Software Product and Process Management. 2024-25.

Coursework 01: Stage 02

Minimum Viable Product and Sprint Plan.

Date: 11 Apr 2025.

Table Of Content.

GROUP MEMBERS.

No.	Full Name	Serial No.	B No.	Role.
1	Tanha Islam Sarani	10413607	B00968375	Scrum Master
2	Mst Saima Islam	10415449	B00974365	Frontend developer
3	Dharmendra Budhathoki	10412118	B00971067	Backend developer
4	Kumal Ravisha Tikiri Bandara	10414641	B00971989	Quality assurance
5	Sharafat Jamil	10419527	B00973753	DevOps/ Development manager
6	Ensar Ozdemir	10377101	B00975380	Business analyst

Introduction.

The **Community Volunteer Platform** is a prototype solution developed to bridge the gap between individuals seeking meaningful volunteer opportunities and the organizations that provide them. In a world where digital engagement plays a crucial role in community involvement, this platform enables seamless interaction among volunteers, non-profit organizations, community leaders, educational institutions, and donors.

This assignment focuses on delivering the **Minimum Viable Product (MVP)**, a streamlined version of the platform that includes the core functionalities required for early deployment and user validation. These include user registration, support request management, and a donation system with transparency and confirmation features.

Through a structured and agile approach, this report presents a detailed **story map**, sprint planning for each feature, individual development tasks, and user-focused test cases. The overall goal is to create a robust foundation for community-driven engagement, ensuring transparency, accessibility, and user satisfaction.

By focusing on real-world use cases and prioritizing essential features, this MVP lays the groundwork for future enhancements while addressing the immediate needs of stakeholders involved in community development.

Minimum Viable Product (MVP).

The **Minimum Viable Product (MVP)** of the Community Volunteer Platform is a **functional prototype** designed to address the core needs of connecting volunteers with community service opportunities. The MVP will include the **most essential features** that deliver immediate value to users and stakeholders while allowing room for improvements based on feedback.

Goals of the MVP:

- Enable **user registration and authentication**
- Allow users to **submit support requests** and **track responses**
- Facilitate **donations with confirmation and transparency**
- Provide basic admin functionality for managing submissions

Story Mapping.

Theme	Registration		Request Support		Donation	
feature	**User Sign-Up / Profile Creation**	**User Login / Authentication**	**Submit Support Request**	**Track Request Status**	**Make a Donation**	**View Donation History & Receipts**
Sprint 1	Design and build sign-up form (name, email, password)	Build login page, connect to backend auth logic	User interface design for request form (title, category, description)	Display submitted requests in a user dashboard	User interface for donation form (amount, currency, method)	Display donation history in user dashboard (list + totals)
	Set up a Database Schema for User profiles	After login save the password in the Email or the device		Create dashboard layout to list user's past requests		
Sprint 2	Input validation, account creation backend, email verification	Add secure password storage (hashing)	Backend for submitting & storing requests, input validation	Backend logic for updating and retrieving request status	Payment gateway integration (PayPal) with currency conversion	Add downloadable receipts (PDF/HTML) for each transaction

		Add "Remember Me", logout feature, and login audit logs		Backend API for fetching & updating request status by admin	Add test payment integration + error handling (invalid card, limits)	
Sprint 3	Profile edit support, profile image upload, user feedback page	Forgot password flow, email-based reset.	Add file attachments, success confirmation, and notification email	Implement email/push notifications for updates, status filters (e.g. pending/resolved)	Confirmation page with receipt, automated thank-you email	Filters for search (date, amount), export full donation report
	Create an interface for "Delete Account Permanently"		Allow draft saving, show submission preview, add success modal	Implement filters (status/date), color-coded status tags	Real payment processing, donation summary page with share options	Add charts (e.g., donation trends), export to Excel

Development Tasks.

Task ID	T1
Story ID	S04
Task Title	Implement Secure Login System for Users
Task Size	Medium
Dependencies	User accounts must already exist (from S03 or DB seeding)
Subtasks	* Create a login form on the frontend with fields for email and password, including appropriate labels and placeholders * Add client-side form validation to ensure no empty inputs and proper email format using RegEx * Set up a backend /api/login endpoint that receives credentials and returns auth status * Use bcrypt to validate entered password against hashed password in the database * Generate a secure JWT token on successful authentication with appropriate expiration time * Set HttpOnly and Secure flags for the token when sending it to the client as a cookie * Create middleware on the backend to protect routes and check for valid tokens before granting access * Design and implement a "Forgot Password" screen with email input and send a reset link using a token * Log each login attempt (success/fail), and flag suspicious behavior after multiple failed attempts from the same IP

Task ID	T2
Story ID	S08
Task Title	Develop Admin Interface for Support Requests
Task Size	Large
Dependencies	T1
Subtasks	* Implement a user-facing support request form with fields for subject, category, and detailed message * Validate the form inputs on both frontend and backend before saving to DB * Create a new SupportRequests table in the

	database with user ID, message, status, priority, and timestamps * Develop an admin-only dashboard view to list and filter support tickets by status (e.g. open, pending, resolved) * Enable status updates, assignment of tickets to admins/support agents, and internal comments on tickets * Implement real-time or email/in-app notifications when a ticket is updated or responded to * Enable threaded conversations between user and support team inside the ticket view * Display admin metrics: total open tickets, average resolution time, etc. * Apply role-based access control (RBAC) to restrict this dashboard to admins and support agents only * Write unit/integration tests to verify correct ticket behavior, access permissions, and filtering

Task ID	T3
Story ID	S10
Task Title	Build Transparent Donation Management System
Task Size	Large
Dependencies	None
Subtasks	* Design a normalized database schema including tables for: Donations, Expenses, Reports, and Donors * Create backend API endpoints to add, update, and retrieve donation and expense entries * Build an admin interface to enter and categorize expenditures linked to donation sources * Implement data visualizations (pie/bar charts) comparing used vs. available funds over time * Automate generation of donation reports (monthly/quarterly) showing income, expense, and allocation * Provide functionality to export donation

	reports as downloadable PDF files * Implement email delivery of personalized reports to donors summarizing their contributions and impact * Add date range and donor/project filters to dynamically adjust report outputs * Maintain a secure, immutable audit log of all financial transactions and report changes * Integrate with donor history and payments for full traceability from incoming donation to expense allocation

Task ID	T4
Story ID	S03
Task Title	Create user account
Task Size	Medium
Dependencies	None
Subtasks	● Enter the donor's data into the database (Name, contact number, email address, etc.) ● Submit all required donor information to the system's portal. ● Generate unique id ● Send email along 3 topics ➞ Personal information check confirmation ➞ Request to create a password (required for logging in with the unique ID and password)

	➞ Link for 2FA setup • Send a welcome email after successful registration including terms and conditions.

