

]

# RoomEase

## High-Fidelity Prototype Report Milestone 3 - Group 25

November 17, 2025

*“Small Updates, Big Harmony.”*

## Contents

<b>1 Project Description</b>	<b>3</b>
1.1 Overview . . . . .	3
1.2 System Purpose . . . . .	3
1.3 Target Users . . . . .	3
<b>2 Technology Stack &amp; Implementation</b>	<b>3</b>
2.1 Core Technologies . . . . .	3
2.2 Design Approach . . . . .	3
2.3 Architecture . . . . .	4
<b>3 Major Features</b>	<b>4</b>
3.1 Horizontal Prototype Coverage . . . . .	4
3.2 Vertical Prototypes . . . . .	6
3.2.1 Vertical Prototype 1: Task Management Workflow . . . . .	6
3.2.2 Vertical Prototype 2: Expense Tracking & Balance Calculation . . . . .	7
<b>4 Design Highlights &amp; Innovation</b>	<b>7</b>
4.1 Visual Design Excellence . . . . .	7
4.2 Innovative Features . . . . .	8
4.3 Usability Enhancements . . . . .	8
<b>5 Interface Screenshots</b>	<b>8</b>
5.1 Home Dashboard . . . . .	9
5.2 Tasks Page . . . . .	10
5.3 Expenses Page . . . . .	11
5.4 Chat Interface . . . . .	12
5.5 Profile Page . . . . .	13

<b>6 Strengths of the System</b>	<b>13</b>
6.1 Comprehensive Feature Set . . . . .	13
6.2 Polished User Experience . . . . .	13
6.3 Technical Implementation . . . . .	14
6.4 Innovation & Creativity . . . . .	14
<b>7 Known Issues &amp; Limitations</b>	<b>14</b>
7.1 Current Limitations . . . . .	14
7.2 Design Trade-offs . . . . .	15
7.3 Future Enhancements . . . . .	15
<b>8 Testing &amp; Evaluation Readiness</b>	<b>15</b>
8.1 Usability Testing Support . . . . .	15
8.2 Testing Scenarios . . . . .	15
<b>9 Conclusion</b>	<b>16</b>

## Project Description

### Overview

**RoomEase** is a comprehensive roommate task management system designed to help students and adults living in shared accommodations manage household responsibilities, expenses, and communication effectively. The system addresses the common challenges of coordination, fairness, and accountability in shared living spaces through an intuitive, visually appealing interface.

### System Purpose

RoomEase serves as a centralized platform for:

- **Task Management:** Creating, assigning, and tracking household chores with support for recurring tasks and urgency levels
- **Expense Tracking:** Logging shared expenses, splitting costs fairly, and maintaining transparent financial records
- **Communication:** Facilitating roommate communication through integrated chat functionality
- **Fairness Monitoring:** Tracking contribution balance through dynamic fairness calculations and activity feeds

### Target Users

The primary users are university students and working adults (ages 18-40) living with 2-5 roommates in shared apartments, houses, or dorm-style accommodations. The system is designed to be accessible on both mobile devices and desktop computers, supporting users' on-the-go lifestyles.

## Technology Stack & Implementation

### Core Technologies

- **HTML5:** Semantic markup for structure and accessibility
- **CSS3:** Advanced styling with custom properties, Flexbox, Grid, and animations
- **Vanilla JavaScript (ES6+):** Client-side logic and DOM manipulation
- **LocalStorage API:** Client-side data persistence
- **Font Awesome 6.4.0:** Icon library for consistent visual elements
- **Google Fonts (Inter):** Modern, readable typography

### Design Approach

The prototype implements a **neumorphic design system** with:

- Soft shadows and subtle depth for a modern, tactile feel
- Consistent color palette with primary blue (#667eea) and accent purple (#764ba2)

- Smooth animations and transitions for enhanced user experience
- Full dark mode support with automatic theme detection
- Responsive design optimized for mobile-first, with full-screen desktop support

## Architecture

The application follows a single-page architecture with:

- **Modular JavaScript:** Functions organized by feature (tasks, expenses, chat, profile)
- **CSS Custom Properties:** Centralized theming and easy dark mode switching
- **LocalStorage Persistence:** All data stored client-side with automatic save/load
- **Event-Driven UI:** Real-time updates through event listeners and DOM manipulation

