University of Massachusetts, Amherst
College of Engineering

ITS LIT

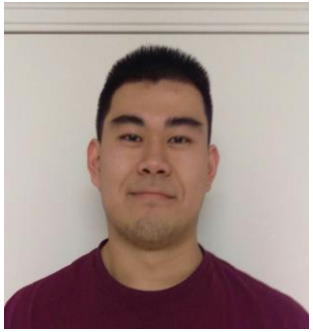
Cumulative Design Review

Senior Design Project ‘17

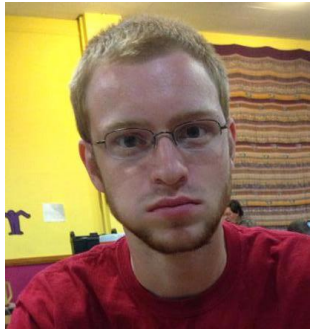
Department of Electrical and Computer Engineering

Meet The Team

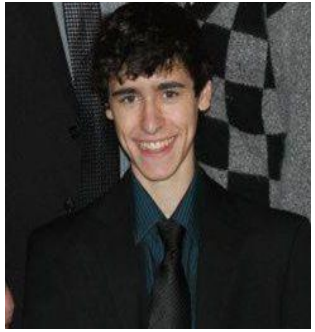
Advisor:
Professor David McLaughlin



Tommy Zhen
CSE



Michael Polin
CSE

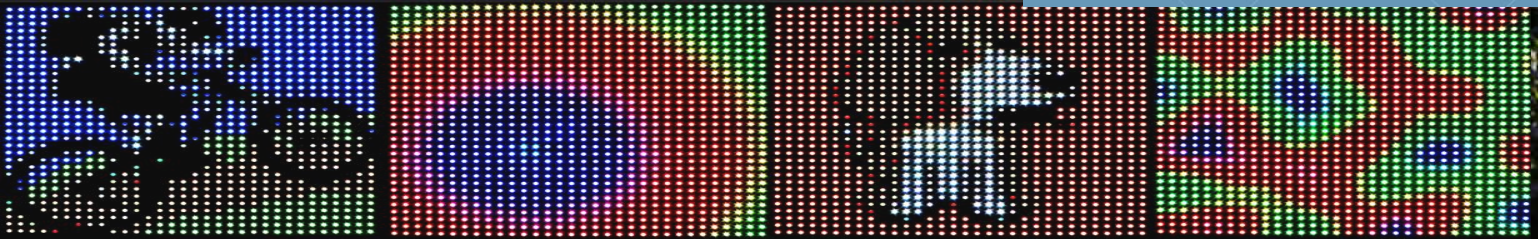


