1.

2. Requirements

1. Functional Requirements

Function /Feature/ Task	Priority	Difficulty	Complexity	Testing	Definition of Done	Resources
	(Scale between 1-5, where 1 is the lowest and 5 is the highest)			*Identify and describe how and when the product will be tested over the course of the project/semester.*	*Describe a set of conditions that allow your team to assess the product's completion for the project/semester.*	*List the required library, API, language, package, connected function(s), connected database etc.*
Fun_1	*2*	*3*	*4*	*(A)* *e.g., see section A of the 'Acceptance Testing' section*	*(1)* *e.g., see section A of the 'Definition of Done' section*	
Community Member Onboarding	5	3	3	Unit test account creation, integration test full onboarding flow	A visitor can become a Community Member.	MongoDB, Node.js, Express, Stripe API
Account Management	5	3	3	Unit test CRUD actions, integration test role updates	User can log in/out, update details, deactivate account, and role changes propagate correctly	MongoDB, JWT, bcrypt
Membership Payment	5	4	4	Simulate Stripe test payments, verify webhook	Payment successfully processed, receipt emailed,	Stripe API

Integration				receipts	membership marked as paid	
Meeting Minute Access	4	2	2	Upload/download test, permission checks	An Administrator can upload and publish Meeting Minutes, and Community Members can view them.	MongoDB GridFS, role-based access middleware
Editorial Dashboard	4	4	4	CRUD content test, image upload test	A Content Manager or Administrator can make changes to the public content.	Vue, Node.js, MongoDB
Member List Viewer	4	2	2	Query test, filter test	Administrators can view details of all members on the list.	MongoDB, Mongoose
Event Calendar	3	3	3	CRUD event test, calendar UI test	Administrators can create events visible to Visitors.	Vue, MongoDB
Blog Submission and Approval	3	3	3	Submission, approval/rejection, publish visibility tests	A Visitor can submit and an Administrator can approve.	MongoDB, Vue, Node.js, Express
Contact Us Form Email Integration	2	2	2	Form validation test, email send test	The authorised email receives the Visitor's enquiry.	Nodemailer (Node.js module for sending emails)

2. Non-Functional Requirements

Function /Feature/ Task	**Priority**	**Difficulty**	**Complexity**	**Testing**	**Definition of Done**	**Resources**
	(Scale between 1-5, where 1 is the lowest and 5 is the highest)			*Identify and describe how and when the product will be tested over the course of*	*Describe a set of conditions that allow your team to assess the product's*	*List the required library, API, language, package, connected*

				the project/semester.	*completion for the project/semester.*	*function(s), connected database etc.*
Fun_1	*2*	*3*	*4*	*(A)* *e.g., see section A of the 'Acceptance Testing' section*	*(1)* *e.g., see section A of the 'Definition of Done' section*	*xxx*
Performance	5	3	3	Load testing (50 concurrent users), response time monitoring	System responds under 2 seconds with 50 concurrent users	Apache JMeter, Chrome DevTools
Security	5	4	4	Penetration testing, JWT expiry checks	All pages behind login require valid JWT, passwords are hashed, sensitive data encrypted	bcrypt, JWT, Helmet.js, TLS
Usability	4	2	2	User acceptance tests, UI/UX review	≥80% of users complete key tasks without guidance and report ease of use	Figma (prototyping), user test participants
Reliability	4	3	3	Uptime monitoring, error logging	≥99% uptime during tests, graceful error handling and logging implemented	PM2, Winston logger
Scalability	3	4	3	Simulated scaling test	Application handles increased traffic by horizontal scaling	Docker, Kubernetes (optional)
Maintainability	3	3	3	Code review, linting, documentation audit	All code is documented and passes linting with no critical errors	ESLint, Prettier, JSDoc

Accessibility	3	2	2	Screen reader test, WCAG checklist	All pages meet WCAG 2.1 AA guidelines	axe DevTools, Lighthouse
Backup and Recovery	2	2	2	Database backup/restore simulation	Weekly backups run successfully, recovery completed within 2 hours	MongoDB Atlas backups

