Training and Assistive tool to understand Danish texts using eye-tracking

Aalborg University
Institute of Electronic Systems
Department of Communication Technology

Project Group 1020, 2012

AALBORG UNIVERSITY Institute of Electronic Systems

Fredrik Bajers Vej 7 ▪ DK-9220 Aalborg Øst Phone +45 96 35 87 00

Title: Training and Assistive tool to understand Danish texts using eye-tracking

Theme: Vision Graphics and Interactive Systems

Project period: February 2012 to June 2012

Project group:
1020

Group member:
Philippe Murison

Supervisors:
Jakob Schou Petersen &
Lars Bo Larsen

Copies: 3
Pages: 90
Finished: 25th May 2012

Abstract: This project is issued from a project proposal concerning eye-tracker and the willingness to make a language study case, the main hypothesis of the project is: "Can a tool using eye-tracking assist and train adult learners of Danish to understand Danish texts?". The study of the reading mechanisms and the methods to learn a language have been the first step in the project. Then several parts have been reviewed and considered for the development of the future tools such as the device that will be used but also some knowledge concerning teaching to adults. A user centric design study has been done to have a convenient application for the learners using two low fidelity prototypes which lead to a final design. The implementation follows as close as possible the prototype. Some user tests have been done to get a general impression about the system. Those tests provides some clues about the fit between the assumption and the actual needs of the Danish learners and such a system could potentially assist to understand a text, however concerning the training part of the hypothesis longer tests are required to have an answer.

Preface

This project is written as a master's thesis at the Institue of Electronic system at Aalborg University. This report documents is the work of group 1020 in the 10th semester of the Vision Graphics and Interactive program. This report had been started the 3rd of February and delivery the 25 of may. Jakob Schou Petersen and Lars Bo Larsen from Aalborg University were the supervisors of this project.

Philippe Murison
<puris11@student.aau.dk>

Aalborg University, May 25th 2012

Contents

Introduction

Worldwide data on literacy show 861 million adults unable to read their own native language [Young-Scholten and Strom, 2006]. Concerning the second language (L2) and according to [Weber, 1995], the L2 population counts 564 million speakers in 1995. The data are slightly old and during the last 17 years the numbers have changed a lot, without the English readers (not available for this data) the L2 speakers population is 688 million [Lewis, 2009]. In [Crystal, 2003] it is mentioned that 1 billion people speak English, this number covers all levels from beginners to advanced speakers.

The proportion of the L2 speakers is evolving due to the geopolitics influence of the emerging countries. The Arabic and Mandarin have exploded during the last 25 years, at the opposite the countries with less influence have be drawn, such as French or Japanese as shown in the Table 1.

Table 1: Number of persons speaking an L2 from two previous studies [Weber, 1995] & [Lewis, 2009]

Language	Weber, 1995	Paul Lewis, 2009
French	190 million	50 million
English	150 million	N/A
Russian	125 million	110 million
Portuguese	28 million	15 million
Arabic	21 million	246 million
Mandarin	20 million	178 million
Spanish	20 million	60 million
German	9 million	28 million
Japanese	8 million	1 million

Concerning the Scandinavian countries, the numbers given by [Lewis, 2009] show that the number of native speakers of Danish is 5.6 million. However the global number of Danish speakers is around 6 million. This difference of 0.4 million between the native and all the speakers is approximately the same for Norwegian and Finnish

speakers. The population speaking those languages is highly concentrated in the corresponding countries and the international communications are made with English. The [Eurobarometer, 2006] study about the foreign languages spoken in the EU countries reveals that in Scandinavian countries and the Netherlands an important part of the population speaks English as shown in Table 2.

Table 2: Percentage of the population who speaks English [Eurobarometer, 2006]

Country	English speakers (in %)
Denmark	86
Finland	63
Netherlands	87
Sweden	89

Even if English is really present in those countries, it is often mandatory to speak the national language to work there. The economic stability and the culture attract a lot of foreign people but some of those Nordic languages are not easy to learn for foreign people, in this context Danish is considered as one of the hardest because of the gap between the spelling and the words' pronunciation [Lundberg, 1999].

This master's thesis being written in Aalborg University (Denmark), the environment trend to be better for the Danish study than any other language. The name of the master being Vision Graphics and Interactive Systems this project will focus on the eye-tracking technology for the vision part. This report will focus on the Danish language and how to train and assist the learners in understanding texts by using eye-tracking.

Chapter 1

Presentation of the project

1.1 Description of the project

Based on the visual aspect of reading and using eye-tracking technology, the project is about to build an assistive and training tool for non-native Danish readers and test it. Eye movements are analyzed through an eye-tracking system composed of an eye-tracker camera which provides data input to an application, this one uses an algorithm to analyze the input data and gives as an output some help on a screen for the reader when a difficulty is detected. As an example, if the reader gets stuck on a word, the eye-tracker detects it and the application displays the translation of the word. The eye-tracking part will be done with the Tobii X120 eye tracker camera [Tobii,], it is an eye-tracking device provided by the university, due to the time frame of a short project, the implementation of a dedicated eye tracker would have been too long to be included into the project.

The system will help the reader to understand a sentence by giving a translation; however, an interaction part can occur if the reader keeps staring at a word. In this case more information about the word such as definition, synonyms, an image or an example of the word in a sentence is provided. In parallel each word who required a help is kept in a database. The reader is thus able to consult a previous encountered word. The understanding of the text will be evaluated by some question (Multiple Choice Questions) concerning the text.

A secondary purpose of this project would be to find a way to automatize the translation process that is currently quite long on a computer. Opening a browser, going to a dictionary page, writing the noun to translate, all those steps take some times and could be optimized with tools such eye-tracking. However new methods would also include new problems that will have to be taken in account (e.g. the Midas touch)

1.2 Main Hypothesis

The Scandinavian countries: Denmark, Norway, Sweden and Finland are most of the time represented as a homogeneous group of unified nations, because of their geographical

neighborhood but also by similar languages, common historical and cultural traditions. The languages associated to each country are close from each other. Finnish is a special case and corresponds more to the Baltic languages (e.g. Latvian or Lithuanian) than the three others with a strong Germanic-Dutch influence. Danish is considered to have a deeper orthography than Swedish and Norwegian. There is a larger "distance" between written and spoken Danish than the others, this orthography involves more difficulty for the beginners than the ones of the other Scandinavian languages [Lundberg, 1999]. For this reason the project is centered upon Danish learning.

Around 6 million people speak Danish and 5.6 million are native speakers . Based on the eye movements and the vision of the reader, the purpose of this project is to know if eye-tracking may be a part of an assistive and training tool for the Danish learners

From this description, the initial problem of the project is: Can a tool using eye-tracking assist and train adult learners of Danish to understand Danish texts?

1.3 Basic scenario

The following basic scenario describes a sequence of actions that occur while using an eye-tracking system. A learner of Danish is reading a text on a computer screen. In one case, he is assisted by the eye-tracking system and not in the other one:

Scenario: Unknown word

- Without: the target starts to read a text in Danish and tries to understand it. When there is a word he does not know, he tries to get the meaning by the orthography, or the context of the sentence. If he does not get the meaning, he skips the word and maybe misses crucial information in the text.
- With: the target start to read a text in Danish and tries to understand it, using the system. When there is a word he does not know, he tries to get the meaning by the orthography, or the context of the sentence. If he does not get the meaning, the application gives him the definition or some synonyms he may know and then continues to read.

1.4 Report structure

This report follows the chronological rollout of the project and has the following structure.

- Pre-analysis part: the basic assumption is divided into sub-problems and previous studies and papers answer some questions about the project
- Analysis part: it explores all the future elements that will be used later on the design and the implementation

- Design part: it describes how from a first design and involving the users is built the final paper prototype
- Implementation part: it details the important elements of the built application
- Final test part: it describes the test done and the answer given by the testers
- Conclusion: answer to the hypothesis and present the future changes

Chapter 2

Pre-analysis

Detect a difficulty is the first step in the purpose to bring help; no difficulty equals no help required. Then what corresponds to a help for understanding a text. The mechanism of reading is performed by the eyes and the brain; a regular reader will read and understand a text at an average rate of 250 words per minute in a given fluent spoken language [Findlay and Gilchrist, 2003].

From the research question: "Can a tool using eye-tracking assist and train adult learners of Danish to understand Danish texts?", several sub-problems emerge, the first one consists in the detection of a reading difficulty and the second one concerns the help to provide to understand.

2.1 Difficulty detection

Reading is first decrypting a text through the vision of the eyes. It includes the recognition and the understanding of the letters then words and finally sentences. The vision and eye movements capture the information from the text and then the brain analyzes those ones. Finally the brain depending of its understanding of the text controls the eyes to reread or not a word. The first sub-part concerns the eye-movements itself and the second one focus on the understanding of the vision.

2.1.1 Saccades, fixations and regressions

The eye movements while reading are composed by three different movements: saccades, fixations and regressions [Frenck-Mestre, 2005]. The visual sampling is the segmentation of the vision, made by the eyes through the saccades and fixations. The fact that the eyes are not going through a text in a continuous manner had been discovered at the end of the 19th century by Javal; while reading eyes follow one line of a text with series of saccades, separated by fixation movements. Then at the end of a line another saccade led the eye to the beginning of the following line. Most of the eye movements are forward movements, however there are also the regressions that correspond to a

backward saccade, most of the time this regression shows a difficulty in processing the text [Findlay and Gilchrist, 2003]. The first pass fixation correspond to the 1st stop made on a group of letters, if there is a regression, there will be a second pass fixation (or refixation) that corresponds to reread a section of words.

As an example a fluent English reader will have an average length of saccade of 7-9 letters and an average time of fixation of 200-250ms and the average time between two saccades is 30ms. Those measures correspond to a quiet read of an English text, in this case regressive movements is not taken in account and the reading rate is about 250 words per min [Findlay and Gilchrist, 2003].

An important impact upon the speed of recognition is the position of the first pass fixation.

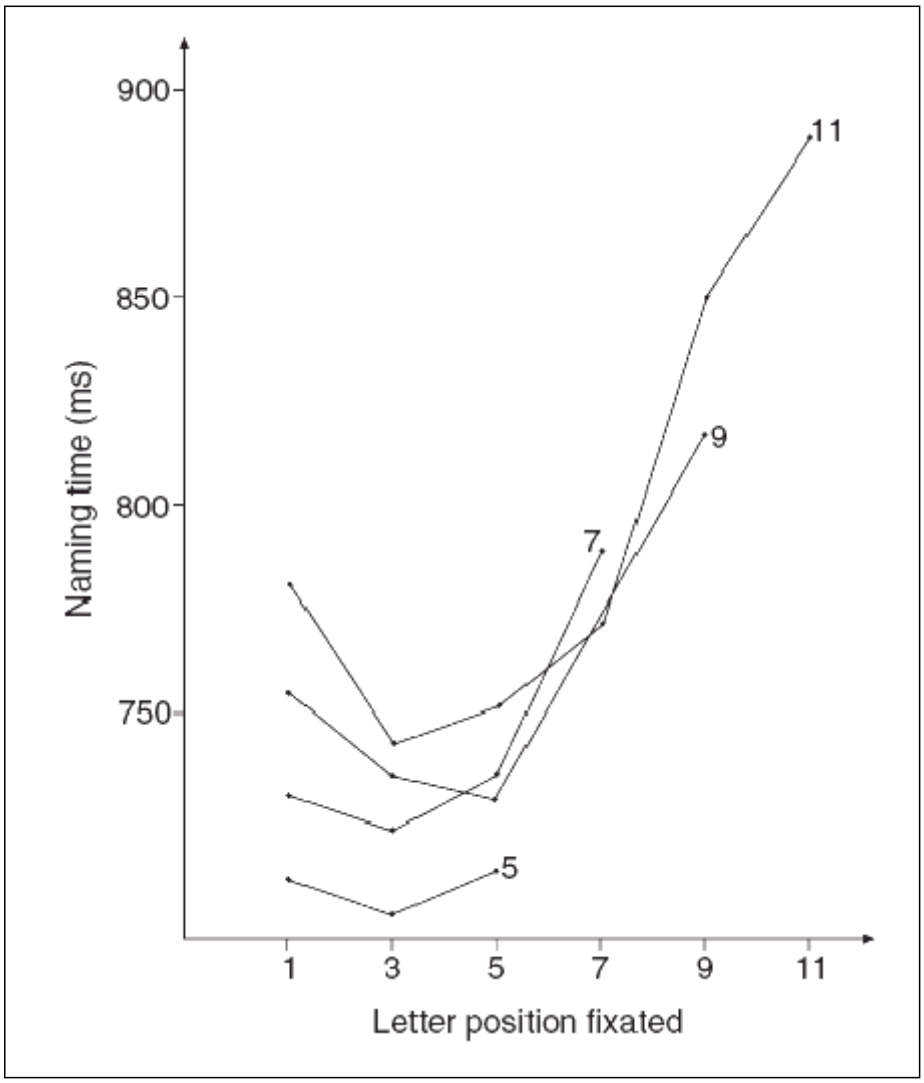


Figure 2.1: Optimal viewing position [Findlay and Gilchrist, 2003]

The naming time of a word depends a lot form the fixation point. The "optimal viewing position" is here located a bit on the left of the middle of a word, the more the fixation

point is far from this position, the more the naming time increase. This phenomenon is due to two facts, the refixation probability increases with the distance between the first fixation point and the optimal point, however even without refixation the named time increase with the size of the word itself (around 20 ms per character). Another point concerns also the position of the important information for recognition. The word "vulnerable" is recognizable after the first five letter that is not common in other words of the language, it is the contrary for words such as "underneath" with a unique second half but a common first one [Findlay and Gilchrist, 2003]. Concerning foreign language this "recognizable" part is not directly recognized, the probability of refixation is thus higher. Words are first recognized if they match the orthographic lexicon of the reader, the meaning is extracted from this amount of known words [J. ward and Parkin, 2000].

In the case of short words, eyes have different behaviors at their approach in comparison to long ones. In terms of saccades and fixations, while approaching a long word (with a length beyond six letters) the eyes will make a larger saccade than when approaching a short word. Fixations of the long words are most of the time shorter than short words ones. The main cause of this is that for long word, two short fixations are made instead of a long one [O'Regan, 1980].

As we can see here, long and difficult words to decode are often refixated which represent a difficulty.

2.1.2 Refixation and processing

Understanding is the part of reading where the meaning is extracted from a text. The visual sampling (cf. previous section) rate has to match with the speed of understanding of each sample. Understanding a word has an influence on the eye movements, this is what leads to the regression phenomenon described in the previous part [Findlay and Gilchrist, 2003]. Lexical access has an impact on language processing and this is applicable for both native and non-native language. An exception concerns the last words of a sentence, the "wrap up effect" increases the time spend on the final words. Not because of the word itself but sometime the sentence is being understood at this word, this area is not considered as the most reliable for measurement [Frenck-Mestre, 2005].

Linguistic structure also interferes during reading processing. An ambiguous sentence will always take more time to read and understand than a clear one. Sentences such as "Mary gave the book to Jane and she felt pleased", the automatic understanding will not be as fast as usual because of the "she" that could represent either Jan or Mary [Findlay and Gilchrist, 2003].

The more the reader is advanced in the language, the less there is regression. [Inhoff and Rayner, 1986] as shown that fixations on low frequency words were about 20 ms longer than fixations on high frequency words. Table 2.1 and Table 2.2 confirm this, when the text contains more specific vocabulary, the fixation duration increases (+60ms max), the saccade length decrease (-2.4 characters max) and there is more regressions (+15 %). The same variation can be observed with the level of the current reader, from the first year of school till

Table 2.1: Average type of parameter measured with different type of text [Findlay and Gilchrist, 2003]

Topic	Fixation duration (ms)	Saccade length (char)	regression (%)
Light fiction	202	9.2	3
Newspaper	209	8.3	6
History	222	8.3	4
Economics	233	7.0	11
Mathematics	254	7.3	18
Biology	264	6.8	18

Table 2.2: Eye movement parameters at various school grades [Findlay and Gilchrist, 2003]

School grade	Fixations per line	Fixation duration (ms)	Regressions per line
School year 1	18.6	660	5.1
School year 5	7.3	268	1.4
High School year 1	7.2	244	1.0
College	5.9	252	0.5

college, the number of fixations per line decrease by 12.7, the fixation duration decrease by 408 ms and the number of regressions pass from 5.1 to 0.5.

2.2 Way of helping to understand

Reading ability can be defined as "the mechanisms to interpret the information from a printed page"; however this definition does not tell about reading skills and strategies. [O'Malley and Chamot, 1990] divided the learning strategies of L2 learners:

- The cognitive strategies help learners to identify, manipulate and change the language. These strategies have been sub-divided in two categories mentioned by [English, 2003]
  - Memory strategies correspond to the remember and recall of key items, including word association and mental images
  - Compensation strategies which correspond to reading strategies like guessing lexical meaning from the text and making inferences
- The metacognitive strategies are used to apply the knowledge of a language (e.g. planify to use a word instead of another and mentally organize a sentence before saying it).

- The social strategies include the ask for help form the others when learning and get some feedbacks and correction (e.g. from a teacher)
- The affective strategies can be considered as the capacity to reduce anxiety and encourage the attitude of the learner towards learning.

These six strategies had been put in 2 groups, the one named "direct" strategies: cognitive, memory and compensation; the other one are the "indirect" strategies: metacognitive, social and affective; even if this classification does not indicates how effective each strategy is, the group corresponding to reading is the direct one. Getting the meaning of a word from the lexicon of the reader or from the context corresponds respectively to the memory strategies and the compensation ones. Word recognition is the first step for a reader to understand an entire sentence. The more he recognizes words through memory strategies, the more he will get the meaning of the corresponding sentences (compensation strategies). Indirect strategies concern more the mechanisms not active while reading but present while the reader as to use and learn the language.

2.2.1 Memory strategies

[Butorff, 1985] stipulates that persons with a Roman alphabet native language will transfer his skills to the targeted learned language. Readers with logographic script background (e.g. Chinese) without any contact with a roman alphabet will have more problem in decoding words. However if the person has used phonetic tools such as Pinyin (phonetic transcription of Mandarin in Latin's letters), it will be easier for him to recognize the word. Especially in Europe, most of the languages are linked and many words are shared in different languages. As an example a large vocabulary is shared between all the Germanic languages, the word "irritation", is the same in French, English, Danish and there is some little differences with: "irritación" in Spanish or "irritazione" in Italian. That kind of word can be given as synonyms and understood by the foreign reader.

Dividing the words into different parts for instance the negation and the word itself: unnatural would become un-natural; also provides help especially for readers with a non Latin native language. It can also be applied for a compound word with a frequent part, for example in Danish: "brødrister" could become "brød-rister". "Brød" is a quite familiar noun compare to "rister", less used. Finding the meaning of a word in a compound-word is a way faster if the first part of the word is a common noun [Y. Ko, 2011].

Starting from an unknown word, giving the translation or the meaning of this one should help both advanced leaners and beginners [Young-Scholten and Strom, 2006]. This gives great clues about the fact that giving the translation, a synonym, a definition or even separate compound words will help the reader (beginner or not) to understand a word or even a sentence.

2.2.2 Compensation strategies

[Alderson, ding] says "The ability to parse syntax into its correct structure appears to be an important element in understanding text" thus syntax also impacts the ability to understand the meaning of a sentence. Knowing that a word is a subject, verb or complement can provide information which could help the reader to understand the general meaning of the sentence. Adult learners of L2 differ from illiterate native speaking adults who read for the first time. If L2 learners do not have notions of phonology, morphology and syntax of the second language when they begin to learn to read, the ability to understand text will be compromised. "the ability to parse syntax into its correct structure appears to be an important element in understanding text". Knowing the syntax of a text is a key condition for understanding a text.

Another important skill is the ability to link the propositionw together, that could be crucial if the first proposition is needed to interpret the second or the contrary. Those mechanisms help the reader to gain access to all range of text understanding from literal comprehension (understanding words by words) to interpretive comprehension (understand a group of words with linked meaning) and finally critical comprehension (differentiate opinions and facts) [English, 2003].

[English, 2003] refers to two hypotheses concerning the impact of L1 reading ability upon the L2 reading one. Those two hypothesis:

- The 'linguistic threshold hypothesis' stipulates that there is a L2 linguistic ability threshold necessary before L1 reading ability can be used for a second language
- The 'linguistic interdependence hypothesis' allows first language reading ability to be transferred from L1 to L2 regardless of L2 level

Some studies shown evidence in favor of the linguistic threshold hypothesis, the L1 reading ability was only transferred to L2 readers with a certain L1 reading level and it was not the case for low-level readers. L2 readers with a high L1 reading ability are more accurate for understanding and reusing a part of the text than those with low L1 ability [English, 2003]. Understanding is linked with reusing the known structures, readers with a more solid L1 background would be able to apply their language knowledge upon the L2.

Providing syntactic information to the reader is also another way to help him. Second language readers are better at understanding a sentence when they know which role has which word in a sentence, but also when they have similar known structures. While reading a non native language text, have other sentences to compared with and the structure of a sentence can be a help for the reader.

2.2.3 Social strategies

To complete the helping, the user may also want more information for himself. This curiosity concerning the language is described by the social strategy [English, 2003],

this strategies include the help of others when learning, feedback and corrections from any person around. When the reader asks for more information, an application could give it and simulate a person's feedback. These strategies also reflect the learner's motivation and interest for a language.

Providing more information when the reader asks for it is a part of helping of the user to get more information about a word and then get a better understanding of a sentence. This kind of interactions while reading was impossible on paper texts but possible nowadays with applications and tools such as eye-tracking to know when the user wants some help.

