

LAB 1

COMP 5541- Winter 2024
Tools and Techniques for Software Engineering

Fahime Nezhadali

fahime.nezhadali@gmail.com

Course Outline

Grading Scheme Assignments:

- Assignments: 20%
- Project: 20%
- Quizzes: 30%
- Final Exam: 30%

Course Outline

In this course, we have three assignments and one course project.

Three Assignments: Write reports for these three assignments and implement them in your final project.

- Requirements gathering
- Architectural design
- Testing

Course Project: In a Team Submit a Project Report and Develop the project which will be evaluated with a 15-20 mins Project Demo.

Students must do (and get non-zero marks in) **at least** two of the three assignments and the project to pass the course.

Course Outline

- **It is very important to complete Assignments on time.**
- Do not delay your assignments for the last moment.
- Develop it step-by-step everyday.
- Ask questions!
- Connect with your TA as they will act as one of the stakeholders for your Assignments and Projects.
- **Each team must submit one Assignment and Project Report per Team**

Course Assignment and Project

- Extending the Open-Source ***Book-Project Web App*** into the “***My Books Web App***”
- You are going to extend the open-source *Book-Project Web App* available at <https://github.com/Project-Books/book-project> into a more personalized Web App named “***My Books Web App***”
- ***Please Refer to Assignment 1 to carefully read the extended requirements for the Book-Project Web App***

Assignment 1: My Books App

Add the Requirements Mentioned in the assignment

Current Book-Project web app is to be extended into a more personalized web app named My Books. The Book-Project is an web app:

Organizes books based on user's interest in 3 Categories (shelves):

- *Reading*: Books that are currently being read,
- *Read*: Books that are already read, and
- *To Read*: Books that will be read in the future.

Extended into the My Books web app:

- Group and display the books in a shelf according to their genres (Sci-fi, Thriller, Novel, Mystery,).
- Add a new shelf called Favorites which contains all the books most liked by the user.
- Add another shelf called Recommendations containing books recommended for the user based on the user's interest.

If any Additional Requirements are added it will considered an add-on!

Assignment 1: My Books App

Establishing Requirements for My Books App

You are going to establish and document the **requirements** for this My Books app. Specifically do the following:

- In consultation with your stakeholders (TA and team members), establish a stable set of requirements.
- Document your established requirements using appropriate mechanisms (user stories, use cases, UML diagrams, etc.).

Your submission should include a **report** listing all your requirements with justifications and their documentation, describing how you established those requirements and how your approach can handle any changes that may happen during the design process.

Assignment 1: My Books App

Requirement Gathering

- Compare System As-Is vs. System-To-Be
- What is the problem?
- What is the impact of the solution?
- Who are the stakeholders of the system?
- What are the Functional Requirements?
- What are the Non-Functional Requirements?

SRS Document Example

Assignment 1: SRS Example

Assignment 1: Report

- Please create a report of no more than 20-25 pages.
- Follow the template or example presented during the LAB session.
- *The following slides include some sections of the Report and what is required within those sections.*

Use- Case

What is a Use-Case ?

A use-case is a written description of how users will perform their task within your system.

It describes from the user's point of view, systems behaviour as it responds to a request.

Why we create Use-Case ?

It helps us visualize user interaction with the system and also helps us understand what and how user is expecting a system to behave in each scenario.

Use- Case

Each use-case is defined as a sequence of simple steps. Beginning with user's goal and ending when the goal is accomplished.

There might some cases where there can possibly be negative outcomes for some goals. For that we use exceptional case to represent the steps when something is going in a negative way how the system must handle and respond to it.

e.g. Scenario 1- User log-in the system – Positive Scenario

What if the user has forgotten password to login within the system ? – Negative Scenario of User login.

Use- Case Format

1. **Use-Case ID** – Unique Identifier for each use-case. e.g. UC-01
2. **Use-Case** – Title of the use-case. e.g. Use-Case : Make a Reservation
3. **Description** – Give a brief on what is the use-case about ? When does it occurs ? What are the constraints ?

