

# 09/13: Team Status Reports

## The Capstone Experience

Dr. Wayne Dyksen

Department of Computer Science and Engineering  
Michigan State University

Fall 2017



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

# Team Amazon

## Status Report

[1 of 4]

### Faia: Fashion Artificial Intelligence Assistant

- Project Overview
  - Virtual Personal Stylist that adapts to shopper preferences
  - Makes clothing suggestions using Machine Learning
  - Text-based communication with user
  - Integrates with Amazon Marketplace services
- Project Plan Document
  - Several sections in progress
  - Completed early sections
  - Still needs images and diagrams
  - Clarified specifications for the project plan with the Amazon clients



# Team Amazon

## Status Report

[2 of 4]

### Faia: Fashion Artificial Intelligence Assistant

- Server Systems / Software
  - Hosting Web Interface on EC2
  - RDS Database System
- Development Systems / Software
  - Web Interface with HTML, Javascript, PHP on Visual Studio
  - Amazon Lex interprets user input for Faia
  - Twilio/Facebook Messenger
  - AWS Machine Learning



# Team Amazon

## Status Report

[3 of 4]

### Faia: Fashion Artificial Intelligence Assistant

- Client Contact
  - Weekly meetings on Tuesdays
  - Draft meeting on Friday
- Team Meetings
  - Weekly meetings on Tuesdays
  - Triage meetings on Thursdays
- Team Organization
  - Client Contact: Danielle
  - Program Manager: David, Machine Learning: Zizhen and Nikhil, Web: Danielle and David, Chatbot: Dominic



# Team Amazon

## Status Report

[4 of 4]

### Faia: Fashion Artificial Intelligence Assistant Risks

- API
  - Description: Integrating all of the APIs together
  - Mitigation: Develop prototype with all APIs
- User Experience
  - Description: Creating a smooth user experience through SMS
  - Mitigation: Utilize embedded HTML over SMS
- Machine Learning
  - Description: Difficult to get good results from machine learning
  - Mitigation: Experiment with multiple machine learning models



# Team Auto-Owners

## Status Report

[1 of 4]

### House of Hazards

- Project Overview
  - Teach employees about home safety
  - Time-based virtual reality game
  - Using an Oculus Rift
  - Various scenarios inside & outside home
- Project Plan Document
  - Project plan skeleton done
  - Finished project plan presentation slides
  - Haven't started writing core of plan
  - Roughly halfway done



# Team Auto-Owners

## Status Report

[2 of 4]

### House of Hazards

- Server Systems / Software
  - Github, repository set up
- Development Systems / Software
  - Unity, installed on all machines, tutorials done
  - Oculus Rift
  - Oculus Rift Touch Controllers
  - Oculus Rift Sensor
  - Oculus & peripherals need to be tested in Unity



# Team Auto-Owners

## Status Report

[3 of 4]

### House of Hazards

- Client Contact
  - Met with client 9/8/17 in person
  - Scheduled meetings for Fridays at 3PM
- Team Meetings
  - Regularly schedule team meetings, whole team meeting at least once a week, Wednesdays after All Hands Meetings or Fridays after Triage Meetings.
  - Met as a team 4 times so far
- Team Organization
  - Brian, Ken, & Kevin assigned to Scripting & Testing
  - Matt & Fred assigned to Modeling & Asset Management





# Team Auto-Owners

## Status Report

[4 of 4]

### House of Hazards

#### Risks

- Risk 1
  - Lack of HDMI port for Oculus
  - Obtain an adapter
- Risk 2
  - Oculus compatibility with MacOS
  - Use Windows VM
- Risk 3
  - Adapting prefab scripts with current Unity version
  - Code reviews of scripts
- Risk 4
  - No available Unity assets for some objects
  - Model assets



# Team Avata

## Status Report

[1 of 4]

### Security Analytics Suite: Configuration Setup Tool

- Project Overview
  - Create new clients, add/edit roles, modules, and shift information
  - Create campuses, beats, and sub-beats
  - Create client specific crime taxonomy
- Project Plan Document
  - Skeleton has been created, with cover page and table of contents
  - Contains UI mockups and use cases
  - Architecture diagram is done
  - 15% completed



# Team Avata

## Status Report

[2 of 4]

### Security Analytics Suite: Configuration Setup Tool

- Server Systems / Software
  - AWS Server is up and hosting backend
  - Client will be providing a MsSQL server in the future
  - Front-end has no server, looking at doing AWS
- Development Systems / Software
  - Two separate GIT repositories for front and back end
  - Jira board has stories for early project requirements
  - Slack is set up for communication. Jira is linked to Slack and provides logging information for story status updates
  - IntelliJ IDEA being used for development



# Team Avata

## Status Report

[3 of 4]

### Security Analytics Suite: Configuration Setup Tool

- Client Contact
  - Have met thrice with client via Google Hangouts
  - Weekly Meeting: Every Tuesday 5pm-6pm
- Team Meetings
  - Have met in person thrice
  - Weekly Meeting: Every Tuesday 4:30pm-6:30pm for sprint planning
- Team Organization
  - Sean – PM                      Meenu – Front-end Lead
  - Ashley – Back-end Lead              Zack – QA              Chantz – SysAd



# Team Avata

## Status Report

[4 of 4]

