

I & E

Project Proposal:

HOMIES


Team 4: Abeera, Chloe, Lamia, Qaisar



# Homies

Harmony in Every Home

We solve the problem of confusion and conflict over household chores in shared homes for roommates, couples, and families by providing an app that allows users to assign, track, and manage tasks transparently, ensuring fairness and accountability.



# Overview

1. **Situation & Problem**
2. **BMC: Revenue Streams, Prices**
3. **Solution & Value Proposition**
4. **Functionalities & Screens**
5. **Code & Demo**
6. **Manual Prototype**
7. **Application: Technology & Tests**
8. **Customers: Acquisition & Retention**
9. **Users List**
10. **Market Study (Users & Customers)**

11. **Interviews**
12. **Questionnaire**
13. **Market Study (Users & Customers): Sources**
14. **Market Study (Competitors)**
15. **Market Study (Competitors): Sources**
16. **Detailed Analysis of User Behaviour**
17. **User Feedback & Analysis: Manual Prototype**
18. **User Feedback & Analysis: Application**
19. **Financial Plan**
20. **Conclusion**
21. **Presentation Responsibility**

# Situation & Problem

## Situation



- **Qaisar**: university student living with 3 roommates, struggles to coordinate household chores in a shared apartment.
- Each roommate has a different schedule, so tasks like cleaning, taking out the trash, or washing dishes often get delayed.
- Occurrence: **Daily / Weekly chores**, recurring issues multiple times per week.

## Problem

- **Confusion** over who is responsible for what task.
- Tasks are forgotten or **unevenly distributed**, leading to **frustration** and **tension** among housemates.
- Lack of transparency causes **repeated arguments** and **decreased motivation** to do chores.

# BMC: Value Proposition

## Qualitative

- Creates **peace of mind** with transparent responsibility allocation
- **Reduce tension** and **misunderstandings** among housemates
- Provides a sense of **organisation & structure**
- Makes chore management **simple**

## Quantitative

- **Reduce time spent** coordinating chores
- Improve **fairness** in task coordination
- **Reduce** household **conflicts**
- **Reduce** number of **forgotten tasks**

# BMC: Customer Relations

## Self-Service

- Users create households, assign chores, navigate features themselves
- Include simple onboarding and default tasks/tutorials

## Automated services

- Automatic reminders for upcoming tasks
- Automatic logging of user actions

## Community Feedback

- Collect suggestions or bug reports

# BMC: Resources

## Human

- Development team: developers, designers, testers
- Project team members: contribute to ideation, organisation, code, testing

## Intellectual

- Concept & App idea
- User Interface design
- Task scheduling logic and flow
- Prototype of application

## Physical

- Laptops/mobile for development & testing
- Hosting infrastructure for backend, database, authentication (Firebase)
- Development tools (VSCode)

# BMC: Key Activities

## 1. Understanding User Needs

- a. Conduct surveys, observe problems with students, colocataires, couples
- b. Collect feedback on painpoints: chore distribution, reminders, fairness, reassignment
- c. Validate useful features for households

## 2. App development

- a. Design and implement core basic features
- b. Build & maintain backend, authentication
- c. Develop user interface

## 3. Customer Acquisition

- a. Share prototype with initial user groups for testing
- b. Promote through word-of-mouth, social networks, student housing groups

## 4. Testing & Iteration

- a. Test the prototype with real users
- b. Identify bugs, usability issues, unclear steps
- c. Iterate on design & features based on feedback

## 5. Platform Maintenance

- a. Manage hosting, database stability, and push notifications during test period



# BMC: Key Partners

## Strategic Alliances

- **University housing offices / student residences:** help reach students living in shared apartments
- **Co-living companies:** potential agreement to recommend the app to residents
- **Lifestyle / home organization influencers:** partnerships to increase visibility and adoption

## Potential Integration Partners

- **Calendar service providers** (Google Calendar): for optional future integration with user calendars.

