

Whoosh Project Management Plan

Course Name: Requirements Engineering

Course #: 4351.001

Instructor: L. Chung

Team Name: Whoosh

Team URL: <http://utdwhoosh.github.io>

Team Members:

Name	Student ID	Email
Sasha Borodin	axb138330	sasha@borodin.info
Luke Carr	lxc112230	luke.carr4@gmail.com
Izuchukwu Elechi	ixe120130	izuchukwu@utdallas.edu
Dustin Endres	dxe110630	dxe110630@utdallas.edu
Dustin Grannemann	dsg101020	dsg101020@utdallas.edu
Thomas Grice	tag130230	augustus.grice@gmail.com
Taber Hust	txh120330	txh120330@utdallas.edu
Marie Imperial	fxi120030	fxi120030@utdallas.edu
Jessica Jennings	jmj112020	jmj112020@utdallas.edu
Elise Keller	exk120130	exk120130@utdallas.edu
Jacinth "Jason" Nguyen	jrn120030	jrn120030@utdallas.edu
Tayaba Saleem	txs121430	txs121430@utdallas.edu
Kesha Shrestha	kxs135430	kxs135430@utdallas.edu
Maelene Tacata	mxt125730	mxt125730@utdallas.edu

Team Member Signatures

Name	Signatures
Sasha Borodin	
Luke Carr	
Izuchukwu Elechi	
Dustin Endres	
Dustin Grannemann	
Thomas Grice	
Taber Hust	
Marie Imperial	
Jessica Jennings	
Elise Keller	
Jacinth "Jason" Nguyen	
Tayaba Saleem	
Kesha Shrestha	
Maelene Tacata	

Outline

1. Introduction

- 1.1 Project overview
- 1.2 Project deliverables
- 1.3 Evolution of this document
- 1.4 References
- 1.5 Definitions, acronyms, and abbreviations

2. Project organization

- 2.1 Process model
 - 2.1.1 Requirements/Analysis
 - 2.1.2 Design
 - 2.1.3 Implementation
 - 2.1.4 Unit Testing
 - 2.1.5 Integration
 - 2.1.6 Integration Testing
 - 2.1.7 Repetition
- 2.2 Organizational structure
 - 2.2.1 Logistics
 - 2.2.2 Development
 - 2.2.3 Design
 - 2.2.4 Project Management
- 2.3 Organizational boundaries and interfaces
- 2.4 Project responsibilities
 - 2.4.1 Project Management Team
 - 2.4.2 Design Team
 - 2.4.3 Logistics Team
 - 2.4.4 Development Team

3. Managerial process

- 3.1 Management objectives and priorities
 - 3.1.1 Planning

- 3.1.2 Organizing
- 3.1.3 Coordinating
- 3.1.4 Reporting
- 3.2 Assumptions, dependencies, and constraints
- 3.3 Risk management
 - 3.3.1 Loss of person
 - 3.3.2 Disagreements within teams
 - 3.3.3 Deliverable preparation
- 3.4 Monitoring and controlling mechanisms

4. Technical process

- 4.1 Methods, tools, and techniques
- 4.2 Software documentation
- 4.3 Project support functions

5. Appendix A - Process Specification

- A.1 Functional Decomposition (IDEF0)
- A.2 Softgoals (SIG)

6. Appendix B - Meeting Minutes

1. Introduction

1.1 Project overview

Whoosh is an information and navigation app specialized for the University of Texas at Dallas. It aims to provide disabled users the information they require to navigate between locations on campus with the most efficient route. The app will include all accommodations across campus for users with a disability, including up to date closure information and construction details. Whoosh aims to make travelling at UTD simple, efficient, and most importantly safe.

We will work with the University and the office of Student AccessAbility to allow the app to route and display accessibility options across campus.

The criteria for the project's success includes the development of a functional app that determines the optimal route for users to take on campus, proper documentation concerning the usage of the app, as well as the finding and utilization of artifacts within the project lifecycle.

