VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY
HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

Programming Integration Project – CO3103

“Requirements, Mockups Design, API Spec”

Advisor(s):	**Mr. Le Dinh Thuan**
Class:	**CC01**
Group:	**N8**

HO CHI MINH CITY, OCTOBER 2025

Member List

No.	Student ID	Full Name
1	2313624	Trần Đỗ Cao Trí
2	2352708	Đinh Cao Thiên Lộc
3	2311883	Nguyễn Thị Kim Loan
4	2352918	Nguyễn Lê Đức Phú
5	2352770	Trần Hà My
6	2153485	Nguyễn Quang Khởi
7	2311987	Nguyễn Song Minh Luân

Contents

1 REQUIREMENT ELICITATION

1.1 Domain Context

In today's group trips or activities, payments are decentralized: different members take turns paying for accommodation, admission tickets, meals or transportation. After the trip, the group often struggles to compile each person's share of the expenses and to settle transfers to the correct recipients. As a result, the need for a shared, transparent, and easy-to-use tool is growing. However, the common practice still relies on manual spreadsheets, scattered notes in chat applications, or even memory. This leads to several limitations: data are not centralized, errors are easy to make, revision history is hard to trace, and real-world situations—such as multiple people jointly paying for a single item—are not well supported. Consolidating all member' expenses typically takes a long time duo to manual reconciliation and writing bank-transfer details, which can cause payments to go to the wrong recipient or be for the wrong amount.

The task is to build a system that standardizes and automates the entire process. Specifically, each trip is managed with a unique ID and a corresponding access link that all members can open to view. Users add the list of participants and record transactions including the description, amount, payer(s), and the list of beneficiaries with allocation weights. The system automatically computes each individual's share. At settlement, it provides a transfer list and allows a designated treasurer (coordinator) to be assigned to collect the funds. At the individual level, the system generates bank-transfer QR codes accordingly: if a person still owes money, their QR directs payment of the exact amount to the treasurer; if a person is to be reimbursed, their QR is used by the treasurer to transfer funds back to them. This approach aligns with the project's core requirements by ensuring transparency, ease of use, and a reduced risk of errors.

1.2 Stakeholders and Needs

Stakeholders	Role	Needs
Event Organizer / Treasurer	Creates and manages events (unique ID/shareable link), manages participants, adds/edits/deletes transactions, assigns a collector, calculates & summarizes expenses, views reports, generates payment QRs, exports data/closes the event	Fast flows, consistent data, easy sharing with the group
Participants	Join via link, update profile, view event info, contribute transactions, view balance summary, view payment QR, request withdrawal, view transaction history, leave event	Transparency, instant access, auditability
Collector	Receives payments per final balances; needs accurate bank info that can be updated so the system can generate standard VietQR codes	Accurate bank info that can be updated so the system can generate standard VietQR codes
Banking/Payment & QR service	Supplies data to generate QRs	VietQR-compliant, high-resolution, fast-generated codes
Database system, File export service	Persists/queries event, participant, and transaction data; encrypts sensitive banking info; exports CSV/PDF	Stable integration, data integrity, good performance

Table 1: Stakeholders, their roles and needs

1.3 Scope of Project

In-Scope:

- Build the event creation & management module: generate a unique Event ID and a shareable link; update/delete events.
- Participant management (target ≥ 50 participants per event): add/edit/remove, store basic profile and optional banking info.
- Transaction management: description, amount, support multiple payers per expense and weighted allocation to beneficiaries.
- Computation & summary: totals paid/benefited and final balances (pay/receive) per participant; view summary reports.
- Collector role: designate a treasurer for the event; store banking information for settlement.
- VietQR payment code generation based on each participant's balance; allow download-/share of QR codes.
- Data export (CSV/PDF, UTF-8) and close event to lock further modifications.

- Participant-side features: join via link, update profile, view event details, view summaries & personal QR, contribute expenses, submit withdrawal requests, view history, and leave the event.
- Key performance targets (objective): fast create/update flows, summary computation within a few seconds for small/medium groups, quick QR generation, responsive UI.

Out-of-Scope (Deferred/Post-MVP):

- Automatic payment processing / payment-gateway integration (bank transfers, collection on behalf, automated reconciliation).
- Automatic bank-statement reconciliation; real-time payment confirmation.
- Advanced features beyond core needs: intelligent/ML assistance, multi-currency, dedicated native apps, complex push-notification system.

1.4 Benefits of the System

- **Transparency & auditability:** Centralized records show who paid, who benefited, and final balances; participants can review summaries and history.
- **Fairness & accuracy:** Automatic computation with allocation weights and multiple payers reduces human errors versus manual spreadsheets.
- **Faster settlement via QR:** Per-participant VietQR codes (debt → pay collector; credit → receive from collector), easy to download/share.
- **Time-saving operations.**
- **Scalability & record-keeping:** Supports $\geq$50 participants per event; export to CSV/PDF UTF-8 for archiving/reconciliation.
- **Sensitive-data protection.**
- **Easy sharing & access.**

2 System Requirements Specification

2.1 Functional Requirements

- **FR1 — Create event**
  The system allows creating a new event, generates a unique Event ID and shareable link bill.thuanle.me/event/{EVENT_ID}, persists it, and shows errors for invalid input.
- **FR2 — Update/Delete event**
  Enable editing event info or deleting an event; confirmation required; success/failure feedback returned.

- **FR3 — Manage participants**
  Add/Edit/Delete participants; optional banking info; prevent deletion if referenced by transactions.
- **FR4 — Assign collector**
  Select a participant as collector and store their banking information for settlement.
- **FR5 — Manage transactions**
  Add/Edit/Delete transactions with: description, amount, multiple payers, and beneficiaries with ratios; validate data before persisting.
- **FR6 — Calculate & summarize expenses**
  Compute total spent, total benefited, and final balance (pay/receive) per participant; allow per-person drill-down and report export.
- **FR7 — View summary report**
  Display summary table (name, spent, benefit, balance), support sort/filter and export CSV/PDF.
- **FR8 — Update collector info**
  Enter/update collector's bank name, account number, account owner; used automatically for payment QR generation.
- **FR9 — Generate VietQR payment codes**
  Generate per-participant QR based on balances: debt → pay collector, credit → receive from collector; allow download/share (ZIP/PDF).
- **FR10 — Export data / Close event**
  Export CSV/PDF (UTF-8, with headers); close event to lock add/edit/delete operations and display Closed status.
- **FR11 — Update personal info**
  User updates profile (phone, address, . . .) after login; system verifies they can access their own profile.
- **FR12 — View event information**
  Joined participant views event details (members, transactions, current fund); non-member sees Join Event invitation.
- **FR13 — Join event via link**
  Access bill.thuanle.me/event/{ID_EVENT}, enter basic info (optional bank), system adds user to member list.
- **FR14 — View money distribution summary**
  View spent/benefit/balance; can switch to transaction detail or payment QR.
- **FR15 — View payment QR**
  Show payment details and QR for remaining amount due.

- **FR16 — Contribute transaction**
  Add own expense: amount, multiple payers (if any), benefit ratios; default to equal split when ratios omitted.
- **FR17 — Submit withdrawal request**
  Submit refund/withdrawal with amount, reason, receiving account; system records and notifies organizer for approval.
- **FR18 — View transaction history**
  View chronological transaction list with filter/sort and per-entry details.
- **FR19 — Leave event**
  Remove oneself from the event's participant list.

2.2 Non-Functional Requirements

2.2.1 Performance & Scalability

- Event creation $\leq$ 3s; name supports Unicode.
- Participant updates reflect instantly ($\leq$ 500ms); support $\geq$ 50 participants/event.
- Save/update transaction $\leq$ 2s; UI remains responsive while loading participants.
- Summary calculation $\leq$ 3s for $\leq$ 50 txns & 20 users; result UI $\leq$ 200ms.
- Load summary $\leq$ 2s for $\leq$ 20 users; sorting/filtering $\leq$ 200ms.
- Generate payment QRs $\leq$ 5s for $\leq$ 20 users; each QR $\geq$ 300×300 px.
- Export files $\leq$ 5s for $\leq$ 50 txns & 20 users.
- Event info load $\leq$ 5s; participant summary load/calc $\leq$ 5s; personal QR generated $\leq$ 2s.

2.2.2 Correctness & Data Integrity

- Amounts use two decimal places; ratio computations accurate with real numbers.
- Updates propagate immediately to dependents; deletions preserve integrity (cascade).
- Once closed, disable all modification APIs and show Closed badge.
- Withdrawal requests stored securely and immutable after submission.

2.2.3 Security & Privacy

Encrypt sensitive banking info at rest; validate account number format; authenticate/authorize access to profiles and event pages.

2.2.4 Availability & Error Handling

Clear error messages for DB errors, duplicate IDs, missing data, unavailable QR service; skip participants lacking bank info during QR generation with notices.

2.2.5 Usability & Accessibility

- Real-time input validation; snappy UI (200ms) for sort/filter; mobile-friendly join flow.
- Event name supports Unicode for multilingual input.

2.2.6 Compatibility & Compliance

QRs must follow the VietQR standard encoding account, amount, note; exports use UTF-8 and preserve labels/amounts.

2.2.7 Business Constraints

The system does not execute bank transfers; it computes & generates QRs so users settle the right amounts to the right recipients.

3 Use-case diagram and Use-case scenario

3.1 Common usecase

3.1.1 UC-SYSTEM-01: Login

Table 2: Use-case: Login

Field	Details
Use-case Name	Login
Use-case ID	UC-SYSTEM-01
Primary Actors	Event Organizer, Participant
Secondary Actors	Database System, Authentication Service
Goal/Purpose	To allow users to securely access the system.
Scope	Travel Expense Splitter Web System
Pre-conditions	User has a registered account in the system.
Post-conditions	**Success:** User is authenticated and redirected to dashboard. **Failure:** Error message is displayed.
Normal Flow	1. User accesses the login page. 2. System displays login form (Email and Password fields). 3. User enters email and password. 4. User clicks "Login" button. 5. System validates credentials against database. 6. System generates session token (JWT) with 7-day expiration. 7. System redirects user based on role: • Event Organizer: Event management dashboard • Participant: List of joined events 8. System displays welcome message.

Continued on next page

Field	Details
Alternative Flow	**A1. Forgot Password:** 1. At Step 2, User clicks "Forgot Password". 2. System displays password reset form. 3. User enters email address. 4. System sends reset link via email. 5. User clicks link and creates new password. **A2. Login with Google:** 1. At Step 2, User clicks "Login with Google". 2. System redirects to Google OAuth. 3. User authorizes and System logs them in.
Exception Flow	**E1. Invalid Credentials:** E1. At Step 5, System detects incorrect email or password. E2. System displays: "Invalid email or password." E3. System limits to 5 attempts per 15 minutes. **E2. Database Error:** E2. At Step 5, System cannot connect to database. E2. System displays: "Unable to process login. Please try again later."
Special Requirements	• Passwords hashed using bcrypt. • Login process completes within 2 seconds. • All endpoints use HTTPS. • Session tokens expire after 7 days of inactivity.

3.1.2 UC-SYSTEM-02: Register

Table 3: Use-case: Register

Field	**Details**
Use-case Name	Register
Use-case ID	UC-SYSTEM-02
Primary Actors	New User
Secondary Actors	Database System, Email Service
Goal/Purpose	To allow new users to create an account.
Scope	Travel Expense Splitter Web System
Pre-conditions	User does not have an existing account.
Post-conditions	**Success:** New account created and user is logged in. **Failure:** No account created, error message displayed.
Normal Flow	1. User clicks "Register" button. 2. System displays registration form with fields: • Full Name (required) • Email (required) • Password (required, min 8 characters) • Confirm Password (required) • Phone Number (optional) 3. User fills in information. 4. User clicks "Create Account". 5. System validates input data. 6. System checks email does not already exist. 7. System hashes password using bcrypt. 8. System creates new user account with default role "Participant". 9. System sends verification email. 10. System automatically logs in user. 11. System redirects to dashboard.

Continued on next page

Field	Details
Alternative Flow	**A1. Register with Google:** 1. At Step 2, User clicks "Sign up with Google". 2. System redirects to Google OAuth. 3. Google returns user profile (email, name). 4. System auto-fills form with Google data. 5. System creates account (no password needed). 6. Continue from Step 10.
Exception Flow	**E1. Email Already Registered:** E1. At Step 6, System detects email exists. E2. System displays: "This email is already registered. Please login." **E2. Invalid Input:** E2. At Step 5, System detects validation errors. E2. System displays specific error messages: - "Password must be at least 8 characters." - "Invalid email format." - "Passwords do not match." E2. User corrects and resubmits. **E3. Database Error:** E3. At Step 8, System cannot save account. E3. System displays: "Unable to create account. Please try again later."
Special Requirements	- Passwords hashed using bcrypt. - Registration completes within 3 seconds. - Email verification required before creating events. - Default role: "Participant".

