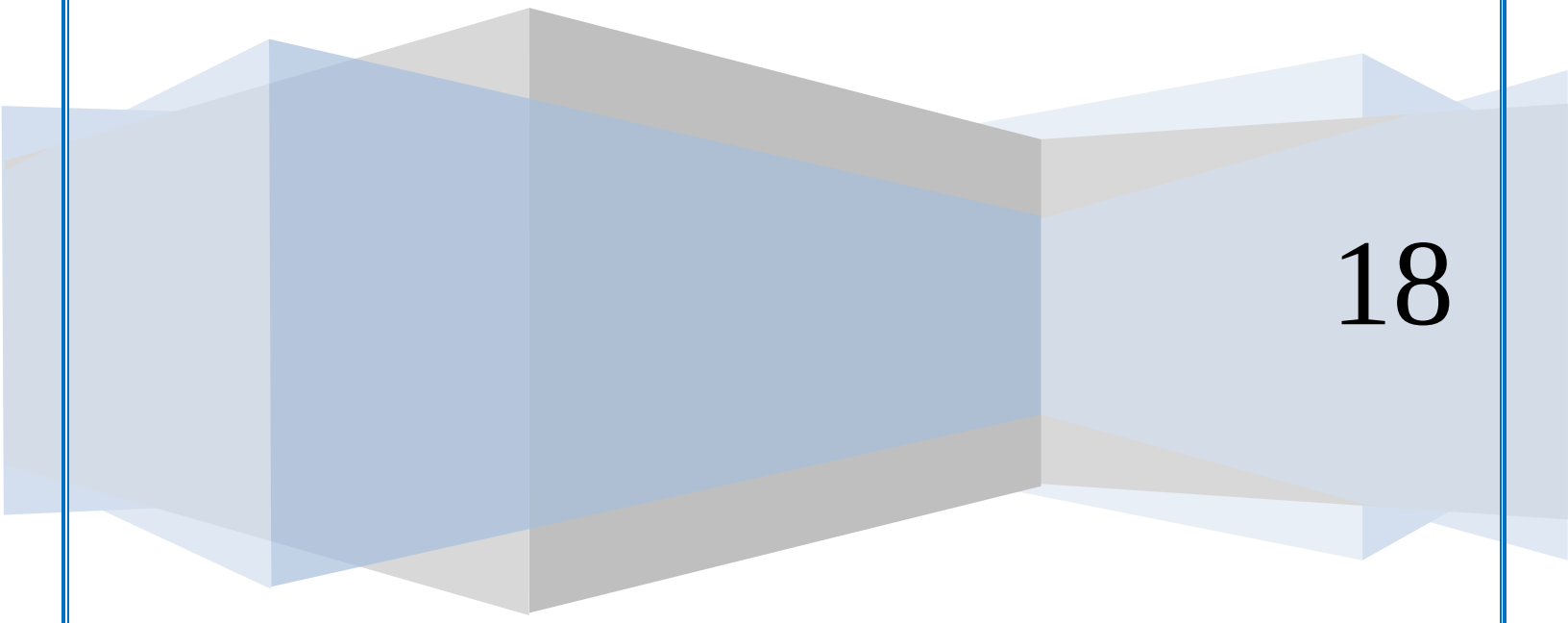
Portland State University - Oregon MESA

MESA EVERYDAY

Project Plan

Fadi Labib



18

Table of Contents

Introduction

MESA Everday is a web app that gamifies the MESA experience for students who are part of the program. This documentation explains the plan for building the web app as well as a risk management plan with mitigation strategy for risks associated with the project.

Purpose and Scope

This document gives the target audience the big picture about how the project will be planned. It starts by defining the project goals and scope, and then provides information about how the project will be staffed in the human resource plan. From there, the document gives a short summary of the major milestones and deliverables, and then breaks down those major milestones into smaller pieces in the work breakdown structure and assigns them to members of the team. The document also includes a list of major risks to the project as well as the mitigation plan put in place to manager those risks in the risk management plan section. The document ends with the plan for communication among team members as well as a stakeholder management plan.

Target Audience

This document primarily targets the web app developers and faculty members who are part of Portland State University (PSU) Computer Science (CS) Capstone.

