

Group 4

Sprint 1 Retrospective

On Monday 27th January, our team held an extended meeting to focus on building a productive and informative retrospective on the work completed during the first sprint of the project.

We utilised two separate approaches for our retrospective in order to gain a more well rounded understanding of what worked well, where we believed there was room to improve, as well as our actionable items for the week ahead.

Using <https://www.funretrospectives.com/> we developed these two boards collaboratively to better organise the teams' thoughts:

1 Humble Pie - Admit a wrong that occurred and apologize for the mistake. <small>Write the content of your card to insert it into the board</small>	2 Shoo Fly Pie - Identify something negative that is hampering the team and you want to remove. <small>Write the content of your card to insert it into the board</small>	3 Cutie Pie - Submit an appreciation for someone or something. <small>Write the content of your card to insert it into the board</small>	4 Pie in the Sky - Present an idea that may be 'impossible' - a stretch goal, fanciful idea, or plan. <small>Write the content of your card to insert it into the board</small>	5 Easy As Pie - Recognize something that was simple and pleasurable that you believe the team should continue doing. <small>Write the content of your card to insert it into the board</small>
I should have made commits atomic	Didn't plan thoroughly enough on Monday	Everyone always shows up to the meetings	role swap so we understand each others roles better	Everyone was easily reached and responded to messages throughout the work day
Should have been more strict about commits on GitHub that didn't have adequate description and layout.	not having the project board up at meetings meant we were less clear about where we were in the overall sprint	Overall, communication between group members was consistently good and we helped each other when needed	turn the daily meeting into a game show	Being quick and effective when communicating about issues being faced
I did not apply cromulent effort when creating and refining user stories and as such, they were not up to par. Apologies	definition of done varied between different user stories and wasn't ticked off	Merylou was very good with handling merge conflicts and resolving errors	Agile escape room to promote teamworking	All attending daily stand ups
	Only doing tasks you are assigned so things don't end up done twice		Throughput on software development being as high as it was in Sprint 1	Progress made toward the software was efficient and fast-paced
	Lacking GitHub management left ambiguity as to what everyone's specific tasks were			
	Branch names and branch structure was disorganised and inconsistent			

Group 4

1 Future directions: Please write down all future directions regarding the project/phase. <small>Write the content of your card to insert it into the board</small>	2 Lessons learned: Please write down the key lessons and takeaways from the project/phase. <small>Write the content of your card to insert it into the board</small>	3 Accomplishments: Please write down the key accomplishments for the project/phase. <small>Write the content of your card to insert it into the board</small>	4 Problem areas: Please write down the problematic areas experienced throughout the specified project/phase. <small>Write the content of your card to insert it into the board</small>
Plan out sprint reviews so it follows an order	use the project board at the meetings so we are all clear on where we are at everyday	Successfully held daily stand up meetings with full group attendance	Sprint review was a bit slapdash and rough around the edges.
Allow the ATM to handle international bank accounts, taking into consideration currency conversion	have a branch for every user story and sub branches for every task for better organisation	Created log files to store transaction data	Ambiguity about who specific tasks were assigned to
Be stricter about GitHub usage - i.e. require descriptions for pull requests and proper user story formatting	Give specific and informative descriptions for commits for easier GitHub management	Created a useable user interface for the ATM	
Smaller pieces of functionality tackled at a time	Assign tasks for every user story on the github and only work on tasks we are assigned so there is less confusion and repeated work	Created a meaningful format for transaction data that could be used throughout the flow of data	
Create multiple simulated networks containing distinct groups of accounts and correctly route data to these in the transaction switch	Merge earlier to deal with conflicts earlier and be merging smaller things together	Allowed communication between all three elements of the system	
Have more emphasis on agile methodology as opposed to software development		hosted network simulator and transaction switch on AWS	
		Created network simulator	
		Created transaction switch	
		Created ATM client	

Compiling these two boards helped us to come up with the following:

What worked well

- We held daily meetings throughout the week during which we had informative discussions and tracked progress.
- Communication within the group was effective and consistent, which was incredibly helpful for our progression.
- We accomplished a high level of tangible progress in the development of our software artefact.
- Our internal discussions always resulted in each member of the group feeling better informed.

What could be improved

- Sprint reviews with the client should follow a more structured approach and be better planned out to improve the flow of our conversation and presentation with them.
 - We believe that during our second sprint review, we did improve in this area but that some aspects, such as setup and demonstration, could still be improved.
- Strictness about the usage of GitHub and the rules team members are to follow should become more defined as we move forward.
 - This was accomplished through the addition of additional templates on GitHub – namely for pull requests – and the refusal to accept requests and new issues if they did not follow the proper format.
- User stories should be cut down to smaller pieces of work in order to ease distribution of tasks.
 - During our daily meetings, we made it a priority to handle issues in an atomic way and any new User Stories were developed with this in mind.
- Fully signing off Definition of Done under every applicable task to ensure good integration and testing as well as a better understanding of the progress of each task.
 - We made an effort to keep a closer eye on tasks being moved across the project board and kept in contact with the assignees to ensure the DOD was being properly followed and signed off.
- Utilise our first day to perform more planning action in relevance to our approach to the sprint and ensure everyone is on the same page.
 - We had an especially long meeting on Monday to discuss and plan out the work for the week, after which nobody started on actual development until after Tuesday's meeting.

Group 4

- Overall, have more emphasis on agile methodologies as opposed to software development.
 - We made an effort to spend more time planning and strategizing this sprint, and less time on actual development. We feel this is possibly still an area in which we could improve, but that we made a distinct effort that paid off nonetheless.
- Make use of the project board during daily meetings.
 - This was helpful as it allowed us to better understand and visualise what tasks would be worked on during the day.

What will be accomplished in the next sprint

During the sprint 2 we will strive to work on our areas of improvement via the following actionable items:

- Build issues as user stories:
 - Each issue is a user story which is labelled accordingly.
 - Each sub issue should have a written description, and be labelled as a task on GitHub.
 - Create different project views for different sprints, only user stories, only tasks, etc.
 - Assign people to different issues and sub issues.
 - Include epics for an overall picture of what we are working towards achieving.
 - Consistency and effort have been added to all stories.
 - Add additional stories if needed as the project evolves.

Overall, our usage of GitHub and breaking down of user stories into actionable tasks has improved greatly over the course of the second sprint. We did include epics during this sprint, however they could have been utilised more effectively. Aside from that, we believe that we managed to implement this set of items successfully.

- Implement a structured approach to the sprint review:
 - Welcome the clients, introduce ourselves, and set the scene by detailing what we covered throughout the sprint, what our goals were and how we achieved them.
 - Give an overview of what we will demonstrate to the clients.
 - Effectively demonstrate the functionality.
 - Discuss any major events or problems that occurred throughout the sprint.

Group 4

- Present our upcoming backlog to illustrate what we plan on working on next.
- Conclude by requesting any questions the client may have and discuss with them.