Task ID	T5
Story ID	S11
Task Title	Smart Donation Through Digital receipt (QR Code)
Task Size	Medium
Dependencies	T4
Subtasks	• Add donor Details (Name ,contact number,email address, payment amount ,unique id of Donor) • Complete payment through correct gateway method (Google pay,paypal) • Generate QR Code with proper information through proper library (Javascript,Python,C#) • Creation of Digital Receipt (PDF format) • Ensure the Functionality of the QR Code

	• Send the code to donor email address

Task ID	T6
Story ID	S12
Task Title	Two Factor Authentication(2FA) for safe transaction
Task Size	Medium
Dependencies	T5
Subtasks	• Activate 2FA option to Donor device (Google Authenticator, Microsoft Authenticator). • Generate code (usually 6 digits and short lived code) and send it via donor’s email or sms. • Verify and implement the code from the donor within the time limit, as it is short-lived. • Confirm the amount and proceed with the transaction. • Complete payment.

Task ID	T7
Story ID	S06
Task Title	Volunteer Request Management and Escalation System
Task Size	Medium
Dependencies	None
Subtasks	• Build a monitoring tool that flags volunteer requests not addressed within 48 hours. • Implement auto-reminder functionality to notify the assigned organization via email or dashboard alert. • Allow volunteers and organizations to mark their requests as "Resolved" with a comment box. • Store resolution status and timestamps in a support ticket database. • Trigger escalation to admin if the request exceeds 72 hours without action.

Task ID	T8
Story ID	S07
Task Title	Smart Resource Request Form with Admin Workflow
Task Size	Medium
Dependencies	None
Subtasks	• Create an accessible form for project leaders to input project name, type of resources, and urgency. • Validate data and store each submission in the backend with a status set to "Pending." • Send automatic notification to the admin with a link to the request. • Build an admin dashboard to approve/reject the request, with notes. • Notify the project leader with the decision and estimated resource delivery timeline.

Task ID	T9
Story ID	S09
Task Title	Donation Logging & Transparency Dashboardk Donations in System
Task Size	Large
Dependencies	None
Subtasks	• Design a robust schema to log donations: donor ID, amount, currency, date, purpose. • Implement real-time transaction logging linked to payment gateways. • Create an admin dashboard for browsing/filtering donation records. • Enable export to PDF/CSV for financial reports. • Automate monthly summary reports sent to stakeholders to enhance transparency

Task ID	T10
Story ID	S01
Task Title	Record Volunteering Hours
Task Size	Medium

Dependencies	None
Subtasks	● Create form to log volunteer hours (date, activity, duration). ● Link hours to user profile. ● Verify hours with organization's approval. ● Update total hours in the user's dashboard. ● Send notification to user upon approval.

Task ID	T11
Story ID	S02
Task Title	Personalized Recommendations System
Task Size	Large
Dependencies	T10
Subtasks	● Develop user interest and skills input form. ● Implement recommendation algorithm using filters (location, interest, skill match). ● Show top 5 recommended opportunities on dashboard. ● Allow feedback on recommendations to improve accuracy. ● Send weekly email with updated suggestions.

Task ID	T12
Story ID	S05
Task Title	Request Help for an Opportunity
Task Size	Medium
Dependencies	None
Subtasks	• Design help request form (title, description, deadline). • Enable attachments for relevant documents or images. • Route request to support team/admin. • Enable user to track the status of request. • Send notification once response is available.

TEST CASES

User story 1

Test ID: TC1	**Test Description:** Make sure admin can review, approve, reject or edit volunteering submissions		
Story ID: S01	**Story Narrative:** As an educational institution, I want to be able to access and manage student volunteering records so that I can attach them to academic records for scholarship shortlisting.		
Step ID:	**Step Description**	**Test Data**	**Expected Result**
1	Admin logs in and access dashboard	Username: admin Password: admin123	Dashboard showing all volunteer submissions
2	Selects a particular submission	Good gym, 3 hours in total	All records of the particular submission is displayed.
3	Approves the submission	Record matches the given guidelines	Students points are updated, green tick is displayed.
4	Admin selects another submission	Charity shop, 20 minutes in total	All records of the particular submission is displayed.
5	Rejects the submission	Record fails to match with the institution guidelines	The submission is removed from the dashboard. Students points remains same.
6	Admin chooses to edit the record.	Modifies the total hours.	Updated record is displayed.

Test ID:TC2	**Test Description:** Ensure that the admin can filter, and view student's volunteering submissions based on organization's name, hours and location.		
Story ID:S01	**Story Narrative:** As an educational institution, I want to be able to access and manage student volunteering records so that I can attach them to academic records for scholarship shortlisting.		
Step ID:	**Step Description**	**Test Data**	**Expected Result**
1	Admin logs in to the system.	Username: admin Password: admin123	Dashboard is displayed.
2	Navigates to the volunteer submission section.	-	List of student submission is displayed
3	Selects a filter with organization's name	"Charity Shop"	Submission only from the Charity Shop organization is shown.
4	Selects a range of hours from 0 to 30 minutes.	Hours filter input: 0 to 30	Only records with volunteering hours upto 30 minutes is shown

USER STORY 2:

Test ID: TC3	Test Description: Ensure that an user can create an account successfully.		
Story ID: S03	**Story Narrative:** As a new user, I would like to register an account so I can access the platform, manage my profile, and take advantage of the features without limitations. The registration process should be smooth and secure, allowing me to easily verify my identity.		
Step ID:	**Step Description**	**Test Data**	**Expected Result**
1	Click on sign in	-	Two options. Register as a new user or Log in.
2	User selects register as a new user.	Name: Email: Password:	All fields expect input.
3	Click “Register”	-	Form is submitted. “Registration link has been sent” is displayed.
4	Opens email and click on verify.	-	Redirects to the dashboard. “Account is successfully created” message displayed.

Test ID: TC4	**Test Description:** Ensure the user and successfully amange and update their profile details.		
Story ID: S03	**Story Narrative:** As a new user, I would like to register an account so I can access the platform, manage my profile, and take advantage of the features without limitations. The registration process should be smooth and secure, allowing me to easily verify my identity.		
Step ID:	**Step Description**	**Test Data**	**Expected Result**
1	User clicks on sign in	Email: Password:	Successfully logged in message is displayed.
2	Clicks on account settings.	-	Different fields are shown.
3	User heads over to “edit profile”	-	Editable fields are displayed.
4	Update full name.	From “Tris” to “Tris Smith”	Field is updated. “Successfully updated” message is shown
5	User clicks on “update password”.	From “” to “Newpassword23”	Field is updated. “Successfully updated” message is shown
6	Click “Save”	-	“Profile updated successfully” message is displayed

User story 3:

Test ID: TC5	**Test Description:** Ensure a volunteer can submit an inquiry to a non-profit organization		
Story ID: S06	**Story Narrative:** As a non-profit organization, I want to receive and respond to volunteer inquiries, so that I can provide clear guidance and improve engagement.		
Step ID:	**Step Description**	**Test Data**	**Expected Result**

1	Volunteer navigates to a particular organization page	"Good Gym"	A side-option showing "Send an enquiry"
2	Clicks on the "Send an enquiry" button.	-	Inquiry form opens
3	Fill in name and email field	Name: John Email: john23@yahoo.com	Details are accepted, shows a green tick beside the fields.
4	Volunteer writes their message	"I want to know more about volunteering options"	Text is accepted.
5	Click on "Submit" option	-	"Your enquiry has been sent, please check your email for confirmation" text is displayed
6	Volunteer check email.	-	"Your enquiry is received, please download your digital receit" email is displayed

Test ID: TC6	**Test Description:** Non-profit organization responds to an inquiry		
Story ID: S06	**Story Narrative:** As a non-profit organization, I want to receive and respond to volunteer inquiries, so that I can provide clear guidance and improve engagement.		
Step ID:	**Step Description**	**Test Data**	**Expected Result**
1	Organization admin signs in	Username: admin Password: admin123	Dashboard loads
2	Heads over to "inquiries" section	-	All inquiries submitted is shown
3	Selects a particular inquiry.	Inquiry from John	All the inquiry details is shown. Inquiry message as well as name and email
4	Types a reply.	"Thanks for you interest! Here are the details ….."	Text is accepted, and "Send" button turns green.
5	Clicks "Send".	-	"Reply sent successfully!" is displayed
6	Volunteer receives email regarding the reply to his inquiry	-	Volunteer can see the reply as well as login option to his portal is suggested.

