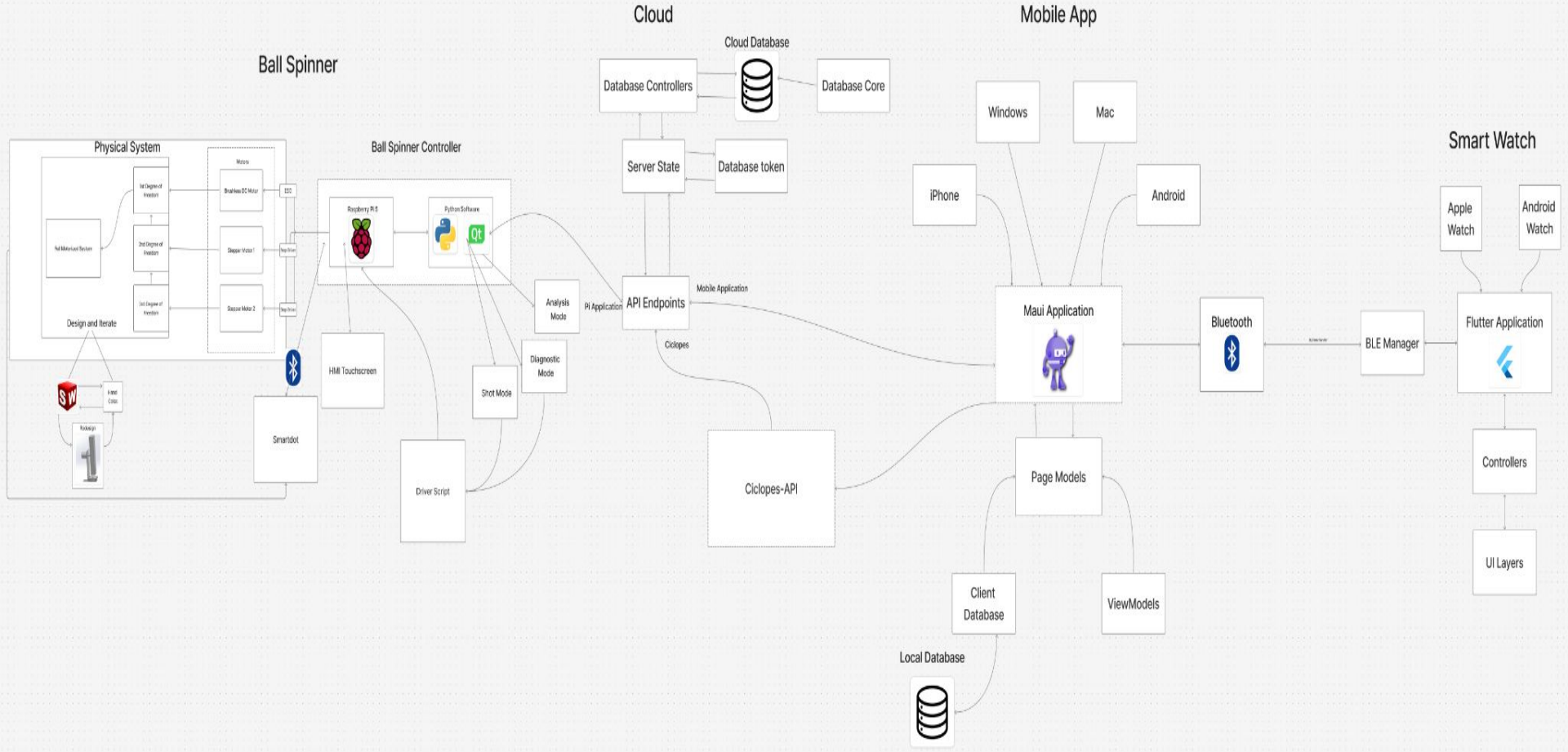
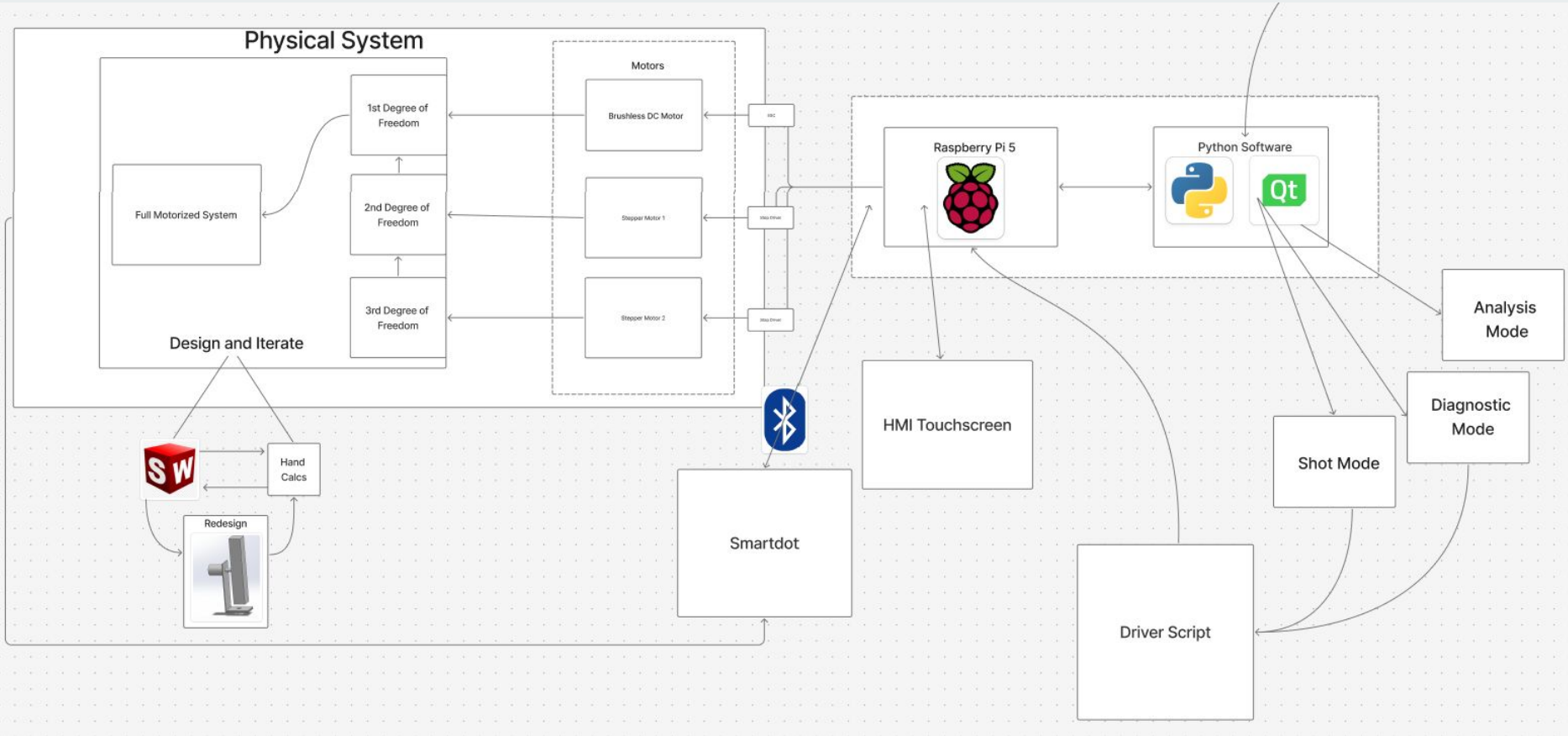
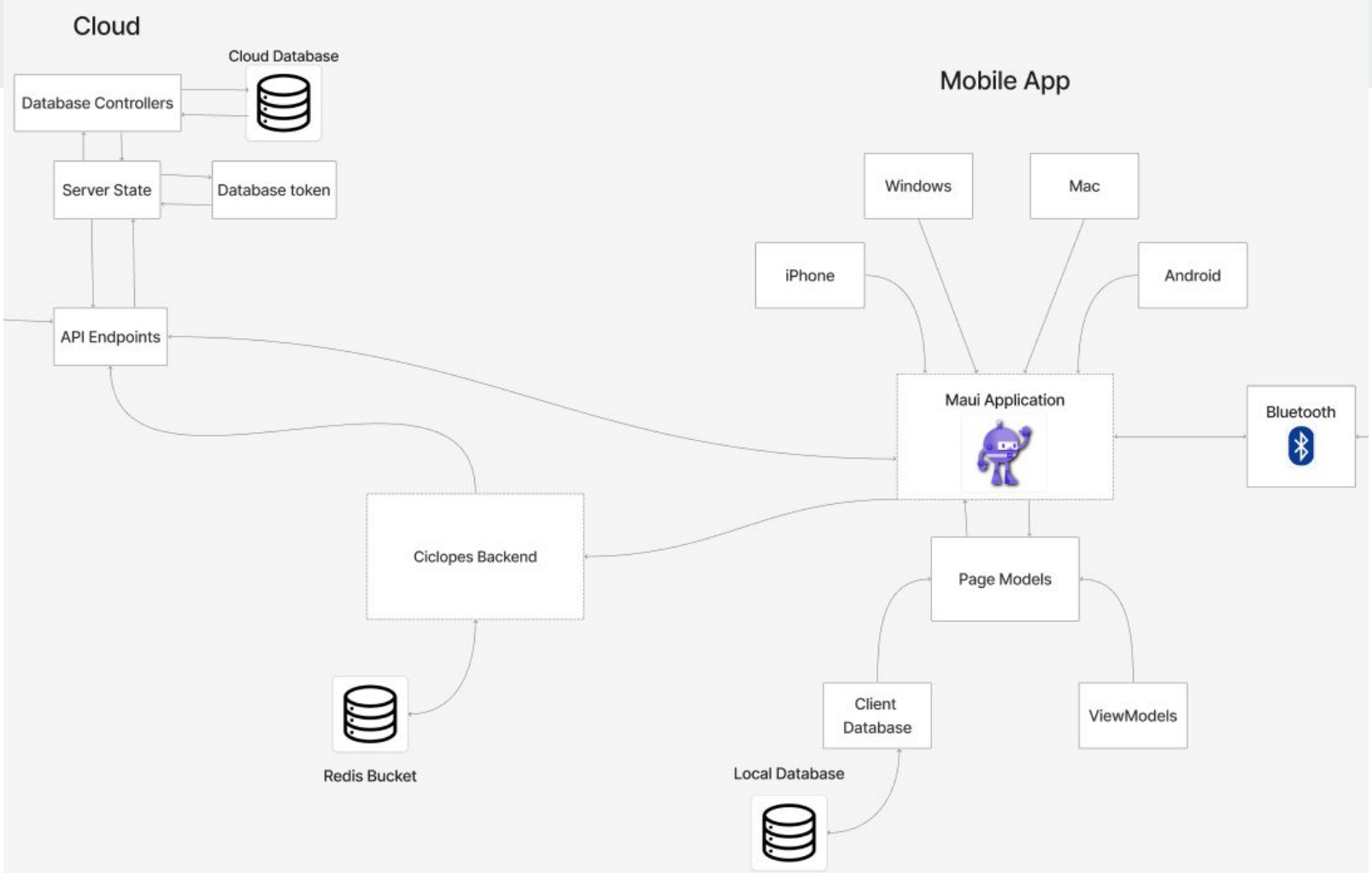
# Milestone 3

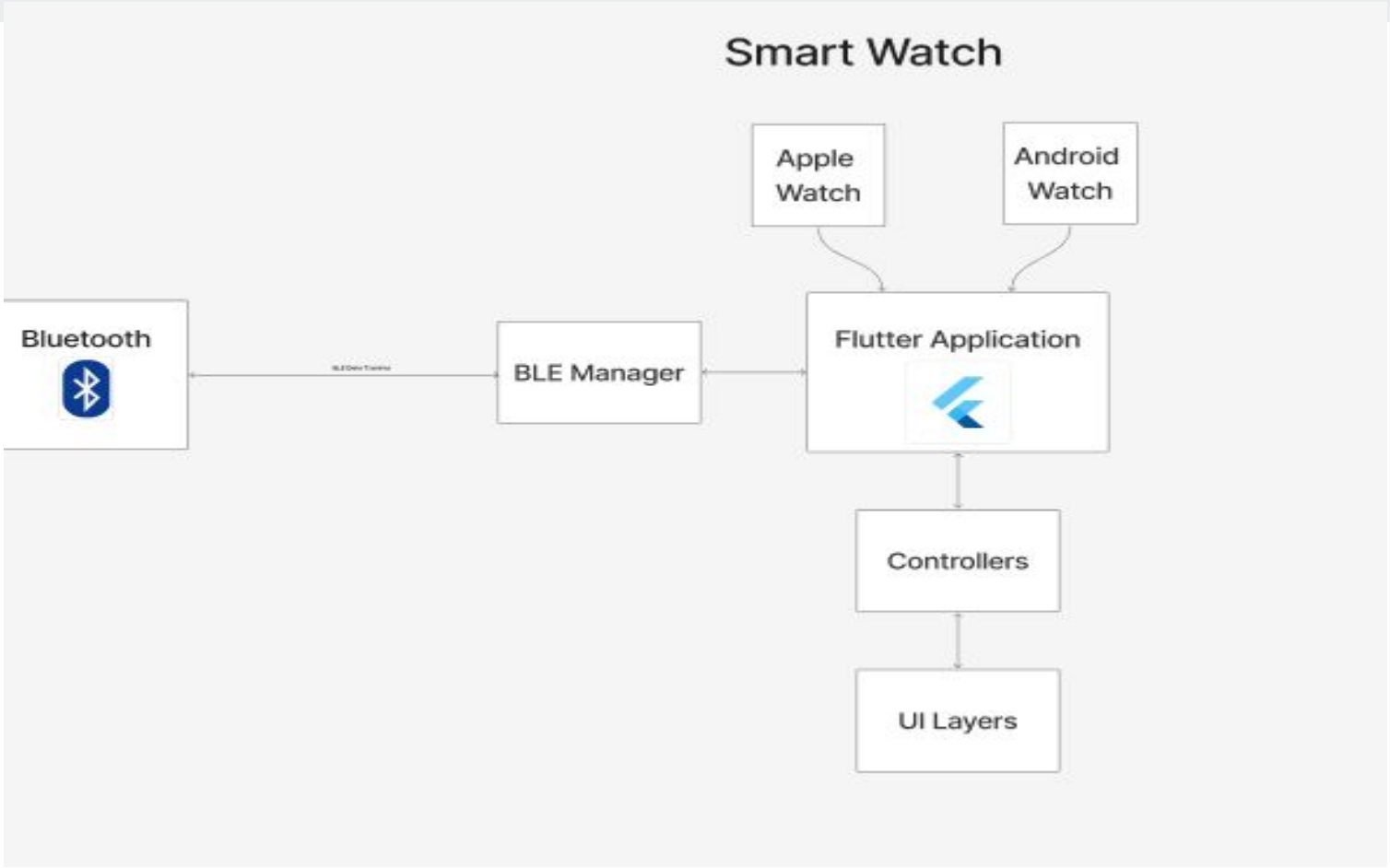
Matt Brown, Josh Byers, Charles Carroll, Zach Cox, Joseph Downey, Gabe Manero, Jakeb Nielsen, Andrew Olvera, Gavin Wentz, Hunter Wolfe

# HIGH LEVEL OVERVIEW




Physical System
Motors
1st Degree of Freedom
2nd Degree of Freedom
3rd Degree of Freedom
Full Motorized System
Design and Iterate
Brushless DC Motor
Stepper Motor 1
Stepper Motor 2
Raspberry Pi 5
Python Software
Qt
Analysis Mode
Diagnostic Mode
Shot Mode
HMI Touchscreen
Smartdot
Driver Script
Hand Calcs
Redesign


Cloud
Cloud Database
Database Controllers
Server State
Database token
API Endpoints
Ciclopes Backend
Redis Bucket
Mobile App
Windows
Mac
iPhone
Android
Maui Application
Bluetooth
Page Models
Client Database
ViewModels
Local Database


Smart Watch
Apple Watch
Android Watch
Bluetooth
BLE Manager
Flutter Application
Controllers
UI Layers

