Student Smart Homes (SSH) EDR

By Sheun Shoyebi 2424078

Introduction

For many university students around the country and globally, knowing what ingredients to buy in order to create a healthy meal, storing those ingredients, and using them before they expire, is one of the most frustrating challenges students face. The money required to buy the ingredients almost feels like it has been wasted once the food goes bad, and for university students that may find themselves in tougher financial circumstances, this could lead to actual problems regarding healthy eating habits and food insecurity. According to experts, 20 to 50 percent of college students in the U.S. experience food insecurity, which is consistently higher than that of the general population. (Shaak, 2021)

Recognising this, at Smart Student Homes (SSH), we look to provide students with the means and technology to streamline their lives, and to make the shared living experience more manageable and seamless. We formulated an idea that expands the functionality of one of our flagship products, the SSH Camera, in order to combat this issue. The SSH Camera uses smart camera technology and computer vision in order to capture and remember what food items and ingredients are currently available, and paired with a customisable database of recipes, meals can be suggested that are able to be made with only the available ingredients.

Implementing this would greatly reduce the wastage that occurs in student homes/flats. It would also allow students to learn how to cook more creatively and healthily by being exposed to new recipes on the app. This can enhance a student's quality of life, and also their satisfaction as a customer. Ultimately, this would help combat the rising issue of food insecurity within the student community, helping limit new eating disorders or health complications that could arise as a result.

Aside from the benefit to the student experience, this feature would bolster SSH's product/service portfolio, especially on the software side in regards to the proprietary computer vision and object recognition technology, which would in turn strengthen our position as leaders within the student housing technology provider market. This could in turn lead to an increase in subscribers for our Cloud service for example, or even lead to sales of our other items, resulting in a continuous flow of revenue for the company.

Goals:

- The primary goal is to reduce food waste in student homes by providing recipe suggestions based on the available ingredients detected by the SSH Camera.
- The system should accurately detect at least 90% of common food items in the fridge and update the inventory in real-time.
- At least 70% of users should engage with the recipe suggestion feature at least once per week, indicating its usefulness and relevance.

Non-goals:

- The system will not handle dietary restrictions or complex meal planning beyond basic recipe suggestions.
- The system will not account for food expiration dates or track the freshness of ingredients beyond their presence in the fridge.

Design Overview

Starting from a high level perspective, the system should combine the SSH Camera's object recognition abilities with the SSH App, which would have a recipe function with sufficient functionality, but also have the option to connect a purpose-built recipe app such as MyFitnessPal in order to have greater options and freedoms for creating recipes. All these connections would be facilitated by the SSH Cloud, using industry-level encryption to ensure the connection between the device hosting the App and the Camera is secure.

SSH Camera
Now focusing on the SSH Camera, this is the main piece of hardware within the system we are working with, and will be our main component. Using advanced computer vision algorithms paired with deep learning models, the Camera should be able to identify a large variety of ingredients with above 90% accuracy. The SSH Camera will use R-CNN-based object detection to identify food items in real-time with over 90% accuracy. The camera will transmit data to the cloud for processing, ensuring the app remains lightweight

This occurs in real-time, with the Camera being powered either by battery, which will be replaced once below a certain predetermined level (with the SSH App notifying the users once the battery is low), or it can be powered by wiring the camera to the main electrical circuit, which is a more permanent option that landlords perhaps could take, as opposed to the battery option which is aimed more towards students. The Camera processes the images locally to identify, and sends the data to the App via the Cloud, which allows for the app to be less memory-intensive as it doesn't have to process much data. The camera is made to work autonomously. If an item is wrongly registered, the user can manually correct it via the App.

SSH Cloud
Moving onto the SSH Cloud, it has been purpose built to match industry standard cloud technology, in order to process and store data that is collected by the Camera. Once a food item is recognised, that information will be sent from Camera to Cloud, and is stored on a cloud-hosted SQL database that is designed to be lightweight and fast in information transfer, as the data we are storing is not large in terms of volume, but there may be multiple changes required per day if items are added and removed to the fridge. Also as the database aims to be a live representation of the food present, it must be fast to update as well.

