

TENCompetence Handbook

on

Integrated Quality Assurance

and

Gender Mainstreaming

Version 2.0

2007

Educational Technology Expertise Centre (OTEC)
Open University of the Netherlands

TENCompetence Handbook on Integrated Quality Assurance
and Gender Mainstreaming

Version 2.0

Internal Deliverable 1.3 (update for DIP-2)

Amendment History

Version	Date	Author/Editor	Description/Comments
0.8	30 June 2006	E. Kluijfhout and P. van Rosmalen	Draft
1.0	10 July 2006	E. Kluijfhout and P. van Rosmalen	DIP-1 version
2.0	1 June 2007	E. Kluijfhout, P. van Rosmalen, S. Maassen	DIP-2 version. Amendments: <ul style="list-style-type: none">• Executive Committee membership and mandate changed• Workshop and conference procedure added• DIP-2 internal review criteria added• Software development standards and procedures added• Risk management simplified

Table of contents

Introduction.....	11
1. Consortium management structure, responsibilities and meetings.....	12
Consortium management structure	12
Detailed responsibilities.....	12
Consortium Board.....	12
Executive Committee.....	14
Project Standards and Quality Assurance Subcommittee.....	16
Technical Standards and Architecture Subcommittee	18
Scientific Subcommittee	19
Valorisation Subcommittee.....	20
TENCompetence Secretariat.....	22
Project Co-ordinator.....	23
Work Package Leader	23
General Manager.....	24
Local Project Co-ordinator	25
Project meetings and workshops.....	25
Attendance closed meetings.....	26
Planning and organising closed project meetings.....	26
Planning and organising open workshops/conferences	26
Proposed article review procedure.....	30
Budgeting and financing project meetings and open workshops.....	32
2. Reporting.....	34
Report types	34
Internal report acceptance procedure	35
Report compilation details	35
Internal Consortium Activity Report - per WP	35
Internal Consortium Costs Report	36
Aggregated Internal Consortium Activity and Costs Report.....	37
Interim Activity Report (6 months)	37
Periodic reports - overview	38
Periodic Activity Report	39
Periodic Management Report	39
Periodic Report on the Distribution of the Community's Contribution	40
Draft Planning for next 18 months (Updating the Implementation Plan).....	41
3. TENCompetence assessment protocol.....	42
TENCompetence output types	42
RTD output types and quality criteria.....	42
Peer-reviewed journal articles.....	43
(Chapters in) scientific books	44
Scientific conference papers	44
Other papers	44
Technology output	45
Awarding RTD output points.....	47
RTD output norms	48
Valorisation impact appraisal	50

Using DSpace for output submission.....	52
4. Internal deliverable quality assurance.....	54
The nature of RTD quality assurance in the Unified Process.....	54
Quality assurance criteria internal deliverables	55
WP1.....	55
WP2.....	56
WP3.....	58
WP4.....	61
WP5.....	64
WP6.....	67
WP7.....	69
WP8.....	73
WP9.....	79
WP10.....	80
Some notes on QA of architecture integration during cycle 1.....	81
5. Deliverable appraisal and submission.....	82
Deliverables overview	82
Deliverable types and internal assessment criteria	83
Internal deliverable acceptance procedure.....	89
Deliverable submission to the Commission.....	89
Hints for writing milestone and deliverable reports	90
Plan adjustments	90
6. External project review procedure.....	91
7. Overall project evaluation and impact assessment methodology	92
The goals of TENCompetence validation.....	92
Effectiveness evaluation of the project outcomes.....	92
The role of the pilots in the three cycles of project activity.....	95
Technical testing and validation	96
Unit and integration testing.....	96
Technical evaluation of the integrated system.....	97
8. Software development standards and procedures.....	98
SourceForge and Bugzilla.....	98
Background.....	98
Bugzilla.....	98
Eclipse integration with Bugzilla provided by Mylar.....	99
References and Links:.....	99
Unit Testing and JUnit.....	100
Background.....	100
Integration with Eclipse	100
JUnit 3 or JUnit 4?	100
JUnit TENCompetence source folder structure and naming conventions.....	101
Automated and Manual Testing frameworks.....	102
Guidance for writing tests.....	102
The Use of a CVS Repository in the TENCompetence Project.....	103
Background.....	103
CVS and SourceForge.....	103
Access rights and user permissions.....	103
Integration with Eclipse	104
Repository details.....	104

Modules and folder structure	104
JUnit folder structure	105
Other considerations when using folders	105
House Rules	106
Component Owners	106
References and Links:	106
Coding conventions and guidelines	107
Naming conventions	107
Conventions for plug-ins	109
User Interface Guidelines	110
Runtime, Platforms, Tools and Libraries	110
Licensing	111
References	111
9. Risk management	113
10. Gender mainstreaming - Gender Project Action Plan	114
1. Introduction	114
2. Gender Actions	115
3. Monitoring Progress and the Reporting Timetable	118

List of Tables and Figures

Tables

Table 1: Example of a conference announcement	27
Table 2: Example of an article review form.	30
Table 3. Report types and submissions intervals	34
Table 4. Overview internal reporting decisions	35
Table 5. Output points per output type	47
Table 6. Output norms per staff type	49
Table 7. Output points for months 1-12.....	49
Table 8. Impact information overview table.....	51
Table 9. Impact information 3-monthly progress reports	51
Table 10. Deliverables overview months 13-30	82
Table 11. Deliverables internal assessment criteria	83
Table 12. Plan adjustment decisions.....	90
Table 13. Seven solutions TENCompetence translated into criteria and indicators.....	93
Table 14. Eclipse tool and libraries used in the project.....	110
Table 15. Third-party tools and libraries used in the project.....	111

Figures

Figure 1. Project Management structure.....	12
Figure 2. Unified Process.....	54
Figure 3. Relation between TENCompetence WP-types.....	81

Introduction

This Handbook contains information that is required for the proper day-to-day running of the TENCompetence project, and to promote quality assurance.

Quality Assurance (QA) in TENCompetence is not a one-time activity, but is fully integrated into the project structure and procedures. More specifically, the following QA aspects are covered in this Handbook:

1. Responsibilities of all those involved in project activities (chapter 1)
2. Regular internal and external monitoring and reporting procedures (chapter 2)
3. The definition of an assessment protocol for scientific, technology and valorisation outputs (chapter 3).
4. Internal Deliverable quality assurance criteria (chapter 4)
5. Internal Deliverable and Deliverable appraisal and submission procedures (chapter 5)
6. External project review procedure (chapter 6)
7. Overall project evaluation and impact assessment methodology (chapter 7)
8. Clearly defined software development standards and procedures (chapter 8)
9. Risk assessment and contingency planning (chapter 9)
10. Gender mainstreaming

This version of the Handbook covers the DIP-2 period (months 13-30), but being an ever-evolving document, updates may become available as the project evolves. The latest version of the Handbook is available electronically at <http://hdl.handle.net/1820/993>

We have decided to make this Handbook a document that can be read on its own, instead of referencing to other project- and Commission documents. This might however give rise to ambiguity. In such case, the order of prevalence is:

1. Contract document, including the DoW (highest level of prevalence)
2. Commission rules, e.g. as contained in manuals
3. Consortium Board decision
4. Consortium Agreement
5. Handbook (lowest level of prevalence)

Questions and suggestions for improvement of the Handbook can be sent to: eric.kluijfhout@ou.nl or peter.vanrosmalen@ou.nl

1. Consortium management structure, responsibilities and meetings

Consortium management structure

The TENCompetence project management structure is as follows:

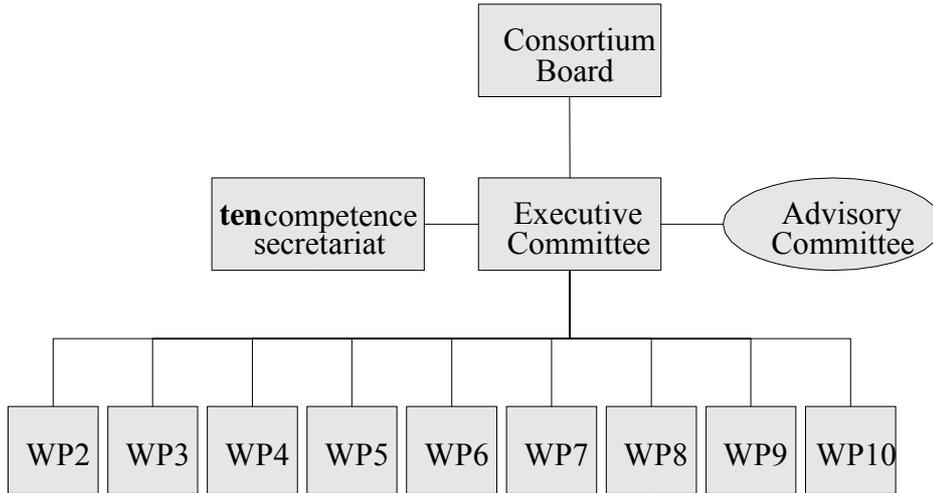


Figure 1. Project Management structure

All major project roles are detailed below.

Detailed responsibilities

Consortium Board
<p>Summary</p> <p>The Consortium Board is the ultimate decision-making body of the Consortium. The representatives to the Consortium Board are of senior management level with the authority to commit their organisation to the decisions of the Consortium Board.</p>
<p>Composition</p> <p>Each Party is entitled to send one voting representative to the Consortium Board. These representatives should be of senior management level with the authority to commit their organisation to the decisions of the Consortium Board. Each partner's representative on the Consortium Board will be responsible for the internal co-ordination of TENCompetence activities in their institution. The Consortium Board is Chaired by the Project Coordinator. The General Manager shall act as the secretary (without voting rights).</p>

Detailed tasks and responsibilities

The Consortium Board is the highest collective decision-making body. It decides on matters relating to:

- The overall project progress, supported by progress reports received from the Executive Committee.
- The preparation and final approval of the annual implementation plan prior to the submission to the Commission.
- All budget-related matters, including approval of the accounts for the past (financial) year, and approval of the budget for the annual implementation plan.
- The structure and restructuring of the work packages.
- Standards including, but not limited to: project standards and quality assurance; the TENCompetence assessment protocol; the technical standards and architecture; the scientific standards; and the valorisation approach.
- The appointment of the members of the Executive Committee.
- The acceptance of new parties as well as the exclusion of Parties.
- The alteration of the Consortium Agreement.
- The resolution of difficulties and disputes.
- The premature completion/ termination of the Project.

The decisions of the Consortium Board in the Project-related matters are legally binding to all Parties.

Meetings

Each Party is entitled to send one voting representative to the Consortium Board. These representatives should be of senior management level with the authority to commit their organisation to the decisions of the Consortium Board. The meetings are Chaired by the Project Coordinator. The General Manager shall act as the secretary (without voting rights).

Ordinary meetings of the Consortium Board shall be convened at least twice a year. Extraordinary meetings of the Consortium Board may be convened either by the Chairperson or at the request of a quarter (25 percent) of the Work Packages Leaders or at the request of one third ($\frac{1}{3}$) of the Parties. Ordinary and extraordinary meetings of the Consortium Board shall constitute a quorum if more than half ($\frac{1}{2}$) of the Parties are present or duly represented by proxy. The Commission may participate as an observer at the meetings of the Consortium Board.

The General Manager shall convene the meeting and provide an agenda to all the Parties not later than fifteen (15) days in advance of the relevant Consortium Board meeting. The agenda must give all relevant background information to any proposed decision. No decision may be made in relation to any matter not mentioned in the agenda unless agreed by unanimous consent of all the Parties present, or duly represented by proxy.

In extraordinary cases the Consortium Board may take decisions through its chairperson consulting with all members via teleconference and/or via e-mail, phone, etc. These decisions must be ratified by an ordinary meeting and shall be made available to all Parties.

Minutes of the meetings shall be submitted to all members without delay. The minutes shall be considered as accepted by the Parties if, within fifteen (15) calendar days from receipt, no Party has objected in a traceable form to the coordinator.

Voting

All decisions of the Consortium Board shall be taken by simple majority. In case of draw the Project Co-ordinator, who is the Chairperson of the Consortium Board, will decide.

A Party may issue its veto only in the case of decisions:

- a) to accept a new party in the Project if a substantial threat to its commercial or strategic interests is likely to exist which cannot be resolved by any other measure,

b) which affects significantly its budget and workload set down in the Description of Work (Contract, Annex I).
A veto in accordance to (a) may be overruled by the Consortium Board by unanimous vote of all the non-objecting Parties as far as these Parties represent more than two third ($\frac{2}{3}$) of all Parties. A veto in accordance to (b) may be overruled by the Consortium Board with votes of three quarters ($\frac{3}{4}$) of the Parties present or duly represented.

Member contact details

Open Universiteit Nederland	Rob Koper (Chair)	rob.koper@ou.nl	+31 45 5762657
Open Universiteit Nederland	Eric Kluijfhout (secretary)	eric.kluijfhout@ou.nl	+31 45 5762921 +31 6 52692093
Software de Base S.A. Altran SDB	Miguel Arjona Villanueva	marjona@altransdb.com	+34 917 440517
LogicaCMG Nederland B.V.	Rob Bakker	r.bakker@logicacmg.com	+31 43 3524200
Fundació Barcelona Media Universitat Pompeu Fabra	Josep Blat	josep.blat@upf.edu	+34 93 5422172
GIUNTI Interactive Labs s.r.l.	Marco Luccini	m.luccini@giuntilabs.com	+39 0185 42123
Centre for Research and Technology Hellas	Demetrios Sampson	sampson@iti.gr	+30 210414.2766
Gottfried Wilhelm Leibniz Universität Hannover	Daniel Olmedilla	olmedilla@l3s.de	+49 511 7629735
Institut Européen d'Administration des Affaires INSEAD CALT	Albert Angehrn	albert.angehrn@insead.edu	+33 1 60724361
Bolton University	Prof. Oleg Liber	o.liber@bolton.ac.uk	+44 1204 903660
University of Amsterdam	Henk Sligte	h.w.sligte@uva.nl	+31 20 5251374
Sofiiski Universitet "Sveta Kliment Ohridski"	Dr. Roumen Nikolov	roumen@fmi.uni-sofia.bg	+359 28161551
Stichting SURF	Tom Dousma	dousma@surf.nl	+31 30 2346600
Synergetics	Luk Vervenne	luk@synergetics.be	+32 476 530021
UNESCO-IHE	Carel W.H. Keuls	c.keuls@unesco-ihe.org	+31 15 2151848
Stad Antwerpen	Karin Nauwelaerts	karin.nauwelaerts@stad.antwerpen.be	+32 33604601

Executive Committee

Summary

The Executive Committee is responsible for the day-to-day management of the project and shall report and be accountable to the Consortium Board. This involves assuming overall responsibility for liaison between the Parties in relation to the Project, for analysing and approving the results, for proper administration of the project and for implementation of the provisions contained in the Consortium Agreement.

<p>Composition</p> <p>The Executive Committee comprises the Chairperson of the Consortium Board who will also be the Chairperson of the Executive Committee, and all the Work Package leaders. The General Manager will act as the secretary for the Executive Committee.</p>
<p>Detailed tasks and responsibilities</p> <p>Under the control of, and in compliance with the decisions of the Consortium Board, the Executive Committee shall co-ordinate the Project. The Executive Committee assumes overall responsibility towards the Consortium Board for liaison between the Parties for analysing and approving the results generated under Work Packages and/or Subcommittees. The Executive Committee shall be specifically responsible for:</p> <ul style="list-style-type: none"> • Taking the necessary decisions in between the meetings of the Consortium Board. • Supporting the Co-ordinator in fulfilling obligations towards the Commission. • Ensuring that all work meets functional requirements. • Providing administrative, financial, legal and logistical support – through the Secretariat - for activities that surpass the level of individual Work Packages. • Providing project management in relation to the activities of the subcommittees on technical, financial and/or exploitation/dissemination issues, as applicable. • Supporting the individual Work Package Leaders in progress monitoring through the administration of Internal Deliverables and deadlines. • Supporting the Board in monitoring progress through the compilation of internal progress reports, based on inputs received from the Local Project Co-ordinators and Work Package Leaders. • Supporting the Board in progress and financial reporting to the Commission through the compilation of annual progress reports on the basis of inputs received from Work Package Leaders, and the Chairs of the Subcommittees and the Board itself. • Supporting the Board in compiling the detailed annual Implementation Plan, on the basis of inputs from the Work Package Leaders, the Local Project Co-ordinators, the Chairs of the Subcommittees and the Board itself, for approval by the Consortium Board prior to its submission to the Commission • Reviewing and proposing to the Consortium Board budget transfers in accordance with the contract and the annual implementation plan. • Proposing changes in work sharing, budget and participants to the Consortium Board. • Making proposals to the Parties (other than a Defaulting Party) to serve notices on a Defaulting Party and to assign the Defaulting Party's tasks to specific entities. <p>The Executive Committee shall have the right to set up Subcommittees to advise and support the Executive Committee in the proper management and co-ordination of the Project. These Subcommittees have an advisory role only. The following Subcommittees will be set up at the start of the project:</p> <ul style="list-style-type: none"> • Project Standards and Quality Assurance Subcommittee • Technical standards and Architecture Subcommittee • Scientific Subcommittee • Valorisation Subcommittee
<p>Meetings</p> <p>The Executive Committee may meet both physically and virtually. It shall meet at least quarterly at the request of its chairperson. Extraordinary meetings may be called at any other time at the request of its chairperson, two of its members or at the request of a quarter (1/4) of the Parties. The Commission may participate as an observer at the meetings of the Executive Committee.</p> <p>Ordinary meetings shall be convened by the chairperson with at least fifteen (15) calendar days prior notice including an agenda. Ordinary and extraordinary meetings of the Executive Committee shall constitute a quorum if at least half of the voting members are present or duly represented by proxy.</p>

<p>Any decision requiring a vote at a Executive Committee meeting must be identified as such on the pre-meeting agenda, unless there is an unanimous agreement to vote on a decision at that meeting and three-quarters ($\frac{3}{4}$) of the members of the Executive Committee are present or duly represented by proxy.</p> <p>Minutes of the meetings shall be transmitted to the members of the Executive Committee without delay. The minutes shall be considered as accepted, if within fifteen (15) calendar days from receipt no member has objected in a traceable form to the Co-ordinator. Subsequent, the minutes of the meetings shall be submitted to all Parties.</p>				
<p>Voting</p> <p>All decisions of the Executive Committee shall be taken by simple majority. In case of draw the Project Co-ordinator, who is the Chairperson of the Executive Committee, will decide.</p>				
<p>Member contact details</p>				
WP1	Open Universiteit Nederland	Rob Koper	rob.koper@ou.nl	+31 45 5762657
WP2	Software de Base S.A. Altran SDB	Miguel Arjona Villanueva	marjona@altransdb.com	+34 917 440517
WP3	LogicaCMG Nederland B.V.	Louis Maessen	louis.maessen@logicacmg.com	+31 43 3524200
WP4	Fundació Barcelona Media Universitat Pompeu Fabra	Josep Blat	josep.blat@upf.edu	+34 93 5422217
WP5	GIUNTI Interactive Labs s.r.l.	Marco Luccini	m.luccini@giuntilabs.com	+39 0185 42123
WP6	Centre for Research and Technology Hellas	Demetrios Sampson	sampson@iti.gr	+30 210414.2766
WP7	Gottfried Wilhelm Leibniz Universität Hannover	Arne Koesling	koesling@l3s.de	+49 511 7629735
WP8	Institut Européen d'Administration des Affaires INSEAD CALT	Albert Angehrn	albert.angehrn@insead.edu	+33 1 60724361
WP9	Open Universiteit Nederland	Marlies Bitter	marlies.bitter@ou.nl	+31 45 5762636
WP10	Bolton University	Christopher Kew	c.kew@bolton.ac.uk	+44 1204 903651

Project Standards and Quality Assurance Subcommittee

Summary

The Project Standards and Quality Assurance subcommittee guides the work package activities and the delivery of internal deliverables and deliverables with quality procedures. The Subcommittee has an advisory role only.

Composition			
Each Party is represented by one member in this subcommittee, which is chaired by a member of the Executive Committee. If appropriate and required, besides the partners, also individual experts may be invited as member.			
Detailed tasks and responsibilities			
<p>The Project Standards and Quality Assurance subcommittee will:</p> <ul style="list-style-type: none"> • Advise the Executive Committee on the quality assurance procedures and criteria to be included in the Project Handbook and Quality Assurance Plan that will be compiled under the coordination of WP1, and will be updated for each project cycle. • Advise the Executive Committee on any required additional quality assurance procedures and criteria during project execution. • Monitor the effectiveness of the implemented quality assurance procedures and criteria in producing and assessing internal deliverables, deliverables and annual implementation plans. <p>The Executive Committee may confer additional tasks and responsibilities to the Subcommittee, while the Subcommittee may propose more detailed tasks and responsibilities for itself to the Executive Committee for approval.</p>			
Meetings			
No meeting rules have been defined beforehand. The Subcommittee may propose such rules to the Executive Committee for approval.			
Voting			
No voting rules have been defined beforehand. The Subcommittee may propose such rules to the Executive Committee for approval.			
Member contact details			
Open Universiteit Nederland	Rob Koper (Chair)	rob.koper@ou.nl	+31 45 5762657
Software de Base S.A. Altran SDB	Miguel Arjona Villanueva	marjona@altransdb.com	+34 917 440517
LogicaCMG Nederland B.V.	Louis Maessen	louis.maessen@logica cmg.com	+31 43 3524200
Fundació Barcelona Media Universitat Pompeu Fabra	Sergio Sayago	sergio.sayago@upf.edu	+34 93 5422711
GIUNTI Interactive Labs s.r.l.	Marco Luccini	m.luccini@giuntilabs.co m	+39 0185 42123
Centre for Research and Technology Hellas	Pythagoras Karampiperis	pythk@iti.gr	+30 2104142766
Gottfried Wilhelm Leibniz Universität Hannover	Daniel Olmedilla	olmedilla@l3s.de	+49 511 7629735
Institut Européen d'Administration des Affaires INSEAD CALT	Alicia Cheak	alicia.Cheak@insead.edu	+33 1 60729169
Bolton University	Prof. Oleg Liber	o.liber@bolton.ac.uk	+44 1204 903660
University of Amsterdam	Judith Schoonenboom	j.i.schoonenboom@uva.n l	+31 20 5251520
Sofiiski Universitet "Sveta Kliment Ohridski"	Mirolyuba Madjarova	mira@fmi.uni-sofia.bg	+359 28161551
UNESCO-IHE	Mr. Biswa Bhattacharia	bha@ihe.nl	+31 15 2151898
Stad Antwerpen			

Technical Standards and Architecture Subcommittee

Summary

The Technology Standards and Architectures Subcommittee advises the Executive Committee on technological standards and architectural constraints for the software to be developed in the project. The Subcommittee has an advisory role only.

Composition

The subcommittee consists of one member of each partner, chaired by a member of the Executive Committee. If appropriate and required, besides the partners, also individual experts may be invited as member.

Detailed tasks and responsibilities

Each software component developed in the project should meet the standards and constraints agreed upon by the Executive Committee. All standards will be open standards or specifications. The Technology Standards and Architectures subcommittee will:

- Advise the Executive Committee on technological standards and architectural constraints for the software to be developed in the project under WP3, WP5, WP6, WP7 and WP8.
- Review software deliverables (from WP3, WP5, WP6, WP7 and WP8) in view of these standards and constraints as part of quality assurance, and advise the Executive Committee accordingly.
- Review annual implementation plans (especially for WP5, WP6, WP7 and WP8) in view of these standards and constraints as part of quality assurance, and advise the Executive Committee accordingly.

The Executive Committee may confer additional tasks and responsibilities to the Subcommittee, while the Subcommittee may propose more detailed tasks and responsibilities for itself to the Executive Committee for approval.

Meetings

No meeting rules have been defined beforehand. The Subcommittee may propose such rules to the Executive Committee for approval.

Voting

No voting rules have been defined beforehand. The Subcommittee may propose such rules to the Executive Committee for approval.

Member contact details

Open Universiteit Nederland	Marcus Specht	marcus.specht@ou.nl	+31 45 5762806
Software de Base S.A. Altran SDB	Miguel Arjona Villanueva	marjona@altransdb.com	+34 917 440517
LogicaCMG Nederland B.V.	Ruud Lemmers	ruud.lemmers@ logicacmg.com	+31 43 3524200
Fundació Barcelona Media Universitat Pompeu Fabra	Toni Navarrete	toni.navarrete@upf.edu	+34 93 5422711
GIUNTI Interactive Labs s.r.l.	Michele Dicerto	m.dicerto@giuntilabs.it	+39 01 02465178
Centre for Research and Technology Hellas	Pythagoras Karampiperis	pythk@iti.gr	+30 210414.2766
Gottfried Wilhelm Leibniz Universität Hannover	Daniel Olmedilla	olmedilla@l3s.de	+49 511 7629735

Institut Européen d'Administration des Affaires INSEAD CALT	Thierry Nabeth	thierry.nabeth@insead.edu	+33 1 60724312
Bolton University	Scott Wilson	scott.bradley.wilson@gmail.com	+44 1248 383645
University of Amsterdam	Judith Schoonenboom	j.i.schoonenboom@uva.nl	+31 20 5251520
Sofiiski Universitet "Sveta Kliment Ohridski"	Sylvia Ilieva	sylvia@acad.bg	+359 28161551
UNESCO-IHE Stad Antwerpen	Mr. Wim Glas	w.glas@unesco-ihe.org	+31 15 2151818

Scientific Subcommittee

Summary

The Scientific Subcommittee advises the Executive Committee on the scientific criteria for project activities and outcomes. The Subcommittee has an advisory role only.

