

MICHIGAN STATE

U N I V E R S I T Y

Project Plan

Asynchronous Service Desk Callback App

The Capstone Experience

Team GM

Sean Rabaut
Corbin Rangler
Michael Palmer
Evan Hlavaty
Brian Hart

Department of Computer Science and Engineering
Michigan State University

Fall 2015



*From Students...
...to Professionals*

Functional Specifications

- The main focus is to reduce the amount of time General Motors employees waste on hold with Global Service Desk
- GM employees submit requests using native iOS application
- Allow employees to better utilize time waiting
- Global Service Desk employees will use the web application to work on these requests

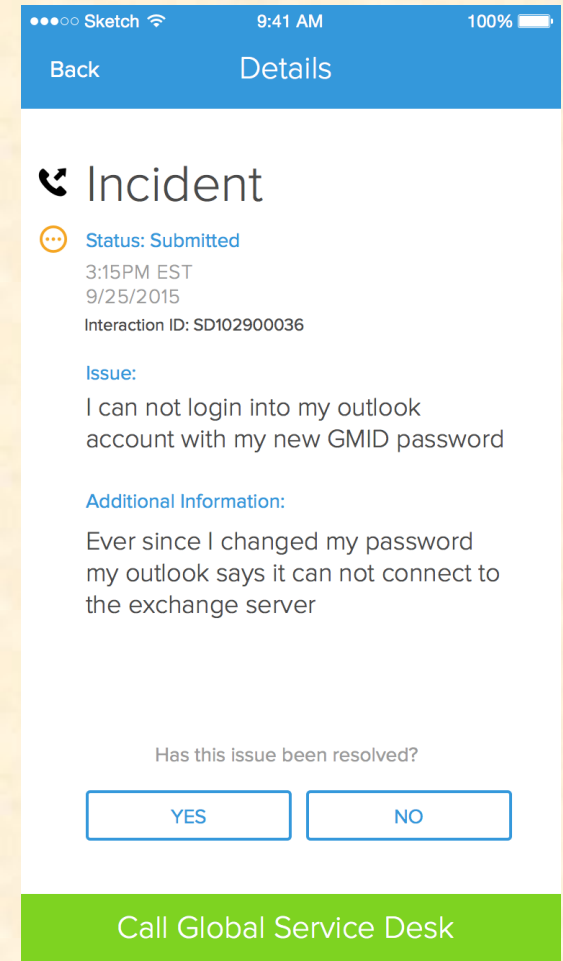
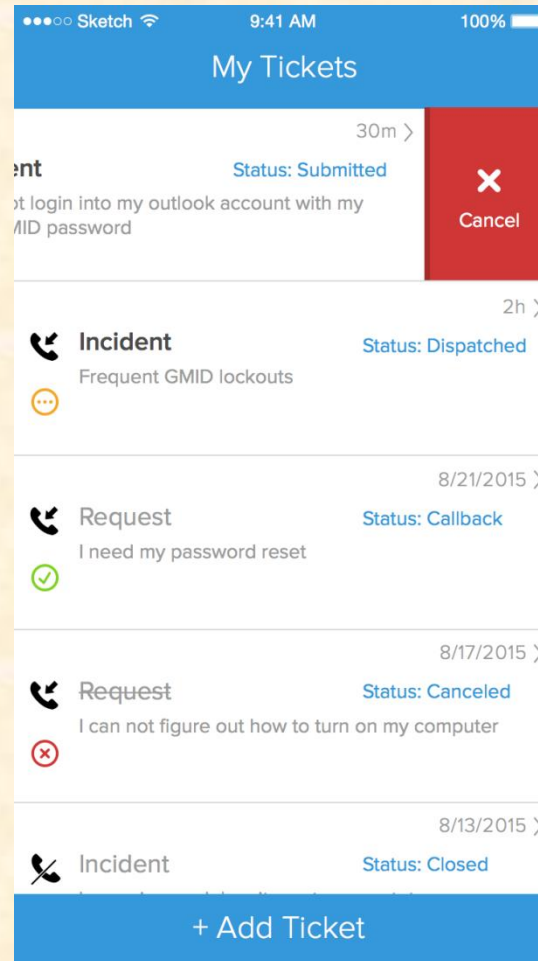
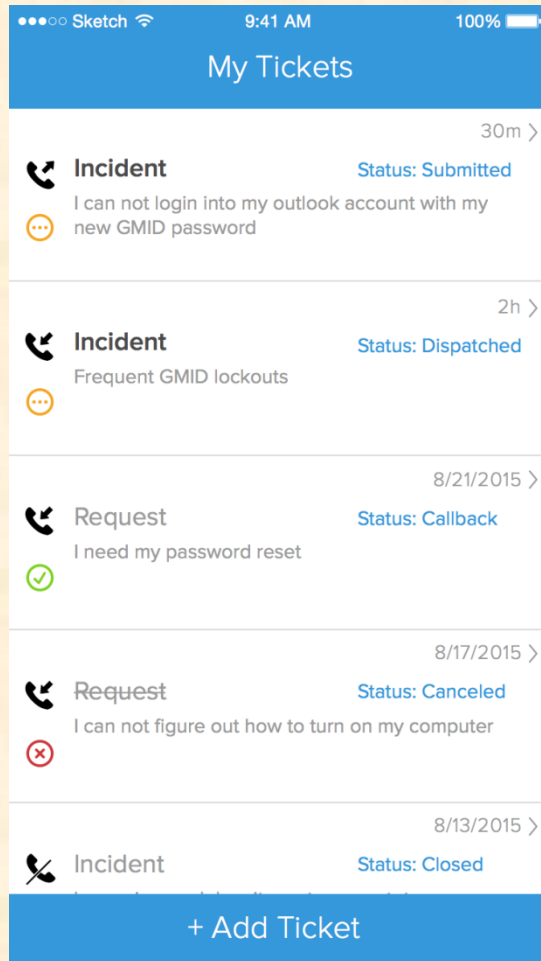


