Project Concordia
Smart Home Assistant
01

The Reason
Discussion about why our project makes sense and what issue it resolves!

02

The Creation
See the parts we are using, the way it all connects and how we plan to do it!

03

The Team
Meet the team members, the sponsors, and learn about our inspiration!
What’s the main issue with Smart Home Automation? The consistent lack of cohesion!

- Smart Home Automation has been consistently **proprietary**
- Only moderate steps taken for improved connectivity between devices
- Need for an easily connectable hub for any and every smart home device

With a system that takes advantage of open source, bluetooth, wifi, and the cloud, we can solve this issue
Best of Both Worlds

**Hardware**
All of our hardware will be based around the smart objects we are controlling and the central unit for control.

**Software**
Our software will be used for connection to the cloud as well as an app for control.
The Team
5 Members, 5 Pieces

1. Smart Light Bulb & Bluetooth Module - Alex
2. Smart Camera - Nick
3. PCB Design & Smart Lock - Logan
4. Smart Speaker - Peter
5. The Cloud & Android Application - Alex, Jack, Nick
Full Block Diagram
Smart Light Bulb & Bluetooth Module

- The Bluetooth Module will be the main communication method between each of the smart devices and the central hub.
- Two way communication is used between the central hub and peripheral devices.
Parts, Progress, & Blockers - Smart Light Bulb & Bluetooth Module

Progress:
- Two way communication between central hub and peripherals

Blockers:
- One-To-Many Communication is limited on the HC-05 Module

Overview:
- UART communication will be used for bluetooth module (HC-05)
  - 9600 baud rate
  - 8 data bits
  - 1 stop bit
  - No parity bits
Demo - Bluetooth Connection
Smart Camera

- The smart camera is going to be connected directly to the cloud via Wifi to stream a live video
  - could be integrated with the smart lock for added security
- Currently uses a HAAR Cascade to detect faces in the frame and draw bounding boxes over them
  - A HAAR Cascade is a set of edge/feature filters trained to identify important areas in an image
Parts, Progress, & Blockers - Smart Camera

- **Parts** -
  - OpenMV RT1062 (embedded WIFI)
  - LIPO 2500mAh Battery Pack
  - USB LiIon/Lipoly Charge Moderator Board
  - SanDisk 128GB micro SD Card

- **Progress** -
  - Currently the board is successfully running a HAAR Cascade face detection model with bounding boxes drawn over the video stream, and it streams this to a web browser where it can be viewed by anyone on that WIFI network (where the cloud bucket can reach it)

- **Blockers** -
  - Getting the video stream connected to the Firebase Cloud from browser
Demo - Smart Camera

```
1 import sensor
2 import time
3 import network
4 import socket
5 import image
6
7 SSID = "Shot27"  # Network SSID
8 KEY = "sageZurlla"  # Network key
9 HOST = "*"  # Use first available interface
10 PORT = 8888  # Arbitrary port
11
12 # Init sensor
13 sensor.reset()
14 sensor.set_framesize(sensor.QVGA)
15 sensor.set_pixformat(sensor.RGB565)
16
17 # Sensor settings
18 sensor.set_contrast(3)
19 sensor.set_gain(celling(16))
20
21 # HQVGA and GRAYSCALE are the best for face tracking.
22 sensor.set_framesize(sensor.HQVGA)
23 sensor.set_pixformat(sensor.GRAYSCALE)
```

```
Traceback (most recent call last):
  File "<stdin>" , line 106, in <module>
  File "<stdin>" , line 62, in start_streaming
Exceptions: IDE interrupt
  OpenMV v4.5.2; MicroPython v1.26-osv-22; OpenMV IMXRT1808-MIMXRT1808DV36A
0:00 e *help() for more information.
```
Smart Lock

- Connected to the main unit via Bluetooth
- Will have an option to either lock or unlock the lock on the app
- Default state of locked door as a safety precaution
Parts

- STM32-L476VGT6 Microcontroller
- Servo Motor TowerPro MG959 - Operating speed (4.8v): 0.16sec / 60 degree at no load, Stall torque @ 4.8V: 28.0kg.cm

Progress

- Have full control of the board and connection to the central hub for bluetooth message receiving
- Connected Servo Motor to example door lock for demonstration

Blockers

- Need full control of the cloud and the android application to have full proof of concept
Demo - Smart Lock
Smart Speaker

- Incorporated into the central hub
- Will function as a feedback device for user actions with the central hub/app
- Will be used as a music player
- Will have a microphone array which will be able to process voice commands
- Will have on board storage to store audio files and recordings
Parts, Progress, & Blockers - Smart Speaker

- Parts -
  - Class D Audio Amplifier Board
    - AA-AB32231
  - Full Range Speaker @ 8 Ohm Impedance
    - PC105-8
  - Micro SD Card expansion board
    - HW-125
  - Microphone expansion board
    - Adafruit PDM MEMS Breakout

- Progress
  - Audio Playback from storage directly using DMA with ADPCM protocol
  - Audio Capture to storage using MEMS/PDM Microphone
The Cloud

- Main endpoint between central hub and Android mobile application
  - Has a realtime database with data fields relating to all peripheral devices
  - Has a storage bucket for video cam stream

---

https://smart-home-assistant-dea36-default-rtdb.firebaseio.com
door_locked: true
light_on: true
test: "Hello from ESP"
value: 7385
Parts, Progress, & Blockers - The Cloud

- **Parts**
  - ESP-8266 NodeMCU wifi board
    - Communicates through UART and HTTPS
  - Firebase Realtime Database and Storage Bucket
- **Progress**
  - Successfully parsing and receiving HTTPS messages from the wifi module and mobile app, both devices can modify data fields
- **Blockers**
  - Database to board communication has high latency
Demo – Android App
PCB

4 Layers
66x73mm
Goals - End of Winter

- Complete bluetooth communication between all peripherals
- Integrate smart light with cloud for full project path proof of concept
- Complete all devices communicating with central hub
- Get Smart Camera Stream viewable over wifi
- Complete setting up the Firebase realtime database and bucket and integrating with wifi board and mobile app
- Test custom PCB with all connected devices to see if a revision is necessary
Goals - Spring

- Integrate smart camera with cloud and mobile app
- Complete a fully functioning mobile app and get it integrated with the cloud
- Have a functional and pleasing to the eye demonstration for each of the devices
- Refine PCB to work seamlessly with components
- Complete cloud work to manage data transfer between the mobile app and the central hub
Risk Assessment

- Unable to form secure connection between smart camera and the cloud
- Battery with smart camera might not be able to handle running large computer vision model for long periods of time
- Firebase cloud could continue to have high latency issues
- Concurrent Tasks (i.e. playing audio and recording from the microphone simultaneously / receiving data from the cloud while playing or recording audio)
Summary

Just in case you missed anything!

- Smart Devices controlled from one central hub
- Phone application for ease of control from anywhere with wifi connection
- Real time updates for any smart devices connected
- Video security camera with human recognition
- Cloud controlled data sending
THANKS!

Do you have any questions?
github.com/Smart-Home-Assistant

Acknowledgments

- Dr. Yogananda Isukapalli
  - TA Brian Li
  - TA Eric Hseih
  - TA Alex Lai

Feel free to reach out!