NYU TANDON SCHOOL OF ENGINEERING

Computer Science and Engineering

EcoFoodie

Project Management Plan

Version 1.0

Document Number: SPMP-001

Project Team Number: B23

Project Team Members:	Vijay Kallem	vrk248
	Nathan Cantu	snc387
	Jonathan Xu	jx990

REVIEW AND APPROVALS

Printed Name and Title	Function (Author, Reviewer, Approval)	Date	Signature
Nathan Cantu	Author, Reviewer, Approval	10/11/2022	Nathan C
Vijay Kallem	Author, Reviewer, Approval	10/11/2022	Vijay K
Jonathan Xu	Author, Reviewer, Approval	10/11/2022	Jonathan X

REVISION LEVEL

Date	Revision Number	Purpose
10/11/2022	Version 1.0	Initial Release
10/11/2022	Version 1.1	Updates/Defect Corrections

Table Of Contents

1. OVERVIEW

1.1 Project Summary

The priority of this service is to allow users to share their leftover produce, leftover unopened food items, leftover ingredients, and homemade meals with their community and neighborhood. Every offering by a supplier (a user who has registered their SSN) is free of cost to every other user. The goal is to help mitigate food insecurity across the city of New York, broadening wider as the service progresses, for families or individuals who are low income but do have an internet capable device.

The document is the software project management plan for the application, EcoFoodie. The document will outline the organization of the project and plans, the management processes and technical processes. Development teams and clients are the intended audience of the SPMP.

1.2 Purpose, Scope, and Objectives

The objective of this project will be to create a software application that will allow users to give their leftover unopened food items or produce to those within a certain radius.

All users for the initial testing phase will be within NYC. A supplier may use EcoFoodie to upload meal offerings accompanied by pictures of the food, a video of the preparation of the food, and a description of ingredients and allergy information. Consumers may use it to reserve food offerings, leave reviews, add other users as friends and create groups, and file complaints against suppliers. Users can send out directed alerts to groups and avoid encountering strangers. All users and suppliers have access to a terms of conduct page dictating the rules that every supplier must follow in the preparation of and distribution of their food.

This release is intended specifically for the development team to starting planning the development of the product

1.3 Assumptions and Constraints

.
The following assumptions and constraints will be made/used in this document:

- The deliverable product needs to be dependable.
- Deadlines will need to be met.
- The software back-end/front-end must be built to support additional features being added on at a later date.

1.4 Project Deliverables

Project Team Selection Form: 9/14/22

Project Proposal: 9/21/22

Software Analysis/Requirements Specification (SRS) : 9/26/22

Software Project Management Plan (SPMP) : 10/11/2022

Project Description : 10/12/2022

Initial Software Design Description : 10/17/22

Final Design Document : 11/28/22 -12/14/2022

Implementation: 11/28/22 -12/14/2022

Presentation and Demonstration: 11/28/22 -12/14/2022

1.5 Schedule and Budget Summary

A summary of the schedule is as follows:

- Requirements workflow (3 weeks) (complete)
  - Software Requirements Specification
- Project Management workflow (2 weeks) (complete)
  - Software Project Management Plan
- Design workflow (2 weeks) (incomplete)
  - Project Description
  - Design Description
- Implementation (6 weeks) (incomplete)
- Testing Workflow (1 week) (incomplete)
- Presentation and Demonstration (1 week) (incomplete)

The total development should take 12 weeks.

2 Evolution of the Plan

Any change of plan must be mutually agreed upon by the three team members. Once a change is planned, the document will be updated and distributed again to the audiences.

3 REFERENCES

- Eco-Foodie, B23
- Project Team Selection Form Version 1.0: 9/14/22
- Project Proposal Version 1.0: 9/21/22
- Software Analysis/Requirements Specification (SRS) Version 1.0 : 9/26/22

4 DEFINITIONS

EcoFoodie - the main product and deliverable

App - the application of the product

SRS – Software Requirements Specification

SPMP – Software Project Management Plan

Client – a participating user, restaurant, or venue

5 PROJECT ORGANIZATION

5.1 External

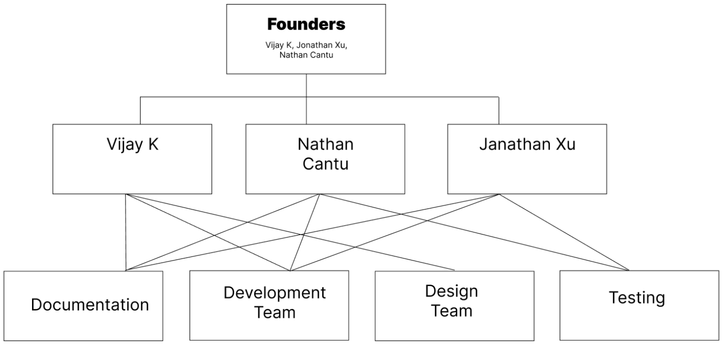
The Software Quality Group will review the development team’s implementation and make suggestions or alterations. The Project Office controls all projects within the organization.
The development team will be meeting the pilot clients, whoever they may be, at the end of the development and begin the testing process