User Story 4:

Test ID: TC7	**Test Description:** Ensure that a leader in community can submit a valid resource request for a project		
Story ID: S08	**Story Narrative:** As a leader in the community, I would like to seek more volunteers or resources for a project to guarantee its successful implementation.		
Step ID:	**Step Description**	**Test Data**	**Expected Result**
1	Community leader logs in	Username:leader123 Password: leaderdoe	Dashboard is displayed showing all available options
2	Navigates to a particular project page.	"Clean water" project	All the details regarding the project is displayed.
3	Clicks on "Request resources"	-	A text-box appears

4	Enters resource details	Item “Pipes. Quantity: 3”	Form accepts input and “Send request” button turns green
5	Clicks “Send request”	-	“Resource successfully sent” message appears.
6	System updates project details	-	Additional requests are displayed along the project.

Test ID: TC8	**Test Description:** Ensure that when the administrator approves a resource request, the community leader receives notification.		
Story ID: S08	**Story Narrative:** As a leader in the community, I would like to seek more volunteers or resources for a project to guarantee its successful implementation.		
Step ID:	**Step Description**	**Test Data**	**Expected Result**
1	Admin logs in the platform	Username: Password:	Dashboard is shown.
2	Navigates to the “pending requests” section	-	List of projects is shown
3	Selects a specific request to review	Request ID: M890	Request details is shown, as well as time when request is received and
4	Admin reviews the request.	Checks the review and checks their available resources.	Two option is displayed “Approve” or “Disapprove”
5	Clicks “Approve” button.	-	Request status changes to “Approve” and turns green.
6	An email is sent to the community leader.	-	Digital receipt is available in the email inbox.

User Story 5:

Test ID: TC9	**Test Description:** Ensure that the system displays real-time conversiona rates and donation is broken down for easier understanding for the donor.		
Story ID: S12	**Story Narrative:** As a donor, I seek a seamless multi-currency payment system that transcends borders, allowing me to contribute effortlessly from my local currency to the charity’s preferred currency. I desire a hassle-free transaction process with real-time conversion rates, ensuring complete transparency and a clear understanding of my donation's exact value.		
Step ID:	**Step Description**	**Test Data**	**Expected Result**
1	User navigates to the donation page.	-	Donation page loads, showing all the available options to donate money and an “amount” box
2	User type a donation amount.	30 pound	Input accepted.
3	Selects country of origin.	Country: Canada	Country of origin changes and amount.
4	System displays real-time conversion	30 Pound is converted to 54 Canadian dollar.	Details showing the converted amount. “Donate” button is turned green.
5	User clicks on “Donate” button	-	Page reloads, then displays “Donation is sent successfully”.

6	User receives a confirmation in their email.	“You donated 54 Canadian dollar.”	User views his digital receipt.

Test ID: TC10	**Test Description:** Ensure that the donor can select from multiple payment methods to complete a donation		
Story ID: S12	**Story Narrative:** As a donor, I seek a seamless multi-currency payment system that transcends borders, allowing me to contribute effortlessly from my local currency to the charity’s preferred currency. I desire a hassle-free transaction process with real-time conversion rates, ensuring complete transparency and a clear understanding of my donation's exact value.		
Step ID:	**Step Description**	**Test Data**	**Expected Result**
1	User navigates to the donation payment page.	-	Different payment options are shown to choose from.
2	User choose particular amount and currency	Amount: 50 BDT	Amount is accepted. A green tick is shown right next to the text field.
3	User clicks on “View payment options”	-	PayPal, Google Pay, Visa, Klarna, Clearpay displayed
4	User clicks on “Visa”	-	“You have selected Visa payment” message is displayed
5	Authenticate and confirm payment	Valid card number, expiry date and address details.	Transaction is successful.
6	Email receipt is sent to the users email.	Email:johnsmith@yahoo.com	Receipt includes currency conversion, transaction ID and payment method that was selected

Conclusion

Overall, we all had a great experience developing our project and implementing essential Agile and Scrum methodologies to structure our workflow. From building the Product Backlog to breaking down each task during Sprint Planning, story mapping, and writing test cases, our team collaborated effectively throughout the process. Regular Sprint Reviews allowed us to gather valuable feedback, helping us continuously improve and stay focused on our goals.

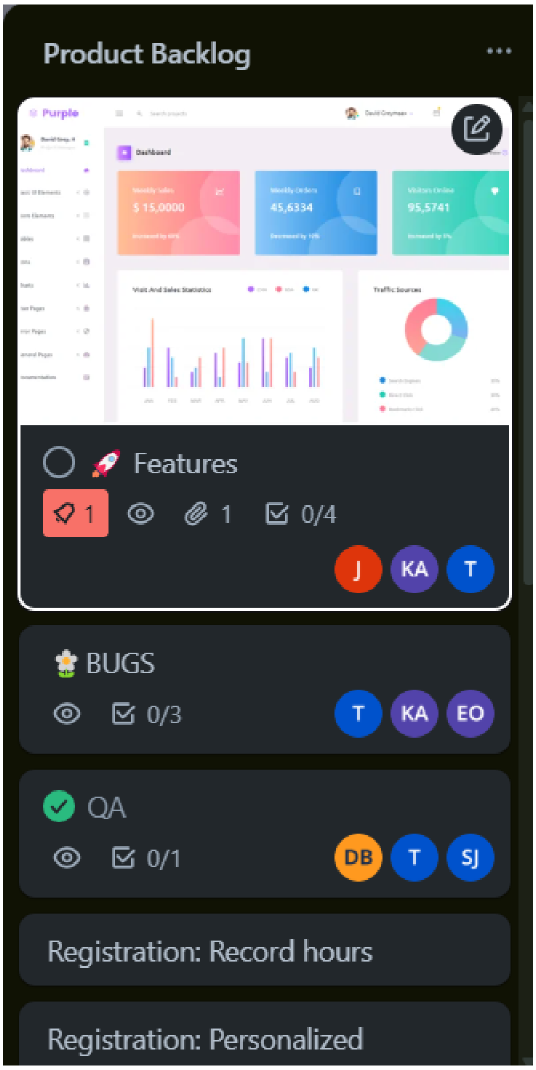
Each sprint gave us a clearer picture of what was working and what needed to change. By identifying priorities, adjusting our approach when necessary, and committing to iterative progress, we were able to produce results more efficiently. The structured yet flexible nature of Agile helped our team stay organized, communicate better, and deliver a solution that meets the project's needs.

