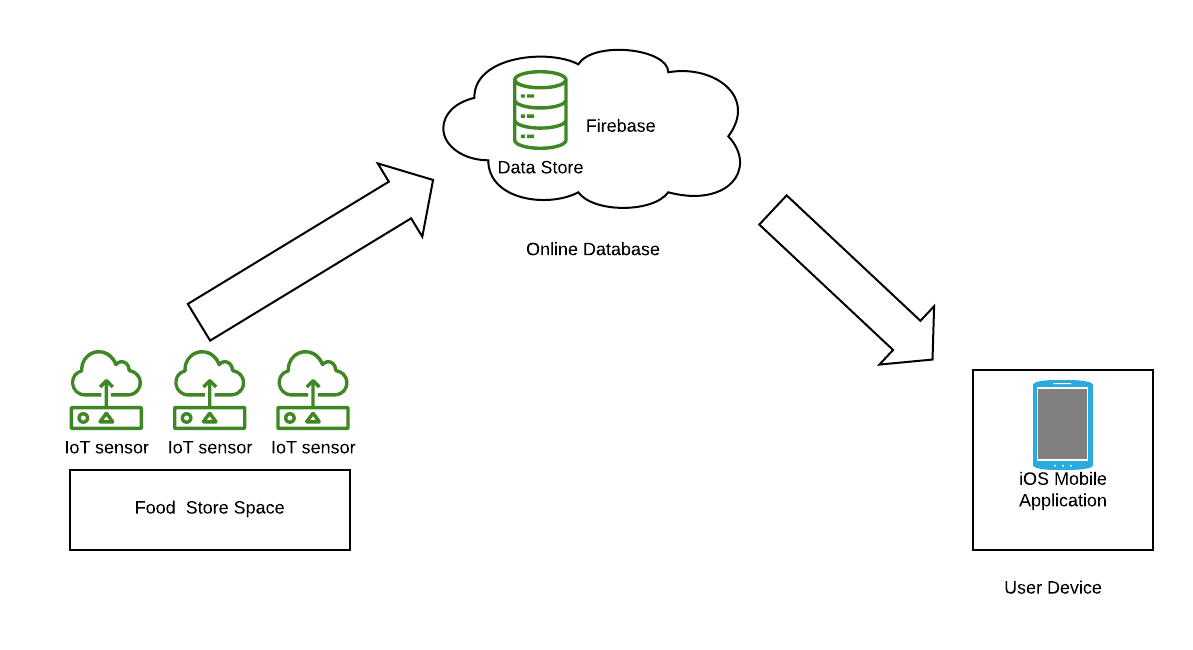
**Project Name: Smart Footorage**

Overall, the general idea of our project is to provide fruit information about days they can be stored and freshmen degree.

Below is the system architecture of our project.



**Sensor End**

At store space, a camera with catch photo of fruits in storage. Sensors will produce real-time data and upload to firebase including temperature, airPressure and humidity.

Some **libraries** used in python documents at raspberry pi:

* picamera

This library was used to make camera capture photos of fruits in storage. After certain times, camera will capture an image and save it at raspberry pi, the original image file will be deleted.

* google.cloud.vision

This API is quite important as we will use it to identify fruit types in the image captured by camera.

Each time a new photo was taken, after all fruit labels were identified, the new types of fruit will be uploaded to firebase fruit collection and removed fruit types will be deleted from the database.

* firebase\_admin

This library was used to interact with firebase data.

* Adafruit\_DHT

This one was needed to make sure dht11 sensor (humidity) to make effect.

* shelve

For offline storage and data backup, we use python shelve to store temporary data.

At raspberry pi, sensor data will be stored locally in shelve daily, every day at 23:59, local data will compare with remote data and upload missing data and all local data will be deleted at the end. Cron service was used to achieve schedule task.

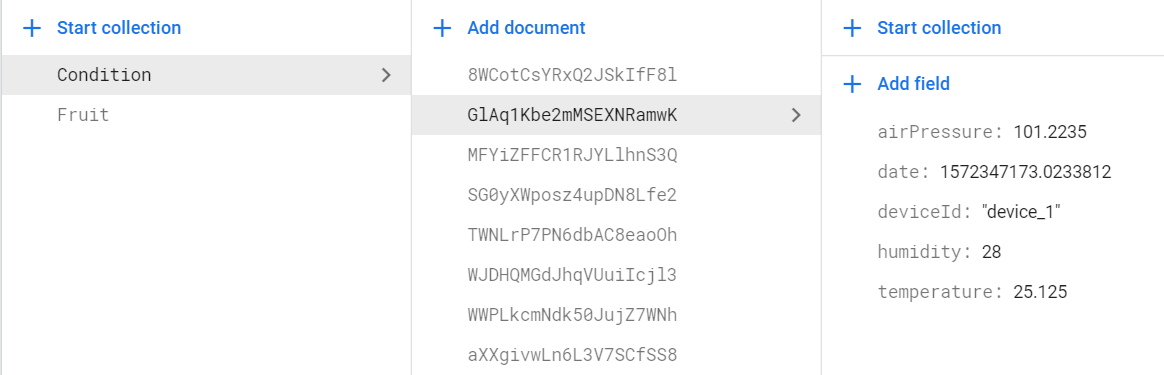
**Database**

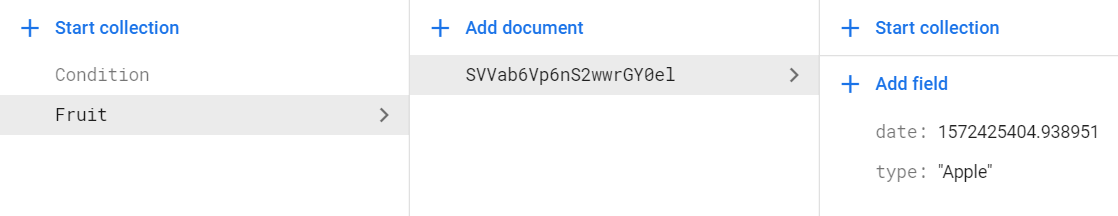
Firebase is used as database of our project.