# Solution

## **Homie, an application that allows users to:**

- Create a shared household space and add members
  - Assign tasks with deadlines
- Track completed tasks and contributions
  - Send reminders and notifications
- Reassign or request help for chores

# Functionalities

## Must-Have

- **Create a household space**
- **Add members or join household via link or QR code**
- **Create a task**
- **Assign task to a household member**
- **Add a deadline (specific date)**
- **Mark task as done**
- **Reassign task / request help**
- **Activity log** (list showing who completed which tasks)
- **Calendar views:** daily / weekly / monthly
- **User account creation + login**
- **Email notifications** for reminders

## Nice-to-Have

- **Point or reward system**
- **Task categories** (cleaning, cooking, shopping...)
- **Color-code for each member**
- **Add a time for a task**
- **Mark urgent vs non-urgent tasks**
- **Delay/postpone a non-urgent task**
- **Comment section under each task**

# Screens

- **Dashboard page** → summary of today's tasks
- **Task creation form** → name, deadline, assignee
- **Calendar page** → day/week/month view
- **Household page** → members and invitation link
- **Task details page** → comments, urgency, mark as done
- **Activity log page** → list of completed tasks
- **Profile / settings page** → notifications, household options

# Code & Demo

## Implementation Overview

- The prototype interfaces and screens were built using **TypeScript** with a modular structure.
- Each screen is organized into reusable components for easy iteration and future scalability.
- The code reflects the app's core logic at a conceptual level-handling navigation, user input, state updates, and simulated data retrieval.

## Code Example (Simplified)

```
ts
// Basic types
type Task = {
  id: string;
  title: string;
  completed: boolean;
  deadline: Date;
  assignedTo?: string;
};

type Member = {
  id: string;
  name: string;
};

// Simple task list component
function TaskList({
  tasks: Task[];
  members: Member[];
  onToggle: (id: string) => void;
  onDelete: (id: string) => void;
}) {
  const getMemberName = (id?: string) =>
    props.members.find((m) => m.id === id)?.name ?? "Unassigned";

  return (
    <div>
      {props.tasks.map((task) => (
        <div key={task.id}>
          <input
            type="checkbox"
            checked={task.completed}
            onChange={() => props.onToggle(task.id)}
          />

          <strong>{task.title}</strong>

          <span> - {getMemberName(task.assignedTo)}</span>

          <span> - {task.deadline.toDateString()}</span>

          <button onClick={() => props.onDelete(task.id)}>Delete</button>
        </div>
      ))}
    </div>
  );
}
```

# Code & Demo

## Key Features Demonstrated

- **User Registration Flow**  
Simple form validation, error handling, and input state management.
- **Appointment Scheduling Flow**  
Allows the user to choose a day, pick a time slot, and confirm an appointment.  
All implemented with a clean TypeScript component architecture.
- **Dashboard Mock Logic**  
Components simulate:
  - Upcoming appointment display
  - Past appointment list
  - Quick actions for booking or canceling