### Security Analytics Suite: Configuration Setup Tool Risks

- ArcGIS API
  - Algorithm for finding center of polygon and preventing collisions
  - Will create prototype using less complicated shapes (lines)
- Separate front and back ends
  - Never hosted front end on different server, normally bundle with back end
  - Research and use AWS server to test
- Data Hierarchy
  - Data structure for graphic campus/beat/sub-beat and crime/group/categories
  - Speak with client about the relationships between data points



# Team Ford

## Status Report

[1 of 4]

### Ford Smart Park App

- Project Overview
  - Send notifications when available spots are in your area
  - Give reward incentives for users to scan open parking spots
- Project Plan Document
  - Some information recorded, not yet documented
  - 10% complete

# Team Ford

## Status Report

[2 of 4]

### Ford Smart Park App

- Server Systems / Software
  - Windows Server 2016 Datacenter, Not Installed
  - SQL Server, Not Installed
- Development Systems / Software
  - Android Studio 2.3.3, Installed
  - Google Tango SDK, Installed
  - Sync 3 Emulator, Installed



# Team Ford

## Status Report

[3 of 4]

### Ford Smart Park App

- Client Contact
  - Weekly Meetings on Fridays, at 1:30pm
  - Met twice via WebEx, in-person meeting today
- Team Meetings
  - Weekly Meetings on Wednesdays at 4:30pm
  - Met seven times in person
- Team Organization
  - Eric, Doug, and Rahul working on Mobile App
  - Helena, and Cheng working on Google Tango





# Team Ford

## Status Report

[4 of 4]

### Ford Smart Park App

#### Risks

- Google Tango Implementation
  - No proof of parking spot identification
  - Using SDK to manipulate Google Tango source code
- Server Installation Process
  - Backend server is not set up
  - Seeking help from faculty and look into alternatives (AWS)



# Team GM

## Status Report

[1 of 4]

### Workplace Safety System

- Project Overview
  - Ensure GM factory workers are using correct Personal Protective Equipment (PPE)
  - Use image/object recognition and machine learning to recognize missing PPEs
  - Alert the area safety manager via SMS when issues arise
  - Provide web reporting of safety incidents to help identify patterns of concern to management
- Project Plan Document
  - Template is laid out
  - 10% completed
  - Expect to complete project plan by the next team meeting



# Team GM

## Status Report

[2 of 4]

### Workplace Safety System

- Server Systems / Software
  - Amazon Web Services (AWS) – plan to train model using AWS
- Development Systems / Software
  - NVidia Deep Learning SDK – Going through Deep Learning Institute lectures and tutorials
  - NVidia Jetson Tx2 – Receiving them today
  - NVidia Deep Learning GPU Training System (DIGITS) – Need to setup on Jetsons



# Team GM

## Status Report

[3 of 4]

### Workplace Safety System

- Client Contact
  - Weekly meetings with client every Wednesday from 9 a.m. – 10 a.m.
  - Plan to meet in person later in September with an on site plant tour
  - Damien Hong will be the student point of contact
- Team Meetings
  - Official team meetings held every Sunday from 3 p.m.
  - Five team meetings have been held
- Team Organization
  - Assigned roles for each part of the project
    - Steve – DIGITS/Web services
    - Michael – AWS
    - Damien – Training models
    - Ike – DIGITS
    - Marc – Web services/Mobile services



# Team GM

## Status Report

[4 of 4]

### Workplace Safety System

#### Risks

- Training model to accurately recognize PPEs
  - Difficulty: Hard
  - Description: Training the model is straightforward, however we don't know how to effectively train a model with high accuracy.
  - Mitigation: Consult Machine Learning experts at NVidia, consult Deep Learning Institute kit, research tips and tricks online
- Emulating factory environment
  - Difficulty: Medium
  - Description: We don't know whether or not the environment will have an impact on model performance.
  - Mitigation: Model test environment to those shown in images. This includes adjusting lighting, wearing actual PPEs, and or the use of props.
- Choosing the right Deep Learning framework
  - Difficulty: Medium
  - Description: There are many Deep Learning frameworks available that DIGITS can utilize. Choosing the right one may have an impact on how well our model works.
  - Mitigation: Consult NVidia, deep learning experts, look through Deep Learning Institute kit
- Implementing mobile notifications
  - Difficulty: Easy
  - Description: Most developers on the team have never implemented mobile push notifications
  - Mitigation: Research SMS APIs, create prototype to receive notifications



# Team Humana

## Status Report

[1 of 4]

### MyHumanaBot

- Project Overview
  - Automated chat service
  - Quickly responds to member questions
  - Humana information and specific member information
  - Alternative to phone call or emailing customer service
- Project Plan Document
  - Outline made
  - Executive Summary written
  - Draft of function specs written and critiqued by client
  - 20-25% complete



# Team Humana

## Status Report

[2 of 4]

### MyHumanaBot

- Server Systems / Software
  - Microsoft Azure cloud hosting – Set up but need license for bot hosting
  - SQL Server – Architecture plan, but not set up
  - API.ai – Set up and proof of concept experimentation
- Development Systems / Software
  - Bot Emulator – Set up and tested
  - Microsoft Bot Framework – “Hello World” type bot
  - C# / .NET Core – Initial MVC set up



# Team Humana

## Status Report

[3 of 4]

