

# Software Engineer II Code Challenge

## ECG Primer

An electrocardiogram (ECG) is a measurement of electrical potential inside of a person's heart. ECGs are used to diagnose and treat many cardiac conditions and can make use of multiple cables (called "leads") placed on a patient's body at different locations around the heart. Spaulding Medical typically deals with 12-lead ECGs, which have the names "I", "II", "III", "aVR", "aVL", "aVF", "V1", "V2", "V3", "V4", "V5", and "V6". Lead II is the lead most commonly used in patient diagnosis and special attention is paid to it.

## Skeleton

A project skeleton has been set up for you in `CMakeLists.txt` and `src/Main.cpp`. It uses GLAD and GLFW to manage initializing the window and OpenGL context. It also sets up a few basic window event callbacks. If you're more comfortable with a library other than GLFW, you are welcome to swap it out.

## Instructions

1. Read `The_ISHNE_Format.pdf` in "docs" to get an understanding of the ISHNE format for Holter files.
2. Implement the `ECGViewer::LoadFile` function in `src/Main.cpp`, which is called when the user drops a file into the window. An example ISHNE file has been provided in `files/5MinExample.ecg`. Make sure to perform proper error handling.
3. Draw 10 seconds of (at least) a single lead of ECG data (II) onto the screen using OpenGL in the `ECGViewer::RenderLoop` function. An example screenshot of all 12 rendered leads is on the next page.
4. Implement the `ECGViewer::MoveView` function in `src/Main.cpp` to move the view of the ECG back and forward by 10 second intervals when the user presses the arrow keys. Make sure to perform proper bounds checking.
5. ZIP up your code (making sure to remove any build artifacts!) and send it to [nick.fetcko@spauldingmedical.com](mailto:nick.fetcko@spauldingmedical.com) along with any comments you may have about this challenge.

## Rules / Restrictions

1. Must be written in modern C++ (C++11/14/17).
2. Must be crossplatform and compile / run on at least Windows and Linux (bonus points for macOS!).
3. OpenGL immediate mode (`glBegin()` / `glEnd()`) is strictly forbidden.
4. CMake must be used (a working `CMakeLists` has been provided).
5. You are not restricted to only the `ECGViewer` class. Use a class structure that works well / makes sense for you.
6. You are free to add additional functionality as you see fit. Rendering ECG can be a lot of fun, and we definitely encourage experimenting!

## 12-lead ECG Example Screenshot