3. User Stories/Narratives

2.3.1. Community Member Onboarding

ID	User Story	Priority	Estimate
U101	As a Visitor, I want to locate and press the “Join” button so that I can start the onboarding process to become a Community Member.	H	4
U102	As a Visitor, I want to provide my personal information when becoming a member, so that I can progress in the onboarding process.	H	6
U103	As a Visitor, I want to securely create a password so that I can access my account in the future.	H	5
U104	As a Visitor, I want to provide my payment details so that I can complete the membership contribution during the onboarding process.	H	6
U105	As a Visitor, I want a new webpage to confirm that onboarding is complete so that I receive a clear indication that it was successful.	L	2

ID	User Story	Priority	Estimate
U106	As a Visitor, I want to be redirected to the Login page so that I can immediately access my account after onboarding.	L	3

2.3.2. Account Management

ID	User Story	Priority	Estimate
U201	As an Account Holder, I want to locate and press the “Login” button, so that I can access my account.	H	4
U202	As an Account Holder, I want to log in using my email and password so that I can access the correct account.	H	6
U203	As an Account Holder, I want my password to be masked so that it is not displayed when in public.	M	1
U204	As an Account Holder, I want to update my personal details so that my account information stays accurate.	M	4
U205	As an Account Holder, I want to deactivate my account so that all my data is removed from the system.	M	2
U206	As an Account Holder, I want to change my password so that I can strengthen the security of my account.	M	2
U207	As an Account Holder, I want to update my email address so that notifications are forwarded to my preferred communication method.	M	2
U208	As an Administrator, I want to create a Content Manager account so that they can edit the public-facing content.	H	5

ID	User Story	Priority	Estimate
U209	As an Administrator, I want to create an Administrator account so that they can access everything on the system.	H	5
U210	As an Administrator, I want to deactivate an Account Holder so that they lose system access and permissions.	M	2
U211	As an Administrator, I want to change the role of an Account Holder so that permissions are enforced correctly.	L	4

2.3.3. Membership Payment Integration

ID	User Story	Priority	Estimate
U301	As a Visitor or Community Member, I want to be redirected to the payment platform (Stripe) so that I can process my payment.	H	7
U302	As a Visitor, I want to use the Stripe payment platform, so that I can securely process my payment with Pyrmont Action.	H	8
U303	As the Treasurer, I want to be sent email notifications when a payment is processed so that I can update financial records.	H	6
U304	As a new or current Community Member, I want to receive an automatic receipt by email after payment so that I have proof of membership.	H	6
U305	As a Community Member, I want to be notified when my membership is about to expire so that I can renew on time.	M	4

ID	User Story	Priority	Estimate
U306	As a Community Member, I want my membership status and renewal date shown so that I am aware when to renew it.	L	3

2.3.4. Meeting Minute Access

ID	User Story	Priority	Estimate
U401	As an Administrator, I want to upload Meeting Minutes as a PDF on a private-facing webpage so that members can access them securely.	H	6
U402	As an Administrator, I want to control when Meeting Minutes are published so that only finalised documents are visible to members.	H	4
U403	As a Community Member, I want to view published Meeting Minutes so that I can stay informed about decisions and updates.	H	3
U404	As an Administrator, I want to edit or replace uploaded Meeting Minutes so that I can correct errors or update content.	M	5
U405	As a Community Member, I want to receive a notification when new Meeting Minutes are published so that I can stay updated.	M	4
U406	As an Administrator or Community Member, I want to organise Meeting Minutes by date so that I can easily find	L	5

ID	User Story	Priority	Estimate
	records.		

2.3.5. Editorial Dashboard

ID	User Story	Priority	Estimate
U501	As an Administrator or Content Manager, I want to access the editorial dashboard so that I can manage public-facing webpages in one place.	H	7
U502	As an Administrator or Content Manager, I want to see a list of all editable webpages so that I can quickly navigate to the one I need.	H	5
U503	As an Administrator or Content Manager, I want to select a webpage from the dashboard so that I can access editing features for it.	H	3
U504	As an Administrator or Content Manager, I want to upload images and text to a webpage so that I can enrich the content.	M	6
U505	As an Administrator or Content Manager, I want to remove outdated items from a webpage so that the content stays relevant.	M	4

