



Listen-In An Android/Stethoscope Application

Team DEC13-12

Adviser

Luke Pint Tyler Treat
Sean Strickland Juan Mozqueda
Daniel Heiniger Alex Swanson

Dr. Akhilesh Tyagi



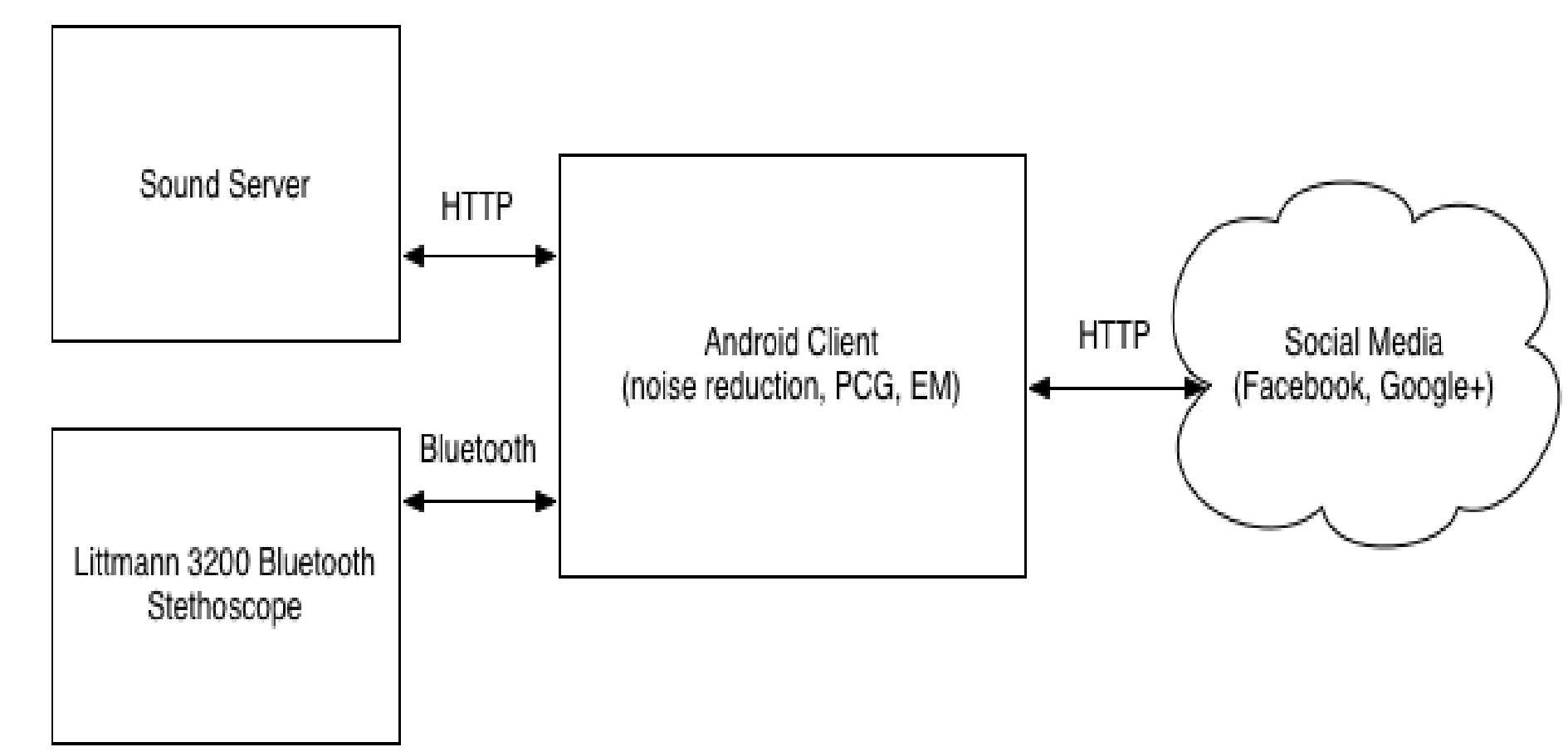
Problem Statement

Medical professionals need the ability to analyze stethoscope data away from their office. In some cases, they need to train medical students and provide guidance on how to accurately diagnose conditions.

