<Game project name>

<Game catch phrase>

Technical Document   
(Homework No.3)

Project team: <Team Name>

Instructor: Dr. Araz Yusubov

Submitted in partial fulfillment of the requirements of the CSCI 4836: Game Development Fundamentals course project

|  |  |
| --- | --- |
| Version date | Version information |
| <Date> | Initial draft |
| <Date> | <Version description> |

|  |  |
| --- | --- |
| Other documents in the package | |
| File name | Brief description of the document |
| <File name> | <Description of the document> |
| <File name> | <Description of the document> |

|  |  |  |
| --- | --- | --- |
| Team member | Contribution to this homework (NOT the project) | Estimated % |
| <Student Name 1> | <Description of the work contributed> | <X>% |
| <Student Name 2> |  |  |
| <Student Name 3> |  |  |
| <Student Name 4> |  |  |

# Table of Contents

<Automatically generate here using Microsoft® Word menu References🡪Table of Contents>

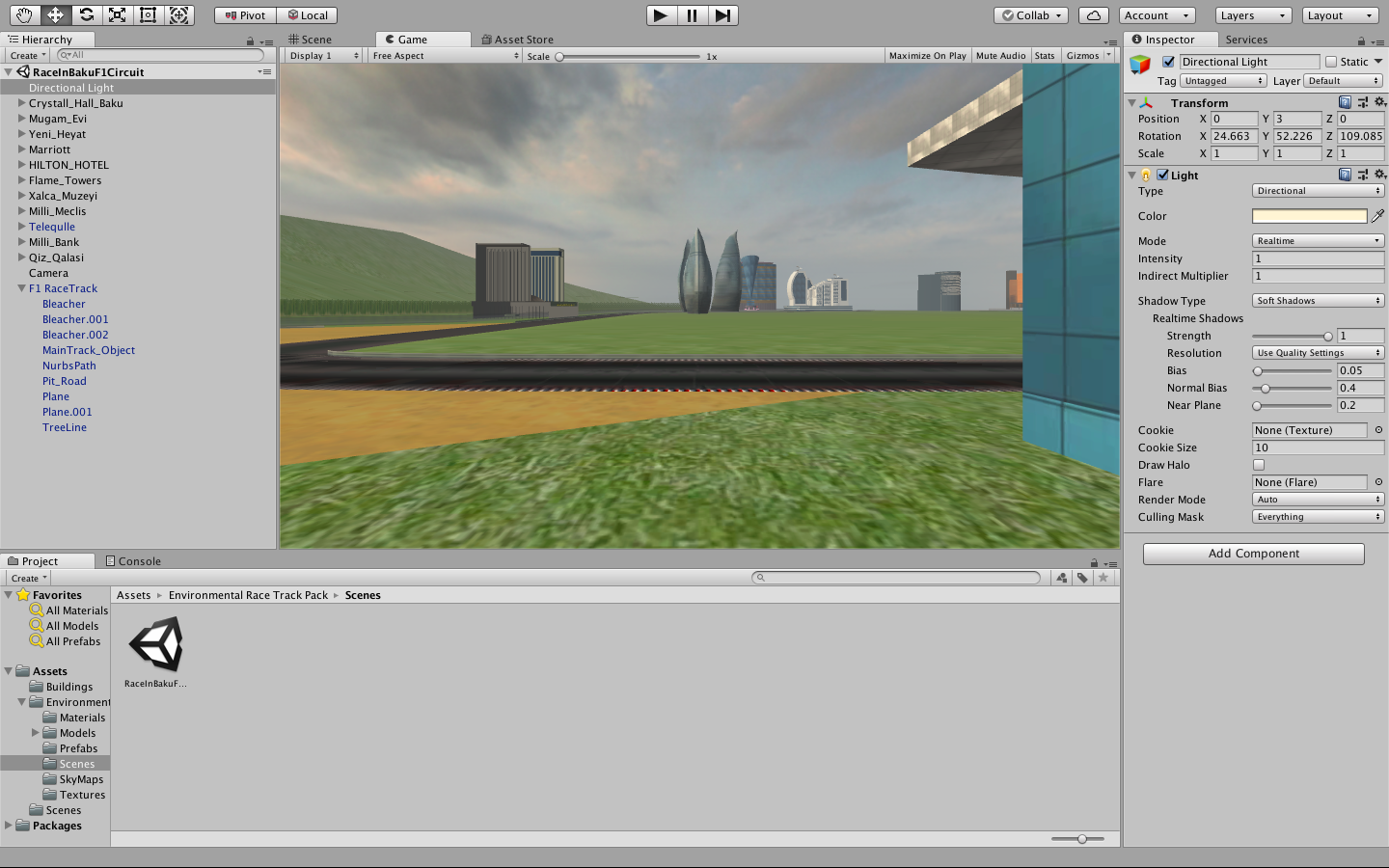
# Introduction

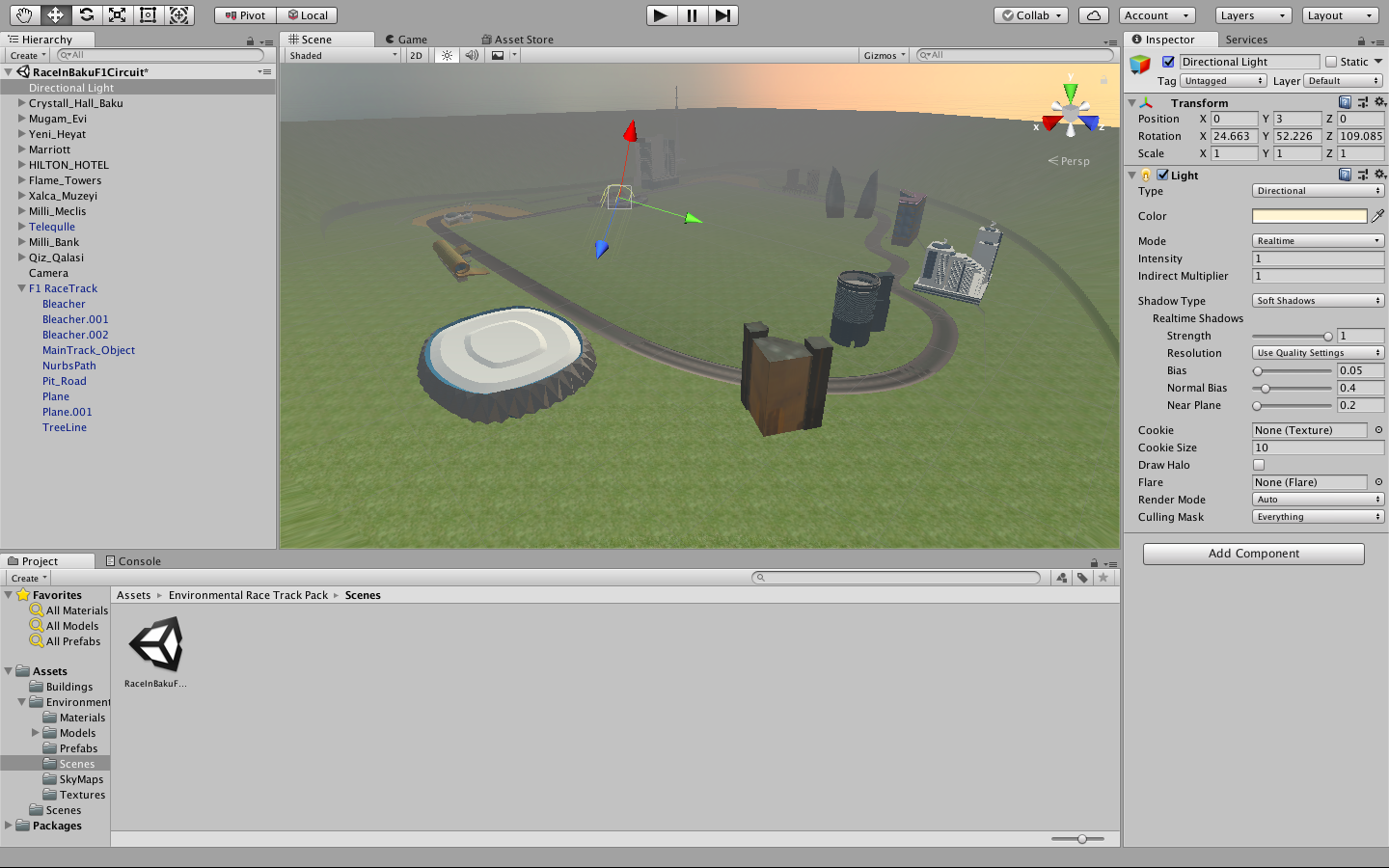
This is part of the Game Design Document for a hypothetical project Race in Baku F1 Circuit

submitted for partial fulfillment of the requirements of the Game Development Fundamentals course in the School of Information Technologies and Engineering at ADA University, Baku, Azerbaijan.

