ENGLISH LANGUAGE TEACHING COMPUTER PROGRAM

A SUPPLEMETARY MATERIAL FOR TEACHERS AND LEARNERS

BEGINNER LEVEL

BASED ON STREAMLINE ENGLISH DEPARTURES

By

Varga Levente

University of Debrecen
Institute of English and American Studies

Supervisor: Nagy Tibor

Debrecen, Hungary
2007

Introduction

Nowadays, computers are continuously getting more and more widespread as technology becomes more available for the masses of people everywhere. Two billion computers have been sold so far[1] and every year more than two hundred million computers are shipped worldwide[2]. The same expansion is true for the English language: over 380 million people speak English as their first language and estimates that include second language speakers vary greatly from 470 million to over a billion depending on how literacy or mastery is defined[3]. Due to the massive globalization and the rapid spread of the Internet, English language gained such an importance and popularity that has never been before. The merger of these two areas is inevitable in a world where many things are being automatised and implemented into computer programs. In my thesis firstly I would like to elaborate on the history of computer-assisted language learning (further abbreviated simply as CALL), then analyse some of the most well-known Hungarian CALL applications and describe the creation and operation of my program.

[1] 1 Billion Served, PC's go over the top, and into the Dump, http://homepage.mac.com/techedgeezine/1billion_served.html, April 10, 2007.
[2] Personal Computer Market Share: 1975-2004, http://www.pegasus3d.com/total_share.html, April 10, 2007.
[3] Wikipedia contributors, "List of countries by English-speaking population," Wikipedia, The Free Encyclopedia, http://en.wikipedia.org/wiki/List_of_countries_by_English-speaking_population, April 9, 2007.

The History of Computer-Assisted Language Learning

The history of teaching English goes back to a very long time, however teaching the language with the help of computers is a relatively new phenomenon. The use of technological devices in teaching English has always been part of English teaching since their appearance. Today it is usual that teachers use tape recorders in the classroom since these devices make teaching more effective by providing authentic audio material for the students from the native country. However, with the upheaval of the appearance of these new devices - such as audio or video - there were always some overuse of these machines, instead of the initially planned application.

> Fascinated by the new technology, many teachers focused on technological issues, neglecting pedagogical and methodological questions and not realising that innovative pedagogy and methodology were required to integrate satisfactorily the use of computers into the foreign languages curriculum[4].

This phenomenon also happened when computers appeared in language teaching. Some researchers thought that computers will revolutionize language teaching and will be able to use such methods that will immediately boost language learning. Too much hope was connected to these devices and due to over-expectations the result was often disappointment,. Many attempts were made to bring a new perspective by using computers; some of them managed to survive, others did not.

Computers have been used since Universities and researchers could try out the primitive computers of the time. Computers have been used for language teaching since the 1960s. According to Mark Warschauer, the professor of University of Hawaii, USA, this 40-year period can be divided into three main stages: behavioristic CALL, communicative CALL, and

[4] Wikipedia contributors, "Computer-assisted language learning," Wikipedia, The Free Encyclopedia, http://en.wikipedia.org/wiki/Computer-assisted_language_learning, April 9, 2007.

integrative CALL[5]. These stages changed according to the actual phases of the technological developments of the time. They were also modified due to the fact that by time more and more new information and expertise was gained in this field of language teaching. Furthermore, certain pedagogical approaches also altered the way how CALL would look like in practice.

Behaviouristic CALL

The theoretical foundations of the so-called behavioristic CALL were establisher in the 1950s, however, the whole idea was implemented and made into practice only in the 1960s and 1970s[6]. As its name suggests it was based on the behaviourist learning model, which was the prevailing learning model of the era, especially in the United Stated of America. This approach mainly tried to achieve language acquisition by continuously repeating certain patterns of the language. In practice, this ideology manifested in computer programs as repetitive language drills, referred to as drill-and-practice.

It was already an initial aim of even the first CALL applications to provide students the chance to advance according to their own pace. The computer was viewed as a automatic facilitator who never grew tired or disapproving and allowed students to work at an individual pace. This possibility to use call in the way as it was intended to use was only possible, when personal computers became available for greater masses of people and not only very big organisations could afford to purchase and maintain a computer. Before the era of personal computers behaviourist CALL applications run on mainframe computers. The most memorable and best-known tutorial system was PLATO[7]. It "ran on its own special hardware

[5] Warschauer, M., & Healey, D. (1998). Computers and language learning: An overview. Language Teaching, 31, 57-71.
[6] The history of CALL & CMC, http://www.msu.edu/~ruimin/cep901.html, April 9, 2007.
[7] PLATO: The Emergence of Online Community, http://thinkofit.com/plato/dwplato.htm, April 10, 2007.

consisting of a central computer and terminals and featured extensive drills, grammatical explanations, and translation tests at various intervals".[8]

Communicative CALL

The second stage, called the communicative CALL came into existence mainly due to two determining factors. The first reason was that at the end of the 1970s due to several reasons the behaviouristic approaches regarding language teaching began to be rejected in many places both at the theoretical and pedagogical level.[9] The other triggering thing was the appearance of personal computers to a wider range of people, which opened a whole set of new possibilities for those, who wanted to create language teaching applications. As these new personal computers were available for a lot more people, both the number of developers and the number of users boomed. Commercial computer programs began to appear. The communicative CALL approach evolved in the late 1970s and early 1980s which is also the period when these two events happened.

> Proponents of communicative CALL stressed that computer-based activities should focus more on using forms than on the forms themselves, teach grammar implicitly rather than implicitly, allow and encourage students to generate original utterances rather than just manipulate prefabricated language, and use the target language predominantly or even exclusively.[10]

In concordance with some cognitive theories, communicative CALL also emphasized that learning should be a progression route, where the student discovers the language instead of mugging up certain forms. It laid more stress on the personal creative discovery and practice of the language[11].

---

[8] Computers and language learning: An overview, http://www.gse.uci.edu/faculty/markw/overview.html, April 9, 2007.
[9] Behaviorism, http://plato.stanford.edu/entries/behaviorism/#4, April 10, 2007.
[10] Computers and language learning: An overview, http://www.gse.uci.edu/faculty/markw/overview.html, April 9, 2007.
[11] Communicative CALL: Principles, http://www.ucs.mun.ca/~emurphy/technologyuse/sld009.htm, April 10, 2007.

Many of the applications that were developed in the attitude of communicative CALL included text reconstruction programs, which taught students the patterns of language and meaning for individual students or student groups by allowing them the discovery to rearrange words and texts[12]. Another common feature was simulation, which stimulated discussion and discovery among students working in pairs or groups.

Although communicative CALL was considered to be advancement in comparison to behaviouristic CALL, after a decade of its appearance it started to be criticised for the definite reason that it was too marginal[13]. This meant that with communicative CALL language learning was still quite isolated in many ways. Students only had opportunity to practice a special kind of exercise, with developing only a definite language skill. Besides, isolation was observable in isolated material that was not much to do with everyday life of the English-speaking countries.

Integrative CALL

The decline of communicative CALL and new technological developments lead to the third stage, called integrative CALL by the late 1980s and early 1990s. This approach consciously strives to join together the teaching of not only one language skill but often integrates listening, speaking, reading and writing exercises. At the same time it tries to integrate technology into the language learning process as much as possible[14].

Similarly to the communicative CALL, this method is also associated with new technological developments. In this case these new technological developments manifested in the appearance of the multimedia-capable personal computers and both the rise of the networked computer and the computer networks. Multimedia computers, which appeared for the masses

[12] Solange Moras, Computer-Assisted Language Learning (Call) And The Internet, http://www3.telus.net/linguisticsissues/CALL.html, April 10, 2007.
[13] Mark Warschauer, Computer Assisted Language Learning: an Introduction, http://www.ict4lt.org/en/warschauer.htm, April 10, 2007.
[14] Kuang-wu Lee, English Teachers' Barriers to the Use of Computer-assisted Language Learning, http://iteslj.org/Articles/Lee-CALLbarriers.html, April 11, 2007.

in the middle of the 1990s opened a wide range of new possibilities in language learning. New computers with greater performance appeared which were capable of using really great multimedia power[15]. This way, creators of multimedia teaching programs could put together sounds, images, video and text into a single application, and use these integrated materials combined together to further enhance language learning. Thus, we can say that they merged several devices - the slide-projector, tape recorder and video player - in one single device. This really meant a great leap at that time; using these technologies simultaneously could open new possibilities for language teachers who wanted to create educational software for language learners. Moreover, not only were computers capable of playing audio, but they were able to record them. Students could record their own voice while articulating the sentences of the foreign language and then, they could compare their version to the sample. This meant that now they could hear their voices from an external perspective and immediately correct their mistakes, which was then impossible without a language teacher. Since this feature proved to be very efficient, it is no surprise that it is present in many multimedia educational software.

At the present the largest developments in the field of CALL are related to the Internet. The World Wide Web opened many new possibilities and perspectives. A wide range of English language teaching applications can be found today, most of them is free to use. These vary greatly both in type and the pedagogical approaches adopted from traditional grammar-based teaching to innovative goal-oriented quests, wile the former still dominates[16]. Moreover, students of the language can practice English all the time, since it is the unofficial language of the Web, most of the web pages are available in English, not to mention chat rooms. There can be a vast potential in this area. With the adoption of broadband internet access and the

[15] The History and Development of Multimedia: A Story of Invention, Ingenuity and Vision, http://www.ucalgary.ca/~edtech/688/hist.htm#ninties, April 10, 2007.
[16] The History of CALL, http://www.ict4lt.org/en/History_of_CALL.pdf, April 11, 2007.

new multimedia possibilities that come with it, new promises open in the field of computer-assisted language learning.