2.3.6. Member List Viewer

ID	User Story	Priority	Estimate
U601	As an Administrator, I want to view a list of all registered Community Members so that I can monitor membership.	M	7
U602	As an Administrator, I want each member's full name to be displayed so that I can identify individuals.	M	2
U603	As an Administrator, I want each member's email address to be shown so that I can contact them if needed.	M	2
U604	As an Administrator, I want to see whether a member is active or inactive so that I can track renewals.	M	3
U605	As an Administrator, I want to filter the list by active/inactive status so that I can manage renewals efficiently.	L	6
U606	As an Administrator, I want to search for a member by name or email so that I can locate specific records.	L	6

2.3.7. Event Calendar

ID	User Story	Priority	Estimate
U701	As an Administrator, I want to add new events to the Event Calendar so that the community can stay informed.	H	6
U702	As a Visitor, I want to view upcoming events on the Event Calendar so that I can participate in community activities.	H	4

ID	User Story	Priority	Estimate
U703	As an Administrator, I want to include the event name, date, location, and description so that all relevant details are visible.	H	6
U704	As an Administrator, I want to edit event details after posting so that I can correct or update information.	M	7
U705	As an Administrator, I want to delete past or cancelled events so that the calendar stays clean and accurate.	M	5
U706	As a Visitor, I want to click on an event to see full details so that I can decide whether to attend.	L	4
U707	As a Visitor, I want to see a calendar view and a list view so that I can choose how to browse events.	L	5
U708	As an Administrator, I want to duplicate past events so that I can reuse them for future planning.	L	7

2.3.8. Blog Submission and Approval

ID	User Story	Priority	Estimate
U801	As a Visitor, I want to write and submit blog content so that I can express my views to the community.	M	5
U802	As a Visitor, I want to draft a blog post anonymously so that I can share my thoughts without revealing my identity.	M	5
U803	As a Visitor, I want to draft a blog post with my name so that I can be credited for my contribution.	M	5

ID	User Story	Priority	Estimate
U804	As a Visitor, I want to be prompted to review my draft before submitting it, so that I can fix grammatical errors and improve clarity.	M	5
U805	As an Administrator, I want to view all submitted blog drafts so that I can manage approvals.	H	7
U806	As an Administrator, I want to reject inappropriate blog posts so that the site remains respectful and safe.	H	7
U807	As an Administrator, I want to edit blog posts before publishing so that I can fix formatting or minor issues.	L	7
U808	As a Visitor, I want to view published blog posts so that I can read community contributions.	H	6

2.3.9. Contact Us Form Email Integration

ID	User Story	Priority	Estimate
U901	As a Visitor, I want to submit a message through the contact form so that I can reach Pyrmont Action.	M	3
U902	As an Enquiry Email Holder, I want to receive the Visitor’s message, name, email address, and subject line via email so that I can respond appropriately.	L	3
U903	As an Enquiry Email Holder, I want the contact form to be configured to send emails to a specific email address so that enquiries reach the correct recipient.	L	3

4. User Narratives

Use Case ID	UC101: Community Member Onboarding
User Story	As a Visitor, I want to become a Community Member so that I can access exclusive content.
Goal	The Visitor completes registration and becomes a Community Member.
Priority	High
Actors	Primary Actor: Visitor Secondary Actors: Web application, Database, Stripe API
Pre-conditions	- Visitor is on the onboarding webpage. - Visitor has a valid email address and payment method.
Post-conditions	- Visitor's account is stored in the database and marked as active. - Visitor is redirected to the login page.
Trigger	The Visitor clicks the "Join" button on the onboarding page.
Main Flow	1. The Visitor clicks the "Join" button. 2. The system displays the registration form. 3. The Visitor enters personal details and creates a secure password. 4. The Visitor is redirected to the Stripe payment portal. 5. The Visitor enters payment details and completes the payment. 6. The system saves the account information, marks the member as active, and shows a success confirmation. 7. The Visitor is redirected to the login page. 8. The use case ends.
Exceptions	E1. Steps 1–8 – closing the browser during onboarding pauses the process. E2. Step 5 – Payment fails due to declined card or timeout.
Includes	UC205: Account Login, UC301: Membership Payment
Supporting Information	All personal data must comply with the organisation's privacy policy and PCI DSS for payments.
Non-functional Requirements	Performance: Average response time < 2 seconds. Security: Passwords hashed with bcrypt, all steps use TLS.