Composition

The Scientific Subcommittee comprises a maximum of seven members from the partners, including the Chair who is the Project Co-ordinator. Members are selected on the basis of their high reputation in one or more fields relevant to the project. If appropriate and required, besides the partners, also individual experts may be invited as member.

Detailed tasks and responsibilities

The Scientific Subcommittee will:

- Set standards for scientific publications and technology output.
- Review project deliverables as measured against scientific criteria
- Review annual implementation plans in view of the scientific standards as part of quality assurance, and advise the Executive Committee accordingly.

The Executive Committee may confer additional tasks and responsibilities to the Subcommittee, while the Subcommittee may propose more detailed tasks and responsibilities for itself to the Executive Committee for approval.

Meetings

No meeting rules have been defined beforehand. The Subcommittee may propose such rules to the Executive Committee for approval.

Voting

No voting rules have been defined beforehand. The Subcommittee may propose such rules to the Executive Committee for approval.

Member contact details

Open Universiteit Nederland	Rob Koper (Chair)	rob.koper@ou.nl	+31 45 5762657
Fundació Barcelona Media Universitat Pompeu Fabra	Josep Blat	josep.blat@upf.edu	+34 93 5422217

Centre for Research and Technology Hellas	Demetrios Sampson	sampson@iti.gr	+30 210414.2766
Gottfried Wilhelm Leibniz Universität Hannover	Wolfgang Nejdil	nejdil@kbs.uni-hannover.de	+49 511 7629735
Institut Européen d'Administration des Affaires INSEAD CALT	Albert Angehrn	albert.angehrn@insead.edu	+33 1 60724361
Bolton University	Oleg Liber	o.liber@bolton.ac.uk	+44 1204 903660
Sofiiski Universitet "Sveta Kliment Ohridski"	Roumen Nikolov	roumen@fmi.uni-sofia.bg	+359 2 8161 551
Athabasca university, CA	Terry Anderson	terrya@athabascau.ca	
QUEM Management of Qualification development, Berlin, Germany	John Erpenbeck	john.erpenbeck@gmx.de	
University of Wollongong, Australia	Barry Harper	bharper@uow.edu.au	
International School of New Media, Lübeck, Germany	Joachim Hasebrook	joachim.hasebrook@isnm.de	
IEEE, NZ	Kinshuk	kinshuk@ieee.org	
University of Leeds, UK	Roger Hartley	j.r.hartley@education.leeds.ac.uk	
Utah State University, USA	David Merrill	mdavid.merrill@gmail.com	
Simon Fraser University, Burnaby, CA	Griff Richards	griff@sfu.ca	
Cranfield University, Swindon, UK	Bernhard Scott	b.c.e.scott@cranfield.ac.uk	
Florida State University, USA	Mike Spector	mspector@lsi.fsu.edu	
Utah State University, USA	David Wiley	david.wiley@gmail.com	
University of Sydney, Australia	Peter Goodyear	p.goodyear@edfac.usyd.edu.au	
LICEF, Montréal, CA	Gilbert Pacquette	gpaquette@licef.teluq.uquebec.ca	
UNESCO-IHE			
Stad Antwerpen			

Valorisation Subcommittee

Summary

The Valorisation Subcommittee advises the Executive Committee on the planning, execution and monitoring of valorisation activities in WP9 and WP10. The Subcommittee has an advisory role only.

Composition

The members of the Subcommittee are representing Partners or Associate Partners. Each partner is represented by one member in this Subcommittee, which is chaired by a member of the Executive Committee. The Subcommittee has an advisory role only.

Detailed tasks and responsibilities

The Valorisation Subcommittee will:

- Advise on the TENCompetence impact criteria, valorisation strategies and underlying business

<p>models.</p> <ul style="list-style-type: none"> • Advise on the relevance of the research and development activities to meet the public and private sector needs (e.g. towards WP2). • Advise on structuring relations with the wider professional community in the various sectors of the world of lifelong competence development (schools, universities, training companies, libraries, different user organisations, etc.; WP9 and WP10) • Review annual implementation plans in view of the impact criteria and valorisation strategies as part of quality assurance (especially for WP9 and WP10), and advise the Executive Committee accordingly. • Review project outcomes and deliverables of WP9 and WP10 as measured against the impact criteria and valorisation strategies <p>The Executive Committee may confer additional tasks and responsibilities to the Subcommittee, while the Subcommittee may propose more detailed tasks and responsibilities for itself to the Executive Committee for approval.</p>			
<p>Meetings</p> <p>No meeting rules have been defined beforehand. The Subcommittee may propose such rules to the Executive Committee for approval.</p>			
<p>Voting</p> <p>No voting rules have been defined beforehand. The Subcommittee may propose such rules to the Executive Committee for approval.</p>			
<p>Member contact details</p>			
Open Universiteit Nederland	Eric Kluijfhout	eric.kluijfhout@ou.nl	+31 45 5762921 +31 6 52692093
Software de Base S.A. Altran SDB	Miguel Arjona Villanueva	marjona@altransdb.com	+34 917 440517
LogicaCMG Nederland B.V.	Wouter-Jan Smeulers	wouter.jan.smeulers@logicacmg.com	+31 70 3756000
Fundació Barcelona Media Universitat Pompeu Fabra	Josep Blat	josep.blat@upf.edu	+34 93 5422217
GIUNTI Interactive Labs s.r.l.	Marco Luccini	m.luccini@giuntilabs.com	+39 0185 42123
Centre for Research and Technology Hellas	Demetrios Sampson	sampson@iti.gr	+30 210414.2766
Institut Européen d'Administration des Affaires INSEAD CALT	Albert Angehrn	albert.angehrn@insead.edu	+33 1 60724361
Bolton University	Prof. Oleg Liber	o.liber@bolton.ac.uk	+44 1204 903660
University of Amsterdam	Henk Sligte	h.w.sligte@uva.nl	+31 20- 5251374
Sofiiski Universitet "Sveta Kliment Ohridski"	Dr. Roumen Nikolov	roumen@fmi.uni-sofia.bg	+359 28161551
UNESCO-IHE	Mr. Andreja Jonoski	a.jonoski@unesco-ihe.org	+31 15 2151813
Stad Antwerpen			

TENCompetence Secretariat

Summary

The Secretariat provides administrative, financial, legal and logistical support to the Consortium Board; to the Executive Committee; to the Project Co-ordinator in his role as intermediary to the Commission with communication and reporting activities; to the Advisory Committees; and to the Work Package Leaders where activities surpass that of individual Work Packages. The Secretariat carries out its activities as part of WPI.

Composition

The Secretariat comprises a General Manager, a Project Officer for interfacing with the Work Package Leaders, and a Financial Administrator. The General Manager assists the Project Co-ordinator in all of his tasks and is the main day-to-day contact person for the European Commission for administrative, financial, legal and logistical issues that are standard procedures. For strategic issues the Project Co-ordinator will remain the contact person for the commission.

Detailed tasks and responsibilities

- Compile a Project Handbook, detailing: project structure, policies and procedures for reporting and submitting deliverables; quality assurance standards and procedures; and a gender mainstreaming policy. The first version of the Handbook will be compiled within six months from the start of the project, and updates will be produced for each project cycle.
- Liaise with the Local Project Co-ordinators to receive periodic (activity and management) and final reports.
- Monitor Work Package performance based on the received periodic progress and financial reports, in support of the Executive Committee.
- Assist the Project Co-ordinator to manage risks and take corrective actions as necessary.
- Coordinate drafting the Annual Implementation Plans and related budgets for approval by the Project Co-ordinator and subsequently the Executive Committee, and final ratification by the Consortium Board
- Monitor the project's gender mainstreaming policy.
- Monitor the project's IPR policy (open source licences, creative common licences, distribution of deliverables and software).
- Monitor project contacts between TENCompetence and external organisations (industrial, trade, governmental, or institutional).

Member contact details

Open Universiteit Nederland	Eric Kluijfhout	eric.kluijfhout@ou.nl	+31 45 5762921 +31 6 52692093
Open Universiteit Nederland	Jos van den Broek	jos.vandenbroek@ou.nl	+31 45 5762501
Open Universiteit Nederland	Peter van Rosmalen	peter.vanrosmalen@ou.nl	+31 45 5762889
Open Universiteit Nederland	Marlies Timmermans	marlies.timmermans@ou.nl	+31 45 5762183
Open Universiteit Nederland	Ruben Obsteter	ruben.obsteter@ou.nl	+31 45 5762971
Open Universiteit Nederland	Mieke Haemers	mieke.haemers@ou.nl	+31 45 5762624
Open Universiteit Nederland	Sabine Maassen	sabine.maassen@ou.nl	+31 45 5762605

Project Co-ordinator

Summary

The Project Co-ordinator, as the intermediary to the Commission, is responsible for project management and will lead the scientific research and development in TENCompetence. The Project Co-ordinator will chair the Consortium Board, the Executive Committee and the Scientific Subcommittee. The TENCompetence Secretariat supports the Project Co-ordinator in the day-to-day co-ordination, and in all administrative tasks.

Detailed tasks and responsibilities

The Project Co-ordinator is responsible for the following:

- Overall management of the Project with the support from the Secretariat,
- Chairing the Executive Committee, the Consortium Board, and the Scientific Subcommittee
- Preparing meetings and decisions of the Consortium Board and the Executive Committee.
- Collecting and preparing, with support from the Executive Committee, statements, including financial audit certificates, from the partners for transmission to the Commission.
- Ensure prompt delivery of all hardware, software and data identified as deliverable items in the Contract or requested by the Commission for reviews and audits, including the results of the financial audits prepared by independent auditors.

In addition, the Co-ordinator is authorised:

- In case there exist serious concerns regarding the financial soundness of a partner, to require a letter of comfort to prove that the partner is able to fulfil the financial obligations with regard to the Contract and this Consortium Agreement. Until this is provided, the Co-ordinator is entitled to refuse the disbursement of the financial contributions of the Commission to this Party.
- To retain any payment if a partner is late in submitting or refuses to provide deliverables as defined in the Description of Work and/or Consortium Agreement.

Contact details

Open Universiteit Nederland	Rob Koper	rob.koper@ou.nl	+31 45 5762657
-----------------------------	-----------	-----------------	----------------

Work Package Leader

Summary

The Work Package Leader is responsible for the co-ordination of activities and the production of deliverables of the Work Package. The Work Package Leader cooperates with the Executive Committee to facilitate day-by-day project management on Work Package level.

Detailed tasks and responsibilities

The Work Package Leader is responsible for:

- Compiling a detailed work plan for each project cycle.
- Coordinating the activities of the partners participating in the Work Package.
- Monitor progress and securing quality of internal deliverables and deliverables
- Keep participating partners informed on Work Package progress.
- Keep the Coordinator, the Consortium Board and the Executive Committee informed on the current status of activities, internal deliverables and deliverables through the Activity Reports and Internal Cost Report.
- Contribute to the compilation of the Annual Implementation Plans for the Work Package.

Contact details				
WP1	Open Universiteit Nederland	Rob Koper	rob.koper@ou.nl	+31 45 5762657
WP2	Software de Base S.A. Altran SDB	Miguel Arjona Villanueva	marjona@altransdb.com	+34 917 440517
WP3	LogicaCMG Nederland B.V.	Ruud Lemmers	ruud.lemmers@logicacmg.com	+31 43 3524200
WP4	Fundació Barcelona Media Universitat Pompeu Fabra	Josep Blat	josep.blat@upf.edu	+34 93 5422217
WP5	GIUNTI Interactive Labs s.r.l.	Marco Luccini	m.luccini@giuntilabs.com	+39 0185 42123
WP6	Bolton University	Dai Griffiths	dai.griffiths.1@gmail.com	+ 34 687 955912
WP7	Gottfried Wilhelm Leibniz Universität Hannover	Arne Koesling	koesling@l3s.de	+49 511 7629735
WP8	Institut Européen d'Administration des Affaires INSEAD CALT	Albert Angehrn	albert.angehrn@insead.edu	+33 1 60724361
WP9	Open Universiteit Nederland	Marlies Bitter	marlies.bitter@ou.nl	+31 45 5762636
WP10	Bolton University	Christopher Kew	c.kew@bolton.ac.uk	+44 1204 903651

General Manager

Summary

The General Manager heads the TENCompetence Secretariat, and serves as Secretary to the Consortium Board and the Executive Committee. The General Manager can represent the Project Coordinator as chair of the Executive Committee.

Detailed tasks and responsibilities:

- Head the Secretariat
- Serve as Secretary to the Consortium Board
- Serve as Secretary to the Executive Committee

Contact details

Open Universiteit Nederland	Eric Kluijfhout	eric.kluijfhout@ou.nl	+31 45 5762921 +31 6 52692093
Open Universiteit Nederland	Jos van de Broek	jos.vandenbroek@ou.nl	+31 45 5762501
Open Universiteit Nederland	Peter van Rosmalen	peter.vanrosmalen@ou.nl	+31 45 5762889

Local Project Co-ordinator			
Summary			
Each partner will appoint a Local Co-ordinator. The Local Project co-ordinator is responsible for the administrative and financial issues. The Local Project co-ordinator cooperates with the Executive Committee to facilitate day-by-day project management on partner level.			
Detailed tasks and responsibilities			
<ul style="list-style-type: none"> • Be the main contact point for all administrative and financial issues for the partner. • Collect and send all information required for the activity and financial reports to the Secretariat. 			
Contact details			
Open Universiteit Nederland	Rob Koper	rob.koper@ou.nl	+31 45 5762657
Software de Base S.A. Altran SDB	Miguel Arjona Villanueva	marjona@altransdb.com	+34 917 440517
LogicaCMG Nederland B.V.	Louis Maessen	louis.maessen@logicacmg.com	+31 43 3524200
Fundació Barcelona Media Universitat Pompeu Fabra	Josep Blat	josep.blat@upf.edu	+34 93 5422217
GIUNTI Interactive Labs s.r.l.	Marco Luccini	m.luccini@giuntilabs.com	+39 0185 42123
Centre for Research and Technology Hellas	Demetrios Sampson	sampson@iti.gr	+30 210414.2766
Gottfried Wilhelm Leibniz Universität Hannover	Daniel Olmedilla	olmedilla@l3s.de	+49 511 7629735
Institut Européen d'Administration des Affaires INSEAD CALT	Albert Angehrn	albert.angehrn@insead.edu	+33 1 60724361
Bolton University	Oleg Liber	o.liber@bolton.ac.uk	+44 1204 903660
University of Amsterdam	Judith Schoonenboom	j.i.schoonenboom@uva.nl	+31 20 5251520
Sofiiski Universitet "Sveta Kliment Ohridski"	Krassen Stevanov	krassen@fmi.uni-sofia.bg	+359 28161551
Stichting SURF	Jocelyn Manderveld	manderveld@surf.nl	+31 30 2346600
Synergetics	Luk Vervenne	luk@synergetics.be	+32 476 530021
UNESCO-IHE	Carel W.H. Keuls	c.keuls@unesco-ihe.org	+31 15 2151848
Stad Antwerpen	Karin Nauwelaerts	karin.nauwelaerts@stad.antwerpen.be	+32 33604601

Project meetings and workshops

In TENCompetence we try to organise a project meeting and open workshop each quarter, to be organised by one of the partners. The format usually is to have a 3-day closed project meeting, and a 2 day open workshop.

The purpose of the closed project meeting is to:

- Meet with the Board or the Executive Committee, depending on the issues at hand

- Allow WP-partners to meet face to face in working sessions
- Coordinate activities *between* WPs
- Monitor and assess overall progress through formal and informal discussions between the Project Coordinator and the WP-leaders

The purpose of the open workshop is to:

- Generate research output by inviting researchers from within and outside the project to meet and present scientific papers
- Generate dissemination outcomes by organising activities with and for (prospective) associate partners and general stakeholders

Attendance closed meetings

All major partners should attend the closed meetings with at least three staff members. All WP-leaders should be present, and are supposed to have prepared their meetings. The smaller partners may attend with less people.

Members of the Board and the Executive Committee can be duly represented by proxy, provided the representative has the same mandate regarding decisions that are taken during the meeting as the person he/she is representing.

Planning and organising closed project meetings

The agenda for the closed meeting is drawn up by the Secretariat, and sent around as a draft to the WP-leaders for comments, and to the hosting partner for logistical preparations. The logistics are arranged by the hosting partner.

Planning and organising open workshops/conferences

The hosting partner is responsible for organising the open workshop, including organising a call for papers, their collection, assessment and selection, and subsequent publication of the proceedings and - preferably - a special issue of one of the endorsed journals (see chapter 3 for the list of endorsed journals).

Suggested timing for organising a conference is as follows:

Months to go till workshop	Activity
6	Draft Call for Papers, and check with the Secretariat (example below)
6	Arrange for journal special issue publication: see chapter 3 for acceptable journals
5	Decide on preferred meeting and workshop venue
5	Publish call for papers and details on acceptance policy, registration policy, venue and fee (see below for an example): put all info on a dedicated conference website and distribute info through contacts
5	Compile a Programme Committee for paper review: should be of sufficient scientific calibre.

5-3	Collect and distribute papers to reviewers. Proposed review procedure and form below.
	Confirm logistics: conference venue, # of rooms, internet access, catering
3	Confirm to potential participants: conference venue, hotels info, travel suggestions, put also on conference website
2	Inform authors whether their papers have been accepted for the conference

Table 1: Example of a conference announcement

<p><i>TENCompetence Open Workshop on Current research on IMS Learning Design and Lifelong Competence Development Infrastructures</i></p>
<p>Location:</p> <p>Venue:</p> <p>Start: <date>, <time></p> <p>End: <date>, <time></p>
<p>Introduction</p> <p>This workshop is organized by the EU 6th Framework Integrated Project TENCompetence. The objective of the workshop is to identify and analyse current research and technologies in the fields that provide the building blocks for the development of an open source infrastructure that contains all the services needed to support individuals, teams and organisations to (further) develop their competences, using all the distributed knowledge resources, learning activities, units of learning and learning routes/programmes that are available online.</p>
<p>Theme</p> <p>The main theme of this workshop is to provide an overview on current research on IMS Learning in relation to competence development. It includes, but is not limited to:</p> <ul style="list-style-type: none"> • Development of LD tools and architectures, eg, editors, content management systems, runtime engines. • Advanced applications of LD (eg, in gaming, collaborative learning, competence development, ePortfolios) • Research and technology development to support users in creating an adequate learning design (e.g. software agents or design aids) • Research to evaluate the ability of the LD specification to integrate the following criteria: completeness, pedagogical flexibility, personalization, re usability, formalization, etc. • Evaluations of the use of LD in practice, including the integration of a variety of specifications in a learning environment • Studies defining the current state-of-the-art in learning design, or positioning LD in the wider context of learning, training and instruction • Learning Design for formal and informal Lifelong Learning and Competence Development
<p>Other topics</p> <p>Other topics relevant for the workshop include, but are not limited to:</p> <ul style="list-style-type: none"> • Knowledge Management issues related to competence development, lifelong learning or HRM • Competence description frameworks for professions

- Monitoring (changes in) competences in professions * Competence assessment and learner positioning
- Methods and tools to support learners in distributed environments, peer tutoring, teaching, mentoring, coaching
- Personal data, profiles, and portable ePortfolios for competence registration
- Technologies for the discovery of suitable learning resources
- Collaborative filtering technologies, recommend or systems for learning
- Use of language technologies to support learning processes
- Navigational support to users in a learning network
- Authoring, discovery, advise and/or exchange of knowledge resources, learning activities or units of learning
- Open standards and specifications usable for learning networks
- Social software usable for lifelong competence development
- Dynamic discovery of, and collaboration between, informal learning groups
- New social theories and practices which have implications for Competence Development
- Models and tools supporting work-based learning
- Personalised and collaborative trails using knowledge resources or learning activities
- Mobile support for lifelong competence development
- Semantic Web and lifelong competence development
- Gender issues related to the design of educational applications

Workshop Language

English

Publication of Proceedings

All papers accepted for the workshop will be published in the workshop proceedings with ISBN and will be distributed to the participants after the workshop. Selected papers will be invited to submit a full paper (..... words) for a special issue of the devoted to the outcomes of the workshop.

Workshop Chair

Rob Koper (Netherlands) - Chair

Josep Blat (Spain) – Co-chair

Programme Committee

<a selection of senior staff from the organising partner and TENCompetence>

Barry Harper, Australia

Joachim Hasebrook, Germany

Roger Hartley, UK

Wolfgang Nejd, Germany

Roumen Nikolov, Bulgaria

Mike Spector, USA

Peter Goodyear, Australia

Terry Anderson, Canada

Albert Angehrn, France

John Erpenbeck, Germany

Kinshuk, New Zealand

Oleg Liber, UK

David Merrill, USA

Gilbert Pacquette, Canada

Griff Richards, Canada

Bernard Scott, UK

Demetrios Sampson, Greece

David Wiley, USA

Local organising Committee contact details

Address:

name 1 : (email)

name 2 : (email)

Tel: +()

Fax: +()

SUBMISSION GUIDELINES

Deadline for paper submissions

During the workshop it will be possible to present a paper of 2000 words. These submissions may include Research, Technology Development, Review and Application but are not limited to these only. Furthermore, participants can submit a proposal for a demonstration session to demonstrate relevant software.

Electronic submissions should be forwarded as a Portable Document Format (.pdf) attachment to: All submissions must be papers of 2000 words that are formatted according to the publication guidelines (note: for an example see the author instructions of the Sofia call at http://ld06.uni-sofia.bg/CFP_Instructions%20for%20Authors.pdf).

All authors will be informed of the results from the reviews of their papers, including if the papers were accepted for the publication in the workshop proceedings latest

Registration

Registration form may be filled in and sent as an email attachment to: @.....

Deadline for the registration:

Fees

..... (note: this is obviously up to you to decide. The standard policy is that costs for materials, lunches etc. are paid through a small conference fee).

Location

The main conference venue will be at

Accommodation

.....

Proposed article review procedure

Each paper is to be reviewed by at least two reviewers (A and B) from the programme committee. The reviewers review the paper independently, using the form below to report back to the chair. The possible review recommendations are:

- rr1: Definitely accept
- rr2: Minor revisions required before accept
- rr3: Major revisions required before accept
- rr4: Definitely reject

The table below lists all possible outcomes and the action to be taken. In case of large differences between the two reviewers, a third reviewer will need to provide the deciding recommendation. *Reviewing such cases thus takes extra time, which should be planned for!*

Reviewer A	rr1	rr2	rr3	rr4
Reviewer B				
rr1	Accept	Accept	Third reviewer decides	Third reviewer decides
rr2	Accept	Accept	Third reviewer decides	Third reviewer decides
rr3	Third reviewer decides	Third reviewer decides	Reject	Reject
rr4	Third reviewer decides	Third reviewer decides	Reject	Reject

Table 2: Example of an article review form.

	Building the European Network For Lifelong Competence Development TENCompetence IST-2005-027087
	Paper Review Form
Reviewer Name:	<input style="width: 100%; height: 20px;" type="text"/>

Affiliation, Country	
Title of the paper:	
<p>TENCompetence Project Coordination at: Open University of the Netherlands Valkenburgerweg 177, 6419 AT Heerlen, The Netherlands Tel: +31 45 5762624– Fax: +31 45 5762800</p>	
<i>Please rate the following 4 points</i>	
1. Relevance of the paper with regard to the Call for Papers [1 (very low) - 6 (very high)]:	
<i>Please explain your rating:</i>	
2. Technical quality of the paper [1 (very low) - 6 (very high)]:	
(E.g.: Is the paper technically sound? Does it carefully evaluate the strengths and limitations of its contributions? Are its claims backed up? Does the paper offer a new form of evidence in support of or against a well-known technique? Does it offer a theoretical analysis of prior experimental results? Related Work?)	
<i>Please explain your rating:</i>	
3. Presentation quality of the paper [1 (very low) - 6 (very high)]:	
(E.g.: Is the paper well written? Does it motivate the research? Are the results described and evaluated? Is the paper organized in a logical fashion? Is the paper written in a manner that makes it accessible to most educators and/or educational system developers? Is the paper written in clear English? Is the readability good, average or poor? Are there any presentation problems?)	
<i>Please explain your rating:</i>	
4. How Original is this paper? [1 (very low) - 6 (very high)]:	
(E.g.: Are the problems and approaches new? Is this a novel combination of existing techniques? Does the paper points out differences from related research? Does it address a new problem or one that has not been studies in depth? Does it introduce an interesting research paradigm? Does it introduce an idea that appears promising or might stimulate others to develop promising alternatives?)	
<i>Please explain your rating:</i>	

Please Summarize the Strength & Weaknesses of the paper:
My overall recommendation is

- Definitely accept
- Minor revisions required before accept
- Major revisions required before accept
- Definitely reject

Recommended minor revisions? (if any)

You can use "track changes" and add your notes directly into the paper. Please indicate if you do so.

Recommended major Revisions? (if any)

(In case of major revisions, the revised paper will be reviewed again and authors will be asked to provide a short summary of the performed revisions along with the revised paper.)

How confident are you in your appropriateness as a referee for this paper?

- Very confident - I am an expert in this area.
- Confident - I have a reasonable knowledge of this area.
- Fairly confident - I have some knowledge of this area.
- Not confident - I have no significant knowledge of this area.

