Test Report – Milestone 4

Team: charlie

Presented by: Michael Le, Debashis Jena, Austin Johnson, Prince Antwi Aboagye, Didimus Kimbi, Damion Sevilla

SWEN 670 – sOFTWARE eNGINEERING pROJECT

August 06, 2021

reVision 1.0

Project name: Mnemosyne, Disability Mobile Application

Date: August 06, 2021

Project Leader: Michael Le

Phase: Design & Engineering and Execution

For approval: Michael Le

Michael le Date: 08/06/2021

For approval: Dr. Mir Mohammed Assadullah

Date: 08/06/2021

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version Number | Date | Description | Approved By |
| 1.0 | 08/06/2021 | Test Report | Michael Le |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[**1. Introduction** 4](#_Toc78957729)

[1.1 Purpose 4](#_Toc78957730)

[1.2 Application Overview 4](#_Toc78957731)

[1.3 Technical Project Stakeholders 5](#_Toc78957732)

[1.4 Testing Limitations 5](#_Toc78957733)

[**2. Type of Testing** 6](#_Toc78957734)

[2.1 Unit testing 6](#_Toc78957735)

[2.2 Integration testing 6](#_Toc78957736)

[2.3 Regression Testing 6](#_Toc78957737)

[2.4 End-to-end System Testing 7](#_Toc78957738)

[2.5 Nonfunctional Testing 7](#_Toc78957739)

[**3 Items not Tested** 8](#_Toc78957740)

[**4. Metrics and Testing** 9](#_Toc78957741)

[**5. Defect Identified** 10](#_Toc78957742)

[**6. Exit Criteria** 11](#_Toc78957743)

[**7 Website Performance Report** 12](#_Toc78957744)

[**8 Conclusion/Sign Off** 13](#_Toc78957745)

# **1. Introduction**

## 1.1 Purpose

The purpose of this document is to show the Quality Assurance and Control the functionalities requirements for Mnemosyne mobile application. The application aims to present the testing activities for each of the application. This test report goal is to show the audiences that Mnemosyne application are working based on the fundamental testing metric and test cases execution. Overall, this testing document delve as follow: Mnemosyne application overview, testing strategy, testing limitations, type of test, test cases, defect, and metrics to present the tests performed.

## 1.2 Application Overview

1. Dr. Mir Assadullah created the Short-Term Memory Assistant mobile application for the UMGC Software Engineering Project course Capstone Project in the summer 2021 term. The Charlie team will be working with DevSecOps Team to complete the overall Capstone Project. Each team will have similar general requirements, and the business analyst will develop a high-level requirement to envision the team for a unique design/approach mobile application solution.
2. The Capstone Project’s customers include disabled people from dementia to Alzheimer’s. The public service professionals, such as doctors, medical analysts, etc., would benefit from the service using this memory-impaired mobile application.
3. The main activities of the Mnemosyne mobile application are to shows mainly the functionalities such as record speech, save/retain text, voice recognition training, record speech of user only, visual options, searching text, training videos, and send notification to users.

## 1.3 Technical Project Stakeholders

The testing strategy in this document delve around mainly test case feature functionalities and aim to present the results of the Mnemosyne application. In addition, the test development includes the following activities:

* Reviewing and analyzing the Software Requirements and the Project Plan
* Development of scenarios and required testing techniques.
* Define acceptance criteria to meet the accepted standard
* Construction of test case which defined by the use case input/output
* Writing and creation of test scripts
* Running tests and documenting the results of each test
* Review of all testing documentation

## 1.4 Testing Limitations

# **2. Type of Testing**

There are various types of testing that the team Charlie went through as part of the project implementation. In the following section the types of testing will be elaborated.

## 2.1 Unit testing

Unit tests are written to help the developers to ensure each of the components are working as designed. In case of this project, the unit tests are also required to be executed as part of the continuous integration and continuous delivery (CICD) process. Typically, the unit tests cover three quarter of the code. However, because of the lack of time, the planned code coverage could not be achieved. Tests were written to verify the text encryption/decryption and to test if the texts are identified as scheduled texts.

## 2.2 Integration testing

In case of Mnemosyne application, there are many components were integrated with each other. As part of the normal system testing, they were verified. Only external system in this application is Google Cloud’s speech-to-text API. The API connection was tested thoroughly with use of the service account credential JSON, by passing a static audio file and a streaming input.

## 2.3 Regression Testing

The project went through 8 execution sprints. The regression testing is performed to ensure the existing features within the application is working after the new releases. Typically, the regression testing for larger projects is done using an automated testing tool like selenium or protractor. However, in case of Mnemosyne the regression testing was done manually. At the end of each sprint, when the weekly release becomes available, the application goes through a round of regression testing. The regression test suites are nothing but the test cases that were executed in the last release.

## 2.4 End-to-end System Testing

The project adopted the scrum method of agile testing. So, after every release the application was expected to work end-to-end with the defined set of features. Keeping that in mind, the test cases were written for the sprint. Each of the use cases were executed keeping the usability in mind. The app was uninstalled and installed again for almost all tests that involved user’s voice training. For the existing notes, the use cases were executed with the installed app.

## 2.5 Nonfunctional Testing

The application went through nonfunctional testing. Usually, the nonfunctional testing cover both performance and load testing. The performance testing in this project was purely manual. In the initial releases a few performance issues were identified, and they were addressed in the next releases. As Mnemosyne is a standalone application with minimum external communication apart from the Google’s speech-to-text, there wasn’t any load testing performed.

# **3 Items not Tested**

The following items are not in scope to be tested by Team Bravo:

* Network Security
* App Store/Google Play deployment
* Load and stress testing

# **4. Metrics and Testing**

# **5. Defect Identified**

# **6. Exit Criteria**

# **7 Website Performance Report**

# **8 Conclusion/Sign Off**