SC2006 – Software Engineering

Lab 1 Deliverables

Lab Group	SCSC
Team	HobbyHive
Members	Teo Rong Xuan (U2421554F)
	Jain Divisha (U2423825L)
	Chua Jing Yi Jax (U2421642A)
	Ishita Dhananjaya (U2420909L)
	Palagiri Afreen Mahtaj (U2422632F)
	Swetha Sudhakar (U2420696G)

Table Of Contents

1. Documentation of Functional and Non-Functional Requirements

A. Functional Requirements

1. HobbyHive shall provide functionality for **Users** to create accounts and securely sign in and access the HobbyHive platform.

1.1. HobbyHive shall allow **Members** to create an account.

1.1.1. HobbyHive shall allow Members to input their username, address, password, email, and profile picture.

1.1.2. HobbyHive shall create an account with the "Member" role with information inputted by the Member.

1.1.3. HobbyHive shall verify that the email address is not already associated with an existing account.

1.1.4. Upon successful account creation, HobbyHive shall assign a unique User ID to the account.

1.2. HobbyHive shall allow **Admins** to create an account.

1.2.1. HobbyHive shall allow Admins to input their email and password.

1.2.2. HobbyHive shall create an account with the "Admin" role with information inputted by the Admin.

1.2.3. HobbyHive shall verify that the email address is not already associated with an existing account.

1.2.4. Upon successful account creation, HobbyHive shall assign a unique User ID to the account.

1.3. HobbyHive shall allow Users to modify their account information.

1.3.1. HobbyHive shall allow Users to update their username, address, email, password, and profile picture.

1.3.2. HobbyHive shall verify that the updated email is not already associated with another account.

1.3.3. HobbyHive shall require Users to enter their current password to authorise account modifications.

1.3.4. Upon successful modification, HobbyHive shall confirm the changes to the User via on-screen confirmation.

1.4. HobbyHive shall allow Users to log in to the HobbyHive platform using their previously registered account.

1.4.1. HobbyHive shall allow Users to enter their email and password to access the HobbyHive platform.

1.4.2. HobbyHive shall mask the password input by default.

1.4.3. HobbyHive shall allow Users to unmask their password input if desired.

1.4.4. If the email or password does not match, HobbyHive shall display an error message: “Email and password do not match.”

1.4.5. If email and password match, HobbyHive shall grant access to the dashboard.

1.4.5.1. If the User is a **Non-admin (Member)**, HobbyHive shall display the User Dashboard, which provides access to features for Participants and Hosts.

1.4.5.2. If the User is an Admin, HobbyHive shall display the Admin Dashboard, which provides access to administrative functions.

2. HobbyHive shall allow **Participants** to perform Participant-specific tasks.
   - 2.1. HobbyHive shall allow Participants to search for events using a search bar.
     - 2.1.1. HobbyHive shall allow Participants to filter search results by category.
       - 2.1.1.1. If the Participant chooses a category, it shall be selected from a predefined list of valid categories.
     - 2.1.2. HobbyHive shall allow Participants to filter search results by date.
       - 2.1.2.1. If the Participant chooses a date, a valid calendar date in YYYY-MM-DD format shall be selected.
     - 2.1.3. HobbyHive shall allow Participants to filter search results by location.
       - 2.1.3.1. HobbyHive shall allow users to set a centre location via their address and define a radius in kilometres.
       - 2.1.3.2. HobbyHive shall display only events located within the specified circular boundary.
     - 2.1.4. HobbyHive shall allow Participants to filter events by keyword.
       - 2.1.4.1. Keyword data shall be text of length greater than 0 and less than 256 characters.
   - 2.2. HobbyHive shall allow Participants to join multiple groups. Joining a group allows a Participant to access the group chat and updates, while RSVP confirms their attendance for the event.
     - 2.2.1. If the Participant wants to join a certain group, they shall do so by pressing the “Join” button.
       - 2.2.1.1. HobbyHive shall send a request to the Host of that event for approval.
       - 2.2.1.2. If the Host approves the Participant’s request, HobbyHive shall admit the Participant into the Group chat created by the Host.
       - 2.2.1.3. If the Host rejects the Participant’s request, HobbyHive shall notify the Participant that the group is full.
     - 2.2.2. Upon joining the group, if the participant is interested to attend the event, the participant shall RSVP by pressing the RSVP button.
     - 2.2.3. HobbyHive shall notify all participants of the event if any changes are made to the event details.
       - 2.2.3.1. If the Participant agrees to still go, they shall RSVP by pressing the ‘RSVP’ button again.
       - 2.2.3.2. If the participant presses the ‘leave’ button, HobbyHive shall remove the participant from the group chat.
   - 2.3. HobbyHive shall recommend events to users based on their preferences and contextual factors.
     - 2.3.1. HobbyHive shall recommend events aligned with the categories or hobbies the user has expressed interest in.
     - 2.3.2. HobbyHive shall prioritize events with higher popularity.
     - 2.3.3. HobbyHive shall recommend events that are geographically closer to the user’s preferred or current location.
   - 2.4. HobbyHive shall allow Participants to follow other Users.
     - 2.4.1. Participants shall be able to follow other Participants or Hosts by pressing the “Follow” button on their profile.

2.4.2. HobbyHive shall notify the user when they gain a new follower.

2.4.3. Participants shall be able to unfollow users at any time.

2.4.4. Following a user shall allow Participants to view their upcoming events or activities more prominently in recommendations.

3. HobbyHive shall allow **Hosts** to perform Host-specific tasks.

3.1. HobbyHive shall allow Hosts to create events.

3.1.1. HobbyHive shall allow Hosts to input basic event details.

3.1.1.1. HobbyHive shall allow Hosts to enter the event title (e.g. “Badminton at 6PM”).

3.1.1.2. HobbyHive shall allow Hosts to enter descriptions of the event they are hosting.

3.1.1.3. HobbyHive shall allow Hosts to select a category for the event (e.g. Sports, Music, Arts).

3.1.2. HobbyHive shall allow Hosts to set event scheduling information.

3.1.2.1. HobbyHive shall allow Hosts to select the event date.

3.1.2.2. HobbyHive shall allow Hosts to select start and end time.

3.1.3. HobbyHive shall allow Hosts to specify event location.

3.1.3.1. HobbyHive shall allow Hosts to enter a venue using postal code or address.

3.1.3.2. HobbyHive shall allow Hosts to select a location using a map pin.

3.1.4. HobbyHive shall allow Hosts to set participation details.

3.1.4.1. HobbyHive shall allow Hosts to set the maximum number of participants.

3.1.4.2. HobbyHive shall allow Hosts to set the type of joining rule (e.g. Automatic or Manual Approval).

3.1.5. HobbyHive shall allow Hosts to enhance events.

3.1.5.1. HobbyHive shall allow Hosts to upload an image/banner for the event.

3.1.5.2. HobbyHive shall allow Hosts to add notes (e.g. “bring your own racket” or “materials provided”).

3.2. HobbyHive shall allow Hosts to manage participants.

3.2.1. HobbyHive shall allow Hosts to review participant join requests.

3.2.2. HobbyHive shall allow Hosts to approve or reject participant join requests.

3.2.3. HobbyHive shall allow Hosts to cancel a participant’s RSVP if necessary.

3.2.4. HobbyHive shall allow Hosts to view a list of confirmed and pending participants.

3.3. HobbyHive shall allow Hosts to edit or cancel events.

3.3.1. HobbyHive shall allow Hosts to update event details (title, description, category, date, time, location, capacity).

3.3.2. HobbyHive shall notify all participants if event details are updated.

3.3.3. HobbyHive shall allow Hosts to cancel events.

3.3.4. HobbyHive shall notify all participants if an event is cancelled.

3.4. HobbyHive shall allow Hosts to follow other Users.

3.4.1. Hosts shall be able to follow other Hosts or Participants by pressing the “Follow” button on their profile.

3.4.2. HobbyHive shall notify the user when they gain a new follower.

3.4.3. Hosts shall be able to unfollow users at any time.

3.4.4. Following a user shall allow Hosts to engage with frequent participants or collaborators more easily.

4. HobbyHive shall allow **Admins** to perform administrative tasks.

4.1. If there is misconduct or violation of community guidelines, HobbyHive shall allow admins to remove these users from the platform.

4.1.1. If a user repeatedly posts offensive or inappropriate content in group chats or event boards, HobbyHive shall allow Admins to remove the user.

4.1.2. If a user consistently fails to attend events they RSVP for (3 or more times), HobbyHive shall allow Admins to remove the user.

4.1.3. If a user creates multiple fake accounts, HobbyHive shall allow Admins to remove the accounts.

4.1.4. If a user harasses or spams other users, HobbyHive shall allow Admins to remove the user.

4.2. When a user submits requests to create a new group, HobbyHive shall allow Admins to review and approve/reject the requests.

4.2.1. If the submitted group contains insufficient description or unclear purpose, HobbyHive shall allow Admins to request more information before approval.

