

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
THE UNIVERSITY OF TEXAS AT ARLINGTON

ARCHITECTURAL DESIGN SPECIFICATION  
CSE 4316: SENIOR DESIGN I  
FALL 2019



**xXPew<sup>3</sup>Xx**  
**LASERS: COMBAT EVOLVED**

DIPTIN DAHAL  
KATARINA GOMEZ  
PHU LY  
JASON AUTRY  
TAUSIF ZAMAN

## REVISION HISTORY

Revision	Date	Author(s)	Description
0.1	10.01.2015	GH	document creation
0.3	11.08.2019	DD, JA, KG, PL, TZ	complete draft
0.4	01.11.2020	KG	Hardware modified
1.0	05.14.2020	TZ	Final Version

## CONTENTS

<b>1</b>	<b>Introduction</b>	<b>5</b>
<b>2</b>	<b>System Overview</b>	<b>6</b>
2.1	Android Client Layer Description . . . . .	6
2.2	Vest Layer Description . . . . .	6
2.3	Gun Layer Description . . . . .	7
2.4	Host Server Layer Description . . . . .	7
<b>3</b>	<b>Subsystem Definitions &amp; Data Flow</b>	<b>8</b>
<b>4</b>	<b>Android Client Layer Subsystems</b>	<b>9</b>
4.1	Application Subsystem . . . . .	9
<b>5</b>	<b>Host Sever Layer Subsystems</b>	<b>10</b>
5.1	Controller Subsystem . . . . .	10
5.2	Database Subsystem . . . . .	11
<b>6</b>	<b>Gun Layer Subsystems</b>	<b>12</b>
6.1	Subsystem 1 . . . . .	12
<b>7</b>	<b>Vest Layer Subsystems</b>	<b>13</b>
7.1	Subsystem 1 . . . . .	13

## LIST OF FIGURES

1	Simple Architecture Layer Diagram . . . . .	6
2	A simple data flow diagram . . . . .	8
3	Example subsystem description diagram . . . . .	9
4	Example of Host Sever diagram . . . . .	10
5	Example of Host Sever diagram . . . . .	11
6	Example subsystem description diagram . . . . .	12
7	Example subsystem description diagram . . . . .	13

## LIST OF TABLES

2	Subsystem interfaces . . . . .	9
3	Subsystem interfaces . . . . .	10
4	Subsystem interfaces . . . . .	11
5	Subsystem interfaces . . . . .	12
6	Subsystem interfaces . . . . .	13

## 1 INTRODUCTION

Lasers Combat Evolved is a simple to use android application that works with a gun and vest pair to allow the user to play a game of laser tag in a geographical area of their choosing by making use of the phone's gps tracking ability. The phone will act as a relay and validator for the vest/gun to communicate with a remote database server that will store information about the game. The phone will also work as a map for the player by retrieving the player locations and displaying them on a map of the user defined game area.

## 2 SYSTEM OVERVIEW

The overall system will be similar to a Real-Time System where information will be updates after a certain amount of time. The system will be consist of 3 main layers Host Sever Layer , Android Client Layer and Hardware Layer. The basic strategy is letting them handle difference data processes as such Hardware Layer will be handle only signal data from any hardware devices and pass them on to the Android Client Layer. Using a certain condition and implementation, these data will decrepit at the Android Client Layer and form a package before communicating with the Host Server Layer. Lastly, Host Server Layer will receive information from Android Client Layer to control and process it for placement in the database. The processed data will then sent to the database for storage until it is needed by the the Android Layer. The relevant processed data will then be sent for display.

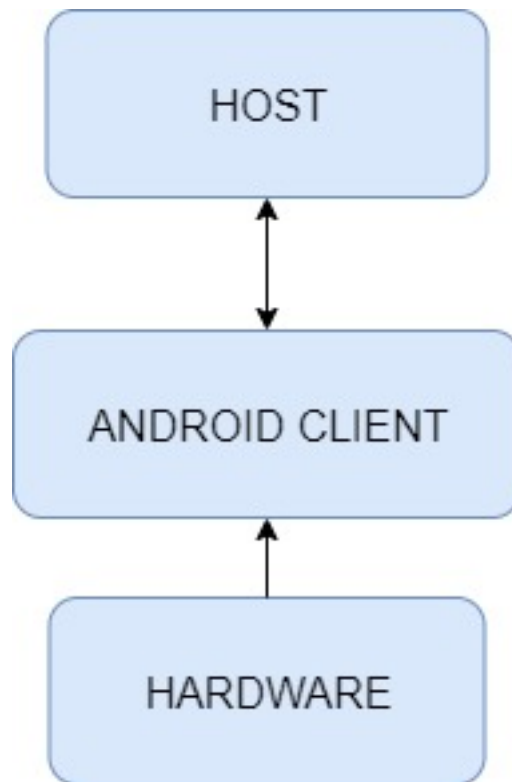


Figure 1: Simple Architecture Layer Diagram

### 2.1 ANDROID CLIENT LAYER DESCRIPTION

The android client layer will receive packets from the hardware layer and do a simple check to make sure the information it received is valid. Next it will send the data to the Host Server Layer for storage. This layer will also request the data it needs from the Host server for it's display.