Note: Keep description very short as 1-2 sentences.

This use case occurs when the customer has successfully found the tool or equipment according to their need and wants to reserve it (only one tool can be reserved with a reservation request. To reserve more than one tool a customer needs to make a separate reservation request).

Use- Case Format

4. Level – User Goal Level/System Level ?

Level specifies with whom is the use-case related to it. Is it the use-case which will be performed by the user or by the system. Most of the use-cases are User Goal Level.

5. Primary Actor- User primarily involved with the use-case.

e.g. Primary Actor – Customer

6. Supporting Actor/s – Users passively involved with the use-case.

e.g. Supporting Actor – Branch Employee.

Use- Case Format

7. **Stakeholders and Interests** – Who are directly and indirectly involved with the use-case and describe what is their involvement.

e.g. Stakeholders and Interests –

- i. Customer - Reserves a tool or equipment according to their need.
- ii. Branch Employee - Verify the customers details, credit card, rental tool deposit amount and hand out the tool or equipment in store to the customer and update the tool status.
- iii. Competitors - Reserves a tool or equipment to check the flow and navigations.
- iv. Developer - Develops a reservation scenario for the customer to smoothly create a reservation request and performs unit testing.
- v. Tester - Tests the developed scenario and suggests any improvements required

Use- Case Format

8. **Pre-conditions** – What are the requirements in-order to implement the use-case ?

E.g. Pre-Conditions.

1. Customers must create separate reservation requests to reserve each tool.
2. Customer must be 18 years old or older.
3. Customers must make a reservation request as early as next day and upto 30 days in advance

9. **Post-conditions** – The result of the use-case.

- i. Success end condition
- ii. Failure end condition

Use- Case Format

10. Main-Scenario – The actual steps the user will be performing with the system.

11. Extensions – Mention the exception cases(negative scenario) if any for a particular use-case.

Note: Mention at-most 3 extensions and not more than that.

12. Special Requirements – How the use-case must be for the users.
i.e. Specify the quality requirements against each use-case.

User-Stories

- User Stories are the features of the system.
- They are written from the perspective of end-users and written on the index cards.

It contains of three main-steps:

1. **Activities = Use-Case.** e.g. Make a reservation.
2. **Steps = Main-Scenario of a use-case.** e.g. Login -> Search Tools....
3. **Details = lowest-level of granularity of user-interaction.** e.g. Enter username. Enter password....

User-Stories Example

User-Story Map: Mobile App Feature for Depositing Checks

NNGROUP.COM NN/g

1. Activities:

High-level tasks users can do in the digital product

Check account balance

Deposit a check

2. Steps:

Steps users go through to complete the activity above

Log in

Access accounts

Enter mobile deposit details

Sign check

Photograph check

Submit deposit

Confirm deposit

Enter username or email

View account balances

Choose account

Read tips for taking check photos

Enable camera access

Confirm deposit

View confirmation message

3. Details:

Granular, discrete interactions to complete the step above

Enter password

See pending transactions

Enter deposit amount

Turn phone horizontal

Understand amount available

Receive email confirmation

Press login button

Open new account

View transaction limits

Take photo of front & back

Cancel deposit

You can segregate the user-stories steps/features in must-haves, should-haves, could-haves, don't-haves using a color code in the diagram to easily understand.

UML Diagram

You can use draw.io which is the best tool to develop all the diagrams.

A total of 3 diagrams are expected in your Assignment 1 Report based on the priorities mentioned below.

- 1. Use-Case Diagram** – Most Important - **Must-have** in your Assignment 1.
- 2. Sequence Diagram** – **Should-have** in your Assignment 1 for two use-cases.
- 3. Activity Diagram (optional)** – **Could-have** in your Assignment 1.
- 4. State-Machine Diagram (optional)** – **Could-have** in your Assignment 1.