3.2 Event Organizer

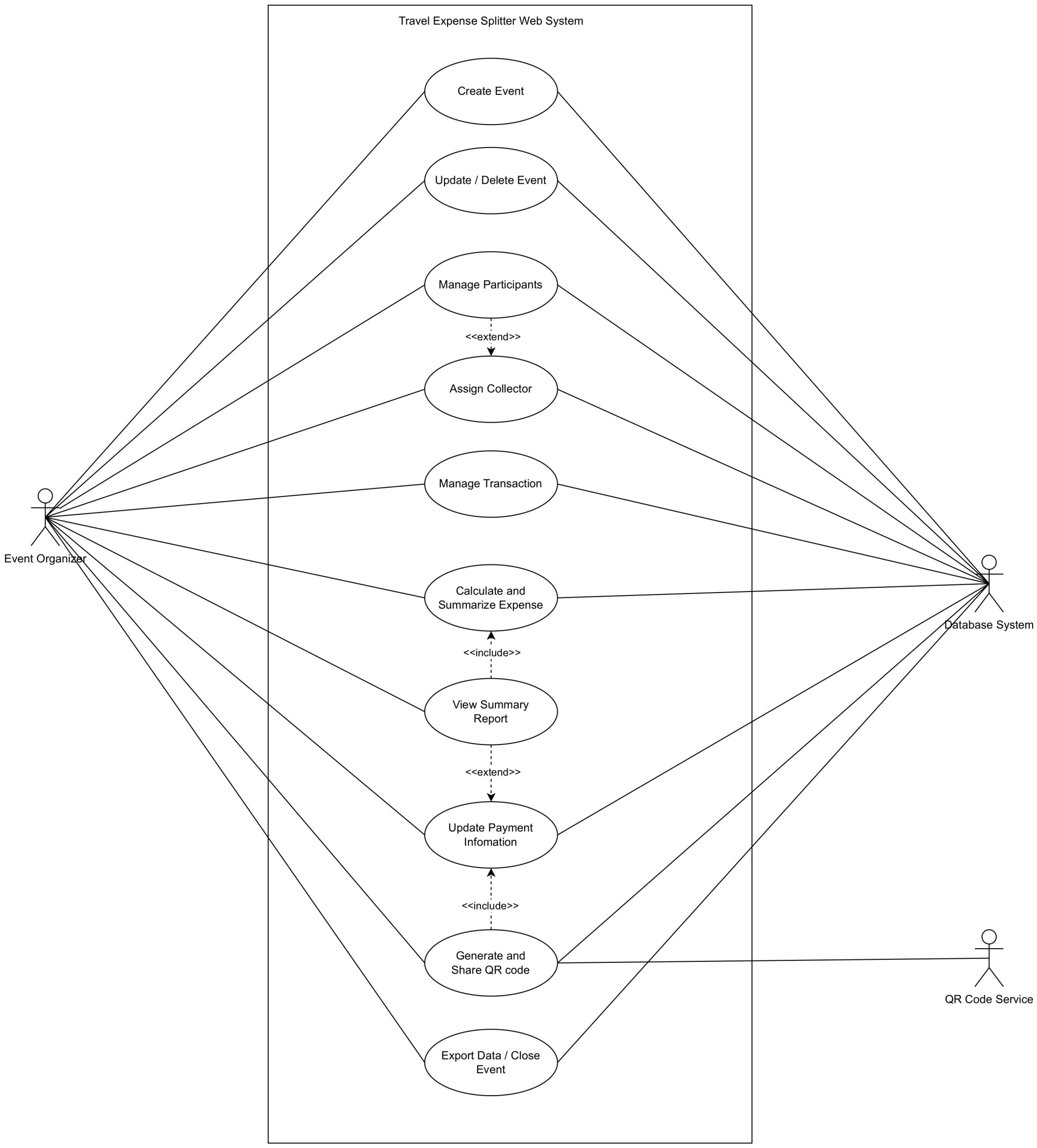


Figure 1: Use-case for event organizer

3.2.1 UC-ADMIN-01: Create Event

Table 4: Use-case: Create Event

Field	**Details**
Use-case Name	Create Event
Use-case ID	UC-ADMIN-01
Primary Actors	Event Organizer
Secondary Actors	Database System – stores and retrieves event data.
Goal/Purpose	To allow the event organizer to create a new trip or event for managing shared expenses among participants.
Scope	Travel Expense Splitter Web System
Level	User-goal level
Pre-conditions	• The organizer has access to the system interface. • No existing event with the same identifier.
Post-conditions	**Success:** • A new event record is created and stored in the database. • The system generates a unique event ID and shareable link (bill.thuanle.me/event/{EVENT_ID}). **Failure:** • The system displays an error message if event creation fails (e.g., missing name, connection error).
Normal Flow	1. The Organizer selects the “Create New Event” function. 2. The System displays the event creation form. 3. The Organizer enters event information (name, description, dates, etc.). 4. The Organizer confirms creation. 5. The System validates the data and generates a unique Event ID. 6. The System stores event data in the database. 7. The System displays confirmation and the event access link.

Continued on next page

Field	Details
Alternative Flow	• **A1.** Organizer leaves mandatory fields empty: The System prompts the user to complete the missing fields. • **A2.** Organizer cancels before confirmation: No event is created, and the system returns to the previous state.
Exception Flow	• **E1.** Database error: The System displays "Event creation failed. Please try again." • **E2.** Duplicate Event ID generated: The System regenerates a new unique ID and retries saving.
Special Requirements	• The event creation process must complete within 3 seconds. • The event name field must support Unicode characters.

3.2.2 UC-ADMIN-02: Update / Delete Event

Table 5: Use-case: Update / Delete Event

Field	Details
Use-case Name	Update / Delete Event
Use-case ID	UC-ADMIN-02
Primary Actors	Event Organizer
Secondary Actors	Database System – stores and maintains event data.
Goal/Purpose	To allow the event organizer to modify event information or delete an event entirely.
Scope	Travel Expense Splitter Web System
Level	User-goal level
Pre-conditions	• The event exists in the system. • The organizer has permission to manage it.

Continued on next page

Field	Details
Post-conditions	**Success:** - Event information is updated, or an event is removed from the database. **Failure:** - The system shows an error message if update or delete operation fails.
Normal Flow	**Update:** 1. The Organizer navigates to the event page and selects an existing event. 2. The Organizer selects "Edit Event." 3. The System displays editable event fields. 4. The Organizer modifies the information and confirms. 5. The System validates input and updates the database. 6. The System displays success confirmation. **Delete:** 1. The Organizer navigates to the event page and selects an existing event. 2. The Organizer selects "Delete Event." 3. The System prompts for confirmation. 4. The Organizer confirms deletion. 5. The System removes the event and associated data.
Alternative Flow	- **A1.** Organizer cancels deletion: The operation is aborted, and no data is changed. - **A2.** Invalid input detected during update: The System requests correction before saving.

Continued on next page

Field	Details
Exception Flow	• **E1.** Database error during update or delete: The System displays "Unable to process your request." • **E2.** Event already deleted by another user: The System displays "Event not found."
Special Requirements	• Update operations must reflect in all dependent modules immediately. • Deletion must ensure data integrity (cascade delete related participants and transactions).

3.2.3 UC-ADMIN-03: Manage Participants

Table 6: Use-case: Manage Participants

Field	Details
Use-case Name	Manage Participants
Use-case ID	UC-ADMIN-03
Primary Actors	Event Organizer
Secondary Actors	Database System – stores participant data for each event.
Goal/Purpose	To allow the event organizer to add, edit, or remove participants in an event.
Scope	Travel Expense Splitter Web System
Level	User-goal level
Pre-conditions	• The event exists. • The organizer has access rights to modify participant data.
Post-conditions	**Success:** • The participant list is updated accordingly in the database. **Failure:** • Operation fails with a suitable error message (e.g., participant used in transaction).

Continued on next page

Field	Details
Normal Flow	**Add:** 1. The Organizer selects “Add Participant.” 2. The System displays participant input form. 3. The Organizer enters participant name and optional bank info. 4. The Organizer confirms the addition. 5. The System stores the new participant record in the database. **Edit:** 1. The Organizer selects an existing participant. 2. The Organizer edits details and confirms changes. 3. The System updates participant information. **Delete:** 1. The Organizer selects “Delete Participant.” 2. The System prompts for confirmation. 3. The Organizer confirms deletion. 4. The System removes the participant from the list.
Alternative Flow	• **A1.** Attempt to delete participant linked to transactions: The System shows “Cannot delete participant with existing transactions.”
Exception Flow	• **E1.** Database connection failure: The System displays “Unable to save participant information.” • **E2.** Duplicate participant name detected: The System displays “Participant already exists.”
Special Requirements	• Participant list updates must reflect instantly without page reload (under 500ms). • The system must support at least 50 participants per event.

3.2.4 UC-ADMIN-04: Assign Collector

Table 7: Use-case: Assign Collector

Field	**Details**
Use-case Name	Assign Collector
Use-case ID	UC-ADMIN-04
Primary Actors	Event Organizer
Secondary Actors	Database System – updates collector information linked to an event.
Goal/Purpose	To allow the event organizer to assign one participant as the money collector who will receive payments from others.
Scope	Travel Expense Splitter Web System
Level	User-goal level
Pre-conditions	• Events and participants exist in the system. • At least one participant has valid banking information.
Post-conditions	**Success:** • Collector information is stored in the database and associated with the event. **Failure:** • The system displays an error if the collector cannot be assigned.
Normal Flow	1. The Organizer navigates to the “Collector” section in the event page. 2. The System displays a list of participants. 3. The Organizer selects one participant as the collector. 4. The Organizer verifies or enters bank account details. 5. The Organizer confirms assignment. 6. The System saves collector information and displays confirmation.

Continued on next page

Field	Details
Alternative Flow	• **A1.** Organizer changes collector → The System updates the new collector and overwrites the previous one. • **A2.** Organizer adds missing bank info before confirmation.
Exception Flow	• **E1.** Selected participant does not exist → The System displays "Invalid collector." • **E2.** Database error during save → The System displays "Failed to assign collector. Please retry."
Special Requirements	• Collector change must update corresponding QR code generation logic immediately. • The system must validate the account number format before saving.

3.2.5 UC-ADMIN-05: Manage Transactions

Table 8: Use-case: Manage Transactions

Field	Details
Use-case Name	Manage Transactions
Use-case ID	UC-ADMIN-05
Primary Actors	Event Organizer
Secondary Actors	Database System – stores and updates transaction data.
Goal/Purpose	To allow the event organizer to add, modify, or delete transactions related to an event.
Scope	Travel Expense Splitter Web System
Level	User-goal level
Pre-conditions	• The event exists in the system. • The participant list has been defined.

Continued on next page

Field	Details
Post-conditions	**Success:** • Transaction information is saved, modified, or removed successfully. **Failure:** • System notifies the user of the issue (e.g., invalid input or database failure).
Normal Flow	1. **Add:** (a) The Organizer selects “Add Transaction.” (b) The System displays the transaction input form. (c) The Organizer enters transaction details (description, amount, payer(s), beneficiaries with ratios). (d) The Organizer confirms addition. (e) The System validates and saves the transaction to the database. 2. **Edit:** (a) The Organizer selects an existing transaction. (b) The Organizer updates details and confirms. (c) The System updates the database and displays confirmation. 3. **Delete:** (a) The Organizer selects a transaction to delete. (b) The System prompts for confirmation. (c) The Organizer confirms deletion. (d) The System removes the transaction from the database.
Alternative Flow	• **A1.** Organizer enters invalid amount (e.g., negative or zero) → The System prompts correction. • **A2.** Organizer cancels action before saving → No change is made.

Continued on next page

Field	Details
Exception Flow	- **E1.** Database connection lost → The System displays "Unable to save transaction. Please try again." - **E2.** Data inconsistency detected (payer not in event) → The System displays "Invalid participant selection."
Special Requirements	- The system must process transaction save/update within 2 seconds. - Amount values must be validated to two decimal places. - The interface should remain responsive while loading participant lists.

3.2.6 UC-ADMIN-06: Calculate and Summarize Expense

Table 9: Use-case: Calculate and Summarize Expense

Field	Details
Use-case Name	Calculate and Summarize Expense
Use-case ID	UC-ADMIN-06
Primary Actors	Event Organizer
Secondary Actors	Database System – stores and queries information of participants and transactions.
Goal/Purpose	To help the organizer generate a spending summary for the entire event, including: - The total amount each participant has paid. - The total amount each participant has benefited from. - The amount each participant still owes or should receive. The final result is a fair and transparent expense summary for the whole group.
Scope	Travel Expense Splitter Web System
Level	User-goal level (user's business objective)

Continued on next page

Field	Details
Pre-conditions	• The event has been successfully created. • The list of participants and transactions has been fully entered. • All stored data is valid (no missing payer or beneficiary, all amounts > 0).
Post-conditions	**Success:** • The system displays a summary table including: – Total amount spent by each participant. – Total amount each participant benefited from. – Remaining amount each participant must pay or receive. • The summary data is temporarily stored for display or report export. **Failure:** • The system displays an error message (e.g., missing data, invalid transaction, or no participants found).

Continued on next page

Field	**Details**
Normal Flow	1. The Organizer selects the event to summarize → The System loads the event, participants, and transaction data. 2. The Organizer chooses the "Calculate Summary" function → The System begins processing calculations. 3. The System calculates the total amount spent by each participant (based on payers). 4. The System calculates the amount each participant benefited from (based on benefit ratios). 5. The System compares the difference between the amount spent and the amount benefited. 6. The Organizer views the calculation results → The System displays a summary list including: name, amount spent, amount benefited, and amount to pay/receive. 7. The Organizer can save or export the summary results → The System saves the summary state or allows exporting as a CSV/PDF file.
Alternative Flow	• **A1.** Organizer wants to view detailed transactions: In Step 6, the Organizer clicks on a participant's name → The System displays a detailed list of that participant's related transactions. • **A2.** Organizer wants to adjust benefit ratios: Before Step 2, the Organizer can return to edit transaction data, then re-run the "Calculate Summary" process.
Exception Flow	• **E1.** No transaction data available: When "Calculate" is clicked, the System displays an error message: "No transactions found for this event." • **E2.** Transaction contains negative amount or missing beneficiaries: The System stops the calculation and shows a specific error message for correction. • **E3.** Database connection error: The System displays: "Unable to access database. Please try again later."

Continued on next page

Field	Details
Special Requirements	• Calculation results must be processed within $<$ 3 seconds for up to 50 transactions and 20 participants. • The result display interface must respond instantly (within 200 ms after computation). • The system must ensure absolute accuracy in ratio calculations (using real numbers with two decimal places).

3.2.7 UC-ADMIN-07: View Summary Report

Table 10: Use-case: View Summary Report

Field	Details
Use-case Name	View Summary Report
Use-case ID	UC-ADMIN-07
Primary Actors	Event Organizer
Secondary Actors	Database System – stores and provides access to event summary data.
Goal/Purpose	To allow the event organizer to view the overall summary report after calculation, including each participant's total spending, benefits, and balance (amount owed or to be received).
Scope	Travel Expense Splitter Web System
Level	User-goal level
Pre-conditions	• The event must already exist. • The expense calculation (UC-ADMIN-06) has been completed successfully. • Summary data has been generated and stored temporarily or permanently in the database.

Continued on next page

Field	Details
Post-conditions	**Success:** • The system displays the event summary report, including: – Participant name – Total paid amount – Total benefit – Balance (positive = receive money, negative = pay more) • The report can be viewed, exported, or printed. **Failure:** • The system shows an appropriate error message if no summary data is found or cannot be loaded.
Normal Flow	1. The Organizer navigates to the event page → The System retrieves event and participant information. 2. The Organizer selects "View Summary Report" function → The System loads the latest calculation results. 3. The System displays the summary table, including each participant's spending, benefit, and balance. 4. The Organizer can sort or filter the data (e.g., by amount, name) → The System refreshes the displayed data accordingly. 5. The Organizer may choose to export the report → The System offers export options (CSV, PDF).
Alternative Flow	• **A1.** Organizer wants to view individual participant details: At Step 3, clicking a participant's name opens a detailed breakdown of that person's transactions. • **A2.** Organizer wants to recalculate before viewing: Before Step 2, the Organizer can trigger "Recalculate Summary" (UC-ADMIN-06) to ensure up-to-date data.

Continued on next page

Field	Details
Exception Flow	- **E1.** No summary data available: The System displays: “No summary data found. Please perform calculation first.” - **E2.** Database connection failure: The System shows: “Unable to load summary. Please try again later.” - **E3.** Data inconsistency detected: The System notifies: “Invalid or incomplete summary data detected.”
Special Requirements	- The summary table must load in $<$ 2 seconds for $\leq$ 20 participants. - The interface must remain responsive while sorting or filtering ($<$ 200ms delay). - Exported reports must maintain consistent numeric formatting (2 decimal places).

