Open Source Software Platform

Chris Corrado

Zhonghua Bao

Anirudh Bezzam

Ziyu Zhang

**Business Platform Team, February 12th, 2019**

**INTRODUCTION**

This project is meant to be a platform for developers to find and independently choose the tools necessary to build out operational services for local businesses. MVP of the platform will include these Microservices:

● Inventory Management

● Core Services and Identity Management

● Order Management

The Software Platform is a way for local businesses to register with a central authority ”the platform” and sell their merchandise to registered customers by delivering the products to user-inputted addresses. Consumers of the platform can browse local businesses that are registered and select products to be delivered to a location of the user’s choosing. This is different from other in market competitors in that the cost of delivery and operation is spread fairly amongst the consumers with each order. The software platform will support business administration of merchandise and information and consumer registration.

**ROLES AND RESPONSIBILITIES**

These vary for each type of product and, for small projects, folks may serve multiple roles. This is a list of common roles we have used for software development:

Development Lead Chris Corrado

Buildmeister (In charge of weekly builds) Das

Architect Bao

Developers All

Test Lead Sam

Testers All

Documentation All

Documentation Editor Das

Designer Chris Corrado

User advocate Das

Risk Management Sam

Modification Request Board All (2 reviewers per change)

Requirements Resource Bao

# **METHOD**

These are unique to software development, although there may be some overlap.

* Software:
  + Frontend
    - Android Kotlin
  + API
    - NodeJS/Javascript
  + Database
    - Java/JVM + MySQL
  + Microservice(s)
    - Python + Flask
  + Coding conventions and contributing documents in github <https://github.com/CCorrado/ssw690ossmgmt/wiki/Contributing>
  + Coding style and conventions should be maintained in each technology stack.
* Review Process:
  + Every change will be reviewed by all of the team members.
  + Every change will require a minimum of 2 approvals to merge.
  + Every change will execute and pass existing unit tests and generate coverage metrics.
  + Every change must have a passing build via CI/CD automation.
  + Every change will trigger a code quality report.
* Build Plan:
  + Github Source Code Management
  + Automation (CI/CD) performed by TravisCI integrated with the Github Repository
  + Builds will deploy daily
  + Test suite will run on every build

# 

# **Virtual and Real Work Space**

**Github** Wiki, Issues, and repository as a single source for all documentation and source code.

Google Docs as a collaboration tool

# **COMMUNICATION PLAN (required)**

## **“Heartbeat” meetings**

Every week on Sunday

## **Status meetings**

Status Meetings will be held every week on Tuesday night in person at class

## **Issues meetings**

Ad-hoc up to the team’s discretion.

Communication channels should be leveraged to bring up a new issue:

* Github
* Meet
* E-Mail

# **TIMELINE AND MILESTONES**

For a full backlog of all issues: <https://github.com/CCorrado/ssw690ossmgmt/issues>

For a most recent list of project milestones and what issues are inside each milestone: <https://github.com/CCorrado/ssw690ossmgmt/milestones>

Note that for this project we have a few time boxes. They are:

· Week of February 12th – Architecture Review

· Week of February 26th first milestone, description of second milestone

· Week of March 12th second milestone, description of third milestone

· Week of March 26th third milestone, description of fourth milestone

· Week of April 9th fourth milestone, description of final product

· Week of April 30th final product

**TESTING POLICY/PLAN**

Testing of every service should be automated, with some manual testing required towards the end of each release. See the Testing overview on the working agreement here: <https://github.com/CCorrado/ssw690ossmgmt/wiki/Working-Agreement>

# **RISKS**

There are a few risks involved:

* The twelve week timeline is tight to deliver four production-ready platforms given the skill level of the overall team.
* Lower risk, but the platform may be delivered and become obsolete if existing services solve the problems that we intend to solve (inventory management issues, costs and fees to the business for using other services)

# **ASSUMPTIONS**

Assumptions are that this platform will be delivered as a working MVP as part of the SSW 690 class. At the end of the semester, the platform should be able to live on and scale as needed to local businesses.

Documentation and Build Plan

For Contributing, Working agreement, and how build deployments are managed, see the project’s wiki page here:

<https://github.com/CCorrado/ssw690ossmgmt/wiki>