# Team Pi (Goals for MS3)

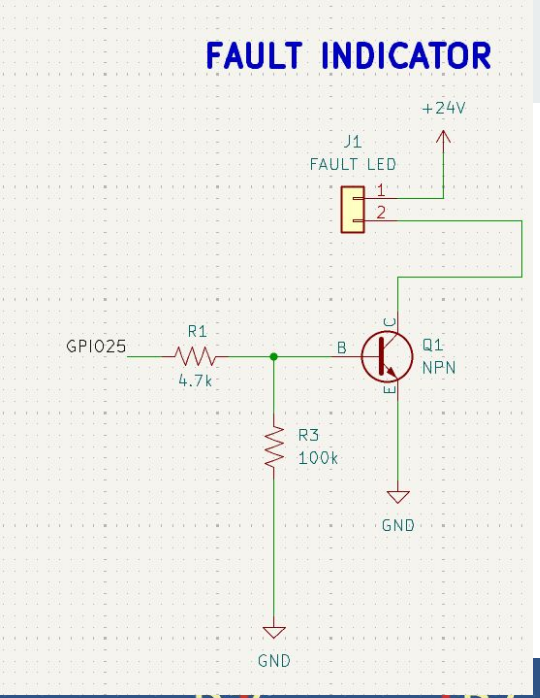
- Add encoders to the secondary and tertiary motors
- Get a releasable standalone version of the BSC application
- Integrate PCB with system
- Rewire the BSC
- Place BSC in enclosure
- Integrate new panel mount LEDs
- Organization of system circuitry
- Investigate power up sequencing after PCB implementation

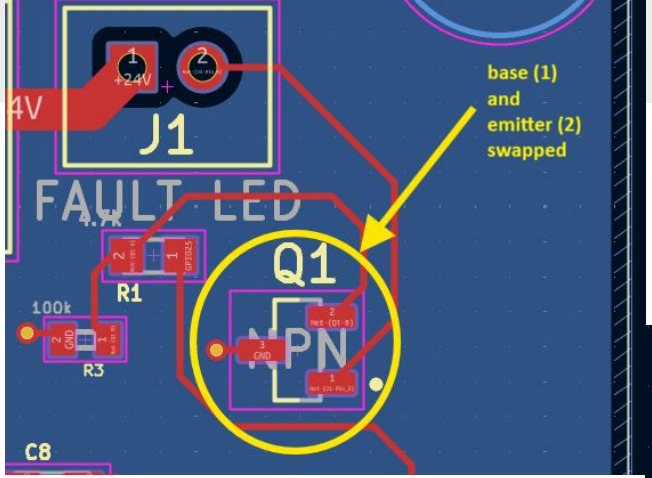
# Team Pi (MS3 Achievements)

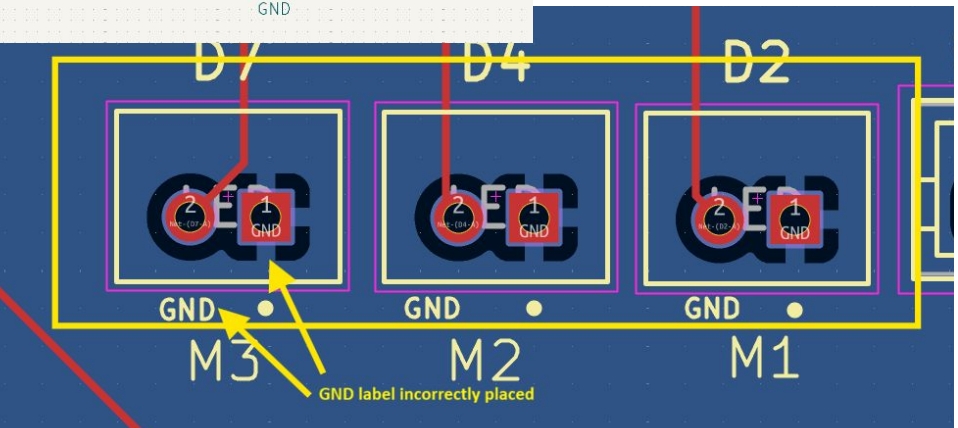
- Alpha build of the BSC application is on github (MacOs and Windows versions)
- BSC can now be run without the VESC connected
- Automated build script created for pi development.
- Diagnostic page overhaul: Motors Enabled is separate from Diagnostic session.
- PCB integrated into system

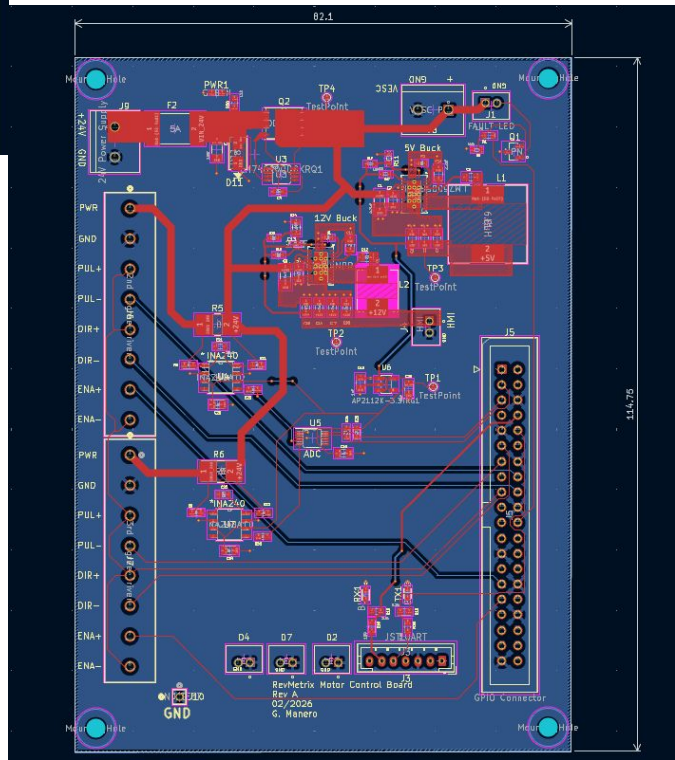
# Team Pi (Future)

- Add pull down resistors to ENA pins
- Investigate power up sequencing
- Integrate LED ESTOP
- Adding Zeroing functionality to the BSC
- AC switch, ground enclosure
- Current sensor
- UART
- Finish Wiring of system


FAULT INDICATOR
+24V
J1
FAULT LED
1
2
GPIO25
R1
4.7k
B
Q1
NPN
R3
100k
GND
GND


4V
+24V
J1
FAULT LED
4.7k
R1
100k
R3
Q1
NPN
C8
base (1)
and
emitter (2)
swapped

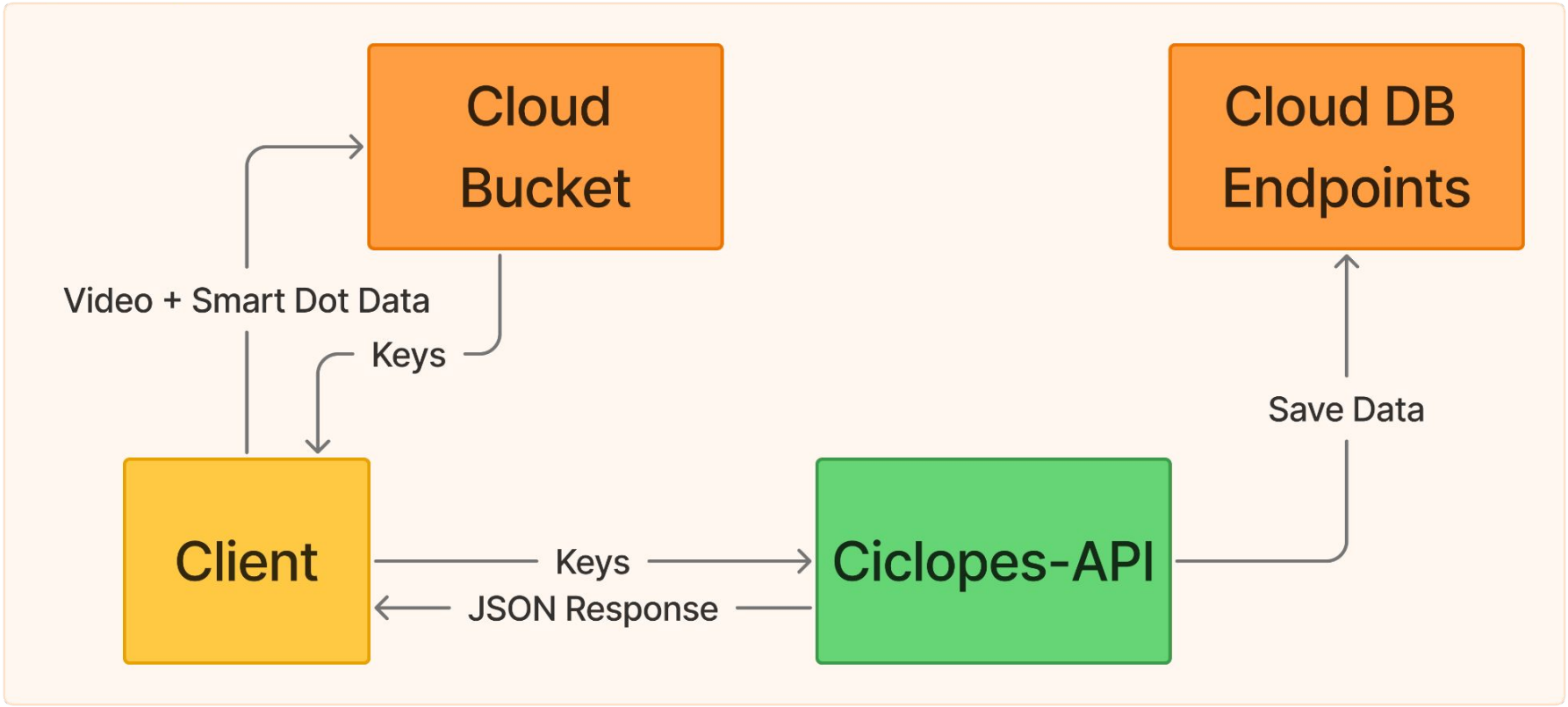

D7
D4
D2
GND
GND
GND
GND
GND
GND
M3
M2
M1
GND label incorrectly placed


82.1
114.75
PWR
GND
PUL+
PUL-
DIR+
DIR-
ENA+
ENA-
12V Buck
5V Buck
ADC
JST UART
GPIO Connector
RevMetrix Motor Control Board
Rev A
02/2026
G. Manero