Database architecture:

‘date’ in condition collection: time of sensor data generated.

‘date’ in fruit collection: time fruit was put in storage

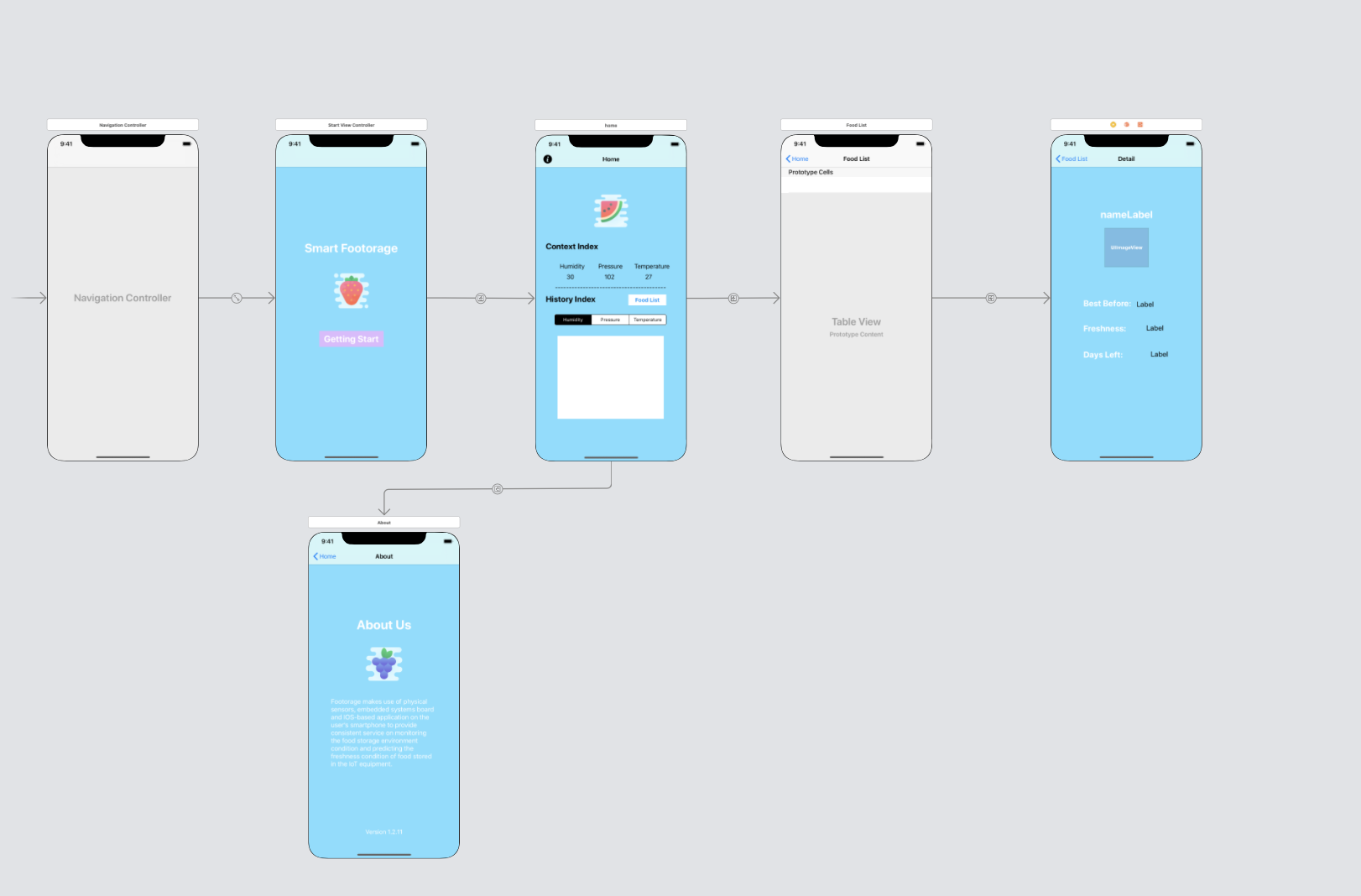




**Front End**

iOS based application was developed to be interface between users and our project.

It concludes 4 screens: home screen, food list screen, detail screen and about screen.



1. Home screen

Home screen is the main screen that controls the workflow. All of the detailed information about the environment within the storage box will be presented on the screen. The real time index of humidity, air pressure and temperature will be displayed dynamically and only the current environment index will be displayed. Also, this screen displays the history record of the environmental index. Users can check the specific number of each parameter of environment and see the historical trend in certain range of time. A food list button is displayed and user can click the button to check the real time food list.

1. Food list screen

The food list screen displays all the food items in the storage box. The items in the list may change based on the real world condition. For example, once a food is removed from the storage space, the food list will be updated and shows the real time information of all the items in the storage space, which shows the food consumed by users is removed. At the same time, all the cells in the table are clickable, which means, once a user clicks a cell, the application will jump to another screen that shows detail about the specified item.

1. Detail screen

The datil screen provide specific information about the condition of the specified item and how long it left to be consumed before it gets spoiled. The application will calculate the deadline and the days left for certain item to be consumed based on the scientific algorithms , which provides users with great insight. Also, all the information provided on the datil screen are dynamic. The calculated deadline may change if the storage condition changes, which improves user experience and provides more accurate predictions.

1. About screen

This screen briefly introduced the food storage application and shows the version of current application.