The goal of this project is to create a software interface for a built-in projector on a smartphone. The software will control all the projector hardware features in order to project the image a flat surface, and with the assistance of multiple sensors, determine if the surface is being touched and respond to the interaction. Basically the projected surface would be an extension of the smartphone screen, working with the same gestures and commands.

This is a totally new implementation of hardware, targeting users of all kind. The most important characteristic is the fast interaction, the range of applications of the Interactive Projector Phone can go from the user better expressing his/her ideas using presentations, to watching movies on a big projected screen at any place where you can turn off the lights.

For this product three main functional requirements describes the most important features: interaction recognition at projected surface, the response of the interaction and the drawing feature. The main non-functional requirements are related to the performance in order to be productive we need to provide a fast response to the user interaction. The precision and accuracy in a sense that the user can only interact with the projected screen. For reliability the system should make a backup of the current phone state and shutdown the application once the battery reaches 5% and finally for the supportability the software should be updated via internet.

The software use-case diagram contains six use cases: “initiate application”, “calibrate projected screen interaction area”, “check network for updates”, “read and execute user interaction”, “take notes at the screen”, and “change projected screen settings”. The actors in this system are the “user”, “projection hardware”, “screen sensors”, and “phone screen”.

There are also six test cases, with each one corresponding to a particular use case. The testing materials consist of a smartphone containing projection hardware and projection sensors. On the software side, the test operating systems are Blackberry OS, iOS, android and windows phone OS.
The design paradigm consists of a hardware-software mapping. There is a two-directional relationship between the interactive projection software and the projection hardware, as well as a two-directional mapping between the interactive projection software and the sensors array. As stated in the design document, the sensors array is built in the smartphone responsible to read all the user interaction at the projected screen (all these interactions are interpreted by the Interactive Projection Application).

With regards to persistent data management, the “backup” subsystem and “settings” subsystem is all a part of a larger system called “persistent data”. “Backup” is comprised of the background application, the projected screen content, the smartphone screen content, and the notes taken on the screen.

Our product will have four stages with regards to its planning that come together in a cyclical relationship. The first stage is the “brainstorming” stage where developers think of ideas on how to develop a certain subcomponent of the software system. The second and third stages are the development and testing stages respectively. After the release stage (the fourth and final stage), the first stage will be reactivated subject to time and schedule constraints.

There are eight classes in the software class diagram depicted below. This class diagram represents the components of the basic software system and their inner-workings.