



**NYU**

**TANDON SCHOOL  
OF ENGINEERING**

# **Computer Science and Engineering**

---

## **Chores App**

## **Project Management Plan**

**Version 1.0**

Document Number: SPMP-001

Project Team Number: A08

Project Team Members (name and NET\_ID): Helen Xu (hxx201), Kevin Grajeda (kag622), Alwyn Zhang (az1436)

**REVIEW AND APPROVALS**

Printed Name and Title	Function (Author, Reviewer, Approval)	Date	Signature
Professor Strauss	Approval	Fall 2020	
Kevin Grajeda	Author	11/05/2020	Kevin Grajeda
Helen Xu	Author	11/05/2020	Helen Xu
Alwyn Zhang	Author	11/05/2020	Alwyn Zhang

**REVISION LEVEL**

Date	Revision Number	Purpose
11/05/2020	Version 1.0	Initial Release

---

**Table of Contents**

<b>1.</b>	<b>OVERVIEW</b>	<b>5</b>
1.1	PROJECT SUMMARY	5
1.2	PURPOSE, SCOPE, AND OBJECTIVES	5
1.3	ASSUMPTIONS AND CONSTRAINTS	5
1.4	PROJECT DELIVERABLES	6
1.5	SCHEDULE AND BUDGET SUMMARY	6
1.6	EVOLUTION OF THE PLAN	6
<b>2</b>	<b>REFERENCES</b>	<b>6</b>
<b>3</b>	<b>DEFINITIONS</b>	<b>6</b>
<b>4</b>	<b>PROJECT ORGANIZATION</b>	<b>6</b>
4.1	EXTERNAL INTERFACES	6
4.2	INTERNAL STRUCTURE	7
4.3	ROLES AND RESPONSIBILITIES	7
<b>5</b>	<b>MANAGEMENT PROCESSES</b>	<b>7</b>
5.1	START-UP PLAN	7
5.1.1	<i>Estimation Plan</i>	7
5.1.2	<i>Staffing Plan</i>	8
5.1.3	<i>Resource Acquisition Plan</i>	8
5.1.4	<i>Training Plan</i>	8
5.2	WORK PLAN	8
5.2.1	<i>Work Activities</i>	9
5.2.2	<i>Schedule Allocation</i>	9
5.2.3	<i>Resource Allocation</i>	9
5.2.4	<i>Budget Allocation</i>	10
5.3	CONTROL PLAN	10
5.3.1	<i>Requirement Control and Traceability</i>	10
5.3.2	<i>Schedule Tacking and Adjustment</i>	10
5.3.3	<i>Budget Tracking and Adjustment</i>	11
5.3.4	<i>Quality Control</i>	11
5.3.5	<i>Reporting Mechanisms</i>	11
5.3.6	<i>Metrics Collection Plan</i>	11
5.4	RISK MANAGEMENT PLAN	12
5.5	POST IMPLEMENTATION PLAN	12
<b>6</b>	<b>TECHNICAL PROCESSES</b>	<b>12</b>
6.1	PROCESS MODEL	12
6.2	METHODS, TOOLS, AND TECHNIQUES	13
6.3	INFRASTRUCTURE PLAN	13
6.4	PRODUCT ACCEPTANCE AND MIGRATION PLAN	13
<b>7</b>	<b>SUPPORTING PROCESSES PLANS</b>	<b>13</b>
7.1	CONFIGURATION MANAGEMENT PLAN	14
7.2	QUALIFICATION (VERIFICATION AND VALIDATION) PLAN	14
7.3	DOCUMENTATION (LIBRARY) PLAN	14
7.4	QUALITY ASSURANCE PLAN	15

7.5	REVIEWS AND AUDITS	15
7.6	PROBLEM RESOLUTION PLANS	15
7.7	ENVIRONMENT MANAGEMENT PLANS	15
7.8	PROCESS IMPROVEMENT PLAN	16
<b>8.</b>	<b>ADDITIONAL PLANS</b>	<b>16</b>
<b>9</b>	<b>INDEX</b>	<b>16</b>
<b>10</b>	<b>RATIONALE</b>	<b>16</b>
<b>11</b>	<b>NOTES</b>	<b>16</b>
<b>12</b>	<b>APPENDICES</b>	<b>17</b>
12.1	SCHEDULE TRACKING	18
12.2	DEFECT TRACKING	20
12.3	GANTT CHART/MICROSOFT PROJECT SCHEDULE	21

## 1. OVERVIEW

---

### 1.1 Project Summary

The motivation for the Chores App is to improve communication among people living together by giving them a way to assign tasks to each other that mitigates awkwardness. The purpose of the SPMP is to describe how we will go about developing the chores app. The intended audience is the team, management, and the client.