5.2 Internal Structure

Nathan, Vijay, and Jonathan will make up the development team. All three will be working full stack with Nathan being the lead developer for the back-end while Jonathan and Vijay work primarily front-end. The internal structure will grow to accommodate more as the company grows.

5.3 Roles and Responsibilitie s

Vijay and Jonathan will work on the development of the front end part of the product. Nate will focus more on the backend and design the architecture and logics. Nathan will work on setting up the database and Jonathan will assist Nathan and Vijay on integrating all parts together. Nathan will upload all documents to NYU Brightspace per the deadlines.



6 MANAGEMENT PROCESSES

6.1 Start-Up Plan

The development team was self-selected. The project was then selected and proposed. The proposal came from the client.

6.1.1 Estimation Plan

We plan to estimate based on our prior experience on working on teamed software projects. We will also research other projects to understand their estimations and get an idea on how we would have to proceed with the plan.

6.1.2 Staffing Plan

All the team members have the required skills to build the application. We will not need any additional staff. The team is also learning skills that may help build the project faster. The present staff is well skilled and well equipped to precede. The team has built a plan to divide the tasks to work on the application.

6.1.3 Resource Acquisition Plan

All the resources we need to build the application are open sourced or are available at free of cost. If needed, the team would request money from the class budget if items are needed for testing. We are also planning on applying to the prototyping fund. The team would also exchange resources with each other.

6.1.4 Training Plan

The team has started training in certain frameworks using available resources. The team would gain certain skills if needed at any stage of the development by using resources available to us through the Internet.

6.2 Work Plan

Please refer to the Gantt Chart in 13.3 for a breakdown of all work activities.

6.2.1 Work Activities

Please refer to the Gantt Chart in 13.3 for a breakdown of all work activities.

6.2.2 Schedule Allocation

Please refer to the Gantt Chart in 13.3 for a breakdown of all work activities.

6.2.3 Resource Allocation

Please refer to the Gantt Chart in 13.3 for a breakdown of all work activities.

6.2.4 Budget Allocation

Budget allocation for each stage of development relies on the hourly wage of each employee. Each employee gets paid $18 an hour regardless of position.

Documentation: Hourly cost of wages

Implementation: Hourly cost of wages, potentially the cost of a cheap Android phone to run our mobile application on

Testing: Hourly cost of wages

6.3 Control Plan

6.3.1 Requirement Control and Traceability

Traceability is the primary method of controlling requirements that will be used. Each requirement has a designated number assigned to it. Each number has a list of other requirement numbers that it relies on (backwards) and a list of other requirement numbers that rely on it (forward).

6.3.2 Schedule Tacking and Adjustment

The development team meets at least twice a week on Wednesday and Friday to give progress updates and anticipate deadlines. A status chart will be used and we will use a weekly agile report. Any adjustments needed to the schedule will be made. The project schedule would then be redistributed.

6.3.3 Budget Tracking and Adjustment

Each week, the anticipated cost per planned budget will be compared to the week's budget spent. All time costs will be analyzed, and if needed corrective actions will be taken to fix any issues causing budget excess.

Cost reporting will occur every week and for the sections of documentation, implementation, and testing all will have their own cumulative costs occurred during the period of each section.

Earned value tracking will be in place to compare anticipated completion against actual completion and cost spent

6.3.4 Quality Control

We will continually test all functionality and the expected output from all scenarios. We will execute the tests and report any defects.
Quality inspections of all documents will be made before submission in order to ensure consistency and completeness.

6.3.5 Reporting Mechanisms

The developer team systematically will report any information to one another through a shared Discord server. Reporting documents or code will use NYU Brightspace. Reporting project status to the teaching assistant will be through Zoom. The final presentation will be shared and presented through Google Slides.

6.3.6 Metrics Collection Plan

Project metrics will be taken continually with each iteration of the documentation and implementation. Metrics include on time delivery, quality of work, number of defects, and amount of time worked for each team member. For code, metrics include run time, memory space, and overall quality. These metrics will be taken as applicable before every project deliverable submission.

