Initial Design Report

for

Building a Client-Server Architecture to Play Card Game, **BLÖFLÜ PİŞTİ** , from Mobile Devices

e-Limon

Utku SAKİL

A.Emirhan ÖZDEMİR

Cuma Tekin TOPUZ

Hamza YILMAZ

Contents

1.Introduction 4
1.1 Problem Definition 4
1.2 Purpose 5
1.3 Scope 5
1.4 Overview 5
1.5 Definitions and Abbreviations 6
1.6 References 6
2. System Overview 6
2.1 Goals 7
2.2 Objectives 7
2.3 Benefits 8
3. Design Considerations 8
3.1. Design Assumptions, Dependencies and Constraints 8
3.1.1. Design Assumptions 8
3.1.2. Design Dependencies 9
3.1.3. Design Constraints 9
3.2. Design Goals and Guidelines 9
4. Data Design 11
5. System Architecture 11
5.1. Architectural Design 11
5.2. Description of Components 13
5.2.1. Server 14
5.2.2. Android Client Application 17
5.2.3. Database 19
5.3. Design Rationale 20
5.4. Traceability of requirements 20
6. User Interface Design 20
6.1. Overview of User Interface 20
6.1.1. Login Menu: 21
6.1.2. Main Menu: 21
6.1.3. Lobby Menu: 21
6.1.4. Settings: 21

6.1.5. In Game Menu:..... 21
6.2. Screen Images 22
6.3. Screen Objects and Actions..... 25
6.3.1. Login Menu 25
6.3.2. Main Menu 26
6.3.3. Lobby Menu 26
6.3.4. Settings 26
6.3.5. In Game Menu..... 27
7. Libraries and Tools..... 27
7.1. Server Management Tool 27
7.2. Android Development IDE 28
7.3. Database Manager 28
8. Time Planning 29
8.1. Term 1 29
8.1. Term 2 29
9. Conclusion..... 30

1.Introduction

This document is an initial design report for **Building Server-Client Architecture to Play Card Games, BLÖFLÜ PİŞTİ, from Mobile Devices.** To introduce you the document well, we will first give problem definition of our project and then the purpose of this IDR, scope of this document, then follow an overview which explain how the IDR is organized and what the report contains and we will state the descriptions and abbreviations that are used in our IDR report and references which state all the documents and sources that are used in our IDR report. After finishing the introduction we supply **a System Overview** which constitutes of a general description of the software system including its functionality and matters related to the overall system and its design. We will briefly explain the goals, objectives and benefits of our Project in this part. Namely this part will provide the basis for the brief description of our product After completing **System Overview**, we will address **Design Considerations** which Special design issues need to be addressed or resolved before attempting to devise a complete design solution. Following **Data Design** which consists two separated part: **Data Description** that explain how the information domain of our system is transformed into data structures and the other **Data Dictionary which** alphabetically list the system entities or major data along with their types and descriptions . Then **System Architecture** and then **User Interface Design, Detailed Design, Libraries and Tools, Time Planning** and finally **Conclusion.**

1.1 Problem Definition

We will build a game server where clients can connect and play a simple card game, ***BLÖFLÜ PİŞTİ***, with each other or with an artificial agent running on the server. We will build:

1. A game server that keeps all the information about the users and ongoing games, manages all the communication between clients (dealing cards, sending moves, etc).
2. The server will be implementing a web service. It is specified a well defined protocol for the communications between clients and the server. There will be no peer to peer communication between clients.

3. A client program with a cute user interface that runs in a mobile platform, i.e. on Android systems. In our case Android system.
4. An agent that runs in the server and acts as a client. (The communications between the agent and the server will use the same protocol) The card game is "blöflü pişti". It is a simple but nontrivial two player card game of luck and skill, played with a standard deck of 52 cards.

1.2 Purpose

The purpose of this document is to show how the software system will be structured to satisfy the requirements **Building Server-Client Architecture to Play Card Games, BLÖFLÜ PİŞTİ, from Mobile Devices** to be developed, it will contain all the information required by a programmer to write code. Any programmer can write all the core codes while reading this report because this developer can see all the requirement information about our project and the diagrams will show the way how developer starts coding and what he will code.

1.3 Scope

The document comprises of our hardware and software requirements, environment, user interfaces, architectural design, project procedures, UML modelings (diagrams), development schedule and planned future works. The aim of the diagrams are to visualize the details of the project, and to show how components interact between themselves.

1.4 Overview

This IDR consists of ten parts. In first part we gave general information about **Building Server-Client Architecture to Play Card Games, BLÖFLÜ PİŞTİ, from Mobile Devices** to the readers, because it is written for developers we gave all the required information in this report and developers can read easily this report and start coding. In second part of this report we focus on **System Overview** which provide a general description of the software system including its functionality and matters related to the overall system and its design. In the third

part **Design Considerations,** we mention about special issues and then we give all information about **Data Design** which explain *Data Desctiption,* clarify how the information domain of our system is transformed into data structures and *Data Dictionary,* Alphabetically list the system entities or major data along with their types and descriptions. In the fifth part **System Architecture** which mention about architecture of the project. In the sixth section **User Interface Design** is the header and we will decsribe the functionality of the system from the user's perspective . It display screenshots showing the interface from the user's perspective and there is discussion of screen objects and actions associated with those objects. In the seventh part of our report the heading is **Detailed Design** which contains the internal details of each design entity/component. The last three parts are **Libraries and Tools, Time Planning and Conclusion.**

1.5 Definitions and Abbreviations

Define any important terms, acronyms, or abbreviations

IDR: Initial Design Report
AI bot/agent : Artificial intelligent game client bot/agent.
ACA: Android Client Application
GUI: Graphical User Interface

1.6 References

[1] IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications
[2] http://market.android.com
[3] http://itunes.apple.com/us/app/pisti-ii/
[4] http://creately.com
[5] http://wikipedia.org

2. System Overview

The product will be for the use of everybody eventually. System will be integrated with an online web site so user can log in the system via this web site and can play card games with multi-users environment system or via the computer. This game is played with 52 card deck

and among 2 people or 4 people. Our system is a product of artificial intelligence. System can compute the most powerful attack and think the sooner move. Our system, **Building Server-Client Architecture to Play Card Games, BLÖFLÜ PİŞTİ, from Mobile Devices** will be a part of a platform which accept members and these members can log in the system with their username and password then they reach their account and they see the games being played by other users and they make a room for a new game and the creator of a new room has the right of choosing the players which send an request. Creator can choose anyone of these requesters or from computer itself(Figure 2.1.1).