### 1.2 Purpose, Scope, and Objectives

The system will give users the ability to manage their shared tasks and responsibilities with their roommates/housemates. Since people often put off responsibilities or forgot about them, this application aims to help users stay on top of their tasks and to communicate with roommates/housemates effectively. Users can create tasks that can be seen by other users within the application and they can be modified by the user based on their status. The system is not intended to help the user complete specific tasks/chores, but to be a management tool resource for users.

### 1.3 Assumptions and Constraints

We assume the user possesses an electronic device to use the app in the first place. On top of that, the system needs internet connection in order to be accessed, as devices need to communicate with each other to assign tasks. The app would also need to identify users, so providing something like an email or username as well as a password would be necessary for login purposes.

### 1.4 Project Deliverables

Deliverable	Date
Project Proposal	September 22, 2020
Software Specifications Requirement (SRS) - Domain Definition	October 8, 2020
SRS - Project Requirements	October 20, 2020
Software Project Management Plan (SPMP)	November 2, 2020
SRS - Project Analysis	December 1, 2020
Presentation	Final Two Weeks of Semester

### 1.5 Schedule and Budget Summary

See Section 12.3 for schedule; budget will be provided in a future release.

### 1.6 Evolution of the Plan

The document will be updated if there is a change in user requirements or the schedule. Changes will be documented and the document will be updated, inspected, and approved.

*Project Proposal 1.0*, Grajeda, Xu, Zhang September 2020.

## 2 REFERENCES

*System Requirements Specification 2.0*, Grajeda, Xu, Zhang  
October 2020.

### 3 DEFINITIONS

---

None at the moment.

## 4 PROJECT

### ORGANIZATION

---

#### 4.1 External Interfaces

The external entities involved include the management team, testing team and quality assurance team. The development team will work in coordination with them in order to make sure the system is the highest quality possible.

#### 4.2 Internal Structure

The development team will be split into three roles with some shared responsibilities. This includes writer, reviewer, and uploader in the documentation phase and programmer, designer, error checker in the development phase. The whole team will interface with the external entities related to the project.

#### 4.3 Roles and Responsibilities

Role	Team Member	Responsibility
Writer	All Team Members	Writes documentation and makes necessary changes
Reviewer	All Team Members	Reviews documentation for any possible errors or inconsistencies
Uploader	Alternating	Uploads documentation

## 5 MANAGEMENT

### PROCESSES

---

#### 5.1 Start-Up Plan

**5.1.1 Estimation Plan** To be added in a future release.

**5.1.2 Staffing Plan** To be added in a future release.

**5.1.3 Resource Acquisition Plan** To be added in a future release.

**5.1.4 Training Plan** The engineering team must have sufficient training and/or experience to be qualified to develop all the components of the system. This includes experience in client/server side programming, interface design, database development/management, and server design/maintenance. The entire team must be familiar with all of the project's documentation, as well. The specific training methods, and entry and exit criteria will be developed at a later time.

#### 5.2 Work Plan

**5.2.1 Work Activities** The work activities for the project in order are:  
Creating documentation, designing the system, coding for the application and database layers, reviewing documentation and code, and testing the application.  
Coding, document revision, and testing will continue to occur throughout the development process after the application is complete as improvements and iterations are made. All team members are expected to be informed regarding all aspects of the project and to invest similar amounts of time and effort as their peers in understanding and contributing to all work activities and throughout the development cycle.



**5.2.2 Schedule Allocation** Individual team members may freely work on the project whenever they are available. At least once a week, all team members will coordinate a meeting to discuss and review their work. Should the team find that they are failing and struggling to meet project milestones and deadlines specified in the SRS in time, the schedule allocation will be revisited and changes will be considered.

**5.2.3 Resource Allocation** All members of the development team will have access to the same resources. If one team member needs a specific piece of equipment or software, they will obtain it promptly. Resources include computers, IDEs, database tools, and server tools. Team members will also have management support for guidance.