Budgeting and financing project meetings and open workshops

In addition to budgets for RTD, Demonstration, Training and Consortium Management activities, an overall budget was calculated for other direct costs like travel, meetings, workshops, hardware and software (see section 9.5.2 of the DoW). From the advance payments of this budget, 3% is put into a 'Common Reserve' as described in Section 6.3 of the Consortium Agreement. Project meeting and workshop costs can be claimed against this Common Reserve. The following rules apply:

1. Claim against valid activity type:

Each organized event should be linked to an activity type against which costs can be claimed, i.e. Management, RTD, Training, or Demonstration. The choice for the activity type has to be justified, as it defines which part of the costs can ultimately be recovered through the EC, depending on the funding regime of the concerning partner.

2. Eligible costs under the Common Reserve regime:

Only part of the event costs can be reimbursed out of the Common reserve, i.e. costs for conference rooms & facilities, coffee/tea, shuttles.

3. Non-eligible costs under the Common Reserve regime:

The following costs have to be covered within the budget of the organizing partner or by the attendees, i.e. lunches, dinners, proceedings etc. The standard policy is that for

open events, non-partner attendees will give a fair contribution to cover their share of the costs of the event.

The organizing partner submits the proposed budget for the event to the TENCompetence Secretariat for approval. At the end of each 12 month period (in relation to the cost statements) any partner who wishes to claim approved costs from the Common Reserve, should send a cost statement to the co-ordinator.

If the total amount of the eligible costs claimed is less than the total of the Common Reserve, all claimed costs will be added to the budget of the parties concerned. If the total amount of the eligible claimed costs is greater than the total of the Common Reserve, a proportional part (the relation between the claimed costs and the Common Reserve) of the claimed costs will be added to the budget of the parties concerned.

2. Reporting

Reporting requirements are based on Commission regulation as contained in the Contract, but also have project-specific aspects. This chapter describes the various report types, details by whom and how to compile them, and the project-internal acceptance procedure.

Report types

Table 3. Report types and submissions intervals

<i>Type</i> ¹	<i>Description</i>	<i>Internal</i>
Internal Consortium Reports	Used for internal monitoring only, and are not submitted to the Commission. Internal Consortium Reports to be submitted to the Secretariat: <ul style="list-style-type: none"> • Internal Consortium Activity Report <ul style="list-style-type: none"> ○ Internal Consortium Activity Report by Work Package Leader ○ Internal Consortium Activity Report by Contractor • Internal Consortium Costs Report 	every 3 months
Interim Activity Reports	To be submitted by the Secretariat to the Commission, if specified in the contract. In TENCompetence we have a verbal agreement with the Project Officer to submit a compilation of the data that are needed for the Periodic Reports at month 6.	month 18, 30, 42
Periodic Reports, including on-line questionnaires	To be submitted to the Commission by the Secretariat at the end of each reporting period (12 months). Periodic Reports to be submitted to the Commission: <ul style="list-style-type: none"> • Periodic Activity Report • Periodic Management Report • Periodic Report on the Distribution of the Community's Contribution • Draft planning for the next 18 months (18-month Implementation Plan) On-line questionnaires to be submitted to the Commission: <ul style="list-style-type: none"> • Interim Science and Society Reporting Questionnaire • Interim Reporting on the Implementation of the Gender Action Plan • Interim Socio-Economic Reporting Questionnaire 	month 24, 36
Final Reports	To be submitted by the Secretariat to the Commission at the end of the project in addition to the periodic reports for the last reporting period. In this project cycle these are not yet relevant.	month 48

¹ For further details see 'Project Reporting in FP6', <http://cordis.europa.eu.int/fp6/find-doc-management.htm#reporting>

Internal report acceptance procedure

The following table lists the decision-making responsibilities regarding the various reports.

Table 4. Overview internal reporting decisions

Product and decision type:	Responsible body:	CB	EC	PC	WPL	AC	PS
Internal Consortium Report endorsement							
Internal Consortium Activity Reports - per WP				D	P		A
Internal Consortium Activity Reports - aggregated	D	P					A
Internal Consortium Costs Reports - per WP				D	P		A
Internal Consortium Costs Reports - aggregated	D	P					A
Interim Activity Report							
			D				P
Periodic Report endorsement							
Periodic Activity Report			D				P
Periodic Management Report			D				P
Periodic Report on the Distribution of the Community's Contribution			D				P
Draft planning for next 18 months:							
• WP 18-month Implementation Plan		D			P		A
• WP 18-month budget				D	P		A
• Overall 18-month Implementation Plan	D	P					A
• Overall 18-month budget	D	P					A

Responsible bodies:

CB = Consortium Board

EC = Executive Committee

PC = Project Co-ordinator

WPL = Work Package Leader

AC = Advisory Committee

PS = Project Secretariat

Decision types:

P = propose

A = advise

D = decide (approve/reject)

Report compilation details

Report details and their compilation procedures are detailed below.

Internal Consortium Activity Report - per WP
<p>Summary</p> <p>The report is compiled every three months for internal progress monitoring at Work Package and overall project level. The information on individual Work Packages is submitted by WP-leaders to the Secretariat, and on partner progress by the Local Project Co-ordinators.</p>

<p>Collected data</p> <p>The data are collected at the level of an individual Work Package, and pertain to: a) started and finished tasks; b) general progress (overall status, results, risks, etc.); c) plans for the next period; d) management issues; and e) progress at Task level.</p>
<p>Compiled report</p> <p>The collected data are compiled into an aggregated version for presentation to the Executive Committee and Consortium Board.</p>
<p>Procedure</p> <ol style="list-style-type: none"> i. The TENCompetence Secretariat sends out the data collection spreadsheet (the latest version will be distributed by email, for an example see the forum in “News and Information for Partners” at http://www.tencompetence.org/course/view.php?id=28) to the Local Project Co-ordinator before the end of the three-month project execution period. ii. The Local Project Co-ordinator collects the required information from the WP Leader, and returns the completed spreadsheet to the Secretariat not later than 1 week after closure of the three-month period. iii. Once the Secretariat has received the data on all Work Packages, it compiles the ‘Aggregated Internal Consortium Activity and Cost Report’ (see below)

Internal Consortium Costs Report

<p>Summary</p> <p>The report is compiled every three months for internal financial monitoring at Work Package, partner and overall project level (by activity types) by the partners, and submitted to the Secretariat.</p>
<p>Collected data</p> <p>The financial data are collected per partner (as the budget holder) and sent to the Secretariat for processing. Cost types - personnel (at the level of individual staff members), equipment, travel & subsistence, other direct costs, subcontracting, and indirect costs (FC only) - are charged against activity type (RTD, training, demonstration).</p>
<p>Compiled report</p> <p>The collected data are compiled into an aggregated version for presentation to the Executive Committee and Consortium Board.</p>
<p>Procedure</p> <ol style="list-style-type: none"> i. At the start of the 12-month reporting period the Secretariat provides each Local Project Co-ordinator with a spreadsheet in which the costs have to be entered on a 3-monthly basis, and which are then automatically accumulated (See http://www.tencompetence.org/mod/forum/discuss.php?d=350 for the current version 06-02-2006). ii. The completed spreadsheet should be received by the Secretariat no later than one week after the end of the three-monthly reporting period. iv. Once the Secretariat has received all Cost Reports it compiles the ‘Aggregated Internal Consortium Activity and Cost Report’ (see below)

Aggregated Internal Consortium Activity and Costs Report

Summary

The report is compiled every three months by the Secretariat, based on the received Internal Consortium Activity Reports and the Internal Consortium Costs Reports.

Collected data

The data are collected through the Internal Consortium Activity Reports and the Internal Consortium Costs Reports.

Compiled report

The collected data are compiled into an aggregated version for presentation to the Executive Committee and Consortium Board.

Procedure

- i. Once the Secretariat has received the Internal Consortium Activity Reports and the Internal Consortium Costs Reports (see above) from all Work Packages, it compiles an aggregated version for presentation to the Executive Committee and/or the Consortium Board. This will be available not later than two weeks after the reception of the last WP-reports.
- ii. In case the reported data raises concerns, the Secretariat may take this up with the Project Co-ordinator and/or Executive Committee immediately.

Interim Activity Report (6 months)

Summary

The Interim Activity Report consists of brief information on key project actions. Interim activity reports are only required if this has been specified in Annex I of the contract. In TENCompetence we have a verbal agreement with the Project Officer to submit the Aggregated Internal Consortium Activity and Costs Report at month 6, in the format of the Period Reports (see below).

Collected data

Based on the Aggregated Internal Consortium Activity Reports and Internal Consortium Costs Reports.

Compiled report

An aggregate of the Internal Consortium Activity and Costs Reports, to be presented to the Project Officer in the format of the Periodic Reports.

Procedure

The Secretariat will provide the Project Officer with the Interim Activity Report at month 6.

Periodic reports - overview

Summary

The Periodic Reports comprise a total of four reports and three on-line questionnaires that have to be submitted to the Commission at the end of each reporting period (12 months)².

Collected data

The Periodic reports are to a large extent the aggregation and consolidation of the Internal Consortium Activity Reports and the Internal Consortium Costs Reports. In addition, the following information has to be provided:

<i>Additional information required for Periodic reports</i>	<i>Provided by</i>
An update of the Plan for using and disseminating the knowledge, as part of the Periodic Activity Report	WP9 + WP10
Financial Statement ('form C'), together with an audit certificate for those partners for whom the financial Community contribution exceeds E. 150.000,- over the cumulative reporting periods, as part of the Periodic Management Report	each contractor
Draft planning for the next 18 months (WP detailed implementation plans).	WP Leaders
Interim science and society reporting questionnaire (on-line)	Co-ordinator
Interim implementation of the gender action plan questionnaire (on-line)	Co-ordinator and each contractor
Interim socio-economic reporting questionnaire (on-line)	each contractor

Compiled reports

- Periodic Activity Report
- Periodic Management Report
- Periodic Report on the Distribution of the Community's Contribution
- Draft planning for next 18 months

On-line questionnaires:

- Interim Science and Society Reporting
- Interim Reporting on the Implementation of the Gender Action Plan
- Interim Socio-Economic Reporting

Procedure

- i. The Secretariat will coordinate report compilation in assistance to the Executive Committee.
- ii. The draft reports will be reviewed at an extraordinary meeting of the Executive Committee not later than two weeks after closure of the 12-month reporting period.
- iii. Depending on the outcome of that meeting, the drafts will be adjusted first, or may be sent directly to the annual meeting of the Consortium Board with a recommendation for their approval. The total set of reports shall be distributed to all Parties at least two weeks before the Consortium Board meeting (not later than five weeks after closure of the 12-month reporting period).
- iv. The Consortium Board discusses reports during its annual meeting, modifies or amends them, if necessary, and accepts the final version with a majority of the Parties.
- v. The final and approved reports will be submitted to the Commission not later than 45 days after closure of the 12-month reporting period.

The procedure for compilation, approval and submission of each of the four periodic reports is detailed below.

² For further details see 'Project Reporting in FP6', <http://cordis.europa.eu.int/fp6/find-doc-management.htm#reporting>

Periodic Activity Report

Summary
Provides an overview of project progress over the first 12 months, to be submitted to the Commission.
Collected data
Data required to compile the report includes: activities carried out; progress in relation to the project objectives; internal deliverables and deliverables status; and relevant problems and corrective actions taken. These data will to a large extent be derived from the Internal Consortium Activity Reports that were compiled over the 12 month reporting period.
Compiled report
The Periodic Activity report to a large extent consolidates the data collected through the Internal Consortium Activity Reports and the Internal Consortium Costs Reports. In addition, the following information will be provided in the Periodic Activity Report: <ul style="list-style-type: none"> • A publishable executive summary • An updated Plan for using and disseminating the knowledge
Procedure
<ol style="list-style-type: none"> i. The Secretariat will start compiling a first draft of the Periodic Activity Report in month 11 of the project. ii. WP9 and WP10 will be asked to deliver an updated Plan for using and disseminating the knowledge during month 11. iii. As soon as the Internal Consortium Activity Reports for months 10-12 have been received (one week after the end of month 12 latest), the draft report will be consolidated. iv. The report will be reviewed at an extraordinary meeting of the Executive Committee. v. For completion of the procedure, see 'Periodic reports - overview' for further details.

Periodic Management Report

Summary
Provides and audited overview and justification of costs incurred and resources deployed over the first 12 months, to be submitted to the Commission.
<p>Note: Contractors requesting a Community financial contribution for one or more reporting periods of less than EUR 150.000 need not submit an audit certificate, until the cumulative request for Community financial contribution is equal to or exceeds EUR 150.000 for the reporting periods for which an audit certificate has not yet been submitted. In all cases an audit certificate shall be submitted at the latest 45 days after the final reporting period. This final audit certificate shall cover all periods for which an audit certificate has not been previously submitted.</p>
Collected data
Data required to compile the report will be derived from the financial administration as kept by the Consortium Secretariat. In addition the following have to be provided by each contractor: <ul style="list-style-type: none"> • Form C to the contract: 'Model of Financial Statement per Activity for an Integrated Project'. • Audit Certificate

Compiled report

The Periodic Management report to a large extent consolidates the data collected through the Internal Consortium Activity Reports and the Internal Consortium Costs Reports. The Forms C and Audit Certificates will be annexed to the report.

Procedure

- i. The Secretariat will start compiling a first draft of the Periodic Management Report in month 11 of the project.
- ii. The contractors submit completed Forms C not later than one week after the end of month 12.
- iii. As soon as the Internal Consortium Costs Reports for months 10-12 have been received (not later than one week after the end of month 12), the draft will be consolidated.
- vi. The report will be reviewed at an extraordinary meeting of the Executive Committee not later than two weeks after closure of the 12-month reporting period (for completion of the procedure, see 'Periodic reports - overview').
- iv. The partners will submit the Audit Certificates not later than three weeks after the end of month 12 for inclusion in the report version that will be sent to the Board (the audit can be invoiced under the next reporting period).

Periodic Report on the Distribution of the Community's Contribution

Summary

Records the distribution of funding to each contractor over the first 12 months, to be submitted to the Commission.

Collected data

Data required to compile the report will be derived from the Internal Consortium Costs Reports that were accumulated over the 12-month reporting period

Compiled report

Overview of funding distribution to each contractor.

Procedure

- i. The Secretariat will compile the distribution overview on the basis of the accumulated Internal Consortium Costs Reports received (not later than one week after the end of month 12)
- ii. The report will be reviewed at an extraordinary meeting of the Executive Committee not later than two weeks after closure of the 12-month reporting period.
- iii. For completion of the procedure, see 'Periodic reports - overview'.

Draft Planning for next 18 months (Updating the Implementation Plan)

Summary

The annual draft planning for the next 18 months involves an update of the detailed implementation plan (see chapter 8 of the DoW), together with the corresponding budget information. On the basis of this information the Commission will decide on approval for the next 18 months.

Collected data

The DoW forms the basis for the updated information plan. The DoW is then detailed and adjusted for the next 18 months on the basis of:

- progress made during the previous 12 months
- new developments in research and technology development domains relevant to the project
- feedback received from the pilots (once available)

Compiled report

The detailed implementation plan will contain an update on:

- General description and internal deliverables
- TENCompetence assessment protocol
- Planning and timetable
- Graphical presentation of work packages
- Work package list
- Deliverables list months 13-30 (respectively 25-42, 36-48)
- Work package descriptions form (for each WP)
- IP Effort form months 13-30 (respectively 25-42, 36-48)
- Budget months 13-30 (respectively 25-42, 36-48)

Procedure

The Project Co-ordinator and the Executive Committee are responsible for the development, extrapolation and harmonisation of the Implementation Plan. The Secretariat will coordinate the day to day compilation of the Implementation Plan.

collect input from WP-leaders, Subcommittee Chairs, the contractors through the Board

- i. The Work Package Leaders, supported by the Work Package participants, design a detailed plan for their Work Packages for the following 18 months period. This is submitted to the Secretariat not later than 2 weeks before the end of month 12.
- ii. The Secretariat compiles an overall update of the implementation plan, including the budget overview.
- iii. This plan will be reviewed at an extraordinary meeting of the Executive Committee not later than two weeks after closure of the 12-month reporting period.
- iv. For completion of the procedure, see 'Periodic reports - overview'.

3. TENCompetence assessment protocol

In TENCompetence we qualify and quantify project outputs, and have devised a mechanism to link output norms to project staff input. This mechanism is explained in this chapter.

TENCompetence output types

TENCompetence deliverables comprise reports and prototypes (see table 10 for an overview). These deliverables are tightly related to the three project cycles. In addition to these deliverables, TENCompetence also produces outputs typically related to RTD activities throughout its four year life-cycle:

- Scientific publications and technologies³ resulting from the RTD activities (WP2-8)
- A variety of valorisation activities (WP9-10)

For the RTD outputs the TENCompetence assessment protocol defines:

- The (qualitative) criteria the output should meet
- The appraisal mechanism for awarding ‘output points’
- The required (quantitative) productivity, or output norm

For the valorisation activities, the impact will be assessed through:

- a set of indicators on project output consultation and use
- according to different target groups.

RTD output types and quality criteria

The quality and the productivity criteria of the RTD outputs (WP2-8) can be divided into two categories: scientific publications and technologies. As an RTD project, TENCompetence output should meet internationally recognised standards. Consequently the TENCompetence Assessment Protocol (TAP) is based upon, and is in line with, internationally approved protocols such as the Standard Evaluation Protocol (Standard Evaluation Protocol 2003 – 2009 For Public Research Organisations - ISBN 90-5588 278x, 2003).

TENCompetence applies the regular scientific output norm, i.e. every scientific full time equivalent (FTE) produces three scientific publications a year, or the equivalent in technologies. The input FTE's are corrected for overhead and for type of staff (see under ‘Output norms’ for details).

³ Technologies are the artefacts that are developed in the project, like software, specifications and methods/models.

Scientific publications cover:

- peer-reviewed journal articles
- (chapters in) scientific books
- scientific conference papers.

Technologies cover artefacts that are developed in the project:

- specifications
- methods/models
- software.

The quality criteria for each of these output types are outlined below. In addition as of 1 November 2006 the following general criteria should be applied:

- The output should contain an acknowledgment similar to "*The authors' efforts were (partly) funded by the European Commission in TENCompetence (IST-2004-02787) (<http://www.tencompetence.org>).*"
- The output should follow from the activities performed within TENCompetence. In case the output is the result of collaboration with an other externally funded project output points will be assigned proportionally.

Peer-reviewed journal articles

Accepted scientific journals are:

- All SCI/SSCI journals, including open access journals, see <http://www.isinet.com/cgi-bin/jrnlst/jloptions.cgi?PC=master>
- Journals that are listed on the extension list (See http://145.20.177.141/mediawiki/index.php/Advanced_Learning_Technology_Journals for the latest version), created by the Scientific Committee.

The journals contained in the extension list were selected for meeting the following criteria:

- Journal must publish papers in the field of Advanced Learning Technologies/Technology Enhanced Learning.
- Journal must have a published peer-review system with an editorial board consisting of well known scholars in the field.
- English language journals only (English is the working language within the project)
- Published papers come from international sources (so journals that publish mainly local papers are excluded)
- Appears in a fixed frequency.
- Publishes papers above 2000 words.

(Chapters in) scientific books

For books only scientific publishers that use an internal scientific review system will be granted output points. The publishers/series will be listed by the Scientific Committee. A first version of the list is provided at:

http://projects.edte.utwente.nl/ico/ico_uitgevers.html and may later be expanded by the Scientific Committee.

Scientific conference papers

Conference papers do only count for output points when the following criteria are met:

- The conference/workshop has had an open call for papers.
- The conference/workshop is international (local workshops are excluded).
- The acceptance rate has not been higher than 50%. The papers are in English.
- The conference/workshop uses a regular peer-reviewed system that is performed by a published scientific board.
- The papers are published by an international scientific publisher (e.g. Springer Lecture Notes, IEEE, ACM), but also less well known publishers can be included as long as the proceedings can be ordered through any book shop in the world).

An advisory list created by the Scientific Committee can be found at:

http://wiki.tencompetence.org/mediawiki/index.php/Advanced_Learning_Technology_Conferences

Other papers

In addition to the peer-reviewed journal articles, (chapters in) scientific books and scientific conference papers described above, the following additional types of papers are of relevance to TENCompetence and its stakeholders:

- Papers which report on project work, but which are not in approved journals (for example short papers presented at project workshops, but which are not selected for inclusion in the journal).
- Papers which discuss TENCompetence, but which are not reporting on the execution of the work plan, and thus are no candidate for output points (even when accepted in an approved journal).

Such articles should also be reported in the three-monthly activity reports and be made available publicly and to the reviewers, stating clearly their status.

Technology output

To make the technology output equivalent to the publication output, similar criteria for technology output are used as for publications:

- The technology is publicly available and accessible (e.g. through open-source channels like source forge).
- Members of the relevant RTD community can replicate the work on the basis of the documentation provided, i.e. rebuild alternative technologies on the basis of the requirements and the design.
- Members of the relevant RTD community can continue to work on and elaborate the technologies.
- The output is peer reviewed, e.g. for software this can be operationalised in that the software is used by others (number downloads, number of questions, number of contributions, etc.).

The output quality of prototypical software is assessed against the following criteria:

- Software scope qualitative
- Software coding qualitative.

Software scope qualitative criteria. Any software eligible for output points should comply with the following three criteria:

- Relevance: How relevant is the software for the further development of the domain?
- Significance: How important is the problem addressed by the software for the domain? Does the software have a community of users?
- Originality: Are the problems and approaches new? Is this a novel combination of existing techniques?

Software coding qualitative criteria. Software that is eligible for any output points should meet the following quality assurance criteria⁴.

- Code readable: Readable code includes systematic naming conventions and good formatting/indentation.
- Code commented: In Java based projects, JavaDoc comments should be used to drive HTML based documentation. Equivalent technologies to drive XML output are available in .NET (for example NDoc)
- Code structured: Well structured code refers to the way in which the code is modularised and the way in which code modules are grouped together. Good package naming conventions in Java and name spacing in .NET are evidence of a thoughtful approach to code structure.
- Code efficient: Although performance profiling is often addressed towards the later stages of a prototype (through tools such as JProfiler), code should be written with performance in mind. Evidence of this will include judicious choice of

⁴ *Software Quality Assurance (QA) and Open Source Maturity Model (OSMM) Development*, retrieved from http://www.jisc.ac.uk/uploaded_documents/ACFCDE.doc

variable data types and appropriate data structures, CPU and memory efficient operations, appropriate choice of XML parsing tools etc.

- Testing: Although there is some overhead required to put unit testing in place, developers who use it, find that unit testing saves development time. In principle, best practice is for each algorithm method to focus on one operation and for each method to be testable using a single unit test. In Java based projects, JUnit is a popular unit test framework. In .NET, NUnit is an equivalent.
- Deployment: All software and documentation should be written in such manner that is possible to install and run this software on a different environment than the one that is used for the development. This should not require any re-factoring in the code.
- API documentation available: Typically linked from the project website. The API documentation should be kept in sync with the source code by using a technology such as JavaDoc (for Java projects) or NDoc (for .NET projects)
- Source code publicly available with archive facility (e.g. Source Forge): Even with a well developed project website, source code should also be maintained on a public website with archiving facility.
- Licensed and download available: Typically a download is linked from the project website. The download should only be made available once appropriate licensing is in place.

Awarding RTD output points

The various RTD outputs described above are appraised as follows:

Table 5. Output points per output type

<i>Category</i>	<i>Output</i>	<i>Points</i>	<i>Comments</i>
Scientific output	Refereed SCI/SSCI journals	1	
	Journals on extension list	1	
	Book chapters	1	
	Conference papers	1	
	Books	3	Divided by the number of authors
	Ph.D. thesis	1	In addition to the published papers that are part of the thesis
Technology output	Software source code	≥ 1	see: RTD output norms, Technology output
	Software documentation	1	
	Specification	≥ 1	note: norms to be decided
	Models, Methods, etc.	1	note: norms to be decided
Non-scientific/non-technology output	Professional publications	0	All publications that do not comply to the norms of scientific publications.
	Papers for a professional audience	0	
	Software source code	0	
	Other	0	

Awarding output points for scientific output is rather straightforward. For technology output there is not yet such a well established and widely accepted procedure. Software eligible for output points in TENCompetence should represent a significant amount of work, equivalent to four months of effort for a senior scientific developer. A useful operational translation of this effort can be made by estimating the average number of lines that a senior scientific programmer can produce per effective working day. This is set to be 10 lines of actual code per day using a third generation programming language like Java or C#. On the average a line of code will consist of a single statement typically terminated by a semi column. In these 10 lines of code per day are included:

- all efforts required to make to functional design
- all efforts required for the technical design
- all efforts needed to make the code compliant to the quality assurance norms mentioned under ‘Software coding, qualitative criteria’ above.

Reuse of code of others should be clearly marked and does not count for the output.

Converting the four man month effort into the number of lines that could be awarded with one output point leads to the following calculation assuming a month has 17 effective working days: $10 * 17 * 4 = 680$ lines of code.

For a software component to become eligible for output points, it must be reviewed by at least two independent peer reviewers. The reviewing process will be blind. (NOTE: this can be in conflict with many open source licenses! Furthermore it might be quite cumbersome to remove all references to any of the authors from the code.). The software should meet the qualitative (scope and coding) and quantitative requirements mentioned above.