6.4 Risk Management Plan

Business Risk:

Description: Advertisements as the primary method of monetization may not be enough to cover the costs. Especially if the project evolves into having screens placed around the city, those costs plus transportation costs for drivers to head to those locations may be too expensive

Probability: Very likely

How Discovered: If the total costs starts exceeding the total revenue

Responsible Party: Product team

Status: Not currently a threat

Mitigation Plan: Introduce more methods of monetization, currently undetermined

Operational Risk:

Description: Users can become ill, sick, or fatally harmed due to poisoned or contaminated dishes offered up on the app

Probability: Extremely likely

How Discovered: If someone complains on the app or directly to the team or if law enforcement notifies the team

Responsible Party: Product Team and the perpetrator

Status: Major current problem

Mitigation Plan: Uploading the video required of them preparing the food alongside pictures of the food, the complaint feature, and suppliers having to register their SSN, should make it easier to hold the supplier liable. This accountability will hopefully prevent suppliers from acting out.

Technology Risk:

Description: One risk is if someone hacked into the system and sent out false alerts to everyone in the area. They would not be registered with a SSN and they could use this feature for extreme harm towards the users.

Probability: Possible

How Discovered: Law enforcement notification, seeing it through our servers

Responsible Party: Product team and hacker

Status: Always extremely important

Mitigation Plan: The mitigation plan is to implement security cryptographic measures to ensure that nobody can reasonably cause corruption

Economic Risk:

Description: Due to the operational risk of one of the users being injured by the contaminated food or by technology risk of being lured by ill intentioned people, the service could be sued.

Probability: Likely

How Discovered: Notification by an attorney that the company is being sued

Responsible Party: Ultimately the product team but the prosecution is the one suing

Status: Of concern currently

Mitigation Plan: Have users agree to a terms and service contract that does not place liability on the service

6.5 Post Implementation Plan

The primary software deliverable is November 28th, but the software development will be broken down into sprints where each week one or more major tasks must be completed. All of the software will be archived through GitHub. To prepare for the PIR, we will continually monitor mistakes, flaws, system shortcomings and errors throughout the development process.

7 TECHNICAL PROCESSES

7.1 Process Model

We will be using the object oriented methodology and our service will follow a waterfall incremental model as stated in the initial SRS.

7.2 Methods, Tools, and Techniques

The method used will be UML (unified modeling language) and the tools used include Github, Trello for organization/planning, Google Docs for documenting, React Native for front end development, Python Django for back end development, and PostgreSQL for the database. The programming languages used will be Python, SQL, and JavaScript.

7.3 Infrastructure Plan

In order to develop the product, we will need a Windows computer with React Native, Python Django, and PostgreSQL downloaded. In order to test the product, we will need a phone running Android/IOS. We will also require a server to store the information after deployment.

7.4 Product Acceptance and Migration Plan

We will start at feature testing and go to integration testing. Once the development is complete, we will test the application by giving access to our application to some of our target customers. We will make changes to the product until the product gets complete acceptance by our target customers.

8 SUPPORTING PROCESSES PLANS

8.1 Configuration Management Plan

Everything will be stored and managed through github, where we will have a repository for everyone to access. Changes and all modifications will be tracked through there too and any time someone commits to the repository, there will be comments detailing everything that has been added/changed/deleted.

8.2 Qualification (Verification and Validation) Plan

Any time a feature/requirement is added to the software, one will proofread and edit the code, checking for any edge cases, while the other will test and debug any other issues that come up. That way, every member will be involved in the process and each will look at the code fresh.

8.3 Documentation (library) Plan

All documents created will be stored within the github repository in its own folder. They will be written by all members, though each will have a specialization in different categories. Each member will then review another's section and then together, go over it all once more to approve what has been written. Each document will also have a unique reference number for accessibility.

8.4 Quality Assurance Plan

Tests will be run every other day on different features and all bugs will be added to a list. At the end of the week, members will all get together to work on the bugs, starting with the most important. That way, when the software is delivered, the code will be as functional as possible without any noticeable issues.

8.5 Reviews and Audits

Along with the weekly testing and fixing, the list of defects will be ordered from most important to least important and each will be assigned to a member of the team.

8.6 Problem Resolution Plans

Each problem/defect will be tracked and have a status (in progress/completed/inspected/distributed). These will all also be documented in Github for everyone to see. Whenever a defect is undergoing revision, testing will sometimes be replaced by revision if the issue is big and requires more time.