Use-Case Diagram

A use-case diagram is a graphical depiction of a user's possible interactions with a system.

A use-case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well.

The **use-cases** are represented by either circle or ellipses.

The **actors** are represented by the actor symbol.

Interaction of the Actor and use-cases are represented using arrows **relations**.

Use-Case Diagram

- Important Relation using <<extend>> and <<include>>

The <<include>> relationship

- Partial behavior that is common across several use cases (models required behavior)
- Separate it into its own subfunction use case, and indicate its inclusion
- Included sub-function shared across several use-cases (duplicated parts!)

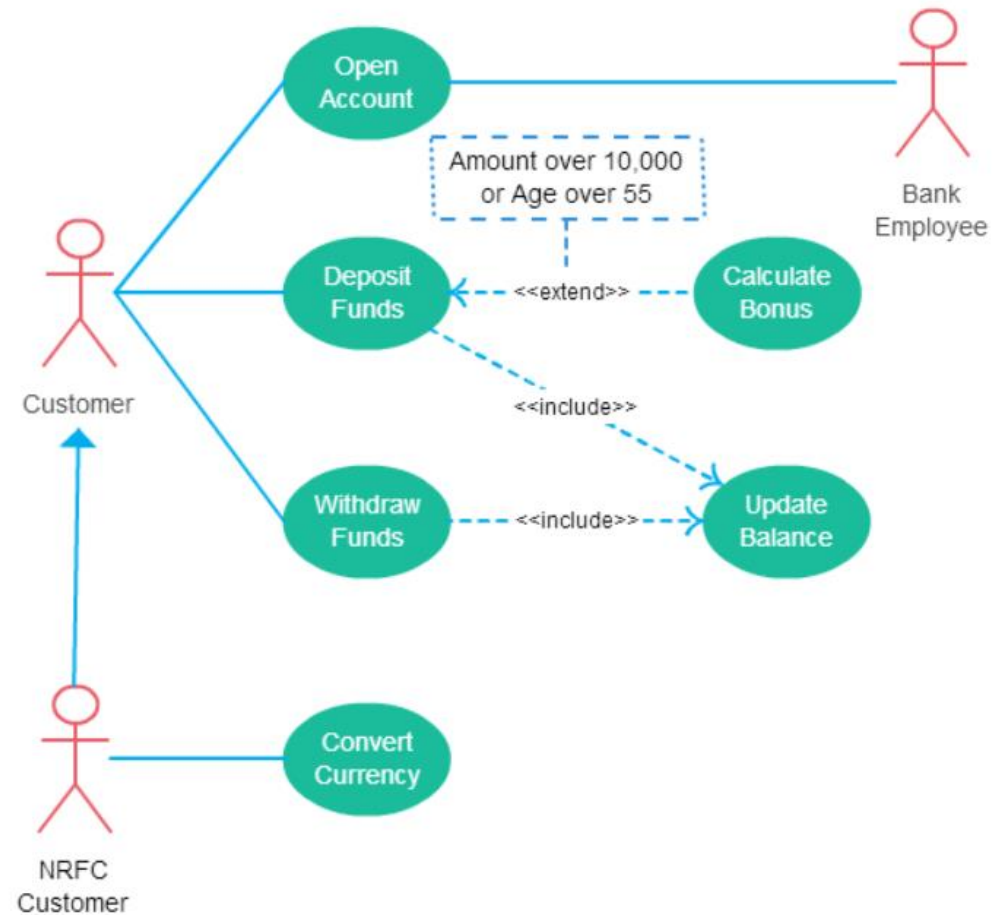
The <<extend>> relationship

- Separate it into its own Subfunction Use Case
- Indicate where and under what condition it extends the behavior of some base use case (models' optional behavior)
- Original use case remains untouched
- Extended functionality may be optional

Use-Case Diagram

- Important Relation using <<extend>> and <<include>>
- <<include>> - reuse functionality.
- <<extend>> - one use-case extends the behavior of another use-case.

