English Language and Information Technology

Instructor : Dr. Abdullah Al Fraidan

LECTURE 1

CALL, NLP, Corpus Linguistics

What is CALL?

CALL = Computer Assisted/Aided Language Learning.
For the purposes of this course we take CALL to embrace any computer software that is usable in some way to help language learners, whether intended for that purpose or not, and whether directly used by them, or used by someone else to create a conventional material (e.g. a coursebook) which learners use.

Though the acronym "CALL" implies a limitation to language learning, we do not, as some do, distinguish that from computer aided language acquisition (CASLA). And we include in our scope language use by learners, and of course language teaching. Computer aided language testing (CALT) is often discussed separately from CALL, and for various reasons will not be much focused on in this course (lack of time and lack of the software!). We are also excluding use of computers in AL and ELT research in general (CASLR), and in the learning of linguistics rather than language (though there is an unclear borderline here, as much language teaching involves teaching about language, especially grammar, or raising awareness of language forms, and so resembles simple linguistics).

There are many other acronyms and terms around with broader scope than CALL, or scope overlapping with CALL. They refer to areas of theory and research which have implications for CALL: e.g. CAL, CAI, CBE, TELL, Telematics, HCI, AI, NLP, Corpus Linguistics. On these neighbouring areas see Chapelle 2001 ch2 and Levy 1997 ch3 and pp77-82.
CALL 'tasks' include what may be otherwise referred to as games, exercises, activities, materials, even tests, and just 'ordinary use' of facilities like word processing. Sometimes they are fully determined by the program, sometimes they are largely in the hands of the teacher or learner using the software. They may be done in class or at home, etc.

Thinking about CALL means thinking about many of the same things one considers when thinking about 'materials' for language learning/teaching (coursebooks, visual aids like posters or videos, pen and paper exercises, dictionaries etc.). Both involve something physical that teachers and learners use alongside a teaching method, syllabus etc. in a taught program OR which may be just used independently by the learner. Both have to be bought (or pirated). Both have a tangible form, but at the same time when exploited form part of a less tangible 'task' or the like. This parallel leads us to the conclusion that there are three main areas of concern (see Hubbard 1996 in ed. Pennington The Power of CALL ***for a fuller exposition, attempting to relate this to the Richards and Rodgers framework for analysing teaching methods):***

1) Development/creation. I.e. the principles and processes of writing software or authoring new materials within some existing software (Cf. Chapelle 2001 p166ff, and Levy 1997 ch4 onwards (esp. p104-108), for concepts rather than practicalities). Compare materials development, course book writing.
2) Use/implementation. I.e. how teachers use software with their learners (in or out of class, individually or in groups, for what sort of tasks, integrated with other aspects of the teaching-learning process or not, etc. etc.)… and how the learners use the software (which may be differently from how the teacher plans, or indeed entirely independently of school), their processes and strategies. Compare discussion of the role of materials like coursebooks or tapes in a course, different 'task types' they can be involved in, learner use of materials like dictionaries or cribs out of class unknown to the teacher etc… (Levy 1997 Ch4 onwards touches on ideas about Use repeatedly, esp p100-103; Jones and Fortescue ch14 old but practical)
3) Evaluation. I.e. how to decide what is good or bad software…. including inevitably considering what is a good or bad use of the software. Compare materials evaluation. (Chapelle 2001 Ch3).

HISTORY OF CALL

In terms of the development of hardware, program types, relation to ideas about language learning and teaching... This is filled out in class. See also Chapelle 2001 ch1 and Levy 1997 ch2 and the online http://www.history-of-call.org/

- The computer-as-big-as-a-room era. Entire courses like that of PLATO organised at a few universities. Audio-lingualism.
- The arrival of the home/school computer (Sinclair, Apple, BBC). CALL tasks as ancillary, and produced by many small publishers such as WIDA and even teacher enthusiasts. Attempts to fit it in with the Communicative approach.
- The era of the powerful PC (and Mac). Professionalisation of software writing but lack of transfer of much software from earlier platforms.
- PC + CD, multimedia. Software out of the hands of teachers, largely audio-lingual in mode. New attempts at entire courses.
- The era of the Internet. Teacher as selector. Learner-centred.
- The future: convergence of media and ‘omnimedia’
- Social networking?

LECTURE 2

UUEG Software (Azar Interactive)

UUEG Software

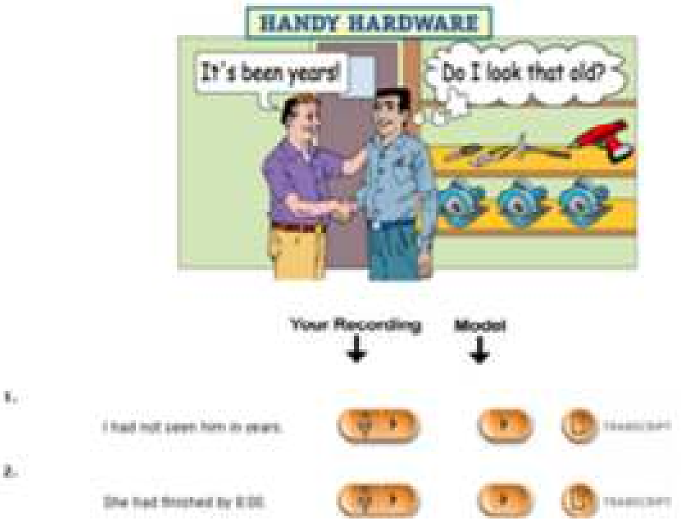
http://www.azarinteractiveonline.com/tour/

Evaluation of UUEG

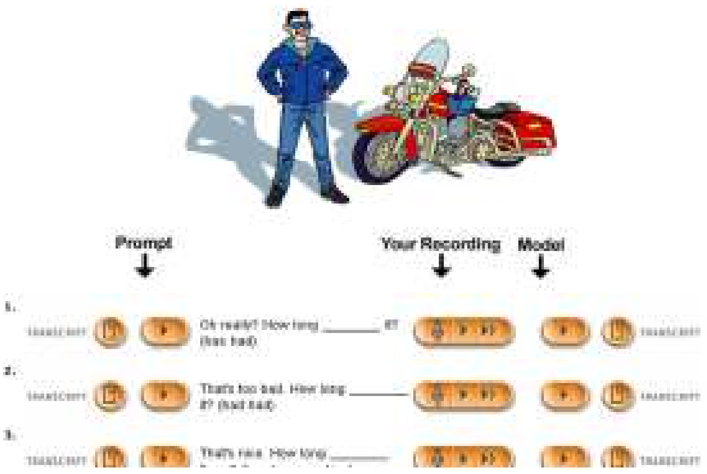
Before beginning the evaluation itself, it is necessary to give a brief description of the software, which is based on Betty Azar's book (2009). Due to space restriction, I will only provide an analysis of just one chapter of the book with intercepted description of the methods used in implementing the software in classroom. The analysed chapter is divided into four parts, each focusing on the following tenses: the present perfect, the present perfect progressive, the past perfect, and the past perfect progressive. Each section includes several quizzes, exercises and one crossword game, and these are followed by three main tasks covering listening, speaking and reading comprehension (named by myself). To finish, there is a test that enables students to assess their achievements.