1.2 Project deliverables

The following is the list of planned project deliverables:

Preliminary Project Management Plan	September 3rd
Phase 1.1: Powerpoint Presentation	September 29th
Phase 1.1: Preliminary Description	September 29th
Phase 1.1: Product Scenario	September 29th
Phase 1.2: Evolved Project Plan	October 15th
Phase 1.2: WRS Document	October 15th
Whoosh Prototype	October 15th
Phase 2.1: Vision Document	November 10th

Phase 2.1: Evolved Project Plan	November 10th
Phase 2.1: Updated WRS Document	November 10th
Phase 2.2: Process Spec	December 1st
Phase 2.2: Traceability Matrix	December 1st
Final Project Plan	December 1st
Final Project Presentation	December 1st
Refined Whoosh Prototype	December 1st

1.3 Evolution of this document

9/2/15 - Created Project Management Plan

1.4 References

UTD's Official Site: <http://www.utdallas.edu/>

UTD Campus Map: <http://www.utdallas.edu/maps/>

UTD Office of Accessibility <http://www.utdallas.edu/studentaccess/>

Dr. Chung's website: <http://www.utdallas.edu/~chung/CS4351/syllabus.htm>

1.5 Definitions, acronyms, and abbreviations

app - Application, typically referring to a mobile application

UTD - University of Texas at Dallas

2. Project organization

2.1 Process model

For this project, we will utilize the Iterative Model:

2.1.1 Requirements/Analysis

Analysis/Re-Analysis of system/module requirements for current iteration.

Cooperation between team and customer to specify and document all requirements for the system or current system component. Research of the capability of the system to meet requirements and conduction of requirements changes accordingly.

2.1.2 Design

Creation and documentation of architecture of the system or design for the current application module. Validation that design is optimal for the requirements specified. Review of design documentation to ensure that the structure is clear for development team.

2.1.3 Implementation

Construction of the system software components from design documentation.

2.1.4 Unit Testing

Creation of unit tests to test the individual components developed during Implementation phase.

2.1.5 Integration

Integration of system components to form the complete software system.

2.1.6 Integration Testing

Testing of the integrated system's ability to satisfy all functional and nonfunctional requirements documented during Requirements Phase.

2.1.7 Repetition

Iteration of the previous phases, adding to or modifying existing specifications of the system.

2.2 Organizational structure

The project will be implemented using 4 teams that were chosen by personal preference. Team members are primarily accountable for their assigned responsibilities in section 2.4, as well as any deliverables and goals specifically assigned to them via Iteration Goals Sheet, Team lead, or Project Management. Team member group associations may change in the future.

2.2.1 Logistics Team:

Elise Keller

Luke Carr

Tayaba Saleem

Kesha Shrestha

2.2.2 Development Team:

Marie Imperial
Taber Hust
Dustin Endres
Dustin Grannemann

2.2.3 Design Team:

Izuchukwu Elechi
Thomas Grice
Jessica Jennings
Maelene Tacata

2.2.4 Project Management Team:

Jason Nguyen
Sasha Borodin

2.3 Organizational boundaries and interfaces

Each team will elect a **team leader**. Team leaders will be in charge of overseeing their team's progress and communicating progress and concerns within the team. Team leaders will communicate with dedicated **project managers** about the concerns and progress of each team.

Project managers will assist in keep teams in contact with one another. Managers will oversee the progression of each team, making sure that the project remains on schedule and within budget. In turn, managers will also keep teams updated on the overall progression of the project and report any areas of concern.

Team members should also feel free to approach leaders, managers, and Dr. Lawrence Chung should any concern arise.

2.4 Project responsibilities

2.4.1 Project Management Team

The Project Management Team is in charge of overseeing the activities of other teams in

addition to pursuing managerial objectives (see Section 3.1). Managers must be able to communicate with team leaders and have good organizational skills. Project managers play a key role in ensuring that development remains on schedule.

2.4.2 Design Team

The Design Team is in charge of creating an engaging user interface for the application. While the team focuses on the front end, designers must also communicate with the development team to integrate both the front and back ends.

2.4.3 Logistics Team

The Logistics Team is in charge of gathering the required information for development of the the application. This includes map information, disability rights, and more. Logistics will also assist in record keeping, keeping meeting minutes, and communicating with external organizations.

2.4.4 Development Team

The Development Team is in programming the application, mainly with focus on the backend. Using the information gathered from the logistics team, the development team will utilize it to develop the core functionality of the application. The development team will also work closely with the design team to ensure quality in the app's design and feel.