### MyHumanaBot

- Client Contact
  - Call every Tuesday at 3pm
  - In person meeting Wednesday, September 20
- Team Meetings
  - Whole group meeting 5 times so far
  - Plan to meet twice a week for remainder of project
- Team Organization
  - Madeline-Project Manager, Tynan-Backend Lead, Tony-Frontend Lead, Jason-Security Lead, Sharon-API.ai Lead
  - Slack, Trello, CSE GitLab





# Team Humana

## Status Report

[4 of 4]

### MyHumanaBot

#### Risks

- Microsoft Azure Bot Services
  - Not sure if we will be able to host here for free
  - Emailing to check about student licenses
- Database
  - Unsure about how to host database and access the database tables from the Bot Framework and admin app
  - Research and watch tutorials, contact people with experience working with databases in .NET
- Machine Learning
  - Not sure if built in machine learning will work for our limited amount of data
  - Experiment and make more data if needed



# Team Meijer

## Status Report

[1 of 4]

### Fresh-ipes

- Project Overview

- Simplify shopping experience using Amazon Echo Show or mobile application
- Determine current ingredients based on shopping history
- Integrate with Yummly API to find and suggest recipes
- Create shopping list of missing ingredients

- Project Plan Document

- Skeleton with basic information
- Individual portions complete
- Initial mockups and system architecture started
- Sent to client for review



# Team Meijer

## Status Report

[2 of 4]

### Fresh-ipes

- Server Systems / Software
  - Microsoft Azure – access to the account
  - Microsoft SQL Server – need confirmation
- Development Systems / Software
  - Alexa Skills – access to dev account
  - Yummly API – access to dev account
  - Microsoft .NET (C#, ASP.NET) – VS installed and connected to VSTS



# Team Meijer

## Status Report

[3 of 4]

### Fresh-ipes

- Client Contact
  - Initial call and high-level architecture explained
  - Scheduled calls Wednesdays before class
  - Email each day if questions arise
- Team Meetings
  - Meet almost daily during the week
  - Team meetings Tuesdays after triage
- Team Organization
  - Mobile/Xamarin: Charley
  - Web: Dan and Olive
  - Alexa Skill/Echo: James
  - SQL/Backend: Justin



# Team Meijer

## Status Report

[4 of 4]

### Fresh-ipes

#### Risks

- Risk 1
  - How to integrate shopping history data
  - Create/gain access to database or parse with regex
- Risk 2
  - Sync state across mobile and Amazon Echo Show
  - Configure web app for mobile and Echo separately then integrate
- Risk 3
  - Integration with mPerks
  - Work closely with Meijer and ask a lot of questions



# Team Michigan State University

## Status Report

[1 of 4]

### Spartan Experience App

- Project Overview
  - Developing the official campus mobile application for Michigan State University
  - Using iOS and Android
  - Target audience is current and prospective MSU students, as well as visitors
- Project Plan Document
  - We have divided sections of the project plan for each team member, and begun writing our project plan document.
  - Functional Specs: 70% done
  - Screen Mockups: 75% done
  - Design Specs: 25% done
  - System Architecture Diagram: 100% done
  - Technical Specs: 30% done
  - Schedule and Testing Plan: 30% done
  - Risks: 90% done



# Team Michigan State University

## Status Report

[2 of 4]

### Spartan Experience App

- Server Systems / Software
  - AWS access is complete
  - Python AWS Lambda environment is running
  - PostgreSQL Database is running
- Development Systems / Software
  - GitLab, Trello, Slack collaboration set up
  - VMware Fusion 8 and Windows 10 VM Set up
  - Android Studio 3.0, Xcode 9 installed and tested
  - Python 3.6 installed and tested



# Team Michigan State University

## Status Report

[3 of 4]

### Spartan Experience App

- Client Contact
  - Met with EJ, Tyler, and Spencer on Tuesday, September 5<sup>th</sup>
  - Met with Tyler and Spencer on Friday, September 8<sup>th</sup>
  - Scheduled a weekly in-person meeting every Friday
- Team Meetings
  - Our team has met four times (08/31, 09/05, 09/06, 09/10)
  - We have scheduled team meetings twice a week, every Sunday and Wednesday evenings
- Team Organization
  - Client Contact: Patrick Pale
  - Project Manager: Nayana Kodur
  - Android Lead: Roy Perryman
  - Back-End Lead: Scott Swarhout
  - iOS Lead: Ryan Johnson
  - Organize work and communicate through Slack and Trello





# Team Michigan State University

## Status Report

[4 of 4]

### Spartan Experience App

#### Risks

- Risk 1
  - Access between separate university services
  - Take an inventory of which services are feasible to use and prioritize them
- Risk 2
  - No well-defined list for context-sensitive information categories.
  - Present a list of possible categories to our client, and prioritize them.
- Risk 3
  - RHS just updated the dining hall website, so we do not know whether or the not the RSS Feeds still exist
  - If RSS feeds to not exist, we will have to implement web-scraping techniques.
- Risk 4
  - No team experience working with beacon push notifications
  - Find example applications and implement a simple proof-of-concept application utilizing the technology.



