



# Fresh Eyes

## Team B3

Alex Strasser, Samuel Leong, Oliver Li

# What is FreshEyes?

## Integrated smart fridge attachment system

### Vision-based Tracking System

Easy way to scan fresh produce before putting them into fridge



### Intuitive Interface + Reminder App

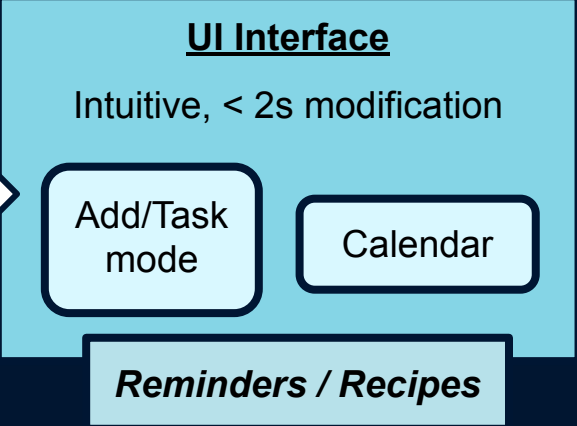
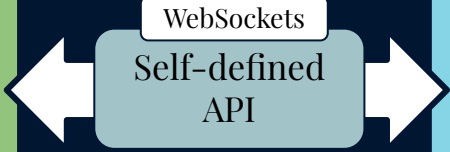
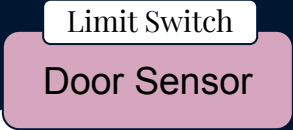
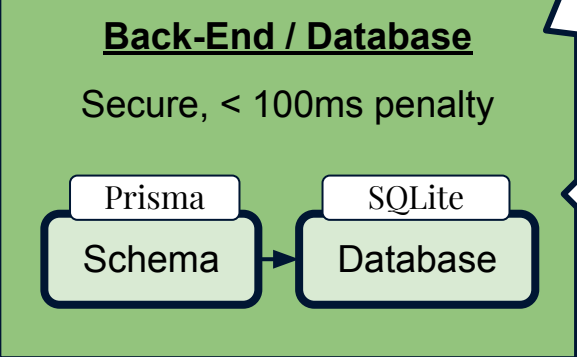
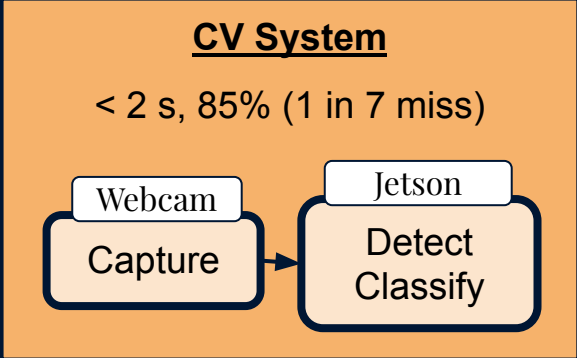
Allows user to intuitively track food  
Reminds user when fresh produce is expiring