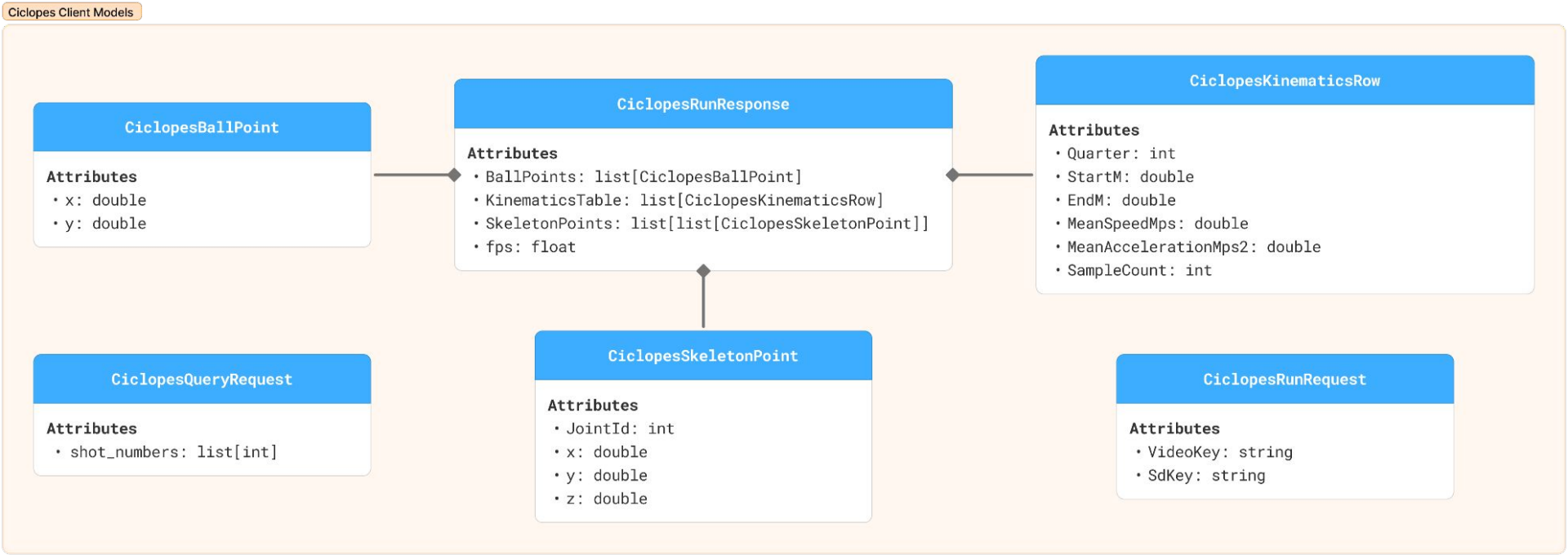
# Team Pi Demo

# Ciclopes - Overview

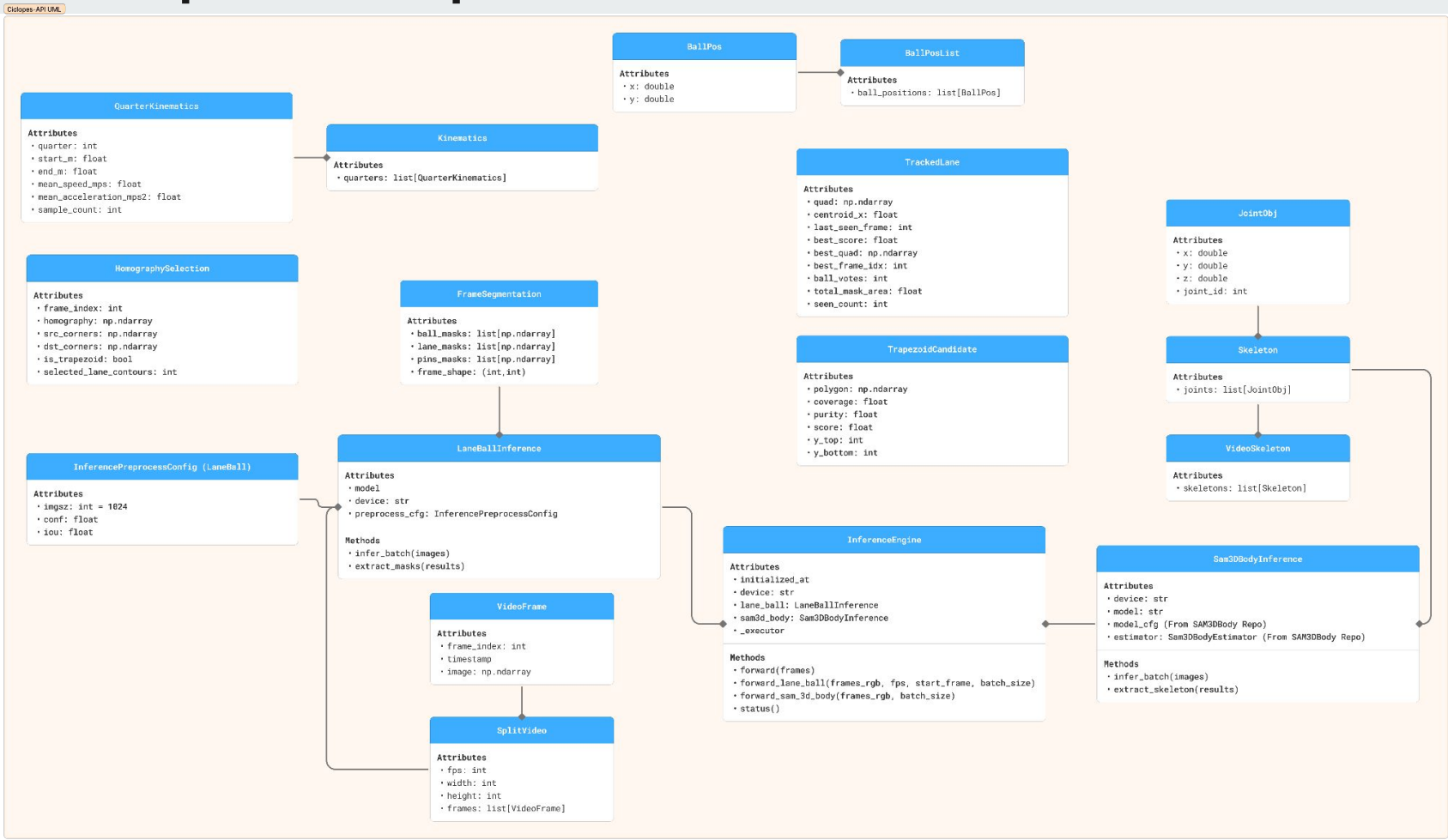
Ciclopes Architecture



# Ciclopes - Client Side Models



# Ciclopes - Ciclopes-API UML / Architecture



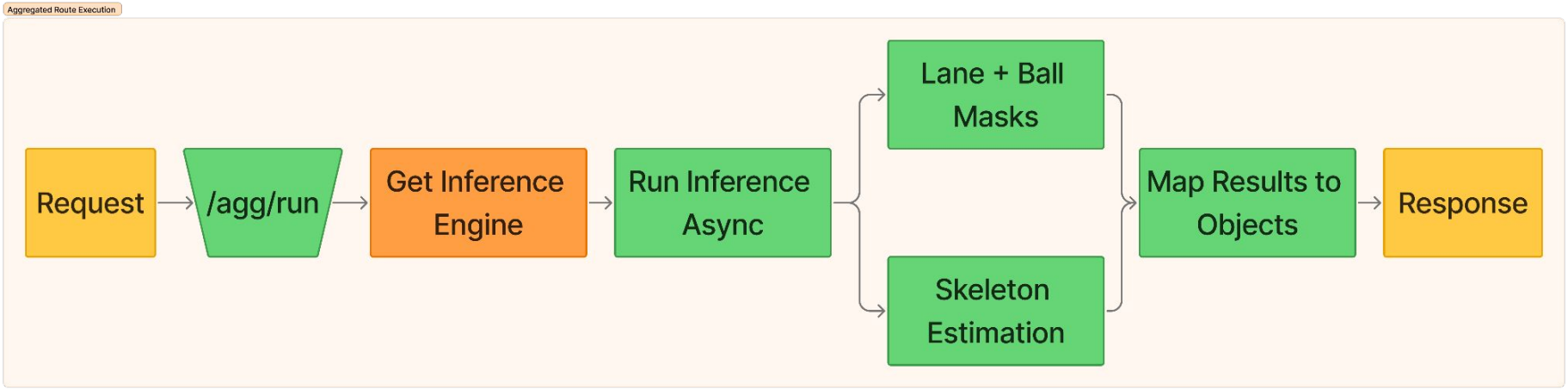
# Ciclopes - Goals for MS3

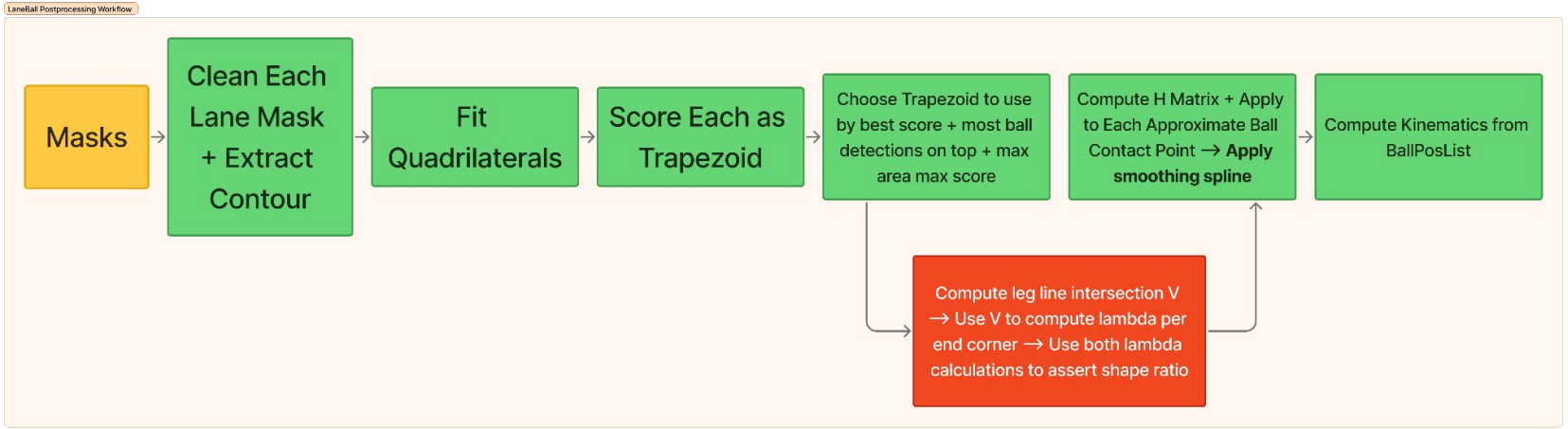
- Implement DB integration
- Add query Ciclopes from shot statistics views
  - Overlay multiple shots and data analysis view (connect with user collected scores / outcomes) - Stretch: AI Insights (LLM Powered but not AI slop)
- Implement SmartDot data integration
  - Record real world use demo (Setup service from Capstone room machine)
- Integrate UI styling + component library (ShadCn.Blazor)
  - UI components + styling build system
- Finalize UI/UX design with team and overhaul UX
- Complete integration into application

# Ciclopes - Accomplishments for MS3

- Implement  Mock DB integration
- Added query Ciclopes past shot data functionality
  - Overlay multiple shots and data analysis view
- Implemented SmartDot data integration
  - Record real world use demo (Setup service from Capstone room machine)
- Integrated UI styling + component library
  - UI components + styling build system
- Finalized UI/UX design with team and overhauled UX
- Complete integration into application

# Ciclopes - Ciclopes-API Execution



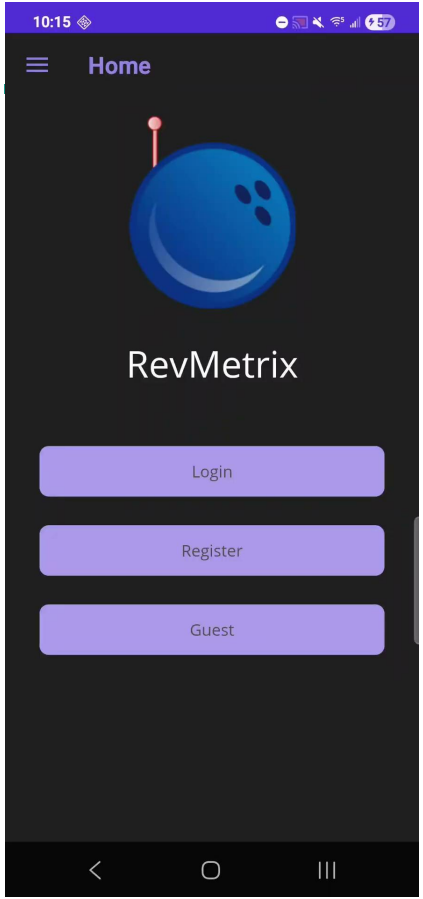
# Ciclopes - Demo of New Video Inference

# Ciclopes - Demo of Past Video Query + Overlay

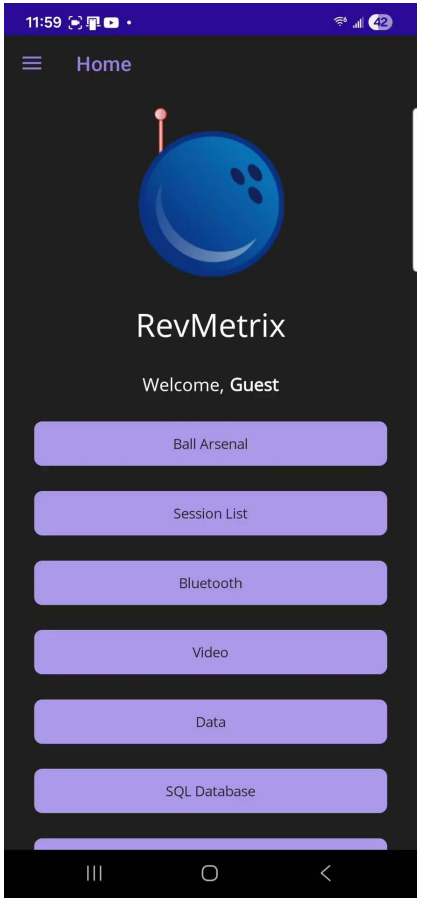
# Ciclopes - Deliverables for Capstone Expo

- Full integrated demo (completed)
- Cloud DB integration for full UX flow when using Ciclopes
- UI tweaks for cleanliness and user enjoyment
- Setup interactive pose capture and rendering feature for fun capstone expo demo
- FULL documentation of training / evaluation pipeline for future team
- FULL documentation of Ciclopes-API service for future team
- Complete update of wiki information

# Mobile App - Demo!




Video page



# Mobile App (MS3 Goals)

- Improve Stats Page/shot input
- Edit/ update events and establishments
- Change app color scheme
- Save shots from watch
- Update MMS page

# Mobile App (MS3 Achievements)

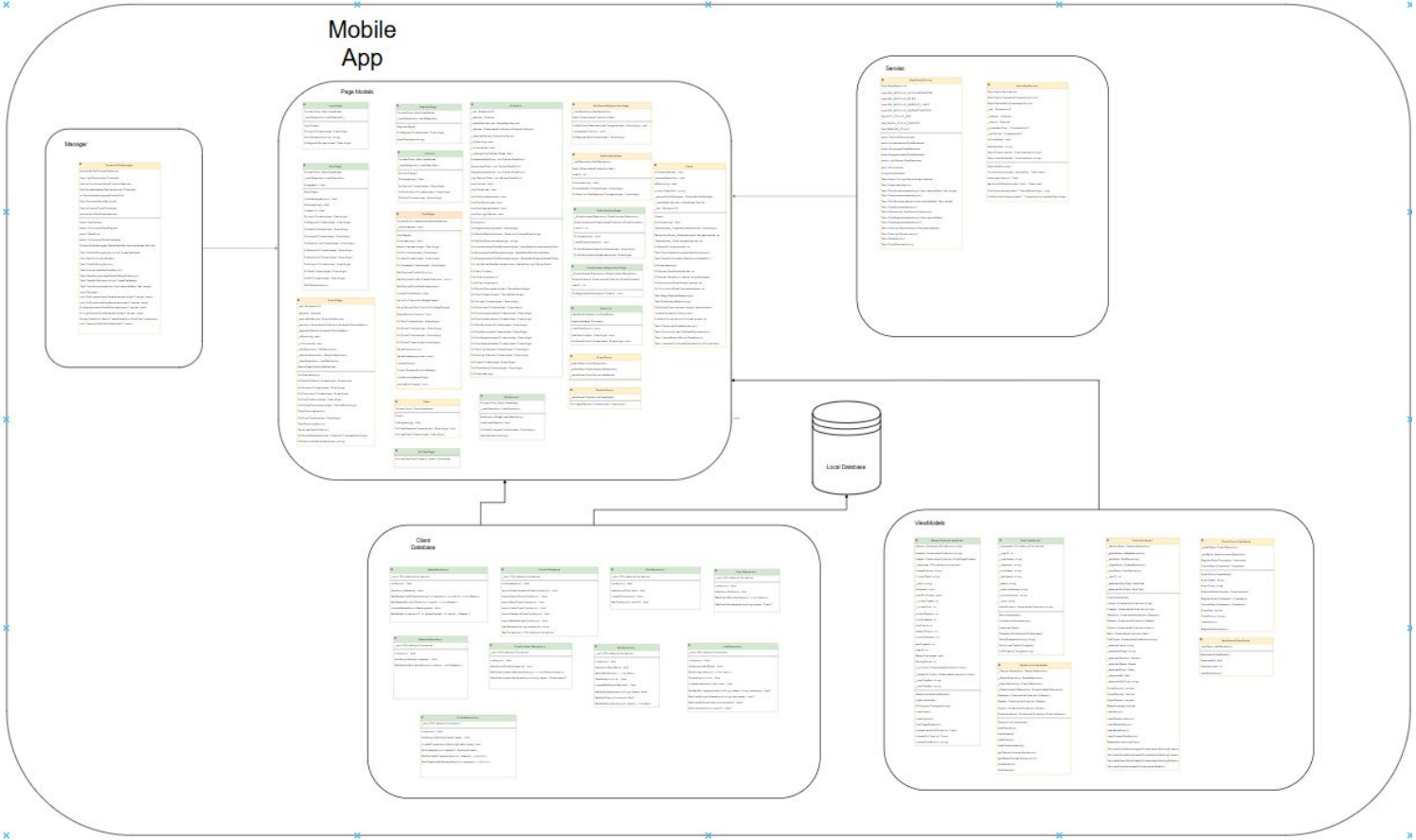
- Establishment Page Upgrades
- Improved shot page input (Speed, Stance, Position, Lane)
- Demo mode for video page
- UI/UX Upgrades
- Data Import
- Improved Stats

- Reworked SmartDot Page
  - Changeable settings for Accelerometer, Ambient Light, Barometer, Gyroscope, Magnetometer
- Fixed Bugs on VideoPage
  - Maui Video Bug
  - Bluetooth Bug
- Fixed what the app sees as a “derivative spike”

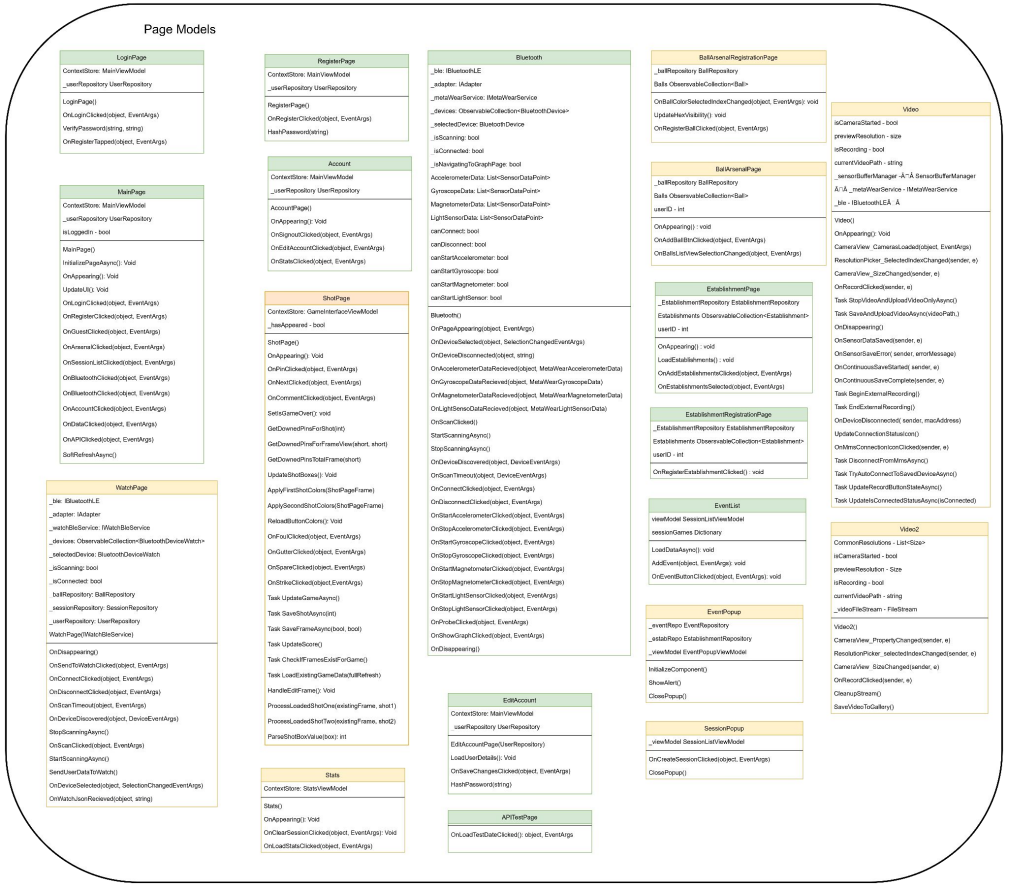
# Mobile App (Future: MS4)

- Add more stats!!!
- Edit Events
- More UI/UX updates (Popups)
- Fix little things for Prof. Hake
- Continue to fix Video Page
- Continue to fix Cloud Sync
- GRADUATE