Analytically speaking, the chapter follows Ur's framework (1988) for teaching grammar: presentation, explanation, practice, and test. The chapter starts with a preview of the tense, comparing it to, and/or contrasting it with, similar tenses – a method that is claimed to be effective by Walker (1967). Learners can either read or listen to the preview before examining a chart that exemplifies the tense. Following this, students are presented with a range of nearly all the typical mechanical drills, such as gap filling, error recognition, cloze, and multiple choices. Some of the quizzes come with animated pictures, and the exercises are represented in a linear progression – i.e. they become more difficult as the students advance. I would consider some of these exercises to be preparatory activities for the main tasks; for example, exercise 11 (Fig.1) prepares the students for the speaking task in exercise 16 (Fig.2).

Within the program there are five main buttons located at the top of every page. These are made up of 'outline' (which outlines the whole chapter in detail), 'report' (enabling students to check their progress after each step), 'glossary', 'help' (where learners find help topics), and 'contents'.



1. I had not seen him in years.
2. She had finished by 8:00.



1. Oh really? How long ______ it? (has had)
2. That's too bad. How long ______ it? (had had)
3. That's nice. How long ______

Evaluation of UUEG   Evaluation of UUEG

The listening task suggests that students listen to the recording of an international student's experience before answering the corresponding questions. A transcript of the dialogue is available.

In the speaking task (Fig.2) there is a 'record and compare' function that enables learners to listen to a prompt before reiterating the sentences whilst recording their speech. This enables them to compare their recordings to those of the model. Transcripts of the prompts and the model's words are available, and it is possible to play both of the recordings again and again.

The reading task comes in the form of a passage that includes some difficult hyperlinked words. By clicking on each, there appears a pop-up window that is linked to the glossary page. This displays the word's meaning along with a list of the other hyperlinked words, thus allowing students to check the meaning of other vocabulary. Multiple-choice comprehension questions follow the passage.

The above outlines what the software suggests for each task. However, it was I 's decision to ask the students to discuss these undertakings in the specially-designed chat rooms, thereby making each task more communicative. I also decided to add further activities to each, and I discussed this idea later on in the evaluation. In order to motivate the students, I offered bonus marks for those who participate in the discussion and extra activities.

Chapelle (2001) evaluation scheme

For the purpose of this evaluation, it will be useful to begin with an outline of Chapelle's Scheme (2001). Chapelle argues that CALL evaluation should be carried out using the theories of second language acquisition. There are two stages in her scheme: judgmental and empirical. In the judgmental stage, Chapelle (2001) analyses the software using two levels: the program and the teacher. In other words, she considers what learning conditions are set out by the software and what the teacher plans to do with the program respectively.

According to Chapelle (2001), however, this is not enough. She also addresses the question of what the learner actually does with the software by conducting an empirical evaluation. Whilst she focuses on different questions in each stage, she uses the same criteria in both. These criteria are: language learning potential, learner fit, meaning focus, positive impact, authenticity, and practicality. I shall judge the software by analysing the tasks using two of Chapelle's criteria: language learning potential, and learner fit.

LECTURE 3

CALL Evaluation

CALL Evaluation

Basic definitions

'CALL software' here can involve any software or programs potentially usable by language learners in connection with learning/teaching or use of language (esp. EFL/ESL). That includes both material claimed as designed for this purpose ('dedicated'), and that not. The latter includes both specific programs like adventure games for native speaker children, and 'generic' or content free software like email or word processing. It also includes whatever hard copy support materials, booklet etc. any software comes with. See further our Intro.

"Evaluation is a matter of judging the fitness of something for a particular purpose" (Hutchinson and Waters 1989: 96). 'Evaluation' therefore implies an activity where something is declared suitable or not and consequent decisions are to be made or action taken. Evaluating something therefore is not the same as researching it, though research may be done to find out things which then inform the value judgment and hopefully make it better. Research on its own may just end up with information, not judgment and action.

CALL software and general teaching materials and tasks - a parallel?

Much of what we say below about evaluation of CALL software is similar to what one would say for 'materials evaluation' generally in language teaching. CALL software is often analogous to an individual exercise or task in a book, though some series of CDROMs constitute entire courses and so are parallel with complete coursebooks. The parallel is valuable... up to a point. There are some important differences, however.

Firstly, a book is not typically dynamic or interactive; a program, by contrast, may not always present an exercise the same way every time you use it, and can usually give some response to the user dependent on what they click or type in. That is why CALL programs have often been seen as replacing a teacher rather than just teaching materials, though that clearly does not fit all software.

Secondly, a book is more limited in its media capability. CALL can involve sound as well as pictures, diagrams and text all in the same package.

Thirdly, use of written materials has few technological prerequisites: eyes and a desk to put them on will do. CALL by contrast requires computers, network access etc.

Fourthly, the language content of material in a coursebook is essentially unalterable, while some CALL software allows 'authoring': i.e. the teacher can put in his/her own choice of text, words etc. for the program to make an exercise out of, or whatever. In fact some software, such as a wordprocessing program, is essentially content-free and is nothing unless someone enters text to make an exercise, or designates a task for learners to do with it (see next).

Fifthly, the activities to be done with each section of a coursebook are usually heavily constrained by the book itself, though there may be some latitude for the teacher to implement exercises in different ways, and of course skip some material. A CALL program on the other hand may be very constrained (e.g. a hangman game), or may be almost entirely open in this respect (e.g. email).

The last two are important for evaluation, as they make it hard to draw a line sometimes between evaluating the software and evaluating the specific language material a teacher has put in, or a specific task done with the software which is not determined by the software itself. I.e. the borderline between evaluating software 'in itself' as a material and evaluating some proposed or imagined use of the software becomes impossible to maintain.