User can be anyone and enter the system with an id and a password. If the user enter the system first time it will be created an account to the user. The id must be a valid e-mail adres and the user must reply the mail sent by out web server to its mail adres.The id and password are kept in our databse and it will have to be matched next time when user can log in to system . Otherwise, if there is an error in matching, system will warn the user wrong id or password please try again or forget your password?

2.1 Goals

The main goal of our project is to create a Android Application for Playing Blöflü Pişti in Mobile Phones using Android Operating System and our system must respond to the user in the most intelligent way. Player can play against the AI player and our system must be created in a smart approach so that system must keep all the necessary information about pre-turns and store the information about opened cards. System must react to the user in a reasonably short time and take place if any user log out or the dropped down from the connection.

2.2 Objectives

Our objectives about AI to store information, is system can compute the most powerful attack and AI will win the game. System do not let user to play peer to peer and so if any of the user log out the system or left from the server somehow the AI will take place in turn of this player.

2.3 Benefits

This project is an entertainment project which is a type of card games, so the main purpose is to enjoy target people. Satisfying the requirements of Bloflu Pisti System will be enough condition to ensuring the entertainment level of game, because the Bloflu Pisti is a game that is ported from real life to digital media (software). Lastly, a slow reacting game cannot be enjoyable, then game software should be enough of quickness for running in an Android device. Our software will work very fast because user communicate with server instead of with each other. (likes distributed manner server architecture)

3. Design Considerations

Building Server-Client Architecture to Play Card Games, BLÖFLÜ PİŞTİ, from Mobile Devices project is a card game on ANDROID mobile devices. This system works on a server and answer client request .

3.1. Design Assumptions, Dependencies and Constraints

3.1.1. Design Assumptions

The target group for this software is assumed as everyone who needs to be entertained. Players of this game software can play either with AI or with other users matched from multiplayer server architecture. Player number is allowed for zero to two players, that is, multiplayer servers with zero human player also can play AI versus AI in an ongoing game. Players can communicate via our servers' network protocol. Player demands the information from the server and server responds to players' request. User can play with an AI Bot or play with others. Players communicate with our server and our system answer their request and system take place of any players who log out the system immediately.

3.1.2. Design Dependencies

The software is considered to run on the Android operating system connected to a web server. It is considered that the software will run on Android operating system on web server. Source codes of project will be implemented on ECLIPSE with Android Java Plug-In. In addition software will be running also on a web server.

3.1.3. Design Constraints

-Hardware Constraints

System will work an all new generation Android Mobile devices such as mobile phones with Hsdpa, touch tablets, hand PDAs and Android netbooks. Backward compatibility will probably be available up to Android 1.6

-Software Constraints

We will implement the project on ANDROID SDK Eclipse Platform and the database of project will be created with SQL, in addition web Server-Client System will be created on PHP. Therefore JAVA, SQL, PHP are mainstream infrastructures for developing our project.

3.2. Design Goals and Guidelines

Primary aim of the software is to make a well infrastructed software with modular software architecture. Server-Client architecture will be well organized with other components of project. We intend that server-client architecture run in a generic way, so that possible changes in design patterns will not highly affect the outline of the project. In addition to this expectations, our software will include some other properties to be a successful software. Performance is important and it will be provided within the project, because android devices have relatively less computation power when comparing with notebooks or desktop computers. In addition simplicity should be embedded in a sophisticated design, so that user

can easily manage the content of a complex design. Lastly server will be tweaked to refresh the users' move interrupts with a rate of half second, that is, client application will try to reach server with a frequency of 2 Hz.

In general, we have several design guidelines that we take into account:

1. Useful: Result of the project will be useful for everybody.

2. Fast: Software product should run with no delay time. Even every millisecond is important nowadays. AI agent will be quick enough to respond against human players' moves and in case of an connection interrupt from player side, an AI agent will take place with reasonable delay.

3. Simple: Keeping design and user interface as simple as possible

4. Engaging: Once a person uses that product, s/he should want to try again, so it must be a challenging task to do.

5. Universal: Design should be something that covers principles which have standards around the world. Design standarts should also be satisfied, and generic mainstream properties should fulfill globally accepted expectations.

6. Attractive: User interface or graphical components of design should be attractive to people, delight the eye without distracting the mind.

-Software System Attributes

Usability: The system should have easy user interface with minimal design. and minimum number of user interfaces. Any person with the knowledge of basic computer usage should be capable of using our system.

Documentation: The system should include a tutorial-like documentation which includes the information about how to play game and use the application.

Availability: The system should be available to user any time s/he wants to access with his/her username and password

Reliability: The system works independent from the users namely user communicate with server and server responds the user. If a user wants to make a move s/he sends the request to the server and then system do the action then responds quickly. There is tolerance of that any user leave the system immediately.

Security: System works in a secure atmosphere and it does not let intruders. Passwords of users are kept secure and sent with an encryption algorithm which is secure enough to use.

4. Data Design

5. System Architecture

A description of the program architecture is presented here.

5.1. Architectural Design

In our project, there are 3 major parts that are server, Android client application and the database.