3.2.8 UC-ADMIN-08: Update Collector Information

Table 11: Use-case: Update Collector Information

Field	Details
Use-case Name	Update Collector Information
Use-case ID	UC-ADMIN-08
Primary Actors	Event Organizer
Secondary Actors	Database System – stores collector information (bank name, account number, account owner).
Goal/Purpose	Allow the event organizer to specify or update the payment collector’s information, which will be used later to generate payment QR codes for participants to transfer money accurately.
Scope	Travel Expense Splitter Web Application
Level	User-goal level

Continued on next page

Field	Details
Pre-conditions	• The event has been created successfully. • Expense calculation has been completed or event data is ready for settlement. • The organizer has permission to edit collector information.
Post-conditions	**Success:** • The collector's banking information is saved in the database. • The information will be used automatically when generating payment QR codes. **Failure:** • System displays an error if required fields (e.g., account number, bank name) are missing or invalid.
Normal Flow	1. The Organizer opens the "Collector Information" section of the event → The System displays current collector data (if any). 2. The Organizer enters or edits collector info: bank name, account number, and account owner → The System validates input format (non-empty, valid account number). 3. The Organizer clicks "Save" → The System saves data to the database. 4. The System confirms the update → Displays message: "Collector information has been successfully updated."
Alternative Flow	• **A1.** Organizer wants to remove collector info: At Step 2, Organizer clicks "Remove Collector Info." The System clears stored collector data after confirmation. • **A2.** Organizer selects an existing participant as collector: The System automatically fills in that participant's bank information (if available).

Continued on next page

Field	Details
Exception Flow	- **E1.** Missing or invalid input: The System highlights missing fields and shows: "Please enter all required collector information." - **E2.** Database connection error: The System shows: "Unable to update collector info. Please try again later."
Special Requirements	- The System must validate all input fields in real-time (no page reload). - Sensitive banking info must be encrypted before storage. - Update operation should complete in under 2 seconds.

3.2.9 UC-ADMIN-09: Generate Payment QR Code

Table 12: Use-case: Generate Payment QR Code

Field	Details
Use-case Name	Generate Payment QR Code
Use-case ID	UC-ADMIN-09
Primary Actors	Event Organizer
Secondary Actors	- Banking / Payment System – provides information for generating QR codes. - Database System – stores participant and payment data.
Goal/Purpose	To generate payment QR codes that allow participants to easily transfer or receive money based on the final summary. - If a participant owes money, the QR code directs them to transfer the exact amount to the organizer. - If a participant should receive money, the QR code contains their banking info so others can transfer money to them.
Scope	Travel Expense Splitter Web System
Level	User-goal level

Continued on next page

Field	Details
Pre-conditions	• Expense summary (UC-ADMIN-06) must be successfully calculated. • Organizer and participant banking details are valid and available. • The system has access to a payment/QR generation service.
Post-conditions	**Success:** • The system generates and displays QR codes for: – The organizer (for collecting payments). – Each participant (for paying or receiving money). • QR codes can be downloaded or shared directly. **Failure:** • The system notifies the organizer if QR generation fails due to invalid or missing data.
Normal Flow	1. The Organizer opens the summary report page → The System retrieves event summary data. 2. The Organizer clicks "Generate QR Codes." → The System validates participant and payment information. 3. The System determines each participant's balance (positive = receive, negative = pay). 4. For each participant, the System generates a QR code: • If balance < 0 → QR directs payment to Organizer's bank info. • If balance > 0 → QR contains Participant's bank info. 5. The System displays all generated QR codes in a table → Each row includes name, balance, and QR image. 6. The Organizer can download QR codes or share the report → The System provides download and share options (ZIP/PDF).

Continued on next page

Field	Details
Alternative Flow	- **A1.** Organizer generates QR for one specific participant only: At Step 2, the Organizer selects a participant → The System generates and displays only that QR code. - **A2.** Organizer updates bank info before generating: Before Step 2, the Organizer edits the participant's bank details → Then regenerates QR.
Exception Flow	- **E1.** Missing or invalid bank info: The System skips that participant and shows: "Missing banking details for [Name]." - **E2.** Payment service unavailable: The System shows: "Cannot generate QR codes. Please try again later." - **E3.** Database error: The System displays: "Error retrieving payment data."
Special Requirements	- Each QR code must encode correct payment info (account, amount, note) following VietQR standard. - QR generation must complete in $<$ 5 seconds for $\leq$ 20 participants. - All generated QR codes must maintain high resolution for scanning ($\geq$ 300x300 px). - Exported file (ZIP or PDF) must preserve each QR's label and amount.

3.2.10 UC-ADMIN-10: Export Data / Close Event

Table 13: Use-case: Export Data / Close Event

Field	Details
Use-case Name	Export Data / Close Event
Use-case ID	UC-ADMIN-10
Primary Actors	Event Organizer

Continued on next page

Field	Details
Secondary Actors	• Database System – provides stored event data for export. • File Export Service – generates downloadable CSV or PDF files.
Goal/Purpose	Allow the organizer to finalize the event by either: • Exporting all event data (participants, transactions, final balances) for record keeping, or • Closing the event to prevent further modification.
Scope	Travel Expense Splitter Web Application
Level	User-goal level
Pre-conditions	• The event exists and contains valid participant and transaction data. • Expense calculation and summary have been completed. • The organizer is authenticated and authorized to finalize the event.
Post-conditions	**Success:** • A data file (CSV or PDF) is generated and downloaded to the organizer's device. • The event status changes to **Closed**, preventing further edits. **Failure:** • The system notifies the organizer if the export fails or data is missing.

Continued on next page

Field	**Details**
Normal Flow	1. The Organizer opens the event summary page → The System displays event overview and available actions. 2. The Organizer clicks "Export Data / Close Event." → The System shows dialog with two options: "Export Data" or "Close Event." label*=1. Organizer selects **Export Data** and chooses file format (CSV or PDF) → The System generates the file and starts download. lbbel*=2. Organizer selects **Close Event** → The System requests confirmation before locking further edits. 3. Organizer confirms the action → The System performs export or status update. 4. The System displays a confirmation message → "Event data exported successfully." or "The event has been closed."
Alternative Flow	- **A1.** Organizer wants to export only specific data: At Step 2, selects custom export options (e.g., only transactions, only participant info). The System generates filtered file accordingly. - **A2.** Organizer cancels the close action: At Step 3b, clicks "Cancel." The System aborts closing and keeps the event open.
Exception Flow	- **E1.** Missing calculated data: The System shows: "Please complete the expense calculation before exporting." - **E2.** Export service failure: The System notifies: "Unable to generate file. Please try again." - **E3.** Database access error: The System shows: "Failed to retrieve event data."

Continued on next page

Field	Details
Special Requirements	• File export must complete within 5 seconds for up to 50 transactions and 20 participants. • Exported files should use UTF-8 encoding and include headers for readability. • Once an event is closed, all modification-related APIs (add/edit/delete) are disabled. • The system must display a clear **Closed** status badge on the event page.

3.3 Participant

This use case focuses on the function that the participants can perform. Provides a transparent, instant, and auditable system for managing communal finances. Specifically, the following use cases define the participant's journey from entering a shared expense to viewing their final, balanced settlement summary.

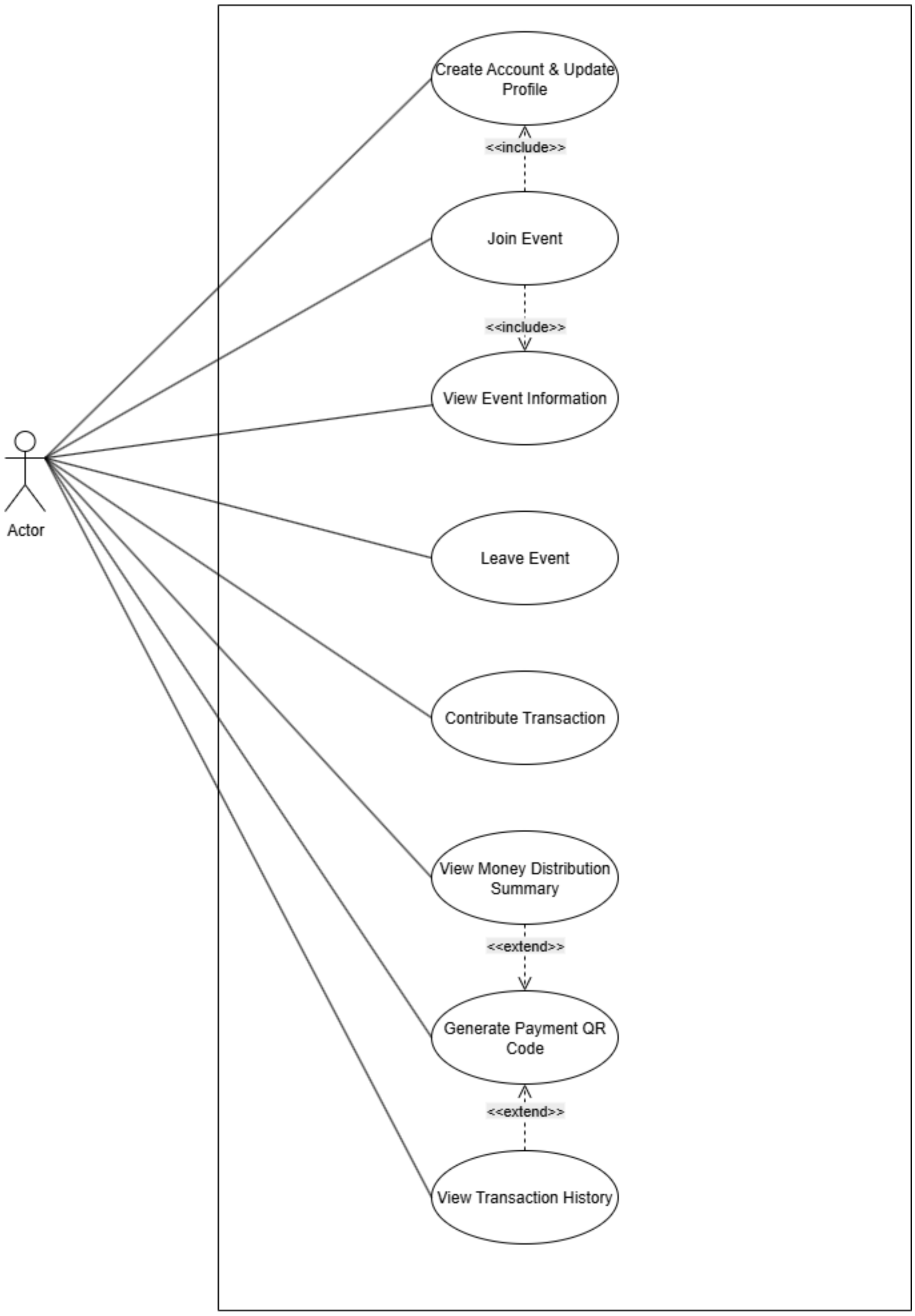


Figure 2: Use-case for participant

3.3.1 UC-PARTICIPANT-01: Update Personal Information

Table 14: Use-case: Update Personal Information

Field	Details
Use-case Name	Update Personal Information
Use-case ID	UC-PARTICIPANT-01
Primary Actors	Participant
Secondary Actors	Database System
Goal/Purpose	To allow the participant to successfully modify and save their personal profile information within the application.
Scope	Travel Expense Splitter Web System
Pre-conditions	The User is logged in and has navigated to the Profile area.
Post-conditions	The User's profile information is updated and committed to the database.
Normal Flow	1. Participant clicks on the "Update personal information" on the profile. 2. The System verifies the participant's authentication and authorization (ensures the user is logged in and permitted to view their own profile). 3. The System retrieves the participant's current personal data from the database. 4. The System displays the "Edit Profile" form, pre-filled with the current information. 5. The Participant reviews the current data and makes changes to one or more fields that includes phone number, address and social security number. 6. The Participant clicks the "Save Changes". 7. The System performs input validation on the submitted data. 8. The System updates the participant's record in the database with the new, validated information. 9. The System displays a confirmation message.

Continued on next page

Field	Details
Alternative Flow	At Step 4, the User decides not to save changes and clicks a "Cancel" or "Back" button. The System returns the user to the previous page without saving any modifications.
Exception Flow	6a. The system prints a message if the information input is not validated.
Special Requirements	None

3.3.2 UC-PARTICIPANT-02: View Event Information

Table 15: Use-case: View Event Information

Field	Details
Use-case Name	View Event Information
Use-case ID	UC-PARTICIPANT-02
Primary Actors	Participant
Secondary Actors	Share-Bill System
Goal/Purpose	The participant wants to view the details of an event they have joined, including the member list, transactions, and the current total fund.
Scope	Travel Expense Splitter System – web application bill.thuanle.me
Pre-conditions	• The participant has already joined a valid event. • The system has data on the member list and transactions for the event.
Post-conditions	• Success: Event details are displayed accurately to the participant. • Failure: The system displays an error message if the event does not exist or data cannot be retrieved.

Continued on next page

Field	Details
Normal Flow	1. The participant accesses the event page they have joined (bill.thuanle.me/event/{ID_EVENT}). 2. The system verifies the user is on the event's member list. 3. The system retrieves data: event name, organizer, member list, transactions, current total fund. 4. The system displays all event information on the user interface. 5. The participant can choose to view transaction details or switch to other tabs (e.g., summary, payment QR).
Alternative Flow	A1. User viewing the event has not joined: → The system displays the "Join Event" invitation page instead of the details. A2. Large event dataset: → The system loads data using pagination or lazy loading to optimize performance.
Exception Flow	E1. Event does not exist or has been deleted: → The system displays an error message "Event does not exist or has been closed." E2. Server error / network connection issue: → Displays message "Could not load event information, please try again later."
Special Requirements	• Data loading time should not exceed 5 seconds.

3.3.3 UC-PARTICIPANT-03: Join Event

Table 16: Use-case: Join Event

Field	Details
Use-case Name	Join Event
Use-case ID	UC-PARTICIPANT-P03
Primary Actors	Participant
Secondary Actors	Share-Bill System

Continued on next page

Field	Details
Goal/Purpose	The participant wants to join a bill-splitting event to view information, contribute transactions, and track the fund summary.
Scope	Travel Expense Splitter Web System – web application bill.thuanle.me
Pre-conditions	• The event has been successfully created by the organizer. • The organizer shares the participation link or code with the user.
Post-conditions	• Success: The participant is added to the event's member list and can access information and perform actions within the event. • Failure: The system displays an error message (e.g., invalid link, event closed).
Normal Flow	1. The Participant accesses the event link (bill.thuanle.me/event/{ID_EVENT}). 2. The System displays the event invitation interface. 3. The Participant enters basic personal information (name, bank info - optional). 4. The Participant clicks "Join Event". 5. The System validates the information and adds the participant to the event's member list. 6. The System displays the event information page.
Alternative Flow	A1. User has previously joined the event: → The system detects the user already exists in the list and automatically redirects to the event page (skipping the information entry step).
Exception Flow	E1. Invalid link or event deleted: → The system displays an error message "Event does not exist or has been closed." E2. Network connection / server error: → Displays message "Cannot connect to the server, please try again later."

Continued on next page

Field	Details
Special Requirements	- User-friendly interface, accessible from mobile devices.

3.3.4 UC-PARTICIPANT-04: View Money Distribution Summary

Table 17: Use-case: View Money Distribution Summary

Field	Details
Use-case Name	View Money Distribution Summary
Use-case ID	UC-PARTICIPANT-P04
Primary Actors	Participant
Secondary Actors	Share-Bill System
Goal/Purpose	The participant wants to see the final summary of the amount they paid, their share, and the amount they still owe or are due to receive.
Scope	Travel Expense Splitter System – web application bill.thuanle.me
Pre-conditions	• The participant has joined a valid event. • The event has transaction data (at least one expense / contribution has been made). • The system has performed an automatic calculation, or the organizer has initiated the summary.
Post-conditions	• Success: The participant can view their personal and the group's summary (who owes whom). • Failure: The system displays an error message if there is no data or the calculation fails.