### 2.2 VEST LAYER DESCRIPTION

Vest Layer consist of two subsystems : IR receiver and Bluetooth transmitter. The main purpose of the vest layer is to recognize when the player gets shot and then transmitting that data. It will take a signal output by the Gun layer and relay it to the Android Client.

### **2.3 GUN LAYER DESCRIPTION**

The gun layer will take in player data from the Android Client Layer and output an IR signal containing that data to other users.

### **2.4 HOST SERVER LAYER DESCRIPTION**

Host Server Layer (HSL) will handle most of the data communication between different Android Client Layers and it also performs the communication with database. Some of the HSL functions is to establish communication with the database and utilize the database as one of its resources. HSL will also create room for players and generate key room for invitation. This method will eliminate the match matching process and will allow players to have more freedom on who they want to play with since they will mostly be playing on the same playground. HSL also handles the scoring process of the game where it will store and generate a table of points or kill/death/rate as the result of the game.

### 3 SUBSYSTEM DEFINITIONS & DATA FLOW

The subsystem consists of three separate layers which work together to get the whole subsystem working in smooth flow. The layers are the following: Android Client, Host Server and Hardware. Here we see the flow of data between different layers. We can assume the origin is the application in the Android Client and when a new action is performed by the hardware the android client receives that signal and logs it. The logged data is sent and saved in the host server. This data can also be pulled back by the android application. So the Android Client is the main sender and receiver of finalized data in the form that can be displayed in the game application. The flow of data goes from hardware to client to host server and it can also go from host server to Android client (but data in this path does not go to the hardware).

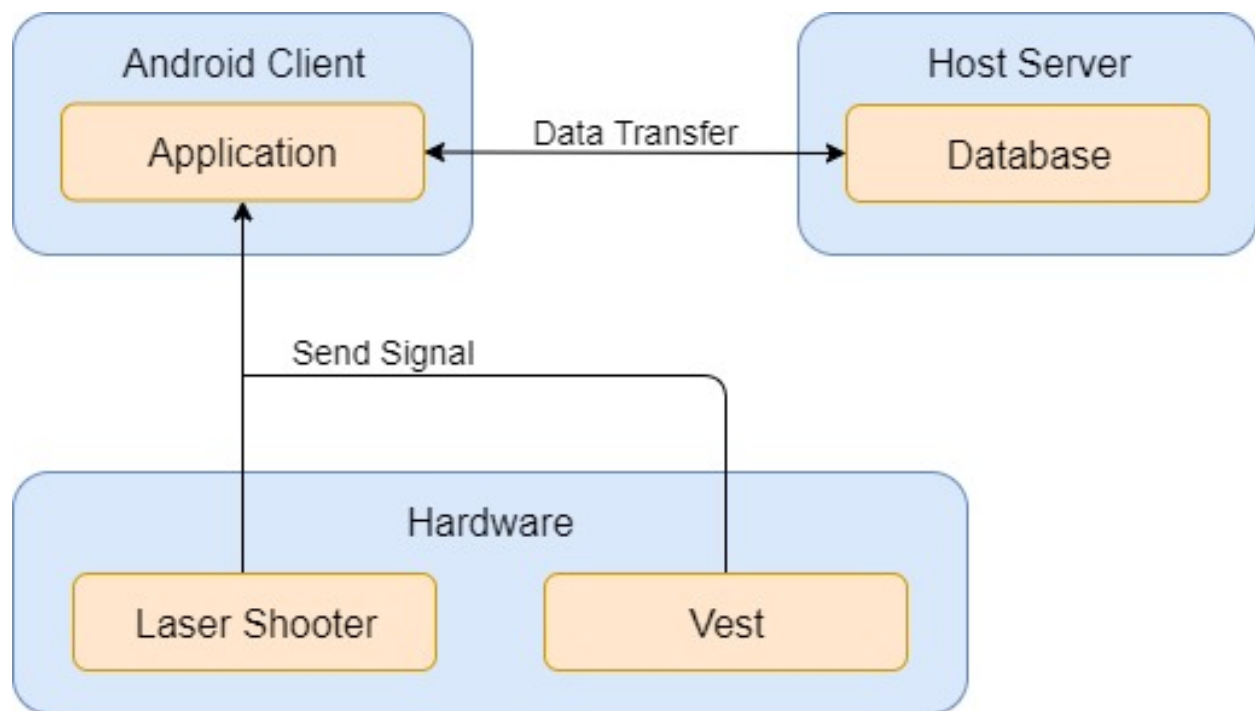


Figure 2: A simple data flow diagram



## 4 ANDROID CLIENT LAYER SUBSYSTEMS

### 4.1 APPLICATION SUBSYSTEM

This subsystem manages the display and the user inputs.



