

Deliverable 1
Quantum Quants
Housefy
Team 04

Team members

Member 1: Wenyuan Yu, Student ID: n01403697

Member 2: Kyrylo Lvov, Student ID: n01414058

Member 3: Artem Tsurkan, Student ID: n01414146

Table of Contents

1. Member's Info and Participation Table
2. Project Scope and goals that are targeted in this course
3. GitHub Repo Link and Strategy explanation
4. Screenshot of the invitation of the Hardware Professor
5. Screenshot of the Table with Stories and breakdown of Tasks, with start/end date, size and priority
6. Meeting the DoD criteria explanation for the completed tasks, plus the actual DoD table itself
7. Business Model Canvas, with all 9 fields clearly showed and explained
8. Gantt Chart with main milestones and work progress (min.10 components and timelines, with focus on main components)
9. Record of Daily Stand-ups

1. Member's Info and Participation Table

Name	ID	Signature	Effort
Kyrylo Lvov	n01414058	Kyrylo Lvov	100%
Wenyuan Yu	n01403697	Wenyuan Yu	100%
Artem Tsurkan	n01414146	Artem Tsurkan	90%

2. Project Scope and goals that are targeted in this course

Project Scope

The scope of the project encompasses the following key areas:

Hardware Integration: This includes the setup and integration of a Raspberry Pi 4 Model B with several sensors including the CCS811 Air Quality Sensor, DHT22/AM2302 Temperature and Humidity Sensor, PZEM-004T Smart Energy Meter, and TSL2561 Digital Light Sensor.

Software Development: The software development component involves the creation of a Python script to be run on the Raspberry Pi, which collects data from the sensors and controls the connected devices. Additionally, an Android application written in Java will be developed, fetching data from an

InfluxDB database and providing a user interface for real-time data visualization and device control.

Database Management: The project includes setting up and managing an InfluxDB time-series database to store sensor data for analysis and visualization.

This Course's Goals:

Understand IoT Systems: Gain a comprehensive understanding of the design, development, and operation of IoT systems, especially in the context of home automation and environment monitoring.

Develop Programming Skills: Improve proficiency in Python and Java, developing both a Python script to collect and manage sensor data and a Java-based Android application for user interaction.

Learn Database Management: Acquire experience with InfluxDB, learning how to manage a time-series database, store sensor data, and retrieve data for analysis and visualization.

Hardware Integration: Learn how to integrate various hardware components such as sensors, microcontrollers, and devices in a coherent IoT system.

Develop Problem Solving Skills: Enhance problem-solving skills by overcoming challenges during the hardware setup, software development, database management, and data visualization stages.

Project Management: Gain practical experience in managing a project from conception to completion, including planning, execution, monitoring, and final presentation.

