



KARE Team Final Presentation

CS481.001, Central Washington University, Spring 2023

Table of Contents

- Project Overview
 - KARE Application, Teams, Client, and Problem Statement
- Development
 - Mobile Development
 - Requirements, Design, Tools
 - Desktop Development
 - Requirements, Tools, Design
 - Database Development
 - Requirements, Design, Tools, Bugs
- Application Demonstrations
 - Mobile App
 - Desktop App
- Conclusion

The Project

- KARE
- Medically Complex Children
- Structure for Parents
- Three Core Components:
 - Mobile
 - Desktop
 - Server



Project Teams

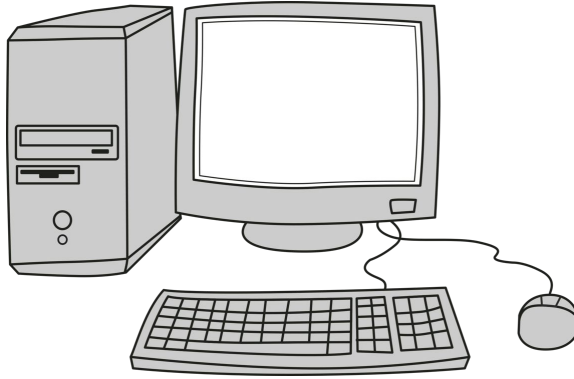


Mobile Team

Hunter Berg

Logan Halverson

Conner Kaul



Desktop Team

Zak Kappenman

Megan Phinney

Stephan Schuller



Server Team

Ryan von Fiedler

Ardit Miftaraj

Supervisor & Customers

- Supervisor: Dr. Szilard Vajda
 - Department of Sciences, Computer Science
- Customers: Dr. Donald Davendra & Dr. Magdalena Bialic-Davendra
 - College of Sciences, Computer Science
 - College of Business, Economics
 - Aid in the Care of MCC
 - Help Parents and Doctors Maintain a Good Record of Care

Problem Statement

The transition from hospital care to home care is a difficult one, and KARE seeks to make that process easier.

Mobile Application

Mobile Tools

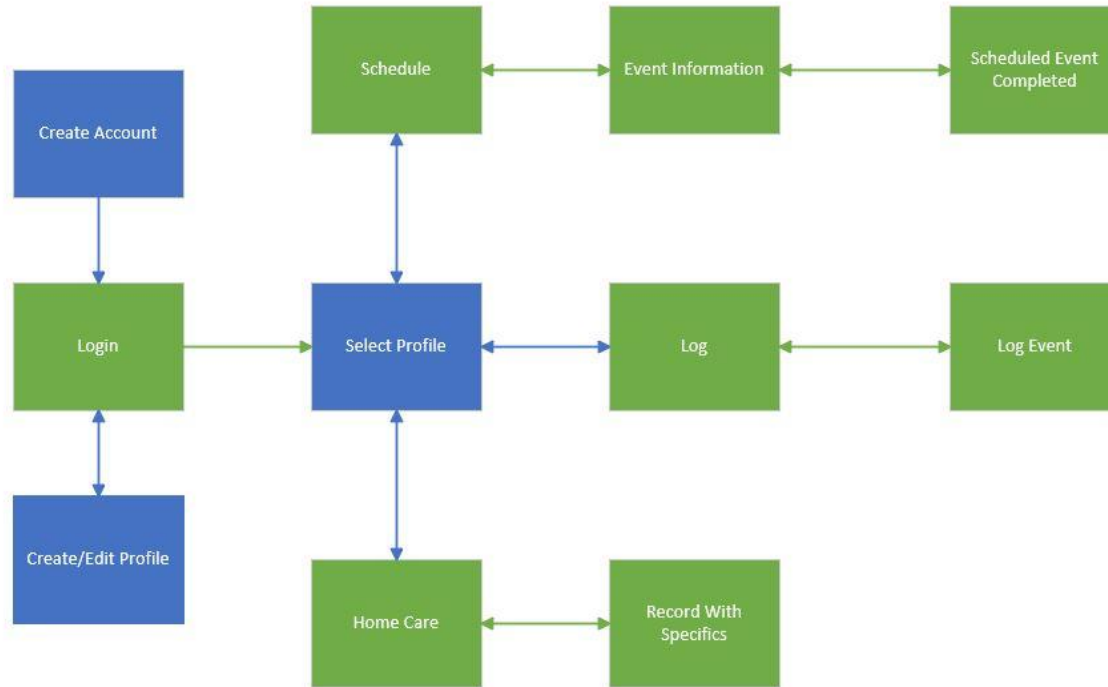
- React Native
 - JavaScript Framework
- Node JS
 - JavaScript Server Backend
- Expo
 - Mobile Platform Built on React-Native