Analysis of some English Teaching Computer Programs

There are a lot of CALL applications in commercial use in the world and in Hungary; some of them are really unique pieces of work, while others tend to make everything after the same pattern.

I have analyzed several applications during my research work. Three of them were created by Woodstone Studios (Nyelvész, Nyelviskola and Nyelvstúdió), which has published a lot of CALL applications in the 1990s and 2000s and became on of the most prominent educational software developers of Hungary[17]. These programs contained many common features and functions, while they differed in the content to some extent. Beside Woodstone programs, I have written about four other CALL applications (Lopva Anlolul, Manó Angol, PicDic and ClipDic), which I could not try out personally, but collected lots of useful pieces of information about them.

Lopva Angolul

The first of these latter three programs is called Lopva Angolul, which is created from the course book of the same title by Czobor Zsuzsa and Horlai György[18]. The title "lopva" is ambiguous. First, it alludes to the idea that the users of the program can learn English in an imperceptible way, by solely watching its episodes and doing the exercises. Secondly "lopva" comes from the fact that the program is built around the story of a crime band.

The program consists of three CD-ROMs with different language levels. It aims to develop language knowledge by using animated cartoon movies: the language learner can watch more than 180 minutes of cartoon episodes with subtitles. For each episode many exercises are attached, that is why there are more than 1800 exercises in the program[19]. These exercises are

[17] Multimédia Holding Céginfó, http://www.woodstone.hu/ceginfo.html?PHPSESSID=8e0f54069605f46bc226acc74822d201, April 12, 2007.
[18] Móricz Attila, „Tanulás számítógéppel", PC WORLD (March 2000)
[19] http://www.tujapont.hu/product_info.php?products_id=16389, April 10, 2007.

to develop text comprehension, reading and listening skills, expand vocabulary and grammar knowledge, and in addition there are microphone drills and games in the program. Moreover, there is a 2000-word vocabulary with the correct English pronunciation of the words and in addition, the program is shipped with a 100 page comprehensive language book[20]. For sample images of the program see Figure 1 in the Appendix section.

Manó Angol

This application is part of the Manó software family which is created for children between 8 and 14 years. Therefore, it has easy to handle user interface and a children-friendly learning content[21]. It contains a lot of games and this way children can learn the language playfully, in a very enjoyable way. There are also many dialogs in the program, backed with subtitles. After listening to these dialogs the student can do the some related exercises, where the instructions are uttered in Hungarian to ease learning. Besides, the program contains fifteen pleasant songs, which can also be played with a traditional CD player[22]. There is also an option to record voice and practice pronunciation. The program contains a speaking vocabulary, which is unique for its more than two hundred built-in animations[23]. For sample images of the program see Figure 2 in the Appendix section.

PicDic

The main idea of this CALL application is to use images to teach English language. It connects words and stories with pictures, which is accompanied with the voice of native speakers. It is unique that learners can click on some selected points of the pictures to select different objects. By this, the user gets the Hungarian and English equialent of the world with its English pronunciation. Moreover, quite surprisingly this CALL application can be used to teach reading, because children can try out putting together letters to form words, and to

[20] Lopva Angolul Sorozat, http://www.profi-media.com/nyelvokt/lopva/lopva_info.html, April 9, 2007.
[21] helyesírási készséget fejlesztõ feladatok, http://www.profi-media.com/szakmai/dudas.html, April 9, 2007.
[22] http://www.ebolt.hu/product2639.html, April 9, 2007.
[23] ManóAngol, http://www.inext.hu/~lugosi/PROFI%20MEDIA/Manonyel.doc, April 9, 2007.

match these words with the drag and drop method with their picture that they belong[24]. Besides, the application contains more than two hundred computer graphics and language learning English is further aided by language tests with audio.[25]

ClipDic

This CALL application uses video clips to teach English language. Most of the material of this program consists of video clips of broadcasts from the news programs and other everyday situations, which can greatly help to develop the listening skills of the learner[26]. For every clip several various kinds of exercises and tests belong. These tests are filling the gap, structuring the words of a sentence in the correct order, multiple-choice tests, vocabulary drills and exercises where the user has to write down the heard text. The program contains a comprehensive grammar summary, which the user can browse and search in[27]. There is an option to print the texts belonging to the clips in English or Hungarian, which is a quite rarely used option but might be useful for some of the users. For sample images of the program see Figure 3 in the Appendix section.

Common Features of the Applications of Woodstone Studios

The material of these applications is built around stories, which are presented as cartoons. By default, all the three programs teach the language by playing automatically these cartoon sequences and the student can observe as the story unfolds. At the same time, these cartoons are backed by the voices of English speakers and subtitled in English or Hungarian language, which can help comprehension. Besides automatic playing, there is an option for the user to manually advance to the following cartoon images and texts. In all the three cases the user can record his or her voice to practice pronunciation and can compare it to the sample English pronunciation. Besides, the three programs provide the feature to choose between the desired

[24] PicDic for Windows, http://www.inext.hu/~lugosi/PROFI%20MEDIA/Pic-clip.doc, April 10, 2007.
[25] PicDic sorozat, http://www.profi-media.com/nyelvokt/picdic/picdic_info.html, April 9, 2007.
[26] http://shop.unas.hu/shop_artdet.php?shop_id=1000&cikk=termek_350312, April 10, 2007.
[27] ClipDic sorozat, http://shop.unas.hu/shop_artdet.php?shop_id=1000&cikk=termek_350312, April 10, 2007.

units in a separate menu. In addition, there are both English and Hungarian vocabularies in all the three programs. Because all the three applications include these common features, during their description I tend to emphasize the things that they differentiate and unique in.

Nyelvész

In this program we can get acquainted with the story of a young woman who is getting familiar with England and many new things. The whole user interface can be controlled by clicking on images that represent a certain function. Since the pop-up helping descriptions for these image buttons appear in Hungarian, to avoid misunderstanding I will use these Hungarian descriptions here. If we choose the Képregény option, the cartoon is automatically played, and the user can watch each cartoon episode of about five minutes. These cartoons contain images, text and audio along with English or Hungarian subtitles on demand. Unfortunately, the pictures are not too various and the drawings are not too nice either. We can see the same two women all the time, while sometimes some other relevant images are added. One reason for the few number of images can be that the creators did not want the application to occupy too much hard drive space. Including images in the program proportionately increases the size of the program, which is not a desired property, especially in older systems. Besides automatic playing, the user can read all the sentences of the dialogues in the "felolvasás" option. The user can practice the pronunciation of these sentences in the "Felvétel" menu. Unfortunately, during some of the sentences the speakers perceptibly speak with a Hungarian accent. The program contains fifteen units, which can be selected from the contents menu, which is created in a user-friendly way. For each of these units a grammatical summary and some tests belong. The printable Hungarian grammatical summaries can be quite useful; they can support the comprehension of the material.

Four kinds of tests can help the students to progress and practice the content of the cartoons. There is a fill in the gap exercise, where the user has to write in the missing word into the

sentences of the cartoons. The problem with this exercise is that it accepts only those words that were contained in the cartoon. Even the smallest deviation is unaccepted, and if the student writes in a solution that is correct syntactically or semantically but not the desired word by the creators, it will be evaluated as a mistake. The next type of exercise is the multiple-choice test. This is very similar to the previous exercise, but here the user does not have to type in the word that is missing from the sentence but has to choose it from three given choices. This implementation is more precise and straightforward than the previous one, since it provides exact answers and only the right solution can be selected. There is also an exercise where the student has to type in the word that is uttered. Fortunately, in this program there are no yelling voices after the correct or incorrect answers. These sounds can be funny and entertaining at first, but after some time they are quite annoying.

The user interface of the program is quite user-friendly, the buttons and functions that belong together got clustered and got next to each other. The user can set the volume level and the pauses between two cartoon images. There is also an exit option at any time, which – despite its importance - is unfortunately omitted from some CALL programs. For sample images of the program see Figure 4 in the Appendix section.

Nyelviskola

This application is intended to be an intensive language training, therefore, it covers more compact material than other CALL applications. Its structure is very similar to the previous application, however, there is a main difference: in some cases the dialogues can be controlled by the user[28]. There are ramifications where the user can choose from two different answers, and the dialog will continue according to the selected branch (see Figure 5 in the Appendix). Although, there are only few places where the direction of the dialogues can be chosen, this feature can make language learning more interactive and thus, more interesting and enjoyable.

[28] http://www.interbook.hu/uj/index.php?op=termek_reszletes&termek_id=2146&fokat=5&alkat=10, April 10, 2007.

In any other aspect, the application is structured in a very similar way to the other two programs: there are also some cartoons with images, audio, texts and subtitles and they can be played automatically or manually. As a curiosity, besides Hungarian subtitles, there are Polish and Romanian subtitles. The buttons are also almost identical to the previous application, however, there are no pop-up describing texts for the buttons, which can greatly render navigation more difficult, especially during the first few uses of the program. The graphical user interface is clearer in comparison to the previous application. The buttons for the tests are on the right side, all the other buttons are on the left side.

Three of the exercises are identical in type to the previous program and there are two new kinds of exercises. In the first one, the user has to choose from three possible answers to a given question, while two of these answers are syntactically or semantically incorrect. In the other exercise, the user is shown a picture and given a related “yes or no” question. The student has to choose between the two options.