**25lb/  
year!**

**\$65/  
year!**

**Reduce Food Waste  
Save Cost**

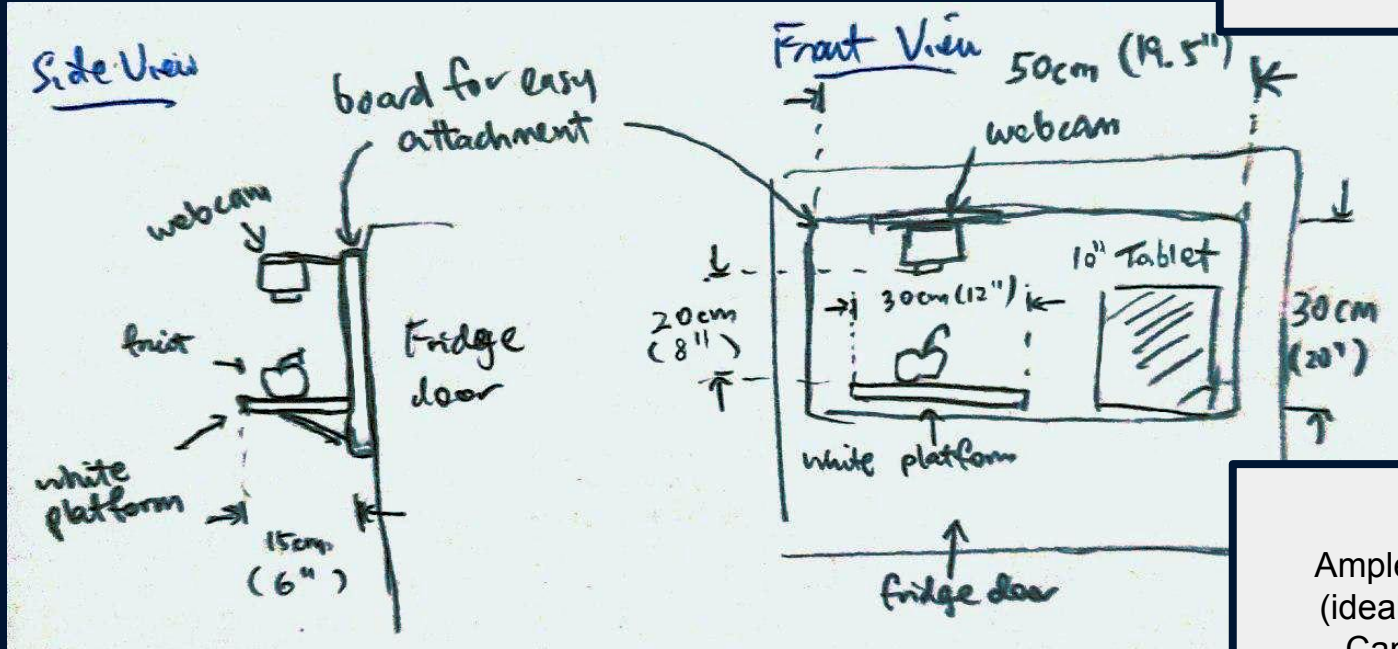
# Solution Approach & Requirements



# Fridge Attachment System

## Integrated

Everything on one large board  
Easily installed/removed with  
standard 3M tape



## Non-intrusive

Ample Space: Fairly large  
(ideally foldable) platform  
Camera on top allows  
ease-of-placement

# CV Detection and Classifier

## Basic White Background Segmenter

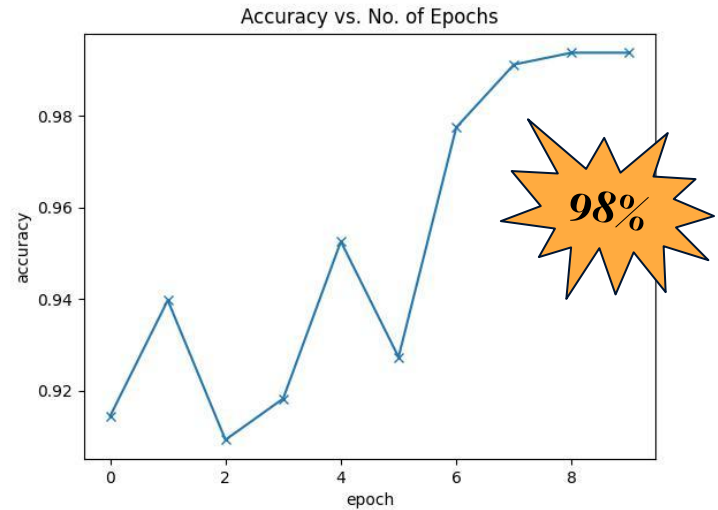
Platform will have white background  
Check for changes in pixels  
Basic white background subtraction  
with bounding box



