**Project Name:** Short-Term Memory Assistant

**Date:** May 25, 2021

**Project Number:** UMGC6702021

**Project Manager:** Malcolm Freeney

**Vision Statement:**

The vision for United Global Masters Coders is to make the best personal speech recognition application for the memory impaired.

**Business Need/Project Objectives:**

About 14 million people in the United States have a disability that impairs their short-term memory. Short-term memory (STM) indicates different systems of memory involved in the retention of pieces of data for a relatively short time (usually up to 30 seconds). The causes of the disability range from aging related dementia, to Alzheimer’s disease. According to the U.S. Centers for Disease Control and Prevention, noticing a decline in your mental abilities ("cognitive decline") is one of the earliest signs of impending Alzheimer's disease or dementia. There is a need to provide a solution to adults that assists them with the short-term memory function. To meet this need, United Global Masters Coders proposes the development of a mobile application that recognizes the speech of the user as they engage in a conversation and help them live a fuller life despite their disability.

**Project Description and How it Meets the Business Need:**

The purpose of this project is to develop the Short-Term Memory Assistant for people with short-term memory disabilities. There are several ways to develop this app and this effort is to determine which way would be the most optimal. The Short-Term Memory Assistant meets the business need by providing a solution for a critical and underserved market segment to have a solution with them that meets their need for memory assistance.

**Project Benefits:**

1. Implementing a mobile application provides convenience to the disabled person when going about the daily lives. The disabled user can remember important conversations and ideas because the application resides on their smartphone.

**Project Requirements:**

1. The application should **not** save any voice recording.
2. The application should **allow** the user to edit any speech converted to text.
3. The application should provide a user **interface that is 508** accessible and that an elderly person should be able to use it without eyeglasses.
4. The application should keep the end user persona in mind for **UI/UX considerations** and offer a flexible environment for the user to **customize.**
5. The application should provide **training videos within** the app to guide the user regarding its various functionalities.
6. The application should recognize that every person has a unique personality and a **unique set of phrases and sentences** that he or she uses while speaking to him or herself or with others. The app's purpose is to make the process unobtrusive and fit naturally for the user.
7. The application should **learn the phrases** that the user wants to use when talking to someone while trying to save important spoken text and also the phrases that the user wants to use speaking to him or herself trying to retrieve the noted information.
8. The application should provide a facility to the user. This facility will note **several phrases that the user may use in natural conversation** to **save** as well as to **retrieve** saved information to play as speech.
9. The application should **ignore everything except what the user speaks**.
10. The application should provide the means for the user to **train the app on its voice**.
11. The application should **bypass** asking everyone **permission** to **record**.

**Project Deliverables:**

1. Project Plan
2. Software Requirements Specification (SRS)
3. Technical Design Document
4. Development Teams: Add Software Test Plan to the Project Plan
5. DevSecOps Team: Deployment and Operations Guide (Runbook)
6. Development Teams: Programmer Guide
7. Development Teams: Deployment and Operations Guide (Runbook)
8. DevSecOps Team: Add Software Test Plan to the Project Plan
9. DevSecOps Team: User Guide
10. Development Teams: User Guide
11. All Teams: Test Report
12. DevSecOps Team: Programmer Guide
13. Peer Evaluations

**Success / Acceptance Criteria:**

1. TBD (This section will change). It is important to define a uniform, objective, and broad criteria for success. The PM shall work with the designated teams to come up with a commonly agreed upon scientific unbiased criteria to evaluate multiple aspects of speech to text, NLP, text to speech, and UI/UX so that in the end we can compare the approaches taken by the teams and assess the approaches. This criterion is essential to determine which team's approach performed better in addressing the problem.

**Estimated Project Schedule:**

Milestone One - Project Initiation **(Due 6/8/21**)

* Project Plan
* Software Requirements Specification (SRS)

Milestone Two (**Due 6/29/21**)

* All Teams: Technical Design Document
* Development Teams: Add Software Test Plan to the Project Plan
* DevSecOps Team: Deployment and Operations Guide (Runbook)

Milestone Three (**Due 7/20/21**)

* Development Teams: Programmer Guide
* Development Teams: Deployment and Operations Guide (Runbook)
* DevSecOps Team: Add Software Test Plan to the Project Plan
* DevSecOps Team: User Guide

Milestone Four (**Due 8/10/21**)

Deliverables for this milestone should be defined in the Project Plan. This milestone must include the finished software.

Project Constraints:

1. The application must be designed using **Flutter**.
2. There isn’t a **monetary** budget for this project.

The words from professor.

There are multiple ways to create this app and this effort is to determine which way would be the best. For instance, we can use **speech to text** and **text to speech** tools from multiple vendors: AWS, Google, Azure, etc. We could also use non-cloud, library-based tools such as **speech recognition** by Dragon or IBM Via Voice. There are some text to speech libraries, likewise, that we can use. Then there is a piece of NLP, which understands the text, that could be cloud or library based. Likewise, there may be some ideas around **UI/UX of the app**. Which of these combinations would work best and which user interface would work best? That is the purpose of this exercise.

I want the teams to be free to choose their technological and UI/UX approaches and be creative. I want you, Malcolm, to work with the teams to come up with a commonly agreed upon scientific unbiased criteria to evaluate multiple aspects of speech to text, NLP, text to speech, and UI/UX so that in the end we can compare the approaches taken by the teams and assess the approaches (not the teams). Next semester, I would like to assign the class one direction to work on. This semester I want to know what that direction is going to be.