Project Goals

MESA Everday is a web app that gamifies the MESA experience for students who are part of the program. Oregon MESA is a pre-college academic program hosted by Portland State University that equips underrepresented students with science, technology, engineering and math (STEM), invention, and 21st-century skills. This app will help MESA collect core data about student experiences and their MESA journey to give testimonials that are more concrete for grant funders as well as help students participate in more MESA sponsored activities and events that might allow them to get college credits for work that they have done. The goal is to build this app in a way that caters to 6th - 12th grade students and make them excited to participate in MESA events and earn high scores.

Project Scope

The project is to build a web app that students could login to and be able to record events and activities that they have participated in and be able to earn points for their work. Actively participating student earn more badges and rank up in levels in a way parallel to the amount of work they put into it. The minimal viable product (MVP) for this project is for the web app to have a front-end user interface tied to a backend database that stores the data students enter. The web app would run queries on the database to calculate scores, grant badges, and display information that help students keep track of their progress. For details on what is part of the minimal viable product, please consult the requirement's document.

If time allows, the developers will built in support for an administrator side to the app that will give MESA representative the ability to control the rules of the game, monitor student activities, and audit their inputs. For details on the capabilities of the administrator side, consult the requirements document.

The product whether including only the MVP or has the added admin side support will be hosted on the internet provided that MESA representative meet certain requirements that has been communicated to them in the hosting plan document. Production hosting will be out of scope if MESA representative are unable to meet those requirement.

Software Development Methodology

The project will be built using primarily the waterfall software development model, while integrating some elements of the agile software development model. The project will mostly follow the waterfall model. The developers start by gathering all requirements and designs before actually implementing the code and most of the testing will occur after the implementation phase. It will integrate elements of the agile method like daily online standup meetings, integrating some testing during development stage, and having roles like a product owner.

Human Resource Plan

This section explains the details of how the project will be staffed and the members of the team as well as the roles and responsibilities of those individuals.

Recruiting and Building Team

The project manager evaluates team members' resumes and project interest then drafts people for their team based on the available pool of applicant experiences and interests.

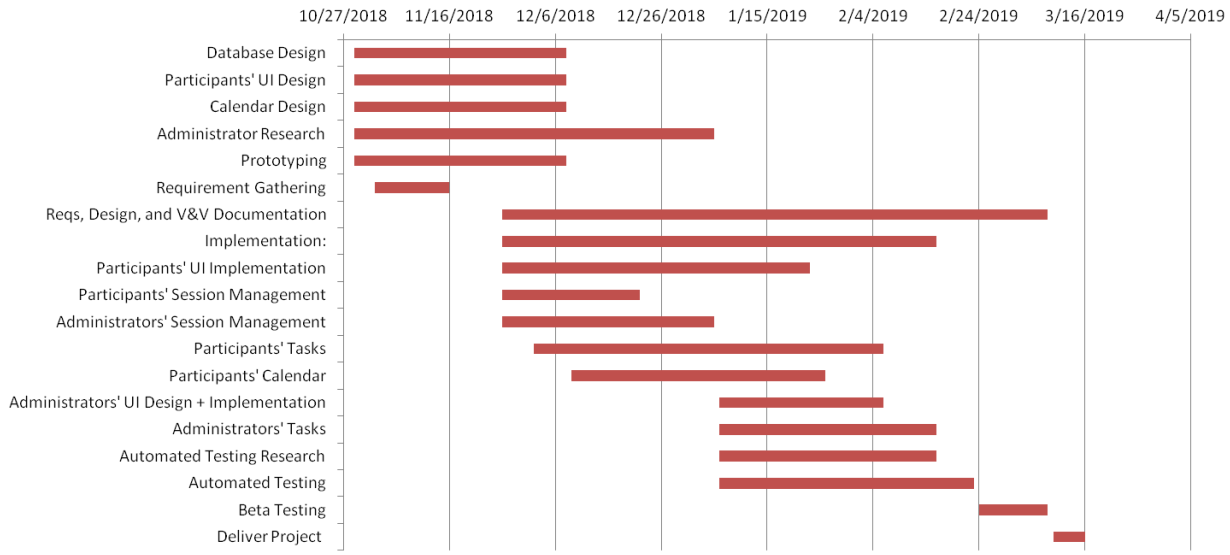
Roles and Responsibilities of Team Members

Below is a table explaining the primary and secondary role for each member. Each member has a primary and secondary role while two-thirds of the team has tertiary roles. Primary and secondary role are of the same importance and time should be dedicated to them as needed to get the tasks done. Tertiary Role usually requires less time commitment, but members should allocate time for it as needed.