Alternate Flow 1	“Invalid Details” Trigger The Visitor enters missing or invalid personal information. Step The system highlights invalid fields and displays error messages. The Visitor corrects the details and resubmits the form. Alt Flow The Visitor’s corrected data is validated successfully, and they continue from Step 3 of the Main Flow. Post conditions The corrected data is stored and used for the account creation process. Exceptions E1. Step 2 – If incorrect five times, the session locks for five minutes.
Alternate Flow 2	“Payment Cancelled” Trigger The Visitor cancels the payment on the Stripe page. Step The system returns to the onboarding page with a banner saying “Payment was not completed.” Alt Flow The Visitor can retry the payment or exit onboarding. Post conditions The account remains inactive until the payment succeeds, at which point the Visitor rejoins Step 5 of the Main Flow. Exceptions E1. Step 5 – If the session times out, the onboarding process restarts from Step 1.

Use Case ID	UC201: Account Management
User Story	As an Account Holder, I want to manage my account so that my information stays current.
Goal	The Account Holder updates personal information and manages access securely.
Priority	High
Actors	Primary Actor: Account Holder Secondary Actors: Database
Pre-conditions	The Account Holder is logged in.
Post-conditions	Account details are updated in the database.
Trigger	The Account Holder selects “My Account” from the menu.

Main Flow	1. The Account Holder logs in. 2. They navigate to My Account. 3. They edit personal details or security settings. 4. They click Save. 5. The system validates and stores the updates. 6. The system confirms success. 7. The use case ends.
Exceptions	E1. Invalid field inputs (e.g. weak password, duplicate email). E2. Session timeout during editing.
Includes	UC206: Change Password
Supporting Information	All personal data must comply with the organisation's privacy policy.
Non-functional Requirements	Performance: Average update < 2 seconds. Security: Passwords hashed, data transmitted via TLS.
Alternate Flow 1	"Weak Password" Trigger The Account Holder enters a password that fails the strength policy. Step The system displays a strength meter and password requirements. Alt Flow The Account Holder chooses a stronger password and resubmits. Post conditions The new password is saved (hashed). Exceptions E1. Step 3 – If weak 5 times, account is temporarily locked.
Alternate Flow 2	"Email Already Used" Trigger The Account Holder enters an email already registered. Step The system shows an error: "Email is already registered." Alt Flow The Account Holder enters a different email. Post conditions The new email is saved. Exceptions E1. Step 3 – System cannot verify email due to timeout.

Use Case ID	UC301: Membership Payment Integration
User Story	As a Member, I want to pay for membership so that I can maintain access.

Goal	The Member completes a secure online payment and renews their membership.
Priority	High
Actors	Primary Actor: Member Secondary Actors: Stripe API, Treasurer
Pre-conditions	- The Member is logged in. - The Stripe payment system is active.
Post-conditions	- Payment is processed and recorded. - Membership expiry is extended. - A receipt is emailed to the Member.
Trigger	The Member clicks “Renew Membership”.
Main Flow	1. The Member clicks Renew Membership. 2. The system redirects to the Stripe checkout page. 3. The Member enters payment details and confirms. 4. Stripe processes the payment and returns success. 5. The system updates the membership status. 6. The system emails a receipt. 7. The use case ends.
Exceptions	E1. Card declined or payment error. E2. Webhook confirmation delayed.
Includes	UC905: Email Notification
Supporting Information	The payment must meet PCI DSS security standards.
Non-functional Requirements	Performance: End-to-end < 3 seconds after Stripe confirmation. Security: TLS 1.2+, Stripe idempotency keys.
Alternate Flow 1	“Card Declined” Trigger The payment is declined by Stripe. Step The system shows the decline reason and asks the Member to try again. Alt Flow The Member enters new card details and resubmits. Post conditions The payment is accepted and processed. Exceptions E1. Step 3 – If declined 3 times, transaction is cancelled.

Alternate Flow 2	“Webhook Delayed” Trigger Stripe confirmation is delayed. Step The system shows a “pending” banner and retries status in the background. Alt Flow When the webhook arrives, the system updates the membership and emails the receipt. Post conditions The payment is recorded correctly. Exceptions E1. Step 4 – Webhook never arrives → flagged for manual review.