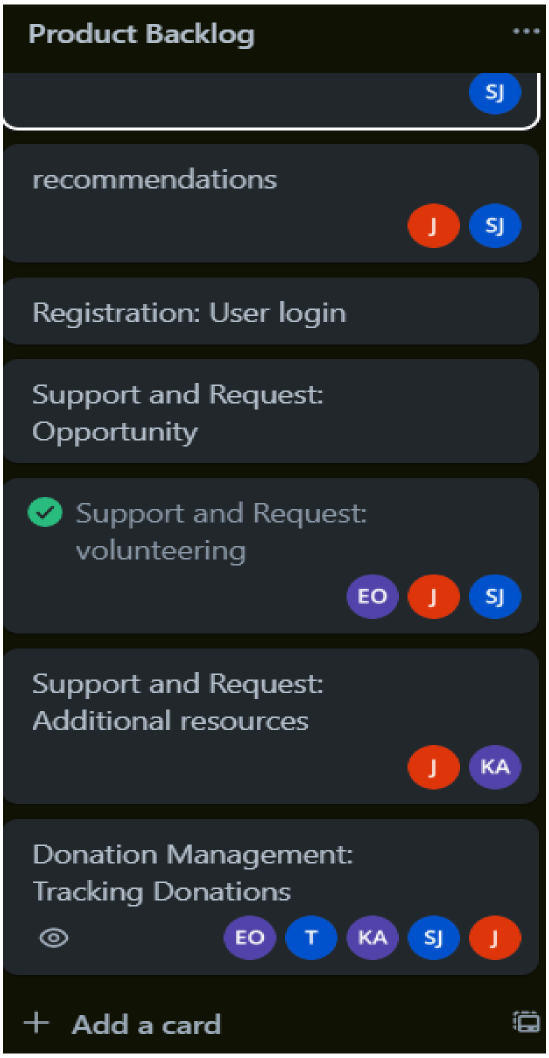
In the end, not only did we complete the technical work, but we also grew as a team—learning how to manage time, solve problems collaboratively, and adapt to challenges in a real-world development environment.

Appendix.

PRODUCT BACKLOG

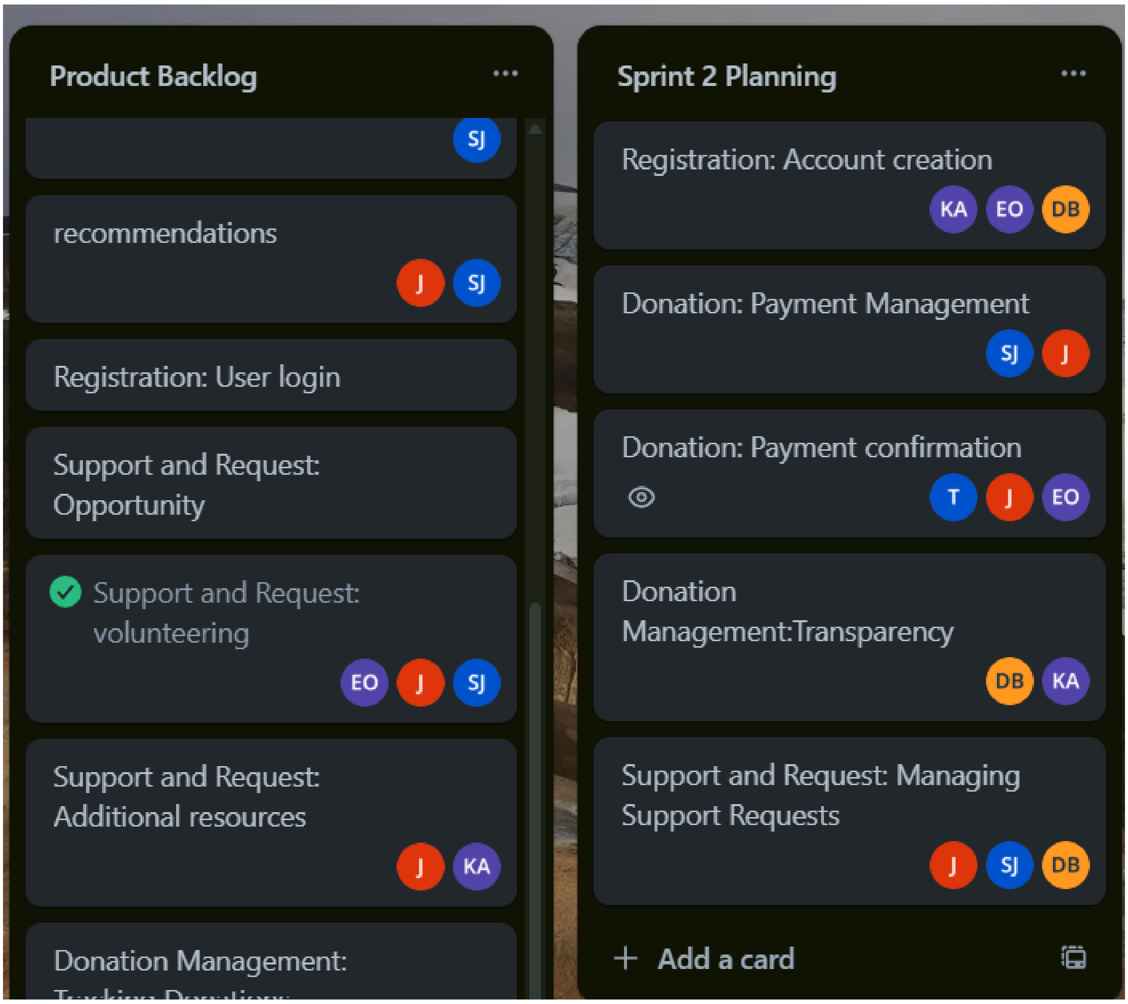
Product Backlog is the central point in Agile and Scrum development. It includes all the tasks required for the project, organized according to priority, and fulfils all the requirements needed to deliver a complete software product. It also includes any necessary bug fixes and enhancements. The Product Owner is responsible for managing and prioritizing the Product Backlog.





SPRINT PLANNING

It is the beginning of a new sprint, where the team decides what they will work on and how they will accomplish it. During the meeting, the Product Owner selects items from the product backlog, and the development team discusses how to implement them. By collaborating, the team selects user stories that can realistically be completed during the sprint and defines a clear sprint goal. These stories are then broken down into smaller, manageable tasks. The outcome of this planning session is the Sprint Backlog — a list of tasks that need to be completed during the sprint. The MVP is discussed to prioritize the core features that are essential for the first release, such as user registration, payment processing, and receipt generation.