Our greeting and introduction to the client was better than that of sprint 1, but still could have been more effectively structured. Such as the way in which we gave our overview of the week as we went through the presentation instead of all at the beginning. Our demonstration of new functionality was, for the most part, smooth and to the point. Major events and/or problems were briefly brushed over, but not discussed in any depth, which is something that could be improved upon. Similarly, we could have enhanced the presentation with a more detailed explanation of our next steps, along with a showing of our GitHub backlog. Hence overall, our second sprint review was better than the first, but still left room for improvement.

- Create stricter GitHub rulesets to make tracking progress and documentation easier:
 - Require atomic commits.
 - Require commits to have messages and comments where necessary.
 - Use git branches more efficiently to aid with integration of features from those branches, i.e. one branch for each issue and sub issue.
 - Standardise issues and pull request formats to enforce proper documentation of features and progress.

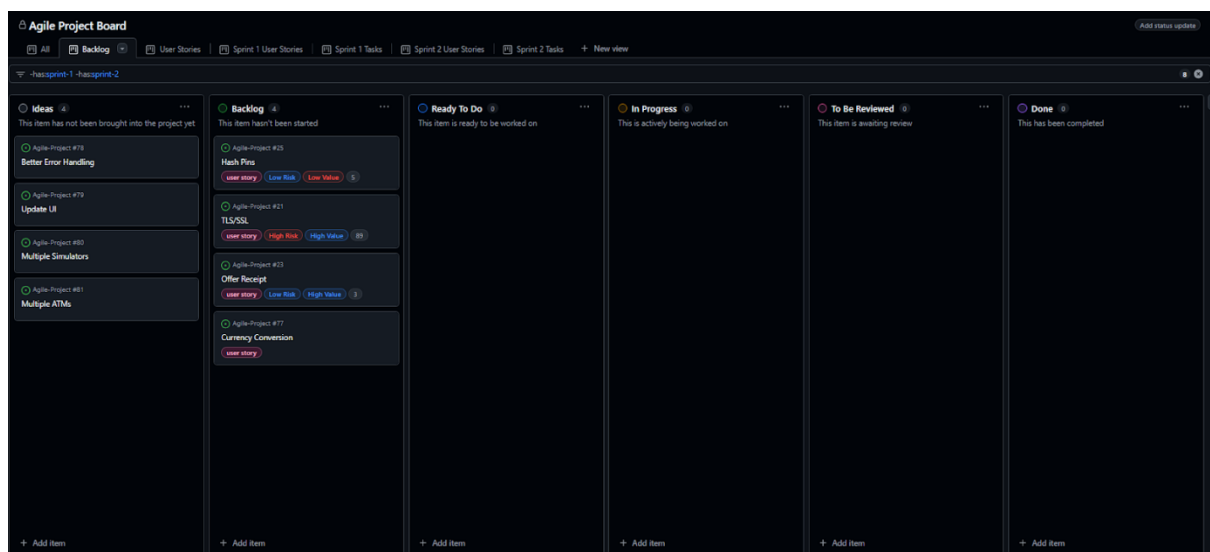
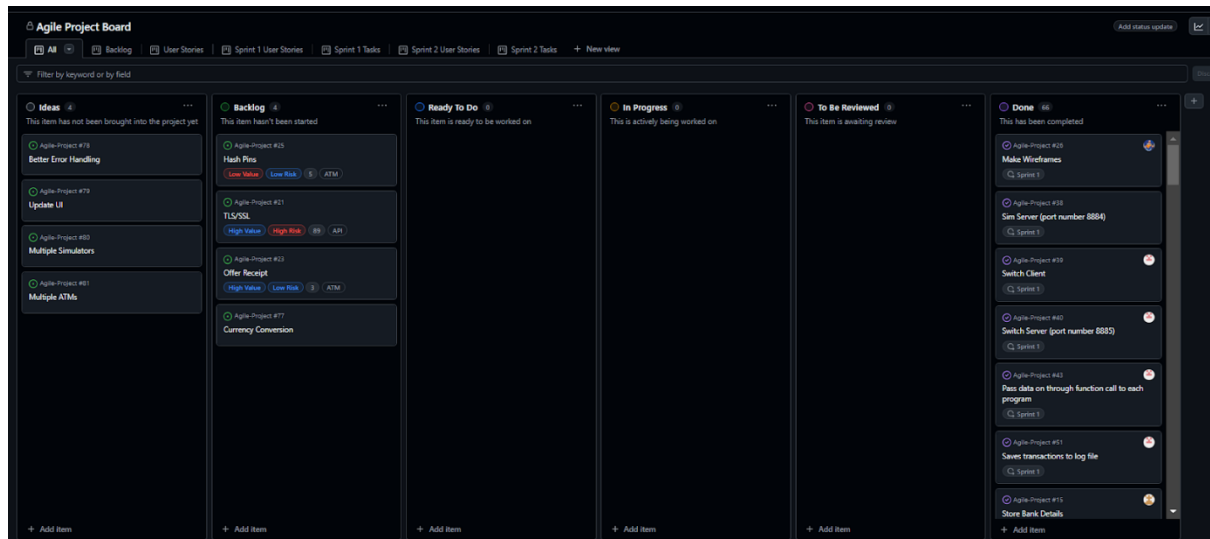
All of these new requirements were put into action at the very start of the second sprint and were adhered to successfully by every team member throughout the week. Near the very end of the sprint, a handful of quick small bug fixes were accepted without thorough documentation, but we as a team decided to allow this in order to quickly tidy up minor loose ends.