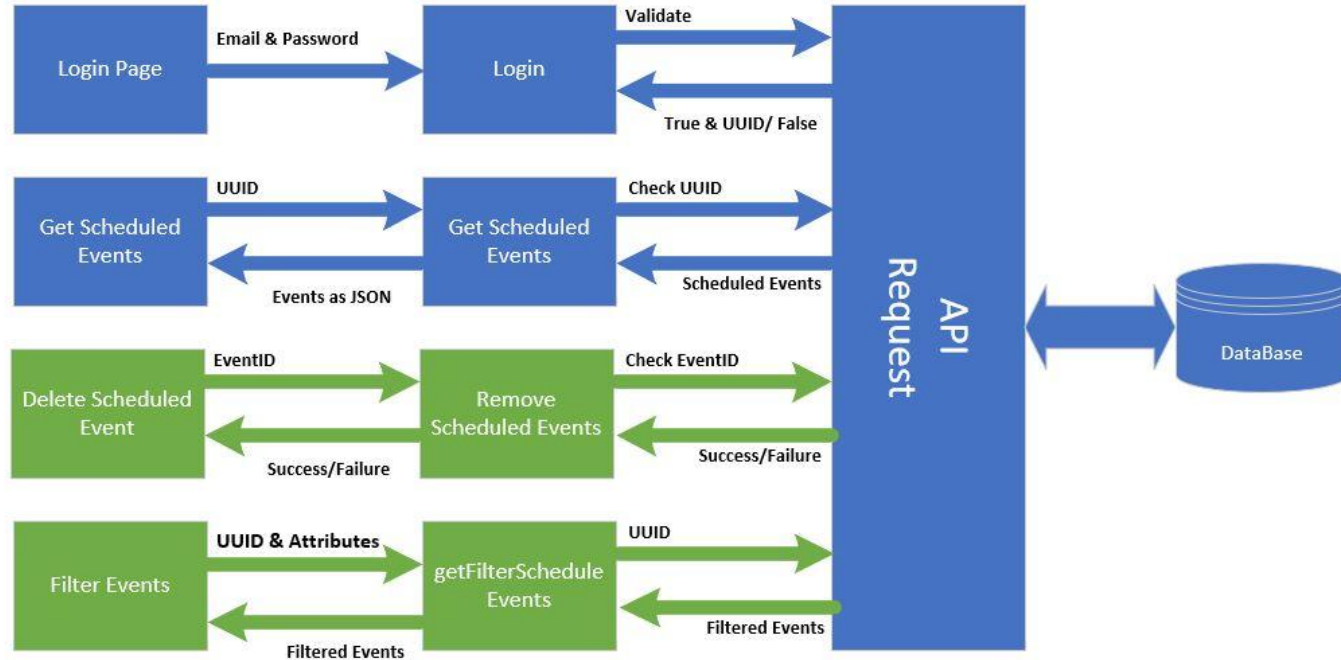
Mobile Progress

- Overhauled User Interface & User Design
- Fixed Various Bugs & Several Features
 - Filter, Status Bar, Logs, Merged Log & Home Care Pages
- Added Functionality
 - Recurring Scheduled Events, Event Deletion, IP Input, New Data Log Page
- Improved API Calls
 - Added Deletion Call & Fixed Scheduled Event API
- Began Offline Implementation
 - Offline Login & Offline API Call Storage

