**Final Presentation Meeting Agenda**

**When & Where:** May 11th, 2020 ~ 4:30pm - 4:55pm via Zoom

Zoom Meeting: <https://csusm.zoom.us/j/182370297>

**Attendees**: Dr. Kristin Stewart, Dr. Shaunn-inn Wu, Stoic Solutions Team

**4:30pm - Formal Greetings**

**4:30pm - Agenda Overview**

**4:31pm - Background**

**4:32pm - Project Requirements Overview**

**Adam/Brian - 4:33pm - Project Management**

**Brian**

- Timeline generated alongside our gantt chart

- Has our primary tasks listed, milestones shown below \*go over\*

* Milestones
  + First team meeting, Research & requirements
  + AWS Setup (Login Info, communication w/ Jason (Prev group)), Report 1
  + AWS Development (360 Image collection, ML testing & algorithm processing)
  + Finalizing images, Database development(Team meeting w/ UI team & setting up)
  + AWS Database Finalization (Fields, communication w/ UI)
  + Final report and meeting

**Adam**

Thanks Brian! Here you can see our Schedule of Tasks and WBS which directly correlate with each other. We have two main sections here; the first being Image and data collection and the second being Litter Data. The Image and data collection portion consists of all the tasks taken to gather the images and data needed in order to fill the database with the proper data. The Litter Data portion consists of all the tasks taken to produce the database. Following our Schedule of Tasks and WBS helped the project run smoothly and allowed us to keep track of our progress in correlation with our Gantt chart. Now, I'm going to hand it back to Brian to talk about the first 3 phases of our project.

**Brian**

-Breakdown our hours and prices for each phase

-Stoic solutions was operating at 28 dollars an hour, whilst our AWS ML ran at .29 cents per hour

PHASE 0 & 1

* Combined phases: Start-up phase, consist of: Initial project start-up, Research & Requirements, JAD 1, Report 1, & AWS Set-up/Configuration
* Our time was heavily allocated to meetings, agenda & resources to those meetings, the development of our first report (statements, gant, cost calculation, time sheets, team info), and gathering info to set-up AWS.
* Went over estimated hours due to startup of everything, no need for AWS uptime as we didn't have the information to start yet

PHASE 2

* JAD 2: Updates/fixes to our reports, meetings, AWS development(Images & ML Algo)
* Time here was primarily allocated to; again the preparation of meetings, and to testing ML algo and collecting 360 images
* Under our estimated hours (not by much), AWS ran a total of 10-hours during this time to collect images

PHASE 3

* PROTOTYPE 1: GSV script development, refining, and testing. Continued gathering of images/finalizing. Report Updates.
* Time here was heavily allocated to the GSV python script development, testing, and refinement. At the same time, finalizing images that were already gathered, and having our first team meet with the UI team to discuss database development.
* Under our estimated hours by big, adjustment to reports required less hours, team was efficient and proactive. AWS did not need to run during this phase.

**Adam**

PHASE 4

* PROTOTYPE 2: Database management and Finalization, Script Adjustments, and updating the report.
* Time here was primarily allocated to AWS Database testing, GSV Python script adjustment, metadata collection, as well as collating & storing the litter metrics.
* This was the biggest phase yet, the team had to adjust any changes to the database fields & GSV Script.
* Most AWS server uptime: at 19.5hr, since this was the phase where all the images were processed through the ML algorithm and the database was created.

PHASE 5

* Final phase: Consisted of the final report, meetings, documentation of the system, and preparation
* Phase 5 was under our estimated hours, and we did not need to run the AWS server at all since we had already processed all the images of san marcos and got the data from them into the database in the previous phase.

At the end of it all, you can see that our total hours and total cost were below what was estimated, and that's an achievement we are very happy about as a team. Now I'm going to hand it off to Jordan to talk about analysis and design.