Continued on next page

Field	Details
Normal Flow	1. The participant accesses the event page. 2. Selects the "Balance Summary" tab or function. 3. The system retrieves all transaction data for the event. 4. The system calculates (if not already available): total amount spent by each person, their share, and the final difference. 5. The system displays the summary table: member name, total paid, total share, amount to pay or receive. 6. The participant views the results and can switch to see transaction details or the payment QR code.
Alternative Flow	A1. Event summary is not yet finalized: → The system displays a message "Temporary summary – data is subject to change." A2. Summary data has been updated by the organizer: → The system displays the final summary with the finalization date.
Exception Flow	E1. No transactions: → The system displays a message "No transactions available to summarize." E2. Calculation error or invalid data: → Displays message "Could not display summary, please try again later."
Special Requirements	• Data loading and calculation time should not exceed 5 seconds.

3.3.5 UC-PARTICIPANT-05: View Payment QR Code

Table 18: Use-case: View Payment QR Code

Field	Details
Use-case Name	View Payment QR Code
Use-case ID	UC-PARTICIPANT-P05

Continued on next page

Field	Details
Primary Actors	Participant
Secondary Actors	Database System
Description	To generate and display a dynamic QR code that allows the participant to settle their outstanding balance.
Scope	Travel Expense Splitter Web System
Pre-conditions	• The User is a registered participant in the selected event. • The User has updated all information.
Post-conditions	The System successfully displays the payment details and an active QR code representing the remaining amount due.
Normal Flow	1. User selects the desired event from their event list. 2. Participant clicks on the "View payment information" on the event setting. 3. The System checks the transaction history for the specific order. 4. The System calculates the Remaining Amount Due. 5. The System displays the Payment Information including Total Amount Required, Amount Already Paid, Remaining Amount Due, A New QR Code is generated for the Remaining Amount Due.
Alternative Flow	None
Exception Flow	3a. Using invalid sign-in credentials results in an error message.
Special Requirements	The QR code must be generated within 2 seconds.

3.3.6 UC-PARTICIPANT-06: Contribute Transaction

Table 19: Use-case: Contribute Transaction

Field	Details
Use-case Name	Contribute Transaction

Continued on next page

Field	Details
Use-case ID	UC-PARTICIPANT-P06
Primary Actors	Participant
Secondary Actors	Share-Bill System
Goal/Purpose	The participant wants to record an expense or their contribution to the event's shared fund.
Scope	Travel Expense Splitter System – web application bill.thuanle.me
Pre-conditions	• The participant has joined a valid event. • The event is still open, allowing new transactions. • The user has permission to contribute (not locked).
Post-conditions	• Success: The transaction is saved to the system; the total expenses and share distribution among members are updated. • Failure: The system does not save the transaction and displays an error message (e.g., incorrect format, missing information).
Normal Flow	1. The participant opens the event page and selects "Add Transaction". 2. The system displays the form to enter new transaction details. 3. The participant enters: description, amount, selects the payer(s) (can be multiple), and the share for each member (e.g., 1, 1, 1.5, 3). 4. The participant clicks "Confirm Contribution". 5. The system validates the data ($amount > 0$, members exist in the event). 6. The system saves the transaction information and updates the total fund. 7. The system displays a "Transaction added successfully" message and updates the transaction list displayed for all members.

Continued on next page

Field	Details
Alternative Flow	A1. Contribution with multiple payers: → The participant selects multiple payers; the system automatically divides the contribution proportionally. A2. Share distribution not entered: → The system defaults to splitting the cost equally among all members.
Exception Flow	E1. Incorrect amount format or missing required information: → The system displays an error message and requests re-entry. E2. Network connection loss or server error: → The system displays "Cannot add transaction, please try again later."
Special Requirements	• Ensure data integrity.

3.3.7 UC-PARTICIPANT-07: View Transaction History

Table 20: Use-case: View Transaction History

Field	Details
Use-case Name	View Transaction History
Use-case ID	UC-PARTICIPANT-P07
Primary Actors	Participant
Secondary Actors	Database System
Description	To allow the participant to securely access and view chronological list of their financial transaction within the system.
Scope	Travel Expense Splitter Web System
Pre-conditions	• The Participant must be logged in. • The participant must have at least one completed transaction recorded in the database.
Post-conditions	The system displays a filtered, sortable list of the participant's transactions.

Continued on next page

Field	Details
Normal Flow	1. The Participant navigates to the “History Transactions” section of the system. 2. The system verifies the Participant’s authentication and authorization. 3. The System retrieves the complete chronological list of the User’s relevant financial transactions from the database 4. The System displays the Transaction History interface, presenting the transactions in a list format. 5. The System ensures the following key details are visible for each entry: Date/Time, Description, Amount, and Type (Debit/Credit). 6. The User reviews the transaction list and can optionally use features like sorting (by date/amount) or filtering.
Alternative Flow	• At Step 3, the System finds no transactions for the User. The System displays the history interface (Step 4) but shows an "empty state" message. • At Step 6, the User applies a filter (e.g., only show transactions from "Event A"). The System performs a new database query (Step 3) and re-renders the history list with the filtered results (Step 4) • At Step 5, the User clicks on a specific transaction entry. The System displays a detailed view showing all relevant information (e.g., full description, associated event, linked participants).
Exception Flow	7a. At Step 2, the User’s session has expired. The System redirects the User to the Login screen. 7b. At Step 3, the database connection fails or the query times out. The System displays an error message.
Special Requirements	None

3.3.8 UC-PARTICIPANT-08: Leave Event

Table 21: Use-case: Leave Event

Field	**Details**
Use-case Name	Leave Event
Use-case ID	UC-PARTICIPANT-P08
Primary Actors	Participant
Secondary Actors	Database System
Description	To allow the participant to successfully remove themselves from the event's participant list.
Scope	Travel Expense Splitter Web System
Pre-conditions	The User is logged in and currently a member of the event.
Post-conditions	The User's participation record is removed from the event and they can no longer access the event details.
Normal Flow	1. The Participant navigates to their list of current events. 2. The Participant chooses the specific event they want to leave. 3. The System displays the Event Information page. 4. The Participant locates and clicks the "Leave Event" button. 5. The System displays a confirmation message. 6. The Participant confirms the action. 7. The System executes the database transaction to remove the Participant from the event's participant list. 8. The System displays a successful message.
Alternative Flow	At Step 6, the User clicks "No, Cancel" or closes the confirmation dialog. The system closes the message and returns the User to the Event Information page.
Exception Flow	7a. The Database System fails to update the record. The system displays a generic error message.
Special Requirements	None

4 Mockup Design and Details

Mockup Design In Figma

4.1 Homepage

4.1.1 Login

Allows users to enter their credentials to authenticate and access the system. This is also the initial interface displayed when users access the website.

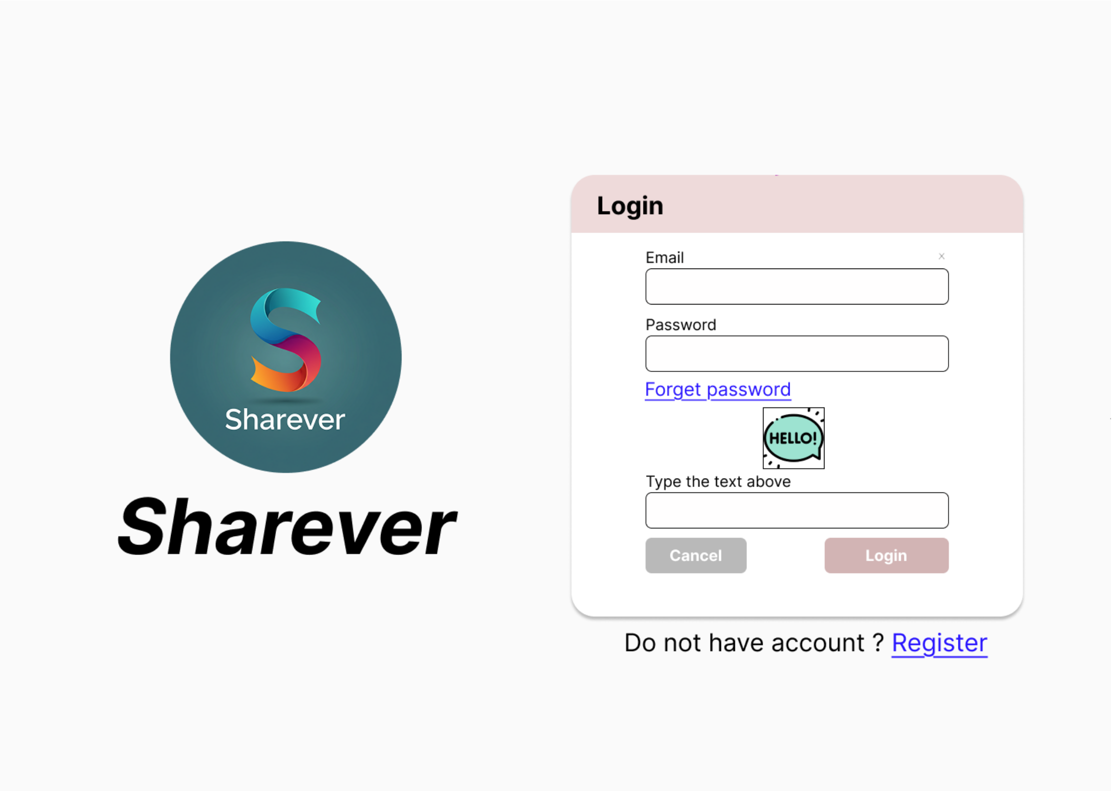


Figure 3: Login

No.	Element	Type	Function	User action	Required
1	Email field	Text input	Enter the user's email address	Type / edit / delete the email	Yes
2	Clear (x) in Email	Icon button	Quickly clear the content of the Email field	Click to clear the entered email	No
3	Password field	Password input	Enter the login password	Type / edit / delete the password	Yes
4	"Forget password" link	Hyperlink	Support users who forgot their password	Click to go to the forgot-password flow	No
5	Captcha image ("HELLO!")	Captcha image	Display the characters that must be retyped	Look at the image and read the text	Indirect
6	"Type the text above" field	Text input	Retype the characters shown in the captcha image	Type the exact same text, edit or clear if needed	Yes
7	"Cancel" button	Button (secondary)	Cancel the current login attempt	Click to clear/close the login form	No
8	"Login" button	Button (primary)	Submit the login form to the server	Click to send Email, Password and Captcha	Yes
9	"Register" link	Hyperlink	Navigate to the account registration screen	Click to open the Register page	No

Table 22: Analysis of elements and actions in the *Login* form

4.1.2 Register

Allows users to create a new account by providing required information.

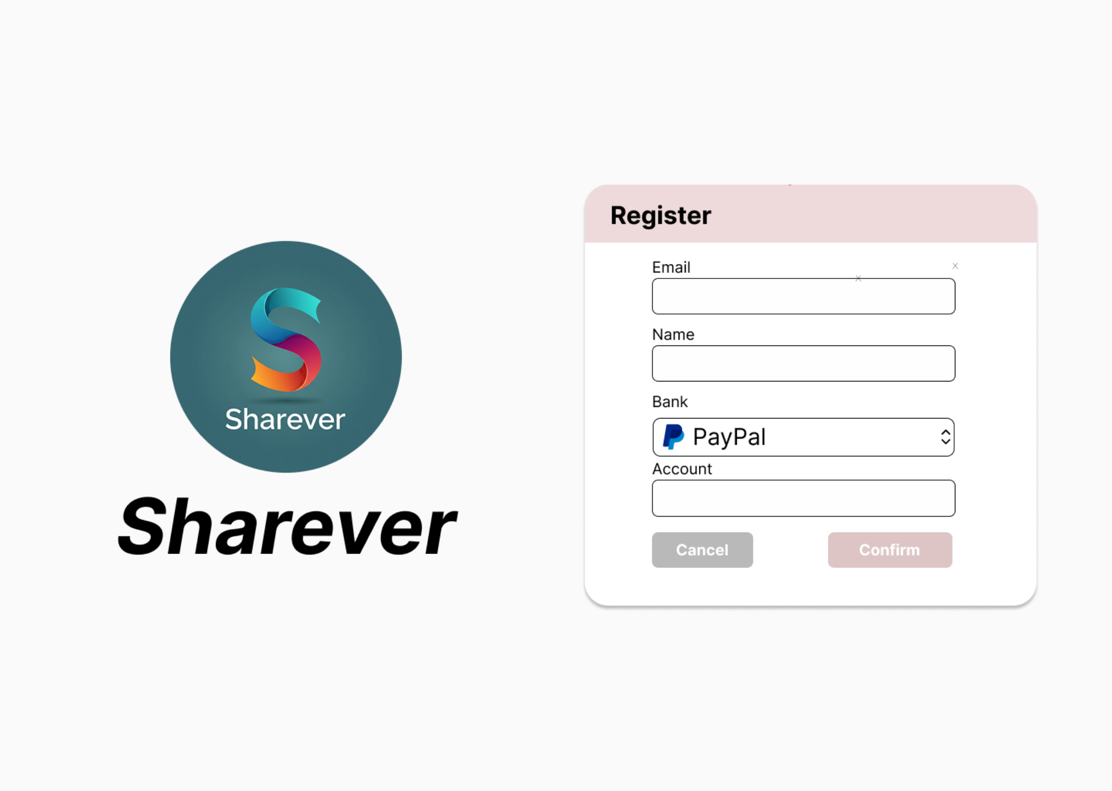


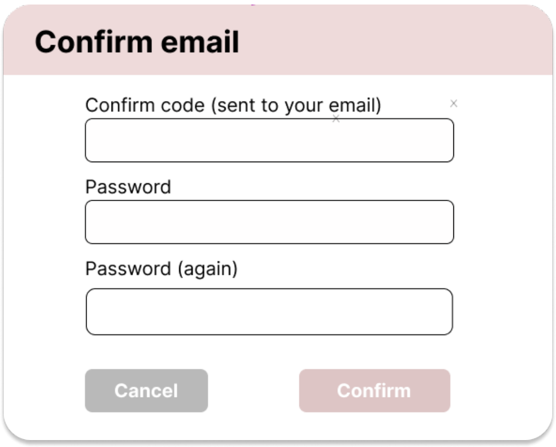
Figure 4: Register

No.	Element	Type	Function	User action	Required
1	Email field	Text input	Enter user email for registration	Type / edit / delete email	Yes
2	Clear (x) in Email	Icon button	Quickly clear content of Email field	Click to clear entered email	No
3	Name field	Text input	Enter user's display name	Type / edit / delete name	Yes
4	Bank selector	Select (dropdown)	Choose payment service or bank (e.g. PayPal)	Open list and choose one option	Yes
5	Account field	Text input	Enter account number / payment account	Type / edit / delete account	Yes
6	Cancel button	Button (secondary)	Cancel registration process	Click to close form without saving	No
7	Confirm button	Button (primary)	Submit registration information to server	Click to confirm and create account	Yes

Table 23: Analysis of elements in the *Register* form

4.1.3 Verification Email

Sends a verification email to confirm the user's email address during registration or login.