8.7 Environment Management Plans

Once the product is released, we will do an audit to find if there are any problems in the environment. We will continue to make these audit regularly. We will also get feedback from our customers and make updates according to their acceptance. We will maintain the same development environment after the project is deployed so as to continue performing bug fixes. We will all continue to work as environment management.

8.8 Process Improvement Plan

We will make note of every problem we come across, analyze the codebase to determine the root of the issue, and document the problem and our solution for future reference. This will make the codebase more memorable and as we develop these problem solving techniques, we should be able to debug issues at a greater pace.

9 ADDITIONAL PLANS

We will make additional plans according to problems we come across in the development process and all the plans will be _well documented.

10. INDEX

11. RATIONALE

None

12. NOTES

None

13. APPENDICES

13.1 Schedule Tracking

Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
SRS	Nathan Cantu	6 hours	10 hours	4 hour
	Jonathan Xu	5 hours	8 hours	3 hours
	Vijay Kallem	5 hours	6 hours	1 hour
	Summary for entire team	16 hours	24 hours	8 hours

Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
SPMP	Nathan Cantu	5 hours	6 hours	1 hours
	Jonathan Xu	4 hours	6 hours	2 hours
	Vijay Kallem	4 hours	4 hours	0 hour
	Summary for entire team	13 hours	16 hours	3 hours

Cumulative

Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
Cumulative	Nathan Cantu	11 hours	16 hours	5 hours
	Jonathan Xu	9 hours	14 hours	5 hours
	Vijay Kallem	9 hours	10 hours	1 hour
	Summary for entire team	29 hours	40 hours	11 hours

13.2 Defect Tracking

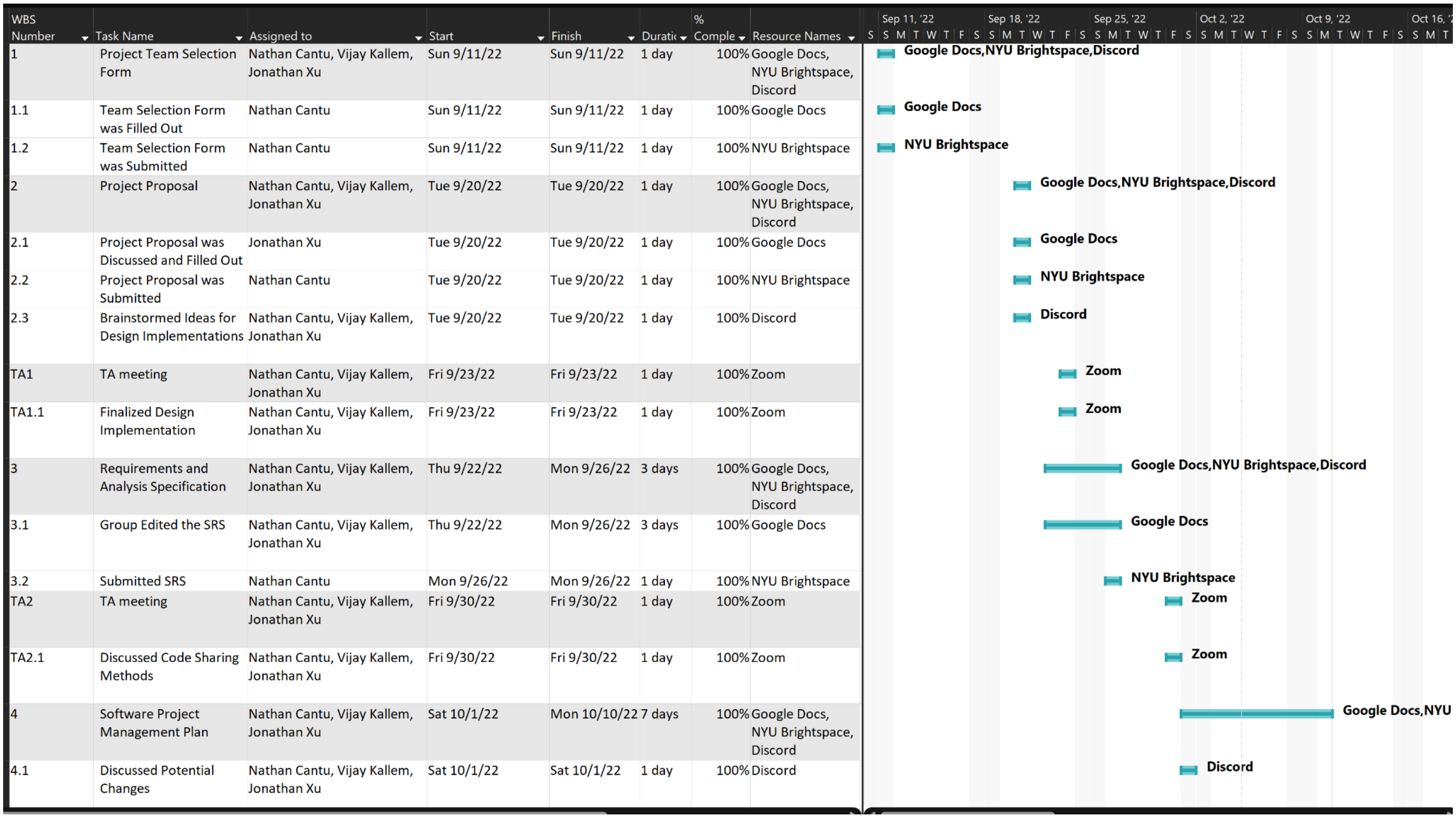
Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
SRS - Domain	Nathan Cantu	5 per page	3 per page	2 per page
	Jonathan Xu	4 per page	3 per page	1 per page
	Vijay Kallem	3 per page	8 per page	5 per page
	Summary for entire team	12 per page	14 per page	8 per page

Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
SPMP	Nathan Cantu	3 per page	5 per page	2 per page
	Jonathan Xu	3 per page	4 per page	1 per page
	Vijay Kallem	5 per page	5 per page	0 per page
	Summary for entire team	11 per page	14 per page	3 per page

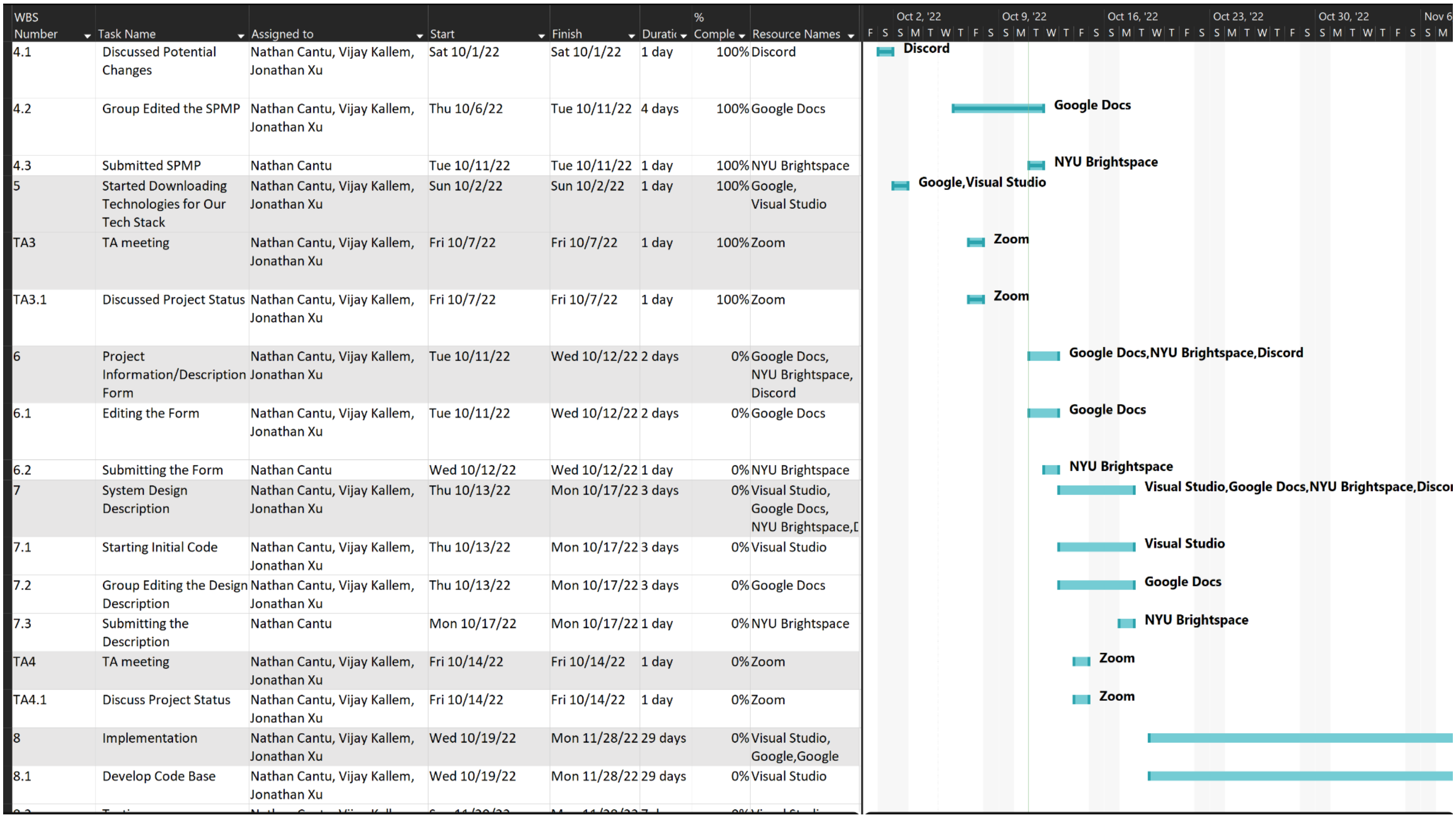
Cumulative

Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
