SCHOOL OF SCIENCE AND ENGINEERING

Fast Reading App in Arabic

Prepared by:
Mehdi Lebbar – ID 51916

Supervisor:
Dr. Violetta Cavalli-Sforza

Coordinator:
Dr. Yassine Salih-Alj

Spring 2015
Acknowledgements

This project would not have come to life without the help of many people, either by their support, help or advice. Hereafter, I would like to thank them for their help. First of all I would like to thank my supervisor Dr. Violetta Cavalli-Sforza for her valuable pieces of advice and great supervising methodology. Secondly, I would like to thank my parents, my siblings and all my friends for their continuous support and ideas they provided me with. Finally, I would like to thank each person who agreed on getting the experiment, and helped me gather the necessary data for the success of this project.
Contents

Abstract ........................................................................................................................................... 3

1. Introduction .................................................................................................................................. 4
  1.1. Project Scope .......................................................................................................................... 5
  1.2. STEEPLE Analysis .................................................................................................................. 5

2. Feasibility Study and Analysis .................................................................................................... 6
  2.1. Economic Feasibility .............................................................................................................. 6
  2.2. Technical Feasibility .............................................................................................................. 6
  2.3. Technological Analysis .......................................................................................................... 6
      2.3.1. Application delivery ....................................................................................................... 6
      2.3.2. IDE ................................................................................................................................. 7

3. Experimentations and Analysis .................................................................................................. 7
  3.1. Background ........................................................................................................................... 7
  3.2. Experiment Design ................................................................................................................ 8
      3.2.1. Terminology .................................................................................................................... 8
      3.2.2. Hypothesis ...................................................................................................................... 8
      3.2.3. Variables ......................................................................................................................... 8
      3.2.4. Independent variables ................................................................................................... 8
      3.2.5. Experiment Evaluation .................................................................................................. 10
      3.2.6. Experiment Structure .................................................................................................... 10
  3.3. Experiment Description ........................................................................................................ 11
  3.4. Results ................................................................................................................................... 12
      3.4.1. TEST 1 and Results: Testing ORP location. (far left, middle or far right) ............... 12
      3.4.2. Test 2 and Results: Achieving the objectives ............................................................... 13
  3.5. Discussion and conclusion .................................................................................................... 15

4. Software Engineering Methodology .......................................................................................... 17
  4.1. Requirements Engineering .................................................................................................... 17
      4.1.1. Specific Requirements ................................................................................................... 17
  4.2. System design ....................................................................................................................... 19
      4.2.3. Use-case Diagram ......................................................................................................... 19
      4.2.4. Class diagram ................................................................................................................ 20
  4.3. Design and implementation .................................................................................................. 20
4.3.1. Application Mockups ................................................................. 21
4.3.2. Implementation and technology enablers ................................... 22
5. Future Work .................................................................................. 23
6. Conclusion .................................................................................... 24
References ....................................................................................... 25
Appendix A: GNU GPLv3 ................................................................... 26
Appendix B: Test 2: List of most common awzan ............................... 27
Appendix C: Screenshots from application ........................................ 28
Abstract

The conventional page format for presentation of text content has advanced over several number of years into its current structure. Rapid Serial Visual Presentation (RSVP) is a relatively new reading method in which text is read by presenting a word at a time, one after another while keeping the word center static. The need for such a new way of text presentation is a consequence of our evolving surroundings: individuals are progressively more portable and use smartphones, which are difficult to read from; information is growing rapidly; and memory on new devices can hold thousands of bits, making it possible to carry a library in your pocket. This capstone project demonstrates RSVP for the Arabic Language. It is based upon research done in an Applied Research course (EGR 4303), taken in the summer of 2014. The project investigates the Arabic reading experience using RSVP through experiments with human subjects, in order to recognize the Optimal Recognition Point (ORP). The project also produces as output an Android application using agile software engineering.
1. Introduction

Reading can be seen as a complex form of pattern recognition that has evolved over millions of years. The traditional page format for presentation of a text has progressed over thousands of years into its present form. This work promotes a relatively new and unconventional form of text presentation, in fact it will be the first Rapid Serial Visual Presentation (RSVP) software for Arabic.

Traditional reading is time consuming because your eyes have to move from word to word and line to line. According to Spritzinc.com, when reading, only around 20% of your time is spent processing content. The remaining 80% is spent physically moving your eyes from word to word and identifying the next word.