Confirm email

Confirm code (sent to your email)

Password

Password (again)

Figure 5: Verification email

No.	Element	Type	Function	User action	Required
1	Confirm code field	Text input	Enter verification code sent to user's email	Type / edit / delete code	Yes
2	Clear (x) in code	Icon button	Quickly clear the Confirm code field	Click to clear the entered code	No
3	Password field	Password input	Define password for the new account	Type / edit / delete password	Yes
4	Password (again) field	Password input	Re-enter password to confirm it matches	Type same password as above	Yes
5	Cancel button	Button (secondary)	Cancel email verification step	Click to close form and go back	No
6	Confirm button	Button (primary)	Submit verification code and password	Click to complete account activation	Yes

Table 24: Analysis of elements in the *Verification email* form

4.2 Dashboard

4.2.1 Home

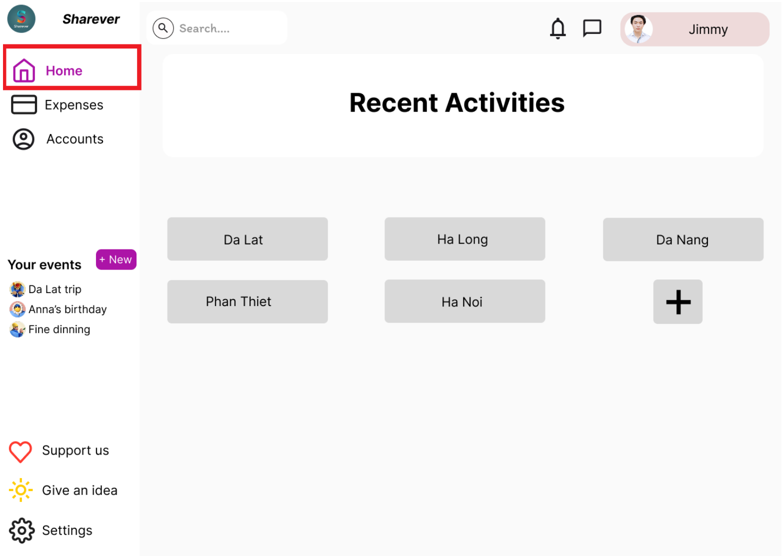
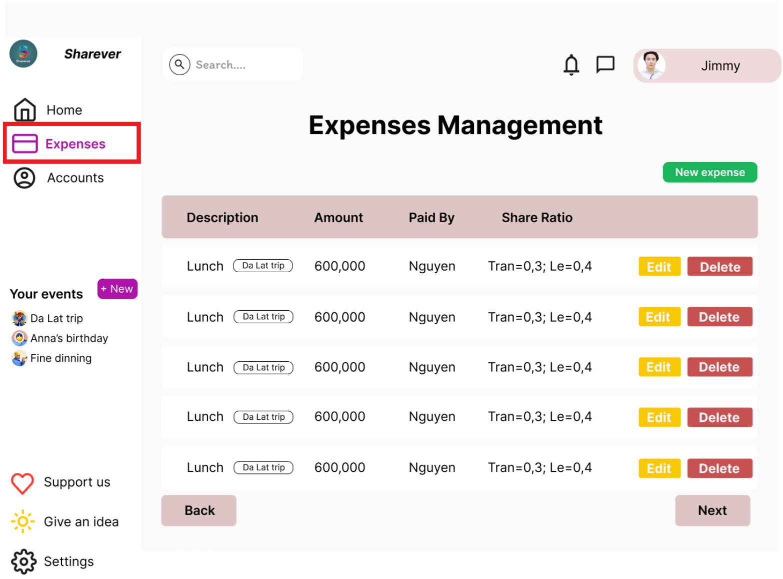


Figure 6: Home Page After Login

No.	Element	Type	Function	User action	Required
1	Left navigation	Navigation menu	Navigate between Home, Expenses, Accounts, Settings, etc.	Click menu items to switch pages	No
2	Home item (highlighted)	Navigation item	Indicates current page is the Home screen	Click to return to Home from other pages	No
3	Search bar	Text input	Search for events / activities by name or keyword	Type keyword to filter results	No
4	Notification / chat icons	Icon buttons	Access notifications or messages related to events	Click corresponding icon to open notification center	No
5	User profile menu	Button / avatar	Show current logged-in account and profile options	Click avatar to open account menu	No
6	“Recent activities” area	Content section	Show shortcuts to recently accessed events	Click an activity tile to open that event	No
7	Event tiles (Da Lat, Ha Long, . . .)	Button tiles	Quick access to specific events from Home screen	Click tile to open event overview	No
8	“Your events” list	Sidebar list	List all user events in a compact navigation block	Click event name to open that event	No

Table 25: Analysis of elements on the *Home page after login*

4.2.2 Expenses



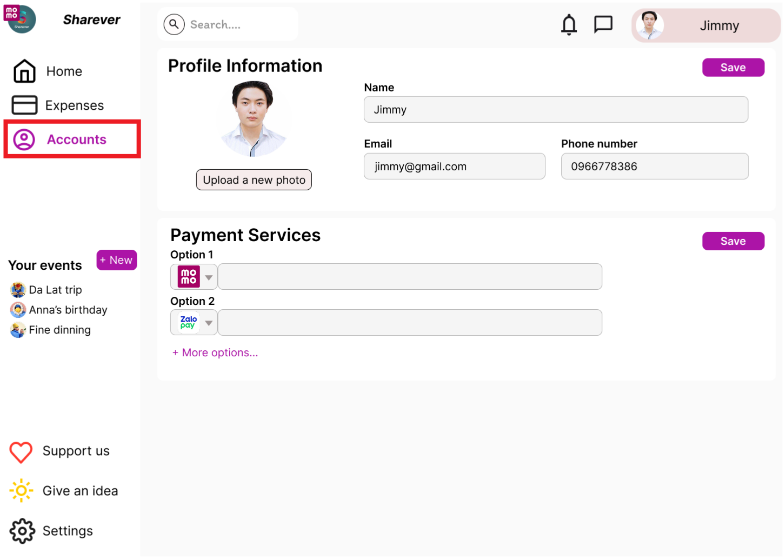
Description	Amount	Paid By	Share Ratio	
Lunch Da Lat trip	600,000	Nguyen	Tran=0,3; Le=0,4	Edit Delete
Lunch Da Lat trip	600,000	Nguyen	Tran=0,3; Le=0,4	Edit Delete
Lunch Da Lat trip	600,000	Nguyen	Tran=0,3; Le=0,4	Edit Delete
Lunch Da Lat trip	600,000	Nguyen	Tran=0,3; Le=0,4	Edit Delete
Lunch Da Lat trip	600,000	Nguyen	Tran=0,3; Le=0,4	Edit Delete

Figure 7: All Expenses

No.	Element	Type	Function	User action	Required
1	Left navigation	Navigation menu	Switch between Home, Expenses, Accounts	Click "Expenses" to open this screen	No
2	Page title "Expenses Management"	Heading	Clarify that this screen manages all expenses	Read-only context information	N/A
3	Expenses table	Table	Show list of expenses (Description, Amount, Paid by, Share ratio)	Scroll and review each row	Yes (to manage expenses)
4	Edit button in table	Button	Modify an existing expense row	Click in a specific row to open Edit expense pop-up	No
5	Delete button in table	Button (danger)	Remove an expense from the event	Click Delete then confirm (if needed)	No
6	"New expense" button	Button (primary)	Create a new expense entry	Click to open New expense pop-up	No
7	Back / Next buttons	Navigation buttons	Navigate between steps or pages in the flow	Click Back or Next to move between screens	No
8	"Your events" list	Sidebar list	Quickly switch to another event whose expenses you want to view	Click another event name	No

Table 26: Analysis of elements on the *All expenses* screen

4.2.3 Accounts



Profile Information

Name: Jimmy

Email: jimmy@gmail.com

Phone number: 0966778386

Payment Services

Figure 8: Account Information

No.	Element	Type	Function	User action	Required
1	Left navigation	Navigation menu	Switch to the Accounts page	Click "Accounts" to open profile settings	No
2	Profile picture	Image	Display current avatar of the user	View or change via related button	No
3	"Upload a new photo" button	Button	Change user's profile picture	Click to choose and upload a new photo	No
4	Name field	Text input	Store user's full name shown in the app	Type / edit / delete name	Yes
5	Email field	Text input	Store contact email for the account	Type / edit / delete email address	Yes
6	Phone number field	Text input	Save user's phone number	Type / edit / delete phone number	Optional
7	Profile Save button	Button (primary)	Save changes in profile information block	Click to update profile on server	Yes (when changes made)
8	Payment services table	Form section	Configure multiple payment services (MoMo, ZaloPay, etc.)	Fill in options and accounts for each row	Optional
9	Payment Save button	Button (primary)	Save configuration of payment services	Click to store services and accounts	Yes (when edited)
10	"More options..." link	Text link	Add more payment service rows if needed	Click to expand and add extra options	No

Table 27: Analysis of elements on the *Account information* screen

4.2.4 Event List

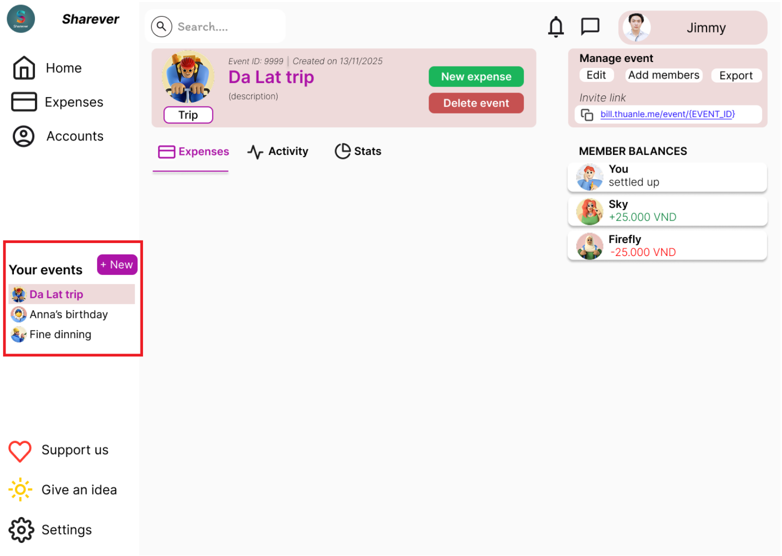


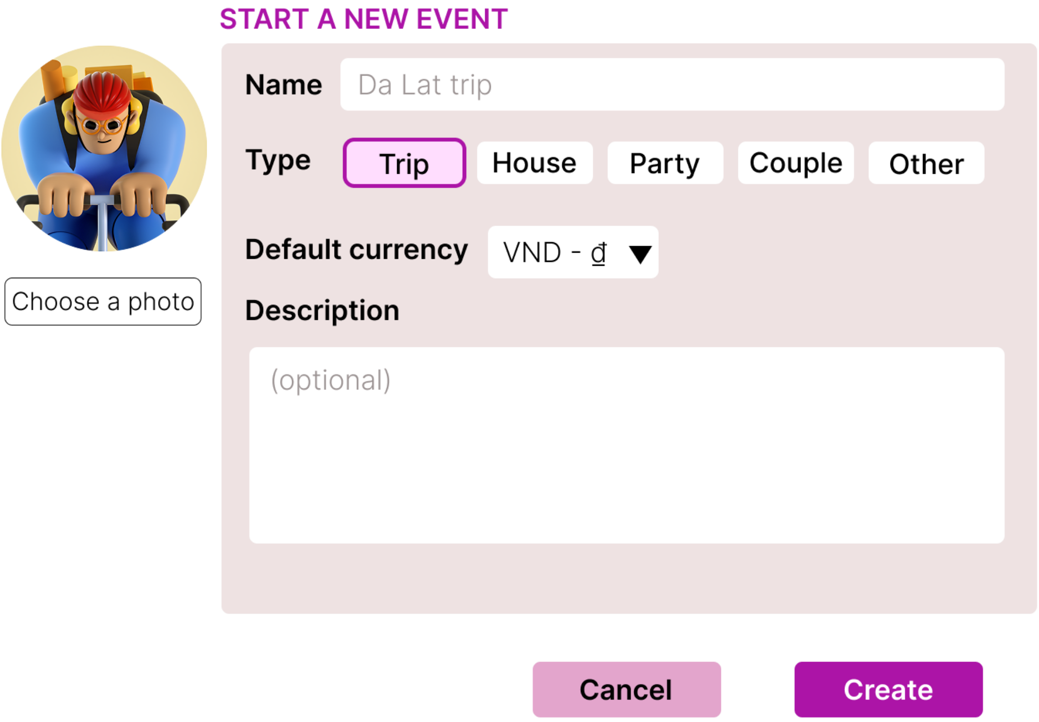
Figure 9: Access Event View

No.	Element	Type	Function	User action	Required
1	“Your events” list	Sidebar list	Show all events; select one to access its details	Click event (e.g. Da Lat trip) to open it	Yes (to pick event)
2	Event header card	Info card	Show event name, type, metadata, and quick actions	Read details or click buttons inside	N/A
3	“New expense” button	Button (primary)	Add a new expense to current event	Click to open New expense pop-up	No
4	“Delete event” button	Button (danger)	Remove the event and all its data	Click to trigger delete flow	No
5	Tabs Expenses / Activity / Stats	Tab control	Switch between expense list, activity log, and statistics	Click a tab to change view	Yes (to view other info)
6	Manage event panel	Action group	Provide actions: Edit, Add members, Export, Invite link	Click corresponding button for desired management task	No
7	Member balances panel	Info panel	Show how much each member owes or should receive	Read balances to understand current settlement state	N/A

Table 28: Analysis of elements in the *Access event view* screen

4.3 Event Management

4.3.1 Create Event



START A NEW EVENT

Name: Da Lat trip

Type: Trip | House | Party | Couple | Other

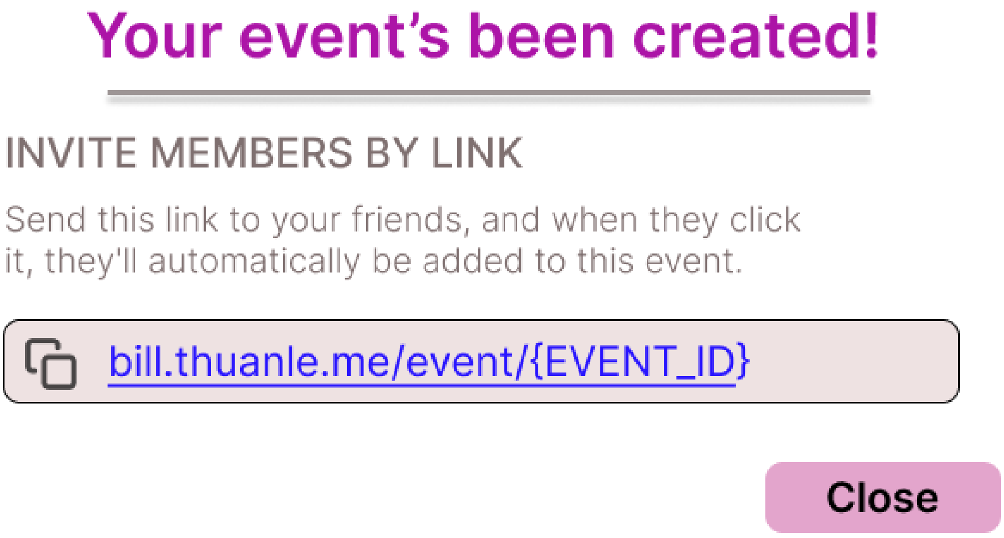
Default currency: VND - ₫ ▼

Description: (optional)

Figure 10: Create Event

No.	Element	Type	Function	User action	Required
1	Event avatar	Image	Display avatar for new event	Optionally change via “Choose a photo”	No
2	“Choose a photo” button	Button	Select picture to use as event avatar	Click to open file picker or gallery	No
3	Name field	Text input	Enter event name (e.g. Da Lat trip)	Type / edit / delete name	Yes
4	Type selector (Trip, House, . . .)	Segmented control	Choose event category (trip, house, party, couple, other)	Click one of the type buttons	Yes
5	Default currency selector	Select (dropdown)	Choose primary currency for the event (VND, USD, . . .)	Open dropdown and select currency	Yes
6	Description field	Textarea	Optional description or notes about event	Type multi-line text or leave empty	No
7	Cancel button	Button (secondary)	Cancel creation of the new event	Click to discard information and close	No
8	Create button	Button (primary)	Create event with current information	Click to submit and create event	Yes

Table 29: Analysis of elements in the *Create event* screen



Your event's been created!