Cumulative - Summary	Nathan Cantu	8 per page	8 per page	0 per page
	Jonathan Xu	7 per page	7 per page	0 per page
	Vijay Kallem	8 per page	13 per page	5 per page
	Summary for entire Team	23 per page	28 per page	5 per page

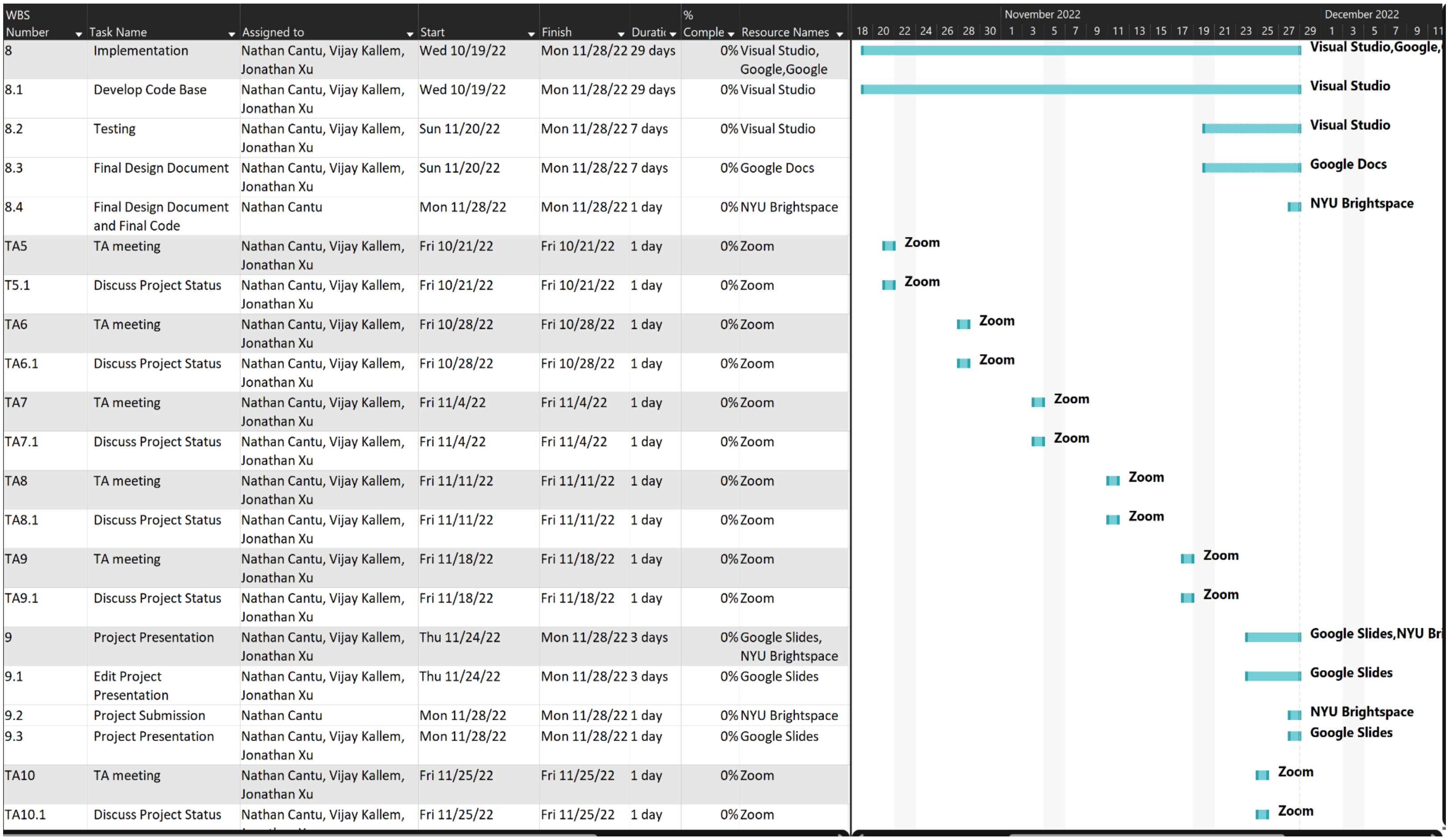
Gantt Chart/Microsoft Project/Spreadsheet Schedule