Group 4

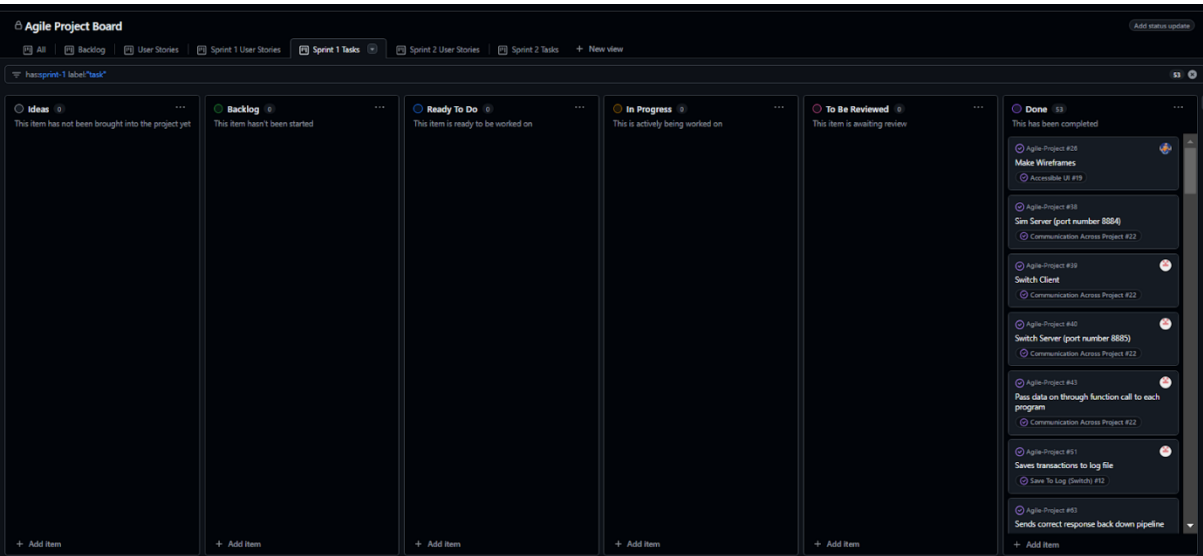
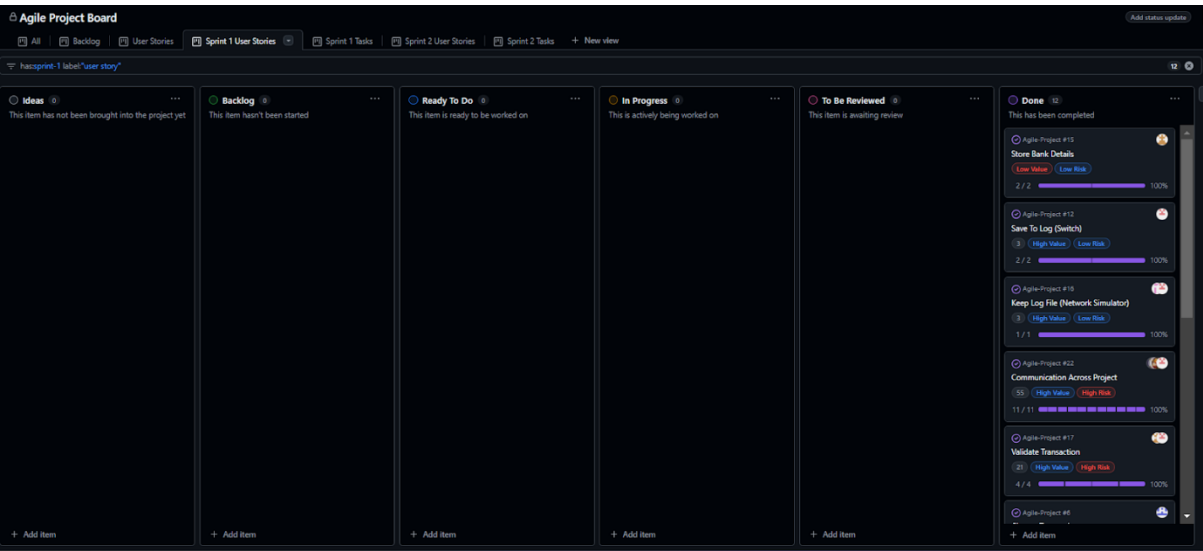
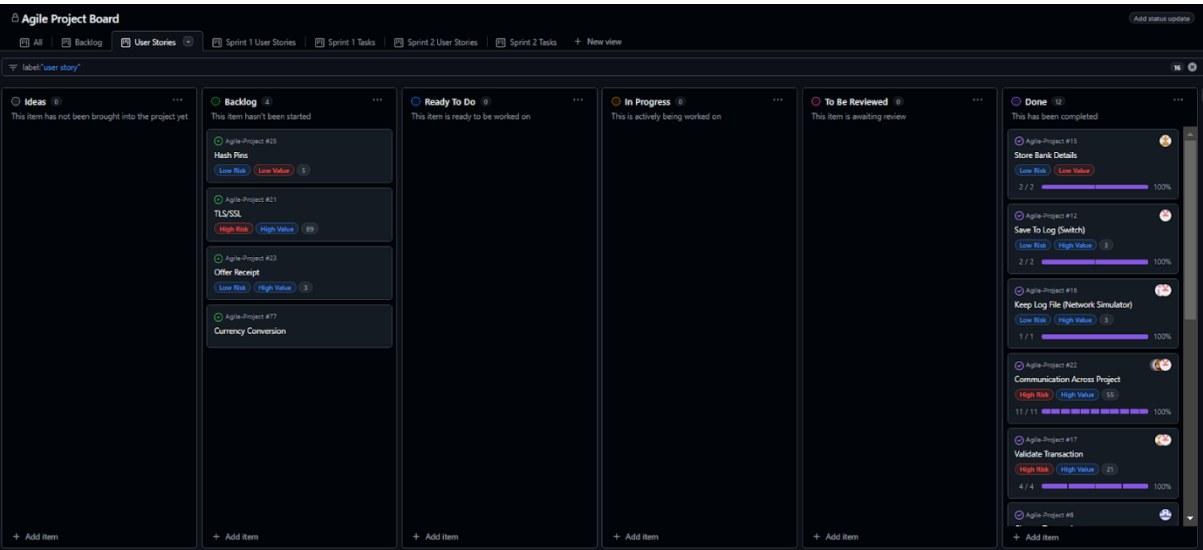
Management of Backlog

After every meeting, screenshots were taken of our GitHub project board in order to keep an accurate note of our management of the project backlog.

