Release Planning Document

COSC 4P02 Software Engineering II

Naser Ezzati-Jivan

January 18, 2025

Brock University

TA: Sarfaroz Yunusov

Project Title: PRJ 4 – Event Management Platform (Festify)

Name	Student Number	Email
Gautam .	7340433	gg21qv@brocku.ca
Harmanjot Malhi	7300973	hm21jv@brocku.ca
Kartikkumar Parekh	7107782	kp20zm@brocku.ca
Parampal Singh	7003114	pp20kv@brocku.ca
Rohal Kabir	7105836	rk20da@brocku.ca
Sumant Patel	6796841	sp19kp@brocku.ca
Vivek Salwan	6951826	vs19bf@brocku.ca

Table of Contents

Introduction	3
User Stories	3-5
Product Backlog	5-8
Sprint Backlog	8
Tool Screenshot	9-19
Team Contribution	20
Challenges and Next Steps	21
Appendices	21-22

Introduction

The Festify project, an event management and ticketing platform designed to transform the way events are planned and attended, is covered in detail in this document. The platform's capabilities, which include event creation, ticket administration, analytics, and promotional tools, have been designed to help event organizers while providing attendees with an easy way to find, book, and interact with events.

This document serves as an overview of the project's user stories, product backlog, and release schedule. As seen in the illustration, the product backlog comprises essential features (epics) grouped under the headings of Organization Management, User Management, and Admin Management. By providing a user-centric platform, these features seek to meet the demands of both event planners and attendees.

User Stories

ID	Title	Description	Acceptance Criteria	Priority
SCRUM- 2	Event Creation	Event planners may establish new events by completing a form with information such as the event title, description, date, time, venue, and ticket choices. After being submitted, the event is included in the database and made available to the general public. The site will allow users to browse and sign up for these events.	Users can fill form with details and it gets saved in the database.	Highest
SCRUM- 5	Customize / Manage Events	Organizers can examine a list of their events using a management interface. Any event may be chosen, and its information can be edited, including the date and description. They may also delete events, which will take the event off the public listings and notify registered attendees of the cancellation.	Organizers can view their events and can customize the details and manage them	Medium
SCRUM- 6	Ticket Manageme nt	The ticket management system will enable organizers to establish pricing, availability, and sales limitations in addition to creating various kinds of tickets. Real-time ticket sales tracking and availability updates will be provided via the system. To safely process ticket purchases, it will also interact with a payment gateway.	Organizers can set ticket types, prices and availability. Ticket sales are updated exclusively.	High
SCRUM-7	Event Feedback	This tool offers post-event feedback to the organizers in terms of participant comments.	Organizers can view the feedback by users	Low
SCRUM- 9	More Visibility (PRO)	This pro feature will allow the organizations to display their events with more priority than regular organizations. This will be done by putting events on the main dashboard for the attendees to view.	Pro events are highlighted on top at the explore page	Lowest
SCRUM- 10	Simultane ous Event (PRO)	This pro-feature enables event planners to arrange and manage several events concurrently. They can effectively manage attendee involvement for all concurrent events, track ticket sales, and effortlessly monitor and amend event information.	Organizers can manage and create multiple events and tickets are trackable	Lowest