The importance of evaluation

Evaluation is one of three key aspects of CALL that need consideration: Creation, Use and Evaluation.

CALL shares one important thing with teaching materials and tasks in general. All these are under-evaluated. Just as new coursebooks and types of task are constantly being proposed and promoted by their creators … and adopted and used… so are CALL programs and activities (Chapelle top of p10). What rarely happens is any proper evaluation of the value or effectiveness of any of this…. by teachers or researchers. Correction: some teachers may well do a lot of evaluation of what they use… but, if so, it remains within their personal teaching process and is not published. Hence we have no idea how much of this goes on, or what evaluation methods and criteria are used; furthermore, nobody else gets the benefit of the information arising from the evaluation.

The three key components in CALL evaluation

Mostly evaluation cannot be done in the abstract. I.e. things are rarely universally good or bad. With CALL you may feel some programs have features which in NO situation would be any good. Possible candidates for 'universal' status could be software glitches (e.g. the program crashes whenever the help icon is clicked) and inaccuracy of language (e.g. multiple choice exercises where the option counted as correct is actually wrong). However, a lot is really 'relative' and it is as well to start off thinking of everything as potentially relative than the reverse. As Chapelle says (2001 p52): 'Evaluation of CALL is a situation-specific argument'.

Clearly most features may be good for one type of person, situation etc. but bad for another. For example the kind of vocabulary included, the kind of computer knowledge required to work it. This is as true of general materials evaluation as of evaluation of CALL specifically. So one important aspect of evaluation is to establish the specific users (learners and teachers), situation, purpose etc. etc. that you are evaluating the materials for. This means that you cannot really evaluate without also thinking of how the material will be used in the learning and teaching process. It is quite possible for one and the same program to seem 'good' when used one way with a class and 'bad' used another way, or with a different class.

Software and materials evaluation in ELT, then, can be seen as an activity where you match materials to teaching/learning situations. I.e. there are three things to think about -

(a) the nature of the materials/software: describe in detail what it consists of/does (especially if your account may be read by someone not familiar with the program). As mentioned above, this may extend to analysing the specific task it is used for/in. 'It's not so much the program, more what you do with it' Jones 1986.

(b) the nature of the T/L situation, the learners and their needs, uses etc.: describe in detail (not just 'intermediate learners'). Levy 1997 has several somewhat theoretical sections on describing CALL e.g. p108f, 156f, 173f.

(c) a rating or judgement to make of suitability of one of the above for the other, with due attention to relevant universal principles of good teaching/learning; explain how this is going to be done (e.g. introspectively or empirically - see below) and execute it.

One may of course do that for just one piece of software at any one time, but it is often easier to evaluate two or more programs of the same type together. Comparisons are often revealing. In addition, one may often usefully compare a CALL activity/program with a non-CALL (pen and paper) counterpart, as has widely been done in writing research (pen versus wordprocessor).

Furthermore you can deal with the above three components one of two ways round:

(i) You can think of a specific type of learner, teaching situation, required activity etc. first and consider whether or not each of a set of materials/each separate activity in a software package would be suitable or not for that one case. A teacher in the field is likely to work this way ("Would this suit my class?"). It is certainly easier to produce a clearly focussed evaluation that way. Note: in this course the idea is not just to evaluate CALL for ourselves as users, but to think further afield of some potential learner user type.

(ii) You can start with the materials/program and consider what range of people, situations, ways of being used etc. etc. it would suit and which not. The courseware 'reviewer' in a journal, and perhaps some of us here as AL/ELT people not currently teaching any learners directly, may prefer to think this way. When software comes with claims by its authors of what learners it is suited to, this can be a way to proceed. (But this can degenerate into letting what software is available drive what one does rather than the reverse Chapelle p44)

When the evaluation is done

It is also worth noting that there can be several types of occasion when evaluation of teaching materials, including CALL, may occur (overlooking evaluation done while the software is actually under development):
1) Evaluation of materials prior to purchasing them or creating access to them for any learners. I.e. as a result of evaluating materials you decide whether to buy or adopt them or not, for some specific learners. (Direction i usually, though ii is also possible).
2) Evaluation **after** purchase or otherwise acquiring availability of software, but before use. Here usually the question is what learners it would suit. So the consequent action is to use it with/recommend it to these learners not those, and so on. (Direction ii, or i).
3) Evaluation **after** the program has been **acquired** and **used** with some learners for a bit. Here the question is whether it was a success and the action is to use/not use the program again with these or other learners, or to alter the way it is used in some way. (Direction ii).
This account is focused more on 1 and 2, since most of us are not teachers who have just been using CALL with any actual learners, but the same ideas pervade all three situations. In all of them you decide if the materials are good or bad, not just what they consist of or 'do' etc.

Who evaluates

The evaluators we are thinking of here are primarily language teachers, though of course other people evaluate materials too - curriculum/program planners, government education departments, reviewers writing for journals, researchers in applied linguistics...etc. In the realm of CALL, it is especially necessary for teachers to be good at evaluating. There is a lot of poor material about; publishers are especially prone to hype; curriculum designers who might evaluate to choose suitable coursebooks for a course are less likely to extend this activity to CALL, so the job is left to the teacher; only a few teachers write their own CALL software (compared with the number who might write bits and pieces of their own non-CALL teaching materials) - most rely on professional products (though remember programs may require or allow some teacher 'authoring').

LECTURE 4

Judgmental Evaluation

The judgmental evaluation

Methods of evaluation (A): Introspective judgmental evaluation; checklists

There are two broad types of way of actually executing evaluation studies (A and B here). In many ways A suits situations 1 and 2 above, B suits situation 3. (Cf Chapelle 2001 p53).

Introspection means relying on one's own judgment/experience, and maybe published consensus on what should be there, what is good or bad, or AL theory.

(A1) Evaluation can be done purely individually, subjectively, globally and introspectively. I.e. the teacher simply looks through the material, or in our case tries out the program (or just reads the blurb about it in a catalogue), and comes to an overall intuitive judgment about whether it would suit their class or what class it would suit. When teachers evaluate in this way it may help in part to try to place themselves in the role of some type of learner using the material. When trying out a CALL program it is especially useful often to make deliberate mistakes to see how the program responds - e.g. give wrong answers and press the wrong keys etc.

This could be described as the global 'expert judgment' method of evaluation. The evaluator introspects and somehow accesses an unanalysed notion of some users of the software, an unanalysed impression of the software, and matches the two using often inexplicit criteria.