WBS Number	Task Name	Assigned to	Start	Finish	Duratio	% Comple	Resource Names
1	Project Team Selection Form	Nathan Cantu, Vijay Kallem, Jonathan Xu	Sun 9/11/22	Sun 9/11/22	1 day	100%	Google Docs, NYU Brightspace, Discord
1.1	Team Selection Form was Filled Out	Nathan Cantu	Sun 9/11/22	Sun 9/11/22	1 day	100%	Google Docs
1.2	Team Selection Form was Submitted	Nathan Cantu	Sun 9/11/22	Sun 9/11/22	1 day	100%	NYU Brightspace
2	Project Proposal	Nathan Cantu, Vijay Kallem, Jonathan Xu	Tue 9/20/22	Tue 9/20/22	1 day	100%	Google Docs, NYU Brightspace, Discord
2.1	Project Proposal was Discussed and Filled Out	Jonathan Xu	Tue 9/20/22	Tue 9/20/22	1 day	100%	Google Docs
2.2	Project Proposal was Submitted	Nathan Cantu	Tue 9/20/22	Tue 9/20/22	1 day	100%	NYU Brightspace
2.3	Brainstormed Ideas for Design Implementations	Nathan Cantu, Vijay Kallem, Jonathan Xu	Tue 9/20/22	Tue 9/20/22	1 day	100%	Discord
TA1	TA meeting	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 9/23/22	Fri 9/23/22	1 day	100%	Zoom
TA1.1	Finalized Design Implementation	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 9/23/22	Fri 9/23/22	1 day	100%	Zoom
3	Requirements and Analysis Specification	Nathan Cantu, Vijay Kallem, Jonathan Xu	Thu 9/22/22	Mon 9/26/22	3 days	100%	Google Docs, NYU Brightspace, Discord
3.1	Group Edited the SRS	Nathan Cantu, Vijay Kallem, Jonathan Xu	Thu 9/22/22	Mon 9/26/22	3 days	100%	Google Docs
3.2	Submitted SRS	Nathan Cantu	Mon 9/26/22	Mon 9/26/22	1 day	100%	NYU Brightspace
TA2	TA meeting	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 9/30/22	Fri 9/30/22	1 day	100%	Zoom
TA2.1	Discussed Code Sharing Methods	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 9/30/22	Fri 9/30/22	1 day	100%	Zoom
4	Software Project Management Plan	Nathan Cantu, Vijay Kallem, Jonathan Xu	Sat 10/1/22	Mon 10/10/22	7 days	100%	Google Docs, NYU Brightspace, Discord
4.1	Discussed Potential Changes	Nathan Cantu, Vijay Kallem, Jonathan Xu	Sat 10/1/22	Sat 10/1/22	1 day	100%	Discord



