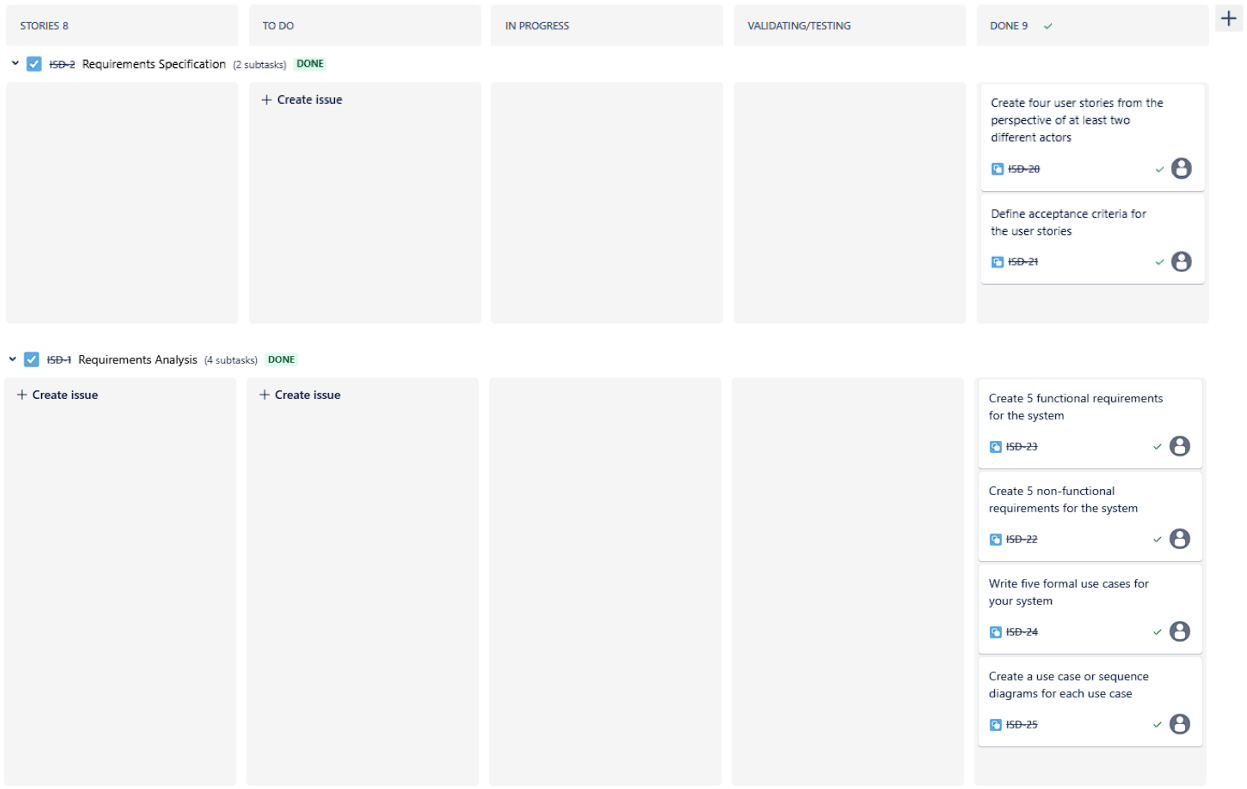
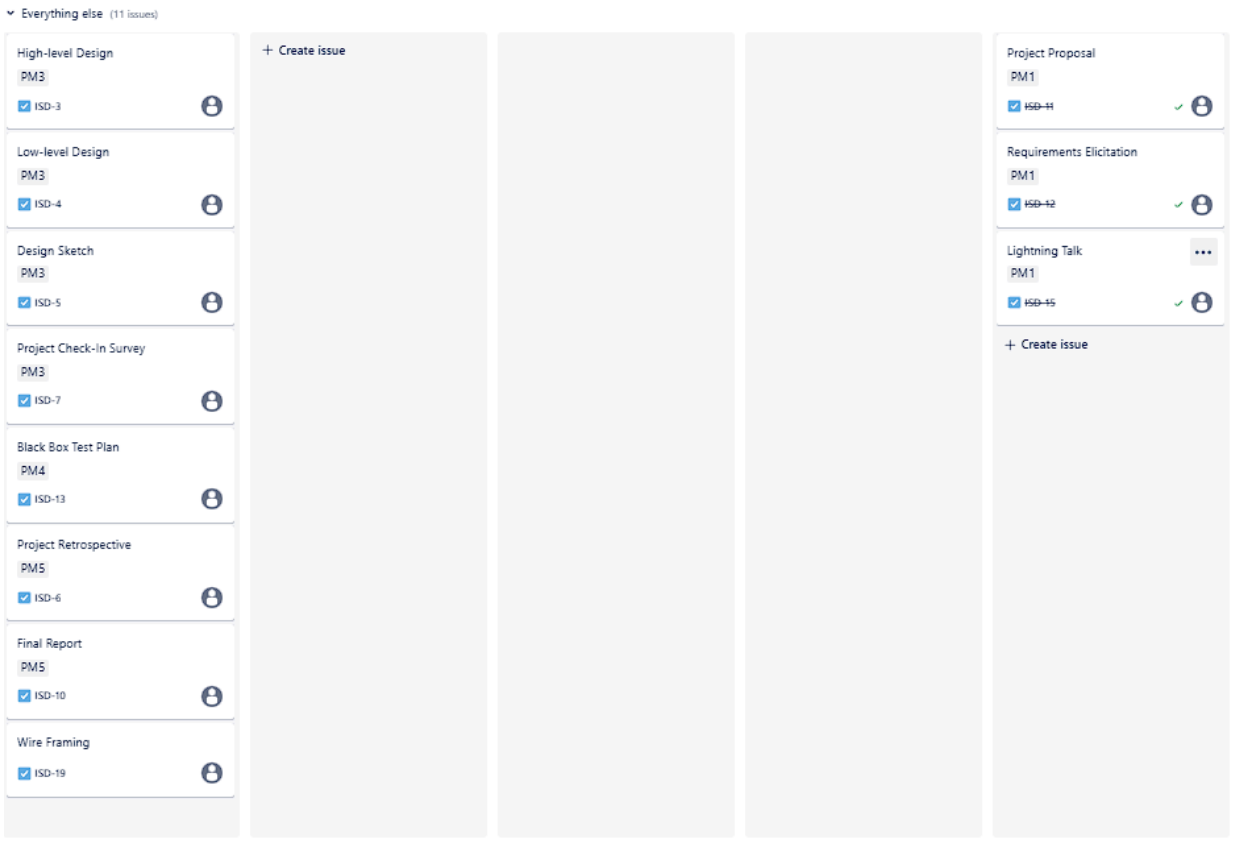
By: Samantha Austin, Adya Haydarpour, Domenic Martin, and Peyton Ludwig
Process Deliverable I
Kanban Board: https://vt-cs3704-team2.atlassian.net/jira/software/projects/ISD/boards/4