## Major Features

### Horizontal Prototype Coverage

The horizontal prototype demonstrates the complete system concept across five main sections:

#### 1. Home Dashboard

The home page provides a comprehensive overview of the shared living space:

- **Personalized Welcome:** Dynamic greeting based on time of day with user's name
- **Quick Stats Bar:** At-a-glance metrics for pending tasks, overdue items, pending expenses, and net balance
- **Today's To-Do Widget:** Focused view of tasks due today with empty state handling
- **Fairness Overview:** Dynamic circular progress indicator showing contribution balance (0-100%)
- **Interactive Calendar:** Monthly view with task indicators and clickable dates
- **Activity Feed:** Chronological feed of recent actions (tasks completed, expenses added, messages sent) grouped by time periods

## 2. Tasks Management

Complete task lifecycle management with:

- **CRUD Operations:** Create, read, update, and delete tasks with full form validation
- **Task Assignment:** Multi-select assignment to one or more roommates
- **Recurrence Support:** One-time, weekly, and monthly recurring tasks
- **Urgency Levels:** High, Moderate, and Low priority with visual indicators
- **Smart Filtering:** Filter by "My Tasks", "Today", "All Tasks", or "Completed"
- **Search Functionality:** Real-time search across task titles and descriptions
- **Status Tracking:** Visual indicators for overdue, today, pending, and completed states
- **Completion Celebration:** Confetti animation on task completion for positive reinforcement

## 3. Expenses Management

Comprehensive expense tracking and splitting:

- **Expense Logging:** Add expenses with description, amount, payer, and participants
- **Automatic Splitting:** Fair division of costs across selected roommates
- **Balance Calculation:** Real-time calculation of "You owe" and "You are owed" amounts
- **Net Balance Display:** Clear indication of overall financial position
- **Status Management:** Track expenses as "Pending" or "Settled"
- **Search & Filter:** Quick search across expense descriptions
- **Visual Indicators:** Color-coded status badges and amount highlighting

#### 4. Chat & Communication

Integrated messaging system:

- **Conversation List:** View all active conversations (1-on-1 and group chats)
- **Message Threading:** Organized message display with sender avatars and timestamps
- **Group Chats:** Support for multiple conversation types (general, bills, etc.)
- **Unread Badges:** Visual indicators for unread message counts
- **Responsive Layout:** Desktop split-view and mobile-optimized layouts
- **Real-time Updates:** Instant message display with smooth animations

#### 5. Profile & Settings

User profile and preferences:

- **User Statistics:** Tasks completed this week, house contribution percentage, current streak
- **Preferences:** Dark mode toggle, task reminders, expense updates, quiet hours
- **Roommate List:** View all roommates with avatars and initials
- **Theme Management:** Persistent theme selection with system preference detection

### Vertical Prototypes

#### 3.2.1 Vertical Prototype 1: Task Management Workflow

**Depth of Implementation:** Fully functional task creation, assignment, completion, and tracking workflow.

##### Key Functionality:

1. User clicks "+ Add Task" button
2. Modal form appears with all task fields (title, description, assignees, due date, recurrence, urgency)
3. Form validation ensures required fields are completed
4. Task is saved to LocalStorage and immediately appears in the task list
5. User can filter tasks by various criteria (My Tasks, Today, All, Completed)
6. User can mark task as complete, triggering confetti animation
7. Completed tasks move to "Completed" filter view
8. Fairness score updates dynamically based on task completion
9. Activity feed updates with task completion event

**Testing Capabilities:** This vertical prototype supports comprehensive usability testing for:

- Task creation efficiency and error handling
- Filter usability and effectiveness
- Visual feedback and confirmation mechanisms
- Data persistence and state management

### 3.2.2 Vertical Prototype 2: Expense Tracking & Balance Calculation

**Depth of Implementation:** Complete expense lifecycle with automatic balance calculation and financial transparency.

**Key Functionality:**

1. User adds new expense with amount, payer, and participants
2. System automatically calculates split amounts per participant
3. Balance summary updates in real-time showing:
  - Total amount user owes to others
  - Total amount others owe to user
  - Net balance (positive or negative)
4. User can mark expenses as "Settled" when paid
5. Balance calculations adjust based on settlement status
6. Expense history is maintained with full edit/delete capabilities
7. Search functionality allows quick expense lookup