Use Case ID	UC401: Meeting Minute Access
User Story	As an Administrator, I want to publish meeting minutes so that members stay informed.
Goal	The Administrator uploads and publishes meeting minutes that members can view.
Priority	High
Actors	Primary Actor: Administrator Secondary Actors: Database, Community Members
Pre-conditions	- The Administrator is logged in with the correct permissions.
Post-conditions	- The meeting minutes are saved in the database and visible to members.
Trigger	The Administrator selects “Upload Minutes” from the admin dashboard.
Main Flow	1. The Administrator selects a PDF file of the minutes. 2. The Administrator clicks Upload. 3. The system validates and saves the file with metadata (date, title). 4. The Administrator toggles Publish. 5. The minutes appear on the members’ page for viewing or download. 6. The use case ends.
Exceptions	E1. The uploaded file is not a PDF. E2. The file size exceeds the maximum limit.
Includes	UC403: Member Views Minutes
Supporting Information	Files must be securely stored and accessible only by authorised users.
Non-functional	Performance: Upload < 5 seconds for 10 MB files.

Requirements	Security: Role-based access enforced.
Alternate Flow 1	“Draft Only” Trigger The Administrator uploads but does not publish. Step The minutes are saved as Draft and visible only to the Administrator. Alt Flow The Administrator can return later to publish them. Post conditions The minutes remain hidden from members. Exceptions E1. Step 4 – Draft is deleted accidentally.
Alternate Flow 2	“Permission Denied” Trigger A non-admin user attempts to upload minutes. Step The system blocks the upload attempt and displays an error message. Alt Flow The user is told to request admin access. Post conditions No file is stored. Exceptions E1. Step 2 – Event not logged due to a system error.

Use Case ID	UC501: Editorial Dashboard
User Story	As an Administrator or Content Manager, I want to edit pages so that content stays current.
Goal	The user edits and publishes content on the website.
Priority	High
Actors	Primary Actor: Administrator or Content Manager Secondary Actors: Database, Web server
Pre-conditions	The Administrator/Content Manager is logged in.
Post-conditions	The changes are saved and appear live on the website.
Trigger	The user selects a webpage from the editorial dashboard.

Main Flow	1. The user opens the editorial dashboard. 2. The user selects a webpage tile. 3. The system shows the editing interface. 4. The user updates text, images, or layout. 5. The user clicks Publish. 6. The system updates the live content. 7. The use case ends.
Exceptions	E1. Invalid image file format. E2. Internet connection lost during save.
Includes	UC504: Upload Content
Supporting Information	Version history should be stored for all edits.
Non-functional Requirements	Performance: Update reflects within 2 seconds. Maintainability: All changes logged with editor name and time.
Alternate Flow 1	“Save as Draft” Trigger The user chooses to save without publishing. Step The content is stored as a Draft only visible to editors. Alt Flow The user can later publish it. Post conditions The content is not visible publicly. Exceptions E1. Step 4 – Draft is lost due to a browser crash.
Alternate Flow 2	“Edit Conflict” Trigger Two users try editing the same page simultaneously. Step The system detects a conflict and blocks the second save. Alt Flow The second user is prompted to refresh and merge changes. Post conditions Only one version is saved. Exceptions E1. Step 5 – Conflict detection fails and overwrites changes.

Use Case ID	UC601: Member List Viewer
User Story	As an Administrator, I want to view all registered members so I can manage the community.
Goal	The Administrator sees a searchable list of members.
Priority	Medium

Actors	Primary Actor: Administrator Secondary Actors: Database
Pre-conditions	The Administrator is logged in.
Post-conditions	The list of members is displayed with details.
Trigger	The Administrator selects “Members” from the admin dashboard.
Main Flow	1. The Administrator clicks Members. 2. The system loads all member records. 3. The Administrator can scroll, search, or filter. 4. The Administrator views individual member details. 5. The use case ends.
Exceptions	E1. No members exist. E2. Search query returns no results.
Includes	UC605: Filter Active/Inactive
Supporting Information	Must support pagination for large lists.
Non-functional Requirements	Performance: Load 500 members in < 2 seconds. Security: Admin-only access.
Alternate Flow 1	“No Results” Trigger The search finds no members. Step The system shows “No members found.” Alt Flow The Administrator resets the search filter. Post conditions The full list reloads. Exceptions E1. Step 3 – The search function times out.
Alternate Flow 2	“Server Error” Trigger The database query fails. Step The system displays “Unable to load members.” Alt Flow The Administrator retries after a few seconds. Post conditions The list loads successfully. Exceptions E1. Step 2 – Multiple retries fail due to server crash.