Member	Primary Role	Secondary Role	Tertiary Role
Fadi Labib	Project Manager	Risk Management Analyst	Product Owner
Michael Cohoe	Developer	Backend Architect	Q&A Analyst
Thong Tran	Developer	Frontend UI Architect	Q&A Analyst
Christopher Bartlett	Developer	Infrastructure & Hosting Engineer	Q&A Analyst
Millen Wan	Developer	Security Researcher	Q&A Analyst
Minwei Luo	Developer	Q&A Analyst	N/A
Minh Nugyen	Developer	Q&A Analyst	N/A

Major Milestones and Deliverables

Below is the Gannet Chart for major milestones and deliverables of the project. Below the chart is brief description of the milestone and explanation of the deadlines.



Database Design

This is designing the schema for all the tables needed for the app, setting up the database and implementing all the tables in the setup database.

Participants UI Design

This is designing the general layout that is consistent to all the pages that will show up for a participant logged in to the app including all navigation bars and containers as well as the design that is specific to each page visible to a participant in the web app.

Calendar Design

This is designing everything that has to do with the calendar that shows upcoming events and deadlines for workshops sponsored by MESA. As well as the displaying of special events, that will have a countdown element to them.

Administrator Research

Researching ways to implement an administrator side to the app and testing that it works when integrated to the current user side. The success of this research will make building an administrator side to the web app possible. This task extends to winter break so the person in charge of this task has to be able to work over winter break. The last week of December, which is the third week of winter break, is off so the amount of time allocated for administrator research is one week less than the time shown in the gannet chart.

Prototyping

This milestone involves building a simple web app that logs in user and logs them out as well as allows them to reset their password and remind them of their username. That simple app will include some basic UI elements like the landing page for registration and signing in as well as the general layout with all navigation bars in place and showing some upcoming test events in a calendar along with countdown for events coming up for a given week. The prototype will be demoed during the midterm presentation.

Requirement Gathering

This milestone includes the deadline to finalize all meeting with client to start and formalize all the requirements and UI expectation in order to draft a requirement document.

Reqs, Design, and V&V Documentation

This milestone is writing the actual requirement document and some parts of the design and V&V documentation based on requirement gathered in the previous milestone. It also includes iteration over those documentations as developers implement more features and thus it spans both the implementation and testing stages.

Implementation:

This milestone is the overall period allotted for implementing the web app including support for an administrator side; it is broken down to simpler milestones listed below.

Participant UI Implementation

This milestone is the actual implementation of the participant's UI, as it will look in production. It will be done based on the prototype built in an earlier milestone. This milestone also includes the implementation of the link to all the pages visible to a participant. This stage ends with all the navigation bars working and directing participants to the correct pages in the web app.

Participant's Session Management

This milestone includes registering a participant, signing them and signing them out. It also includes resetting their passwords sending them email to remind them of their usernames, deactivating their account, and editing basic profile features. The stage ends will all the tasks above working correctly.

Administrators' Session Management

This milestone includes declaring a single account as administrator and giving it credentials that will be only know to MESA representatives that will allow them to sign in and sign out. It will also allow them to reset their password, and reminds them of their username by sending emails to the email address tied to the admin account. The stage ends will all the tasks above working correctly.

Participants' Tasks

This milestone includes all the features that participants will be able to access upon their successful authentication. Multiple people will implement this section and it will end with all participants features, enumerated in the requirements document, working correctly.

Participants' Calendar

This milestone is every feature delivered to successfully authenticated participants that has to do with calendar; a detailed list of those features is available in the requirements document.

Administrator UI Design + Implementation

This milestone is finalizing the UI design for the administrator side of the web app and implementing it based on the participant UI design but with different links to all the pages visible to an administrator. This stage ends with all the navigation bars working and directing administrators to the correct pages in the web app and the pages are not visible to unauthenticated users and users who authenticate as participants.

