Design Document

for

Pocket Tanks

Prepared by Rahul Nirania and Abhishek Yadav

April 15, 2018

Contents

1 Introduction

1.1 Abstract

The intended goal of Design document is to give brief description about what we are going to implement how we are going to implement,risk Analysis and Test plans.We are to build an online game named as **Pocket Tanks**.User can refer to SRS to know more about our game mechanics,User interface and game story.

1.2 Pocket Tanks

1.3 Developement Strategy

For simplicity we dividing this project into three phases as:
1. Phase-1: to develop an offline game.
2. Phase-2: host offline game on server.
3. Phase-3: inbuild chat functionality.

We are going to set deadline of various phases.Testing and risk management will be done in parralel with the development phase.

1.4 Prerequisite

(1)Javascript
(2)AngularJS
(3)CreateJS
(4)HTML and CSS
(5)NodeJS
(6)TCP Server
(7)Firebase(as database)
(8)socket.IO

2 Phase-1:Offline Game

2.1 Overview

In this phase we are going to describe how we are going to develop an offline version of Pocket Tanks.We are further dividing this phase in various stages.These stages will include building of weapons,Tanks,dynamic generated terrain.We will be then displaying score of players along with player name on top such that they get update according to rules of game.At bottom of screen we will be displaying weapons,various parameter of tanks and moves according to degree of freedom of tank.Two webpage will be created,on first webpage user will be provided with various options and after selecting one of them a new webpage.A new webpage displayed on screen will consist of various weapon,among which user will be allowed to pick five among them.This will lead user to play mode in which user is free to resume,pause or exit.

2.2 Application of AngularJS and CreateJS

2.2.1 Weapons

We are going to implement only 10 weapons.CreateJS and AngularJS will be used to design these weapons.Further description about these weapons will be given in ReadMe.md.User will be allowed to fire any of the selected wealpons.

2.2.2 Tanks

Tanks will be randomly generated on terrain while the user will be free to move the tank,change its inclination between 0-180.User can also control the power with which various weapons can be fired on opponent.We will be using creteJs to design tank.

2.2.3 Terrain

Terrain will be randomly generated with help of javascript framework.This randomly generated terrain will remain same during game play while firing of weapons may change shape,size of terrain.User will be allowed to move this tank freely on this terrain.But due to some constraint over degree of freedom of tank it will be not allowed moved freely in every case.

2.2.4 User Score and Tanks Attribute

After every successful firing of weapon on its opponent players score will be upgraded according rules of game.Tank attribute as described above will be displayed at botoom of screen.This functionality will be implemented using AngularJS.

2.2.5 Play Mode

When user will first load game a screen will be displayed asking user to select playing mode whether user wants to play in single player mode,dual player mode or wants to play on local server asking third party to join game on that server.

2.3 Single Player Mode

As user selects single player mode,user will be allowed to select difficulty level.There are three difficulty level i.e Easy,Medium,Hard.In this mode user will be playing against CPU or a bot.This bot will be behaving diffrently in every level.When user select hard level bot will take intelligent decision.Degrading difficulty level will limit bot to take intelligent decision.We will be coding these decision that bot will take during various game play in javascript.

2.4 Dual Player Mode

As user select dual mode a screen will be displayed asking user to set their name.Further the two user will be allowed to select weapons for their game play.After selecting weapons a new screen will be rendered.Now the two players are free to fire any weapon on one another.At the end the user that will score more points will be winner,detailed description of game rules is given in SRS.Uptill this part we will be using javascript,angularJs and cresteJs.User will be provided enough time to make strategy against opponent and fire weapon accordingly.Every weapon produces unique sound when fired,we will be implementing this feature using SoundJS.Finally when particular user wins,then over tank of that user "winner" will be displayed and particular sound will be produced.

2.5 Local Server Mode

This part is targeted to be completed in phase-2.Detailed description will be given on this part in upcoming chapters.

3 Phase-2:Server Side

3.1 Overview

Server will be created by one of the user that listens for players wanting to join that game.This will be implemented using NodeJS and database will be maintained using firebase.User authentication will be done using email-Id and password.But at first time user have to create an account using email-Id and unique password will be selected by user only.Thus authentication will be helpful in tracking history.

3.2 Firebase

Thus directly we are going to use firebase for authentication purpose.We had not thought much about the implementation but we are having only high level idea.User can be authenticated with email-Id/password,either facebook or with google login.

3.3 Application of NodeJS

We can create server with help of nodeJS,the server created will listen to client's requests similar to Apache HTTP Server. The server created can read http request made by the client which can be a browser or a console and return the response.We had this much Knowledge only about nodeJS.

3.4 Transmission Control Protocol

Protocol are set of instruction set by international body which are followed while communicating with each other.TCP helps to exchange packet of information biderctionally once the connection is established.Primarily, TCP ensures end-to-end delivery of data between distinct nodes.This flow of information will be till they are connected.

3.5 Risk Analysis

There can be some case in which weapon fired by one user will appear after some time lag to other,this will create a bad user experience.Thus game should be synchronous i.e. "like you are facing each other across a board in real life".

4 Phase-3:Chat System

4.1 Text Chat

We will be using socket.io to enable text chat functionality.

4.2 Video Chat

We had not thought about this till now but if time permits we will add this functionality too.

5 Refrences

1. For CreateJS https://createjs.com/getting-started.
2. For AngularJS https://angularjs.org/.
3. For NodeJs https://nodejs.org/en/.
4. For firebase authentication https://howtofirebase.com/firebase-authentication-for-web-d58aac
5. For text chat functionality https://socket.io/get-started/chat/.