**4:xxpm - Analysis of Design Decision 1 (Jordan: Slides 8,9,10)**

[Start: Slide 8] Thanks Adam! Throughout the semester we conducted Joint Application Development meetings and through those we encountered some important development decisions. The first of these was to define a geographical region. However, Google’s API doesn’t allow you to simply traverse streets without knowing the coordinates beforehand. Our first attempt at solving this design decision was to try brute force.

[Slide 9] As you saw, an interval of 36 feet just isn’t feasible. We tried other intervals too but even at 100 feet data points are still too dense. With this approach of generating coordinate points, as the interval increases so too does the chance of missing streets entirely. Clearly brute force won’t work.

[Slide 10] Our implementation involved manually defining the region to San Marcos. This isn’t the optimal choice but it does allow us to strictly control our data set and gives us an easily parsable file. I’ll hand it over to Jerry to explain our next design decision.

**4:xxpm - Analysis of Design Decision 2 (Jerry: Slides:11,12,13)**

**4:xxpm - Analysis of ML Processing (Jordan: Slides:14,15)**

[Start: Slide 14] Thanks Jerry! From here on out the complexity of the project was a little lower. All that was required was to alter the 2018 team’s script for our purposes and process all the images we had collected. In total it took about 7 hours for our ~4000 images to process.

[Slide 15] This slide just shows the differences between the two outputs of the past group and ours. We no longer needed the images and instead the latitude, longitude, rank, litter count and date were parsed out and redirected to a text file. I’ll pass it on to Connor to explain our database script.

--

[Slide 16] From here we created a python script that took the raw output from the machine learning server. In database.py we combined duplicate coordinate pairs of angle 1 and angle2 by summing their litter counts and altering their KAB Ranks to reflect those changes. We used PyMySQL to accomplish the task of insertion into the database.

[Slide 17 – Video Demonstration] Here’s a quick demonstration of database.py. There’s our raw text file from the ML Algo. Once we run the script you can see it begins processing the text file and inserting all the rows. Pretty exciting, right? Once it finishes all our data now resides our AWS RDS where it can be queried. Now I’ll hand it over to Connor to talk about the database.

**--**

**4:xxpm - Analysis of Exporting to Database (Connor:Slides:16?,17?,18,19)**

**4:xxpm - Deliverables/Documentation (Matthew:Slides: 20,21,22)**

[Slide 20] Thank you Connor. As we close out this semester the Stoic Solutions team will be handing the keys back to Dr. Wesley Schultz & Dr. Kristin Stewart. This includes the script which takes a route of coordinates, fetches the corresponding Google Street View. In addition we have the script from our predecessor group which we converted from creating an output image to an output text file. Our successful small test produced a 4000 line file called SanMarcosData.txt. Lastly we will be handing off the login credentials to access our database

[Slide 21]We were very thankful that our predecessor's from the Spring 2018 student team titled, "FixIT", created over 30 pages of Documentation which greatly helped in getting us up to speed.

Following that theme, the GitHub link below contains not only our documentation but also the corresponding files being discussed. We hope that the next team to pick up the batton of this long-term project also appreciates this level of documentation.

[Slide 22]As we approach the end of this presentation let us discuss one of our first challenges this semester: How to traverse a geographic area. We asked ourselves, should we automate the data collection by dividing a city into a grid or by searching for litter on all 500 unique street names in San Marcos. For the sake of quickly creating a database to be used by the UI team we plotted routes by hand.

Late in the semester we found that someone had already solved the problem of visiting every street name in the city and it may be key to the next phase of this project which is using automation to scale up from a neighborhood to city. Ricky will be next as we continue looking to the future.

**4:xxpm - Future Enhancements (Enrique: Slides: 23,24,25)**

* Automatization of image collection
  + Current method is time consuming
  + Automated method is needed to scale geographically
* Refining database structure to accommodate larger geographic regions
* Speeding up the Machine Learning Algorithm
  + Finding a way to increase the processing speed of the ML algorithm would allow for less AWS server runtime and a reduction in costs.