## 

## Edenz logo_for monitor(RGB)_Large

**Diploma in Software Development**

DSE 760 Cloud and Web Software Development

Assessment: Project Report  
Total marks: 100  
Course Weighting: 100%

Due Date: Friday, 9th November at 5:00 p.m., 2018

**Student Name(s):……Michael Braverman……………………………………**

# EXECUTIVE SUMMARY

I created a tool to support foreign language learners. It will help people to save new foreign words together at one place with translations, and learn them through a simple game.

I decided to write this software because it may be useful for people. I have been learning English for about 2 years and have not found yet a simple application that can save new words and help to learn them. There are a lot in the internet, but they are too complicated or not good enough. I took as a model Android App “My Vocabulary” that does not exist in Google Play with some improvements.

I used Microsoft technologies for development such as ASP and MS SQL Server because they are extremely popular in New Zealand.

# Introduction

## Background

*Describe why you chose this application and what you were planning to achieve.*

My decision of choosing of dev technologies is based on job market’s demand.

I started to look for a job directly after arrival and found out that Microsoft Technologies prevail in the market. During job interviews with representatives of different HR companies such as [Enterprise](https://www.enterprise.co.nz/) and [Potentia](http://potentia.co.nz/), I was informed that MS ASP is the most popular skill in the job market. When I looked for a job through <https://www.seek.co.nz/> I also figured out that in New Zealand between 60 and 80% of the all jobs are .NET jobs. Most of these jobs require knowledge in ASP.NET. Most in-demand DB technology (about 80%) is MS SQL Server. I did not make exact calculation and this is my subjection perception of the situation. Together with information from HR agencies I believe that choosing these technologies it is a right decision.

I created my application with following technologies:

1. ASP.NET core Razor Page
   1. 60-80% of all jobs in New Zealand are in ASP.NET MVC
   2. Even though companies still use ASP Web Forms, ASP MVC or even ASP classic, ASP.NET core is the most perspective technology for Web.
2. MS SQL Server as a data storage
   1. Although I dealt last years with Mongo Db, I decided to work with MS Server to polish up my SQL skills.
   2. Entity Framework Core I used as object-relational mapper (O/RM)

My goal was to learn newest cutting-edge Microsoft web development technology to prepare myself better to NZ local job market.

## Scope

*Give a short overview of what you have included and what extensions are possible later.*

Application is available to the registered users only. The application is a dynamic dictionary with option to play memory game. You can create unlimited number of independent dictionaries. Every dictionary is the *List*. The List keeps Records, where every record is a foreign word with translation. User can create some lists with a number of records. For comfortable and not too long Memory Game, a list should keep not more than 100 records, but it is up to the user. The Memory Game initiates for every List. User should select the list with more than five Records in the List. The goal is to train the memory of user by showing him a foreign word with three option of translations, where one is correct and two are wrong. User will win if he guesses right all records in the selected list. User can make not more than two mistakes. With a third wrong selection the game will be over. Application will show how many records are left, what is your record in all games you did, what is your current score and how many lives you have.

Through your configuration, you can choose the game direction: from foreign language or from your native language. If you choose from foreign language – you will see the foreign word and three options for translations. If you select from native language – you will see translation and three options for foreign word.

List of feature

What app does?

No technology. What not how.

# Project Planning and Execution

## Project Plan and Gantt Chart

*Show all the tasks you carried out, in what order and when*

1. Understanding what do I want to do
2. Creation of User Stories
3. Creation of Use Cases
4. Learning ASP Razor Page technology from [Microsoft Docs](https://docs.microsoft.com/en-us/aspnet/core/tutorials/razor-pages/?view=aspnetcore-2.1) together with building my project structure
   1. Building of Models: how my DB structure will look like
   2. Generation of Data Base by Entity Framework Core
   3. Creation of appropriate Razor Pages
   4. Implementation of Authentication and Registration with default MS authentication model
   5. Implementation of Email validation
   6. Making web pages available only by authenticated user
   7. Implementation of the game
   8. Configuration development
5. Bugs fixing
6. UI improvements
7. Code refactoring
8. Code cleaning up
9. Report creation