Imagine reading your favorite 120,000 words novel in 2 hours, or reading your emails, text messages, news and magazine articles in minutes. Indeed with Spritz technology, users have been able to read up to 1000 words per minute. On the one hand, in the era of speed and technology, reading quickly can be an important asset in your daily life activities. On the other hand, Spritz came to compensate for the drawbacks of using small screen displays, such as smart watches, google glasses, and smartphones, while dramatically improving their readability.

Drawing knowledge from Spritz, this capstone research project leads the way for developing an Arabic version of Spritz. To that end, this project identifies the ORP of Arabic language by applying a standard experimental research methodology. Through some experiments, I was able to accurately identify the ORP in different word patterns, then to produce a suitable mobile application.
1.1. Project Scope

This capstone project called “Fast Reading App in Arabic” is in partial fulfillment of the Bachelor of Science in Computer Science’ requirements in Al Akhawayn University. This project is the result of my work during this whole semester under the supervision of Dr. Violetta Cavalli-Sforza and continues work done under Applied Research course (EGR4303) in summer.

1.2. STEEPLE Analysis

In developing this project, I took into account different considerations known as STEEPLE:

- **Societal**: This application will definitely be a social help for Arabic readers. A 2008 UN Survey [8] comes to an alarming conclusion: “The average person in the Arab world reads no more than 4 pages a year. Americans read an average of 11 books a year.” While this is partially due to the high illiteracy in Morocco, this survey also argues that the lack of book printing in the Arab world explains why Arabs don't read nearly as much as they could. Hopefully, this application will compensate the scarcity of book printing to increase the reading habits of the Arab world.

- **Technical**: The application is intuitive and does not involve any complex technologies that the user cannot handle. The only knowledge required of the user is how to use a smartphone.

- **Environmental**: The application does not have any direct relation with the environmental aspect. As an Android application, the only environmental concern is the source of energy used to power the device.

- **Ethical**: The application does not harm or lead to any unethical behavior; instead it will be there to improve the quality of life.

- **Political & Legal**: The application has nothing to do with any political concern, and does not break any law.
• **Economic**: The application does not require any commercial resources and is intended to be free of use. [ See Appendix - A ]

# 2. Feasibility Study and Analysis

## 2.1. Economic Feasibility

The project is economically feasible as it simply requires a mobile phone with Android operating system. The application is intended to be free to download once released into Google Play Store. The users must be able to connect to internet and download the application, this would be the only cost incurred in the project.

## 2.2. Technical Feasibility

The current project is technically feasible as the application was previously deployed on Google Play Store for Latin languages.

## 2.3. Technological Analysis

### 2.3.1. Application delivery

Since the application should execute without internet access, the delivery choice is constrained by the nature of the application. Thus the application will be native one i.e. it will be installed through the Google Play Store directly into the device.

*Advantage*: Because native apps are written for a specific platform, they can interact with and take advantage of operating system features and other software that is typically installed on that platform.

*Disadvantage*: Because all the application modules will be on one side, it can be heavy to download.
2.3.2. IDE

There are two primary integrated development environments (IDE) for Android: Eclipse and Android Studio. Eclipse was once considered as the default IDE for Android development. However, Google finally released the long awaited stable version of Android Studio 1.0 [2], which is fully tested and bug free. For this project, I chose to develop using Android Studio.

3. Experimentations and Analysis

3.1. Background

Rapid Serial Visual Presentation (RSVP) is a relatively new method of text presentation. As the name suggests, RSVP serially presents text one word at a time in a rapid sequence at the same location on the screen. RSVP method is a means of increasing reading speed by eliminating the need to make eye movements.

Rapid Serial Visualization Presentation has been researched [7] and found to be faster than standard reading and with the same or higher comprehension. RSVP research [7] has shown that readers read and understand RSVP texts presented as fast as 600 wpm (words per minute).

Projecting words one at a time is not sufficient for an efficient and comfortable reading. Words, in this project, are helpfully presented at their Optimal Recognition Point (ORP), highlighting the letter that is most crucial for your brain to process the word. ORP is based on the Parallel Letter Recognition model [3], which is the model most agreed upon by psychologists at this time. This model argues that readers are focusing on a fixation point, usually a letter. Thus, the letter information is used to recognize the words.
3.2. Experiment Design

3.2.1. Terminology

Root: The root in Arabic language is the verb of which the rest of the forms are derived. The root is a sequence of consonants or semi-vowel sounds ‘y’ and ‘w’ that can be found in all the words that are derived from it. Most roots are composed of three letters, though some have four.

