Language Technologies for Lifelong Learning

LTfLL -2008-212578

Project Deliverable Report

D7.1 – Validation Design

Work Package	WP7
Task	
Date of delivery	**Contractual**: 01-09-2008 **Actual**: 28-10-2008
Code name	D7.1 **Version**: 1.0 Draft ☐ Final ☒
Type of deliverable	Report
Security (distribution level)	PU
Contributors	UNIMAN, AURUS, OUNL, UTU, PUB-NCIT, UPMF, UU, WUW
Authors (Partner)	Gillian Armitt, Isobel Braidman, Tim Dornan (UNIMAN) Jan Hensgens (AURUS) Howard Spoelstra (OUNL) Gaston Burek (UTU) Stefan Trausan-Matu, Vlad Posea (PUB-NCIT) Philippe Dessus (UPMF) Paola Monachesi, Eelco Mossel (UU) Fridolin Wild, Bernhard Hoisl (WUW)
Contact Person	Gillian Armitt (UNIMAN) Tel. 00 44 161 275 8108 University of Manchester Gillian.Armitt@manchester.ac.uk 186 Waterloo Place Oxford Road Manchester M13 9PL UK
WP/Task responsible	WP7.1 (UNIMAN)
EC Project Officer	Mr. M. Májek
Abstract (for dissemination)	This deliverable describes the framework, approach and early planning for validation of the software services and evaluation of the LTfLL methodology, project and work approach.
Keywords List	Evaluation, Validation, Assessment, Verification, Usability

Table of Contents

Executive Summary

The Language Technologies for Lifelong Learning (LTfLL) project aims to provide the following high-level software services:

- Positioning the learner in the domain (workpackage 4 task 1)
- Diagnosis of conceptual development (workpackage 4 task 2)
- Feedback based on interaction analysis (workpackage 5 task 1)
- Feedback based on assessing textual products (workpackage 5 task 2)
- Knowledge-sharing network (workpackage 6 task 1)
- Adding a social component to public knowledge (workpackage 6 task 2)

The purpose of this document is to describe the framework, approach and early planning for validation of the software services and evaluation of the LTfLL methodology, project and work approach. Validation of software is concerned with answering the question “have we built the right software? (is this what the customer wants?)” rather than “does the software match the specification?”.

Validation of the software services will take place throughout the software development life cycle to provide formative feedback to designers, developers and language technologists at each stage. The final validation cycle will also be formative, informing an end of project roadmap for future development.

Validation of the software services will be scenario-driven and will use the criteria for validation as expressed by stakeholders during the design of scenarios and as subsequently enhanced by formative feedback through the evaluation process. It is anticipated that criteria for validation will include aspects of pedagogic/andragogic effectiveness and need, usability, user acceptance, efficiency, and transferability between different systems and situations, though the specific aspects of each will emerge during scenario design.

We have chosen to adopt and adapt the Scenario-Based Design (SBD) Method (Rosson & Carroll, 2002[1]). SBD puts from the very beginning the focus on real user problems and tasks with their proposed solutions. It achieves this by creating a series of scenarios in close cooperation with stakeholders, culminating with the **validation scenario**. These will encapsulate many of the key decisions about validation, including the stakeholders, the criteria for validation and methodologies which are appropriate to the validation criteria, including sampling and indicators. **The validation scenarios are due in later months of the project; accordingly, this document provides a framework within which key decisions required for validation will be taken later.**

[1] Rosson M.B. and Carroll J.M. (2002), Usability engineering: scenario-based development of human-computer interaction, Morgan Kaufmann Publishers Inc., San Francisco.

Validation of the high-level software services will take place at the scenario design stage, as well as at each of the three software releases (showcase, version 1 and version 2). The releases will be evaluated through pilots with real users, and other stakeholders where appropriate and realistic within the pre-defined time frame. Each service will be evaluated in at least two European languages, such that all the languages of the Consortium are used. Some pilots will be in the information technology domain and some in medicine.

As well as validating the software services, the project will evaluate the LTfLL methodology, project and work approach. This will take place throughout the lifetime of the project to promote reflection and formative feedback on the effectiveness of the technical tools and human aspects of the project, including communication and management. It will also be summative, to inform others outside the project of lessons learnt.

SECTION 1: Introduction

1.1 Overview of LTfLL project

The LTfLL (Language Technologies for Lifelong Learning) project will create next-generation support and advice services to enhance individual and collaborative building of competences and knowledge creation in educational and organisational settings. The project makes extensive use of language technologies and cognitive models in the services.

The research activities are enveloped by activities that ensure common ground in the use cases and pedagogically/andragogically sound scenarios that steer the design and development of the services and guide the validation; a technical infrastructure for the creation and integration of the services and a validation structure that ensures rigorous evaluation in realistic settings, with several languages supported.

The research of the project is organised in three themes, each leading to particular types of services and infrastructures:

In theme 1 (workpackage 4) services are developed to establish the current position of the learner in a domain. Services will offer semi-automatic analysis and comparison of learner portfolios to the domain knowledge and continuous modelling and measurement of conceptual development.

In theme 2 (workpackage 5) support and feedback services are developed based on analysis of the interactions of students – using Natural Language Processing (NLP) and Social Network Analysis (SNA) – and textual output of students – using Latent Semantic Analysis (LSA) with contributions from NLP.

In theme 3 (workpackage 6) a knowledge sharing infrastructure is construed that allows comparison and sharing of private knowledge to give rise to new common knowledge and social learning. Ontologies for formal domain representation are combined with social tagging.

The services are expected to result in improved appreciation of learner requirements, which are in turn expected to lead to better recommendations on study plans and resources. Progress monitoring based on learning activities, rather than on formal assessments, is expected to improve recommendations for further competence building and may improve co-construction of knowledge in social and informal learning.

1.2 Purpose of validation plan

The purpose of this document is to define an initial validation plan, which will form the basis of an evolving validation plan as the project progresses. Validation will concern itself with two aspects of the project: validation of the services produced, and validation of the LTfLL methodology, project and work approach.

1.2.1 Validation of the LTfLL services

Validation of the LTfLL services will be driven by use cases and scenarios, and aims to determine the extent to which the services meet the expectations of stakeholders.

The expectations of stakeholders will be expressed as criteria within the scenarios. Validation will take place through pilots that will test the criteria expressed in the scenarios.

The criteria for evaluation will emerge as scenarios are developed, but are expected to include:

- Pedagogic/andragogic effectiveness, including the extent to which the services foster learning
- Usability, user acceptance and efficiency
- Transferability between different systems and organisational/educational environments

Because the specification of scenarios is currently at an early stage, this document will focus on:
- the framework for validation planning
- the approach to validation
- planning for the first round of pilots

Where incomplete information is currently available, this plan indicates the process and timescale over which the information will become available.

1.2.2 Evaluation of the overall project approach

In any project, there is much to learn through reflection on the tools used and human processes by which the project reaches its conclusion. This validation plan will make provision for formative reflections by the project team on the project approach in areas including the following:

- Appropriateness of technical methods and approaches
- Effectiveness of approaches to partner collaboration
- Management of the project and workpackages

These reflections will take place throughout the lifetime of the project and will also be summative, to inform others outside the project of lessons learnt.

1.3 Overview of validation plan

This plan comprises:

Section 2: the LTfLL context – the partners, the scenarios approach, and services to be validated.

Section 3: the framework for validation of the LTfLL services and evaluation of the LTfLL approach.

Section 4: the validation approach – validation of the scenario design and software products, and evaluation of the LTfLL overall approach.

Section 5: the pilot planning framework.

Section 6: pilot planning for the initial “showcase” version.

Section 7: a diary of the activities and milestones.

Appendices: the appendices give more detail on *methodologies* and the *criteria for validation* that are to emerge from the scenarios approach.

1.4 Relationship of validation plan to other LTfLL workpackages and deliverables

The LTfLL project adopts the Scenario-Based Design (SBD) method of Rosson and Carroll[2]. Scenarios are developed in workpackage 3 and form the basis for the development of the LTfLL services in workpackages 4, 5 and 6. Scenarios also direct the validation of the LTfLL services in workpackage 7, and pilot design is dependent on the requirements emerging from the scenarios. Workpackage 2 integrates the services emerging from workpackages 4 - 6. It can therefore be seen that scenarios are critically important to the delivery of LTfLL.

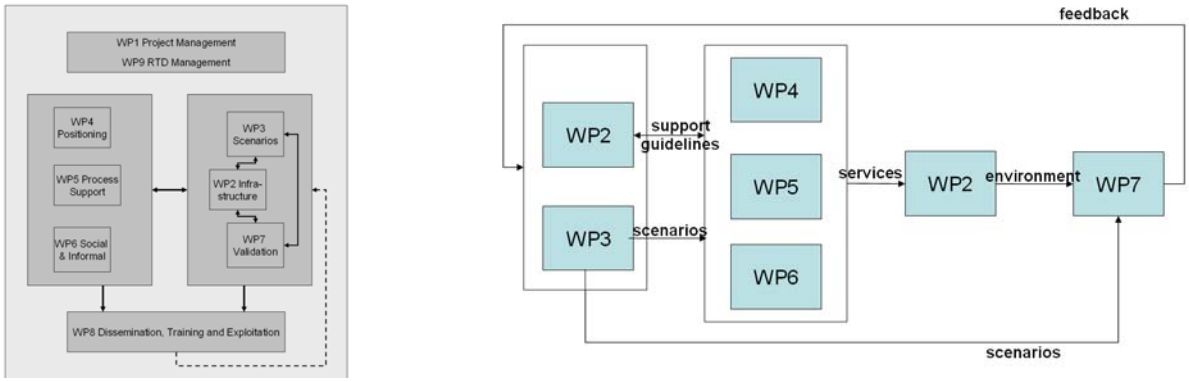

Figure 1. Relationship of the workpackages

Three phases of pilots are planned:

- a preliminary 'mash-up' using existing services ("showcase")
- Version 1 of the LTfLL services
- Version 2 of the LTfLL services

Workpackage 7 is responsible for validating the services in each of the three pilots. The results of the first two evaluations (deliverables D7.2 and D7.3) provide formative feedback into the scenarios and service development, and the results of the third evaluation are used to generate a 'roadmap' (deliverable D7.4) to inform future work after the end of the project.