Use Case ID	UC701: Event Calendar

User Story	As an Administrator, I want to add events so that the community stays informed.
Goal	The Administrator creates a new event and publishes it.
Priority	High
Actors	Primary Actor: Administrator Secondary Actors: Database
Pre-conditions	The Administrator is logged in.
Post-conditions	The event is stored and visible on the calendar.
Trigger	The Administrator clicks “Add Event”.
Main Flow	1. The Administrator clicks Add Event. 2. The system opens the event form. 3. The Administrator fills in event details. 4. The Administrator clicks Publish. 5. The event appears on the calendar. 6. The use case ends.
Exceptions	E1. Invalid or missing fields. E2. Invalid date format.
Includes	UC704: Edit Events
Supporting Information	The event should include title, date, time, location, and description.
Non-functional Requirements	Performance: Events display within 2 seconds. Usability: Calendar view supports both list and month views.
Alternate Flow 1	“Save as Draft” Trigger The Administrator saves without publishing. Step The event is stored as Draft. Alt Flow The Administrator can publish later. Post conditions The event is hidden from public view. Exceptions E1. Step 3 – Draft not saved due to session timeout.

Alternate Flow 2	“Duplicate Event” Trigger The Administrator duplicates a past event. Step The system copies the details into a new form. Alt Flow The Administrator edits and publishes it. Post conditions The new event appears on the calendar. Exceptions E1. Step 2 – Wrong date copied from original.