4.2.2. If the submitted group contains offensive or unrelated content, HobbyHive shall allow Admins to reject the group creation request.

4.2.3. If the submitted group contains sufficient and appropriate information, HobbyHive shall allow Admins to approve the group creation.

4.3. When inappropriate events are created, HobbyHive shall allow Admins to moderate them.

4.3.1. If an event promotes illegal activities, HobbyHive shall allow Admins to remove the event.

4.3.2. If an event contains misleading details or false scheduling, HobbyHive shall allow Admins to remove the event.

4.3.3. If an event is reported multiple times by users, HobbyHive shall allow Admins to review and decide on its removal.

5. HobbyHive shall create and manage **group chats** for events with the Host and Participants who join the event.

5.1. HobbyHive shall create group chats automatically when an event is successfully created by the Host.

5.1.1. Each group chat shall be uniquely associated with its respective event.

5.2. HobbyHive shall manage group chat membership automatically.

5.2.1. Participants shall be added to the group chat once they have joined.

5.2.2. Participants shall be removed from the group chat if they press the “Leave” button or if their RSVP is cancelled by the Host.

5.3. HobbyHive shall allow communication between the Host and Participants within the group chat.

5.3.1. Participants and the Host shall be able to send and receive messages.

5.3.2. HobbyHive shall notify all group chat members if event details are updated (e.g., date, time, or location).

5.4. HobbyHive shall close group chats when an event is cancelled or after the event expiry period.

5.4.1. All Participants shall be removed from the chat upon cancellation.

5.4.2. HobbyHive shall automatically close the group chat one month after the event has occurred.

B. Non Functional Requirements

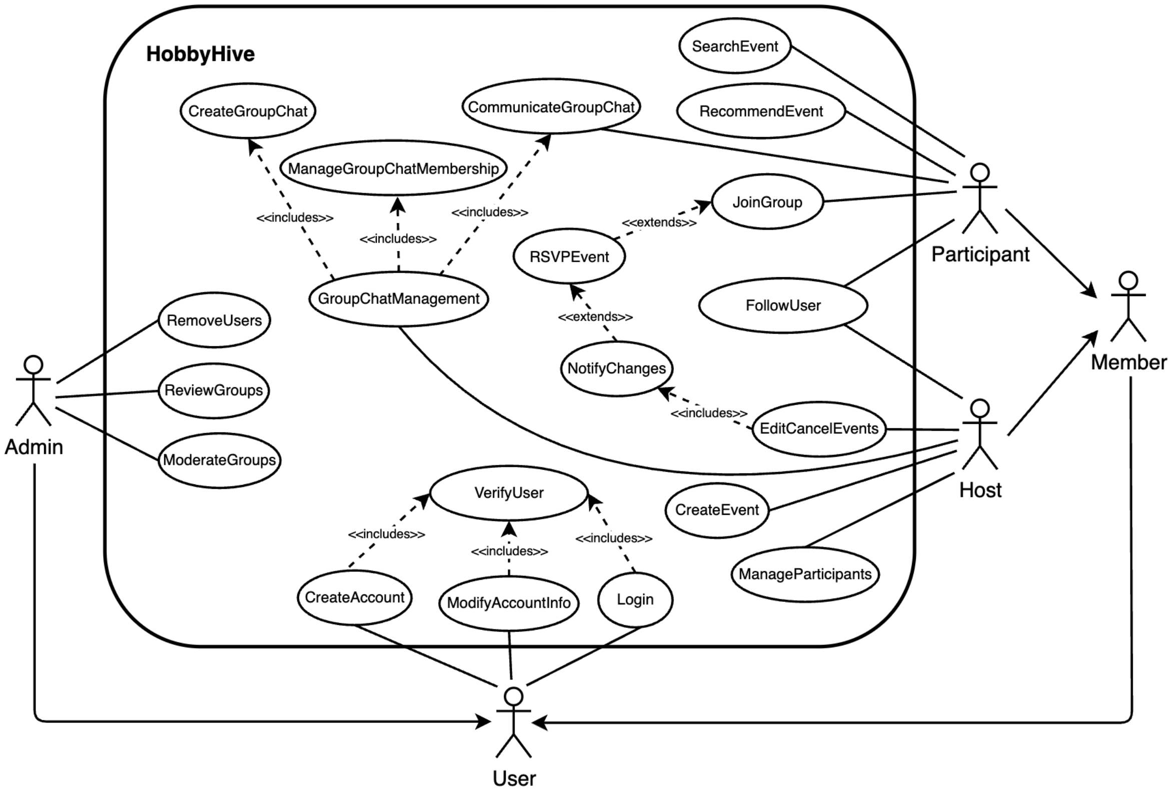
Usability	Easy Sign Up ● The sign up process should be simple, taking no more than 2 minutes under normal operating conditions.
	Mobile Responsive ● All content must be visible on any mobile device, regardless of screen size or resolution. ● Navigation menus and buttons should remain accessible and functional on both portrait and landscape modes.
	Responsive User Interface ● Pages and interactive elements should respond to user actions (clicks, searches, filters) within three seconds under normal operating conditions. ● Users should be able to navigate between sections of the site without unnecessary delays or page reloads.
Reliability	Website Availability ● The website should maintain minimal downtime, targeting at least 95% uptime.
	Recovery ● If there is a problem, the website should recover within 10 min without losing user information.
Performance	Page Load Time ● Pages and listings should load within 2 seconds under normal network conditions.
	Search and Filter ● Searching and filtering hobby listings should return results within 3 seconds.
Supportability	The database must be replaceable with any commercial product supporting standard SQL queries.

2. Data Dictionary

Term	Definition
Account	The credentials and profile information that identify a user within the application.
Admin	A user with elevated privileges who manages the application and performs administrative tasks.
Category	A label that classifies events based on its type or theme (eg. “Sports” or “Tech”).
Description	A textual field that provides additional details or context about a group or event.
Event	Activity or gathering organized within the web app by Organizer(s). Events have a title, description, date and time, location, category, and tags.
Filter	A set of conditions used to narrow down or sort content, such as events, groups, or users.
Group chat	A shared conversation within a group which allows users to send , view and receive messages.
Host	The individual, organization, or system responsible for creating, organizing, or managing an event, session, or activity. The host typically controls scheduling, and event descriptions.
Location	A place associated with an event, which can be physical or virtual.
Member	A participant or host.
Membership role	A user’s position/role in a group such as host or participant.
Participant	An individual, group, or entity that might join/has joined a group created by the host.
RSVP	A user’s indication of whether they plan to attend an event.
RSVP Status	A record of a user’s RSVP-ed events.
Tag	Keywords that summarize events to aid in search and discovery. The tags provide additional context for what an event is about or might entail.
Text message	A communication sent by a user to another user or a group.
User	The person using the Hobbyhive site after creating an account. A User may be an Admin, Participant or Host.

3. Initial Use Case Model (Use Case Diagram and Use Case Descriptions)

A. Use Case Diagram



B. Use Case Descriptions

I. For Functional Requirement #1 (Login)

I.I CreateAccount

Use Case ID:	#1-1		
Use Case Name:	CreateAccount		
Created By:	Jain Divisha	Last Updated By:	Jain Divisha
Date Created:	7 September 2025	Date Last Updated:	7 September 2025

Actor:	User
Description:	Allows a person to create an account with HobbyHive and be registered as a User.
Pre-Conditions:	None
Post-Conditions	An account with the details provided by the User is created.
Priority:	High
Frequency Of Use:	High
Flow Of Events:	1. The system prompts the User to input particular details like role, username, password, email and profile picture. 2. The User enters the required information and clicks on “Create Account” to confirm details. 3. The system includes the VerifyUser use case to validate the entered information (check for unique email/username and proper format). 4. An account with the role and other provided details is created for the User.
Alternative Flows:	**AF-S2a**: If User inputs username that is being used by another account already 1. The system displays a message “Username is already taken. Please enter a new username.” 2. The system returns to step 2. **AF-S2b**: If User inputs email that is linked to a pre-existing account

	1. The system displays a message “Email is already linked to an account” 2. The system returns to step 2.
Exceptions:	If any required information is missing or invalid, an error message “Please enter all valid details” is displayed.
Includes:	VerifyUser
Extends:	None
Special Requirements:	The system must validate all input data and store it securely.
Assumptions:	None
Notes and Issues:	None

I.II ModifyAccountInfo

Use Case ID:	#1-2		
Use Case Name:	ModifyAccountInfo		
Created By:	Jain Divisha	Last Updated By:	Jain Divisha
Date Created:	7 September 2025	Date Last Updated:	7 September 2025

Actor:	User
Description:	Allows User to change details linked to their account.
Pre-Conditions:	The User must have an account with HobbyHive and must be logged into the account.
Post-Conditions	The account details related to the User are modified.
Priority:	Medium
Frequency Of Use:	Medium
Flow Of Events:	1. The User opens the Account Menu and clicks on **Modify Account Info**. 2. The system redirects the User to a new page with options **Change Password**, **Change Email**, **Change Username**, **Change Profile Picture**. 3. If User clicks on **Change Password** 3.1. The system prompts User to enter **New Password** and **Confirm New Password**. This information is displayed as dots which can be unmasked by the user by clicking on an eye icon in the text bar. 3.2. User inputs required details and clicks on Confirm Changes. 3.3. The system includes the VerifyUser use case to validate matching fields and confirm the current password before saving changes. 3.4. The system updates the User's account information.
Alternative Flows:	**AF-S3a**: If User clicks on **Change Email** 1. The system prompts User to enter New Email and Confirm New Email. 2. User inputs required details and clicks on Confirm Changes.