2.2.4 Training the user

Providing help to read is a first step and then another aspect of this project will be to give the reader some training possibilities. A reader will be more skillful and proficient if the reading process is more fluent and automatic for him, L2 readers are required to have a minimum vocabulary stored in the long-term memory to understand well a text [English, 2003].

While reading, understand a word or a sentence is based on the personal lexicon of the reader and its personal knowledge of the language's syntax and grammar. The time for word recognition is highly dependent of the familiarity of the word. The more the reader is familiar with this word, the more he will read and then understand it fast. From the beginning of learning a language to read and understand this language is directly linked with practice of this this language, this corresponds to the social strategy [Kurvers, 1007].

The speed of the lexical access is under the influence of some factors described by [Findlay and Gilchrist, 2003]:

- The word frequency, how much the word is present in every day's language. Obviously, word more present will be recognized quicker
- If a word had been recently met, it will be recognized faster
- The link with another word can be used to understand faster a word. When the word "nurse" follows the word "doctor", the recognition is faster because of the same lexical area
- The length and the 1st landing position of the eye on the word can also impact on recognition speed (cf. previous section)

The daily uses of the language and contact with printed texts arise the vocabulary [Young-Scholten and Strom, 2006]. Few solutions concerning trainings are possible, remind the reader the words he did not know as often as possible under the form of a database is profitable for him, then creating links between words (e.g. creating categories of word depending of their meaning) will also help to train the reader.

2.2.5 Conclusion

The papers and previous studies gave clues concerning the difficulties and the way to go through them. Reading speed varies with the difficulty of the text and the skill of the reader. Based on those studies and papers 3 workable eye movements are present at difficulties:

- Fixation time: if a reader spend too much time on a word, it can be considered as a difficultly
- Saccades length: if the saccades are short, the sentence is more complex for the reader to understand
- Number of refixations: if the reader reread a word several times, it is a harder/unknown word.

The understanding of a text depends of the grammar knowledge and the vocabulary of the reader. Having a daily contact with a words (both written and oral way) make the reader recognizing those faster. Create a list of words containing the initially unknown words is also a great solution for the reader to find back the word he needed a translation. This database should not only contains the translation but also all the related information such a definition, synonyms or an example. However training the user suggests to have enough time and long term testers that will be problematic with a semester project.

Chapter 3

Analysis

First the targeted group and the text that will be read by this one are defined, then the information about the Tobii camera and what it is possible to do with it is analyzed. This chapter contains more information about the tools and the future system will lead to the next one: the design of the application.

3.1 Textual context and Target group

Including the users in the design process allows to get a more optimal and usable design before building the application. A discussion with a Danish teacher has first been made to get some information about teaching then come the textual context and the target group.

3.1.1 Teacher exchange

Presentation

The teacher point of view is important in terms of education and especially about some points concerning how the application works. First of all, the main hypothesis is presented, then the description of the project is made and then the discussion starts. The purpose of this is to have the point of view of a professional in teaching.

- Education and teaching, concerning young adults and the first year of education
- The vocabulary that should be used for this kind of application for beginners.
- The helping and training part of an application

Results of the dialog

The discussion had been made the 21/03/2012 with a Danish teacher at Aalborg Sprogcenter. First, concerning teaching to young adults, the main difference with teenagers is that learning a language for them is not an obligation (mandatory to have L2 at school). The motivation of the learner and its maturity impact a lot on the behavior of the learner. The social strategy (i.e. asking for more information) is developed in the adult learner because it makes sense for him to get more information about the language. Most of the time, adults learners have a goal for acquiring a language like work or live in the subject country, that is not always the case for the teenager. The first year of study is essential to establish a solid base for the learner, the oral part (pronunciation, speaking and oral understanding) should be more important than the written one (read, write and text understanding); this is actually the natural way for a kid to learn a language, first speak then read and write. However the two must be developed in the purpose to help the word recognition.

The text read by the learner should for most of its content the vocabulary of the first year of learning. Domains such as the family or the daily activity are a common base for beginners. However including less-frequency words or words out of these domains would be essential to test an application because of the higher probability of help in those cases, which would also lead to an improvement of the lexicon of the reader.

Concerning the helping part of the application, first giving the translation of a word had been debated and it is nowadays recognized as a natural and good way to help to understand. It depends of the frequency of the vocabulary and the meaning of the word itself. However extra help for a word really depends of its grammar function:

- Nouns and verbs: a definition and a picture
- Adjectives: a definition and a synonym
- Adverbs: example in a sentence, a definition
- Interjection: translation is enough
- Conjunction and preposition: example in a sentence

Concerning the training part, a lexicon that corresponds to what it is in every dictionary, combining the translation, the definition and application of the words in few sentences.

While German is really structured and each word belongs to a particular category, Danish is full of exceptions and specific sentence structures. Thus giving the lexicon information and examples of structures matches Danish.

3.1.2 Target group

The target group is young adult learners of Danish (20 to 40 years old) in the first year of learning Danish; the group must have a level of English high enough to understand the translations and definitions given by the application.

The Tobii eye-tracker, regardless of a subject's age, use of glasses or ethnic background should work with both bright and dark pupil and allows working with a wide section of the population.

3.1.3 Textual concept

The Danish alphabet is composed by 29 letters, it is three more than the English one: 'æ', 'ø' and 'å'. These letters are present in Norwegian alphabet, but are not present in other alphabets. Concerning the Germanic origins of Danish, a lot of words are the same as in English and German. The vocabulary used can be a mix of words common in many languages "gas", specific common Danish words such as "fjernsyn" (television) or specific non-frequent word "udvikling" (development).

After running few tests with the Tobii camera, the size of the font cannot be less than 20-points and the font used will be Lucia Console because each character has the same width and it will be easier for the calculation of the of the words' position. For the vocabulary, as it learners in the first year it will concern the semantic field of the daily activities and the family. However adding less-frequency words is also an important part of the project because of the difficulty they can represent, some words from other semantic fields will be added to the text in the purpose to evaluate the understanding of the readers even with less common words.

3.2 Eye gaze tracking

3.2.1 Tobii camera

The Tobii X120 will be used in this project because of the time limit. However building an eye-tracker from USB camera is possible and this could complete this project, in the purpose to make the tool be used by persons that do not have a Tobii eye-tracker at their disposition.

According to the camera documentation [Tobii, 2010a]The Tobii X120 camera eye-tracking (figure 3.1) is based on the Pupil Center Corneal Reflexion. An infrared light source illuminates the eye causing reflections and then a camera takes an image of the eye showing these reflections. This image is used to identify the present reflections; one is the reflection of the source on the cornea and another one on the pupil. The Tobii camera uses then an algorithm to calculate a vector formed by the angle between pupil and cornea reflections, it is then used to determine the gaze direction. The camera uses infrared light and two image sensors to capture the images then algorithms using image

Figure 3.1: Tobii X120 eye-tracker [Tobii, 2010b]

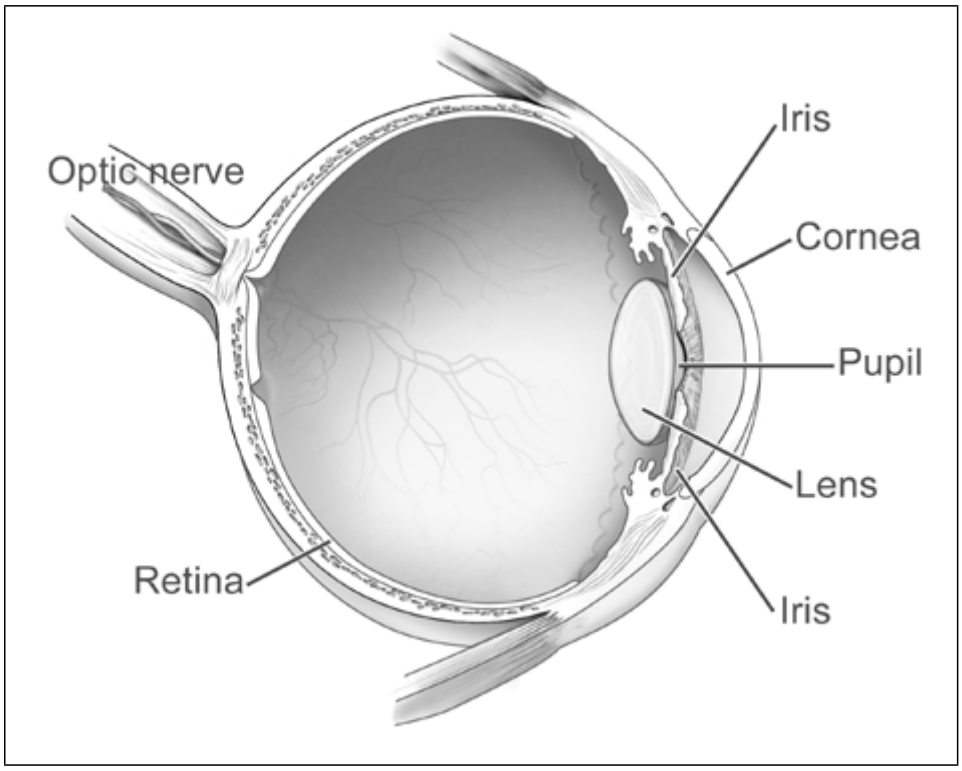


Figure 3.2: Eye Diagram [Institue,]

processing and a 3D model of the eye are used to estimate the point of gaze. The figure 3.2 shows an eye with all the different part of it.

Two different illumination setups are used with PCCR eye-tracking.

- The bright pupil eye-tracking, the light source is placed close to the optical axis of the device, the pupil will appear lighter; this is what causes the "red eye" effect in normal cameras
- The dark pupil eye-tracking, it is exactly the contrary, the light source is placed away from the optical axis of the imaging device. Then the pupil appears darker than the iris.

Age, environmental light and ethnicity factors can affect the bright or dark pupil eye-tracking. The Tobii eye-tracker uses both of these methods to calculate the gaze position.

For each subject, the two methods are tested at the beginning of the calibration and the one with the highest accuracy is chosen for the tests. The Tobii X120 camera is able to register eye-movements with an accuracy of 0.5 degrees [Tobii, 2010b] that corresponds in our configuration approximately to 0.6 cm on the screen; however there it can also be 0.5 degrees of error corresponding to the lighting environment, the noise and the head movements of the reader which corresponds to an accuracy of 1.1 cm on the screen.

The figure 3.3 represents the schema of a tobii system.

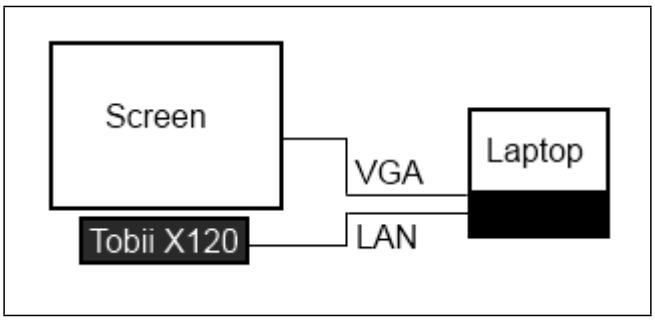


Figure 3.3: Schema of a system using the Tobii camera

3.2.2 Positioning and calibration of the camera

To properly collect the data during a test, it is important to put the subject in the right position according to the maximum angle of the eye-tracker. The figure 3.4 (from [Tobii,]) shows the disposition of a system using the Tobii camera, the maximum angles the eye-tracker can measure and the distance between the subject and the camera

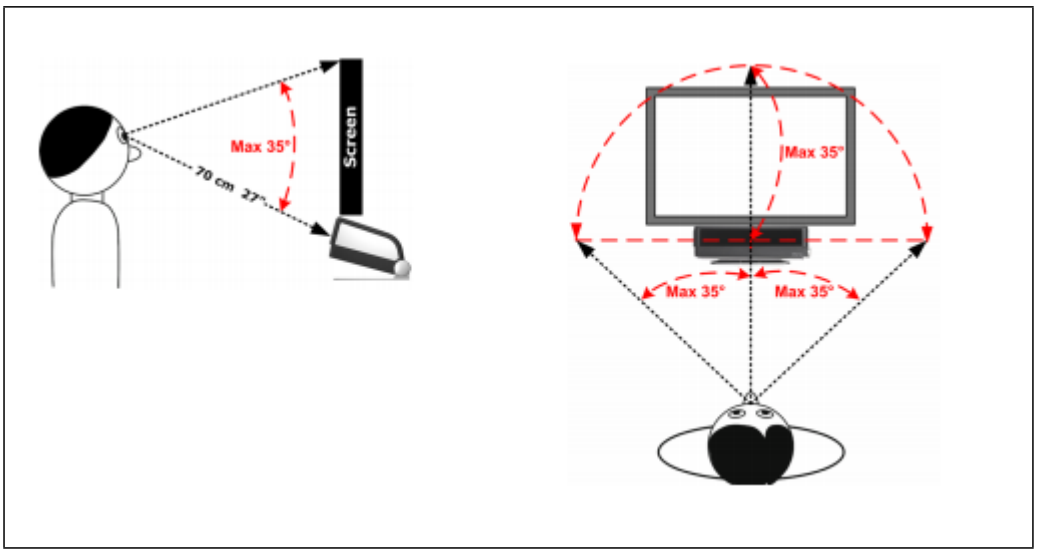


Figure 3.4: Maximum angle of the Tobii eye-tracker [Tobii,]

To finish to set up the camera, the parameters corresponding to its position and the used screen size must be completed, with the configuration tool of the Tobii Studio software. It contains the following values: the eye tracker angle, its distance to screen, the screen angle, its size and the height difference between the two devices

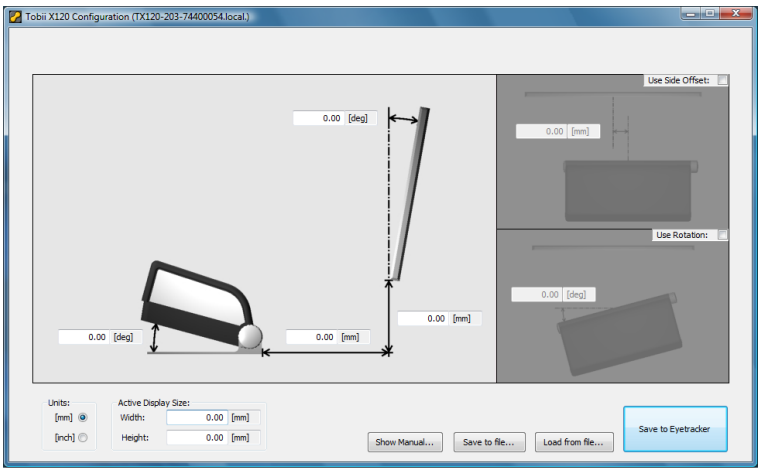


Figure 3.5: Camera configuration file [Tobii,]

Then the eye-tracker has to be calibrated for each subject and this must be done before the subject read any text. The calibration procedure for an eye-tracker according to the developers guides [Tobii, 2011] should follow these steps:

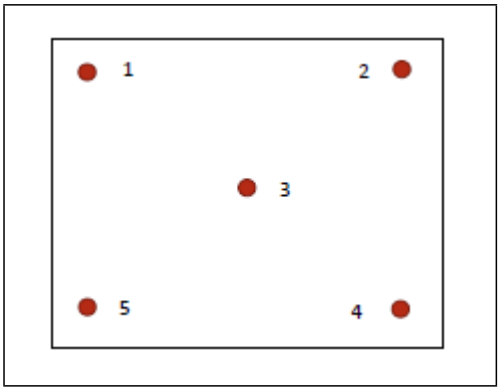


Figure 3.6: 5 Typical calibration points [Tobii, 2011]

1. An animated object should appear on the screen to catch the subject's attention.
2. Move the object to a calibration point and stay 0.5 seconds and minimize the size of the object to make the subject focuses the gaze.
3. The eye-tracker collects the data of the corresponding point.
4. Enlarge the object again
5. Repeat the previous steps for different calibration points.

The usual number of calibration points is 2, 5, or 9, "more points can be used but the result will not increase significantly for more than 9 points". The developers manual

recommends 5 points which gives good results shown in the figure 3.6. During this procedure, a 3D model of the eye is created from taken images; this one is used to calculate the gaze data. This model includes information concerning light refraction and reflection of different parts of the eyes. The calibration process is reiterated until the calibration is accepted by the camera.

The SDK used: "Tobii Analytics Software Development Kit" provides a set of function to retrieve and analyze data from the Tobii eye-trackers. The SDK is free and contains the functions for is compatible with C#, Python, C++ and objective-C but also sample of codes to start the development.

The Tobii eye-tracker outputs used in the project are mainly the 2D coordinates (x and y) of the gaze of each eye. They are under the form of two 64 bits Float numbers varying from 0 to 1 (with 0 the first point on the left and 1 the point on the right). Another data used is about the validity of every detection, this validity code is given for each right and left eye. This code corresponds to how certain the eye tracker is about a data given for an eye. The table 3.1 summarizes the different possibilities given in [Tobii, 2011]

Table 3.1: Data validity codes [Tobii, 2011]

		Right Validity Code				
		0	1	2	3	4
Left Validity Code	0	Found two eyes				Found one eye. Most probable the left eye.
	1				Found one eye. Probably the left eye.	
	2			Found one eye. The tracker cannot with any certainty determine which eye it is.		
	3		Found one eye. Probably the right eye.			
	4	Found one eye. Most probable the right eye.				No eyes found

The validity codes are used to filter out the incorrect data "it is recommended that all samples with validity code 2 or higher are removed or ignored.".

3.2.3 Midas touch problem

Using eye gaze as the "cursor" for an application make the interaction direct between the human and the machine; however due to the fact that the eyes constantly look at objects, determine if the user is just looking or wants an action to occur becomes the key for the application using eye-tracking. The possibility to enable/disable the eye-tracking should be mandatory in every application using eye-gaze. [Glenstrup and Engell, 1995]

However this lead to another problem concerning this "switch", it should be quick enough to do not lose the advantages of the eye-tracking, but not too much to not make this one enable constantly by mistake. Eye-tracking systems has to not bother the user at all time but also make it possible to be used when this one requires it.

This project concerns a tool to assist and train the user to understand Danish text, the Midas touch problem is thus limited because the user always wants the system to provide him the translation in case of difficulty. However is such a system should be used in other cases (e.g. reading a Danish web page) this would become bothering for the user in certain cases.

3.3 Helps and design principles

3.3.1 Reading help

As defined in the analysis, helping reading can be done through different aspects.

- Translation: Kontakt = Switch
- Definition Kontakt:
  - a device for making and breaking an electrical connection (lys kontakt)
  - communication or cooperation between people or organizations
  - the action of communicating or meeting
- Synonyms of Kontakt:
  - afbryder
- Example of sentence: jeg installerer den nye kontakt = I install the new switch
- A picture of: kontakt (figure 3.7)

The help can be divided in two parts; the translation is always the starting point of the help and then come complementary information.

If more help is required then it will depends of the kind of word (i.e. verb, noun, adjective etc.). As an example, a definition of an article is less relevant than the one of a noun. The table 3.2 shows the different helps for each kind of words.

Concerning other possible help, dividing compounded words can be done during the lecture of a word, as a secondary help too.

Figure 3.7: Picture for the word "switch"

Table 3.2: Helps depending of the word

Kind of word	Translation	Definition	Example	Synonym	Picture
Adjectives	X	X		X	
Adverbs	X	X	X		
Articles	X	X	X		
Conjunction	X		X		
Interjection	X				
Nouns	X	X			X
Preposition	X		X		
Verbs	X	X			X

3.3.2 Personal lexicon

The training part of the tool consists of allowing the user to consult and look. Each time a reader has a problem to read one word, the application provides help but also put the word in a personal lexicon. Instead of just having the translation of a sentence, the word has added information such as an example or a definition.

Consulting the lexicon can also let the user evaluate his progress in reading. For example, if the reader always has a difficulty on a particular word, this one can appears in another color. Also if the word is all the time recognized, this one can be removed from the lexicon or appearing with a mark. This way the user can focus on some words than the others.

The possibility to order the words by name, number of time missing, when they have been meet let the user more flexibility about his lexicon and work on it.

The application will be a tool to assist reading; however it is also a tool to train. In this case depending of the reader's wish to understand the word, it could be interesting to have the pronunciation of the word in the dictionary. The user will be able to understand the word and to pronounce it. This could be done both in an oral way (by saying the word) and a written one using phonological alphabet.

3.3.3 Suitable design principles

The design is an essential point of building an application because it will be the visual part of this one. The [Jacquemin, 2011] course contains some design keys for application. Those design principles are here impacting two aspects of the design:

- Perception: segmentation of an image, attention of the reader and colors
- Response execution: guide the user to make the right decision, in a shorter time

For the application, three bases of pages are defined: the menu, the text and the personal lexicon.

The Menu page

The menu has in every application must remain simple. Most of the time a menu always contains a "quit" "help/about" and "launch the application" buttons. In this application, the two features (reading a text with help and the personal lexicon) could each represent a button, the menu is completed by the quit button, the help/about one and the title of the application.

Considering the fact that the two main features are reading a text and the lexicon, those buttons should be bigger to guide the user while using the application. The figure 3.8 shows a possible first menu.