SCRUM- 12	Create / Manage Profile	Organizations can use this tool to build and update their profiles, including adding or changing contact data, and a name. Visitors may access information about the organization via the profile.	Organizers can create/edit profile and visitors can view their profile	Highest
SCRUM- 17	Promotion s	Organizers will have tools to create and send email campaigns to attendees, promoting events with custom messages, discount codes, or exclusive offers. Additionally, the system will allow organizers to generate visually appealing social media posters for platforms like Instagram, Facebook, and Twitter, enabling them to share event promotions easily.	Organizers can create email campaigns with custom text and discount codes. posters are generated to share across social media	Lowest
SCRUM- 16	Organizati on Dashboard	The organizational dashboard lets organizers see all their events, track ticket sales, check revenue, view attendee lists and easily manage their events.	Organizers can track events, ticket sales, revenue and attendee lists in one view	High
SCRUM- 13	User Registratio n	The user will need to register to be able to access any booking features. They will be sent to a form which will ask for details such as personal information as well as payment information	Users must complete form with personal and payment details to register	High
SCRUM- 25	User Dashboard	A User will be able to see their details and be able to edit the information if needed. This will be done by allowing the user to edit the UI and replace information in a form.	Users can view and edit their profile information	High
SCRUM- 14	Discover Events	This enables users to explore through the page and look at all the events happening that could be filtered down based on choices. Users can expand those events and look at more related details.	Users can browse and filter events and their details are expandable	Highest
SCRUM- 15	Ticket Booking	This allows attendees to select events, choose ticket types, and complete the booking process. It provides a smooth and secure experience, including payment options and booking confirmation.	Users can select event and choose ticket for payment. Confirmation of payment is given	Medium
SCRUM- 31	Database Creation	The database will include tables for user accounts, attendees, organizers, and events. Each event will be linked to the corresponding organizer using their unique organizer ID.	Database include tables for users, organizers, attendees and events.	Highest
SCRUM- 19	Payment Handling	The system will handle payments securely using trusted gateways like credit cards or online wallets. Admins will have the ability to track transactions and resolve payment-related issues.	Payments are processed through trusted gateways and admin can address transaction issues	Low

SCRUM- 20	Event Analytics	Admins can access a dashboard displaying data like ticket sales, popular events, and user activity. These insights are then provided to event organisers to help optimize the event and improve the user experience.	Dashboards displaying ticket sales, popular events. Details are shared with organizers.	Lowest
SCRUM- 18	Notify Attendees	The system will allow admins to send updates to attendees, such as reminders to upcoming bookings, changes to event schedules, cancellations, or additional instructions. Notifications will be sent via email or push notifications.	Email or push notifications sent for updates to the event, or cancellations and reminders	Lowest
SCRUM- 21	Moderate Events	Admins will have the ability to check event details before publishing them. They can approve, reject, or request modifications, and notify organizers of their decision. They can also edit or delete the events later once the event is hosted out completely.	Admins have ability to approve, reject, ask for changes for publishing. They can also delete the events if necessary.	Low
SCRUM- 24	QR - Code Generation	The system will generate a unique QR code for each booking. Admins will ensure that QR codes are linked to the correct event and can be used for seamless attendee check-ins at the venue.	Unique QR code for each booking which are linked to the event for easy check in.	Lowest

Product Backlog

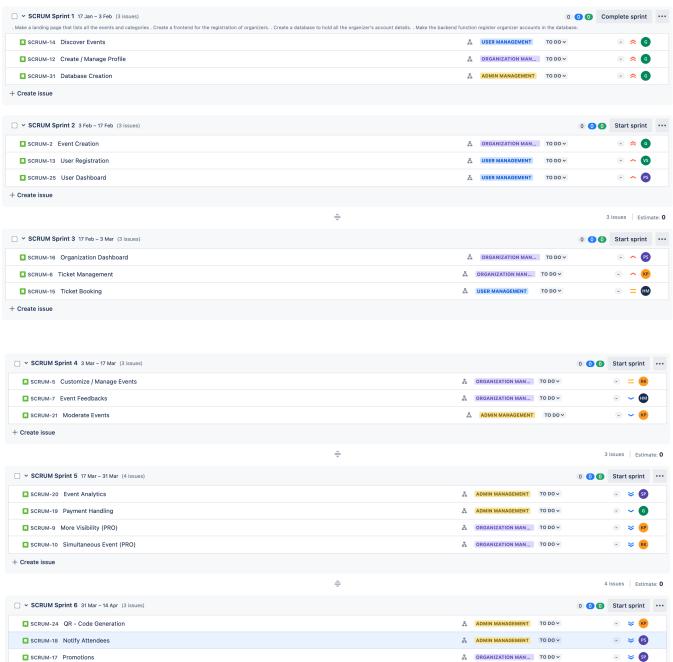
Task	Associated	Priority
	User Story	
SRUM-26 (Implement the event creation form.)	SCRUM-2	High
SCRUM-27 (Add input validation for mandatory fields (title, description,	SCRUM-2	Medium
date, time, venue).		
SCRUM-29 (Integrate the form with database to store event details.)	SCRUM-2	High
SCRUM-28 (Add a success or error notification after form submission.)	SCRUM-2	Medium
SCRUM-54 (Design the form for the event)	SCRUM-2	High
SCRUM-55 (Test the event creation functionality)	SCRUM-2	Low
SCRUM-37 (Profile Creation Page)	SCRUM-12	High
SCRUM-38 (Edit/Update Profile)	SCRUM-12	High
SCRUM-130 (Test this feature by adding profile details in the page)	SCRUM-12	Medium
SCRUM-97 (Design database schema using MySQL)	SCRUM-31	High
SCRUM-99 (Tables for user accounts, attendees, organizers, events,	SCRUM-31	High
and tickets)		
SCRUM-100 (Test Database queries)	SCRUM-31	Medium
SCRUM-131 (Create database for Promotional Codes)	SCRUM-31	Low

