Project Plan

Date: September 21, 2018

*NPR Telegram Bot*

Team #2

Yoseph Hailemariam

Erikton Konomi

Sarah Obi

Andrew Williams

# Introduction

This product will harness Telegram, a messaging app with built-in functionality that supports the usage and creation of bots. Our team will create a bot that will allow users to interact with the nationally syndicated public broadcast radio station, NPR (the National Public Radio). Users will be able to use the bot to automate the process of finding a podcast or program from this station based on their interests. The bot interface will assist users in searching for these programs and downloading them on their device for offline access.

# Business needs/requirements

Users (regular listeners of NPR) will require the bot to be able to download podcasts or programs for offline access. Functionality should include the organization of audio files after they have been saved onto storage. An important consideration will be the size of the audio file. It cannot be too large and should include options to optimize file size, with or without affecting audio quality.

In terms of business needs, our interface should allow for companies to subscribe to the NPR service so that their employees will be granted unlimited access i.e. (medical establishments would subscribe to the service to allow their employees to listen to programs that discuss current issues and topics within the medical field).

# Product or solution overview

The interface should allow for users to choose which file type to save their program as i.e. (.mp3, .mp4, .mov, etc.). In addition, a search feature will allow users to find programs and podcasts based on criteria such as title, topic of the broadcast, host(s) involved, guest(s), and date aired. To solve the issue of file size, our solution will be to optimize/compress the audio file upon download. The type of device being used should not affect functionality. Computers, mobile devices, and tablets will all be supported.

# major Features

* Allows user to search the NPR database for podcasts using a variety of fields (name, host name, date)
* Allows users to further request information from NPR servers using wildcards such as: /latest, /today, /<date>
* The bot responds to the user’s inquiries by presenting a list of relevant matches in the form of web links that the user can interact with.
* Upon choosing a result, the desired podcast would be downloaded in the bot message window, allowing the user to listen to the file from their device.

# Project Process description

For this project, we will be implementing Agile methodologies, specifically Kanban. Due to the nature of our project, we have several tasks that can be completed concurrently. Kanban allows for all team members to work in tandem before proceeding to the next. Our application’s behavior is determined by the user’s actions; a dynamic process modeling will be used.

**Requirement Analysis**

Identify, evaluate, and categorize the end goals of the application as must-have, would-like-to-have, or nice-to-have.

*Entrance criteria*: Target audience, Goals, Resources and Time Limitations

*Exit criteria*: Goal priorities and Risk Analysis

**System Design**

Identify all individual components needed for the application to function per design. Create a system diagram / flowchart that shows the interactions between entities at a high-level.

*Entrance criteria*: Software Requirements, Platform Dependencies, Communication Protocols needed

*Exit criteria*: Mockup of final product, Documentation for overall system operation

**Application Design**

Based on findings from previous step, determine the programing language(s) and technologies to be used. For each of the sub-systems above, draft a possible implementation using the determined language, leveraging embedded features and/or existing frameworks. Also assign development priorities for each building block.

*Entrance criteria*: Sub-systems, their functionality, as well as their interdependence (flowchart)

*Exit criteria*: Skeleton code for application that includes overall structure (e.g. libraries, entry points etc.)

**Application Implementation**

Following the priorities from the prior phase, write code for each block to fulfill its purpose. Unit-test the code as it is being developed.

*Entrance criteria*: Prioritized sub-system blocks

*Exit criteria*: Fully coded and Unit-tested functional sub-system blocks

**Application Testing**

Test the application on a system-level using common methodologies such as black-box testing and user-acceptance testing. Identify bugs and fix them.

*Entrance criteria*: Fully-Developed application

*Exit criteria*: Identified and fixed bugs not encountered during unit-testing

**Application Deployment**

Deploy application on Telegram Messenger and ask for user feedback. This step is inherent in our project from the early phases.

*Entrance criteria*: Completed Application

*Exit criteria*: Satisfied Users

# Project Schedule

| **Date (**YYYY-MM-DD) | **Milestone/ Event** | **Entry Deliverable & Criteria** | **Exit / Notes** |
| --- | --- | --- | --- |
| 2018-09-21 | Project Plan Baselined | Inspection completed | In progress |
| 2018-10-02 | Requirement Analysis | Software requirement and analysis | Requirements Specification |
| 2018-10-09 | System Design | Subsystem identification and establishment of overall system flow | System Design Document and System Flowchart |
| 2018-10-23 | Application Design | Research and analysis of technologies to be used for development | Application Framework and Dependencies |
| 2018-11-27 | Application Implementation | Coding and development of individual sub-systems and unit testing | Working Prototype |
| 2018-12-04 | Application Testing | System testing, evaluation and bug fixing | Application Tested and Ready for Deployment |
| 2018-12-04 | Application Deployment | Release of application and gathering of user feedback if any | Application meets re |

# Project Work and Product Estimates

* Effort Hours per team-member = 10 (hours per week) x 10 (weeks) = 100
* Lines of Code = 3,000

# Team

**Yoseph Hailemariam**

Programmer, Application Design, Server Setup

Critical skills: Communication, Development, Testing

**Erikton Konomi**

Team Leader, Designer: oversees team communication and project progress; works on system architecture, application design and infrastructure.

Critical skills: Communication, Leadership, Framework Design, Infrastructure

**Sarah Obi**

Team Member: gathers the requirements and determines which process model is appropriate for the project

Critical skills: Planning, Eliciting Requirements

**Andrew Williams**

Team member: provides general application and team support, reviews and revises any data to follow more clear, concise, more presentable and business formal language standards

Critical skills: Technical Writing and Language Skills, Organizational skills, Coding

# Scope and limitations

The main limitation this project faces is the availability of NPR servers, upon which our up-time is dependant. A solution that is currently out of our scope would be an interface that stores top programs and/or programs aired within the past 24-48 hours directly on our server, thus improving the user's overall experience. The current scope of this project will be a free service. Optimally we would like to implement a subscription service for businesses and individuals to allow for unlimited access to program downloads, but as of now this will be listed as a limitation. In terms of the scope of this project cycle, allowing users to choose which type of audio file they would like to download their program as will serve as another limitation.