Use-Case Diagram



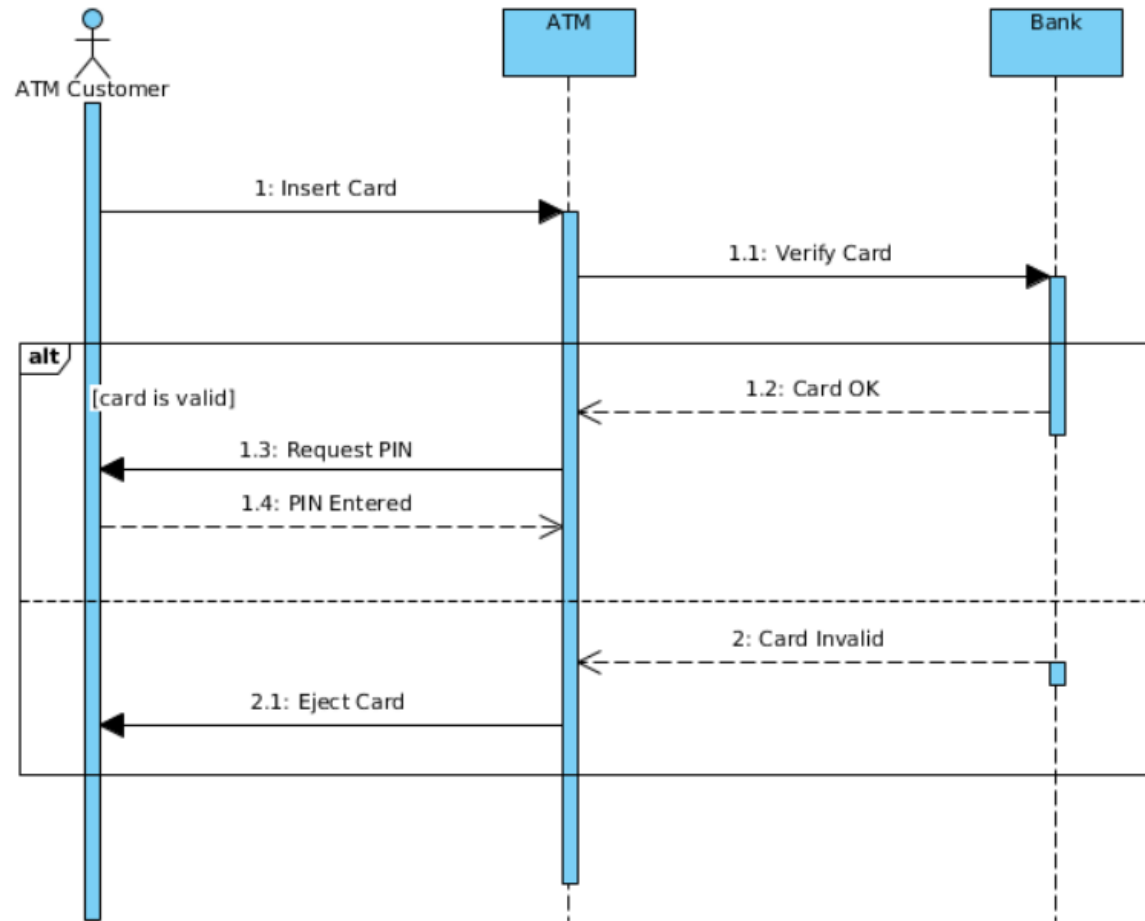
Sequence Diagram

Sequence Diagram illustrates the sequence of messages between objects in an interaction.

It describes how and in-what order a group of objects work together.

Used by software developers and business professionals to understand requirements of a new system.