**Testing Capabilities:** This vertical prototype enables testing of:

- Expense entry accuracy and validation
- Balance calculation correctness
- Financial transparency and clarity
- Settlement workflow efficiency

## Design Highlights & Innovation

### Visual Design Excellence

- **Neumorphic Aesthetics:** Modern, soft interface design that feels tactile and approachable
- **Smooth Animations:** Page transitions, card animations, and micro-interactions enhance perceived performance
- **Loading Screen:** Branded splash screen with animated logo and spinner for professional first impression
- **Empty States:** Thoughtful empty state designs with helpful messaging and clear calls-to-action
- **Color Psychology:** Strategic use of colors for urgency (red), success (green), and information (blue)

## Innovative Features

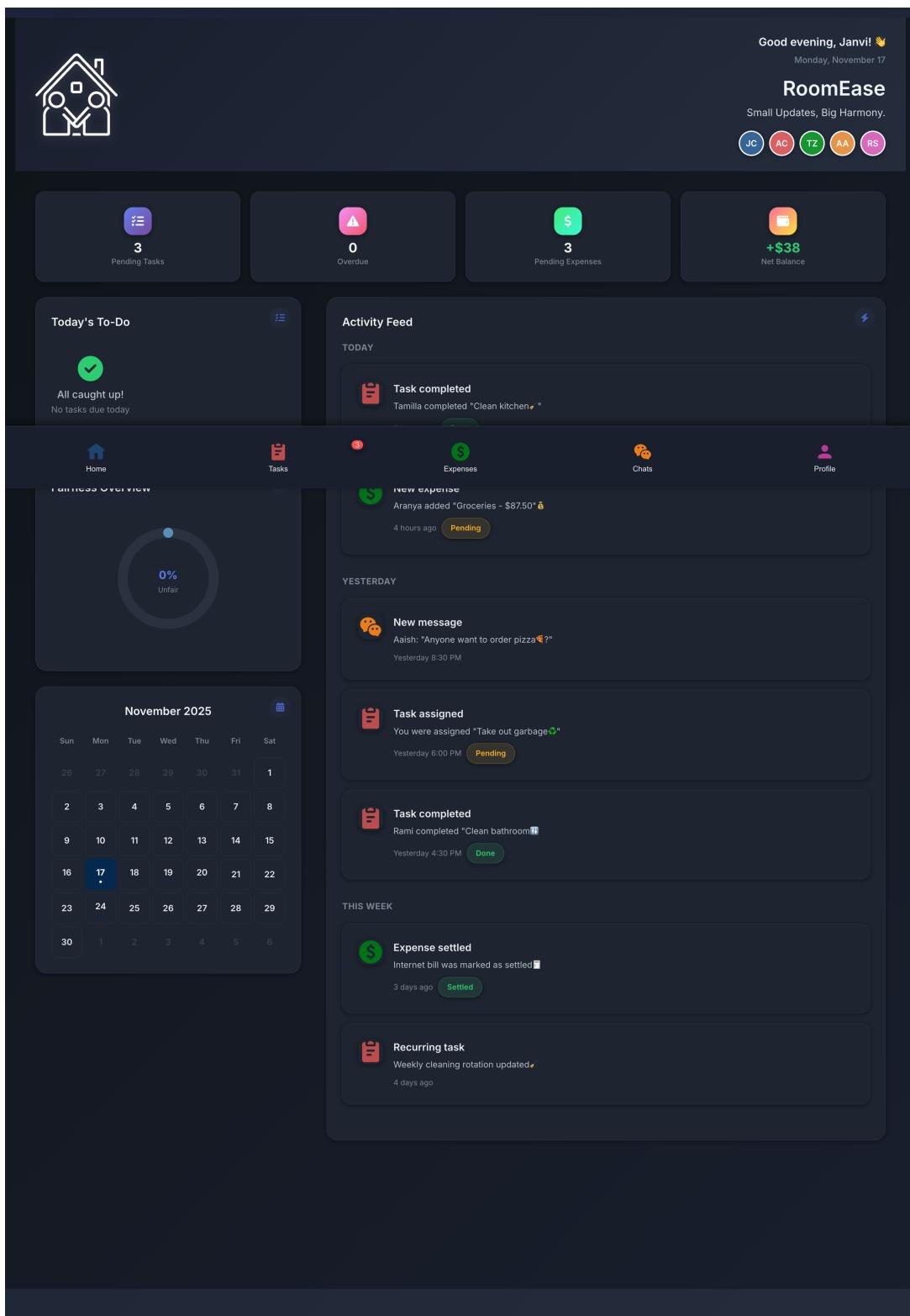
1. **Dynamic Fairness Calculation:** Real-time fairness score based on task completion distribution, providing transparency and motivation
2. **Activity Feed:** Chronological feed of all household activities, creating a sense of community and accountability
3. **Confetti Celebration:** Positive reinforcement through visual celebration when tasks are completed
4. **Integrated Communication:** Chat functionality embedded within the task management system, reducing context switching
5. **Smart Quick Stats:** Home page provides immediate insight into pending tasks, overdue items, expenses, and financial balance
6. **Personalized Experience:** Dynamic greetings, date displays, and user-specific filtering create a tailored experience