Server manages:

- ✓ Communication between players
- ✓ Recording statistics into the database
- ✓ Login and logout operations
- ✓ Running the AI agent when necessary

Android Client Application:

- ✓ Provides the cute user interface
- ✓ Running the game
- ✓ Gets inputs from user via the interface

Database:

- ✓ Keeps users' all information
- ✓ Keeps statistics about each user for helping the AI agent
- ✓ Holds ranking tables

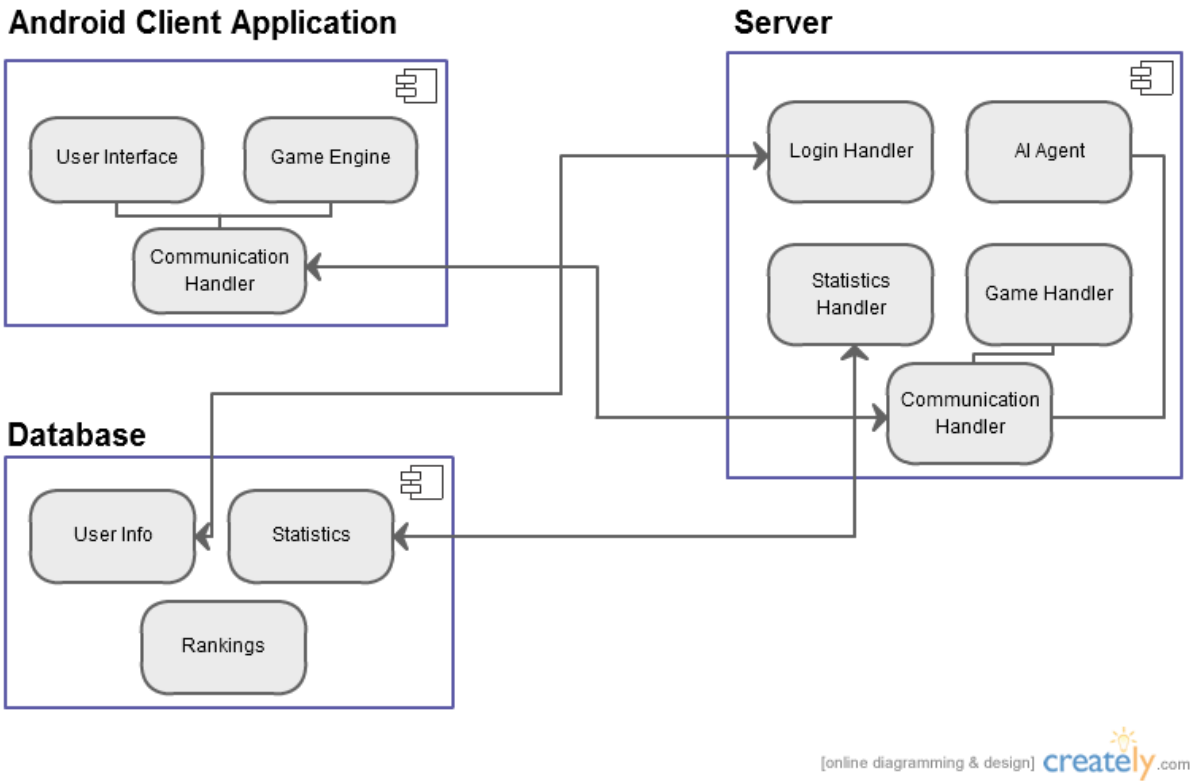


Figure 5.1.a: Component Diagram of the System

When a user opens the client application first the login menu [Figure 6.2.a] will be shown connects to the system, server will check user's username and password from the database. If authentication is done, main menu interface [Figure 6.2.b] will be shown to the user. The user can select one of three options which are "Play Online", "Play vs Computer" and "Settings. If the user selects the "Play Online" option, lobby menu [Figure 6.2.c] will be shown. In the lobby, existing game tables are shown, and the user can join one of them. In the "Play vs Computer" option, user can play with the AI agent. In the "Settings" option menu [Figure 6.2.d], user can change his/her information such as password, e-mail, or profile icon.

