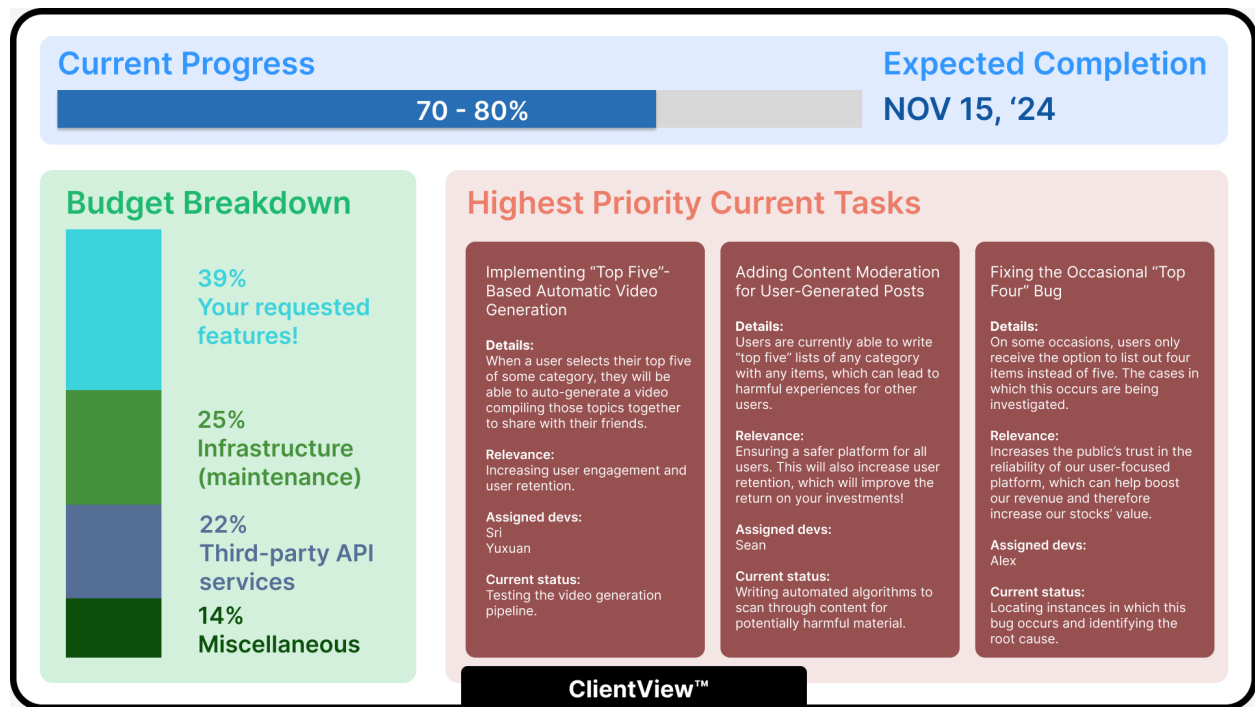


Project Milestone 2

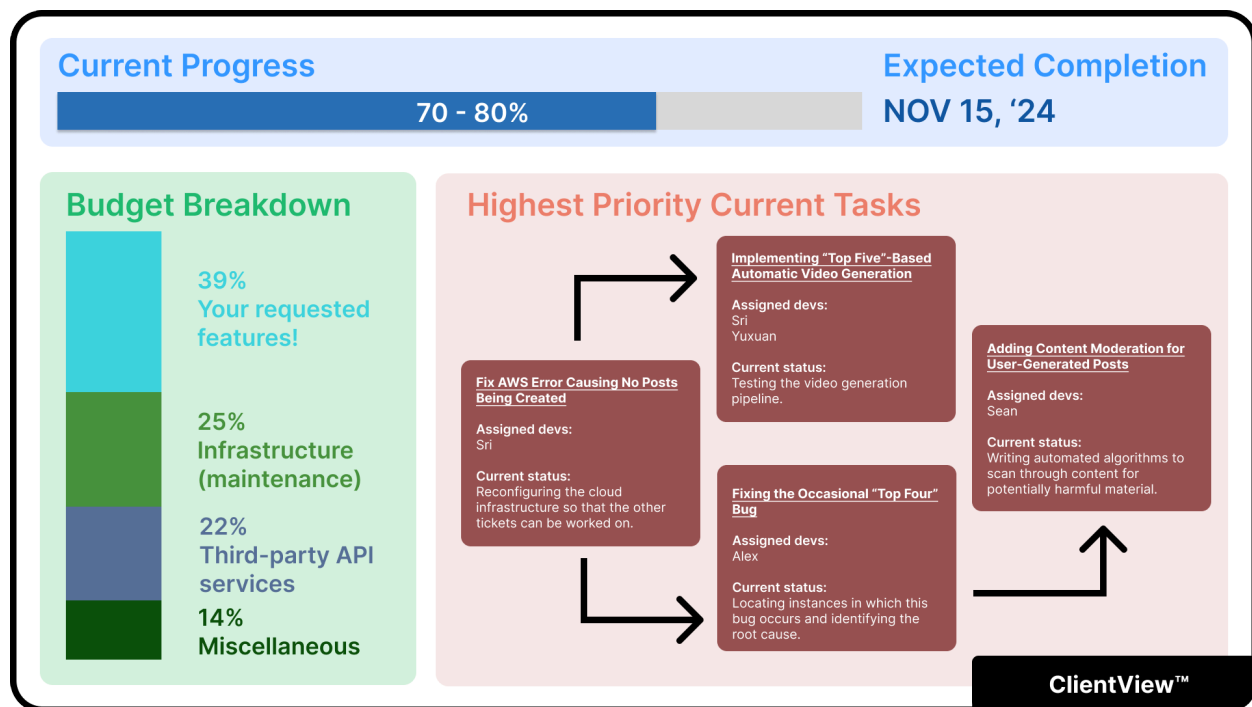
Group Members: Sri Vangaru, Alex Urbina, Yuxuan Guo, Sean Sullivan

Process Deliverable

An older prototype:



A newer prototype (still remains to be further refined):



Requirements Analysis

Non-Functional Requirements:

1. Usability:
 - The dashboard interface must display all text in a minimum of 16pt font size to ensure readability within 1 meter.
 - **Rationale:** Based on our requirements elicitation where Group 09 emphasized the need for easy-to-understand information presentation.
2. Reliability:
 - The system must maintain a minimum uptime of 120 hours during business operations, and should be able to recover within 5 minutes of any failure.
 - **Rationale:** Since this is tracking project progress that clients rely on for business decisions, high reliability is crucial.
3. Performance:
 - The dashboard must load initial project data within 2 seconds for 95% of page requests, and reflect Jira updates within 30 seconds of changes
 - **Rationale:** From our requirements elicitation, clients need real-time tracking and quick access to project status.

4. Supportability:
 - The system must support configuration changes to Jira integration settings without requiring system restart or code modifications.
 - **Rationale:** Based on Group 09's feedback about too many configuration options in current systems, this needs to be streamlined.
5. Implementation/Constraints:
 - The system must be implemented using React.js for the frontend and must integrate with Jira API version 3.0 or higher
 - **Rationale:** Since we're integrating with Jira, this ensures compatibility with existing systems.