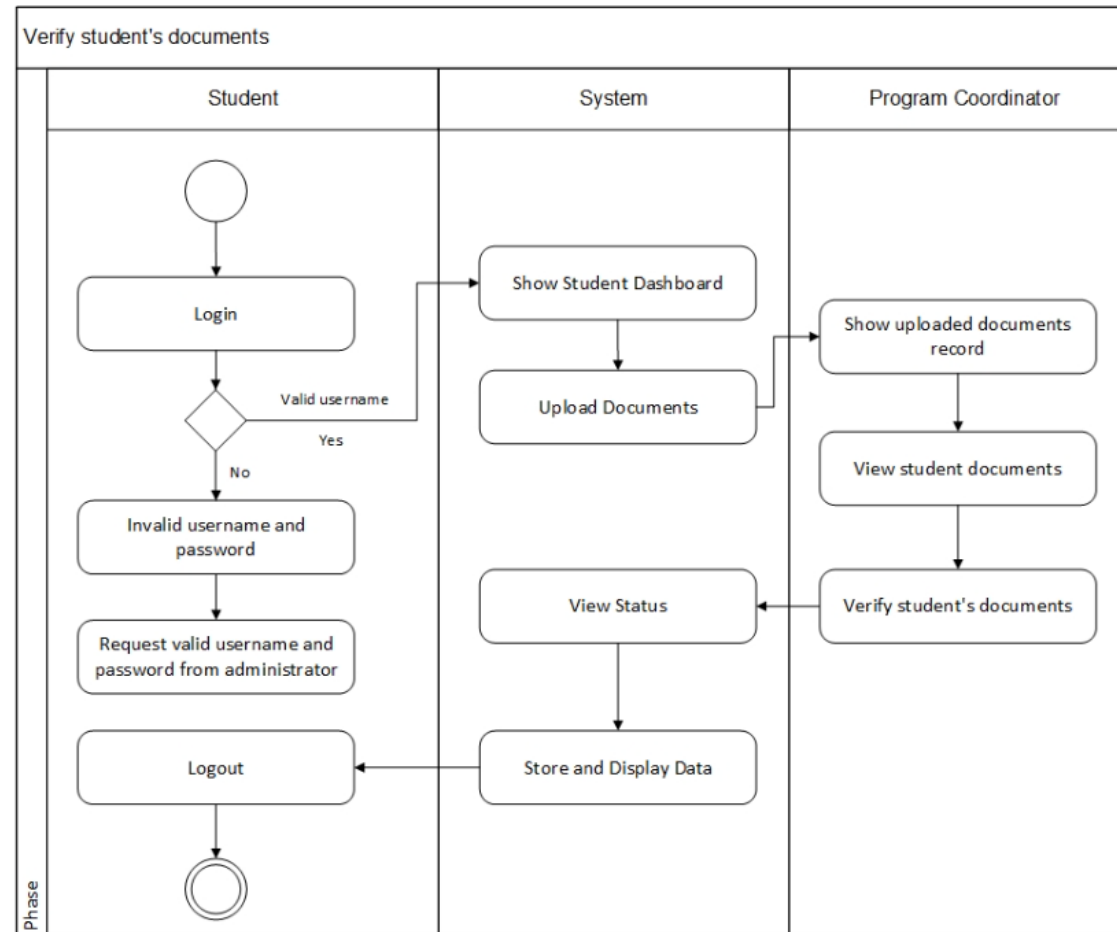
Sequence Diagram



Activity Diagram

- An activity diagram graphically represents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram.
- It also helps to see the negative scenarios of the system.

Activity Diagram



State-Machine Diagram

- Describe the behavior of system in the form of state.
- It helps visualize the developers what is the state-changed when user-performs a particular task(an event).
- State diagrams require that the system described is composed of a finite number of states.

State-Machine Diagram

- It has three main components:

1. Event

- An event is a significant or noteworthy occurrence.