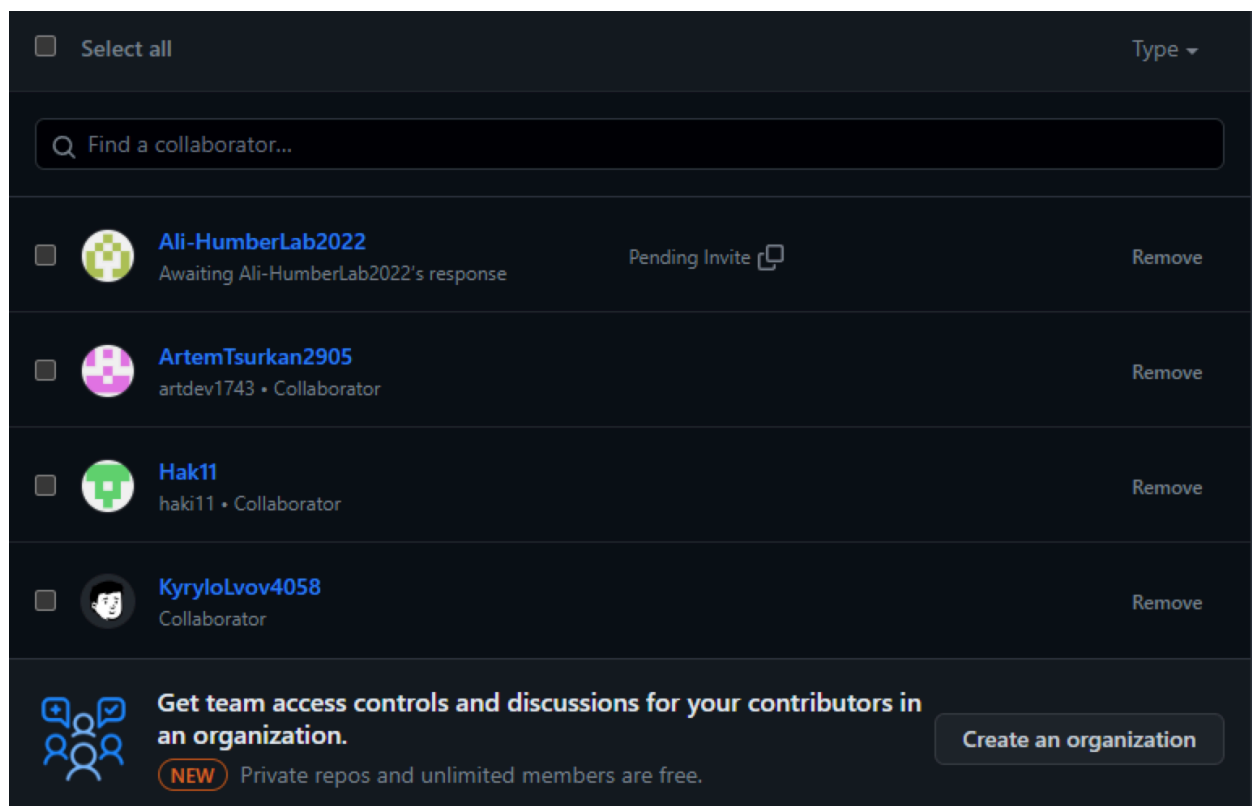
3. GitHub Repo Link and Strategy Explanation

<https://github.com/WenyuanYu3697/housefy>

Our github strategy is the following:

We develop our app by creating a new github branch for each new feature that we implement. After each new feature is “Done” as per the DoD (described further in this document), it is then merged to Master.

4. Screenshot of the invitation of the Hardware Professor



5. Screenshot of the Table with Stories and breakdown of Tasks, with start/end date, size and priority

Story 2: Air quality fragment	Story 3: A/C Control Fragment	Story 4: Smart Light Control Fragment	Story 5: Menu Functionality	Story 6: Permissions Functionality
<p>Task 1: Design the layout for the air quality fragment.</p> <p>Task 2: Hardcode the metric values</p> <p>Task 3: Implement any necessary UI animations or transitions for an improved user experience.</p> <p>+ Add a card</p>	<p>Task 1: Design the layout for the A/C control fragment, including buttons or sliders for temperature control.</p> <p>Task 2: Implement the code to simulate A/C control functionality without the actual backend integration.</p> <p>Task 3: Implement logic to handle user interactions with the A/C control UI elements.</p> <p>Task 4: Display the current temperature settings and update them based on user input.</p> <p>Task 5: Validate the responsiveness and accuracy of the A/C control fragment's UI.</p> <p>+ Add a card</p>	<p>Task 1: Design the layout for the smart light control fragment, including switches or sliders for controlling lights.</p> <p>Task 2: Implement the hardcoding to simulate smart light control functionality without actual backend integration.</p> <p>Task 3: Handle user interactions with the smart light control UI elements.</p> <p>Task 4: Update the UI to reflect the current state of the lights (e.g., on/off status, brightness level).</p> <p>Task 5: Verify the responsiveness and accuracy of the smart light control fragment's UI.</p> <p>+ Add a card</p>	<p>Task 1: Design the layout and structure for the app's menu, considering navigation and settings options.</p> <p>Task 2: Implement code to handle menu item selections and navigate to respective screens.</p> <p>Task 3: Create the settings screen UI and link it to the menu.</p> <p>Task 4: Ensure proper navigation between the different screens of the app.</p> <p>Task 5: Validate the correct screen transitions and menu item behavior within the menu functionality.</p> <p>+ Add a card</p>	<p>Task 1: Identify the required permissions for accessing specific device features (e.g., location, camera).</p> <p>Task 2: Implement code to request the necessary permissions from the user.</p> <p>Task 3: Handle permission request responses (e.g., granted, denied) and provide appropriate feedback to the user.</p> <p>Task 4: Design UI elements to handle scenarios requiring permission rationale explanations.</p> <p>Task 5: Test the permissions functionality for different scenarios, such as granting or denying permissions.</p> <p>Enter a title for this card...</p>