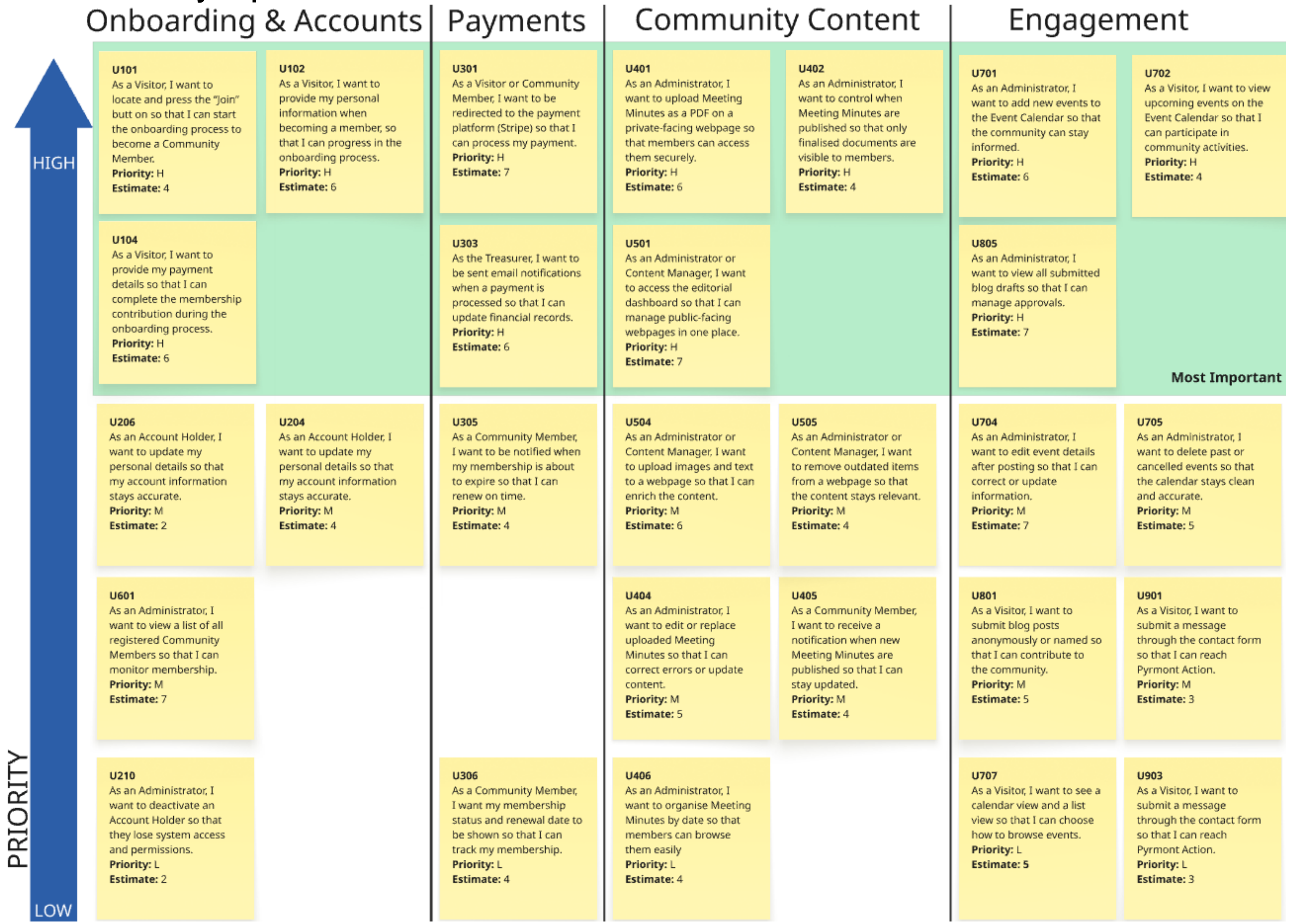
Use Case ID	UC801: Blog Submission and Approval
User Story	As a Visitor, I want to submit a blog post so that I can contribute to the community.
Goal	The Visitor submits a blog draft which the Administrator approves or rejects.
Priority	Medium
Actors	Primary Actor: Visitor Secondary Actors: Administrator, Database
Pre-conditions	The Visitor is logged in or provides name/email.
Post-conditions	The draft is saved and sent to the Administrator for approval.
Trigger	The Visitor clicks “Submit Blog”.
Main Flow	1. The Visitor writes the blog draft. 2. The Visitor clicks Submit. 3. The system saves the draft and notifies the Administrator. 4. The Administrator reviews the draft. 5. The Administrator approves and publishes it. 6. The use case ends.
Exceptions	E1. Draft content is empty. E2. File upload fails.
Includes	UC806: Reject Posts
Supporting Information	Content should meet the community guidelines.
Non-functional Requirements	Performance: Autosaves every 30 seconds. Security: Only admins can publish posts.

Alternate Flow 1	“Rejected Draft” Trigger The Administrator rejects the draft. Step The system notifies the Visitor of the rejection. Alt Flow The Visitor can edit and resubmit. Post conditions The updated draft is stored again. Exceptions E1. Step 4 – Rejection message not delivered.
Alternate Flow 2	“Anonymous Draft” Trigger The Visitor opts to submit anonymously. Step The system removes identifying details. Alt Flow The Administrator reviews and publishes as anonymous. Post conditions The post appears without author details. Exceptions E1. Step 2 – Name not stripped properly.

Use Case ID	UC901: Contact Us Form Email Integration
User Story	As a Visitor, I want to send a message so that I can contact the organisation.
Goal	The Visitor submits the contact form and the message is emailed to the organisation.
Priority	Medium
Actors	Primary Actor: Visitor Secondary Actors: Enquiry Email Holder
Pre-conditions	The contact form is available online.
Post-conditions	The enquiry email is delivered.
Trigger	The Visitor clicks “Send” on the contact form.
Main Flow	1. The Visitor enters their name, email, subject and message. 2. The Visitor clicks Send. 3. The system validates the fields. 4. The system sends the email. 5. The enquiry email holder receives the message. 6. The use case ends.
Exceptions	E1. Invalid email format. E2. Email server unavailable.