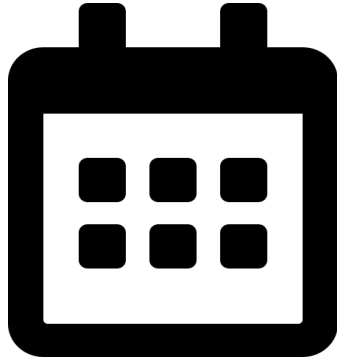
Mobile High-Level Design



Mobile Low-Level Design



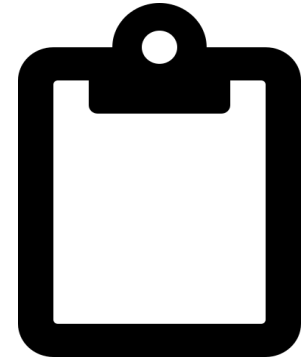
Mobile App



Schedule



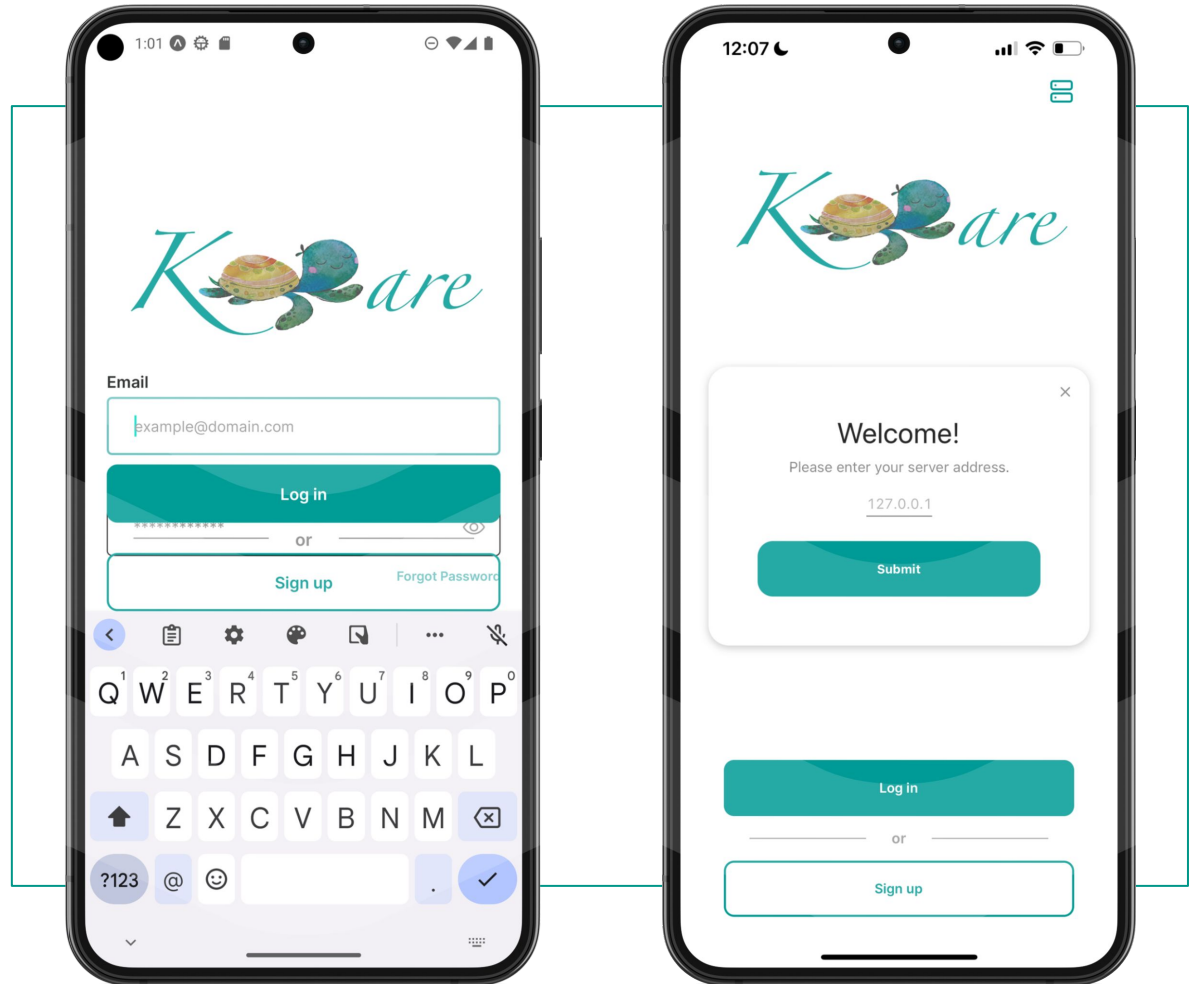
Home Care



Log

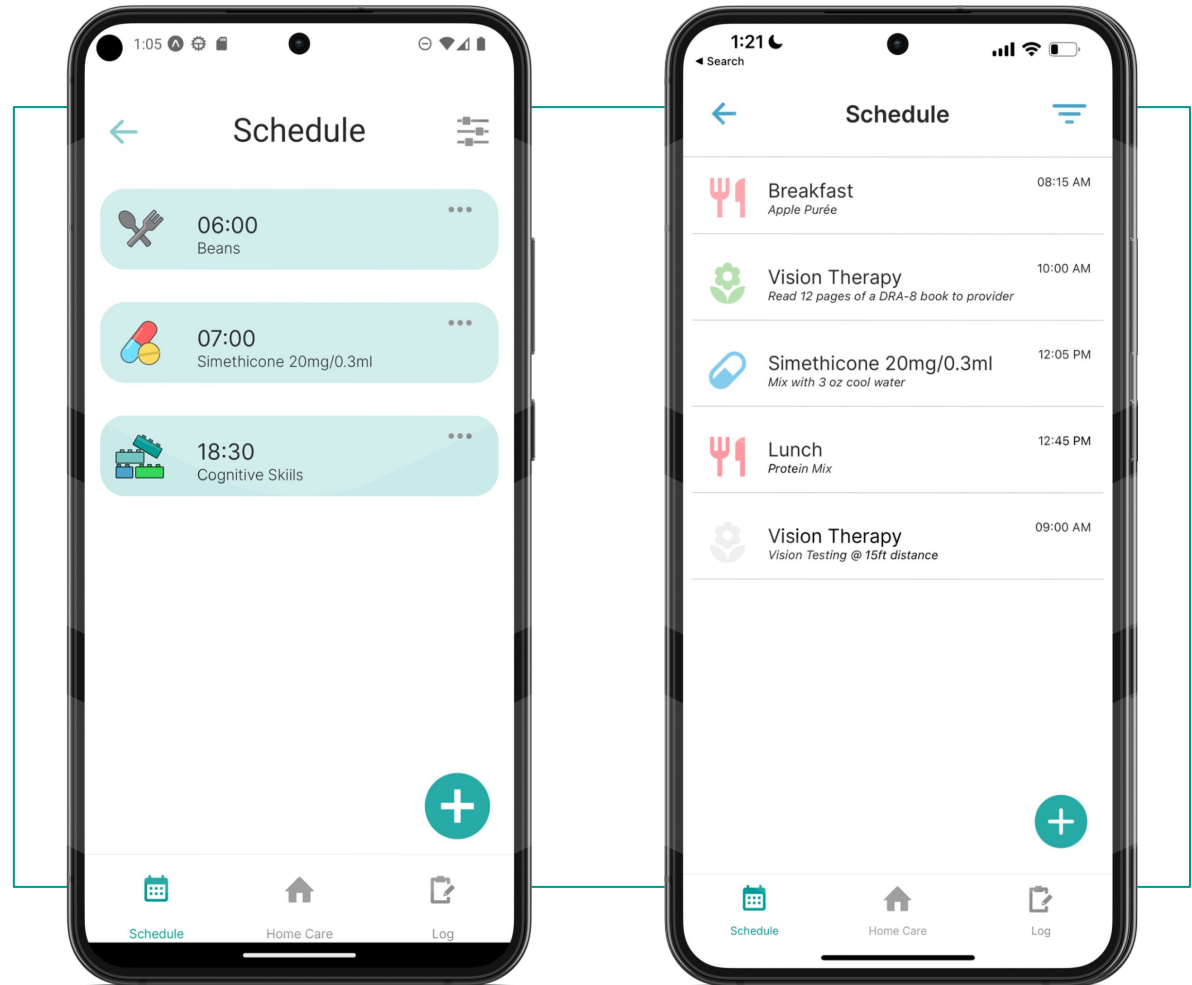
Login

- Server Address Changing
- Responsive Scaling



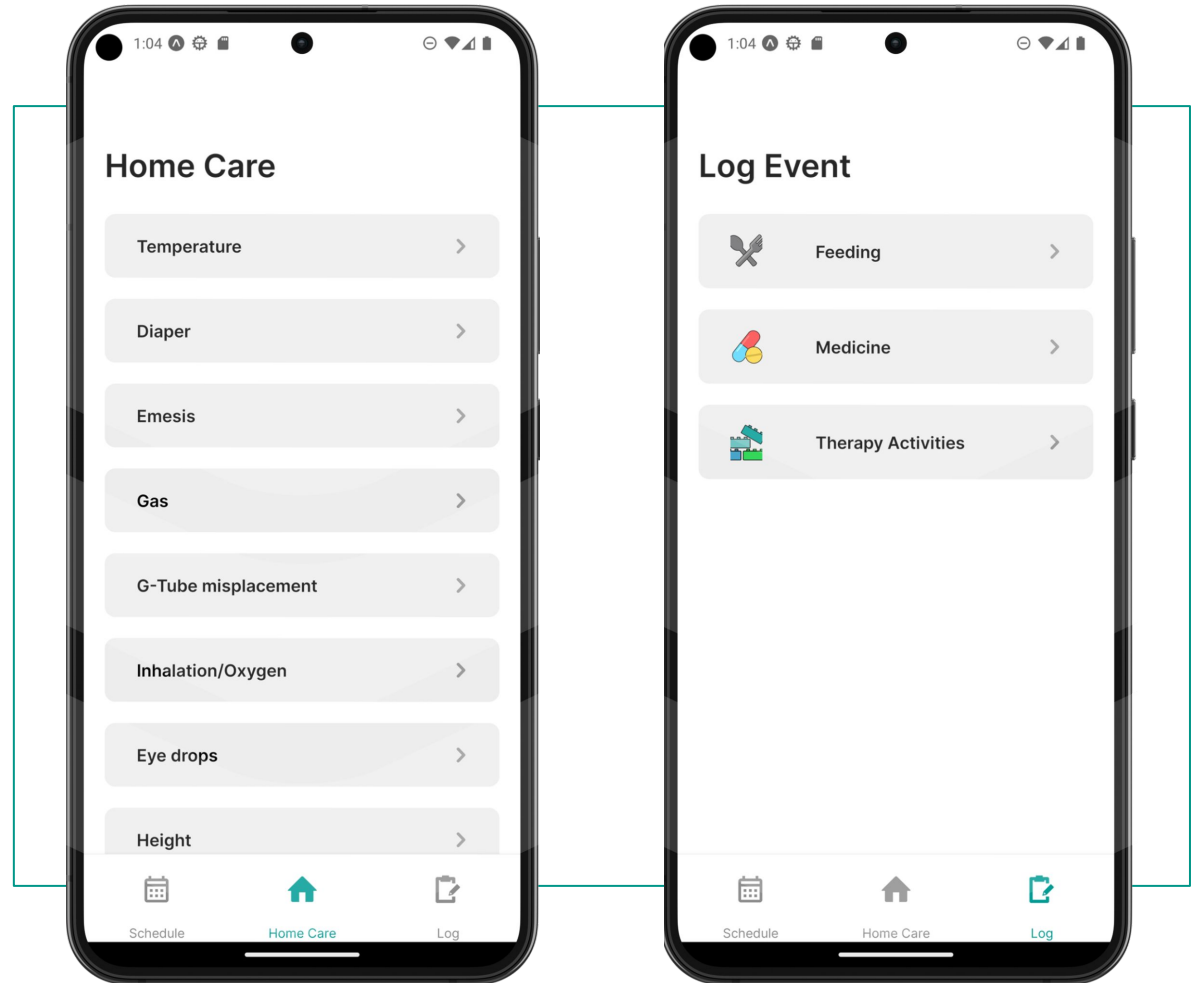
Schedule

- Scheduling Repeating Events
- Filtering, Logging, Deleting Events



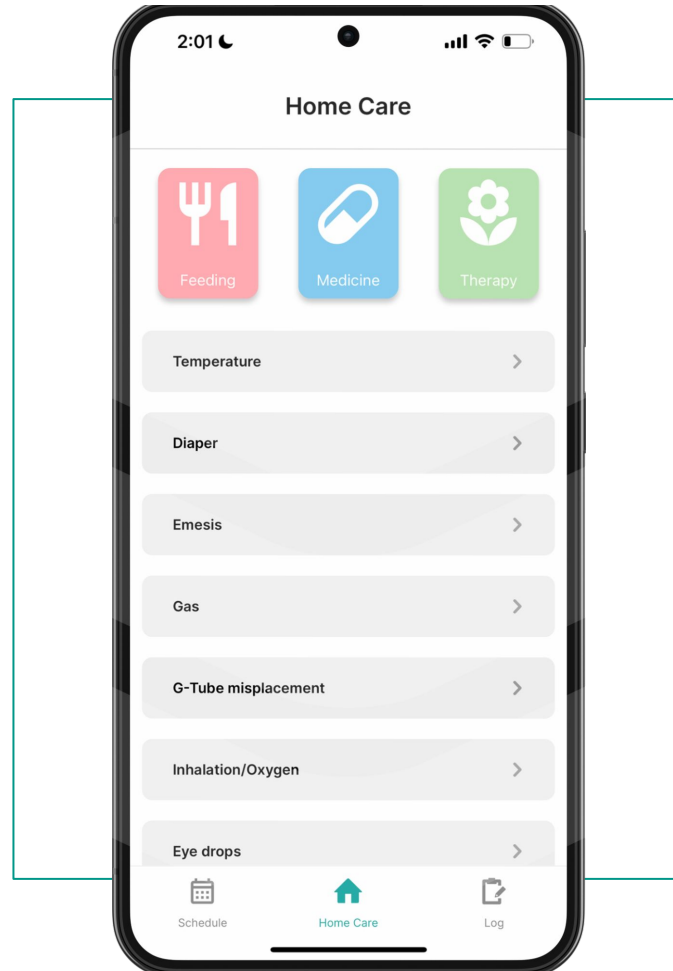
Home Care - Old

- Manual Logging
- Separate Screens