Similarly to the previous program, the drawings are not too nice and various here. What is more, the program contains only an English-Hungarian vocabulary and there are no grammar summaries here. Another problem is that the name of the program is nowhere to be found: neither is it printed on the CD-ROM, nor written in the program, which can be very confusing (see Figure 6 in the Appendix section).

Nyelvstúdió

This application is the finest one of the three, it comprehends all the good features of the previous two programs, while omits many of their mistakes. The graphical user interface is implemented in a high standard; it is structured very well, it is clearer and more transparent. Moreover, there is a separate settings menu where the user can set the volume, speed and some other important settings. The separate contents menu also worth mentioning, since it was created in such a beautiful and creative way that is unique among Hungarian CALL

applications. The user can choose units by clicking on books that are stored in a bookshelf of a nice reading room (see Figure 7 in the Appendix).

The drawings of the program are also much nicer than in the case of the previous two applications. Interestingly, the program was created with the same Toolbook developer program that was used to create my application.

This program is also based on cartoons, which can be played automatically or manually or with sounds and text only. Printable grammar summary for each unit, English-Hungarian and Hungarian-English vocabularies are also featured built-in parts of the program (see Figure 8 in the Appendix).

The recording function of this program is more developed, the user can not only compare his or her pronunciation with the correct English one by playing one after another, but the program is capable of playing the two audio files simultaneously, thus, providing a better opportunity for comparison.

Besides, there are five different types of exercises for each unit. Four of these tests are similar to the ones that were described in relation to the previous exercises. However, there is a crossword puzzle built in the program, which is a real curiosity among CALL applications. The user has to answer a question and the solution will be written into the crossword puzzle. This is a very creative and entertaining way of language teaching and at the same time it is quite effective (see Figure 9 in the Appendix).

The only negative aspect of tests is the continuous yelling sounds that follow each correct or incorrect answer. After some time this feature can be very annoying to the user. However, after all, this program was the best among the programs that I have tried out.

The Material of the Program

The written application is not a tutorial program for students but rather a digital supplementary material for teachers who want to make their lessons more colourful. Therefore, the main aim of the program is not to provide a tutoring system that stands on its own, but to provide digital supplementary material in a wide range of fields through the adaptation of a well-known English teaching book, Streamline English.

This usage determined the main principles concerning the selected material of the program. It contains a lot of thematic units, which teach different grammars. Teachers can pick out very colourful material for their students from lots of different themes. It is a great source for a great number of different topics, especially everyday conversations.

Since the program is a supplementary material for skilled English teachers, it is not aimed to provide a vocabulary of the contained words or to include exercises. The reason for omitting vocabulary is quite straightforward: teachers do not need vocabulary to understand and translate the words that are present in an English language teaching book for beginners. Besides, since most of the English teachers are got used to create or collect exercises for their students, including exercises and test was not necessary.

Although the program is aimed to support English teachers with digital material, its use as an intensive English teaching CALL program is not impossible. However, this is rather advised for students who want to brush up their language knowledge or improve their listening comprehension.

For a sample image of the program see Figure 10 in Appendix.

Adapted Book – Streamline English

My program is based on the material of Streamline English. This book was written by Bernard Hartley and Peter Viney and was published by Oxford University Press in 1978. The book was a real novelty in Hungary in the 1980s for its colourful pages, interesting stories and

teaching material. It was received with enormous enthusiasm and hope and many of the English teachers chose to teach from the book. However, after a few years it turned out that the book is not as fantastic as it was previously anticipated. It was sometimes and methodologically wrong in some aspects, for example in the case of many of its exercises. In spite of these negative properties, it contained many useful pieces of material. In writing my program and choosing the book, my aim was to save these useful pieces of Streamline English for the future in digital format and hand it to help teachers to find digital supplementary material for language teaching.

I decided to choose Streamline English for several other reasons. Firstly, the book was not hard to make a digital version of. Secondly, probably the best property and advantage of the book is that it is humorous. There is always some entertaining material included in almost every unit, whether it is materialised either in comical images, humorous text or audio. What is more, the content is various in topics, methods and material. These things make language learning more enjoyable and more interesting and therefore the users of the program will be more motivated to use the program and learn the English language. Furthermore, the book contains appropriate amount of material to support the basic English knowledge. The amount of material is not insufficient for this aim, and at the same time not too much to make language learning a boring thing. Besides, the book is an intensive course, which also provides faster language learning.

In addition, I wanted to create a program from a book which is made for beginner language learners and at the same time has no CD-ROM version. The program can be made from the foundations, I can decide what the final application will look like and do not have to adjust the design to a previous design. Besides, this program can be continued; a sequel can be created by digitalising Streamline English Connections and Departures.

Moreover, throughout the book, users can read and listen to British English accent. The book contains many everyday conversations of the British people, which take place in such common places as shops or pubs. By this, students can glance at the everyday life of the British people, get acquainted with their culture and habits, which makes the book more authentic.

The Software Developing Environment - Toolbook

The application was written with a developer program named ToolBook version 5 (see figure 11 in Appendix). The decision to choose this program was due to several reasons. Firstly, this developing environment is especially suitable for creating educational software with multimedia support. The program is more than 10 years old- the fifth version was created ten years ago in 1997 - which is an enormous time in the life of computer science because most of the life cycle of the applications lasts no more than only a few years because new technologies come and the old software often become obsolete. However, this application is still used in several places because it contains most of the features that are present in today's newer multimedia software developer applications with only minor deficiencies. Moreover, in comparison with other integrated developing environments made for program building – such as the integrated developer environments of C++, C# or Java - developing multimedia software is quite easy with Toolbook even for engineers without advanced programming knowledge.

Secondly, I chose Toolbook because I have already completed a seminar which was about developing applications with the software. I learned how to use its graphical user interface, how to insert and format text, multimedia files and how to set the properties for these constituents. Besides, I have acquired knowledge in using Toolbook's own scripting language named Openscript.

Adapted Material

Most of my program consists of pages that are created from the texts and dialoguess of Streamline English. Therefore, the pages mainly comprise of one ore more images, the text and audio material connected to a unit. Some of the units have been split up into several separate or related pages.

Text

A large amount of the application consists of text. This is not surprising fact, since no English teaching material can work without text, especially those ones that are aimed to teach beginner language learners. Texts help to improve reading skills and to some extent the writing skills of the student. It also helps to improve develop correct spelling. What is more, the texts of Streamline English are often funny, entertaining and varied, which makes language learning more enjoyable.

In the following section I will described how I retrieved the text from the book and also how I structured it and how and why I split it up into separate pages if it was necessary. I chose to digitalise the narrations of a story and the dialogues of each unit. In the case of dialogues I did include neither any dashes nor the names of the speakers to help transparency and to help students' comprehension. In every other aspect, I strived to keep the format and content of the text of Streamline English.

Images

Using images in language teaching and especially in a multimedia language teaching application has a lot of positive effects. The first and foremost advantage of using images is that they can hold a lot of information, which can help the student in the comprehension of the material and overall language learning. Pictures have universal meaning - they express the same or very similar message that is shared by every people all around the world. This is especially true for images used in language teaching books, because these books are intentionally created for a very large user basis, they are intentionally created to be understood

by as many students as possible. Even a student, who cannot speak a single word in the foreign language that he or she is learning, can obtain a lot of information by simply observing the pictures. And since the book is created for beginners who have little or no knowledge of the language, it is vital to include these images in the program. Furthermore, it is obvious that images provide the fastest way of understanding in comparison to text or audio and since the book is aimed to be a very “intensive course for beginners”, this fact is another supporting idea for using images in the program.

In addition, images are the best way to provide immediate information on the context of the actual material of the situation or any other text of the unit. Images provide context because they can be instantly understood and contain clues for the content of the unit and express the meaning of the unit. Before even reading or hearing anything, the student will acquire a lot of information related to the unit. By simply looking on the image or images, the student gets some help to determine the context and gets an immediate glimpse of what the text and the audio material will be about. This will help understanding while reading or listening the material of the unit.

Besides providing information prior to reading or listening and providing information that is contained in the text or audio material, images can provide additional information to these sources. A little smile on the face, a sneaky look or a bored expression can slightly or greatly alter the meaning that has been acquired during reading the text or listening to the audio.

What is more, images can contribute to the creation of an education program of high standard that has immense style. It can give colour to language teaching and learning particularly in the case of Streamline English, because in contrast with many modern language books that are created for adults, this book often has very funny and humorous pictures that try to represent the comical side of every situation. Therefore, using the images of the book may greatly make

language learning more enjoyable, more fun and at the same time avoiding students get bored of using the program.

Audio

Using audio material also bears many advantages. For a student who wants to learn any language, it is quintessential to listen how the users of the language speak. To begin with, probably the most important positive aspect of using audio in a CALL application is that it can show the student how the speakers of the language speak, how they utter words, how they stress word, what is their speed of voice and many other things that they could not obtain otherwise. The speakers of the foreign language provide a model for the learner, which he or she can imitate and after appropriate amount of practice, master. Furthermore, using audio material in the program makes language learning more enjoyable, less dry and monotonous.

Besides these general positive effects, Streamline English provides some additional advantages. First, during the whole audio material students can hear original British speakers with original British accent. In contrast to many contemporary Hungarian language teaching applications, where the creators often use Hungarian speakers who speak English language fluently, though there is always a conspicuous difference between these kind of speakers and native British speakers. This book uses native speakers, which makes it – and thus the program - much more authentic and reliable. It also makes the student more comfortable when being in England or speaking with a native speaker of the language, since the pronunciation will not be so uncommon, they will be more accustomed to it. This fact is further supported by the fact that the book contains a lot of everyday conversations, where English people speak about everyday topics in such common places as pubs, shops or at the hairdressers. This may be much better than the stagy, cramped, artificial conversations used by many English books and applications.