# Team Microsoft

## Status Report

[1 of 4]

### Enhanced Company Portal with Graph

- Project Overview
  - Android App
  - Mobile Device Management
  - Microsoft Graph API
  - Social Integration and IT Help Desk Chat
- Project Plan Document
  - Status: Finishing Touches
  - Risks Identified
  - Specifications identified
  - Mockups made
  - 85% complete



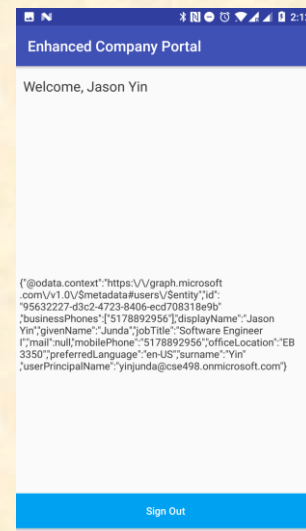
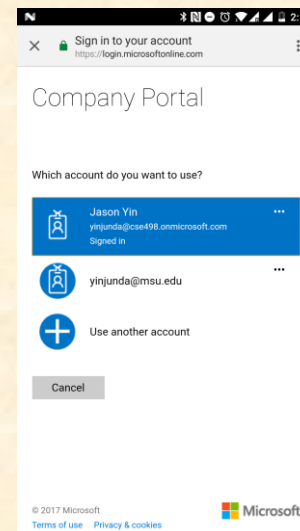
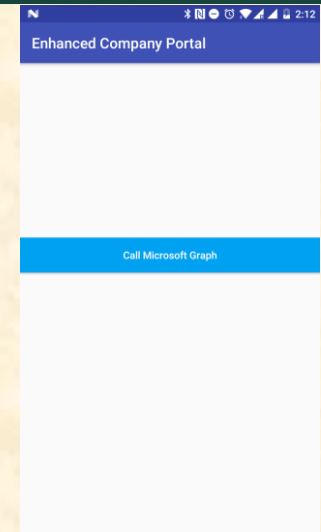
# Team Microsoft

## Status Report

[2 of 4]

### Enhanced Company Portal with Graph

- Server Systems / Software
  - No On Premises Servers, Azure Setup
  - Enrollment with Microsoft Active Directory
  - Accounts with Microsoft Graph and Microsoft Intune
- Development Systems / Software
  - Android Studio installed
  - Physical Android Device tested
  - Skeleton App started
  - In-App MS Account Authentication tested
  - API Request tested



# Team Microsoft

## Status Report

[3 of 4]

### Enhanced Company Portal with Graph

- Client Contact
  - Weekly calls Monday and Wednesdays at 9:45AM
  - 3 Team Calls made
- Team Meetings
  - 6 Team meetings
  - Meetings before all hands
  - Meet briefly on Tuesday/Thursday
- Team Organization
  - Structure
    - 1 Project Manager / Business Logic Developer
    - 2 App Designers
    - 2 Developers
  - OneDrive, Trello, Slack



# Team Microsoft

## Status Report

[4 of 4]

### Enhanced Company Portal with Graph

#### Risks

- Microsoft Graph API (Hard)
  - Microsoft Graph REST API for all data consumption. Graph contains a vast amount of API's, must identify necessary API's
  - Review Graph API documentation to identify necessary API's. Design data models for JSON.
- Social Integration (Hard)
  - The team will identify features to incorporate in the app to drive application usage.
  - Review Graph API documentation to identify API's to use
- Identify Account Level (Moderate)
  - IT Admins can control/manage all users in domain. Currently no information in API documentation that displays this information.
  - Talk with client, use HTTP status to determine if user has permission
- Microsoft Teams (Easy)
  - Trial accounts cannot enroll with Microsoft Teams. Teams API will facilitate IT support chat.
  - Work with client to get our Azure tenant upgraded



# Team Mozilla

## Status Report

[1 of 4]

### Taking Firefox Screenshot Testing Suite to 11

- Project Overview
  - Better integrate testing suite with Mozilla testing framework
  - Improve functionality and performance of testing suite
  - Increase reliability
  - Add test cases
- Project Plan Document
  - Functional skeleton set up
  - Developing rough draft
  - Communicating with clients to flesh out specifications
  - 10% complete



# Team Mozilla

## Status Report

[2 of 4]

### Taking Firefox Screenshot Testing Suite to 11

- Server Systems / Software
  - Mozilla VC repositories
  - Mozilla Try servers
  - Mozilla automated testing
  - All maintained by Mozilla
- Development Systems / Software
  - Building local Firefox (Windows bug)
  - Connecting to Try
  - Mozilla technologies accessed by us



# Team Mozilla

## Status Report

[3 of 4]

### Taking Firefox Screenshot Testing Suite to 11

- Client Contact
  - 2 meetings, every Friday
  - IRC
  - Weekend hackathon scheduled for Oct 14-15
- Team Meetings
  - 7 meetings, every Friday at least
  - Planning to meet 3-5 times a week
- Team Organization
  - Trello, IRC, GroupMe
  - Communications Director – Rand, Project Status Organization – Mike, Meetings/Scheduling Coordinator – Chris, Technical Specialist – Robin, Quality Control - Jack





# Team Mozilla

## Status Report

[4 of 4]

### Taking Firefox Screenshot Testing Suite to 11 Risks

- Process of moving from Tier 3 to Tier 2
  - The internal process of Mozilla moving the code from Tier 3 to Tier 2 may take time
  - Attempt to leave a reasonable amount of time for bureaucratic processes
- Limited and disorganized documentation
  - Information is dispersed and sometimes nonexistent
  - Communication via IRC with specific developers, personal documentation
- Learning Mozilla specific technologies
  - Mozilla has many in-house technologies: JSM, XUL/XBL, Mach, Try servers, Mercurial
  - Identify the relevant technologies and restrict our focus to those only