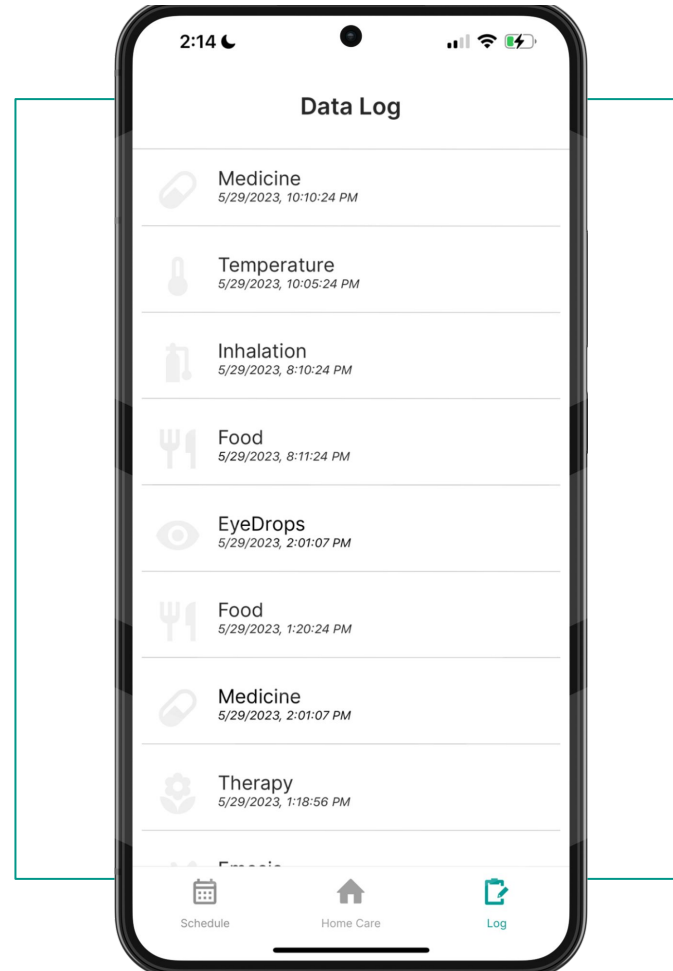
Home Care - New

- Emphasize the Most Important Events
- Centralize Manual Logging



Data Log

- See Recent Entries
- Stored for One Week Only



Mobile Test Cases

Bugs Fixed As a Result of Test Cases

- 21 Mobile Development Trello tasks accomplished
- Interchangeable usage of cancel and back
- Security issues with scheduled events
- Schedule filtering not working correctly
- Scheduled events had no capacity for repetition
- Scroll functionality not working on Android devices