Administrator Tasks
This milestone includes all the features that administrators will be able to access upon their successful authentication. Multiple people will implement this section and it will end with all administrators features, enumerated in the requirements document, working correctly.

Automated Testing Research
This is the time allocated for all developers to start researching how to implement automated testing. It is recommended that developers start working on this task earlier than the time set above in order to reduce the volume of work they have to deliver in the allotted amount of time.

Automated Testing
This milestone starts at the same time as automated testing research as both learning and developing the actual test will happen simultaneously, but an additional week is given for this milestone in order to allow for a slack space for developers to really understand how to create automated tests.

Beta Testing
This milestone may or may not happen depending on whether or not the client brings in some of their students and have them test try the app. If it will not be possible then this time will be used by us the developers to do more testing and fix bugs.

Deliver Project
This is the final milestone and it involves hosting the product in production server if it makes it to that stage and delivering to the client at least the minimal viable product hosted in a testing server with limited access.

Work Breakdown Structure
This section deconstructs the milestones and major deliverables in the web app into smaller chunks and assigns them to team members.

Michael Cohoe Tasks
- Participant: View Current Obtained Stamps
- Administrator: Alter Badge Max Level (Adding More Levels)
- Administrator: Viewing Single Student (Search and Display)
- Administrator: Exporting Data to Excel Research + Implementation
- Administrator: Removing Old Accounts
- Documentation
- Testing

Christopher Bartlett Tasks
- Calendar UI Implementation
- Calendar and Event Management
- Viewing Countdown for Important MESA Dates
- Viewing Upcoming Tasks Based on Badge
- Documentation
- Testing

Minh Nugyen Tasks
- Participants: Session Management
- Participants: View Current Level
- Administrator: Adding School
- Administrator: Adding Extra Badges + Stamps
- Administrator: Alter Point Required for Badge level
- Administrators: Removing Schools
- Administrator: Removing Badges & Removing Stamps
- Documentation
- Testing

Thong Tran Tasks
- Landing Page Implementation
- Participants Layout Implementation
- Dashboard Implementation
- Administrator Layout Implementation
- Implementing All Pages Routes
- Implementing Resetting Password Page
- Documentation
- Testing

Millen Wan Tasks
- Administrators: Session Management
- Participant: Replicating work in all the badges
- SQL Injection Mitigation
- Documentation
- Testing

Minwei Luo Tasks
- Participants: All tasks except removing stamps
- Administrator: Removing Stamps
- Administrator: Viewing Top Three Per Badge
- Administrator: Setting Academic Year Date
- Administrator: Deleting User Account
- Documentation
- Testing

A Detailed deadline breakdown for these tasks is included in a supplemental document shared with the developers and is not part of this documentation.

Risk Management Plan

There are multiple risks associated with this project. While it is very hard to enumerate every possible disaster or hiccup that impedes project progress, below is a list of predicted failure each assigned a probability and a severity level along with trigger, response, and contingency plan in the case that the perceived risk becomes a reality.

Client Disappears

Description Summary	Probability	Severity	Risk
The client disappears all the sudden and all communication attempts fail	2	4	8
OWNER(s):	Fadi Labib		

Trigger Event(s) & Indicator(s)

Not hearing from the client after contacting them three or more times trying different communication methods and giving them a chance to respond. The chance to respond will be based on the urgency of communication.

Risk Response Description

The alternate client will be our point of contact in regards to project deliverables. For deliverables that the alternate client does not know about it, the project manager and the developers will use their discretion to build the product and deliver it to the alternate client.

Contingency Plan

The project manager may decide to take away some of the deliverables that are not part of the minimal viable product (MVP) and will be updating the alternate client about any decisions they made in regards to the final deliverables of the project.

Team Member(s) Leave Capstone

Description Summary	Probability	Severity	Risk
Team member decides to leave capstone or is removed from the team	2	4	8
OWNER(s):	Fadi Labib		

Trigger Event(s) & Indicator(s)

One or more members of the team start missing many deadlines or not show up for two or more all team meetings.