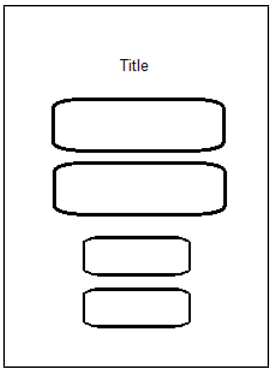


Figure 3.8: Draft of the menu

The Text page

This is the page with the more interaction between the user and the application. The information given by the Tobii camera will make this page evolving while on it. At the contrary of the menu presented previously, the number of elements on this can change, as an example the number of words in the text or the help(s) present(s) or not.

The figure 3.9 represents the process from the global picture to the exact information. First step, the global processing divide the picture in shapes and areas of interest. Then each region is analyzed and for each area of interest, which one looks more useful than the others. Then the attention can be separated in two parts.

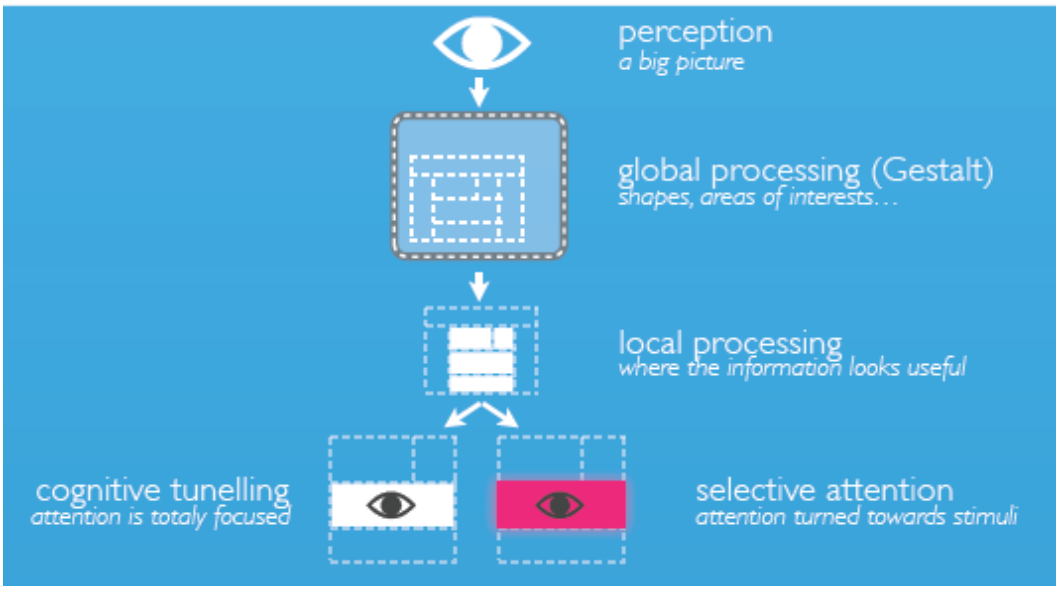
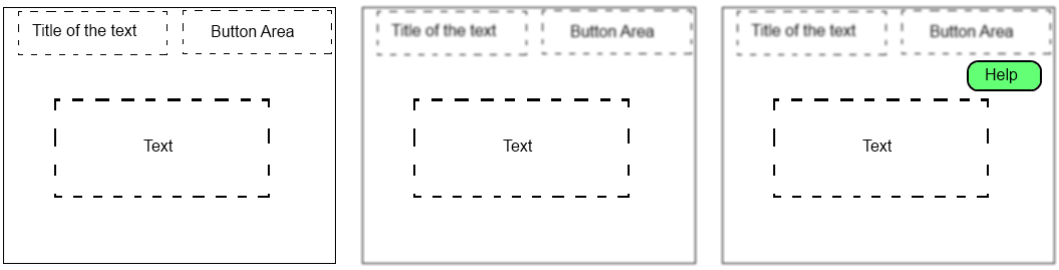


Figure 3.9: From global picture to details [Jacquemin, 2011]

- The selective attention, the attention is turned toward stimuli (e.g. an alarm)
- The cognitive tunneling, a focused attention (e.g. speedometer of a car)

In this case, the cognitive tunneling is the text, the reader keeps his attention on it at all time and the selective attention is the help provided in case of difficulty. The figures 3.10 are first schemas of the segmentation of a page and the representation of the attentions, starting from the global picture until the help.



(a) Global processing **(b)** Cognitive tunneling **(c)** Selective attention

Figure 3.10: Schema of the segmentation of a page

However, the selective attention makes a new object appears and according to the Fitt's law that is:

$$T = a + b \cdot \log_2 \left(\frac{D}{W} + 1 \right)$$

- T is the time to hit the target

- a and b are empirical constants
- D is the distance between the starting point and the center of the target
- W is the width of the target measured along the axis of the motion.

The further the object is from the target, the higher the response time will be and it is also true for the size of an object, the smaller it is the longer will be the response time. The help should appear close to the corresponding word. At the same time to differentiate this help from the text, as shown in the figure 3.10 (c), another color should be used to take the attention of the reader. In this case, using a light color (figure 3.11) should not annoy the reader while reading.

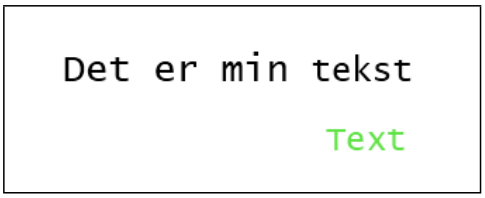


Figure 3.11: Example of a translation

Personal lexicon

The principle of the personal lexicon is to provide the user with a database containing the words where a translation has been provided. Using it after or before reading a text or to help him improve his vocabulary, this personal lexicon must be simple to use and contain all the added information related to a word. The user of the application cannot use it in parallel of reading a text but between two readings as the words to remember.

The basic design will be a list of words that makes appear the corresponding translation/definition/image when you click on it. With the possibility to search for a word in English or Danish and sort the list by: alphabetic order, the number of missing occurences, the last time encountered etc. The figure 3.12 is a first draft of the personal lexicon

A color code can be apply on those words (figure 3.13), corresponding to the number of missing time or the last time there was a fault. Going from the green to the red for the proportion of missing times and if a word is not missed anymore it can be put in a light gray which corresponds to the fact that the word is not missing anymore.

3.4 Conclusion

This analysis part contains all the elements to build a first prototype that will be used to lead interviews and finally give the final prototype before the implementation. The analysis started with some points given by a Danish teacher, information concerning the motivation of the students but also some key points concerning the teaching method

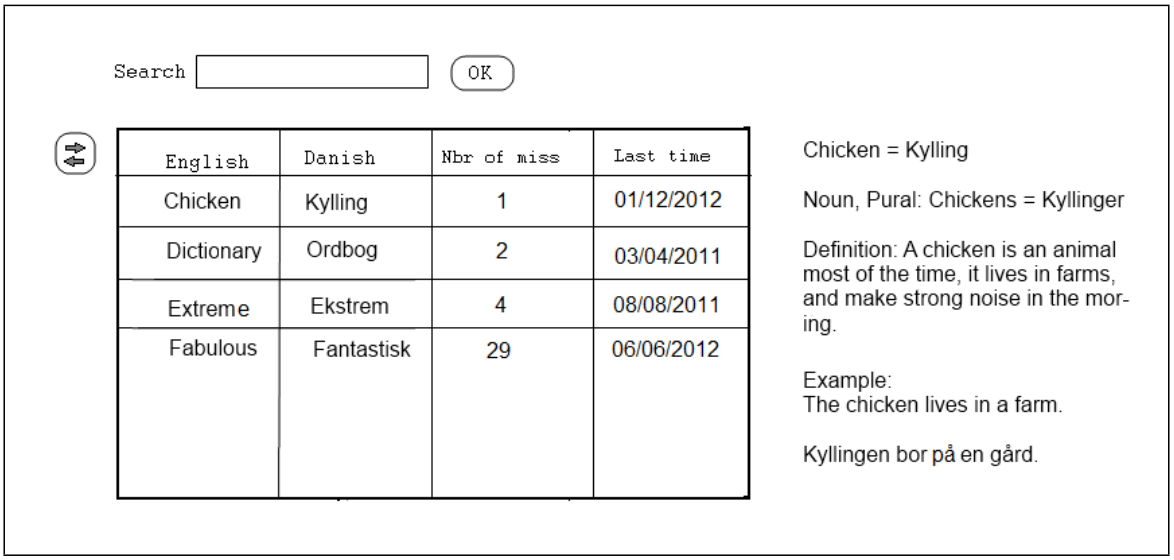


Figure 3.12: Draft of the personal lexicon

English	Danish	Nbr of miss	Last time
Chicken	Kylling	1	01/12/2012
Dictionary	Ordbog	2	03/04/2011
Extreme	Ekstrem	4	08/08/2011
Fabulous	Fantastisk	29	06/06/2012

Figure 3.13: Lexicon with colors

have been discussed. The text used in the application will concern daily activities which matches for a first year of study level.

The study of the Tobii camera has provided information concerning the possibilities and limits of such a system. The calibration and the configuration files are essential to get an accurate system, however some problem emerges from the eye-tracking such as the Midas touch one concerning the usability.

Finally the different way of helping and training the user and suitable design for the application conclude this chapter. These preliminary designs will be used to build to first paper prototype in the Design part. All the different elements present in this chapter are bases for building a system using the Tobii camera.

Chapter 4

Design

Based on the elements of the analysis chapter, the design will define the final prototype used for the implementation. The application follows an iterative design process, it starts from a first design which evolves by consideration of the remarks and the answers of interviews. The user centric design based on an iterative design process is represented in the figure 4.1. Compare to the iterative development process, the iterative design process does not include the implementation and testing in the loop. These last steps will be done after getting the final design.

4.1 First design and Interview

4.1.1 First prototype

The first design is based on the elements resulting from the analysis part. The first paper prototype contains four pages, the menu, the text and two pages for the personal lexicon.

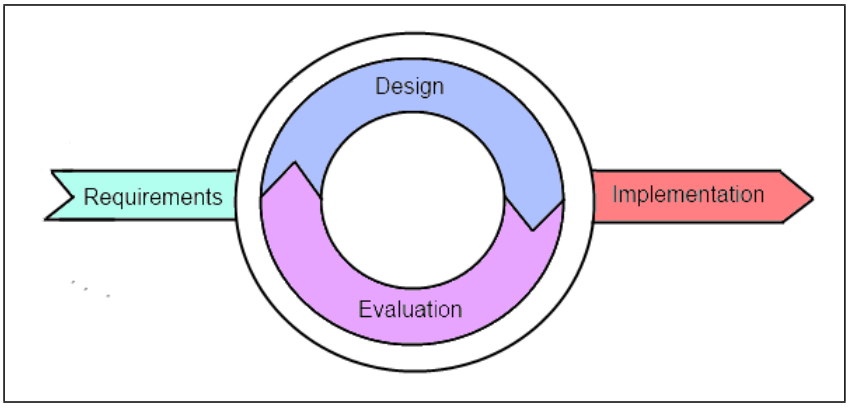


Figure 4.1: Iterative Design process [Jacquemin, 2011]

Menu

It represents directly what appears when the application is started. It does not appear in full screen and contains 5 buttons: Read a text, Personal lexicon, About, Help and Quit. The figure 4.2 illustrates the menu of the first paper prototype.

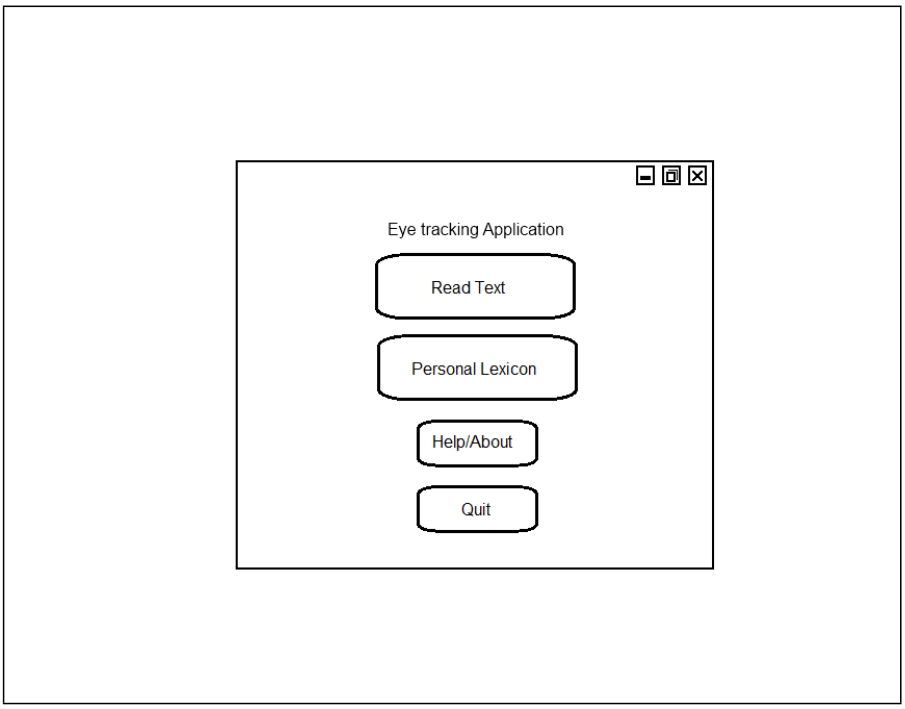


Figure 4.2: Prototype 1 - Menu

Text

Concerning the text page, the purpose of the first interview is to determine the position of the help module and have the overall view of the structure of the application. Thus this first text page represents only the text and different elements around it such as the title and some buttons. No colors are used at that stage, the interview focus on where to place the help while reading for this page. The figure 4.3 illustrates the text page of the first paper prototype.

Personal lexicon

There are two pages to represent the personal lexicon. The dictionary is represented with a first page containing the list of words, without the color code; the second page contains the lexicon with the detail of one word. This first prototype focuses on the structure of the application. The figures 4.4 and 4.5 illustrate the personal lexicon of the first paper prototype.

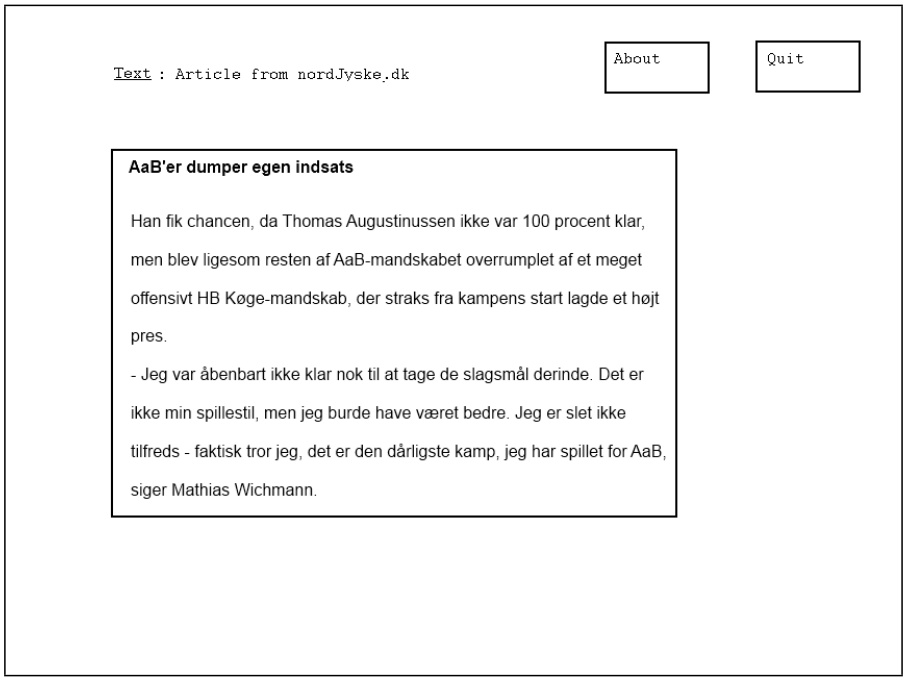


Figure 4.3: Prototype 1 - Text page

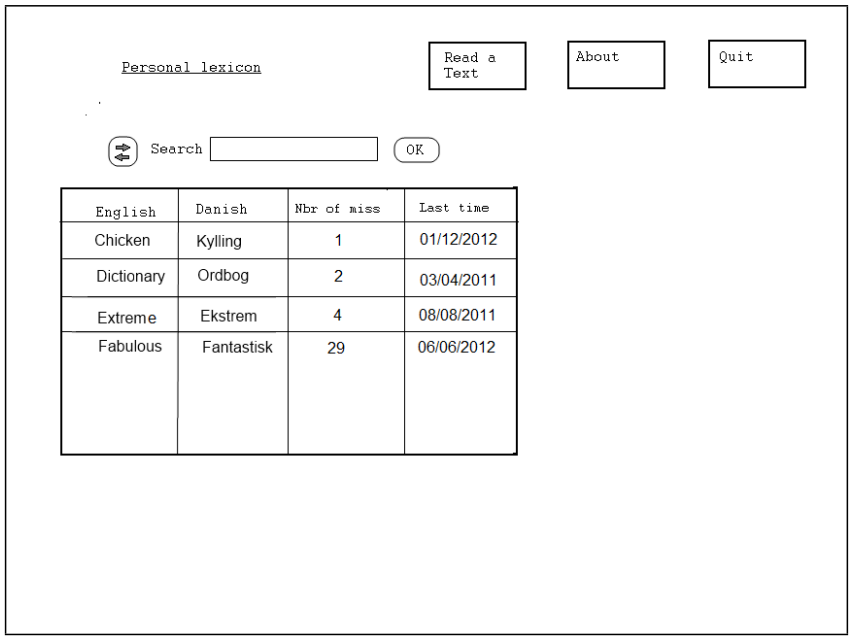


Figure 4.4: Prototype 1 - Personal lexicon

4.1.2 First interview

The purpose of the first interview is to present first a low fidelity prototype to people and get a general feedback about the organization of the first design. It allows also to receive different suggestions for the placement of some elements. As an example, as where to place a translation when help is required. This interview is shown as a simulation of the

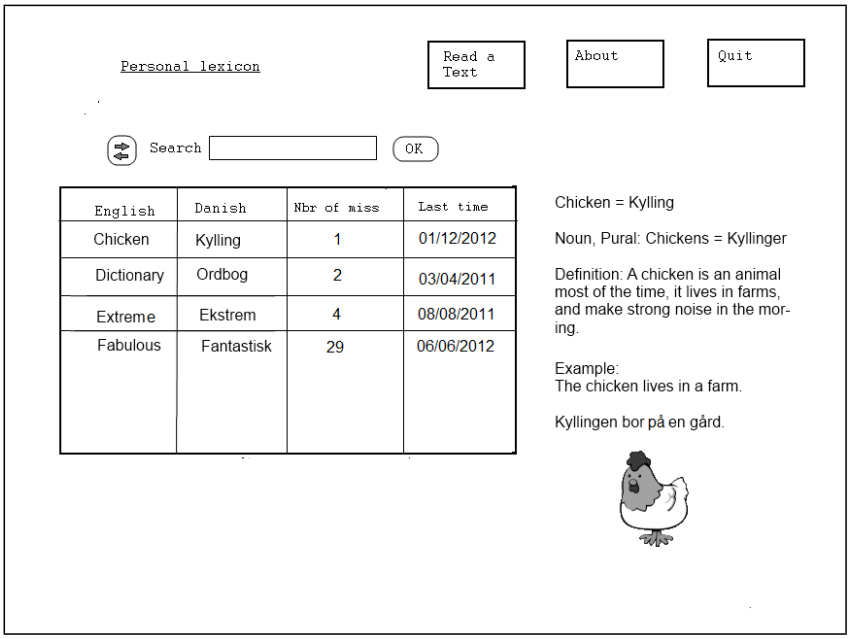


Figure 4.5: Prototype 1 - Personal lexicon with details

application's use for the interviewee. He has to complete few tasks and give his feedback. For example he has to try to access to the right page indicate if it is easy or not using it. After the simulation, some questions are asked to the interviewee concerning some features of the application.

During the interview, the presentation of the prototype is divided in two parts: first the text page and then the personal lexicon. Test of the text page: Starting from the menu (figure 4.2), the interviewee has to access the text (figure 4.3). He then starts to read and simulates a difficulty on a word, at this point a sticker with the translation of the "difficult" word is given to him. This first help is the one that will be shown: he has to place it on the text where he thinks he would prefer the help. Right after that, a second sticker is given to him, this one represents the advanced help (definition, synonym, example ...) and should appear if the reader wants more information. The interviewee is able to give the shape and the size he wants for the information and locate it where he thinks it is the best place to look when searching for information.

Test of the lexicon: the reader has to start again from the menu. This time he has to access the personal lexicon. The word he had a problem with in the first lecture appears in the "personal lexicon page" (figure 4.4 and 4.5). There the interviewee has to open the details of the word and gives his feedbacks about the information provided and the general aspect of the pages.

Finally some questions concerning the additional help provided and the possibility to give the pronunciation of a word conclude the interview. These additional questions has two different purposes, the first one is to get the preferences of the user on the existing features and the other one concern the possibility to add a sound part in the application.

- What added information would you give to the user: definition, synonym, example, images, other?
- How long should stay the translation of a word and the advanced help on the screen?
- Does the pronunciation of a word help you to understand it, even in quiet reading?