Another advantage that the book provides with audio is humour. Most of the audio clips are humorous, funny, varied and accordingly make listening a joyful activity for the language learner. Besides these, the audio clips are appropriate in length, not too short and not too long either.

In addition to the general advantages of audio material in language learning and the additional advantages of Stramline English in relation to audio, using a computer application to support the advancement of listening has also got some great positive effects. The users of the program can listen to the clip not only once but several times, can play back the audio file and clarify any unclear points that occurred during the initial listening. They can also pause the clip any time they want. These features make it possible to effectively practice and polish their articulation and pronunciation and also to compare the written and uttered versions of the text of the unit.

Split-Up Units

Unfortunately, most of the time a unit had to be split up into more than one page in the program since the whole material would not fit into a single ToolBook page. Sometimes there were too many or too large images in the unit and sometimes the text or the audio was too long to fit into a single ToolBook page. Another common situation was that the unit was structured in a way that called for cut it into several logically coherent pages.

Text

Another problem why the units of the book had to be split in the program into more than one page is that the text of the unit was often too long to be displayed. In such a case, there is an option to decrease font size but this change has several negative effects. Firstly, it would render more difficulty to read the text and thus, have a negative result towards understanding. The student would have to concentrate more to read the text, would slow down in the reading and this would slow down the overall learning process, which might cause dissatisfaction and

peevishness. Moreover, while listening to the audio, the student would concentrate too much to the text and this would distract his or her attention from listening, which would make the listening task less effective. What is more, too small font size would exclude out those students from using the program, who have some problems with their sights. For these reasons, I decided to split up the unit where the amount of text reaches the level where it does not fit to one page with an appropriately large font size.

Images

Pasting too many images into a page would affect negatively the overall aim of the program, to effectively teach English. The flow of images would distract attention from the text, audio and overall content of the material. Students would spend too much time with the examination of these images and could not really concentrate to gain information from any other useful material. Moreover, these images are often funny and in correlation with the aforementioned things this positive trait might affect understanding negatively if there would be too many images in a single page. Furthermore, too many images would occupy too much space on the screen which would leave only a small space for the text. If the text has only a small place, the size of the font must be decreased and this would have a negative effect for the user.

Audio

Another determinant which caused a unit to be split up was the audio material. Sometimes the audio of a unit was too long and sometimes it was structured in a way that splitting was required. Having a too long audio stream for a relatively long time would cause the student to loose concentration and this would result in decreased effectiveness level of the listening task. Besides, a long audio stream most of the time results longer text, which – as described above – infer splitting up the unit. Secondly, sometimes the audio was structured into separate parts, which made it necessary to split up the unit. For example, if a unit had four dialogs, it seemed more logical to split the unit into four pages with four dialog texts, four images and four audio

files instead of including all these twelve elements - or leaving out some of them - into a single page, which would be quite impossible.

Of course, splitting up a unit into two, three or even four pages has the negative effect that the user experiences the correlated material of the unit in a fragmented way, this situation is far not as wrong as the negative effects of not splitting.

Omitted Material

I decided to incorporate some images, part of the text and the related audio material of the units in a ToolBook page. Most of the pages in my program contain these three elements, however, there are some pages that omit some of the elements, or on the contrary, contain some additional supplementary material. My main aim was to make to material of the program as simple and easy to understand as possible and to help the user to comprehend the material and master the language as much as possible. I tried to subordinate all the designing directives to these principles, even if it meant leaving out some material that was included in Streamline English.

I had to decide which parts of the units of Streamline English should get into its digital version. Of course, I could not use all the material that was included in the book for several reasons. First, if all the exercises and other material would have got into the program, it would have result such a robust, long and never-ending application that would be boring for the user, would create dissatisfaction and disillusion and would take too much time to go through the units.

Secondly, there were some materials in the book that were very difficult to create a digital version of. Some images were hard to scan in a proper way because of their setting or position, and even if scanning was successful it was not uncommon that some text or another picture covered some part of the picture. These flaws are very hard to correct even with

advanced photo editing applications. Therefore, I decided to leave out these images or replace them with another image.

Furthermore, there were many exercises that by their nature were very hard to be programmed into the application. Due to the constraints of programming, there are some things that are easier to be programmed than others. While some exercises were multiple-choice tests which are easy to be programmed, many of the exercises needed a teacher or a tutor who could check whether the student gave the right answers. Today, computer science and artificial intelligence is far not developed enough to fulfil this task and replace the teacher. Though, it would be possible to accept only a single solution in written form, but this might make the student believe that there is only one single solution and the least deviation from the sample solution could not be processed as a right answer by the computer.

There were other technical difficulties. Many of these exercises were made rather to be oral exercises and not a writing exercise, which is hard to implement in a computer program. In addition, there were some exercises in the book, where the student had to write the solution to an image and this could be hardly realizable in a computer program. For example in the instance of unit 36 the student has to fill in a form that is printed on the page. (see Figure 12 in Appendix) In this case cutting out the page, pasting it in ToolBook and achieving that the student could write on the image would have been quite complicated. One solution could have been to leave out the image and create only a table where the student has to fill in the asked data, but this would have greatly reduce the pleasure of the exercise and make it a rather dull exercise. As a consequence, I decided to leave out these kinds of exercises.

I also left out those exercises that I considered unnecessary for some reason. Firstly, there were a few parts and exercises that did not really connect closely to the unit. Secondly, there were some omitted exercises because they were objectionable in their methodology. However,

since the book was written by acknowledged experts in the field of English teaching the occurrence of the latter two problems occurred only in a few cases.

Although the book is considered of high standard and to have immense style and besides aimed to give colour to language teaching and make it enjoyable, there were some often recurring elements. An often occurring phenomenon that resulted in leaving out a part of the unit was the recurring of certain types of exercises. There were typical exercises that appeared in a large number of units with the same or very similar structure and there were cases when more these similar exercises occurred in a single unit. Of course, there were variations in the content and style of these exercises but after all they were very similar in structure. Making these pieces part of the program might have had several negative effects. Firstly, using the same or very similar type of exercises would make the program quite dull to use. Furthermore, it would lead to disillusionment and dissatisfaction since the user should do the similar exercises and watch the similar parts over and over again. This would go contrary to the principle of making the application enjoyable. What is more, by having to do the same exercises continuously, students would get used to it, would answer extempore with little demand to think and this would decrease the efficiency of language learning.

Sometimes some components of the units were merely left out because they were not as interesting and enjoyable as other parts of the book. Since the material had already been very enormous, having dull exercises in a program might cause the negative effects that I have described in the previous problem. I decided to leave out these components because they would were not sufficient for the quality measures of my application and would decrease the efficiency and enjoyment of the program.

Another reason why I left out the exercises was because in many cases there was no need for any further practicing of the unit. Over-practicing would cause the same negative effects that I have described above in the dull exercises.

Technical Implementation of the Program

First, the overall look and design of the program had to be planned. The whole program in ToolBook is called a Book. This book consists of several pages. In these pages text, images, video, hyperlinks, buttons and many other more elements can be placed at. These elements have their own properties which can be changed during the design of the program or during the running of the program. But Toolbook is not only capable of inserting these things in a page. There is a programming language called Openscript implemented into ToolBook with which a programmer can reach a very wide range of programming tools. Openscript is a very high level language which aims to ease the creation of multimedia applications while maintaining flexibility. With this scripting ability it can be said that ToolBook provides most of the programming possibilities that professional languages, for example C and Java, nowadays provides. It does this in a very comfortable and quite flexible way.

Before creating the whole application, it was logical and necessary to plan how the program would look like and be structured and to create some sample pages from the first few units. I processed the first five units of the book and created their digital versions. This occasion provided me an opportunity to experience with the possibilities of ToolBook, try out how the script would be built up and how I should digitalise, split up and categorise the multimedia files that I would use in the final program. Besides, I could also avoid the mistake of building up the program without preliminary planning and finding out the mistakes during the middle of the work. In this case even a minor mistake or error could result in restructuring and reediting all the material that I have put into the program, which would have been an enormous job to execute. After deciding the final design and structure, my task was to go through the units of the book one-by-one and to work them up according to the first five units that I have created and tested earlier.

A large part of the program content is based on the multimedia source files: text, wave files and images. These files make up the majority of the size of the application. The text and image files became integrated into the running exe file of the application, while the audio files remained in separate files in the folder named Audio.

To create a digital CD-ROM version of Streamline English, most of its content had to be digitalized. This means, that its pages had to be scanned, its text had to be typed in, and its audio cassette also had to be recorded into digital wave format. In some cases I had to adapt to Toolbook and use those file types that Toolbok was compatible with, for example wav instead of mp3 or the compressed jpg instead of bmp.

The program is made to be run from the C:\Stream directory. Due to the fact that this version of Toolbook was publisher more than 10 years ago and used the older file-name system, I had the constraint to reduce file and directory names to no longer than eight characters. This property had its effect in all the file and directory names that can be found in the Stream directory or its subdirectories. Generally, the name of a source file of the program consists of a unit identifier and a page identifier and the file extension – for example U07P02.wav. The unit identifier (U07) shows which unit the source file is originated from. Some units of Streamline English had to be broken into more than one page in the program, that is the reason why page identifiers had to be used in file names. The page identifier (P02 in this case) then indicates the serial number of a unit. In the above example the wave file belongs to the seventh unit of Streamline English. In addition, the unit is broken into more than one page and the wave file belongs to the second page related to unit seven.