Risk Response Description

The project manager will put in all the effort to the extent possible to reach the member(s) and figure out if they actually left capstone or if there is some sort of critical emergency. Capstone faculty members will be notified of such issues immediately, as it becomes known. The project manager will start planning to divide the tasks among the leftover team members and may take certain deliverables out of the project that are not part of the MVP.

Contingency Plan

All team members will be notified about it as soon as a capstone faculty clears the sharing of such information. Due to privacy limitation, the project manager may not share the reason behind a team member leaving the team. The project manager will shift deadlines accordingly and may end up removing tasks that are not part of the MVP in order to balance the load without affecting other team members or giving them too much work.

Losing Code Base

Description Summary	Probability	Severity	Risk
The repository containing the codebase is lost	1	4	4
OWNER(s):	Fadi Labib, Tong Tran, Michael Cohoe		

Trigger Event(s) & Indicator(s)

The project manager checks with team and finds that backup is off schedule or not done according to the "project code backup plan" mentioned later on this document. An even more catastrophic trigger is finding that the code repository has been overwritten or lost.

Risk Response Description

The project manager will contact appropriate team members according to the procedure in "project code backup plan" and the owners of this risk will make all efforts to fill in missing backups. In the case that the code repository is deleted or overwritten, the procedure in the "project and code plan" will be followed to try to recover as much of the code as possible. Owners assigned to this risk must insure all backups are done according to the procedure in " project code backup plan" and be able to recover as much of the data as possible in the case of it being lost.

Contingency Plan

After owners respond to the risk, the person who lost the data, to the extent possible, will try to recreate any unrecoverable data. If necessary, owners will contact appropriate capstone faculty member to see if the data is recoverable. The project manager may resort to using basic forensic tools to try to recover data. If all efforts to recover data fail and the missing parts cannot be recreated, one of two things will happen depending on the type of data that is unrecoverable. First, for any unrecoverable data relating to tasks that is part of the MVP, it will have to be recreated and non-MVP tasks will be removed as necessary to allow more room for MVP tasks to be recreated. Second, for any unrecoverable data relating to tasks that is no part of the MVP, they will be canceled and development will continue regularly following the regular schedule.

Admin Session Management Failure

Description Summary	Probability	Severity	Risk
Failing to implement admin account because of complexity/deadline concerns	4	2	8
OWNER(s):	Fadi Labib, Michael Cohoe, Millen Wan		

Trigger Event(s) & Indicator(s)

By second week of winter break, the person assigned the task of figuring out how to add admin to the web app declares it not possible.

Risk Response Description

Project manager will be talking with the person in charge of implementing admin session management, who is also one of the owners of this risk, to see why it fails. The owners will then come up together with alternative solutions to get it working and implement any changes to the code as necessary (even if it involves modifying other people's code).

Contingency Plan

If it appears that, the failure of admin side implementation is due to design restriction or that coding is too complicated and missy that it will start making other deadline fall behind, the project manager will drop the admin side from the set of final deliverables. The client is already aware that admin is a stretch goal and that it may not work or be too risky to deliver at such an early stage of the app development.

Team Member(s) not Meeting Deadlines

Description Summary	Probability	Severity	Risk
Team member starts missing multiple deadline or refusing to do work	4	4	16
OWNER(s):	Fadi Labib		

Trigger Event(s) & Indicator(s)

The first time a team member misses a deadline.

Risk Response Description

For first time missing a deadline, it will be noted in the closest all team meeting notes and a new deadline will be negotiated as long as that new deadline will not take away from other tasks or hinder the progress of the project. Second time a deadline is missed the project manager will grant extension up to the closest upcoming all team meeting and whether it gets done or not will be noted in the all team meeting with how many days the deadline has been extended and the result of either completed or missed. The project manager will also set a one on one meeting with the person who had a second penalty missing a deadline and do two things. First have a recorded discussion with the member about why deadline is missed and what is the best way to avoid this from happening a third time. The recording will be kept confidential and will only be shared with capstone faculty members upon request. The project manager may also share it with capstone faculty when a third deadline is missed. Second, the project manager will break all upcoming tasks for that member to small hourly tasks. The third time the same member misses a deadline, after breaking down the tasks, will results in having all notes about the missed deadlines forwarded to the capstone faculty in charge of the capstone that this team is part of, and it will be up to him to declare how things will go forward.