The outcome of the process may be any of these four:

- Accepted: the software meets all the required criteria and may be counted as scientific output according to the rules specified in section 'RTD output points awarding'.
- Accepted with minor revisions: the software does not meet at the criteria yet, but can be fairly easily be modified so it will be meeting all the set criteria. The modified software does not have to be peer reviewed hereafter. The authors have to describe how the modified the software to meet the reviewers remarks
- Accepted with major revisions: the software does not meet the criteria and needs some elaborate changes in order the meet the criteria. After re-submission the software will be peer reviewed again by the same reviewers to check if the changes led to a satisfying result.
- Rejected: the software does not meet the criteria by far and there are no foreseeable changes that could solve the situation. The software is not eligible to any output points.

RTD output norms

TENCompetence uses the regular scientific output norm, i.e. every scientific full time equivalent (FTE) produces three scientific publications a year, or the equivalent in technologies.

The scientific norm of three publications applies to assistant-, associate- and full professors. For other types of staff this norm is corrected for overhead and academic seniority (distinguished staff types are: professor, post-doc, and PhD student): postdocs and PhD students are expected to produce less than a professor. For technology output a distinction is made between senior technologists and junior technologists (equivalent to postdoc).

The number of points that a certain technology output will gain depends on the normative input size: three output points are earned for a technology that is developed and delivered by each FTE average senior technologist in the period of one year (to be corrected for overhead).

The overhead in European projects for scientific and technology work is estimated at 30% per scientific or technology FTE. This overhead consists of: additional distance communication, meetings, travel, and costs of inter-cultural communications/delays (e.g. language difficulties). The table below specifies the number of required output points for each scientific and technology FTE in the RTD activities.

Table 6. Output norms per staff type

Job title	Output norm in points per year per FTE
Full/Assoc./Assistant professor	$2.1 = 0.7 * 3$ output points
Post-doc	$1.4 = 0.7 * 2$ output points
Ph.D. student	$1.05 = 0.7 * 1.5$ output points
ICT senior staff	$2.1 = 0.7 * 3$ output points
ICT junior staff	$1.4 = 0.7 * 2$ output points
Supporting staff	n.a.

Following this approach the estimated, calculated output for year 1 (months 1-12) is 40 points.

Table 7. Output points for months 1-12

FTEs in the RTD activities (2/3 of project cycle 1)	19
Type of staff: professor & senior ICT staff (norm = 2.1)	2.1
Output norm year 1 ($19 * 2.1$)	40

However, because of the following two reasons, we expect to produce 20 output points in year 1. First of all, giving the Unified Process approach followed, the project output will not be distributed evenly over the full project period. And in particular for the technology and academic publications the output can only be produced after some considerable work in the project (e.g. after the first half year). Secondly, there is the regular delay in the publication process. An output point is counted for every published paper, but the publication process for some journals may take between 1 – 1.5 year. For this reason we will always report on publications that are 'in the pipeline' also (with the status: submitted or in press).

Example for calculating the expected output points per partner.

Example:

The output points per WP or per Partner can be derived from the allocated manmonths (translated to a budget based on number and costs per partner manmonth) from WP2-WP8 by following two steps:

1. Determine your budget for staff for WP2-WP8;
2. Map your staff to the categories according to table 6.

Suppose you have the following budget/manmonths available: 12 manmonths with average manmonths costs 5000. Two of this manmonths belong to WP9. This leaves us with 10 manmonths for WP2-WP8 and a budget of 50.000 for staff.

Next we determine our staff input (column B) on the basis of the required work and the available budget in such a way that it fits with the available budget of 50000: in this case by assigning 2 professor mm, 4 post-doc mm and 8 ICT junior mm. With the help of the output norms (column E, see table 7), we now can easily calculate the required output (column F). Finally, please note for the 1st year we only require 50% of the output norm (column G).

A	B	C	D	E	F	G
Staff category	mm	mm costs	total staff	output norm	output	output year 1
Professor	2	7000	14000	2.1	0.35	0.18
Post-doc	4	3500	14000	1.4	0.47	0.23
PhD			0	1.05	0.00	0.00
ICT-staff			0	2.1	0.00	0.00
ICT junior	8	2750	22000	1.4	0.93	0.47
			50000		1.75	0.88

Valorisation impact appraisal

The impact of the activities carried out in WP9 and WP10 on the target audiences cannot be directly measured and will be derived from a set of indicators. The following indicators will be made available for the project review.

Table 8. Impact information overview table

	<i>Impact indicator</i>	<i>Verification method</i>	<i>Responsible</i>
1	Number of training sessions, the number of attendees, their background and their interests and appreciation	questionnaire	WP9
2	Number of workshops, the number of attendees, their background and their interests and appreciation	questionnaire	WP10
3	The number of professional presentations and/or papers (including title, event, magazine info)		WP1
4	Set-up of the Ph.D. network, the number of participants, their background and their appreciation	questionnaire	WP9
5	Set-up of the Competence Network of Associate Partners, the number of participants, their background and their appreciation	questionnaire	WP9+10
6	Number of software downloads		WP10
7	Number of organisations using or planning to use (part of) the software, their type of activities and their user/customer base	survey	WP9+10
8	Number and type of spin-off projects	survey	WP9+10
9	Number of questions on the website, the background of the questioner, the response time, the appreciation	questionnaire	WP10
10	Number of news items posted, the number of news members; their background and their appreciation	questionnaire	WP10
11	Number of visits of the web site, the background of the visitors, the type of information they read, their appreciation	statistics, survey	WP10
12	Number of web references to TENCompetence	web statistics	WP10
13	Selection of key publications	scientific committee	WP1 through the Scientific Committee

Through the 3-monthly progress reports collected by WP1 the following information will be collected:

Table 9. Impact information 3-monthly progress reports

<i>Planned/ actual Dates</i>	<i>Type</i>	<i>Type of audience</i>	<i>Countries addressed</i>	<i>Size of audience</i>	<i>Source Link</i>	<i>Partner responsible /involved</i>
ddmmyy	Press release (press/radio/TV)	General public			Dspace handle	
	Media briefing	Higher education				
	Conference	Research				
	Exhibition	Industry (sector x)				
	Publications					
	Project web-site					
	Posters					
	Flyers					
	Direct e-mailing					
	Film/video					
	Training					
	Other ...					

Note 1: The overview table should be accompanied by a short description for each major activity (conference, exhibition, etc.) having taken place or planned since the last report. This should be part of the task of WP9 or WP10

Note 2: Scientific output is collected separately.

Using DSpace for output submission

All outputs produced by TENCompetence - articles, papers, presentations, software, specifications, etc - should be submitted to DSpace in the proper collection. *Materials that are not submitted to DSpace will not get you output points!*

DSpace can be accessed through the TENCompetence internal Moodle site, top right-hand corner of the opening page.

Please keep the following in mind in using DSpace:

- Until September Rob will perform the administrative tasks. From September onwards somebody else will take over. We will keep you informed.
- All documents should be in PDF format (except of course for source code); so create PDF of your documents and PowerPoint presentations before submitting.
- When you want to submit a document to more than one collection, you never should submit multiple items. You should always have only a single handle for every document. Add the document to the main collection and ask the administrator (Rob for now, from September somebody else) to map this item into the other collection(s), of course specifying the collections into which you want to submit the document.
- After you have submitted an item, keep an record of the handle issued.
- When you want to update an item, for example when you had to revise your pre-print, do not submit the new version as a new item. Ask Rob to change the original item and replace the document with the new item.
- If your pre-print gets published, send Rob the proper APA reference and handle-id to which the APA reference has to be added.
- Use always the same spelling for author names. First check whether an author already exists in the system and keep the spelling consistent. If you have to add a new author, use Last name and Initials.
- Be sure to enter Last name and Initials in the correct fields; the form requires Last name in the first field, and Initials in the second field.
- Be sure to select the correct collection. Do not submit presentations to the Publications and Preprints collection.
- Be sure to use the proper Creative Commons license for TENCompetence products:

“Creative Commons License” is a license that offers (some of) your rights (e.g. the right to read, redistribute, modify and use) to any taker on certain favourable conditions (e.g. for credits only or free for non-commercial use). In this contract when referring to a Creative Commons License we mean that any taker is free:

- a. to copy, distribute, display, and perform the work;
- b. to make derivative works (optional condition can be removed by the Parties if and where required);
- c. to make commercial use of the work.

Under the conditions that:

1. they attribute the work in the manner specified by the author or licensor;
2. for any reuse or distribution, they must make clear to others the licence terms of this work.

Any of these conditions can be waived if they get permission from the copyright holder.

“Open Source Software” means software licensed and distributed under an OSI certified Open Source agreement which guarantees the right to read, redistribute, modify, and use the software freely. (Examples of OSI certified agreements are the GNU GPL and the GNU LGPL.)

4. Internal deliverable quality assurance

This chapter starts with a brief description of specific characteristics of QA in RTD projects as opposed to regular software development projects. Then each work package is described in terms of a number of internal deliverables and their acceptance criteria. We distinguish two types of internal deliverables: critical ones to be reviewed externally to the WP, and non-critical ones to be reviewed internally within the WP.

The nature of RTD quality assurance in the Unified Process

The TENCompetence project has adopted the Unified Process (UP) to structure its development activities through three project cycles. The quality assurance strategy is therefore closely linked to the Unified Process.

Most quality assurance methodologies, like for example applied by Prince2, start from the premises that before the actual software development process starts 1) the project's business case is clear, which provides a yardstick against which to validate overall project progress, and 2) the system requirements have been clearly defined, which provides unambiguous guidelines for component- and integrated product testing as the project progresses.

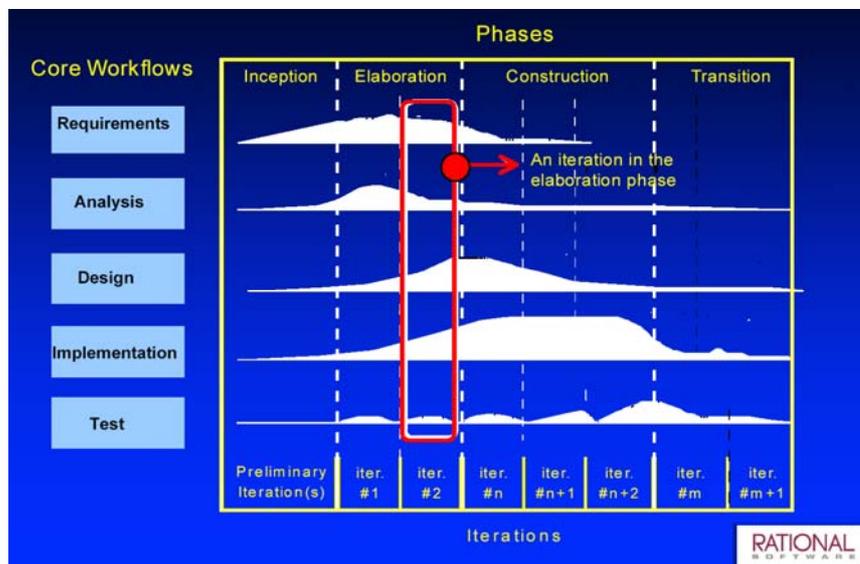


Figure 2. Unified Process

In an RTD project like TENCompetence these criteria are not so clear-cut. In fact, during project cycle 1 (UP Inception and Elaboration phases), the requirements for cycle 2 are formulated. At the same time however - following the UP axiom that each cycle covers the full workflows of requirements definition, analysis, design,

implementation and testing, a first integrated version of the TENCompetence infrastructure will already be produced and tested in a pilot.

This cyclic development approach - as opposed to the ‘waterfall approach’ to software development - has the following implications:

- During project cycle 1 the seven major problems that informed the TENCompetence objective served as a temporary business case.
- During the second and third project cycle the business case and requirements definition will become increasingly explicit, to be used as validation yardsticks.

Quality Assurance in TENCompetence focuses on outcomes (non-critical internal deliverables, critical internal deliverables, and ‘normal’ deliverables) rather than on activities, and combines three approaches to integral QA:

1. QA of scientific outputs and impact criteria contributed to individual work packages.
2. QA of technology outputs produced by work packages, to be integrated into the TENCompetence architecture.
3. Overall project validation criteria and an evaluation plan.

Each of these three mechanisms is detailed below.

Quality assurance criteria internal deliverables

For each of the work packages the internal deliverables and related quality criteria are detailed below. Critical internal deliverables, indicated in ***bold italics***, are reviewed externally to the WO. Those reviews are organised through the Secretariat. Non-critical internal deliverables are assessed within the WP itself.

WP1

	<i>Internal Deliverable</i>	<i>Acceptance criteria</i>	<i>Format</i>	<i>M</i>
1.3	Update of the TENCompetence Handbook	<ul style="list-style-type: none"> • Accessibility: structure and language • Effective in supporting the day-to-day project management tasks of WP-leaders, local project coordinators, Committee members and others involved in the project. 	Report	15
1.4	Three-monthly Internal Consortium Reports	<ul style="list-style-type: none"> • To be discussed in the Executive Committee or Consortium Board, depending on which meeting is held in that quarter 	Report	16 19 22 24
1.5	Informal Interim Activity Report	<ul style="list-style-type: none"> • Similar structure - but summarised - of the Periodic Activity Report 	Report	18

WP2

2.2	Integrated Roadmap (carried over from DIP-1)	<ul style="list-style-type: none"> Equal to deliverable D2.1 		18
2.3	Gender Mainstreaming Requirements Action Plan	<ul style="list-style-type: none"> It must include the perspective of women as user of the system Give a brief overview of gender related problems in reference to the project scope Specify the gender differentiation aspects in the system if they are needed Reasons why the differentiation is needed 	Report	17
2.4	Analysis Report	<ul style="list-style-type: none"> The methodology selected and applied is appropriate to address the problem as described above The gap analysis as carried out covers the requirements, the first version of the domain model and already available software components/systems It is clear how the results have been assessed by WP4 from a pilot perspective, and what this contributed to the analysis. (as part of WP4 task 4) The recommendations are <ol style="list-style-type: none"> relevant for the further elaboration of Use Cases and the Domain Model in internal deliverable ID2.5. correct, i.e. based on a correct interpretation of the present domain model (UML syntax), the use cases in the Requirements Report, and the available software components/systems of sufficient detail/granularity as compared to the level of detail of the first version of the domain model and use cases in the Requirements Report clear - concrete, unambiguous - to be usable as input to more elaborate Use Cases and Domain Model (internal deliverable ID2.5). 	UML diagrams, report	15
2.5	Elaborate Use Cases and Domain Model	<p>This updates, completes and further details the Use Cases and Domain model provided earlier in the Initial Requirements Report and the Analysis report (ID2.4).</p> <ul style="list-style-type: none"> Elaborate Use Cases <ul style="list-style-type: none"> explicitly linked to the domains of WP5, WP6, WP7 and WP8 exposing any additional functionality received from WP4 Domain model <ul style="list-style-type: none"> overall updated domain model expressed in UML Class and Activity diagrams specific domain models for WP5, WP6, WP7 and WP8 published widely through an open web site to collect feedback validated against the elaborate Use Cases (cover all major classes used in the Use Case descriptions) Reviewed by WP3 on quality and relevance for the WP3 development tasks. Validated by WP5-8 on relevance, correctness, completeness, granularity and clarity for cycle 2 activities (>month 18) 	UML diagrams, report	17

2.6	Reports on vision sessions, to be collected into an overall vision document	<ul style="list-style-type: none"> • Addressing the seven problems underlying TENCompetence • Which can be shared with external experts for validation • Concrete enough to be used as the basis for updating the Use Cases and Domain Model (ID2.7) • Concrete and explicit enough to be used by WP5-8 to inform their development activities in cycle 2 (>month 18) 	Report	17
2.7	Updated high level Use Cases, Domain Model, and underlying vision documents and pedagogical model definitions	<p>This updates ID2.5 on the basis of the vision developed as part of ID2.6.</p> <ul style="list-style-type: none"> • The Pedagogical models for lifelong competence development MUST: <ul style="list-style-type: none"> ○ address new, promising and innovative approaches to competence development ○ apply innovative learning technologies ○ integrate individual, collaborative and organizational learning • The Use Cases of ID2.5 are updated: <ul style="list-style-type: none"> ○ to explicitly reflect the vision document ID2.6 ○ to explicitly reflect the pedagogical models ○ in proper UML notation • The Domain Model of ID2.5 is updated: <ul style="list-style-type: none"> ○ cover all major classes used in the Use Case descriptions ○ in proper UML notation • Specific (sub)domain models for WP5, WP6, WP7 and WP8 	UML diagrams, report	22
2.8	Updated functional and non-functional requirements, and process definitions based on ID2.7	<p>This represents an update of the Initial Requirements Report (DIP-1) taking into account:</p> <ul style="list-style-type: none"> • the updated Use Cases and Domain model of ID2.7 • any functional and non-functional requirements communicated through WP4 for the future pilots <p>This ID2.8 should in turn inform WP5-8 in planning for DIP-3 (>month 24)</p> <ul style="list-style-type: none"> • Functional requirements: <ul style="list-style-type: none"> ○ explicitly linked to WP5-8 requirements ○ prioritised ○ in proper UML notation • Non-functional requirements, at a minimum covering: <ul style="list-style-type: none"> ○ security ○ performance ○ scalability ○ accessibility ○ compliance to standards ○ including requirements from future WP3 pilots • Reviewed by WP3 • Validated by WP4-8 on relevance, correctness, completeness, granularity and clarity for DIP-3 activities 	UML diagrams, report	20

WP3

3.2	Incident management procedure	<ul style="list-style-type: none"> • Procedure description, at least specifying: <ul style="list-style-type: none"> ○ incident notification ○ incident classification ○ handling procedure ○ response procedure • Underlying template in software tool 	Report	14
3.3	First release of the TENCompetence software (milestone 1)	<p>The TENCompetence software will be validated as part of WP4 pilots. The unit testing, integration testing and system testing takes place within WP3 itself. Integration testing and system testing may also involve others. Appendix 3 from deliverable D4.1 may serve as the guideline for this (see p. 120 for the proposed test approach), outlining the following steps:</p> <ul style="list-style-type: none"> • Make a test plan. • Scope definition (e.g. only the calculating functions and not the personal administration will be tested and which test methods used, e.g. decision table). • Cluster definitions. • Description of test conditions. • Test cases. • Description of test environment. • Documentation of the configuration of the test environment (to make the test repeatable). • Test results of each test. • Report of the test results (to report on the number of failed and passed tests). 	Software, report, Excel	17
3.4	Setup of the infrastructure for the pilot(s)	<p>For the first pilots WP3 will set up the infrastructure. Successful technical testing (ID3.3) is a prerequisite. Acceptance of the infrastructure by WP4 as the 'customer' will focus on the following criteria (based on appendix 4, deliverable D4.1):</p> <ul style="list-style-type: none"> • the 'Critical Functionality Test' results • Extensibility test results • Security test results <p>All the other criteria listed in appendix 4 are for the time being the internal responsibility of WP3 and therefore do not require external assessment.</p>	Sign-off by user organization	14
3.5	<i>Intake procedure, including checklists for WP5-8 components</i>	<ul style="list-style-type: none"> • Peer reviewed <p>Should contain:</p> <ul style="list-style-type: none"> • Process flow of the procedure • Assignment of responsibilities for the steps in the intake procedure • Checklist for control of submitted components 	MS-Word	15

3.6	Version 2.0 of the overall design report of the TENCompetence architecture, according to the SOA approach, and the WP5-8 components to integrate, and an updated user interaction model.	<ul style="list-style-type: none"> • Availability of the document • Good English (doesn't need to be perfect, but it has to be good enough) • Peer reviewed <p>It should contain:</p> <ul style="list-style-type: none"> • Sections for the 4 + 1 views as from P. Kruchten's ideas. • Architectural model, including: <ul style="list-style-type: none"> ○ Use case view (aspects of architectural relevant use cases and non-functional requirements) ○ Logical view (analysis model) ○ Implementation view (layers and components, high level description of the components per layer, integration with external components) ○ Process view (performance, load, processes & threads) ○ Deployment view (hardware infrastructure, libraries, deployment and configuration). • Updated developer guidelines, showing concretely how to fit in the WP5-8 components for tool and application developers, including <ul style="list-style-type: none"> ○ Naming conventions ○ Documentation templates ○ Component interface how-to (protocols, standards, ...) ○ Use of version control system (CVS /SourceSafe /Subversion /...) ○ Coding standards 	Report	20
3.7	Final API definitions for the second release, beta version.	<ul style="list-style-type: none"> • Peer reviewed <p>API definitions should contain descriptions for:</p> <ol style="list-style-type: none"> 1. Used communication protocol(s) 2. Method signature: input and output parameters, name of the method 3. Used dataformat(s) 4. Error codes and explanations 5. Preconditions (requirements for the state when starting execution) 6. Postconditions (description of the result and end state when the execution is completed) 7. General description 	Specification of the APIs	24
3.8	Second major release of the TENCompetence software, realising version 2.0 of the architectural design report	<ul style="list-style-type: none"> • Software has to be available • Code compliant to the developer guidelines from the Architecture Design Report (ID3.6) • Code compliant to the list of quality attributes (this is the list that was defined by WP5 and Ruud) • Peer reviewed • Completed system test, including test report, <i>according to guidelines of Appendix 3 from deliverable D4.1</i> • No outstanding high priority bugs • Functionality for realising the WP2 and WP4 requirements, if delivered to WP3 timely 	Source code and test report	27

3.9	Updated design for the third release of the TENCompetence software	<ul style="list-style-type: none"> • Availability of the document • Good English (doesn't need to be perfect, but it has to be good enough) • Peer reviewed <p>It should contain:</p> <ul style="list-style-type: none"> • Sections for the 4 + 1 views as from P. Kruchten's ideas. • Architectural model, including: <ul style="list-style-type: none"> ○ Use case view (aspects of architectural relevant use cases and non-functional requirements) ○ Logical view (analysis model) ○ Implementation view (layers and components, high level description of the components per layer, integration with external components) ○ Process view (performance, load, processes & threads) ○ Deployment view (hardware infrastructure, libraries, deployment and configuration). • Updated developer guidelines, showing concretely how to fit in the WP5-8 components for tool and application developers, including <ul style="list-style-type: none"> ○ Naming conventions ○ Documentation templates ○ Component interface how-to (protocols, standards, ...) ○ Use of version control system (CVS /SourceSafe /Subversion /...) • Coding standards 	Report, wireframe	30
-----	--	---	-------------------	----

WP4

4.2	<i>Evaluation outcomes of cycle 1 pilots of the initial system</i>	<ul style="list-style-type: none"> • Should at least produce the following information <ul style="list-style-type: none"> ○ report on the proceedings of the pilots ○ report on the application of the evaluation methodology ○ tools, methods, and approaches taken during the evaluation ○ questions posed and answered by the evaluation ○ evaluation results and recommendations for the TENCompetence client ○ how the TENCompetence aspects were validated ○ list of recommendations and suggestions for Cycle 2 • Should at least signals the completion of the following activities <ul style="list-style-type: none"> ○ end of the pilots' curricula and related courses ○ end of the evaluation activities ○ finalization of data collection and analysis ○ completion of related reporting and information diffusion 	Report	20
4.3	<i>Execution and Evaluation Plan for cycle 2 pilots</i>	<ul style="list-style-type: none"> • Should at least produce the following information <ul style="list-style-type: none"> ○ pilots academic materials and students' profile ○ competence development programs and pedagogical models definitions ○ assessment methodologies ○ learning scenarios, use cases ○ registration policy and expected academic results ○ evaluation plans for cycle 2 pilots • Would signal the end of the following activities <ul style="list-style-type: none"> ○ planning the evaluation of cycle 2 pilots, methodology defined ○ finishing the preparations for execution of the pilots. ○ related materials collected and developed ○ related tools selected or defined 	Report	26
4.4	Evaluation implementation plan Digital Cinema pilot	<ul style="list-style-type: none"> • Implementation plan of the proposed evaluation methodology (below) in terms of: <ul style="list-style-type: none"> ○ required resources ○ roles and responsibilities ○ time plan • The evaluation methodology should contain: <ul style="list-style-type: none"> ○ development of the seven TENCompetence objectives and (a selection of) the research questions of the pilot evaluation plan into measurable outcomes; and vice versa it should indicate how the expected outcomes relate back to the research questions ○ description of methods for: <ul style="list-style-type: none"> ▪ data selection (why these samples of respondents) ▪ data collection (how will the data be collected), including instruments for measurement and evaluation ▪ data analysis (which general and more specific analytic strategies will be pursued) ▪ enhancing validity ▪ dealing with ethical issues, including gender-specific dimensions ○ indicate which aspects of the methodology are specific 	Report	14