There are two broad types of way of actually executing evaluation studies (A and B here). In many ways A suits situations 1 and 2 above, B suits situation 3. (Cf Chapelle 2001 p53).

Introspection means relying on one's own judgment/experience, and maybe published consensus on what should be there, what is good or bad, or AL theory.

(A1) Evaluation can be done purely individually, subjectively, globally and introspectively. I.e. the teacher simply looks through the material, or in our case tries out the program (or just reads the blurb about it in a catalogue), and comes to an overall intuitive judgment about whether it would suit their class or what class it would suit. When teachers evaluate in this way it may help in part to try to place themselves in the role of some type of learner using the material. When trying out a CALL program it is especially useful often to make deliberate mistakes to see how the program responds - e.g. give wrong answers and press the wrong keys etc.

This could be described as the global 'expert judgment' method of evaluation. The evaluator introspects and somehow accesses an unanalysed notion of some users of the software, an unanalysed impression of the software, and matches the two using often inexplicit criteria.

(A2) However, to regard evaluation as in any way systematic it is necessary at the very least to 'unpack' this armchair approach a bit. The teacher (or anyone else) acting alone as evaluator should break down the 'overall' or global judgment into parts. This means (a) looking carefully at different aspects of the materials separately and (b) thinking of all the relevant different aspects of the learning situation, learners, potential use etc. etc. and (c) judging aspects of (a) in respect of (b), broken down into points. This last in part resembles the process of assessing 'content validity', often talked about in language testing: one can check on an achievement test by analysing the aspects of language tested and comparing them with what the syllabus or the teaching course before the test covered. Another general principle of language testing also applies here: it is known that tests with more items are more reliable than shorter ones, and a set of agree/disagree items circling round some issue is more reliable than a single one targeting it. So here, the summary of a whole series of introspective judgments of specific aspects is more reliable than one global one.

This is where 'checklists' come in. These are written records of the sort of 'breakdowns' just described. They may be made by the teacher/evaluator, or adopted from someone else. They at least provide a way of ensuring that important aspects do not get forgotten and that there is some consistency if the same person evaluates several things. However, the evaluation still remains individual, introspective and maybe pretty subjective. Checklists generally take the form of sets of headings to be considered or sets of questions to ask oneself. They may or may not include a system for weighting different elements, or adding up a total score in some way. Two I know of for CALL are the list of points in Jones and Fortescue, and a more reasoned and systematic framework by Odell (in

Leech and Candlin). Recently Chapelle has a set of 6 points formed from an SLA research perspective (2001 p54ff). John Roberts has a much bigger collection of such checklist used in general materials evaluation.

However, many published checklists strike one as a rather miscellaneous collection of points or questions, not clearly distinguishing between (a) and (b) and (c) above, and not obviously exhausting the types of point that should be considered, or organising them in a motivated way.

For teachers, often the checklist-based evaluation just described is the only one feasible, since it is the one that can be done quickly and easily and before the materials have been extensively used or even bought. It can be enhanced by incorporating the views, arrived at in a similar way perhaps, of more than one person. I.e. the teacher can get other teachers to do the same sort of evaluation, or read reviews in journals etc. This makes it less individual, though still introspective and rather subjective.

(A3) Additionally the teacher may enhance the checklist approach, if he/she has the time and energy......, by doing things that in a loose sense could be called 'research'. By this I mean looking systematically with some analytic techniques etc. at aspects under the (a) or (b) head above, not just deciding what they are on an instant introspective basis. This may focus more on the (a) side: e.g. linguistic analysis of the structures used in the content of the program (if it is fixed), checking the frequency level of the vocabulary against a standard reference list, grading the exercise types that are incorporated on a recognised scale of task difficulty etc. This might be called 'materials analysis'. Or it may focus on the (b) side: e.g. finding out what the syllabus for the current year actually says my learners should be doing, doing an analysis of learners' needs or interests, finding out what the school budget actually has available, etc. This is in effect 'analysis of the learning/teaching situation'. These are all things that might appear on a checklist and of course can all alternatively be decided by the evaluator just "off the top of his/her head".

Further, with respect esp. to (c) the suitability judgment itself, these may bear some 'research' in the form of reading up what theory, research studies and so forth have to say. You have a program with certain characteristics and you want to use it with young learners (as the publishers indeed claim it is suited to be). Instead of just relying on one's own judgment of what is suitable, one can read up what the collective wisdom of psychologists, educators etc. have to say about what the characteristics are of young learners and so what suits them. Similarly the general wisdom on how to construct multiple choice items (e.g. in books on testing) may help evaluate the suitability of m/c items in a CALL package. Research studies of the way learners use CALL, teaching with CALL etc. may also be worth looking at, and indeed if a program is supposedly designed to aid reading, the general wisdom on the teaching of reading and reading strategies, and so forth. However, there is always the danger that supposedly 'general' research findings do not actually apply in your situation for some reason.

But if you are using the checklist approach there are some key things not to forget:

Be explicit about where the list comes from, which existing one is being used/adapted, and have as many detailed subsections as possible. Make sure whatever system/list you use covers all three of the (a) (b) and (c) aspects

Cover the (a) aspect. A description of detailed aspects of how the program works, with examples of actual items, screens etc., and what it does (a) has to be incorporated, since the reader cannot be assumed to be familiar with the software. If part of what you are evaluating is a particular task that is not part of the software itself, or some language element supplied by the teacher, make that clear. But that alone is not an evaluation.

Cover the (b) aspect. Give a full account of (imagined or real) target learners in a situation in a particular country at a particular level etc. Evaluation for some generalised 'learner' is not very convincing.

Don't forget (c) i.e. explanation of how each feature of the program (a) does or doesn't fit (b). This needs to be supported wherever possible by more than your expert intuition - reference to applied linguistic concepts, research, models etc. (E.g. Chapelle 2001 pp45-51). This is the crux of evaluation.

The actual organisation of the writeup of such an evaluation can be done several ways. The most popular and sensible probably is to describe (b) fully in advance, and the relevant research/theory background to (c). Then go through a systematic set of (a) points - different aspects of the materials - giving a clear description of each aspect and the actual evaluation (c) of each in relation to (b).

Some people use the overt structure of the specific materials themselves as the (a) basis for proceeding. E.g. instead of having a prior idea of what categories to look at (e.g. from a published checklist), and using headings such as 'language content', 'balance of focus on the four skills' etc., they proceed through a list like 'reading passage', 'cloze exercises' (i.e. things the programmers present as separate parts of the materials). That is in some ways 'easier' but of course instead of the evaluator imposing a relevant set of categories of things to look at it puts the materials in the driving seat and may mean that relevant things do not get looked at. Compare what happens when you visit TESCO without a shopping list of one's own made in advance, and just uses the shelves of the store as a prompt for what to buy as one goes round!