Requirements Analysis

Based on the results of your requirements elicitation, goals for your project, and course materials, please complete the following tasks:

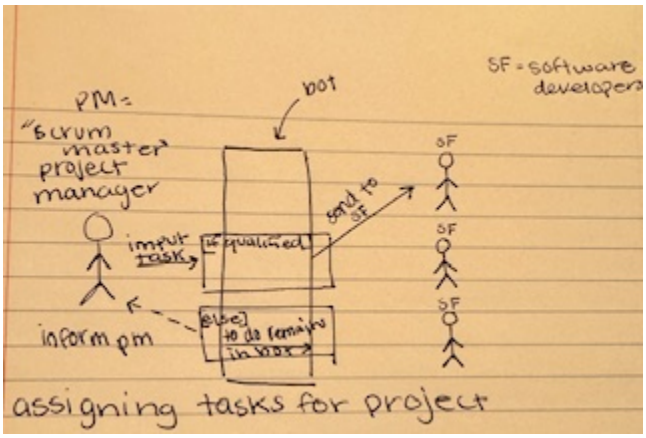
1. Provide an example of five hypothetical non-functional requirements for your system. Be sure to include the specific type of requirement discussed in class, with each requirement coming from a unique category.
    a. Usability: The bot's interface should be user-friendly, allowing developers to view tasks without navigating complex commands.
    b. Reliability: In the event of a failure, the bot should be able to recover and restore the last 5 minutes of data.
    c. Performance: Assigning new issues to developers should take less than 2 seconds.
    d. Supportability: The system should allow administrators to update developer profiles without taking the bot offline.
    e. Implementation/ Constraints: The bot must be compatible with Windows and Mac operating systems.
2. Provide an example of five hypothetical functional requirements for your system.
    a. The system must provide a user interface for users to input issues into the workflow including their effort and priority values and the skills needed to complete the task

b. The system must automatically assign issues to developers based on their current workload, skills, and experience level as well as the issue's effort value, skills needed, and priority value
c. The system must allow developers to update the stage of progress an issue is in
d. The system must constantly update a dashboard showing the current status of each issue
e. The system must identify bottlenecks in development and adjust its distribution of issues accordingly