Includes	UC902: Email Delivery
Supporting Information	The message must include sender name, email, subject, and body.
Non-functional Requirements	Performance: Delivered within 1 minute. Security: TLS-encrypted SMTP.
Alternate Flow 1	“Invalid Email” Trigger The Visitor enters an invalid email address. Step The system shows an error message. Alt Flow The Visitor corrects it and resubmits. Post conditions The form submits successfully. Exceptions E1. Step 3 – Validation fails silently.
Alternate Flow 2	“Email Server Down” Trigger The mail server is down. Step The system saves the message locally. Alt Flow The system retries sending later. Post conditions The message eventually sends. Exceptions E1. Step 4 – Retry queue is full and drops the message.

5. User Story Map



Onboarding & Accounts

U101
As a Visitor, I want to locate and press the "Join" butt on so that I can start the onboarding process to become a Community Member.
Priority: H
Estimate: 4

U102
As a Visitor, I want to provide my personal information when becoming a member, so that I can progress in the onboarding process.
Priority: H
Estimate: 6

U104
As a Visitor, I want to provide my payment details so that I can complete the membership contribution during the onboarding process.
Priority: H
Estimate: 6

U206
As an Account Holder, I want to update my personal details so that my account information stays accurate.
Priority: M
Estimate: 2

U204
As an Account Holder, I want to update my personal details so that my account information stays accurate.
Priority: M
Estimate: 4

U601
As an Administrator, I want to view a list of all registered Community Members so that I can monitor membership.
Priority: M
Estimate: 7

U210
As an Administrator, I want to deactivate an Account Holder so that they lose system access and permissions.
Priority: L
Estimate: 2

Payments

U301
As a Visitor or Community Member, I want to be redirected to the payment platform (Stripe) so that I can process my payment.
Priority: H
Estimate: 7

U303
As the Treasurer, I want to be sent email notifications when a payment is processed so that I can update financial records.
Priority: H
Estimate: 6

U305
As a Community Member, I want to be notified when my membership is about to expire so that I can renew on time.
Priority: M
Estimate: 4

U306
As a Community Member, I want my membership status and renewal date to be shown so that I can track my membership.
Priority: L
Estimate: 4

Community Content

U401
As an Administrator, I want to upload Meeting Minutes as a PDF on a private-facing webpage so that members can access them securely.
Priority: H
Estimate: 6

U402
As an Administrator, I want to control when Meeting Minutes are published so that only finalised documents are visible to members.
Priority: H
Estimate: 4

U501
As an Administrator or Content Manager, I want to access the editorial dashboard so that I can manage public-facing webpages in one place.
Priority: H
Estimate: 7

U504
As an Administrator or Content Manager, I want to upload images and text to a webpage so that I can enrich the content.
Priority: M
Estimate: 6

U505
As an Administrator or Content Manager, I want to remove outdated items from a webpage so that the content stays relevant.
Priority: M
Estimate: 4

U404
As an Administrator, I want to edit or replace uploaded Meeting Minutes so that I can correct errors or update content.
Priority: M
Estimate: 5

U405
As a Community Member, I want to receive a notification when new Meeting Minutes are published so that I can stay updated.
Priority: M
Estimate: 4

U406
As an Administrator, I want to organise Meeting Minutes by date so that members can browse them easily
Priority: L
Estimate: 4

Engagement

U701
As an Administrator, I want to add new events to the Event Calendar so that the community can stay informed.
Priority: H
Estimate: 6

U702
As a Visitor, I want to view upcoming events on the Event Calendar so that I can participate in community activities.
Priority: H
Estimate: 4

U805
As an Administrator, I want to view all submitted blog drafts so that I can manage approvals.
Priority: H
Estimate: 7

U704
As an Administrator, I want to edit event details after posting so that I can correct or update information.
Priority: M
Estimate: 7

U705
As an Administrator, I want to delete past or cancelled events so that the calendar stays clean and accurate.
Priority: M
Estimate: 5

U801
As a Visitor, I want to submit blog posts anonymously or named so that I can contribute to the community.
Priority: M
Estimate: 5

U901
As a Visitor, I want to submit a message through the contact form so that I can reach Pyrmont Action.
Priority: M
Estimate: 3

U707
As a Visitor, I want to see a calendar view and a list view so that I can choose how to browse events.
Priority: L
Estimate: 5

U903
As a Visitor, I want to submit a message through the contact form so that I can reach Pyrmont Action.
Priority: L
Estimate: 3