SCRUM-32 (Build an event management interface for organizers to view	SCRUM-5	Medium
their events.)		
SCRUM-33 (Create edit functionality to update event details.)	SCRUM-5	Medium
SCRUM-34 (Create delete functionality to remove events from listings.)	SCRUM-5	Medium
SCRUM-35 (Update the database for edited or deleted events.)	SCRUM-5	Medium
SCRUM-56 (test the feature by customizing / deleting the events)	SCRUM-5	Medium
SCRUM-86 (Create a UI for browsing, searching, and filtering)	SCRUM-14	High
SCRUM-87 (Display event cards with key details)	SCRUM-14	Medium
SCRUM-88 (Set up API endpoints for fetching events based on search	SCRUM-14	High
and filter criteria)		
SCRUM-89 (Write queries to retrieve events based on filters)	SCRUM-14	High
SCRUM-90 (Test the accuracy of searching and filtering events)	SCRUM-14	Low
SCRUM-57 (Implement pricing and availability for fields for tickets.)	SCRUM-6	High
SCRUM-58 (Develop real-time ticket tracking and availability updates.)	SCRUM-6	Medium
SCRUM-59 (Create Ticket Type Management Interface)	SCRUM-6	High
SCRUM-60 (test by adding different types of tickets and manage	SCRUM-6	Low
tickets)		
SCRUM-62 (Integrate the feedback section to organization dashboard)	SCRUM-7	High
SCRUM-72 (Give the functionality to the organizer to reply back to the	SCRUM-7	Medium
comments)		
SCRUM-70 (test the comments and replies)	SCRUM-7	Low
SCRUM-61 (Implement priority event tagging for PRO accounts)	SCRUM-9	High
SCRUM-63 (Develop filtering logic for priority events)	SCRUM-9	Medium
SCRUM-128 (test an validate PRO event visibility feature)	SCRUM-9	Low
SCRUM-67 (enable PRO permissions for simultaneous event posting)	SCRUM-10	High
SCRUM-68 (Create multiple events posting form)	SCRUM-10	High
SCRUM-69 (Integrate backend support for multiple event submissions)	SCRUM-10	High
SCRUM-71 (Notify PRO organizers of submission errors)	SCRUM-10	Medium
SCRUM-129 (Test this pro feature for multiple events)	SCRUM-10	Low
SCRUM-79 (Implement a editable form for account Info)	SCRUM-25	High
SCRUM-80 (List view for past booking)	SCRUM-25	Medium
SCRUM-81 (Account delete button)	SCRUM-25	Medium
SCRUM-83 (Write logic for fetching past booking from database)	SCRUM-25	High
SCRUM-84 (Implement Account deletion from database, also deleting	SCRUM-25	Medium
bookings associated with that account)		
SCRUM-85 (test if it's fetching the right info and validate account	SCRUM-25	Low
deletion)		
SCRUM-42 (Develop the form for sending promotional events)	SCRUM-17	High
SCRUM-44 (Social Media Poster Generator)	t	Medium
Serier 44 (Secial Field Feeter Series at 197	SCRUM-17	Medium
SCRUM-45 (Social Media Sharing Integration)	SCRUM-17 SCRUM-17	Medium
SCRUM-45 (Social Media Sharing Integration)	SCRUM-17	Medium