3. Managerial process

3.1 Management objectives and priorities

Whoosh is a simple application consisting of several intricate parts handled by different teams (refer to section 2). To make sure development runs smoothly, managers will have the following objectives in the project lifecycle:

3.1.1 Planning

Managers must be aware of the progress of the teams within the project. Based on that knowledge, managers must decide the best course of actions for teams to take with the goal of completing the overall project in mind.

3.1.2 Organizing

Managers should maintain documents and other artifacts from project development in an organized manner. It is also important for managers to help teams keep track of their progress and obtain information related to the project, mainly previous artifacts.

3.1.3 Coordinating

Due to the overlapping nature of the teams, managers must be able to oversee communications between teams. In addition to communication, managers must also help handle dependencies between teams, i.e. making sure requirements between teams are being met.

3.1.4 Reporting

Managers must be able to report any events or concerns should they occur. This objective is important as events may impact the activities of the teams which may affect the overall project.

3.2 Assumptions, dependencies, and constraints

As of now (9/2/15) the app is limited to the main campus, i.e. residential areas, sports fields, and parking lots will not be factored in. Current development of the application will also be based around existing construction. The limitations may be subject to change as the development of Whoosh progresses.

Team dependencies are a primary critical factor of this project. Team Leads and Project Managers must remain vigilant in order to keep teams on track, in sync, and in touch with one another. Failure to do so may result in project delays, and in the worst case, project failure.

Additionally, time will also play a critical factor into the quality of the app. The current given time limit (one semester) should be enough to develop a refined prototype, however as classes and work continue to progress, it may impact the development.

3.3 Risk management

In order to meet project goals and deliverables on time, all team members are

responsible for complying with and enforcing risk management. The following policies will be used in order to mitigate the following risks

3.3.1 Loss of person

If, due to sickness, unforeseen emergencies or events, a person is not able to attend meetings, class due dates, or perform assigned work, they shall notify the team as soon as possible through appropriate channels of communication (i.e. texting a teammate, posting in slack, or email). Team members will coordinate with each other and other teams if necessary to reassign work.

3.3.2 Disagreements within teams

In the case that team members within a team have disagreements and cannot come to an agreement, they will discuss with the Team Lead and Project Managers in order to come to a resolution.

3.3.3 Deliverable Preparation

In order to ensure that all deliverables are delivered on time, Project Managers are responsible for ensuring that a meeting is scheduled before the deliverable is due. All team members on all teams are responsible for attending the meeting and or class to sign the deliverable as needed.

3.4 Monitoring and controlling mechanisms

In order to mitigate the concern of team dependencies, regular meetings will be held between project managers and team leads. In meetings, attendees will discuss progress and areas of concern that are to be addressed as soon as possible. By discussing progress, it will also be easier to stay on schedule, helping mitigate the risk falling behind.

4. Technical process

4.1 Methods, tools, and techniques

The team will be utilizing the standard Android SDK for all core user interface development. Maps will be stored and rendered using the Google Maps API, which is standard for most Android applications that display graphical maps or use the user's location. The Java programming language will be used for all front-end application development.

The team will collaborate using Slack for project discussion and GitHub for code collaboration and revision control. Android Studio will be used as the primary IDE for all app development.

4.2 Software documentation

All software documentation artifacts will be stored on our Google Drive account. Public information can be view on our team website <http://utdwhoosh.github.io/>.