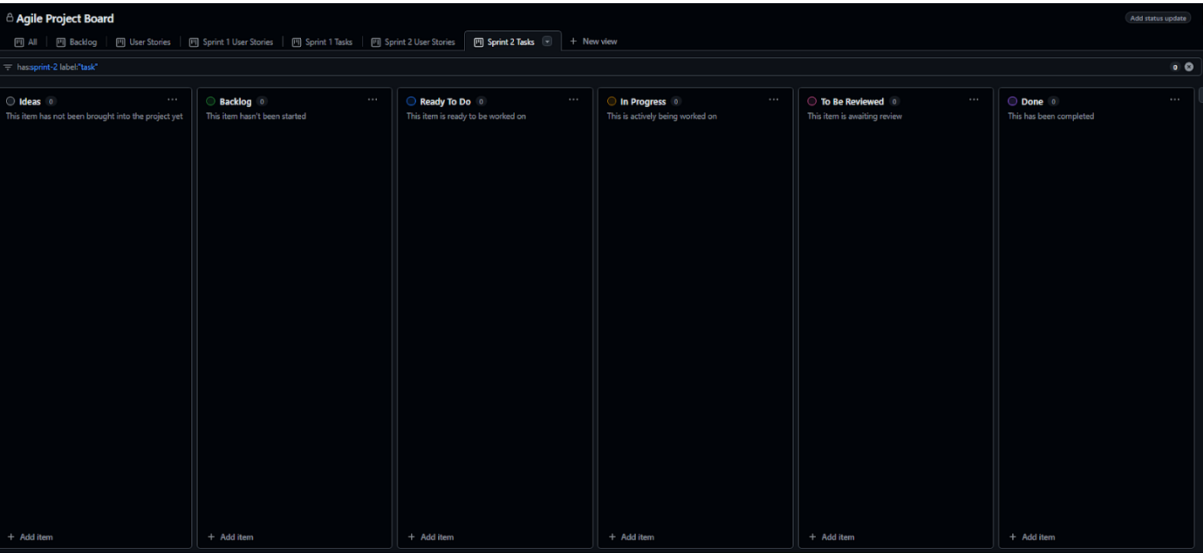
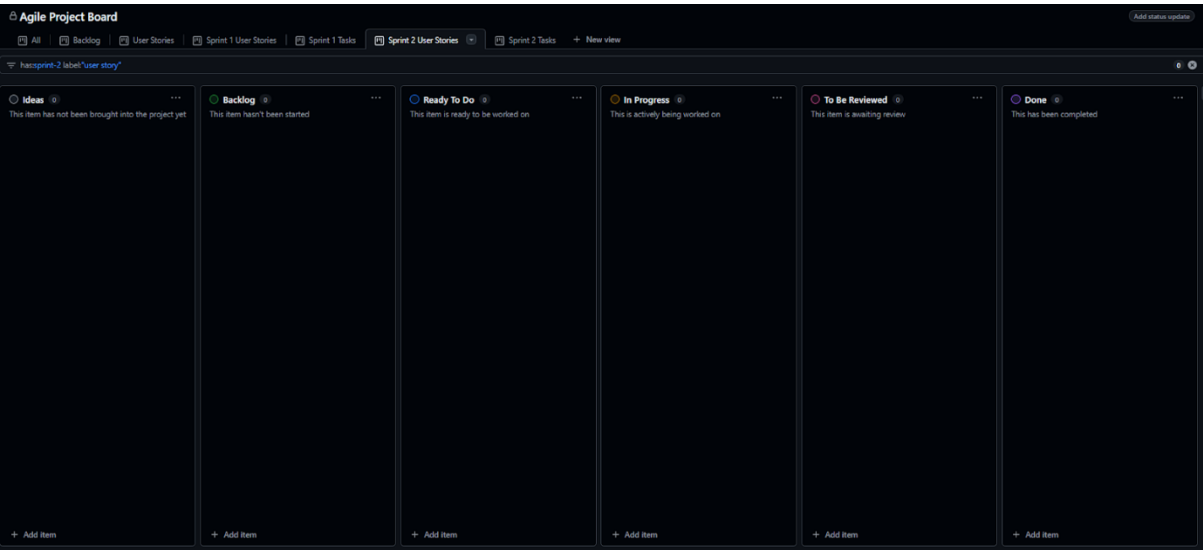
Post Sprint 1



Group 4

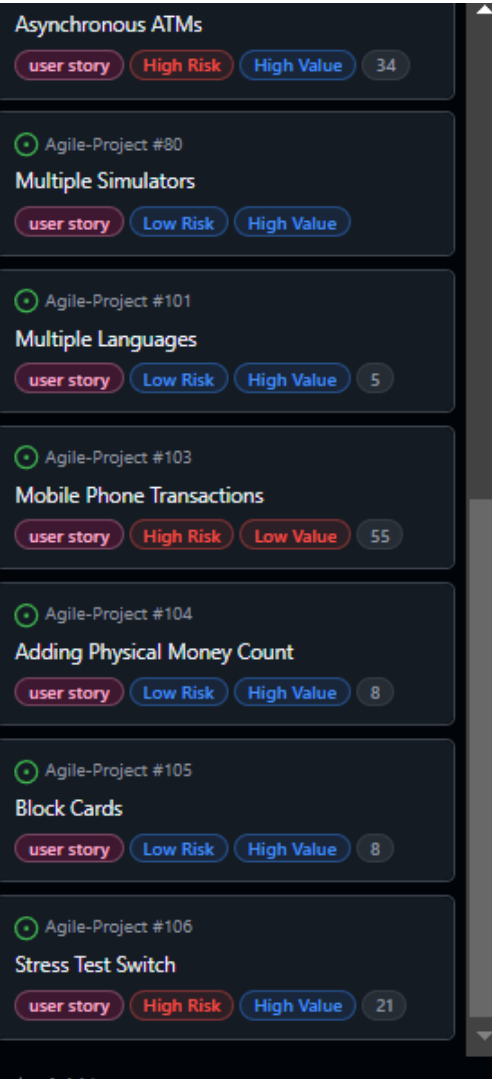
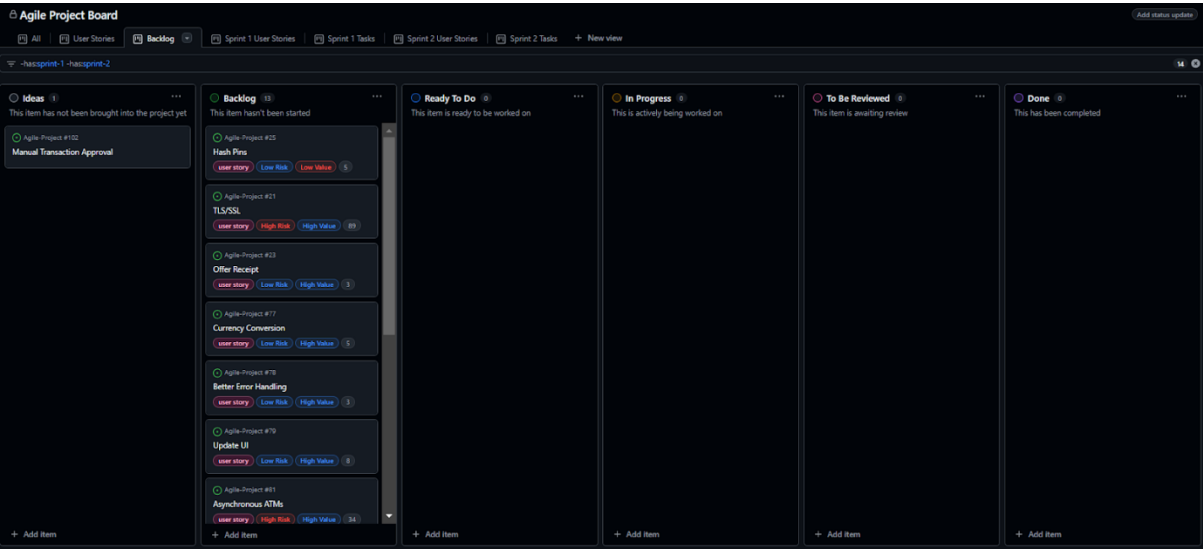