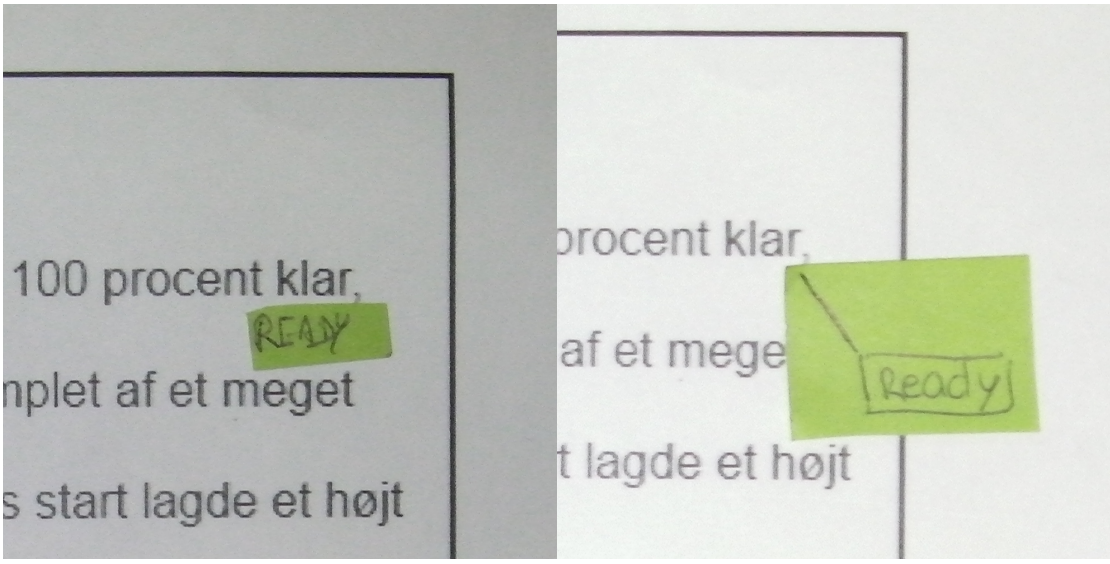
4.2 Iterative process

4.2.1 Process and Results of the first interviews

The first interviews took place between Mars 21 and 23 2012 five students between the bachelor and master degree. All of them were learning Danish. The interviews took an average time of 21 minutes and had been lead in a room of the University, the material used had been the first paper based prototype printed and stickers.

Concerning the menu, all the participants reached the right page. It had been mentioned that the menu is "classic and simple", it will thus not change in the next prototypes. The name "personal lexicon" puzzled 1 of the participant who did not understand it.

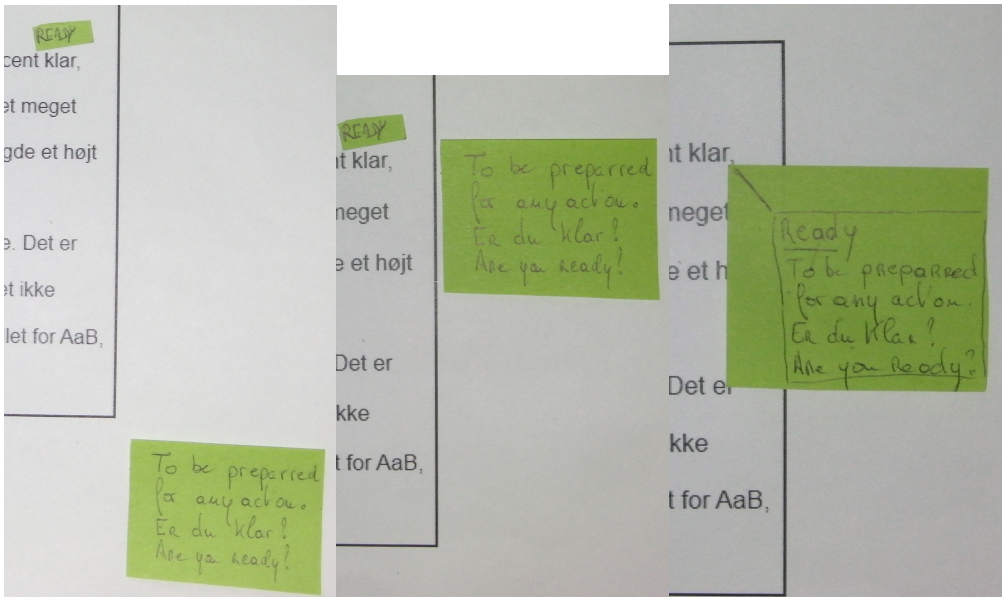
The Text page has been described as "simple but with the right information", however "the text should be more centered on the page". For the location of the translation, four of the interviewees proposed to place it in the interline right under or above the corresponding word, however one of them proposed to place it in a "bubble" linked to the word. Figures 4.6 show the different possibilities proposed by the interviewees for the translation.



(a) Translation under the word **(b)** Translation in a bubble

Figure 4.6: Pictures of the positions of the translation

Concerning the advanced help, one more time four of the interviewees proposed a panel in the space around the text. Two of them proposed it static in a corner and the two others proposed to move it depending of the word's vertical position (figure 4.7.a and 4.7.b). The last proposition was to complete the previous bubble (see figure 4.6.b). The information shown in the advanced help will be some sample of the ones contained in the personal lexicon.



(a) Static Advanced help in a corner **(b)** Moving Advanced help **(c)** Advanced help in a bubble

Figure 4.7: Pictures of the positions for the advanced help

The personal lexicon looked clear and organized for the 5 interviewees. For three of them the "Danish column should be placed first" and the number of "nbr of miss" columns should use some colors to recognize the words more difficult to remember for the user. 4 of the interviewees wanted a more realistic personal lexicon, containing words of the text from the text page.
Finally concerning the questions:

- Question 1: two considered the image too childish but for the three others it is a really good help for learning. All other possible helps were relevant, depending on the kind of word (e.g. verb, nouns etc.)
- Question 2: four of the interviewees considered that the translation of a word should stay until the end of the text. Concerning the advanced help, 3 of them said it should stay until the next information was asked and the others proposed to hide it this after aw while.
- Question 3: three of the five persons though that the pronunciation is a real plus for learning and helping the people to remember the words, however, it should be added in the dictionary but not in the advanced helps.

4.2.2 Changes applied to the prototype and interview v2.0

Second prototype

The second prototype consisted in an evolution of the first prototype considering the feedbacks of the first interviews. The menu remained the same because it was simple and clear enough in the first prototype; however the three other pages changed. Two pages of the prototype are present in this section, all the other remaining ones are given in the appendix.

The two kinds of help provided (translation and advanced) have two attributes each:

- The position: where to place the help on the page
- Its aspect: the color and shape of the help.

According to the answers of the first interviews, the table 4.1 contains the possibility of the text page for the second paper prototype. This is why the text page of the second version of the prototype is split in two; the figure 4.8 and 4.9 contains all the possibilities. The text of the page has been changed to be more realistic, this one corresponds to the level of Danish learner in the first year of study.

Table 4.1: Possibilities for the helps of the text pages

Factor	Prototype v2(a)	Prototype v2(b)
Position Translation	Upper the text	Under the text
Form Translation	Colored background	Colored text
Position Advanced help	Bottom right corner	Position depending of the word
Form Advanced help	Colored background	Colored text

For the personal lexicon, all the elements are kept; a color code is added to it as shown in the figure 3.12. The words present in the personal lexicon are now some words taken from the text.

Interview v2.0

This new interview will improve one more time the design. All the participants are different from the people interviewed for the first version. The purpose of this interview is to define which design is the best for the text page and if the color of the personal lexicon is understood by the user.

As for the first interview, the interviewee has to start from the menu and first open a text. While reading the text, a difficulty is simulated for the reader and then the two possible designs for the translation defined in the table 4.1 are shown. The reader gives his opinion about both the pages and simulates that he wants more information; then

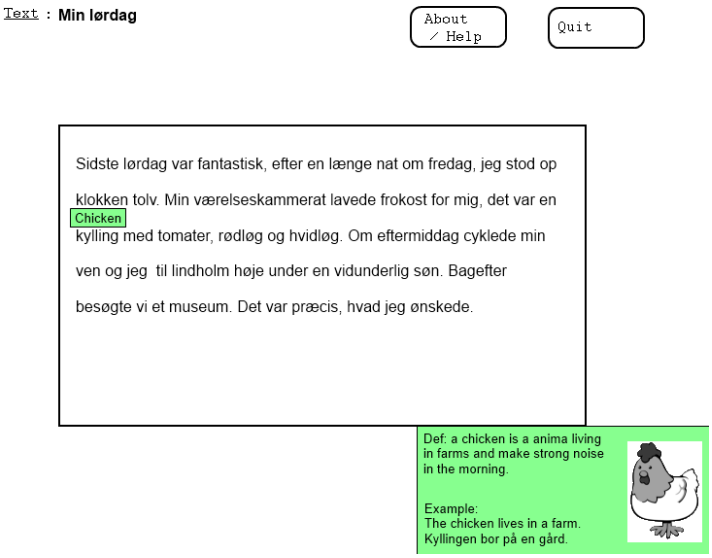


Figure 4.8: Prototype 2(a) - Text with translation and advanced help

Figure 4.9: Prototype 2(b) - Text with translation and advanced help

the two other possibilities for the advanced help is shown to the user (figure 4.8 and 4.9). One more time the interviewee gives his opinion about the two possible designs.

After the text pages, the personal lexicon is shown. The questions asked to the interviewee concern only the colors associated with the words (see figure A.7 and A.8 in the appendix). Here is the list of the questions with the expected answers (table4.2).

Table 4.2: Questions about the personal lexicon of Interview v2.0

Question	Expected answer
Why the word "Fabulous" has a red square ?	Because the word has been missed a lot of time recently
Why the word "Red Onion" has a green square ?	Because the word has been missed a lot of time recently
Why the word "Exactly" has a yellow square ?	Because the word has been missed a lot of time recently
Why all the row of the word "Dictionary" is in gray ?	Because the word has not been missed for a long time

4.2.3 Results of the interviews for the second version

Two interviews have been conducted on March 27, 2012. These two interviews have led to a version 2.1 of the paper prototype. The appendix A. corresponds to the version 2.1 of the prototype. The changes concern the shapes and names of the buttons in the lexicon and the text page. Moreover in the lexicon, the way to sort the list of words has also changed, the "double arrows" button was replaced by arrows directly in the column. After these two interviews three others had been done on March 29th with the prototype v2.1.

Concerning the help, four of the interviewees told that the translation in color under the word seems better (figure 4.8) and "It does not annoy to read the translation this way compare to the frame" (figure 4.9). Concerning the advanced help, all the participants considered that it was easier to read it when it was in a frame and four of them said that the added information should be close to the word instead of a static place, here it was the bottom right corner.

For the five interviewees, the color code was understood and one of them told "the lexicon should not be changed from now". However two of them made a remark about the fact that the details of a word should be organized like a table or just more structured visually.

4.3 Use cases

According to the analysis and the interviews, the list of the features possible for the Final prototype is the following:

- Open a text
- Provide a first help (translation) where the reader has difficulty
- Provide more information when the user requests it.

- Access to the complete list of the personal lexicon
- Display the details of a word in the personal lexicon
- Open the help/about page

This section presents the use cases that correspond to the previous list of tasks. The abbreviation P.L. represents: "Personal Lexicon". The use cases start when the user is on the menu. The use cases and the final prototype are the theoretical models used for the implementation part.

4.3.1 Use case Diagram

The figure 4.10 is the class diagram and the table from 4.3 to 4.8 are the 6 corresponding use cases.

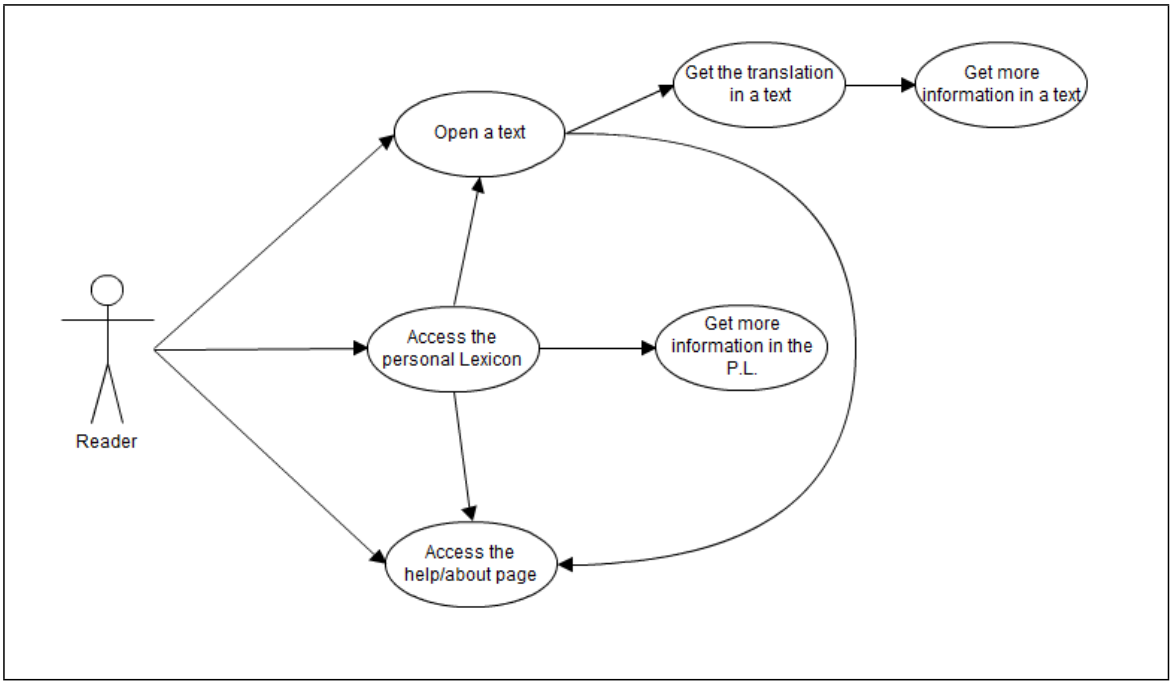


Figure 4.10: Use case Diagram

4.3.2 Use case Tables

Table 4.3: Use Case 1: Open a text

Use case 1	Open a Text
Actor	Reader
Description	The reader wants to open a new text
Basic course of action	1 The reader clicks on the Read Text button of the menu 2 The application opens the calibration page 3 The reader follows the calibration procedure 4 The calibration is accepted 5 The Text opens
Alternate courses of action	1A The reader clicks on any other button of the menu 2A The reader restarts the application 3A The reader re-attempts action 1 4B The calibration is not accepted 5B The reader re-attempts a calibration until this one is accepted

Table 4.4: Use Case 2: Get a translation in a text

Use case 2	Get a translation in a text
Actor	Reader
Description	The reader has a difficulty to read a word, the application provide him the translation of this one
Basic course of action	1 The reader has a difficulty to read a word 2 The application detects the difficulty 3 The application writes the translation of the corresponding word right under it
Alternate courses of action	2A The application does not detects the difficulty 3A The camera has to be re-calibrated

Table 4.5: Use Case 3: Get more information about a word in a text

Use Case 3:	Get more information about a word in a text
Actor	Reader
Description	The reader had a difficulty to read a word and wants more information about this one
Basic course of action	1 The reader keeps staring at the word 2 The application detects the gaze of the reader on the word 3 The application provides more information about the word on it side
Alternate courses of action	2A The application does not detects the difficulty 3A The camera has to be re-calibrated

Table 4.6: Use Case 4: Access the personal lexicon

Use Case 4:	Access the personal lexicon
Actor	Reader
Description	The reader wants to access his personal lexicon
Basic course of action	1 The reader clicks on the Personal Lexicon button of the menu 2 The application opens the personal Lexicon
Alternate courses of action	1A The reader clicks on any other button of the menu 2A The reader restarts the application 3A The reader re-attempts action 1

Table 4.7: Use Case 5: Get more information in the personal lexicon

Use Case 5:	Get more information in the personal lexicon
Actor	Reader
Description	The reader wants to get more information about a specific word in his personal lexicon
Basic course of action	1 The reader clicks on the name of the word 2 The application displays all the details of the corresponding word
Alternate courses of action	1A The reader wants to type the word in the search bar 2A The word does not exist 3A The reader re-attempts action 1 1B The reader wants to type the word in the search bar 2B The word is found in the list 3B The application displays all the details of the corresponding word

Table 4.8: Use Case 6: Access the help/about page

Use Case 6:	Access the help/about page
Actor	Reader
Description	The reader wants to access the help/about page
Basic course of action	1 The reader clicks on the help/about button of the menu 2 The application opens the help/about page
Alternate courses of action	1A The reader clicks on any other button of the menu 2A The reader restarts the application 3A The reader re-attempts action 1

4.4 Final paper based prototype

The final paper based prototype has been built considering the remarks of the previous interviews applied to the different versions of the prototypes. Two images of the final paper prototype are provided below; all the others are given in appendix B. at the end of this report.

The figure 4.11 and 4.12 are respectively the text page with helps and the personal lexicon. The only page not determined now is the calibration one, it will depend of the different possibilities offered by the Tobii SDK.

Figure 4.11: Final prototype - Text with helps

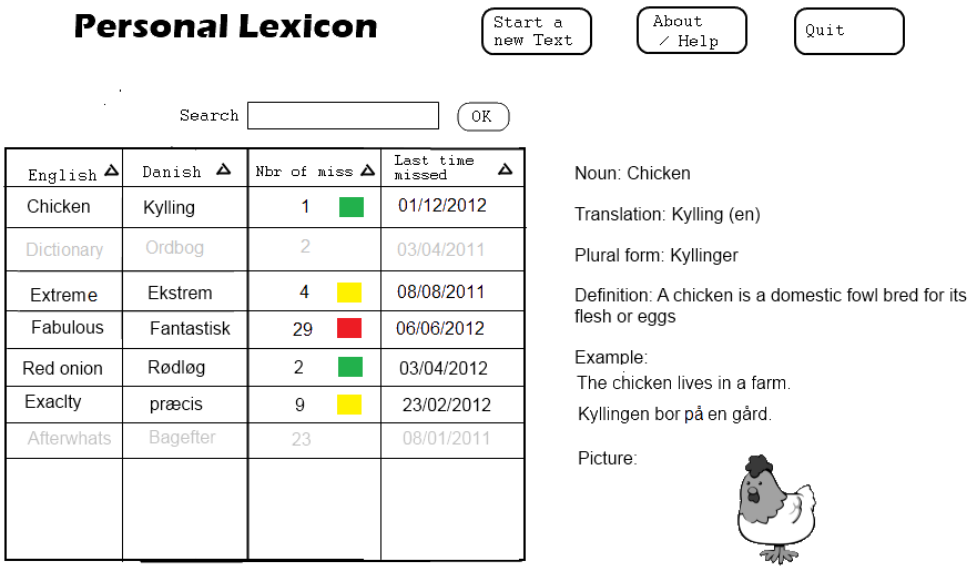


Figure 4.12: Final Prototype - personal lexicon

Chapter 5

Implementation

This part contains the information about the structure of the application built starting the prototypes and also the modifications applied for technical or time reasons. First, the global structure of the application is presented through some diagrams, then the calibration of the camera is presented. After comes the part concerning the words and finally the detection with some pictures of the final prototype are shown. More detailed explanations are present in the appendix part C.

This application has been developed in C# because it is one of the supported languages by the Tobii camera. It also uses the library Tobii SDK v3.0.2, its structure is presented in the figure 5.1 taken from [Tobii, 2011]:

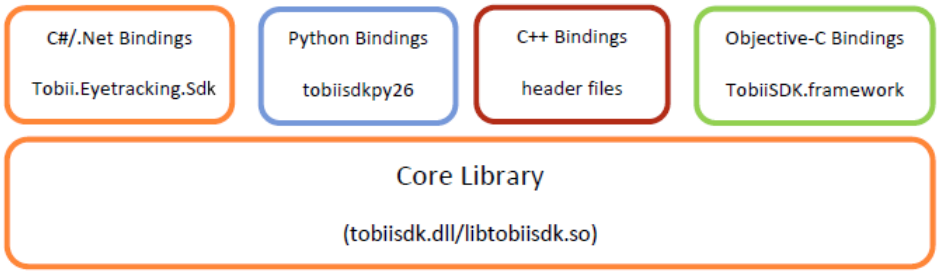


Figure 5.1: Tobii SDK components [Tobii, 2011]

Presented with the library, some examples concerning the first calibration of the camera and gaze data acquisition were given. Some of the classes present in the open source examples were re-used in the application with the purpose to create the connection between the camera and a computer.

5.1 Structure of the system

5.1.1 Deployment diagram

The system is composed of three elements: a laptop, a second screen and the Tobii camera. The computer node contains itself three objects: the GUI of the application,

the words database that will allow using the dictionary and the personal lexicon, and finally and the eye tracker data analyzer. Its purpose is to receive and analyze the data provided by the Tobii eye-tracker node. The second screen node contains the display of the application because it is aligned with the Tobii eye-tracker. Finally the Tobii eye-tracker node only contains the eye-gaze data that are provided to the computer node.

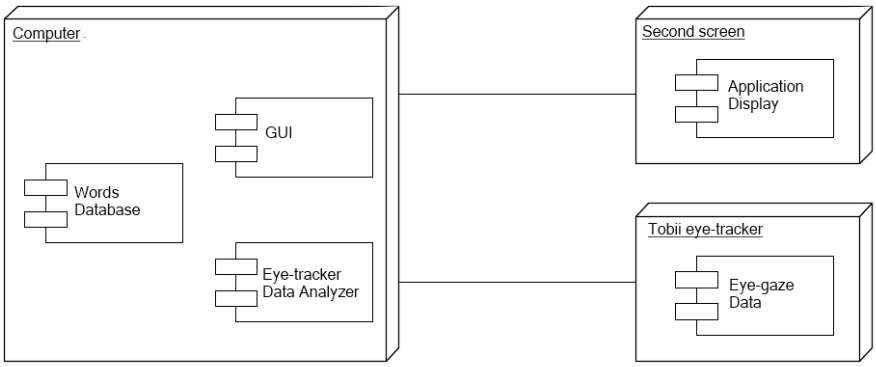


Figure 5.2: Deployment Diagram of the system

5.1.2 Class diagram

The software architecture is structured around a main class "Master" containing an instance of all the graphical components of the application which are all the classes in the packages "Form". The ones inherit from the class "Form". The class "Master" hides and shows the different forms in the purpose to create the GUI of the application.