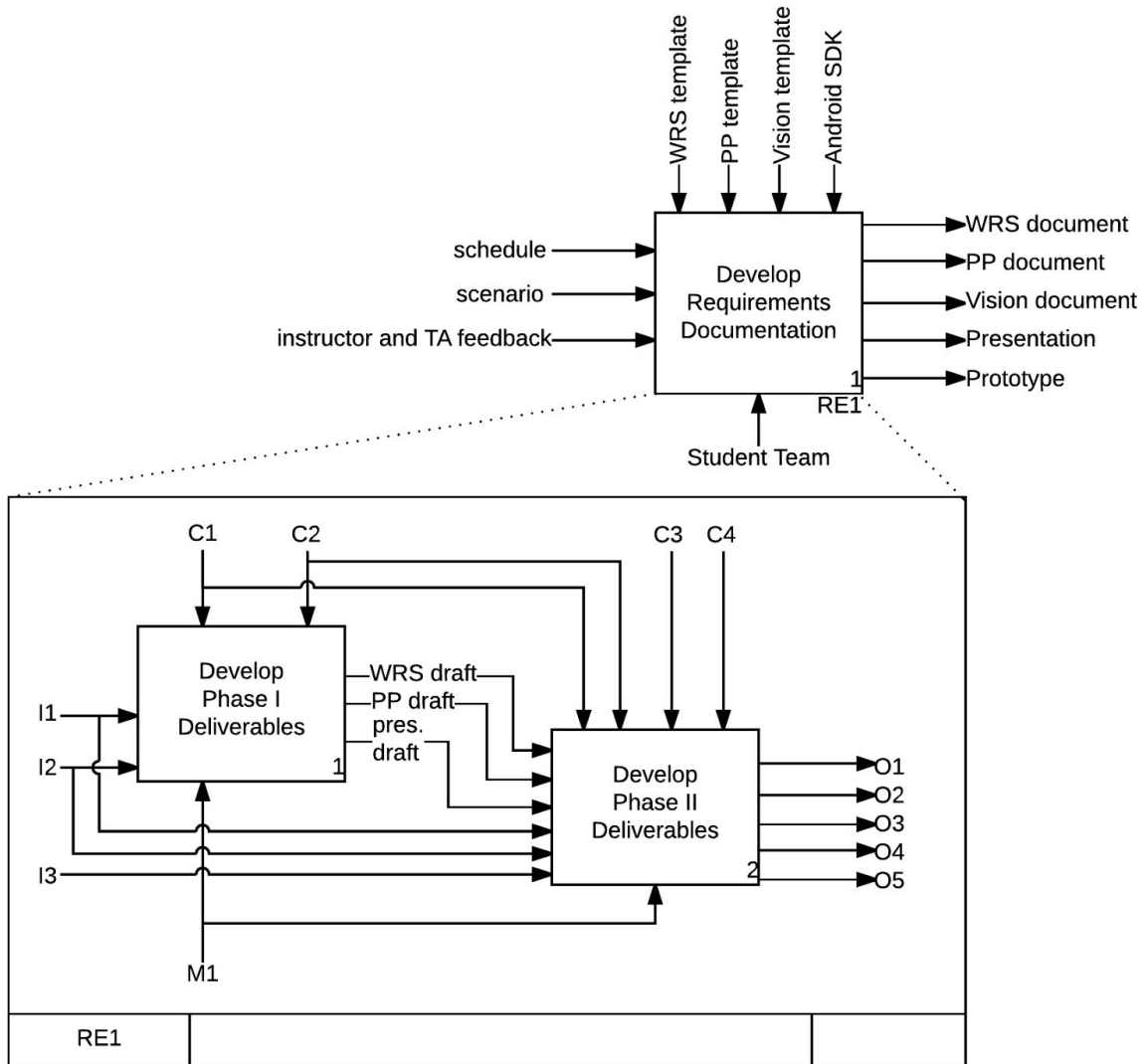
Deliverable Justification

This deliverable has been thoughtfully developed by members of the team with diverse backgrounds, including design, development, web and more. The plan has helped to lay out the foundation for the currently existing teams, as well as help set goals for current sprint.