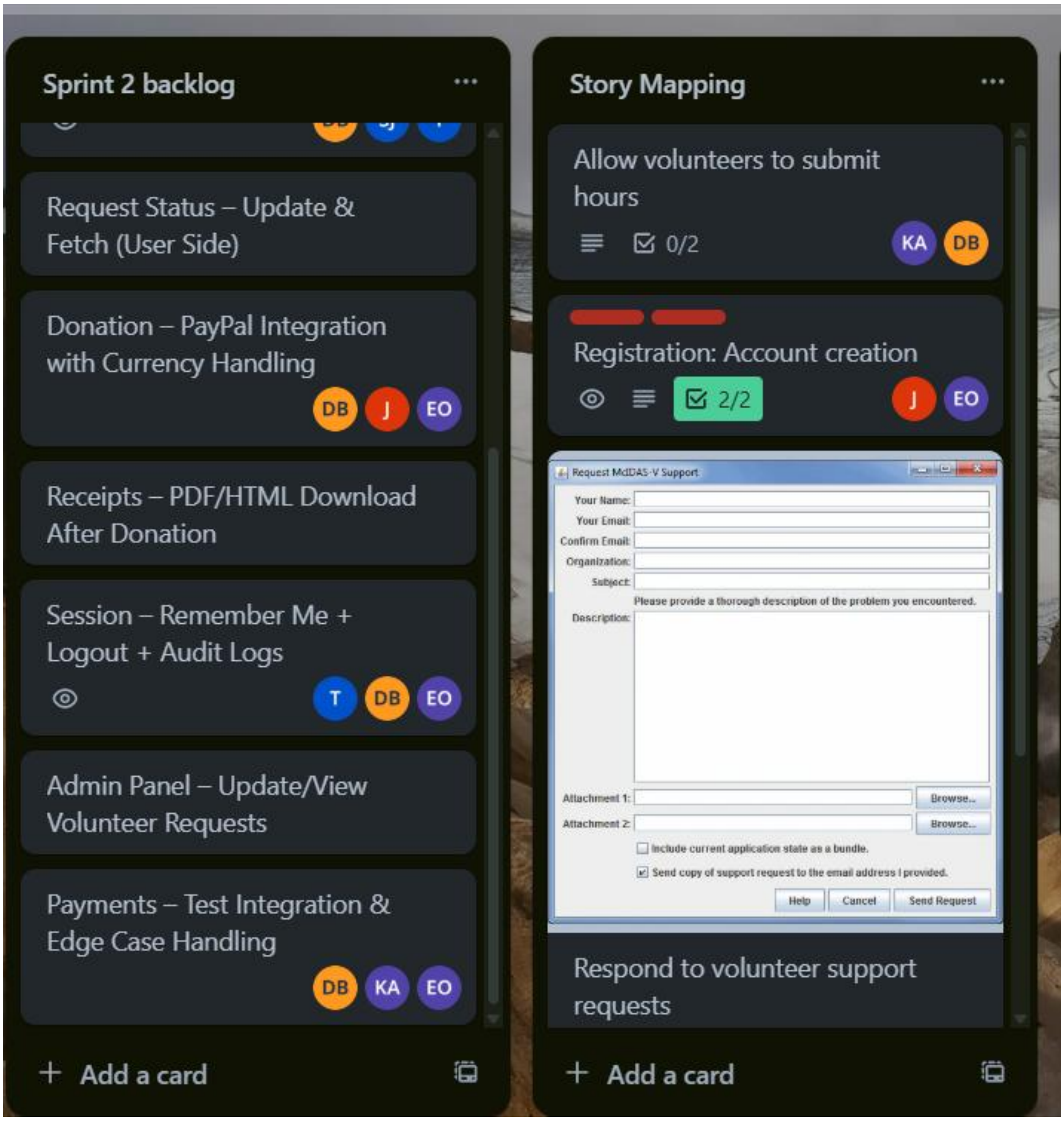


SPRINT BACKLOG

The Sprint Backlog contains a list of selected tasks and user stories that the development team commits to completing during the given sprint. It is created during the Sprint Planning meeting by the team and includes all the work needed to achieve the sprint goal. The difference between the Sprint Backlog and the Product Backlog is that the Product Backlog contains everything that may be needed for the entire project, while the Sprint Backlog focuses only on what needs to be completed within a specific sprint. This focused list helps the team break down work into smaller tasks, estimate time effectively, and stay aligned and productive throughout the sprint.

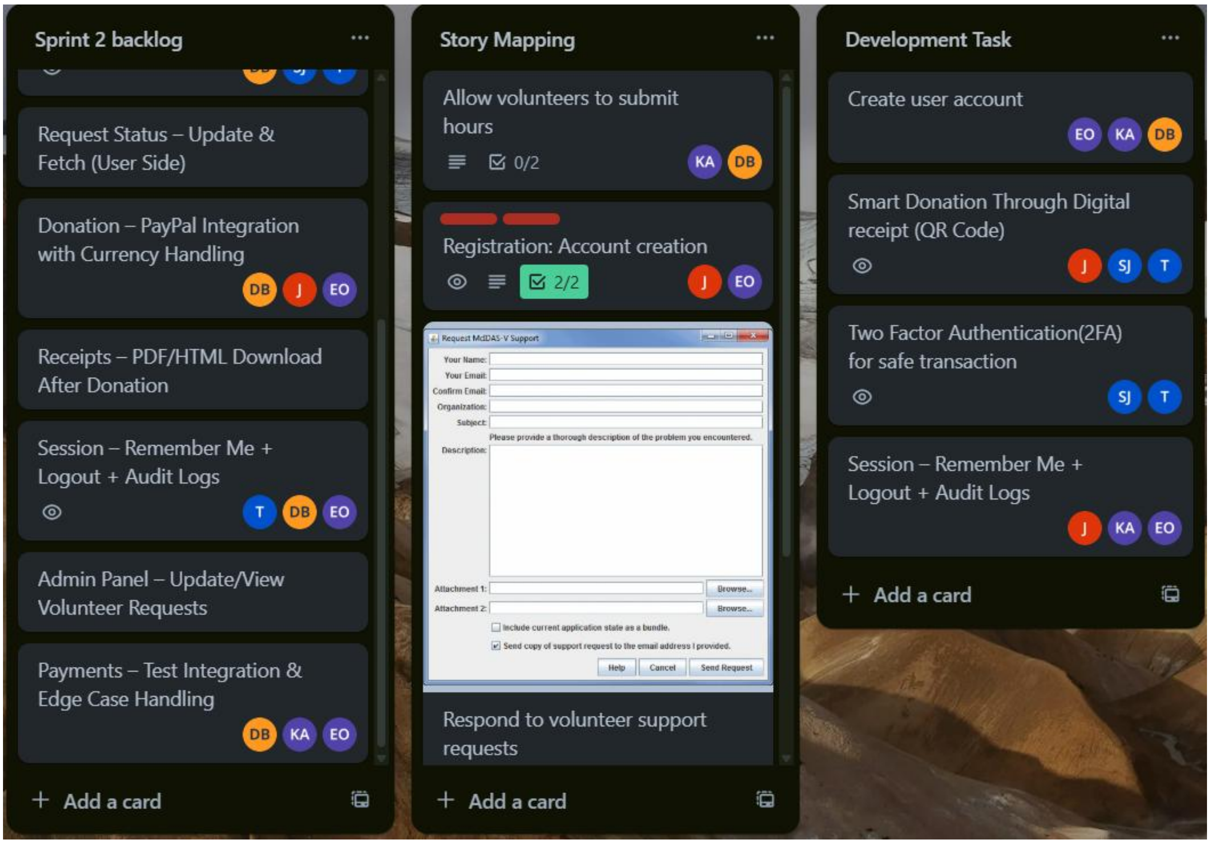
STORY MAPPING

In Agile development, story mapping is a technique used to organize user stories based on the requirements of the project. It helps the team analyze what the user needs and identify which features are required to support those needs. In a typical story map, high-level user activities and core features are placed at the top and given the highest priority, while related user stories and supporting tasks are listed underneath. This structure makes it easier to identify the Minimum Viable Product (MVP), which the team can then focus on delivering first.