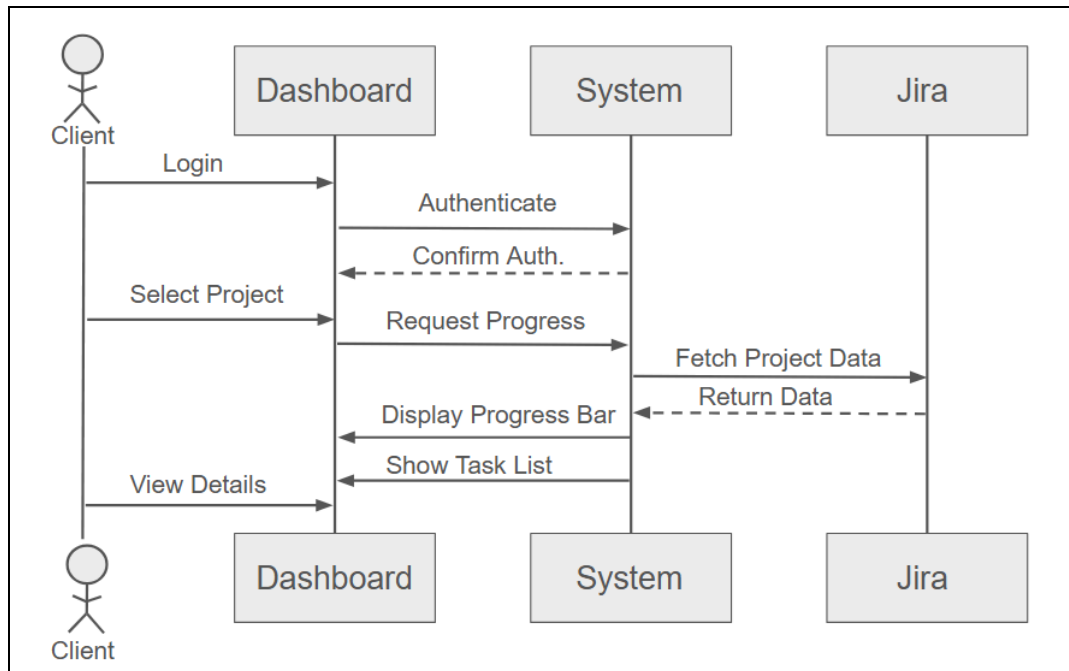
Functional Requirements:

1. The system should automatically sync project data from Jira to the ClientView dashboard
2. The system should display a progress bar showing completed and in-progress tasks for each project
3. The system should provide a visualization of task dependencies and relationships
4. The system should display estimated completion times/dates for projects
5. The system should show current task assignments for each team member

Formal Use Cases:

1. View Project Progress
 - Actor: Client
 - Precondition: Client has a project
 - Main Flow:
 1. Client logs into dashboard
 2. Selects specific project
 3. Views progress bar
 4. Reviews task list
 5. Can open more task details
 - Alternative Flow: System shows error if Jira data unavailable
 - Postcondition: Client views updated project status

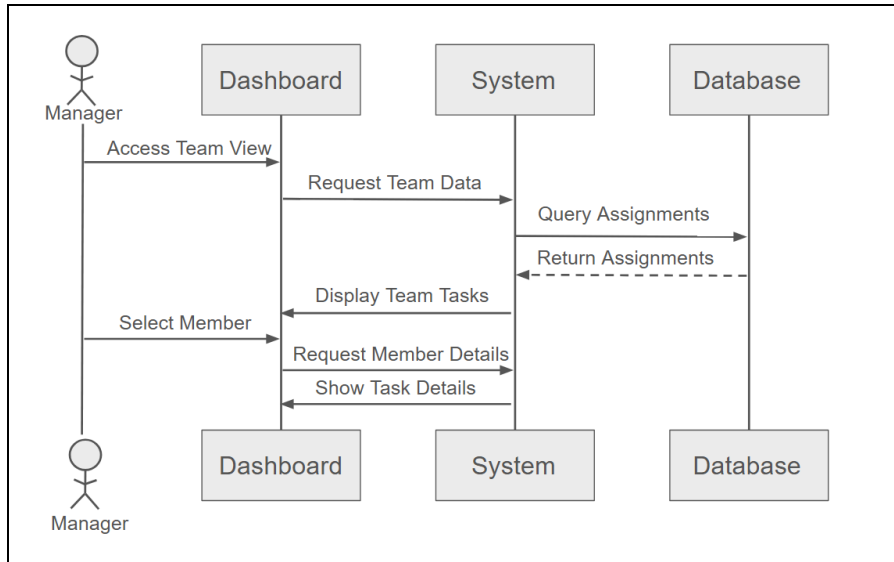
Sequence Diagram:



2. Track Team Member Tasks

- Actor: Project Manager
- Precondition: Manager is authenticated
- Main Flow:
 1. Manager accesses team view
 2. Views team workload distribution
 3. Selects specific team member
 4. Reviews assigned tasks
 5. Views task details
- Alternative Flow: Notification if team member is overloaded
- Postcondition: Manager has current task assignment overview

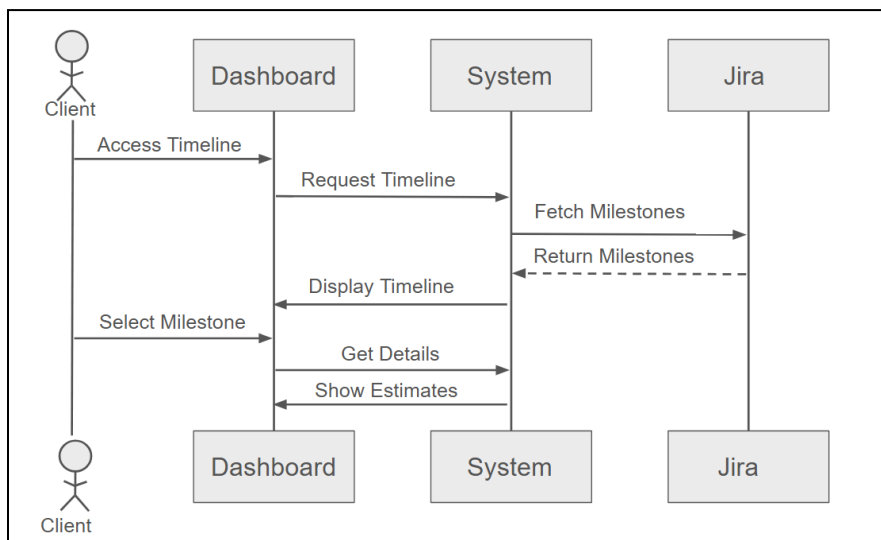
Sequence Diagram:



3. Check Project Timeline

- Actor: Client
- Precondition: Client is authenticated
- Main Flow:
 1. Client accesses timeline view
 2. Views project milestones
 3. Selects specific milestone
 4. Reviews estimated completion dates
- Alternative Flow: Warning if timeline at risk
- Postcondition: Client understands project timeline

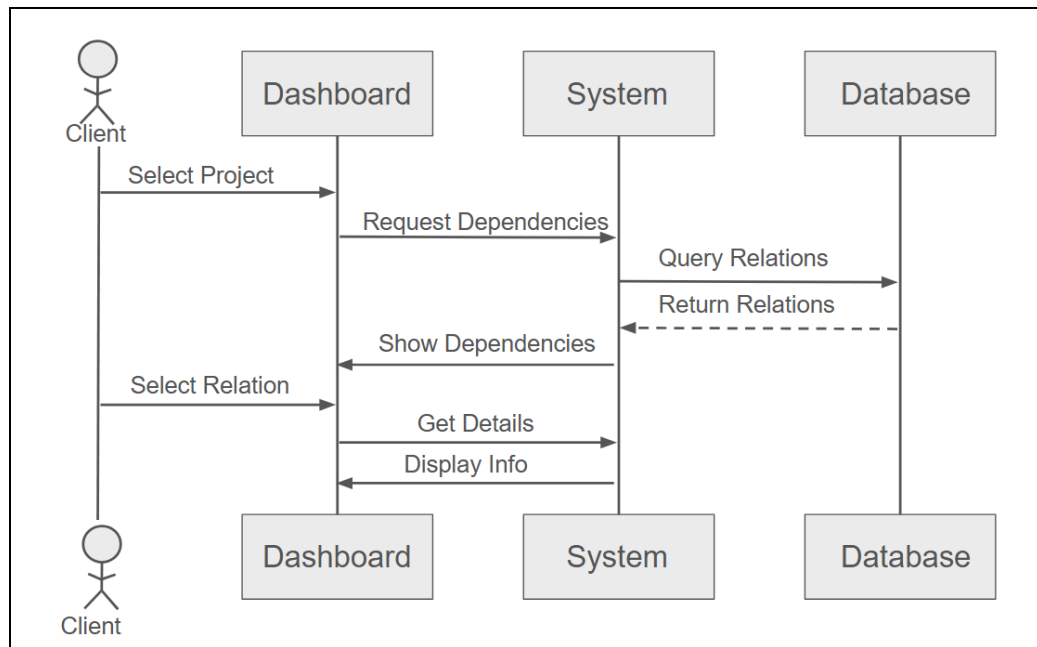
Sequence Diagram:



4. View Task Dependencies

- Actor: Client
- Precondition: Client is authenticated
- Main Flow:
 1. Client selects project
 2. Views dependency graph
 3. Selects specific relationship
 4. Reviews dependency details
- Alternative Flow: Alert if critical path affected
- Postcondition: Client understands task relationships

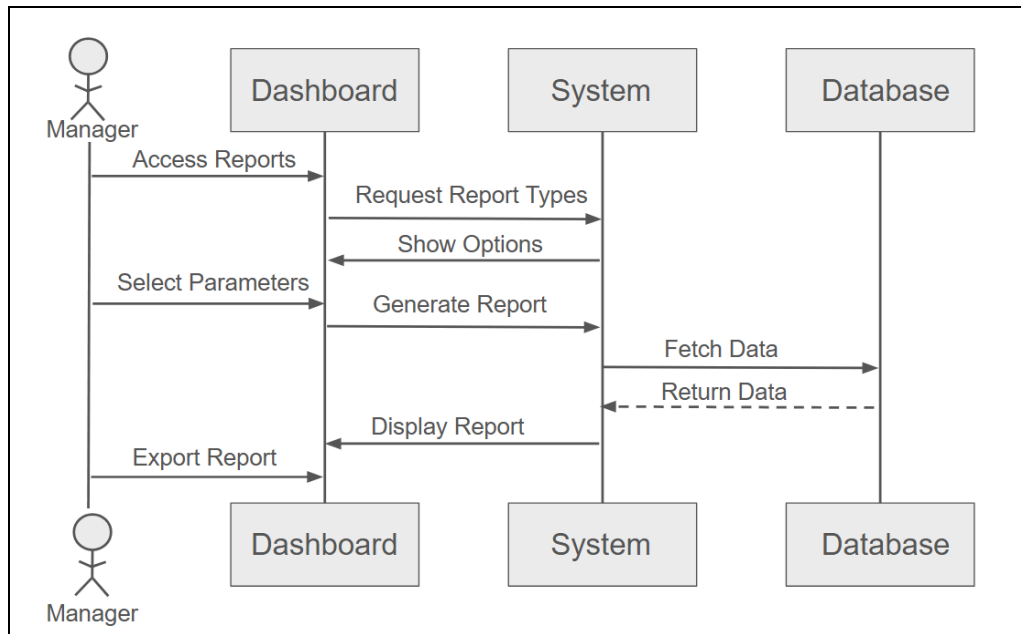
Sequence Diagram:



5. Generate Progress Report

- Actor: Project Manager
- Precondition: Manager is authenticated
- Main Flow:
 1. Manager accesses report section
 2. Selects report parameters
 3. Generates report
 4. Reviews and exports report
- Alternative Flow: Can save report templates
- Postcondition: Report generated and exported