WBS Number	Task Name	Assigned to	Start	Finish	Duratic	% Comple	Resource Names
4.1	Discussed Potential Changes	Nathan Cantu, Vijay Kallem, Jonathan Xu	Sat 10/1/22	Sat 10/1/22	1 day	100%	Discord
4.2	Group Edited the SPMP	Nathan Cantu, Vijay Kallem, Jonathan Xu	Thu 10/6/22	Tue 10/11/22	4 days	100%	Google Docs
4.3	Submitted SPMP	Nathan Cantu	Tue 10/11/22	Tue 10/11/22	1 day	100%	NYU Brightspace
5	Started Downloading Technologies for Our Tech Stack	Nathan Cantu, Vijay Kallem, Jonathan Xu	Sun 10/2/22	Sun 10/2/22	1 day	100%	Google, Visual Studio
TA3	TA meeting	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 10/7/22	Fri 10/7/22	1 day	100%	Zoom
TA3.1	Discussed Project Status	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 10/7/22	Fri 10/7/22	1 day	100%	Zoom
6	Project Information/Description Form	Nathan Cantu, Vijay Kallem, Jonathan Xu	Tue 10/11/22	Wed 10/12/22	2 days	0%	Google Docs, NYU Brightspace, Discord
6.1	Editing the Form	Nathan Cantu, Vijay Kallem, Jonathan Xu	Tue 10/11/22	Wed 10/12/22	2 days	0%	Google Docs
6.2	Submitting the Form	Nathan Cantu	Wed 10/12/22	Wed 10/12/22	1 day	0%	NYU Brightspace
7	System Design Description	Nathan Cantu, Vijay Kallem, Jonathan Xu	Thu 10/13/22	Mon 10/17/22	3 days	0%	Visual Studio, Google Docs, NYU Brightspace,[
7.1	Starting Initial Code	Nathan Cantu, Vijay Kallem, Jonathan Xu	Thu 10/13/22	Mon 10/17/22	3 days	0%	Visual Studio
7.2	Group Editing the Design Description	Nathan Cantu, Vijay Kallem, Jonathan Xu	Thu 10/13/22	Mon 10/17/22	3 days	0%	Google Docs
7.3	Submitting the Description	Nathan Cantu	Mon 10/17/22	Mon 10/17/22	1 day	0%	NYU Brightspace
TA4	TA meeting	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 10/14/22	Fri 10/14/22	1 day	0%	Zoom
TA4.1	Discuss Project Status	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 10/14/22	Fri 10/14/22	1 day	0%	Zoom
8	Implementation	Nathan Cantu, Vijay Kallem, Jonathan Xu	Wed 10/19/22	Mon 11/28/22	29 days	0%	Visual Studio, Google,Google
8.1	Develop Code Base	Nathan Cantu, Vijay Kallem, Jonathan Xu	Wed 10/19/22	Mon 11/28/22	29 days	0%	Visual Studio



WBS Number	Task Name	Assigned to	Start	Finish	Duratic	% Comple	Resource Names
8	Implementation	Nathan Cantu, Vijay Kallem, Jonathan Xu	Wed 10/19/22	Mon 11/28/22	29 days	0%	Visual Studio, Google,Google
8.1	Develop Code Base	Nathan Cantu, Vijay Kallem, Jonathan Xu	Wed 10/19/22	Mon 11/28/22	29 days	0%	Visual Studio
8.2	Testing	Nathan Cantu, Vijay Kallem, Jonathan Xu	Sun 11/20/22	Mon 11/28/22	7 days	0%	Visual Studio
8.3	Final Design Document	Nathan Cantu, Vijay Kallem, Jonathan Xu	Sun 11/20/22	Mon 11/28/22	7 days	0%	Google Docs
8.4	Final Design Document and Final Code	Nathan Cantu	Mon 11/28/22	Mon 11/28/22	1 day	0%	NYU Brightspace
TA5	TA meeting	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 10/21/22	Fri 10/21/22	1 day	0%	Zoom
T5.1	Discuss Project Status	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 10/21/22	Fri 10/21/22	1 day	0%	Zoom
TA6	TA meeting	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 10/28/22	Fri 10/28/22	1 day	0%	Zoom
TA6.1	Discuss Project Status	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 10/28/22	Fri 10/28/22	1 day	0%	Zoom
TA7	TA meeting	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 11/4/22	Fri 11/4/22	1 day	0%	Zoom
TA7.1	Discuss Project Status	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 11/4/22	Fri 11/4/22	1 day	0%	Zoom
TA8	TA meeting	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 11/11/22	Fri 11/11/22	1 day	0%	Zoom
TA8.1	Discuss Project Status	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 11/11/22	Fri 11/11/22	1 day	0%	Zoom
TA9	TA meeting	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 11/18/22	Fri 11/18/22	1 day	0%	Zoom
TA9.1	Discuss Project Status	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 11/18/22	Fri 11/18/22	1 day	0%	Zoom
9	Project Presentation	Nathan Cantu, Vijay Kallem, Jonathan Xu	Thu 11/24/22	Mon 11/28/22	3 days	0%	Google Slides, NYU Brightspace
9.1	Edit Project Presentation	Nathan Cantu, Vijay Kallem, Jonathan Xu	Thu 11/24/22	Mon 11/28/22	3 days	0%	Google Slides
9.2	Project Submission	Nathan Cantu	Mon 11/28/22	Mon 11/28/22	1 day	0%	NYU Brightspace
9.3	Project Presentation	Nathan Cantu, Vijay Kallem, Jonathan Xu	Mon 11/28/22	Mon 11/28/22	1 day	0%	Google Slides
TA10	TA meeting	Nathan Cantu, Vijay Kallem, Jonathan Xu	Fri 11/25/22	Fri 11/25/22	1 day	0%	Zoom
TA10.1	Discuss Project Status	Nathan Cantu, Vijay Kallem,	Fri 11/25/22	Fri 11/25/22	1 day	0%	Zoom