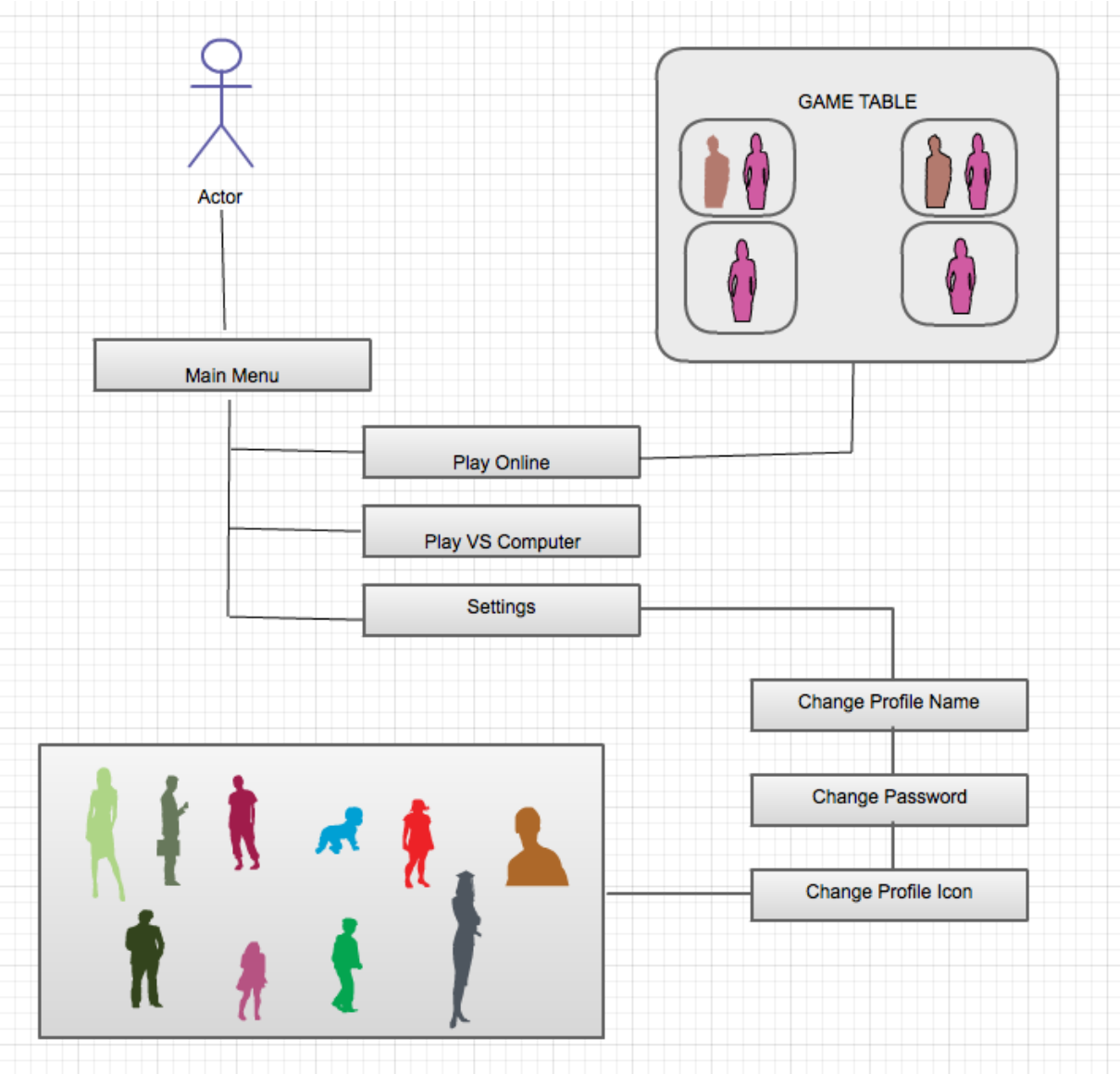


Figure 5.1.b: Use Case of the System

5.2. Description of Components

Employing the object oriented programming paradigm is the best way to ensure the reliability of structured programming techniques, so we are also using object oriented programming paradigm as well in our project. Componentwise structured architecture is very precise and well handled in object oriented systems, the system benefits from this paradigm also from this aspect, because every component should be in maximal consistency among themselves. Modularity is also provided at an extreme level. So it reduces the complexity of project by

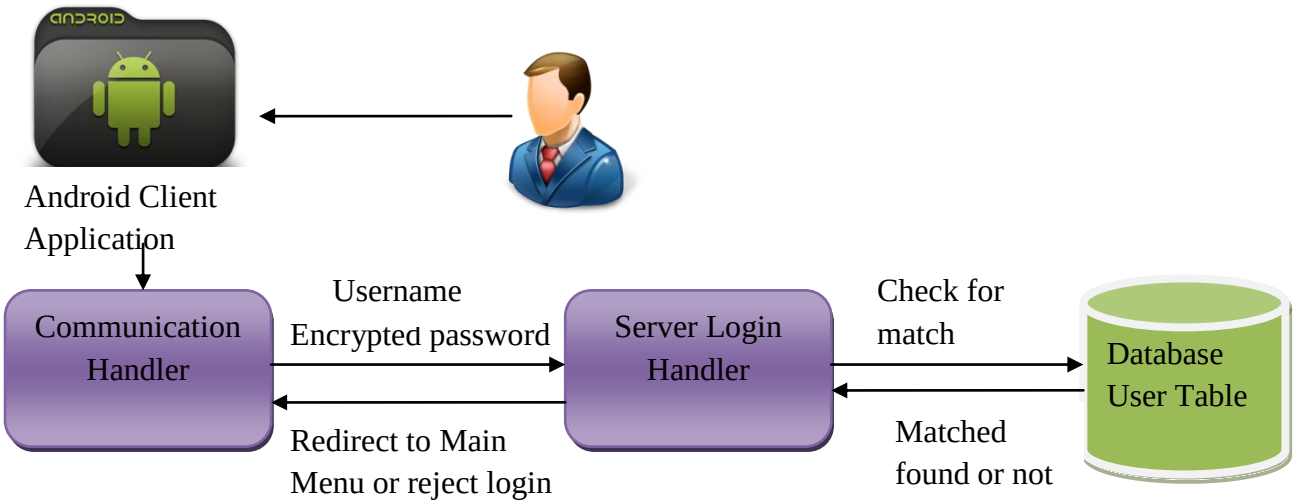
providing ‘part to whole’ approach.Well organized little working parts always increases the success rate of a software project, also this will be convenient to succeed

5.2.1. Server

The server that we used in our project is a PHP server. The reason why we choose PHP for server, is the fact that PHP is the most appropriate script language for server side. Server part has got five components as follows; Login Handler, Game Handler, Statistics Handler, Communication Handler, Artificial Intelligence Handler