2 Rosson M.B. and Carroll J.M. (2002), Usability engineering: scenario-based development of human-computer interaction, Morgan Kaufmann Publishers Inc., San Francisco.

1.5 Smart pilots

The LTfLL Description of Work (DoW) specifies that the pilot schedule will cover all six languages represented by the Consortium: Bulgarian, Dutch, English, French, German and Romanian. It also specifies that the pilots will cover the two domains of Medicine and Information Technology (IT).

The DoW specifies six high-level services to be piloted. A smart validation approach will be applied, in which each service will be tested in two languages. This will give a minimum of twelve pilots, each comprising the three phases described in Section 1.4. Some pilots will take place in Medicine and some in the IT domain; however, not all services will be tested in both domains.

The DoW also requires a minimum of at least twenty five students to take part in each pilot, recruited from regular registered students as well as life-long learners. Under smart validation, only the final phase of each pilot has to contain as many as twenty-five students; earlier phases may be based on smaller numbers. This is discussed further in Section 3.2.2.6.

SECTION 2: Context

2.1 Partners

The partners in the LTfLL consortium represent seven different countries, six different languages and the Higher Education subject domains of Information Technology (IT) and Medicine.

Institution	Country	Language	Subject Domain
Open Universiteit Nederland, Heerlen	Netherlands	Dutch	IT
Universiteit Utrecht	Netherlands	Dutch	IT
Eberhard Karls Universität Tübingen	Germany	German	IT
Wirtschaftsuniversität Wien	Austria	German	IT
Université Pierre-Mendès, Grenoble	France	French	IT
Polytechnica University of Bucharest – National Center for Information Technology	Romania	Romanian	IT
Aurus Kennis- en Trainingssystemen BV	Netherlands	Dutch	N/A
The University of Manchester	UK	English	Medicine
Institute of Parallel Processing of the Bulgarian Academy of Sciences	Bulgaria	Bulgarian	IT
BIT MEDIA e-Learning Solution GMBH & CO KG	Austria	German	N/A

Table 1: Partners in LTfLL consortium

2.2 The scenarios approach

The experiences of several research and development projects in Europe have shown that too much focus on technology, rather than the needs of the stakeholders, can be quite risky for the project. With such a focus, the usability of the developed system or component often appears to be low, and the impact of the technology on the tasks of the different user groups to be unclear. One other risk is that the research starts to drift from the original ideas, leaving insufficient time to finalise the products, which endangers the robustness of the final results. This can lead to difficulties in testing realistic tasks in a less controlled environment. The only possible solution can be to perform testing in a 'laboratory' setting. This leaves the steps between such micro-testing and the potential educational market to be huge. In some cases the benefits of the final results are difficult to explain to people who are not directly involved in the project. This obscurity of the project aims and its products could also create serious problems in finding the required testing groups.

In the past, methods such as user-centered design were shown to relieve the above-mentioned problems. For the LTfLL project, we have chosen to adopt and adapt the Scenario-Based Design (SBD) Method (Rosson & Carroll, 2002[3]). The idea behind SBD is that it puts from the very beginning the focus on real user problems and tasks with their proposed solutions.

[3] Rosson M.B. and Carroll J.M. (2002), Usability engineering: scenario-based development of human-computer interaction, Morgan Kaufmann Publishers Inc., San Francisco.

The scenarios are easy adaptable to the different project phases, where the main communication with all parties involved is achieved using narratives written in natural language. This supports common and open communications between technical and non-technical project members and other stakeholders. SBD enables a continuous reflection on the development in progress and combines concreteness with flexibility, which fits perfectly the fluidness in the design processes (see also Carroll, 2000[4]).

For the LTfLL project, we are using customised templates for writing scenarios. We pay special attention to the educational settings as well as to the validation aspects. We take these into consideration from the start, and monitor and specify these into more details later on. Our motto will be just enough (details) and just in time (design phase). Our SBD approach will support parallel working for the workpackages, i.e. design, research and development for subsequent versions taking place concurrently with validation of the current formal release and testing of the intermediate versions of the next release. In the first project cycle, SBD directs the research, and in the following cycles it directs the development. In the following picture an overview is given of the whole process of scenario writing including the implications for the validation aspects. The writing process for each successive scenario starts with the information provided in the previous one, with updates taking place throughout the development cycle in response to research, development and testing. The progressive completion of the scenarios in response to on-going feedback implies that the timing of freezing of the development ("formal" release) and the final update to the scenario are quite close to each other. Therefore scenarios appear to be ready just before the final implementation of each version.

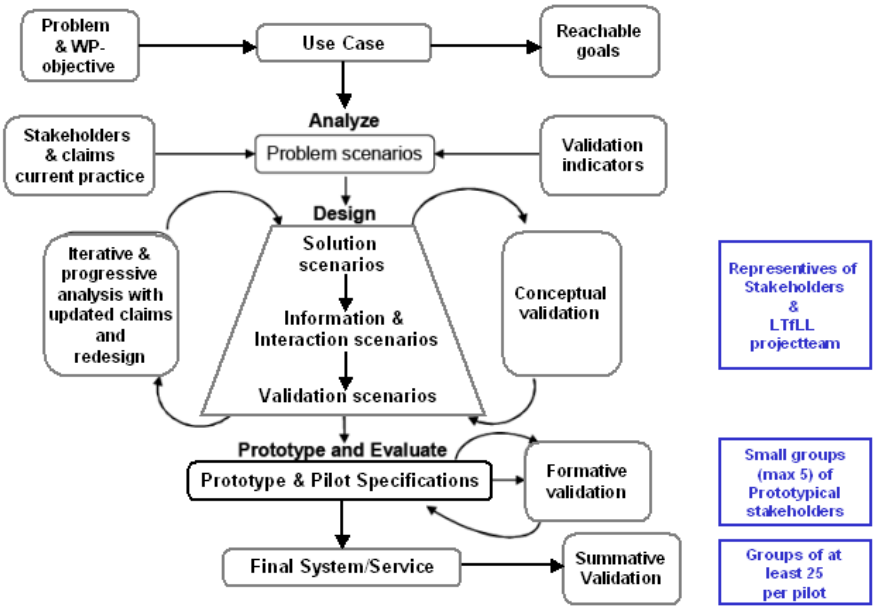

Figure 2. Scenario-Based Design

By using the scenarios, we want to minimise the possible gap between the promises and the final services and tools. By disciplining ourselves to specify the objectives, the pre-conditions, the assumptions etc. in advance, we want to realise our motto to promise only that which can be delivered.

[4] Carroll, J.M. (2000), Five Reasons for Scenario-Based Design In Interacting with Computer, 13 (1), 43-60.

The LTfLL SBD approach and its templates are also objects for reflection, as well as validation. We expect to learn during the project and want to use our reflective insights to improve our methodology. We already foresee some additional benefits of our SBD approach: it offers opportunities of validating the concepts and ideas in the scenarios long before finalising the pilot implementation plan. The problem and first solution scenarios can be used to discuss the possible benefits with the main stakeholders and in doing so, to facilitate the search for pilot institutions to provide test groups. However, perhaps the most important expected effect is that it enables the project to make considered and shared choices during the development process, based on usability, availability of test groups, availability of better solutions etc.

In defining use cases, the context and scope for each workpackage task will be set. Within the scope of each use case, educational problems and possible solutions are inventorised to write and design the associated scenarios. In a common discussion each workpackage team develops the scenarios. To make the writing process more efficient, parts of the scenarios can be shared; however ultimately each pilot (expressed as a unique combination of sub-task, language and domain) will have its own unique and customised validation scenario. The entire process is supported by an editorial board, who are providing templates, guidelines and customised advice during the writing process.

Within the problem scenario a stakeholder analysis is performed. The solution scenario constitutes a 'business case' to be used for consulting the important stakeholders. A validation scenario contains a Plan of Action to be used at validation time.

2.3 Services to be validated

This section contains a summary description of the services emerging from workpackages 4, 5 and 6. It should be emphasised that these summaries were written at an early stage in the project and will be refined in the LTfLL deliverables D3.1, D4.1, D5.1 and D6.1, due in Month 8. These deliverables will provide more detailed descriptions of the stakeholder requirements that will drive the validation. Therefore only an indication of validation criteria can be given in this section.

2.3.1 Services to be validated – workpackage 4

The services to be validated in workpackage 4 are portfolio analysis (task 4.1) and the diagnosis of conceptual development (task 4.2). Task 4.1 will develop services positioning the learner. The services developed by task 4.2 will test methodologies for measuring conceptual development.

2.3.1.1 Task 4.1 Portfolio analysis

The goal of this task is to enable students and tutors to identify knowledge gaps and potentially to recommend optimal sequences of learning materials to cover those gaps. Validation will focus on the effectiveness of the software in meeting the requirements of students, tutors and other stakeholders.

In the first part of the project, evaluation will take place concerning the utility of LSA, PLSA and methodologies implemented by rich knowledge approaches that involve the use of lexical knowledge resources. This will lead to Version 1 of the portfolio analysis tools that include among others the following functionalities:

- Identifying fragments of texts from learning materials that overlap in their content/meaning with texts contained in the learner e-portfolio. Texts from the e-portfolio will be used to query the learning materials.
- Organising those results according to a rating of confidence.
- Training system parameters.
- Deploying those results data in a standardised format.

Version 2 of the software will contain the functionalities from Version 1 as well as new functionalities:

- A user interface that provides a set of learning paths (sequences of learning materials) as recommendations to the user. Those recommendations are based on texts contained in the learner portfolio and learner knowledge gaps, learning materials and learning goals.
- The user will be asked to select preferences from a menu of options. This functionality will be particularly useful for specifying the learners learning goals.
- Knowledge gap analysis will be performed with a set of alternative methodologies tested in Version 1.
- Graphical interface that display recommended learning paths together with their associated confidence ratings.

2.3.1.2 Task 4.2 Diagnosis of conceptual development

Task 4.2 investigates the conceptual development of the learner. The emphasis in this second task is on studying how learners progress in their acquisition of knowledge and understanding from the evidence of their textual output, so that further services can be offered to both tutor/facilitators and learners, which are appropriate to their stage of conceptual development and to learner needs. Initially, medicine will be used as a domain for these studies.