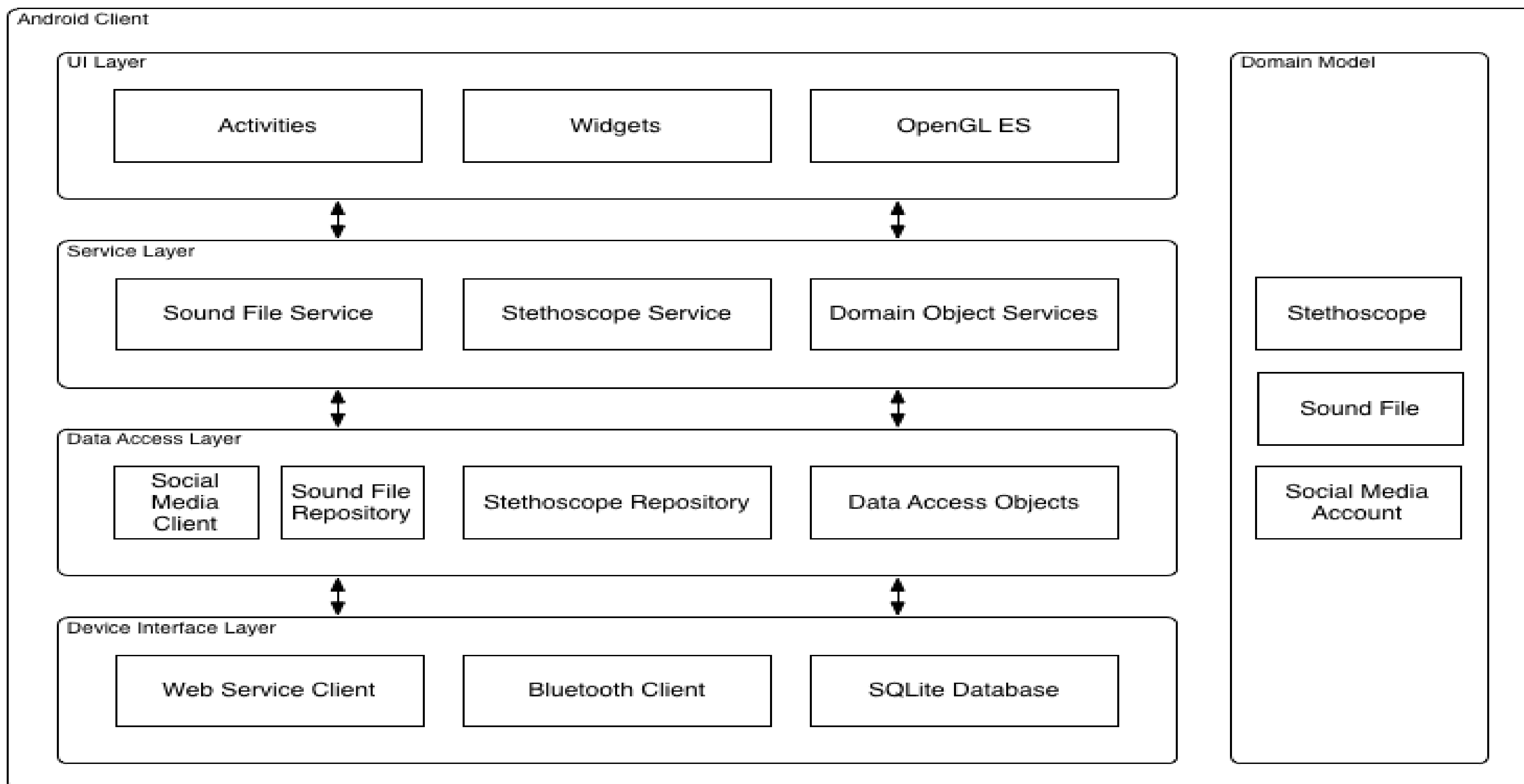
For our project we were asked to design an Android Application that interfaces with a Littmann 3200 Bluetooth Stethoscope. This application consists of four main components:

- Dashboard:** Allows users to manage their stethoscope bluetooth connections.
- Phonocardiogram:** Provides a graphical representation of the data being collected by the stethoscope.
- Audio Player:** Responsible for recording and playback of data.
- Sound Server:** A web based server that is used to share recorded files both publically and privately.