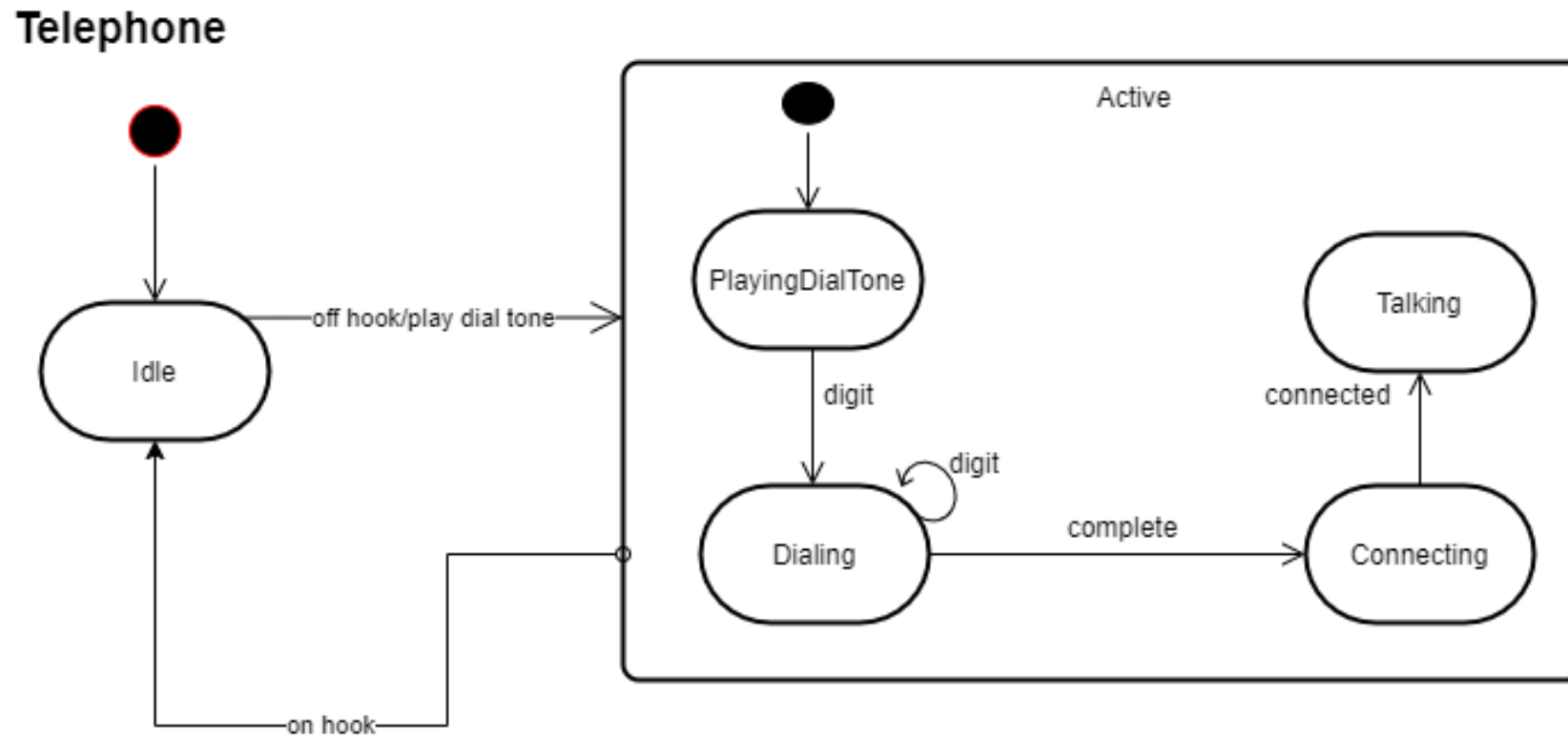
2. State

- A state is the condition of an object at a moment in time between events.