	3. The system includes the VerifyUser use case to check if the email is not already associated with another account. 4. The system updates the User’s account information. **AF-S3b**: If User clicks on **Change Username** 1. The system prompts User to enter New Username and Confirm New Username. 2. User inputs required details and clicks on Confirm Changes. 3. The system includes the VerifyUser use case to check if the username is not already associated with another account. 4. The system updates the User’s account information. **AF-S3c**: If User clicks on **Change Profile Picture** 1. The system prompts User to select a new picture. 2. The User selects a picture with appropriate dimensions. 3. The system updates the User’s account information
Exceptions:	If the data provided by the User in both the fields does not match, the system issues an error message “Mismatched Data! Please ensure correct entry.”
Includes:	VerifyUser
Extends:	None
Special Requirements:	The system must verify data if both fields match and only then modify the Account Information.
Assumptions:	The User is already logged in.
Notes and Issues:	None

I.III Login

Use Case ID:	#1-3		
Use Case Name:	Login		
Created By:	Jain Divisha	Last Updated By:	Jain Divisha
Date Created:	7 September 2025	Date Last Updated:	7 September 2025

Actor:	User
Description:	Allows User to login to and access HobbyHive with their Account Information.
Pre-Conditions:	1. User has an account with HobbyHive that has been made through the CreateAccount use case. 2. User is logged out of the system.
Post-Conditions	The User is logged into and can access HobbyHive.
Priority:	High
Frequency Of Use:	High
Flow Of Events:	1. The system prompts User to enter email/username and password. 2. The User provides the information. The password is masked as dots, but the User can choose to unmask it by clicking on the eye icon in the text bar. 3. The User clicks on Login. 4. The system includes the VerifyUser use case to verify the provided credentials against the database. 5. If credentials are correct, access is granted.
Alternative Flows:	**AF-S4**: If upon verification, the information does not match with any entry 1. The system outputs a message “This account does not exist. Please try again!” for 3 seconds. 2. The system returns to step 1
Exceptions:	If any required information is missing, an error message is displayed.
Includes:	VerifyUser

Extends:	None
Special Requirements:	System must validate user input data
Assumptions:	The User has an account with HobbyHive.
Notes and Issues:	None

I.IV VerifyUser

Use Case ID:	#1-4		
Use Case Name:	VerifyUser		
Created By:	Jain Divisha	Last Updated By:	Jain Divisha
Date Created:	7 September 2025	Date Last Updated:	7 September 2025

Actor:	System (triggered during user interactions in CreateAccount, ModifyAccountInfo and Login)
Description:	Ensures the user-provided information (email, username, password, or other credentials) is valid, unique (when required), and matches the system records before proceeding with account creation, modification, or login.
Pre-Conditions:	1. User is in the process of creating, modifying, or accessing their account. 2. Relevant input fields have been provided by the user.
Post-Conditions	1. User data is verified for correctness, validity, and uniqueness (as applicable). 2. If verification fails, an appropriate error is displayed.
Priority:	High
Frequency Of Use:	High
Flow Of Events:	1. The system receives input data (email, username, password, etc.) from the primary use case. 2. The system checks for mandatory fields (none should be empty). 3. System validates data format: 3.1. Email is in valid format (or) Username follows allowed character rules. 3.2. Password meets security criteria (if relevant). 4. If invoked during CreateAccount: 4.1. The system checks that email and username is not already in use. 5. The system returns verification result (success or specific error)

Alternative Flows:	**AF-S1**: Missing mandatory fields 1. The system displays an error message. 2. It returns to the data entry step in the calling use case. **AF-S2**: Invalid email format 1. The system displays an error message 2. It returns to the data entry step in the calling use case. **AF-S4a**: If invoked during Login: 1. The system compares provided credentials with stored records. 2. Moves to step 5. **AF-S4b**: If invoked during ModifyAccountInfo: 1. The system checks for conflicts (e.g., email already in use, username already in use) and verifies if the current password is the same as the new password before applying changes. 2. Moves to step 5
Exceptions:	None
Includes:	None
Extends:	None
Special Requirements:	All passwords must be hashed. Verification must be completed in under 3 seconds.
Assumptions:	The database of users is always up to date and available for verification queries.
Notes and Issues:	Future versions can include multi-factor authentication (MFA).

II. For Functional Requirement #2 (Participant)

II.I SearchEvent

Use Case ID:	#2-1		
Use Case Name:	SearchEvent		
Created By:	Afreen Mahtaj	Last Updated By:	Afreen Mahtaj
Date Created:	8 September 2025	Date Last Updated:	8 September 2025

Actor:	Participant
Description:	Allows a Participant to search for events using keywords, categories, dates, and locations. The system ensures valid inputs and displays only matching events.
Pre-Conditions:	1. Participant must be logged in. 2. Events must exist in the database.
Post-Conditions	1. A filtered list of events is displayed to the Participant.
Priority:	High
Frequency Of Use:	High
Flow Of Events:	1. Participant navigates to the Search bar. 2. System prompts Participant to enter filters (keyword, category, date, location). 3. Participant enters one or more search criteria. 3.1. Keyword: text of length 1–255. 3.2. Category: selected from a predefined list. 3.3. Date: valid date in YYYY-MM-DD format. 3.4. Location: set by address and radius in km. 4. Participant clicks “Search.” 5. System validates inputs. 6. System retrieves and displays all events matching criteria within the specified circular boundary.
Alternative Flows:	**AF-S1**: If no results match, the system displays “No events found. Try adjusting filters.”
Exceptions:	1. Invalid category selection → system shows error “Invalid category.”

	2. Invalid date format → system shows error "Please enter date in YYYY-MM-DD format." 3. Empty keyword or >255 characters → system shows error "Keyword must be 1–255 characters."
Includes:	None
Extends:	None
Special Requirements:	1. Search results must be displayed within 2 seconds. 2. Location-based search must calculate distance using defined radius.
Assumptions:	1. The event database is indexed and supports filtering efficiently.
Notes and Issues:	None

II.II JoinGroup

Use Case ID:	#2-2		
Use Case Name:	JoinGroup		
Created By:	Afreen Mahtaj	Last Updated By:	Afreen Mahtaj
Date Created:	8 September 2025	Date Last Updated:	8 September 2025

Actor:	Participant
Description:	Allows a Participant to join a group linked to an event. The Host approves or rejects the request. Joining grants access to group chat and updates.
Pre-Conditions:	1. Participant must be logged in. 2. Event group must exist.
Post-Conditions	1. Participant becomes a Member of the group if approved by Host. 2. Participant gains access to group chat.
Priority:	High
Frequency Of Use:	Medium
Flow Of Events:	1. Participant views event details. 2. Participant clicks “Join” button. 3. System sends join request to the Host. 4. Host reviews request: 4.1. If approved, the system admits Participant into group chat. 4.2. If rejected, the system notifies Participant: “The group is full.”
Alternative Flows:	**AF-S1**: If the Participant withdraws request before Host response, the system cancels join request.
Exceptions:	Group does not exist, system shows “This group is unavailable.”
Includes:	GroupChatManagement
Extends:	RSVPEvent
Special Requirements:	1. Host must be notified instantly of join requests. 2. Participants must be notified of

	approval/rejection.
Assumptions:	1. Groups may have maximum capacity set by Host.
Notes and Issues:	None

II.III RSVPEvent

Use Case ID:	#2-3		
Use Case Name:	RSVPEvent		
Created By:	Afreen Mahtaj	Last Updated By:	Afreen Mahtaj
Date Created:	8 September 2025	Date Last Updated:	8 September 2025

Actor:	Participant
Description:	Allows a Participant to confirm attendance for an event after joining its group. Participants can update their RSVP in response to changes.
Pre-Conditions:	1. Participant must be a Member of the event's group. 2. Event must exist.
Post-Conditions	1. RSVP status is recorded in system and visible to Host. 2. Notifications of RSVP changes are sent to all event participants.
Priority:	High
Frequency Of Use:	Medium
Flow Of Events:	1. Participant opens event details. 2. Participant clicks "RSVP" button (options: Going / Not Going). 3. System records RSVP status. 4. System confirms: "Your RSVP has been recorded." 5. If event details change, system notifies all participants. 6. Participant re-confirms RSVP status if still attending.
Alternative Flows:	**AF-S1**: Participant presses "Leave" button, system removes them from group chat.
Exceptions:	1. RSVP deadline has passed → system shows "RSVP closed."
Includes:	NotifyChanges