\*sample, not actual image/output

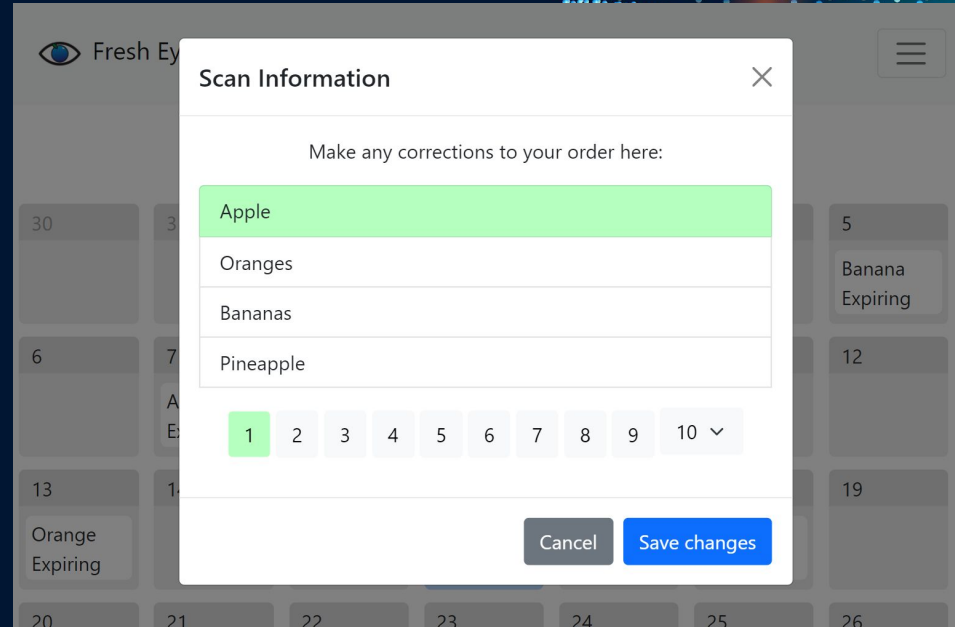
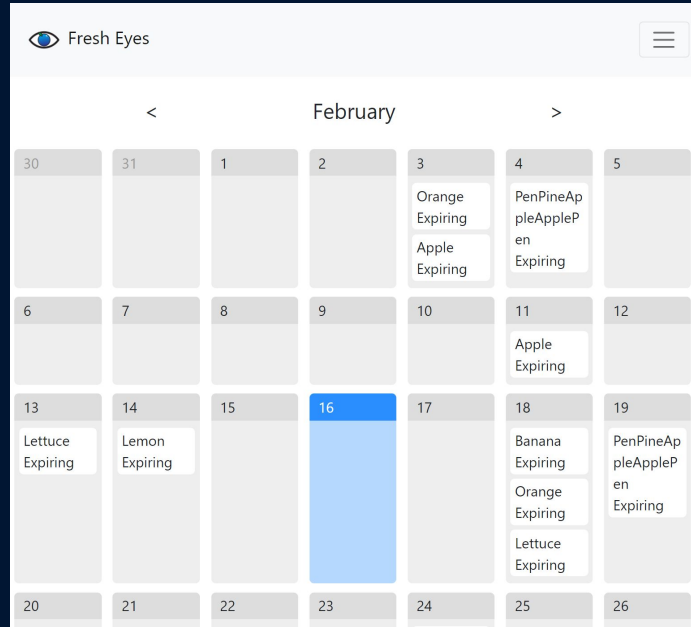
## 9 Layer ResNet CNN

Currently 98% accuracy!  
Need to test/train on real-world dataset



# Intuitive UI

<https://capstone.astrasser.com:2096/>



## Web-based UI

Since the UI is a website, it can be accessed from your phone, laptop, or the fridge tablet.

Communication will be fluid between all components.

## Minimal Clicks

We use an intuitive interface with limited dropdowns to minimize user clicks and interaction time

# Back-End Database

Schema defined as code  
We use an ORM library (Prisma) to define the DB schema

The screenshot shows a database management interface. At the top, there's a tab labeled 'Item' with a close button and a plus sign. Below that, there are controls for filters (set to 'None'), fields (set to 'All'), and a 'Showing 2 of 2' indicator, along with an 'Add record' button. The main table has columns: 'id #', 'name A', 'shelfLife #', 'unit A', and 'transactions []'. Two records are visible: 'Banana' with shelf life 4 and 'Apple' with shelf life 7. The 'Apple' record is selected, and a detailed view of its transactions is shown below. This view has columns: 'id', 'createdAt', 'updatedAt', 'item', 'itemId', and 'quantity'. It lists three transactions for the 'Apple' item.

id #	name A	shelfLife #	unit A	transactions []
1	Banana	4	piece	1 Transaction
2	Apple	7	piece	2 Transaction

id	createdAt	updatedAt	item	itemId	quantity
<input checked="" type="checkbox"/>	2022-02-16T15:22:06.5...	2022-02-16T15:22:06.5...	Item	2	10
<input checked="" type="checkbox"/>	2022-02-16T15:22:06.5...	2022-02-16T15:22:06.5...	Item	2	5
<input type="checkbox"/>	2022-02-16T15:22:06.5...	2022-02-16T15:22:06.5...	Item	1	5

Open in new tab   Skip to unconnected records

Maximum deployment flexibility  
We can easily switch between SQLite on dev environments and PostgreSQL on prod environments

```
model Item {
  id          Int          @id @default(autoincrement())
  name        String
  shelfLife   Int // Number of days item can be stored
  unit        String
  transactions Transaction[]
}
```

# Implementation Plan - Overview

## CV System

Bought webcam, Jetson

Modified online CNN classifier

Writing our own detection/segmentation and optimised program for Jetson

## Attachment System

To buy plywood for platform and support

Will build own basic platform

## Back-End / Database

Self-defined API (schemas done)

Writing own code for database management

## UI Interface

Bought tablet

Writing own web app (basic UI done)