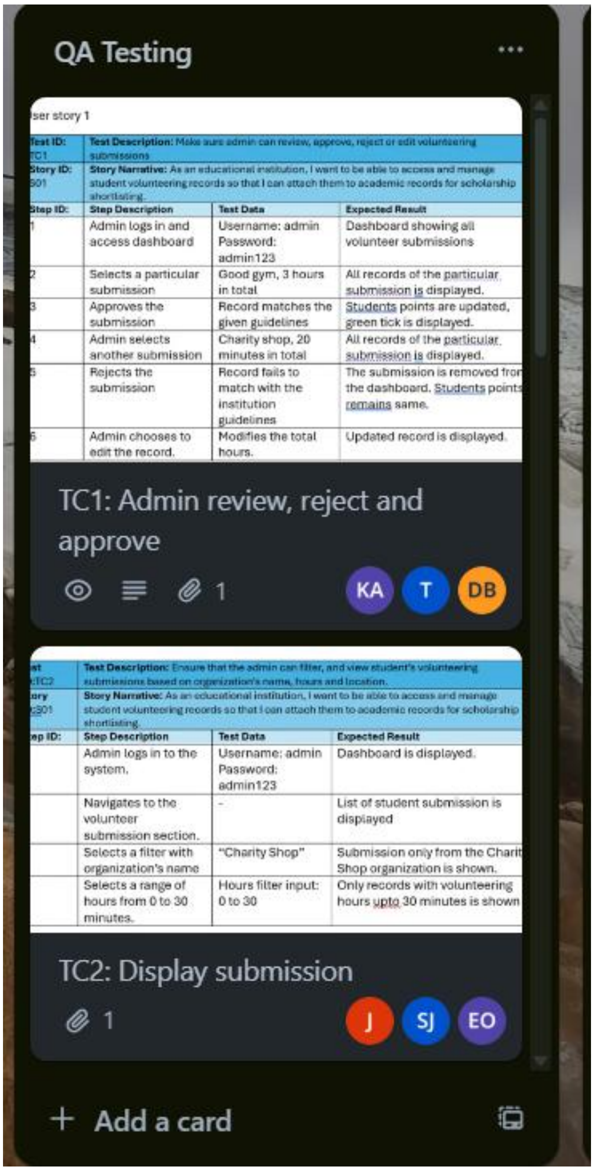
DEVELOPMENT TASK

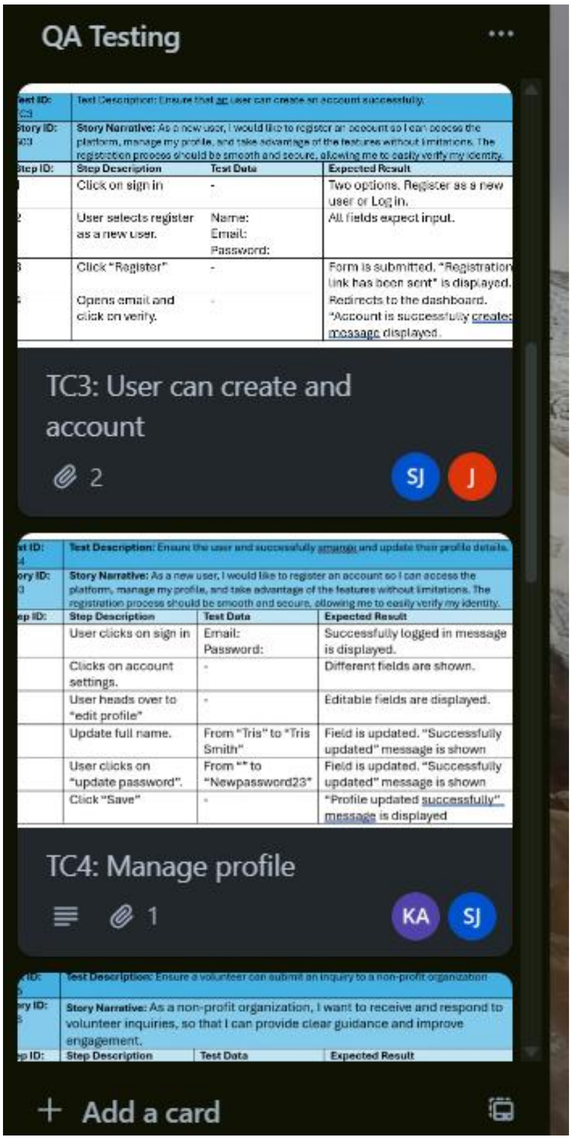
It mainly focuses on technical work such as coding, debugging, optimizing, and testing. The larger tasks from the sprint backlog are broken down into smaller, more granular actions, typically with a shorter time span. This method helps the team track progress quickly, often through the use of checklists, ensuring that each part of a feature or fix is completed systematically.