# Team MSUFCU

## Status Report

[1 of 4]

### Digital Banking with Chatbots

- Project Overview
  - Automate support with the ability to still transfer to live chat if needed
  - Leverage natural language processing
  - Integrate with Alexa, Mobile App, Website
  - Support SMS and iMessage
- Project Plan Document
  - Functional and Design Specs(50% done)
  - Mockups(75% done)



# Team MSUFCU

## Status Report

[2 of 4]

### Digital Banking with Chatbots

- Server Systems / Software
  - MSUFCU API to access sample bank data (requested, not obtained yet)
  - Twilio API to integrate SMS messaging (have tested, working)
  - API.AI, Amazon Lex (have played with a bit)
- Development Systems / Software
  - Android Studio for Mobile App is running
  - API.AI, Amazon Lex (have played with a bit)
  - Node.JS/Express (not setup yet)
  - Webstorm (have tested, working)



# Team MSUFCU

## Status Report

[3 of 4]

### Digital Banking with Chatbots

- Client Contact
  - Met with twice(1 conference call, 1 in person)
  - Weekly in person meetings on Wednesdays @ 5pm
- Team Meetings
  - Weekly client meetings on Wed., met once and called once
  - Weekly team meetings before class on Wed., met 5 times
- Team Organization
  - Project Managing: Josh and Cori
  - Gus: iOS/Android, Chuanyun: Web App/Database, Cori: NLP/Android, Josh: Google Action/API, Syed: Alexa/API



# Team MSUFCU

## Status Report

[4 of 4]

### Digital Banking with Chatbots

#### Risks

- Risk 1
  - Potential vulnerability of intercepting confidential data from database
  - Integrate SSL to create an encrypted tunnel between client and server
- Risk 2
  - Gaining access to an unauthorized bank account
  - Implement robust user access control to provide access to authorized users only
- Risk 3
  - Client wants apps on a variety of different platforms
  - Determined Google's API.AI is the best fit as it has integrations for multiple platforms. Prioritizing which platforms to focus on through user usage statistics
- Risk 4
  - Google and Amazon's NLP understanding inputs like bank account #'s
  - Testing recognition percentages of complex numbers and inputs on devices



# Team Phoenix Group

## Status Report

[1 of 4]

### OPEN v2.0: Smart Order Picking

- Project Overview
  - Goal is to improve warehouse operations
  - Plan is to use BLE devices for micro-location services based on a digital map of the warehouse
  - Workers will be given an optimized/quickest route to traverse the warehouse and fulfill their pick ticket
  - Inventory Control, things like shelf quantities and other changes to bin locations will be tracked as well
- Project Plan Document
  - Project Plan is coming along well, everyone is contributing to their portion of the document.
  - It has been started, should be finished soon so we can practice our parts for the presentation.
  - Percentage complete is about 85%



# Team Phoenix Group

## Status Report

[2 of 4]

### OPEN v2.0: Smart Order Picking

- Server Systems / Software
  - Amazon Web Service (AWS) Server has been set up
  - Amazon R2 Server is up
  - Ubuntu Server installed on our server in capstone lab
- Development Systems / Software
  - Source code has been received and compiled in Visual Studio 2015. Runs well
  - BLE Devices, both short and long range have been ordered
  - VM has been configured along with needed IDE's on capstone lab computers, "Hello World!" test in C# WPF .



# Team Phoenix Group

## Status Report

[3 of 4]

### OPEN v2.0: Smart Order Picking

- Client Contact
  - We have had multiple Zoom meetings with our client and one Face to Face meeting (cut short) along with various Emails
  - We will be in contact to schedule a weekly Zoom meeting to talk about the week's goals and progress of the project. Every Tuesday/Thursday.
- Team Meetings
  - We have met almost every weekday since the project was assigned, much more progress due to in person meetings
  - We have two scheduled team meetings per week, as well as impromptu meetings throughout based on availability.
- Team Organization
  - We have divided what work we think needs to be done throughout the semester. (Eg. Server Side, Client Side, SQL and Fastest Path Algorithm)
  - We have a git repo, Slack Messaging, a Trello Board, Outlook Calendar, Zoom Web Conferencing to help us stay organized and on task!





# Team Phoenix Group

## Status Report

[4 of 4]

### OPEN v2.0: Smart Order Picking

#### Risks

- BLE Beacon Integration
  - Working with BLE devices, worried about the range as well as the fluctuation in power of the signals that they emit. Staying connected and triangulating a current location of the user.
  - Research different software (3 types) to use along side the devices. Test beacons in several rooms with different types of obstacles and environments.
- Intense calculations on the tablet
  - Power consumption on Windows Surface Pro due to intense calculations. We will have to calculate shortest/most effective path, along with making calls to SQL server for Inventory Control
  - Look into doing the calculations on the backend. Constant critiquing of code to ensure a minimal loss of efficiency.
- Best Absolute Path Algorithm
  - Finding the absolute path for our picker in real time. This algorithm will have to be fast/efficient.
  - Research fastest path algorithms to either find a library that will provide an efficient solution, or create our own.
- Distributed System
  - Working on making the app have a distributed system. Multiple pieces running in different places can work, but we must be careful when connecting them.
  - Make sure that our servers are setup correctly, research efficient ways to connect them.



