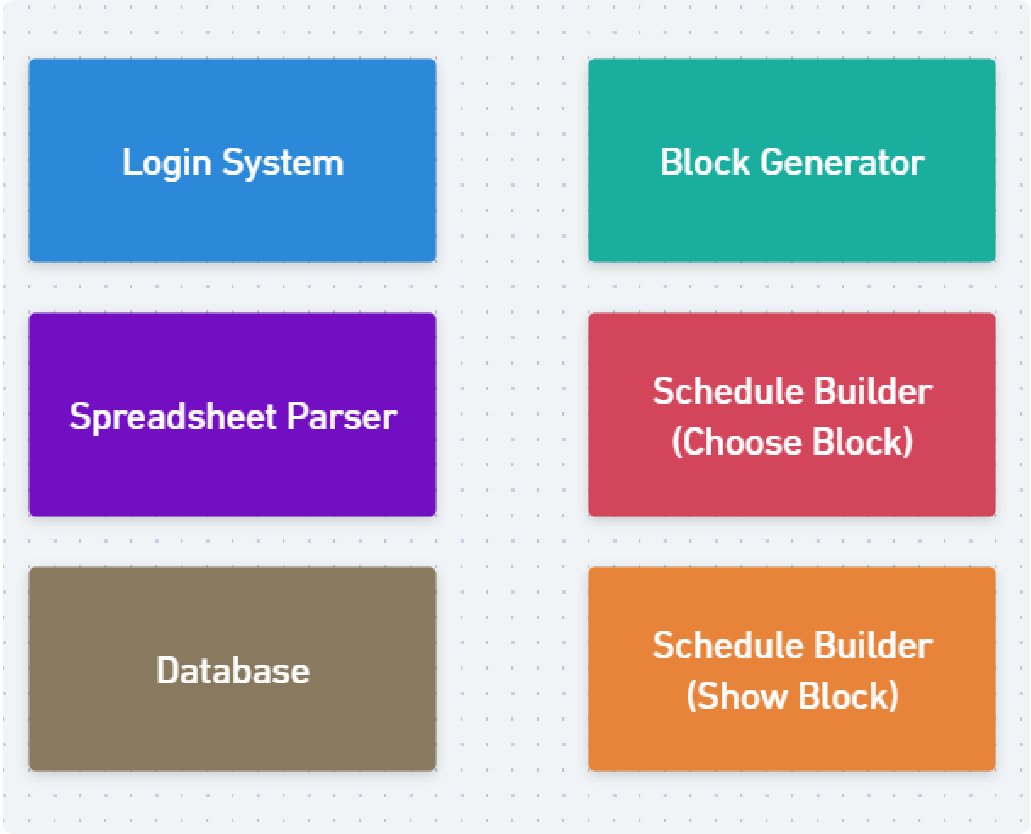