Story 1: Icons for Nav. Drawer Links
<p>Size: S</p> <p>Task 1: Design or find appropriate icons for each navigation drawer link.</p> <p>Jun 6 - Jun 6</p>
<p>Size: S</p> <p>Task 2: Integrate the icon library or assets into the project.</p> <p>Jun 6 - Jun 6</p>
<p>Size: M</p> <p>Task 3: Update the navigation drawer layout to display the icons alongside the links.</p> <p>Jun 6 - Jun 6</p>
<p>Size: M</p> <p>Task 4: Implement click listeners for the navigation drawer links.</p> <p>Jun 6 - Jun 6</p>
<p>+ Add a card</p>

6. Meeting the DoD criteria explanation for the completed tasks, plus the actual DoD table itself

Before marking each task as “Done”, it was evaluated if it meets our Definition of Done, which is outlined below:

Our Definition of Done (DoD):

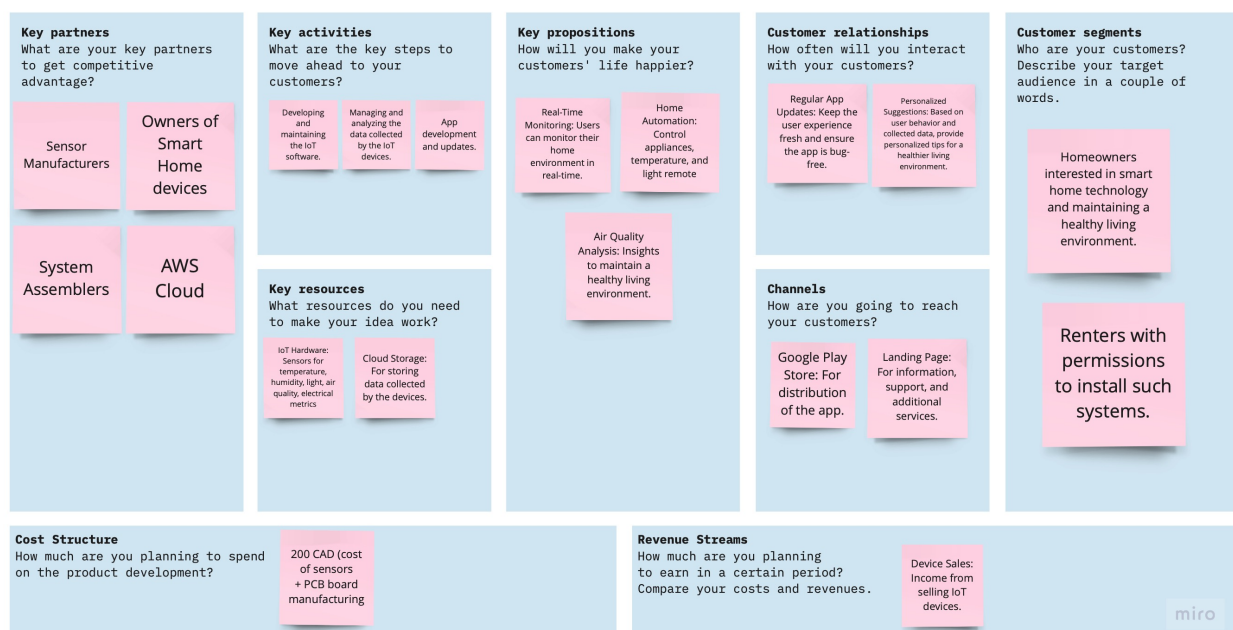
1. For Design Tasks:

1. The team has reviewed and agreed to proceed with the design of an element/screen etc. that is being proposed

2. For Coding Tasks:

1. The code runs smoothly, without errors. The purpose of this line is to remove “Testing” from the task lists, because everyone can click “Run” and see if their code runs.
2. Integrated, that is the code can run together with other elements and features of the app that have already been developed without conflicts.
3. The code has been reviewed by all team members and every team member has agreed that the code is meets the criterion outlined above and is ready to be pushed to Master.