# Testing, Verification, and Metrics

## CV System

Buy selection of fruits and vegetables

Test under simulated real-world conditions

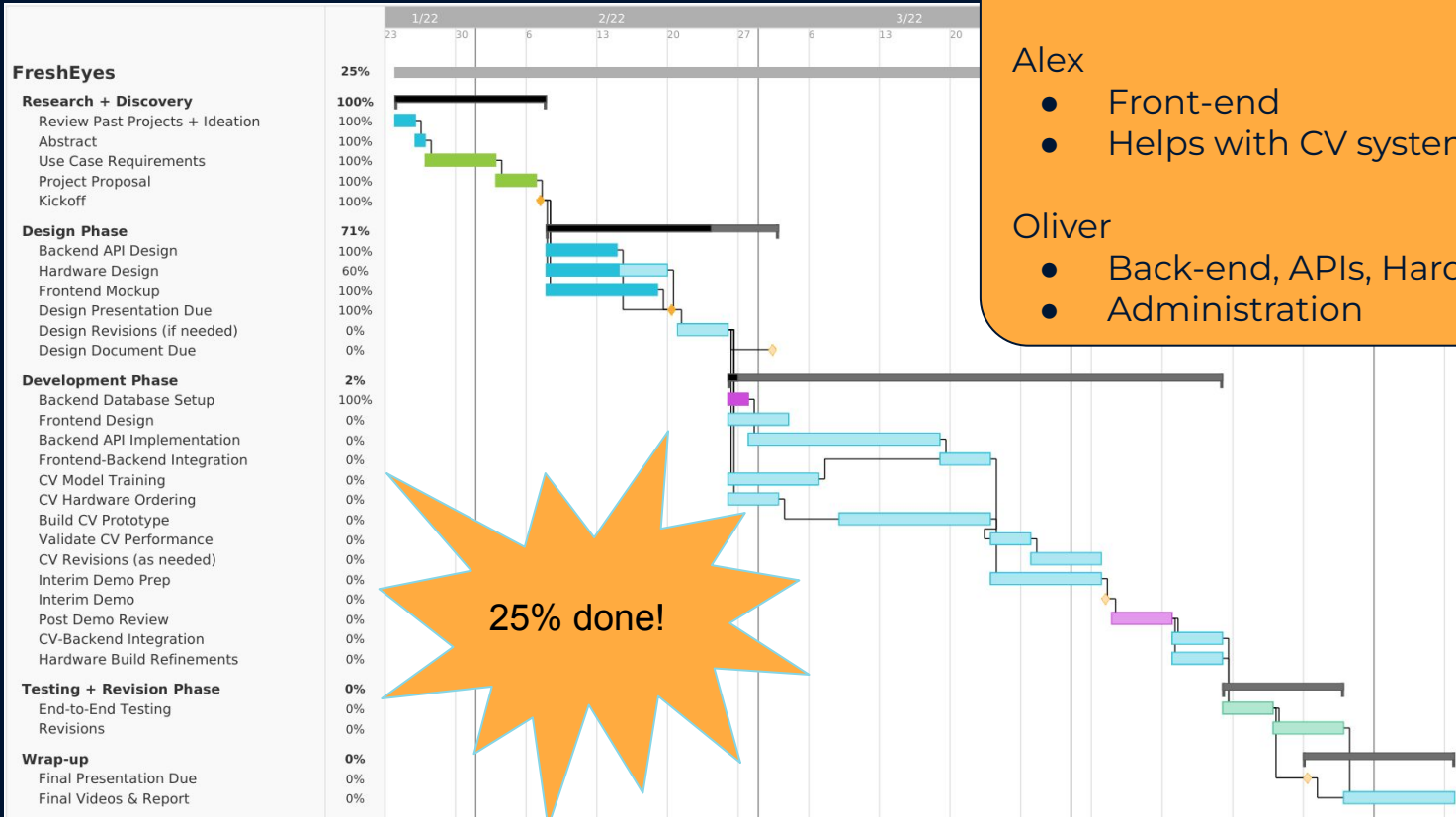
85% accuracy, < 2s response

## Interface (Front-Back-End + API)

Automated tests: Correctness of API endpoints

Feedback collection: Surveys on UX design, system performance, ease of use

# Schedule



**Samuel**

- CV System
- Testing
- Helps with front-end

**Alex**

- Front-end
- Helps with CV system, back-end

**Oliver**

- Back-end, APIs, Hardware
- Administration



The background features a dark blue field with two prominent, glowing particle trails that curve from the top corners towards the center. These trails are composed of numerous small, bright blue dots. Interspersed among these dots are diagonal streaks of light in shades of teal and orange, creating a sense of motion and energy. The overall aesthetic is futuristic and high-tech.

**Q&A**