# Team Rook

## Status Report

[1 of 4]

### Cloud Security Event Processing and Alerting Platform

- Project Overview
  - Build a web application used by security analysts.
  - Manage Correlation Rules
    - Edit existing rules.
    - Examine performance of new rules.
  - Upgrade and migrate current rule engine.
  - Notify support and stakeholders when rule is triggered.
  - Integrate into Rook's Force Platform.
- Project Plan Document
  - 25% complete
  - Rough draft of executive summary complete.
  - Rough draft of functional specifications complete.
  - Risk section complete



# Team Rook

## Status Report

[2 of 4]

### Cloud Security Event Processing and Alerting Platform

- **Server Systems / Software**
  - **Ubuntu Server with Nginx + Django (Amazon Web Services EC2)**
    - Serving front end application
    - RESTful API to reach backend
  - **Amazon Web Services**
    - S3, Athena, Simple Email Services, Lambda, and Elastic Search
    - Python, SQL, and JSON
  - **Front End**
    - HTML, CSS, and Javascript (React/Redux)
- **Development Systems / Software**
  - Visual Studio Code and PyCharm for IDEs
  - Git (Bitbucket) for version control
  - Jira (Atlassian) for sprint planning



# Team Rook

## Status Report

[3 of 4]

### Cloud Security Event Processing and Alerting Platform

- Client Contact
  - Met with client in person twice last week. Daily communication via hipchat and gmail.
  - Weekly conference call scheduled for Wednesday mornings at 9:10.
- Team Meetings
  - To date we have met 5 times.
  - We plan to meet on Tuesdays, Thursday, and Fridays.
- Team Organization
  - Team Roles
    - Front End – Jake, Brad
    - Back End – Brian, Alex
    - Full Stack - Kaushik



# Team Rook

## Status Report

[4 of 4]

### Cloud Security Event Processing and Alerting Platform

#### Risks

- Risk 1 - Integrating Django w/ AWS Authentication. Where do we store AWS credentials?
  - Will figure out mitigation strategy out as soon as we get our teams AWS credentials.
- Risk 2 - How will we notify people of alerts? Needs to be configurable from frontend.
  - Build a “hello world” using Amazon Simple Email Services.
- Risk 3 - Migrate the existing log processing engine to serverless lambda functions.
  - Lots of redundancy; Refactor? Doing so could constitute project scope creep.
  - Continue dialogue on the topic with client. Further explore the possibility by reviewing existing source code.
- Risk 4 - Cannot Start UI until backend is functional.
  - We will attempt to use “storybook” to build front end components without a functioning backend.
- Risk 5 – Everything must be within the Rook network.
  - Use existing vpn client provided by Rook.



# Team Spectrum Health

## Status Report

[1 of 4]

### Symptom Checker

- Project Overview
  - Users input symptoms.
  - From there, the app determines what type of care they need (urgent care, ER, etc.).
  - The app can then schedule an appointment for the user if necessary.
- Project Plan Document
  - Plan document skeleton has been created with names filled in.
  - Executive Summary also written.
  - Functional, technical, and design specifications still need to be written.



# Team Spectrum Health

## Status Report

[2 of 4]

### Symptom Checker

- Server Systems / Software
  - Azure server set up with Microsoft SQL Server configured.
- Development Systems / Software
  - Android Studio installed and starter project created.
  - Xcode installed and starter project created.
  - Github repositories set up with projects committed.
  - Zenhub agile development cards set up for Github repos.
  - Visual Studio with .NET Core 2.0 is set up with a starter project and sample models.



# Team Spectrum Health

## Status Report

[3 of 4]

### Symptom Checker

- Client Contact
  - Met on Friday, 9/8, in Grand Rapids
  - Scheduled weekly conference calls, as well as team lead check-in.
- Team Meetings
  - Scheduled weekly meeting with GVSU team.
  - Have met 4 times so far.
- Team Organization
  - Appointed client contact/team lead, Chris McGrath.
  - Other members have decided on development focuses, but have yet to be assigned strict roles.





# Team Spectrum Health

## Status Report

[4 of 4]

### Symptom Checker

#### Risks

- Risk 1
  - We are working out of a shared codebase/architecture with a capstone team from GVSU.
  - To mitigate this, we are planning on using our previous mobile development experience to establish the architecture as quickly as possible.
- Risk 2
  - We need to determine and implement a machine learning algorithm for matching user input to clinical symptoms.
  - One of our group members has experience in algorithms, as well as contact with professors in this field.
- Risk 3
  - We have to handle/potentially secure a portion of the app for paying bills.
  - We will look into their current web platform for bill payment, as well as discussing this feature further with the client.