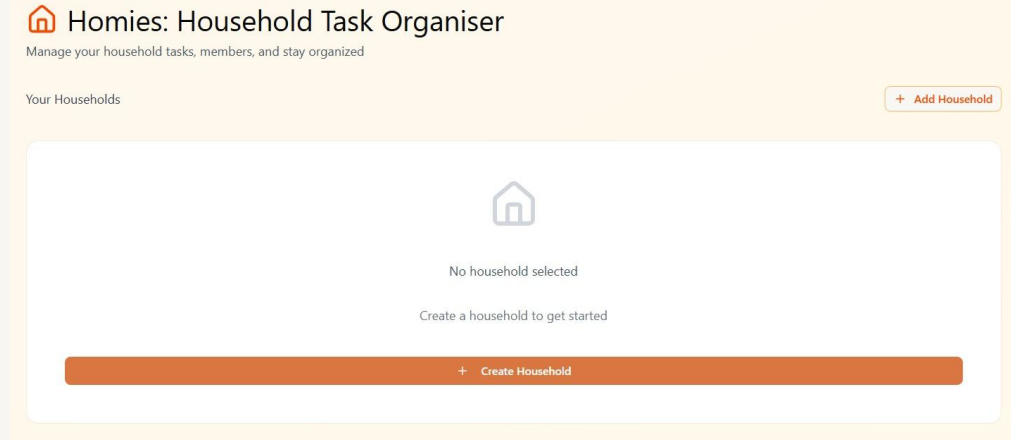
## Demonstration Summary

- The prototype runs through the full booking cycle:
  - **Home** → **Choose Service** → **Pick Date/Time** → **Confirm**
  - **View** **appointments**
  - **(Simulated)** **modify** **or** **cancel**
- The aim is to illustrate core app logic before implementing the backend and final design.

# Manual Prototype

## Why a Manual Prototype?

- To validate the concept early.
- To test user flow and usability with minimal development investment.
- To communicate the idea clearly during development planning.



# Manual Prototype

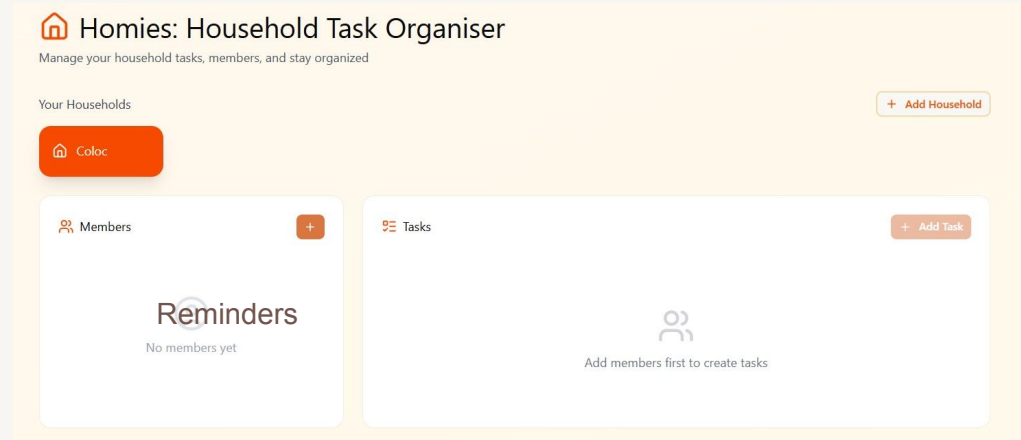
## Prototype Structure

The manual prototype contains all main screens of the planned application,

representing:

- Onboarding
- Home/dashboard
- Task Assignment workflow
- Tasks details
- Completion

and



These layouts show exactly how a user will move from one step to the next.



# Manual Prototype

## How It Works (Concept Flow)

- A household is created.
- Members are added to the household.
- Member Adds/confirms a task.
- Backend generates:
  - optional calendar event data
- User receives reminders at scheduled times.
- If user changes or cancels the appointment:
  - notifications are updated or deleted accordingly

The image displays two overlapping mobile application screens. The top screen, titled 'Add Member', features a subtitle 'Add a new member to your household'. It contains a 'Name' input field with the placeholder text 'e.g., John Smith', a 'Role (optional)' dropdown menu with the example 'e.g., Parent, Child, Roommate', and two buttons at the bottom: 'Cancel' and 'Add Member'. The bottom screen, titled 'Add Task', has a subtitle 'Create a new task and assign it to a member'. It includes a 'Task Title' input field with the placeholder 'e.g., Clean the kitchen', a 'Description (optional)' text area with the placeholder 'Add details about the task...', an 'Assign To' dropdown menu with the placeholder 'Select a member', and a 'Deadline' section with a calendar icon and the text 'Pick a date'. This screen also has 'Cancel' and 'Add Task' buttons at the bottom.