Extends:	JoinGroup
Special Requirements:	1. Notifications must be real-time.
Assumptions:	1. All participants receive event updates instantly.
Notes and Issues:	None

II.IV RecommendEvent

Use Case ID:	#2-4		
Use Case Name:	RecommendEvent		
Created By:	Afreen Mahtaj	Last Updated By:	Afreen Mahtaj
Date Created:	8 September 2025	Date Last Updated:	8 September 2025

Actor:	Participant
Description:	System recommends events to a Participant based on preferences (categories, hobbies), popularity, and proximity.
Pre-Conditions:	1. Participant must be logged in. 2. Event database must exist.
Post-Conditions	1. Recommended events are displayed on Participant's homepage.
Priority:	Medium
Frequency Of Use:	Medium
Flow Of Events:	1. Participant navigates to "Recommended Events." 2. System retrieves Participant's preferences (categories, hobbies, followed users). 3. System identifies events based on: 3.1. Matching categories/hobbies. 3.2. Event popularity. 3.3. Location proximity. 4. System displays recommended events in a ranked list.
Alternative Flows:	**AF-S1**: If no personalized matches exist, the system displays popular nearby events.
Exceptions:	If the system fails to fetch user preferences, fallback to general recommendations.
Includes:	None
Extends:	None
Special Requirements:	1. Recommendations should refresh daily. 2. Distance calculations must be accurate to within

	1 km.
Assumptions:	1. User preference data is up to date.
Notes and Issues:	None

II.V FollowUser

Use Case ID:	#2-5		
Use Case Name:	FollowUser		
Created By:	Afreen Mahtaj	Last Updated By:	Afreen Mahtaj
Date Created:	8 September 2025	Date Last Updated:	8 September 2025

Actor:	Participant
Description:	Allows a Participant to follow another User (Host or Participant) to see their updates and events more prominently in recommendations.
Pre-Conditions:	1. Participant must be logged in. 2. The other User exists in system.
Post-Conditions	1. Followed User is added to Participant's "Following" list. 2. Participant receives updates about followed User's activities.
Priority:	Medium
Frequency Of Use:	Medium
Flow Of Events:	1. Participant visits another User's profile. 2. Participant clicks "Follow" button. 3. System adds User to "Following" list. 4. System confirms: "You are now following this user." 5. System notifies the followed User: "You have a new follower."
Alternative Flows:	**AF-S1**: If Participant clicks "Unfollow," system removes User from following list.
Exceptions:	1. If target User no longer exists, system displays: "This user is unavailable."
Includes:	None
Extends:	None
Special Requirements:	1. Notifications must be sent for new follows. 2. Unfollow should remove all related

	notifications/recommendations.
Assumptions:	1. All Users allow being followed unless the account is deleted.
Notes and Issues:	None

III. For Functional Requirement #3 (Host)

III.I CreateEvent

Use Case ID:	#3-1		
Use Case Name:	CreateEvent		
Created By:	Swetha Sudhakar	Last Updated By:	Swetha Sudhakar
Date Created:	8 September 2025	Date Last Updated:	8 September 2025

Actor:	Host
Description:	Allows Hosts to create an event on HobbyHive by providing event details such as title, description, category, date, time, and location.
Pre-Conditions:	The Host must be logged in.
Post-Conditions	A new event is created and visible to Participants for joining.
Priority:	High
Frequency Of Use:	Medium–High
Flow Of Events:	1. Host clicks on "Create Event. 2. The system prompts the Host to input event details (title, description, category, date, time, location, capacity). 3. Host enters all required details and uploads an optional image/banner. 4. Host confirms creation. 5. The system validates details and creates the event. 6. The system generates a group chat for the event automatically.
Alternative Flows:	**AF-S1:** If mandatory details are missing, the system shows “Please fill in all required fields.” **AF-S2:** If invalid date/time is entered, the system shows “Invalid scheduling information.”

Exceptions:	Database error → “Event could not be created. Please try again later.”
Includes:	VerifyUser
Extends:	None
Special Requirements:	Events must have a unique title and valid date/time.
Assumptions:	Hosts have stable network access.
Notes and Issues:	Need to consider timezone handling for events.

III.II ManageParticipants

Use Case ID:	#3-2		
Use Case Name:	ManageParticipants		
Created By:	Swetha Sudhakar	Last Updated By:	Swetha Sudhakar
Date Created:	8 September 2025	Date Last Updated:	8 September 2025

Actor:	Host
Description:	Allows Hosts to manage event participants by approving/rejecting requests, cancelling RSVPs, and viewing participant lists.
Pre-Conditions:	Host must be logged in and have at least one active event.
Post-Conditions	Participant statuses are updated based on Host's actions.
Priority:	High
Frequency Of Use:	Medium
Flow Of Events:	1. Host selects their event from the dashboard. 2. Host opens the Participants tab. 3. Host reviews join requests. 4. Host approves/rejects participants. 5. Host can cancel RSVPs if necessary. 6. System updates the participant list and sends notifications.
Alternative Flows:	**AF-S1:** If participant limit is reached, the system shows "Maximum participants reached." **AF-S2:** If Host rejects, system notifies the participant "Request denied."
Exceptions:	Connection failure during approval → "Unable to process request. Try again later."
Includes:	NotifyChanges
Extends:	None

Special Requirements:	Notifications must be sent in real time.
Assumptions:	Hosts act fairly and follow community guidelines.
Notes and Issues:	Consider escalation to Admin if Host repeatedly removes participants unfairly.

III.III EditCancelEvents

Use Case ID:	#3-3		
Use Case Name:	EditCancelEvents		
Created By:	Swetha Sudhakar	Last Updated By:	Swetha Sudhakar
Date Created:	8 September 2025	Date Last Updated:	8 September 2025

Actor:	Host
Description:	Allows Hosts to edit or cancel existing events.
Pre-Conditions:	Host must have an active event.
Post-Conditions	Event details are updated OR event is cancelled. Participants are notified.
Priority:	High
Frequency Of Use:	Medium
Flow Of Events:	1. Host selects an event. 2. Host clicks Edit or Cancel. 3. If Edit, the system shows editable fields. 4. Host updates fields and confirms. 5. If Cancel, Host confirms cancellation. 6. System applies changes and notifies participants.
Alternative Flows:	**AF-S1:** If edited event date is in the past, system shows “Invalid event date.”
Exceptions:	Database error during update.
Includes:	NotifyChanges
Extends:	None
Special Requirements:	Event cancellation must automatically close group chats.
Assumptions:	Hosts edit responsibly.
Notes and Issues:	Ensure data consistency when updating/cancelling events.

III.IV NotifyChanges

Use Case ID:	#3-4		
Use Case Name:	NotifyChanges		
Created By:	Swetha Sudhakar	Last Updated By:	Swetha Sudhakar
Date Created:	8 September 2025	Date Last Updated:	8 September 2025

Actor:	System
Description:	Automatically notifies Participants of event updates or cancellations.
Pre-Conditions:	An event must exist and have Participants.
Post-Conditions	Notifications are delivered to all Participants.
Priority:	High
Frequency Of Use:	Medium
Flow Of Events:	1. Host edits or cancels an event. 2. The system detects the change. 3. The system generates notifications. 4. All Participants receive the notification via dashboard and group chat.
Alternative Flows:	**AF-S1:** If Participant has disabled notifications, updates are shown only in dashboard.
Exceptions:	Notification delivery fails due to connectivity issues.
Includes:	None
Extends:	None
Special Requirements:	Notifications must be near real-time (< 1 min).
Assumptions:	Participants check their dashboard/group chat regularly.
Notes and Issues:	Consider adding email alerts as backup.