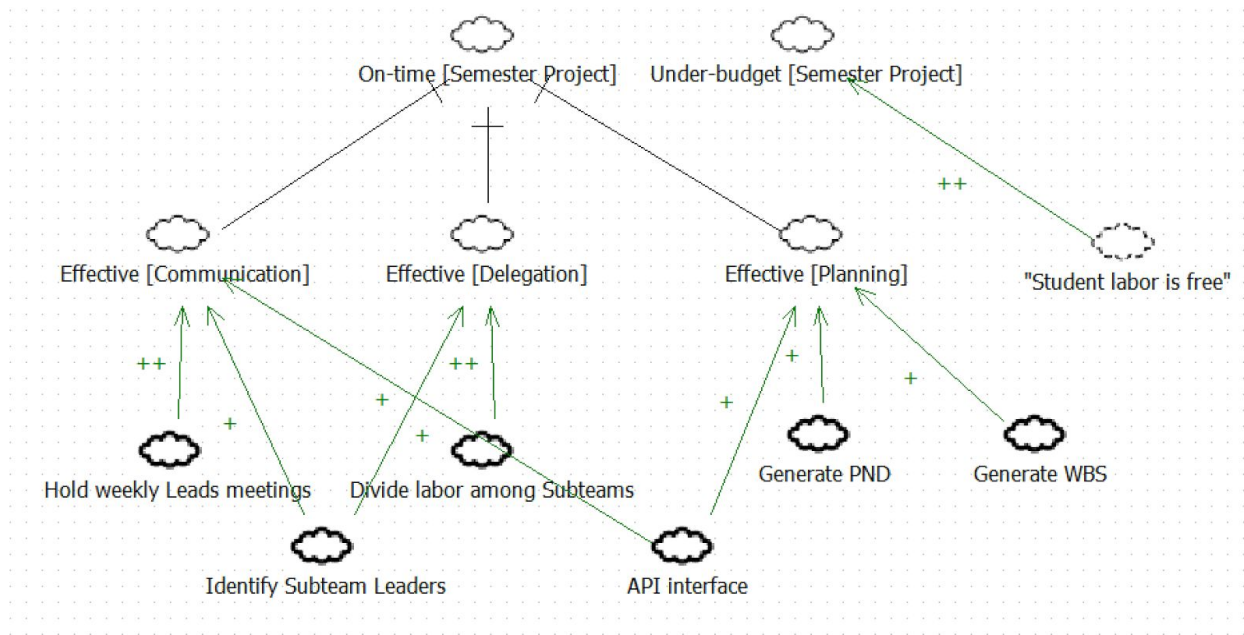
Objectively, the tasks that this deliverable has helped to accomplish account greatly for its quality. Each section was also carefully written to adhere to best practices of software engineering.

5. Appendix A - Process Specification

A.1 - Functional Decomposition (IDEF0)



A.2 - Softgoals (SIG)



5. Appendix B - Meeting Minutes

Date: 9/2/15 | 5pm |

Location: McDermott Library

Participants: Jason, Izu, Luke, Elise, Taber, Tayaba

Agenda:

- App will be focused on helping students, visitors, and faculty with disabilities around campus. Locations of elevators, ramps, etc will be gathered and compiled on to a map using the google maps API.
 - Logistics team is contacting various offices to see what data already exists. Our last resort will be manually gathering data manually.
- Decided on team structure and team responsibilities
 - Front-end (design) and back-end (dev) teams will agree on a set of requirements to ensure that they are building compatible components.
 - Teams should help each other. For example, if somebody on the “logistics” team has nothing to do halfway through the project they should help other teams if possible.
- Decided how to have team leads
 - Each group (design, dev, logistics, PM) will have a team lead. Team leaders will coordinate their group and meet with each other to coordinate the project.
- Decided repository, communication methods
 - Git, slack, google drive
- Team Website
 - Hosted on a new github account where everybody has access.
 - URL:
- Team name: Whoosh with the two o's being wheelchair wheels
 - Somebody please design us an awesome logo

Action items:

- Izu: created Github URL
- Elise: email Office of Student Accessibility and Facilities to ask for handicap accessible route map
- All: Break large group into sub-teams based on preferences and availability. Sub-teams may not have an equal breakdown of members. We grouped members onto sub-teams for the purpose of our 9/3 project management plan deliverable; however, this is subject to change as we progress with the project.

Date: 9/3/15 | 1pm |

Location: ECSS

Participants: Jason Nguyen, Izuchukwu Elechi, Marie Imperial, Maeline Tacata, Dustin Grannemann

Agenda:

- Complete the Software Project Plan Document
 - 3.3 Risk Management
 - 4.2 Software Documentation
 - 4.1 Methods, Tools, and Techniques
- Update team member listings

- update team member listing on project site
- Deliverable justification
- Deciding on first iteration goals for each team

Action items:

- Izu: updated team formations, set up and notified teams of website setup
- All: Completed Software Project Plan Document, discuss potential risks and handling methods, further discuss requirements/design/implementation ideas, set current sprint goals, discussed technologies to use for database, as well as issues and scope constraints.