Contingency Plan

The project manager, to make sure the overall team progress is not hindered, due to the failures of a single person, will implement an alternate plan the moment the first deadline is missed. This alternate plan is the project manager exercising his right to remove certain tasks, switch "undone tasks" of two different people, or reshuffle tasks around so that critical tasks are guaranteed to be done.

Web App Production Hosting Failure

Description Summary	Probability	Severity	Risk
Failing to get a domain to deploy the website in production mode	4	3	12
OWNER(s):	Fadi Labib, Christopher Bartlett		

Trigger Event(s) & Indicator(s)

Client notifies team lead that they do not have the ability to get a domain.

Risk Response Description

The owners of this risk management plan will meet with client and take hosting out of the requirements. Client is already made aware that if they cannot provide a domain, web app cannot be hosted.

Contingency Plan

The developers will continue to build the code as a proof of concept and turn it with instruction on how to run it as well as good comments to explain what it does so that someone can pick it up and host it in the future or even add more features to it.

CAT Infrastructure Stops Working

Description Summary	Probability	Severity	Risk
CAT updates and the likes may break our app impeding work progress	4	3	12
OWNER(s):	Fadi Labib, Christopher Bartlett		

Trigger Event(s) & Indicator(s)

While running the app we figure that it does not run or suddenly while testing the app we find the app out of service with a 404 error.

Risk Response Description

Team member in charge of infrastructure support will contact CAT immediately to see what they can do about it. Deadlines will be pushed as needed if it is severe enough that it affects deliverables deadlines. If the impact is severe enough, certain deliverables that are not part of the MVP will be declared out of scope by the project manager.

Contingency Plan

The developers will run the app locally and a local database will be setup to continue the app away from CAT infrastructure. Tasks not part of the MVP will be cut out to allow for rebuilding the infrastructure without adding extra un-necessary load on team members.

Failure to Receive Needed No-Reply Email

Description Summary	Probability	Severity	Risk
Client doesn't provide us a no-reply email to use to send emails to users	1	2	2
OWNER(s):	Fadi Labib		

Trigger Event(s) & Indicator(s)

Client notifies team lead that they do not have the ability to create a no reply email.

Risk Response Description

The developers will set up a regular Gmail account and share its login credentials with the client and the account will be used to generate emails. It will be the responsibility of the client to maintain this email.

Contingency Plan

No contingency plan necessary as risk is too low.

Failure to Receive Needed Administrator Account

Description Summary	Probability	Severity	Risk

Client doesn't provide us an email that gets admin rights to the web app	2	4	8
OWNER(s):	Fadi Labib, Millen Wan		

Trigger Event(s) & Indicator(s)

Client notifies team lead that they do not have the ability to create an account that the developers can tie to the administrator account of the web app.

Risk Response Description

The owners of this risk management plan will meet with client and give them one of two options. First, the developers create an admin account for the client and the client has to maintain that account. Second, the client is notified that administrator account is not part of the final deliverable. Client is already made aware that admin account is a stretch goal, but in this case, they will be notified that it is actually off the table.

Contingency Plan

For testing purpose, one of the owners will create a local Gmail account and tie it to the admin account for the app to be able to implement some of the admin features that are of the most importance. This way, if client is later, before the ending of the developing stage, able to finds a way to create an email account that they can use as the admin then developers will use that email to build the admin account. Since most of the admin features will already have been implemented by following this contingency plan, the developers will be set to exceed the minimal goal and deliver important admin functionalities that the app desperately needs.

Failure to Receive Needed Design Materials

Description Summary	Probability	Severity	Risk
Client doesn't provide us needed material to design the web app to their liking	1	2	2
OWNER(s):	Fadi Labib, Thong Tran		

Trigger Event(s) & Indicator(s)

Client notifies team lead that they do not have the ability to get certain needed design materials.

Risk Response Description

Most of the need design materials has been already delivered by the client so there is a very low probability the rest of the needed design materials will not be delivered. However, if this ends up being the case then we will use what we have to build the web app and deliver it tailored to the organization as much as possible with design elements we have.

Contingency Plan

No contingency plan necessary as risk is too low.