SCRUM-48 (Event Overview Panel)	SCRUM-16	High
SCRUM-49 (Tickets Sales Tracker)	SCRUM-16	High
SCRUM-50 (Revenue Monitoring)	SCRUM-16	Medium
SCRUM-51 (Attendee List Management)	SCRUM-16	Medium
SCRUM-52 (Feedback Center)	SCRUM-16	Low
CONTROL (FOODBOOK CONTO)	CONOTTIO	2000
SCRUM-73 (Create a registration form to collect account info)	SCRUM-13	High
SCRUM-74 (Validate user input based on formatting)	SCRUM-13	High
SCRUM-76 (Store the account information securely in the database)	SCRUM-13	High
SCRUM-77 (Send a confirmation email upon account registration)	SCRUM-13	Medium
SCRUM-78 (testing registration for valid inputs and email confirmation	SCRUM-13	Low
functionality)		
SCRUM-91 (Create a UI to present the events like the details and the	SCRUM-15	High
ticket types, also showing real-time availability)	CODUM 45	11:
SCRUM-92 (Smooth checkout process with payment integration)	SCRUM-15	High
SCRUM-94 (Implement the functionality to handle payment and	SCRUM-15	High
purchasing of a ticket)	0001114.45	Mariliana
SCRUM-95 (Write logic for sending confirmation email)	SCRUM-15	Medium
SCRUM-96 (Test the payment process with different kinds of Input, test	SCRUM-15	Low
if the real time ticketing thing is working)		
SCRUM-102 (Create a payment form for processing payments with all	SCRUM-19	High
valid options)		16
SCRUM-103 (Display payment statuses (Success, pending, failed)	SCRUM-19	High
SCRUM-105 (Store transaction details in the database for tracking and	SCRUM-19	High
record)		lg.
SCRUM-106 (Update the revenue generated per event)	SCRUM-19	Medium
SCRUM-108 (test payment handling system)	SCRUM-19	Low
, , , , , , , , , , , , , , , , , , , ,		
SCRUM-110 (Create an analysis dashboard)	SCRUM-20	High
SCRUM-111 (Give organizer the option to export the analysis (Excel	SCRUM-20	Medium
file))		
SCRUM-112 (Fetch data from the front end)	SCRUM-20	High
SCRUM-113 (Write queries to aggregate data)	SCRUM-20	High
SCRUM-114 (Schedule periodic jobs to update analysis)	SCRUM-20	Medium
SCRUM-116 (Test the analysis the system is generating)	SCRUM-20	Low
CODUM 440 (Develop notification and analysis	CODUNATO	1 Cat
SCRUM-119 (Develop notification and email template)	SCRUM-18	High
SCRUM-120 (Create an option for the admin to send custom	SCRUM-18	High
notification)	CODUM 40	Llieda
SCRUM-122 (Configure email service and notifications)	SCRUM-18	High
SCRUM-123 (Write logic for schedule reminders)	SCRUM-18	Medium
SCRUM-124 (Track notification status)	SCRUM-18	Medium
SCRUM-125 (Test the notifications)	SCRUM-18	Low
SCRUM-115 (Create a form (approval/rejection/request changes)) for	SCRUM-21	High
events and comments space for admin	33110111-21	1 11811
SCRUM-117 (send email and portal notification for admin decision)	SCRUM-21	Medium
SCRUM-118 (Update records in the database)	SCRUM-21	Medium
Solvon-1 to (Ohagie Lecolas III file agranase)	JUNUIYI-Z I	เาเซนเนเไไ

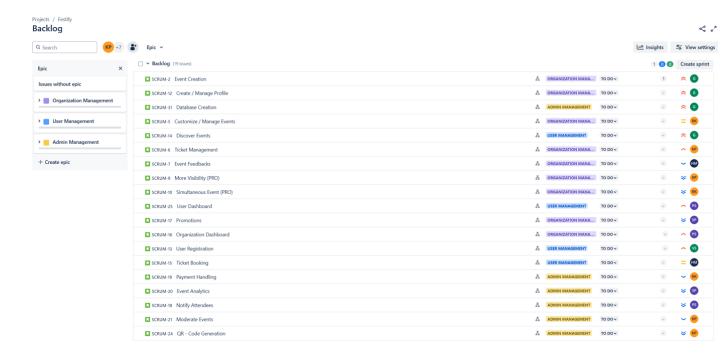
SCRUM-121 (Test the functionality)	SCRUM-21	Low
SCRUM-98 (Display generated QR code to the user)	SCRUM-24	High
SCRUM-101 (Use library to create QR codes with the ticket information)	SCRUM-24	High
SCRUM-104 (Save the QR code in the database)	SCRUM-24	Medium
SCRUM-107 (Send QR code via email to the user)	SCRUM-24	Medium
SCRUM-109 (test the QR Functionality)	SCRUM-24	Low