IV. For Functional Requirement #4 (Admin)

IV.I RemoveUsers

Use Case ID:	#4-1		
Use Case Name:	RemoveUsers		
Created By:	Jain Divisha	Last Updated By:	Jain Divisha
Date Created:	7 September 2025	Date Last Updated:	7 September 2025

Actor:	Admin
Description:	Allows Admin to remove a User by their username for a specified duration.
Pre-Conditions:	Admin is logged into their HobbyHive account.
Post-Conditions	A User account is removed from HobbyHive and the email can not be used to make future accounts.
Priority:	High
Frequency Of Use:	Low
Flow Of Events:	1. Admin inputs the username of the user that the Admin wants to ban. 2. Admin inputs the duration of the ban. 3. Admin confirms the ban by selecting **Confirm**.
Alternative Flows:	None
Exceptions:	If the username is not found, the system shall display an error message and prompt the Admin to input another username.
Includes:	None
Extends:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

IV.II ReviewGroups

Use Case ID:	#4-2		
Use Case Name:	ReviewGroups		
Created By:	Jain Divisha	Last Updated By:	Jain Divisha
Date Created:	7 September 2025	Date Last Updated:	7 September 2025

Actor:	Admin
Description:	Allows Admin to review group creation requests submitted by Users and decide whether to approve, reject, or request more information based on the group's description and purpose.
Pre-Conditions:	1. Admin is logged into their HobbyHive account. 2. One or more group creation requests have been submitted by Users.
Post-Conditions	1. The group creation request is approved, rejected, or set to "pending additional information." 2. The requesting User is notified of the decision.
Priority:	High
Frequency Of Use:	High
Flow Of Events:	1. The system displays a list of pending group creation requests to the Admin. 2. Admin selects a group creation request to review. 3. The system displays the group's details, including its description. 4. Admin reviews the details and chooses one of the following actions: 4.1. **Approve** the request 4.2. **Reject** the request 4.3. **Request more information** from the User 5. If the Admin selects **Approve**, the system updates the group's status accordingly and notifies the User of the decision.
Alternative Flows:	**AF-S5a**: Request more information 1. Admin determines that the group description is

	insufficient or unclear. 2. Admin selects “Request More Information.” 3. The system notifies the User to provide additional details. **AF-S5b**: Reject offensive or unrelated group 1. Admin identifies offensive, inappropriate, or unrelated content in the group description. 2. Admin selects “Reject.” 3. The system notifies the User of rejection and provides a reason.
Exceptions:	None
Includes:	None
Extends:	None
Special Requirements:	1. All group descriptions must be stored securely and remain editable only until Admin takes action. 2. Notifications to Users must be sent in real-time.
Assumptions:	User has submitted the complete initial group creation request before Admin review.
Notes and Issues:	None

IV.III ModerateGroups

Use Case ID:	#4-3		
Use Case Name:	ModerateGroups		
Created By:	Jain Divisha	Last Updated By:	Jain Divisha
Date Created:	7 September 2025	Date Last Updated:	7 September 2025

Actor:	Admin
Description:	Allows Admin to review and moderate inappropriate or problematic groups/events on HobbyHive by removing those that violate guidelines, promote illegal activities, contain misleading information, or receive multiple user reports.
Pre-Conditions:	1. Admin is logged into their HobbyHive account. 2. One or more groups/events are flagged or reported for review.
Post-Conditions	1. The inappropriate group/event is removed, or its status is updated to “Reviewed”. 2. Relevant Users (e.g., group creators or reporters) are notified of the action.
Priority:	High
Frequency Of Use:	Medium
Flow Of Events:	1. The system displays a list of flagged or reported groups/events to the Admin. 2. Admin selects a group/event for moderation. 3. The system shows details including description, schedule, reports, and flagged content. 4. Admin reviews the details and chooses one of the following actions: 4.1. Remove the group/event 4.2. Retain the group/event (mark as reviewed) 5. If the Admin clicks on **Remove**, the system removes the group/event listing from HobbyHive and notifies the creator and, if applicable, the reporting Users.
Alternative Flows:	**AF-S5**: If the admin clicks on **Retain** 1. The system updates the group/event status to

	“Reviewed” and notifies the creator and, if applicable, the reporting Users.
Exceptions:	None
Includes:	None
Extends:	None
Special Requirements:	All moderation actions must be logged with timestamps and Admin’s username for audit purposes.
Assumptions:	Groups/events can be reported by Users or flagged automatically by the system.
Notes and Issues:	None

V. For Functional Requirement #5 (Group Chats)

V.I GroupChatManagement

Use Case ID:	#5-1		
Use Case Name:	GroupChatManagement		
Created By:	Swetha Sudhakar	Last Updated By:	Swetha Sudhakar
Date Created:	8 September 2025	Date Last Updated:	8 September 2025

Actor:	System, Host, Participant
Description:	Handles the lifecycle of event group chats, including creation, membership, and closure.
Pre-Conditions:	An event must exist.
Post-Conditions	Group chat is created, managed, and eventually closed.
Priority:	High
Frequency Of Use:	High
Flow Of Events:	1. When a Host creates an event, the system generates a group chat. 2. Participants are added once they join. 3. Participants are removed when they leave or RSVP is cancelled. 4. Group chat closes after event cancellation or one month post-event.
Alternative Flows:	**AF-S1:** If chat cannot be created, the system retries.
Exceptions:	Chat service failure will show “Group chat unavailable.”
Includes:	CreateGroupChat, ManageGroupChatMembership, CommunicateGroupChat
Extends:	None
Special Requirements:	Chats must support real-time messaging.
Assumptions:	All users have access to messaging features.

Notes and Issues:	Need to consider data retention policy for chat logs.

V.II CreateGroupChat

Use Case ID:	#5-2		
Use Case Name:	CreateGroupChat		
Created By:	Swetha Sudhakar	Last Updated By:	Swetha Sudhakar
Date Created:	8 September 2025	Date Last Updated:	8 September 2025

Actor:	System
Description:	Automatically creates a group chat for each new event.
Pre-Conditions:	A Host successfully creates an event.
Post-Conditions	A group chat linked to the event is created.
Priority:	High
Frequency Of Use:	High
Flow Of Events:	1. Host creates an event. 2. The system generates a unique group chat ID. 3. The chat is linked to the event.
Alternative Flows:	**AF-S1:** If creation fails, retry process.
Exceptions:	Chat service error.
Includes:	None
Extends:	None
Special Requirements:	Group chats must be unique per event.
Assumptions:	Chat system is integrated with event service.
Notes and Issues:	Consider scalability if many events are created.

V.III ManageGroupChatMembership

Use Case ID:	#5-3		
Use Case Name:	ManageGroupChatMembership		
Created By:	Swetha Sudhakar	Last Updated By:	Swetha Sudhakar
Date Created:	8 September 2025	Date Last Updated:	8 September 2025

Actor:	System, Participant, Host
Description:	Controls adding/removing users from group chats based on join/leave actions.
Pre-Conditions:	Event exists and group chat is created.
Post-Conditions	Group chat membership is updated accordingly.
Priority:	High
Frequency Of Use:	High
Flow Of Events:	1. The participant joins events, and the system adds them to group chat. 2. Participant leaves or RSVP cancelled, system removes them.
Alternative Flows:	**AF-S1:** If Host removes a participant, system notifies them.
Exceptions:	Membership sync error which leads to delayed updates.
Includes:	None
Extends:	None
Special Requirements:	Membership must be updated in real time.
Assumptions:	Hosts/Participants follow group chat rules.
Notes and Issues:	Ensure proper logging for auditing.

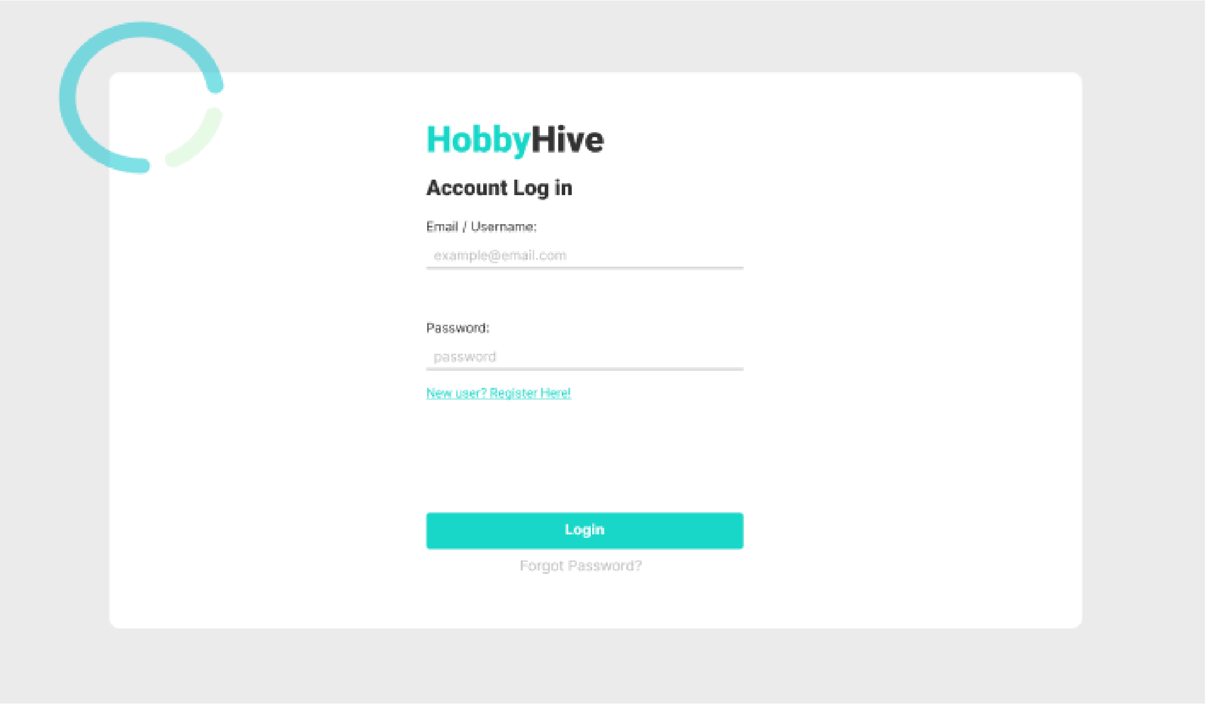
V.IV CommunicateGroupChat

Use Case ID:	#5-4		
Use Case Name:	CommunicateGroupChat		
Created By:	Swetha Sudhakar	Last Updated By:	Swetha Sudhakar
Date Created:	8 September 2025	Date Last Updated:	8 September 2025

