

## Heads Up Display (HUD) Using Google Glass

Henrique Savelli Abelha Gomes and Renan Martins Kuba

Illinois Institute of Technology

## Abstract

This paper describes the process to find a solution for displaying data from a repository populated by wireless mesh networks with attached sensors. The idea was put into practice developing an Android application for Google Glass and for smartphones, providing a user friendly interface for viewing the data within MySQL and integrating voice activation with a variety of output formats. The system was designed to be adaptable and interactive, creating a connection between users and sensors coming up with an easy way to read what was stored into the database.

Google Glass was chosen as a front end to demonstrate an unobtrusive and intuitive way to view the data, but hypothetically, the system could include applications for other mobile devices or a web interface.

**Keywords:** Google Glass, application, database, human-machine interaction

### Heads Up Display (HUD) Using Google Glass

Technology has changed the way that people see and interact with the environment.

Every passing year, humans try to reinvent their connection with machines, in order to make it friendlier and more accessible. Smart-phones and wearable devices have become more and more common in people's lives, and this fact inspired our group to develop a project for displaying data from a repository populated by wireless mesh networks with attached sensors.

Subsequently, a solution came up with the development of two Android applications, which gave us another perspective of the power of technology and how we can use it to improve the way that employers work.

We started our project understanding how the database work and what would be the best option to receive real time information. After that, we decided to created a My Structured Query Language (MySQL) server and a table that would store all the information from the mesh network sensors. Finally, we shared this database with our partners, who created the wireless mesh network, and they populated it with the information captured by temperature, humidity and luminosity sensors.

Next, we discussed about all the possible solutions for the project. At first, we did not have a Google Glass available to work on. Therefore, we decided to use our previous knowledge about Android and we started to create a friendly application for smart-phones with a simple design. Our goal was to make an application that would provide all the possible information to the user with different outputs. We wanted to make the application very interactive, so we used resources like graphics and voice commands to create a better connection with the app. In

addition, we made a functionality that permits the user to change the names of the connected boards, what improve the communication with the database.

In order to create a well structured application, the Android app's architecture was developed in three layers based on model-view-controller (MVC) architectural pattern, each layer of our project represents one piece of the MVC pattern. Following this method and using the Android API, which leads to a MVC development, different kinds of classes were developed.

Classes such as Board and Sensors were created to represent models. Those classes have the behavior of storing data to be treated and displayed. Classes such as BoardDao and SensorsDao are classes that interact with the database, those classes are responsible for every data manipulation. Classes such as MainActivity, SensorsActivity, DetailsActivity and Settings, were created to make the interaction between view and model. Those classes represent the controller on MVC pattern. The view layer was developed using the XML files that the Android API gives the choice to use. This way, the whole user interface could stay in a separate file, facilitating the system maintenance.

Besides the Android app's architecture, the connection between app and database was well structured during the development. The connection is made by a class called ConnectionFactory, which is responsible for passing the credentials, such as login and password, for the correct server. Thus, the application can exchange information with databases and tables. For that, the class ConnectionFactory makes use of a Java API called Java Database Connectivity (JDBC),

After finishing the development of the Android application for smart-phones, we finally had the opportunity to use a Google Glass. Our first impression with this wearable device was completely immersive. We used it for a few hours without stop, learning about the technology and how the user would interact with it. Then, we started thinking about all the possible ways that we could make this Android application works giving the best user experience. Because the hardware and the software of Google Glass are both nascent technologies, we did not find a lot of documentation about it, what made the process of creation more difficult. Even though we had some issues, we developed an application that could read the data from the database and we could display it in the screen, giving a unique value of information for each of the connected sensor.

Although Google Glass is a different device, it uses Android as its Operational System, thus, its development is not too different from the smartphone development. With that in mind, some classes previously created for smartphone could be reused on Google Glass application, making the structure very similar to the smartphone application. The only difference between them is the form to manipulate models, the classes that do this also send information directly to the screen instead of passing it through a view layer. In addition, for the Google Glass application, the connection with the database works in the same way that the smart-phone software does.

In this project we have achieved the goal of creating a new Android application that could present all the data collected from a wireless mesh network with sensors in a user friendly connection. Our objective was to change the way that people see the information sent by a

machine, giving a new perspective of how a simple wearable device or a smartphone can immerse into the market as more than an entertainment platform, but as a helpful tool for professionals who work with research and analysis of data. Successfully, we completed the project with two different working solutions and we hope it impacts in the way that people use technology for their daily activities.

In case we had more time to work on this project, we would do different implementations for both of the applications. First, for smart-phone software, we would create a web-service that would get the data from the MySQL database and share it with the application. Read data directly from the database consumes battery and reduces the performance of the device. The web-service would correct this problem, working as a bridge between the database and the machine. Moreover, for the Google Glass, we would use the same web-service. We would also implement voice commands inside the application and use graphics to show the information collected, in order to extend the user experience for this device.