Awzan: Each three letter Arabic root can be transformed into one of fourteen possible verb forms (الأوزان, al-awzan), in this research, the focus will be on verb forms.

Wazn: is the singular form of awzan.

3.2.2. Hypothesis

The hypothesis of this experiment is that the ORP is located within the center of the wazn. For 4 letters awzan, the center can either be the second letter of the wazn or the third letter. For 5 letters awzan, the center will be the third letter of the wazn. However two possibilities were tested: whether the ORP is located at the second or third letter. Depending on the wazn, this places the ORP in the root or not. As for the 6-letter wazn, two positions were tested: whether the ORP is located within the root (4th letter) or not (3rd letter).

3.2.3. Variables

When performing experiments, the key is to vary one factor at a time or, occasionally, a combination of factors, and to observe and measure the impact of this change on the user behavior. By manipulating one or more independent variables, where the variables are independently determined and constructed by the researcher, we try to cause a variation in participant responses. [4]

3.2.4. Independent variables

An independent variable is an independent characteristic (e.g., man vs woman, 18-25 years vs. 45-60 years, experienced vs. novice driver) or the physical environment (e.g., presence / absence
of others, wall color, noisy vs calm ...) or a presented stimuli which is manipulated by the researcher in order to monitor or analyze its impact on the behavior, condition mind or mental processes studied.

In this project, the independent variables relevant to the experiment are the reading fluency and the highlighted & centered letter.

The reading fluency is a given variable that is inherent to the subject, which the experimenter cannot directly act upon. Hence, I was able to distinguish three levels of readers’ fluency using a questionnaire presented before starting the experiment (Figure 3.1). Because the ability to recognize a word may differ among readers of different levels of fluency, reading fluency is defined as being “Reading rapid, efficient, accurate word recognition skills that permit a reader to construct the meaning of text.” [5]

Figure 3.1 Questionnaire used to distinguish participants’ reading habits.
The second independent variable is the highlighted and centered letter. By changing its location, we will be able to recognize which one is optimal in the most common Arabic word forms [1][10]. The word list in Appendix B consist of the most common derivations of the three letter root. The first two awzan فعل and فعل فعل were omitted, since they are only three letters awzan, so the cost of testing would likely not be worth the amount of information they would provide. The highlighted and centered letter is our first independent variable. The key is to change the focus point of two examples of the same wazn so as to identify the best position in that particular wazn.

3.2.5. Experiment Evaluation

The experiment will be able to achieve the following objectives:

1. Identifying the ORP in different awzan.
2. Identifying the correlation, if any, between the word length and the location of the ORP.
3. Knowing if the ORP is always within the root.

Several ways can be used in order to achieve the above objectives. I chose the following approach. By experimenting with the awzan in the “ماضى” (perfective) I was able to determine whether the reader was able to recognize variations of a particular wazn. I did this by choosing a particular wazn, and making 3 variants of the word by changing the first, second or third letter of the root in each variant. At the same time, I changed the highlighted & centered letter. The participant was then prompted to choose between the three examples. Performance with the different awzan, the one-letter changes and the different highlighted letters allowed us to know what centered position is most accurate for recognizing the word.

3.2.6. Experiment Structure

Two experiments were conducted. The first one aimed to know if the ORP falls within the center of the word or not. Indeed, scientifically speaking, I tested all the positions so as to be able to most effective one. Then, depending on the results, a second test was conducted.
• **If the ORP falls within the center**: In order to recognize which letter in that *wazn* is the optimal for the word recognition, the participant will be prompted with two examples of the same *wazn*, only the centered and highlighted letter will be changed. Table 3.1 represents an example of the words to be presented to the participants in this case.

<table>
<thead>
<tr>
<th>صَا</th>
<th>دَرَ</th>
<th>صَا</th>
<th>دَرَ</th>
<th>صَا</th>
<th>دَرَ</th>
<th>عَاكْسَ</th>
<th>عَاَلَ</th>
<th>عَاَمَلَ</th>
<th>عَاَدَلَ</th>
<th>صَا</th>
<th>دَرَ</th>
<th>صَا</th>
<th>دَرَ</th>
</tr>
</thead>
</table>

Table 3.1: Example of words to be presented to the participant

• **If the ORP is not necessarily within the center**: We will have to test a range of possibilities for the same *wazn*, in order to identify one or multiple ORPs.