Methods of evaluation (B): Empirical evaluation

Other methods of evaluation generally require much more work, and for the materials to have been used for some time by learners/in actual classes (compare situation 3), so they are often firmly fixed in a specific teaching/learning situation (b). However, they do move away from the purely introspective approach. These are the ones that incorporate activities that are just like those we would otherwise regard as typical of regular empirical 'research' - measurement, surveys etc. I.e. they may entail using questionnaires and interviews, systematically observing, eliciting 'think-aloud' data from software users, or testing users. They may mean doing 'studies' (experimental or not) comparing the success of one material against another and so forth, or indeed doing 'action research' with CALL. (See Chapelle, Jamieson and Park 1996 in ed. Pennington The Power of CALL for an overview of types of empirical research done on CALL classified by the kinds of methods used; and Chapelle 2001 pp66-94 for a more detailed coverage, in relation to CALL tasks of the more communicative type, and classic SLA research issues looked at in CALL)

In themselves these 'research' type activities are non-evaluative, in the sense considered here (except action research). They are best seen as scientific means of gathering facts and testing hypotheses which can then either remain as cold statements of fact about what the effectiveness of the materials is or what people's opinions about them are, or be exploited for practical ends as part of an evaluation exercise - i.e. to make decisions like those described at the start.

Examples are:

Doing a survey of teachers and/or learners who have used the material and finding out how they use it, their difficulties, attitudes to the interest and usefulness of the content, tasks etc. Checklists can come in here again. E.g. one can base a questionnaire to users around the same set of (a) and (b) points that might otherwise be the points one asks oneself about in A above.

Observing a class using the program, taping and making systematic notes on their difficulties, actions, strategies, what they say, the teacher's involvement etc. Or one can ask learners to keep a diary of their reactions.

Getting the computer to store records of actions performed by learners using a program and analysing them to infer learner strategies and processes. (E.g. revisions when wordprocessing, accesses made to an online glossary when reading). Example in T. Johns 1997 ‘Contexts’ in ed Wichmann et al Teaching and Language Corpora (Longman).

The classic research comparison of those using one program with those using another differing in a small or large way (or no program… just doing non-computer equivalent tasks) over a period, with before and after tests to check on how much has been learnt.

If A type and B type evaluation are both done, the connection between the two needs to be spelt out. If the A evaluation resulted in adoption of the software, did the B evaluation show that was a good decision?

LECTURE 5

A Checklist for Judgmental CALL Evaluation

A Checklist for Judgmental CALL Evaluation

The beginnings of a CALL checklist follow, inspired mainly by Odell 1986 'Evaluating CALL software' in ed. Leech and Candlin Computers in English Language Teaching and Research and John Roberts' 1996 article in System 24, but not exactly following either. This is definitely not meant to be exhaustive. You are invited to add to it, and subdivide into more detail, especially in the pedagogical area, as you look at actual software and think of points that aren't covered. It is meant to apply as much to generic software like the Internet used in some way for CALL as to a dedicated MMCD.

Remember you can organise an account in various ways – e.g. describe all the (b) first, then the (a) then finally do (c); or you can make a list of points each of which deals with (a,b,c) in one.

Some side questions I am not sure of the answer to:

How much CALL evaluation can be done using 'universal' criteria, how much is inevitably local to particular learners and situations? Chapelle 2001 ch3, from an SLA perspective, tends to emphasise the former, I, from an ELT perspective, the latter.

Should one pay any attention to the claims of the producers of software? Should one just evaluate the program for one's own purposes regardless? Or should one separately consider also (i) if the program does what it says it does, and (ii) if what it says it does is suitable to the target teaching/learning situation? Some suggest evaluation should have these two stages - External: Relevance to particular needs of particular learners (e.g. specific level, ESP, syllabus). Internal: quality of the work per se in meeting its declared specification/ aims. A prog. may be unsuitable (alone, or compared with another) EITHER because it is perfectly good but the wrong level of sophistication, coverage of items etc. for some class OR because it is just badly made.

As you try out CALL software: BOTH evaluate the software using the checklist, whatever comes to your 'expert' mind, and my hints (aimed to make you focus in more depth on either (a) or (b) elements), AND revise the checklist to become more comprehensive.

The beginnings of a CALL checklist follow, inspired mainly by Odell 1986 'Evaluating CALL software' in ed. Leech and Candlin Computers in English Language Teaching and Research and John Roberts' 1996 article in System 24, but not exactly following either. This is definitely not meant to be exhaustive. You are invited to add to it, and subdivide into more detail, especially in the pedagogical area, as you look at actual software and think of points that aren't covered. It is meant to apply as much to generic software like the Internet used in some way for CALL as to a dedicated MMCD.

Remember you can organise an account in various ways – e.g. describe all the (b) first, then the (a) then finally do (c); or you can make a list of points each of which deals with (a,b,c) in one.

Some side questions I am not sure of the answer to:

How much CALL evaluation can be done using 'universal' criteria, how much is inevitably local to particular learners and situations? Chapelle 2001 ch3, from an SLA perspective, tends to emphasise the former, I, from an ELT perspective, the latter.

Should one pay any attention to the claims of the producers of software? Should one just evaluate the program for one's own purposes regardless? Or should one separately consider also (i) if the program does what it says it does, and (ii) if what it says it does is suitable to the target teaching/learning situation? Some suggest evaluation should have these two stages - External: Relevance to particular needs of particular learners (e.g. specific level, ESP, syllabus). Internal: quality of the work per se in meeting its declared specification/ aims. A prog. may be unsuitable (alone, or compared with another) EITHER because it is perfectly good but the wrong level of sophistication, coverage of items etc. for some class OR because it is just badly made.

As you try out CALL software: BOTH evaluate the software using the checklist, whatever comes to your 'expert' mind, and my hints (aimed to make you focus in more depth on either (a) or (b) elements), AND revise the checklist to become more comprehensive.

Specification

(External pre-requisites of the software, consideration of which usually needs to be prior to any consideration of real pedagogical value. Used to assess basic practicality of using the software.)