Digitalising Text

The largest and most difficult part of creating the program was digitalizing its text. This was not easy because the book consists of 80 units and the dialogs and other texts had to be typed in to the computer. At first I tried to manually type in the dialogs, starting from the beginning,

but this method took too much time, and not included any other text from the units. Besides, it eventually turned out that Toolbook cannot handle the new clipboard format that Microsoft Office uses. Therefore, I needed to find a new way to digitalise the text of Streamline English.

I decided to scan all the pages and use a character recognition program. I scanned the pages at 300 dpi resolution with 24 bit colours. This resolution was enough both to recognize text and to use the images later for other purposes. After scanning all the images I used the most widely recognized character recognition software, called OmniPage. This program is capable of differentiating text from image in a scanned page, recognize the characters and create a Microsoft Word document containing the text of the selected scanned pages. This way, I did not have to manually type in all the text from Streamline English and the whole digitalizing process became much faster.

Unfortunately the output Microsoft Word file produced by OmniPage could not be directly used with Toolbook. The first problem was that the text derived from the scanned pages was often fragmented or recognized in a wrong way (see Figure 13 in the Appendix). Another problem was that Toolbook used a different clipboard format from the new Microsoft Office software-families, and as a result the text from a Word or Excel document could not be directly copied into ToolBook. My solution was that I used a temporary single text file, where I copied the text of each page in the correct order and spelling. I saved these files in the C:\Text directory starting the naming from U01P01.txt. Using these files I simply had to copy the content of these text files into the ToolBook pages that I have created.

In the program every text is placed in a textbox, which is aligned to the left side of the page in most of the cases. This placement was reasonable, because most of the time the textbox of the pasted text occupied a great amount of space. Since it needed more space than the majority of the images and due to the design of the common background, there is more free space on the left side of the page. For the sake of consistency I tried to stick to this alignment, it was only

necessary to align the text to the right side, when a large-sized image should be placed to the left side.

The format of the character was also chosen with intent. By experimenting with the font size I tried to find the golden mean. On the one hand, if fonts are smaller in size, more text can be pasted into a single page and therefore there will be less split units, which is better for the student. However, with a too small font size, reading the text would be harder and probably slower, not to speak about those users who are handicapped in their sight. On the other hand, if fonts are larger in size less text can be pasted into a single ToolBook page, and an inappropriately large font size would not guarantee better readability either. In the end, the font size was chosen to 16 with bold font style. With choosing the font type I did not have too many problems. I chose Arial font type because it is very widely used.

The colour of the text became red, since this colour is quite conspicuous and at the same time easy to read. Besides it fits to the colour of the background. Around the text the thin border of the containing textbox is chosen to be light orange to harmonise both with the text and the background colour. This colour composition and text formatting was only changed when the text had to be separated in a single page according to the difference between the two texts. For example if a story had an introductory part, this introductory part was placed in a disjunctive textbox, it became smaller and was coloured green to be differentiated from the body of the text.

Digitalising Images

I also included the pictures of Streamline English in my program. First I had to scan all the pages of the book and then cut out the relevant images that I needed for the software. I had to take care for not setting the scanning settings to too large because it would have taken too much time unnecessarily and I could not even use the advantage of it since the screen size of the program is set to 800x600 pixels. On the other hand, I also had to take care not to set the

quality to a too low level because then the smaller images of the book would become pixeliated during their enlarging. Therefore, I decided to scan the pages at 300 dpi resolution with 24 bit colours, which is approximately the optimal quality for my actual application. For this quality it was enough to use an average scanner with its scanning software. These scans were later used for both cutting out the relevant images and for the character recognition of the text of the book that I have described above.

After the scanning was over I had got 96 scanned pages but these could not be directly used for my program since it was a raw image containing the text, images and unnecessary parts of every unit. Therefore, I had to cut out the images that were relevant and important for my application. I used to cut out those images that were connected to the dialogs and those ones which were otherwise necessary for the program (esetleg ide egy mintakép és hogy azokból miket vágtam ki). I used the built-in image editor program of Microsoft Windows, named MS Paint, and sometimes Adobe Photoshop. Since advanced manipulation was not needed in most of the cases it was enough to load an image to Paint cut out the part of the scanned image that I needed, create a new image file and paste in the cut-out part and save it. This process was repeated with the scanned image while no other part was needed to cut out. This whole procedure was done until all the scanned images were processed. The result was 170 images in 12 Megabytes.

The images always had to be imported into ToolBook by the Import Graphic menu item of the File menu. Originally I wanted to save the images in Bitmap format since it is not compressed and thus the quality of the original scanned images would have remained the same. However, since Toolbook is not able to resize BMP images, I had to use the compressed jpeg format. Although jpeg uses lossy compression, that is, some part of the data loses during the compression process in order to provide the smaller file size, this property did not make its

harmful effect felt while using these images. No quality-loss or pixilation is perceptible even while closely observing these images in the program.

However, I still had to take care for maintaining the appropriate quality of the image. The reason for this is the fact that most of the time I had to resize the images and if not taken carefully, this process can easily lead to the loss of quality in some way. Although, the pages of the book were scanned in good quality, it contained a lot of small pictures that were necessary to be enlarged while creating the computer program. If these images are enlarged in a great extent, their size is multiplied and the phenomenon of pixilation occurs. This means that because raster images consist of determined number of pixels and the number of these pixels needs to be multiplied in the enlarging process, if we extend the size of the images too much, the quality of the image severely decreases and the picture becomes raunchy. On the other side, leaving the images in their original small size would have resulted that the users of the program would not have been able to see it correctly, which would lead to dissatisfaction and to the loss of efficiency. Therefore, I had to find the balance, had to find the optimal size of the image during resizing, not making the image too small or, on the other hand, pixilated.

On the contrary, scanning the pages of the book in good quality caused another problem that is the opposite of the previous one: if the size of the mages is decreased too much, the image also loses from its quality and the smaller items that can be clearly observed in the original picture can be hardly seen in the small image. The sizes of the scanned images were 826x1169 pixels, while the application runs at 800x600 pixels. There were many cases when a picture that had to be pasted into a ToolBook page almost took the whole page in the Streamline English book, and since they did not fit into the page in their original size, they had to be decreased. This caused the aforementioned problem of losing quality; therefore I had to take care for this issue.

Another problem was that while decreasing the size I had to take care for mainly maintaining the aspect ratio of the image. This was a frequently recurring problem during reducing image size, since the majority of these large images was longer either horizontally or vertically. One solution for this was cutting a part of the picture that had little or no relevance to the overall image. Another solution was reducing the size in a way that left most of the information that the picture hold.

While using the images the program continuously grew in size. Therefore I had to take care not to expand the size of the executable file in an enormous extent since this would require more memory to be occupied, which would cause the program not to run in older systems.

Most of the images are aligned to the right side of the pages, only some larger images were aligned to the left side, because there was more space for them. While placing the images I also had to take care not to cover the background image and buttons with the image. I decided to use no border around the images since they already differentiate from the background and no transition is needed.

Digitalising Audio

I did not have an easy case with the audio material of the book either. With Streamline English comes an audio cassette, which is a very important part of the book, since many of the book material builds upon it. Therefore, the whole audio content was used, which had to be digitalised. Firstly the content was digitalised in standard PCM format with 44 kHz sampling rate and 16 bit per sample. The cassette was not digitalised in one large file, but was separated into separate files in concordance to the units.

Unfortunately, these audio files could not be directly used in my software. First, they contained a voice at the beginning of each file saying the unit numbers and titles. Since this information is already available to the user on the user interface of my program, which always

shows the actual unit number, this was unnecessary for the application, and the beginning of each unit audio file had to be cut out.

While cutting the audio files I have also left out many of the intermissions, where not a single sound could be heard. This was necessary, because these pauses were often quite long and this would have left an unnecessarily long pause in the voice which is not present in the text of the units. Besides, while listening to the audio material the user of the program might believe that the audio is over. I have also cut out the mute ends of the audio files. On the other hand, I left those part of the audio files unchanged where sound from the surrounding environment could be heard. This could be the background sounds of a pub, an aeroplane that took off or chattering voices in the background at the beginning of an audio file. I intended to leave these sounds untouched and remain part of the audio material of the program since they can help the students, the users of the program to recognize the context of the actual audio file.

Moreover, in many cases the dialogs were so long that they must have been divided into several pieces in order to make their corresponding text fit into one page in Toolbook. I had to use a wave splitter program and since there is no such an application built into Windows, I had to obtain one. I decided to choose the audio software named Visual MP3 Splitter and Joiner for several reasons. First, it was absolutely suitable for the job, since it was created almost solely for this task. Secondly, it was surprisingly flexible and easy to use with its easily understandable graphical user interface and easy editing features. Moreover, another advantage was that it could split an audio file into pieces in a way that its original format stayed unchanged. The only negative property of the application was that it was a trial version and I could only split up files into five pieces, but fortunately this constraint did not set back my work because there were no audio files that were needed to split up into more than five pieces.

I began with splitting these files one-by-one. There were seventy units for which audio material was provided and most of the audio files had to be split up into two to five pieces. From the 70 original audio files of the units 180 split files came into existence, which means that as an average a unit will take 2,5 pages in my final program. Unfortunately the name of the output file of the splitter program could not be changed by the user and the end-results were file names such as Unit03-01.wav. The problem with these filenames was that they exceeded the permissible eight-character limit. Therefore I had to rename all the files, which would have been a really hard work if I would have not used a batch renaming application. Fortunately, I found such application built in the popular and widely-used file manager named Total Commander 6.56. With the program's batch renaming ability I could easily and rapidly rename all the audio files. The final file names have a unit identifier, a page identifier and wav extensions. The overall size of the split audio files was 303 Megabytes. With lossy compressed audio files this size could have been radically decreased. Unfortunately these extensions could not be changed into the compressed mp3 format since Toolbook could not handle mp3 audio files, because in the time of its creating there was no such format present.

