

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.

# BMC: Channels

The channels explain how the app reaches and interacts with customers.

- **Social Media:** Instagram, TikTok, Facebook, WhatsApp for viral roommate content & demonstration videos.
- **Referral System:** Existing users invite household members via link.
- **Email Campaigns:** Onboarding guidance, reminders, updates.
- **University Partnerships:** Promote to student housing/residence halls.
- **Community Platforms:** Reddit, Discord communities.

# BMC: Customer Segments

## Primary Users

- Students living in shared apartments/dorms
- Young professionals in co-living spaces
- Shared households (friends, roommates)

## Secondary Users

- Families wanting chore distribution among members
- Domestic helpers/cleaning services coordinating tasks

# BMC: Cost Structure

## 1. App Development (Free)

- All development done by the project team
- No paid developers involved
- Open-source libraries and frameworks used

## 2. UI/UX Design (Free)

- Designed in house using free tools (e.g., Figma free plan)

## 3. Cloud Hosting & Infrastructure (Free Tier)

- Vercel (frontend hosting)
- NeonDB(database)
- Firebase Authentication (login & user management)
- Free object storage (ObjectModel FireBase)

# BMC: Cost Structure

## 4. Maintenance & Updates (Free)

- Managed by the team
- Includes fixing bugs, adding improvements, monitoring

## 5. Administrative Costs (Not applicable)

- No business registration
- No legal or accounting costs

# BMC: Cost Structure

## 6. Marketing & Advertising (None)

- No paid ads
- Only organic sharing or word-of-mouth

## 8. Email Delivery Services

- SendGrid/Mailgun free plans (if emails are required)

## 7. App Store Fees

- App is not published on Apple/Google stores
- Runs as a web/mobile prototype

# BMC: Revenue Streams

**The main revenue stream is €1.99 micro-fee per premium membership.**

## Free Tier

The free tier app will allow the user to:

- Create household
- Basic tasks & reminders
- Basic dashboard

## Premium Subscription

- Day view, week, monthly week
- Point/reward system
- Categorisation of task
- Remind people to do task
- Adding calendar view
- Different colors for different members
- Add time
- Ability to delay non-urgent/low-priority
- Mark a task as urgent vs non-urgent
- Comments on task
- Notification to remind people: through email and/or pop up notification

# 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)

```
// 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 TasksList(props: {
  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={() => propsonDelete(task.id)}>Delete</button>
        </div>
      ))
    </div>
  );
}
```

# Code & Demo

## Key Features Demonstrated

- |   |              |   |
|---|--------------|---|
| ● <b>User Registration</b>  | <b>Flow</b>  | ● The prototype runs through the full booking cycle:  |
| Simple form validation, error handling, and input state management.   |              | ○ <b>Home → Choose Service → Pick Date/Time → Confirm</b>                                   |
| ● <b>Appointment Scheduling</b>   | <b>Flow</b>  | ○ <b>View appointments</b>  |
| Allows the user to choose a day, pick a time slot, and confirm an appointment.<br>All implemented with a clean TypeScript component architecture. |              | ○ <b>(Simulated) modify or cancel</b>   |
| ● <b>Dashboard Mock</b>   | <b>Logic</b> | ● The aim is to illustrate core app logic before implementing the backend and final design. |
| Components simulate:  |              |   |
| ○ Upcoming appointment display  |              |   |
| ○ Past appointment list   |              |   |
| ○ Quick actions for booking or canceling  |              |   |