		<p>to this pilot and which are general, and also to be used in other pilots</p> <ul style="list-style-type: none"> • how the evaluation of this pilot and its results will relate to other pilots 		
4.5	Evaluation implementation plan ICT training pilot	<ul style="list-style-type: none"> • Implementation plan of the proposed evaluation methodology (below) in terms of: <ul style="list-style-type: none"> ○ required resources ○ roles and responsibilities ○ time plan • The evaluation methodology should contain: <ul style="list-style-type: none"> ○ development of the seven TENCompetence objectives and (a selection of) the research questions of the pilot evaluation plan into measurable outcomes; and vice versa it should indicate how the expected outcomes relate back to the research questions ○ description of methods of: <ul style="list-style-type: none"> ▪ data selection (why these samples of respondents) ▪ data collection (how will the data be collected), including instruments for measurement and evaluation ▪ data analysis (which general and more specific analytic strategies will be pursued) ▪ enhancing validity ▪ dealing with ethical issues, including gender-specific dimensions ○ indicate which aspects of the methodology are specific to this pilot and which are general, and also to be used in other pilots ○ how the evaluation of this pilot and its results will relate to other pilots 	Report	15
4.6	<i>Evaluation methodology for cycle-2 pilots</i>	<ul style="list-style-type: none"> • The evaluation methodology can be considered successful to the extent to which: <ul style="list-style-type: none"> ○ the expected outcomes will <ul style="list-style-type: none"> ▪ give different stakeholders information about what has taken place in the various pilots, ▪ make what happens within the pilots understandable in all its complexity, and explain how various factors are connected ▪ (to a lesser extent) make it possible to predict what will happen in future pilots ○ the translation of the TENCompetence objectives into outcomes is adequate ○ the various methods are adequate for answering the research questions and will deliver valid results ○ the combination of methods will together provide a more valid result (triangulation) ○ ethical issues are adequately dealt with ○ the methodology will enable conclusions from comparison of the results of the various pilots ○ the evaluation is feasible given the available effort, time and other constraints. 	Report	24

4.7	Finalize distribution of cycle 1 pilots information	<ul style="list-style-type: none"> • Should at least communicate the following information <ul style="list-style-type: none"> ○ functional requirements of the pilots to WP3 ○ non-functional requirements of the pilots to WP2 ○ CDPs of the pilots' curricula to WP7 ○ developed UOL to WP6 ○ scenarios and use-cases to WP2 • Feedback on at least the following information should be gathered <ul style="list-style-type: none"> ○ WP3's feedback on functional requirements • WP2's feedback on non-functional requirements 	On-line information	13
4.8	Finalize distribution of cycle 2 pilots information	<ul style="list-style-type: none"> • Should at least communicate the following information <ul style="list-style-type: none"> ○ functional requirements of the pilots to WP3 ○ non-functional requirements of the pilots to WP2 ○ CDPs of the pilots' curricula to WP7 ○ developed UOL, pedagogical models, and assessment methods to WP6 ○ scenarios and use-cases to WP2 ○ relevant social networking plans to WP8 • Feedback on at least the following information should be gathered <ul style="list-style-type: none"> ○ WP3's feedback on functional requirements ○ WP2's feedback on non-functional requirements 	On-line information	30

WP5

5.2	Roadmap for Knowledge Resource Sharing & Management RTD	<ul style="list-style-type: none"> • Roadmap should at least indicate: <ul style="list-style-type: none"> ○ Summary state of the art in the field ○ Summary of WP5 contributions to RTD over first 18 months (based on underlying Milestones/Internal Deliverables) ○ Summary of trends/'next steps' in the field and TENCompetence needs ○ Priority RTD actions for next 18 months • Accessible not only for WP5 partners but also for external audiences active in the field 	Report	17
5.3	First cycle KRSM prototype system ready for validation	<ul style="list-style-type: none"> • Open availability of the prototype • Proposal for prototype validation strategy according to the guidelines in Appendix 3 of Deliverable D4.1, and ID5.8 in the form of a validation protocol • Accessible to user-testing 	Software component, report	14
5.4	<i>First cycle KRSM validated prototype system for delivery to WP3</i>	<ul style="list-style-type: none"> • Code and documentation openly available • Validation proof, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, for Unit Testing: <ul style="list-style-type: none"> ○ Annex 1, Checklist <ul style="list-style-type: none"> ▪ Calamities ▪ Functionality ▪ Storage ○ Annex 3, quality attributes: <ul style="list-style-type: none"> ▪ reliability ▪ usability ▪ efficiency ▪ maintainability ▪ portability • Coding quality: <ul style="list-style-type: none"> ○ Correct use of Object Oriented programming (e.g. clear separation of concerns) ○ Put all the code in TENCompetence CVS ○ Proper use of try/catch and managing of Exceptions ○ Java doc (or java doc like) ○ Inline comments ○ Java source code conventions (e.g. Eclipse, D3.1), ○ Human readable variable names ○ Indentation ○ No dead code ○ Efficient code (hard to specify, to be assessed by an experienced programmer) ○ No hard coded values (for items that might change in the future) ○ English (code, documentation, all) • Quantity (i.e. Number of code lines as described in the project handbook) • List of proposed improvements/enhancements for the next release 	Software component	18

5.5	API definitions to be delivered to WP3	<ul style="list-style-type: none"> • General description • Input and output parameters • Used data format(s) • Preconditions (requirements for the state when starting execution) • Post conditions (description of the result and end state when the execution is completed) • Likely (if any) errors and their handling 	Specification	18
5.6	Second cycle KRSM system – beta release, comprising integrated components/service implementations ready for validation	<ul style="list-style-type: none"> • Code and documentation openly available • Motivated and documented functional updates from ID5.9 • Proposal for prototype validation strategy according to the guidelines of Deliverable D4.1, Appendix 3 in the form of a validation protocol • Accessible to user-testing 	Software component, report	24
5.7	<i>Second cycle KRSM system – final release, comprising integrated components/service implementations for delivery to WP3</i>	<ul style="list-style-type: none"> • Code and documentation openly available • Availability of the APIs to be used for integration in the WP3 client • Documentation on bug fixes on the basis of KRSM second cycle system evaluation results (ID5.10) • List of proposed improvements/enhancements for the next release 	Software component	30
5.8	KRSM first cycle prototype evaluation plan	<ul style="list-style-type: none"> • According to guidelines of deliverable D4.1, Appendix 3, Annex (2) - Test Plan 	Report	13
5.9	KRSM first cycle prototype evaluation outcomes	<ul style="list-style-type: none"> • Reported along the lines of the Evaluation Plan (ID5.8) • List of proposed improvements/enhancements for the next release 	Report	16

5.10	KRSM second cycle system evaluation results	<ul style="list-style-type: none"> • Validation proof based on tests with real users, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, for Unit Testing: <ul style="list-style-type: none"> ○ Annex 1, Checklist <ul style="list-style-type: none"> ▪ Calamities ▪ Functionality ▪ Storage ○ Annex 3, quality attributes: <ul style="list-style-type: none"> ▪ reliability ▪ usability ▪ efficiency ▪ maintainability ▪ portability • Coding quality: <ul style="list-style-type: none"> ○ Correct use of Object Oriented programming (e.g. clear separation of concerns) ○ Put all the code in TENCompetence CVS ○ Proper use of try/catch and managing of Exceptions ○ Java doc (or java doc like) ○ Inline comments ○ Java source code conventions (e.g. Eclipse, D3.1), ○ Human readable variable names ○ Indentation ○ No dead code ○ Efficient code (hard to specify, to be assessed by an experienced programmer) ○ No hard coded values (for items that might change in the future) ○ English (code, documentation, all) ○ Quantity (i.e. Number of code lines as described in the project handbook) • Impact analysis • List of proposed improvements/enhancements for the next release 	Report	28
5.11	Updated Roadmap as outcome of task 1 running till month 30	<ul style="list-style-type: none"> • Update of ID5.2, at least indicating: <ul style="list-style-type: none"> ○ Summary state of the art in the field ○ Summary of WP5 contributions to RTD over first 30 months (based on underlying Milestones/Internal Deliverables) ○ Summary of trends/'next steps' in the field and TENCompetence needs ○ Priority RTD actions for next 18 months • Accessible not only for WP5 partners but also for external audiences active in the field 	Report	30

WP6

6.2	WP6 prototype tools - tested first release	<p>For the authoring and runtime tools:</p> <ul style="list-style-type: none"> • Code and documentation openly available • Test results, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, for Unit Testing: <ul style="list-style-type: none"> ○ Annex 1, Checklist <ul style="list-style-type: none"> ▪ Calamities ▪ Functionality ▪ Storage ○ Annex 3, quality attributes (limited testing): <ul style="list-style-type: none"> ▪ reliability, ▪ usability, ▪ efficiency, ▪ maintainability, ▪ portability • Coding quality: <ul style="list-style-type: none"> ○ Correct use of Object Oriented programming (e.g. clear separation of concerns) ○ Put all the code in TENCompetence CVS ○ Proper use of try/catch and managing of Exceptions ○ Java doc (or java doc like) ○ Inline comments ○ Java source code conventions (e.g. Eclipse, D3.1), ○ Human readable variable names ○ Indentation ○ No dead code ○ Efficient code (hard to specify, to be assessed by an experienced programmer) ○ No hard coded values (for items that might change in the future) ○ English (code, documentation, all) • Quantity (i.e. Number of code lines as described in the project handbook) • List of proposed improvements/enhancements for the next release 		18
6.3	WP6 specifications for second release	<ul style="list-style-type: none"> • Based on the evaluation outcomes (ID6.2) • Taking into account any new requirements resulting from WP2 and WP4 activities • Specified in functional terms in such a way that they can also be validated by other WPs on relevance (like WP3 and WP4) • In addition, ‘technical specifications’ that are understandable to programmers to produce a first implementation - no vague discussions or ‘google’ inventories • Including a listing of specifications that may impact/overlap with functionality being developed in other Aspect WPs 	Report, wireframe (?)	18

6.4	WP6 prototypical tools - second release	<p>For the authoring and runtime tools:</p> <ul style="list-style-type: none"> • Code and documentation openly available • Listing of added functionality since the first release • Test results, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, for Unit Testing: <ul style="list-style-type: none"> ○ Annex 1, Checklist <ul style="list-style-type: none"> ▪ Calamities ▪ Functionality ▪ Storage ○ Annex 3, quality attributes (limited testing): <ul style="list-style-type: none"> ▪ reliability, ▪ usability, ▪ efficiency, ▪ maintainability, ▪ portability • Coding quality: <ul style="list-style-type: none"> ○ Correct use of Object Oriented programming (e.g. clear separation of concerns) ○ Put all the code in TENCompetence CVS ○ Proper use of try/catch and managing of Exceptions ○ Java doc (or java doc like) ○ Inline comments ○ Java source code conventions (e.g. Eclipse, D3.1), ○ Human readable variable names ○ Indentation ○ No dead code ○ Efficient code (hard to specify, to be assessed by an experienced programmer) ○ No hard coded values (for items that might change in the future) ○ English (code, documentation, all) • Quantity (i.e. Number of code lines as described in the project handbook) • List of proposed improvements/enhancements for the next release • Tools accessible to user testing (ID6.5) 	Software, report	24
6.5	Validation testing and usability evaluation results	<ul style="list-style-type: none"> • Availability of improved tools (bugs fixed and additional functionality since ID6.4) and documentation • Validation test results with users, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, annex 3 'Quality attributes' (extensive testing): <ul style="list-style-type: none"> ○ reliability ○ usability ○ efficiency ○ maintainability ○ portability • Impact analysis • List of proposed improvements/enhancements for the next release 	Reports	30

WP7

7.2	Prototypical navigation service and learner support service; Report with summary of WP outputs; and a Roadmap of competence development programmes RTD	<p>Validated prototypical navigation service:</p> <ul style="list-style-type: none"> • Code and documentation openly available • Validation outcomes of intended functionality <ul style="list-style-type: none"> ○ Delivering navigation service to advise next most suitable learning activity and –path. ○ Predictions based on a combination of collaborative filtering and group profiles • List of proposed functional improvements/enhancements for the next release • Learner support will be delivered by the University of Sofia and referred to in the report • Report with WP summary outcomes: <ul style="list-style-type: none"> ○ Main body of text brief summary of underlying milestones/internal deliverables; include also references to publications and dissemination activities ○ Well structured, good English, and also accessible to the wider TENCompetence audiences ○ Underlying milestones/internal deliverables themselves included as annexes, or references to dSpace handles • Roadmap of competence development programmes RTD should at least indicate: <ul style="list-style-type: none"> ○ Summary state of the art in the field ○ Summary of WP7 contributions to RTD over first 18 months (based on underlying Internal Deliverables) ○ Indication of the WP impact - to what extent has the state of the art been advanced ○ Summary of trends/‘next steps’ in the field and TENCompetence needs ○ Priority RTD actions for next 18 months ○ Accessible not only for WP partners but also for external audiences active in the field 	Software; report	18
7.3	<i>Refined version of the learning path description, suitable to be brought as input to standardization bodies</i>	<ul style="list-style-type: none"> • Specification of the gaps it fills, in comparison with current practices in e.g. LOM, IMS LD and SCORM • Including learner preferences • With feedback from external experts 	Report	24
7.4	Specifications Report based on analysis of current practices and user needs for selecting or creating competence development programs	<ul style="list-style-type: none"> • Based on: <ul style="list-style-type: none"> ○ Practices include the Digital Cinema Pilot, at least one more non-University domain and two University curricula ○ Comparison and explanation of differences in practice ○ Recommendations for future interfaces for addressing identified user needs • ‘Specifications’ in terms of descriptions that are understandable to programmers to produce a first implementation - no vague discussions or ‘google inventories’ 	Report, wireframe(?)	20

7.5	Evaluated user interface designs for selecting or creating CDPs	<ul style="list-style-type: none"> • Based on outcomes of ID7.4 • Motivation of design decisions • Prototype at least working at user interface level • Evaluated in a well-designed study including verification of hypotheses • List of proposed improvements/enhancements for the next release 	Report	22
7.6	API definition of M7.2 to be delivered to WP3	<ul style="list-style-type: none"> • General description • Input and output parameters • Used dataformat(s) • Preconditions (requirements for the state when starting execution) • Postconditions (description of the result and end state when the execution is completed) • Likely (if any) errors and their handling 	Documentation	18
7.7	Validated service for selecting or creating CDPs	<ul style="list-style-type: none"> • Code and documentation openly available • Test results, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, for Unit Testing: <ul style="list-style-type: none"> ○ Annex 1, Checklist <ul style="list-style-type: none"> ▪ Calamities ▪ Functionality <ul style="list-style-type: none"> ➢ Means for adding learning objects to the repository ➢ Graphical overviews of CDPs ➢ Visual feedback on prerequisite relations, learner interest and knowledge level ➢ Storage of selected or created CDPs ➢ Curriculum planning and time planning ▪ Storage ○ Annex 3, quality attributes: <ul style="list-style-type: none"> ▪ reliability ▪ usability: tested with real users ▪ efficiency ▪ maintainability ▪ portability • Coding quality: <ul style="list-style-type: none"> ○ Correct use of Object Oriented programming (e.g. clear separation of concerns) ○ Put all the code in TENCompetence CVS ○ Proper use of try/catch and managing of Exceptions ○ Java doc (or java doc like) ○ Inline comments ○ Java source code conventions (e.g. Eclipse, D3.1), ○ Human readable variable names ○ Indentation ○ No dead code ○ Efficient code (hard to specify, to be assessed by an experienced programmer) ○ No hard coded values (for items that might change in the future) ○ English (code, documentation, all) • Quantity (i.e. Number of code lines as described in the project handbook) • Impact analysis • A plan for the adaptation and integration of the component during project cycle 3 	Software	30

7.8	Validated positioning service, first release	<p>Validated prototypical positioning service:</p> <ul style="list-style-type: none"> • Code and documentation openly available • Validation outcomes of intended functionality <ul style="list-style-type: none"> ○ Delivering positioning results comparable to experts using latent semantic analysis ○ Computing document similarity in learner profiles and learning networks, simulating /validating advice • List of proposed functional improvements/enhancements for the next release • Learner support will be delivered by the University of Sofia and referred to in the report 	Software, documentation	18
7.9	API definition of ID7.8 to be delivered to WP3	<ul style="list-style-type: none"> • General description • Input and output parameters • Used dataformat(s) • Preconditions (requirements for the state when starting execution) • Postconditions (description of the result and end state when the execution is completed) • Likely (if any) errors and their handling 	Documentation	30
7.10	Validated positioning service, second release	<ul style="list-style-type: none"> • Motivated changes with respect to ID7.8, based on experimental outcomes • Code and documentation openly available • Test results, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, for Unit Testing: <ul style="list-style-type: none"> ○ Annex 1, Checklist <ul style="list-style-type: none"> ▪ Calamities ▪ Functionality <ul style="list-style-type: none"> ➢ Delivering positioning results comparable to experts using latent semantic analysis ➢ Computing document similarity in learner profiles and learning networks, simulating / validating advise ▪ Storage ○ Annex 3, quality attributes (extensive testing): <ul style="list-style-type: none"> ▪ reliability ▪ usability: tested with real users ▪ efficiency ▪ maintainability ▪ portability • Coding quality: <ul style="list-style-type: none"> ○ Correct use of Object Oriented programming (e.g. clear separation of concerns) ○ Put all the code in TENCompetence CVS ○ Proper use of try/catch and managing of Exceptions ○ Java doc (or java doc like) ○ Inline comments ○ Java source code conventions (e.g. Eclipse, D3.1), ○ Human readable variable names ○ Indentation ○ No dead code ○ Efficient code (hard to specify, to be assessed by an experienced programmer) ○ No hard coded values (for items that might change in the future) ○ English (code, documentation, all) • Quantity (i.e. Number of code lines as described in the 	Software	30

		<ul style="list-style-type: none"> project handbook) • Impact analysis • A plan for the adaptation and integration of the component during project cycle 3 		
7.11	Validated navigation service	<ul style="list-style-type: none"> • Motivated changes with respect to ID7.2, based on experimental outcomes • Code and documentation openly available • Test results, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, for Unit Testing: <ul style="list-style-type: none"> ○ Annex 1, Checklist <ul style="list-style-type: none"> ▪ Calamities ▪ Functionality <ul style="list-style-type: none"> ➢ Delivering positioning results comparable to experts using latent semantic analysis ➢ Computing document similarity in learner profiles and learning networks, simulating / validating advise ▪ Storage ○ Annex 3, quality attributes (extensive testing): <ul style="list-style-type: none"> ▪ reliability ▪ usability: tested with real users ▪ efficiency ▪ maintainability ▪ portability • Coding quality: <ul style="list-style-type: none"> ○ Correct use of Object Oriented programming (e.g. clear separation of concerns) ○ Put all the code in TENCompetence CVS ○ Proper use of try/catch and managing of Exceptions ○ Java doc (or java doc like) ○ Inline comments ○ Java source code conventions (e.g. Eclipse, D3.1), ○ Human readable variable names ○ Indentation ○ No dead code ○ Efficient code (hard to specify, to be assessed by an experienced programmer) ○ No hard coded values (for items that might change in the future) ○ English (code, documentation, all) • Quantity (i.e. Number of code lines as described in the project handbook) • Impact analysis • A plan for the adaptation and integration of the component during project cycle 3 	Software	30

WP8

8.2	State-of-the-Art, design and validated 1st release of TENCompetence Competence Observatory Prototype	<p>For the design:</p> <ul style="list-style-type: none"> • Software component specification for the prototype • Meeting the TENCompetence architecture requirements for integration at level 2 • Properly documented to allow continued development • Evaluation plan <p>This document will be continuously updated and will represent the basis for the selection and development of existing tools and components</p> <p>For the validated 1st release:</p> <ul style="list-style-type: none"> • Code and documentation openly available • Test results, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, for Unit Testing: <ul style="list-style-type: none"> ○ Annex 1, Checklist <ul style="list-style-type: none"> ▪ Calamities ▪ Functionality ▪ Storage ○ Annex 3, quality attributes (limited testing): <ul style="list-style-type: none"> ▪ reliability ▪ usability ▪ efficiency ▪ maintainability ▪ portability • Coding quality: <ul style="list-style-type: none"> ○ Correct use of Object Oriented programming (e.g. clear separation of concerns) ○ Put all the code in TENCompetence CVS ○ Proper use of try/catch and managing of Exceptions ○ Java doc (or java doc like) ○ Inline comments ○ Java source code conventions (e.g. Eclipse, D3.1), ○ Human readable variable names ○ Indentation ○ No dead code ○ Efficient code (hard to specify, to be assessed by an experienced programmer) ○ No hard coded values (for items that might change in the future) ○ English (code, documentation, all) • Quantity (i.e. Number of code lines as described in the project handbook) • List of proposed improvements/enhancements for the next release 	Software, documentation, wireframe(?)	18
8.3	API definition to be delivered to WP3	<ul style="list-style-type: none"> • General description • Input and output parameters • Used dataformat(s) • Preconditions (requirements for the state when starting execution) • Postconditions (description of the result and end state when the execution is completed) • Likely (if any) errors and their handling 	Specification	18

8.4	2nd Release of TENCompetence Competence Observatory Prototype, ready for validation	<ul style="list-style-type: none"> • Motivated changes with respect to ID8.2, based on experimental outcomes • Open availability of coding and documentation • Accessible to user-testing • Validation protocol available 	Software, documentation	24
8.5	Competences ontology	<ul style="list-style-type: none"> • State of the art - reflecting work carries out in various standardisation fora • Accessible to the external TENCompetence stakeholders 	Report	24
8.6	Validated release of the Competence Observatory prototype, based on functional testing and usability evaluation results	<ul style="list-style-type: none"> • Code and documentation openly available • Overview of corrections and enhancements since ID8.2 • Test results, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, for Unit Testing: <ul style="list-style-type: none"> ○ Annex 1, Checklist <ul style="list-style-type: none"> ▪ Calamities ▪ Functionality ▪ Storage ○ Annex 3, quality attributes (extensive testing): <ul style="list-style-type: none"> ▪ reliability ▪ usability: tested with real users ▪ efficiency ▪ maintainability ▪ portability • Coding quality: <ul style="list-style-type: none"> ○ Correct use of Object Oriented programming (e.g. clear separation of concerns) ○ Put all the code in TENCompetence CVS ○ Proper use of try/catch and managing of Exceptions ○ Java doc (or java doc like) ○ Inline comments ○ Java source code conventions (e.g. Eclipse, D3.1), ○ Human readable variable names ○ Indentation ○ No dead code ○ Efficient code (hard to specify, to be assessed by an experienced programmer) ○ No hard coded values (for items that might change in the future) ○ English (code, documentation, all) • Quantity (i.e. Number of code lines as described in the project handbook) • Impact analysis • A plan for the adaptation and integration of the component during project cycle 3 	Software, report	30
8.7	State-of-the-Art, design and validated 1st release of TENCompetence Overview tool, proactive sharing agent and game-like embedded dynamics	<p>For the design:</p> <ul style="list-style-type: none"> • Software component specification for a tool providing an overview of competence development opportunities • Meeting the TENCompetence architecture requirements for integration at level 2 • Properly documented to allow continued development and • Evaluation plan <p>This document will be continuously updated and will represent the basis for the selection and development of existing tools and components</p> <p>For the prototypical tool:</p> <ul style="list-style-type: none"> • Code and documentation openly available 	Software, specifications, wireframe(?)	18

		<ul style="list-style-type: none"> • Test results, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, for Unit Testing: <ul style="list-style-type: none"> ○ Annex 1, Checklist <ul style="list-style-type: none"> ▪ Calamities ▪ Functionality <ul style="list-style-type: none"> ○ search facilities ○ collaborative filtering and social interaction support ○ data-mining ○ visualization of the search results ○ agents and game like dynamics as social connectors ▪ Storage ○ Annex 3, quality attributes (limited testing): <ul style="list-style-type: none"> ▪ reliability ▪ usability ▪ efficiency ▪ maintainability ▪ portability • Coding quality: <ul style="list-style-type: none"> ○ Correct use of Object Oriented programming (e.g. clear separation of concerns) ○ Put all the code in TENCompetence CVS ○ Proper use of try/catch and managing of Exceptions ○ Java doc (or java doc like) ○ Inline comments ○ Java source code conventions (e.g. Eclipse, D3.1), ○ Human readable variable names ○ Indentation ○ No dead code ○ Efficient code (hard to specify, to be assessed by an experienced programmer) ○ No hard coded values (for items that might change in the future) ○ English (code, documentation, all) • Quantity (i.e. Number of code lines as described in the project handbook) • List of proposed improvements/enhancements for the next release 		
8.8	API definition to be delivered to WP3	<ul style="list-style-type: none"> • Description of the API to be used for by the WP3 client/server functionality 	Specification	18
8.9	2nd release of TENCompetence Overview tool, proactive sharing agent and game-like embedded dynamics, ready for validation	<ul style="list-style-type: none"> • Motivated changes with respect to ID8.7, based on experimental outcomes • Open availability of coding and documentation • Accessible to user-testing • Validation protocol available 	Software, report	24

8.10	Validated release, based on functional testing and usability evaluation results for TENCompetence Overview tool, proactive sharing agent and game-like embedded dynamics	<p>Final release, comprising integrated components/service implementations for delivery to WP3</p> <ul style="list-style-type: none"> • Code and documentation openly available • Available Services Interfaces • Test results, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, for Unit Testing: <ul style="list-style-type: none"> ○ Annex 1, Checklist <ul style="list-style-type: none"> ▪ Calamities ▪ Functionality ▪ Storage ○ Annex 3, quality attributes (extensive testing): <ul style="list-style-type: none"> ▪ reliability ▪ usability: tested with real users, validating (or not) of research hypothesis as formulated in SOA. ▪ efficiency ▪ maintainability ▪ portability • Coding quality: <ul style="list-style-type: none"> ○ Correct use of Object Oriented programming (e.g. clear separation of concerns) ○ Put all the code in TENCompetence CVS ○ Proper use of try/catch and managing of Exceptions ○ Java doc (or java doc like) ○ Inline comments ○ Java source code conventions (e.g. Eclipse, D3.1), ○ Human readable variable names ○ Indentation ○ No dead code ○ Efficient code (hard to specify, to be assessed by an experienced programmer) ○ No hard coded values (for items that might change in the future) ○ English (code, documentation, all) • Quantity (i.e. Number of code lines as described in the project handbook) • Impact analysis • A plan for the adaptation and integration of the component during project cycle 3 	Software, report	30
8.11	State-of-the-Art, design and validated 1st release of TENCompetence Network Management Tool Prototype	<p>For the design:</p> <ul style="list-style-type: none"> • Software component specification for the network management tool • Meeting the TENCompetence architecture requirements for integration at level 2 • Properly documented to allow continued development and • Evaluation plan <p>This document will be continuously updated and will represent the basis for the selection and development of existing tools and components</p> <p>For the prototypical tool:</p> <ul style="list-style-type: none"> • Code and documentation openly available • Test results, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, for Unit Testing: <ul style="list-style-type: none"> ○ Annex 1, Checklist <ul style="list-style-type: none"> ▪ Calamities ▪ Functionality, fostering social exchange: 	Software, specifications, wireframe(?)	18