After creating these files I had to import them into my ToolBook application. Since ToolBook was created in a time when the management of image and audio files was still in its infancy, the import, use, and management of these multimedia files was quite uncomfortable and uneasy during the creation of my application. Firstly audio files had to be imported to the so-called Clip Manager, which - as its name suggests – is intended to hold together and help the management of these and other multimedia files. The problem with Clip Manager is that firstly each file has to be imported one-by-one and has to be named one-by-one to be able to be referred during programming. No group selection was possible like in newer applications. With such a large number of files that I used, this was quite a great effort to make. The naming of the audio clips was important, since the scripting was created in a way that that

audio clip was played after giving the play command (by clicking on the play button or pushing the P key) which had the same name as the actual page. Therefore each page had its corresponding audio clip in my ToolBook program with the same name.

Background

There is an option to define backgrounds in ToolBook. These backgrounds are useful when common elements occur in several pages such as a common background image. My program has one background page which constitutes the general outlook of the program.

In this background the user can find the title of the program in the upper left corner. I considered it important to continuously represent the program and book title because by this anyone, even those who have never seen the application before, could identify which program he or she is dealing with from a single glance. Although, I could give a different name for my application from the book from which it was created, for the sake of unity, I decided to select the same title. The font of the title was also consciously selected, I wanted to choose a font type which would correspond to the spirit of Streamline English and at the same time shows professionalism and ease. The size of the title was also important because a too large size would distract attention and take too much space, while a small size would make it insignificant. The final decision was Californian FB font with 20 point font size.

Besides the book title, the actual unit number also appears on the background page and thus in every page. The representation of this kind of information is important to the same reason as in the case of the title: to motivate and to inform the user about where he or she is at the moment in the overall material and how much he or she has progressed since the beginning. The size of this text is smaller than the title size since this information must not be as eye-catching as the title. The font type is also different to differentiate it from the title express.

Navigation

Easy and user-friendly navigation was also an important object of my program. I wanted my application to be as easy and straightforward to navigate as possible. That is why the user has the chance both to navigate by clicking on buttons and pressing a key on the keyboard.

The navigation panel was placed in the upper right corner. This placement is not by accident; buttons were placed there because many applications have their navigation panel in that place. The size of the panel is aimed not to be too small or too large but optimal in size. Too small buttons would be both hard to find and hard to click on, while large buttons would take too much space and distract attention from the significant parts of the interface.

During the creation of the buttons another very important goal was to avoid texts in any language on them but use pictograms and other signs instead to indicate their function. I did not want to write Hungarian text on the buttons because this program is aimed to develop English knowledge of its users and a Hungarian label would go against this idea. Besides, there is often a problem of drawing some Hungarian vowels such as õ or û because of the insufficient font definition. This phenomenon tends to appear especially in older systems and programs such as ToolBook. On the other hand I did not want to use English labels on the navigation buttons because a text label would involve the need to increase the size of the buttons. Instead of text labels I used images and signs that are familiar for users from other applications and represent a commonly shared and known meaning.

There are six buttons: a previous page button, a next page button, a home page button, a play/pause button, a stop button and an exit button. These five buttons are surrounded with a circle, which serves only to improve style and to apparently join the five buttons together.

In the upper row there is the previous and next page buttons and the home page button. The first two buttons help the user to navigate to the previous and next page. To help navigation these buttons are made very similar to the buttons used in other applications to indicate

previous and next. The previous page is indicated by a << sign and the next page is indicated by the >> sign.

Above these two buttons the Home button can be found. With this button the user can jump directly to the first page of the program which is often called the Home Page or the Main Menu. This menu helps to select between the options and also serves as an initial page when the application is loaded. To help understanding I used a well-known icon that is widely recognised as the home sign or home button. This button sign is used for example in most of the browsers to indicate the home button.

Under these three buttons the user can find two buttons that are related to audio control: the play/pause button and the stop button. These buttons were placed into the graphical user interface to help the user control the dialogs and other audio material. This feature makes it possible for the learners of the language to examine the audio more closely and clarify understanding. Moreover, this way students can further develop their listening skills by hearing the text again. This option may also come handy when the user is interrupted due to some external reason and has to make an intermission during the learning. Both the icons of the play/pause and stop buttons are iconic representations of their respective meaning, known by everyone all around the world.

The play/pause button could have been separated into two distinct play and pause buttons but the merged solution applied in the program is more widely used today due to the new digital devices. What is more, by this solution some space could be saved in the user interface. The play button appears by default when the program starts. During this state if the user clicks on this button, the audio file that belongs to the page will began to be played. While any audio file is being played the button changes into a pause sign. During this state if the user clicks on this button, the audio file that is being played will be paused and remain at its paused position. If the user wants to continue listening to the audio file, the only thing he or she has to do is

clicking on the play/pause button again. During scripting I had to take care for changing the icon of the button into pause or play not only when the play/pause button is activated, but also when the audio file is started to play by a different method, for example by pushing P in the keyboard.

The stop button works in a similar way, that is, it was designed to resemble the commonly known features of the button. It takes place right to the play button to resemble to common placing habit. Moreover, its icon was chosen to resemble its iconic image and works as any other stop button. If the user clicks on the stop button while playing an audio file, playing will stop. However, if the user clicks on the stop button while not playing an audio file, nothing will happen. Besides, any audio file that is being played in the program will also stop if the user advances to the next page.

Besides these five buttons there is an Exit button in the application in the upper right corner. With this button the user can immediately quit the program whenever he or she wants. This gives the user the feeling of safety, because the student feels that he or she is in control. This feature should be a basic minimal requirement in creating any educational or non-educational software, however, it is absent in many of these applications. The icon of this button is well-known for anyone who has ever used any operating system. This cross represents quitting from an application both in Microsoft Windows operating systems and in Linux operating systems. Moreover, I consciously placed this button to the upper-right corner because in these operation systems the close button is also in this place.

Besides using the navigation and control buttons of the graphical user interface, the user can also activate the commands triggered by these buttons by using some selected keys keyboard. Using the keyboard to control the program is often faster, more comfortable and simpler than having to click on the buttons. It is faster, because there is no need to reach for the mouse, position the pointer to the selected button and click. If this may not sound enough convincing,

we may consider that today more and more people buy laptop computers and these devices use touchpad as their default pointing device, which is quite uneasy and slow when used as a pointer device. On the contrary, by using keyboard, after some time the user will intentionally find the right key to push on the keyboard and in this case the program is created in a way that mouse usage can be completely omitted. This way, the user do not has to move the hands off the keyboard and using the program will be faster and more comfortable.

The setting of the navigation and controlling keys also supports this idea. I consciously strived to use the keys for navigation that are widely accepted and used, known by most of the potential users, and besides, easy to remember. For example, I used the Home key of the keyboard to navigate the program to the initial page, the right arrow to advance to the next page and the left arrow to go back to the previous page. In addition, the user can play and pause audio clips by pushing the P key and stop these clips by pushing the S key. It is apparent that P and S are the initial letters of Play/Pause and Stop and therefore they are very easy to remember, while the Home key, the left and the right arrow are used in many applications for the same purpose as it is used in my program.

Conclusion

During the creation of the program there were some barriers, but most of the time they could be solved or evaded. Throughout the design, construction and implementation of the application one of the main aims was to make its structure as simple, clear and logical as possible. It was an important objective to create a user-friendly navigation with simple buttons and clearly arranged graphical user interface. I feel that these goals were successfully executed and the usage of the program is clear and straightforward for its potential users.

It was also an aspiration to include a wide variety of subjects and topics for teachers. This was achieved by adopting the vast majority of the units of Streamline English. The text, images and audio of almost all the units was digitalised. Since the aim was not to provide a stand-alone Enlgish language teaching tutorial for students, the exercises for the units were omitted. Teachers are free to incorporate related tests and exercises, even by using the exercises of the book.

Therefore, in this form the program is a digital supplementary material for teachers, which includes a wide range of topics with texts, images and audio. Teachers can choose to use these units to make their lessons more various, rich and colourful. These pieces of material used in beginner English classes can make the introduction of new knowledge or practicing the language more interesting and entertaining.

Besides, the program can be useful by all language learners, who want to revise and brush up their language knowledge. In addition, since the program contains a great deal of audio material, students can develop their listening comprehension.

As a possible way of further development, the program could be changed to be an English language teaching application for beginner students. For this purpose, practice exercises and a vocabulary should be implemented. Although these parts are not yet available, at the present state the program can be a great source for language teachers who want to make their lessons more colourful by using digital supporting material in teaching.