(a) Aspects of software that are usually present and need to be looked at separately for evaluation:
What price (if not free), for multiple or single users? (Bought? Shareware? Freeware? Licenced? Homemade?)
Is it readily available?
What hardware platform required (type of computer PC/Macintosh, speed of processor, amount of memory, type of CD/disk drive, type of graphics screen capability, printer...)?
What other software needed as prerequisite (e.g. Windows, Soundblaster, particular fonts...)?
Does it have restricted compatibility with operating systems (e.g. Windows NT) or networks? Does it allow multiple use, backups?
What management required - i.e. someone's time to set things up and keep them running properly?
(b) Aspects of the teaching/learning situation that are usually present and which are relevant to deciding if (a) is suitable or not:
Specific school/learners - what do they have or can they afford in the above categories?
What school resources of staff and expertise are there to get things working and manage them?
(c) Does a fit b ? OR What b would a fit?
…. Go through all the a/b points above checking the match.
Can one even begin to consider this program - no point unless one has or can afford the platform etc?

Program design

(A lot of these points broadly relate to 'userfriendliness' of the software, or the 'computer-user interface', largely independently of any pedagogical value, but overlapping a bit)
(a) Aspects of software that are usually present and need to be looked at separately for evaluation:
How is the program loaded and run?
Speed?
What typing, deleting, mouse use, clicking buttons and suchlike basics are required?
What is the navigation means (menus, buttons, icons etc.) to jump back, forward, begin again, see where you are in the program etc? Organisation of component exercises etc.?
What means like Escape/f10/Home etc. to exit program at any point?
Does the program readily crash or hang when the wrong keys are pressed (e.g. Break, Escape...)? Or when you click fast with the mouse? Idiotproof?
Does it deal with responses with trailing spaces, mixed cases, numbers when words are required etc. etc., or consider them 'wrong' or crash?
Does it cope with typos, slight misspellings?
What output features: Sound, Graphics, Video, Written fonts, Screen layout? Presentation? How multimedia is it?
Clarity of screen layout – e.g. text size, chunking, margins?
Clarity of icons and their style (cartoon?)?

Clarity of icons and their style (cartoon?)?
Can features like sound be switched on and off? Can graphics be skipped when one doesn't want to wait while they appear, but get on with the task?
What instructions provided - amount of them and the language they are in, and level of difficulty? (A reflection of how far the software is general purpose versus targeted on a specific set of learners in a particular class/country/level)
Separate booklet and/or online help about how to work things?
Opportunity to print?
Opportunity to save uncompleted tasks or scores under individual ID and carry on next time?
Is content fixed or allowing/requiring to be provided by teacher etc? Authoring procedures? Or indeed is the software only an authoring language?
Kind of program in computational terms (pattern matching, AI, parsing....)? If on WWW is it in HTML, Java…?

(b) Aspects of the teaching/learning situation that are usually present and which are relevant to deciding if (a) is suitable or not:
Specific users - what can they manage, given their prior experience of computers? What do they find clear and 'friendly'? Are they even familiar with the querty keyboard?
Specific users - what appeals to them as attractive/important in a program? How sophisticated are they?
Specific users - what instructions can they understand easily (given their competence in the language the instructions are in). What computer actions do they know already as against need to be trained to do?
What facilities for hard copy and individual scoring are needed by course requirements?
Teacher - what time/inclination to author, what expertise at authoring?

(c) Does a fit b ? OR What b would a fit?
…. Go through all the a/b points above checking the match. E.g.
Are the program features too poor? too unattractive? sound obtrusive/irrelevant? … given the experience and expectations of these learners.
Is there so much that is unfamiliar that the students and/or teacher would spend too much time just mastering the technology, not doing real language work?
etc.

LECTURE 6

Chapelle (2001) Evaluation

The judgmental evaluation

Language Learning Potential

Chapelle (2001) describes this criterion as the degree of 'beneficial' focus on form that the software provides to its learners. It corresponds to the following questions: does the software present students with opportunities to learn the language or just to use it? To what extent does the software shift the learners' attention towards beneficial focus on form?

Chapelle (1998) also argues that if the input has been made salient it will help with language learning. UUEG focuses intensively on the forms of the perfect tense. It promotes input saliency by highlighting these forms and writing them in italicized, bold letters. Indeed, previous research has proven such a technique to be very effective (Long & Robinson 1998). Furthermore, both the colourful, animated pictures and the quizzes contribute to 'input enhancement' as termed by Sharwood Smith (1993).

Chapelle (2001) describes this criterion as the degree of 'beneficial' focus on form that the software provides to its learners. It corresponds to the following questions: does the software present students with opportunities to learn the language or just to use it? To what extent does the software shift the learners' attention towards beneficial focus on form?

Chapelle (1998) also argues that if the input has been made salient it will help with language learning. UUEG focuses intensively on the forms of the perfect tense. It promotes input saliency by highlighting these forms and writing them in italicized, bold letters. Indeed, previous research has proven such a technique to be very effective (Long & Robinson 1998). Furthermore, both the colourful, animated pictures and the quizzes contribute to 'input enhancement' as termed by Sharwood Smith (1993).

During the speaking task the focus is entirely on the contracted forms. In the listening and reading tasks, learners are tested on their comprehension of both the dialogue and text respectively, with a moderate focus on the forms.

Chapelle (2001) and Skehan (1998 in Chapelle 2001) suggest some conditions which might characterise a task that draws learners' attention to the form. I will focus on two of them – namely, 'modified interaction' and 'modified input'.

Similarly, in the speaking task the students ere asked to log into the chat rooms to compare their pronunciations (after they have compared their recordings with those of the model). Consequently, the author expected an interactional modification to take place. The author also devoted a portion of time to focus on irregular and regular verb forms and their pronunciation, mainly using the verbs in the program.

It is obvious that when using UUEG an interactional modification between the learners and the computer is to be expected, and Chapelle (1998) suggests this to be a key element in developing a CALL task. The reading exercise provides a prime example of this theory, as meaning is expected to be broken down when students are shown the hyperlinked words. These students were expected to obtain help by clicking on each word to get its meaning. However, while this element is considered to be one of the strengths of the software, there is no other way for learners to get help with other words that they might find difficult. Therefore, in the author's opinion, a link to an online dictionary was a solution for this.