**5.2.4 Budget Allocation** To be added in a future release.

## 5.3 Control Plan

**5.3.1 Requirement Control and Traceability** The system's functional and non-functional requirements have been identified and documented in the SRS. The team must abide by the documentation. In order to ensure that the project is following the requirements, any changes done will be made traceable through documentation. The proposed change will be analyzed on whether they will affect the requirements of the project. They will also be reviewed in terms of impact to cost and schedule. Only then will the change be approved/denied and implemented.

**5.3.2 Schedule Tacking and Adjustment** The development team will have a schedule that they must follow strictly. Milestones will be set for the team throughout the schedule in order to measure progress. In order to be objective, the milestones will be tangible deliverables. Should progress meet the progress outlined by the schedule, the team will convene and address the issues that may have set them behind and create a plan to get the project back on schedule. Any necessary adjustments will be proposed and added if necessary.

**5.3.3 Budget Tracking and Adjustment** To be added in a future release.

**5.3.4 Quality Control**

Any software created for the project will be assessed by the original author first and then peer reviewed by other team members. If the product has reached a state intended to be usable, the team will run it through various potential use cases to find potential defects, which will be documented immediately and resolved as soon as possible.

**5.3.5 Reporting Mechanisms**

To be added in a future release.

**5.3.6 Metrics Collection Plan**

The project metrics to be collected are time spent working on the project, number of defects. Team members will record their hours working on specific documents or programs and tally the number of defects they discover each time they create, review, or change something. The cumulative number of hours spent and defects found will be recorded and updated accordingly.

**5.4 Risk Management Plan**

The team will routinely check that all aspects of the system are functional and secure. Processes that involve the database will be tested for security and consistency before being deployed for the user base. Any defects or vulnerabilities discovered will be addressed as soon as possible. If necessary, the system will be made temporarily available until the issue is resolved. Beyond ensuring that the system is reliable and secure, the team will not be liable for what occurs within individual groups. Therefore, the major risks would be the following:

1.

Business Risk: Server Crash

Description: The system's server goes down unexpectedly

Probability: Low

How discovered: System becomes unavailable

Responsible Party: The development team

Status: System is not operational

Mitigation Plan: Creating a server backup

2.

Business Risk: Database Security Issues

Description: There is a vulnerability discovered with the system's database

Probability: Low

How discovered: Regular checks and reviews of system

Responsible Party: The development team

Status: System operates but is not secured

Mitigation Plan: System is temporarily shut down until issues are fixed by development team

To be added in a future release.

## 5.5 Post Implementation Plan

## 6 TECHNICAL PROCESSES

---

### 6.1 Process Model

The project will use an object-oriented methodology and an iterative life-cycle model. All team members will conduct weekly reviews of any work products created or edited for the project. If a functional version of the program is available, it will be run through several use cases as part of this review process. All deliverables specified in the SRS will be completed and submitted for approval by their respective deadlines and function as the major milestones to be achieved for the project.

### 6.2 Methods, Tools, and Techniques

The project will be developed with an object-oriented UML method. Google Docs will be used to create and edit diagrams and documents. Google Drive will be used to store and share documentation. Microsoft Project will be used to create schedules and Gantt charts. The application and database will be created and maintained using the Python and MySQL programming languages.

### 6.3 Infrastructure Plan

The application will be hosted on Amazon Elastic Compute Cloud and the database will be hosted on Amazon Aurora. The team will use Google Docs and Google Drive to create, store, and share documentation. Team members will use Discord or Zoom to communicate and collaborate on the project.

To be added in a future release.

## **6.4 Product Acceptance and Migration Plan**

## **7 SUPPORTING PROCESSES PLANS**

---

### **7.1 Configuration Management Plan**

The software development team will be using Github and Google Drive to manage software and documentation, respectively. Team members must approve of the progress done before uploading to Github. Google Drive will allow for documentation to be quickly updated along with keeping track of changes made. If a change is proposed it must also be approved by the whole team.

### **7.2 Qualification (Verification and Validation) Plan**

In order for the project to be validated it must be approved by every member in the development team. The process involves checking for defects and making sure that the project meets all the requirements outlined in the documentation. Validation also involves testing the system on multiple devices.