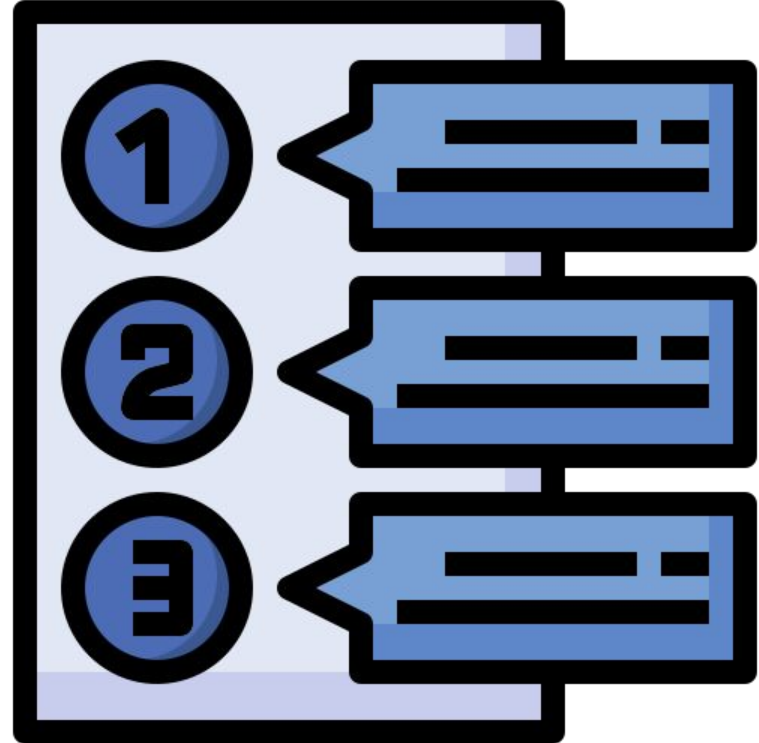


Desktop Application

Desktop Requirements

Previous Iteration

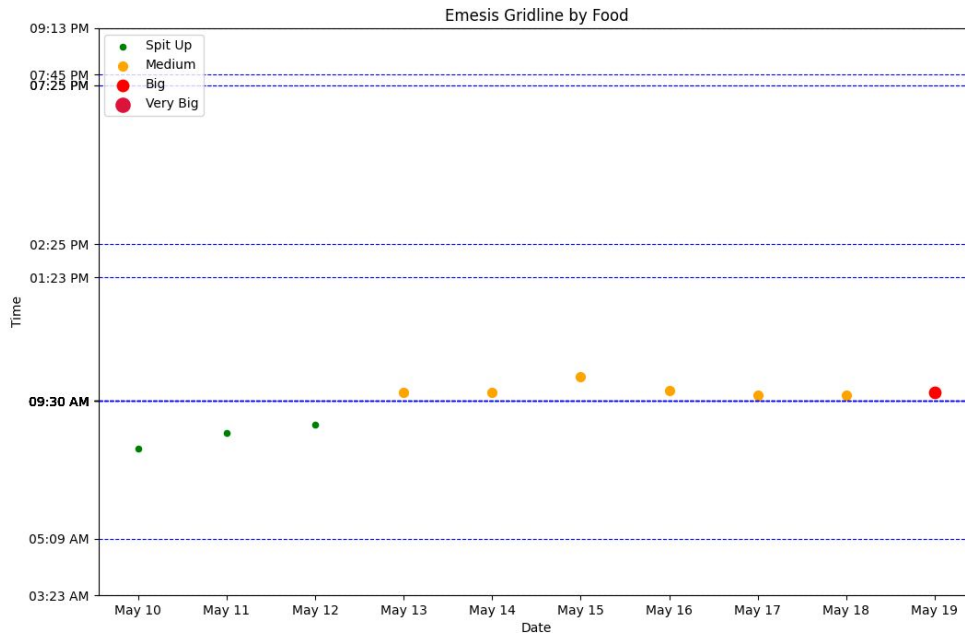
- Data Visualization
- Data Storage
- Exporting Graphs and Data
- Log New Entries



Desktop Requirements Cont.

Current Iteration

- Improve Data Visualization
- Implement a Log Page
- Clear the Server of Old Events
- Testing and Improving
- Offline Functionality



