Get It Done - Final Report

By

Phoning It In

Cesar Santiago

Joseph Quinn

Mason Riley

Sean McMillon

Capstone Project [Spring, 2020]

The University of West Florida

4/17/2020

CIS4595 Capstone Systems Project

Dr. Bernd Owsnicki-Klewe

Executive Summary

Get It Done is an event planning application for Android devices, which will allow users to plan events with tasks that the user associates as needed for completion for the event to be possible. Currently due to not having an AWS database multiple users can not access the same event. This will be covered further down in the report. For now, SQLite is being used as proof of concept.

Get It Done was developed for use on Android devices. The team developed this project using API 19 (KitKat) on Android Studio. The code for the application is stored in and pulled from the group's GitHub repository. New branches were produced for each new method that will be developed. JUnit will be used to conduct unit tests for methods. The database is being kept in SQLite. The next step for the database is to implement an AWS database and do away with the SQLite database.

Table of Contents

1 Final Requirements and Comparison

1.1 Final Requirements

1.1.1 Main Components

1. Android Device

2. Database to hold user credentials, events, and tasks associated with this event

1.1.2 Component Interaction

The client android device will connect to an SQLite database that will store the events and tasks associated with the event. When a new event is created, the database is updated. Every time an event is opened, the information about the event is refreshed from the database. If the information within the event is changed, the database receives the updated information.

1.2 Comparison with Initial Requirements

We were unable to allow users to create and log into unique accounts to show only their events. Due to costs and time constraints, we were unable to implement the AWS server with the database.

2 Final Timeline and Comparison With the Initial Timeline

We stuck with the two-sprint partition of the agile framework. We had planned to change to a three sprint partition however that plan did not occur once the complication of COVID-19 occurred. Some of the dates were thrown back a few days due to the due date for the final submission being pulled back.

2.1 Original Timeline

Project Plan: Jan/31
Sprint 1 Planning: Feb/1 - Feb/3
Sprint 1: Feb/04 - Mar/02
2nd Presentation: Mar/03
Sprint 2 Planning: Mar/04 - Mar/07
Spring Break: Mar/08 - Mar/13
Sprint 2: Mar/14 - Apr/13
Final Presentation and deliverables Submission: Apr/21

2.2 Current Timeline

Project Plan: Jan/31
Sprint 1 Planning: Feb/01 - Feb/03
Sprint 1: Feb/04 - Feb/25
Review 1: Feb/26 - Feb/29
2nd Presentation: Mar/03
Sprint 2 Planning: Mar/04 - Mar/07
Spring Break: Mar/08 - Mar/13
Sprint 2: Mar/13 - Mar/24
Review 2: Mar/25 - Apr/13
Final Presentation and deliverables: Apr/24

3 Project Results Compared with Expectations

We expected to have a fully completed project complete with the basic “must-have” features and one or two extra “should have” features. Instead, due to COVID-19 and its associated delays, we were forced to focus on completing as many basic features as possible. As a result, we have a very primitive project lacking most features rather than one with a firm foundation and a few bells and whistles.

4 Project Process Review

Throughout the project, we strived to follow the Scrum methodology. We successfully executed two sprints. We opted for weekly stand-ups rather than daily ones, which we feel were effective. There was a difficulty increase in team communication caused by COVID-19. The team initially created a Trello board to track the progress of the project but was neglected to be updated throughout development. In spite of this, we feel that we made good progress.

5 Work to be Done

1. AWS server implementation.
2. User account creation.
3. Ability for a user to log into their account.
4. Ability to share an event with another user.

5. Ability to mark a task as completed.

6 Acknowledgements

This work was conducted as part of the Spring 2020 Capstone Project Systems CIS4595 class conducted by Dr. Bernd Oswinicki-Klewe. The development team is; Cesar Santiago, Joseph Quinn, Mason Riley, and Sean McMillon.