Figure 3: Example subsystem description diagram

#### 4.1.1 ASSUMPTIONS

The application will be on version 8.0 or later of Android. And the OS will manage any back end tasks that deal with interfacing with the hardware.

#### 4.1.2 RESPONSIBILITIES

This subsystem manages the display, user inputs, and requesting information from the server.

#### 4.1.3 SUBSYSTEM INTERFACES

Table 2: Subsystem interfaces

ID	Description	Inputs	Outputs
#01	Screen	user touches	information packets
#02	Bluetooth Signal	Decoded IR signal from gun	packets for server

## 5 HOST SEVER LAYER SUBSYSTEMS

### 5.1 CONTROLLER SUBSYSTEM

Controller will be an abstracted subsystem that handling and organize the data received from Android Client Server. It will also communicate and work with the Database to create objects and assigning players into appropriate room.

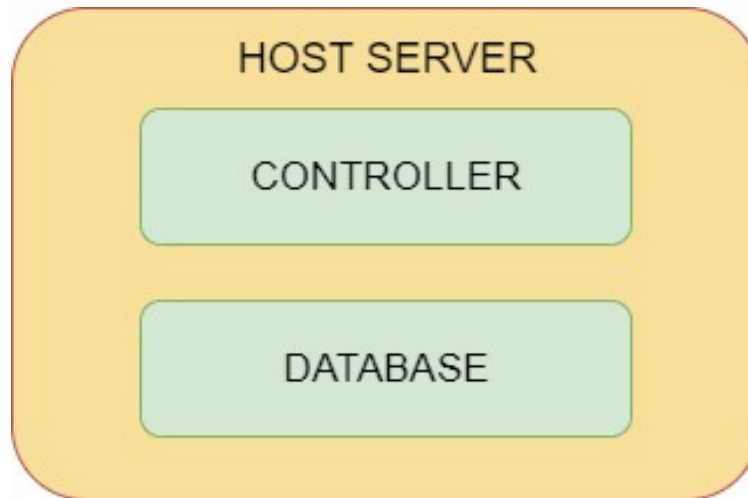


Figure 4: Example of Host Sever diagram

#### 5.1.1 ASSUMPTIONS

Assume all Data will be in a package and also assume that the connection between Host Server and Android Client Data will be established before hand .

#### 5.1.2 RESPONSIBILITIES

Main responsibilities for this Subsystem are first, organizing data and modify its structure so that it is easier to change and import into database as well as retrieving information from database. Second, Controller also respond in recognizing Multiple Android Client Layers to defined them in what room and interacting with what type of group. Data processing is a critical functionality of controller and a unique algorithm will be used to process specific type of data will be necessary.

#### 5.1.3 SUBSYSTEM INTERFACES

Controller is an abstracted subsystem and only developer is allowed to make any changes and/or implementation on top of it.

Table 3: Subsystem interfaces

ID	Description	Inputs	Outputs
00	No Interfaces due to the abstraction level	N/A	N/A

## 5.2 DATABASE SUBSYSTEM

Database Subsystem is as the name itself had said. Database will handle the data stored for generating room, room's hash-key and player's name, game scores. Entry and export of data will be handled through controller using values from database.

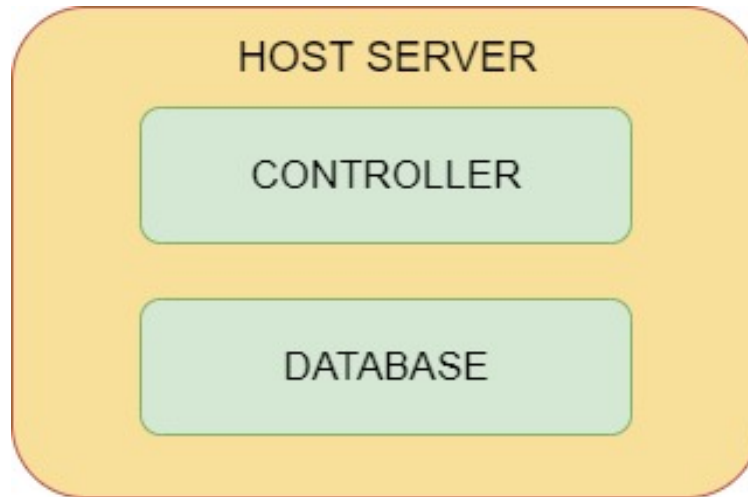


Figure 5: Example of Host Sever diagram

### 5.2.1 ASSUMPTIONS

Assuming there are unlimited data space.