## Usability Enhancements

- **Keyboard Shortcuts:** Ctrl/Cmd+N for new task/expense, Esc to close modals
- **Real-time Search:** Instant filtering as user types in search bars
- **Toast Notifications:** Non-intrusive feedback for all user actions
- **Form Validation:** Real-time field validation with clear error messages
- **Responsive Design:** Seamless experience across mobile, tablet, and desktop devices
- **Accessibility:** Semantic HTML, proper contrast ratios, and keyboard navigation support

## Interface Screenshots

## Home Dashboard



figureHome dashboard showing welcome message, quick stats, today's to-do, fairness overview, calendar, and activity feed

## Tasks Page

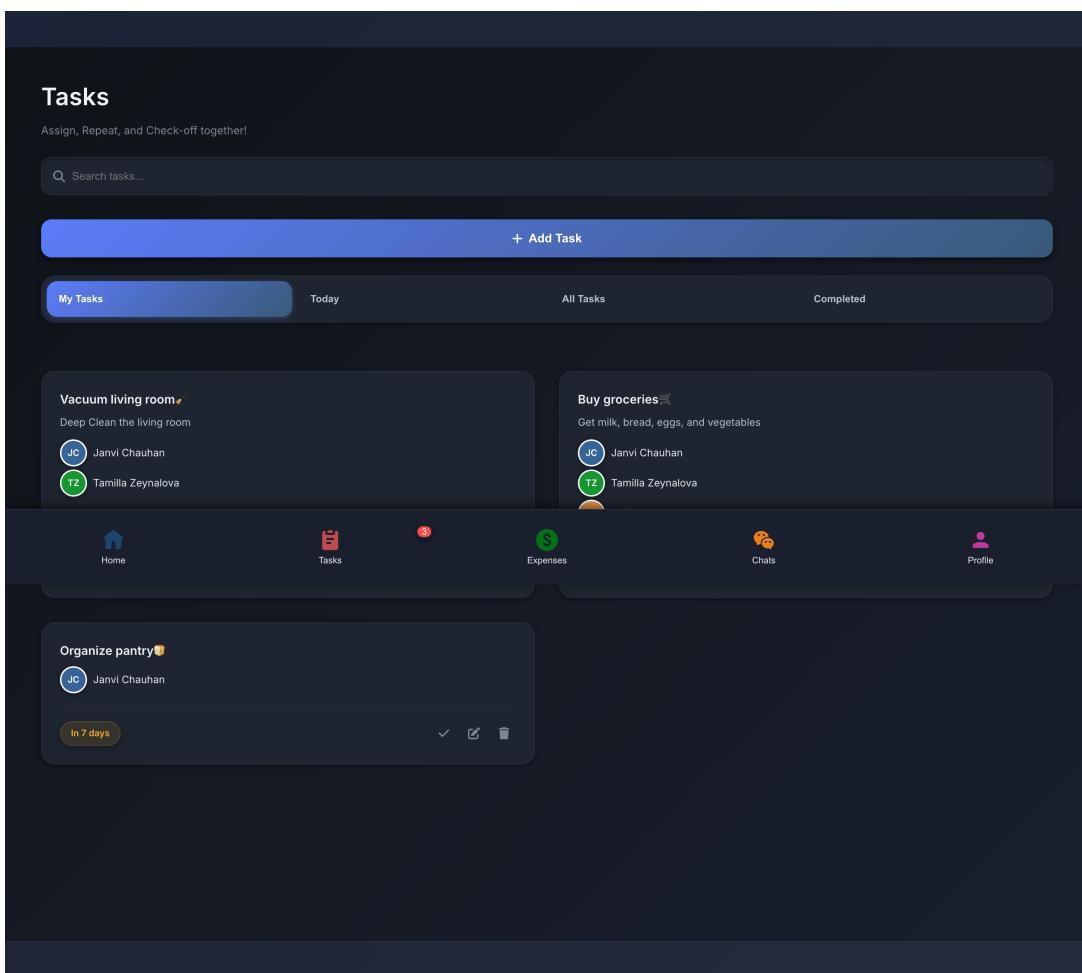
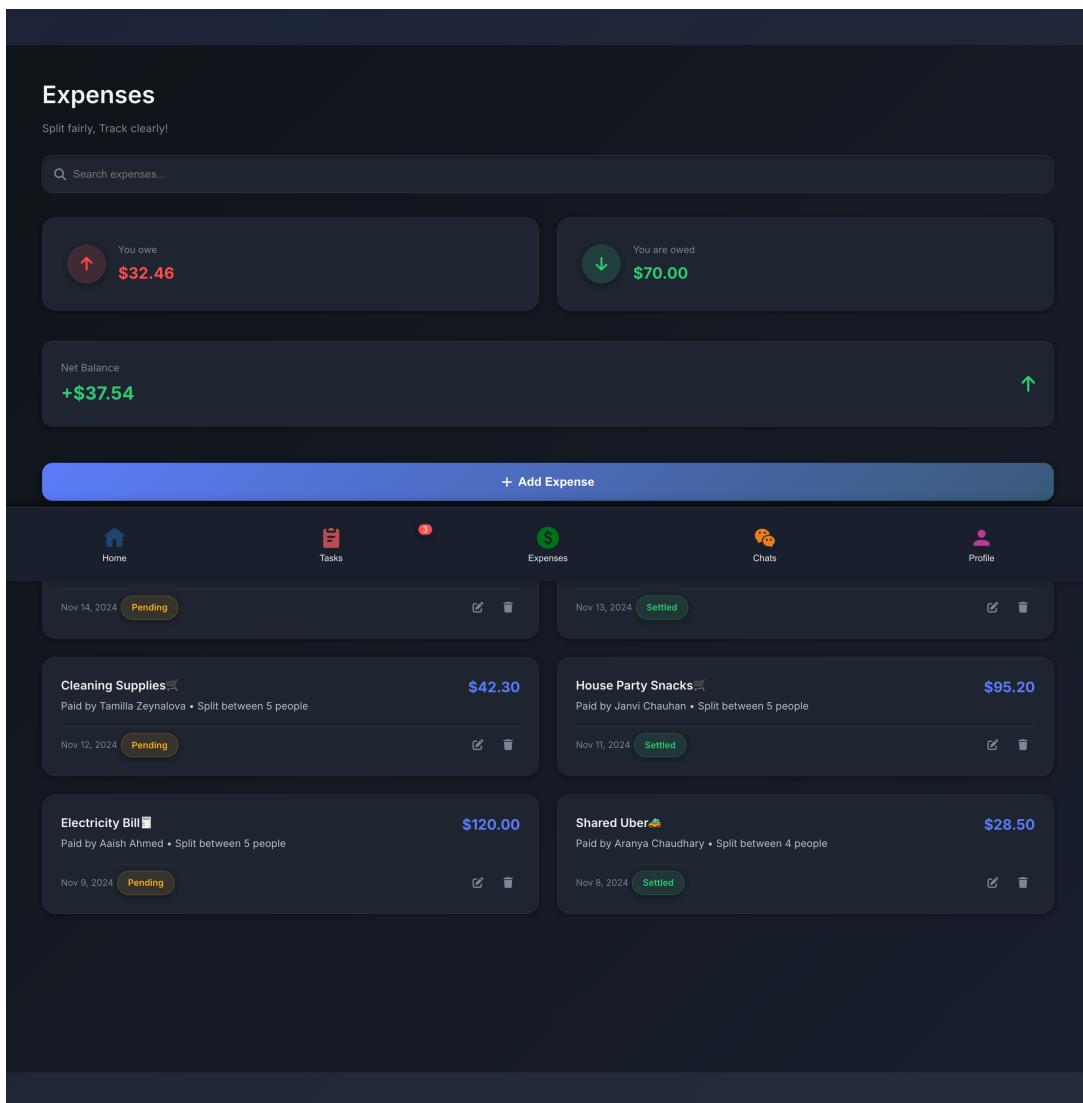