3. Transition

- A transition is a relationship between two states that indicates that when an event occurs, the object moves from the prior state to the subsequent state

State-Machine Diagram



Road-Map

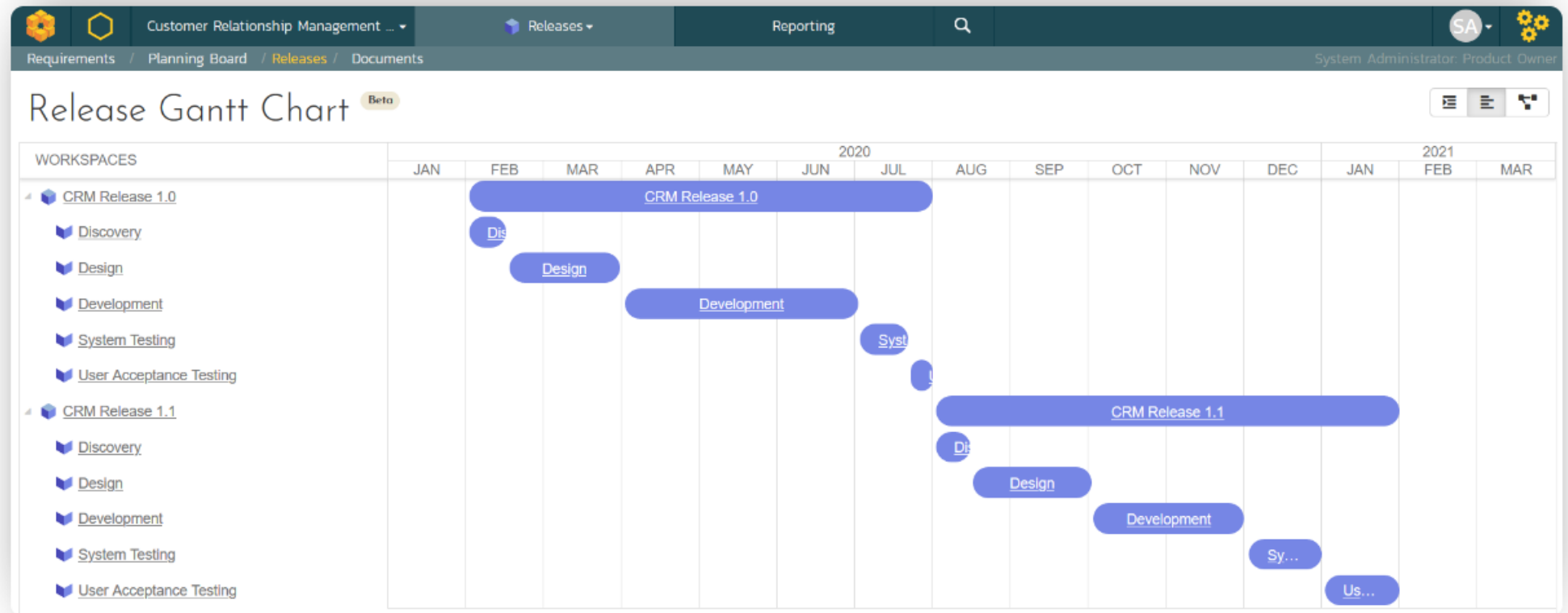
1. Prioritization:

In Agile Methodology we prioritize the requirements based on Must-haves of the systems.

2. Release Plan:

Represent the initial release plan of your system in the form of Gantt Charts.

Road-Map



Appendix

- **Describe the Version Details**

What is the version of the document.

When it was completed.

Reason for creating/changing the document.

- **Contribution Log**

Describe the contribution of each member of the team in brief.

e.g Jim – Project Description.

Kate – Specific Requirements

1. Functional
2. Performance

References

- <https://www.nngroup.com/articles/user-story-mapping/>
- <https://creately.com/blog/diagrams/use-case-diagram-relationships/>
- <https://stackoverflow.com/questions/52231702/uml-sequence-diagram-auto-numbering-in-visual-paradigm>
- <https://www.usability.gov/how-to-and-tools/methods/use-cases.html#:~:text=A%20use%20case%20is%20a,when%20that%20goal%20is%20fulfilled>.