Studies will commence by testing appropriate knowledge measurement techniques, for example card sorting and concept mapping, against extended reports and essays and online material generated through discussion fora, to select the most appropriate methodology for the study. LSA will also be used to analyse the textual material. These methods will be tested over a wide spread of abilities and as learners progress through the curriculum. We have already reviewed the methodologies available for this project.

The validation criteria that will be applied to this component of workpackage 4 are therefore considered from an investigative perspective. They will apply to both the methods selected and the observations made. These are of two types:

Reliability:
- Inter-rater/observer reliability
- Repeatability of observations
- Comparison of results from different groups of learners at the similar levels

Validity
- Agreement of results obtained using different knowledge measurement techniques
- Comparison of results obtained for individual learners with external academic assessments (grades, examination results where appropriate)

The exact type of statistical analysis used can only be identified after initial results have been obtained as there is no existing indication of the data spread. Likewise, the validation methods for any services developed will be selected according to the nature of those services.

2.3.2 Services to be validated - workpackage 5

Workpackage 5 aims to develop tools to provide support and feedback during learning, based on the automatic analysis of learners' textual interactions and discourse. Recommendations will be provided on the writing style of the students' essays, on the selection of learning topics, and on the quality and degree of participation in collaborative activities in forums and chats. Validation will focus on the effectiveness of the software in meeting the requirements of students, tutors and other stakeholders.

Although the detailed specification of the WP5 services is still in progress, the tools developed in this workpackage are expected to provide feedback on many of the following issues:

- to what degree a student's essay or summary of a document read previously corresponds to the expected topics and/or its similarity to tutor-provided documents,
- which of the main topic ideas of the course the student appear to understand, after reading around a topic, and which topics have/have not been covered,
- what are the most important sentences/paragraphs of the course,
- qualitative data about the covered topics in chats, forums and written essays

and for collaborative discussion fora:

- the effectiveness of the conversation as a whole, from the collaboration point of view,
- the contribution of each of the utterances in chats towards achieving the shared goal,
- diagram of the interactions between students ('inter-animation'), including graphical representations of the conversation,
- measures of the polyphonic structure of the chats and forums,
- summary of the conversation,
- assessment of the contribution of each student at different moments in time in the collaborative activities on forums and chats
- the competence of the students on a given topic (competence map),
- the degree of participation and other quantitative data in chats and forums,
- the time of day of participation (peak periods) in forums,
- who is a reader-only (lurker - only looking) and who is a writer, i.e. engaged in the collaborative activities
- what causes an ineffective collaboration?
- positive examples of collaboration segments (message adapted for students' use, for teachers' use, and for developers' use)
- recommendations to the tutor about the psycho-social profile (e.g. who is a leader, a lurker, an extrovert etc. in order to inform future group composition, e.g. one leader only in each group).

Validation of this workpackage will centre on the effectiveness of the feedback services for supporting learners and tutors.

T**ask 1** provides a service that analyses the interactions among students collaborating with instant messaging (chat) systems or discussion forums. The discussion can be synchronous (e.g. chat) or asynchronous (e.g. forum) depending on the nature of the task and the objectives of the evaluation (e.g. chat sessions might be used as tests, while forums can be more useful in research activities).

The pilot functionality for task 1 includes the analysis of transcripts of chat sessions and the delivery of feedback, both to students and tutors, comprising the following reports:

- topics that have been discussed
- graphical representation of the conversation and of inter-animation
- automatic evaluation (grading) of the learners
- social network analysis of the interactions

Task 2 provides software based on LSA-based cognitive models. These models mainly relate on providing feedback on students' summaries concerning their understanding of the course. We focused on student summaries because this activity fosters and demonstrates comprehension (van Dijk and Kintsch, 1983[5]): the better understood a text is, the better it will be summarised. Also, summarising a text fosters its understanding.

The pilot functionality for task 2 includes the analysis of learners' essays and the delivery of feedback composed of the following reports:

- to what extent the summary reflects the most important course sentences
- detection of summary parts that are "off the subject"
- automatic assessment of the learners' understanding
- automatic selection of the next course topic according to the previous assessment.

2.3.3 Services to be validated - Workpackage 6

The objective of WP6 is to support informal and social learning by:

- setting the basis for a knowledge-sharing network which can be used to support the emergence of new common knowledge in a community of learners (task 1);
- creating a link between the formal representation of a given domain in the form of ontologies (which allow for easy retrieval and reuse of the learning material) and the informal descriptions produced by social tagging (which allows for the inclusion of the appreciation people express for different kinds of material and thus identify their needs and interests (task 2)).

Validation will focus on the effectiveness of the software in meeting the requirements of students, tutors and other stakeholders, more information becoming available as the scenarios mature in the next few months of the project.

To support **task 1**, a Common Semantic Framework will be created that can support the emergence of new common knowledge arising from the learning experience, as well as the formal and informal learning process. The formal process is driven by an ontology that guides the learner in his/her learning task, while the informal process is driven by social tagging and annotation.

[5] van Dijk, T.A. & Kintsch, W. (1983) Strategies of Discourse Comprehension, New York: Academic Press

The Common Semantic Framework will support the learner in:

- providing multilingual access to the knowledge he is looking for,
- filtering specific knowledge, through cultural and social adaptation of the knowledge domain to different users,
- accessing various resources by means of a unified model to which the different resources can be mapped,
- searching for the appropriate knowledge by means of different kinds of search,
- visualising the resources in an interactive way.

To support **task 2**, we will develop a knowledge-sharing system that connects (1) learners to resources and (2) learners to other learners by means of user profiles and social tagging. We will also link the social tags to an existing domain ontology in order to complement the formal knowledge with the informal knowledge. The system will be part of the Common Semantic Framework developed in task 1. There will be an interaction between the services developed in the two tasks which are thus closely interrelated.

The system will support the learner by:

- adding a social dimension to the search system available via the Common Semantic Framework since the learner's social network and profile will be taken into account,
- accessing the informal classification of the knowledge provided by other learners which might be different from formal knowledge associated with the ontology;
- recommending to him new learners (or tutors) with knowledge/skills relevant for a given task, to help him in the learning process either by giving feedback or to share learning activities;
- choosing the appropriate learning path on the basis of categorisation and prioritising of the available resources.

In addition, it will support the teacher in:

- being able to monitor what the learners are doing and have acquired;
- being able to see which resources are accessed through the system.

SECTION 3: Framework for Validation and Evaluation

3.1 Introduction

The validation plan will concern itself with two aspects of the project:

- validation of the LTfLL services (both the design and the developed product)
- evaluation of the LTfLL approach.

Validation of software is concerned with answering the question "have we built the right software? (is this what the customer wants?)". This should not be confused with **verification**, which answers the question "does the software match the specification?".

Validation of the services will take place throughout the software development life cycle to provide formative feedback to designers, developers and language technologists at each stage. The final validation cycle will also be formative, informing an end of project roadmap for future development. The formative evaluations are by their nature summative for the phase of the project concerned.

Evaluation is concerned with establishing the value of something to its stakeholders through critical appraisal. In the LTfLL project, a number of criteria will be evaluated to contribute towards answering the validation question "have we built the right software?".

Evaluation of the LTfLL project methodologies and approach will take place throughout the lifetime of the project to promote reflection and formative feedback on the effectiveness of the technical tools and human aspects of the project. It will also be summative, to inform others outside the project of lessons learnt.

The validation plan will evolve throughout the lifetime of the project, being subject to discussion at each Consortium meeting and will be updated as decisions are made during the project.

3.2 Validation of the LTfLL Services

The purpose of validating the LTfLL services is to determine the extent to which the design or the developed product meets stakeholder expectations, in order to:

- inform the design and development of the LTfLL services, throughout the project and as a final roadmap
- demonstrate the value of the LTfLL services to others outside the project

Validation is embedded within and integral to the LTfLL software development lifecycle. Both the design of the software as expressed in the scenarios, and the developed software will be validated within a quality framework for software development that supports the validation.

The identification of stakeholders and their involvement in the development of scenarios as well as in evaluating the developed services is crucial to the success of the project.

3.2.1 Identification of stakeholders and development of scenarios

Users and other stakeholders will be involved throughout the development life cycle and will inform the direction of product development and refinement.

Given the core position of scenarios in the LTfLL Scenario-Based Design approach (see section 2.2 of this report), the stakeholders as well as the criteria to be used are integrated in the initial use cases and scenarios, due in Month 8 of the project. Within the use case, the initial analysis of stakeholders is made under the heading “problem situation”. Within this are described the “who has what problem?”, “What makes this problem a problem?” and “Why is it important to solve the problem?”. The attainable goals and the first outline of solutions are also part of the use case.

These specifications are used for the next phase of writing the “problem scenario”, where the main actors and their perspectives concerning the (problematic) task execution are introduced. This scenario describes the current practice. For the “problem scenario”, an analysis is performed to collect the current features and claims with their pros and contras in one table.

In the specification of “solution scenarios” phase, the proposed solutions are elaborated, including a new or adapted task execution with the interactions of the main actors. The features and claims table with the pros and contras is updated and the benefits of the solution can be seen by comparing both tables. Therefore the “solution scenario” will act as the business case and will be discussed with the main stakeholders. If the presentation of a solution is not convincing, the development of this scenario will be stopped. More criteria can be used to stop a further elaboration of a scenario, even during subsequent phases. These criteria include usability, performance and the availability of pilot institutions and real test groups. The final set will be completed and reported during the project life cycle.

The LTfLL project plan does not enable us to make an initial list of scenarios for this report at this stage. The first substantial “problem” and “solution” scenarios are scheduled to be developed in the autumn and winter of 2008, and these will be developed progressively using feedback loops and iteration. A full description of the LTfLL SBD method, its templates and examples is planned for the first deliverable of workpackage 3 (October 2008). In future validation reports (deliverables D7.2, D7.3 and D7.4), the elaborated scenarios will be described in more detail.

3.2.2 The software development lifecycle and validation process

This section provides an overview of the software development lifecycle and validation process to be applied in the LTfLL project.

3.2.2.1 Development of scenarios and criteria for validation

The development lifecycle starts with the identification of use cases and scenarios in the early months of the project. These were developed by domain experts in teaching and learning within the project team, supported by AURUS and OUNL. The scenarios identify the expected ‘value’ of the development, and the grounding of the scenario in contemporary pedagogic/andragogic theory and practice. There are a number of different types of scenario, which are developed sequentially culminating in the ‘validation scenario’. These validation scenarios will drive much of the validation activity.