Where: McDermott library, Jalapeno room

When: 9/7 10am

Attendees: Elise K, Marie I, Dustin G, Jason N, Izuchuckwu

Agenda:

- Reviewed how each subteam is coming along
 - Dustin G has created shell of app
- Discussed how to store and display maps
 - currently have PDFs of campus maps
 - pull from server, so you can update maps. Izu and Taber also recommended. Possibly using Parse
 - store in DB
 - store buildings as nodes in graphs
 - stores maps in json objects, Mongo DB service.
 - Representing data? Store in json file
 - User authentication
 - How to host? We host Mongo DB on server. Possibly buy small one. Trouble with updating
 - -->Parse may be easier to update across
- Scenarios
 - Must-haves:
 - Can direct user from building to building (including floors)
 - Navigation for crutches and wheelchair
 - Gets user location
 - user can input destination location
 - Nice-to-haves:
 - user accounts
 - inclusions/exclusions (ex: user can choose to not be shown elevators)
 - OSA contact information
 - Contact information for other things (buildings, events)
- Discussed approach:
 - create app with first floor of each building, built up from there with elevators and other floors

- Can overlay our campus maps with Google maps
- Team lead meetings
 - Mondays 5:30pm, McDermott library
- Izuchuckwu discussed UI design brainstorm

Action Items:

- Logistics items:
 - create json file for map objects, populate data coordinates (can get from Google Maps)
 - Project Management:
 - finalize roles for Design & Dev regarding UI implementation
-

Where: ECSS lab

When: 9/8 1pm

Attendees: Elise Keller, Luke Carr, Jason Nguyen, Marie Imperial, Maelene Taccata

Agenda:

- Limiting scope to main part of campus (not parking lot or residential area)
- Looked at current campus maps formats
- Buildings we will be mapping
 - AB Activity center
 - AD Administration
 - AH1, AH2 Arts & Humanities
 - AS Art Barn
 - ATC ATEC
 - BE Berkner
 - CB1, CB2, CB3 Classroom buildings
 - CN Clark
 - CR Callier Center
 - ECSN, ECSS
 - FA Founders West Annex
 - FO Founders
 - FN Founders North
 - GR Green
 - GC Green Center
 - HH Hoblitzelle
 - JO Jonsson
 - JSOM
 - MC McDermott Library
 - MO1, MO2 Modular Labs
 - PHA Physics Annex
 - PHY Physics
 - RL - NSERL

- SLC Science Learning Center
- SSB Student Services Building
- SU
- SUFC - Food court
- TH Theatre
- VCB Visitors Center
- Listed information needed for maps
 - ID
 - building
 - restrooms
 - elevators
 - floor
 - automatic door
 - latitude/longitude (and elevation?)
 - skywalks
 - rooms
 - pathways
 - inactive - for Construction
 - active
 - General node class
 - name
 - id
 - type?
- General Hierarchy:
 - Campus
 - Pathway
 - Skywalk
 - Sidewalk
 - Ramp
 - Building
 - Door
 - Automatic Doors
 - Elevator
 - Floor
 - Room
- Discussed ways to map area
 - iPhone pin dropping
 - Android find current location
 - Google maps latitude/longitude coordinates
- Nice to have app doing text to speech

When: September 10th | 5:15pm

Where: JO, outside of classroom

Attendees: Dustin G, Izu, Elise, Jessica J, Sasha B, Thomas G, Jason N

- Design team working on map data organization
- Logistics responsible for gathering information
- May use crutch and wheelchair speeds for ETA and not different map paths
- Dustin proposed data storing schema
 - possibly adjacency list for each node = building
 - weight of node = coordinates
 - discussed computing coordinates at the time versus storing it
 - I/O bound will be fetching data from Google
 - possibly use one data structure for storing and another data structure to running shortest path algorithms on
 - discussed where it would be faster to perform algorithms
 - Izu recommended cloud computing. Possibly using Parse
 - Dustin and Sasha recommended having it hardcoded and done in the app
 - Host data online and app pulls data from that
- JSON file format for map data
- Parse for hosting map data
- Discussed use of Google Maps API
 - for maps framework
 - rendering map of campus and coordinates, we will be overlaying our own floor plans
 - destinations: ideally rooms, second choice the floor of that building (navigate to nearest entrance)
- Differentiate between milestones and sprints
 - Milestones set by professor, hard deadline
 - subteams do not necessarily need same sprint timing
- Jason described requirements idea
 - Navigations
 - Help
 - OCA
 - Event help
 - Emergency
 - emergency contact interface
- Prototype idea
 - start with going from floor to floor, add rooms later
- Navigation
 - Will we be using Google to render map directions?
 - may do our own navigation or show map directions
 - may show in separate steps
 - live tracking IF POSSIBLE - but probably not