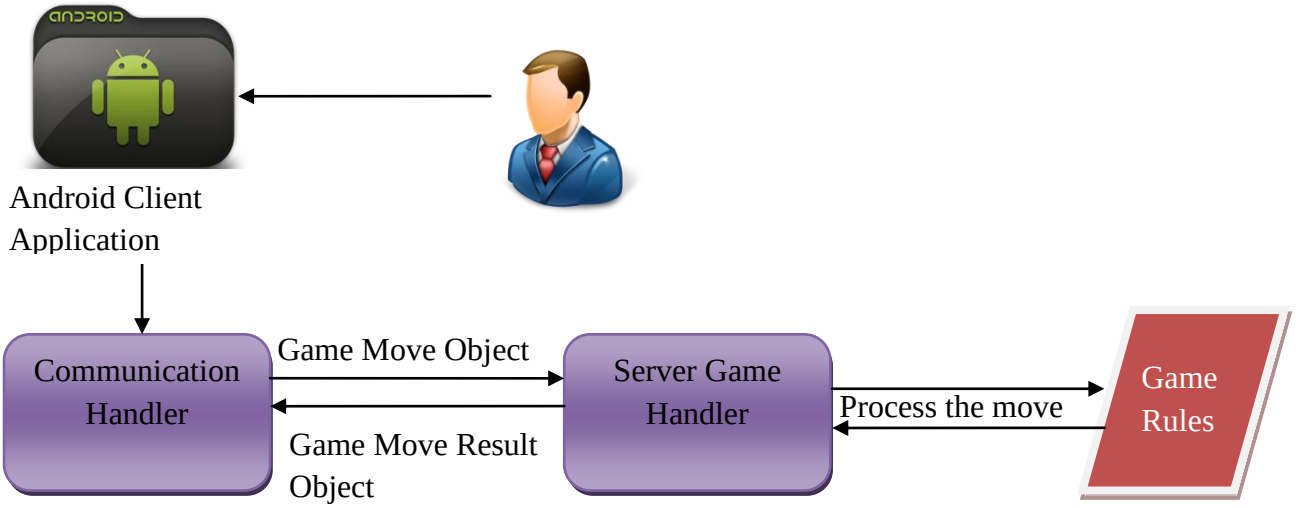
5.2.1.1. Login Handler

In this component of the server module, users’ login attempts will be conducted. Android Client Application will send the username and encrypted password information to the server and Login Handler in the server is about to decrypt the password. Then, username and password will be checked whether they are matched with any entry in the User database. In case of a mismatch, login attempt of user will be rejected. Encryption algorithm will take action when sending the password data online.



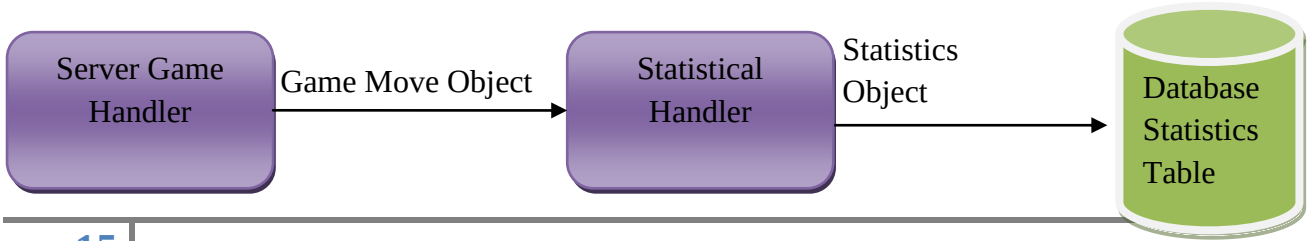
5.2.1.2. Game Handler

In this component of the server module, game rules, moves and other requirements of the game such as scoring, ranking, matching the player with game room, online gamer tables and in-game logs will be handled with this component. Drawing of the cards to players is also done in this component. Move logs of player in game is also recorded and managed within game handler.



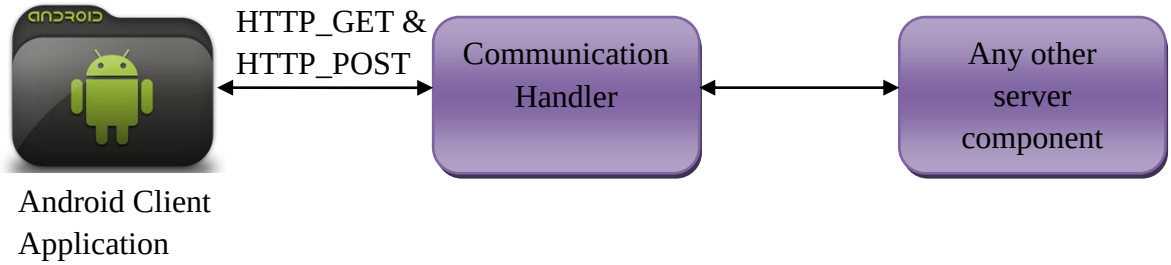
5.2.1.3. Statictics Handler

This module of server is to manage game move objects which comes from the game handler. Game handler sends the items demanded from the statistics handler. Statistics of a corresponding each registered user of game database will be managed and ordered from statistics handler. Move statistics, general rankings, inclinations of a specified gamer will also handled and transmitted to database via this module.



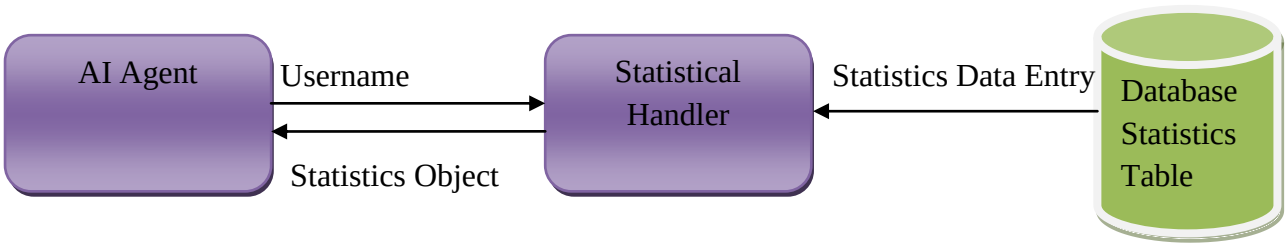
5.2.1.4. Communication Handler

This component handles all the data exchange between the Android Client Application and the server on HTTP port with the help of HTTP_GET and HTTP_POST methods. Game Move Objects will be sent within this handler to server, every data to be sent is transmitted by communication handler.



5.2.1.5. Artificial Intelligence Agent

In this component of the server module, computers countermoves against users moves are determined, game rules and statistics are blended to form a usable game data object. AI agents are administrated via this module. Statistics may affect the behavior of AI agents, for example whether a move is bluff or not will be changed after a series period of games. Learning ability will make the remarkable difference in playing styles. This means that AI agents can change its characteristics according to opponent player.