Sprint Backlogs

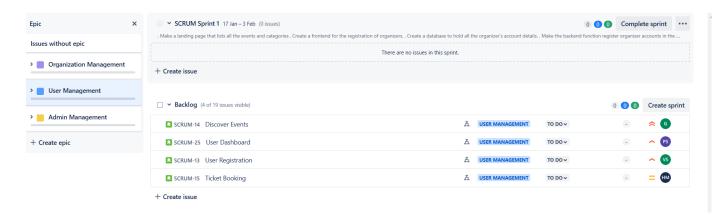


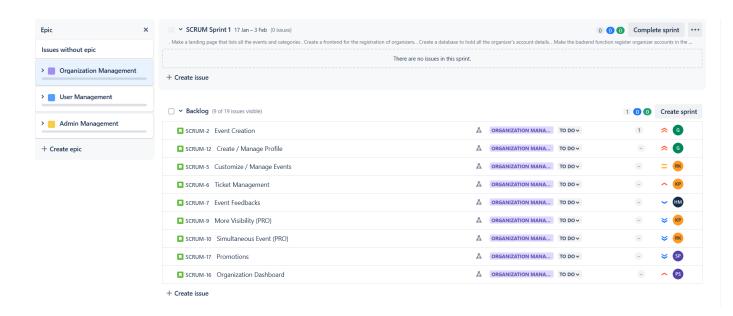
Tool Screenshots

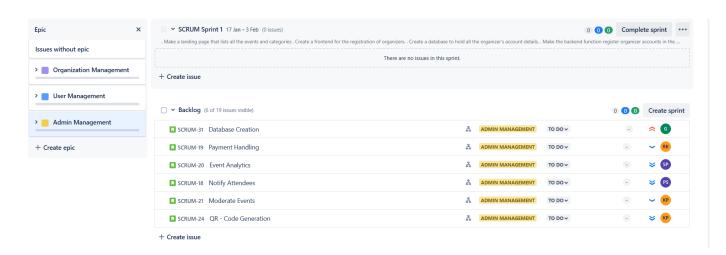
EPICS



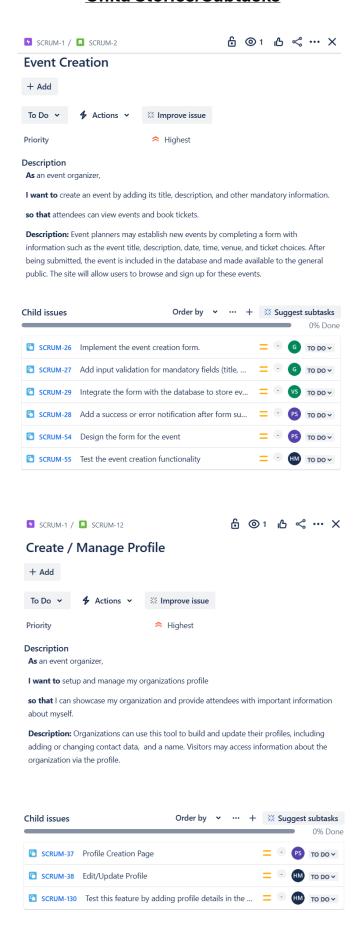
USER STORIES

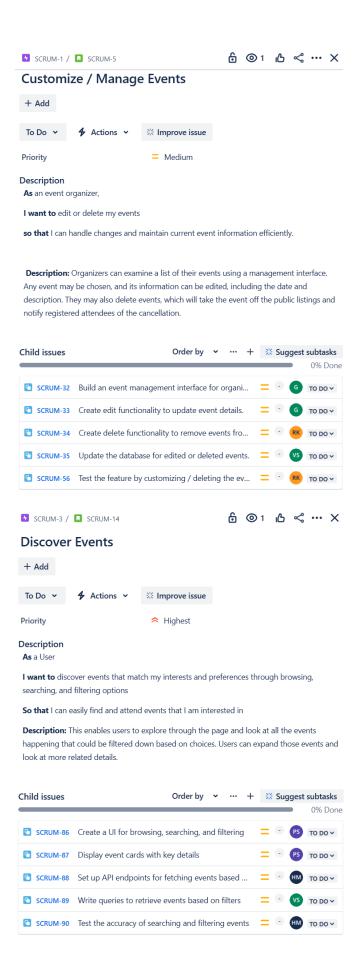


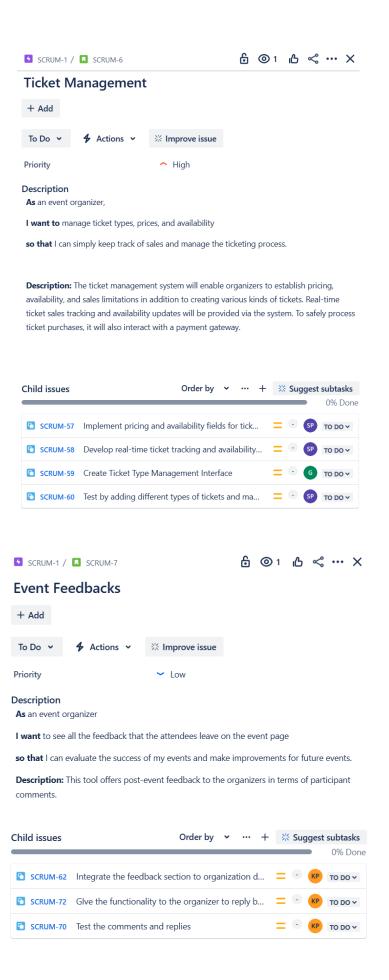


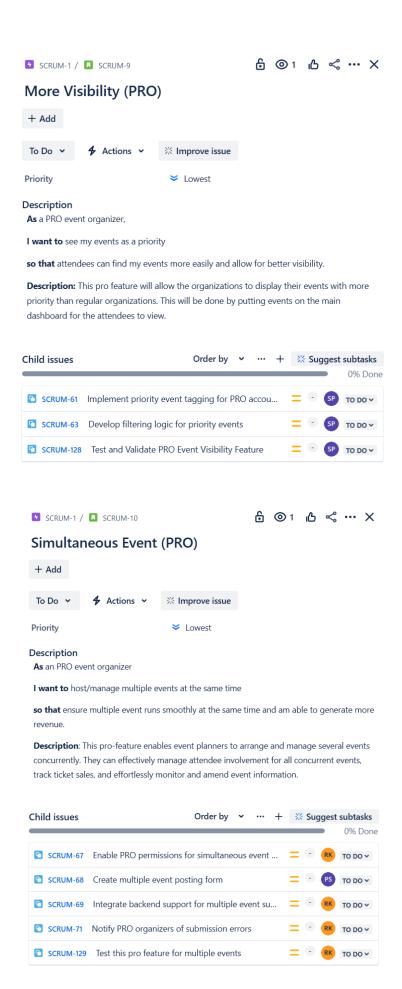


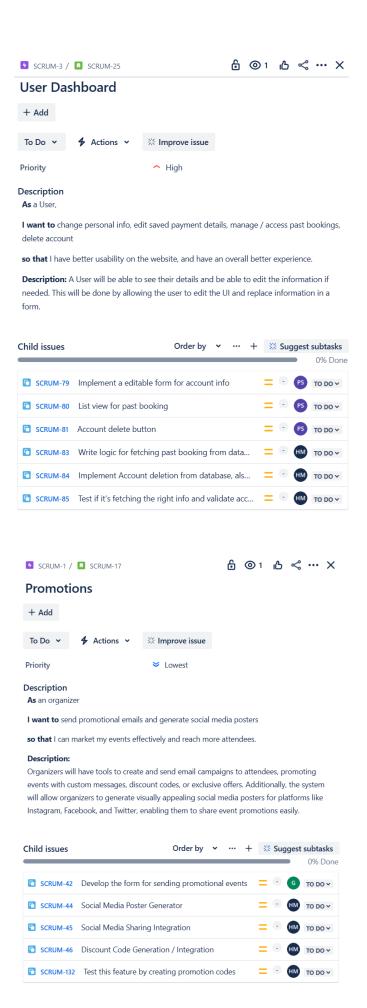
Child Stories/Subtasks

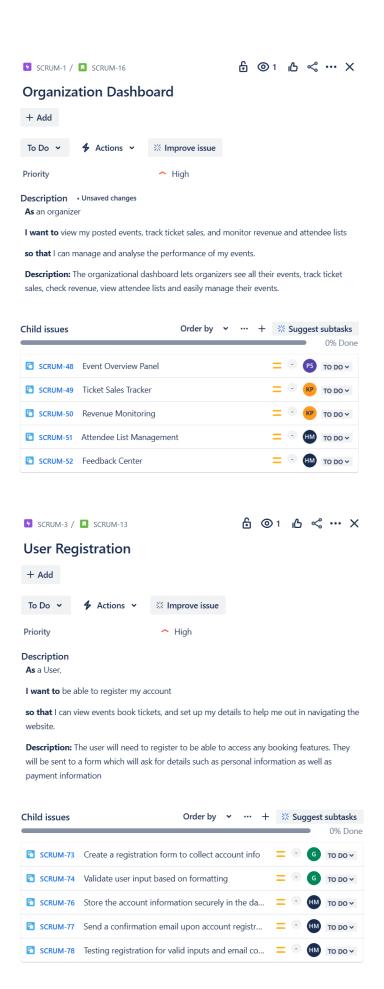


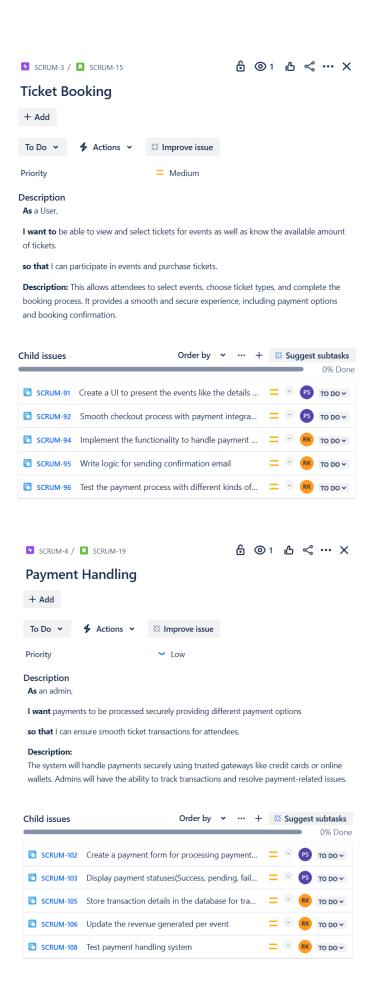


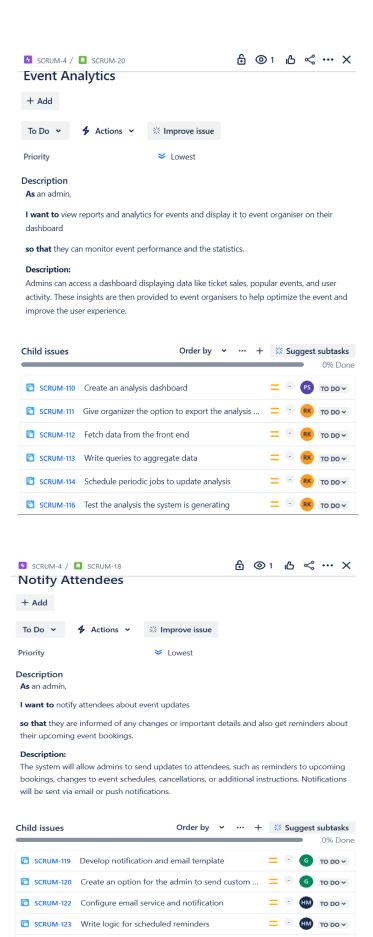










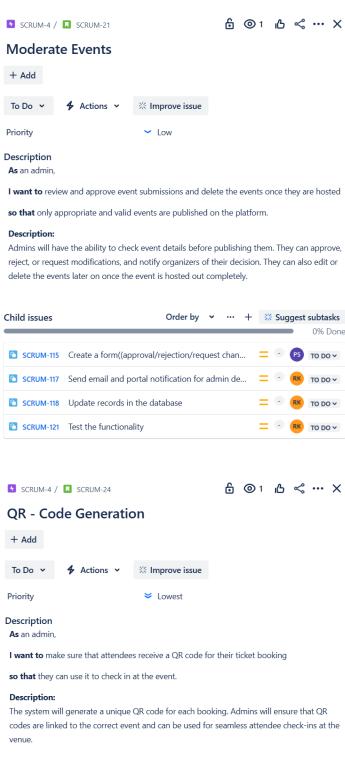


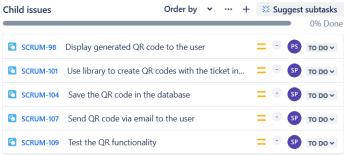
SCRUM-124 Track notification status

SCRUM-125 Test the notifications

_ - HM TO DO Y

_ - HM TO DO Y





Team Contributions

Team Member	Contribution
Gautam . (SCRUM MASTER)	Arranged Scrum meetings, User Story Creation,
	Sprint Backlog management, and content
	contributor on release planning documents.
Harmanjot Malhi	User Story Creation, Tasks and Descriptions,
	Ranking and Assigning tasks.
Kartikkumar Parekh	Tasks and Descriptions, Content contributor for
	release planning and development research.