## Demonstration Summary

# 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.

### Homies: Household Task Organiser

Manage your household tasks, members, and stay organized

Your Households

+ Add Household



No household selected

Create a household to get started

+ Create Household

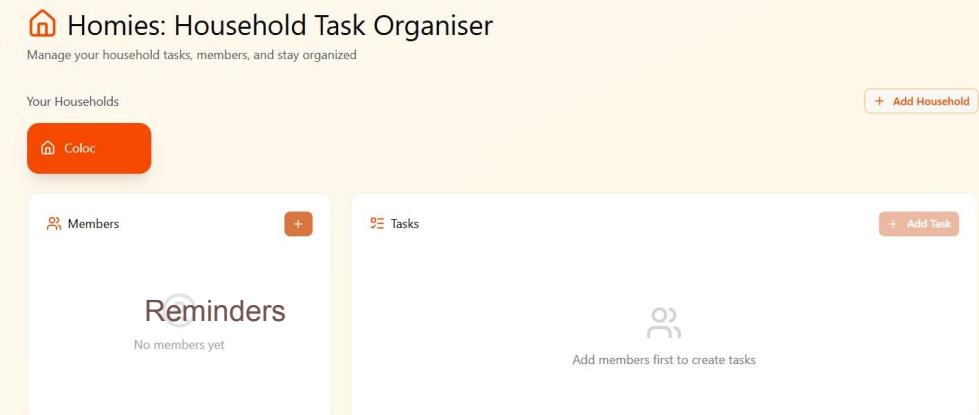
# Manual Prototype

## 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 shows two overlapping modal windows. The top window is titled 'Add Member' and has a subtitle 'Add a new member to your household'. It contains fields for 'Name' (with placeholder 'e.g., John Smith') and 'Role (optional)' (with placeholder 'e.g., Parent, Child, Roommate'). It includes 'Cancel' and 'Add Member' buttons. The bottom window is titled 'Add Task' and has a subtitle 'Create a new task and assign it to a member'. It contains fields for 'Task Title' (placeholder 'e.g., Clean the kitchen'), 'Description (optional)' (placeholder 'Add details about the task...'), 'Assign To' (placeholder 'Select a member'), and 'Deadline' (button 'Pick a date'). It also includes 'Cancel' and 'Add Task' buttons.

# Application – Technology Chosen

- **Web application** accessible via browser on desktop or mobile
- 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