INVITE MEMBERS BY LINK

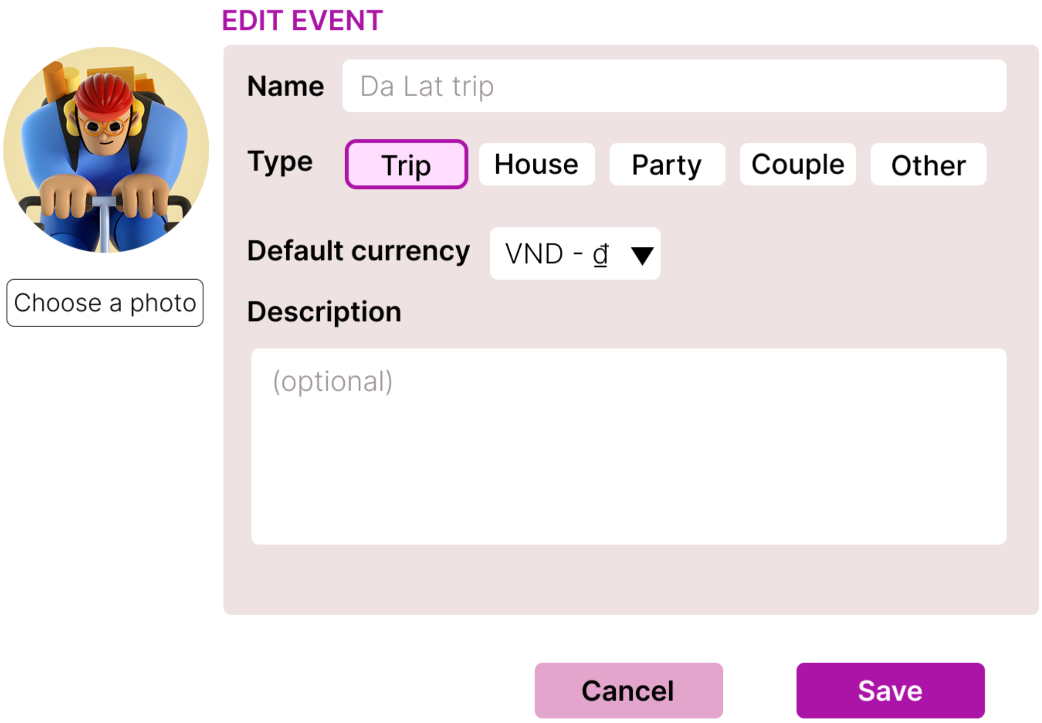
Send this link to your friends, and when they click it, they'll automatically be added to this event.

Figure 11: Pop-up Invite Members By Link

No.	Element	Type	Function	User action	Required
1	Success message	Text heading	Inform user that the event has been created	Read-only context	N/A
2	Instruction text	Paragraph	Explain how to share link so friends can join event	Read to understand share flow	N/A
3	Invite link field	Read-only input	Contain URL that adds users to event when they click	Select and copy link, share via any channel	Yes
4	Copy icon (if any)	Icon button	Quick copy of invite URL to clipboard	Click to copy without manual selection	No
5	Close button	Button (primary)	Close pop-up after user copied/share link	Click to return to previous screen	No

Table 30: Analysis of elements in the pop-up *Invite members by link*

4.3.2 Edit Events



EDIT EVENT

Name: Da Lat trip

Type: [✓ Trip] House Party Couple Other

Default currency: VND - ₫ ▼

Description: (optional)

Figure 12: Edit Event

No.	Element	Type	Function	User action	Required
1	Event avatar	Image	Show current event avatar	Optionally change via “Choose a photo”	No
2	Name field	Text input	Edit event name	Modify existing text	Yes
3	Type selector	Segmented control	Change event category (Trip, House, Party, . . .)	Click another type button	Yes
4	Default currency selector	Select (dropdown)	Change default currency used in event	Choose another currency	Yes
5	Description field	Textarea	Update description / notes for event	Edit or clear existing text	No
6	Cancel button	Button (secondary)	Cancel editing and keep previous event info	Click to discard changes	No
7	Save button	Button (primary)	Save updated event information	Click to apply changes	Yes

Table 31: Analysis of elements in the *Edit event* screen

4.3.3 Participants Management

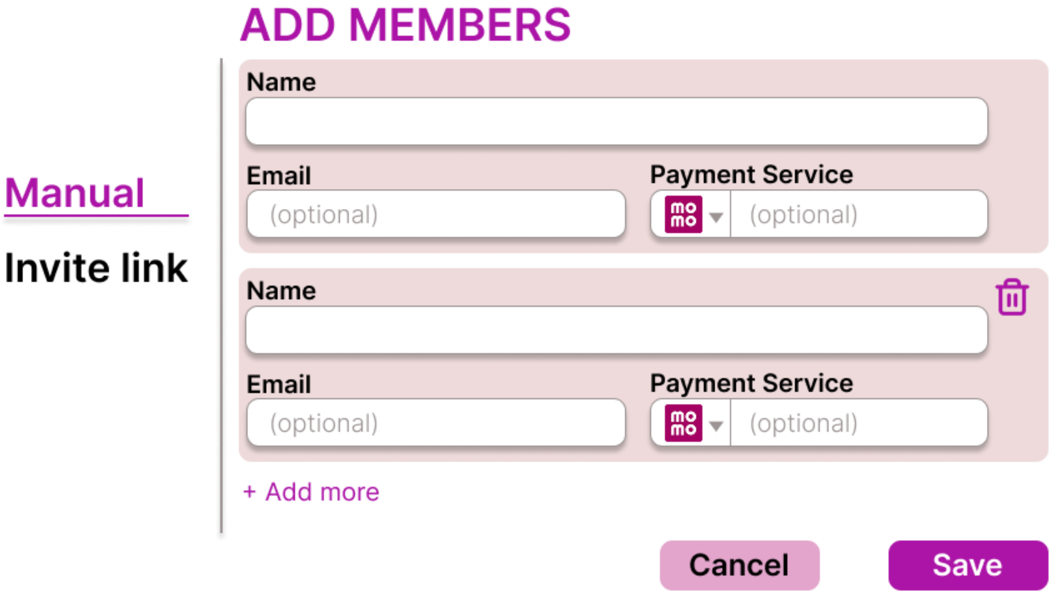
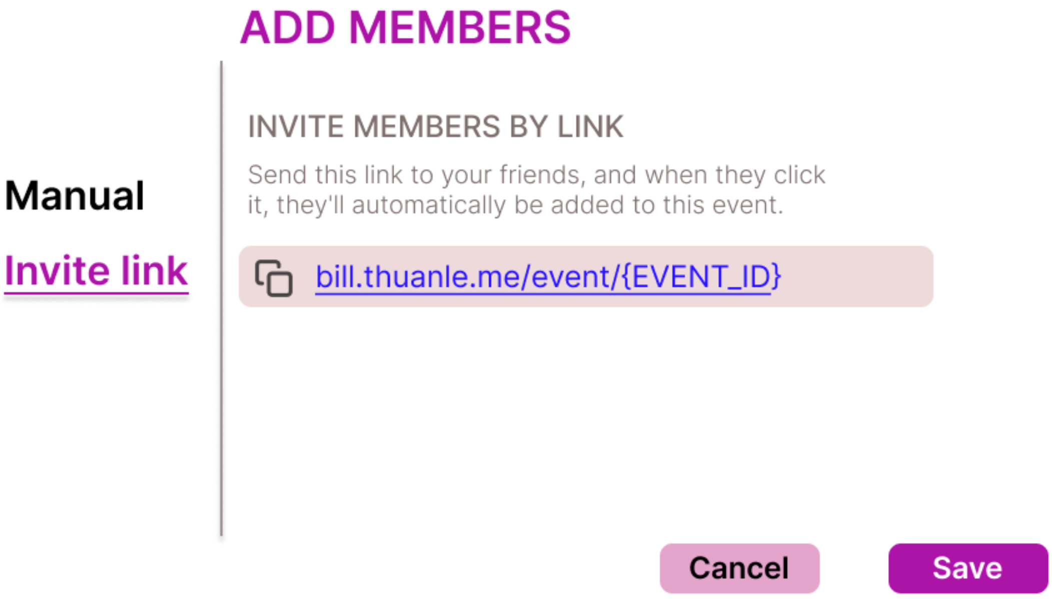


Figure 13: Add Participants Manually

No.	Element	Type	Function	User action	Required
1	Tabs "Manual / Invite link"	Tab control	Choose method to add participants (manual or by link)	Click "Manual" tab to use this screen	Yes
2	Name field (each row)	Text input	Enter participant name for the event	Type / edit / delete name	Yes
3	Email field	Text input	Optional contact email of participant	Type email or leave empty	No
4	Payment service selector	Select (dropdown)	Choose payment service (e.g. MoMo) used by participant	Select service from dropdown	Optional
5	Delete row icon	Icon button	Remove this participant from list	Click trash icon to delete row	No
6	"+ Add more" link	Text link	Add another participant row	Click to append new empty row	No
7	Cancel button	Button (secondary)	Cancel adding / editing participants manually	Click to close dialog without saving	No
8	Save button	Button (primary)	Save participants list for the event	Click to persist all entered members	Yes

Table 32: Analysis of elements in *Add participants manually*



Send this link to your friends, and when they click it, they'll automatically be added to this event.

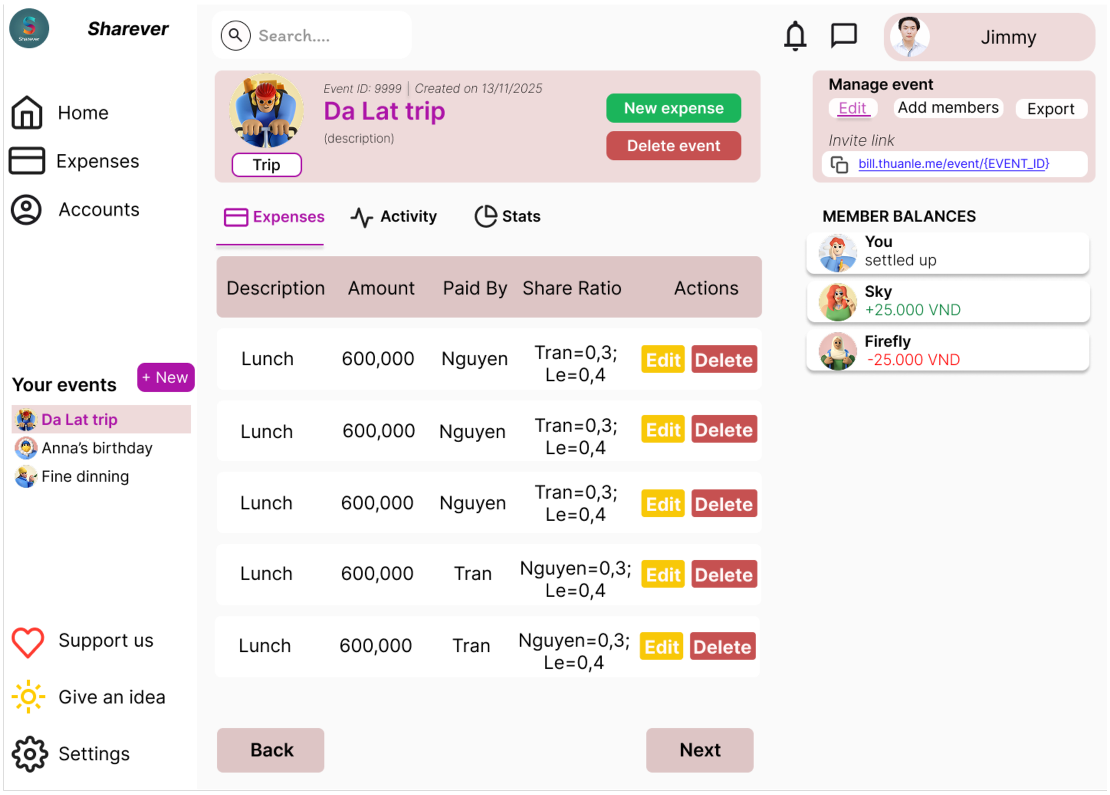
bill.thuanle.me/event/{EVENT_ID}

Figure 14: Add Participants Automatically By Link

No.	Element	Type	Function	User action	Required
1	Tabs “Manual / Invite link”	Tab control	Let user choose to add members manually or by invite link	Click “Invite link” tab to open this view	Yes
2	Heading “INVITE MEMBERS BY LINK”	Text heading	Explain that members will join automatically via link	Read-only information	N/A
3	Instruction text	Paragraph	Describe that friends clicking the link will be added to the event	Read instructions before sharing	N/A
4	Invite link field	Read-only input	Contains event invite URL for automatic joining	Select and copy the URL	Yes
5	Copy icon	Icon button	Quickly copy invite link to clipboard	Click icon to copy link	No
6	Cancel button	Button (secondary)	Cancel adding members via link and close dialog	Click to go back without changes	No
7	Save button	Button (primary)	Confirm that invite link configuration is done	Click after copying / sharing the link	No

Table 33: Analysis of elements in *Add participants automatically by link*

4.4 Expense Management



Sharever

Search....