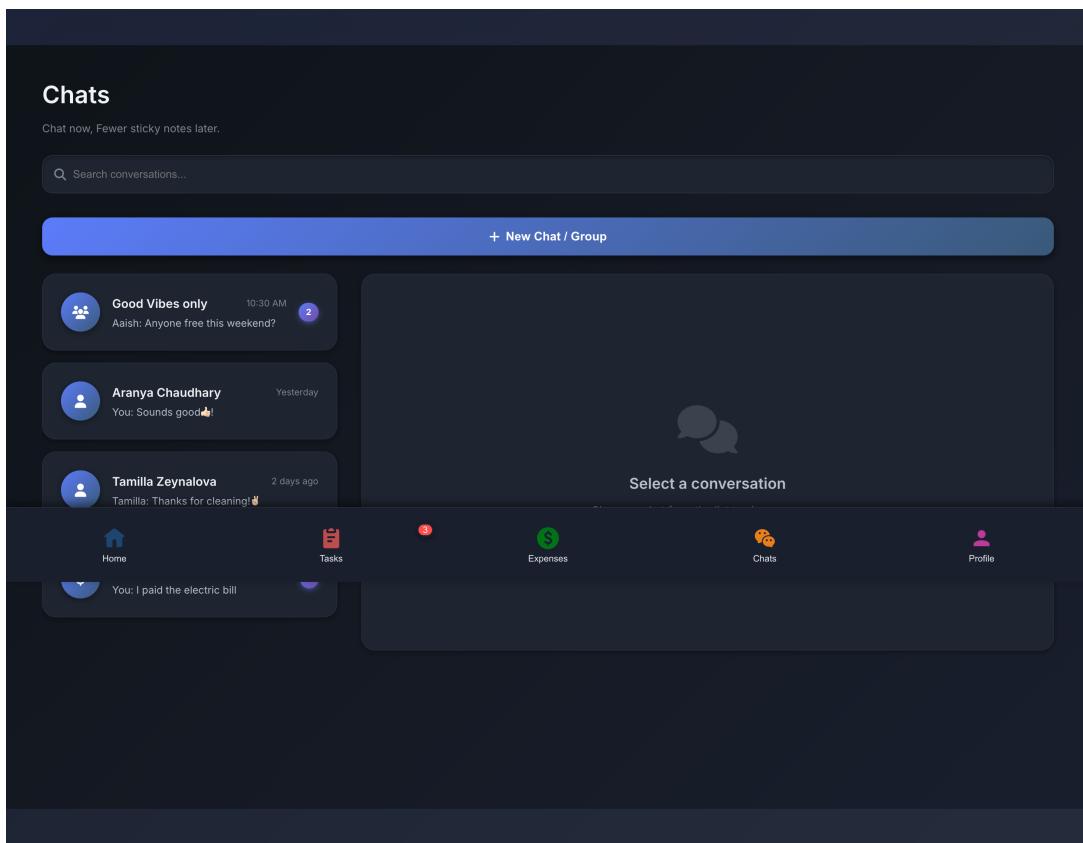
figure Tasks management page with filtering options, search functionality, and task cards showing assignments and due dates