Action Items:

- Design team to meet tomorrow and brainstorm logo creating, wireframe creation
 - PM team to start next deliverable - PPT and WRS
 - Need to see accuracy of the find my location call from Google
 - may not be accurate inside
-

When: September 11th | 6:15pm

Where: FN 2.302

Attendees: Design Team: Izu, Jessica J, Sasha B, Jason N, Maelene, Thomas G

- Decided on main features
 - Map
 - Directory
- Discussed ideas for wireframes
 - Drew out storyboard based on use cases
- Main screen: shows campus map
 - Side menu - swipe to the left
 - Search bar
- Zooms in on the map after searching for location
- Decided on use cases
 - Known/known directions
 - Current/known direction
 - Browsing floorplans - up and down arrows to navigate between floors
 - Directory - contact numbers for Office of Accessibility
- Figured out what should be ready for the next sprint
 - Wireframes
 - Map layering - superimposing the campus map on top
 - Partial implementation
 - Floor selector component

Action Items:

- Next sprint: wireframes for two major use cases
 - All: Meetup on Monday 7pm @ FN 2.302 to design wireframes
 - Individuals: Research on map layering and design prototype
-

When: 9/14 | 5:30pm

Where: McDermott library

Attendees: Izuchuckwu E, Sasha B, Dustin G, Jason G, Elise K

Agenda:

- Design update:
 - worked on overall flow of app
 - planned next goals
 - Wireframes

- Mockups
 - looked at APIs they'll be using to know limits
 - Clarifying roles of Design vs Development vs Logistics
 - Who reads in initial data set in (CRUD):
 - Logistics: data gathering and input into Parse
 - Development: data modeling, graphing/routing
 - Design: Android U/I, human directions, directory
 - Interfaces - not sure who?
 - Discussing data storage:
 - potentially JSON or Parse object
 - id: integer //arbitrary, primary key so to speak
 - coords: float, float //latitude, longitude
 - edges: [integer] //adjacent nodes. ex: elevator
 - group: Enum //building name. ex: 0 = ECSS, 1 = ATC
 - subgroup: Enum //floor
 - name: String
 - type: Enum //room, building door, elevator, pathway (most likely just at intersection, but possibly more if curved path) skywalk, bathroom
//possibly using icons to denote type of node
 - Example:
 - id: 22
 - coords: 213.12, 50.67
 - group: ECSS
 - subgroup: 2.206
 - name: 2.206/TI Auditorium
 - type: room
 - Could work image and image metadata all into one data structure with node info

Action items:

- Jason and Elise and logistics team to gather data
- Design and Dev to coordinate working session
- Izu to look into Parse account

When: 9/21 | 5pm

Where: McDermott library

Attendees: Sasha B, Dustin G, Izu E, Elise K, Jason N

Agenda:

- Sasha explained api.pdf
 - Design team working on LocationService primarily
 - searchGroupNodes() limites search nodes to group. May be able to use Parse API
 - getGroupMap() produce map for floor. Bitmap replaced by Drawable

- Node
 - switching IDs to string. Assuming we're going to deal with Parse keys
 - getDefaultFloor() need to be able to get bottom floor of building
- Directions
 - centered around node
 - type: turn, head, around <left, right, slight left, slight right, ahead>
 - has to convert set of nodes and translate into directions
- Decide on hard deadlines for project deadline
 - pull up map (getCampusMap)
 - tap and hold current location (getClosestNode)
 - search current & destination node (searchNodes)
 - Use Cases
 - Case 1: User goes from place they know to another known place (class -->next class)
 - Case 2: User plots from current location to known location
 - Case 3: User browses map, views nodes and floor plans
 - includes campus maps and floor plans
 - includes searching for building
 - getClosestNode
 - getCampusMap
 - getGroupMap
 - **Going to focus on for next deliverable**
 - Case 4: User views contact directory
- Presentation for next Tuesday
 - scenarios
 - As-Is
 - To-Be
 - demo of app
 - mockups or wireframes
 - evolving Project Plan
- Reviewed wireframes from Thomas
 - currently building to building
 - would like to see node to node on floor