		<ul style="list-style-type: none"> ➤ peer tutoring ➤ member participation ➤ terms of use ▪ Storage ○ Annex 3, quality attributes (limited testing): <ul style="list-style-type: none"> ▪ reliability ▪ usability ▪ efficiency ▪ maintainability ▪ portability • Coding quality: <ul style="list-style-type: none"> ○ Correct use of Object Oriented programming (e.g. clear separation of concerns) ○ Put all the code in TENCompetence CVS ○ Proper use of try/catch and managing of Exceptions ○ Java doc (or java doc like) ○ Inline comments ○ Java source code conventions (e.g. Eclipse, D3.1), ○ Human readable variable names ○ Indentation ○ No dead code ○ Efficient code (hard to specify, to be assessed by an experienced programmer) ○ No hard coded values (for items that might change in the future) ○ English (code, documentation, all) • Quantity (i.e. Number of code lines as described in the project handbook) • List of proposed improvements/enhancements for the next release 		
8.12	API definition to be delivered to WP3	<ul style="list-style-type: none"> • Description of the API to be used for by the WP3 client/server functionality 	Specification	18
8.13	2nd release of TENCompetence Network Management Tool, ready for validation	<ul style="list-style-type: none"> • Open availability of coding and documentation • Motivated changes with respect to ID8.11, based on experimental outcomes • Accessible to user-testing • Validation protocol available 	Software, report	24

8.14	Validated release, based on functional testing and usability evaluation results for TENCompetence Network Management Tool	<p>Final release, comprising integrated components/service implementations for delivery to WP3 as integrated component of ID8.10, Overview Tool prototype</p> <ul style="list-style-type: none"> • Code and documentation openly available • Available Services Interface • Test results, presented on the basis of the guidelines as contained in deliverable D4.1, appendix 3, for Unit Testing: <ul style="list-style-type: none"> ○ Annex 1, Checklist <ul style="list-style-type: none"> ▪ Calamities ▪ Functionality ▪ Storage ○ Annex 3, quality attributes (extensive testing): <ul style="list-style-type: none"> ▪ reliability ▪ usability: tested with real users, validating (or not) of research hypothesis as formulated in SOA. ▪ efficiency ▪ maintainability ▪ portability • Coding quality: <ul style="list-style-type: none"> ○ Correct use of Object Oriented programming (e.g. clear separation of concerns) ○ Put all the code in TENCompetence CVS ○ Proper use of try/catch and managing of Exceptions ○ Java doc (or java doc like) ○ Inline comments ○ Java source code conventions (e.g. Eclipse, D3.1), ○ Human readable variable names ○ Indentation ○ No dead code ○ Efficient code (hard to specify, to be assessed by an experienced programmer) ○ No hard coded values (for items that might change in the future) ○ English (code, documentation, all) • Quantity (i.e. Number of code lines as described in the project handbook) • Impact analysis • A plan for the adaptation and integration of the component during project cycle 3 	Software, report	30
------	--	---	------------------	----

WP9

9.2	<i>Training Plan for Participants in Cycle 1 Pilots</i>	<ul style="list-style-type: none"> • Specification of training objectives • Specification of required learning resources • Specification of planned training activities 	Plan	13
9.3	Report on Winter School 2007	<ul style="list-style-type: none"> • Successful organization of events as specified in DOW • Summary of events, and links to presentations from all • Analysis of evaluation feedback, attendance and topics addressed 	Report	15
9.4	Report on Internal Workshop for Researchers and Developers	<ul style="list-style-type: none"> • Successful organization of events as specified in DOW • Summary of events, and links to presentations from all • Analysis of evaluation feedback, attendance and topics addressed 	Report	20
9.5	Report on Training for Cycle 1 Pilots	<ul style="list-style-type: none"> • Achievement of training objectives • Provided learning resources • Organized training activities • Analysis of evaluation feedback, attendance and topics addressed 	Report	21
9.6	Report on Design and Evaluation plan for TENCompetence Training Pilot	<ul style="list-style-type: none"> • Specification of training objectives • Specification of required learning resources • Specification of planned training activities 	Report	20
9.7	<i>Training Plan for Participants in Cycle 2 Pilots</i>	<ul style="list-style-type: none"> • Specification of training objectives • Specification of required learning resources • Specification of planned training activities 	Plan	24
9.8	Report on Competence Network for Associate Partners	<ul style="list-style-type: none"> • Successful organization of events as specified in DOW • Summary of events, and links to presentations from all • Analysis of evaluation feedback, attendance and topics addressed 	Report	24
9.9	Report on Training for General Stakeholders	<ul style="list-style-type: none"> • Successful organization of events as specified in DOW • Summary of events, and links to presentations from all • Analysis of evaluation feedback, attendance and topics addressed 	Report	24
9.10	Report on Winter School 2008	<ul style="list-style-type: none"> • Successful organization of events as specified in DOW • Summary of events, and links to presentations from all • Analysis of evaluation feedback, attendance and topics addressed 	Report	27
9.11	Report on Training for Cycle 2 Pilots	<ul style="list-style-type: none"> • Achievement of training objectives • Provided learning resources • Organized training activities • Analysis of evaluation feedback, attendance and topics addressed 	Report	30
9.12	<i>Report on the Evaluation of the TENCompetence Training Pilot</i>	<ul style="list-style-type: none"> • Achievement of training objectives • Provided learning resources • Organized training activities • Analysis of evaluation feedback, attendance and topics addressed 	Report	30

WP10

10.2	Critical Use Cases & three potential business model outlines (m 18)	To be acceptable the business models must involve effective requirements gathering. The result will be the identification of niches within which actors can add value through activities with the TENCompetence infrastructure.	Report	18
10.3	List with contacts, potential users, subscribers and associated partners (ongoing, consolidated version month 30).	Acceptability criteria are <ul style="list-style-type: none"> • Identification of potential groups of users, subscribers and associates. • Subscribers enrolled in online system to receive TENCompetence information and invitations to activities • Associate Partners have signed MOUs identifying their contribution to the project, and a designated project worker to manage their participation. 	Report	30
10.4	Compendium of workshop papers, special issues and book chapters edited by TENCompetence	Availability must include: <ul style="list-style-type: none"> • Specifications and standards • Architectures • Software components • Reports and manuals All deliverables must be: <ul style="list-style-type: none"> • Easy to find informal and formal project outcomes • Easy to access deliverable A transparent and efficient content management process should be in place.	Web and paper	24
10.5	Overview of workshops and events organised by TENCompetence and their impact. (month 24)	Acceptability criteria are: <ul style="list-style-type: none"> • Successful organization of events as specified in DOW • Summary of events, and links to presentations from all • Summary of peer reviewed proceedings for all academic workshops • Analysis of evaluation feedback, attendance and topics addressed 	Report	24
10.6	Report on TENCompetence standardisation initiatives, gathering together and presenting standardization initiatives and outcomes provided by other WPs (month 30)	An acceptable report should include <ul style="list-style-type: none"> • Analysis of the need for the specifications developed, within TENCompetence and in wider context • Description of the specifications developed • Approach to wider adoption and (if appropriate) standardization. • An appendix including all specification documentation 	Report	30

Some notes on QA of architecture integration during cycle 1

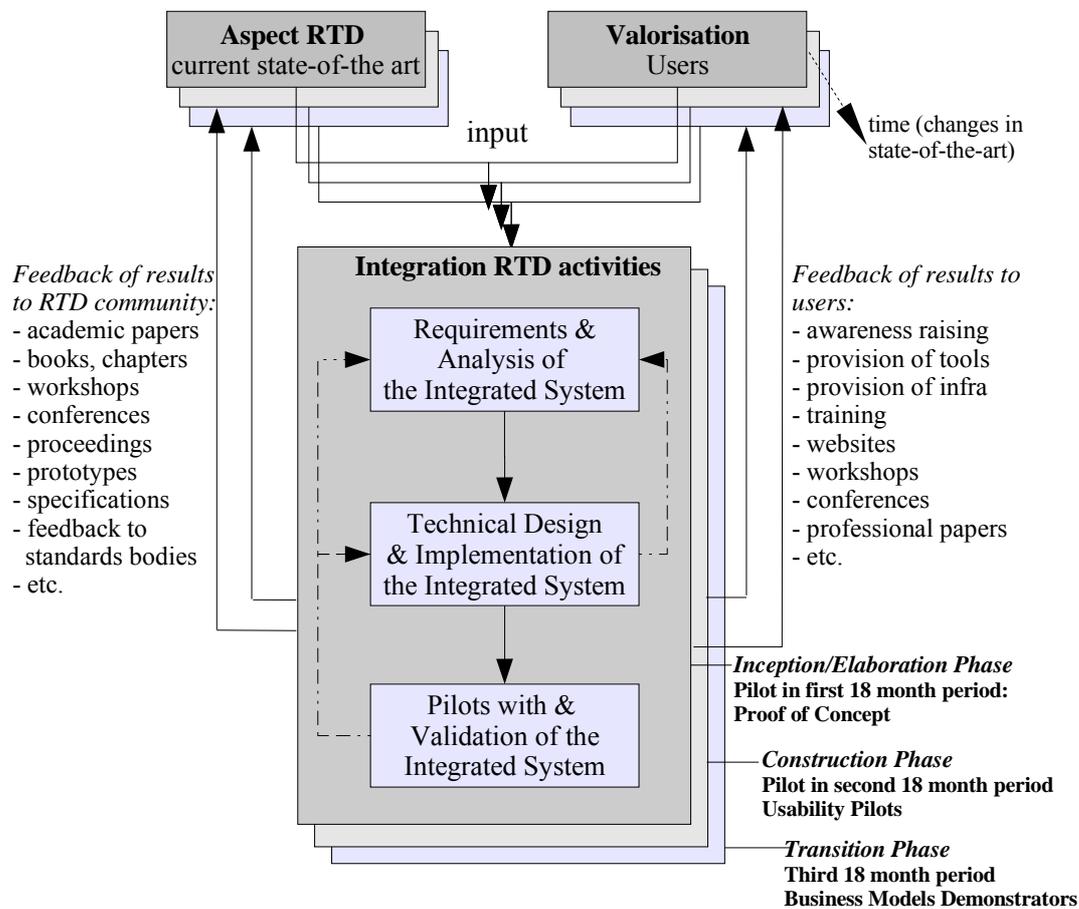


Figure 3. Relation between TENCompetence WP-types

QA in TENCompetence is organised according to the following principles:

- Within the integration RTD activities the cycle of Requirements & Analysis of the Integrated System (WP2) feeds Technical Design & Implementation of the Integrated System (WP3), which is then validated through Pilots with & Validation of the Integrated System (WP4).
- The Aspect RTD work packages (WP5-8) deliver inputs (software products) to be integrated into the technical architecture (WP3).
- All software development activities - whether in the aspect RTD WPs 5-8 or the integration WP3 - have to adhere to proper change management, version management, etc. procedures. These are described in chapter 8 of this Handbook.

5. Deliverable appraisal and submission

In this chapter we describe the procedure for internal deliverable appraisal and submission towards the Commission.

Deliverables overview

The DIP-2 contains the following overview of deliverables for months 13-30:

Table 10. Deliverables overview months 13-30

	<i>Deliverable title</i>	<i>WP</i>	<i>p.m.</i>	<i>type</i> ⁵	<i>diss. level</i> ⁶	<i>month</i> ⁷
D1.2	Periodic Report (covering progress and expenditure reporting over months 13-24), including the DIP-3 for the period covering month 25-42 (month 24).	1	34	R	CO	24
D2.1	Integrated roadmap	2	12	R	PU	18
D2.2	Updated use case models and underlying vision documents and pedagogical model definitions	2	24	R	PU	24
D2.3	Updated functional and non-functional requirements, and process definitions	2	11	R	PU	30
D3.2	First tested version of the Integrated TENCompetence System	3	35	P	PU	18
D3.3	Aggregate of internal deliverable M3.8 and internal deliverables ID3.6, ID3.7 and ID3.9, including updated design and second release software	3	70	P	PU	30
D4.2	Report on the results of the evaluation of the cycle 1 pilots, including internal deliverable outcomes ID4.2, ID4.3, ID4.4, ID4.6, ID4.7 and ID4.9.	4	38	R	PU	21
D4.3	Report, containing internal deliverable outcomes ID4.5, ID4.8, ID4.10	4	24	R	PU	30
D5.1	Report with summary of WP outputs over month 1-18, and a roadmap for knowledge resources sharing and management	5	35	R+P	PU	18
D5.2	Report containing internal deliverable outcomes ID5.3-ID5.10	5	28	R+P	PU	30
D6.1	Report with summary of WP outputs over month 1-18, and a roadmap of learning activities and units of learning RTD	6	15	R	PU	18
D6.2	Compilation of internal deliverable outcomes ID6.4-ID6.5	6	56	R+P	PU	30

⁵ Please indicate the nature of the deliverable using one of the following codes:

R = Report; **P** = Prototype; **D** = Demonstrator; **O** = Other

⁶ Please indicate the dissemination level using one of the following codes:

PU = Public

PP = Restricted to other programme participants (including the Commission Services).

RE = Restricted to a group specified by the consortium (including the Commission Services).

CO = Confidential, only for members of the consortium (including the Commission Services).

⁷ Month in which the deliverables will be available. Month 0 marking the start of the project, and all delivery dates being relative to this start date.

D7.1	Report with summary of WP outputs over first 18 months and a roadmap of competence development programmes RTD	7	15	R+P	PU	18
D7.2	Compilation of internal deliverable outcomes ID7.3-ID7.11	7	29	R+P	PU	30
D8.1	Report with overall WP8 results during month 1-18, and a roadmap of networks for lifelong competence development RTD	8	18	R+P	PU	18
D8.2	Compilation of internal deliverable outcomes ID8.2-ID8.14	8	36	R+P	PU	30
D9.2	Report, containing internal deliverable outcomes ID9.2-ID9.12	9	32	R	PU	30
D10.2	Report with an assessment of the WP-results including ID10.2 – ID10.6	10	58	R	PU	30

Deliverable types and internal assessment criteria

TENCompetence produces deliverables related to different activity-types:

- Project management activities
- Integration activities
- RTD activities
- Training and dissemination

Chapter 6 of the DoW ‘Outline implementation plan for the full duration of the project’, lists the objectives, results and assessment indicators for these different activity types for the full 4-year project period. These are further detailed for cycle 1 in chapter 8 ‘Detailed implementation plan – first 18 months’. Here the criteria are summarised for the purpose of internal assessment (see also chapter 4: “Quality assurance criteria per work package”):

Table 11. Deliverables internal assessment criteria.

	<i>Deliverable title</i>	<i>Assessment criteria</i>	<i>Type</i>	<i>M</i>
D1.2	Periodic Report (covering progress and expenditure reporting over months 13-24), including the DIP-3 for the period covering month 25-42 (month 24).	<ul style="list-style-type: none"> • Defined by the Commission 	Report	24
D2.1	Integrated roadmap	Initially the criteria were defined as: Integrated Roadmap report should incorporate and integrate the recommendations of D5.1, D6.1, D7.1, D8.1, D9.1 and D10.1 on: <ul style="list-style-type: none"> ○ State of the art in the field ○ Relevance to the seven core issues of TENCompetence ○ Summary of WP contributions to state of the art ○ Summary of trends/‘next steps’ in the field ○ Summary TENCompetence needs 	Report	18

		<ul style="list-style-type: none"> ○ Priority RTD actions for next 18 months <p>But the ambition has been reduced to:</p> <ul style="list-style-type: none"> ● Summarise and incorporate the underlying internal deliverables: <ul style="list-style-type: none"> ○ ID2.3: Gender Mainstreaming Requirements Action Plan (month 17) ○ ID2.4: Analysis Report based on gap analysis (month 15) ○ ID2.5: Elaborate Use Cases and Domain Model (month 17) ● In addition, the report should: <ul style="list-style-type: none"> ○ Be accessible: <ul style="list-style-type: none"> ▪ well structured ▪ main body max. 30 p., rest as annexes ▪ clear English ○ not only for WP5 partners but also for external audiences active in the field 		
D2.2	Updated use case models and underlying vision documents and pedagogical model definitions	<ul style="list-style-type: none"> ● All criteria of the underlying internal deliverables ● In addition, the following criteria pertaining to the overall deliverable report: <ul style="list-style-type: none"> ○ Well structured ○ Proper English 	Report, wireframes (?)	24
D2.3	Updated functional and non-functional requirements, and process definitions	<ul style="list-style-type: none"> ● Functional requirements: <ul style="list-style-type: none"> ○ explicitly linked to WP5-8 requirements ○ prioritised ● Non-functional requirements, at a minimum covering: <ul style="list-style-type: none"> ○ security ○ performance ○ scalability ○ accessibility ○ compliance to standards ○ including requirements from future WP3 pilots ● Reviewed by WP3 ● Validated by WP4-8 on relevance, correctness, completeness, granularity and clarity for DIP-3 activities 	Report, wireframes (?)	30

D3.2	First tested version of the Integrated TENCompetence System	<ul style="list-style-type: none"> • Open availability of code and documentation • Test results <p>The first release will become available in month 17 as ID3.3. The TENCompetence software will be validated as part of WP4 pilots. The unit testing, integration testing and system testing takes place within WP3 itself. Integration testing and system testing may also involve others. Appendix 3 from deliverable D4.1 may serve as the guideline for this (see p. 120 for the proposed test approach), outlining the following steps:</p> <ul style="list-style-type: none"> ○ Make a test plan. ○ Scope definition (e.g. only the calculating functions and not the personal administration will be tested and which test methods used, e.g. decision table). ○ Cluster definitions. ○ Description of test conditions. ○ Test cases. ○ Description of test environment. ○ Documentation of the configuration of the test environment (to make the test repeatable). ○ Test results of each test. 	Software, documentation and test report	18
D3.3	Aggregate of internal deliverable M3.8 and internal deliverables ID3.6, ID3.7 and ID3.9, including updated design and second release software	<ul style="list-style-type: none"> • All criteria of the underlying internal deliverables • In addition, the following criteria pertaining to the overall deliverable report: <ul style="list-style-type: none"> ○ Well structured ○ Proper English 	Report, software	30
D4.2	Report on the results of the evaluation of the cycle 1 pilots, including internal deliverable outcomes ID4.2, ID4.4, ID4.5, and ID4.7.	<ul style="list-style-type: none"> • All criteria of the underlying internal deliverables • In addition, the following criteria pertaining to the overall deliverable report: <ul style="list-style-type: none"> ○ Well structured ○ Proper English 	Report	21
D4.3	Report, containing internal deliverable outcomes ID4.3, ID4.6, ID4.8	<ul style="list-style-type: none"> • All criteria of the underlying internal deliverables • In addition, the following criteria pertaining to the overall deliverable report: <ul style="list-style-type: none"> ○ Well structured ○ Proper English 	Report	30

D5.1	Report with summary of WP outputs over month 1-18, and a roadmap for knowledge resources sharing and management	<ul style="list-style-type: none"> • Report with WP summary outcomes: <ul style="list-style-type: none"> ○ Main body of the text is a brief summary of underlying milestones/internal deliverables; include also references to publications and dissemination activities ○ Assessment criteria for the underlying milestones/internal deliverables apply ○ Accessible: <ul style="list-style-type: none"> ▪ well structured <ul style="list-style-type: none"> ➢ main body of text max. 30 pages ➢ underlying milestones/internal deliverables themselves as annexes, or even reference to DSpace handles ▪ clear English ▪ also be understood by external TENCompetence stakeholders • Roadmap of competence development programmes RTD should at least indicate: <ul style="list-style-type: none"> ○ Summary state of the art in the field ○ Relevant to the seven core issues of TENCompetence ○ Summary of WP5 contributions to RTD over first 18 months (based on underlying Internal Deliverables) ○ Summary of trends/'next steps' in the field ○ TENCompetence needs ○ Priority RTD actions for next 18 months <p>Accessible not only for WP partners but also for external audiences active in the field</p>	Report, software	18
D5.2	Report containing internal deliverable outcomes ID5.3-ID5.10	<ul style="list-style-type: none"> • All criteria of the underlying internal deliverables • In addition, the following criteria pertaining to the overall deliverable report: <ul style="list-style-type: none"> ○ Well structured ○ Proper English 	Report, software	30
D6.1	Report with summary of WP outputs over month 1-18, and a roadmap of learning activities and units of learning RTD	<ul style="list-style-type: none"> • Report with WP summary outcomes: <ul style="list-style-type: none"> ○ Main body of the text is a brief summary of underlying milestones/internal deliverables; include also references to publications and dissemination activities ○ Assessment criteria for the underlying milestones/internal deliverables apply ○ Accessible: <ul style="list-style-type: none"> ▪ well structured <ul style="list-style-type: none"> ➢ main body of text max. 30 pages ➢ underlying milestones/internal deliverables themselves as annexes, or even reference to DSpace handles ▪ clear English ▪ also be understood by external TENCompetence stakeholders • Roadmap of learning activities and units of learning RTD should at least indicate: <ul style="list-style-type: none"> ○ Summary state of the art in the field ○ Relevant to the seven core issues of TENCompetence ○ Summary of WP6 contributions to RTD over first 	Report, software	18

		<p>18 months (based on underlying Internal Deliverables)</p> <ul style="list-style-type: none"> ○ Summary of trends/'next steps' in the field ○ TENCompetence needs ○ Priority RTD actions for next 18 months <p>Accessible not only for WP partners but also for external audiences active in the field</p>		
D6.2	Compilation of internal deliverable outcomes ID6.4-ID6.5	<ul style="list-style-type: none"> ● All criteria of the underlying internal deliverables ● In addition, the following criteria pertaining to the overall deliverable report: <ul style="list-style-type: none"> ○ Well structured ○ Proper English 	Report, software	30
D7.1	Report with summary of WP outputs over first 18 months and a roadmap of competence development programmes RTD	<ul style="list-style-type: none"> ● Report with WP summary outcomes: <ul style="list-style-type: none"> ○ Main body of the text is a brief summary of underlying milestones/internal deliverables; include also references to publications and dissemination activities ○ Assessment criteria for the underlying milestones/internal deliverables apply ○ Accessible: <ul style="list-style-type: none"> ▪ well structured <ul style="list-style-type: none"> ➢ main body of text max. 30 pages ➢ underlying milestones/internal deliverables themselves as annexes, or even reference to DSpace handles ▪ clear English ▪ also be understood by external TENCompetence stakeholders ● Roadmap of competence development programmes RTD should at least indicate: <ul style="list-style-type: none"> ○ Summary state of the art in the field ○ Relevant to the seven core issues of TENCompetence ○ Summary of WP7 contributions to RTD over first 18 months (based on underlying Internal Deliverables) ○ Summary of trends/'next steps' in the field ○ TENCompetence needs ○ Priority RTD actions for next 18 months <p>Accessible not only for WP partners but also for external audiences active in the field</p> 	Report, software	18
D7.2	Compilation of internal deliverable outcomes ID7.3-ID7.11	<ul style="list-style-type: none"> ● All criteria of the underlying internal deliverables ● In addition, the following criteria pertaining to the overall deliverable report: <ul style="list-style-type: none"> ○ Well structured ○ Proper English 	Report, software	30

D8.1	Report with overall WP8 results during month 1-18, and a roadmap of networks for lifelong competence development RTD	<ul style="list-style-type: none"> • Report with WP summary outcomes: <ul style="list-style-type: none"> ○ Main body of the text is a brief summary of underlying milestones/internal deliverables; include also references to publications and dissemination activities ○ Assessment criteria for the underlying milestones/internal deliverables apply ○ Accessible: <ul style="list-style-type: none"> ▪ well structured <ul style="list-style-type: none"> ➢ main body of text max. 30 pages ➢ underlying milestones/internal deliverables themselves as annexes, or even reference to DSpace handles ▪ clear English ▪ also be understood by external TENCompetence stakeholders • Roadmap of competence development programmes RTD should at least indicate: <ul style="list-style-type: none"> ○ Summary state of the art in the field ○ Relevant to the seven core issues of TENCompetence ○ Summary of WP8 contributions to RTD over first 18 months (based on underlying Internal Deliverables) ○ Summary of trends/'next steps' in the field ○ TENCompetence needs ○ Priority RTD actions for next 18 months <p>Accessible not only for WP partners but also for external audiences active in the field</p>	Report, software	18
D8.2	Compilation of internal deliverable outcomes ID8.2-ID8.14	<ul style="list-style-type: none"> • All criteria of the underlying internal deliverables • In addition, the following criteria pertaining to the overall deliverable report: <ul style="list-style-type: none"> ○ Well structured ○ Proper English 	Report, software	30
D9.2	Report, containing internal deliverable outcomes ID9.2-ID9.12	<ul style="list-style-type: none"> • All criteria of the underlying internal deliverables • In addition, the following criteria pertaining to the overall deliverable report: <ul style="list-style-type: none"> ○ Well structured ○ Proper English • To be acceptable this report should <ul style="list-style-type: none"> ○ provide a clear overview of the training activities ○ provide a review of training strategy with specific recommendations for the coming period 	Report	30
D10.2	Report with an assessment of the WP-results including ID10.2 – ID10.6	<ul style="list-style-type: none"> • All criteria of the underlying internal deliverables • In addition, the following criteria pertaining to the overall deliverable report: <ul style="list-style-type: none"> ○ Well structured ○ Proper English • To be acceptable this report should <ul style="list-style-type: none"> ○ provide a clear overview of the training activities ○ provide a review of training strategy with specific recommendations for the coming period 	Report	30

Internal deliverable acceptance procedure

Provision of deliverables is the responsibility of the Work Package Leader. Advice on the quality of a deliverable will be provided by (one of) the Advisory Committees, depending on the nature of the deliverable, or in special cases a selected team that may include external advisors. Final endorsement of the deliverable before submission to the Commission is the responsibility of the Project Co-ordinator.