# Mobile UML Overview



# Mobile UML - Page Models

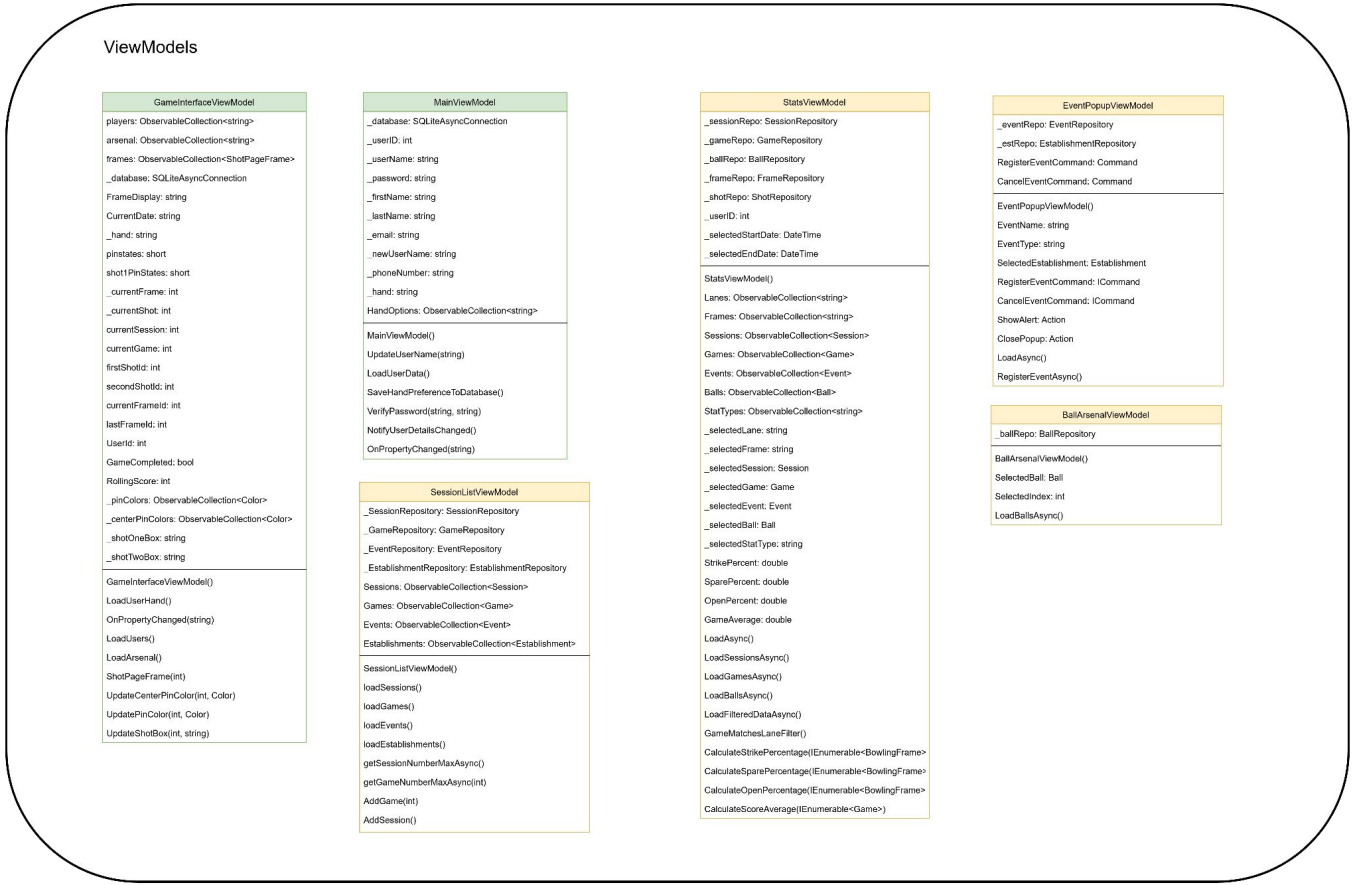



Implemented

Partially Implemented

Not Yet Implemented

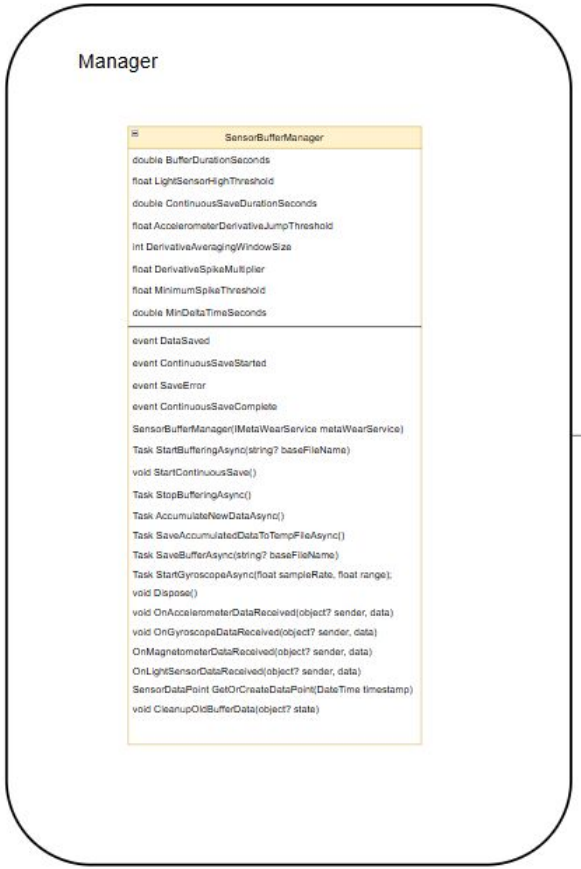
# Mobile UML - View Models



# Mobile UML - Database

Client
Database

# Mobile UML - Managers

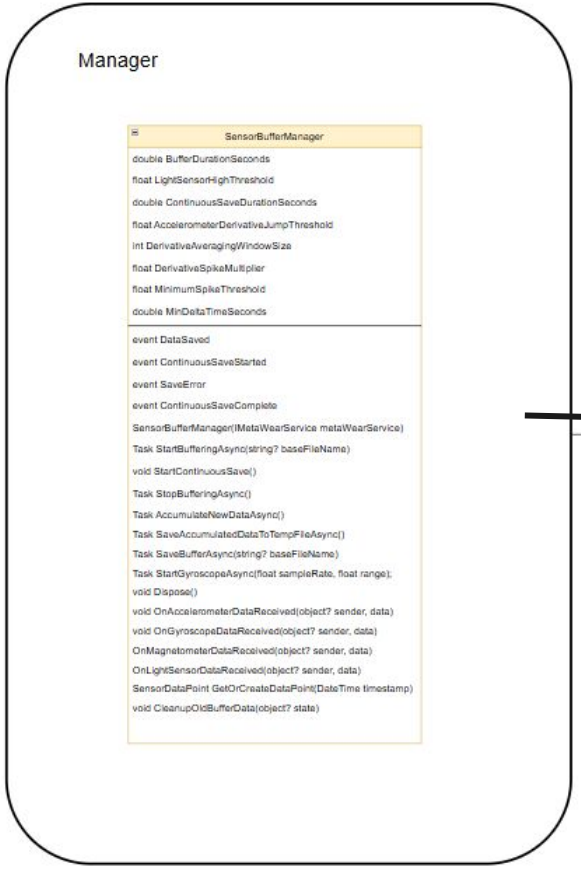


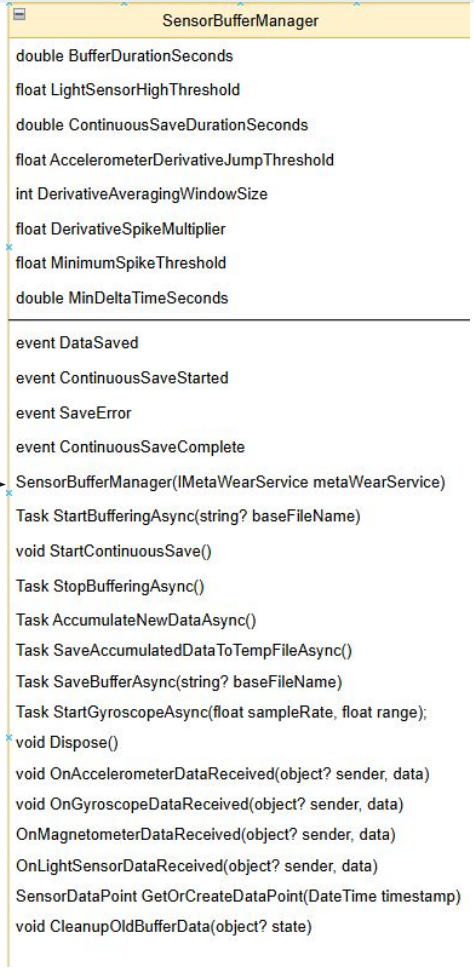

Implemented

Partially Implemented

Not Yet Implemented

# Mobile UML - SensorBufferManager



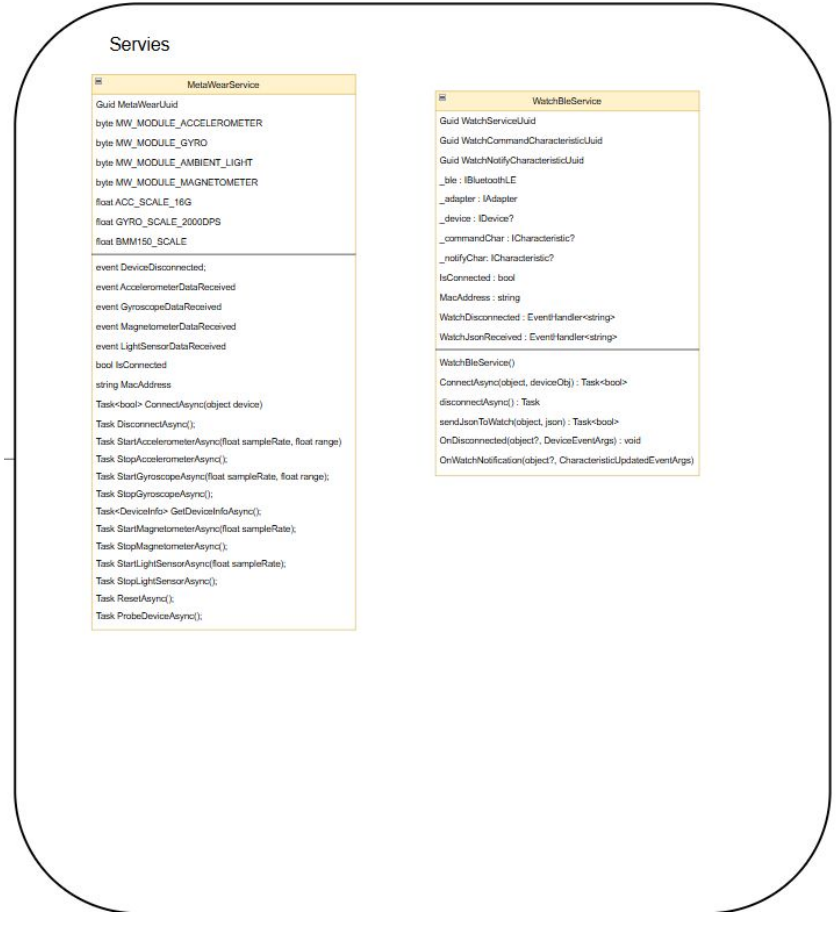




Implemented

Partially Implemented

Not Yet Implemented

# Mobile UML - Services




Implemented

Partially Implemented

Not Yet Implemented

# Mobile UML - MetaWearService

Servies

MetaWearService

Guid MetaWearUuid
byte MW_MODULE_ACCELEROMETER
byte MW_MODULE_GYRO
byte MW_MODULE_AMBIENT_LIGHT
byte MW_MODULE_MAGNETOMETER
float ACC_SCALE_16G
float GYRO_SCALE_2000DPS
float BMM150_SCALE

event DeviceDisconnected;
event AccelerometerDataReceived
event GyroscopeDataReceived
event MagnetometerDataReceived
event LightSensorDataReceived
bool IsConnected
string MacAddress
Task<bool> ConnectAsync(object device)
Task DisconnectAsync();
Task StartAccelerometerAsync(float sampleRate, float range)
Task StopAccelerometerAsync();
Task StartGyroscopeAsync(float sampleRate, float range);
Task StopGyroscopeAsync();
Task<DeviceInfo> GetDeviceInfoAsync();
Task StartMagnetometerAsync(float sampleRate);
Task StopMagnetometerAsync();
Task StartLightSensorAsync(float sampleRate);
Task StopLightSensorAsync();
Task ResetAsync();
Task ProbeDeviceAsync();

WatchBleService

Guid WatchServiceUuid
Guid WatchCommandCharacteristicUuid
Guid WatchNotifyCharacteristicUuid
_ble : IBluetoothLE
_adapter : IAdapter
_device : IDevice?
_commandChar : ICharacteristic?
_notifyChar: ICharacteristic?
IsConnected : bool
MacAddress : string
WatchDisconnected : EventHandler<string>
WatchJsonReceived : EventHandler<string>

WatchBleService()
ConnectAsync(object, deviceObj) : Task<bool>
disconnectAsync() : Task
sendJsonToWatch(object, json) : Task<bool>
OnDisconnected(object?, DeviceEventArgs) : void
OnWatchNotification(object?, CharacteristicUpdatedEventArgs)

MetaWearService

Guid MetaWearUuid
byte MW_MODULE_ACCELEROMETER
byte MW_MODULE_GYRO
byte MW_MODULE_AMBIENT_LIGHT
byte MW_MODULE_MAGNETOMETER
float ACC_SCALE_16G
float GYRO_SCALE_2000DPS
float BMM150_SCALE

event DeviceDisconnected;
event AccelerometerDataReceived
event GyroscopeDataReceived
event MagnetometerDataReceived
event LightSensorDataReceived
bool IsConnected
string MacAddress
Task<bool> ConnectAsync(object device)
Task DisconnectAsync();
Task StartAccelerometerAsync(float sampleRate, float range)
Task StopAccelerometerAsync();
Task StartGyroscopeAsync(float sampleRate, float range);
Task StopGyroscopeAsync();
Task<DeviceInfo> GetDeviceInfoAsync();
Task StartMagnetometerAsync(float sampleRate);
Task StopMagnetometerAsync();
Task StartLightSensorAsync(float sampleRate);
Task StopLightSensorAsync();
Task ResetAsync();
Task ProbeDeviceAsync();