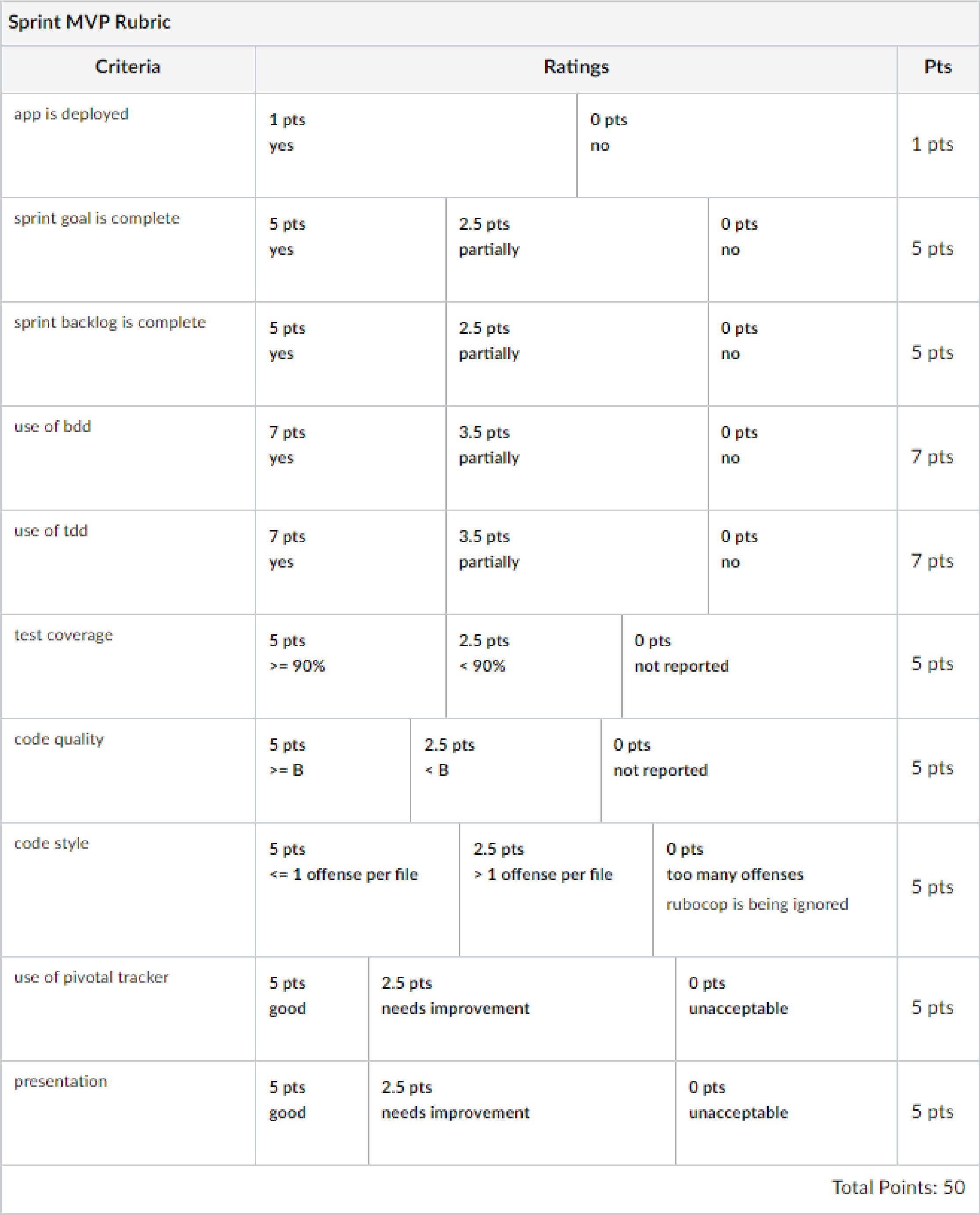
Sprint 1 MVP Design Document