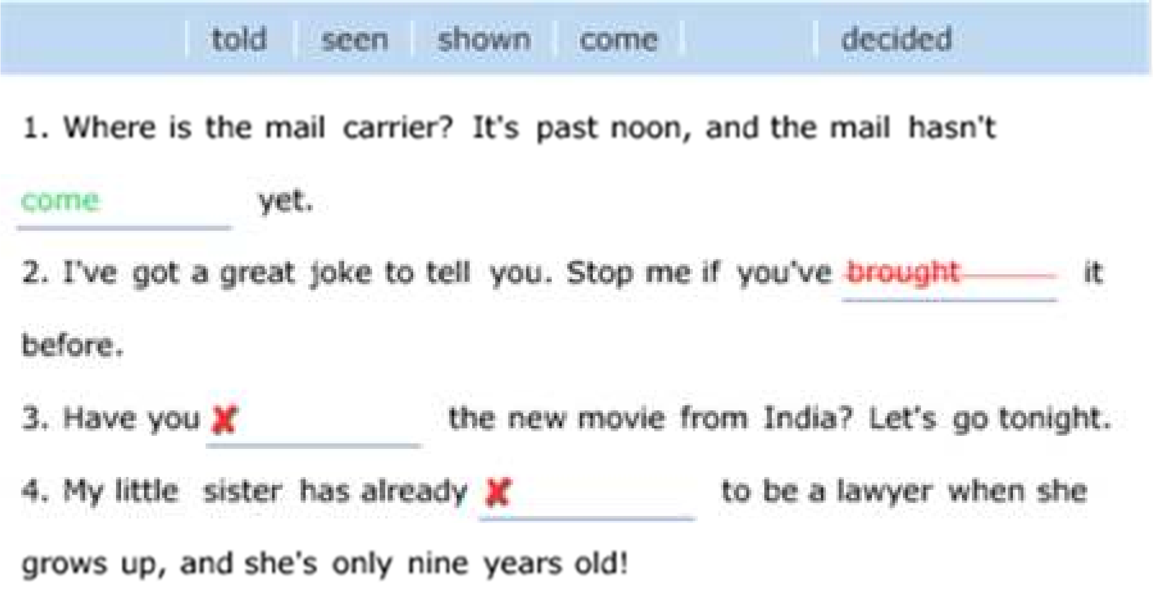
Moreover, learners were given a chance to preview the passage to help them answer the questions. By consulting the passage, learners were interacting with the computer. Interactional modification can also be achieved in the speaking task; when observing students during their performance of this exercise, it is clear that modifications can come in the form of repetition requests whilst comparing or checking the transcripts. If the software were to give a statistic of how many times options such as ‘preview the passage’, ‘compare’, and ‘transcript’ were accessed, it would give us a real indication of interactional modification between learners and the computer. Unfortunately, such a feature is not supported by UUEG.

Modified output

Chapelle argues that CALL software should have the ability to let students 'notice' their errors as this would help them to shift to 'a syntactic mode' that aids in internalizing the new form (1998, p.4). Borg (1999) also claims that error awareness helps students to 'monitor and self-correct their use of language' (p. 158). In UUEG, the feedback is very appropriate and one of the potential strengths of the software. By pressing the ‘check answer’ button that is found at the bottom of every page that has exercises, errors are crossed with a red line (or with a red cross if no answer has been given)

Chapelle (1998) also argues that learners should be given the chance to correct their errors, and in the exercises discussed earlier students were given a second chance to do just this. If an error still persists, the computer will eventually display the answer in green. When the mouse is moved to the corrected answer, it flashes the error in red and the right answer in green. The author believes learners will benefit greatly from this feature. In the case of more than two errors being made, the computer will advise learners to go back to the previous charts and check their information. The author supports Chapelle's (1998) view that it is advisable to have access to some online references that can help learners make corrections.

When all of the answers are correct, the software displays a 'well done' message in red at the top of the exercise, and changes the answers into the colour green. The coloured feedback is of significance: apart from giving a focus on form, it allows the computer to take on the occupational role of teacher, as people in this profession tend to use the colour red when making corrections.



told | seen | shown | come | decided

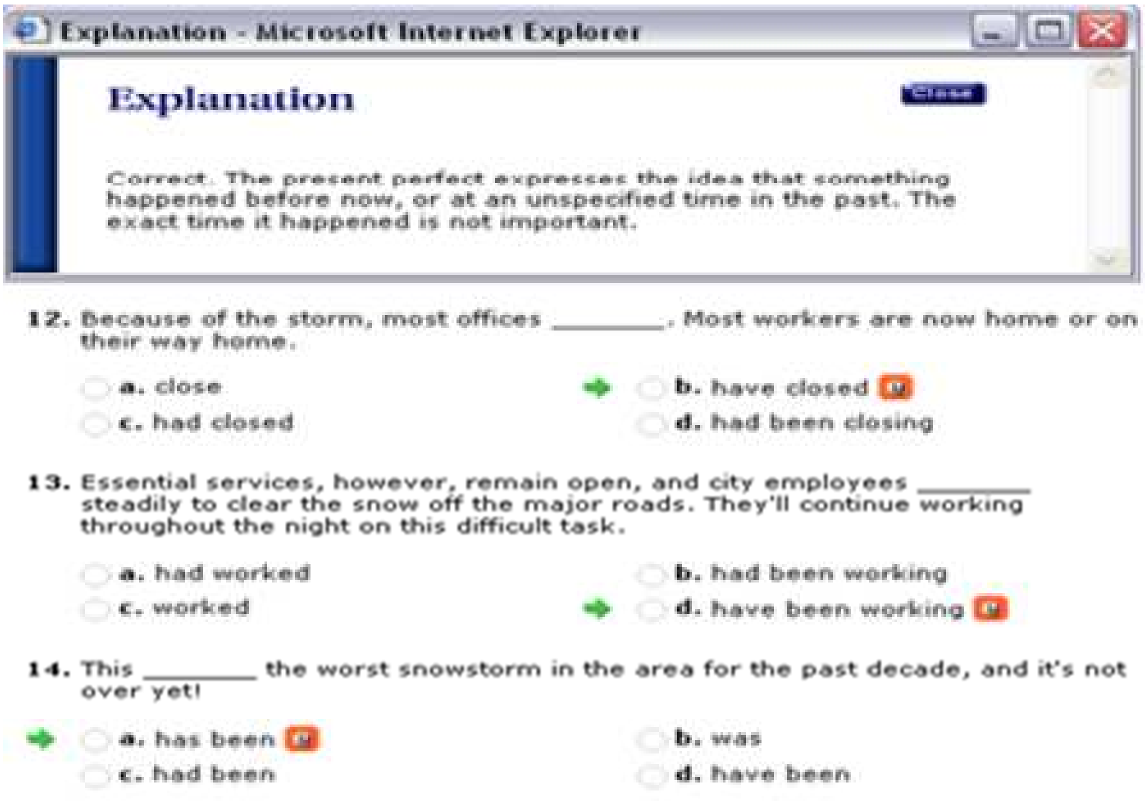
1. Where is the mail carrier? It's past noon, and the mail hasn't come yet.