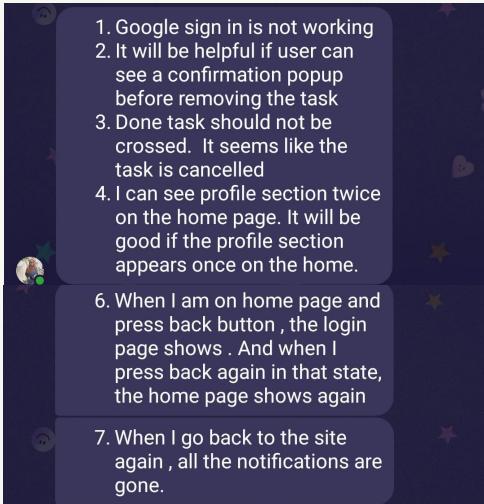
# Detailed Analysis of User Behaviour



<b>id</b>	<b>userId</b>	<b>action</b>	<b>details</b>	<b>timestamp</b>
00c4d126-7167-4317-9ea0-5b4fca212311	iaACH6KC8cNKMEEwgNdEPIMsrrx1	TASK_COMPLETED	{"taskId": "8e07d118-cf7f-44da-8ea5-bd9c20ec11e4"}	2025-11-24 15:45:19.455
0158544d-3cc1-48cb-91cb-6338ddef9474	iaACH6KC8cNKMEEwgNdEPIMsrrx1	USER_LOGOUT		2025-11-24 16:05:41.747
01f72161-5d13-4dc3-9d57-f271d4699bdf	iaACH6KC8cNKMEEwgNdEPIMsrrx1	DASHBOARD_LOADED		2025-11-24 16:14:06.032
03f36f53-d991-4c84-ac80-f05f07242080	MkfIiLDLwQyfYEeqyUyG4q4XjCxce2	HOUSE_CREATED_CLIENT	{"name": "Beach House with Mountain view", "houseId": "fbfbdf5c-5283-45b5-bbd2-5df0d6ae5a8"}	2025-11-24 16:08:31.839
045704f4-2985-420f-8d40-8f691803cf0b	iaACH6KC8cNKMEEwgNdEPIMsrrx1	NAV_DASHBOARD_VIEW		2025-11-24 16:04:02.371
0487ccb2-c5e1-418f-9df0-796a66cf715c	YcZQJlQMSNPPKZiBAOEdSaLJGGr2	TASK_REOPENED	{"taskId": "ca7a74b6-34ac-4fcf-bfbe-5e1821d1fa91"}	2025-11-24 10:52:19.197
06d37f37-92c6-486c-9c11-d1decde4bb89	dAhN1lzuOscko0g5zNkWplmrAj	HOUSE_SELECTED	{"houseId": "d8ac7a21-d38b-4c01-81f4-22c8e18a14d4"}	2025-11-24 15:29:36.421
083e6882-f690-4f8e-863f-74a53b5fb056	YcZQJlQMSNPPKZiBAOEdSaLJGGr2	DASHBOARD_LOADED		2025-11-24 16:07:57.173
0cb8576a-588d-4e99-82ff-9ac608b37b	iaACH6KC8cNKMEEwgNdEPIMsrrx1	DASHBOARD_LOADED		2025-11-24 16:21:24.647
0daee3258-38e9-4250-aecb-33e98812bab6	R4VWAOYRqHRmsvy5IAF1RwHm1U32	DASHBOARD_LOADED		2025-11-24 20:05:13.685
0f729634-6df3-4be1-9b78-24ba2ca62505	9deRbFUVcnMxXskcjAy71zeIE8z1	TASK_COMPLETED	{"taskId": "c0237eaa-7245-46ea-a332-2910f13dd22e"}	2025-11-24 18:22:15.147
			{"url": "/api/houses?"}	
107fcc5d-fc48-4b95-ab83-35534b129d6e	YcZQJlQMSNPPKZiBAOEdSaLJGGr2	NETWORK_CONNECT	userId: YcZQJlQMSNPPKZiBAOEdSaLJGGr2", "error": "Error: Failed to fetch houses"}	2025-11-24 16:07:57.59
11d723b3-6a2d-409d-b4e2-82d681ed4bba	iaACH6KC8cNKMEEwgNdEPIMsrrx1	NAV_PROFILE_VIEW		2025-11-24 16:20:07.484
12fbaf62-5c47-4655-9269-b449f27d7f81	R4VWAOYRqHRmsvy5IAF1RwHm1U32	TASK_CREATED	{"title": "Read paper too", "houseId": "a6daafec-1c3f-4038-a20f-491b4fb29bb8"}	2025-11-24 12:50:50.825
14c62bdf-9d54-43ba-9617-004cd08766b	iaACH6KC8cNKMEEwgNdEPIMsrrx1	TASK_CREATED	{"title": "Clean the kitchen", "houseId": "669e2c1b-b765-4c59-89ce-ac2d20314f26"}	2025-11-24 16:18:07.819
170c746c-048f-43c4-a457-2a6d89ed87ac	YcZQJlQMSNPPKZiBAOEdSaLJGGr2	TASK_CREATED	{"title": "Floor", "houseId": "d8ac7a21-d38b-4c01-81f4-22c8e18a14d4"}	2025-11-24 10:48:42.632
17279bf6-2ae8-463e-9915-6df27f6599d7	iaACH6KC8cNKMEEwgNdEPIMsrrx1	HOUSE_SELECTED	{"houseId": "669e2c1b-b765-4c59-89ce-ac2d20314f26"}	2025-11-24 16:18:36.583
1888e314-a0b7-4f15-bd0b-79149a0220e7	YcZQJlQMSNPPKZiBAOEdSaLJGGr2	HOUSE_SELECTED	{"houseId": "d8ac7a21-d38b-4c01-81f4-22c8e18a14d4"}	2025-11-24 10:42:05.423
18ee32c0-ca6b-4d01-9c46-d2889e8c36fa	MkfIiLDLwQyfYEeqyUyG4q4XjCxce2	DASHBOARD_LOADED		2025-11-24 16:08:58.533
1e41411d-ae07-4ad4-91d5-8a2510b81f15	9deRbFUVcnMxXskcjAy71zeIE8z1	NAV_DASHBOARD_VIEW		2025-11-24 18:20:19.988
1d824c73-42a2-4d21-8f73-62a9215ec9ff	YcZQJlQMSNPPKZiBAOEdSaLJGGr2	DASHBOARD_LOADED		2025-11-24 16:10:48.844
1f036fce-44d2-407c-858a-f8ad4e733544	R4VWAOYRqHRmsvy5IAF1RwHm1U32	MEMBER_ADDED	{"email": "abeeranaaeem99@gmail.com", "houseId": "a6daafec-1c3f-4038-a20f-491b4fb29bb8"}	2025-11-24 12:47:11.995
2016b14f-3311-44f7-9f47-16e18d36cd61	YcZQJlQMSNPPKZiBAOEdSaLJGGr2	NAV_DASHBOARD_VIEW		2025-11-24 10:47:17.411
2717f7a7-6325-4da3-b057-56a4ae42ea3d	iaACH6KC8cNKMEEwgNdEPIMsrrx1	TASK_DELETED	{"taskId": "7da9c0b2-f802-4342-923f-d257ecdd26db"}	2025-11-24 16:03:27.163
27972417-50df-4a6d-8aeb-941d17ed308	iaACH6KC8cNKMEEwgNdEPIMsrrx1	HOUSE_SELECTED	{"houseId": "669e2c1b-b765-4c59-89ce-ac2d20314f26"}	2025-11-24 16:19:18.426
28705172-8739-4b35-bef6-7ee5c9c45a92	iaACH6KC8cNKMEEwgNdEPIMsrrx1	USER_LOGOUT		2025-11-24 16:13:49.993
29b38c6-dadd-42b7-9adb-02dcf2e58ce4	9deRbFUVcnMxXskcjAy71zeIE8z1	NAV_DASHBOARD_VIEW		2025-11-24 19:02:10.676
2b706c4-a731-4b06-9d8d-b84d20fcd469	iaACH6KC8cNKMEEwgNdEPIMsrrx1	DASHBOARD_LOADED		2025-11-24 16:16:42.918
2da8c2a1-aecf-40fd-8983-31435a6b828a	iaACH6KC8cNKMEEwgNdEPIMsrrx1	DASHBOARD_LOADED		2025-11-24 16:14:02.365

# User Feedback & Analysis

## Manual Prototype



- 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