Communications Plan

This section includes the policy and procedures for communicating will all stakeholders of this project including capstone faculty members, the project manager, developers, representative of Oregon MESA (our client), and representatives of the Computer Action Team (CAT).

Communication with Capstone Faculty Members

Developers may contact a capstone faculty member as needed, but for project related inquires, they should follow the chain of command and go through the project manager first. All developers should be aware that capstone faculty members are mandatory reporters and filter their communication with them accordingly.

Project manager communicates with a capstone faculty member in daily bases to give status of their work as well as the team progress and issues via Slack. Project manager also meets either weekly with a capstone faculty member face to face or through slack to discuss team management related topics and upcoming deadlines that are then disseminated to team members. Project manager may also contact a capstone faculty member as needed and may ask for their intervention in project related problems.

Communication with Project Manger

Developers may contact the project manager as needed. Developers can reach the project manager via slack, email, phone, or in-person contact in campus.

Developers also meet with project manager once or twice a week in person, via Slack or Google Hangout in order to discuss project progress and align the team in upcoming tasks. Developers also report daily to project manager what they have done the day before and what they plan to work on today preferably in the morning before noon, but no later than 11:30 PM to give project manager a chance to read them and respond accordingly. All developers should be aware that the project manager is a mandatory reporter and filter their communication with him accordingly.

Clients and members of the CAT team my contact the project manager regarding any concerns via either phone or the university email.

Communication among Developers

Developers may contact other developers as needed. It is encouraged that developers communicate with each other to solve issues before escalating it to team lead. Developers can communicate with each other via slack, email, phone, or in-person contact in campus.

Developers may call an all team meeting if necessary but project manager must be present. Developers may meet with each other without having the project manager present but they must let project manager know about and keep notes about what they have discussed and send it to the project manager.

Communication with Client

Project manager must be present in all face-to-face client meetings and must always have at least one but no more than three team members with him. Developers may not initiate a meeting with client or contact them without letting the project manager know about it first and then the project manager decides whether it is ok for the developer to ask the client directly or if he must initiate the communication himself. Project manager may contact client at anytime as necessary but for project related stuff only using communication methods that the client has approved which in this case are in-person, email, or phone call.

Communication with Computer Action Team (CAT)

Project manager and/or Infrastructure & Hosting Engineer must be present in all face-to-face meeting with any representative of the CAT team if it is in regards to the project. Project manager and Infrastructure & Hosting Engineer must always have at least one but no more than three team members with them in all such communications. Developers may not initiate a meeting with a representative of the CAT team relating to the project or contact them regarding such matters without letting the project manager and Infrastructure & Hosting Engineer know about it first. Project manager or the infrastructure & hosting engineer then decide whether it is ok for the developer to ask the client directly or if they must initiate the communication themselves. Project manager or infrastructure & hosting engineer must always notify each other about developers initiated communication with CAT representative that relate to the project and explain the purpose of the communication as soon as possible. Project manager may contact CAT representatives at anytime as necessary, but for project related stuff, they must notify the infrastructure & hosting Engineer as soon as possible preferably before the meeting, but after it is fine. Same expectation holds true for the infrastructure & hosting engineer. Project manager and infrastructure & hosting engineer should use communication methods that CAT has approved which currently is face-to-face or through the CAT support email and any other method available to the infrastructure & hosting engineer.

Stakeholder Management Plan

Stakeholders in this case are MESA representatives who asked us to build the app or any affiliates they include in any of our communications. They will mostly be receiving information and updates regarding the project implementation and testing progress. Project manager will communicate to stakeholders any decisions that will directly affect the requirements set by them, at least in writing, and he will ask them to provide some form of a written input of their decision. Developers will ask stakeholders to provide UI related material needed for the website including high-resolution pictures, logos, colors, pre-set avatars for user profile pictures, and stamps design. Developers will also ask stakeholders to buy a domain and provide us that domain name if they want to have their website hosted in a production environment. Developers will also ask stakeholders to provide a no reply email that they set up and that will be used to send reset password link and username reminders. Developers will ask them to provide an email that they create which the developers will tie to the administrator role and an e-mail associated with the Google calendar that they update with all MESA events and that will be visible on the web app. Developers will work with stakeholder to find student participants to beta test the app.