Overview


Login System
Block Generator
Spreadsheet Parser
Schedule Builder (Choose Block)
Database
Schedule Builder (Show Block)

- In sprint 1, we split the block scheduling tool into 6 different parts.
  - **Each developer should focus on implement features for one indepedent system in this sprint.**
  - We will integrate them in the future.
- Each developer should complete **at least one feature** based on user story.
  - **Pls write down user stories** before you start working on the feature because user stories will help a lot in the test phase.
- Note that the criteria includes **code quality, code style, and test coverage**
  - Each developer should write unit tests for their features.
  - Use `simplecov` to check your test coverage (like PA3)
  - Run `rubocop` and `rubycritic` to check your code -quality (like PA3)
- **Convention:** In user stories, we use three roles: `user` , `admin` , and `developer`

- Feel free to discuss the document with Scrum Master.
- If you cannot deliver the feature on time, **PLEASE INFORM YOUR TEAM IN ADVANCE**. We understand that unexpected issues can arise, but it's important to keep us informed.



Sprint MVP Rubric

Criteria	Ratings			Pts
app is deployed	1 pts yes	0 pts no		1 pts
sprint goal is complete	5 pts yes	2.5 pts partially	0 pts no	5 pts
sprint backlog is complete	5 pts yes	2.5 pts partially	0 pts no	5 pts
use of bdd	7 pts yes	3.5 pts partially	0 pts no	7 pts
use of tdd	7 pts yes	3.5 pts partially	0 pts no	7 pts
test coverage	5 pts >= 90%	2.5 pts < 90%	0 pts not reported	5 pts
code quality	5 pts >= B	2.5 pts < B	0 pts not reported	5 pts
code style	5 pts <= 1 offense per file	2.5 pts > 1 offense per file	0 pts too many offenses rubocop is being ignored	5 pts
use of pivotal tracker	5 pts good	2.5 pts needs improvement	0 pts unacceptable	5 pts
presentation	5 pts good	2.5 pts needs improvement	0 pts unacceptable	5 pts

Total Points: 50

Initialize and deploy - Adithi

Sprint 1 MVP

1. Create the basic app and deploy it

I. Login System - William

Sprint 1 MVP

1. Make basic UI for the login page(Button, Header...)

User Stories

	Step1	Step2	Step3	Step4	Step5
1	As a user	When I open the login page	I see the button for sign up		
2	As a user	When I open the login page	I see the button for sign in		

Backlog

2. Use 3rd-party authetication
3. Design the basic user profile page

II. Spreadsheet Parser - Mahima

Sprint 1 MVP

1. Admin could upload the spreadsheet
2. The spreadsheet should be stored in database

User Stories

	Step1	Step2	Step3	Step4	Step5
1	As an admin	When I open the spreadsheet upload page	I can see the upload button	Then I press the button to upload file from local	I see the file has been successfully uploaded
2					

Backlog

3. Spreadsheet can be parsed correctly. After that, data will be stored in database.

III. Block Generator - Junhyuk

Sprint 1 MVP

1. Generate schedule algorithm as list - ex: (MWF 10a. MWF 10:30 <- these shouldn't overlap)
    - When I input mock class data, I should see a list of possible schedules with a simple algorithm, such as separating overlap

User Stories

	Step1	Step2	Step3	Step4	Step5
1	As a developer	I add a class on MWF 10am-11am	I add a class on MWF 10:30am-11:30am	When I generate the schedule	Then I see two possible schedules
2					

Backlog

IV. Database - Aaron

Design the database schema based on client's requirement.

- Reference

1. Design a data structure for classes to keep track of my classes and requirements
    - Term
    - Department Code
    - Section Name
    - Bldg, Room
    - Start Time, End Time
    - ...
2. Design a data structure for Student profile data
    - Name
    - UIN

    - email
    - Classes already taken (For prerequisites)
    - ...
3. Implement the data structure using sqlite3

User Stories

	Step1	Step2	Step3	Step4	Step5
1	As a developer	I create a student named Bob	When I retrieve Bob's data from database	I can see the name 'Bob'	
2					

Backlog

V. Schedule Builder (Show Block) - Ryann

Sprint 1 MVP

1. Show example schedule chart (time, days of week)

User Stories

	Step1	Step2	Step3	Step4	Step5
1	As a user	I want to see my schedule chart	So I open the schedule viewer page	Then I can see time blocks on chart (Unit Block: 15 min)	And I can see class blocks shown correctly
2					

Backlog

2. The chart can show different classes with differen colors
3. You can check the class detail by moving cursor on the class block

VI. Schedule Builder (Choose Block) - Adithi

Sprint 1 MVP

1. Build basic UI page like Aggie Schedule Builder

  - It will show the selected classes
  - It has a `Generate` button for generating schedule
  - After generating possible schedules, user can view the schedule through `View` Button
  - ...

User Stories

	Step1	Step2	Step3	Step4	Step5
1	As a user	When I open the UI schedule builder page	I can see the classes I selected		
2					

Backlog