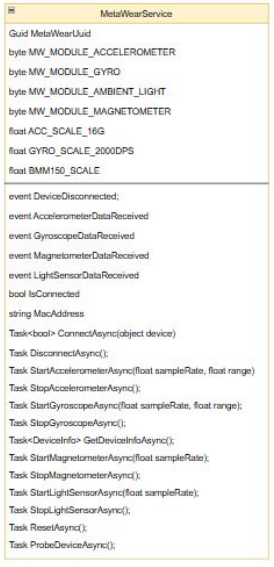
Implemented

Partially Implemented

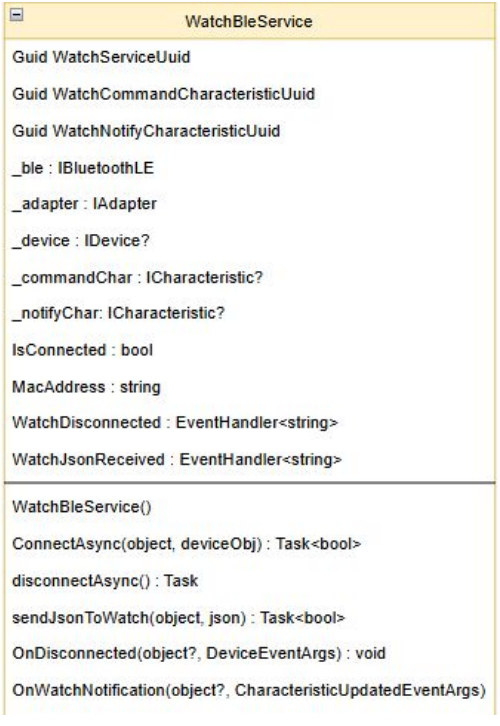
Not Yet Implemented

# Mobile UML - WatchBleService

Servies





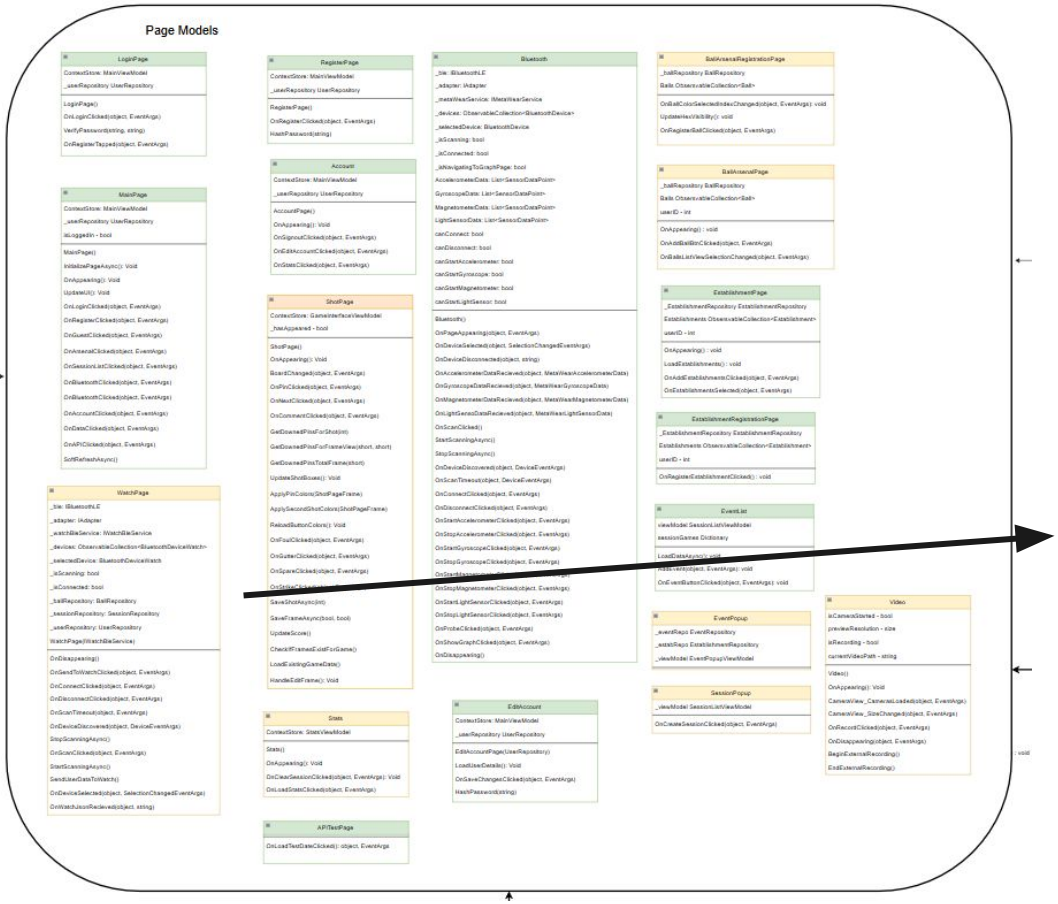



Implemented

Partially Implemented

Not Yet Implemented

# Mobile UML - WatchPage




## WatchPage

_ble: IBluetoothLE

_adapter: IAdapter

_watchBleService: IWatchBleService

_devices: ObservableCollection<BluetoothDeviceWatch>

_selectedDevice: BluetoothDeviceWatch

_isScanning: bool

_isConnected: bool

_ballRepository: BallRepository

_sessionRepository: SessionRepository

_userRepository: UserRepository

WatchPage(IWatchBleService)

---

OnDisappearing()

OnSendToWatchClicked(object, EventArgs)

OnConnectClicked(object, EventArgs)

OnDisconnectClicked(object, EventArgs)

OnScanTimeout(object, EventArgs)

OnDeviceDiscovered(object, DeviceEventArgs)

StopScanningAsync()

OnScanClicked(object, EventArgs)

StartScanningAsync()

SendUserDataToWatch()

OnDeviceSelected(object, SelectionChangedEventArgs)

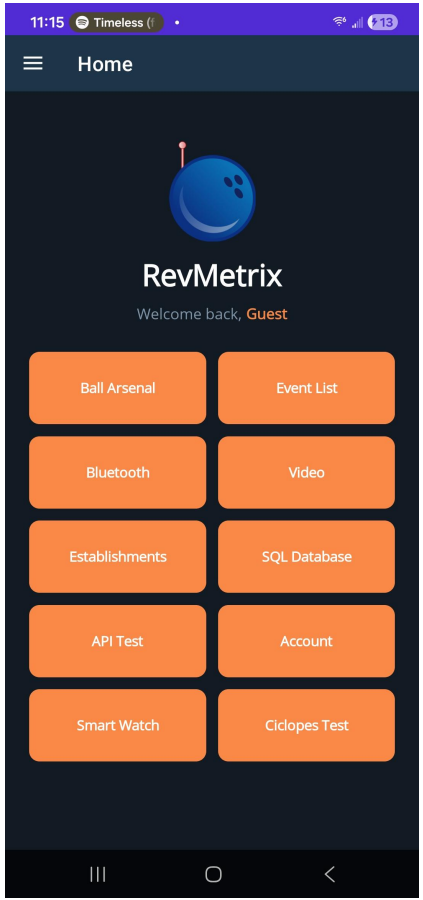
OnWatchJsonRecieved(object, string)