Patrick Browne
EE

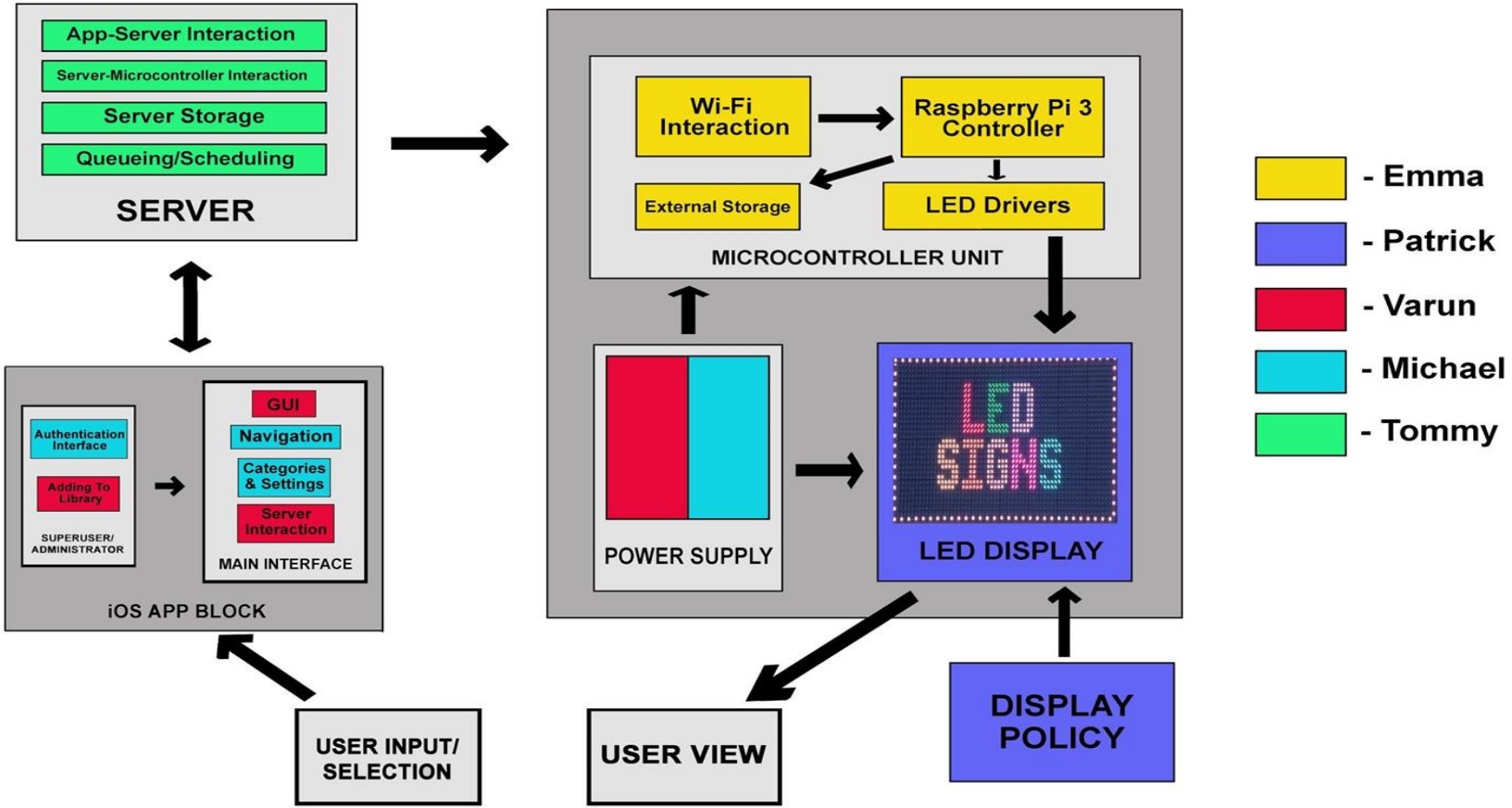
Varun Menon
EE

Emma Bryce
EE

ITS LIT



Block Diagram



Promised CDR Deliverables

PJ

- PCB design & display policy complete

Tommy

- Server and app transmit/receive data and queuing algorithm has been started

Emma

- Raspberry Pi computer controls a 2-dimensional display for CDR and renders at least one image type from server