### 3.3. Experiment Description

The experimental method consisted in creating a situation that tested and measured a specific phenomenon based on the manipulation of one or more factors.

The aim of the experiment was the comparison between equivalent groups of participants in order to recognize the optimal recognition point under different *awzan*.

By using an experimental tool, previously developed in the EGR 4303 Applied Research course, I was able recognize the most efficient and comfortable way of reading, concretely speaking the Optimal Recognition Point. Using the data acquired, I analyzed and was able identify the ORP then I followed software engineering standards to lead to a standardized and viable mobile application.

In both experiments, the participants were told that the words do not necessarily convey any meaning, and they were asked to keep their calm and concentrate. The experiment tool randomizes the *awzan* so each two examples of the same *wazn* are not necessarily successive.
3.4. Results

3.4.1. TEST 1 and Results: Testing ORP location. (far left, middle or far right)

20 subjects participated in this experiment, so as to know where the optimal recognition point falls within, either far left or far right. The list of words below (Table 3.2) represents the words that were tested. They include 3 triliteral and 1 quadriliteral *awzan*. The goal of this test was to know whether the *awzan* are best recognized when the highlighted and centered letter (not necessarily in the root) is at the center, at the beginning or at the end of the word.

<table>
<thead>
<tr>
<th>اِنْطَلَق</th>
<th>اِنْطَلَفَ</th>
<th>اِنْطَلَفَ</th>
<th>اِنْضَرَب</th>
<th>اِنْضَرَبَ</th>
<th>اِنْضَرَبَ</th>
</tr>
</thead>
<tbody>
<tr>
<td>أَعْقَلَ</td>
<td>أَعْقَلَ</td>
<td>أَعْقَلَ</td>
<td>أَغْتَكَ</td>
<td>أَغْتَكَ</td>
<td>أَغْتَكَ</td>
</tr>
<tr>
<td>أَعْقَلَ</td>
<td>أَعْقَلَ</td>
<td>أَعْقَلَ</td>
<td>أَغْتَكَ</td>
<td>أَغْتَكَ</td>
<td>أَغْتَكَ</td>
</tr>
</tbody>
</table>

Table 3.2: The full test set of 1st experiment.

The results were mostly conclusive as almost all the words were easily recognizable when they were highlighted and centered in the middle. This test gave us a stepping stone on what to test for the next experiment. On the other hand, having the ORP within the middle will prevent us from testing a whole range of possibilities. Figure 3.2 presents the results of our first experiment.

![Figure 3.2: Test 1 Results](image)
3.4.2. Test 2 and Results: Achieving the objectives

40 Subject participated in this experiment; most of them are students at Al Akhawayn University. Our sample however was diversified, three types of readers were distinguished in this experiment, according to their reading levels (Figure 3.3). The participants they were first presented with the questionnaire (Figure 3.1) to fill in, then started the experiment. In the experiment they were presented with the two examples of each *wazn* for the eight *awzan* shown in Appendix – B. The examples in the Appendix also constitute the test set for this second experiment.

![Figure 3.3: Representation of the number of participants and their reading fluency](image)

After the experiment was conducted, the results were gathered and plotted in a scattered column graph, to be able to compare values across categories. The test result, as shown in (Figure 3.4, 3.5, and 3.6), records only the correct answers. The highlighted & centered letters can be seen in Appendix B, 1st example and 2nd example are “مثال 1” and “مثال 2” respectively.

Each category of readers was separated from the others, in order to analyze each group alone.
Figure 3.4: Test result for the 8 occasional readers

Figure 3.5: Test result for the 22 average readers
3.5. Discussion and conclusion

After analyzing at the data, it seems that three categories of readers have consistent results. Therefore, gathering the result all together was possible (Figure 3.7). Afterwards, I was able to identify the focal character under each *wazn*.

![Figure 3.6: Test result for the 10 frequent readers](image)

<table>
<thead>
<tr>
<th>Most accurate focal point</th>
<th>Word Length</th>
<th><em>wazn</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>ُفاعل</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>أفعال</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>تفاعل</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>تفاعل</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>إنفعال</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>افتعال</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>إفعال</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>يستفعال</td>
<td>6</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 3.7: Representation of the *awzan*, and the most accurate focal point.
An analysis of the experiment results is as follows:

I was able to identify the ORP in different word cases: After an analysis of the result, we can identify now the ORP for the most commonly used awzan. And these are shown in Table 3.7.