Crucially, the criteria against which the software will be validated will be specified in the scenarios. The criteria for validation are emerging as scenario development takes place, and are expected to fall within the following three broad areas:

1. Pedagogic/andragogic effectiveness
2. Usability, user acceptance and efficiency
3. Transferability between different systems and organisational/educational environments

An overview of these three broad areas is given in Appendices 2, 3 and 4.
While usability of the software will be tested throughout the project, validation of pedagogic/andragogic effectiveness and transferability will take place in the later versions of the software only.

3.2.2.2 Validation of the scenarios

It is important that the scenarios are validated so that development can proceed with the confidence that the scenarios meet initial stakeholder requirements. Amongst the stakeholders is the EU; scenarios will be checked to ensure that they fall within the scope of the LTfLL Description of Work.

Changes at the design stage in response to stakeholder feedback are much less costly than those following development. It should be noted that validation at this stage will not be perfect, as it is difficult for stakeholders to comprehend fully how the finished product will behave when in live running; also stakeholder requirements may evolve during the course of the project because of changes in their business environment.

Usability of user interface should be tested as part of the validation of scenarios where appropriate. This is important to ensure that users' attitudes towards the user interface do not impact on other criteria being tested, e.g. pedagogic effectiveness.

3.2.2.3 Software standards

Standards for software development will be drawn up by workpackage 2. These will include international standards to be applied, good coding practice, standards for interface design and arrangements for software testing prior to release for validation.

3.2.2.4 Software development

The LTfLL project will comprise three cycles of development leading to product 'releases': a preliminary 'mash-up' of existing technologies (the **showcase**), an intermediate **version 1** and a final **version 2**. The products will be developed using a service-oriented architecture approach, in which discrete 'services' will be developed and integrated with each other through interfaces.

The purpose of the showcase is to demonstrate how language tools can be used in an educational setting illustrating the use of (parts of) the LTfLL tools, in order to inform development of version 1. Criteria for the showcase include:

- the conditions, infrastructure and pilot groups for the chosen domain and language for the pre-pilot testing should be available
- existing and stable language technologies will be used
- each workpackage task will produce a showcase

The products will comprise six separate high-level services:

Positioning the learner	(WP4 task 1)
Diagnosis of conceptual development	(WP4 task 2)
Feedback based on interaction analysis	(WP5 task 1)
Feedback based on assessing textual products	(WP5 task 2)
Knowledge-sharing network	(WP6 task 1)
Adding a social component to public knowledge	(WP6 task 2)

Each high-level service will comprise a number of integrated reusable services. The extent to which the high-level services will be integrated with each other during the course of the project will be established later in the project; however, it is likely that users would perceive each high-level service as a distinct unit of functionality. Accordingly, validation will take place at workpackage task level for all versions of the software, even where an integrated product is piloted.

3.2.2.5 Software testing

Each of the three phased releases (showcase, version 1 and version 2) will be subject to testing by the development teams prior to being handed over for validation. Each development team will draw up its own arrangements to cover unit, module, system, integration and load testing, as appropriate.

Alpha-testing will be undertaken by domain experts who are members of the project team, and decisions will be made within workpackages 4 – 6 concerning how this will take place and criteria for acceptance of software for beta-testing.

Each development team will arrange beta-testing of Versions 1 and 2 by a minimum of one (version 1) or three (version 2) real target users outside the project team. Beta-testing will take place in the production environment.

Following successful beta-testing, a formal process will take place by which the software is handed over for validation. At this point, software in the production environment will be 'frozen' until the end of the validation pilots, unless fatal flaws are exposed necessitating a software update after pilots have started.

Usability of the software should be tested during alpha and beta-testing, as well as following each major release. This is important to ensure that users' attitudes towards the user interface do not impact on other criteria being tested, e.g. pedagogic effectiveness.

3.2.2.6 Pilots

Each pilot is defined as a unique combination of (scenario, language, domain). Each pilot will have three phases, corresponding to validation of the three releases (showcase, version 1 and version 2).

Each high-level service will be piloted in two languages in version 2, such that all the six languages of the Consortium are included (Bulgarian, Dutch, English, French, German, Romanian). One or two languages may be used for the showcase and version 1.

Services will be piloted in partner institutions in the domains of Medicine and/or Information Technology, though not all high-level services will be piloted in both domains.

Pilots will take place with real target users where possible and appropriate. The number of users will be specified in the pilot planning template, but will include at least 25 users per high level service for version 2. A maximum of 5 users is recommended for the showcase and version 1, in accordance with Nielsen's recommendations[6,7]

A plan will be drawn up at the Consortium's autumn meeting to ensure that the educational transferability of the software to different student groups and organisational environments is addressed, including transferability to regular and lifelong learners.

3.2.2.7 Evaluation

For each pilot, appropriate evaluation methodologies will be applied to address the criteria to be tested as defined in the validation scenario. These will be selected as part of the development of the validation scenario. Appendix 1 gives indicative methodologies.

Validation of whether the software works acceptably in different systems and organisational environments will be validated by comparison of the results of the evaluations of the same service in different partner institutions. Validation of the transferability of the software will include consideration of whether the software is effective for different student groups. In concept, different attributes of students will be identified, e.g. age, level, gender. A plan for probing differences between groups of students with different attributes will then be drawn up.

The quality and value of the data from candidate evaluation methodologies will be considered very carefully, to produce best value from the resources available within this workpackage. Some aspects of evaluation may require data to be collected in the language of the pilot, thus limiting the number of partners who can process the data. However, where it is possible without compromising the value of the data collected, evaluation methodologies will produce data in language-neutral format. This will allow the major partners in workpackage 7 to process data from pilots in other languages, thus reducing the burden on the minor partners in this workpackage. The possibility of data being machine-processed will be investigated as part of the development of scenarios.

3.2.2.8 Revising the user requirements and informing further development

Following data gathering and analysis, a document recommending revisions to the software will be produced. These recommendations will enter a process of revisions to the scenarios, and acceptance for development in later versions of the software or in the roadmap as appropriate.

3.3 Evaluation of the LTfLL Approach

By the 'LTfLL approach', we mean the way in which we manage the project and the effectiveness of the decisions we make. The evaluation will focus on both management and technical decisions impacting on the success of the project in both positive and negative ways. The purposes of evaluating the LTfLL approach are:

[6] Nielsen, J. (2000) "Why you only need to test with 5 users", www.useit.com/alertbox/20000319.html (last accessed 7 August 2008)
[7] Nielsen, J."How to conduct a heuristic evaluation", www.useit.com/papers/heuristic/heuristic_evaluation.html (last accessed 7 August 2008)

- to provide on-going formative feedback to the project team on areas for improvement
- to provide summative information on the lessons learnt about the project approach, from which others may benefit

The LTfLL approach will be validated in the following general areas, though more areas may emerge as evaluation progresses.

1. Appropriateness of technical methodologies and approaches
2. Effectiveness of approaches to partner collaboration
3. Management of the project and workpackages

Most of these areas over-arch the project and are relevant to all partners; therefore, all partners are expected to contribute on an on-going basis to data collection for this evaluation. The methodology for partner input and evaluation will be incorporated into existing communication processes so far as is practicable.

The effectiveness of the LTfLL approach will impact not only on the project team but also on external stakeholders involved with the project, who will view project activities from a different perspective. Therefore input will be sought from stakeholders concerning their experience of contributing to the project and the effectiveness of their interactions with the Consortium. Users taking part in the early pilots will be asked to provide formative input. The wider stakeholder base will contribute to the summative evaluation towards the end of the project, and to formative evaluation at earlier stages in the project where appropriate.

SECTION 4 – Validation Approach

4.1 Introduction

This section describes the process by which validation will take place, including the partners involved in each activity. Where decisions remain to be made, the decision-making process is described.

This section contains three sub-sections, covering:

- validation of the LTfLL scenarios
- validation of the software products
- evaluation of the LTfLL approach

4.2 Approach to validation of the LTfLL scenarios

This section describes how scenarios will be validated before being handed over for development.

4.2.1 Approach

The ideal is for each scenario to be validated by stakeholder groups beyond the Consortium at each round of scenarios: problem scenarios, solution scenarios, information and interaction scenarios etc. Pragmatically we recognise that workpackages may not have the resources to consult widely at every stage. Workpackage 3 has set up an editorial board to advise on scenarios as they are being developed. The editorial board will encourage workpackages to validate scenarios with stakeholders as a means of improving the quality of the scenarios. It will achieve this by including the following additional question on the scenario template: “List of stakeholder groups consulted, including numbers in each group”.

The editorial board will monitor the extent of stakeholder consultation taking place, and provide data and recommendations for periodic reports on validation activities.

It is important that usability of the planned user interface is considered as part of validation. This may be achieved by walk-through of a storyboard, a paper pilot, or a think-aloud heuristic evaluation of a demonstrator (see Appendix 1).

4.2.2 Decisions to be taken

Workpackage 3 will achieve agreement of a guideline on its expectations regarding the extent of validation by stakeholder groups at the Consortium autumn meeting.

4.2.3 Responsibilities

Action	By whom	When
Include validation question on the scenario template	WP3 editorial board	30 September 2009
Agree guideline on expected extent of consultation with stakeholders	WP3 editorial board	Consortium autumn meeting 2008
Monitor compliance with the guideline	WP3 editorial board	On-going

Table 2: responsibilities – scenario validation

4.3 Approach to validation of the LTfLL software products

Validation is embedded within and integral to the LTfLL software development lifecycle. Crucially, the criteria against which the software will be validated will be specified in the scenarios.