Group 4

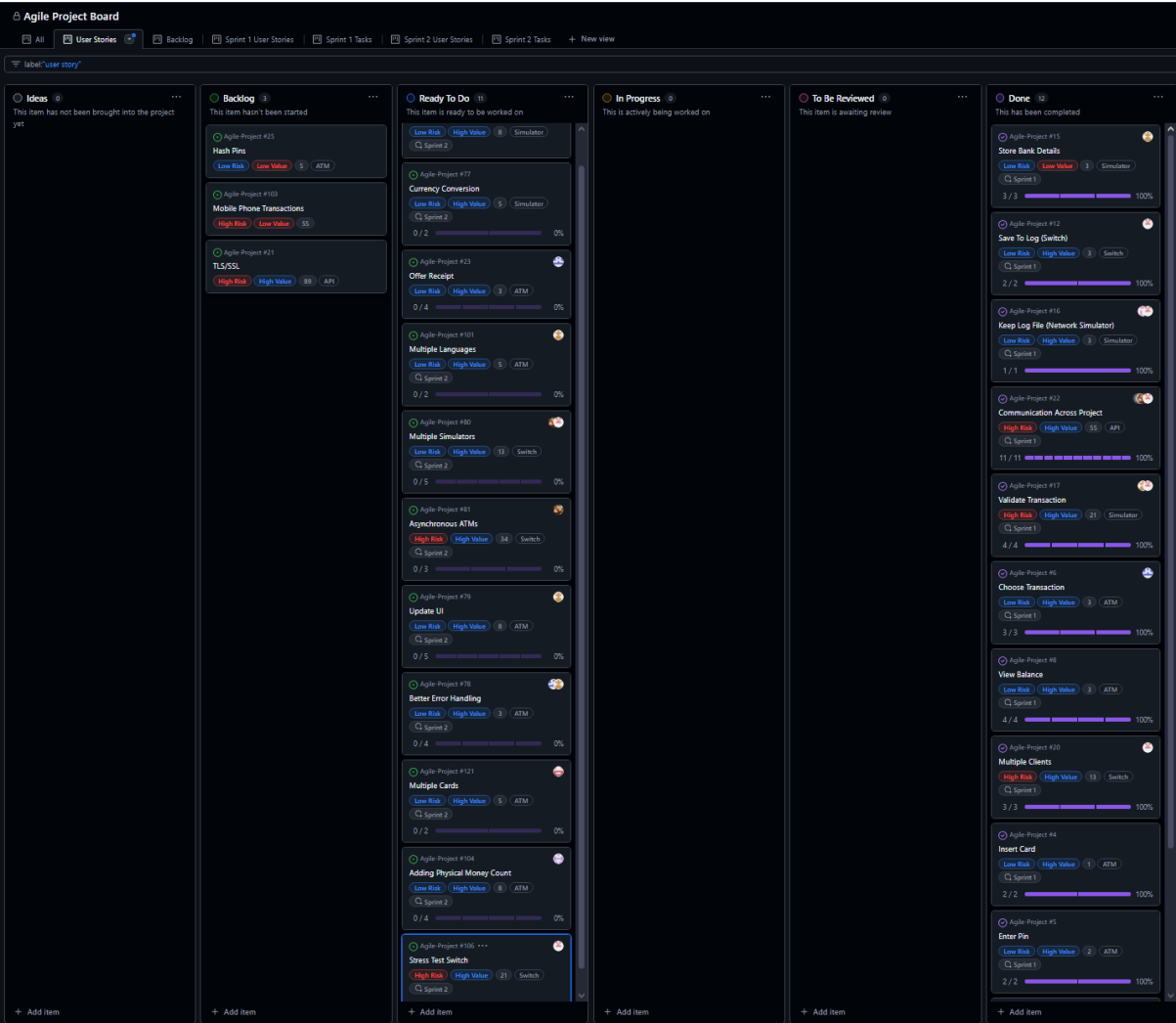


Group 4

Monday 27/01/25 Stand Up

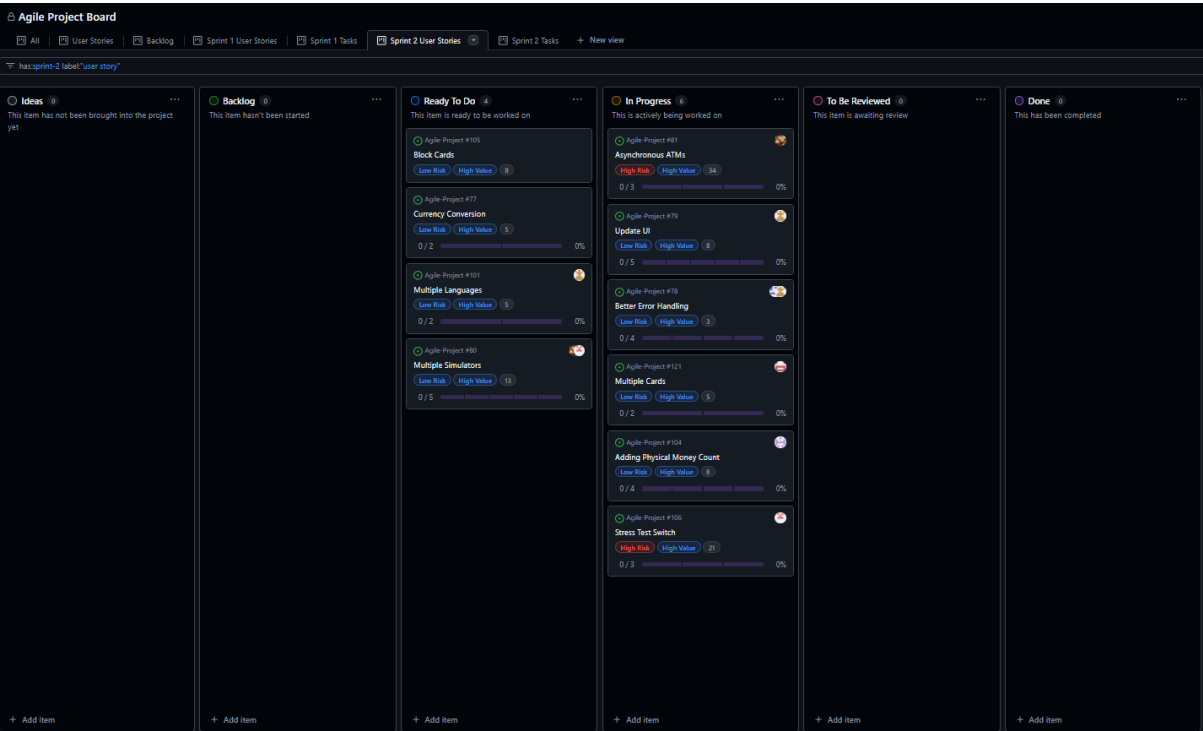
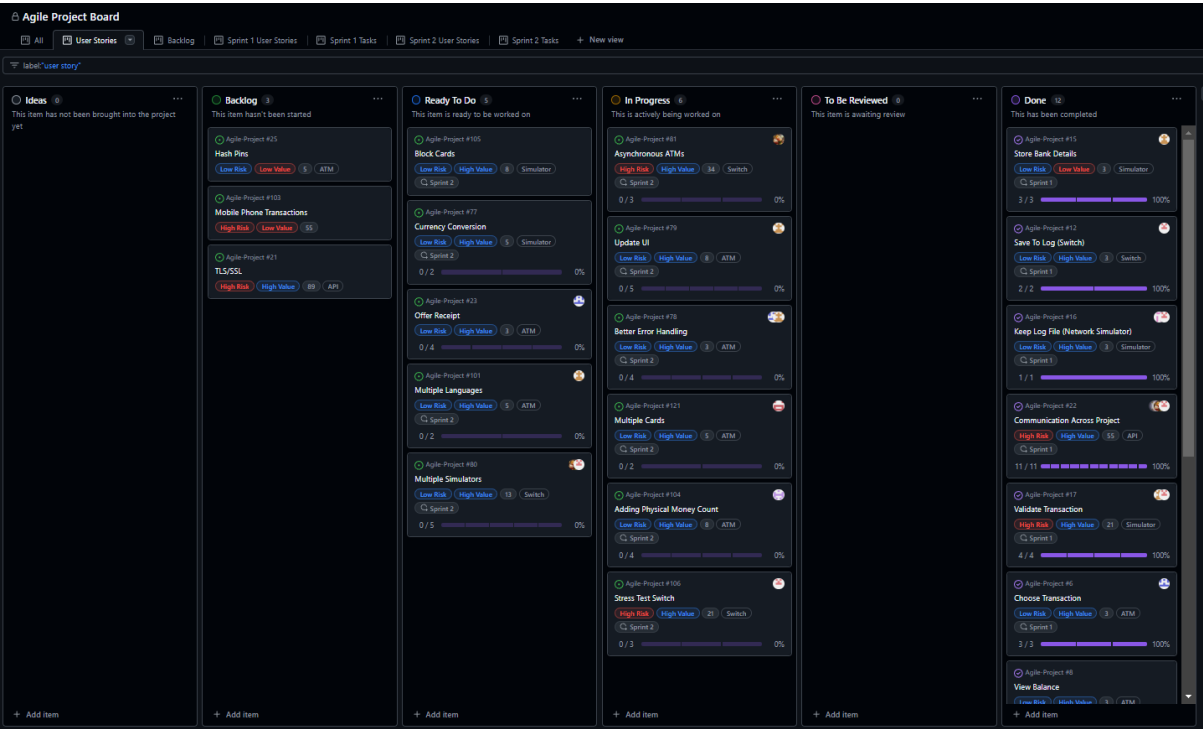