Jimmy

Home

Expenses

Accounts

Event ID: 9999 | Created on 13/11/2025

Da Lat trip

(description)

Trip

New expense

Delete event

Manage event

Edit Add members Export

Invite link

bill.thuanle.me/event/{EVENT_ID}

Expenses Activity Stats

Description	Amount	Paid By	Share Ratio	Actions

Lunch	600,000	Nguyen	Tran=0,3;
			Le=0,4
				Edit Delete

Lunch	600,000	Nguyen	Tran=0,3;
			Le=0,4
				Edit Delete

Lunch	600,000	Nguyen	Tran=0,3;
			Le=0,4
				Edit Delete

Lunch	600,000	Tran	Nguyen=0,3;
			Le=0,4
				Edit Delete

Lunch	600,000	Tran	Nguyen=0,3;
			Le=0,4
				Edit Delete

MEMBER BALANCES

You
settled up

Sky
+25.000 VND

Firefly
-25.000 VND

Your events + New

Da Lat trip

Anna's birthday

Fine dinning

Support us

Give an idea

Settings

Back

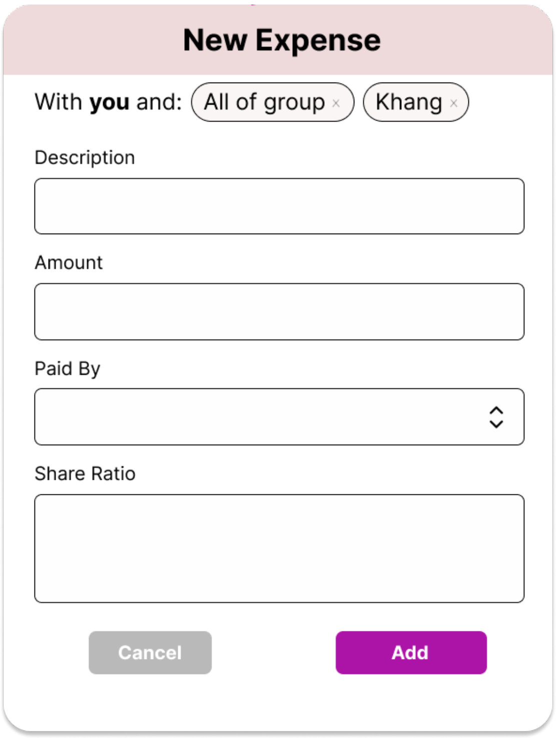
Next

Figure 15: Expenses Within An Event

No.	Element	Type	Function	User action	Required
1	Left navigation	Navigation menu	Switch between Home, Expenses, Accounts	Click menu items to change section	No
2	“Your events” list	Sidebar list	Show all events belonging to the user	Click an event (e.g. Da Lat trip) to open its details	Yes
3	Event header card	Info card	Display event name, icon, dates and quick actions	Read info or click inside buttons	N/A
4	Tabs Expenses / Activity / Stats	Tab control	Change current view within the event	Click “Expenses” tab to see expense list	Yes
5	Expenses table	Table	Show rows of expenses (Description, Amount, Paid By, Share Ratio)	Scroll and review each row	Yes
6	Edit button (per row)	Button	Open pop-up to edit selected expense	Click “Edit” in corresponding row	No
7	Delete button (per row)	Button (danger)	Remove selected expense from the event	Click “Delete” then confirm	No
8	Back / Next buttons	Navigation buttons	Move backward or forward in the event flow	Click “Back” or “Next”	No
9	Manage event panel	Action group	Provide actions: Edit, Add members, Export, Invite link	Click appropriate button to manage event	No
10	Member balances panel	Info panel	Summarize each member’s balance (owe / receive)	Read balances to know who should pay or receive	N/A

Table 34: Analysis of elements in *Expenses within an event*

4.4.1 Add New Expense



New Expense

With **you** and: All of group × Khang ×

Description

Amount

Paid By

Share Ratio

Figure 16: Pop-up New Expense

No.	Element	Type	Function	User action	Required
1	Chips "With you and . . . "	Token selector	Select who participates in this expense (all group / specific people)	Click tokens to add or remove participants	Yes
2	Description field	Text input	Enter short description of the expense	Type / edit / delete text	Yes
3	Amount field	Number input	Enter amount of the expense	Type numeric value	Yes
4	Paid By selector	Select (dropdown)	Choose who actually paid for the expense	Open dropdown and select one member	Yes
5	Share Ratio field	Textarea / grid	Define how the amount is split among participants	Enter ratios or notes on splitting rule	Yes
6	Cancel button	Button (secondary)	Cancel creation of new expense	Click to close pop-up without saving	No
7	Add button	Button (primary)	Save new expense into the event	Click to submit the form	Yes

Table 35: Analysis of elements in the pop-up *New expense*

4.4.2 Edit Expense

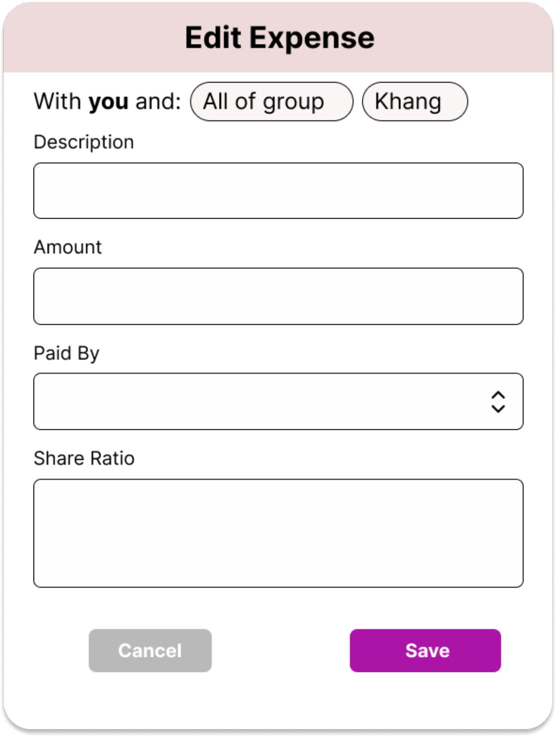
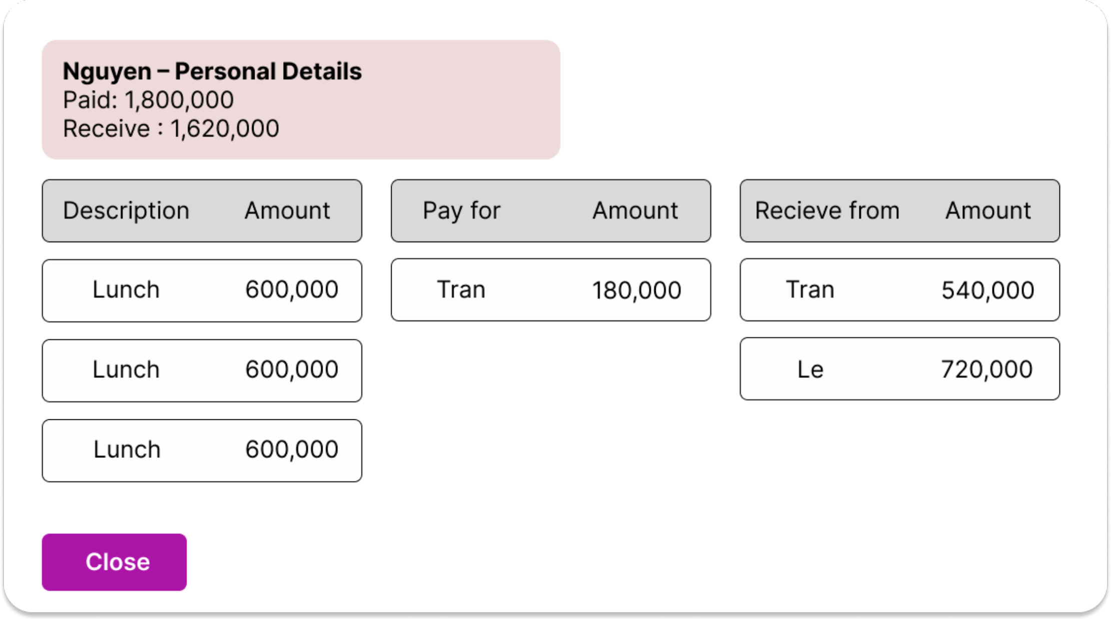


Figure 17: Pop-up Editing Expense

No.	Element	Type	Function	User action	Required
1	Chips "With you and . . . "	Token selector	Show / update which members share this expense	Add or remove participants via chips	Yes
2	Description field	Text input	Edit description of existing expense	Modify text content	Yes
3	Amount field	Number input	Change amount of the expense	Edit numeric value	Yes
4	Paid By selector	Select (drop-down)	Update payer of the expense	Choose different member if needed	Yes
5	Share Ratio field	Textarea / grid	Edit rules/ratios of how cost is split	Update ratios or notes	Yes
6	Cancel button	Button (secondary)	Cancel editing and keep old data	Click to close pop-up without saving	No
7	Save button	Button (primary)	Save changes to the expense	Click to apply modifications	Yes

Table 36: Analysis of elements in the pop-up *Editing expense*

4.4.3 Expense Summary Of Participant



Nguyen – Personal Details
Paid: 1,800,000
Receive : 1,620,000

Description	Amount
Lunch	600,000
Lunch	600,000
Lunch	600,000

Pay for	Amount
Tran	180,000

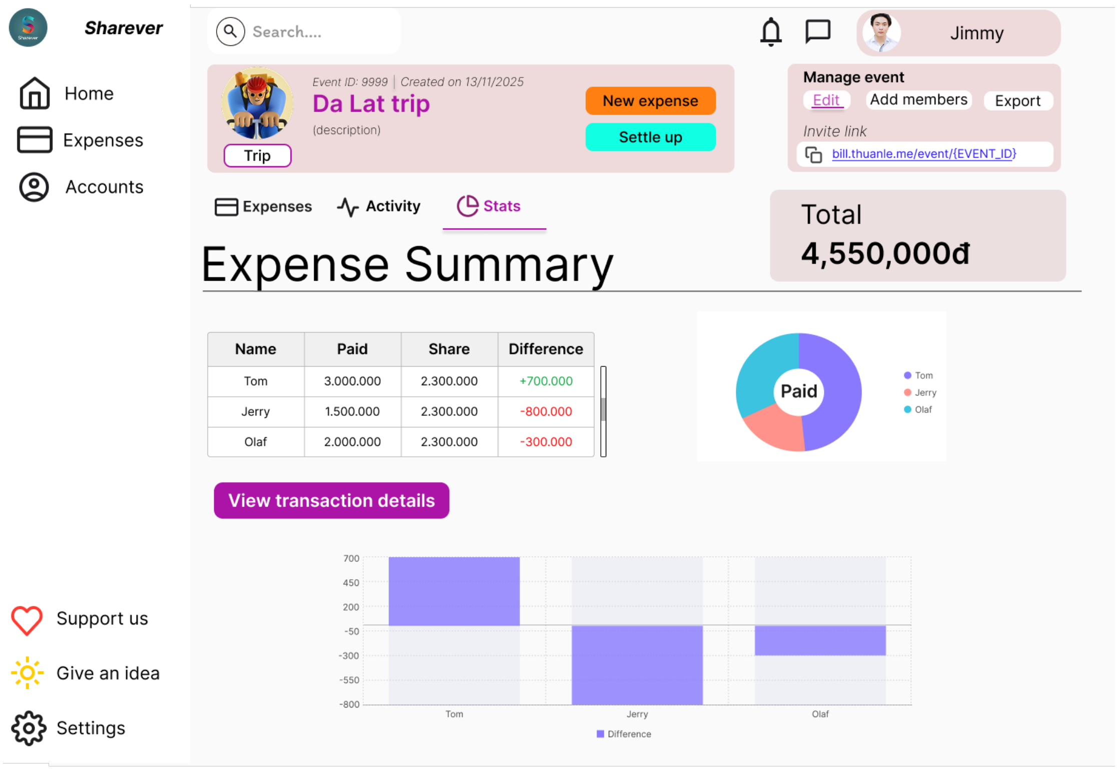
Recieve from	Amount
Tran	540,000
Le	720,000

Figure 18: Pop-up Expense Summary Of Participant

No.	Element	Type	Function	User action	Required
1	Personal details header	Info panel	Show participant name, total amount paid and total to receive	Read summary before checking rows below	N/A
2	Column "Description – Amount"	Table column	List all expenses this person is involved in	Review each item they joined	Indirect
3	Column "Pay for – Amount"	Table column	Show amounts this person paid on behalf of others	Understand how much others owe them	Indirect
4	Column "Receive from – Amount"	Table column	Show amounts this person will receive from others	See who will pay them and how much	Indirect
5	Close button	Button (primary)	Close participant summary pop-up	Click to go back to previous screen	No

Table 37: Analysis of elements in the pop-up *Expense summary of participant*

4.4.4 Expense Summary Of Event



Event ID: 9999 | Created on 13/11/2025

Da Lat trip

(description)

Invite link

bill.thuanle.me/event/{EVENT_ID}

Total

4,550,000đ

Expense Summary

Name	Paid	Share	Difference
Tom	3.000.000	2.300.000	+700.000
Jerry	1.500.000	2.300.000	-800.000
Olaf	2.000.000	2.300.000	-300.000

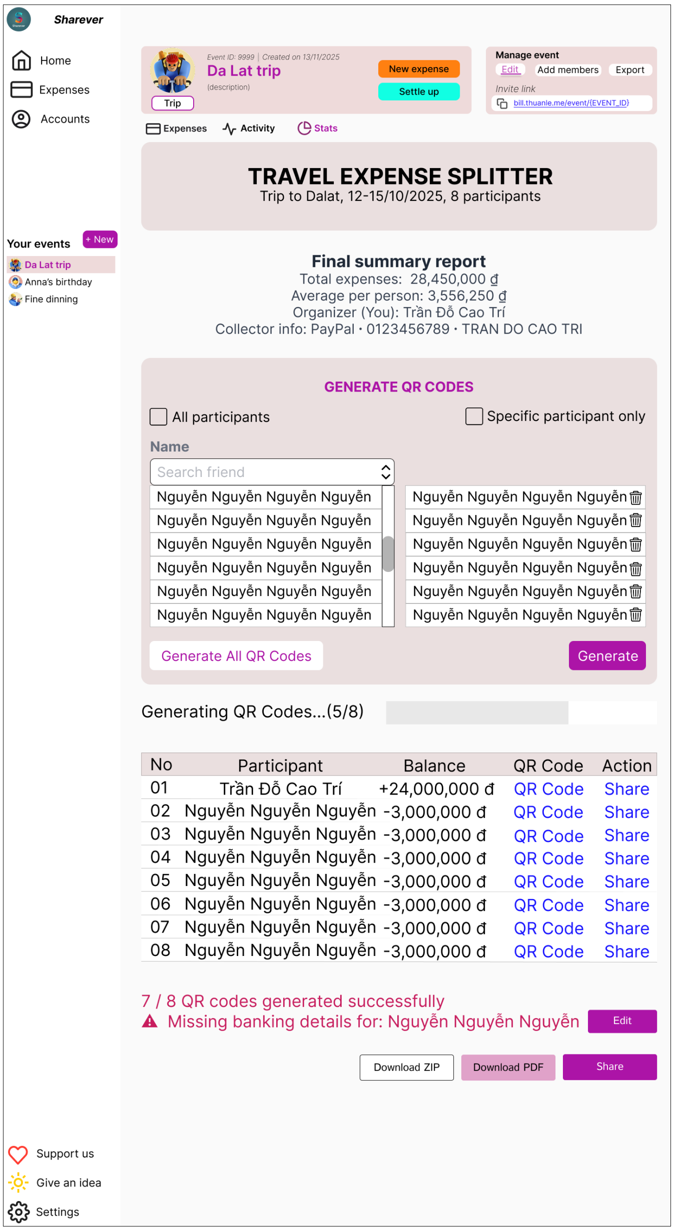
Figure 19: Expense Summary Of Event

No.	Element	Type	Function	User action	Required
1	Event header	Info card	Show event name, metadata and quick actions (New expense, Settle up)	Optionally click buttons to create expense or settle	No
2	Tabs Expenses / Activity / Stats	Tab control	Switch between event sub-views	Select "Stats" to open summary view	Yes
3	Total card	Summary card	Display total expense amount for event	Read-only; gives overview of total	N/A
4	Expense summary table	Table	For each member, show Paid, Share and Difference	Inspect who owes or should receive how much	Yes
5	"View transaction details" button	Button	Open detailed list of transactions behind the summary	Click to navigate to detailed view	No
6	Pie chart "Paid"	Chart (pie)	Visualize distribution of amounts paid by each member	Look at chart for quick comparison	Indirect
7	Bar chart "Difference"	Chart (bar)	Visualize final difference per member	Observe who is positive/negative	Indirect