Action Items:

- Logistics to look at Parse Data browser
 - see if easier to update data directly into Parse
 - backup: updated in Google sheets, import into Parse
 - possibility of using memo in Parse structure to create human-readable
- Sasha to make updates to UML
- Izu & design team to provide mockups
- Dev to highlight some implementation details
- PM team to create PPT shell and divide into subteams

When: 9/25 | 4pm

Where: ATEC computer lab

Attendees: Luke C, Dustin E, Elise K, Jason N, Tayaba S

Agenda:

- Working session for PPT next week
 - added AS-IS, TO-BE scenarios, more Logistics information
- Working session for map Node data
 - finishing up adding coordinates
 - also worked on inputting room numbers

Where: ATEC 2nd floor

When: 9/28 | 5:30pm

Attendees: Sasha B, Dustin G, Elise K, Jason N

Agenda:

- Final presentation
 - assigning/letting people sign up for slides
 - reviewing each section
 - discussing placement of sections
 - scope
 - use cases
 - future enhancements/limitations

When: 10/5 | 5:30pm

Where: McDermott library

Attendees: Sasha B, Dustin G, Elise K, Jason N, Izu C

Agenda:

- Set internal deadlines
 - Design and Dev - what use cases they can do in next week and a half
 - Design: learning platform, before going into front-end dev
 - Logistics:
 - Tab to describe row for each ATC, SSB, EXTERNAL (node group records)
 - Need map to represent this node group, floor ID and lat/long bound
 - <Filename> <Node group> <floor ID> <NE lat/long> <SW lat/long>
 - lat/long for center of building
 - need lat/long for corners of building, central node for Building group
 - getting data into Parse
 - need to transform natural keys to Parse keys, will need code

- Ask team if someone willing to write import utility to take Node data from CSV to Parse using Parse API
 - Sasha to do import utility
 - Basic prototype due (internally) by weekend so time to test before Oct 15
 - breakdown of Dev and Design
 - scope: map browsing
 - update to Node API to return group of nodes
 - Izu went through interactive mockup
-

When: 11/2 | 5pm

Where: McDermott library

Who: Sasha B, Dustin G, Elise K, Jason N

- Design team
 - need update
 - Thomas to be new team lead
 - Need boilerplate/template
 - possibly assigning new people
 - Give them until end of week to produce something, otherwise shuffle the resources
 - Logistics
 - doing Vision Document
 - Need to wait on PM team to review template
 - PM
 - to work on deliverables for next week
 - Development
 - on track
 - need some testing in Location Service
 - almost done with Routing Service
-

When: 11/15 | 5pm

Where: McDermott Library

Who: Sasha B, Dustin G, Elise K, Jason N

- Updates
 - Sasha about 80% done with data import
 - Dustin demonstrated app so far
 - Design not represented, talking about where we think they are
- Is directory necessary?
 - instead of returning list of rooms in search, expect exact match
 - Izu to do map
- Timeline
 - 1 week

- priorities:
 - finish getting data input into Parse
 - test Marie's work in Parse
 - 1: type in room and have pin on map
 - 2: navigation on first floor
 - already have method to return route, doesn't return map though
 - overlaying floor plans
 - walking through steps of navigation
 - Dustin G to work on navigation
 - right now just plotting nodes in navigation
 - Izu to work on overlaying maps during navigation
 - directory should be easy to do
 - Thomas working on sidebar right now
 - Done
 - LocationService done except not fully tested with Parse
 - RoutingService done, have not tested
 - Documentation
 - flesh out PIG and SIG diagram
 - Action Items
 - By Thursday, test code. if doesn't work, hard code
 - Code session on Monday
-

When: 11/24 | 6pm

Where: McDermott library

Who: Sash B, Dustin G, Thomas G, Elise K, Jason N

Agenda:

- Status check
 - prioritized remaining items
 - route navigation
 - floor plan viewing
 - divided up work and did working session
 - Jason: traceability matrix
 - Elise: final presentation, Vision document
 - Rest: app programming
-

When: 11/30 | 6pm

Where: McDermott library, Horned Lizard room

Who: Sasha B, Dustin G, Elise K, Jason N

What:

- Working session
 - Remaining diagrams

- Final presentation
 - fixing nodes for application
 - progress check
-