Architecture



Application Design



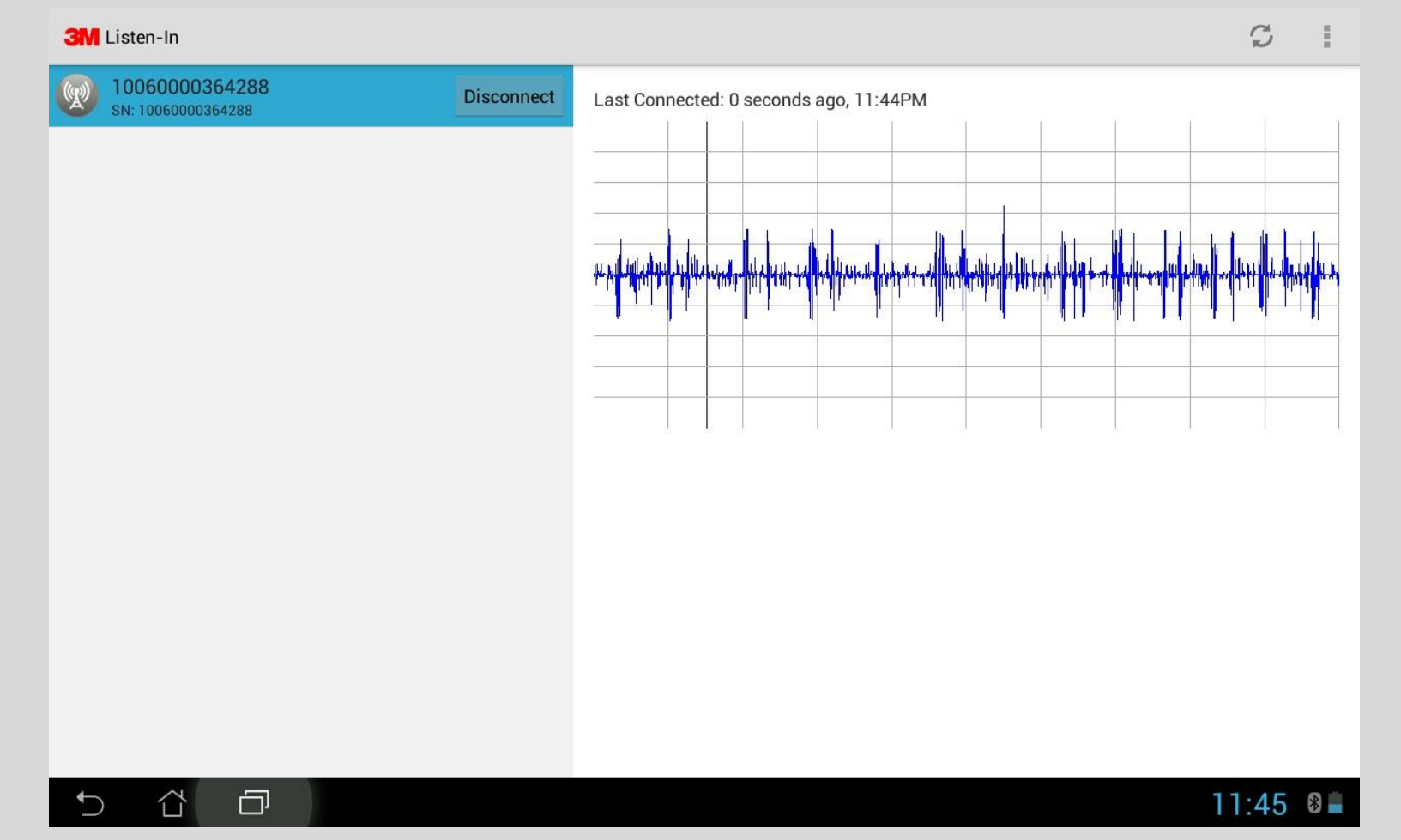
Application Dashboard

Contains access to the core features.

Device discovery automatically searches for stethoscopes in range.

Device pairing and renaming features are accessible by long clicking on a device.

Once a device is connected, a graphical representation of the data is displayed in a phonocardiogram.