### **7.3 Documentation (library) Plan**

All team members are responsible for writing and maintaining documentation for the project. All documentation will be saved and shared on Google Drive, allowing team members to access and edit at any time. For most internal and deliverable work products, all team members will convene and collaborate to create, review, and approve new documents or changes to existing documentation. Outside of these sessions, team members may freely edit or add to documents to reflect their individual work, but all changes must be recorded. Other team members will review any changes made and either approve or deny them. As part of this review process, team members will look for and take note of any defects. All team members are to be held accountable for any defects in documentation and any defects are to be brought to the attention of the rest of the team as soon as possible.

## **7.4 Quality Assurance Plan**

The system's functional and non-functional requirements must be approved before any development begins. Any possible changes must be proposed to the entire team, approved by the entire team, and documented. Once that is done, only then can the change be implemented. Team members will carry out regular review processes and fix any possible defects promptly. Team members must regularly reference the documentation in order to make sure that the system follows its requirements.

## **7.5 Reviews and Audits**

The project will be reviewed by development team members individually at least once per week. If an issue is found while reviewing, the team will carry out the problem resolution plan. Otherwise, if all team members reviewed the project without any issues or concerns arising, they can move on in the development process. The project will receive an extra extensive review before the deadline date, as well.

## **7.6 Problem Resolution Plans**

If an issue is found within the project by one member of the development team, they will notify the rest of the team. They will then discuss the issue with the team and come up with a resolution together. The team member(s) least occupied by important work will implement a solution to the issue. Once the issue is resolved, the rest of the team will be informed to keep the team caught up with the process of the project.

## **7.7 Environment Management Plans**

To be added in a future release.

## **7.8 Process Improvement Plan**

To be added in a future release.

## **8. ADDITIONAL PLANS**

---

To be added in a future release.

## **9 INDEX**

---

May be added in a future release.

## 10 RATIONALE

---

None

## 11 NOTES

---

None

## 12 APPENDICES

---

### 12.1 Schedule Tracking

Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
SRS Domain	Alwyn Zhang	3 hours	2 hours	1 hour
SRS Domain	Kevin Grajeda	4 hours	2 hours	2 hours
SRS Domain	Helen Xu	3 hours	1 hour	2 hours
	Entire Team	10 Hours	5 hours	5 hours

Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
SRS Requirements	Alwyn Zhang	2 hours	2 hours	0 hours
SRS Requirements	Kevin Grajeda	3 hours	2 hours	1 hour
SRS Requirements	Helen Xu	3 hours	2 hours	1 hour
	Entire Team	8	6	2

Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
SPMP	Alwyn Zhang	5 hours	3 hours	2 hours
SPMP	Kevin Grajeda	4 hours	3 hours	1 hour
SPMP	Helen Xu	3 hours	3 hours	0 hours
	Entire Team			

Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
SRS Analysis				
SRS Analysis				
SRS Analysis				
	Entire Team			

**Cumulative**

Who (individual or Team)	Estimated	Actual	Difference
Alwyn Zhang	10 hours	7 hours	3 hour
Kevin Grajeda	11 Hours	7 Hours	4 hours
Helen Xu	9 hours	6 hours	3 hours
Entire Team	30 Hours	20 hours	10 hours

**12.2 Defect Tracking**

Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
SRS Domain	Alwyn Zhang	5	6	1
SRS Domain	Kevin Grajeda	12	8	4
SRS Domain	Helen Xu	7	5	2
	Entire Team	24	17	7

Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
SRS Requirements	Alwyn Zhang	2	4	2
SRS Requirements	Kevin Grajeda	10	7	3
SRS Requirements	Helen Xu	5	6	1
	Entire Team	17	17	0

Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
SPMP	Alwyn Zhang	3	5	2
SPMP	Kevin Grajeda	8	7	1
SPMP	Helen Xu	3	5	2
	Entire Team	14	17	3

Artifact or Deliverable	Who (individual or Team)	Estimated	Actual	Difference
SRS Analysis				
SRS Analysis				
SRS Analysis				
	Entire Team			

**Cumulative**

Who (individual or Team)	Estimated	Actual	Difference
Alwyn Zhang	7	10	3
Kevin Grajeda	12	8	4
Helen Xu	7	5	2
Entire Team	24	17	7



## 12.3 Gantt Chart/Microsoft Project Schedule

