**Final Report**

**Group Upsilon**

**Introduction.**

Students often get tired of memorizing words when trying to learn a new language. Teachers also get tired of forcing students to memorize and tests. We thought students can enjoy learning foreign language by playing game and we produced Sudoku Vocabulary game to help users learn foreign languages. The usual way of memorizing new words would be using que cards or writing down multiple times, however, with Sudoku Vocabulary game, users can learn in a fun and effective way by solving word puzzles.

Sudoku Vocabulary game consists pairs of words such as English words on the grid cells and Korean words on the buttons for the input, instead of regular Sudoku format; numbers from one to nine. The app also includes features to service both teachers and various ages of students.

We developed the Android app with Java Android Studio IDE. The application’s code is organized well into distinct classes according to model-view-controller design pattern.

**Project description and operation.**

The project applied agile development because the project must be completed within a semester which is less than four months. These four months were divided into four iterations and each iteration we got new requirements to implement.

The requirement for the first iteration was setting up the GitLab repository with group members, implementing a playable Sudoku Vocabulary game, and providing documents that analyse user requirements in the form of user stories and executable test-driven development examples.

There were several features added in the second iteration. The grid cells fit into different devices, apply flexible display on portrait and landscape modes, import csv vocabulary lists, add listening comprehension mode, add tracking vocabulary words that users having difficulty recognizing, populate the grid algorithmically, and set up the unit test. Finally, the application classes had to be refactored according to MVC design pattern.

The third iteration involved with adding different size Sudoku grids, for example 4x4, 6x6, and 12x12. We also made drastic change to the design of the Grid cells and added an array that detects horizontal, vertical, and regional locations from onClickListener.

The final iteration was free of will for adding new features. Therefore, we decided to add a timer function and We were also required to write a project report, and peer evaluations.

**Software engineering issues discussion.**

First of all, this was the first group project for us. In our personal project or assignments, we don’t need to take care of other things except algorithms. However, in group project almost everything that wasn’t a problem be a problem. For example, each of us use different naming rules for variables and none of us use comments to make easier for other members to understand. We should define rules and communicate consistently to connect each other’s work.

We couldn’t figure the basic algorithm for Sudoku at first. We googled the basic principle of generating and checking Sudoku. The most commonly know algorithm was for square region sudoku so we should figure out by ourselves for 6X6 and 12X12 modes.

When we write the code, we focused on the implementation rather than the planning and evaluation to meet the deadline. This led MVC refactoring difficult to do because multiple patterns for example control-view, or control-model in a single class. Furthermore, some requirements in late iterations conflict with our existing codes. We should add and delete codes to meet requirements and this experience was a great opportunity for agile project experience. For example, for iteration2 we add mp3 files to make pronunciation and then iteration3 asked us to import words that users want. That means we should use other way to make sounds instead of put in every mp3 files. This caused us to change our program much and we used TextToSpeech API to solve this problem.

**Concluding remarks and future work.**

The project demo was encouraging to watch since we learned how other students did with the same project ideas. It was impressing to watch because each group had a unique app, friendly user interface, and no one failed their project. In future, we wish to add some other functions, for example, timer function or add algorithm that keeps tracking of difficulty words and show them in the next game.

We gained a lot of knowledge and experience in software development in this course. We learned not only about android but also programming methods and other background knowledge needed for developing a program. We are pretty sure that this experience will help us to work well in the future.

The thing is we have is that it was our first time working with Java object-oriented programing, so the learning curve was stiff. Also, we only have 3 members and that makes us be in difficulty compare to other groups. We hope this point would be considered for marking.