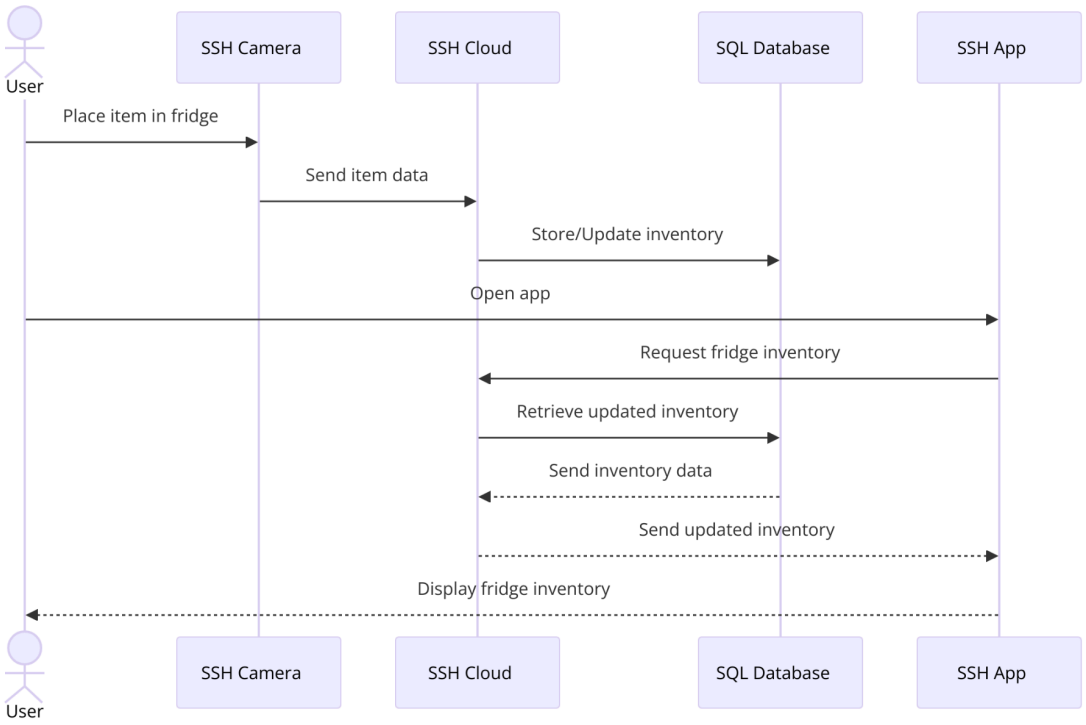
As well as the Cloud being quick to transfer information, it is also built to scale for many users across various households, and to manage the load of each household simultaneously. The SSH Cloud stores fridge data and recipe options, ensuring real-time updates across multiple households using scalable cloud infrastructure. Data is securely transmitted using OpenSSH encryption.

SSH App

Then finally for the SSH App, this would serve as our user facing interface that would allow students to interact with and customise the fridge inventory and recipe suggestion system. It would be fully integrated with the SSH ecosystem, being available on Android and iOS. Kotlin will be used to develop the Android App, and Swift to develop the iOS app. The app design will be minimal and clean, with intuitive ways to navigate to allow for easy use for students.

Users can opt to receive notifications when items are added/removed from the fridge, or when new recipes are available with the resources present. These recipes will provide step by step cooking instructions provided by an online database that is initially made by us at SSH, that will then be expanded by our user base. This will introduce recipes to the SSH community from cultures that our developers may not initially know, allowing for inclusion of all demographics. Users can also filter through cuisine type, dietary preferences or preparation time in order to suit everyone's respective needs.