## Expenses Page



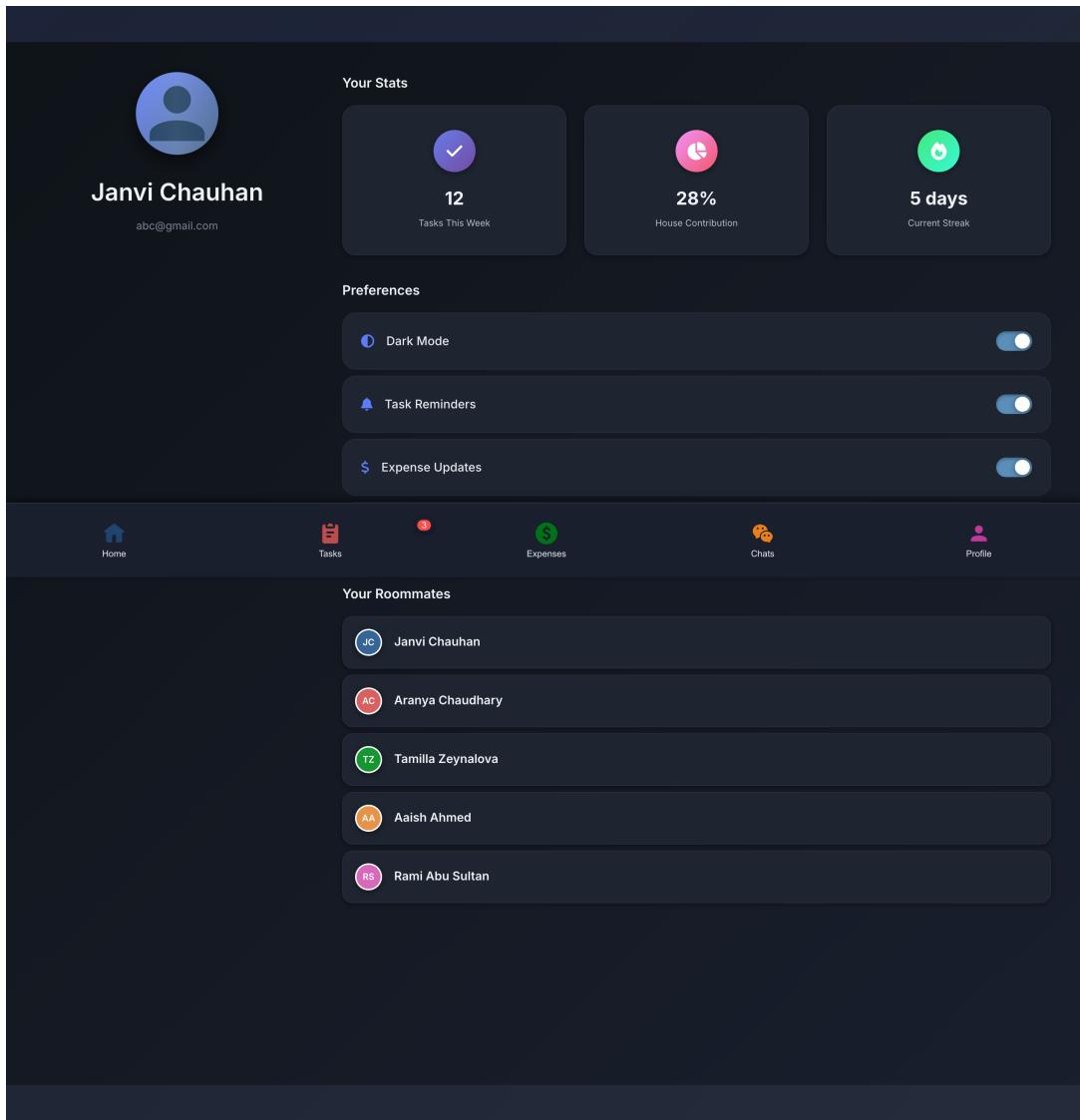
figureExpenses page displaying balance summary (You owe, You are owed, Net Balance) and expense list with status indicators

## Chat Interface



figureChat page showing conversation list with unread badges and message previews

## Profile Page



figureProfile page with user statistics, preferences, and roommate list

## Strengths of the System

### Comprehensive Feature Set

The prototype successfully implements all core requirements from Milestone 1, including task management, expense tracking, and communication features. The system goes beyond basic requirements with additional features like fairness tracking, activity feeds, and smart filtering.