2. I've got a great joke to tell you. Stop me if you've ~~brought~~ it before.

3. Have you ✘ the new movie from India? Let's go tonight.

4. My little sister has already ✘ to be a lawyer when she grows up, and she's only nine years old!

A further strength of the program is the feedback provided in the test sections (Fig. 6). By pressing an orange 'e' button that appears next to each error, learners are given an explanation of each of their mistakes. However, in order to imitate the challenging conditions and characteristics of an exam, the program does not offer learners the chance to correct any errors made during the test section (unless it is uninstalled then reinstalled again). Unfortunately, there are no notifications of this in either the tests’ rubrics or anywhere else in the software.



Explanation

Correct. The present perfect expresses the idea that something happened before now, or at an unspecified time in the past. The exact time it happened is not important.

12. Because of the storm, most offices ________. Most workers are now home or on their way home.

○ a. close
○ b. have closed
○ c. had closed
○ d. had been closing

13. Essential services, however, remain open, and city employees ________ steadily to clear the snow off the major roads. They'll continue working throughout the night on this difficult task.

○ a. had worked
○ b. had been working
○ c. worked
○ d. have been working

14. This ________ the worst snowstorm in the area for the past decade, and it's not over yet!

○ a. has been
○ b. was
○ c. had been
○ d. have been

Learning consecutive interpreting

Learner fit

In Chapelle's description (2001), learner fit takes account of both the language level and its learners' characteristics. CALL materials must suit the target learners, and accordingly its tasks should be set at a level that is neither too simple nor too difficult (Skehan in Chapelle 2001). UUEG is appropriate in terms of content for learners whose levels range from lower intermediate to upper intermediate, and it is designed specifically for those who want to improve their grammar in an innovative way. As for the author's students, the program is well suited to their needs. The author's claim is based on the past evaluation of the original book that has been used for more than ten years.

In Chapelle's description (2001), learner fit takes account of both the language level and its learners' characteristics. CALL materials must suit the target learners, and accordingly its tasks should be set at a level that is neither too simple nor too difficult (Skehan in Chapelle 2001). UUEG is appropriate in terms of content for learners whose levels range from lower intermediate to upper intermediate, and it is designed specifically for those who want to improve their grammar in an innovative way. As for the author's students, the program is well suited to their needs. The author's claim is based on the past evaluation of the original book that has been used for more than ten years.

With regards to difficulty and control, the help section claims that there is also an 'orientation' page within the program, but the demo version used in this evaluation does not provide this facility. Nevertheless, the orientation page equips learners with the information necessary to operate the program, thus allowing them to have full control over it, which in turn gives the software more strength. Indeed, students can move freely from one section to another, record and repeat as applicable, and modify their recordings whenever necessary. Furthermore, they can record as many times as they wish, as once they click the button any previous recording will be erased.

Research shows that learner control is beneficial. However, giving full control to novice learners (i.e. those with poor knowledge) might affect them in a negative way (Clark & Mayer 2003; Hannafin & Hooper 1993 in Lawler-King 2004). Whilst the majority of the exercises and their rubrics are clear and set at the correct level for the author's students, this cannot be said of those designed for error recognition. Moreover, the author has a view which is consistent with that of Heaton (1991): error recognition is not an adequate way of helping students to learn. In the author's opinion they should be exposed to the correct forms, which in turn would help them to produce the language correctly themselves. Nevertheless, this is only true when considering the first stages of learning; advanced students, the author believe, need to be able to distinguish between correct and incorrect forms. the author say this as his students still face problems with the language and still produce errors, and the author doubted that these particular exercises were easy enough for them.

The tasks, like the exercises, are appropriate for teaching language at the level required. In the listening task, the dialogue is simple and the speakers talk at a suitable speed. In the reading task, the language used in the passage matches the students' abilities perfectly. The author doubted that they would encounter any difficulties in either of these two tasks as they already have been exposed to the same materials. All in all, the software presents the students with materials that are new to them, and this enhances second language acquisition (Krashen 1982 in Chapelle et al. 1996).

Another issue relevant to learner fit is the level of the program's appeal to learners. If it were repetitious and dull, it might generate the unwanted factor of boredom. Yet filled with colours, different cartoon characters, animated visuals, games, drag and drop quizzes, and record and compare exercises, the author considered UUEG to be very appealing and joyful.

Furthermore, the 'help' and 'report' options make this programme even more attractive. Learners can find help and support for the most frequent technical problems encountered, and there is information at hand about the system requirements and how to set up the microphone (which is not easy to do). Installation instructions are also available, along with a contact number and an email address through which it is possible to leave feedback about the software. Indeed, it is the author's intention to set the author's students some homework, in which they must write (using the perfect tenses) their own feedback about UUEG, detailing their experience and opinion of the program. These can then be sent to customer support. The purpose behind this is to overcome one major drawback of UUEG: the software does not cover the important skill of writing, and this is of great significance as the author's students are keen to improve their skills in this medium.

In the 'report' option, students can monitor their progress from one section to another within a single chapter. The report shows the learner's name alongside his or her score in each of these sections, and after finishing each chapter learners can compare their most recent score with those gained earlier in the program. An overall average will then be shown at the end of the course. Characteristics and controls such as these demonstrate that UUEG makes a provision for self-study.

Furthermore, the 'help' and 'report' options make this programme even more attractive. Learners can find help and support for the most frequent technical problems encountered, and there is information at hand about the system requirements and how to set up the microphone (which is not easy to do). Installation instructions are also available, along with a contact number and an email address through which it is possible to leave feedback about the software. Indeed, it is the author's intention to set the author's students some homework, in which they must write (using the perfect tenses) their own feedback about UUEG, detailing their experience and opinion of the program. These can then be sent to customer support. The purpose behind this is to overcome one major drawback of UUEG: the software does not cover the important skill of writing, and this is of great significance as the author's students are keen to improve their skills in this medium.

In the 'report' option, students can monitor their progress from one section to another within a single chapter. The report shows the learner's name alongside his or her score in each of these sections, and after finishing each chapter learners can compare their most recent score with those gained earlier in the program. An overall average will then be shown at the end of the course. Characteristics and controls such as these demonstrate that UUEG makes a provision for self-study.