4.3.1 Timing and dependencies

For versions 1 and 2 of the software, there will be time dependencies according to the dates of delivery of scenario reports to the European Commission. Dependencies (including important items from other workpackages) are as follows:

Table 3. Timing and dependencies of validation activities

Month end relative to scenario delivery:	**-1**	**0**		**+4**	**+5**	**+6**	**+7**	**+8**	**+9**	**+10**
Decision on methods of validation										
Decision on who will collect data and who will analyse it										
Decision on training and support materials required *(WP8)*										
Scenarios submitted to EU *(WP3)*										
Completion of preparation of evaluation instruments										
Completion of beta-testing *(WPs 4-6)*										
Completion of support materials and provision of training *(WP8)*										
Handover for pilot running *(WPs 4-6 / WP7)*										
Tested software submitted to EU *(WPs 4-6)*										
Completion of pilot data collection										
Completion of analysis of data and generation of new user requirements; results passed to OUNL and UNIMAN										
Evaluation report submitted to EU										
Decisions on which recommendations to include in next version of software										

For the showcase, the same steps will apply, but on a shortened timescale.

4.3.2 Handover for pilot running

Handover for pilot running will be effected as follows. Having undertaken successful beta-testing and having corrected and re-tested any significant issues arising, the workpackage task leads for workpackages 4, 5 and 6 will inform the workpackage 7 lead and the Project Coordinator that the software has been released for pilot.

In this context, "significant issues arising" are issues likely to prevent any part of the evaluation of the software according to the planned evaluation methodology.

4.3.3 Scenarios

Stakeholders and all criteria for validation will be included in the scenarios. Each stakeholder group will have their own criteria, and validation will be planned from the point of view of each group.

The showcase will be evaluated on the basis of abbreviated scenarios based on the template in Section 5 (**Table 6. Showcase scenario template**).

The first set of full scenarios for workpackages 4, 5 and 6 will be submitted to the EU in Month 17 (July 2009) and will inform the development and validation of version 1 of the services; the second set will be submitted in Month 26 (April 2010) and will inform version 2.

4.3.4 Methods of evaluation

Appropriate methods of evaluation will be included in the validation scenarios, and will be informed by experience gained from any previous evaluation. The UsabilityNet website[8] and Appendix 1 give indications of possible methods to be applied.

Decisions on the method(s) of evaluation and which partner(s) will undertake the collection and analysis of the data will be taken together for each pilot, as these are interdependent. The former influences which partners can undertake the data processing, because of language. For example, if a method produces output in a particular language (e.g. a focus group) or requires particular technical understanding, only specific partners may be able to analyse the data. This needs careful management in view of the current distribution of funds between the partners.

Methods will be selected on the basis of:

- the nature of the data likely to be obtained: its perceived quality and its value for informing future releases of the software product

as informed by pragmatic considerations of available resources:

- the size of the group from which data might be collected
- whether data addressing the criteria can be collected automatically without human intervention

[8] http://www.usabilitynet.org/tools/methods.htm

- whether data can be collected in non-language-specific ways, e.g. by the use of questionnaires with Likert scales

Validation of the transferability of the software to different systems, organisational environments and groups of students will be validated by comparing the results of evaluations of the same services for different sites or groups of students. The mechanism whereby this will take place will be determined at the Sofia meeting.

At the same time as selecting the methods of evaluation, the sampling sizes and where appropriate, the indicators of a successful outcome will be determined. Also at this time, decisions will be made within workpackage 8 concerning training needs and support, which should be notified to UNIMAN.

4.3.5 Analysis and reporting of validation data

When methods of evaluation have been agreed, a common format for presentation of raw and processed data will be agreed for each method.

Following data analysis, the partner undertaking the analysis will pass the results to OUNL and UNIMAN. OUNL will prepare a set of recommendations to inform the next version of the software, or to inform the end-of-project roadmap. UNIMAN will feed the results and the recommendations into the next validation deliverable for the EU. Owing to limited time being available for preparing this deliverable, UNIMAN and OUNL will work concurrently on drafting the report and recommendations.

Recommendations will be referred to the Executive Board for decisions on which to action, including changes to be incorporated into the next version of the software. These decisions will be incorporated into the validation report.

4.3.6 Decisions to be taken

1. Selection of appropriate methods of validation (including sampling and indicators) for the pilots of versions 1 and 2, in conjunction with decisions on which partner(s) will be responsible for the collection and analysis of the data.
2. Agreement of the methodologies for validating transferability of the software to different systems, organisations and student groups.
3. Decisions on who will prepare the evaluation instruments for each pilot.
4. Agreement of common formats for the preparation of data.
5. Following each evaluation round, decisions are required on which recommendations to accept for action.

4.3.7 Responsibilities

Table 4. Responsibilities – validation of services

Action	By whom	Showcase	Version 1	Version 2
Completion of the showcase scenarios, incorporating the validation methods to be used for the showcase	WP 4-6 task leaders	September 2008		
Agreement of the methodologies for validating transferability of the software to different systems, organisations and student groups	UNIMAN	Consortium meeting, autumn 2008.		
Completion of the validation (and preceding) scenarios, including: - Selection of appropriate methods of validation - Decisions on who will prepare the evaluation instruments for each pilot - Decisions on who will collect data, and who will analyse the data - Decisions on training needs and support, which should be notified to UNIMAN (WP8)	WP 4-6 task leaders/WP3 in collaboration with UNIMAN and OUNL		By end June 2009	By end March 2010
Agreement of common formats for the preparation of data (method-dependent)	UNIMAN, WP 4-6 task leaders	October 2008	By end July 2009	By end April 2010
Completion of preparation of evaluation instruments	As agreed	By end Jan 2009	By end Nov 2009	By end Aug 2010
Handover of the software for live running	WP 4-6 task leaders	By end Jan 2009	By end Nov 2009	By end Aug 2010
Completion of pilot data collection	As agreed	By end March 2009	By end Feb 2010	By end Nov 2010
Completion of analysis of data and generation of new user requirements; results passed to OUNL and UNIMAN	As agreed	By end April 2009	By end April 2010	By end Jan 2011
Decisions on which recommendations in the report to accept for action	Executive Board	31 May 2009	31 May 2010	28 Feb 2011
Submission of validation report to the EU	UNIMAN	31 May 2009	31 May 2010	28 Feb 2011

4.4 Evaluation of the LTfLL approach

4.4.1 Approach

In accordance with the validation framework (Section 3), data will be sought not only from Consortium members but also from users and other stakeholders. Details of the approach will be agreed at the Consortium's autumn meeting, so the following is illustrative.

Consortium questionnaire: evaluation of the approach will be based on a questionnaire comprising 5 – 7 questions. Five of the questions will be agreed at the autumn meeting, and will be used in all Consortium evaluations of the project approach. Up to a further two questions may be included in particular instances of the questionnaire; these may be derived from a question bank or alternatively groups may be able to specify their own questions to address important current issues.

Questions will included in the areas of:

- Appropriateness of technical methods and approaches
- Effectiveness of approaches to partner collaboration
- Management of the project and workpackages

Data collection from Consortium members will take place:

- At Consortium meetings
- At interim face-to-face or on-line meetings of three or more partners concerning more than one workpackage, including the monthly Executive Committee meeting.
- Via a discussion forum in the project website, with sub-sections corresponding to the agreed questions
- Via the partners' periodic activity reports

In advance of meetings, the partner responsible for the agenda will draft a response to the five questions and up to two extra questions if (s)he wishes, including recommendations and points for discussion. The meeting will consider the response (briefly in the case of interim meetings) and will agree any changes.

The workpackage 7 lead will monitor issues emerging through the discussion forum, and will resolve them where possible, bringing remaining issues to the Executive Board for resolution.

Data collection from users and other stakeholders will take place during validation activities for versions 1 and 2 of the LTfLL services, using a bank of questions to be agreed as part of the planning for version 1.

4.4.2 Timing of reporting

A section on the evaluation of the LTfLL approach will be included in each of the three validation reports submitted to the EU.

4.4.3 Decisions to be taken

1. The Consortium questions: UNIMAN will prepare a preliminary list for discussion and agreement at the Consortium autumn meeting
2. The questions for users and other stakeholders

4.4.4 Responsibilities

Table 5: Responsibilities: evaluation of approach

Action	By whom	When
Agree Consortium questions	Executive Board (with UNIMAN)	Discussion at Consortium autumn meeting; decision by 30 November 2008
Decision on questions for users and stakeholders	Executive Board (with UNIMAN)	By 30 June 2009
Set up discussion forum and prepare guidance notes for partners	UNIMAN lead (with OUNL)	By 30 November 2008
Production of monitoring reports for Executive Committee based on discussion forum	UNIMAN	As required
Inclusion of time on meeting agendas; preparation of response to the question set	Partner responsible for agenda	Each meeting
Chairing and minuting session at Consortium meetings	UNIMAN	Each meeting
Preparing a section on evaluation of the LTfLL approach for the three validation plans for the EU	UNIMAN	(showcase) By end April 2009 (version 1) By end April 2010 (version 2) By end January 2011

SECTION 5: Pilot Planning Framework

5.1 Introduction

This section describes the overall framework for the planning of each pilot.

The reader should note that one aspect of pilot planning, i.e. training and support materials, falls outside the scope of this document. Planning for training and support is included in the Workpackage 8 deliverable "Dissemination and training strategy" due to be delivered to the EU at the same time as this document. The relevant deadlines for training and support matters are however included in Section 7 of this document ("diary of activities and milestones").

5.2 Pilot and evaluation planning – introduction

The aim of the pilots is to test and validate the main concepts developed in the scenarios, and implemented in the LTfLL services, concerning functionality and usability within an educational environment ('Proof of concept'). In total three pilot rounds are planned:

- The first 'showcase' pilot integrates the current state-of-the-art and the technologies the partners bring into the project within the framework of an educational use case and showcase scenario. This showcase scenario will be piloted with a limited number of users (up to a maximum of 5).
- The second pilot is based on enhanced services that incorporate the results of the evaluation and pilot activities in the showcases. Again, these services will be tested with a limited number of users.
- The third pilot integrates all previous evaluation results into a final version of the services, which will be tested with a minimum of 25 users. The validation results of this pilot will lead to a roadmap, describing how the results achieved can be sustained and proposing new directions for research.

The results of these rounds are improved services and, if necessary, additional user requirements. Proposed dates for activities contributing to the pilots are given in Section 7.