We can notice that there is somehow a correlation between the word length and the location of the ORP, where 7th and 8th awzan which have 5 letters, the ORP was identified to be the third letter except for the 6th wazn.

We know now that the ORP is not always within the root, because we can notice that for the 4th, 5th, 6th, 7th, and 9th awzan the ORP is located in the root, whereas in the 3rd, 8th and 10th awzan the ORP is not located in the root.

Example: for the 3rd wazn "فعالَ" the ORP is not located in the root of the wazn.

![Graph](image-url)

**Figure 2.8: Result of all participants all together.**

Finally, the experiment did confirm the hypothesis that ORP is located in the center of the word. Moreover, the result of the experiments was a key of the development of the application, where each word was classified among the 10 awzan, and was highlighted and centred following to the obtained results.
4. Software Engineering Methodology

The development of this application implements the methodology of the agile software engineering practical to mobile applications developed by Samsung [6].

The agile software engineering method, is an adaptive and flexible way that presents the following characteristics:

- Iterative, and Incremental; it allows the requirements to be changed or adapted even after the first requirements are approved.
- Focuses on the Quality of Project and Agility.

4.1. Requirements Engineering

This section contains all the functional and non functional requirements for the project Fast Reading App in Arabic.

4.1.1. Specific Requirements

This section of the document lists specific requirements for the project Fast Reading App. Requirements are divided into the following sections:

1. User requirements.
2. Non-Functional requirements.
   a. Interface requirements.

4.1.1.1. User Requirements

a. View reading list
   i. Add
      1. Browse mobile device for pdf, txt or epub and add it to the reading list.
      2. Paste content previously coppied.
3. Add a RSS feed.

ii. Remove.

iii. Remove all.

iv. Read Content.

1. Pause.

2. Rewind.
   a. From last punctuation mark.
   b. From last paragraph.

3. Change font.


5. Change font size.

b. Settings.

   i. Do a test to configure your ideal settings.

   ii. Set Word per minutes default value.

   iii. Set default font.

   iv. Set font size.

   v. Stop at punctuations.

4.1.1.2. Non-Functional requirements.

The Application should guarantee the four freedoms to users as defined any the GNU GPL v3.0 or any compatible License. [See Appendix - A]

4.1.1.2.1. Interface Requirements

The interface should be intuitive and friendly with the small screens dimensions. A UI/UX Mockup Design Section is integrated to this report to give an overview to the Application Design.
4.2. System design

Once the features to implement were decided, a system design is needed to implement the application. This section illustrates the design using UML. The following were the two diagrams that are used to model the application.

- Use Case Diagram
- Class Diagram

4.2.3. Use-case Diagram

In the Unified Modeling Language (UML), the use case diagram is a type of behavioral diagram defined by and created from a use-case analysis. (Figure 4.1) It represents a graphical overview of a set of interaction sequences that a system performs to provide a result of observable or measurable value to one or more actors. [9]

In this application there is only actor – the reader.

Figure 4.1: User-Case Diagram for Fast Reading App in Arabic.
4.2.4. Class diagram

The class diagram below characterizes the important classes in the application with their functions and return types. The diagram also indicates how each class is associated with other classes in the system. (Figure 4.2)

![Class diagram for Fast Reading App in Arabic.](image)

4.3. Design and implementation

The section will supply solutions for the requirements defined above, providing a decomposition of the system into modules. By adopting an incremental model, implementation and testing will go hand in hand to produce an efficient and bug free application. Each module will be implemented alone then tested to finally be integrated into the application.
4.3.1. Application Mockups

In order to have an idea about the user interface and experience before getting to the implementation, I developed mockups for the different screens using Balsamiq Mockups. (Figures 5.1, 5.2, 5.3)

Figure 5.1: Home screen  
Figure 5.2: Reading List screen  
Figure 5.3: The reading screen
4.3.2. Implementation and technology enablers

This section discusses the key aspects of the implementation process such as the development environment used. On the other hand, the implementation of Fast Reading App in Arabic used the results obtained in the experimental section. See Appendix C for the application screenshots.

4.3.2.1. DEVELOPMENT ENVIRONMENT

My primary development platform consisted of the following software and hardware configuration:

- **Android SDK**: The Android SDK provides a set of integrated development and testing tools including core Android libraries, a built in emulator, debugger, logger, and device drivers to allow running applications on an Android phone.