Group 4



Group 4

Tuesday 28/01/25 Stand Up



Group 4

Agile Project Board

All

User Stories

Backlog

Sprint 1 User Stories

Sprint 1 Tasks

Sprint 2 User Stories

Sprint 2 Tasks

New view

has sprint-2 label="task"

Ideas

This item has not been brought into the project yet

+ Add item

Backlog

This item hasn't been started

+ Add item

Ready To Do

This item is ready to be worked on

Agile Project #111

Refactor code to reflect asynchronous traffic

Asynchronous ATMs #81

Agile Project #112

Ensure procedures non-blocking

Asynchronous ATMs #81

Agile Project #113

Transactions are returned to the ATM from which they were sent successfully

Asynchronous ATMs #81

Agile Project #114

Test switch can handle 100 ATMs sending 2 transactions per ATM per minute for 5 minutes

Stress Test Switch #106

Agile Project #115

A second application should be made to simulate this environment

Stress Test Switch #106

Agile Project #116

As part of the simulation the number of ATMs and transactions should be variable

Stress Test Switch #106

Agile Project #122

Insert card prompt is updated to feature multiple cards

Multiple Cards #121

Agile Project #123

Choice of card affects bank details sent with transactions

Multiple Cards #121

Agile Project #125

Increase size of UI elements

+ Add item

In Progress

This is actively being worked on

+ Add item

To Be Reviewed

This item is awaiting review

+ Add item

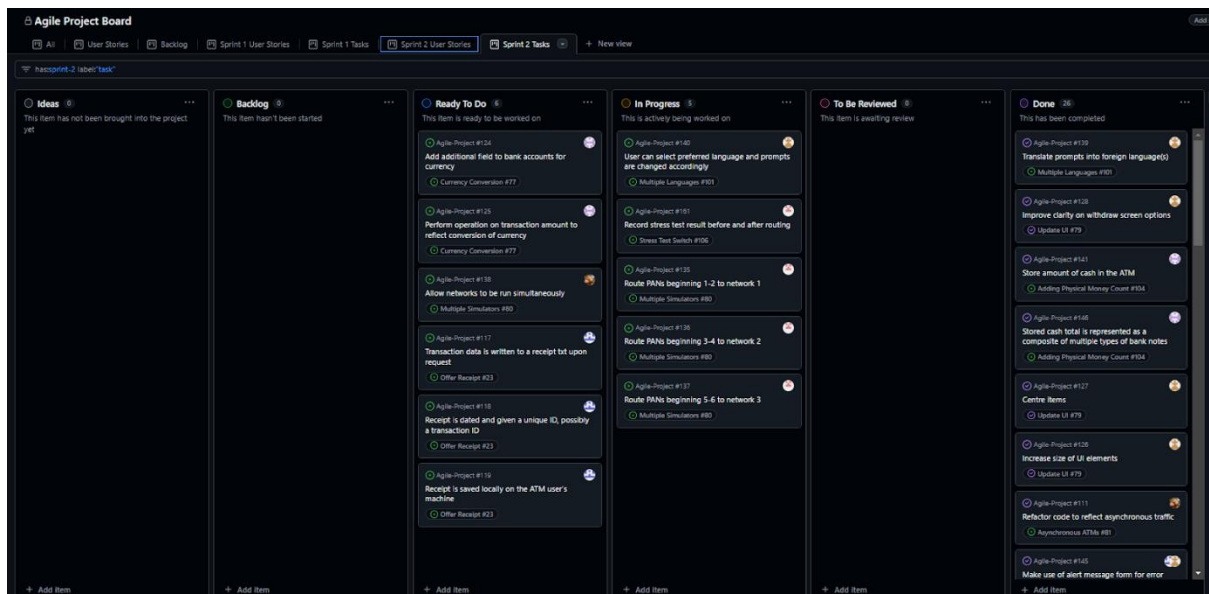
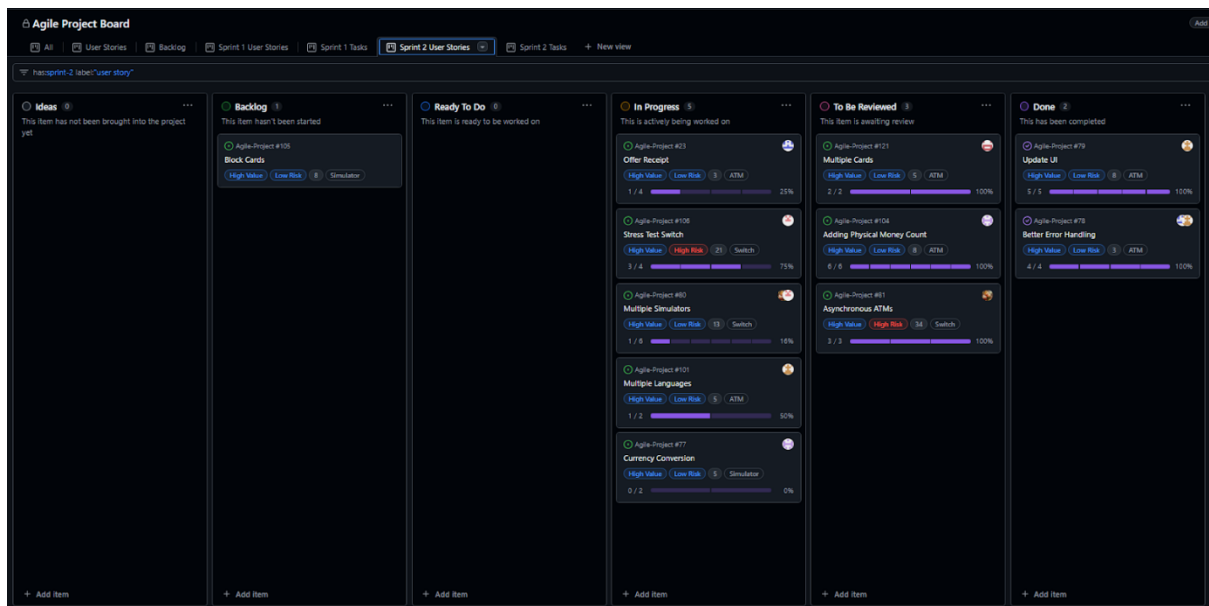
Done