Project Code Backup Plan

This section of the plan defines how individual branches will be backed up as well as the master branch of the github repository containing the source code.

Backup Individual Branches

Each team member, will be required to backup their own branch daily between 10:00 PM and 11:59 PM even if no changes were made to that branch. The backup will be in a folder with the member's last name followed by the date and time the backup was created. The format shall be as shown here: Lastname_mmddyy_hhmm. For team members who don't own their own device they will be required to

backup their own branch whenever they finished working on their branch for the day using a flash drive and following the naming convention indicated earlier.

On Saturday night at 10:00 PM project manager will receive a zipped directory (.zip or .rar) using the school's Gmail account containing the entire backup for the week from Sunday to Saturday for each team member. If the code repo contains certain scripts that Gmail would flag as malicious then the deadline will be Monday during the team's meeting, that is part of the capstone class time, and the delivery method will be some external media. In that case, it will include all backups since the last date that the project manager has received backup for the individual branches.

Team members shall immediately report any incident that involves the deletion of backed up data to project manager, frontend UI architect, and backend architect. The data will be recovered up to the most recent week of update that the project manager has for the team member. Any branch data that were not part of the backup will have to be recovered or recreated by the branch owner.

Backup Master Branch

Project manager as well as frontend UI architect and Backend architect shall backup the master branch every time changes are merged to it regardless of who merged the changes. Anytime changes are merged to master, the project manager as well as frontend UI architect and Backend architect must know about it and backup the branch as soon as possible using the naming convention shown here: Lastname_mmddyy_hhmm. Like the individual branches, the project manager will back up the master branch to an external drive in addition to the copy on his own computer. No backup for the master branch shall ever be deleted from the project manager, frontend UI architect, and backend UI architect computers until the project has been delivered to the client and the team has been cleared of all responsibilities related to the project. If for any reason the project manager, frontend UI architect, and backend UI architect lost their data they should notify the other two immediately that this is the case and one of the two who have their data intact will re-send their backup to the person who lost it.

Archiving Project Related Communications Plan

This section describes how different communication methods related to the project will be archived.

Slack Communications

All slack communication is archived on slack itself.

Face-to-Face Communications

All face-to-face communication is recorded in some form of a written document.

All Team Meetings

For all team meetings one person is randomly picked from the team to take notes using a pre-built template created by the project manager and once everyone in the team had a chance to take notes, including project manager, everyone is entered back to the drawing again and a person is randomly picked to write the notes. All team meetings' notes are reviewed by the project manager, modified as necessary, and then shared with the whole team. Team members should all archive it on their devices for

their reference. Project manager archives all team meeting notes on his own device. The naming standard for all team meetings notes is MESA_Meeting_mmddyy.

Sub Team Meetings

For other meeting that involves few members of the team one of the attendees is assigned notes duty by the project manager, alternatively the project manager sometimes takes the notes himself. For those meeting that notes can be taken in a regular text editor program and does not have to follow in template or format. For meetings that do not involve the project manager one of the attendees, should take notes during the meeting or shortly thereafter and share their notes with the project manager. Everyone involved in the meeting should archive those notes and if project manager is not present, someone must send him a copy of the notes so he can archive it as well.

Client Meetings

Client meetings follow the same rule as sub team meetings in that anyone of the attendees, must take notes and the notes can be taken in format and using any text editor. Everyone involved in the meeting should archive those notes.

CAT Meetings

CAT meetings follow the same rule as sub team meetings in that anyone of the attendees, must take notes and the notes can be taken in format and using any text editor. Everyone involved in the meeting should archive those notes.

Email Communications

All email communication between the project manager and developers, MESA representatives, or CAT representatives are archived in the email itself and saved as a PDF on the project manager's device. All email communication not involving project manager should be backed up by the people involved in the communication and be ready to share it with project manager or a capstone representative upon request.

Phone Communications

Either all phone communications related to the project must be voice recorded, and everyone in the meeting is aware of that recording or written in some form of a document immediately at the end of the phone call and saved by the people involved in the communication. It does not matter who take the notes as long as someone takes them and send those notes to all other parties involved in the communication. Those notes should be backed up by all parties involved and be available to the project manager or a capstone representative upon request.