To be able to review a deliverable properly, it should be submitted to the Secretariat not later than one month before its due date to allow enough time for review and adjustments.

Deliverable submission to the Commission

Deliverables should be submitted on the due date as specified in table 10, with a maximum contractually permitted delay of 45 days. Any further delay in the submission of a deliverable must be reported in the Interim activity report or the Periodic activity report. If one or more of the partners is late in submitting deliverables, the Co-ordinator may submit the other partners' deliverables to the Commission.

Deliverables are often written reports but can also take another form, for example the completion of a prototype, etc. In such cases the deliverable should also be documented in a written record of the achievement of the deliverable, including any available supporting material.

Deliverables are submitted to the Commission electronically and on paper as for project reports, unless otherwise specified in the DoW. Each deliverable has a standard front page.

If no modifications to any reports are required, they will be deemed to be approved within 90 days of receipt. Approval of any report does not imply exemption from any audit or review.

In those cases where, following the evaluation of the reports, the Commission considers that the consortium is not performing satisfactorily, it may:

- Reject the reports submitted and request that the consortium complete the work foreseen within a deadline to be established by the Commission. Reports and deliverables shall be re-submitted once completed;
- Approve the reports and deliverables but subject the project to re-negotiation. In this case, the Commission may suspend the project.
- Terminate the contract.

The formal approval of deliverables by the Commission services forms part of the Periodic review process (see next chapter).

Hints for writing milestone and deliverable reports

In writing milestone and deliverable reports, please keep in mind the following:

- You are not only writing for your close colleagues, but also for the members of the Review Committee!
- Structure the document well:
 - Include a management summary (and have it proof-read by your partner).
 - Use the main text body for the ‘central message’ and use annexes for (technical) details, research findings, articles, etc. that are not necessary to follow the general line of reasoning.
 - When a document may be of interest to different audiences, include a ‘readers guide’.
- Use the same - preferably rather formal - style throughout the report.
- If you do not feel secure about writing in English, have your text edited by a colleague who is a native English speaker, or by a professional translator.
- Use a front page meeting the EU requirements.

Plan adjustments

The following table lists the decision-making responsibilities regarding project plan adjustment.

Table 12. Plan adjustment decisions

Product and decision type:	Responsible body:	CB	EC	PC	WPL	AC	PS
Plan adjustment							
WP mid-cycle milestones & deliverables			D		P		A
WP mid-cycle budget			D		P		A
Consortium Agreement* and membership		D	P				A

Responsible bodies:

CB = Consortium Board

EC = Executive Committee

PC = Project Co-ordinator

WPL = Work Package Leader

AC = Advisory Committee

PS = Project Secretariat

Decision types:

P = propose

A = advise

D = decide (approve/reject)

* Note: An adjustment to the Consortium Agreement has to be endorsed by the authorized representative of each partner.

6. External project review procedure

A periodic project reviews will be organised by the Commission at the end of the reporting period (month 12) to assess the work carried out. The review may be carried out by the Commission services alone, or by the Commission services with the support of external experts appointed by the Commission.

The review will principally assess:

- the degree of fulfilment of the project work plan for the period
- the degree of fulfilment of the deliverables
- the necessity of the resources that the contractors have employed
- the management aspects of the project
- the likelihood to achieve the results aimed at by the project
- the planning of the next 18 months period
- the plan for using and disseminating the knowledge.

The review will be based only on the written material submitted by the project (reports and deliverables, see previous section), and may additionally involve a “hearing and/or review procedure” with project representatives. The exact timing for performing the review, including any hearing, will be fixed in such a way that the 45 days deadline for report submission, and the 45 days for approval by the Commission can be met (thus latest 90 days after the end of the reporting period, in our case early February 2006).

The outcome of the review will be communicated in writing to the project coordinator. This may include technical recommendations to be taken into account in the project’s planning for the work of the next period, and may recommend amendments to the draft 18-month plan, in which case the consortium will present an amended plan to be appended to Annex I (DoW) to the contract. In all cases the final version of the updated Annex I, including the updated Contract Preparation Forms, will be attached to a formal amendment request submitted by the co-ordinator to the Commission.

7. Overall project evaluation and impact assessment methodology

The goals of TENCompetence validation

The validation strategy as described in general terms in the DOW focuses on three aspects:

1. ***Evaluation of the effectiveness of project outcomes***. This will be carried out as part of the activity cluster ‘Pilots with & Validation of the Integrated System’. Highly focused pilots in a challenging and authentic environment are the major instruments in this strategy.
2. ***Validation of the technical performance of project outcomes***. This consists of ensuring that the technical systems produced by the project conform to their requirements set out in verifiable form in the specifications documents, and that the component parts of the system interoperate as planned.
3. ***(Self-)assessment of project processes and documentation***. This consists of ensuring that optimum processes are in place for the development of project documentation and deliverables, and that all project deliverables and services are evaluated and optimised.

In this chapter we cover the first two aspects; the third aspect is covered in chapters 2, 3, 4 and 5 of this Handbook. Chapter 8 contains guidelines for software development and integration - including testing - to be used *within* Work Packages.

Effectiveness evaluation of the project outcomes

The whole TENCompetence project may be seen as a research undertaking which

- Identifies a set of problems in Life Long Learning
- Hypothesises that a specific set of functionalities will improve the provision of Life Long Learning
- Develops an infrastructure to deliver those functionalities
- Runs pilots to test the hypothesis.

The domain which TENCompetence addresses is defined by the seven problems and their solutions as described in the DOW:

- 1) new, promising, innovative pedagogical approaches for lifelong competence development, supported by the TENCompetence infrastructure.
- 2) tools to support individuals, groups and organisations in Europe to find the best solution for their formal or informal learning problem.
- 3) policies and software agents that support the pro-active sharing of knowledge and learning resources.

- 4) models and software tools to assess the competences of individuals, groups and organisations in an exchangeable way.
- 5) software for the effective and efficient support of users who create, store, use and exchange knowledge resources, learning activities, units of learning, competence development programmes and networks for lifelong competence development.
- 6) software solutions to establish a decentralized, self-organized and empowered management model when using the TENCompetence infrastructure.
- 7) Integration of isolated tools that are available in the field.

Effectiveness evaluation therefore should focus on these problems and their solutions as provided by the project. The major vehicle for effectiveness evaluation is the use of intensive pilots throughout the three project cycles, and the incorporation of associate partners as the project progresses. The following table shows how the seven solutions, translated into criteria and indicators:

Table 13. Seven solutions TENCompetence translated into criteria and indicators

<i>Aspect</i>	<i>Criteria</i>	<i>Indicator</i>
Aspect 1: In what ways, and to what extent, does the TENCompetence project provide innovative approaches for life long learning?	Enables LLL to be provided in new ways in new contexts	- Associate partners and learners report that pilots provide opportunities for life long which were previously difficult or impossible to obtain.
	Provision of LLL that links individuals, groups and organisations.	- Pilots are observed to generate information flows across these groups
	Provision of ways of providing providing life long learning which have not previously used and/or proposed	- LLL actors in pilots (learners, teachers, Human Resource responsables, learning providers) identify innovation. - Peer reviewed publication of results innovative case studies.
Aspect 2: Does the TENCompetence infrastructure and methodologies enable users to find the best solutions for their formal and informal problems? In what ways, and to what extent?	Use of the system to identify learning solutions	- Pilots show that users are active in identifying solutions on the system. - Learning solutions identified are adopted.
	Good fit of learner and learning solution	- Learning solutions adopted are completed - LLL actors in pilots identify good fit.
	Improved learning outcomes	- Improved performance of individuals groups and organisations involved in pilots, as assessed by selves, peers and managers.

Aspect 3: Do the TENCompetence infrastructure and methodologies provide policies and software agents that support the pro-active sharing of knowledge and learning resources? In what ways and to what extent?	Knowledge and learning resources are shared.	- Pilots demonstrate sharing activity
	Policies and software agents increase sharing of knowledge and sharing resources when used.	- Correlation between use of policies and software agents, and levels of sharing.
Aspect 4: Do the TENCompetence models and software tools enable users to assess the competences of individuals, groups and organisations in an exchangeable way? In what ways and to what extent?	Competence definitions, assessments, and the results of assessments can be exchanged by the different parts of the system	- Competence definitions, assessments and results are exchanged smoothly between the parts of the system in pilots.
	Competence definitions can be exchanged with users outside the system	- Pilot implementations are able to export competence definitions in a way which can be read by other systems.
	Competence definitions and assessments provided are effective basis for identifying LLL opportunities	- Evaluation with pilot learners indicates that competence assessments led to appropriate learning opportunities and programmes for individuals, groups and organisations.
Aspect 5: Do the TENCompetence infrastructure and methodologies provide users with an effective and integrated system which enables them to create, store, use and exchange knowledge resources, learning activities, units of learning, competence development programmes and networks for lifelong competence development? In what ways and to what extent?	Users with understanding of the task to be undertaken and basic IT skills can use the system without the need for external support.	- In pilots users with basic IT skills are observed use the system without the need for external support. - In pilots use of online help is evenly distributed with regard to functionality
	The system can be used to deliver effective LLL.	- Pilots are completed and their objectives achieved. - Participants do not stop their activity because they cannot use the system. - Evaluation responses from users report that functionality provided is effective
Aspect 6 Do the TENCompetence infrastructure and methodologies enable users to establish a decentralized, self-organized and empowered management model? : In which ways and to what extent?	Use is made of the system to manage LLL by individuals and groups.	- Individuals and groups are observed to use system functionality to manage their own LLL.
	Individuals and groups find the functionality offered effective in managing their own LLL	- Individuals and groups report that they have been empowered. - Change in LLL management practice identified in pilots at individual, group and organisational levels

Aspect 7: To what extent are the TENCompetence infrastructure and methodologies successful in integrating the isolated tools that are available in the field	An integrated solution is released	<ul style="list-style-type: none"> - Technical staff report that the TENCompetence system release used in pilots provides all the functionality required - Qualified server administrator can install and operate the software without external support
	Pilot participants experience the system as a single integrated environment	<ul style="list-style-type: none"> - In pilots users do not have to launch a number of applications to complete their tasks - Users in pilots perceive that they are using a single system

How this will be applied to the pilots is described in the next section.

The role of the pilots in the three cycles of project activity

The TENCompetence project is carrying out research into the best way to address the real life solutions to the challenges presented by Life Long Learning. Consequently the only way of validating the effectiveness outcomes of the project is to use them in authentic and challenging Life Long Learning tasks. This means, firstly, identifying scenarios in which users can use the system while they are carrying out their normal job and training activities, rather than setting up trials, which are attended by users out of their work context. Secondly the pilots must be challenging, in the sense that they should resolve the Life Long Learning need which is at present not adequately addressed.

The pilots are scheduled around the three project cycles, with a planning stage at the start of each cycle and a report produced at the end. Additional pilots run by Associate Partners may be organised at any point, and it is anticipated that this activity will increase as the project progresses.

Cycle 1.

In Cycle 1 (inception/elaboration) the existing tools and services are integrated to create the first release of the TENCompetence system. The Cycle 1 pilots will use this system in an authentic and challenging context to provide 'proof of concept'. The work carried out in the Aspect RTD activities in this cycle concentrates on the research and development of new models and technologies, and will be validated in Cycle 2 pilots.

Cycle 2.

In the second cycle (construction) the evaluation results of the pilots and the output of the first cycle of Aspect RTD activities will be taken as the input for the integration activities and the infrastructure will be redesigned and extended to accommodate these new results. In this second cycle the main integrated technology development activities will take place to construct the infrastructure.

The pilots programme will be greatly expanded in this cycle, with major planned pilots in professional development in medicine, water management in Indonesia, and

integrated competence management in the city of Antwerp, as well as in digital cinema. This second cycle of pilots has the character of 'usability pilots', and they will validate that the solutions developed to make the TENCompetence concept a reality are usable, that is to say that they provide effective solutions to real problems in an authentic context.

Cycle 3.

In the third cycle (transition) the evaluation results of the pilots and the output of the Aspect RTD activities in Cycle 2 are again taken into account as input for this last cycle. The pilots are now 'business models demonstrators', mainly involving external parties to increase the sustainability.

In this cycle the project pilots will demonstrate to the wider Life Long Learning community the advantages of adopting the TENCompetence concept and the infrastructure which will have been developed to support it. Consequently it is intended to leverage and extend the successful pilots which have been established in Cycle 2. The role of associate partners is also expected to be a key factor in organising pilots in Cycle 3.

The pilots and their evaluation strategy are described in detail in deliverable D4.1 'Pilot Evaluation Plan', <http://dspace.ou.nl/handle/1820/684>

Technical testing and validation

Technical testing and validation of the integrated system is carried out in separate actions from the project pilots.

Unit and integration testing

It is essential that the system perform satisfactorily from a technical point of view before it is piloted with users in order to establish its effectiveness for the purposes for which it was designed. The role of technical testing in the project is to ensure the quality and coherence of the integrated system, and in particular the code produced by the project in the context of WP 3. The technical testing focuses on unit testing and integration testing.

The purpose of **unit testing** is to verify that each individual component functions according to the technical specifications. The procedure searches for defects in, and verifies the functioning of, software elements (e.g. modules, programs, objects, classes, etc.) that are separately testable. This may be done in isolation from the rest of the system, depending on the context of the development life cycle and the system. Unit testing typically done by the programmer and not by testers, as it requires detailed knowledge of the internal program design and code.

The purpose of **integration testing**, on the other hand, is to verify that the interaction between the various units which make up the integrated system is satisfactory. According to the plan the *implementer* delivers a component which has been successfully unit tested to the *integrator*. The *integrator* merges this component into

intermediate builds. Step by step, bottom-up, each component will be integrated according to the build plan. After each step, the intermediate build is submitted to the integration test. This procedure ensures that each component added is compatible with the components which have already been integrated.

The details for unit and integration testing are covered in the ‘Unit and Integration Test plan’ (detailed in appendix 3 of deliverable D4.1 ‘Pilot Evaluation Plan’, <http://dspace.ou.nl/handle/1820/684>).

Technical evaluation of the integrated system

Once the integrated system has passed through unit and integration testing it becomes a candidate for a release. Each full release of the system (as opposed to minor updates with bug fixes) undergoes a technical evaluation process.

It is important to establish the performance of the system⁸ in the widest possible range of technical environments through a systematic programme of tests. This information is important for setting up pilots with associate partners, as well as for wider dissemination purposes. This aspect is addressed by the ‘Technical evaluation of the integrated system’ (detailed in appendix 4 of deliverable D4.1 ‘Pilot Evaluation Plan’, <http://dspace.ou.nl/handle/1820/684>).

⁸ With the exception of the first (Antelope) version, as this will only run with a small number of data sets and users.

8. Software development standards and procedures

SourceForge and Bugzilla

Background

Development work in the TENCompetence project requires the use of a server-based solution in order to track bugs, feature requests, and enhancements. The project is already using SourceForge (<http://sourceforge.net/projects/tencompetence>) as a CVS repository and point of presence. The SourceForge hosting facility includes a bug tracking service. However, the TENCompetence project requires a more substantial and fully featured bug tracking service which we propose to be an externally hosted Bugzilla Service.

Bugzilla

Bugzilla is a “defect tracking” system that can track bugs, feature requests, and patches. In spite of its moniker, it used for more than just bug tracking. An example of a “bug” could be an item that tracks the progress of a feature request or enhancement. Bugzilla is free and customisable but has to be set up and hosted on a server. Bugzilla is used by many “big name” projects including the Mozilla Project, Eclipse, and NASA.

Features included can be found at <http://www.bugzilla.org/features/>

Bugzilla relies on an installed web server, such as Apache and a Database Management System, such as MySQL. Bugs and issues can be submitted by anybody, and will be assigned to a particular developer after a process of triage. Various status updates for each bug are allowed, together with user notes and bug examples.

An example of a Bugzilla bug report page can be found here -

https://bugs.eclipse.org/bugs/show_bug.cgi?id=64672

Release notes such as those for Bugzilla 2.20.1 indicate the exact set of dependencies, which include:

- A compatible database server (often a version of MySQL)
- A suitable release of Perl 5
- An assortment of Perl modules
- A compatible web server such as Apache (though any web server that supports CGI can work)
- A suitable mail transfer agent such as Sendmail, Qmail, Postfix, or Exam.

So, in order to set this up the project would require an available server space and a developer to install, configure and maintain the system. The documentation states that it is easier to install on a Unix-based system than on a Windows system, though it can be achieved (see <http://www.bugzilla.org/docs/2.20/html/os-specific.html>)

An alternative is to pay for a Bugzilla hosting service such as that found at <http://wush.net/bugzilla.php>. This works out at about \$15 per month. This will be the preferred option.

Eclipse integration with Bugzilla provided by Mylar

Mylar is an Eclipse Technology Project that consists of a plug-in for the Eclipse platform. This plug-in provides many useful functions allowing the Eclipse user to narrow in and focus on specific development tasks:

- Provides a task focused UI for Eclipse that makes working with very large workspaces as easy as working with small ones.
- Supports task management and monitors your work activity to identify information relevant to the task-at-hand.
- Provides integrated Bugzilla support.

Perhaps the best way to get a sense of what Mylar offers is to watch the Camtasia demos here:

<http://www.eclipse.org/mylar/doc/demo/mylar-demo-04.html>

<http://www.eclipse.org/mylar/doc/demo/mylar-demo-04-reports.html>

Mylar is potentially of use to the TENCompetence development team apart from the Bugzilla integration features because of its task-oriented functionality. With Mylar, a developer can manage personal task and planning, and manage views and editors.

Bugzilla integration in Eclipse provided by Mylar means that a developer can link tasks to bugs and feature requests. The developer can quickly file a bug report from within Eclipse, query the bug base from within Eclipse and the resulting bugs and feature requests can be allocated as tasks to individual developers.

References and Links:

SourceForge Home Page - <http://sourceforge.net>

TENCompetence SourceForge - <http://sourceforge.net/projects/tencompetence>

Bugzilla - <http://www.bugzilla.org>

Mylar - <http://www.eclipse.org/mylar>

Unit Testing and JUnit

Background

Testing of the TENCompetence system software components is an integral part of the development process

JUnit is a unit testing framework written for the developer who implements unit tests in Java. It is free, Open Source and is easily integrated into the developer's test environment as either a standalone library or in tight integration with the Eclipse IDE.

The JUnit.jar library jar is linked to the code containing the tests' classpath. JUnit is currently at version 4.1, the previous version being 3.8.1 which is the default version used by Eclipse 3.2. JUnit ships with many useful features:

- A variety of front-ends that can display the results of the tests – command line, AWT, Swing
- Separate class loaders for each unit test
- Integration with popular tools like Ant, Maven and Eclipse.

Integration with Eclipse

The development IDE and target platform chosen for the project is Eclipse. The Eclipse IDE workbench ships with a built in JUnit testing framework that is fully integrated into the Java development process. This makes it extremely convenient for developers to test their code instantly with graphical feedback in the JUnit View or by running test suites.

Eclipse contains the following JUnit integration features:

- JUnit View provides instant feedback of passing and failing tests
- Ability to run single or multiple tests from Eclipse
- JUnit functions available from contextual menus.

JUnit 3 or JUnit 4?

The default version of JUnit used in Eclipse is version 3.8.1. This version works with Java versions 1.4.x and above. The current version of JUnit is 4.1 but depends on features provided by Java version 5.0 and above in order to compile and run.

The main features of JUnit 4 are

- It Requires JDK 5 to run
- Test classes do not have to extend from junit.framework.TestCase
- Test methods do not have to be prefixed with 'test'
- There is no difference between the old assert methods and the new assert methods
- Use @Test annotations to mark a method as a test case

- `@Before` and `@After` annotations take care of set up and tear down
- `@BeforeClass` and `@AfterClass` annotations take care of one time set up and one time tear down
- `@Test` annotations can take a parameter for timeout. Test fails if the test takes more time to execute
- `@Test` annotations can take a parameter that declares the type of exception to be thrown
- `JUnit4Adapter` enables running the new JUnit4 tests using the old JUnit runners
- Old JUnit tests can be run in the new JUnit4 runner.

It is yet to be decided whether to use this latest version of JUnit, and it would seem logical to do so since code is being written against Java version 5.0. However, it should be noted that **the current version of Ant (1.6.5), which is used for running JUnit automation scripts and reporting, does not work with JUnit 4.** Therefore, if the project wishes to use Ant for automating JUnit tests and reporting it will either have to wait until Ant fully supports JUnit 4, or continue to use JUnit 3.8.1.

JUnit TENCompetence source folder structure and naming conventions

JUnit tests should be located in a separate module that maintains a mirror package structure of the original module that is being tested. This ensures that the main module does not have to link to or contain the JUnit.jar library, and also so that users can check out the code from CVS without having to get the tests as well, these remaining optional in the corresponding JUnit tests module. This is the practice that the Eclipse development team use themselves.

For example, a plug-in module called `org.tencompetence.tencc` would have its tests in a separate module called `org.tencompetence.tencc.tests` and the package structure would be as follows:

The original module:

```
org.tencompetence.tencc
  |----src
    |----org.tencompetence.tencc
      |----package1
        |----StringUtils.java
      |----package2
      |----package3
```

And the JUnit test module structure:

```
org.tencompetence.tencc.tests
  |----src
    |----org.tencompetence.tencc
      |----package1
        |----StringUtilsTest.java
      |----package2
      |----package3
```

File names of tests should follow the naming convention of “Class to be tested” followed by the suffix “Test”. For example:

Class to be tested:	StringUtils
Name of test class:	StringUtilsTest

Automated and Manual Testing frameworks

A developer can run single or multiple tests from within Eclipse “on the fly” as needed. Additionally, we will be running nightly builds and automated scripts that will run all JUnit tests and report the results as HTML files. We have already created an Ant script to perform the task of running automated JUnit tests. However, as noted above, the current version of Ant is only able to invoke JUnit tasks for JUnit version 3.8.1, not version 4.

Guidance for writing tests

Providing best practice guidance for writing JUnit tests is beyond the scope of this document. However, developers are recommended to study the material available here:

JUnit Home Page - <http://www.junit.org>

JUnit Tutorial - <http://clarkware.com/articles/JUnitPrimer.html>

The New Features of JUnit 4 - <http://www.devx.com/Java/Article/31983>

Massol, V. with Husted, T. (2004). JUnit in Action. Manning Publications Co. USA

The Use of a CVS Repository in the TENCompetence Project

Background

The TENCompetence development process works in an open manner, utilising an Open Source licensing model and attempting to maintain transparency at all levels. One means to accomplish this transparency is to ensure that all code, documentation and development artefacts are available in a **Concurrent Versions System (CVS) Repository**.

CVS and SourceForge

The TENCompetence project uses SourceForge for its CVS repository. This is a free service that provides hosting for Open Source projects and provides additional services such as issue tracking, forums, project presence, CVS and Subversion Repositories.

Access rights and user permissions

We have allocated key project staff as CVS administrators and assigned key developers with CVS developer status. By default, CVS access is read only, and this includes public anonymous access. However, developers are granted write access on a per-module basis

In order for a developer to be granted write access to files in a module or modules, the following steps have to be taken:

1. The aspiring developer has to register at SourceForge with a user name.
2. A SourceForge administrator has to add the user as a developer to the SourceForge project admin page.
3. A SourceForge administrator has to edit the `avail` file to include the user name and module name(s) required for read/write access.

As of the time of writing of this document, the `avail` file is as follows:

```
unavail
avail|hubertvogten,harriemartens,rulem,phillipus,scottwilson
avail|arnewolf2|org.tencompetence.tencc.questionauthoring
avail|dicerto|org.tencompetence.tencc
avail|dicerto|org.tencompetence.tencc.coordination
avail|dicerto|org.tencompetence.tencc.policy
avail|ps3com|wp6/org.tencompetence.imsmodel
avail|ps3com|wp6/org.tencompetence.imsmodel.tests
avail|ps3com|wp6/org.tencompetence.ldauthor
avail|ps3com,pzervas,mloizos,pythk|wp6/org.tencompetence.ldauthor.transform
avail|ps3com|wp6/org.tencompetence.xstream
```

The first line, “unavail”, ensures that by default all developers do not have write access to the repository. The line “avail” is followed by user name(s) and the name of the module that is given write access. Full documentation for editing the avail file is available at the SourceForge website.