Table 38: Analysis of elements in *Expense summary of event*

4.5 Payment Management

4.5.1 Generate Payment QR Code



TRAVEL EXPENSE SPLITTER

Trip to Dalat, 12-15/10/2025, 8 participants

Final summary report

Total expenses: 28,450,000 ₫
Average per person: 3,556,250 ₫
Organizer (You): Trần Đỗ Cao Trí
Collector info: PayPal · 0123456789 · TRAN DO CAO TRI

GENERATE QR CODES

□ All participants □ Specific participant only

Name: Search friend

Nguyễn Nguyễn Nguyễn Nguyễn	Nguyễn Nguyễn Nguyễn Nguyễn
Nguyễn Nguyễn Nguyễn Nguyễn	Nguyễn Nguyễn Nguyễn Nguyễn
Nguyễn Nguyễn Nguyễn Nguyễn	Nguyễn Nguyễn Nguyễn Nguyễn
Nguyễn Nguyễn Nguyễn Nguyễn	Nguyễn Nguyễn Nguyễn Nguyễn
Nguyễn Nguyễn Nguyễn Nguyễn	Nguyễn Nguyễn Nguyễn Nguyễn
Nguyễn Nguyễn Nguyễn Nguyễn	Nguyễn Nguyễn Nguyễn Nguyễn

Generating QR Codes...(5/8)

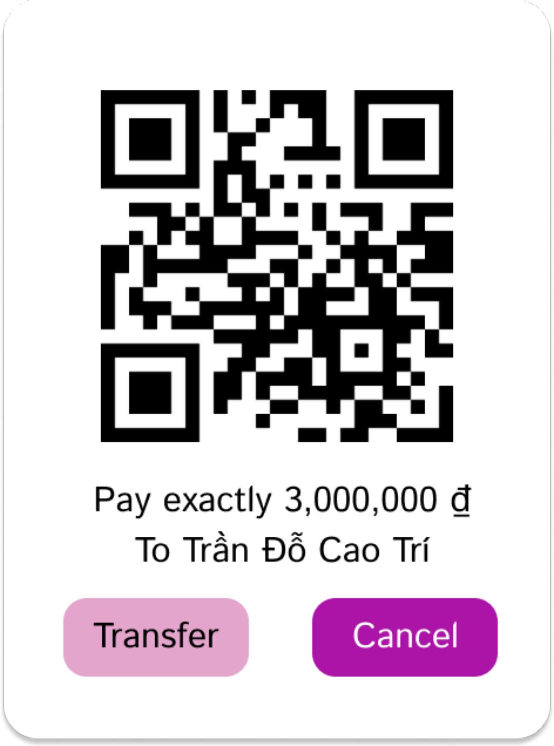
No	Participant	Balance	QR Code	Action
01	Trần Đỗ Cao Trí	+24,000,000 đ	QR Code	Share
02	Nguyễn Nguyễn Nguyễn	-3,000,000 đ	QR Code	Share
03	Nguyễn Nguyễn Nguyễn	-3,000,000 đ	QR Code	Share
04	Nguyễn Nguyễn Nguyễn	-3,000,000 đ	QR Code	Share
05	Nguyễn Nguyễn Nguyễn	-3,000,000 đ	QR Code	Share
06	Nguyễn Nguyễn Nguyễn	-3,000,000 đ	QR Code	Share
07	Nguyễn Nguyễn Nguyễn	-3,000,000 đ	QR Code	Share
08	Nguyễn Nguyễn Nguyễn	-3,000,000 đ	QR Code	Share

7 / 8 QR codes generated successfully
⚠ Missing banking details for: Nguyễn Nguyễn Nguyễn

Figure 20: Generate Payment QR Code

No.	Element	Type	Function	User action	Required
1	Event header and manage panel	Info/Action group	Keep context of current event and give quick actions	Click Edit, Add members, Export, Invite link if needed	No
2	"Final summary report" block	Summary panel	Show total expenses, average per person and collector info	Read to verify numbers before generating QR	N/A
3	Checkbox "All participants"	Checkbox	Generate QR codes for all members at once	Tick to choose mass generation mode	No
4	Checkbox "Specific participant only"	Checkbox	Limit QR generation to selected participants	Tick and select participants from lists	No
5	Name list (left)	Multi-select list	Choose which participants to generate QR for	Select one or multiple names	Yes (in specific mode)
6	Corresponding QR/receiver list (right)	Multi-select list	Map participants to which balances/QR they receive	Review mapping before generating	Indirect
7	"Generate All QR Codes" button	Button	Generate QR for everyone in a single step	Click when All participants mode is selected	No
8	"Generate" button	Button (primary)	Generate QR according to current selection	Click to start generation process	Yes
9	Progress / status text	Status indicator	Inform how many QR codes finished, and any missing info	Read to know success/failure per participant	N/A
10	Result table (No, Participant, Balance, QR Code, Action)	Table	Show each participant's balance and QR-related actions	Click "QR Code" or "Share" for each row	Yes
11	Download ZIP / Download PDF / Share buttons	Button group	Export generated QR codes or share them	Choose one of the export / sharing actions	No
12	Edit button in warning row	Button	Fix missing banking details for a specific participant	Click to open collector/-payment info editing	No

Table 39: Analysis of elements in *Generate payment QR code*

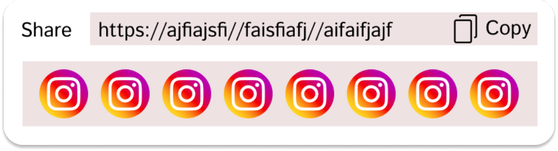


Pay exactly 3,000,000 đ
To Trần Đỗ Cao Trí

Figure 21: Pop-up Generated QR Code

No.	Element	Type	Function	User action	Required
1	QR code image	Image	Encodes payment information (receiver and exact amount)	Scan with banking/-payment app	Yes
2	Text "Pay exactly . . . To . . . "	Text label	Human-readable confirmation of amount and receiver	Check text matches expected payment	Indirect
3	Transfer button	Button (primary)	Start payment flow (if integrated) or confirm payment	Click after scanning / to proceed with payment	No
4	Cancel button	Button (secondary)	Close QR pop-up without performing payment	Click to return to previous screen	No

Table 40: Analysis of elements in pop-up *Generated QR code*



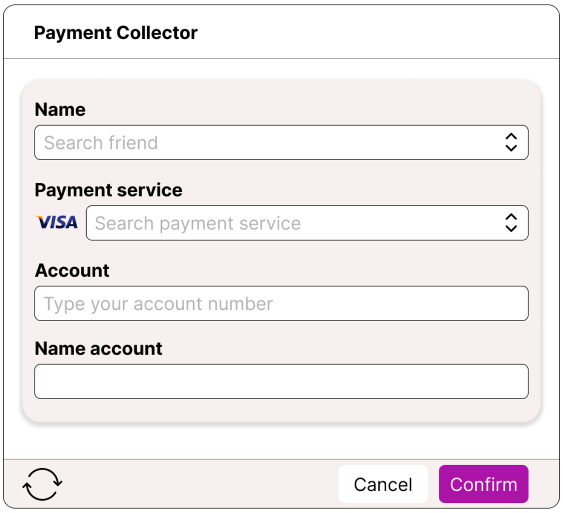
Share https://ajfiajsfi//faisfiafj//aifaifjajf Copy

Figure 22: Pop-up Sharing

No.	Element	Type	Function	User action	Required
1	Share label	Text label	Describe that the following URL is for sharing	Read-only	N/A
2	Share link field	Read-only input	Contain URL for event / report / QR page	Select and copy the link	Yes
3	Copy button	Icon button	Quickly copy share link to clipboard	Click to copy without selecting text	No
4	Row of social icons	Social buttons	Provide shortcuts to share via social networks (e.g. Instagram)	Click an icon to open corresponding app/site	No

Table 41: Analysis of elements in pop-up *Sharing*

4.5.2 Update Collector Information



Payment Collector

Name
Search friend

Payment service
VISA Search payment service

Account
Type your account number

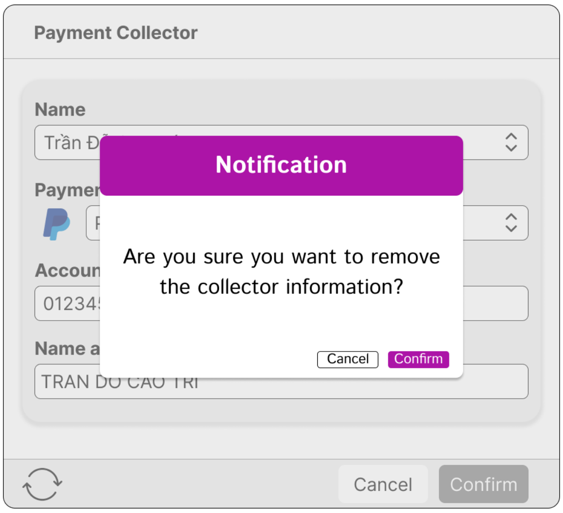
Name account

Cancel Confirm

Figure 23: Update Collector Information

No.	Element	Type	Function	User action	Required
1	Name selector	Select (searchable)	Choose which member is the collector (receiver of payments)	Search and select a friend from list	Yes
2	Payment service selector	Select (dropdown)	Choose payment service for the collector (Visa, PayPal, etc.)	Open list and choose service	Yes
3	Account field	Text input	Enter collector's account number / wallet identifier	Type / edit / delete account	Yes
4	Name account field	Text input	Enter legal name of account holder	Type / edit / delete account name	Yes
5	Refresh icon	Icon button	Reload or reset collector information (if implemented)	Click to refresh data	No
6	Cancel button	Button (secondary)	Cancel update of collector information	Click to close dialog without saving	No
7	Confirm button	Button (primary)	Save collector information for later payments	Click to store updated collector data	Yes

Table 42: Analysis of elements in *Update collector information*



01234

TRAN DO CAO TRI

Are you sure you want to remove the collector information?

Figure 24: Pop-up Confirmation to Cancel Update Collector Information

No.	Element	Type	Function	User action	Required
1	Notification title	Text heading	Inform user that this is an important confirmation dialog	Read-only context	N/A
2	Confirmation message	Paragraph text	Ask user whether they really want to remove collector information	Read message before choosing an action	Yes
3	Cancel button	Button (secondary)	Keep current collector information and close pop-up	Click to abort removal and return to previous screen	No
4	Confirm button	Button (primary / danger)	Confirm removal of existing collector information	Click to delete collector data and close pop-up	Yes

Table 43: Analysis of elements in pop-up *Confirmation to cancel update collector information*

4.6 Export Data

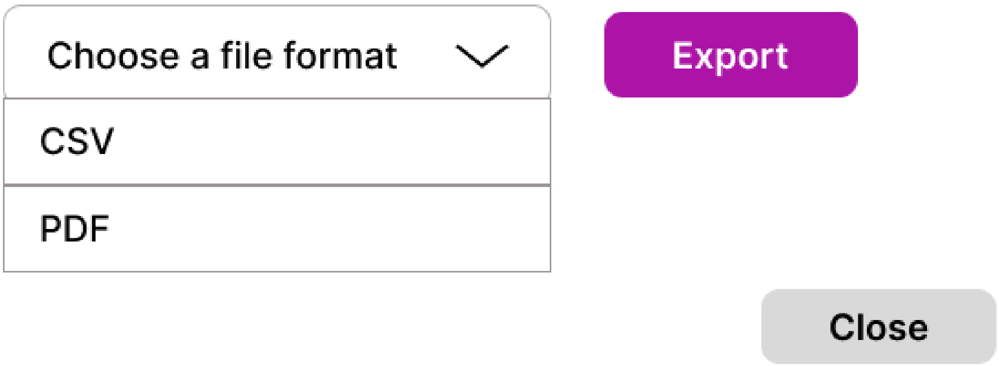


Figure 25: Export Data - File Type Selection

No.	Element	Type	Function	User action	Required
1	File format dropdown	Select (dropdown)	Let user choose type of exported file (CSV or PDF)	Open dropdown and select desired format	Yes
2	Option "CSV"	Dropdown option	Export data as CSV file for spreadsheet tools	Choose when user wants raw tabular data	Optional
3	Option "PDF"	Dropdown option	Export data as formatted PDF report	Choose when user wants printable report	Optional
4	Export button	Button (primary)	Start export process using selected file format	Click after choosing format to download file	Yes
5	Close button	Button (secondary)	Close export dialog without exporting any file	Click to cancel and return to previous screen	No

Table 44: Analysis of elements in *Export data – file type selection*

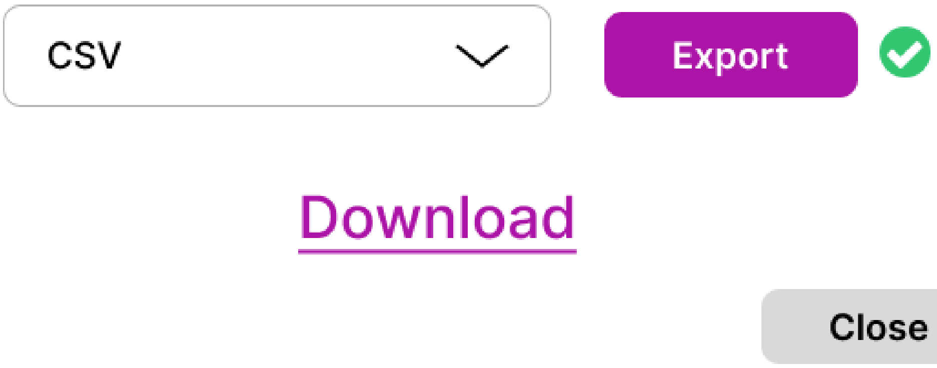


Figure 26: Export Data - Download Ready

No.	**Element**	**Type**	**Function**	**User action**	**Required**
1	File format dropdown	Select (dropdown)	Show selected export format (e.g. CSV) and allow changing it	Open dropdown to switch to another format if needed	Optional
2	Export button (with check icon)	Button (primary, success state)	Indicates that export has finished successfully for the chosen format	Read-only state; no further click required after success	N/A
3	Success check icon	Status icon	Visual confirmation that data export is ready to download	Observe icon to confirm export status	N/A
4	"Download" link	Hyperlink	Provide direct link to download exported file	Click to start downloading the generated export file	Yes
5	Close button	Button (secondary)	Close export dialog after user has downloaded (or skipped) file	Click to return to previous screen	No

Table 45: Analysis of elements in *Export data – download ready*