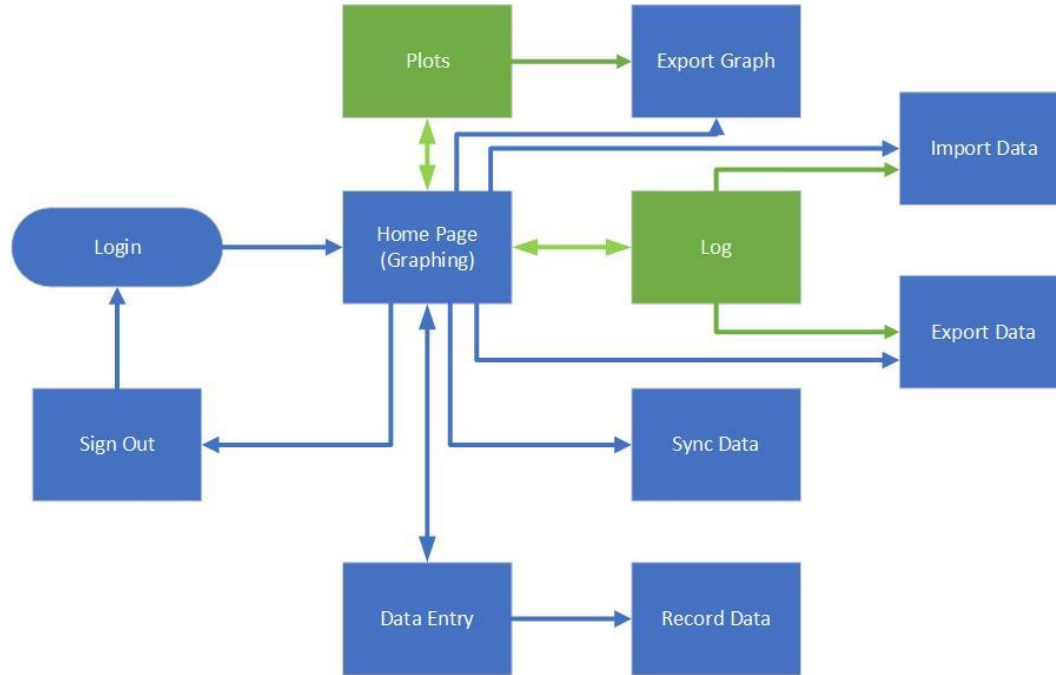
Desktop Tools

Python Packages and Libraries

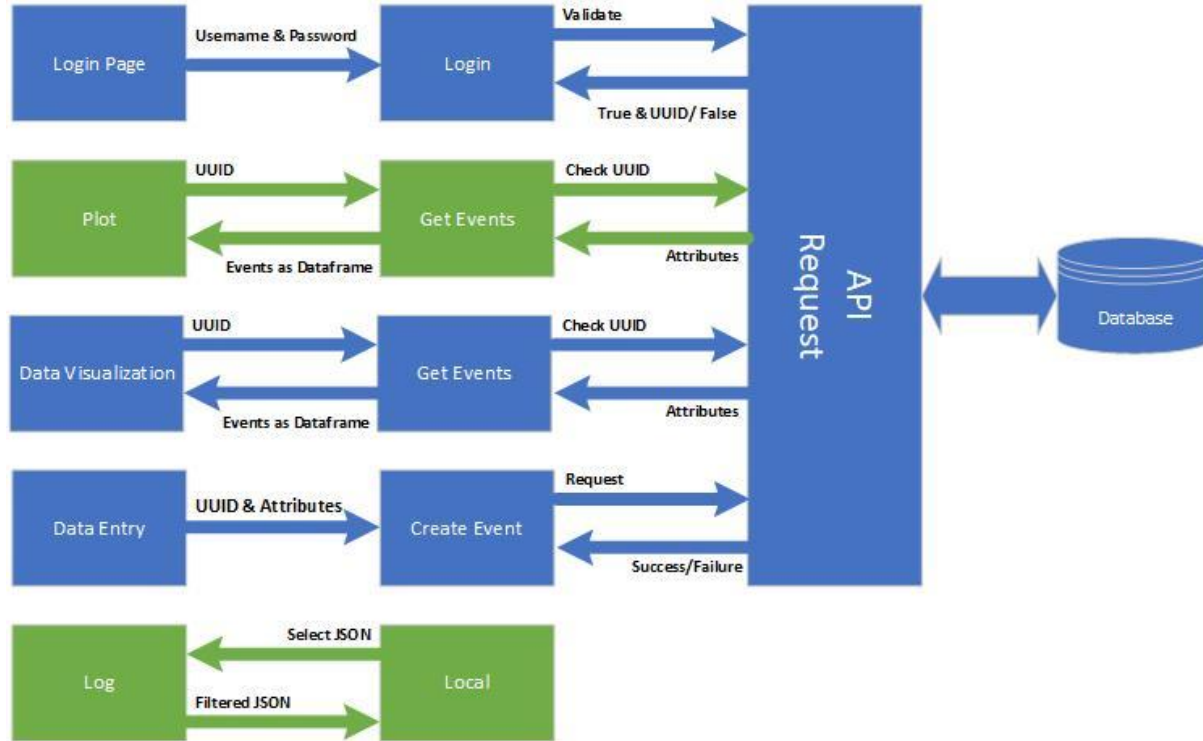
- Matplotlib
- Pandas
- PIL
- Requests
- PyInstaller
- TKinter



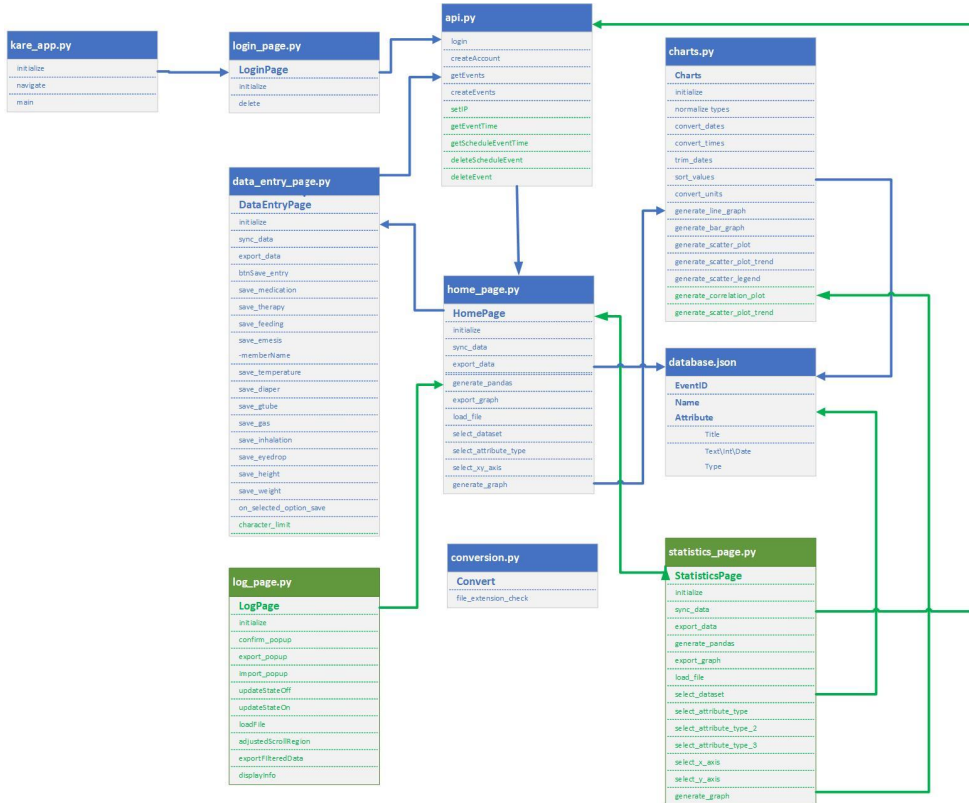
Desktop High-Level Design



Desktop Low-Level Design



Desktop Low-Level Design Continued



Desktop Testing

Stages of Testing

- Legacy Testing
- New Requirements Testing
- “Quality of Life” Testing
- Final Exploitative Testing

By the Numbers

- Started with 8 Test Cases
- Ended with 26 Test Cases
- Final Test Report 85 Pages