5.3 The first pilot round ('showcase')

In order to organise the first round of pilots, based on available state of the art technology, project partners were asked to fill in a form to describe the showcases and their setup, derived from their use cases and scenario descriptions. These forms supplemented the use case and scenario descriptions, enabling partners to plan, conduct and evaluate pilots and communicate about them to the other partners. Important criteria for the selection of the tools to be used within these showcases were:

- They demonstrate how language tools can be used in an educational setting, illustrating the use of (parts of) the tools to be developed within the LTfLL project
- Conditions, infrastructure and pilot groups, including the chosen domain and language for testing, should be available
- The pilots should use existing and stable language tools, to keep the research and development efforts to a minimum for the first round.

The following table fields were specified:

Table 6. Showcase scenario template

Showcase scenario template	
The objectives	A short description. Write down what you want to demonstrate (and if relevant/possible how this supports or illustrates the future development) and the main actor(s). Limit yourself only to those actors who are supposed to be active during the pre-pilot testing.
The Narrative	A short narrative story of the user tasks and the possible sequences of actions of the user and the system. In this showcase, phase the narrative points briefly to the educational problems and to the direction of one possible solution. We are using a kind of mock-up, so describe which parts of the sequence and/or data exchanges are covered by the tool and which will be simulated.
Interactions(s)	Describe the interaction only from the perspective(s) of the actor(s) involved in the testing.
Used technology and services	Describe the tool(s) and their configuration to be used in the pre-pilot testing – remember this can be a mix with even paper-based parts.
Testing opportunities and requirements	Describe what is needed for the testing (infrastructure, the test environment, resources, domain, language and as far as possible: the test group, the type of data you want to collect and the test methods) and where the testing will be done. Define a small Plan of Action to ensure that the testing process goes as planned.

Two showcase pilots are described in more detail in Section 6, using this template. We will report on the other showcase scenarios in the LTfLL-project deliverable 3.1.

Data derived from the preliminary showcase scenarios has led to the following overview of showcase pilots. All pilots aim to end in March 2009.

5.4 Showcase pilots

Table 7. The showcase pilot overview

Institution	**Language**	**Subject Domain**	**Showcase**
Eberhard Karls Universität Tübingen (UTU)	English	Medicine	Task 4.1: Tool for positioning the learner in relation to a specific learning goal and a collection of learning materials
The University of Manchester (UNIMAN)	English	Medicine	Task 4.2: Tool for investigating the conceptual development of the learner through analysis of text language output
Polytechnica University of Bucharest – National Center for Information Technology (PUB-NCIT)	English	IT	Task 5.1: Generate Feedback for Collaborative Chat Conversations
Université Pierre-Mendès, Grenoble (UPMF)	French	IT	Task 5.2: Using students' summaries of course texts to assess their learning

Institution	Language	Subject Domain	Showcase
Institute of Parallel Processing of the Bulgarian Academy of Sciences (IPP-BAS)	Bulgarian English	IT IT, Medicine	Task 6.1: Supporting the tutor in preparing his/her teaching course
Universiteit Utrecht (UU),' Polytechnica University of Bucharest – National Center for Information Technology (PUB-NCIT)	English	IT	Task 6.2a: Comparison and integration of social tagging with domain ontologies Task 6.2b: Managing the social learning network

5.5 The second and third phase pilot rounds

The definition and setup of the second and third phase pilots depends on a choice from the scenarios contributed by partners and delivered from WP3, based on quality assurance and selection criteria. Once this selection process has completed, in order to ensure completeness in the pilot design, the following form will be used to plan the pilot:

Table 8. Second and third phase pilot setup form

Name	(reference number and name of the scenario to be tested)
Aim	(general scenario aim)
Version	(second or third pilot iteration envisioned in LTfLL)
Pilot partner(s)	
Requirements, planning and milestones • software availability/installation/ configuration/testing, • acquire participants, • participant instruction • materials needed • collect data (+ description of used methods), • analyse data (+ description of used methods), • report data	
Outcomes	
Validation requirements	
Software used	
Technical architecture overview/diagram	

SECTION 6: Pilot Planning for Showcase

6.1 Introduction

In general there will be one showcase pilot for each work package task. This section describes one finalised showcase pilot scenario. The full set of showcases forms part of deliverable D3.1, due in Month 8.

6.2 Sample showcase pilot – work package 5 task 2

Tool:	Using students' summaries of course texts to assess their learning
Language:	French
Domain:	IT (negotiation of more precise content in progress, depending on students' availability: either NLP or ICT applications for teaching, or educational research on ICT). So the alternatives are indicated in the scenario.
Number of students:	5
Topic:	Text reading and summarization about a chosen topic and LSA-based assessments allowing self-regulation processes of comprehension.
Period:	1 hour

Table 9. Showcase planning for work package 5 task 2

Pre-pilot scenario	5.2. – Generate feedback for students' course summaries.
The objectives	The main objectives of this pre-pilot test are as follows: • to check the usefulness of a broader educational scenario, where learners' self-assessments on the comprehension of course texts are combined with automatic feedback on their textual summaries. • to collect in a controlled setting information on what kind of support and feedback is required by students during that task execution (finding some lines on how to generate human-like feedback). • to evaluate what can already be used with the current stage of the tools and what should be developed. The main actors are: • students • a teacher
The Narrative	1. Students attending an IT course (see comment above on topic) have to find out more information about the topics of the course not very well understood. 2. They could type queries in Google to retrieve course material on these domains and write out memos, they also could chat in order to help each other understanding the different topics of the course. They also could ask their teacher to propose such texts. These alternatives lead to several problems: students can be overwhelmed by the huge amount (and the diversity as well) of the retrieved texts; these texts could be off-the-subject; there is no tracking on the successive queries of the students; the teacher could be too busy in both proposing new texts and assessing student's understanding of these texts; students couldn't possibly be expert enough in domains not well understood by their peers to provide them with adequate help.

	3. Our aim is to propose a system that addresses the latter problems by allowing students to: a. perform queries within a preselected set of course texts; b. read the course texts and evaluate their understanding themselves; c. write out summaries of these texts; d. be given an assessment about how well the written summaries are related to the source texts; e. be given new texts whose difficulty matches their level of understanding; f. be given questions that foster their level of understanding. 4. Separately, each of these solutions has already been implemented. Our aim is to validate the whole process in which students could be immersed. 5. This whole process allows students to be more aware of the notions they attempt to grasp (self-regulation processes) and lessens the teacher's cognitive load during her assessment task. The latter can focus only on higher levels of student's activity.
Interactions(s)	**Overall interaction:** Student: I'm working alone with the system, but I also need help from my teacher to get hints and suggestions on queries to perform, ideas to type for summarising, and so on. Teacher: I'm working in looking over each student's shoulder, for help on demand. I also manage the whole learning session (time, content, assessment supervision). **A session in detail, from the student viewpoint:** 1. The teacher suggests to me to use Apex 2, software that assesses to what extent summaries are close to their source texts, and refers that assessment to my comprehension of these texts. 2. First of all, I ask myself what were the notions I didn't grasp well: in that case, I found problems in understanding the way Latent Semantic Analysis functions. So I type a query in order to retrieve a set of texts (from a course texts database/from the proceedings of an educational research conference) to have a better understanding of that (e.g. the query "Latent Semantic Analysis"). If necessary, my teacher helps me in reformulating my query. I'm then engaged in a reading loop wherein I carefully read the texts retrieved by the query and try to grasp them. I can ask my teacher for help. If required (by myself or by my teacher) I can refine my query and be given more adequate texts, e.g., "Latent Semantic Analysis process". Once each text is read, I'm prompted to judge whether I have understood it or not; again, my teacher can give some advice about that, in rereading my summary. As the first set of texts has been completely read, I either can read some more texts or be engaged in a second (writing) loop. In the latter one, I have to write out a summary for each of the texts that were judged as well understood: for instance, I found it difficult to understand the Singular Value Decomposition. Eventually, the summaries are compared with their source texts and comments about judgment discrepancies between myself and Apex are highlighted (e.g., the computer prompts "You said you could summarise the text about the cognitive implications of LSA but apparently you cannot"). I then review the summary about that notion to make it more adequately match with the source texts; I also ask the teacher for new terms to type in the query for refining that notion (as a result, well-understood texts will be no longer presented in the next reading loops whereas bad-understood texts could be). 3. At each step of this session, my teacher provides guidance to the student (helps refine queries, answers additional questions, provides comments).

	4. Once one reading-writing whole loops have been performed, I can stop working. It is noteworthy that, in our target system, steps 1 to 5 are likely to be processed as many times as required by the students. For the pre-pilot test, the whole duration of the session will be of 1 hour; so only one reading-writing loop is expected to be performed by the participants during the pre-pilot test. Moreover, in our target system, the way the teacher gets or communicate information from/to the student will be partly computer-based.
Used technology and services	• Apex 2 (LSA on top of C routines plus html/php interface)[9]. See Dessus and Lemaire (2002) for more information. • Corpus of French course texts on IT (either NLP or ICT applications for teaching, or educational research conference). • Corpus of French texts (e.g., newspaper, novels) to get general knowledge of word usage.
Testing opportunities and requirements	**Test environment:** • Apex 2 • domain: IT • language: French **Resources**: (see also previous section) • two computers with Apex 2 installed; **Where the testing will be done:** Either at UPMF university or at the Teacher Training Institute of Grenoble or at Stendhal university. **Plan of action (pre-pilot only):** 1. Decide more precisely the domain and the participants. 2. Find out the corresponding corpus of course texts. 3. Perform Apex 2 sessions with 5 students and record them. 4. Define the architecture of the system and the technologies that are used 5. Questionnaires assessing students' satisfaction and cognitive load (adapted version of the NASA-TLX) 6. Further analyses of the recorded sessions (e.g., number of: queries, texts read, judged as understood or not). **Broader plan of action (for the target system).** 1. Definition of the architecture of the system 2. Corpora selection 3. Apex 2 update according to current technologies. 4. Integrate a module simulating student's understanding (CI-LSA) 5. Integrate a module for helping teachers manage their annotations 6. Integrate a module for detecting macrorule use and providing specific advice to students. 7. Integrate a module for providing questions aimed at fostering student's understanding of the course.

[9] Dessus, P. & Lemaire, B. (2002). Using production to assess learning: an ILE that fosters Self-Regulated Learning. In S. A. Cerri, G. Gouardères, & F. Paraguaçu (Eds.), *Intelligent Tutoring Systems (ITS 2002)* (pp. 772-781). Berlin: Springer, LNCS 2363.