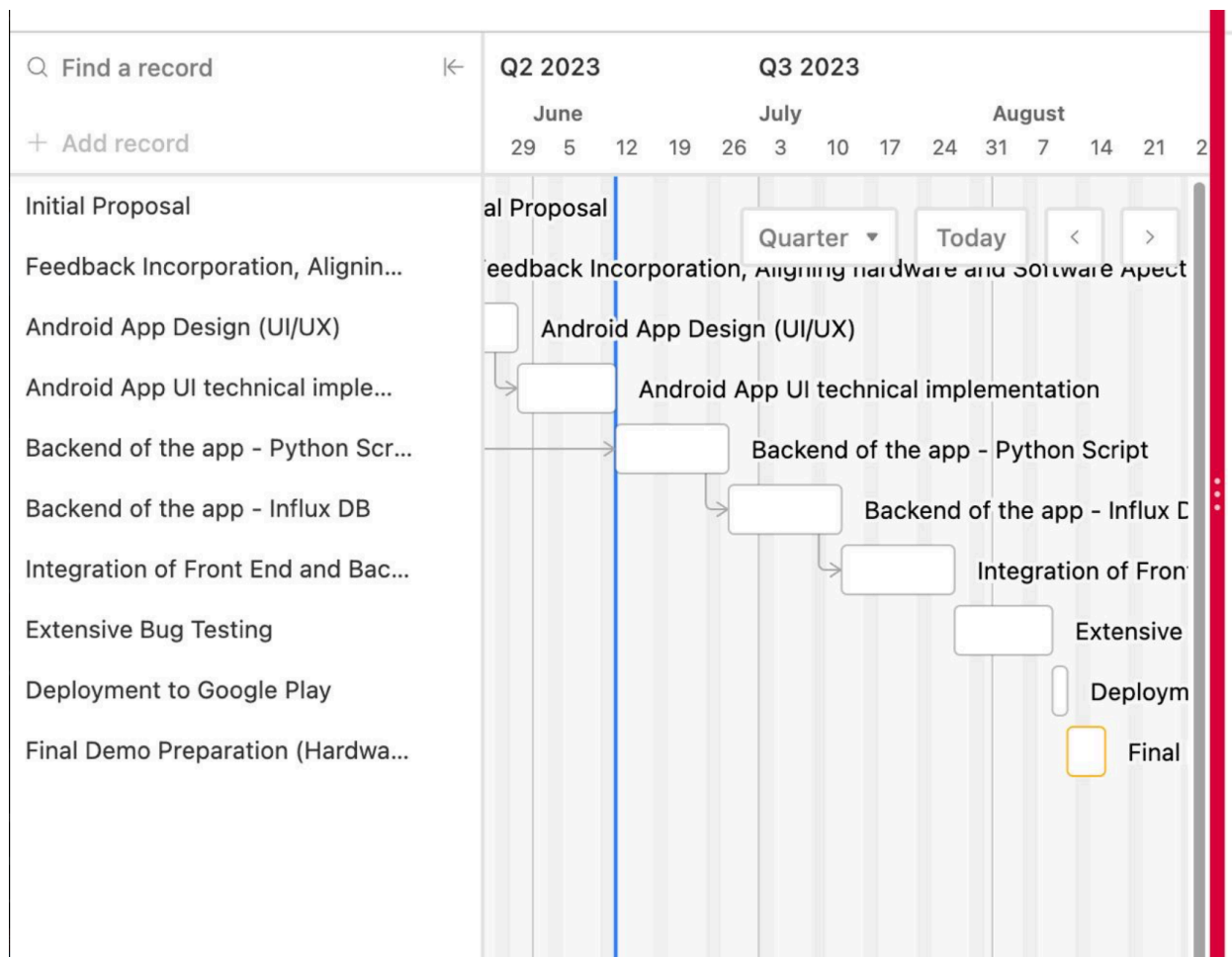
The Business Model Canvas



7. Business Model Canvas, with all 9 fields clearly showed and explained

8. Gantt Chart with main milestones and work progress (min.10 components and timelines, with focus on main components)

<https://airtable.com/embed/shrCfZMXHG7mRPXDf?backgroundColor=red>



9. Record of Daily Stand-ups

<https://www.notion.so/>

[Deliverable-2-763fafcd414d4eb6b09fa25558cb0c53](https://www.notion.so/Deliverable-2-763fafcd414d4eb6b09fa25558cb0c53)