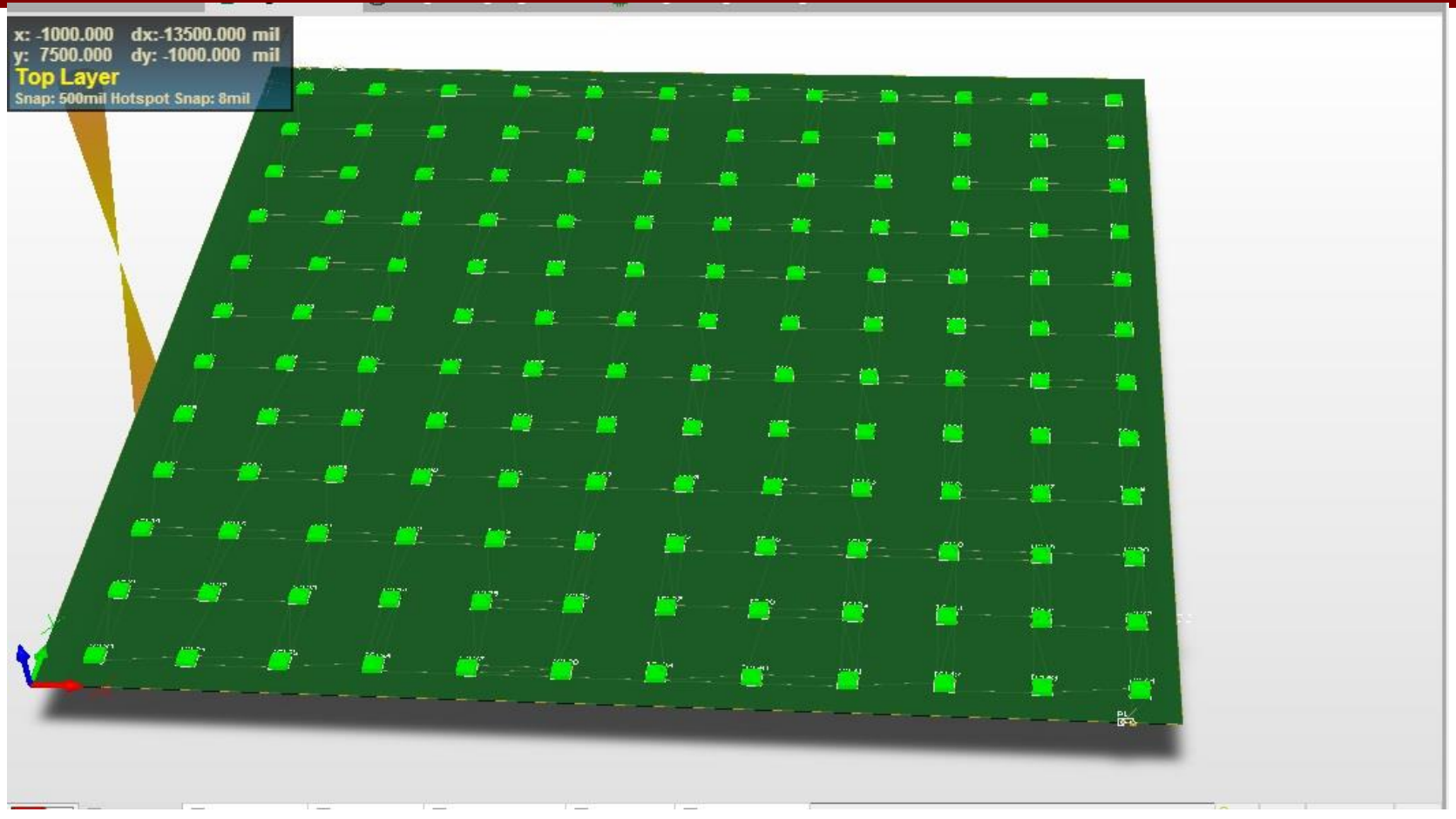
Mike

- Multiple views for categories/settings complete and help with app/server interaction