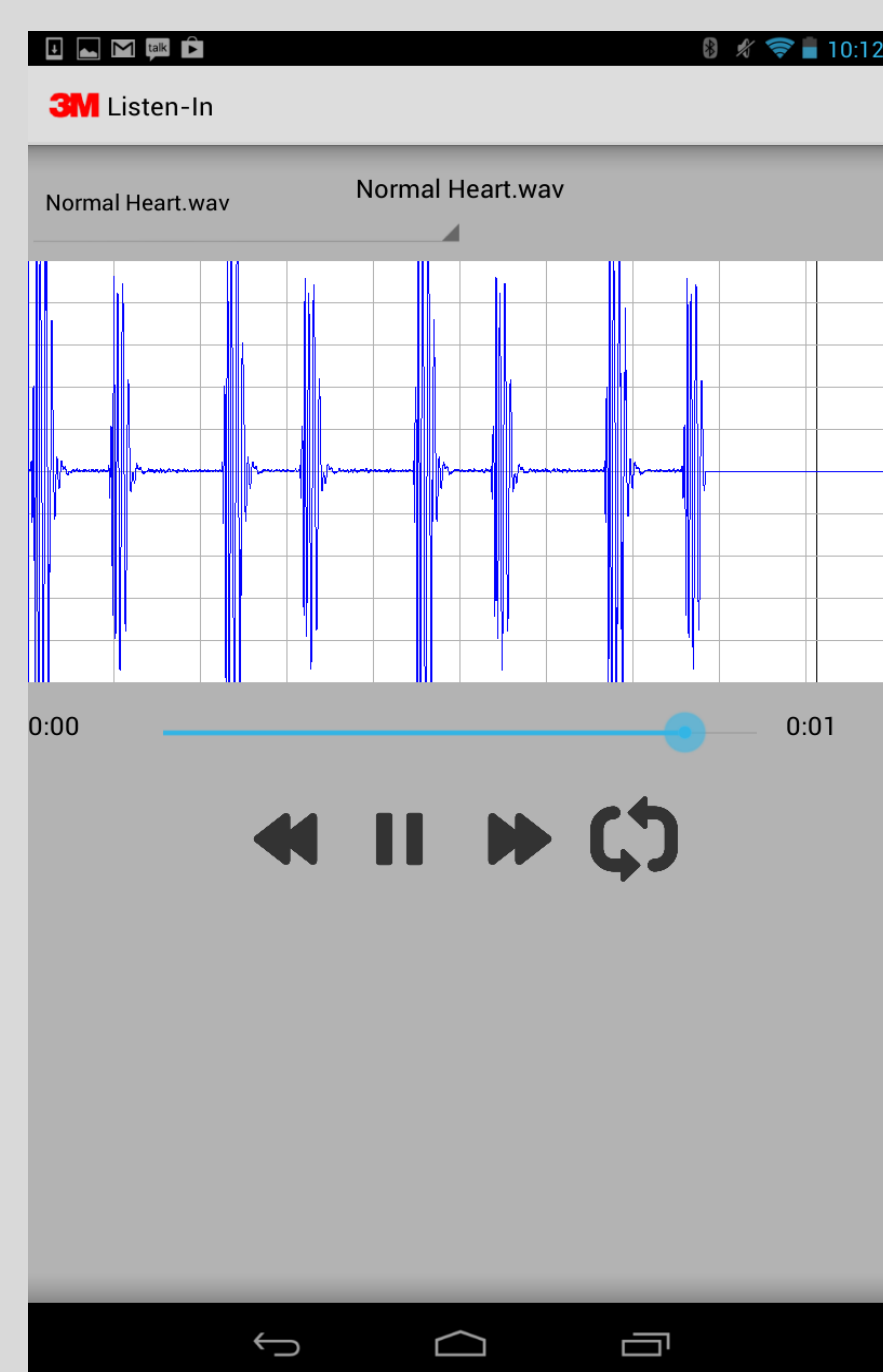
Recording and Playback

Overview

Ability to record cardiac sound files for playback.

Audio Player for playback sample files and recorded files.

Contains a phonocardiogram in the audio player to show visual representation of a heart beat.



Challenges

Debugging was nearly impossible because the stethoscope data comes at a live stream rate which causes latency problems.

We had to create a data manger for the stethoscope input stream that components subscribe to. The need arose when multiple components were attempting to read from the stream at the same time. This caused data being read to be incomplete.

Sound Server

Sound files can be uploaded to a remote server from which they can be downloaded later.