The "Database" package manage the words, its classes are used to get the corresponding information such as the translation, definition, example etc.

Finally the EyeTracking package is the one which receives and analyzes the data sent by the eye tracker.

The figure 5.3 is the simplified class diagram with the packages; the detailed class diagram is presented in the appendix.

5.2 Calibration in the application

The calibration of the camera is essential to get a good accuracy. This is done through calibration points and can be visualized with a calibration plot.

5.2.1 Calibration points

The points are given one after the other to the camera. During the implementation, 5 points did not give enough accurate results, so 9 points are used instead. The 9 points "cover" the text centered on the screen as shown on the figure 5.4

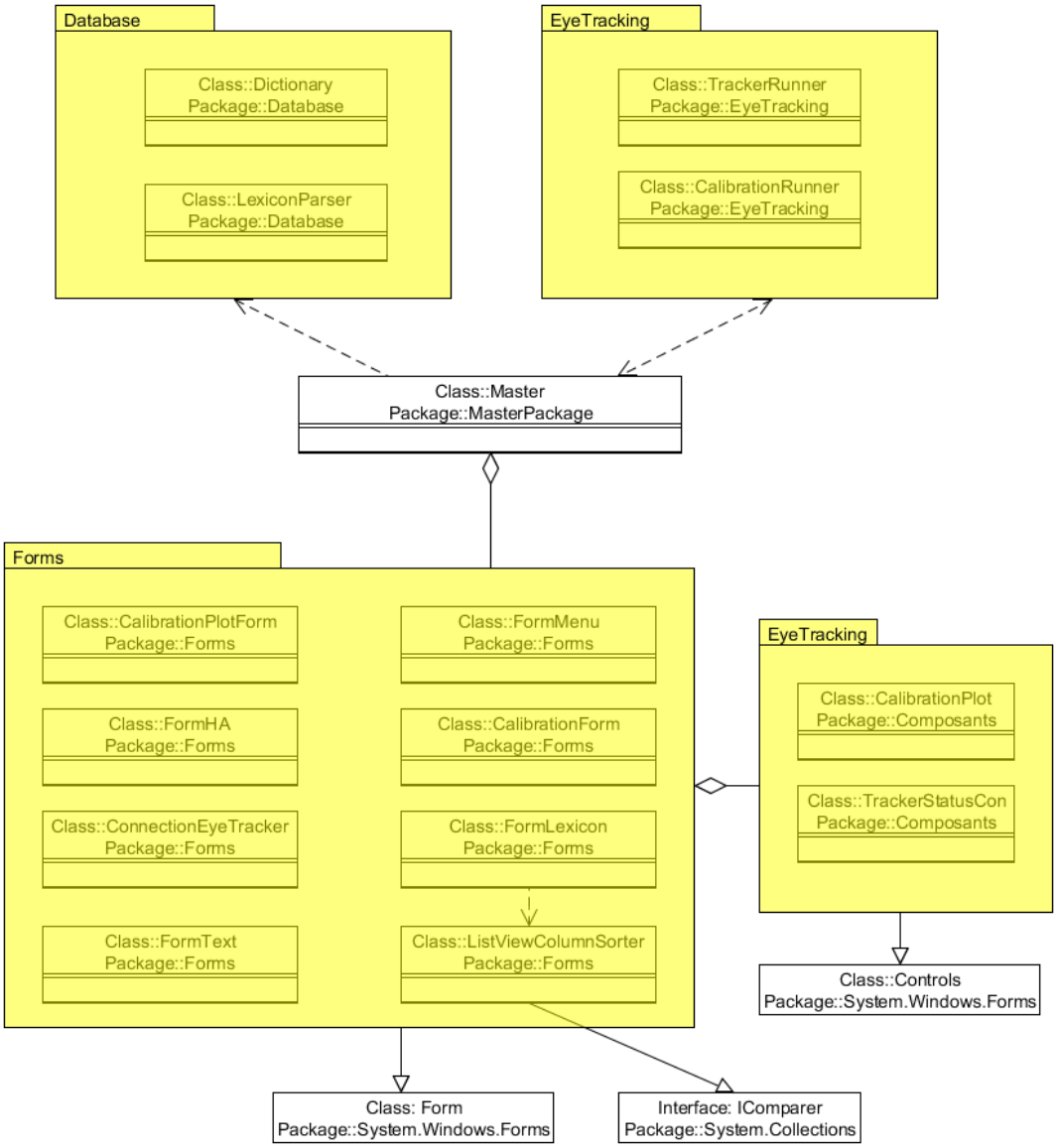


Figure 5.3: Class Diagram of the application

```
private void CreatePointList()
{
    calibrationPoints = new Queue<Point2D>();
    calibrationPoints.Enqueue(new Point2D(0.175, 0.2));
    calibrationPoints.Enqueue(new Point2D(0.5, 0.5));
    calibrationPoints.Enqueue(new Point2D(0.825, 0.2));
    calibrationPoints.Enqueue(new Point2D(0.825, 0.8));
    calibrationPoints.Enqueue(new Point2D(0.175, 0.8));
    calibrationPoints.Enqueue(new Point2D(0.35, 0.5));
    calibrationPoints.Enqueue(new Point2D(0.65, 0.5));
    calibrationPoints.Enqueue(new Point2D(0.5, 0.65));
    calibrationPoints.Enqueue(new Point2D(0.5, 0.35));
}
```

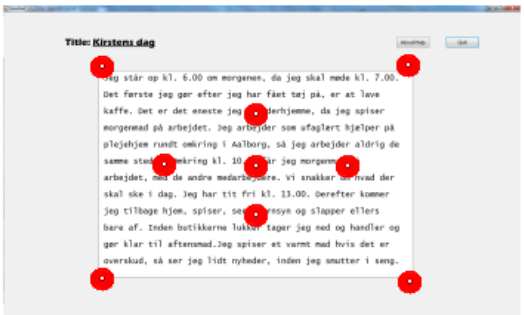


Figure 5.4: Calibration points

5.2.2 Calibration plots

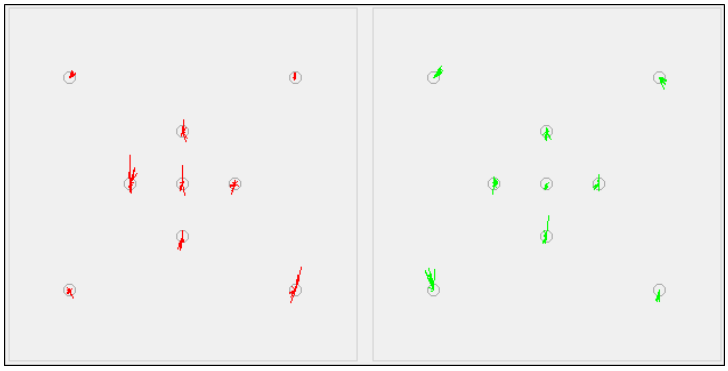
After running the calibration, the errors from each eye can be visualized in a calibration plot; two plots are required to visualize the calibration errors of each eye. The controls get the calibration data and extract from it the 9 points of the calibration (targets watched by the user see figure 5.4). Then it paints a line between the calibration points center and every point of the data with a validity code equals to 1(cf table 3.1); the points with a validity code higher or equal to 2 are ignored. The figures 5.5 represent two calibrations, the figure 5.5.a represents a good one and 5.5.b a bad one. The longer the lines are the worse the calibration is around this point and will affect the detection of the eye gaze.

5.3 Dictionary and words

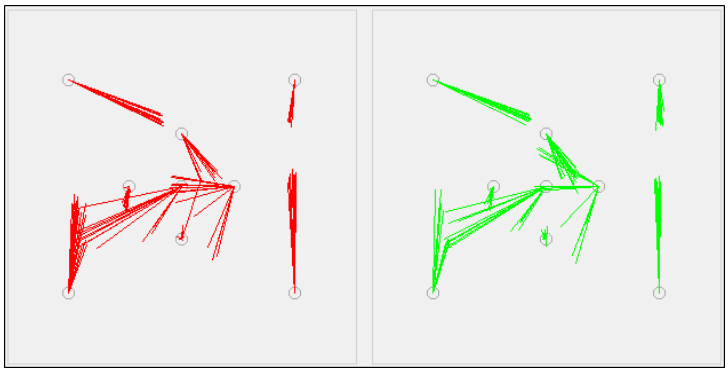
The words and their added information are stored in a text file. Each line represents one word with its corresponding information separated by the ';' or the '-' character, the figure 5.6 shows a part of the file.

The encoding for reading and write in the file must be specified by default which correspond to the actual ANSI code actual of the operating system, otherwise characters such as å,æ and ø will not be read correctly.

For the moment the dictionary only contains the words of the text, however in further development a dictionary API could be used.



(a) Good calibration: accepted



(b) Bad calibration: must be done again

Figure 5.5: Two examples of calibration

Part from Dico.txt

```
7   when;da;at or during a time; when I bike-da jeg cykler;conjunction;because-then;
8   have to;skal;be obliged to, must, represents an obligation;I have to go-jeg skal
9   meet;møde;happen to come into the presence; I meet a friend - jeg møder in ven;ve
10  the;det;design something; The biggest monkey i saw - Det største abe jeg så;artic
11  first;første;comming before all the others;The first exercise - Det første øvelse
```

Figure 5.6: Part of the dictionary text document

5.4 Detection

5.4.1 Loading the text

The "Master" class has a double list of the class "wordShape". When the text page is loading, this list is initialized; there is one instance of "wordShape" for each word in the text. Each one contains:

- int location: corresponds to the X coordinate of the word (left side) in pixel
- int width: corresponds to the width of the word in pixel
- String word: the word itself

- Boolean shown: a boolean initialized at false, but when a word is shown this one turn to true

The "Master" with this information knows the place of each word on a line but also it X coordinate in pixel (location and width variables), the figure 5.7 represents the double list where the blue nodes are the lines and the orange ones are the words themselces. Those data will be used to determine where a help must be provided when a difficulty has been detected.

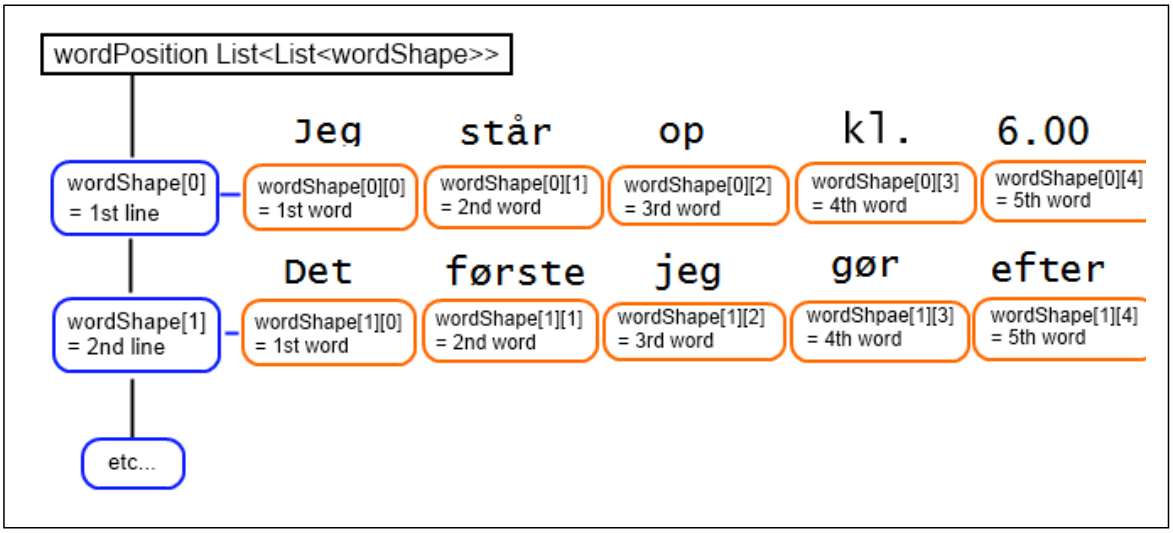


Figure 5.7: Schema of the double list

5.4.2 Difficulty and help

The application is based on the analysis of fixations, 60 times a second, the values X and Y (floats between 0 and 1) for each eye is received from the eye-tracker. If the validation code (see Analysis part), this one is converted in the corresponding pixel position and saved in the history.

The average of the last sixties values is calculated and every value is compared to it. Then each value of the history is compared to this average number, if the distance between a point in the history and the average is too big a counter is decremented, when the counter is under a certain value, the history is cleaned and the reader is not considered as being stuck on a word, in the contrary an help is provided.

To determine which word the reader is stuck on, as the space between two lines does not change (27 px), the average Y coordinate allows to determine which line the reader is reading. Then average X coordinate is compared to the position of each word of the corresponding line (or wordShape[line] see figure 5.7) present in the "Master" class. The corresponding translation is painted on the text page right under the word (as the coordinates are known).

5.5 High fidelity prototype

All the functions present in the application used for the interaction between the pages are based on click events and hide/show functions.

The translation appears when the reader needs help and if he requires more information, the application provides him the corresponding ones in a panel outside the text, the figure 5.8, 5.9 and 5.10 shows few screen shots of the high fidelity prototype while being used. The use cases of the Design chapter have been used as a base for the application, the only difference with it concern the use case nř1: "Open a text" where the text is defined by default and another text cannot be chosen in the purpose to fit the corresponding dictionary (because of the time limit, an API has not been integrated to replace the actual dictionary).

The other element that has been removed is the search bar in the personal lexicon; once again it is because of the time limit. Otherwise no other tasks have been removed or modified and the high fidelity prototype is almost similar to the final paper prototype.

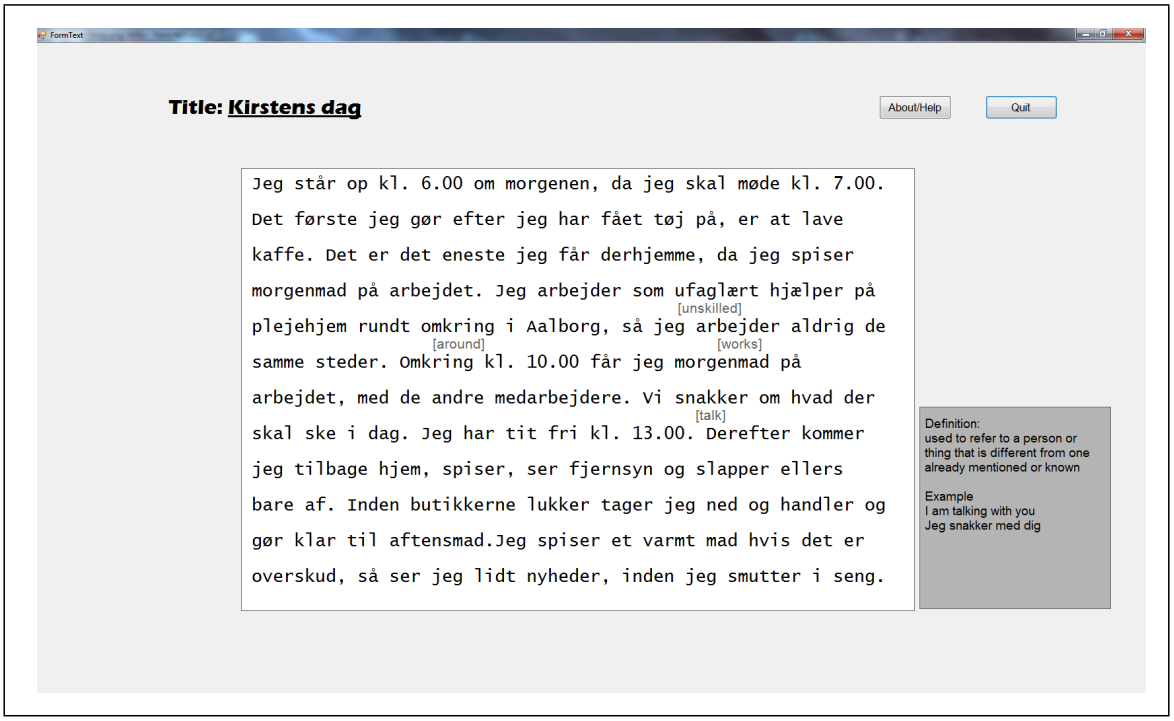


Figure 5.8: High Fidelity Prototype - Text page with few helps provided

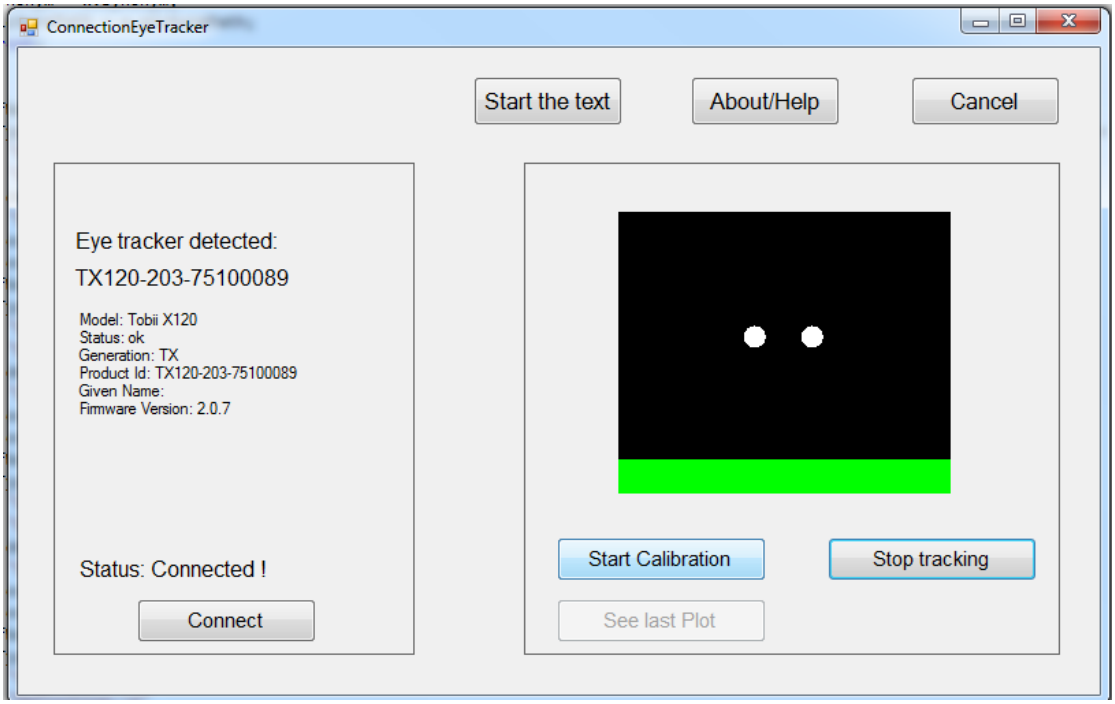


Figure 5.9: High Fidelity Prototype - Calibration window

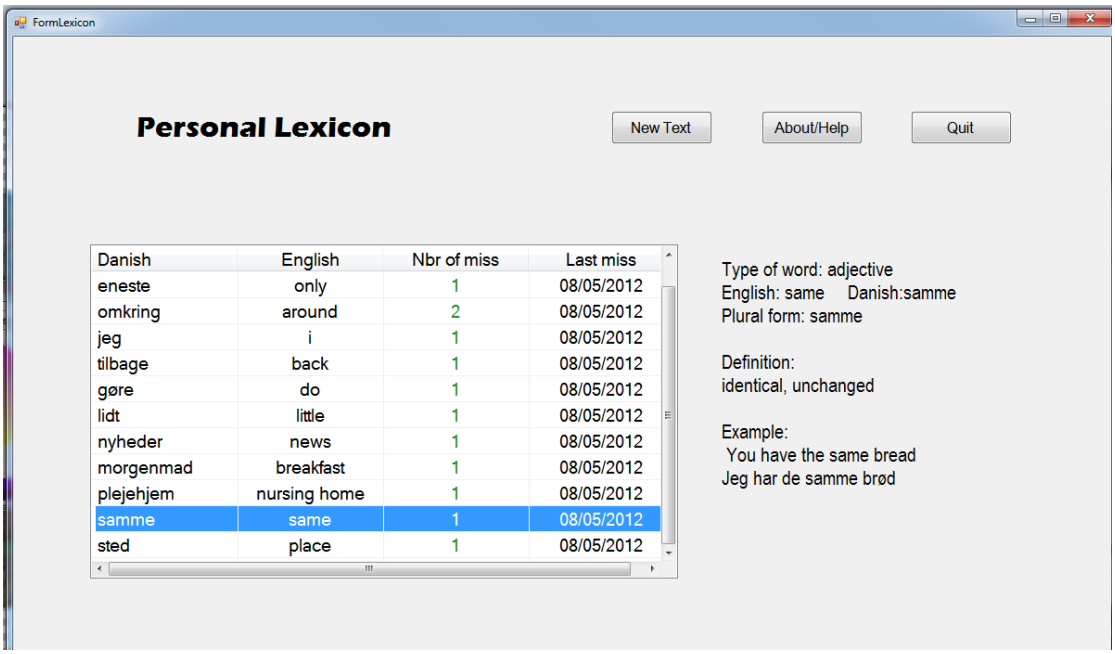


Figure 5.10: High Fidelity Prototype - Personal lexicon page

Chapter 6

System Test and Improvements

The test is here to give a general view of the built high fidelity prototype and provide elements to answer to the research question: "Can a tool using eye-tracking assist and train adult learners of Danish to understand Danish texts?".