This has been completed

+ Add item

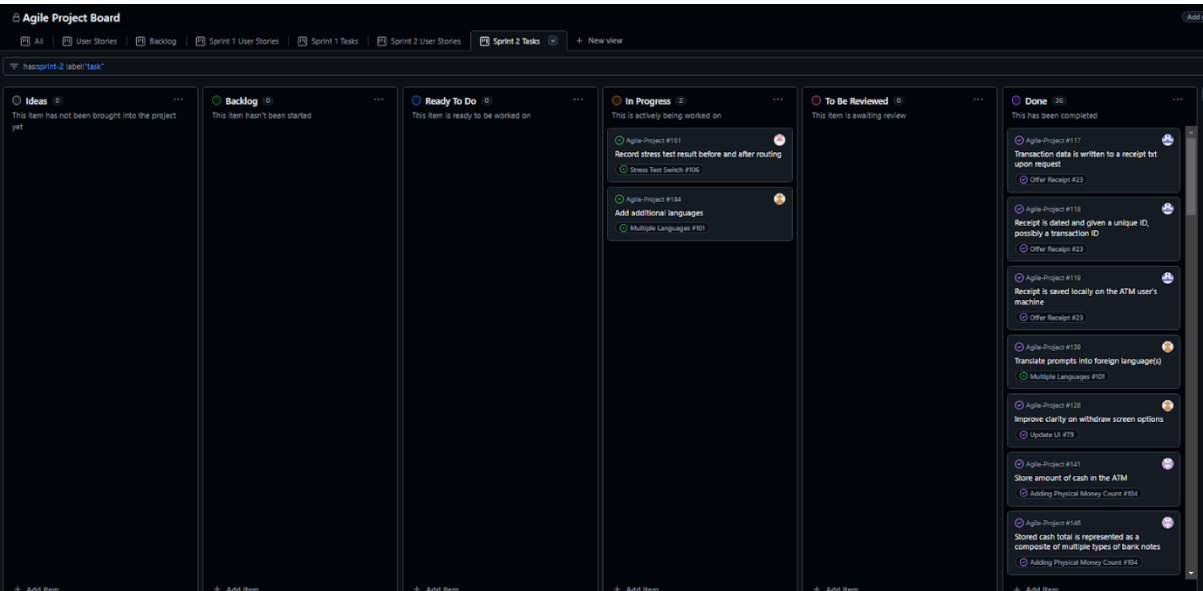
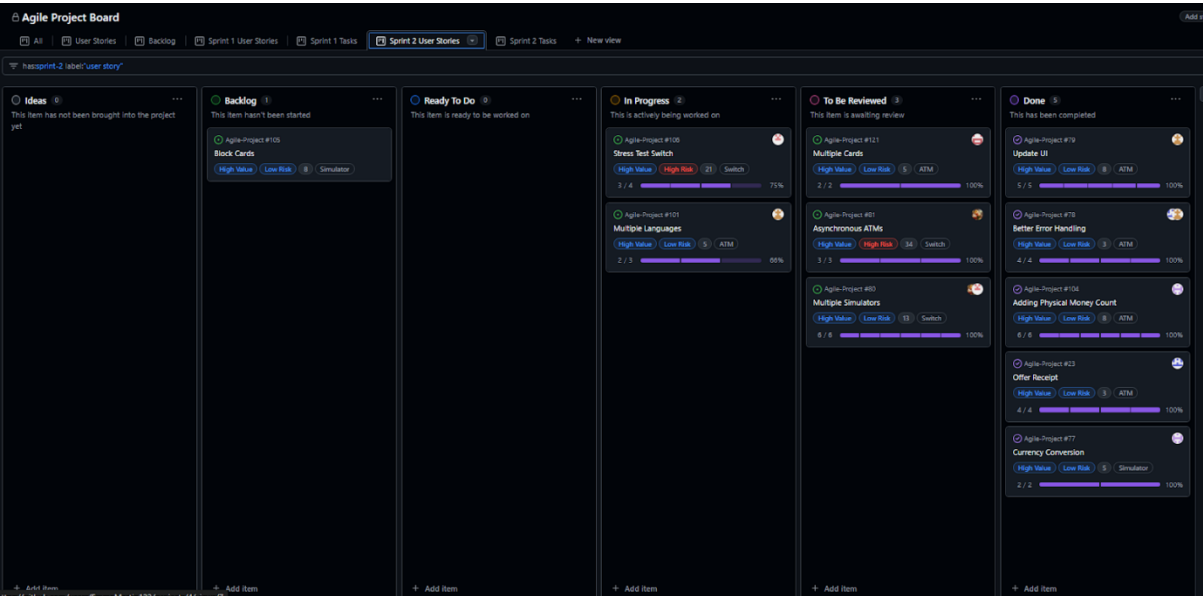
Group 4