Sequence Diagram:



Requirements Specification

1. As a customer, I want to see the progress of my project on a dashboard so I can track completed and ongoing tasks.
 - Acceptance Criteria:
 - The dashboard displays a progress bar that shows the percentage of completed and ongoing tasks for the selected project.
 - The task list is divided into "Completed" and "Ongoing" sections.
 - The customer can click on individual tasks to view additional details, such as the task description, assigned team member, and deadline.
 - The dashboard updates progress in real time as tasks are completed or moved to the ongoing section.
 - Function point analysis:
 - User Input: Selecting a project and accessing task details (+4)
 - User Output: Displaying progress bar with completed and ongoing tasks (+5)
 - Inquiry: Allowing the customer to click on tasks for more details (+3)
 - Internal Logic Complexity: Real-time progress update (+3)
 - Total Function Points: 15
 - Explanation: This requires displaying a dashboard with a progress bar, categorizing tasks and enabling real-time updates and user interaction with task details.

2. As a customer, I want to see the types of task assignments for each team member so I know who is responsible for a specific part of the project.
 - Acceptance Criteria:
 - The dashboard displays a list of team members and their assigned tasks.
 - Each task includes a label indicating the type of assignment.
 - Customers can filter tasks by assignment type to view only a specific category of work.
 - The dashboard allows customers to view assigned tasks and current status.
 - Function point analysis:
 - User Output: Display of team members with task assignments (+5)
 - Inquiry: Filter tasks by assignment type (e.g., development, testing) (+3)
 - External Interface Files: Data integration to fetch team members and tasks (+7)
 - Internal Logic Complexity: Medium complexity to organize and categorize task types per team member (+4)
 - Total Function Points: 19
 - Explanation: Workload includes displaying assigned tasks, enabling task type filters, and integrating external data sources to obtain team member information and task assignments.
3. As a project manager, I want the system to automatically sync project data from Jira so the dashboard always shows the latest information.
 - Acceptance Criteria:
 - The system synchronizes project data from Jira every 30 seconds.
 - The dashboard is updated with the latest task status and progress indicators without manual refresh.
 - If data synchronization from Jira fails, the system provides an error notification to the project manager.
 - In case of delays or failures, the project manager can manually trigger data synchronization.
 - Function point analysis:
 - External Interface Files: Integration for automated Jira sync (+7)
 - User Output: Display of updated project information on the dashboard (+5)
 - Internal Logic Complexity: Medium complexity for syncing data in intervals (+4)
 - Error Handling: Notification system for sync failures (+3)
 - Total Function Points: 19
 - Explanation: It reflects the integration with Jira, automatic data updates and real-time reflection on the dashboard. The synchronization feature requires error handling and manual synchronization capabilities, which adds complexity.
4. As a project manager, I want to see task dependencies and relationships on a dashboard so I can better understand the project's workflow and potential bottlenecks.
 - Acceptance Criteria:

- The dashboard displays task dependencies in an intuitive way, showing relationships and sequential workflows.
 - Each task displays related dependencies and identifies tasks that must be completed beforehand.
 - Dependencies are clearly labeled and critical paths in the project are highlighted.
- Function point analysis:
 - User Output: Displaying dependencies and relationships visually on the dashboard. (+5)
 - Internal Logic Complexity: Identifying sequential workflows, labeling dependencies, and calculating critical paths. (+3)
 - Inquiry: Enabling interaction to view task details, including dependencies that must be completed beforehand. (+4)
 - External Interface Files: Integration with project management tools like Jira for pulling real-time task data and dependencies. (+5)
 - Total Function Points: 17
 - Explanation: This estimate reflects the effort required to build an intuitive dashboard that shows dependencies, ordered workflows, and critical paths. The function points account for the difficulty in integrating our product with external data sources, maintaining real-time updates, and enabling interactivity to ensure project managers can effectively view up-to-date task dependencies.