ID	Requirement	Description	Story Points	Priority	Sprint Number
R1	Platform & Architecture Research	Research and validate optimal architecture using Android Studio with Unity SDK bridging to Meta Quest Pro.	13	P0	1
R2	SDK & Toolchain Validation	Evaluate Meta Quest Pro SDKs, Unity XR pipeline, Android build targets, and integration constraints.	8	P0	1
R3	Data Flow Design	Define end-to-end data flow between phone sensors, Android app, Unity bridge, and headset.	5	P0	1
R4	Security & Privacy Baseline	Research Android health data access, permissions, encryption options, and local-only storage models.	5	P0	1
R5	HUD Overlay Baseline (MVP)	Implement a minimal in-headset HUD overlay with proper placement, anchoring, and visibility controls using Unity.	21	P0	2
R6	Widget System Core	Implement modular widget framework supporting widget creation, placement, repositioning, and removal.	8	P0	2
R7	HUD Performance Optimization	Optimize rendering and update loops to maintain stable ≥72 FPS on Meta Quest Pro.	8	P0	2
R8	Session Persistence	Persist HUD layout, widget positions, and configuration across sessions.	5	P1	2
R9	Proof-of-Concept Communication	Establish basic communication between Android device and Unity application using defined messaging protocol.	8	P0	3
R10	Health Data Ingestion	Implement Android-side ingestion for heart rate and basic health metrics using Health Connect or custom sources.	8	P0	3
R11	Unity Bridge Foundation	Implement Unity-side data receiver and internal routing system for incoming phone data.	8	P0	3
R12	Android App Skeleton	Create Android Studio project with modular architecture, module layout, and build configuration.	5	P0	3
R13	Permission & Consent Framework	Implement explicit, revocable permission handling for health and sensor data on Android.	5	P0	3
R14	Data Serialization Layer	Define structured, versioned data models for metrics to be sent to the headset.	5	P0	3
R15	Local Data Handling	Ensure all collected data remains local by default with clear lifecycle and retention rules.	5	P0	3
R16	Connection Reliability	Handle connection loss, retries, and safe fallback states between Android and Unity.	8	P0	4
R17	Latency & Smoothing	Implement smoothing and buffering to keep perceived data latency under 3 seconds.	5	P1	4
R18	Telemetry & Debug Logging	Add internal logging for data flow timing, dropouts, and message integrity.	5	P1	4
R19	Initial Performance Profiling	Measure baseline CPU, memory, and data throughput costs under real update conditions.	5	P1	4
R20	Heart Rate Display	Render real-time heart rate with bpm indicator, update state, and no-signal handling.	5	P0	4
R21	Extended Health Stats	Integrate additional health metrics such as steps, calories, HRV, or SpO_2 when available.	8	P1	5
R22	GPS Widget	Integrate phone-sourced GPS speed and heading with graceful failure handling.	5	P1	5
R23	Safety Mode	Add low-distraction HUD mode with optional dimming and reduced motion effects.	5	P1	5
R24	Error Handling & Recovery	Implement unified error handling and recovery across Android app, bridge, and HUD.	5	P1	5
R25	Cross-Device Compatibility	Test functionality across multiple Android devices and sensor sources where possible.	5	P1	5
R26	Line-of-Sight Mapping (Experimental)	Investigate gaze-based association of nearby users with strict fail-safe behavior.	13	P3	6
R27	Proximity Discovery (Experimental)	Explore opt-in nearby presence detection using anonymous identifiers.	8	P3	6

ID	Requirement	Description	Story Points	Priority	Sprint Number
R28	Shared Stats Rendering	Render minimal shared stats with explicit, mutual user consent.	8	P3	6
R29	System Refactor & Cleanup	Refactor codebase, clean dependencies, and stabilize architecture for final delivery.	8	P1	6
R30	Final Integration & Demo	Complete end-to-end integration, testing, documentation, and final demonstration.	8	P0	6
R31	Proximity Security Controls	Enforce mutual consent, session-based opt-in, and immediate revoke controls.	5	P3	6
R32	Ethical & Privacy Evaluation	Evaluate privacy, misuse risk, and ethical implications of social AR features.	5	P3	6