Design Specifications

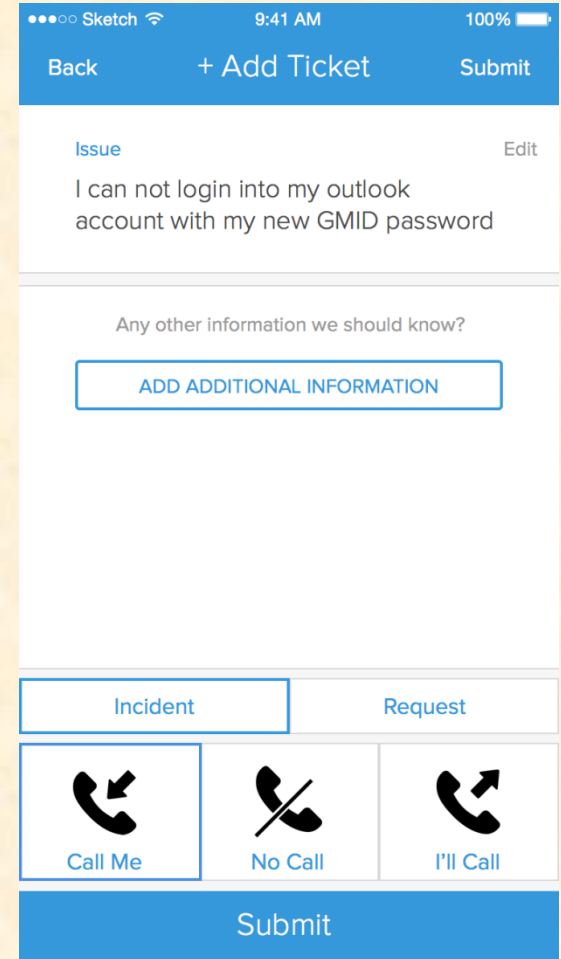
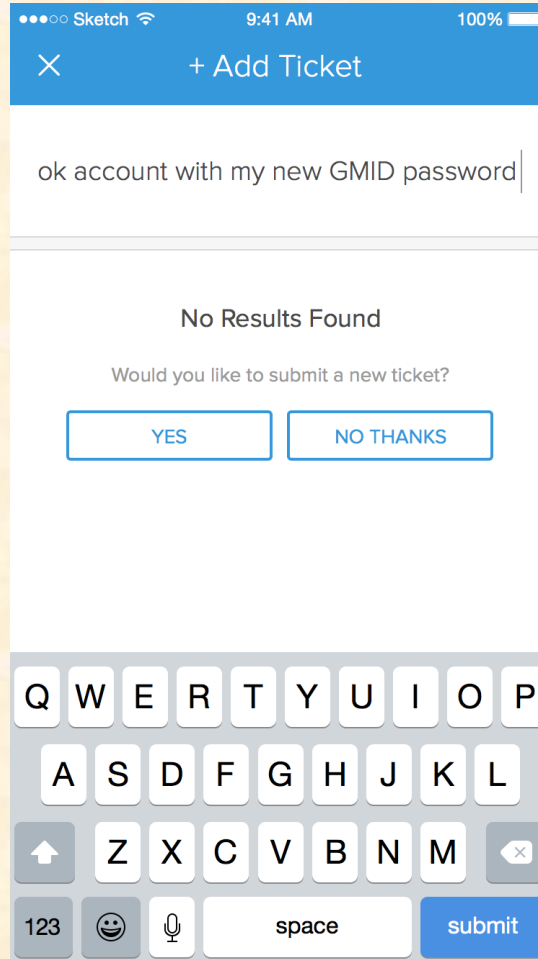
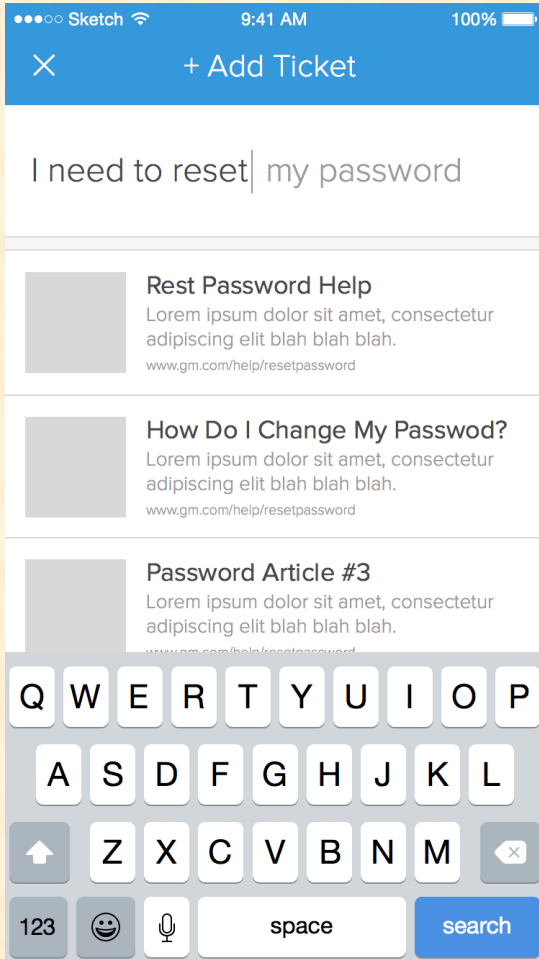
- Submit help tickets
- View submitted help tickets
 - Specific details
 - Status of ticket
 - Ability to cancel ticket
- Query information to help you solve your issue
- Links to help articles and step by step wizards



Screen Mockup: Ticket List/Details



Screen Mockup: Search/Submit Ticket



Screen Mockup: Web Service Manager

Two Tabs –

- Global Ticket Queue List
- Personal Ticket Queue List

Ticket Details –

- Name
- Phone
- Email
- Category
- Issue
- Description
- Interaction ID
- Timestamp