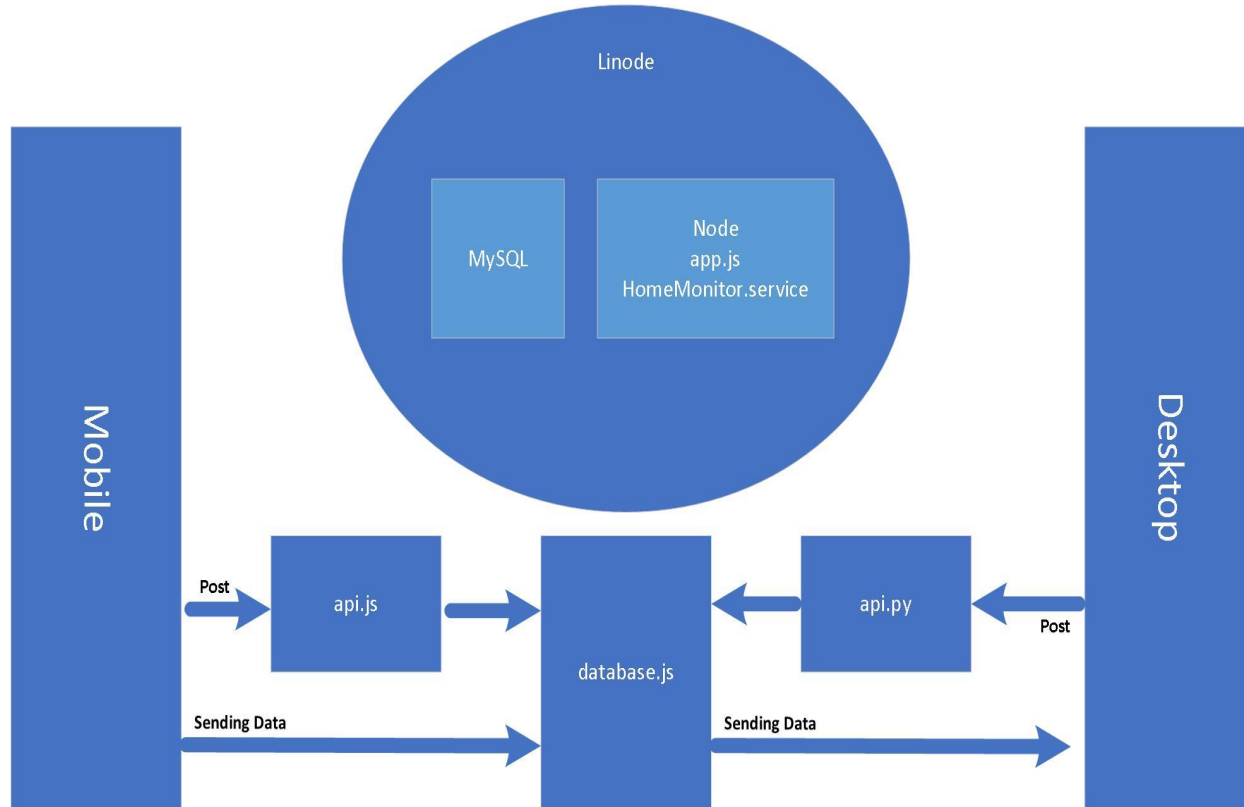
Server Component

Server Requirements

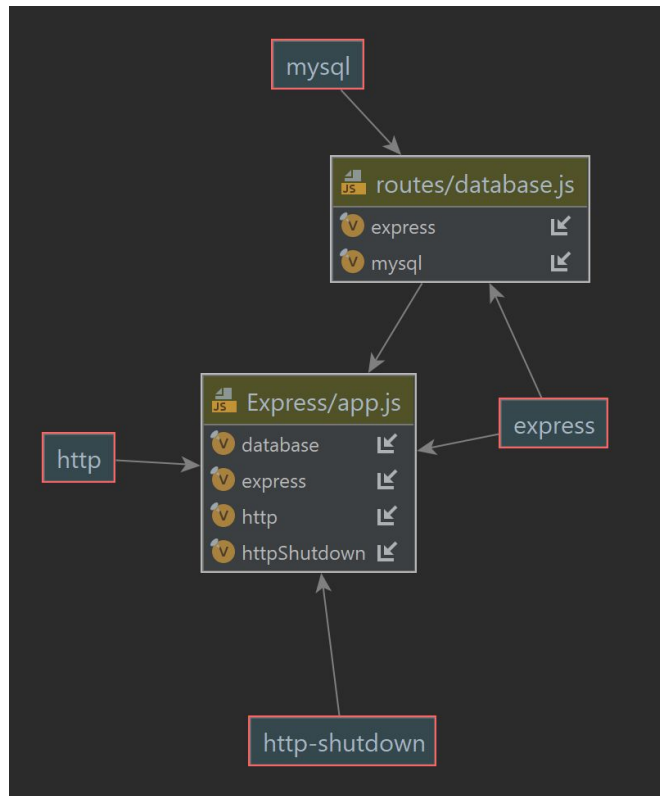
- Bug Fixing
- Stability
- API Creation
- Data Normalization
- Security Fixes