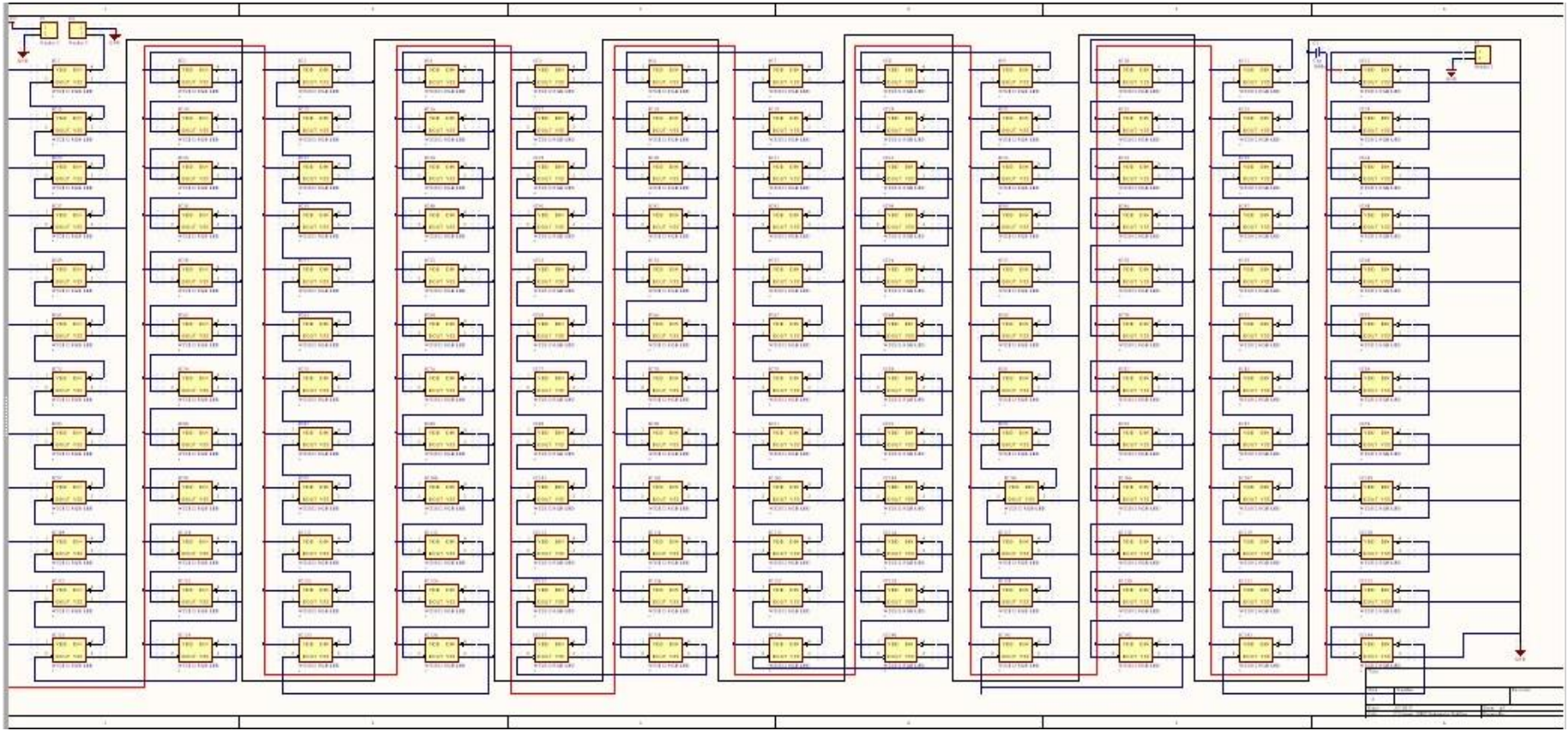
Varun

- App is able to send, receive requests and interact with server

Neopixel Display PCB



PCB



PCB

- 9 total PCB’s
- Each is 1’ x 1’
- Pixel Pitch of 1”
- Each Board has 144 LED’s
- Each Board has a Data line, Power, and Ground
- PCB designed using Altium

Display Policy

- Working on a proposal to send to the University Public Art Committee
- They will determine if we can deploy the display

Display Control

Deliverable: Raspberry Pi computer controls a 2-dimensional display for CDR and renders at least one image type from server