## Risk Management

1. Even though I consider myself as a experienced developer, I started my project very early with plenty of time in advance to be able to finish it slowly, quietly without rush.
2. I planned to fully finish my development 2 weeks before deadline to have more time for report
3. One student spoil his water close to my PC and I thought it is a good idea to keep your code somewhere in a cloud to have it saved in the case of disaster. For instance, hardware breakdown. To prevent loosing of all code, I used [GitHub](https://github.com/) source control. The [code and documentation](https://github.com/mickeybrave/MemoryGame) is available online.
4. Another risk is to be able implement all possible technologies. For instance, I wanted to implement all code with server-side C#. However, when my development was almost done, I faced some fatal error: “HTTP Error 502.3 - Bad Gateway”. After a day of hitting head against a brick wall, I decided to change technology. Solution from internet did not work. Consequently, I rewrote all business logic in JavaScript and removed redundant C# code.

I learnt that it is robust idea to start development ASAP, make balance between quality and speed and leave a “dead-end” to consider another solution for a stubborn problem.

*List what risks you anticipated and describe what you learned about managing risks to your project*

# Requirements Management

I thought about this software for quite a lot time, because I used a couple of similar application and learnt about their benefits and flaws. For instance, if you do not separate your words to lists, your memory game will be too long. From my experience more than 15 minutes it is too long, consequently, words should be separated to independent lists about 100 records length.

I used two approaches to gather requirements:

1. Use case

|  |  |  |
| --- | --- | --- |
| Actor | Wants | Reason |
| user | Play memorization game | To learn foreign words |
| user | Create list of words and translations | In order I can memorize foreign words |
| user | View the list of words and translations | To see how many words in the list |
| user | Add and remove words and translations to and from the list | To allow the game |
| User | Navigate between created list | To see all lists, add words to different lists, initiate the game with particular list |
| user | See the number of words in the top of list | To help me keep the list short |
| user | Change game direction settings (from Russian to English or from English to Russian) | From native to foreign will improve my speaking, from foreign to native improves my reading. |
| user | See a word along with 3 translation and to choose the correct one | To test my knowledge and memory |
| User | See label “correct” | To have indication that user is right |
| User | See label “wrong” | To have indication that user is wrong |
| User | See label with indication how many words are left | To see approximately how long he needs to play to finish the game |
| User | See the best score/record | To aspire for improvements |

1. User story

*Discuss your understanding of issues relating to managing software requirements. Describe any approaches or techniques you used for establishing and managing requirements in your project.*

# User Experience Design (UX)

*Discuss your understanding of issues relating to user experience in web applications. Describe any UX approaches or techniques you used in your project.*

# Architecture and Design

## Content structure

*Describe what you learned about how to structure your web content.*

## Navigation

*Describe what you learned about helping the user to navigate your app.*

## Functionality

*Describe the decisions you made about the functionality you will offer. For example, can the user work off-line; how much processing takes place on the server and how much in the client; how do you handle large quantities of data?*

## Styling

*Discuss your approach to decisions on styling.*

## Accessibility

*Discuss your understanding about accessibility and explain how this relates to your application.*

## Security

*Discuss the issues relating to web security and explain how this might affect your application.*

## Reflection

*Explain what you learned from your experience of designing web apps*

# Implementation

## Discussion of technologies used

*Discuss the technologies used in your application along with alternatives considered and your reasons for final choices.*

Limitation: not enough support, very new technology s

## Database schema

*Show the database schema used*

## Screen shots

*Screen shots of working app*

## Reflection

My Application is ASP .NET Core Razor Page web site that works online only. However, I want to extend it in further courses (Windows and Mobile). I want to

*Explain what you learned from your experience of implementing web apps*

# Cloud Deployment

## Discussion of cloud technologies

*Describe and compare architectures and infrastructures for three cloud platforms.*

## Deployment

*Discuss the process of deploying an application to the cloud and your experiences with this process.*

# Testing

## Test planning

*Provide a table with test cases and expected results*

## Test results

*Discuss your test results and describe any bugs remaining in your app*

## Reflection

*Explain what you learned about testing web apps*

# Conclusion

## Reflection

*Explain what you learned during this project. You should aim to identify all problems and how these affected your project. What would you do differently next time?*

## Summary

*Provide a summary of your project. This should tie up with the Executive Summary at the beginning of the report.*

# References

*You must use the APA referencing system*