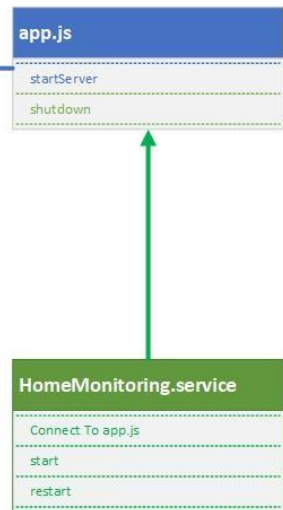
Server High-Level Design



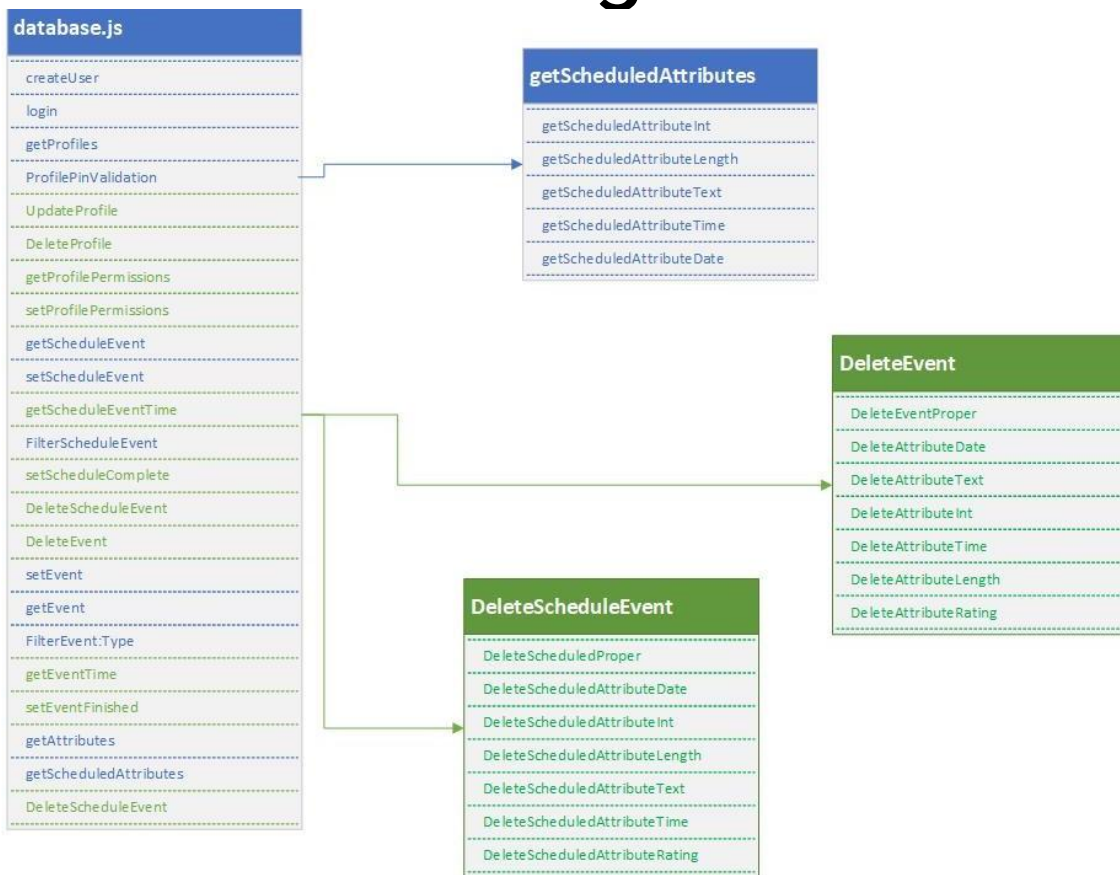
Server Low-Level Design



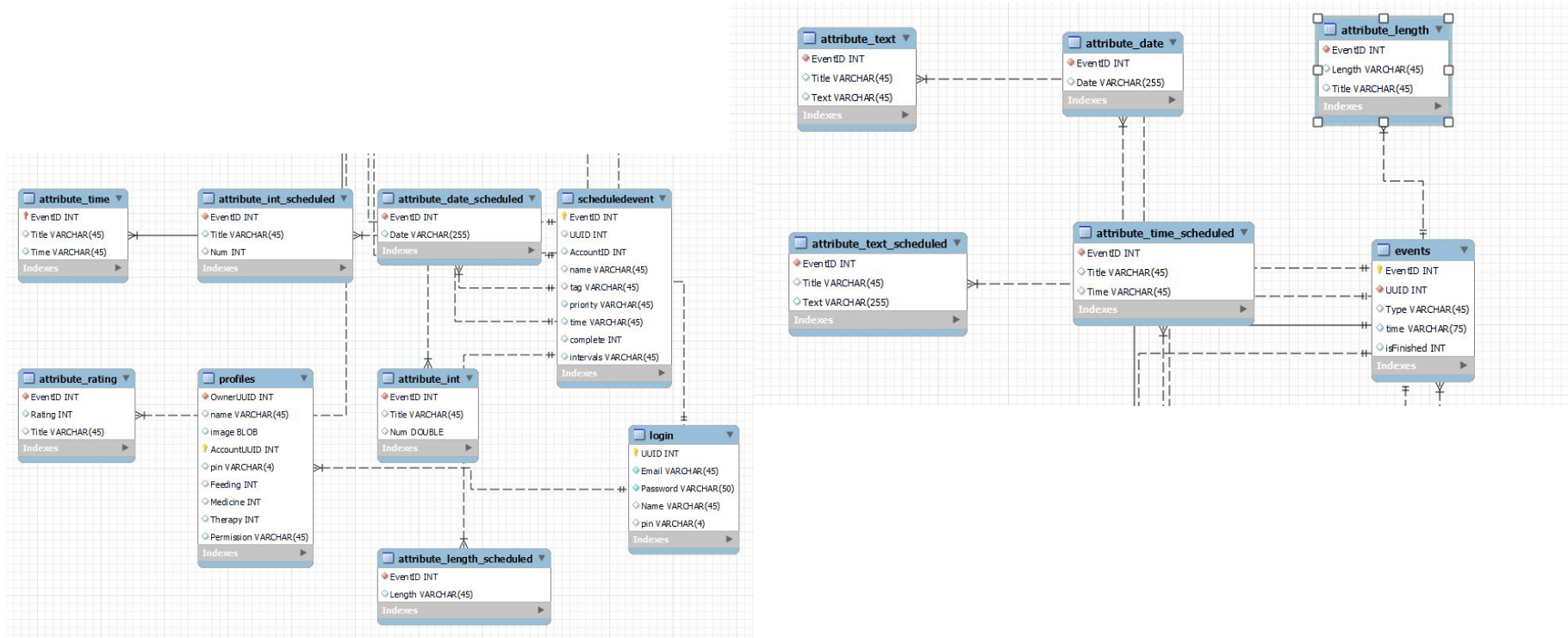
database.js
createUser
login
getProfiles
ProfilePinValidation
UpdateProfile
DeleteProfile
getProfilePermissions
setProfilePermissions
getScheduleEvent
setScheduleEvent
getScheduleEventTime
FilterScheduleEvent
setScheduleComplete
DeleteScheduleEvent
DeleteEvent
setEvent
getEvent
FilterEvent.Type
getEventTime
setEventFinished
getAttributes
getScheduledAttributes
DeleteScheduleEvent



Server Low-Level Design



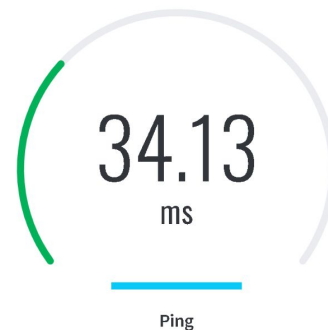
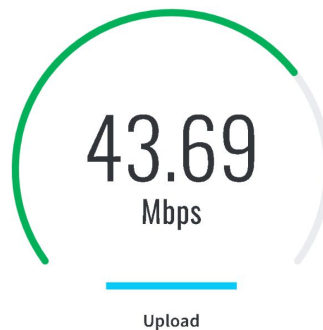
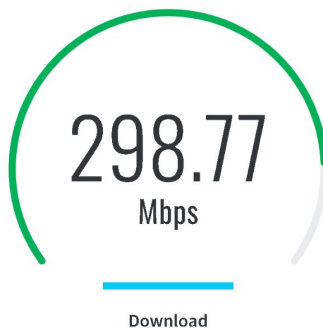
Database - Entity Relationship Diagram (ERD)



Server Tools

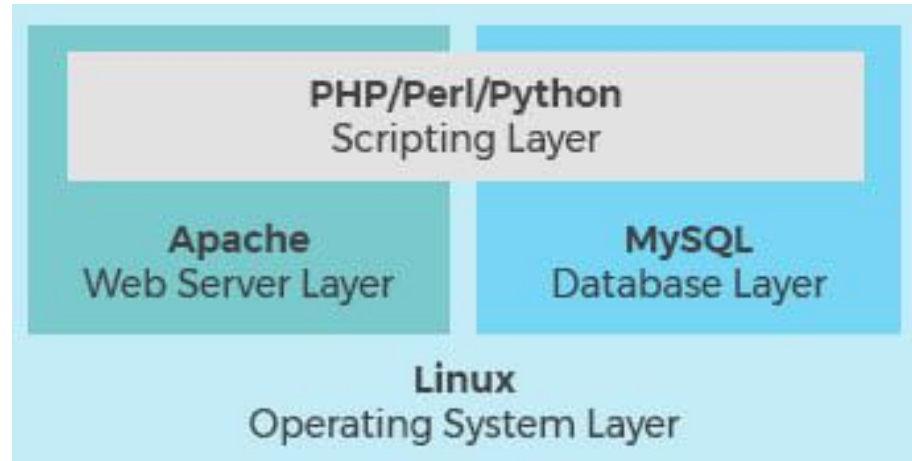


- Nanode 1GB
- SSL Certification
- SSH Access
- Easy to Clone
- Easy to Transfer



Server Tools

- Linux (Ubuntu 20.04)
- Node
- MySQL
- PHP
- Systemd



Server Bugs

- Port 80 Resource Contention
- Data Confusion
- API Typos
- SQL Injection Attacks

Mobile Demo

Logan will now livestream a mobile demo

Desktop Demo

Megan will now livestream a desktop demo

Conclusion

- Rigorous Testing
- Functionality
- User Friendly
- Security
- Deployment



Any Questions?