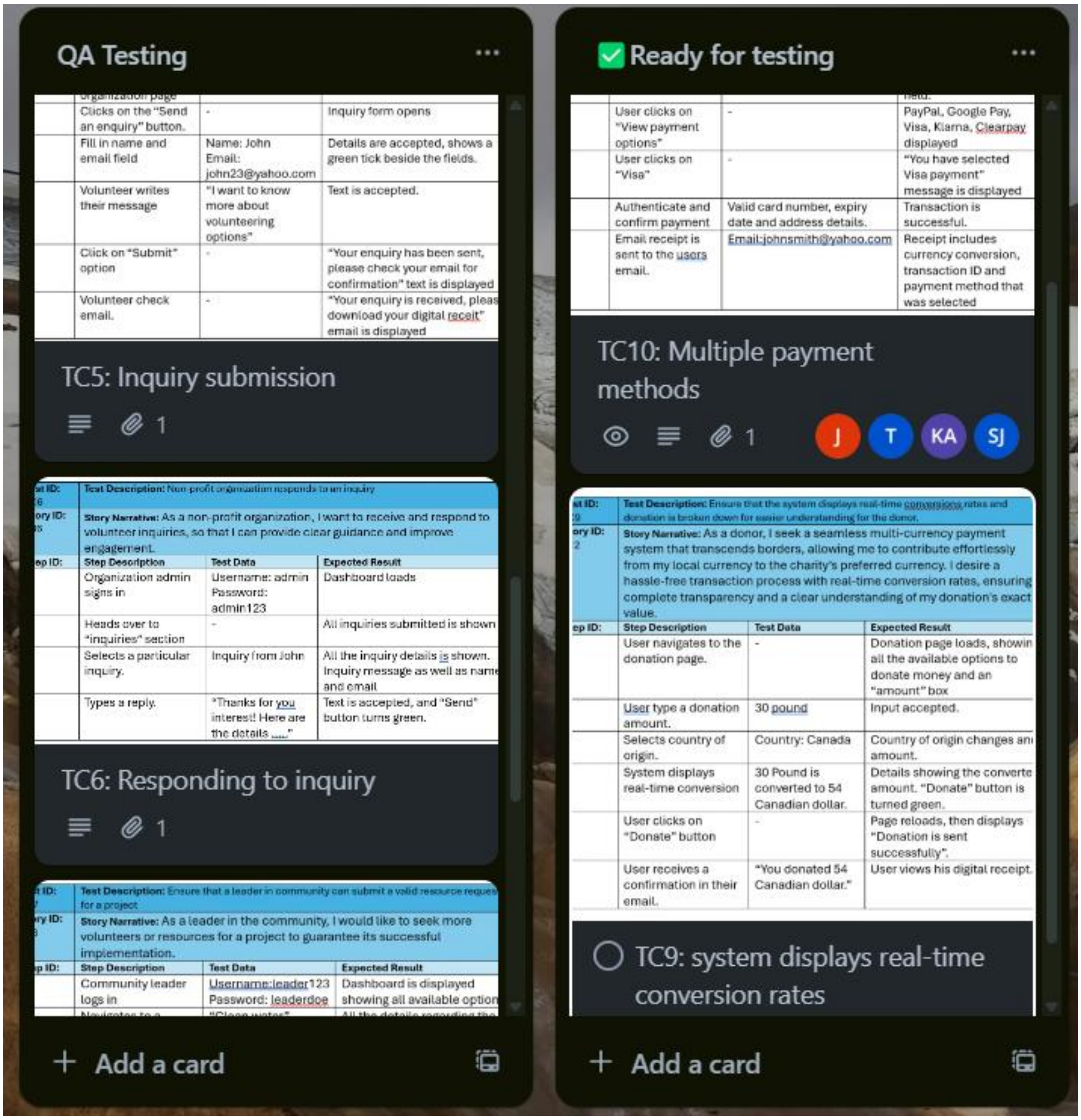


TEST CASES

Test cases are specifically developed to verify that certain features of a software application are behaving as expected. They typically contain inputs, expected results, and the final result. In Agile and software development, test cases are important for validating that the features and functionalities meet the required specifications outlined by the project stakeholders.


QA Testing
ser story 1
Test ID: TC1
Test Description: Make sure admin can review, approve, reject or edit volunteering submissions
Story ID: S01
Story Narrative: As an educational institution, I want to be able to access and manage student volunteering records so that I can attach them to academic records for scholarship shortlisting.
Step ID: | Step Description | Test Data | Expected Result
1 | Admin logs in and access dashboard | Username: admin Password: admin123 | Dashboard showing all volunteer submissions
2 | Selects a particular submission | Good gym, 3 hours in total | All records of the particular submission is displayed.
3 | Approves the submission | Record matches the given guidelines | Students points are updated, green tick is displayed.
4 | Admin selects another submission | Charity shop, 20 minutes in total | All records of the particular submission is displayed.
5 | Rejects the submission | Record fails to match with the institution guidelines | The submission is removed fro the dashboard. Students points remains same.
6 | Admin chooses to edit the record. | Modifies the total hours. | Updated record is displayed.
TC1: Admin review, reject and approve
1
KA
T
DB
Test Description: Ensure that the admin can filter, and view student's volunteering submissions based on organization's name, hours and location.
Story Narrative: As an educational institution, I want to be able to access and manage student volunteering records so that I can attach them to academic records for scholarship shortlisting.
Step Description | Test Data | Expected Result
Admin logs in to the system. | Username: admin Password: admin123 | Dashboard is displayed.
Navigates to the volunteer submission section. | - | List of student submission is displayed
Selects a filter with organization's name | "Charity Shop" | Submission only from the Charit Shop organization is shown.
Selects a range of hours from 0 to 30 minutes. | Hours filter input: 0 to 30 | Only records with volunteering hours upto 30 minutes is shown
TC2: Display submission
1
J
SJ
EO
Add a card