# Team Symantec

## Status Report

[1 of 4]

### Secure Application Layer API Proxy

- Project Overview
  - Modernize web interface to Symantec's VIP System
  - Acts as a secure intermediate layer for pre-existing interface
- Project Plan Document
  - Project Plan is 30% done
  - Primitive architecture design has been created
  - Development systems agreed upon



# Team Symantec

## Status Report

[2 of 4]

### Secure Application Layer API Proxy

- Server Systems / Software
  - VIP Manager account created
  - Access to VIP's SOAP API granted
  
- Development Systems / Software
  - Windows 10 VM installed and working
  - Using C# and Visual Studio 2017
  - Version control using Gitlab
  - "Hello World" API setup and working



# Team Symantec

## Status Report

[3 of 4]

### Secure Application Layer API Proxy

- Client Contact
  - Conference called with Shantanu on 9/7
  - Recurring call with Shantanu on Thursdays @ 4:30 PM
- Team Meetings
  - Team has met 6 times (Triage meetings, Conf. calls, etc.)
  - Team will be meeting on Monday-Thursday after 4:30pm.
- Team Organization
  - Jacob Carl – Client Contact, Security Engineer
  - TJ Kelly – Software Design & Test Engineer
  - Steven Kneiser – Software Design & Test Engineer
  - Lauren Allswede – Project Manager, Security Engineer
  - Yili Luo – Security & Test Engineer



# Team Symantec

## Status Report

[4 of 4]

### Secure Application Layer API Proxy Risks

- Load testing abilities are slim
  - Cannot test multiple calls for requests at once
  - Talk with client about multiple mock accounts
- VIP API documentation
  - Documentation may be incomplete
  - Work with client to clarify misunderstandings
- No Access to Symantec's VIP API
  - Do not have admin access to work with existing SOAP API
  - Contact Dr. D if not resolved by Thursday



# Team TechSmith

## Status Report

[1 of 4]

### TechSmith Director

- Project Overview
  - Creating video content is a challenge
  - Enable a friendly interface to stage a video
  - User will speak into a microphone
  - Application will return scene for video based on user request
- Project Plan Document
  - 90% Done
  - Still need to finalize Database Schema
  - Proofread



# Team TechSmith

## Status Report

[2 of 4]

### TechSmith Director

- Server Systems / Software
  - Microsoft Azure – Account Activated by Client
  - SQL Server – Test DB Setup and Connected to Visual Studio
- Development Systems / Software
  - .NET “Hello World” Web Application Setup In Visual Studio and published to Azure
  - Microsoft Cognitive Services API – Tested, but Not Yet Setup
  - SoundCloud API – Not Yet Setup
  - Bing Image Search API – Not Yet Setup



# Team TechSmith

## Status Report

[3 of 4]

### TechSmith Director

- Client Contact
  - Initial Client Meeting On-Site last week Tuesday
  - Weekly Conference Call Set for Tuesdays at 4 PM over Google Hangouts
- Team Meetings
  - Weekly Meeting after class Monday and Wednesday
  - Set Tuesdays at 3 PM in Capstone Lab
  - Have Met 6 Times Already
  - Meeting As Available Other Times – Organizing Over Slack
- Team Organization
  - Pranay – Client Contact
  - Jake – Project Manager
  - Planning to split up frontend and backend work





# Team TechSmith

## Status Report

[4 of 4]

### TechSmith Director Risks

- Microsoft Azure Computing Resources
  - Video playback needs a lot of computing power
  - Limit the number of API Calls outside of Azure
  - Research each Azure tier and consult with Client to determine optimal resource allocation
- Video Animation and Playback in .NET
  - We don't know how we will animate the videos and play them back
  - Determine capabilities of Microsoft Media Foundation SDK to determine if it will suit the needs of the application
  - Find another SDK to enable video animation and playback in our application
- Storing Video Project State
  - Users can have multiple projects and our application must be able to store and retrieve the progress they have made on each one
  - Research how to store and retrieve the state of assets in the video using JSON or XML
- Microsoft Cognitive Services Speech API
  - We do not know the extent of the capabilities of the API
  - Run "stress tests" by making long, outrageous requests and record the accuracy of the results



# Team Two Men and a Truck

## Status Report

[1 of 4]

### Online Moving Estimator

- Project Overview
  - Online moving quote estimator
  - Text and video chat with Customer Service Representative (CSR)
  - Mobile app with image recognition
  - Appointment scheduling feature
- Project Plan Document
  - Early stages - 10% complete
  - Assigned team members to specific sections



# Team Two Men and a Truck

## Status Report

[2 of 4]

### Online Moving Estimator

- Server Systems / Software
  - Already have chat prototype running on Arctic
  - Will set up AWS instance this week for signaling server
- Development Systems / Software
  - webRTC for video and text chat
  - PHP/Javascript, PHPStorm
  - openCV (Python) for image processing
  - PHP Ratchet for signaling server
  - PHPMyAdmin for SQL database



# Team Two Men and a Truck

## Status Report

[3 of 4]

### Online Moving Estimator

- Client Contact
  - Scheduled weekly conference calls
  - Initial prototype meeting
  - Currently coordinating an in person meeting
- Team Meetings
  - Weekly meeting Wednesday after class
  - Met several times already
- Team Organization
  - Clay - Program Manager, Daria - Image recognition SME, James - webRTC SME, Kevin - Database SME, Liyang - UX SME



# Team Two Men and a Truck

## Status Report

[4 of 4]

### Online Moving Estimator

#### Risks

- Video Conferencing
  - No team member has experience with webRTC or video conferencing
  - Figured out text chat, already sourced useful tutorials
- Recognizing volume based on image classification
  - Don't know if estimate will be accurate
  - Sourced examples of a similar process with food density on a plate
- Writing a signaling server
  - No team member has experience with this
  - Sourced libraries and tutorials, also can use existing services as backup



# Team Union Pacific

## Status Report

[1 of 4]