Appendices

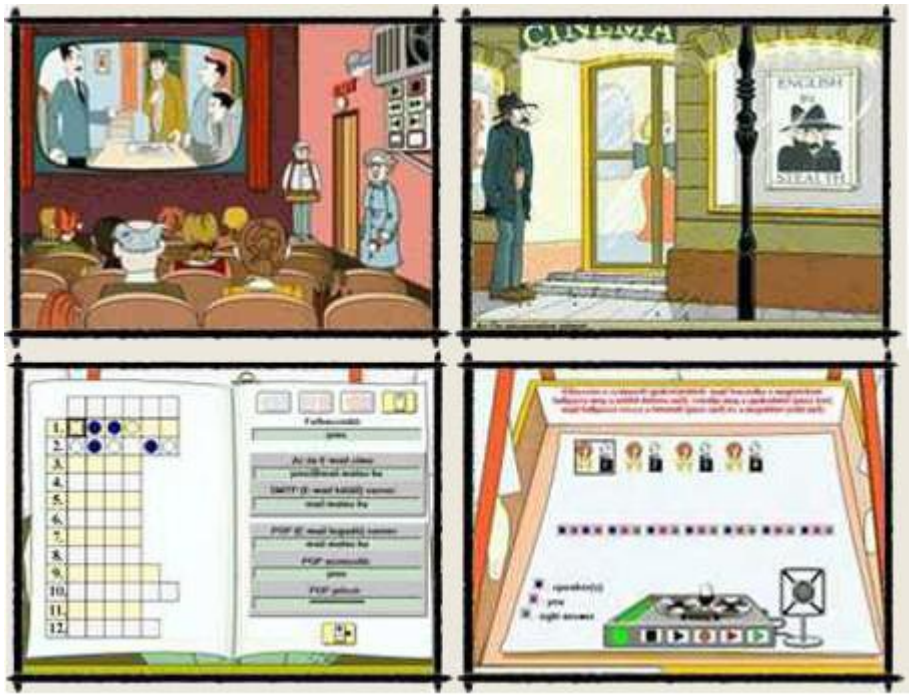

Figure 1. – Screenshots from Lopva Angolul

Figure 2 – Screenshots from Manó Angol

Figure 3 – Screenshots from ClipDic

Figure 4 – Screenshot from Nyelvész 3

Figure 5 – A ramification in Nyelviskola.

Figure 6 – The CD cover of Nyelviskola

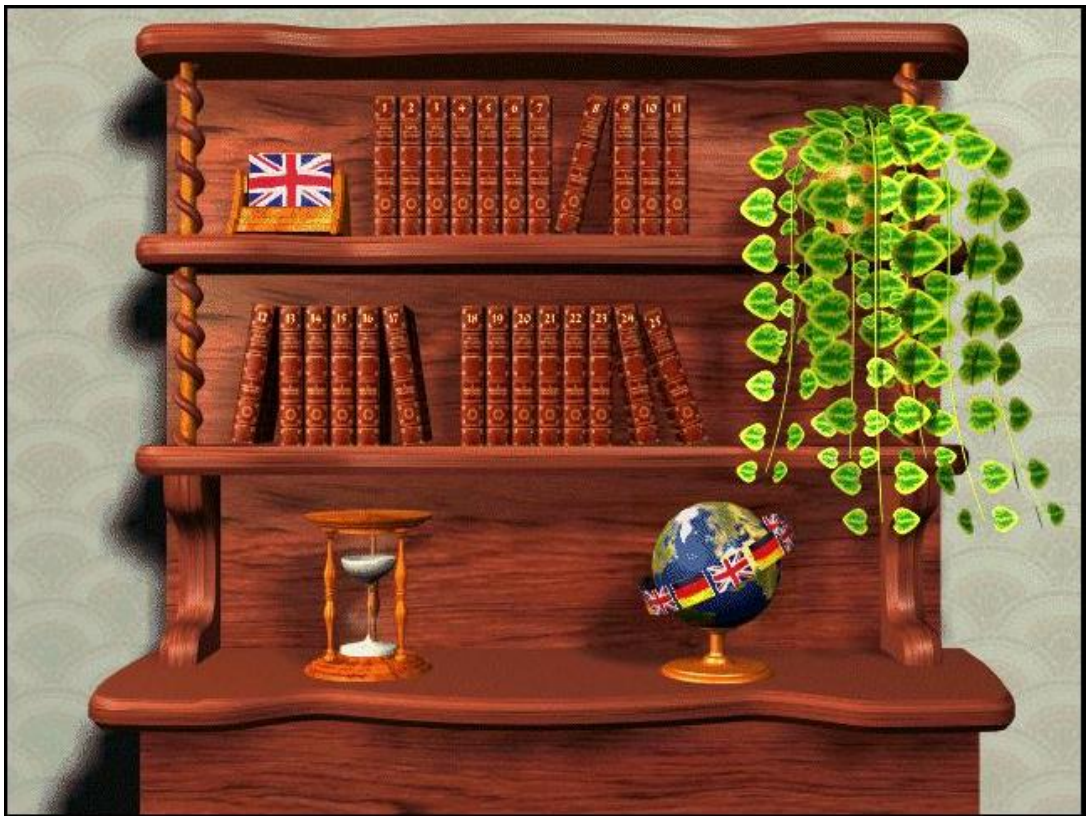

Figure 7 – The contents menu in Nyelvstúdió

Figure 8 – A screenshot from Nyelvstúdió

Figure 9 – Crossword puzzle in Nyelvstúdió

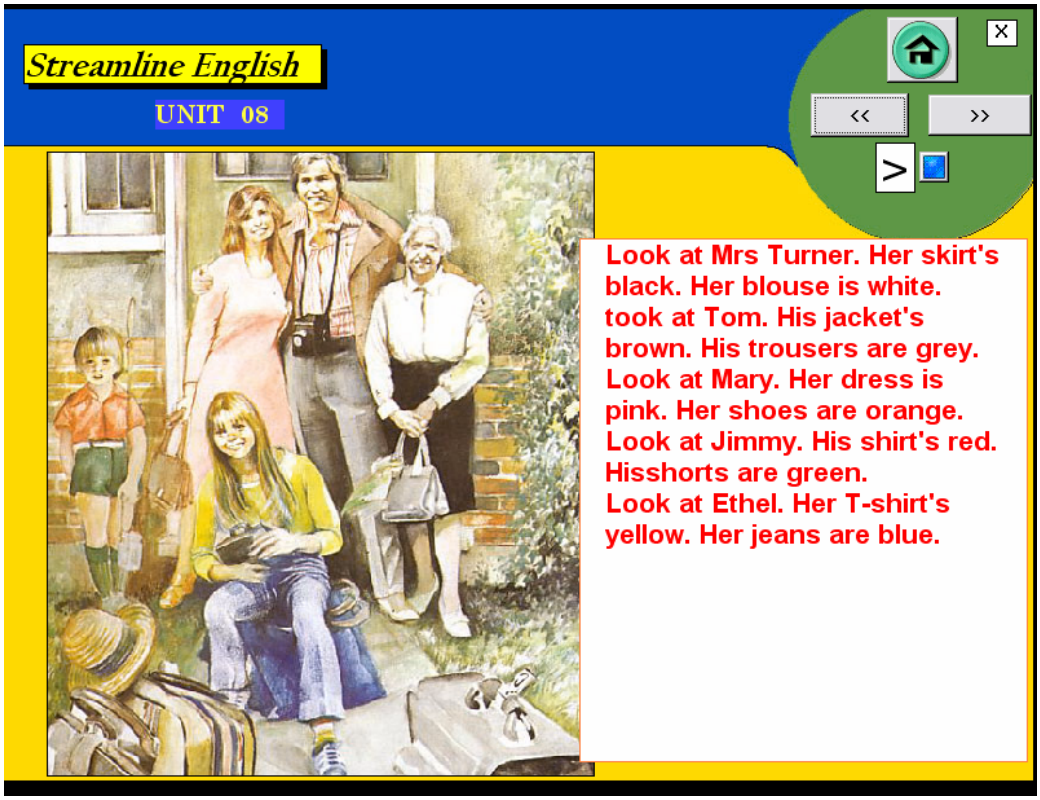

Figure 10 – A screenshot from the created program.

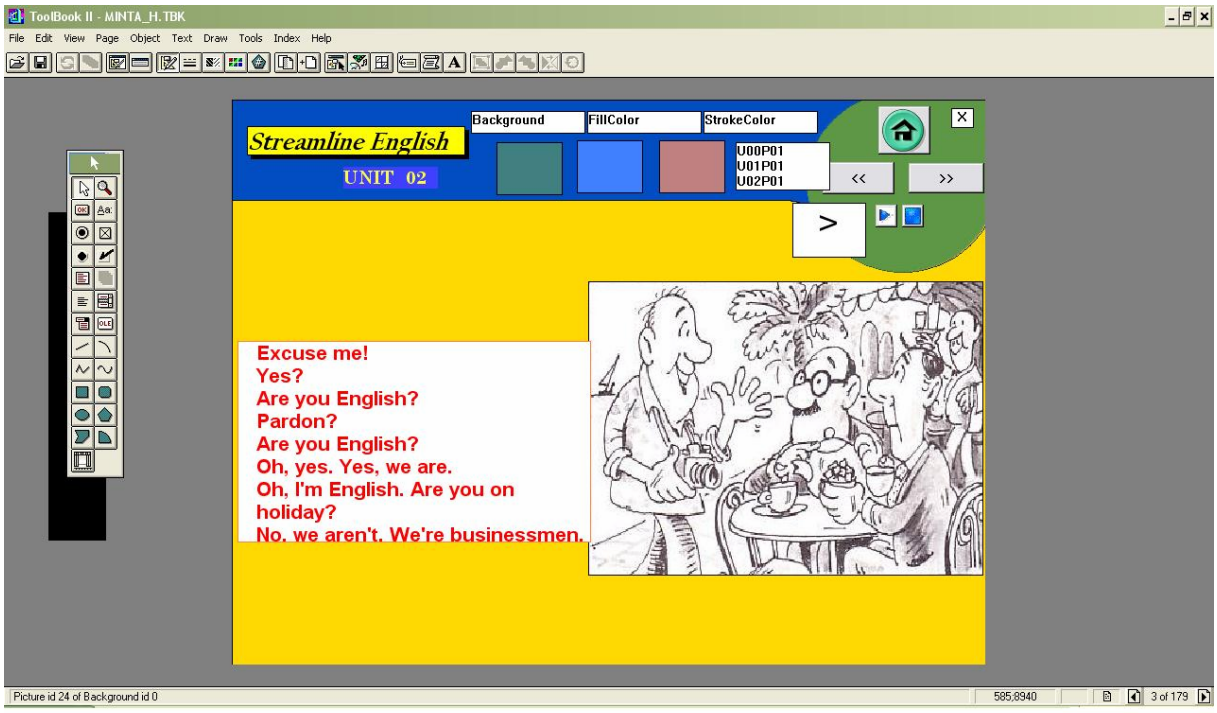

Figure 11 - A screenshot from ToolBook V.