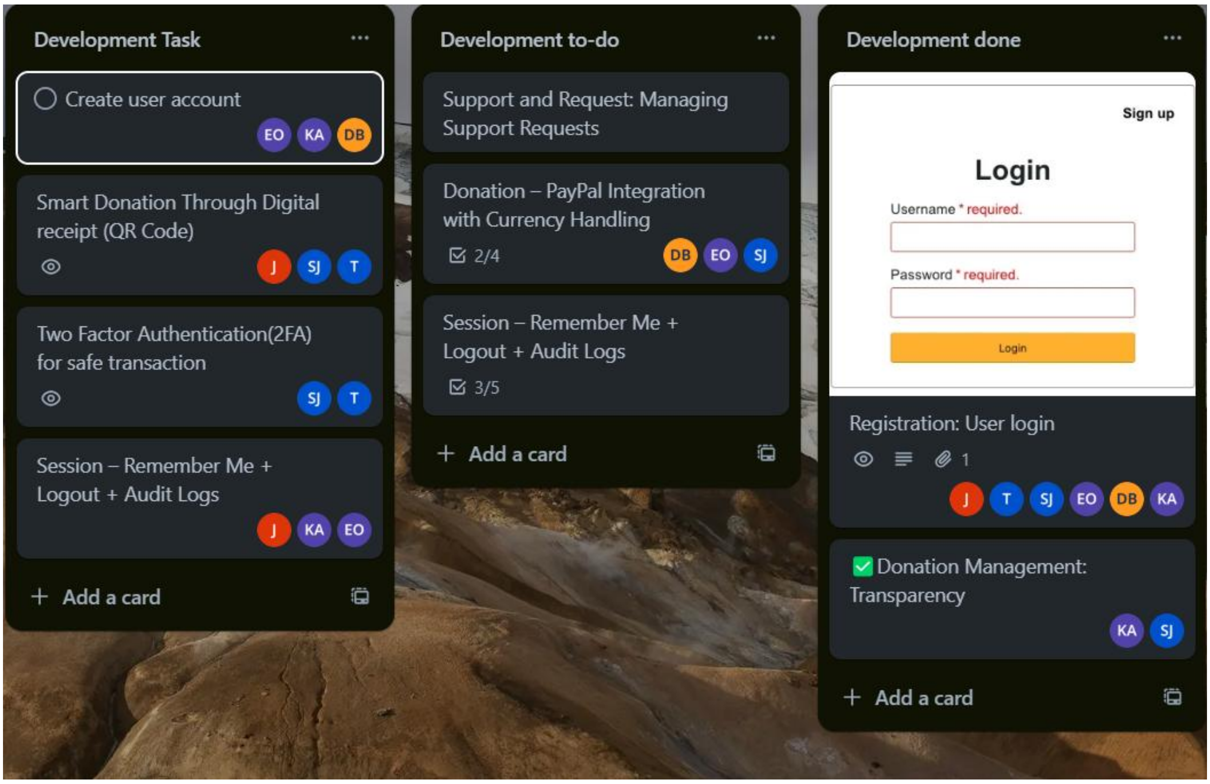

QA Testing
Test Description: Ensure that an user can create an account successfully.
Story Narrative: As a new user, I would like to register an account so I can access the platform, manage my profile, and take advantage of the features without limitations. The registration process should be smooth and secure, allowing me to easily verify my identity.
Step ID: | Step Description | Test Data | Expected Result
1 | Click on sign in | - | Two options. Register as a new user or Log in.
2 | User selects register as a new user. | Name: Email: Password: | All fields expect input.
3 | Click "Register" | - | Form is submitted. "Registration link has been sent" is displayed.
4 | Opens email and click on verify. | - | Redirects to the dashboard. "Account is successfully created message displayed.
TC3: User can create and account
2
SJ
J
Test Description: Ensure the user and successfully amange and update their profile details.
Story Narrative: As a new user, I would like to register an account so I can access the platform, manage my profile, and take advantage of the features without limitations. The registration process should be smooth and secure, allowing me to easily verify my identity.
Step Description | Test Data | Expected Result
User clicks on sign in | Email: Password: | Successfully logged in message is displayed.
Clicks on account settings. | - | Different fields are shown.
User heads over to "edit profile" | - | Editable fields are displayed.
Update full name. | From "Tris" to "Tris Smith" | Field is updated. "Successfully updated" message is shown
User clicks on "update password". | From "" to "Newpassword23" | Field is updated. "Successfully updated" message is shown
Click "Save" | - | "Profile updated successfully" message is displayed
TC4: Manage profile
1
KA
SJ
Test Description: Ensure a volunteer can submit an inquiry to a non-profit organization
Story Narrative: As a non-profit organization, I want to receive and respond to volunteer inquiries, so that I can provide clear guidance and improve engagement.
Step Description | Test Data | Expected Result
Add a card



In our project, we used the test case template provided to us. It contains the test ID, story narrative, and test description. Each test is broken down into several steps, which include a step description, expected result, and input.

DEVELOPMENT PROGRSS

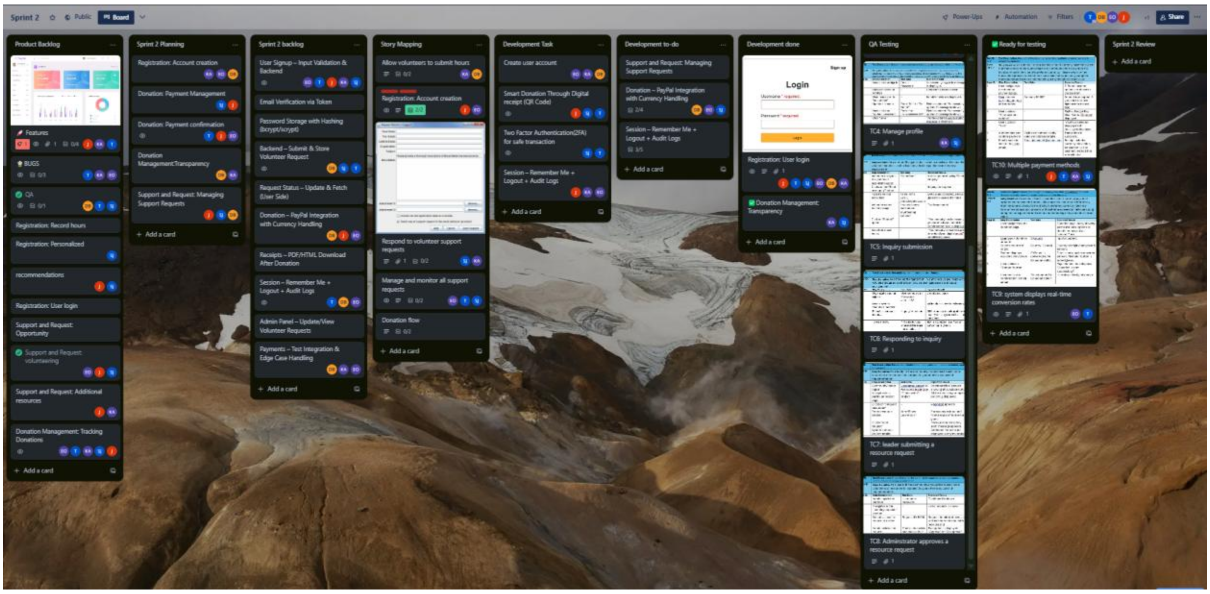
In this step, the developer continues working on completing tasks. They use a template to visualize what's done, what needs to be done, and how much work has been completed. This helps the team track how much time is spent on each task and analyze whether they will be able to meet deadlines.



SPRINT REVIEW

At the end of each sprint, we conducted a **Sprint Review** to gather feedback from all involved stakeholders, including team members and the Product Owner. This review was essential for ensuring that the work completed during the sprint aligned with the project goals and met expectations. During the review, the development team demonstrated the features or improvements they had completed, while stakeholders had the opportunity to provide valuable insights and feedback.

This process not only helped us identify areas for improvement but also provided an opportunity for the team to highlight challenges they faced, which could then be addressed in future sprints. It was an important step in refining our work process and ensuring that we stayed on track.



Sprint 2
Public
Board
Power-Ups
Automation
Filters
Share
Product Backlog
Features
BUGS
QA
Registration: Record hours
Registration: Personalized
recommendations
Registration: User login
Support and Request: Opportunity
Support and Request: volunteering
Support and Request: Additional resources
Donation Management: Tracking Donations
Add a card
Sprint 2 Planning
Registration: Account creation
Donation: Payment Management
Donation: Payment confirmation
Donation ManagementTransparency
Support and Request: Managing Support Requests
Add a card
Sprint 2 backlog
User Signup – Input Validation & Backend
Email Verification via Token
Password Storage with Hashing (bcrypt/scrypt)
Backend – Submit & Store Volunteer Request
Request Status – Update & Fetch (User Side)
Donation – PayPal Integration with Currency Handling
Receipts – PDF/HTML Download After Donation
Session – Remember Me + Logout + Audit Logs
Admin Panel – Update/View Volunteer Requests
Payments – Test Integration & Edge Case Handling
Add a card
Story Mapping
Allow volunteers to submit hours
Registration: Account creation
Respond to volunteer support requests
Manage and monitor all support requests
Donation flow
Add a card
Development Task
Create user account
Smart Donation Through Digital receipt (QR Code)
Two Factor Authentication(2FA) for safe transaction
Session – Remember Me + Logout + Audit Logs
Add a card
Development to-do
Support and Request: Managing Support Requests
Donation – PayPal Integration with Currency Handling
Session – Remember Me + Logout + Audit Logs
Add a card
Development done
Sign up
Login
Registration: User login
Donation Management: Transparency
Add a card
QA Testing
TC4: Manage profile
TC5: Inquiry submission
TC6: Responding to inquiry
TC7: leader submitting a resource request
TC8: Administrator approves a resource request
Add a card
Ready for testing
TC10: Multiple payment methods
TC9: system displays real-time conversion rates
Add a card
Sprint 2 Review
Add a card