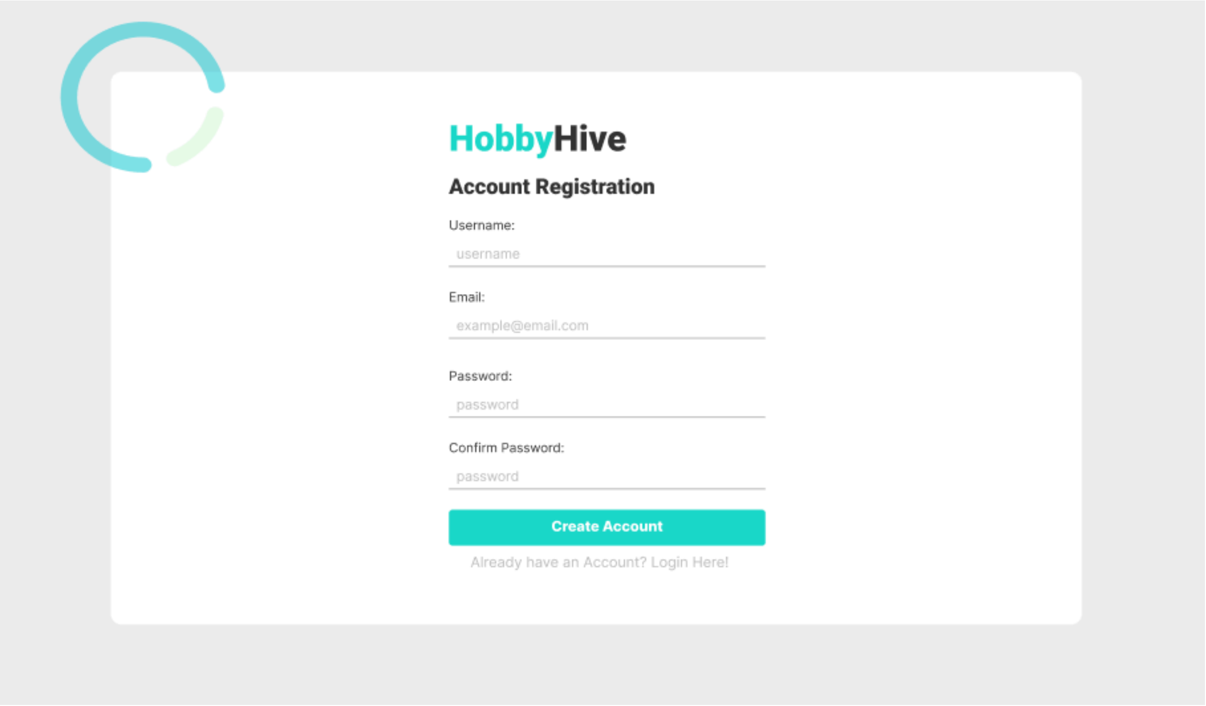
Actor:	Host, Participant
Description:	Allows Hosts and Participants to exchange messages in the group chat.
Pre-Conditions:	User must be part of the group chat.
Post-Conditions	Messages are delivered to all chat members.
Priority:	High
Frequency Of Use:	High
Flow Of Events:	1. User types a message. 2. System transmits message. 3. All members see the message in real time.
Alternative Flows:	**AF-S1:** If message fails to send, system retries.
Exceptions:	Connectivity issue.
Includes:	NotifyChanges (for event updates)
Extends:	None
Special Requirements:	Must support text of up to 256 characters per message.
Assumptions:	Users communicate respectfully.
Notes and Issues:	Consider adding media support in the future.