SECTION 7: Diary of Activities and Milestones

This section gives a diary in table format of pilot and validation activities, including related activities in other workpackages and milestones, to enable ready monitoring of activities. Milestones are taken from the LTfLL Description of Work.

Dates are indicative at this stage and subject to change by agreement with the WP7 lead.

Table 10. Diary of activities and milestones

Task	Lead partner	Contributors	Date
Validation planning - start-up			
Inclusion of validation question in scenario templates	WP3 editorial board		30 September 2008
Agree guideline on extent of consultation with stakeholders during development of scenarios	WP3 editorial board		Consortium autumn meeting
Agree common approach to validation of educational/organisational transferability	UNIMAN	All other partners	Consortium autumn meeting
Agree Consortium questions for evaluation of the LTfLL approach	UNIMAN	All other partners	Discussion at Consortium autumn meeting; final agreement by 30 November 2008
Evaluation of LTfLL approach: set up discussion forum and prepare guidance notes	UNIMAN	OUNL	30 November 2008
Pilot planning	OUNL	Potential pilot institutions	Between November 2008 – March 2009
"Showcase" phase			
Completion of showcase scenarios, including validation methods (WP3)	AURUS	WP 4-6 members	September 2008
Decisions on training and support required (WP8)	UNIMAN	WP 4-6 members	September 2008
Agreement of common formats for presentation of data	UNIMAN, WP 4-6 task leads	WP 4-6 members, OUNL	October 2008
Preparation of deliverable 3.1: Use cases, sample scenario, inventory and assessment of existing tools (WP3)	OUNL	WP3 members	**MILESTONE**: Submission to EU 31 October 2008.
Refinement of validation plan ready for first pilot (at and around the Sofia meeting) Agreement of questions for evaluation of the LTfLL approach. Agreement of approach for validation of transferability between systems,	UNIMAN	WP7	30 November 2008

Task	Lead partner	Contributors	Date
organisations and different student groups.			
Alpha-testing of 'showcase' version (WPs 4 – 6)	WP 4-6 task lead	WP 4-6 lead, WP 4-6 partners	As agreed within the WP concerned
Completion of beta-testing of 'showcase' version (WPs 4 – 6)	Pilot institutions to be determined		31 December 2008
Completion of preparation of evaluation instruments	As agreed	As agreed	31 January 2009
Completion of support materials and provision of training (WP8)	UNIMAN	WP8 members	31 January 2009
Handover of software for live running (WP 4-6)	WP 4-6 task leads		31 January 2009
Completion of integrated 'showcase' version (WP2): Submission to EU	BIT-MEDIA (learning tools) WUW (NLP tools) OUNL		 **MILESTONE**: 28 February 2009.
Evaluation of LTfLL approach: preparation of questions for inclusion in validation questionnaires	UNIMAN		28 February 2009
Completion of first pilot (showcase) and collection of validation data	Pilot institutions to be determined	UNIMAN, OUNL	31 March 2009
Completion of analysis of validation data Completion of recommendations on revised user requirements	UNIMAN OUNL	Pilot institutions	30 April 2009
Decision on which recommendations to carry forward to version 1	Executive Board	WP 4-6 task leads	31 May 2009
Completion of evaluation report 1 Submission to EU	UNIMAN OUNL	OUNL, pilot institutions	Internal review starts 17 May 2009 **MILESTONE**: Submission to EU 31 May 2009.
'Version 1' phase			
Decisions on: • appropriate methods of validation • who will prepare validation instruments for each pilot • who will collect data • who will analyse data • training and support required (WP8)	UNIMAN	WP 4-6 task leads	30 June 2009
Decision on the bank of questions for stakeholder evaluation of the LTfLL	Executive Board	UNIMAN, WP7	30 June 2009

Task	Lead partner	Contributors	Date
approach			
Preparation of deliverable 3.2: First set of educational scenarios (WP3)	OUNL	Partners contributing scenarios, OUNL	**MILESTONE**: Submission to EU 31 July 2009.
Agreement of common formats for presentation of data	UNIMAN, WP 4-6 task leads	WP 4-6 members, OUNL	31 July 2009
Alpha-testing of version 1 (WPs 4 – 6)	WP 4-6 task lead	WP 4-6 lead, WP 4-6 partners	As agreed within the WP concerned
Completion of preparation of evaluation instruments	As agreed	As agreed	30 November 2009
Completion of beta-testing of version 1 (WP 4-6)	Pilot institutions to be determined		30 November 2009
Handover of software for live running (WP4-6)	WP 4-6 task leads		30 November 2009
Completion of support materials and provision of training (WP8)	UNIMAN	WP8 members	30 November 2009
Integration of version 1 (WP2) Submission to EU	BIT-MEDIA? WUW?		**MILESTONE**: Submission to EU 31 December 2009.
Completion of version 1 pilot and collection of validation data	Pilot institutions to be determined	UNIMAN, OUNL	28 February 2010
Completion of analysis of validation data Completion of recommendations on revised user requirements	UNIMAN OUNL	Pilot institutions	30 April 2009
Decision on which recommendations to carry forward to version 3	Executive Board	WP 4-6 task leads	31 May 2009
Preparation of evaluation report 1	UNIMAN	OUNL, pilot institutions	**MILESTONE**: Submission to EU 31 May 2010.
'Version 2' phase			
Decisions on: • appropriate methods of validation • who will prepare validation instruments for each pilot • who will collect data • who will analyse data • training and support required (WP8)	UNIMAN	WP 4-6 task leads	31 March 2010
Preparation of deliverable 3.3: Updated educational scenarios (WP3)	OUNL	Partners contributing scenarios, OUNL	**MILESTONE**: Submission to EU 30 April 2010.
Agreement of common formats for presentation of data	UNIMAN, WP 4-6 task leads	WP 4-6 members, OUNL	30 April 2010

Task	Lead partner	Contributors	Date
Alpha-testing of version 2 (WPs 4 – 6)	WP 4-6 task lead	WP 4-6 lead, WP 4-6 partners	As agreed within the WP concerned
Completion of preparation of evaluation instruments	As agreed	As agreed	31 August 2010
Completion of beta-testing of version 2 (WP 4-6)	Pilot institutions to be determined		31 August 2010
Handover of software for live running (WP4-6)	WP 4-6 task leads		31 August 2010
Completion of support materials and provision of training (WP8)	UNIMAN	WP8 members as agreed	31 August 2010
Integration of version 2 (WP2) Submission to EU	BIT-MEDIA? WUW?		**MILESTONE**: Submission to EU 31 October 2010.
Completion of version 2 pilot and collection of validation data	Pilot institutions to be determined	UNIMAN, OUNL	30 November 2010
Completion of analysis of validation data Completion of recommendations on revised user requirements	UNIMAN OUNL	Pilot institutions	31 January 2011
Decision on which recommendations to carry forward to roadmap	Executive Board	WP 4-6 task leads	28 February 2011
Preparation of evaluation report 3	UNIMAN	OUNL, pilot institutions	**MILESTONE**: Submission to EU 28 February 2011.

APPENDICES

Appendix 1: Methodologies

Appendix 2: Pedagogic/Andragogic Effectiveness and Need

Appendix 3: Usability, User Acceptance and Efficiency

Appendix 4: Transferability between Different Systems and Situations

Appendix 1: Methodologies

This table provides a preliminary list of methodologies.

Methodology	Suitable for criteria:	When to use	Sampling	Indicators	Comments
Paper prototype	Validation of interaction scenarios	With different stakeholder groups during user interface design.	1 – 3 representatives of each stakeholder group	Qualitative data indicating areas of difficulty in using the software and missing/erroneous functionality	
Demonstrator in conjunction with think-aloud evaluation (see below)	Validation of interaction scenarios	With different stakeholder groups during user interface design.	1 – 3 representatives of each stakeholder group	Qualitative data indicating areas of difficulty in using the software and missing/erroneous functionality	Quickly developed proof of concept software providing key functionality to be tested
Heuristic evaluation *Comparison of software with use cases/other listed criteria*.	Usability	For expert reviewers in the project team to identify problems for correction prior to exposing users to the software.	Minimum one project team member per service	Qualitative data indicating areas of difficulty in using the software and missing/erroneous functionality	
Think-aloud evaluation *At-elbow walk-through of software in which user describes his/her thought processes.*	Usability	Where an in depth investigation with small numbers of users is required.	Ideally 3 – 5 users per LTfLL service, selected randomly	Qualitative data indicating areas of difficulty in using the software and missing/erroneous functionality	Very effective. Individual face to face session required for each user, so time-intensive.

Methodology	Suitable for criteria:	When to use	Sampling	Indicators	Comments
Questionnaire	Pedagogic/andragogic effectiveness Transferability – technical, organisational	Where quantitative data collection (with supporting comments) is required from large numbers of stakeholders.	20+	Quantitative data – targets to be achieved can be set for each question or group of questions. Also qualitative data.	Effectiveness depends on selection of questions. Should be preceded by a focus group to fine-tune the questions
Focus group	Validation of scenarios Pedagogic/andragogic effectiveness Transferability – technical, organisational	Where an in depth investigation with small numbers of stakeholders is required.	Ideally 5 – 8 stakeholders	Qualitative data addressing the questions asked.	Very effective at identifying unsuspected issues. Ideally should be taped and transcribed. Can assist in decision-making through including nominal group or Delphi techniques.
Interview	Validation of scenarios Pedagogic/andragogic effectiveness Transferability – technical, organisational	Where an in depth investigation with a single stakeholder is required. Ideal for management where there may be a single individual in a particular role.	One stakeholder	Qualitative data addressing the questions asked.	Very effective at identifying unsuspected issues. Ideally should be taped and transcribed.