Wednesday 29/01/25 Stand Up



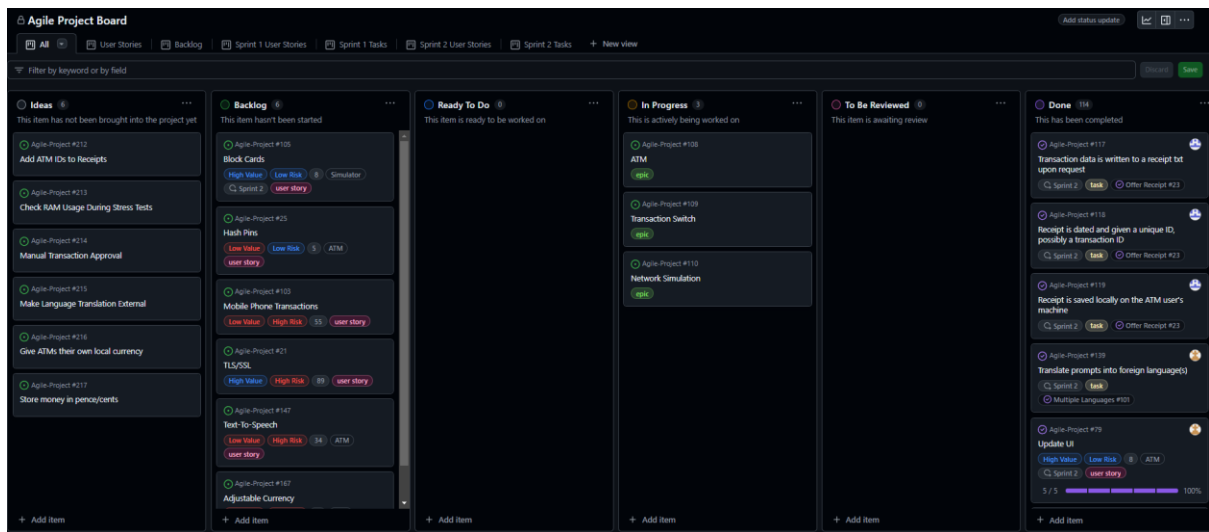
Group 4

Thursday 30/01/25 Stand Up



Group 4

Post Sprint 2



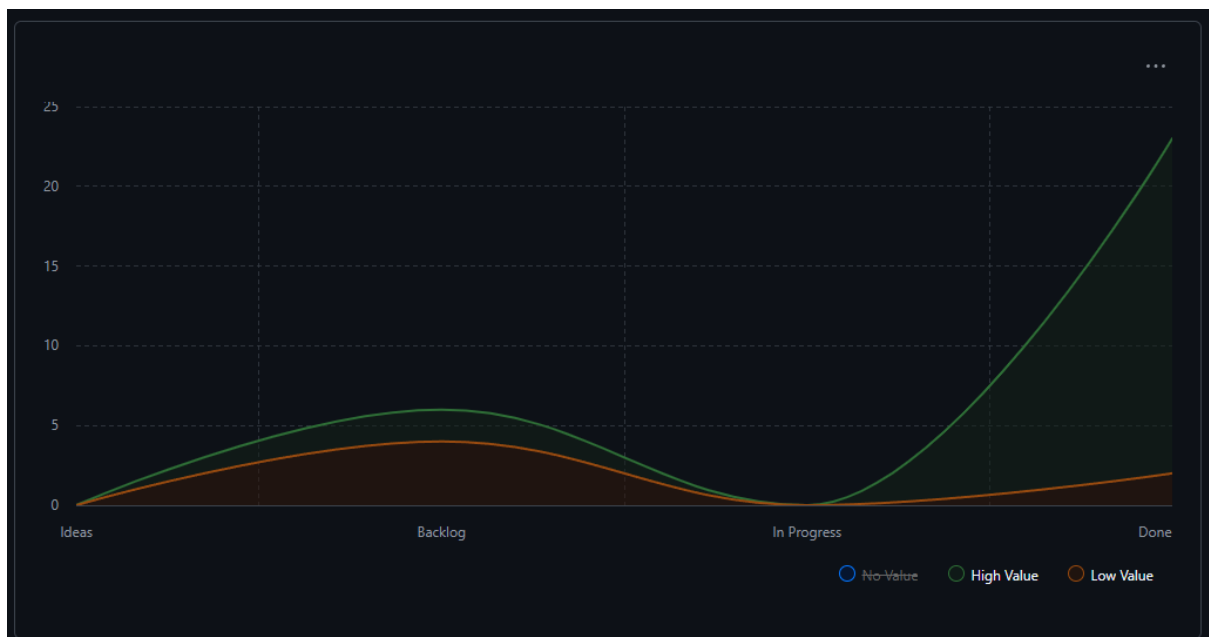
<https://github.com/users/EmmaMartin123/projects/4/views/1>

Group 4

GitHub Charts

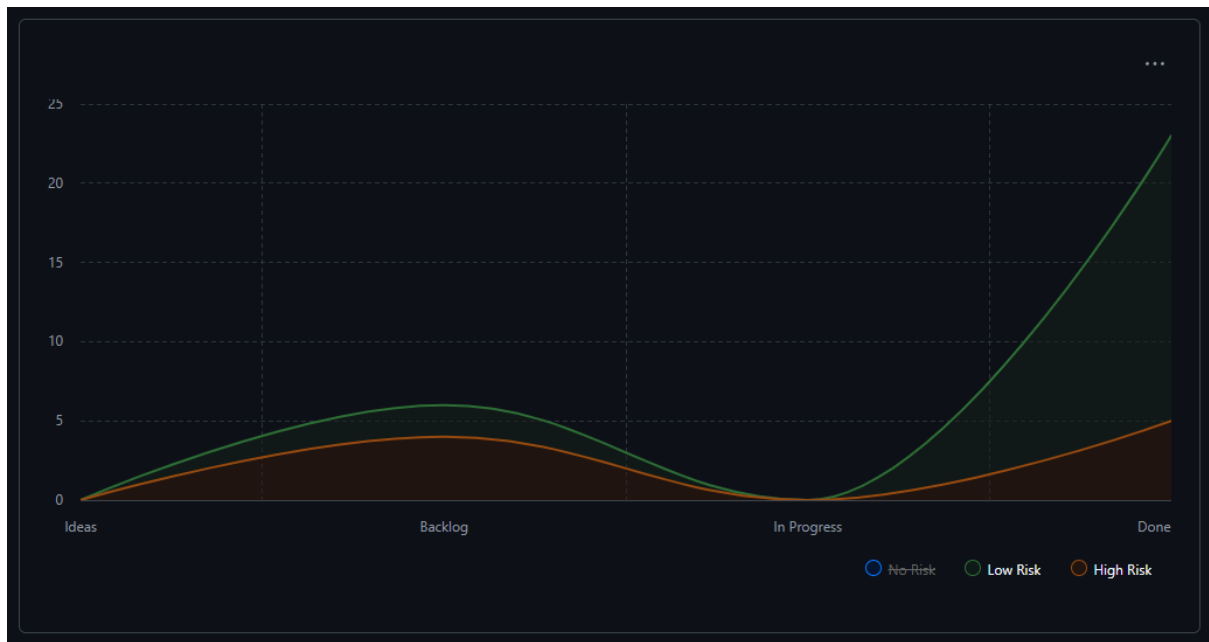


How many issues team members were assigned during Sprint1 and Sprint2



How many issues of High/Low Value are at each status

Group 4



How many issues of High/Low Risk are at each status