36 A Questionnaire

Desmond Philton works for a Market Research company. He's asking people about their free time.

Desmond Good evening, sir.
Mr Norris Good evening.
Desmond I'm from Market Research Ltd. May I ask you some questions?
Mr Norris Yes, . . . yes, all right.
Desmond Thank you. . . . Now, what time do you usually arrive home from work?
Mr Norris Hmm . . . I usually arrive home at six o'clock.
Desmond When do you usually have dinner?
Mr Norris Oh, I usually eat at seven o'clock, but I sometimes eat at eight o'clock or nine o'clock. My wife works too!
Desmond What do you usually do after dinner?
Mr Norris Well, I sometimes go out, but I usually stay at home and watch television.
Desmond How often do you go out?
Mr Norris Oh, not often . . . once or twice a week.
Desmond Do you often visit your friends?
Mr Norris Yes, I do, quite often. I sometimes visit them, and they sometimes visit me.
Desmond Do you ever go to the cinema?
Mr Norris Oh, yes . . . yes, I do.
Desmond How often?
Mr Norris Well, I occasionally see a film . . . I like horror films . . . *Frankenstein* or *Dracula!*
Desmond . . . and the theatre? Do you ever go to the theatre?
Mr Norris Yes, I do . . . but not often. I rarely go to the theatre.
Desmond Hmm . . . Do you ever go to the ballet?
Mr Norris No, never. I don't like ballet.
Desmond Well, thank you Mr Norris . . .
Mr Norris May I ask you a question?
Desmond Yes?
Mr Norris What do you do in your free time?
Desmond I ask questions, Mr Norris. . . . I never answer them.
Mr Norris Oh!

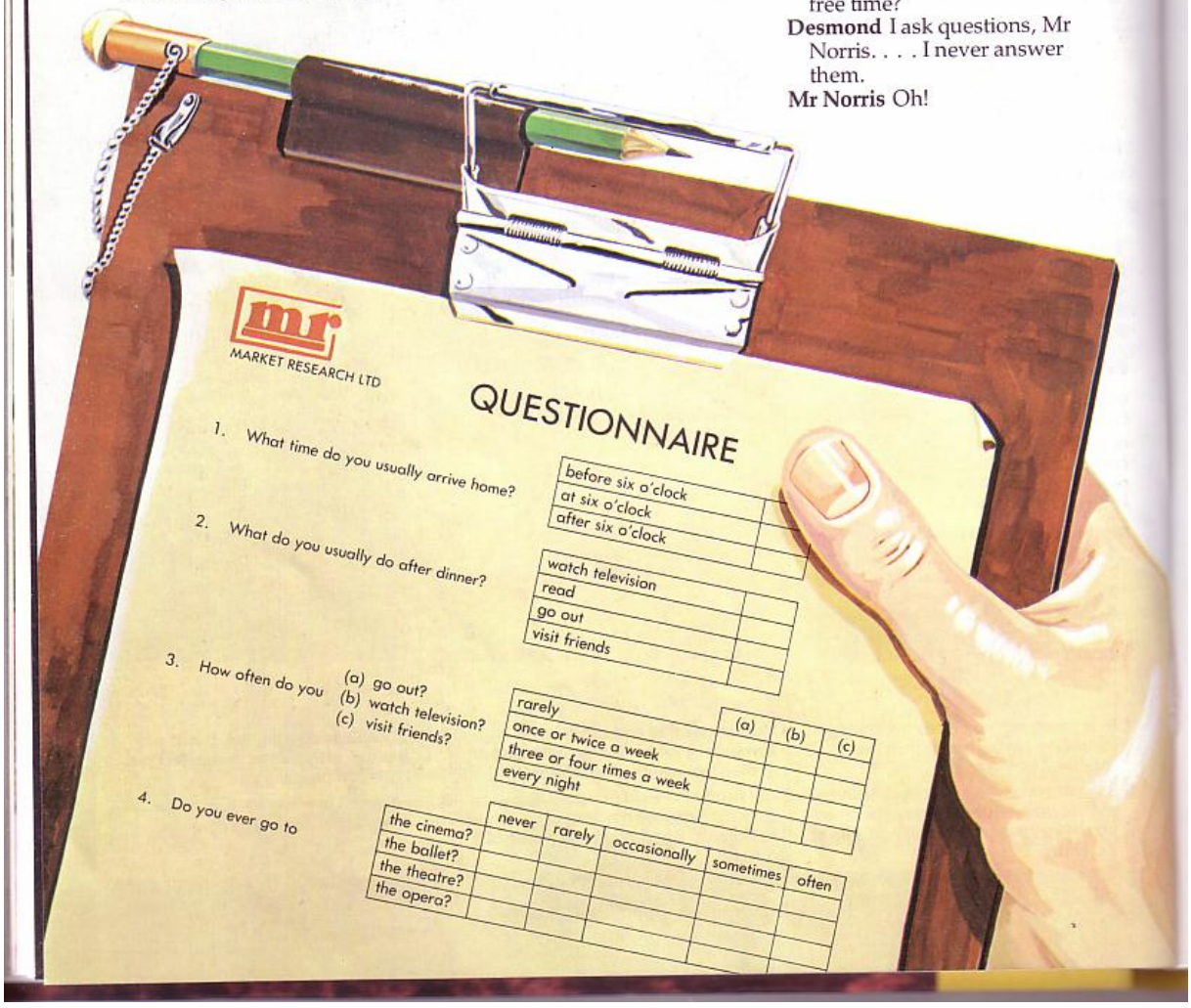

Figure 12 – The scanned image of unit 36 from Streamline English.

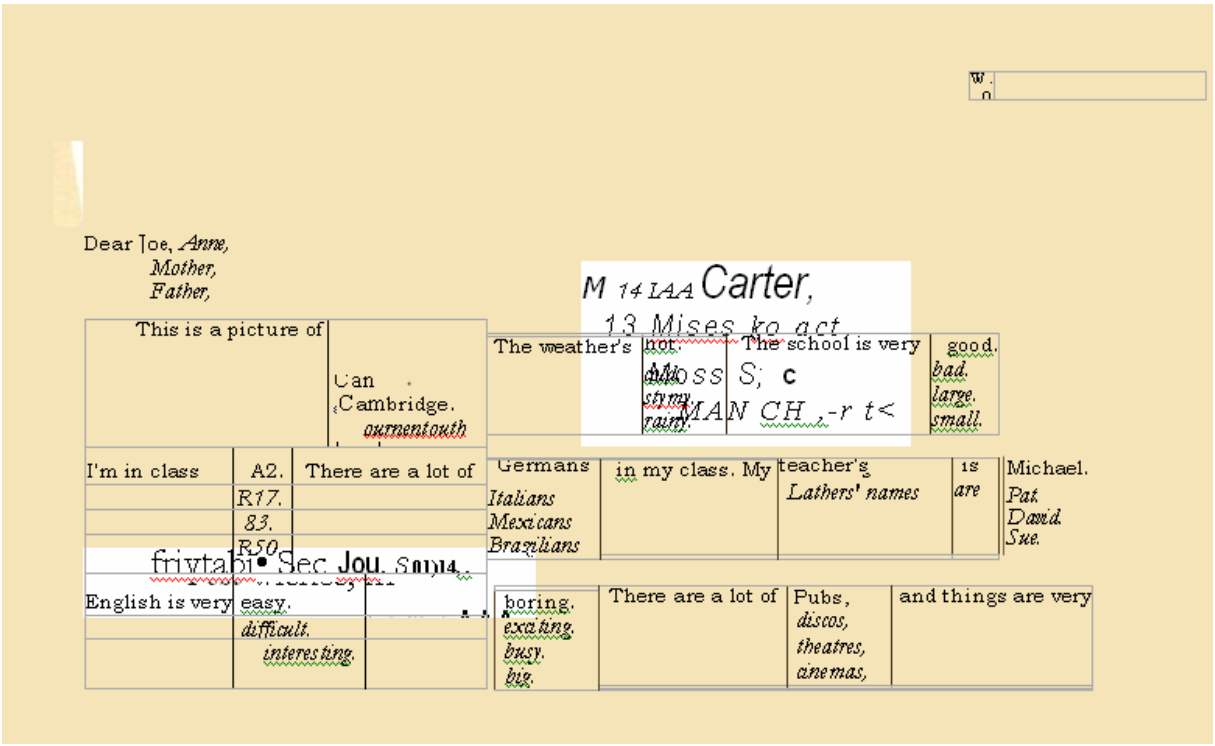

Figure 13 – Errors in the character recognition.