Implemented

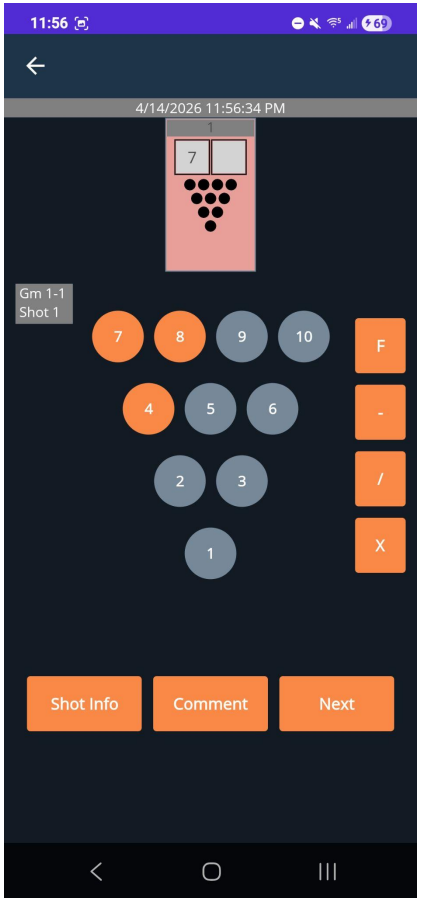
Partially Implemented

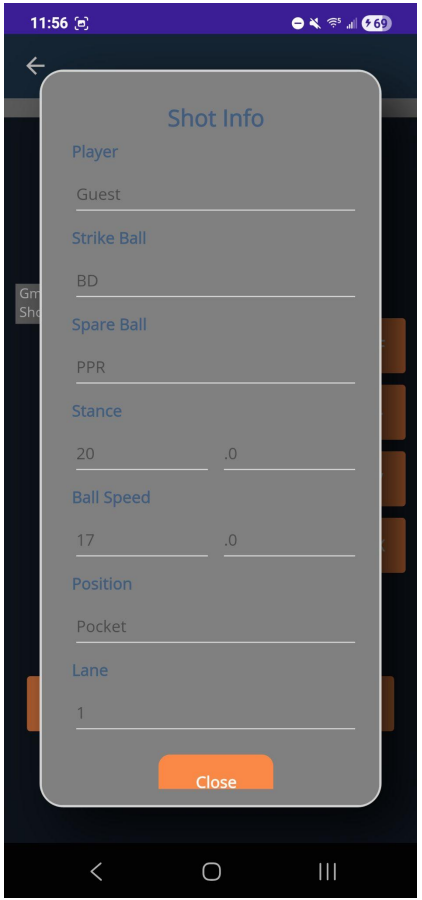
Not Yet Implemented

# Mobile App - Main Page (UI Update)

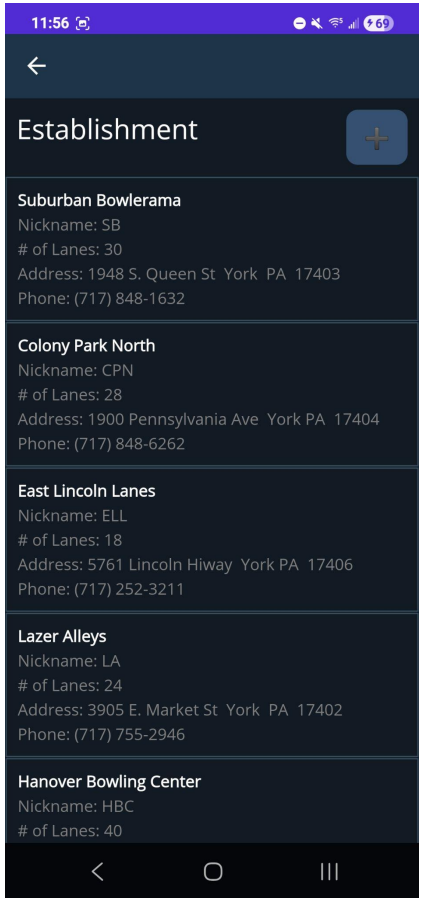


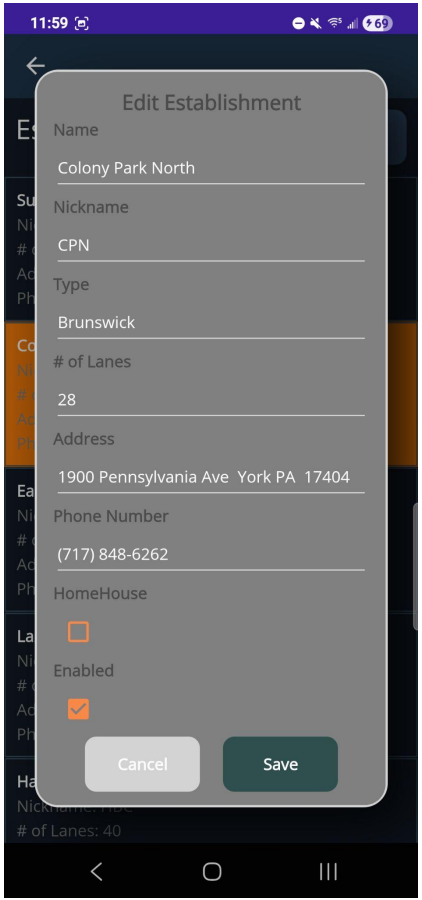
# Mobile App - Shot Page



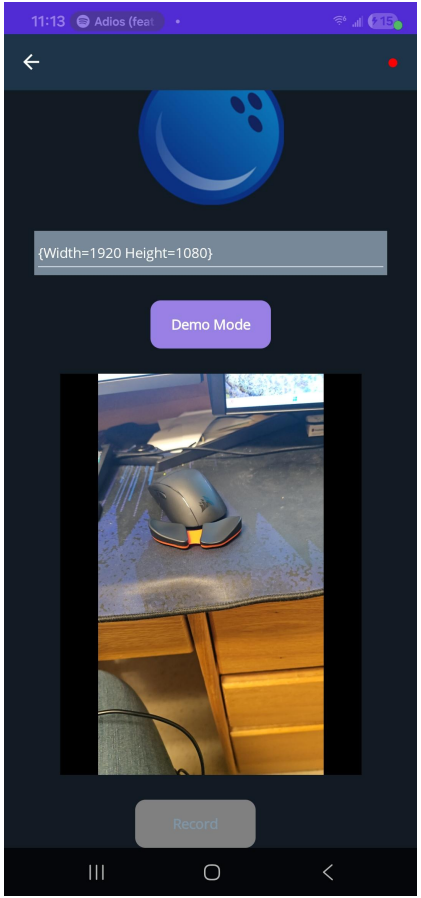



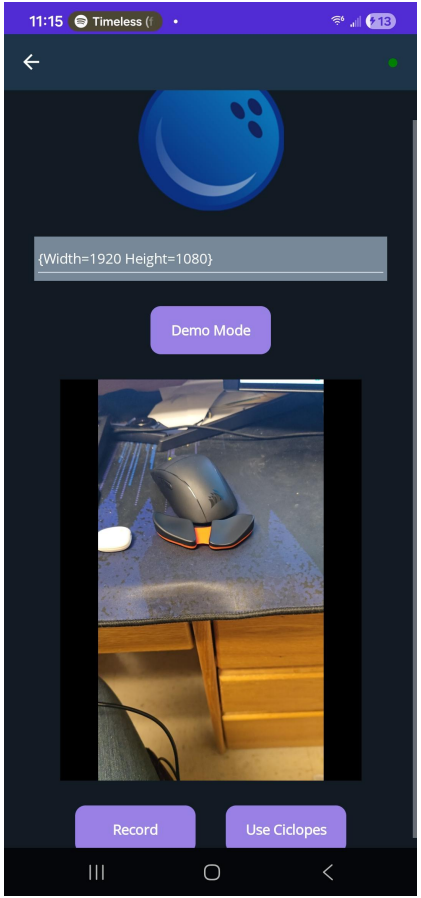
# Mobile App - Establishment Page

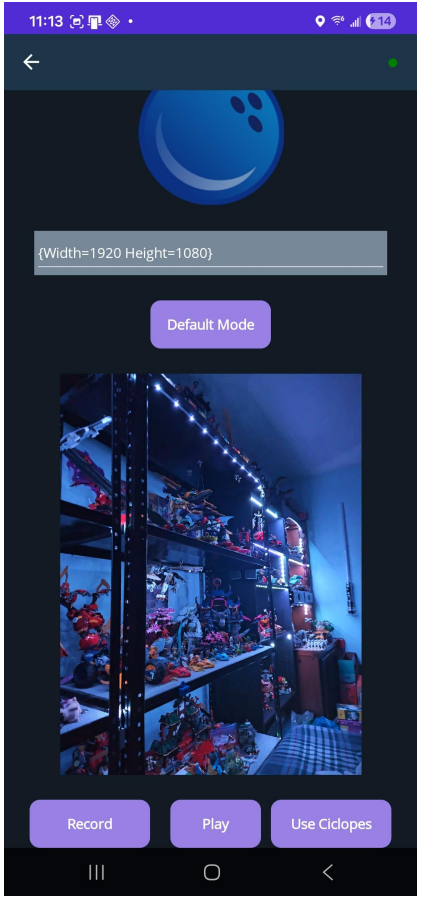


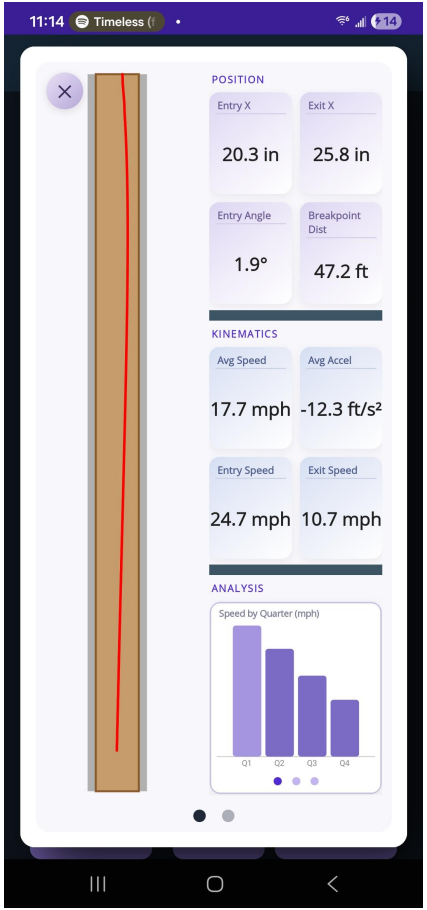



# Mobile App - Video Page

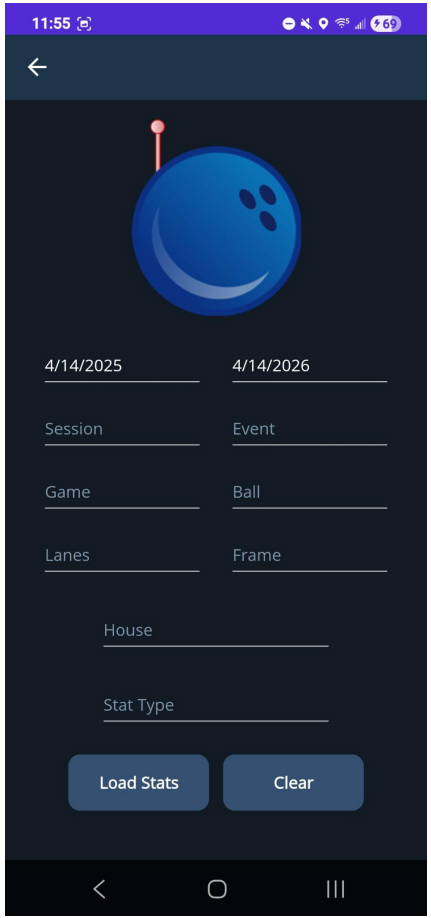


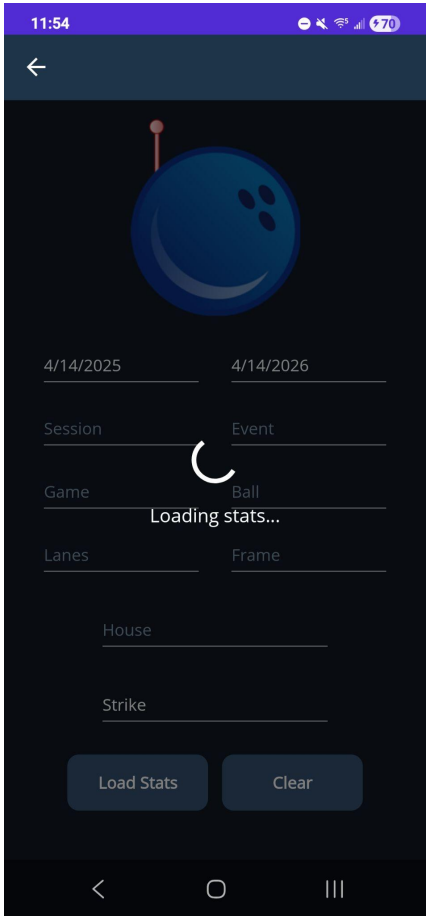



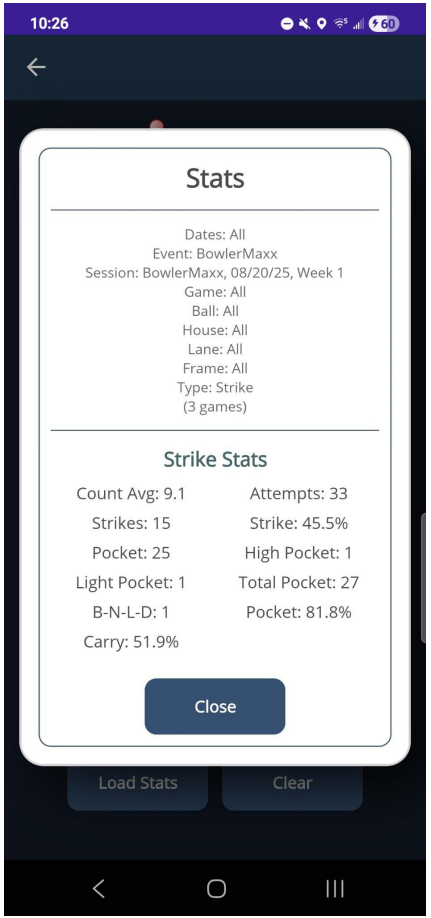



# Mobile App - Stats Update

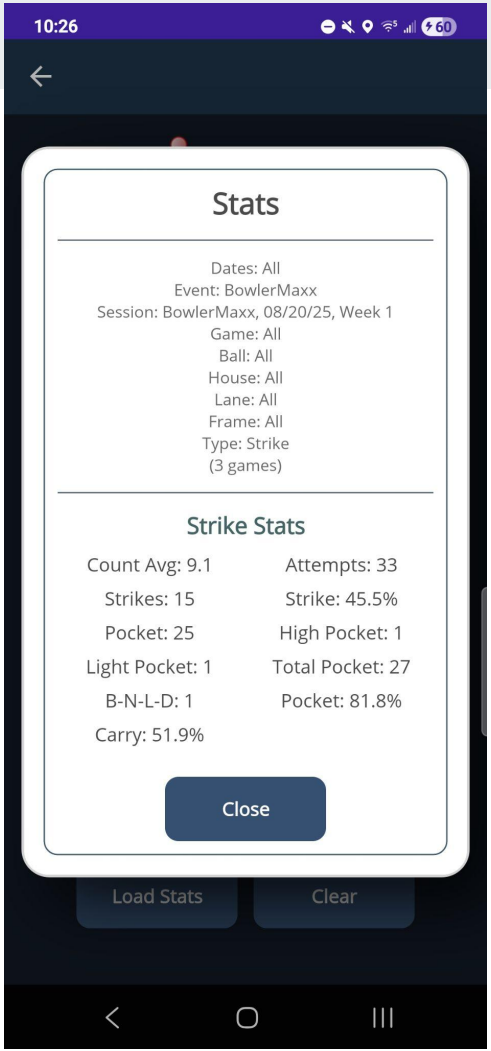


| | Event | Season | Week | Date | Game | Start lane | Ball Strike | Ball Spare |
|---|---|---|---|---|---|---|---|---|
| COUNT | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 1 | 20 | NUB | RPH |
| LEAVE | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 1 | 20 | | |
| SCORE | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 1 | 20 | | |
| TYPE | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 1 | 20 | | |
| BOARD | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 1 | 20 | | |
| LANE | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 1 | 20 | | |
| BALL | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 1 | 20 | | |
| COUNT | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 2 | 19 | NUB | RPH |
| LEAVE | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 2 | 19 | | |
| SCORE | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 2 | 19 | | |
| TYPE | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 2 | 19 | | |
| BOARD | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 2 | 19 | | |
| LANE | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 2 | 19 | | |
| BALL | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 2 | 19 | | |
| COUNT | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 3 | 20 | NUB | RPH |
| LEAVE | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 3 | 20 | | |
| SCORE | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 3 | 20 | | |
| TYPE | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 3 | 20 | | |
| BOARD | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 3 | 20 | | |
| LANE | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 3 | 20 | | |
| BALL | BowlerMaxx | Fa-25 | 1 | 8/20/25 | 3 | 20 | | |
| SERIES | BowlerMaxx | Fa-25 | 1 | 8/20/25 | | | | |

| Average Count | | Strike Statistics Attempts | Strikes | Strike % | Pocket | H-Pocket | L-Pocket | Tot-Pocket | B-N-L-D | Pocket % | Carry % |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 9.455 | | 11 | 5 | 45.5% | 11 | 0 | 0 | 11 | 0 | 100.0% | 45.5% |
| 9.455 | | 11 | 8 | 72.7% | 9 | 0 | 1 | 10 | 0 | 90.9% | 80.0% |
| 8.364 | | 11 | 2 | 18.2% | 5 | 1 | 0 | 6 | 1 | 54.5% | 16.7% |
| 9.091 | | 33 | 15 | 45.5% | 25 | 1 | 1 | 27 | 1 | 81.8% | 51.9% |



# Wiki

**MS3 Goals**

- **Continue updating mobile progress**
- **Get help updating other parts of the project**

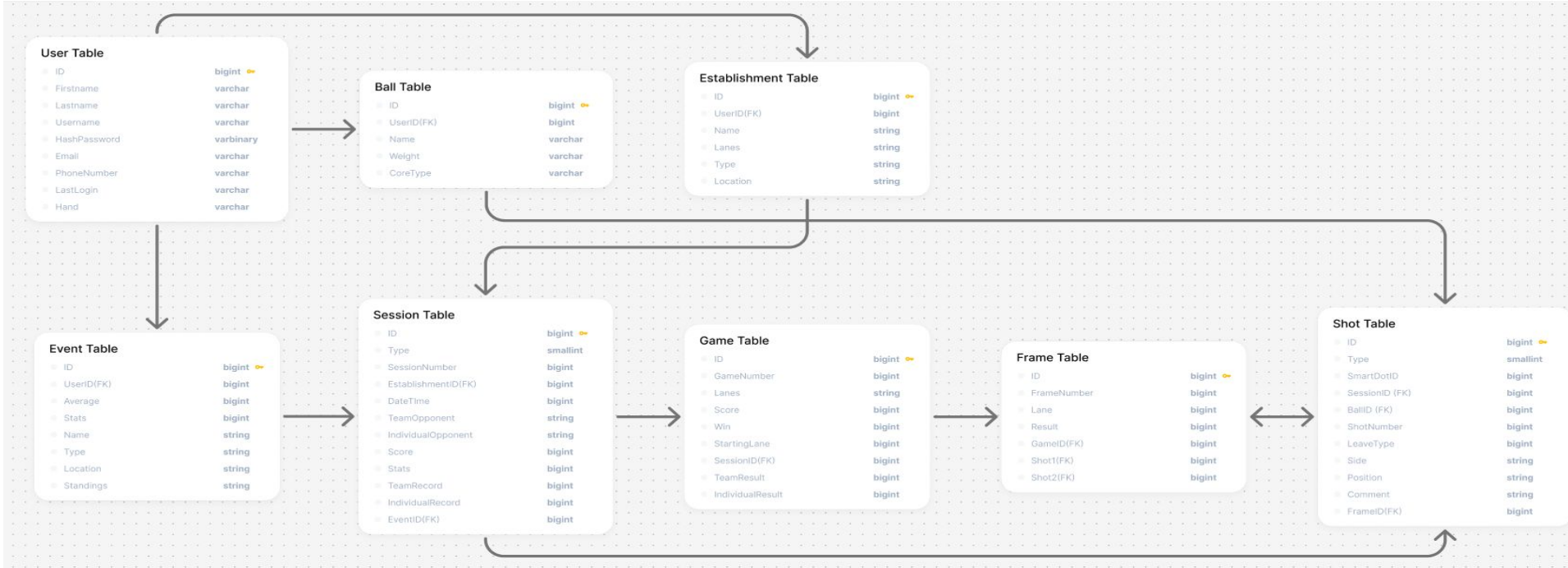
**MS3 Achievements**

- **N/A**

**Future**

- **Update all sections of the project for future team!**

# Database Schema



# Cloud (MS3 Goals and Accomplishments)

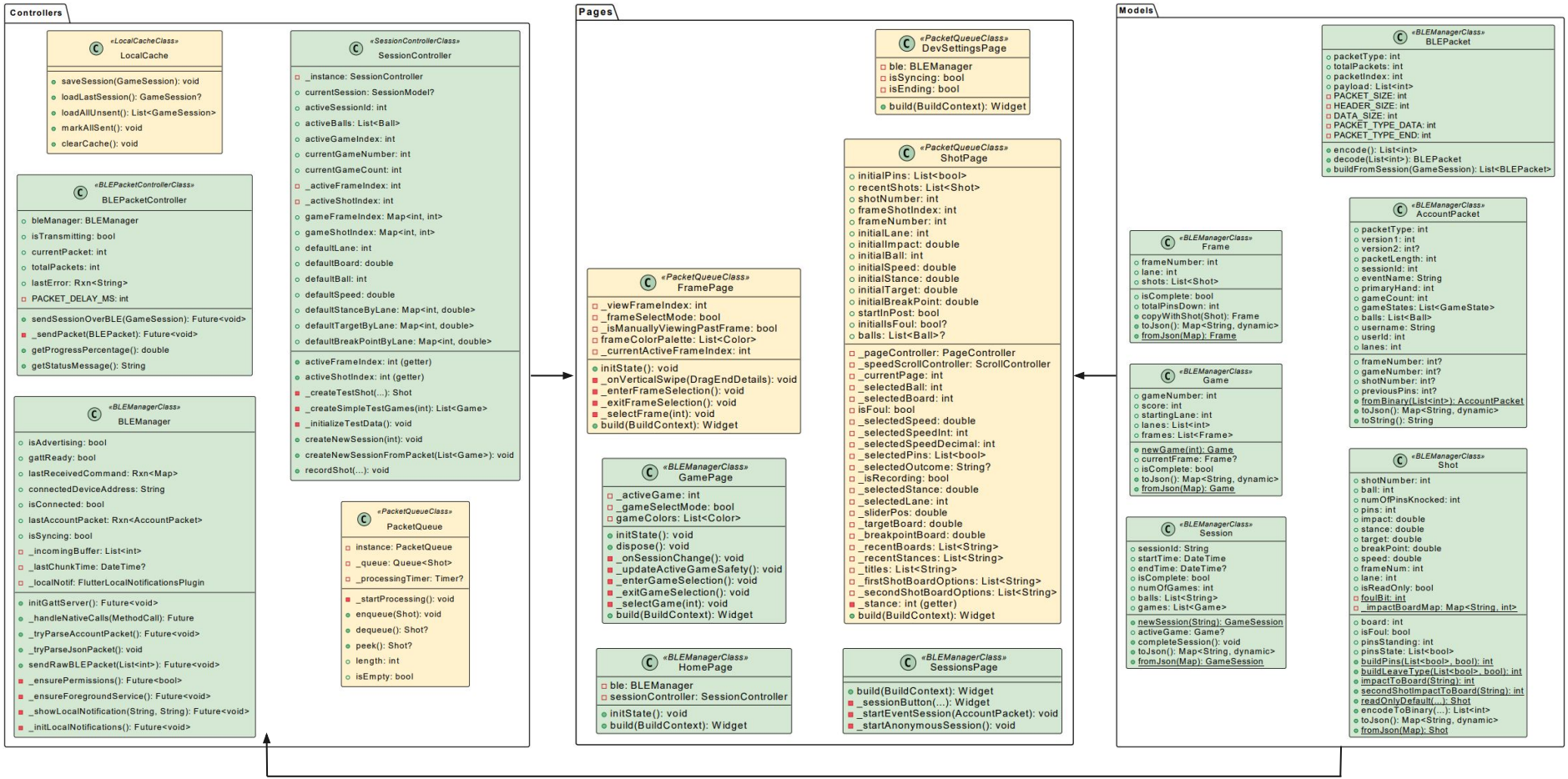
- Update all the Tables to be Correct with the Mobile App
- Update Deletes to just mark some data as “Deleted”
- Fix any other issue that arises

- Update all the Tables to be Correct with the Mobile App

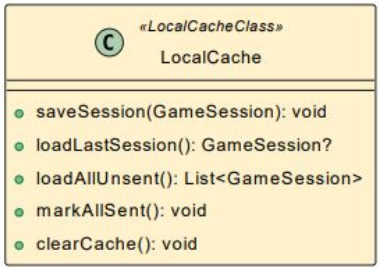
# Cloud (Future: MS4)

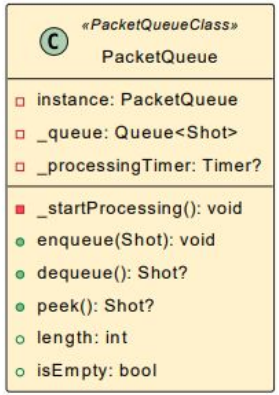
- Update Deletes to just mark some data as “Deleted”
- Fix some bugs with Sync Feature for phone(Prof Hake)
- Fix any other issue that arises

