Software Project Management Plan

Introduction

1.1 Project Overview

Create a website that allows anonymous users to schedule text-based reminders, serving as an alternative to an ineffective stock-phone reminders app.

1.2 Project Deliverables

React-based text reminders website (Delivered by August 1st, 2020)

1.3 Evolution of the Software Project Management Plan

Version	Message	Date
Draft	Initial Commit	July 1st, 2020

1.4 Reference Materials

Document Template — http://users.csc.calpoly.edu/~jdalbey/205/Mgmt/SPMP

1.5 Definitions and Acronyms

React — *A JavaScript library for building user interfaces*

Project Organization

2.1 Process Model

Major Work Products	Due	Approved
SPMP	July 1st, 2020	No
Requirements	July 1st, 2021	No
User Stories	July 1st, 2022	No

2.2 Organizational Structure

All project responsibility and internal management is delegated to Garrett Partenza

2.3 Project Responsibilities

Worker	Responsible	Analyst
Garrett Partenza	Design, Implementation, Testing, Deployment, and all documentation there of.	Professor Broadwater

Technical Process

3.1 Methods, Tools, and Techniques

Computing Environment - *Linux*
Software Tools - *Node.js, Reactjs, Python, EC2 Instance, and Firebase Database.*
Development Methodologies - *Agile Methodology (Due to this being an individual project, an inherent freedom arises that fits well with an agile development methodology).*

3.2 Software Documentation

- SPMP
- Requirements
- User Stories
- TBD

3.3 Technical Pipeline

1. ReactJS web-app takes in information regarding an anonymous users reminder, and ships it to Firebase DB.
2. An elastic compute instance will monitor this database for incoming reminders.
3. When a reminder is detected by the elastic compute instance, a python script will generate a cron job to send schedule the reminder using a package called “send mail through python” at the designated reminder time.

Tools Used Above — ReactJS (NodeJS and Nginx), Firebase DB, EC2, Python, CronJob.

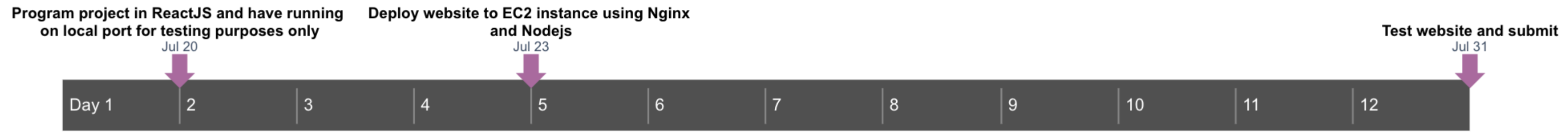
4.1 Managing Change

A set of backup plans have been developed in case of needing to change the project structure.

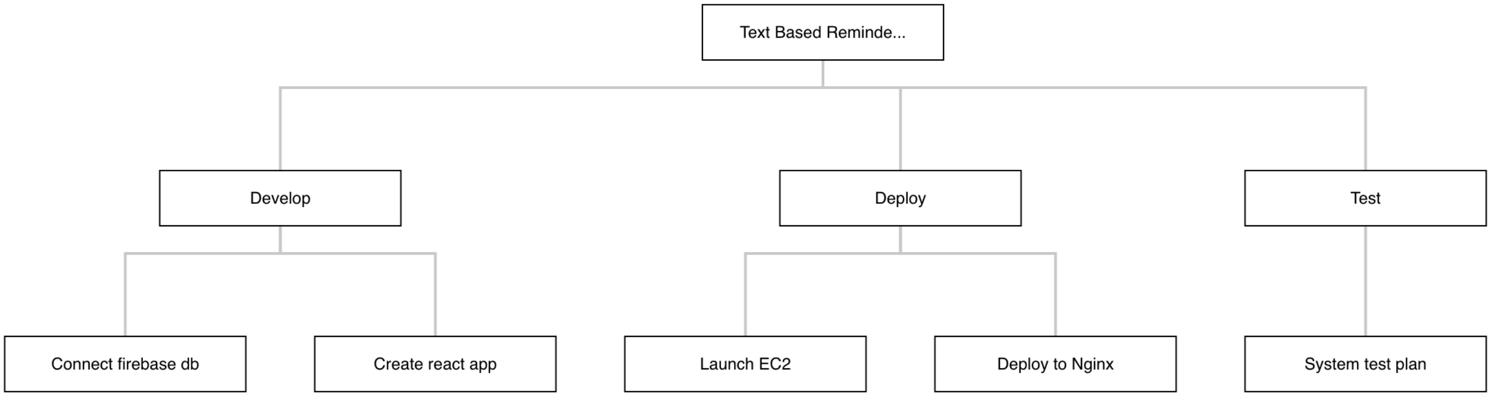
Current plan: React -> Firebase -> CronJob
Plan B: React -> AWS S3 -> CronJob
Plan C: React -> AWS S3 -> AWS Lambda
Plan D: React -> AWS S3 -> CronJob

Due to this being an individual project, all change can be handled as long as the Gant Chart and WPB deadlines are met below. SPMP, UserRoles and Requirements documents will be updated according to changes.

4.1 Gant Chart Schedule



4.2 WPB


Text Based Reminde...
Develop
Deploy
Test
Connect firebase db
Create react app
Launch EC2
Deploy to Nginx
System test plan