### 5.2.2 RESPONSIBILITIES

Database will be responsible for accounting and manage data. Database will also responsible to organizing database within itself.

### 5.2.3 SUBSYSTEM INTERFACES

Database will be abstracted away from the player so there is no necessary interfaces to be shown or display on the application

Table 4: Subsystem interfaces

ID	Description	Inputs	Outputs
00	No Interfaces due to the abstraction level	N/A	N/A

## 6 GUN LAYER SUBSYSTEMS

### 6.1 SUBSYSTEM 1

This subsystem comprises of all the hardware aspects of the gun device. The system comprises of a IR emitter, which will be used to send the necessary packets of information to the opponent's vests.

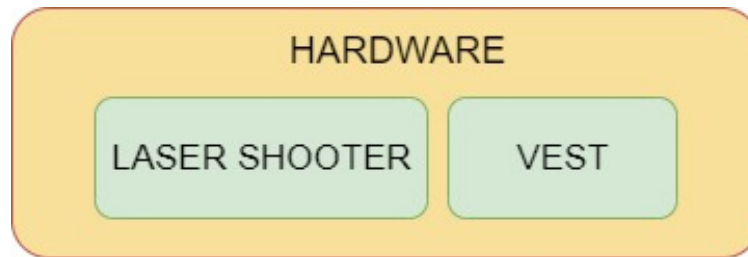


Figure 6: Example subsystem description diagram

#### 6.1.1 ASSUMPTIONS

It is assumed that the signal sent and received is not interfered by the external signals. It is also assumed that there is no transfer and receive latency between the Android Client subsystem and the Hardware Client subsystem.

#### 6.1.2 RESPONSIBILITIES

This layer is responsible for transmitting digital signals into the real world as analog signals to transmit information. Signals should be generated with high accuracy when the trigger is pressed on the gun, and the receiver on the vest should transmit the received "gun shot" signal with high accuracy and filter.

#### 6.1.3 SUBSYSTEM INTERFACES

This layer handles the output of the overall system, it will transmit the IR when trigger is pressed by the user.

Table 5: Subsystem interfaces

ID	Description	Inputs	Outputs
#xx	IR Emitter	Trigger Pulled	Shoot Laser

## 7 VEST LAYER SUBSYSTEMS

### 7.1 SUBSYSTEM 1

This subsystem comprises of all the hardware aspects of the vest device. The system comprises of a IR receiver, which will be used to receive the necessary packets of information the opponent's gun have sent, and a Bluetooth transmitter to communicate the packages it receives to the phone.

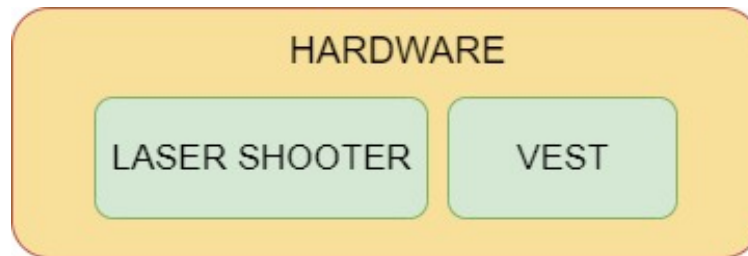


Figure 7: Example subsystem description diagram

#### 7.1.1 ASSUMPTIONS

It is assumed that the signal sent and received is not interfered by the external signals. It is also assumed that there is no transfer and receive latency between the Android Client subsystem and the Hardware Client subsystem.

#### 7.1.2 RESPONSIBILITIES

This layer is responsible for transmitting digital signals into the real world as analog signals to transmit information. Signals should be generated with high accuracy when the trigger is pressed on the gun, and the receiver on the vest should transmit the received "gun shot" signal with high accuracy and filter.

#### 7.1.3 SUBSYSTEM INTERFACES

This layer handles the output of the overall system, it will transmit the IR when trigger is pressed by the user.

Table 6: Subsystem interfaces

ID	Description	Inputs	Outputs
#xx	IR Receiver	IR Signal Received	Data to Bluetooth Module
#xx	Bluetooth Module	Data from Receiver	Data to Android Client