5.2.2. Android Client Application

This module of the project is client game application running on Android based systems mainly smart phones. ACA will have :

- Nice graphical user interface
- Communication handler that interact with server
- Game engine that runs the game in application

5.2.2.1. Graphical User Interface

Android has very high potential in user interface designs. We are going to use this opportunity in our project. Escpecially the Touch Screen is very useful for games. Drag and drop feature, dynamic menus are also very beneficial points of Android systems.

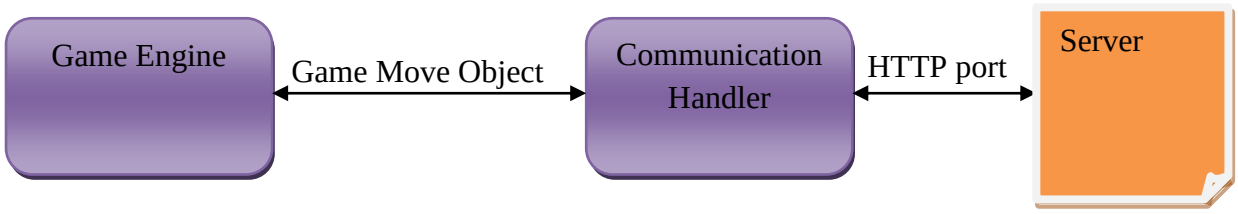
Our ACA has mainly 5 user interface window, we may provide some extra windows later.

- Login Menu window [Figure 6.1.a]
- Main Menu window [Figure 6.1.b]
- Lobby Menu [Figure 6.1.c]
- Settings Menu [Figure 6.1.d]
- In-Game Menu [Figure 6.1.e]

Details of these menus will be explained in 6.User Interface Design.

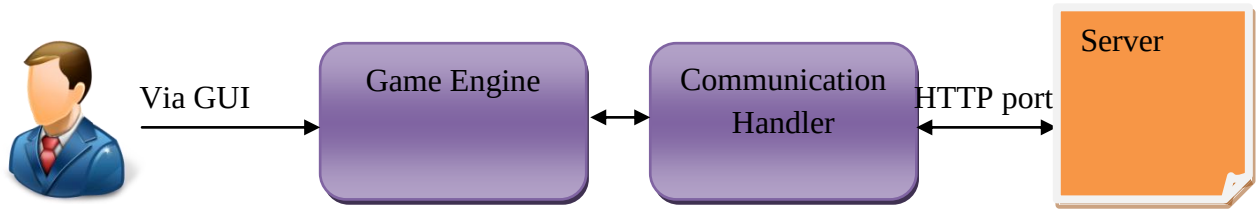
5.2.2.2. Communication Handler

This part of ACA is going to handle all the data interaction with the web server via HTTP port. Every data object that game engine produced will be delivered to the server via Communication Handler. There is no way to interact with outside world for ACA.



5.2.2.3. Game Engine

Game engine handles all the instructions comes from the user via the graphical user interface. Inputs from the gui is converted to input objects and sent to the game engine. Then game engine process the input and produce a proper output. Procuduced output is sent to communication handler and then to the server.



5.2.3. Database

In our project, we use MySql database and for managing database entries we use MySql Workbench 5.2.27.

5.2.3.1. User Info Table

User Table	**Column1(Username)**	**Column2(Password)**	**Column3(E-Mail)**	**Column4 Total Points)**
Row1	*R1 C1 Cuma*	*R1 C2 pass123*	*R1 C3 cuma@...*	*R1 C4 2100*
Row2	*R2 C1 Emirhan*	*R2 C2 pass124*	*R2 C3 emir@...*	*R2 C4 900*
Row3	*R3 C1 Hamza*	*R3 C2 pass135*	*R3 C3 hamza@...*	*R3 C4 2000*
Row4	*R4 C1 Utku*	*R4 C2 pass156*	*R4 C3 utku@...*	*R4 C4 0*

5.2.3.2. Statistics Table

Statistical Table	**Column1(Username)**	**Column2(# of bluffs made)**	**Column3(# of successive bluffs)**	**Column4 (win rate)**
Row1	*R1 C1 Cuma*	*R1 C2 25*	*R1 C3 13*	*R1 C4 %50.0*
Row2	*R2 C1 Emirhan*	*R2 C2 45*	*R2 C3 30*	*R2 C4 %80.5*
Row3	*R3 C1 Hamza*	*R3 C2 10*	*R3 C3 7*	*R3 C4 %81.3*
Row4	*R4 C1 Utku*	*R4 C2 3*	*R4 C3 0*	*R4 C4 %10.3*

5.2.3.3. Rankings Table

Rankings Table	**Column1(Username)**	**Column2 Total Points)**
Row1	*R1 C1 Cuma*	*R1 C4 2100*
Row2	*R2 C1 Emirhan*	*R2 C4 900*
Row3	*R3 C1 Hamza*	*R3 C4 2000*
Row4	*R4 C1 Utku*	*R4 C4 0*

5.3. Design Rationale

We have initially divided the whole system into two modules namely Client Application and Server. Then we realize that, AI Agent needs some database to improve its moves based on past game patterns. Now the modules available in the system are Client Application Server and the Database modules that they are able to work independently from each other.

5.4. Traceability of requirements

In the Software Requirements Specification report has some functional requirements that need to be satisfied in this design report. Those requirements are as follows and corresponding design solutions:

- ✓ Log in to the system : 6.1.a. Login Menu
- ✓ Arranging personal information : 6.1.d. Settings
- ✓ Opening a table : 6.1.c. Lobby
- ✓ Joining an opened game : 6.1.c. Lobby
- ✓ Playing the game or choosing an Al agent on the server : 6.1.b. Main Menu
- ✓ See Rankings : 6.1.e. In-Game Menu

We decided to delete Game History component from the system.

6. User Interface Design

In this section, we will describe user interface of our software project.

