Software Project Management Lab 2

UberRUSH

January 28, 2019

Mohamed Ibrahim 100626201
Amalnnath Parameswaran 100585138
Mohtasim Siddiqui 100635463

Description:

E-commerce is a growing market in modern society. In a world where companies such as Amazon and Ebay exist, consumers are given more ways to add convenience into their daily shopping needs. However, not all retail stores are able to provide an online presence for their customers. Small businesses and niche markets struggle with the capital needed to set up an online presence. This is where UberRUSH is able to bridge the gap between the small businesses and the customers.

UberRUSH will be able to streamline the delivery process for the small businesses, the RUSH drivers and the consumer. The consumer will be able to browse the items available at any participating stores. Once the transaction has been completed, the retail staff will be able to package and prepare the products for delivery. The package is then handed off to a contractor and with tracking capabilities within the app, would deliver the package on time and reliably.

UberRUSH helps small businesses to move to the 21st Century, while also providing jobs to many contract drivers. It adds convenience to the consumer and to the business owners. We can also see UberRUSH expanding to larger businesses, who would prefer to prioritize their same day delivery services.

Objectives:

1. Outsource product delivery for small businesses.
2. Build two mobile applications:
    a. One for businesses and clients to use.
        i. Mobile app should allow businesses to track deliveries and get confirmation on received packages.
        ii. Mobile app should allow clients to request products for delivery from different businesses.
    b. One for RUSH drivers to get notifications of delivery requests, provide driving instructions, and map routes for delivery.
3. Implement a web client application that combines the functionality of both mobile applications.
    a. The web app will have 3 separate logins:
        i. One for clients looking to order products.
        ii. One for businesses to add their products and track deliveries.
        iii. One for RUSH drivers to get notifications of delivery requests, provide driving instructions, and map routes for delivery.
4. Allow clients to review products provided by partner business to maintain quality of service.
5. Match requests with drivers in close proximity to ensure efficient delivery times.

Measures of Success.

1. When businesses want to deliver a package, they send a notification to the RUSH drivers. This request is uploaded to the cloud and all nearby drivers are notified.
2. When a RUSH driver picks up the request, the driver will be prompted with directions to the business location. Upon pickup, the driver will be prompted with directions to the customer.
3. When a RUSH driver picks up a package, the application will assign a tracking number to monitor the status of delivery.
4. Each package has a tracking ID attached. Upon delivery, the tracking ID is removed which sends delivery confirmation to businesses and their clients.
5. GPS locations of RUSH drivers are tracked to show how close they are to delivery. This information can be observed by both the businesses and the clients.

Hardware and software infrastructure

1. Cross Platform mobile application that works on Android and iOS devices.
    a. Two different options available:
        i. Develop native applications using Swift for iOS and Java for Android.
        ii. Develop a cross platform solution using React Native or Flutter.
2. Web application hosted on a remote server.
    a. Web client should be developed using single page technologies like React, Angular, or Vue for increased performance and user experience.
    b. Options for hosting are:
        i. Amazon EC2
        ii. Google Cloud
        iii. Microsoft Azure
        iv. Heroku
        v. Firebase
3. Database server hosted remotely for storing data about clients, businesses, and RUSH drivers.
    a. Options:
        i. Amazon DynamoDB
        ii. MongoDB
        iii. PostgreSQL
4. Server application that provides apis for the web and mobile clients.
    a. Options:
        i. ExpressJS
        ii. Ruby on Rails
        iii. Django
    b. API can be restful or using GraphQL