In order to validate the system and application before the usability tests, a validation test has been made in the purpose to test every possible action of the application. Every translation, added information of the words and connections between the buttons have been checked.

6.1 Between subjects experimental design

For this user test, the purpose is to compare the results between a treatment group using the eye-tracker and a control group without it. Both of them will have the text on a screen with the same font (Lucia Console) at the same size (20 points). The time to read and understand the text (present in the appendix D.1) is the same for both of the groups: 4 minutes, then they have to answer to Multiple Choice Questions (MCQ) without any time limit, all the MCQ are present in the appendix D.2 The results of the MCQ are calculated with the number of errors and all the questions have the same value. The results of two groups will be compared with the purpose to get some clues to answer to the research question.

6.1.1 Treatment group

The test environment for the treatment group is composed by a laptop, a 22" inches screen and the Tobii camera X120. The laptop is connected to the screen by a VGA cable and the eye tracker by an Ethernet cable (cf figure 3.3). The curtains of the room are closed with the purpose to limit the reflection of the exterior light on the testers' eyes. After running some tests, the system worked for some people wearing glasses but not all of them. This is maybe due to different kind of glasses and their treatments (e.g. anti-reflective coat).

The usability test for the treatment group follows this plan:

- The tester starts the program and reads the help
- The tester starts the calibration, re-does until the calibration is accepted by the observer
- The tester starts a text and has 4 minutes to read and understand it
- The tester answers the MCQ
- The tester looks at the personal lexicon
- The tester answers to few questions concerning the application and the system.

Using the eye tracker system, this group can give its opinion about the application, its usability, its accuracy and the other features (personal lexicon). That is why other questions are added to the process, those ones concern the application but not the text. Here is the list of the questions asked for the treatment group.

- Would you use this system? Why? advantages and disadvantages
- What do you think about the speed of the help?
- Accuracy of the system: how many times the translation of another word appears (false positive) and how many times did he try to get a word and it was never shown (false negative)?
- Any suggestion to the system?

6.1.2 Control group

The test environment for this group is composed by a 15,4 inches laptop which is easier to carry, as the eye-tracker system is not required the tests can be done in any other place. The size of the screen is different from the one of the treatment group; however the screen is big enough to display the text at the same format as for the 22 inches screen. The laptop runs a version of the application that does not include the eye-tracking. The elements of the text page are the same with the same disposition, the impact on the reading is thus limited, the figure 6.1 shows the two screens at the same proportion.

As the control group is able to use a dictionary, a web browser is open with the Danish-English dictionary page: http://da.bab.la/ordbog/dansk-engelsk/ and the English definition dictionary page: http://www.wordreference.com/ in parallel of the application. The participant can look at the dictionaries as much as he wants. The dictionaries are the ones used during the test, because they are used to build the "dictionary" of the application. The first one provided the translation, examples and synonyms and the second one the definitions. The two groups will have the same information for the words.

The usability test for the control group follows this plan:

- The tester starts the program
- The tester starts a text and has 4 minutes to read and understand it
- The tester answers to the MCQ

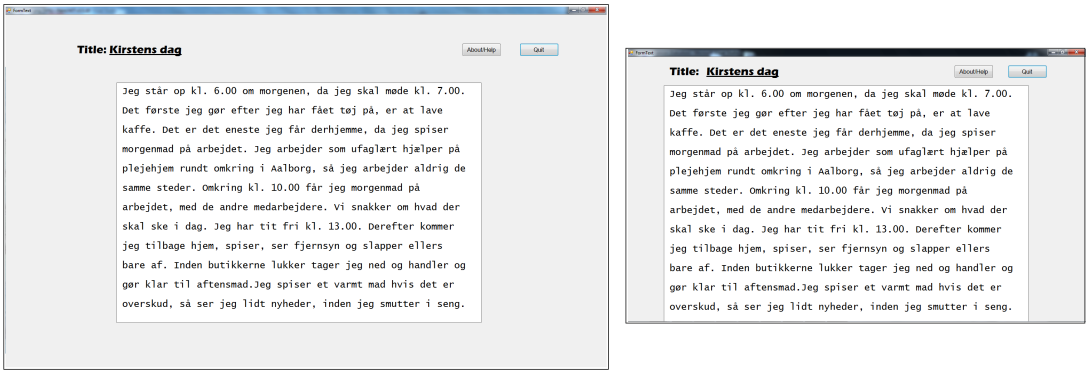


Figure 6.1: Program on the 22" (left) and 15,4" (right)

6.1.3 Possible source of errors

As the test follows a "between subjects experimental design" only one factor should change: in this case the use of the eye-tracking or not. However due to some simplification, some slight changes exist between the two groups. This is the possible list of errors that may impact the results of the tests.

- The screen size is not the same
- The room where take place the test for the participant of the control group can change
- The control group has a web browser open with the dictionaries and have to switch the window to pass from the dictionaries to the application

6.2 Test process and results

6.2.1 Process

The tests took place the first weeks of May. In total 15 participants, 8 in the control group and 7 in the group using the camera. No one of the testers is native Danish speakers (the nationalities are given in the appendix D.3) and all of them are following the courses in Aalborg sprogcenter. The level of Danish there is split in 6 modules, all the participants are in the second one, so they are considered as having the same level.

An observer is always with the participant during the reading process and the questions. For the treatment group, he also led the discussion after using the application. The control group test took an average of 10 minutes and the group using the application took an average of 21 minutes. This difference between the two times is due to the fact that the treatment group has the calibration part to achieve but also the discussion concerning the system.

6.2.2 Results

Answers to the Multiple Choice Questions

The figure 6.2 shows the result plot box for the treatment and the control groups. The median is respectively 7 and 6. Concerning the point at 3 in the treatment group, this value is considered as an outlier because it is out of the scope.

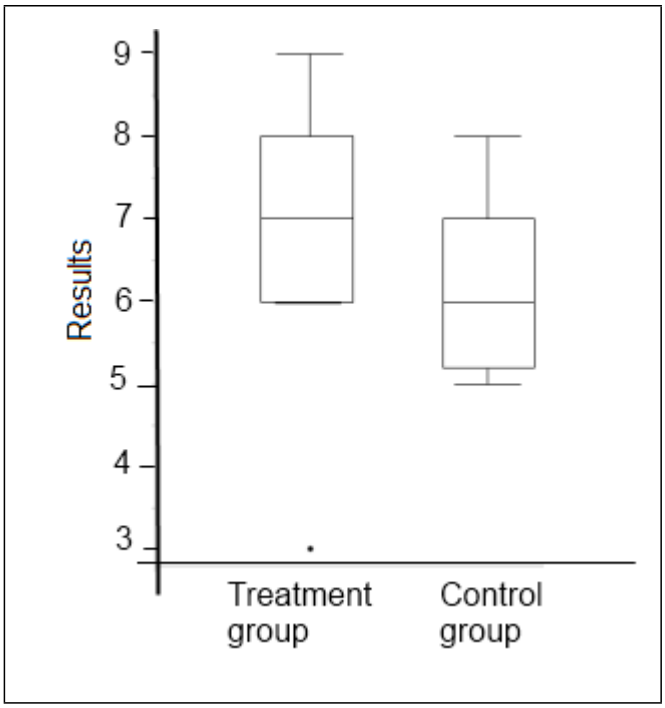


Figure 6.2: Box plot of the results

Answers concerning the systems

Concerning the discussion for the group using the application: all the participants talked about the concept. Five of them qualified it "cool". Four of them talked about the process to automatize or make the translation faster this way than "copy/paste in Google Translate". Two of them said that it should be this way the translations on computer in the future should be handled this way. However 4 of them mentioned the fact that it can be inconvenient to get too much translations when such a program is running and recalls the Midas touch problem (subsection 3.2.3). For the 7 participants it was "quite natural" to look the words and get the translation. Only 3 persons used the complementary information functionality.

Concerning the accuracy, there is an average of 0.6 false negative: 4 in total and an average of 1.7 false positive: 12 in total ((in every cases it concerned a word in a line above or under). For five of the testers the speed of the help was good; however for the two others it should be slightly less fast.

The participants did remarks about the personal lexicon, however as if it would be more a training functionality, it should be used in a longer period of time to have relevant feedbacks.

Three of the persons have seen through this application a tool for teaching. They said: "it should be used in teaching centers or by teachers to provide statistics concerning the level of a class". This functionality would require to choose the text.

6.3 Discussion concerning the results

Answers to the MCQ

Due to the limited number of participants to the test, the results concerning the MCQ cannot be generalized to a larger population. However this preliminary test can give an indication about the trend for a larger population. When comparing the results, it appears that the treatment group got slight better results than the control group.

A fact observed during the test is that looking at another page (online dictionary) makes the reader loose his focus on the text. Tehn it takes time for him to find back where he was in the reading. With the eye tracker the reader keeps his attention on the text during the 4 minutes duration of the reading. The participants of the treatment group had the translation staying until the end of the reading. It was not the case of the control group who had sometimes to search again for a translation.

Discussion about the system

The participants talked about the fact that such an application could become annoying for the user. It is not a problem for an application like this one where the main purpose is to read one text. But if the eye-tracking is used in other conditions (e.g. on a web page) this could become a real problem to have words popping every time a person is looking at a word. The possibility to deactivate the eye-tracking should be considered for every system using an eye-tracker. Another point is how fast the help occurs. It should be possible to choose different levels of speed, at least 3 like: slow - regular - fast. The user could have the possibility to choose it at the beginning, according to his language level.

The horizontal accuracy showed no error, while the vertical one had shown some. For all the false positive errors the translation appears on the line above or under and especially in the corners of the text. This is due to the fact that the height of the screen is smaller than the width; the camera is thus less precise on the Y dimension. The calibration impacts the accuracy; however the mistakes rate was not significantly and decreased when the calibration was better. Some solutions can be tried to improve the system:

- make the font bigger or improve the space between two lines of text: the font is already big enough and making the font bigger also means less text on the same page.
- The linear nature of the reading could be a used, a reader always starts at the first line and keeps a reading flow. Following the reading line by line and word after word is a possibility (assuming that at the end of a line, the next word would be on the following one), however it could happen to re-read the same line

Chapter 7

Conclusion and perspective

7.1 Project achievement

Starting to learn a new language is a task impacted by the motivation of the learner. Providing new tools for the learners is a challenge possible with the new technologies. The eye-tracking allows solbing some problems detected through the eye movements (e.g. saccades and fixations), it is possible to determine when a person meets difficulties to read or understand a word while reading. The project's aim is to answer to the question: "Can a tool using eye-tracking assist and train adult learners of Danish to understand Danish texts?"

The eye-tracking has some limits concerning its usability and accuracy. The eyes are moving almost constantly and fix different objects. The analysis of these movements may determine the difference between the conscious ones and the unintentional ones. The accuracy is also a factor that will determine the right place of the gaze in the purpose to launch the right functionality.

Documentation about the fact of reading and learning a language brings information for the future prototypes. After this step, an user centric design study gives the aspect of the application with the purpose to build it. The high fidelity prototype is close to the design of the final paper prototype and allows making some user tests to answer to the hypothesis.

The results given by the system tests cannot be generalized to the entire population due to the number of participants. However they allow to have a preliminary answer to the question concerning the assist part of the application. The results obtained during the tests shown a tool using eye-tracking can potentially assist adult learners of Danish to understand Danish texts. However, for the training part of the application the question cannot be answered, such a tool should be tested on a longer period.

7.2 Limitation and further development

In a first time, getting better and more complete results concerning the actual application should be done before following the development of the application. Some adaptation

of the system tests must be done to continue to work on the system, the following list corresponds to some points that must be changed.

- Increase significantly the number of participants of the two groups, using people with different level of Danish and have results generalized to an entire population of learners
- The conditions of the tests for the two groups of the "between subjects experimental design" should be exactly the same (expect for using the eye-tracking system) to limit the number of errors
- Making the test on two or three different texts with more questions would provide a more complete set of results
- Make the user test the personal lexicon on a period (e.g. a least daily during a month), to see how it impacts they learning and having a real feedback on the training part of the project..

In a second time improving the actual application by adding some features make and it more usable, is the second point of the further development of the system. The following list is the points that should be modified or updated in the actual application:

- Adapt the application to work with any text chosen by the user. Instead of opening the text directly in the text page, let the possibility to the reader to choose it will allow the application to be used by a larger scope of people.
- The previous point requires then the use of a complete dictionary. Based on existing API providing the translations and information of the words, the readers will have the possibility to get the information of every word in a text. For example, Google translate provides an API for using the translation [Google,], or others such as my gengo [myGengo Inc.,] (these API are not free).
- Adding the pronunciation, as a language as to be spoken and understood training the user to say the words out loud is an important part of learning. An adult learner will be more interested in getting the pronunciation because of its motivation (cf the discussion with the teacher). A button which makes the user be able to get the pronunciation should be added to the application.

After improving the user tests and completing the application, the system could has further developments. First of all if a complete dictionary can be used, all the languages could be used, and the application could be extended. Today it is limited to a Danish text with an English help. This will lead to some problems concerning the languages with a vertical or right to left writing. Secondary, as proposed by the some of the treatment group participants, this kind of application could be used to provide statistics for a teacher about the level of his class learning a language.

7.3 Personal achievement

This master Thesis completes the VGIS program, this was a time of self-management and it shows the limits of an entire project alone from A to Z. The supervisors provided another vision of the project which is essential especially for a one person project. The major problem to making a project alone is to become blind to the own mistakes of the writer and sometime does not see alternative visions of a problem.

Concerning the knowledge part, exploring the human vision and the language learning open a new area, where it was crucial to read papers to understand those mechanisms corresponding more to the psychology domains. Having the possibility to use new tool such as the eye-tracker is also an opportunity to increase the range of the devices and plenty other ideas for further projects/applications.

Chapter 8

glossary

ANSI code

ANSI escape sequences are characters embedded in the text used to control formatting, color, and other output options on video text terminals. [http://en.wikipedia.org/wiki/ANSI_escape_c

API

An application programming interface (API) is a specification intended to be used as an interface by software components to communicate with each other. An API may include specifications for routines, data structures, object classes, and variables. [http://en.wikipedia.org/wiki/API]

Eye-tracking

Technology that monitors eye movements as a means of detecting abnormalities or of studying how people interact with text or online documents: a company that uses eye tracking to evaluate visual products [The Oxford Pocket Dictionary of Current English. 2009]

Language proficiency

also called linguistic proficiency is the ability of an individual to speak or perform in an acquired language. [http://en.wikipedia.org/wiki/Language_proficiency]

Lexicon

In linguistics, the description of a language is split into two parts, the grammar consisting of rules describing correct sentence formation and the lexicon listing words and phrases that can be used in the sentences. The lexicon (or wordstock) of a language is its vocabulary. [http://en.wikipedia.org/wiki/Lexicon]

Outlier

It is an observation that is numerically distant from the rest of the data. There is no rigid mathematical definition of what constitutes an outlier; determining whether or not an observation is an outlier is ultimately a subjective exercise.

Median

It is the numerical value separating the higher half of a sample, a population, or a probability distribution, from the lower half. [http://en.wikipedia.org/wiki/Median]

Semantic process

Process which lead to the understanding of a sentence by getting the meanings of the words [Findlay and Gilchrist, 2003]

zeroconf

Zero Configuration Networking is a set of techniques that automatically creates a usable Internet Protocol (IP) network without manual operator intervention or special configuration servers. [http://fr.wikipedia.org/wiki/Zeroconf]

Bibliography

[Alderson, ding] Alderson, C. (Assessing Reading). *2000*. Cambridge: Cambridge University Press.

[Butorff, 1985] Butorff, M. (1985). *Haitian Creole literacy evaluation study: Final Report*. Washington, DC : Center for Applied Linguistics.

[Crystal, 2003] Crystal, D. (2003). *English as a Global Language*. Cambridge University press, 2nd edition.

[English, 2003] English, S. (2003). Strategy intervention and reading proficiency in the modern tertiary efl classroom. *On-line source: http://www.nyu.edu/classes/keefer/waoe/englishs.html*.

[Eurobarometer, 2006] Eurobarometer (2006). Europeans and their languages. Technical report, European commission.

[Findlay and Gilchrist, 2003] Findlay, J. M. and Gilchrist, I. D. (2003). *Active Vision*. Oxford University press.

[Frenck-Mestre, 2005] Frenck-Mestre, C. (2005). Eye-movement recording as a tool for studying syntactic processing in a second language: a review of methodologies and experimental findings. *Sage publication*.

[Glenstrup and Engell, 1995] Glenstrup, A. and Engell, T. (1995). Eye controlled media: Present and future state. *On-line source: http://www.diku.dk/hjemmesider/ansatte/panic/eyegaze/article.html*.

[Google,] Google. *On-line source: https://developers.google.com/translate/*.

[Inhoff and Rayner, 1986] Inhoff, A. W. and Rayner, K. (1986). Parafoveal word processing during eye fixations in reading: Effects of word frequency. *Attention, Perception & Psychophysics*, 40.

[Institue,] Institue, N. E. *On-line source: http://www.nei.nih.gov/health/eyediagram/eyeimages2.asp*.

[J. ward and Parkin, 2000] J. ward, R. S. and Parkin, A. J. (2000). The role of semantics in reading and spelling: evidence for the Ŝsummation hypothesisŠ. *Neuropsychologia*, 38.

[Jacquemin, 2011] Jacquemin, G. (2011). *User experience in Master 1 IT, ECE - Engineer School Paris.*

[Kurvers, 1007] Kurvers, J. (1007). Development of word recognition skills of adult l2 beginning readers. *Low-educated second language and literacy acquisition: Research, policy, and practice.*

[Lewis, 2009] Lewis, P. (2009). *Ethnologue: Languages of the World.* SIL International, 16th edition.

[Lundberg, 1999] Lundberg, I. (1999). *Learning to read and write : a cross-linguistic perspective*, chapter 9, Learning to read in Scandinavia. Campbridge university press.

[myGengo Inc.,] myGengo Inc. *On-line source: http://mygengo.com/api/libraries/.*

[O'Malley and Chamot, 1990] O'Malley, J. M. and Chamot, A. U. (1990). *Learning Strategies in Second Language Acquisition.* Cambridge University Press.

[O'Regan, 1980] O'Regan, J. (1980). The control of saccade size andfixation duration in reading: The limits of linguistic control. *Attention, Perception & Psychophysics*, 28.

[Tobii, 2010a] Tobii (2010a). An introduction to eye tracking and tobii eye trackers.

[Tobii, 2010b] Tobii (2010b). *Tobii T/X product description.*

[Tobii, 2010c] Tobii (2010c). *Tobii User Manual: Tobii X60 & X120 Eye Trackers.*

[Tobii, 2011] Tobii (2011). *Tobii SDK 3.0 Release Candidate 1 - Developers guide.*

[Weber, 1995] Weber, G. (1995). The world's 10 most influential languages. *Language Today.*

[Y. Ko, 2011] Y. Ko, M. Wang, S. Y. K. (2011). Bilingual reading of compound words. *On-line source: http://umd.academia.edu.*

[Young-Scholten and Strom, 2006] Young-Scholten, M. and Strom, N. (2006). First time l2 readers: is there a critical period? pages 45–68.