6.1. Overview of User Interface

There will be 5 main menus associated with our software project in the user interface. These are:

6.1.1. Login Menu:

The user interface of the system will first expect from the user to input his/her password by theprovided Login Menu. If the user has not had an account yet, he/she will be able to create anaccount by this menu also. If the user inputs wrong log in data the menu will warn the userinforming that wrong log in data is inputted. Unless the user input his/her existing account orcreate one, he/she will not be able to see any further information about the product.

6.1.2. Main Menu:

After creating an account or using an existing account, and logging in the main menu will be seen at the user interface. In this menu the user will be able toplay *blöflü pişti*and adjust his/her settings. In this menu, Play Online, Play vs CPU and Settings options will be displayed.

6.1.3. Lobby Menu:

In the Lobby Menu,the user will able to see rooms, user/s who joined them and their user icons. At this Menu, user can pass to In Game Menu by choosing a room.

6.1.4. Settings:

At the Settings Menu, the user will be able to see profile name, password and choose profile icon options and profile icons. Also, there will be a save button in this menu.

6.1.5. In Game Menu:

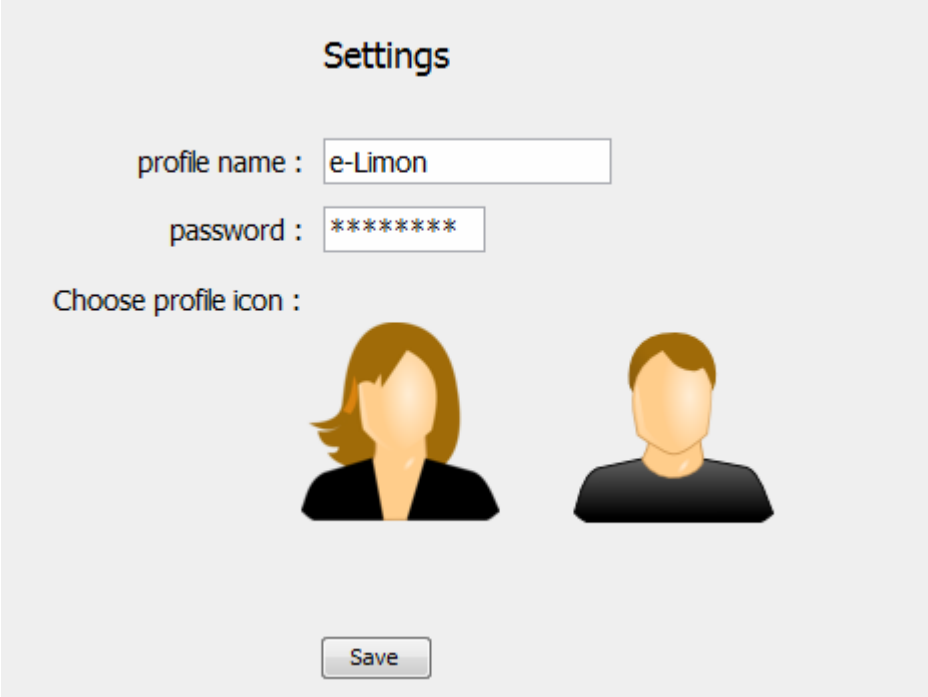
At the In Game Menu, the user will able to see his/her and the opponent's profile names and points. Also, game cards in the hand of user and in the table will be displayed at this menu. In addition, there will be an exit button.

6.2. Screen Images

Figure 6.2.a: Login Menu

Figure 6.2.b: Main Menu

Figure 6.2.c: Lobby Menu



6.2.d: Settings Menu

6.2.e: In Game Menu

6.3. Screen Objects and Actions

6.3.1. Login Menu

In the user interface, the user will be welcomed by a login menu to the system. In Login Menu, there will be Username, Password, Log In and Sign Up screen objects.

Username and Password: In this spaces, username and password fields will be handled when typed via the virtual keyboard of touch-sensitive device.Behind the background of password box, encryption module will take action, and password

string will not be shown openly, each typed character will be represented as asteriks symbol.

Log In: User will able to pass the Main Menu by this button.

Sign Up: User will able to create an account by this button.

6.3.2. Main Menu

In this menu, there will be Play Online, Play vs CPU and Settings screen objects.

Play Online: User can pass to Lobby Menu by choosing this option.

Play vs CPU: User can pass to In Game Menu by choosing this option.

Settings: User can pass to Settings Menu by chosing this option.

6.3.3. Lobby Menu

In this menu,game rooms will be seen.

Rooms: In this objects, there will be room name, user/s who joined them and their user icons. User can able to join a game by this objects.

6.3.4. Settings

In this menu,profile name, password andprofile iconfields will be seen.

Profile name: In this field, profile name can be typed.

Password: In this field, user can change his/her password and the password characters will be seen as asteriks symbol.

Choose profile icon: User can choose his/her profile icon by the user icons in this field.

Save: User can save his/her information by this button.

6.3.5. In Game Menu

In this menu, there will be profile names, points, game cards and exit objects.

Profile Name: Profile names of the both user will be seen by this item.

Point: Points of the both user will be seen by this item.

Game Cards: Game cards in the hand of user and in the table will be displayed by this items.

Exit: User can leave the room and pass to Lobby Menu by this button.