Integration with Eclipse

The development IDE and target platform chosen for the project is Eclipse. The Eclipse IDE workbench ships with a built in CVS client which is fully integrated into the Java development process. This makes it extremely convenient for developers to synchronize their workspace with the CVS repository and to monitor history, annotations and file comparisons within the IDE. Thus, no other CVS client is needed to perform all common CVS tasks.

Repository details

The SourceForge CVS connection details are as follows:

Web Page:	http://sourceforge.net/cvs/?group_id=159487
Host location:	tencompetence.cvs.sourceforge.net
Repository Path:	/cvsroot/tencompetence
Connection type:	extssh
Public User Name:	anonymous

Modules and folder structure

All code is deposited under the “HEAD” CVS location. Work Package 3 code is placed at the top level under HEAD, while other Work Package code is placed in appropriately named subfolders (“wp6”, “wp7”, “wp8”, and so on).

The naming convention for a code module for the TenCompetence client application is the project namespace (`org.tencompetence.tencc`) followed by the module name, as follows:

```
org.tencompetence.tencc.modulename
```

Where “modulename” is an appropriate name of the module.

As an example, if one had a module named "org.tencompetence.myproject" that came under the remit of WP5 then a developer using Eclipse would perform the following steps:

1. Right-click on the Project in the Package Explorer in Eclipse
2. Choose Team->Share Project...
3. Select the 10Competence Repository
4. Choose "Use specified module name" which would be
`wp5/org.tencompetence.myproject`

JUnit folder structure

JUnit tests relating to a module should be organised in a separate module that maintains a mirror package structure of the original module. This ensures that the main module does not have to link to or contain the JUnit.jar library, and also so that users can check out the code without having to get the tests as well, these remaining optional in the corresponding JUnit tests module. This is the practice that the Eclipse development team use themselves.

For example, a plug-in module called `org.tencompetence.tencc` would have its tests in a separate module called `org.tencompetence.tencc.tests` and the package structure would be as follows:

The original module:

```
org.tencompetence.tencc
|----src
|----org.tencompetence.tencc
|----package1
|----package2
|----package3
```

And the JUnit test module structure:

```
org.tencompetence.tencc.tests
|----src
|----org.tencompetence.tencc
|----package1
|----package2
|----package3
```

For more information on JUnit and package structures, see the section on Unit testing.

Other considerations when using folders

The "bin" folder and any folders that contain compiled code files (*.class files for Java) should be excluded from the CVS, as it should only contain source files and binary library files.

3rd party libraries can be uploaded to CVS (together with their licences) provided that a version number is used as part of their file name or set in the Eclipse manifest if it is delivered as a "wrapped" Eclipse plug-in. Versions of popular libraries can vary enormously. JDOM 1.0 is *very* different to JDOM 0.9. The same goes for Tomcat - some things work fine for 5.5, but not for version 5.0. Name the file `jdom-1.0.jar`, not `jdom.jar`.

Note that modules can not be deleted permanently by users. For this to happen a formal request has to be submitted to the SourceForge technical support team by a project administrator. This process can take a few days to complete.

House Rules

In order to maintain good housekeeping for using the CVS repository in the project the following rules have been put in place:

1. Committed code should at all times be able to be compiled.
2. Committed code should at all times be able to be run.
3. Developers should check out the latest code from CVS before committing their own changes
4. Any code conflicts should be resolved between developers preferably informally, but failing that via a project co-ordinator
5. Folders (a.k.a "modules") should **not** be created under CVSROOT but only directly under HEAD. See above for more details as to folder structure.
6. Module naming conventions should be followed, see above
7. JUnit tests should be given their own modules, see above.
8. Compiled class files and the compiled output folder (usually "bin") should not be added to the CVS. Only source files and libraries can be uploaded.
9. Library files should be named according to their version and/or the Eclipse manifest version number set accordingly.

Component Owners

Each component or other coherent piece of software in the TENCompetence system will be assigned a Component Owner. The Component Owner is the organisation, university, or company that created it. When a component requires changes from a non-Component Owner, any changes should be discussed with the Component Owner first. The Component Owner decides how to handle changes. For example, a developer working on a QTI tool that requires changes to the central data API, would have to contact Harrie Martens or Hubert Vogten. Depending on the required changes Harrie or Hubert could either

- update the central data API, or
- allow the developer to make the changes herself, or
- allow the developer to submit a patch file

References and Links:

CVS Home Page - <http://www.nongnu.org/cvs/>

SourceForge Home Page - <http://sourceforge.net>

SourceForge and CVS - <http://sourceforge.net/docs/E04/>

Open Source Development with CVS 2nd Edition, Fogel and Bar - <http://cvsbook.red-bean.com/cvsbook.html>

TenCompetence SourceForge - <http://sourceforge.net/projects/tencompetence>

Eclipse – <http://www.eclipse.org>

Coding conventions and guidelines

Note - the following are modelled on the Eclipse / Sun coding conventions and guidelines.

Naming conventions

Classes and Interfaces

Class names should be nouns, in mixed case with the first letter of each internal word capitalized. Try to keep the class names simple and descriptive. Use whole words - avoid acronyms and abbreviations (unless the abbreviation is much more widely used than the long form, such as URL or HTML).

Examples:

- `class Raster;`
- `class ImageSprite;`

Interface names should be capitalized like class names. For interface names, we follow the "I"-for-interface convention: all interface names are prefixed with an "I". For example, "IWorkspace" or "IIndex". This convention aids code readability by making interface names more readily recognizable.

Additional rules:

The names of exception classes (subclasses of Exception) should follow the common practice of ending in "Exception".

Methods

Methods should be verbs, in mixed case with the first letter lowercase, with the first letter of each internal word capitalized.

Examples:

- `run();`
- `runFast();`
- `getBackground();`

Additional rules:

The naming of methods should follow common practice for naming getters (`getX()`), setters (`setX()`), and predicates (`isX()`, `hasX()`).

Variables

Instance, static, and class constants are in mixed case with a lowercase first letter. Internal words start with capital letters. Variable names should not start with dollar sign \$ characters. Variable names should be short yet meaningful. The choice of a variable name should be mnemonic - that is, designed to indicate to the casual observer the intent of its use. One-character variable names should be avoided except for temporary "throwaway" variables. Common names for temporary variables are i, j, k, m, and n for integers; c, d, and e for characters.

Examples:

- `int i;`
- `char c;`
- `float myWidth;`

Constants

The names of class constants and of ANSI constants should be all uppercase with words separated by underscores ("_").

Examples:

- `static final int MIN_WIDTH = 4;`
- `static final int MAX_WIDTH = 999;`
- `static final int GET_THE_CPU = 1;`

Plug-ins and Extension Points

All plug-ins (and plug-in fragments) must have unique identifiers following the same style of naming convention as Java packages. For example, the Eclipse workbench plug-in is named `org.eclipse.ui`.

Extension points that expect multiple extensions should have plural names. For example, "builders" rather than "builder".

Further guidelines as specified by Sun can be found here - <http://java.sun.com/docs/codeconv/CodeConventions.pdf>

Conventions for plug-ins

Shared code should be packaged into plug-ins so that each plug-in offers a unique set of functionality allowing for optimal re-use. A proposed **core** plug-in structure is as follows:

1. TENCompetence Client Host
2. General Utilities Plug-in
3. GUI Utilities Plug-in
4. Persistence / File handling Plug-in
5. XML / JDOM Utilities Plug-in
6. Conduits/Co-ordination Framework Plug-in.

Dependencies between UI components and business logic components should be eliminated. For example, utility type plug-ins should be factored such that they do not depend on or reference UI components. A Model-View-Controller architecture and Event Model facilitates this pattern. References to Java Interfaces rather than classes also contributes to the plug-in architecture.

Care should be taken when exporting packages and classes. Public packages and classes can be declared in the plug-in manifest and these should usually be limited to public API packages. All other implementations can be made private. This requires the separation of the API from internal implementation. By exporting only API classes one can limit the visibility to downstream clients. Negotiation between plug-in developers may be necessary in order to determine the right balance between public and private dependencies, and developers should reveal only the API in which they have confidence, but be prepared to reveal more of the API as clients ask for it.

Re-usable third-party libraries should be wrapped in plug-in manifests for deployment as Eclipse plug-ins and named accordingly. For example, the XML library JDOM 1.0 could be packaged with a manifest file and made available as `org.jdom.plugin` version 1.0.0. Plug-ins that depend on a particular third-party library may instead ship the required library file(s) in the packaged plug-in.

JUnit tests for each component should be provided in their own plug-in modules – see section 0 for details.

Extension Points

Developers are encouraged to provide and declare Extension Points in their plug-in manifests. This Eclipse mechanism provides an XML-based schema approach for declaring the mechanics of a plug-in's extension point(s).

Extension points define new function points for the platform or application that other plug-ins can plug into. An Extension Point is declared in XML-based schema file that defines the grammar that formally expresses elements, attributes, and types that it relates to. This information can be used by plug-ins to validate extensions. An example of an Eclipse Extension Point is a Menu Item, a View, or an Editor.

Existing Extension Points in the TENCompetence Client are:

```
org.tencompetence.tencc.datamodelfactory
org.tencompetence.tencc.datamodelproviders
org.tencompetence.tencc.conduits.sharing_conduits
org.tencompetence.tencc.coordination.providers
```

These are described in more detail elsewhere in this document.

Future plug-in contributors may then leverage these extension points to provide further functionality to the client.

Naming conventions

TENCompetence plug-ins must follow the naming convention of project namespace followed by module name. The project namespace is `org.tencompetence`. TENCompetence client plug-ins have the “tencc” name appended to the namespace. For example, a new plug-in for the client application would be named `org.tencompetence.tencc.myplugin`.

The convention for naming private and public API packages is for every plug-in intended for external use to have two name spaces. If a package name contains `internal`, the classes it contains are not intended to be used publicly.

User Interface Guidelines

Again, these are modelled after the Eclipse User Interface Guidelines and being too lengthy to reprint here can be found at:

http://wiki.eclipse.org/index.php/User_Interface_Guidelines

Runtime, Platforms, Tools and Libraries

Java 5 (v1.5.0) is to be used for the SDK and target runtime. Java 6 (v1.6.0) may be used at a later date once it has been firmly proved and established on all platforms (Windows, Linux, Mac).

The following table lists the **Eclipse tools and libraries** that are to be used in the project. Note that versions may be changed as updates are made available:

Table 14. Eclipse tool and libraries used in the project

Name	Version	Description	Link
Eclipse Platform	3.2.1	Eclipse SDK and RCP	www.eclipse.org
Eclipse Web Tools	1.5.1	Eclipse Web Tools plug-ins	www.eclipse.org/wtp/
GEF	3.2.1	Graphical Editing Framework	www.eclipse.org/gef/
EMF	2.2.1	Eclipse Modelling Framework	www.eclipse.org/emf/

The following table lists the **third-party tools and libraries** that are currently used in the project. Note that versions may be changed as updates are made available:

Table 15. Third-party tools and libraries used in the project

Name	Version	Description	Link
JDOM	1.0	Java XML Modelling library	www.jdom.org
Commons Logging	1.5.1	Logging services	http://jakarta.apache.org/commons/logging/
Commons Codec	3.2.1	Implementations of common encoders and decoders such as Base64, Hex, Phonetic and URLs	http://jakarta.apache.org/commons/codec/
Commons HTTP	2.2.1	Client side of the most recent HTTP standards and recommendations	http://jakarta.apache.org/commons/httpclient/
Google API	latest	Google search client	http://code.google.com/
Skype API	latest	Skype client	https://developer.skype.com/
Blowfish library	2.1.4	Encryption services	http://www.schneier.com/blowfish.html
Rome	0.8	RSS and Atom Feed parsers	https://rome.dev.java.net/

Additional libraries and tools will be specified and recommended as the project proceeds.

Licensing

The Technical Standards and Architecture sub-committee has selected the BSD software license (New-BSD, 3-clause; <http://www.opensource.org/licenses/bsd-license.php>) as the compulsory TENCompetence software license.

References

- [1] Kruchten, P. (1995), *The 4+1 View Model of Architecture*, <http://doi.ieeecomputersociety.org/10.1109/52.469759>
- [2] Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). *Design Patterns, Elements of Reusable Object-Oriented Software*. Addison-Wesley.
- [3] Koper, R. (07-06-2006), *TENCompetence Domain Model*, <http://dspace.ou.nl/handle/1820/649>
- [4] OSGi framework, <http://en.wikipedia.org/wiki/OSGi>
- [5] *Eclipse – an open development platform*, <http://www.eclipse.org/>

- [6] *Personal Learning Environment Download Page*,
<http://reload.ces.strath.ac.uk/plex/>
- [7] *Hecate - a rich client application enabling the end-user to actively participate in a learning network*, <http://sourceforge.net/projects/hecate/>
- [8] Martens, H., Vogten, H. & Koper, R. (25-01-2006), *Hecate*,
<http://hdl.handle.net/1820/565>
- [9] *The IMS Learning Design Engine*, <http://coppercore.org/>
- [10] *RELOAD – Reusable eLearning Object Authoring & Delivery*,
<http://www.reload.ac.uk>
- [11] *phpBB.com :: Creating Communities*, <http://www.phpbb.com/>
- [12] *Development SmartFeed*,
<http://www.phpbb.com/phpBB/viewtopic.php?t=344050&highlight=smartfeed>
- [13] *Discussions SmartFeed*,
<http://www.phpbb.com/phpBB/viewtopic.php?t=352785&highlight=smartfeed>

9. Risk management

Risk management in TENCompetence takes place at three levels:

1. At the strategic level. Risk management here concentrates on the relation between the project and the consortium with its environment. Risk management at this level is the responsibility of the Consortium Board.
2. At the tactical level. Risk management here concentrates on the Work Packages' contribution to the project objective. Risk management at this level is the responsibility of the Executive Committee.
3. At the operational level. Risk management here concentrates on the activities within the work packages, which is the responsibility of the work package leader.

During the first twelve months of the project, an elaborate risk analysis and contingency planning procedure was implemented. Looking back, this was overdone. The following procedures and tools proved sufficient for risk management:

- Monitoring through the three-monthly Activity Reports (see chapter 2).
- The TENCompetence assessment protocol to quantify partners' scientific outputs (chapter 3)
- The quality assurance procedure for internal deliverables and deliverables (chapters 4 and 5)
- The software development procedures and standards (chapter 8)
- The project workshops, on average every three months, with all project partners somewhere in Europe.
- The Moodle on-line project environment where all WPs discuss and present their day-to-day activities.

10. Gender mainstreaming - Gender Project Action Plan

1. Introduction

1.1. TENCompetence is committed to Gender Mainstreaming – proactively incorporating gender awareness and actions into Project activities. TENCompetence’s Gender Action Plan, in compliance with EU Framework Programme 6 requirements, has two aspects:

(i) to analyse, monitor and promote the participation of women (as researchers, project staff, case-study and pilot participants, etc) within the Project. This part of the Gender Action Plan is called the **Gender Project Action Plan**, which was covered in an earlier version of this document.

(ii) to ensure that all research content and product development content within the Project take appropriate account of gender issues. This part of TENCompetence’s Gender Action Plan is covered in Internal Deliverable 2.3 ‘Gender Mainstreaming Requirements Action Plan’.

In the covering letter to the first review report (ref. D (2007) / E 3 / HJW), the Project Officer Mr. Westhoff indicated that the efforts and funds dedicated to gender mainstreaming in general, can better be dedicated to dissemination activities. As such, TENCompetence will *monitor and promote* gender issues within the project in a manner as indicated below, but will not take into account gender-specific requirements in the system design.

1.2. This TENCompetence **Gender Project Action Plan (GPAP)**, for monitoring and encouraging the participation of women, has been developed in accordance with:

- the overarching EU goal to increase the volume of research and development generally – for which more women are needed to increase the pool of available expertise – and the EU equality aim to improve the participation of women in scientific, technical and industrial research and development. Relevant information about these EU policies and related Commission data can be found at <http://cordis.europa.eu/improving/women/documents.htm> and at http://ec.europa.eu/research/science-society/women/wir/index_en.html
- the concept of Gender Mainstreaming as an integral, shared, cross-project responsibility to build gender-sensitive strategies and actions into Project activity.
- the principles of gender mainstreaming as set out in EU documentation such as the EU Commission Science and Society Research Directorate’s reports: ‘*Vademecum*’ *Gender Mainstreaming in the 6th Framework Programme – Reference Guide for Scientific Officers/Project Officers*, March 2003; and *Gender Action Plans – A Compendium of Good Practices*, December 2005.

- alignment with other reporting and administration cycles of TENCompetence (for administrative ease and more effective mainstreaming).
- TENCompetence's commitment to gender actions as stated in the Project Proposal.

1.3. Further Guidance. Further information and advice about implementing the GPAP are contained in the **GPAP Guidance Notes which** will be circulated to Contractors and Work Package leaders and posted on the TENCompetence website.

1.4. GPAP dissemination and awareness raising. There is a Gender Action Plan (GAP) forum on the TENCompetence Partners' website. Guidance and practical advice, ideas and concrete examples of Gender Actions will be posted and shared on this Forum. Opportunities to include workshops and seminars on the Gender Project Action Plan, and to disseminate good practice, at TENCompetence conferences and meetings will be sought. Calls for conference papers will include calls for papers related to gender participation issues.

1.5. There are two key elements of the Gender Project Action Plan – *Gender Actions*; and *Monitoring Progress and Reporting*.

The GPAP thus covers:

- Gender Actions to be taken to mainstream gender awareness and to promote women's participation in the project's activities (section 2. below)
- The recording and reporting of Gender Actions so that monitoring and evaluation can inform further Actions and progress and improvement can be identified (section 3 below).

The Gender Project Action Plan involves a set of integrated process and actions: without Actions, monitoring and reporting processes are meaningless; without monitoring processes it is impossible to identify and plan meaningful Actions.

2. Gender Actions

2.1. It is an EU FP6 requirement that two types of gender-focussed activity related to women's participation in the Project must take place:

(i) Recording the Project's workforce statistics periodically during the Project (disaggregating male and female participants) in 7 categories of roles:

Type of Position	Note: To make these statistics as meaningful as possible, it is desirable to compare the proportion of women working in the Project to the proportion of women working/researching in the scientific and technical sectors related to the Project's development topics. Look for relevant statistical data to compare representation/participation of women within TENCompetence with the overall gender situation in the related work/academic/ professional/technical sectors.
Scientific manager (Full Professor/Senior Lecturer)	
Scientific team leader / work package manager	
Experienced researcher (> 4 years)	
Early researcher (< 4 years)	
PhD students	
Technical staff	
Other	

The data is collected from each Project Contractor and is used by each Contractor, as well as across the whole Project, to monitor and identify where improvements are needed in employment of women in the Project. Recording the workforce gender data is not a Gender Action in itself; but proactive monitoring of the gathered data and taking steps accordingly to improve the proportion of women employees are Actions – as shown below.

(ii) Undertaking and periodically recording Gender Actions which promote participation of women in the Project's activities and which raise awareness of gender issues more generally.

For all FP6 Projects, Gender Actions are reported by each Work Package under the following Actions headings:

Gender balance within Project workforce
Raising gender awareness
Promotion of Women in Science
Monitoring Action Plan Implementation
Other – please specify

Further details about Gender Actions relevant to these headings are in the GPAP Guidance Notes.

2.2. Actions are key to success of gender mainstreaming.

Each Contractor and Work Package will need to clearly identify who will do what with regard to Gender Actions, and put in place the mechanisms to make sure that Gender Actions and monitoring do happen.

An early Gender Action for each Contractor and Work Package is to set up their internal mechanism and structure for allocating responsibilities and implementing the GPAP.

Guidance on implementing Gender Actions is included in the GPAP Guidance Notes.

Gender Actions are the responsibility of everyone in the Project, especially those who:

- recruit and employ staff (Contractors)
- lead Work Packages
- lead teams within Work Packages
- organize in-Project events and activities at all levels
- organize external evaluation or dissemination events.

But everyone can proactively introduce a gender dimension to their work by encouraging and monitoring participation of women in their own teams/meetings/forums/networks.

While the employed ratio of women and the participation of women in internal events and meetings is relevant to every part of the Project, some different types of gender issues will arise in different Work Packages, For example:

- those Work Packages carrying out evaluations or integration work with external partners will have opportunities to consider and influence the gender of participants in the evaluation activities and to discuss gender implications with external evaluation partners.
- those Work Packages engaged in dissemination activities can include dissemination about TENCompetence Gender Actions/issues/findings as part of the messages about the Project that are conveyed externally.

2.3. Examples of Actions

Gender Actions can be quite simple but they need to be planned, monitored and followed-through. The GPAP Guidance Notes give further details about Gender Actions; two illustrative examples are given below:

Example of Gender Action. Imagined Scenario: women's conference attendance.

You have organized an internal TENCompetence conference or event and you note that women's attendance has been low (ie. the proportion of women attending the event is lower than the proportion of women in the potential pool of attendees). You want to improve this situation for the next conference: when arranging the next event you send targeted invitations to women encouraging them to attend. You monitor whether this Action has any impact – is there a higher proportion of women at the next event? If so, repeat the Action for other events. If not, investigate why not and look for other Actions.

For this Action to be effective you will need to find out what the proportion of women is in the potential pool of attendees so that you can compare the proportion actually attending. You may need to identify the relevant women in order to target invitations to them; you could ask them what will increase their likelihood of attending. If the women identify something to increase their likelihood of attending that is within your scope, take action accordingly. If what they identify is something that requires the input of others, or some extra resources, take it further to discuss with the appropriate decision-makers/budget holders/managers.

Example of Gender Action. Imagined Scenario: raising awareness of gender

The GPAP (within the Project Handbook) has been posted on the website but has not yet been discussed and disseminated within your Work Package/Team. Colleagues either are unaware of it or appear to view it as low priority or an administrative chore. Take Awareness-Raising Action by:

- *Put the GPAP on Meeting agendas, with the question 'How does this apply to our work?'*
- *Set up mechanisms and responsibilities regarding who does what for the GPAP within your Work Package/team*
- *Start up and contribute to discussions on the GAP Forum on the website*
- *Find out and disseminate information about current EU gender issues (eg see <http://cordis.europa.eu/improving/women/documents.htm>)*

3. Monitoring Progress and the Reporting Timetable

The Reporting timetable for the Gender Project Action Plan is in line with TENCompetence's reporting cycle (Table 3) and is explained below.

3.1. Reporting Workforce data disaggregated by gender - Annually

Except for year 1 of the Project (2006), the Project Co-ordinator will collect Workforce Statistics from each Contractor (in the form of an electronic questionnaire) at the end of each 12 months, at the same time as collection of other periodic annual reports.

In the first year of the Project (2006), the Workforce gender data will be collected in October 2006 – separately from other annual reports; for the remainder of the Project it will be collected at the same time as other annual reports.

The required FP6 format for reporting this Workforce gender data is as follows and this is what the questionnaire looks like:

Please complete the table below on a headcount basis.

(Previously supplied data will be inserted automatically)

Type of Position	Number of Women	Number of Men	Total	% Women	% Men
Scientific manager					
Scientific team leader / work package manager					
Experienced researcher (> 4 years)					
Early researcher (<= 4 years)					
PhD students					
Technical staff					
Other					

3.2. Reporting on Gender Actions - Quarterly

Except for year 1 of the Project (2006), the Co-ordinator will request quarterly reports (online questionnaire) on Gender Actions at the same time as requesting other quarterly internal activity reports.

The required FP6 format for recording Gender Actions is as follows and this is what the questionnaire looks like:

Please use the table below to describe gender actions that have been undertaken, any problems that have been encountered and rate the success level.

	Actions taken	Description	Problems encountered	Success rate (score 1- 5)	Comment
Gender balance within Project workforce					
Raising gender awareness					
Promotion of Women in Science					
Monitoring Action Plan Implementation					
Other – please specify					

Advice about compiling both the above reports is included in the GPAP Guidance Notes.

3.3. Monitoring Progress - by Contractors and Work Packages

The information and data collected in these two reports will be collated by the Project Co-ordinator for the purposes of whole-Project reporting (see 3.4. below).

These reports are the basis for each Contractor and Work Package Leader to plan ongoing monitoring and Gender Actions within their own spheres of responsibility. The reports should be discussed at all levels and disseminated to colleagues.

The annual Workforce gender statistics provide an overview and should be made available by Contractors to all staff involved in recruitment and selection of employees for a Work Package, so that when posts are being filled or replaced during the life of the Project, Actions can be taken towards improving any gender imbalances shown in the statistics. (Contractors will also have institutional Equal Opportunities policies and other organisational guidelines which will underpin their recruitment and employment processes.)

The quarterly reports on Gender Actions should be regularly discussed within Work Packages and disseminated to colleagues. It should be made clear, within each Work Package, what the mechanisms are and who is responsible for identifying possible Gender Actions, taking the Actions, monitoring and reporting.

Further details about monitoring progress are in the GPAP Guidance Notes.

3.4. Whole-Project Monitoring and Reporting for the GPAP

The Project Co-ordinator will ensure that reports from each Contractor and Work Package are collated to give a cross-Project picture.

There will be an Annual GPAP Report, in which the collated reports and the whole-Project picture will be analysed, made to the Project Consortium Board. The Board may identify and disseminate responses to progress being made and may recommend Gender Actions to be taken on a cross-Project basis.

The Project Co-ordinator will submit Gender Action Plan Reports to the Commission in accordance with the external reporting requirements – at end of first year and a final report.