4. UI Mockups

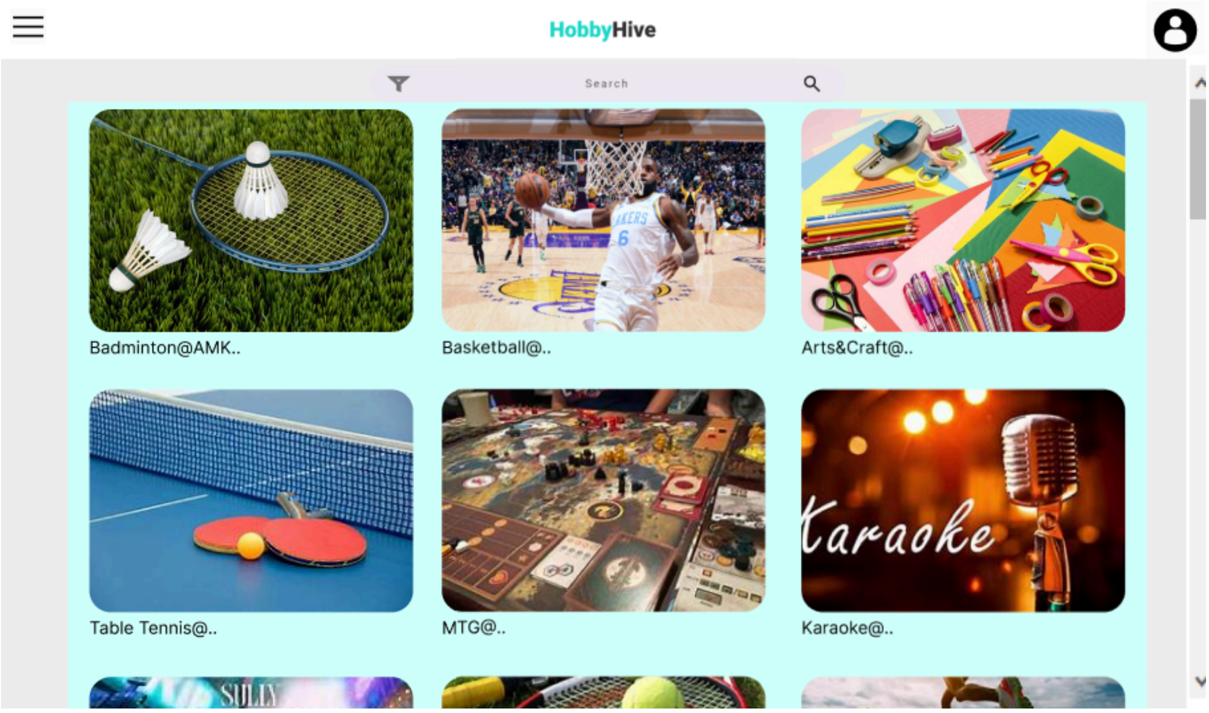
4.1. Login



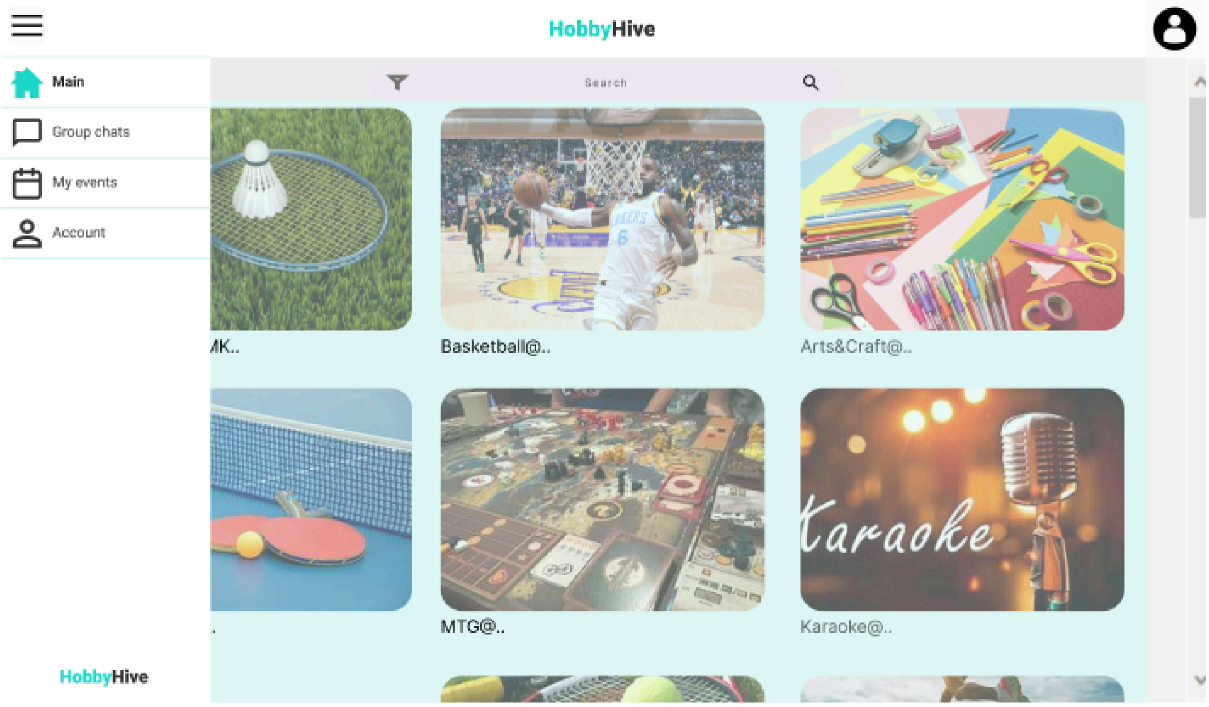
4.2. Register Account



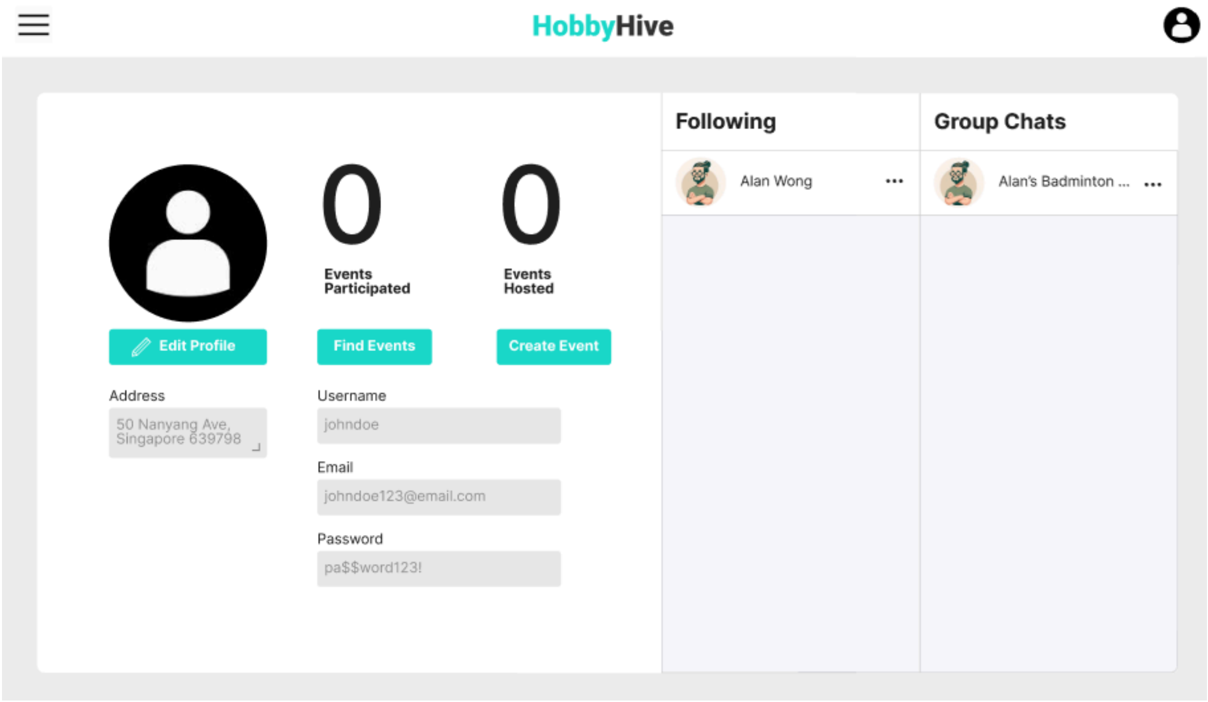
4.3. Home Page



4.4. Dropdown on Home Page



4.5. Profile Page



0 Events Participated

0 Events Hosted

Address: 50 Nanyang Ave, Singapore 639798

Username: johndoe

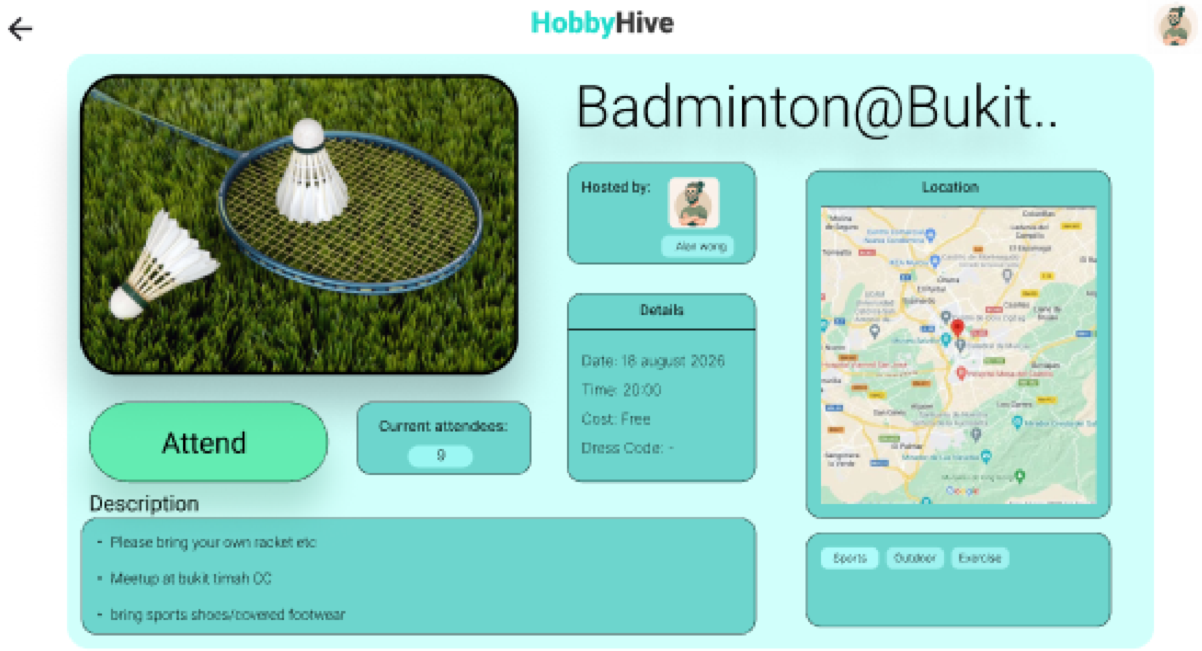
Email: johndoe123@email.com

Password: pa$$word123!

Alan Wong ...

Alan's Badminton ... ...

4.6. Events Listing Page



Badminton@Bukit..

Hosted by: Alex wong

Details

Date: 18 august 2026

Time: 20:00

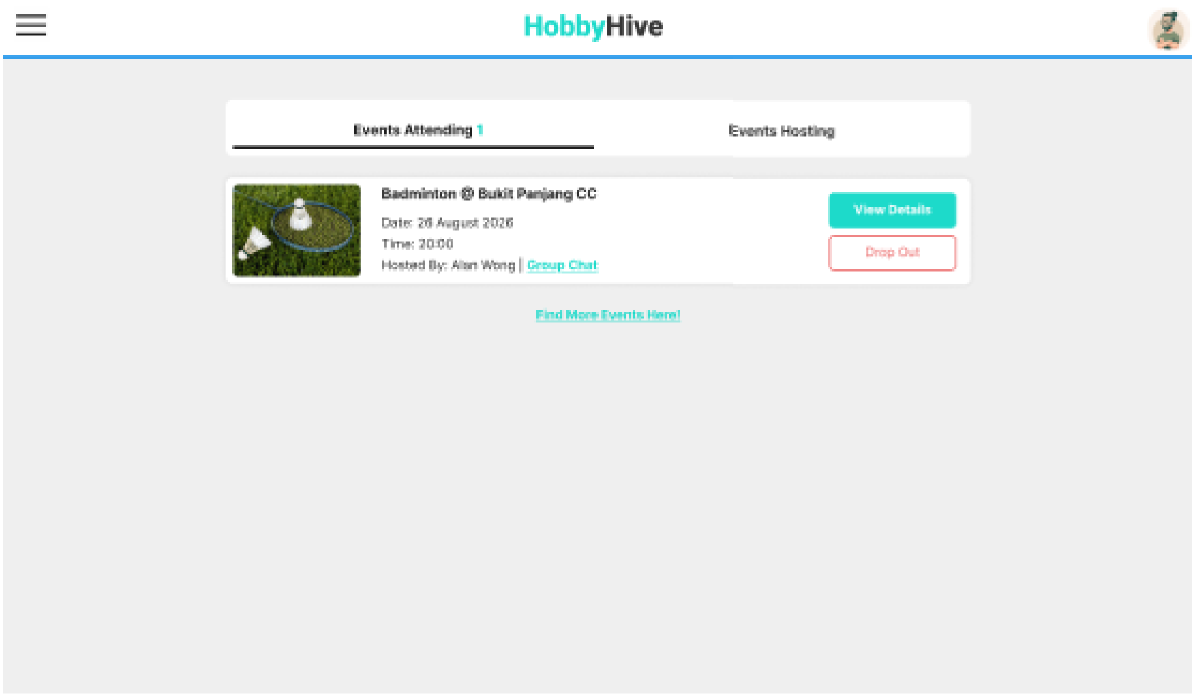
Cost: Free

Dress Code: -

Current attendees: 9

- Please bring your own racket etc
- Meetup at bukit timah CC
- bring sports shoes/covered footwear