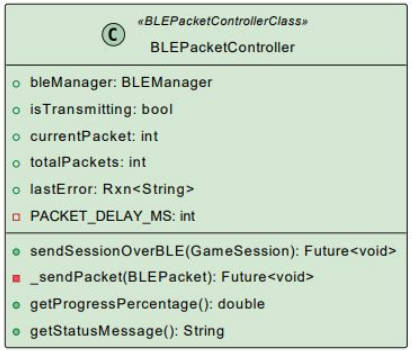
# Smart Watch UML

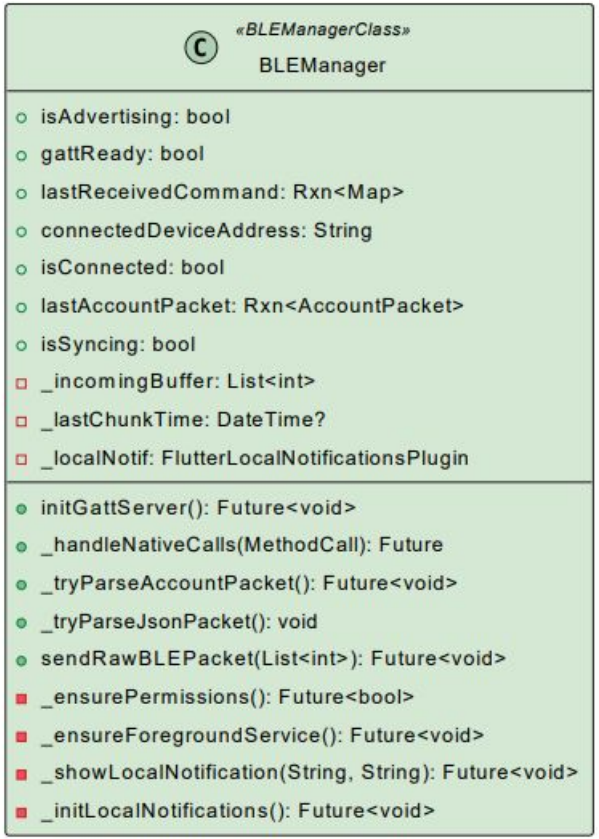


# Smart Watch UML (Controllers)

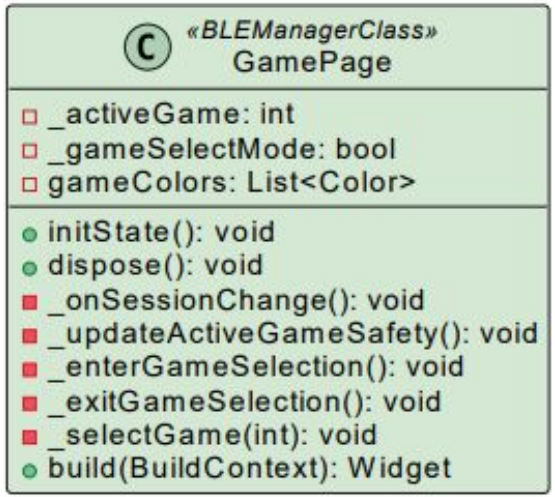


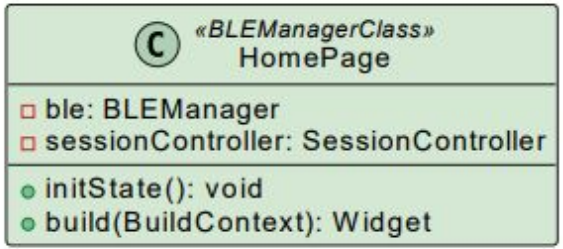



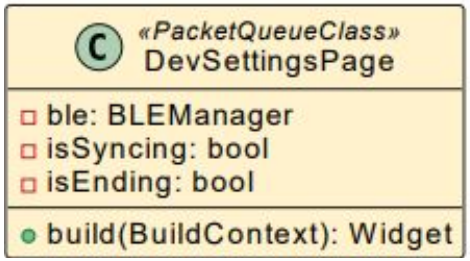



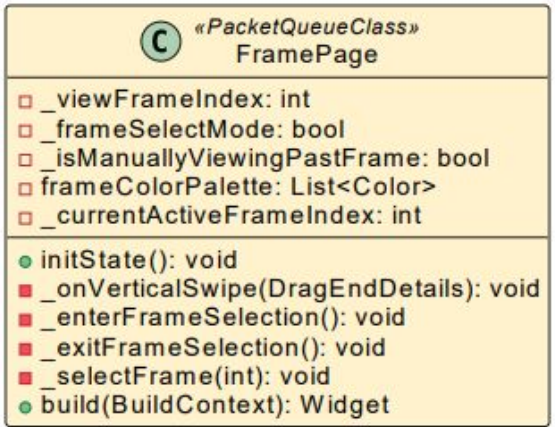



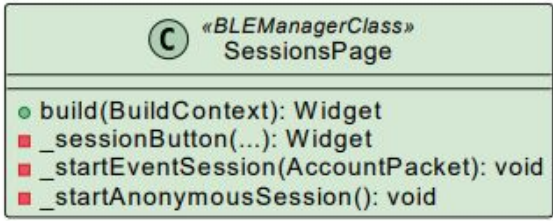

# Smart Watch UML (Pages)

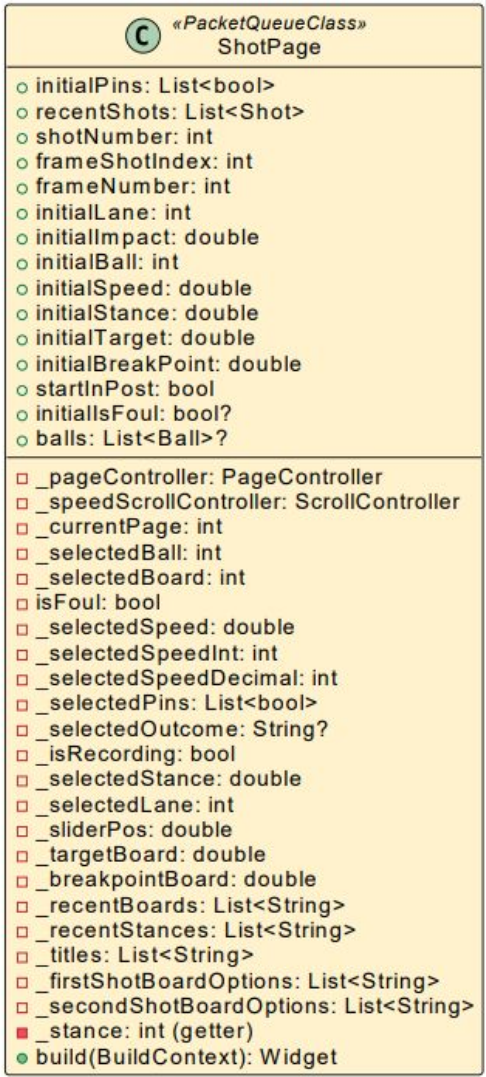


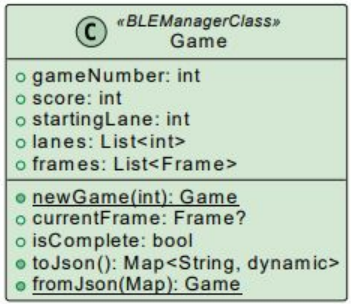



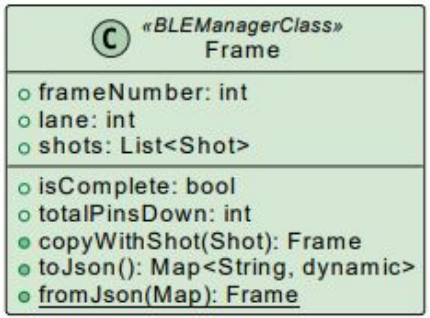



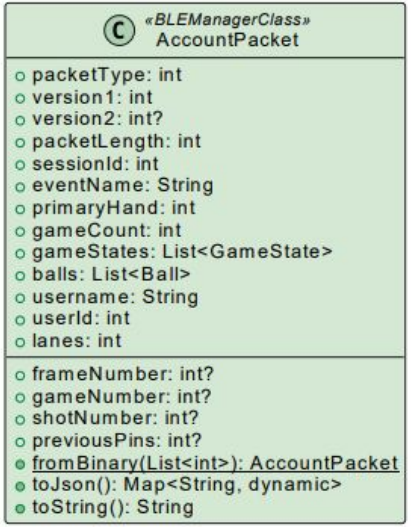



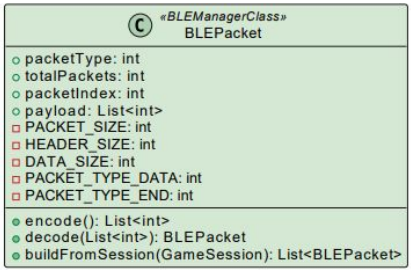
# Smart Watch UML (Models)








# Tools and Technologies

- **Flutter** - framework
- **Dart** - programming language
- **Kotlin** - Android BLE foreground services
- **Visual Studio Code** - IDE
- **Bluetooth Low Energy** (BLE)
- **Android/IOS Launcher** - Emulator

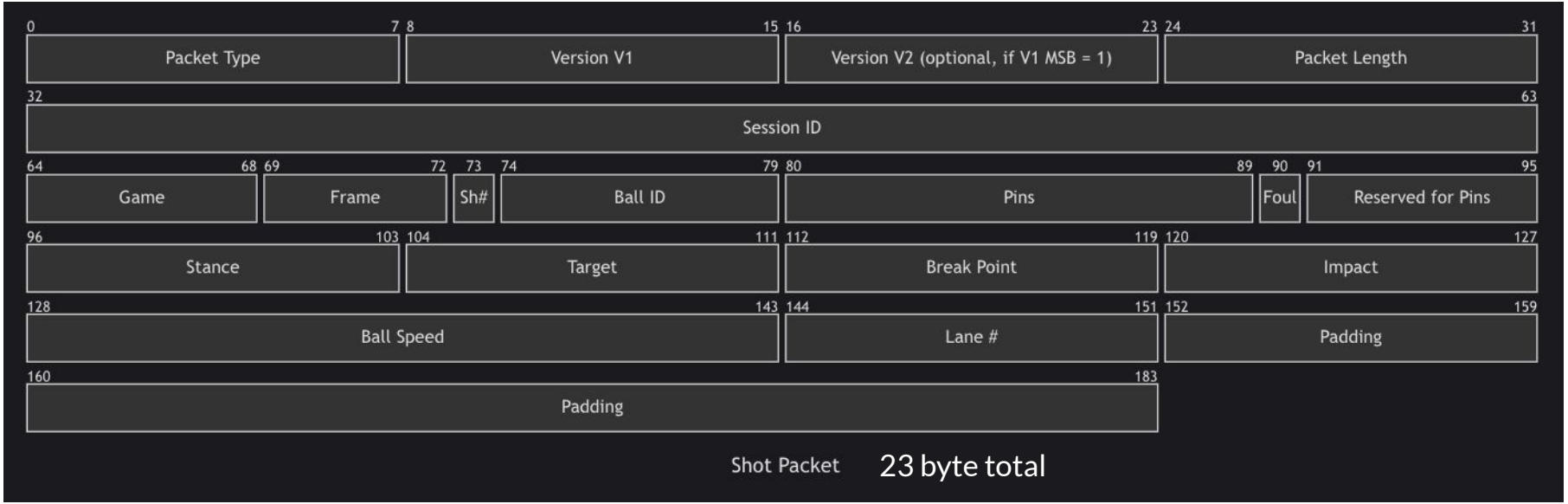

# Smart Watch (MS3 Goals)

- Unpack shot packets and place in cellular database
- Update UI on cellular to show the incoming shots
- Remove scoring from packets and watch
- Implement end of game logic on watch
- Add End session and Logout functionality
- Adjust initial data packet to send session based on date/time
- Anonymous session correctly being placed in database
- Implement lanes
- Update Info bar
- Fix recent results page
- Begin BLE packet queue

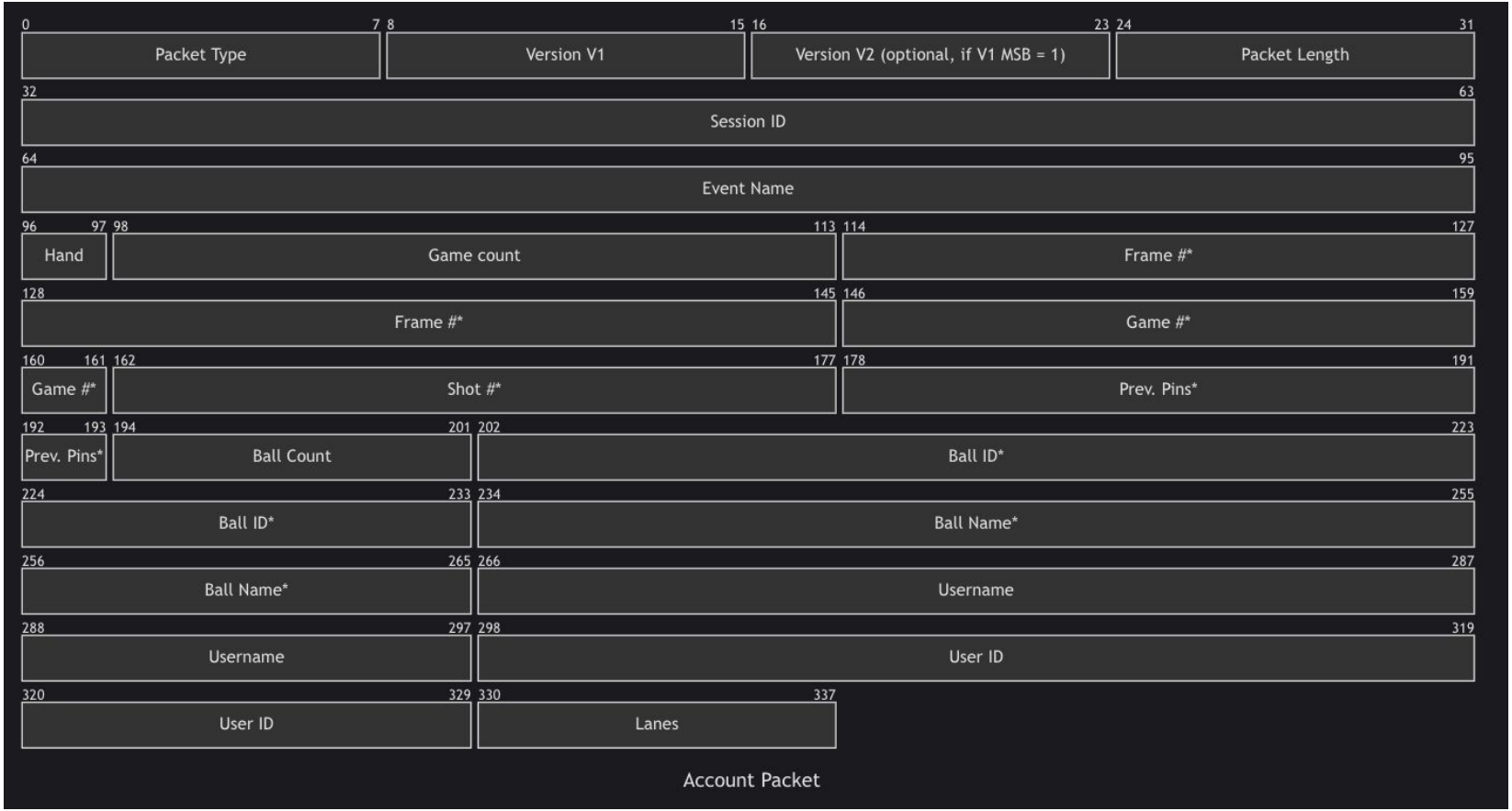
# Smart Watch (MS3 Achievements pt. 1)

- Shots saving to database after being received
  - Game created on watch add to database
- Added Sync button in watch settings
  - Command that resend account packet
  - Re-initializes the watch session UI
- Added parsing for anonymous session
  - Gets a random ID > 1,000,000
- End session functionality
  - Re-initializes with new session
- Log out functionality
  - Disconnects devices
- Lanes added to initial packet
  - Populating in shot page on watch
- Ui updating for each shot sent
- Deprecated unused pages
- Color scheme change on watch
- End game functionality adjusted in watch logic
  - Last frame challenges