3. Write five formal use cases for your system and provide use case or sequence diagrams to represent each use case.
   a. 1. Precondition
      User must have KanBan issue tracking device software downloaded onto their system.
      2. Main Flow
      User will request more work to do [S1]. Bot will check what is left to do and if qualified[S2]. Bot will push task to do list [S3].
      3. Subflows
      [S1] User will need to provide username and password.
      [S2] Bot will need to check user qualifications and find matching to dos
      [S3] Bot will confirm a task for user and submit to their to do list.
      4. Alternative Flows
      [E1] user is unqualified for any of the tasks therefore given none.
   b. 1. Precondition
      User must have KanBan issue tracking device software downloaded onto their system.
      2. Main Flow
      User will request less work to do [S1]. Bot will check on whats user has left and if their are others qualified [S2]. Bot will redirect something on the users to do list to someone else[S3].
      3. Subflows
      [S1] User will need to provide username and password.
      [S2] Bot will need to check user qualifications and find another user to take on said task.
      [S3] Bot will confirm that task being moved with user. Then move to another user.
      4. Alternative Flows
      [E1]No one else is qualified at the moment therefore the task is left unassigned.
   c. 1. Precondition
      Project manager must be assigned to project and titled in KanBan software
      2. Main Flow
      Bot will notify the project manager if a task has been moved around more than 3 times [S1]. Project manager will find a qualifying developer to place task with[S2]. Bot will clear the warning on the list[S3].
      3. Subflows
      [S1] Bot will need to have project manager email contact to inform

[S2] Project manager will need to discuss and work with the team to figure out why the task has been moved so many times.
[S3] Project Manager will manually assign the task to a user in the project group
4. Alternative Flows
[E1] No one is willing to take the task.

d. 1. Precondition
Must have all users and project manager assigned to same kanban project.
2. Main Flow
Users and project manager will input their information[S1]. Project manager will insert tasks for project[S2]. Bot will organize and distribute tasks to everyone in the group [S3].
3. Subflows
[S1] Users will need to provide qualifications
[S2] Project manager will be asked by bot why level of qualification each task has.
[S3] Bot will one by one move a task to each developer in the group
4. Alternative Flows
[E1] There will be only one task to distribute therefore it goes to most qualified.

e. 

Requirements Specification

Based on the results of your requirements elicitation, goals for your project, and course materials, please complete the following tasks:

1. Write four user stories from the perspective of at least two different actors.
Provide the acceptance criteria for these stories.

1) "**As a** developer, **I want to** receive tasks that match my skills and current workload **so that** I can focus on work suited to my experience and availability."
Acceptance criteria:
   - **Given** the system has information on developer skills and workload
   - **When** a new task is submitted,

- **Then** the system assigns the task to a developer based on their skills, experience, and current workload
- **And** the task assignment occurs within 2 seconds.

2) "**As a** developer, **I want to** have the ability to update the progress of my tasks **so that** the project manager can track real-time progress."
   Acceptance criteria:
   - **Given** the developer is working on an assigned task
   - **When** the developer updates the task's status
   - **Then** the task status is updated on the dashboard within 5 seconds
   - **And** the progress updates do not disrupt other ongoing tasks.

3) "**As a** project manager, **I want to** be notified when a task has been reassigned multiple times **so that** I can reassign issues when necessary."
   Acceptance criteria:
   - **Given** a task has been reassigned 3 times
   - **When** the task is reassigned for the 3rd time,
   - **Then** the project manager receives a notification with details on task and developer assignment history
   - **And** the project manager can manually reassign an issue.

4) "**As a** project manager, **I want to** input tasks and required qualifications **so that** the system can assign tasks accurately."
   Acceptance criteria:
   - **Given** the project manager inputs a new task,
   - **When** the task is entered with effort, priority, and required skills,
   - **Then** the system assigns the task within 5 seconds based on the provided qualifications,
   - **And** the assignment is logged for reference.

2. For each user story mentioned above, estimate the amount of effort needed to complete relevant subtasks using function points. Explain your answer.

- User Story 1 - developers receiving tasks:
  - User Interface for input - 2 function points - interface for task input
  - Assignment algorithm - 4 FP - requires logic for matching tasks to developers based on skills and workload
  - Database search - 3 FP - retrieves developer data to aid assignment process
  - Assignment algorithm has most FPs because it has the most complicated logic, interface has least because least complicated

- User Story 2 - progress update:
  - Update interface - 2 FP - interface to update task progress
  - Update database - 2 FP - saves progress data in real time

    - Update dashboard - 3 FP - ensures updates are reflected on dashboard
    - All have similar levels of difficulty

- User Story 3 - reassignment notification:
    - Notification System - 3 FP - notifies project manager after task is reassigned multiple times
    - Logging history - 2 FP - logs assignment details & history
    - Dashboard view - 2 FP - provides view of dashboard
    - Similar levels of complexities, with logic for notification system being most complex

- User Story 4 - task input & qualification match:
    - Task input - 3 FP - Form for project manager to enter new tasks & qualifications
    - Qualification check - 4 FP - processes specified qualifications
    - Assignment logging - 2 FP - logs task assignments
    - Most complicated is the logic for the qualification check algorithm, least complicated is logging logic