4.7. Current Attending/Hosting events



HobbyHive

Events Attending 1 | Events Hosting

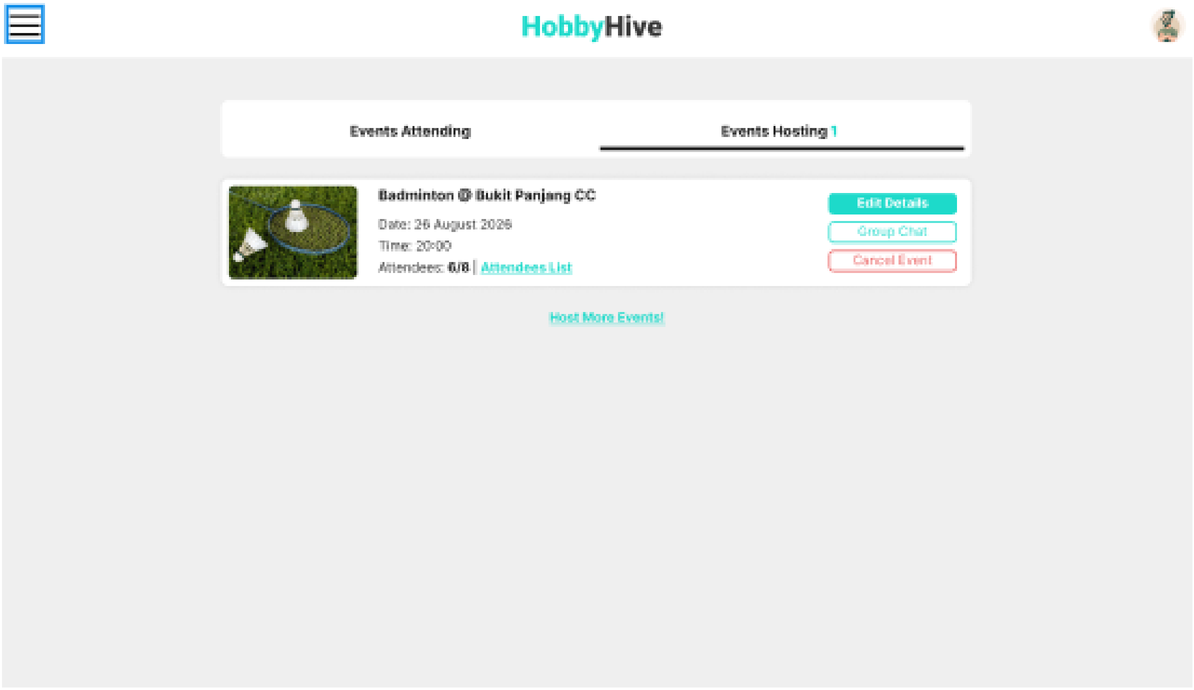
Badminton @ Bukit Panjang CC

Date: 26 August 2026

Time: 20:00

Hosted By: Alan Wong | Group Chat

Find More Events Here!



HobbyHive

Events Attending | Events Hosting 1

Badminton @ Bukit Panjang CC

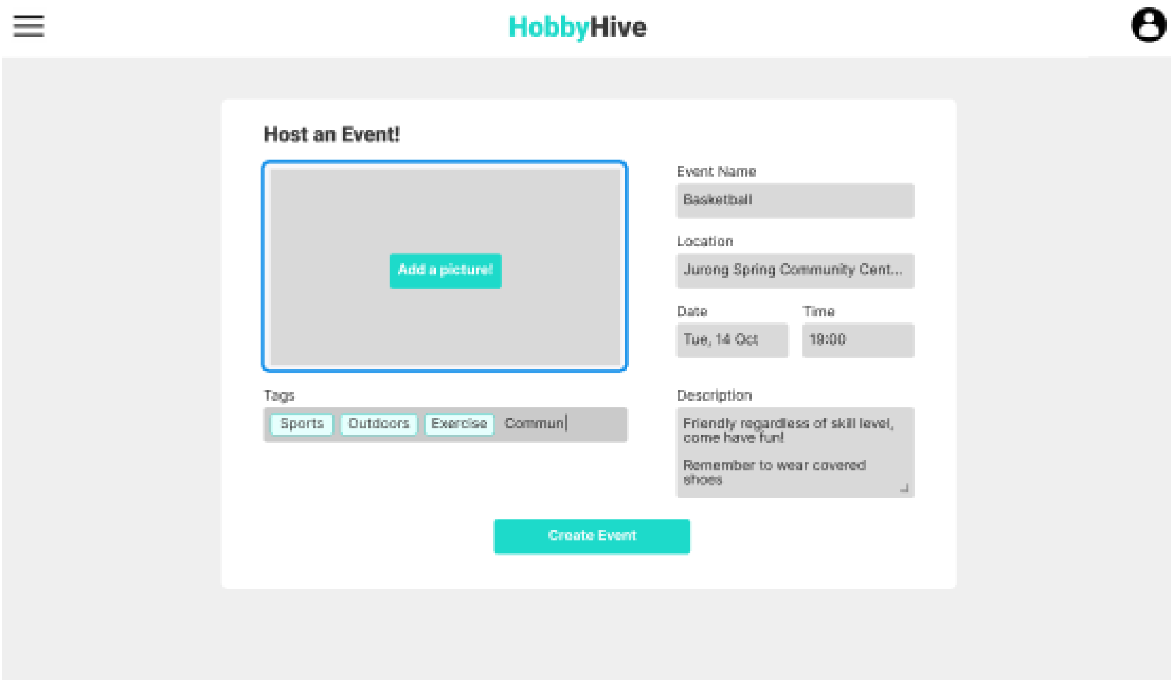
Date: 26 August 2026

Time: 20:00

Attendees: **6/8** | Attendees List

Host More Events!

4.8. Create Listings Page



Event Name: Basketball

Location: Jurong Spring Community Cent...

Date: Tue, 14 Oct

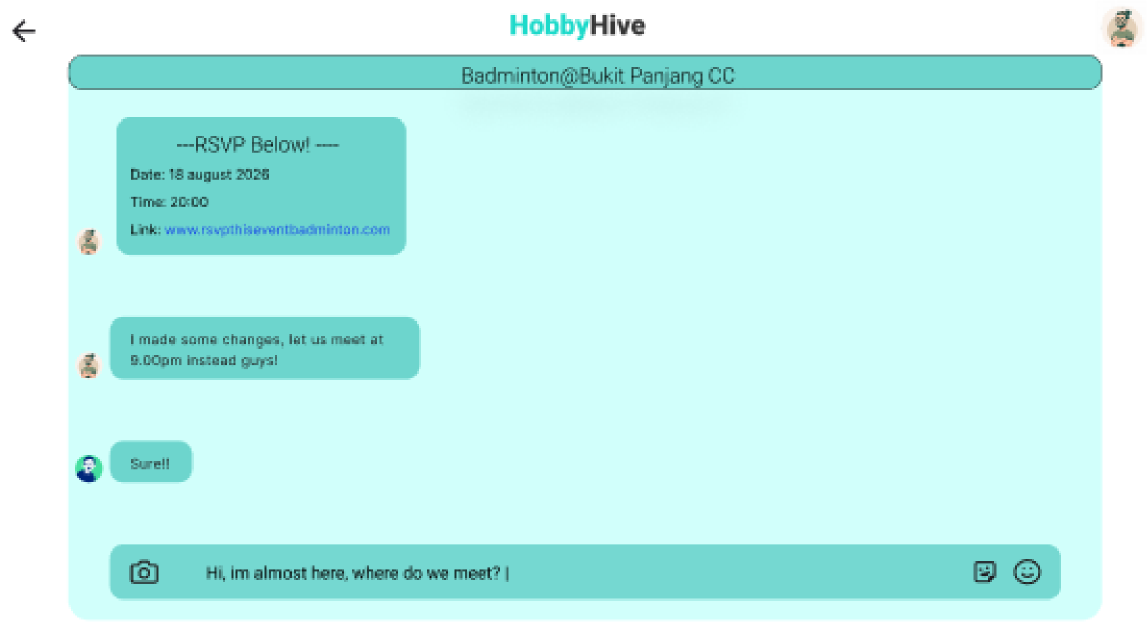
Time: 19:00

Tags: Sports | Outdoors | Exercise | Commun|

Description: Friendly regardless of skill level, come have fun!

Remember to wear covered shoes

4.9. Group Chat



---RSVP Below! ----

Date: 18 august 2026

Time: 20:00

Link: www.rsvpthiseventbadminton.com

I made some changes, let us meet at 9.00pm instead guys!

Sure!!

Hi, im almost here, where do we meet? |