# Application – Technology Chosen

- **Web application** accessible via browser on desktop or mobile
- To developed using **modern web technologies** (TypeScript + Next.js)
- Focus on **fast, responsive, and collaborative experience**
- Designed to support **real-time updates**, modifications, and user friendly dashboards

# Why a Web App?

**Accessible Anywhere:** Works on desktops, laptops, tablets, and mobile browsers hence no installation needed.

**Cross-Platform:** All household members can access the app regardless of device type.

**Easy Updates:** New features or bug fixes are instantly available to all users.

**Centralized Data:** All tasks, logs, and notifications are synced in real time for everyone.

**Scalable & Future-Ready:** Can later add features without major changes.

# Customer Acquisition

## Acquisition:

- Target **shared households and student dorms**
- Promote via **social media, university campaigns, referrals**
- **Simple onboarding** for multiple members

# Customer Retention

## Retention:

- **Transparent logs & dashboards** fairness & accountability
- **Notifications & reminders** for task completion
- **Gamification:** points, badges
- **Personalization:** color coded members, task categories, urgency labels

# Customer Acquisition & Retention

## CRM VIEW:

- **New Users:** Help them get started quickly → onboarding support
- **Active Users:** Keep them motivated → rewards & engagement incentives
- **Passive Users:** Encourage participation → gentle reminders & tips
- **Dormant Users:** Bring them back → re-engagement notifications
- **Household Admins:** Manage the household → assign tasks & encourage members

# Customer Acquisition & Retention

## Customer Acquisition

- **Shareable link / QR code to join a household**
- **Social media presence**
- **Onboarding tutorial**
- **Referral system** (optional)

## Retention Strategy

- **Reminder notifications through email**
- **Email reminders for overdue tasks**
- **Comments under tasks**
- **Weekly summary delivered by email** (optional)
- **Household stats**

## Accounting & Analysis

- **Monthly Active Users**
- **Monthly Active Households**
- **Task completion rate**
- **New accounts created per week**
- **Retention after 1 week, 4 weeks, 12 weeks**
- **Churn rate** (how many users stop using the website)
- **Bug reports & feedback form responses**

# Users List





# Market Study (Users & Customers)



# Interviews



# Questionnaire



# Market Study (Users & Customers): Sources



# Market Study (Competitors)



# Market Study (Competitors): Sources



# Detailed Analysis of User Behaviour



# User Feedback & Analysis

## Manual Prototype



1. Google sign in is not working
2. It will be helpful if user can see a confirmation popup before removing the task
3. Done task should not be crossed. It seems like the task is cancelled
4. I can see profile section twice on the home page. It will be good if the profile section appears once on the home.
6. When I am on home page and press back button , the login page shows . And when I press back again in that state, the home page shows again
7. When I go back to the site again , all the notifications are gone.

- Could we get a feature to track expenses and budget for household items?
- It would be helpful if we could customize the app to fit our specific household needs!

10:18 PM

Thank you for inviting me to use your very useful application. First I would like to say that the idea of making this app is useful as someone who always procrastinate and leave my task till the very last minute or even forgetting to do chores, your app gave me a clear step of documenting what chores I want to do and after finished, the thrill of ticking off every single task. In real world context, I would say it's very useful. However, some improvement I could see is adding visualisation like charts to see a percentage of how much I've done so far. This would give me more motivation to finish everything I need to do!

10:53 PM

Here's my feedback after testing the app:

I really liked how intuitive and easy to use it is and the overall experience is smooth and efficient!! However, one thing felt limiting it's the fact that a task can only be assigned to one person at a time.

In group situations (like roommates or larger teams), some tasks need to be prepared or handled by several people together before being split into individual responsibilities. It would be much more practical if a single task could be assigned to multiple users.

Right now, the only workaround is to create the same task several times and assign it separately to each person, which is repetitive and not very convenient. Adding a "multi-assignee" option for tasks would make the app more flexible and better suited for larger groups.

6:21 PM



# User Feedback & Analysis

## Application



# Financial Plan



# Conclusion



# Presentation Responsibilities