Parampal Singh	Jira management, User Story Creation, Content
	contributor for release planning, Tasks and
	Descriptions
Rohal Kabir	Jira management, User Story Creation, Tasks and
	Descriptions, Content contributor for release
	planning
Sumant Patel	User Story Creation, Tasks and Descriptions,
	Development research and Content contributor
	for release planning.
Vivek Salwan	Development research and Content contributor
	for release planning. User stories and backlog
	lead on release document.

This team had made a very even contribution, there were very productive team meetings where ideas were shared and built upon, as well as the required work needed to complete this release document, which was completed during these times.

Challenges and Next Steps

Challenges faced in the planning process:

We, as a team, didn't face any major challenges in the process, but we just had minor challenges with certain things. We had to devote some time beforehand to figure out the working flow for Jira, for example, maintaining backlog and knowing about user story and subtask levels. We also had a bit of a challenge deciding on the scope of the app for what features must be included and what not. These challenges were resolved through teamwork and clear communication.

Next Steps:

In the two-week Sprint 1, we'll focus on completing three key user stories:

- Discover Events
- o Database Creation
- Create/Manage Profile

Tasks have been divided and assigned to members based on; frontend for UI/UX, backend for application logic, and database tasks. We'll have periodic check-ins to ensure progress, resolve issues, and keep us aligned. By the sprint's end, we aim to have these core features ready.

Appendices

Meeting 1

9 January 2025, 11:00 am to 1:00 pm, ST 236

- Discussed the project schedule and sprint size.
- Selected the Product Owner and Scrum Master

Meeting 2

10 January 2025, 8:00 pm to 10:00 pm, ST 237A

- Explored and finalized the project idea.
- Defined product features and core functionalities.
- Prepared the project proposal and finalized all necessary details

Meeting 3

15 January 2025, 11:30 am to 3:30 pm, Library 6th floor

- Discussion on Features
 - o Identified and finalized all features to be implemented in the project.
- Release Planning
 - o Created a Word document for release planning.
 - o The document includes a rough idea of what has to be implemented.
- Epics and User Stories
 - Set up Jira Dashboard.
 - Defined all epics based on project goals.
 - o Broke down epics into user stories according to the requirements.

Meeting 4

16 January 2025, 10:30 am to 1:00 pm, ST 503B

- User Stories Refinement
 - Reviewed and refined existing user stories to ensure clarity and alignment with requirements.
- Adding Descriptions
 - Added detailed descriptions to each user story to provide context and clarity.
- Child Tasks/Subtasks
 - Began listing child tasks/subtasks under each user story for better task management and implementation planning.

Meeting 5

17 January 2025, 8:00 pm to 11:00 pm, Plaza building

- User Stories
 - Refined all child issues
 - o Completed all child issues
- Task Management
 - o Established priorities for all tasks
 - o Assigned tasks to group members
- Release Planning
 - o Initiated the release planning document/report
- Next Steps
 - o Finalize and review the release planning document
 - o Schedule follow-up meetings as needed