Appendix A

Second paper based prototype

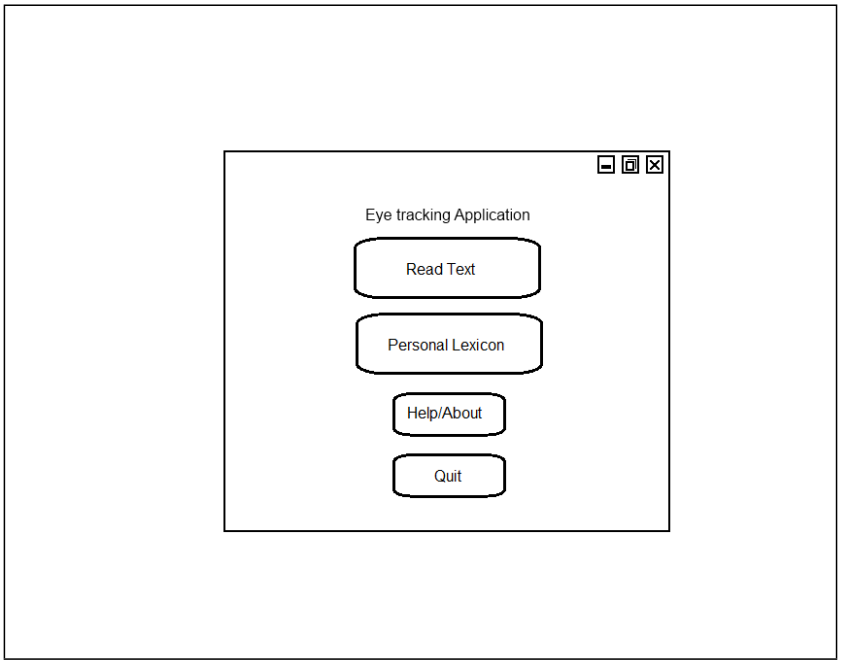


Figure A.1: Prototype 2 - Menu

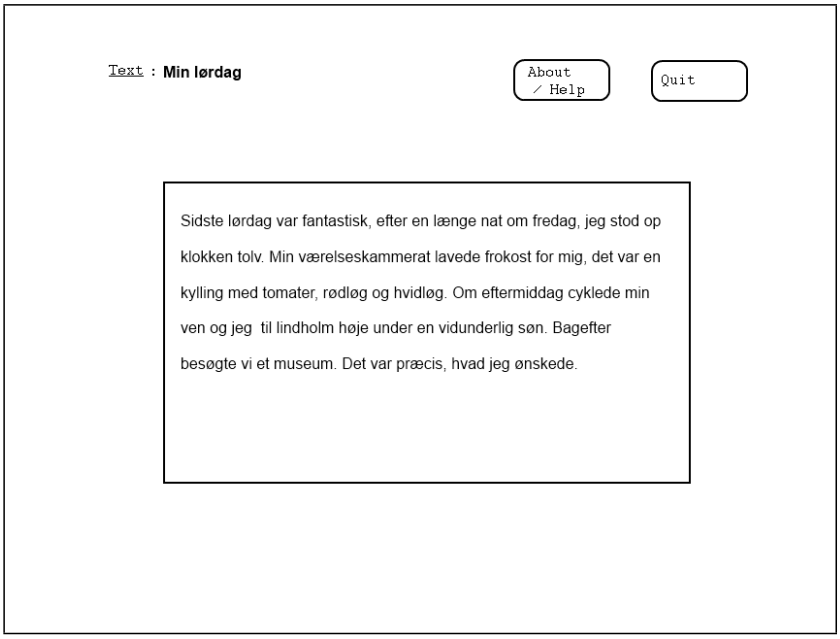


Figure A.2: Prototype 2 - Text alone

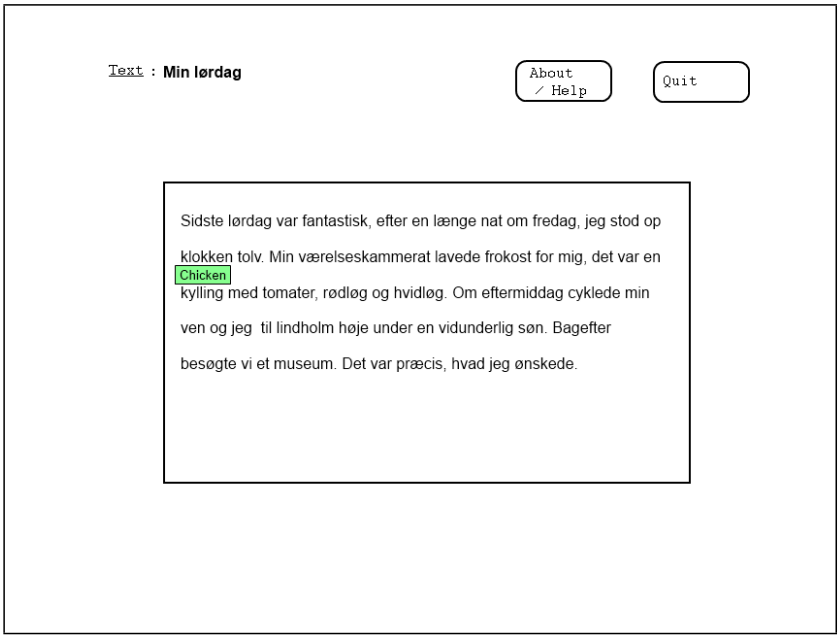


Figure A.3: Prototype 2(a) - Text with translation

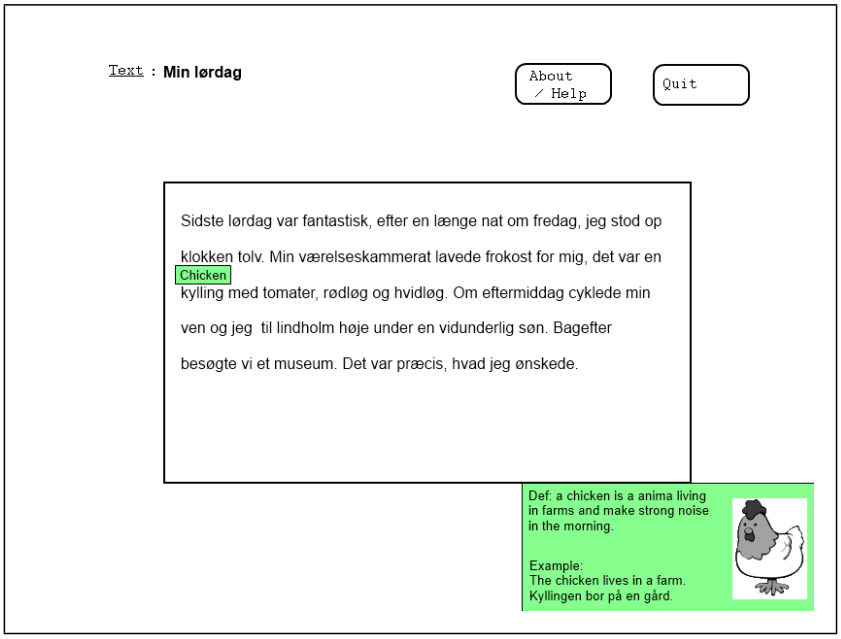


Figure A.4: Prototype 2(a) - Text with translation and advanced help

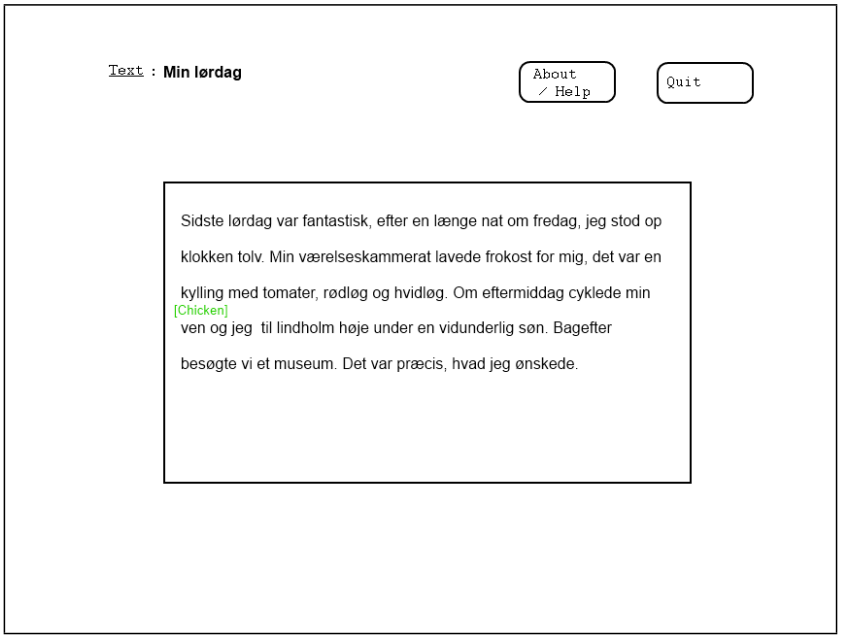


Figure A.5: Prototype 2(b) - Text with translation and advanced help

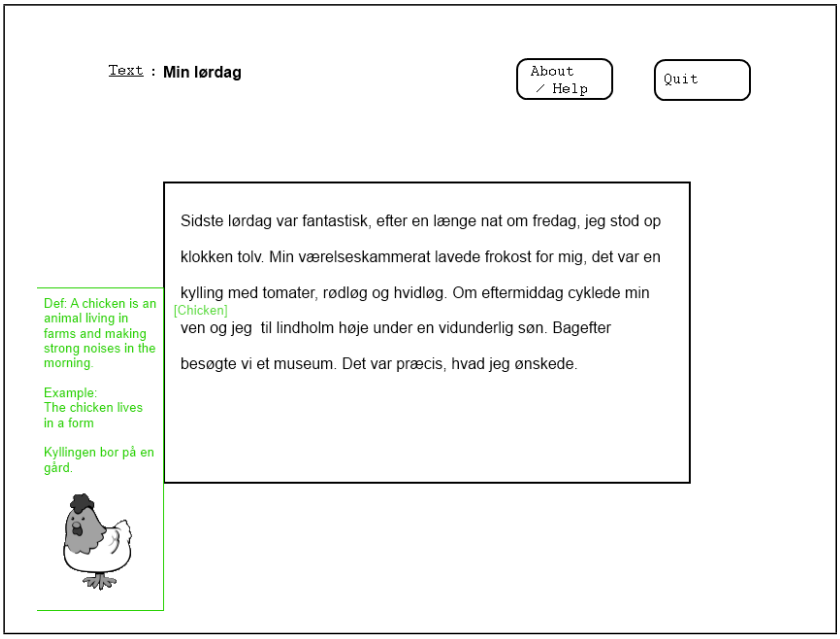


Figure A.6: Prototype 2(b) - Text with translation and advanced help

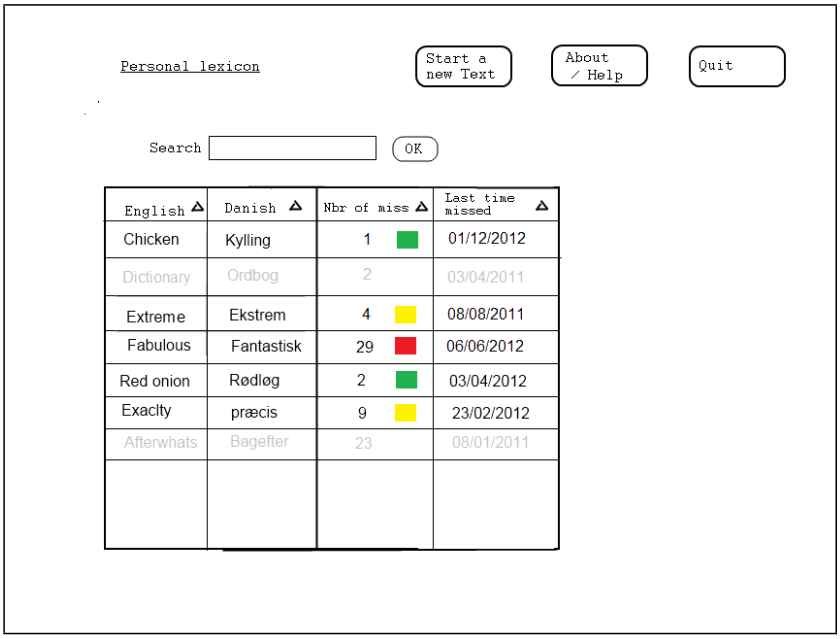


Figure A.7: Prototype 2 - personal Lexicon

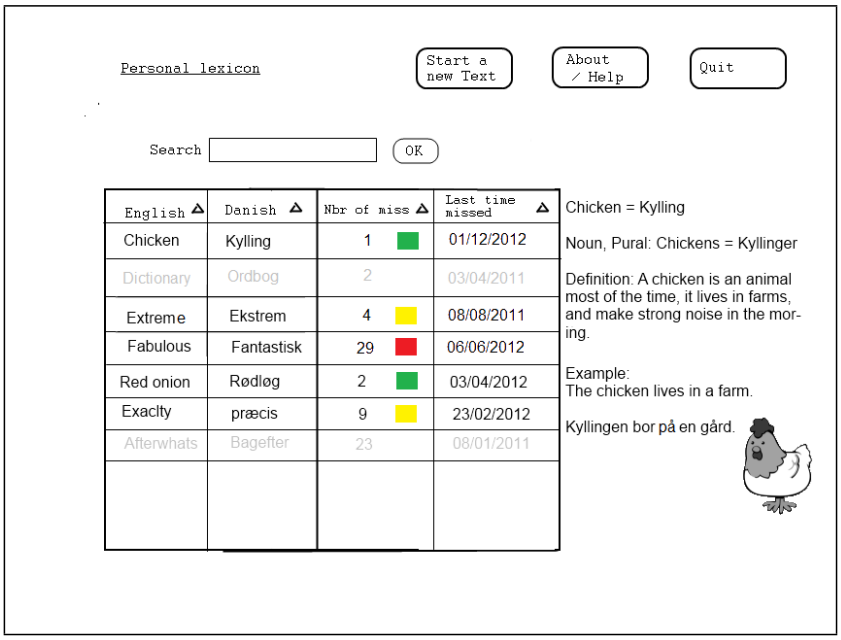


Figure A.8: Prototype 2 - personal Lexicon

Appendix B

Final paper based prototype

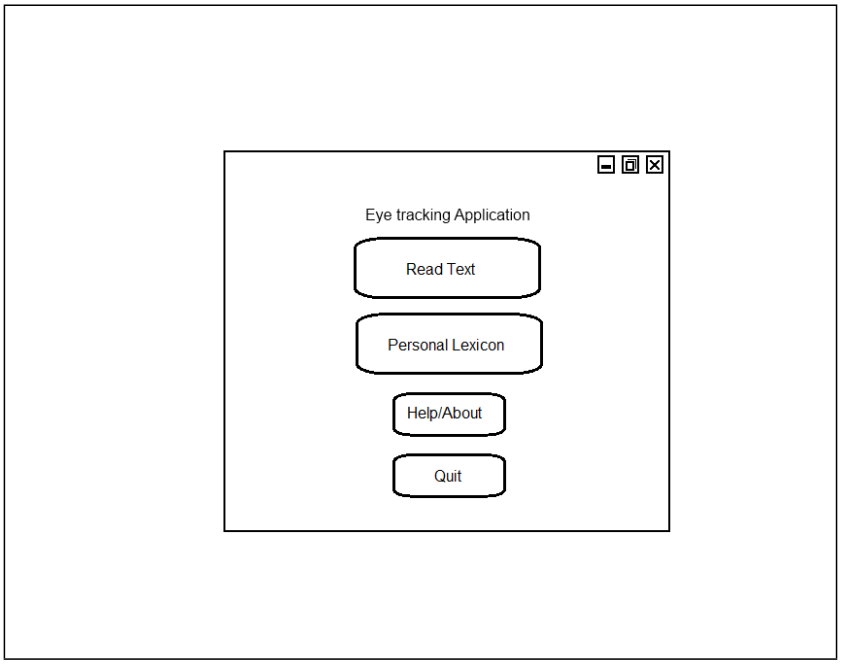


Figure B.1: Final prototype - menu page

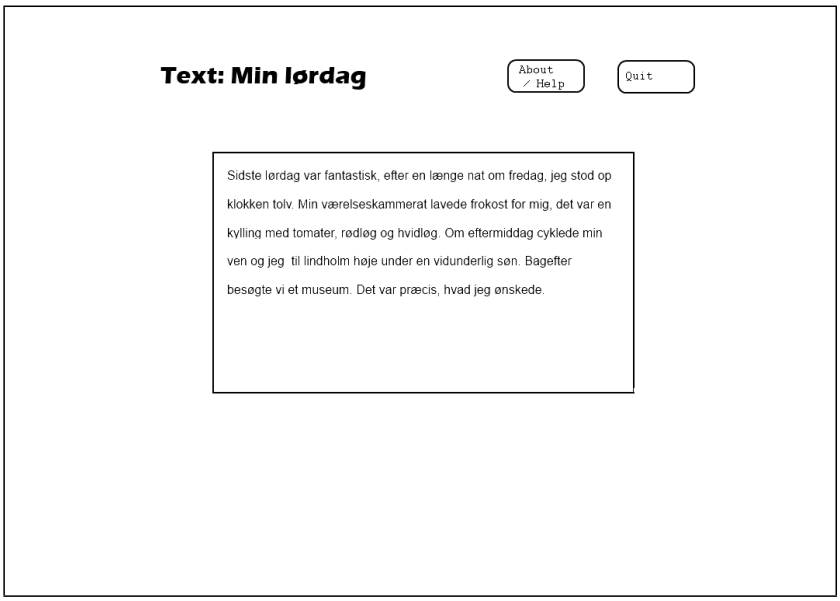


Figure B.2: Final prototype - text page

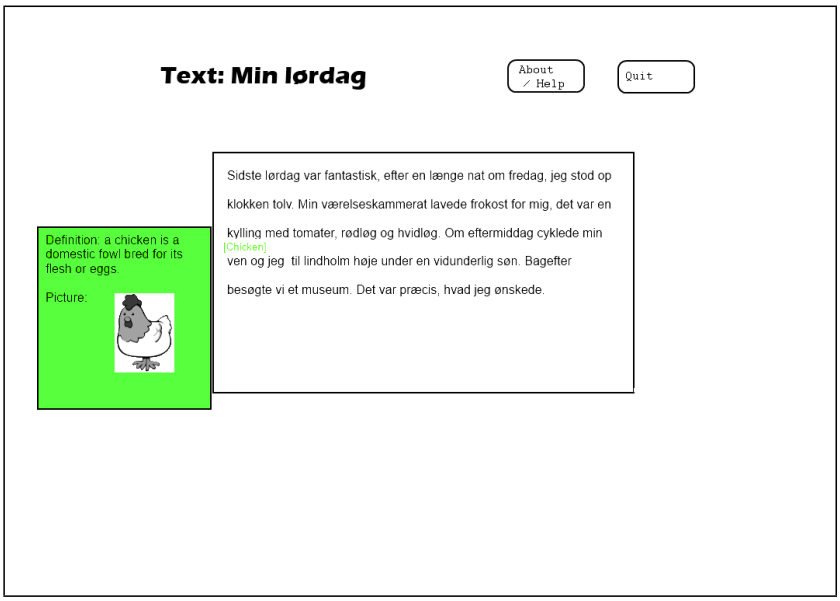


Figure B.3: Final prototype - text page with helps

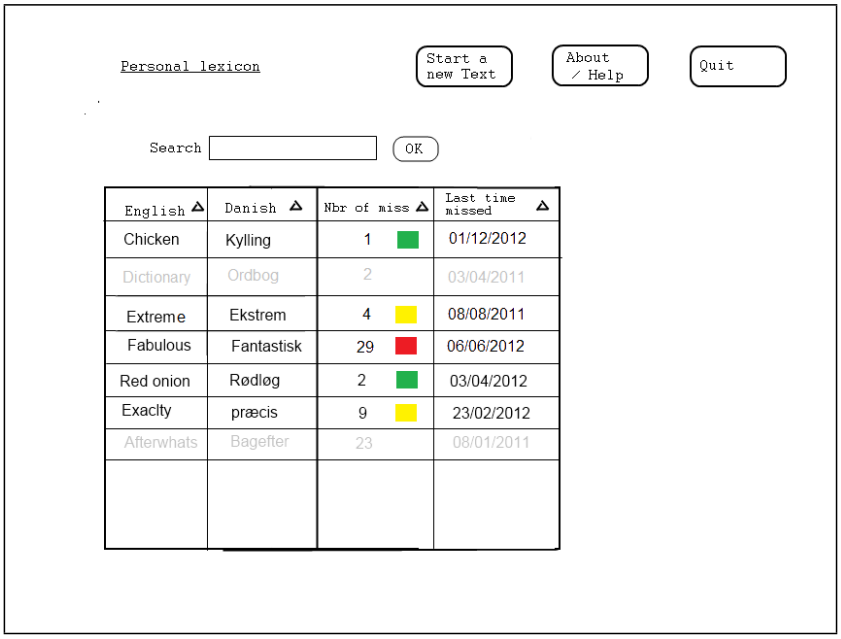


Figure B.4: Final prototype - personal lexicon page

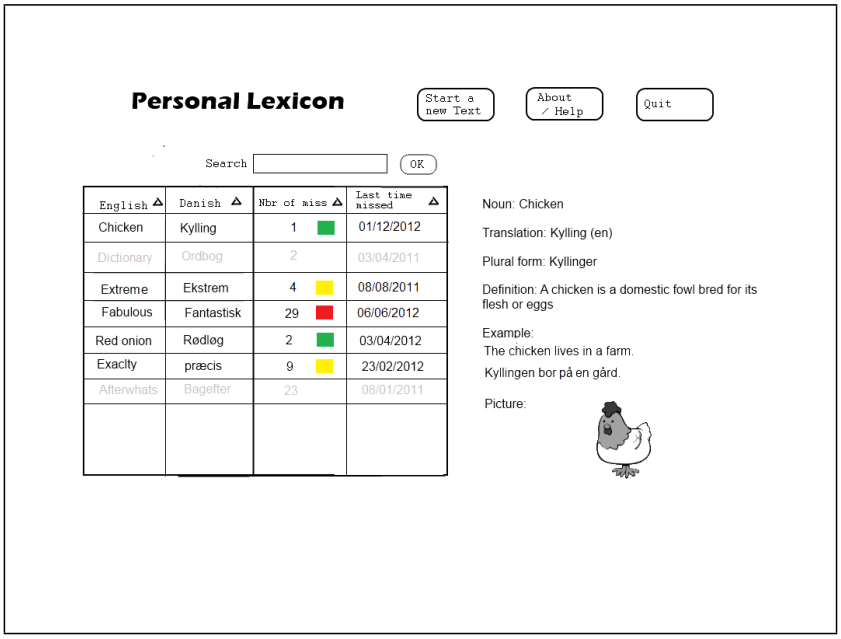


Figure B.5: Final prototype - personal lexicon page with definition

Appendix C

Implementation details

This part contains all the details about how the program had been implemented. First the full class diagram is presented, then the different classes of the application are presented : the master class, the eye tracker data analysis, the word database, the added controls and the forms. Finally the final prototype is presented.