### RailBuilder: The Great Race to Promontory

- Project Overview
  - Generate 3D terrain using information from USGS
  - Populate and texture the terrain based on land classification
  - Build a game to showcase the above functionality
- Project Plan Document
  - Approximately 85% completed
  - Working on generating graphics and concept images
  - ~22 pages in length



# Team Union Pacific

## Status Report

[2 of 4]

### RailBuilder: The Great Race to Promontory

- Server Systems / Software
  - Currently working on pulling data from USGS
  - All files for this project will be stored locally
- Development Systems / Software
  - Unity 2017 has been set up on our personal and lab machines
  - Visual Studio 2017 will be used to support the newest .NET framework within unity
  - Looking into using external libraries for unzipping USGS compressed files
  - All 3D models and assets have been provided by Union Pacific



# Team Union Pacific

## Status Report

[3 of 4]

### RailBuilder: The Great Race to Promontory

- Client Contact
  - First in person client meeting was Thursday the 7th
  - Weekly meetings will occur on Thursdays at 3pm
- Team Meetings
  - Team Meetings are scheduled for Tuesdays and Thursdays at 2pm, and Wednesdays before and after class
- Team Organization
  - Team lead – Trever Daniels
  - Client Contact - Jacob Young
  - Front End Systems – Zach Brenz, Kyle Bush, Trever Daniels
  - Back End Systems – Jacob Young, Declan McClintock





# Team Union Pacific

## Status Report

[4 of 4]

### RailBuilder: The Great Race to Promontory

#### Risks

- Retrieving and Interpreting Terrain Data From USGS
  - Pulling zip files from the USGS site and translating them into numbers for our system
  - Mitigation: Working with our contacts from Union Pacific to see how they get the data
- Creating Terrain Using Height Values
  - Applying the height values from USGS maps into a 3D visualization
  - Mitigation: Generating terrain with random heights
- Texturing Terrain Based on Land Classification
  - The 3D terrain should look like its real world counterpart
  - Mitigation: Using basic colors to classify specific parts of the terrain
- Intelligently Place Environment Assets on the Terrain
  - Place trees, roads, and buildings onto the terrain at different heights
  - Mitigation: Placing assets randomly based on terrain and asset height



# Team Urban Science

## Status Report

[1 of 4]

### KPI Kruncherz

- Project Overview
  - Create responsive web application
  - User Base: Car Dealership Employees
  - Recommend actions based on user's questions and current performance
- Project Plan Document
  - Skeleton Document
  - Analyzed Risks
  - Technical Specifications Outline



# Team Urban Science

## Status Report

[2 of 4]

### KPI Kruncherz

- Server Systems / Software
  - Localhost set up for initial deployments
  - Microsoft MySQL Server Management Studio
    - Populated table with created data
- Development Systems / Software
  - ASP.NET
    - Simple app running
  - .NET Core 2.0
  - Azure – MS Cognitive Services
    - Keyword Finder integrated
    - Other APIs in works



# Team Urban Science

## Status Report

[3 of 4]

### KPI Kruncherz

- Client Contact
  - Weekly meetings every Wednesday afternoon
  - On-site meeting last Friday - discussed business practices
- Team Meetings
  - Weekly meetings Monday and Wednesday
  - Three meetings completed
- Team Organization
  - Will Renius - Project Manager
  - James Grenfell - Client Contact
  - Development tasks divided



# Team Urban Science

## Status Report

[4 of 4]

### KPI Kruncherz

#### Risks

- Web app vs native app
  - May want to create native mobile app to best serve client
  - Develop responsiveness early to determine if web-app serves goals well
- Keyword/KPI mapping
  - Accurate guesses to KPIs from user input
  - Test other NLP Api
- Dataset
  - We are not provided with any real data and are creating dummy data
  - Use NLP methods that won't require training on questions and results



# Team Yello

## Status Report

[1 of 4]

### Automatic Resume Verification

- Project Overview
  - Verifying work and school credentials.
  - Submit credentials to web application.
  - Uses blockchain technologies.
  - Protects user data through one-way hashing.
- Project Plan Document
  - Underway
  - Outline finished
  - 15% done
  - Sections assigned



# Team Yello

## Status Report

[2 of 4]

### Automatic Resume Verification

- Server Systems / Software
  - AWS Server Instance
  - Python 3.6
  - Blockchain
- Development Systems / Software
  - Ruby on Rails
  - Web App
  - RESTful API



# Team Yello

## Status Report

[3 of 4]

### Automatic Resume Verification

- Client Contact
  - Weekly conference calls scheduled
  - Client point of contact assigned
- Team Meetings
  - Triage Meeting Wednesdays
  - Group Meetings Fridays
- Team Organization
  - Ryan is point of contact
  - Nathaniel is in charge of the blockchain
  - Brandon has the most web development experience





# Team Yello

## Status Report

[4 of 4]

### Automatic Resume Verification

#### Risks

- Scalability and Security
  - Might not meet client specifications, proof of source issues
  - Explain problems to client and re-structure the specifications
- Custom Blockchain
  - Client requested Ethereum although a custom one would work better
  - Explain to client the benefits of a custom blockchain
- Unfamiliar Web Framework
  - Group is not familiar with web application development
  - We will take time to learn it together as a team, pdfs sent out (books).
- Unfamiliar with RESTful APIs
  - In addition, group has not worked with the creation of RESTful APIs.
  - Documentation and research on how to create a RESTful API.