# Shot Packet Diagram

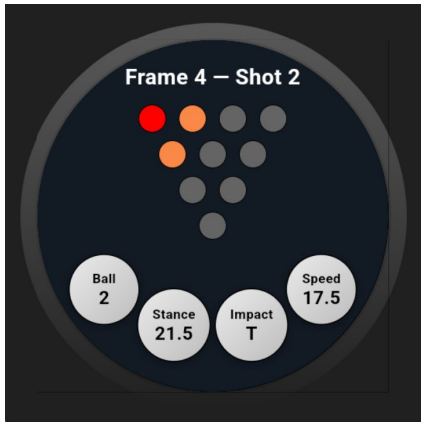


# Account Packet Diagram

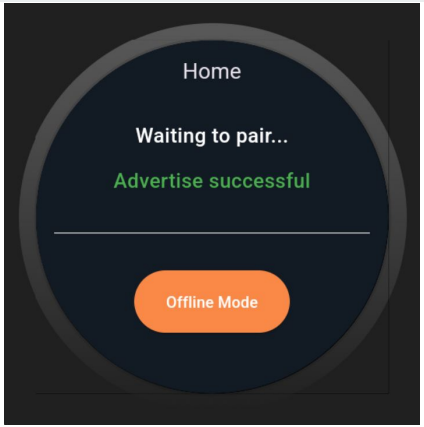


# Updated UI

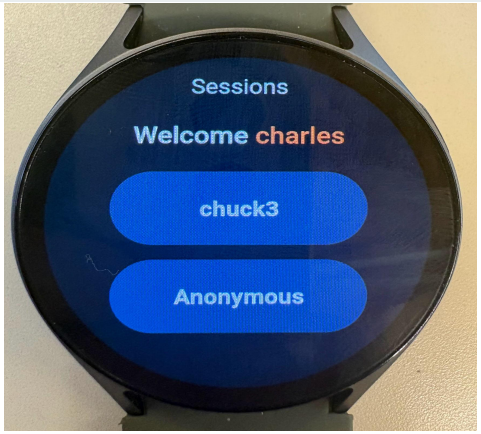

New UI coloring



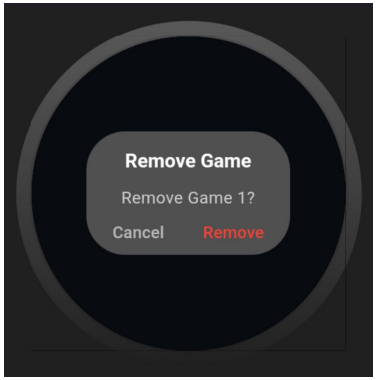

New coloring + Info Bar



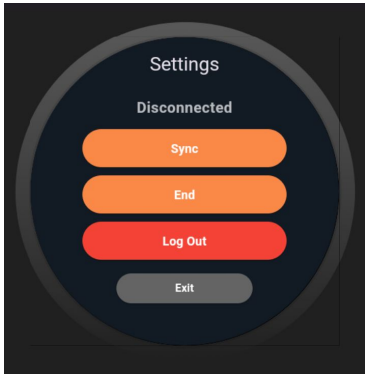

Home page



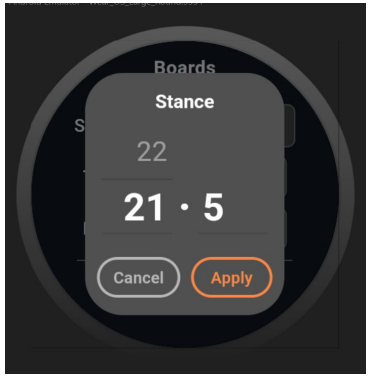

Sessions page




Remove confirmation



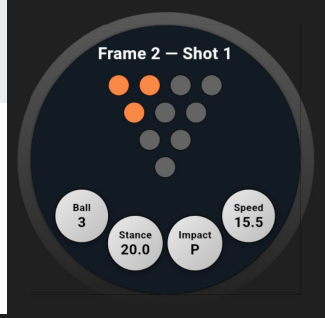

Settings page

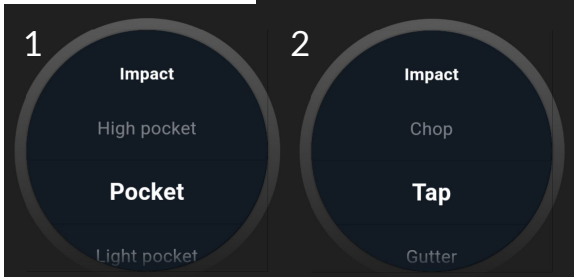


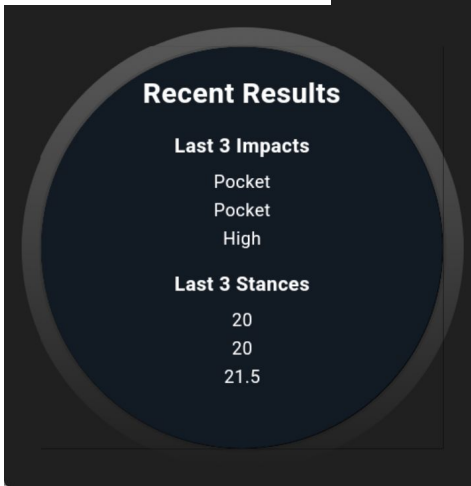

Stance Popup

# Smart Watch (MS3 Achievements pt. 2)

- Updated info bar
  - Shows more relevant information
  - New UI
- Shot input updates
  - Recent results now works, per shot #
  - Shot 2 impact now has a different
- BLE packet queue
  - Outgoing packets go through queue
  - Reattempts if failed
  - Triggers for local storage
  - Loop only runs when there is a packet




```
static const Map<String, int> _impactBoardMap = {
  'gutter': 0,
  'right': 11,
  'light': 13,
  'light pocket': 16,
  'pocket': 17,
  'high pocket': 18,
  'high': 21,
  'nose': 20,
  'brooklyn': 23,
  'left': 27,
};
static int secondShotImpactToBoard(String impact) {
  switch (impact.trim().toLowerCase()) {
    case 'right': return 1;
    case 'left': return 2;
    case 'chop': return 3;
    case 'tap': return 4;
    case 'gutter': return 5;
    case 'foul': return 6;
    case 'spare': return 7;
    default: return 0;
  }
}
```

```
// Called manually from BLEManager's _handleNativeCalls when 'onIndicationComplete' fires
void handleNativeIndicationComplete(MethodCall call) {
  if (call.method == 'onIndicationComplete') {
    final args = call.arguments as Map<dynamic, dynamic>?;
    final success = args?['success'] as bool? ?? false;
    if (_indicationCompleter != null && !_indicationCompleter!.isCompleted) {
      _indicationCompleter!.complete(success);
    }
  }
}
```

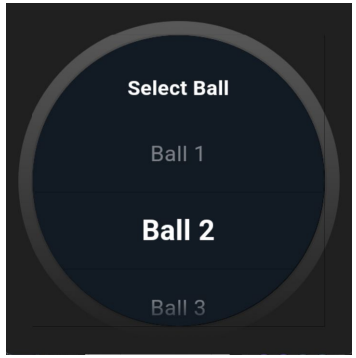
# Smart Watch (MS2 Achievements pt. 2 cont.)

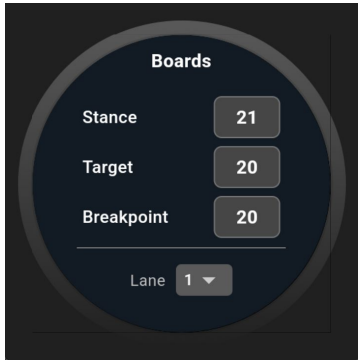
- BLE packet queue
  - Outgoing packets go through queue
  - Reattempts if failed
  - Triggers for local storage
  - Loop only runs when there is a packet
  - Watch can send new Indications, was Notifications

```
I/flutter (23054): Shot submitted: {shotNumber: 1, ball: 9, numOfPinsKnocked: 7, pins: 200, impact
: 17.0, board: 17.0, stance: 20.0, target: 20.0, breakPoint: 20.0, speed: 15.5, frameNum: 1, lane:
 1, isReadOnly: false}
I/flutter (23054): WATCH SESSION: Sending shot packet (23 bytes):
I/flutter (23054): WATCH SESSION Packet hex: 0x03 0x01 0x00 0x17 0x00 0x00 0x00 0x12 0x20 0x89 0x6
0 0xc8 0x28 0x28 0x28 0x22 0x00 0x9b 0x01 0x00 0x00 0x00 0x00
I/flutter (23054): WATCH SESSION Details - SessionId: 18, Game: 4, Frame: 1, Shot: 1
I/flutter (23054): [PacketQueue] Enqueued packet of size 23. Queue length: 1
I/flutter (23054): BLEManager.sendRawBLEPacket -> queued 23 bytes
I/flutter (23054): WATCH SESSION: Shot packet sent to phone successfully
I/flutter (23054): [PacketQueue] sent 23 bytes successfully
```

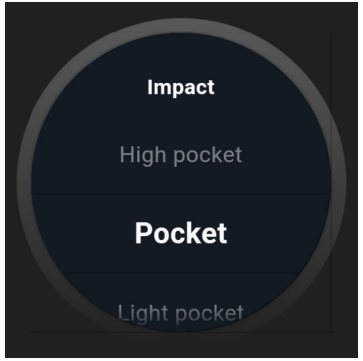
```
try {
  _indicationCompleter = Completer<bool>();

  // Attempt to send via the original BLE service channel
  await _channel.invokeMethod('sendNotification', {
    'serviceUuid': BleGattManagerConstants.serviceUuid,
    'charUuid': BleGattManagerConstants.notifyUuid,
    'bytes': packet,
  });

  // Wait for hardware ACK from Kotlin via onIndicationComplete
  final ackReceived = await _indicationCompleter!.future;

  if (ackReceived) {
    print('[PacketQueue] sent ${packet.length} bytes successfully and ACKed');

    // Confirmation successful, dequeue it
    dequeue();

    // Small delay before next packet to avoid swamping BLE buffer
    await Future.delayed(const Duration(milliseconds: 50));
  } else {
    print('[PacketQueue] Send failed (no ACK) or errored natively');
    // Wait before retrying the exact same packet
    await Future.delayed(const Duration(milliseconds: 5000));
  }
```

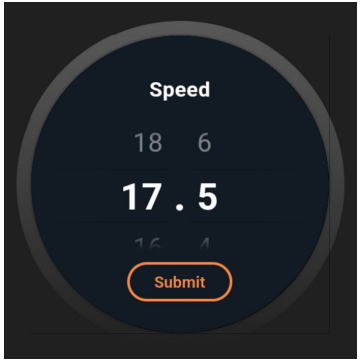
# Updated Workflow



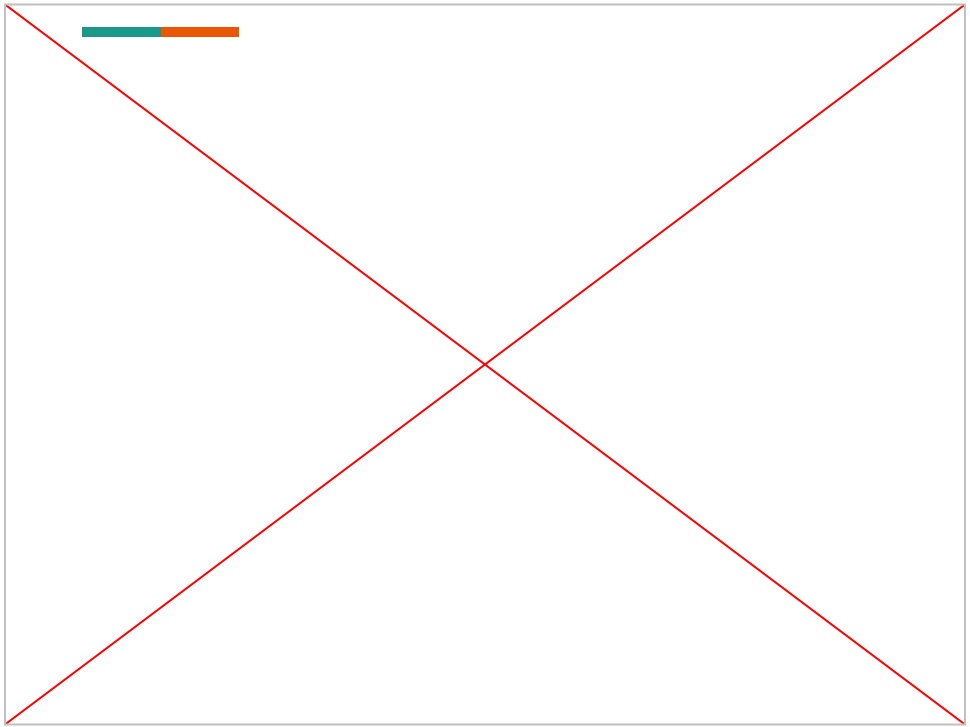



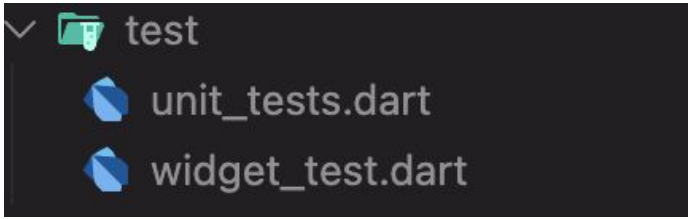

Submission of Shot

# Smart Watch (Future )

- BLE queue
  - Need an ack from phone upon receiving a packet
  - Change from Notification to Indication
    - Indications require ACK
- Bug fixes
  - End of game logic
  - Watch handler for UI
    - Next frame created before previous one finished
  - Disconnect issues
  - Anonymous session
- Testing
- Wiki page updated

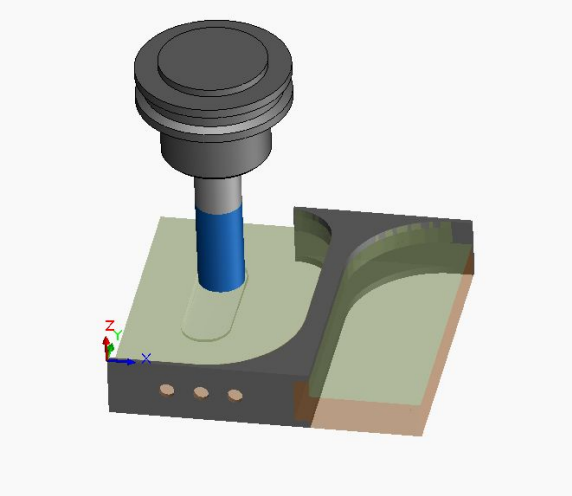
# Smartwatch Demo + Unit Tests



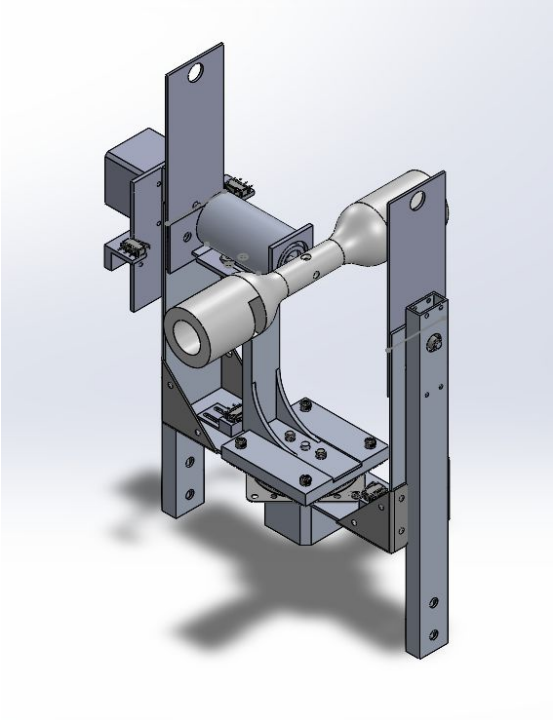



```
charlescarroll@DESKTOP-C7TGMLH flutter_prototype % flutter test test/unit_tests.dart
00:03 +31: All tests passed!
charlescarroll@DESKTOP-C7TGMLH flutter_prototype % flutter test ./test/widget_test.dart
00:01 +3: All tests passed!
```

# Physical Design Overview



CNC Milling Operation for Modified L Bracket



Final Design for Ball Spinner

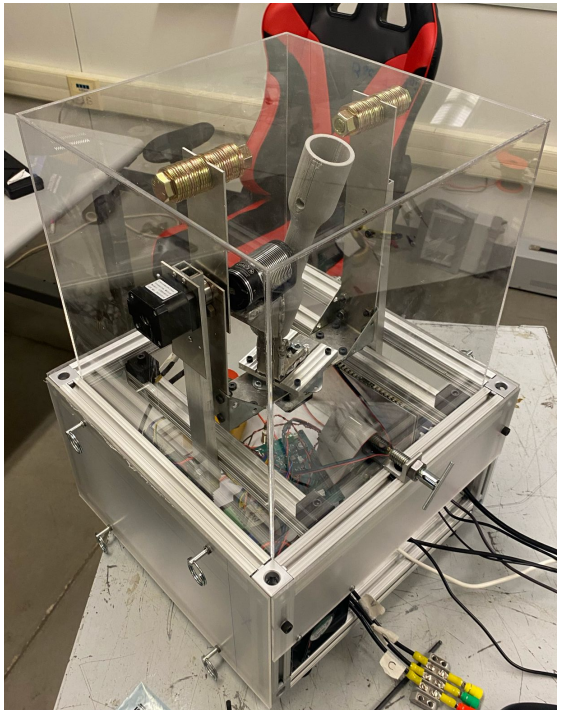
# Physical Design Team (Goals and Achievements for MS2)

## Goals

- Fabricate and assemble 3rd Degree of Freedom
- Finish testing to ensure motors meet required acceleration
- Limit switches implemented

## Achievements

- Acrylic Panels cut and machined
- Completed 3rd Degree Assembly with shaft collar
- L-Bracket reinforced and re-designed
- Testing of 3rd degree motor holding torque
- Counterweight system fabricated



Fully aluminum model with machined acrylic panels, counterweight system, and limit switch holes ready

# Physical Design Team (Future: Capstone Expo)

## Goals

- Finish testing to ensure motors meet required acceleration
- Limit switches implemented
- Fabricate new L-bracket
- 3D-Printed back plate
- Wiring/ Electrical Mounting