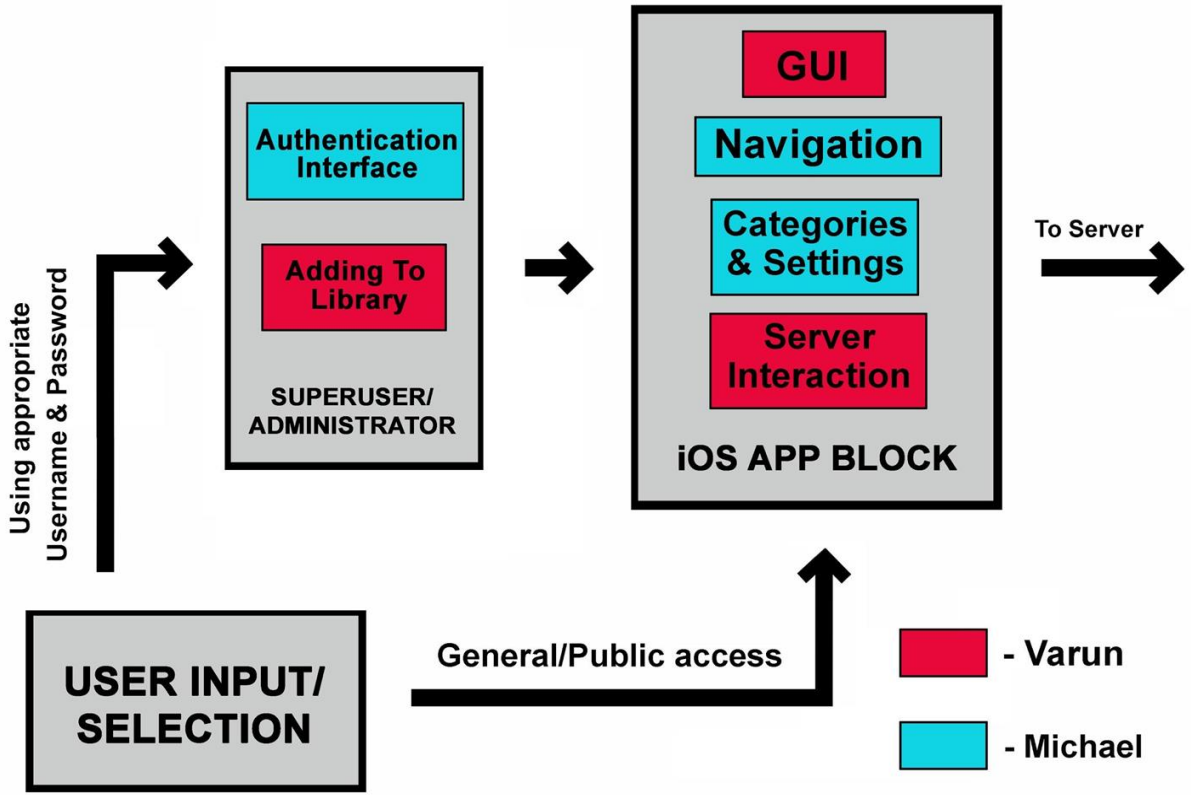
Delivered: Pi renders jpeg images on 84 RGB LED prototype display

- Upgraded from Arduino Uno to Raspberry Pi 2
- jpg format files resized to fit display with good results
- Hardware calibration Gamma correction for brightness/duty cycle done via 255 value lookup table
- Neopixel 2D prototype used
- Logic level shift between Pi and pixel data signal

To Do:

- Drive upgraded 1296 LED display

iOS App

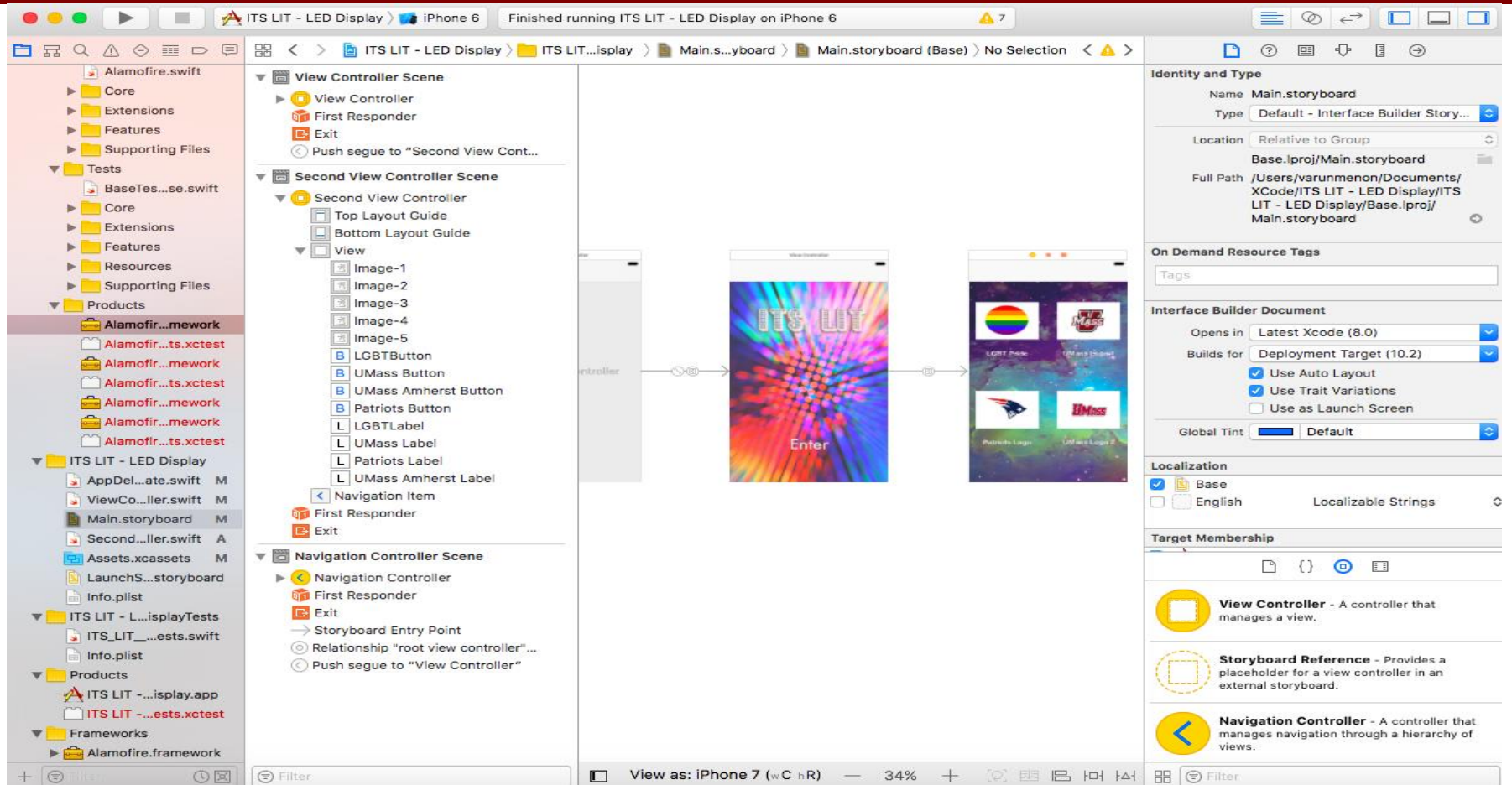

Authentication Interface
Adding To Library
SUPERUSER/ ADMINISTRATOR
GUI
Navigation
Categories & Settings
Server Interaction
iOS APP BLOCK
To Server
Using appropriate Username & Password
USER INPUT/ SELECTION
General/Public access
- Varun
- Michael

iOS App

Tasks Completed:

- Intermediate GUI and Navigation completed
- Sample library with presets
- App is able to take in user input (image selection) and submit a request to the apache server
- App sends data to server through ‘post’ request using the Alamofire library
- App runs on iOS 10.2 simulator for iPhone 6/6+/7/7+

iOS App


ITS LIT - LED Display
iPhone 6
Finished running ITS LIT - LED Display on iPhone 6
View Controller Scene
Second View Controller Scene
Navigation Controller Scene
Enter
Identity and Type
Interface Builder Document
Localization
Target Membership

iOS App

Tasks Remaining:

- Finish up advanced GUI and Navigation
- Populate entire library with categories and presets
- Super-user authentication, privileges and settings
- App geo-fencing (using Location services & Maps)

Power Supply

- Each LED draws 60 mA when white (max)
- Each board (12 x 12 = 144 LED's) draws 8.64 A when all white (max)
- Total power consumption for the entire 9 boards is 388.8 W when all white (max)

Server

- Apache webserver hosted on a Raspberry Pi
  - Requests from app are received via an HTTP post request through a PHP script
  - Requests are then stored on an SQL database
- Queuing algorithm written in python and has access to database to retrieve requests
  - FIFO queue
  - Manages how long each image stays up on display (set time)

Integration Achieved and Required

Achieved design integration:

- App and server integration
- Pi and display hardware integration

Required still:

- Pi and server integration
- Power supply and hardware integration

Demo

Proposed FPR Deliverables

PJ

- Display complete, incl. integration with power supply and Pi

Tommy

- Display is able to receive requests from queue and server

Emma

- Pi able to render images on full display incl. some hardware interface

Mike

- App geofence, super-user authentication process & display chassis complete

Varun

- Advanced app GUI, super-user privileges & power supply complete

Questions?