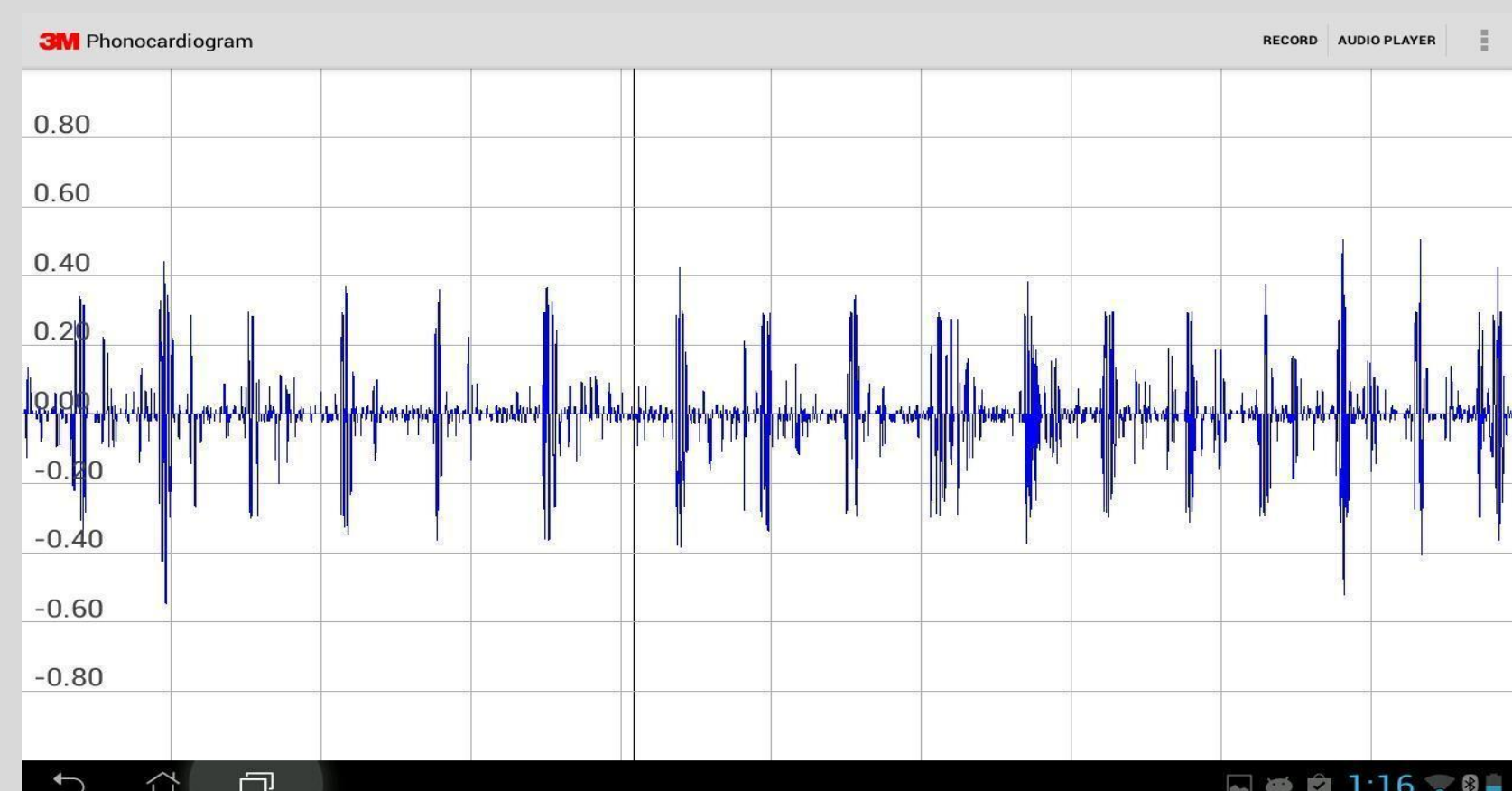
Uploaded files are accessible through the application.

Recorded files can be shared through social media platforms to connect medical professionals. A download link to the remote file is published to the desired platform.

Phonocardiogram (PCG)

Overview

The phonocardiogram (PCG) is the main component of the application. It is a widget developed using OpenGL ES for displaying sound amplitudes to represent heart rhythms. The PCG can be embedded in any application view. A key feature is the ability to process both live streaming data and recorded data.



Challenges

Problem: Drawing a 2D phonocardiogram in a 3D space.
Solution: Positioned the viewing camera such that we can focus on drawing only in the Z = 0 plane.

Problem: Drawing labels not natively supported in OpenGL.
Solution: Created a layer on top of the phonocardiogram for labeling.

Problem: Synchronizing sweep line and data plotting.
Solution: Made sweep line an independent "time keeper" which dictates when the data graph vertices update and the number of vertices that update.

Test Plan

Unit tests for the logic behind individual components.

Unit tests for integration of multiple components verifying they still interact as desired.

Manual Regression Testing for components dependent on streaming data from the stethoscope. These include:

- Sound playback
- Phonocardiogram
- Recording