### Polished User Experience

- **Visual Consistency:** Cohesive design language throughout all pages
- **Smooth Interactions:** Animations and transitions make the interface feel responsive and modern
- **Clear Information Hierarchy:** Important information is prominently displayed with appropriate visual weight

- **Helpful Feedback:** Toast notifications, loading states, and empty states guide users effectively

## Technical Implementation

- **Clean Code Structure:** Well-organized JavaScript with clear function separation
- **Data Persistence:** Reliable LocalStorage implementation ensures data is not lost
- **Performance:** Efficient DOM manipulation and event handling
- **Maintainability:** CSS custom properties and modular code structure facilitate future updates

## Innovation & Creativity

- **Fairness Algorithm:** Dynamic calculation provides unique value proposition
- **Activity Feed:** Creates transparency and accountability in shared living
- **Integrated Design:** Seamless combination of task management, expenses, and communication
- **Visual Polish:** Attention to detail in animations, empty states, and micro-interactions

## Known Issues & Limitations

### Current Limitations

1. **No Backend Integration:** All data is stored locally in the browser's LocalStorage, meaning data is device-specific and not shared across devices or users
2. **No Real-time Synchronization:** Changes made by one user are not reflected for other users in real-time (would require backend infrastructure)
3. **Limited Chat Functionality:** While the chat interface is implemented, message sending is simulated with hardcoded data rather than actual real-time messaging
4. **No User Authentication:** The system assumes a single user session without login/logout functionality
5. **No Data Export:** Users cannot export their data for backup or migration purposes
6. **No Recurring Task Automation:** Recurring tasks are created but do not automatically generate new instances on schedule (would require background job processing)
7. **No Notification System:** While reminders are mentioned in preferences, actual push notifications are not implemented (would require service worker and backend)
8. **No Receipt Upload:** Expense tracking does not support image uploads for receipts (would require file storage)

## Design Trade-offs

- **Client-side Only:** Chosen to keep the prototype simple and testable without requiring server setup, but limits multi-user functionality
- **Hardcoded Sample Data:** Initial data is hardcoded to demonstrate features, but real implementation would start with empty state
- **Desktop-First Responsive Design:** While mobile-optimized, some features (like split-view chat) are optimized for larger screens

## Future Enhancements

For a production version, the following would be priorities:

- Backend API with database for multi-user support
- Real-time synchronization using WebSockets
- User authentication and authorization
- Push notifications for task reminders
- Receipt image upload and OCR for automatic expense entry
- Data export/import functionality
- Mobile app versions (iOS/Android)
- Integration with payment apps (Venmo, PayPal)

## Testing & Evaluation Readiness

### Usability Testing Support

The prototype is fully prepared for usability testing with:

- **Complete Task Flows:** Both vertical prototypes support end-to-end task completion
- **Realistic Data:** Sample data provides context for testing scenarios
- **Error Handling:** Form validation and user feedback enable testing of error scenarios
- **Multiple Interaction Methods:** Mouse, keyboard, and touch interactions are all supported

### Testing Scenarios

The following scenarios can be tested:

1. **Task Creation:** Create a new task, assign it to multiple roommates, set due date and recurrence
2. **Task Completion:** Filter tasks, mark task as complete, observe fairness score update
3. **Expense Entry:** Add a new expense, split it among roommates, verify balance calculations
4. **Expense Settlement:** Mark expense as settled, observe balance updates
5. **Navigation:** Move between different sections, test search and filter functionality
6. **Theme Switching:** Toggle dark mode, verify theme persistence

## Conclusion

RoomEase successfully demonstrates a comprehensive, high-fidelity prototype that addresses the core requirements identified in Milestone 1. The system provides a polished, innovative solution for roommate task and expense management with strong attention to user experience, visual design, and technical implementation.

The prototype is ready for usability testing and evaluation, with two fully functional vertical prototypes (task management and expense tracking) that can support detailed user testing. While the current implementation is client-side only, the architecture and design decisions provide a solid foundation for future backend integration and multi-user support.

The combination of comprehensive features, innovative design elements (fairness tracking, activity feed), and polished user experience positions RoomEase as a strong candidate for further development and real-world deployment.

— *End of Report* —