Contents

[Introduction 2](#_Toc457727060)

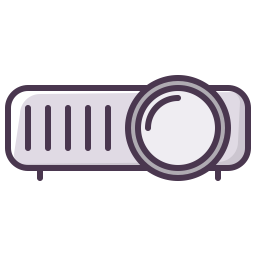
[Issues we faced during the project 3](#_Toc457727061)

[Differences between our implementation and the original paper 4](#_Toc457727062)

[Project in detail 5](#_Toc457727063)

# Introduction

# Project in detail



## Idea

All data save in a database which is in the server and by using the API on that server web client and then mobile app could sync their data frequently.

## Database

Because we used .net technology we preferred to use MSSQL Server 2014. The database contains four tables as you can see in the diagram:



#### Sentences:

All the sentences which we want to ask the users to enter should be saved here.

#### GameBoards:

Active and inactive boards are here and isStarted field contains three value null: waiting to start, true: playing and false is game is finished.

#### Shoots:

New orders to shoot are here. username is for shooter ,TargetUserName is the target and CER is the angel of shoot.

#### AjdarUsers:

User information is here. They have Health, Weapon, Error and status to show is user alive or not.

## Web API

We used Asp.net MVC to build this API. Input should be send by GET method from http protocol and output are JSON. All the processes are happening here which we are describe them below.

GetBoards(): Return a list of boards which are waiting to start.

MakeBoard(name,adminName): Name is the board name and adminName is the user name of board creator.

GetUsers(boardId): Return a list of users which belong to the board with this boardId.

JoinUser.

RequestToStartGame(boardId): Make the status of board to ready for start for the first time and for the second time make the status of board is started. Because we have to step for starting the game first is with admin app when all users join to the game then he should start the game from app. Then someone in web browser should find the board name in list and start it as well.

IsGameStarted(boardId): Return the board status. Using in web browser.

GetNewSentence(): Return a new sentence by random.

Shoot(boardId, userName, cer): Select a target user by random and then save a new shoot in database.

GetNewShoots(boardId): Return all new shoots.

Shooted(ID): Find a shoot with this Id in database and remove it.

UpdateUsersStatus(jsonUsers,boardId): In jsonUsers we have a JSON format of all users with their new status then we update all user status in database with this new status.

GetUserStatus(name,boardId): Return an object of this user.

## Web Page

There are two pages for this part based on HTML and JQuery. First page is showing board names which are just created and has not started yet. After selecting

A simple HTML page which is showing the game using HTML5 canvas objects and communicating with server using JQuery.

## Mobile APP

## How different parts communicate

# Issues we faced during the project

# Differences between our implementation and the original paper