- **Android Studio**: The Android Studio IDE v1.0 allows developing and running apps from within the IDE and provides access to various features of the Android SDK.

- **PC**: Windows 8.1 Pro 64-bit, Java Development Kit 1.8.0, 8 GB RAM, 2.2 GHz Intel Core i7

- **Android Virtual Phone**: Google Nexus 6 with 5.1.0 Android OS.
5. Future Work

The Fast Reading App in Arabic can be improved and upgraded in several ways. Here I explore some of the possibilities and discuss future plans for adding new features that build upon and compliment the current functionalities.

Most of the improvements that can be done are at a technical level. First of all, the Arabic words needs to be displayed as a whole word, for some reasons, Arabic words were sometimes displayed in a discontinuous fashion, that is, letters that should have been joined were not. Hence, further research in Android development needs to be done, in order to come up with possible solutions or libraries to use. On the other hand, this research is based on word recognition, further research need to be done on text comprehension. Furthermore, some of the technical enhancements that can be done are where the user can go directly to a web page and be able to read the page using the application. Another feature that could be added is having a voice recognition system that could voice control the functions of the application. Another great and functional feature is to link this application with Android smart watches, so the user can read emails news and other things on that small screen, which will be very practical.
6. Conclusion

This project has been a way for me to learn methodologies and technologies I was not familiar with. Indeed, I plunged myself into the exciting world of experimental methodologies and tried to apply it in my project. Besides, I learned about Android mobile development and was able to design and implement the desired mobile application.

The application has been designed effectively to meet all the user requirements. However, at this early stage of the project development, it is too early for a public release. Therefore, a further revision of the system implementation needs to be taken care of. Afterwards, it would be wise to release an alpha version of the app to a handful of users, in order to have some user experience feedback and bug reports, before finally releasing it to the general public.
References


10. إسماعيل ع. ج., (2012). معاني الزيادة في الفعل الثلاثي في اللغة العربية: مجلة الجامعات الإسلامية للبحوث الإنسانية, المجلد العشرون. http://resportal.iugaza.edu.ps/articles/%D8%AF.%20%D8%AD%D9%86%D8%A7%D9%86%20%D8%B9%D9%85%D8%A7%D9%8A%D8%B1%D8%A9.pdf
Appendix A: GNU GPLv3

Nobody should be restricted by the software they use. There are four freedoms that every user should have:

- The freedom to use the software for any purpose,
- The freedom to change the software to suit your needs,
- The freedom to share the software with your friends and neighbors, and
- The freedom to share the changes you make.

When a program offers users all of these freedoms, we call it free software.

Developers who write software can release it under the terms of the GNU GPL. When they do, it will be free software and stay free software, no matter who changes or distributes the program. We call this copyleft: the software is copyrighted, but instead of using those rights to restrict users like proprietary software does, we use them to ensure that every user has freedom.
### Appendix B: Test 2: List of most common *awzan*

<table>
<thead>
<tr>
<th>مثال 2</th>
<th>مثال 1</th>
<th>wasn</th>
</tr>
</thead>
<tbody>
<tr>
<td>فأَسْتَفَعَلْ</td>
<td>عاَمَلْ</td>
<td>3</td>
</tr>
<tr>
<td>أَفْعَلْ</td>
<td>عاَمَلْ</td>
<td>4</td>
</tr>
<tr>
<td>تَفَعَّلْ</td>
<td>تَحَمَّسْ</td>
<td>5</td>
</tr>
<tr>
<td>تَفَعَّلْ</td>
<td>تَقَابَلْ</td>
<td>6</td>
</tr>
<tr>
<td>اِنْفَعَلْ</td>
<td>اِنْضَرَّبَ</td>
<td>7</td>
</tr>
<tr>
<td>اِسْتَفْعَلْ</td>
<td>اِسْتَفْعَلْ</td>
<td>8</td>
</tr>
<tr>
<td>اِسْتَفْعَلْ</td>
<td>اِسْتَفْعَلْ</td>
<td>9</td>
</tr>
<tr>
<td>اِسْتَفْعَلْ</td>
<td>اِسْتَفْعَلْ</td>
<td>10</td>
</tr>
</tbody>
</table>
Appendix C: Screenshots from application

Home screen

Settings Screen
Reading Screen

Open File Screen