Methodology	Suitable for criteria:	When to use	Sampling	Indicators	Comments
Action research	Pedagogic/andragogic effectiveness User acceptance and usability	Technology that has progressed beyond the prototype stage	No 'one size fits all' answer; the number must be sufficient to constitute a robust test but not too big for thorough evaluation or to expose a large population to unproven technology	Quantitative: Uptake, usage, user reactions to the technology Qualitative: In-depth evaluation of the experience of using the technology and any specific pedagogic issues	An essential bridge between prototyping and implementation; an experimental method that is very well suited to evaluation of education technology
Unified Theory of Acceptance and Use of Technology (UTAUT)[12]	User acceptance	The model provides a theoretical framework and suggested questions for use in questionnaires, focus groups and interviews.	See questionnaire, focus group and interview above.	See questionnaire, focus group and interview above.	The questions can be used quantitatively as Likert questions or qualitatively.
Accuracy of web-page output	Transferability - technical	For expert reviewers in the project team to identify problems for correction prior to exposing users to the software.	Framework of the web pages	Qualitative data indicating areas of difficulty in using the software and missing/erroneous functionality	To ensure software is displayed correctly to the end-user (through W3C validator). Also for disabled persons.
Evaluation of 'calculation' speed	Transferability – technical*	For expert reviewers in the project team to identify problems for correction prior to exposing users to the software.	Some test routines	Quantitative data	Ensure calculation speed doesn't slow down on different systems and situations.

Methodology	Suitable for criteria:	When to use	Sampling	Indicators	Comments
Test collection evaluation	Validation of data sets for IR-effectiveness evaluation – technical* **(IR = information retrieval)**	Before using large text corpora as input for IR tasks.	1000+ learner text documents of different origins Queries and human relevance judgments	Quantitative data indicating in-/appropriate data sets	Data driven approach Answers the question: how good is a test collection? Identification of ideal number of queries and assessors
IR-effectiveness evaluation **(IR = information retrieval)**	Validation of IR algorithms – technical*	When an implemented IR-algorithm has to be tested.	Positively evaluated test collection Computed relevance judgments	Quantitative measures (precision, recall, f-measure, ranked half life, etc.); following TREC standards	How good is the IR algorithm? Compare different approaches (LSA, etc.) Depends on test collection quality

* Note: Technical issues will be mainly validated in the concerning WP, i.e. WP2, WP4, WP5 or WP6.

Appendix 2: Pedagogic/Andragogic Effectiveness and Need

A2.1 Introduction

The overall aim of the LTfLL project is to produce services to address the themes described in the LTfLL Description of Work (DoW)[10]. The DoW specifies that the pedagogic/andragogic effectiveness and need of the services will be validated. This appendix provides an introduction to the evaluation of pedagogic/andragogic effectiveness and need.

The distinction between pedagogy (the art and science of teaching children, or of being a teacher) and andragogy (the art and science of how adults learn) was originally expressed by Knowles[11], who noted the different ways in which children and adults learn. Today, pedagogy is generally understood to mean instructor-led learning (i.e. teaching), and andragogy to mean facilitated self-directed and/or experiential learning. The purpose of LTfLL is to develop tools for lifelong learning, which requires an understanding of adult learning (andragogy). In contemporary pedagogy/andragogy, emotional learning is becoming important, i.e. how learners learn about themselves, their reactions and how they relate to and respond to other people.

In the LTfLL project, pedagogic/andragogic effectiveness and need are expressed within the scenarios as the desired outcome or value to the stakeholders. Each scenario is expected to relate its anticipated outcome(s) to contemporary pedagogy/andragogy. Accordingly, 'pedagogic/andragogic effectiveness and need' will be primarily validated with reference to how well the services meet the stakeholder 'values' or 'problems' concerning pedagogy/andragogy as expressed in the scenarios. However, stakeholders may not express their needs fully in the scenarios, and their needs may evolve over time. Accordingly, the validation approach must be sensitive to changing needs.

The indicators of pedagogic/andragogic effectiveness will emerge from the scenarios, but are expected to include:

- the accuracy of the output of the LTfLL service compared with that of domain experts
- the perception of domain experts and students concerning whether the services meet their pedagogic/andragogic requirements
- the effectiveness of learning, i.e. the extent to which the services can foster learning, by comparing the learning progress of students using the services with students using a more traditional approach. This will consider the value of services in supporting cognitive and social development and shared knowledge. This may include the response of the users to the services in terms of whether learner performance and/or behaviour have changed.

A2.2 Objectives

The objective of validating pedagogic/andragogic effectiveness is to establish the extent to which the services address stakeholder values or problems with regard to learning and teaching. Validation will be primarily with reference to the values or problems expressed in the scenarios, as revised to take account of the changing needs of stakeholders.

[10] LTfLL Description of Work (drafting date 19 November 2007), page 4
[11] Knowles, M. S. (1968). Androgogy, not pedagogy! *Adult Leadership, 16*, 350-352, 386.

Appendix 3: Usability, User Acceptance and Efficiency

A3.1 Introduction

The Unified Theory of Acceptance and Use of Technology (UTAUT) model[12] is a synthesis of eight prior models of user acceptance of technology. The model proposes that the following four constructs are direct determinants of user acceptance and behaviour:

1. Performance expectancy:	the degree to which an individual believes that using a system will help him/her achieve gains in job performance
2. Effort expectancy:	the degree of ease of use associated with the system
3. Social influence:	the degree to which the individual perceives that important others believe that he/she should use the new system
4. Facilitating conditions:	the degree to which an individual believes that an organisational and technical infrastructure exists to support use of the software

Within the scope of the LTfLL pilots, social influence and facilitating conditions will be somewhat artificial compared with those for mature software in full production; however, performance expectancy and effort expectancy can be evaluated more realistically. Both of these concern user perceptions of usability.

The international standard ISO 9241-11 defines usability as:

The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.

Implicit in the definition are the possibly different perceptions of usability by (1) different stakeholder groups, (2) similar stakeholder groups in different national and institutional contexts, and (3) when undertaking different tasks.

The literature shows many concepts associated with usability. ISO 9241-11 concerns the three aspects of effectiveness, efficiency and user satisfaction, defined as follows[13]:

- Effectiveness - can users complete tasks, achieve goals with the product, i.e. do what they want to do?
- Efficiency - how much effort do users require to do this? (often measured in time)
- Satisfaction – what do users think about the products' ease of use?

Jakob Nielsen (1994)[14] describes five aspects of usability:

- Learnability: How easy is it for users to accomplish basic tasks the first time they encounter the design?
- Efficiency: Once users have learned the design, how quickly can they perform tasks?

[12] V. Venkatesh, M.G. Morris, G.B. Davis & F.D. Davis (2003), "User acceptance of information technology: towards a unified view", MIS Quarterly, 27(3), 425-478
[13] www.usabilitynet.org
[14] J. Nielsen (1994), "Usability Engineering", Morgan Freeman, San Francisco ISBN 0-12-518406-9

- Memorability: When users return to the design after a period of not using it, how easily can they re-establish proficiency?
- Errors: How many errors do users make, how severe are these errors, and how easily can they recover from the errors?
- Satisfaction: How pleasant is it to use the design?

ISO/IEC 9126 includes 'attractiveness' within usability, and includes 'resource efficiency' as well as 'time efficiency'. It also includes 'reliability', defined as maturity, fault tolerance, recoverability and availability.

The aspects of usability that LTfLL will probe in its validation will include the above as well as any additional usability-related requirements derived from the scenarios.

A3.2 Objectives

The objective of evaluating user acceptance and usability is to obtain formative feedback, which will be used to the next major or minor version of the product.

The formative evaluation of user acceptance may consider aspects arising from the UTAUT model, which lists a number of items as a basis for Likert-style questions.

The formative evaluation of usability may also consider the following high level questions, as informed by the scenarios.

- How effective is the product in enabling users to complete the tasks identified in the validation scenarios?
- How efficient is the product, in terms of how long it takes the users to complete the tasks in the validation scenarios?
- How efficient is the product, in terms of system response times?
- How intuitive is the product, in terms of users completing the tasks defined in the validation scenarios unaided and without errors?
- How easy is it for users to recover from errors?
- How satisfied are the users with the ease of use of the product?
- How satisfied are the users with the appearance of the product?
- How satisfied are users with the reliability of the product?

Appendix 4: Transferability between Different Systems and Situations

A4.1 Introduction

There are several aspects of transferability of software (sometimes known as 'generalisation of service'):

- the extent to which the software will run in other technical environments, e.g. on different operating systems
- the framework for interoperability with other related pieces of software, e.g. interfaces with institutional e-portfolio systems
- whether the software can be used by different user groups, including accessibility issues
- whether the software can be tailored for use in different educational and institutional environments

Technical aspects of transferability: Natural language processing services based on latent semantic analysis bring restrictions regarding memory and calculation power of the servers on which they are run. Corpus size, preprocessing activities, and the core algorithm (singular-value decomposition) are the main factors restricting their applicability.

To ensure transferability and interoperability of all services within the project it is strongly recommended that existing standards are adhered to. Looking more closely, we can break down the question of transferability into three different levels:

6. *Data level*
    - Data formats (handling and conversion of different data formats)
    - Storage (limited storage capabilities)
    - Memory (limited memory capabilities)
    - Operating systems (services must run on different operating systems)
    - Calculation routines (optimised design of calculation routines, separation of pre-calculation and folding-in)

7. *Service level*
    - Encapsulation of generic services (object-oriented approach)
    - Precondition for re-use in different applications
    - Corpus size for LSA routines (different corpora, transferability across domains)
    - Communication protocols

8. *Presentation level*
    - Widgetised user interface integration (for use on different user interfaces, e.g. different browsers, different end-user presentation tools, e.g. computers, PDAs, mobile phones)
    - Accessibility for disabled people
    - Availability of integration into existing e-learning environments

Transferability to different institutional and educational environments: It is important that the LTfLL software is effective in a wide range of institutional and educational environments. It cannot be assumed that software developed for one institutional environment will run without tailoring in another institution or even if a different department

within the same institution. The role and responsibility of students and tutors, type of education, educational model followed, privacy, availability of learning materials may all differ. Similarly, students differ in age, background, educational level and aspirations, and in many other ways.

Tailoring to different environments may be achieved through providing user-configurable parameters within the software. The LTfLL project will evaluate the roll-out process for the major releases, to identify and analyse local organisational issues affecting implementation and the response of different student groups to the software. The project will also identify different student groups whose response to the software will be compared. These activities will provide formative feedback to the development team, regarding aspects of the software to be parameterised to allow tailoring to different environments.

A4.2 Objectives

The objectives of the validation of transferability are to ensure that the software works acceptably in the following circumstances, and to provide formative feedback where it does not:

- when run on, or interfacing with, different technical environments
- when used in different organisational environments
- when used by different groups of users.