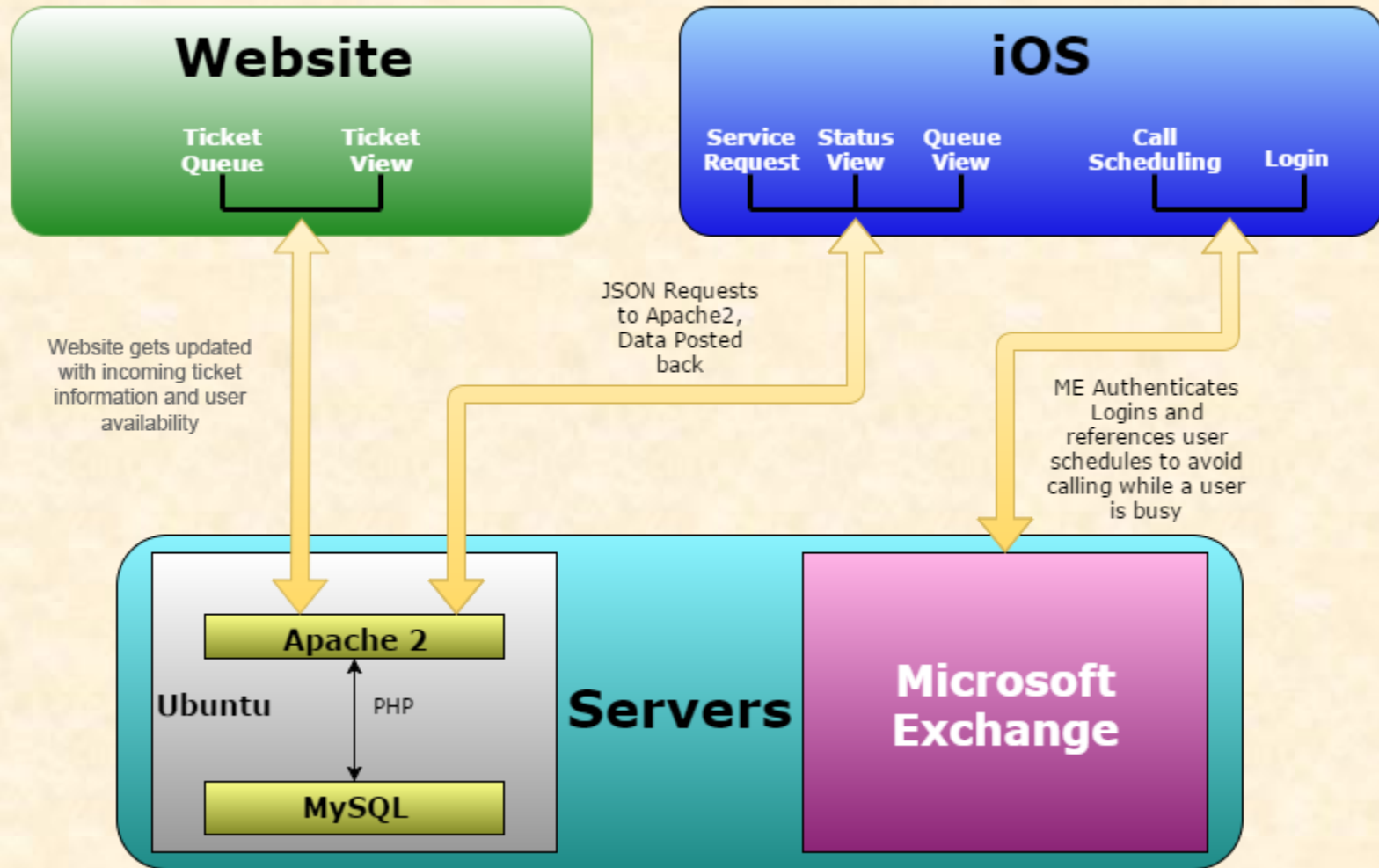


Technical Specifications

- The application and website communicate with an Apache2 webserver to sync with the MySQL database, as seen in the next slide
- The application uses the Microsoft Exchange server to authenticate users and upload schedules
- The front-end of our application uses Swift 2 to perform on the user end, and JSON to communicate with the webserver



System Architecture



System Components

- Hardware Platforms
 - Server Rack
 - iOS Device
- Software Platforms / Technologies
 - Windows Server 2012 with Hyper-V
 - Ubuntu 14.04.3 with Apache2 with SSL, MySQL, phpMyAdmin, OpenSSH
 - Microsoft Exchange 2013
 - Xcode v6.4
 - PhpStorm v8.0.3



Testing

- Web/API Testing
 - Codeception testing framework
 - Behavior Driven Development (BDD) style suites
 - Unit and API tests
 - Selenium for frontend testing
- iOS Testing
 - Apple's XCTest framework
 - Unit and Performance tests
 - Manual testing on various devices



Risks

- Integration into GM's system
 - Need to implement solution with access to GM databases and user accounts.
 - Mitigation: Generalize our solution, keep consistent documentation.
- Microsoft Exchange integration
 - Need to integrate with user's Exchange schedule.
 - Mitigation: Time management, researching API's
- Code Consistency
 - Need to keep our code consistent between all members, as well as have backups in case of emergency.
 - Mitigation: Utilization of code sourcing tools such as Github or SourceTree
- Feature creep
 - GM wants our app to support various features not essential to the core function.
 - Mitigation: Consulting with client, time management