7. Libraries and Tools

7.1. Server Management Tool

Name of the Software: Xampp Control Panel Application

Version Number: 2.5

Mnemonic: 477 MB

Source: http://www.nat32.com/xampp/xampp25.zip

7.2. Android Development IDE

Name of the Software: Eclipse Java EE IDE for Web Developers

Version Number: Indigo Service Release 1

Mnemonic: 264 MB

Source: http://download.eclipse.org/webtools/downloads

7.3. Database Manager

Name of the Software: MySQL Workbench

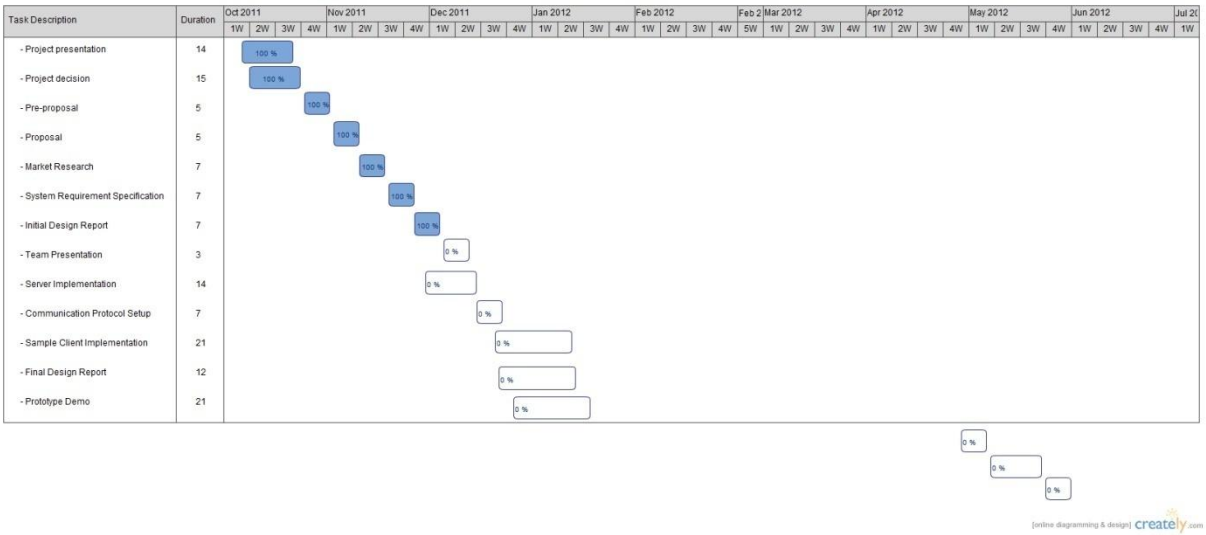
Version Number: 5.2.25

Mnemonic: 57 MB

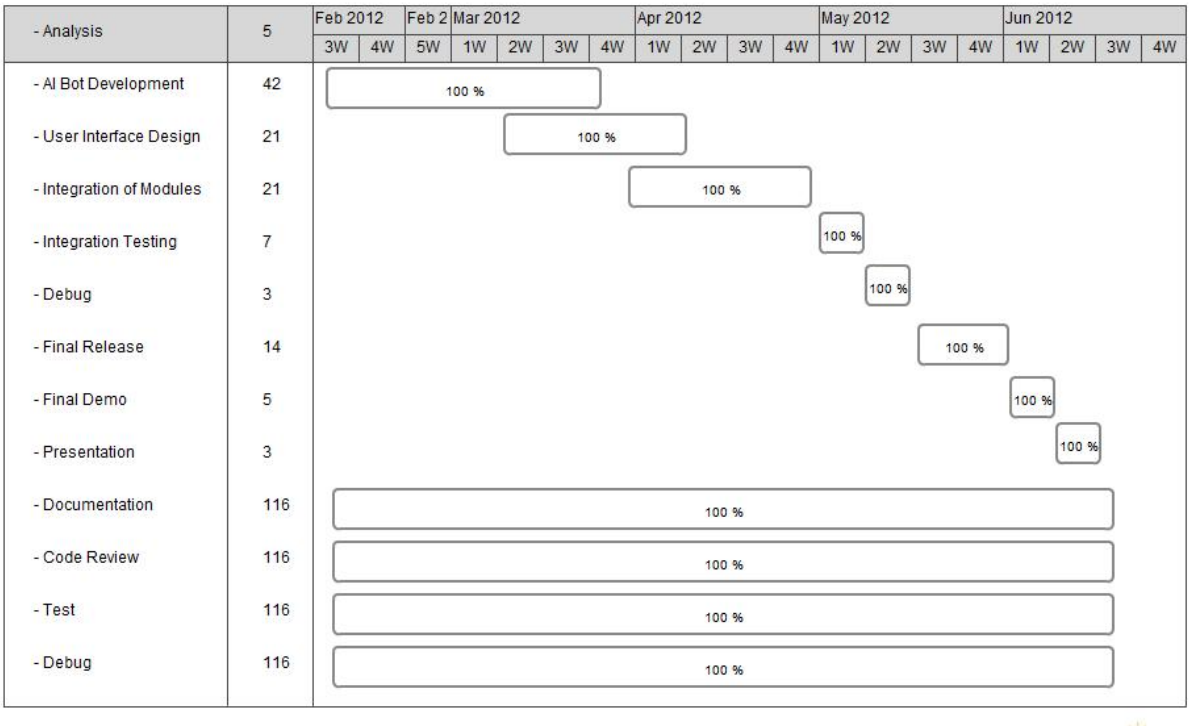
Source: http://www.mysql.com/product/workbench

8. Time Planning

8.1. Term 1



8.1. Term 2



9. Conclusion

This document is prepared to explain the detailed design of the project **Building Server-Client Architecture to Play Card Games, BLÖFLÜ PİŞTİ** supported from e-Limon.
At the earlier steps of the project design, it is implied that what our problem is and what its scope and purpose will be. After that components of the project have been researched. Since our design will be different from others that are in the market, it will be a completely new project, a pathfinder. It is mentioned that what our design should do, what features it will provide, which components it will include, what the user interface will look like and so on. After when we get an initial idea about project, we get into more detail about system architecture and data models. Each module of the system is described. Constraints of the project is discussed. Since this is an initial design document, more details will be provided later in the detailed design report.
Finally, as a plan for the future through the end of the semester, all aspects of the project would be agreed on. It will be a good practice and motivation to make a prototype of project for us when we start the implementation part in the second semester since it will include new topics for all of us such as artificial intelligence and server architecture.