Diagram outlining the process for Inventory viewing:



Alternatives

When coming up with the implementation for the recipe suggestion and fridge inventory, there were multiple viable alternatives we came up with, and each of them were evaluated on their strengths and weaknesses. We will list some below.

Manual Barcode Scanning System
This was a system where users would scan the barcode of every item they would put into the fridge, which would be checked against a database of widely available store barcodes.

Strengths	Weaknesses
Greater accuracy due to unique barcode, less reliance on AI systems for recognition, hence less resource intensive	Manual effort from user required, some products may have no barcode, time-consuming as well

Passive Recipe Suggestion System
Users would simply input their available ingredients into the app via barcode, in order to then receive suggestions from the bank of available recipes that are possible with the input ingredients.

Strengths	Weaknesses
Allows for automation while limiting reliance on AI/Computer Vision, less resource intensive	Still requires manual effort from user, fresh products may have no barcode

Use of AWS/Google Cloud for cloud services
Instead of proprietary cloud technology, use widely used cloud systems in order to benefit from their cost economies of scale.

Strengths	Weaknesses
Potential cost economies of scale from using their systems with pay-as-you-go pricing, vs the cost of RnD and setup of SSH Cloud. More advanced and experienced cloud experts can be hired to run the system, less chance for unexpected behaviour.	Not open source technology, open to security vulnerabilities from the vendor side, over time the cost of Pay as you go can be greater than SSH Cloud setup.

Milestones

Milestone 1: Develop Image Recognition - Implementing and testing the computer vision model for recognising food items using the SSH Camera.
Milestone 2: Integrate Camera with SSH Cloud - Set up a secure data transfer from the SSH Camera to the SSH Cloud and develop the inventory database.
Milestone 3: Create Recipe Matching Algorithm - Develop the algorithm in the SSH Cloud to match fridge contents with relevant recipes.
Milestone 4: Build SSH App Interface - Develop the app's UI for viewing fridge contents and recipe suggestions.
Milestone 5: End-to-End Testing - Conduct testing to ensure seamless integration of the Camera, Cloud, and App.
Milestone 6: Beta Testing - Release to a beta group, gather feedback, and make adjustments.
Milestone 7: Full Launch - Roll out the feature to all users and provide ongoing support.

Dependencies

Camera Development Team: Implements and refines the camera's computer vision algorithms.
Cloud Infrastructure Team: Manages secure data transfer, inventory, and recipe databases.
App Development Team: Builds the user interface and integrates it with the SSH Cloud.
Database Team: Maintains the SSH Recipe and inventory databases.
Security Team: Ensures secure data transmission and manages authentication.
Testing Team: Conducts end-to-end and beta testing.

Cost

Costs should be minimal as the hardware used already exists at SSH, it simply would be creation of new software to integrate with our current software setups. The main costs would be from Research and Development for the computer vision, increased cloud storage space and processing power for the recipe matching and inventory data.

Privacy and Security Concerns

All user data will be stored in the SSH Cloud securely, with strict limits to access. Only permitted users can modify and even view the data. Data will be securely transferred using OpenSSH protocols in order to encrypt the data and protect against unauthorised access. Users will consent to use of the app before inputting any information or setting up any of our products, to ensure we are compliant with legal requirements and in order to act in good faith towards our customers.

Risks

Risk	Impact	Mitigation
Inaccurate food detection by the camera	Poor user experience	Allow manual input of missing or incorrect items
Data security breach	Compromised user data	Implement OpenSSH encryption and regular audits
Low adoption of recipe suggestion feature	Limited user engagement	Provide in-app tutorials and incentives to use
Increased cloud processing load	Potential performance issues	Scale cloud infrastructure as demand grows

Supporting Material

Work Cited

Shaak, Natalie. “College Student Food Insecurity: How to Make Real Change - Center for Hunger Free Communities.” *Drexel University*, 21 December 2021, https://drexel.edu/hunger-free-center/news-events/voices-blog/2021/December/college-student-food-insecurity/. Accessed 21 October 2024.