**Sprint Planning Document (Sprint 3)**

**Sprint Goal Backlog (Sprint 3)**

March 25 – April 21, 2025

Aiden Carroll, Gabe Gros, Kendall Hamm, Chris Hellen, Ethan Schwalbach

**High-level Project Overview**

**Project Mission:**

1. Basketball SmartGlasses Analytics is committed to delivering live statistical information from the Synergy live sports API to the Brilliant Labs AR glasses over a Bluetooth connection established by a custom-built Android app from our team.  
     
     
   **Problems We Are Solving:**
2. Live statistical information is not readily available on the sideline in real environments.
3. Coaches frequently must go look at stats written down or on tablets, taking their eyes away from the game for precious seconds.
4. Assistant coaches and other staff cannot quickly sort, filter, and deliver relevant live statistical information to the head coach in real time.

**Project Overview (High-Level Features):**

1. **Android Application:**
   1. **Starting Lineup:** Have widgets available that show all players currently on the floor, access their statistics with a press of the button corresponding to their position.
   2. **Team stats:** Have a button available to view stats like team shooting percentage, rebounds, etc.
   3. **Transmit button:** After selecting relevant stats, send the stats to the glasses over Bluetooth
   4. **Establish Bluetooth connection:** Use Kotlin over bluetooth to connect to glasses
   5. **Connect to Synergy API:** Make api requests, store information in a database for use in the rest of the app (all the stat buttons listed above)
   6. **Run python code in Android App:** Use available libraries to compile executable python code for API access
2. **Brilliant Labs Smartglasses**
   1. Connect over Bluetooth with a connection established by the android app, choose what information to send to the glasses and present it in a digestible manner.
3. **Backend Services**
   1. **Realtime Database**: Storing live data in the app for filtering and sending to the glasses.
   2. **API:** The team uses a service called Synergy for their live stat information.
   3. **Authentication:** The API has built in verification techniques like usernames and passwords to access the information.

**Sprint 3 Planning**

**Sprint 3 Goals:**

1. **Upgrade UI for App to include more features**
2. **Synergy API Integration to retrieve real statistics**
3. **Import full roster along with calculated advanced statistics**
4. **Connect to Frame Smartglasses from app**
5. **Display selected statistics on Frame Smartglasses**

**Sprint 3 Deliverables:**

1. **Upgrade UI for App to include more features:**
   1. **Assigned:** {Aiden Carroll, Ethan Schwalbach}
   2. Add and style buttons for “Connect to Glasses”, “Get Stats”, and “Send to Glasses.”
   3. Integrate full‑screen loading and error states in the new UI flows.
   4. Polish typography, spacing, and color scheme across all screens.
2. **Synergy API Integration to retrieve real statistics**
   1. **Assigned:** {Aiden Carroll, Ethan Schwalbach}]
   2. Integrate and configure API client module to authenticate and fetch game data.
   3. Expose fetched data to the app’s ViewModel layer for downstream UI consumption.
   4. Ensure structured error handling and retry logic for robust live-data retrieval.
3. **Import full roster along with calculated advanced statistics:**
   1. **Assigned:** {Kendall Hamm, Chris Hellen}
   2. Parse incoming result.json to extract the complete team roster.
   3. Calculate advanced metrics (PER, EFG%, TS%) in Kotlin.
   4. Display the full roster in a scrollable Compose list with placeholder images for unknown players.
4. **Connect to Frame Smartglasses from app:**
   1. **Assigned:** {Gabe Gros}
   2. Architect a robust BLE module with advanced scanning filters, auto-reconnect logic, and lifecycle management.
   3. Include metrics logging (RSSI, throughput, errors) to validate real‑world reliability.
   4. Implement secure pairing and encrypted communication, managing GATT services and characteristics.
5. **Display selected statistics on Frame Smartglasses:**
   1. **Assigned:** {Entire team}
   2. Offer a dynamic stats‑selection UI and translate user choices into structured payloads.
   3. Implement on‑device acknowledgment: verify the Frame confirms receipt and rendering of each payload.