Before to start, from [Tobii, 2011], it is important to consider the two following points in order to implement an application using this SDK:

- Having the dnssd.dll from Bonjour, it is an implementation of the zeroconf standard (developed by apple).
- At the beginning of every program using the Tobii to initialize the library through the command (in C#): Tobii.Eyetracking.Sdk.Library.Init()

C.1 Class diagram

The software architecture is structured around a main class "Master" containing an instance of all the graphical components of the application which are the Forms. All the forms inherit from the Form class and two of them use the two custom controls: "CalibrationPlotForm" and "CalibrationForm". The "FormLexicon" class use the class: "ListViewColumnSorter" which sort the columns of its "listView" control.

The Database package contains two classes; the "dictionary" one creates the dictionary from a text file, allows research in it; the other one "LexiconParser" manages the file containing the missed words by reading, modifying and saving it.

Finally the EyeTracking package is the one which receives and analyzes the data sent by the eye tracker and launches the corresponding functions in the "Forms" package.

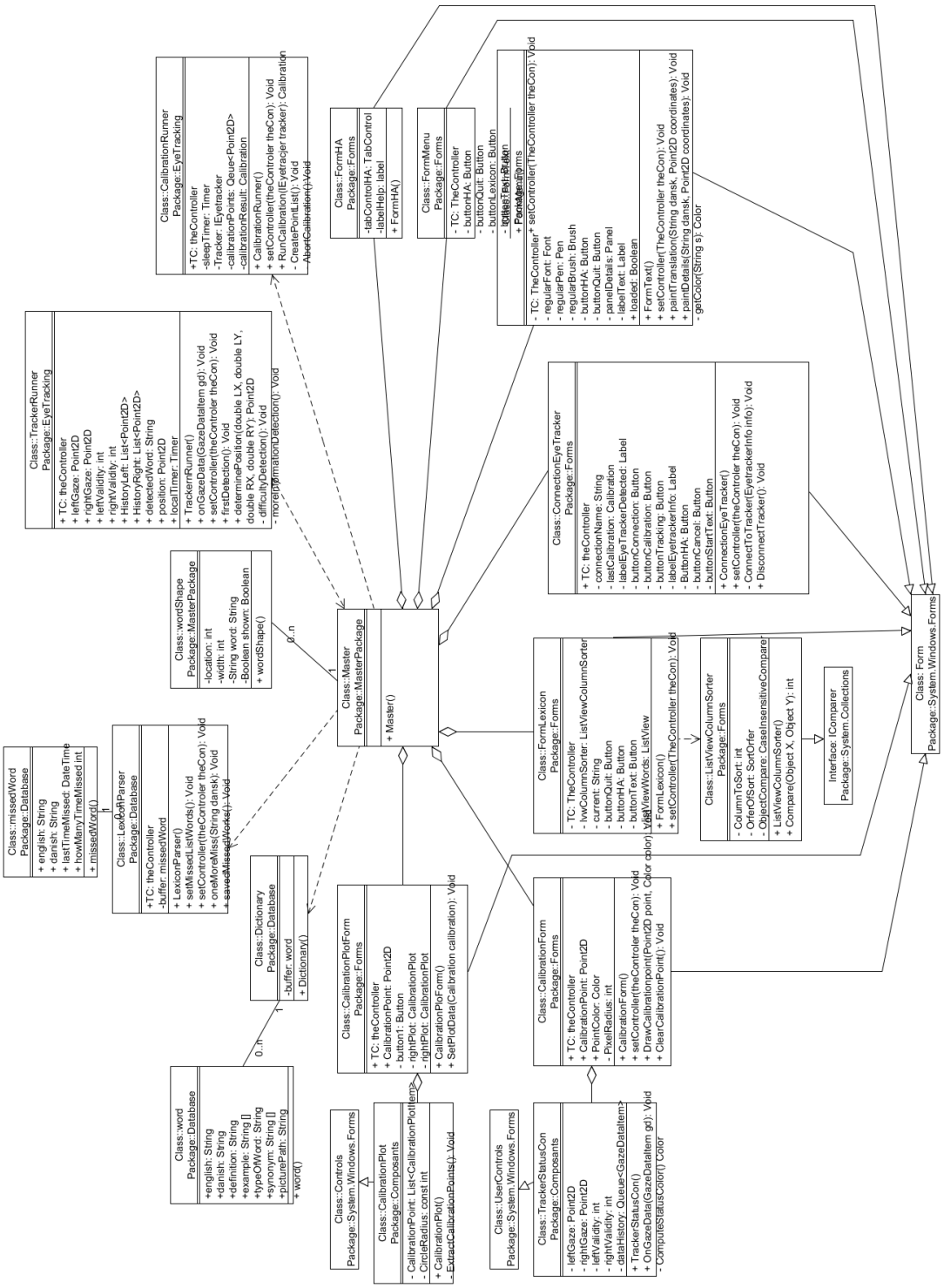


Figure C.1: Global Class Diagram

C.2 Master class

"Master" is the class managing the application. It has an instance of each form and also the list of the words that is empty at the beginning but will be completed while loading the text (see figure X.X):

wordPosition = new List<List<wordShape»();

This double list of "wordShape" corresponds to the list (List<List<») of each line of words (List<WordShape>). "Master" as the order of the words through the position of the "wordShape" in the double List but also the pixel position of each word on the screen. When the text is loading the location and the width of all word is given in the arguments of the "wordShape" constructor. They will be used to determine where a help must be provided when a difficulty has been detected.

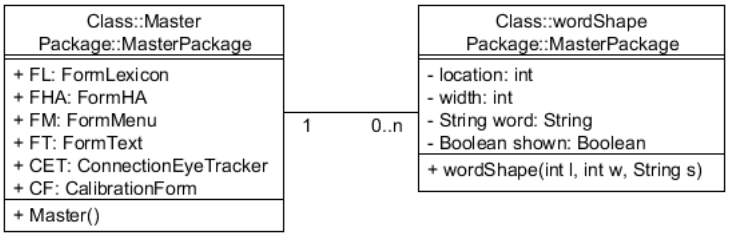


Figure C.2: Master package

C.3 Eye tracking data analysis

The two classes presented in this section are the most important for the tracking. "CalibrationRunner" allows to calibrate the camera and "TrackerRunner" analyzes the eye gaze data.

C.3.1 Calibration runner

The "CalibrationRunner" runs beside the "CalibrationForm", it creates the list of points that have to be shown for the calibration and send it one by one to the "CalibrationForm". Then it also informs the eye-tracker that the calibration is done. The figure X.5(Calibration Points) present the 9 calibrations points and their placement on the screen.

```
private void CreatePointList()
{
    calibrationPoints = new Queue<Point2D>();
    calibrationPoints.Enqueue(new Point2D(0.175, 0.2));
    calibrationPoints.Enqueue(new Point2D(0.5, 0.5));
    calibrationPoints.Enqueue(new Point2D(0.825, 0.2));
    calibrationPoints.Enqueue(new Point2D(0.825, 0.8));
    calibrationPoints.Enqueue(new Point2D(0.175, 0.8));
    calibrationPoints.Enqueue(new Point2D(0.35, 0.5));
    calibrationPoints.Enqueue(new Point2D(0.65, 0.5));
    calibrationPoints.Enqueue(new Point2D(0.5, 0.65));
    calibrationPoints.Enqueue(new Point2D(0.5, 0.35));
}
```

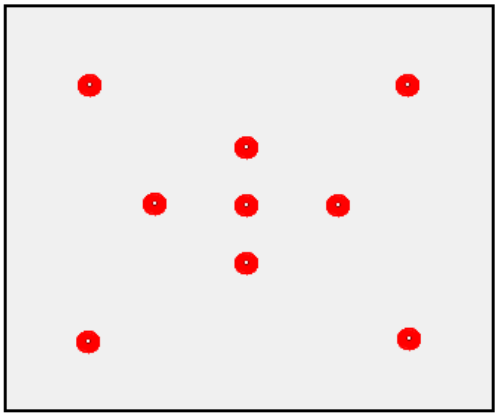


Figure C.3: Calibration Points

C.3.2 Tracker runner

As for the "CalibrationRunner", the "TrackerRunner" runs beside the "FormText" and receives the gaze data from the tracker at every time. The frequency of the ticks is 60 tick/second, at each tick (see figure 5.6 Action at each tick) the left and right gaze is saved if the validity variable (cf Table 3.1) is <=2, then the corresponding pixel position in the text is calculated. If the history lists are bigger than the defined limit, the first element is removed, then if the number of elements is enough (equal to the limit) the detection is launched.

```
  if (leftValidity <= 2 && rightValidity <= 2)
  {
      HistoryLeft.Add(leftGaze);
      HistoryRight.Add(rightGaze);
      HistoryPosition.Add(this.determinePosition(leftGaze.X, leftGaze.Y, rightGaze.X, rightGaze.Y));
  }

  if (HistoryLeft.Count >= (limit+1))
      HistoryLeft.RemoveAt(0);
  if (HistoryRight.Count >= (limit + 1))
      HistoryRight.RemoveAt(0);
  if (HistoryPosition.Count >= (limit + 1))
      HistoryPosition.RemoveAt(0);
  if (HistoryLeft.Count == limit && HistoryRight.Count == limit)
firstDetection();
```

Figure C.4: Action at each tick

The pixel position from the data provided from the tracker (a double type with a value between 0 and 1) is determined by using the function of the figure 5.7. The four arguments are the X and Y double variables for each eye. The purpose is to determine the pixel position in the text but not on the main window, that is why the numbers (dx and dy) corresponding to the distance between the the right top corners of the text and the main window in the "FormText" are subtracted to the sent arguments.

```
public Point2D determinePosition(double LX, double LY, double RX, double RY)
{
     double dx = 0.1904762;
     double dy = 0.1904762;
   Size monitorSize = Screen.PrimaryScreen.Bounds.Size;
   Point2D p2 = new Point2D(
                  (((LX - dx) * monitorSize.Width) + ((RX - dx) * monitorSize.Width)) / 2,
                  (((LY - dy) * monitorSize.Height) + ((RY - dy) * monitorSize.Height)) /
                  );
    return p2;
}
```

Figure C.5: Function determinePosition

To detect the right word where the user is looking at, the average of the X and Y contained in the history of each eye positions is calculated. Then each value of the history is compared to this average number, if the distance between a point in the history and the average is too big a counter is decremented, when the counter is under a certain value, the history is cleaned and the reader is not considered as being stuck on a word, see the code in the figure 5.8.

```
int counter = (HistoryPosition.Count - 1);
  for (int i = 1;i<(HistoryPosition.Count);  i++)
  {
      if (Math.Abs(HistoryPosition[i].X - averagePosition.X) > 50
          || Math.Abs(HistoryPosition[i].Y - averagePosition.Y) > 30)
      {
          counter--;
      }
  }
  if (counter <= (limit*66/100))
      toPaint = false;
```

Figure C.6: Paint depending of the history

If the help is required (the gaze remains in a certain area for a period of time) then first step is to determined the line, as the space between two lines does not change (27 px), the Y coordinate of the average position allows to determine which line the reader is reading. Then the corresponding list of words present in the "Master" class is used to determine the word by comparing the average X coordinate to the position of every word. If the position matches and the translation has not been shown already, then the translation appears and the word is sent to the list of missed words; else if it has been shown before, the corresponding detailed information appears. In every case, the history is cleaned to make each detection independent from the previous data. The figure 5.9 corresponds to the code of the detection of the right word.

```
if (toPaint == true)
  {
   int line=0;
   line = (int)(((averagePosition.Y-10) / 54));
   foreach (wordShape ws in TC.wordPosition[(line)])
   {
            if (averagePosition.X >= (ws.location - 8) && averagePosition.X < (ws.location + ws.width + 8))
            {
                positionX = ws.location;
                detectedWord = ws.word;
                position = new Point2D(positionX + ws.width / 2, (line + line + 1) * 27);

                if (ws.shown == true)
                {
                    this.moreInformationDetection();
                }
                if (ws.shown == false)
                {
                    this.difficultyDetection();
                    ws.shown = true;
                }
                break;
            }
        }
```

Figure C.7: Code for word detection

C.4 The words' database

This part concerns the way the words are stored, the dictionary and the lexicon parser simply read a file containing the words. The two classes present in the corresponding package (figure x.10) simply read this file.

When the two classes are instantiated, they both initialize a list of object corresponding to the words and their added information (the "word" class for "Dictionary" and "missedWord" for the lexicon parser) present in two distinct text files.

The two files read are list of words where each line represents one word with its corresponding information separated by the ';' or the '-' character (see figure X.11). Every time a part of the application needs information about a word, the "List<word>" of the dictionary will be solicited however concerning the "LexiconParser" its list is only used for the personal lexicon.

C.5 Added controls

The Tobii SDK gives the library but also some examples of code, such the calibration of the camera, two controls present in those examples have been slightly changed and reused in this application. The two controls are used to check that the gaze data are well received ("TrackerStatusControl") and to see the calibration plot: "CalibrationPlot".

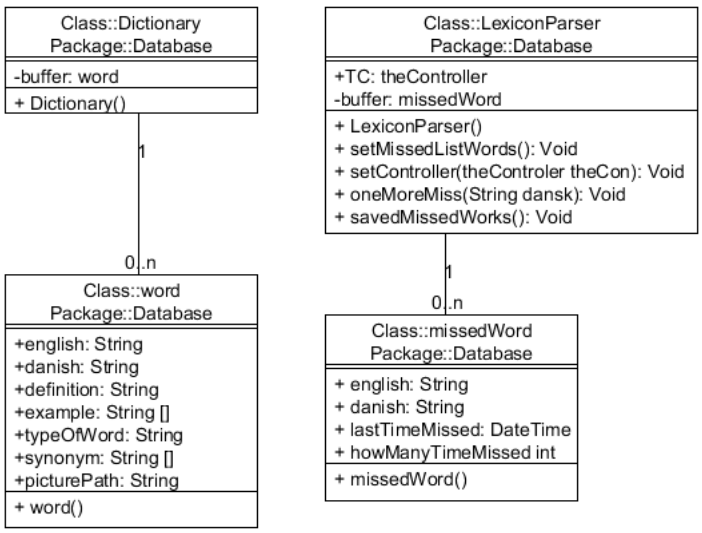


Figure C.8: Database package

Part from Dico.txt

```
when;da;at or during a time; when I bike-da jeg cykler;conjunction;because-then;
have to;skal;be obliged to, must, represents an obligation;I have to go-jeg skal gå;verb;skulle-skal- - ;
meet;møde;happen to come into the presence; I meet a friend - jeg møder in ven;verb;mødte-møder-møde-mødt;
the;det;design something; The biggest monkey i saw - Det største abe jeg så;article;de-den;
first;første;comming before all the others;The first exercise - Det første øvelse;adjective;første;
```

Part from missedWords.txt

```
when;da;24-4-2012;11
meet;møde;24-4-2012;7
first;første;26-4-2012;19
o'clock;kl.;23-4-2012;5
```

Figure C.9: Part of the text documents

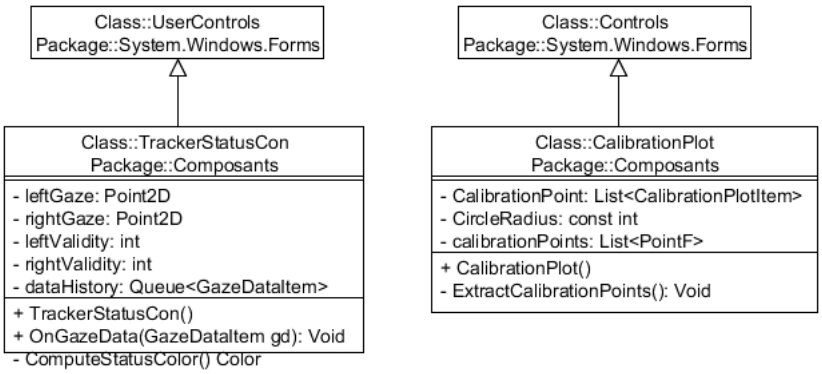


Figure C.10: Composants package

C.6 Forms

All the classes in the Form package inherit from the "Form" class, the only exception is the column sorter used by the "FormLexicon". The functions described in this section

concern the connection and calibration of the Tobii eye-tracker, the "painting" of the help on the text and the loading of this one. All the other functions in the package are based on click events and hide/show functions that creates the interactions between the forms.

C.6.1 Eye tracker detection and calibration

The "ConnectionEyeTracker" class instantiates an "EyeTrackerBrowser" class which detects any eye tracker on the network and starts it while the form is loading. When one eye tracker is detected, its corresponding information are displayed on the form. This form also starts the communication between the eye-tracker and the "TrackerRunner" class concerning the eye gaze data when clicking on the button "Start Tracking".

When clicking on the "start calibration" button, the "CalibrationForm" shows 9 points with 1,3 seconds of interval (see figure 5.5). The calibration takes 12 seconds and finish by showing the calibration plot of each eye.

C.6.2 Read a Text

When loading the text in the "FormText" class, the double list "wordPosition" in the "Master" class is initialized. The text is first divided in String (one for each word) and each one of those is then used to know the two positions: the one in the text (word nř on line nř) and the one on the screen (coordinates X and Y). The label used to contain the text has a define width and the font is Lucia Console which as the same size for every character; thus the maximum number of characters in each line remains the same: 62 and each one of those has a width of 16 pixel. The figure 5.14 is the corresponding code.

When a difficulty has been detected, the following function is used in "FormText"

```
public void paintTranslation(String dansk, Point2D coordinates)
```

The word send in danish is first compared with all the words present in the dictionary then its English translation and type of word is extracted from it, see figure X.15. Then the translation is simply draw under the word using the coordinates in argument (figure X.16). The same principle is used for the detailed information, however in this case more information are extracted from the dictionary (definition, example etc..).

```
List<String> line = new List<String>();
List<wordShape> wordshape = new List<wordShape>();

int sizeLetter = 0;
int locationPixel=5;

foreach (String s in text)
{
    sizeLetter = sizeLetter + s.Length + 1;
    if (sizeLetter >=62)
    {
        TC.wordPosition.Add(wordshape);
        this.labelText.Text = this.labelText.Text + "\n\n";
        sizeLetter = 0 + s.Length + 1;
        locationPixel = 5;
        wordshape = new List<wordShape>();
    }
    wordshape.Add(new wordShape(locationPixel, (s.Length * 16), s));

    locationPixel = locationPixel + (s.Length + 1) * 16;
    this.labelText.Text = this.labelText.Text + s + " ";
}
TC.wordPosition.Add(wordshape);
```

Figure C.11: Text load and initialization

```
foreach (word w in TC.DI.AllWords)
{
    if (w.danish.Equals(dansk))
    {
        englishTranslation = w.english;
        typeOfWord = w.typeOfWord;
        originalDico = w.danish;
        break;
    }
```

Figure C.12: Code of the dictionary extraction

```
regularBrush = new SolidBrush(this.getColor(typeOfWord));
ps.DrawString("[" + englishTranslation + "]", regularFont, regularBrush,
    //test
 new Point((int)coordinates.X - (englishTranslation.Length * 7), (int)coordinates.Y));
```

Figure C.13: Code for drawing the translation

Appendix D

User tests

D.1 Text used

Here is the text used for the user tests:

Title: Kirstens dag

Jeg står op kl. 6.00 om morgenen, da jeg skal møde kl. 7.00. Det første jeg gør efter jeg har fået tøj på, er at lave kaffe. Det er det eneste jeg får derhjemme, da jeg spiser morgenmad på arbejdet. Jeg arbejder som ufaglært hjælper på plejehjem rundt omkring i Aalborg, så jeg arbejder aldrig de samme steder. Omkring kl. 10.00 får jeg morgenmad på arbejdet, med de andre medarbejdere. Vi snakker om hvad der skal ske i dag. Jeg har tit fri kl. 13.00. Derefter kommer jeg tilbage hjem, spiser, ser fjernsyn og slapper ellers bare af. Inden butikkerne lukker tager jeg ned og handler og gør klar til aftensmad. Jeg spiser et varmt mad hvis det er overskud, så ser jeg lidt nyheder, inden jeg smutter i seng.

D.2 MCQ

This is the MCQ used during the user tests. For each question the "Other" option was possible with the possibility to enter an alternative answer.

1. At what time does Kirsten have to be at work in the morning?
    - 6 a.m.
    - 7 a.m.
    - 8 a.m.
2. What is the first action she does after put her clothes on?
    - Go to work
    - Eat the breakfast

- Make some coffee
- Brush her teeth

3. When does she eat her breakfast?

    - Between 6 and 7 a.m.
    - She does not eat breakfast
    - At 7 a.m.
    - At 10 a.m.

4. Where is she working?

    - In nurse houses in Aalborg
    - At home
    - In a hospital

5. What is her job?

    - Doctor
    - Nurse
    - She does not work
    - Unskilled helper

6. What is she doing with the other employees at 10 o'clock?

    - Talk about what have to be done
    - Talk about yesterday
    - Talk about what have been done
    - Drink a coffee

7. What is the first thing she does when she is free?

    - Watch TV
    - Eat
    - Go back home
    - Relax

8. What is the main plan in the afternoon?

    - Visit a friend
    - Go shopping
    - Go running

9. What is the last task she does before going to bed

- Read a book
- Watch a movie
- Watch the news
- Eat diner

D.3 Nationalities of the participants

- Estonian
- French
- German
- Greek
- Lithuanian
- Polish
- Romanian
- Spanish
- Vietnamese