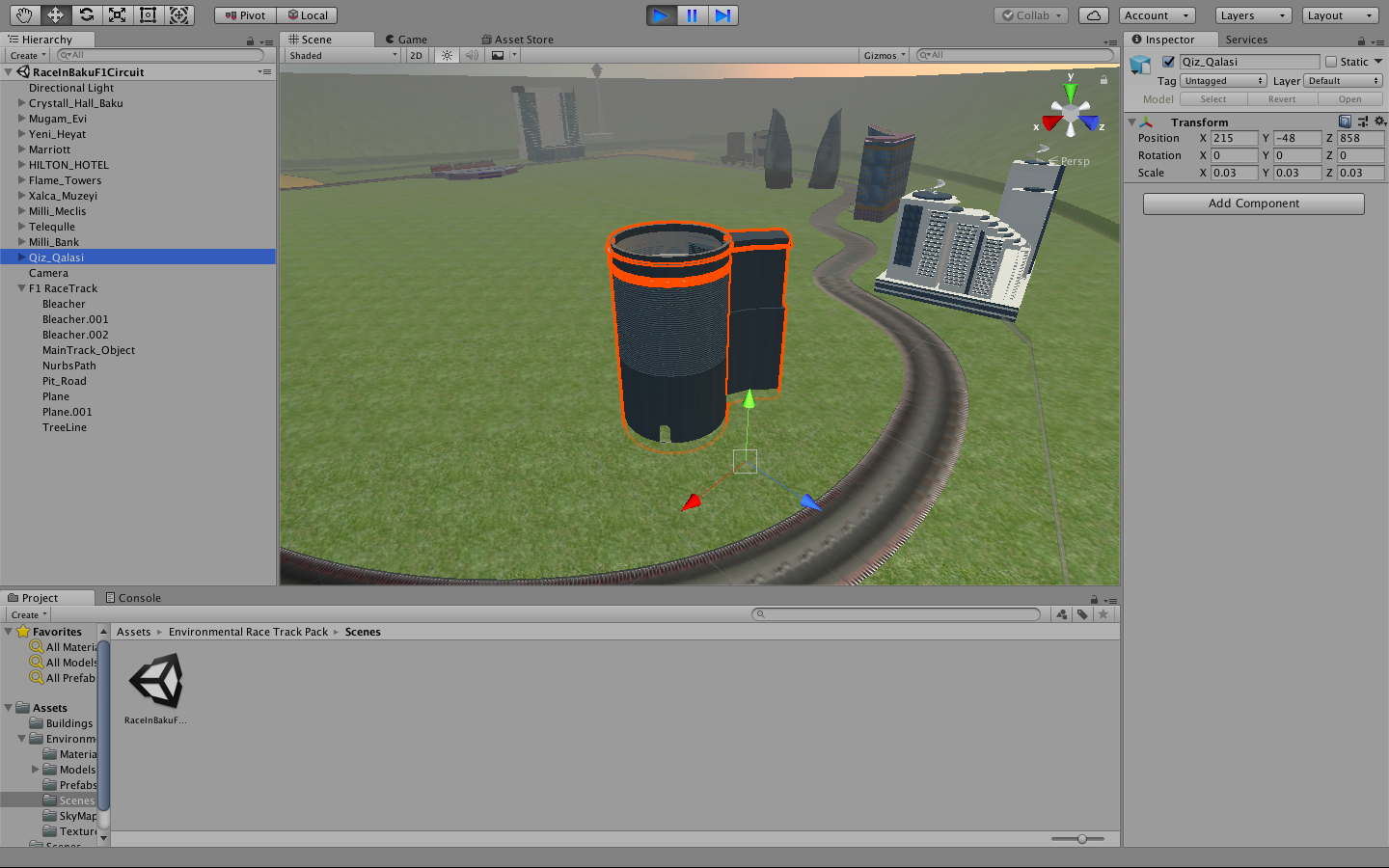
 The game which is planned to be made by GameOPS team is “Race in Baku F1”. This game is single player, competitive, non-violent, extremely exciting and take place in Baku, Azerbaijan. Regarding to genre of the game, sport is one of the most popular and loved game types in the history and therefore we are choosing to create game with the same genre.  The mode of the game will be single player, and users will be enabled to play with computer players in order to get an memorable experience. The game is given a cityscape look to give a player more realistic impression. The territory will be based on three-dimensional model of real-world location of the circuit, which is in Baku. In addition, our three-D models include central streets of Baku, modern buildings and old towers of the city. Players will feel excited in Baku F1 Circuit and try to compete against opponents until the finish line. The theme of our game is to compete with the other opponents that are controlled by computer in a racing tournament, the player’s goal is to reach to the destination as soon as possible while trying to avoid bumping to other cars or road objects, which slows down speed of the car.Lastly, it worth mentioning that the genre of the game is somewhat cloned, however the game is in Baku street which makes it unique.

Furthermore, our team has created 3D models of Baku  where race will happen . Players will feel stuck in curved road of city and it makes game more interesting and entertaining. We are trying to make our model more realistic and similar to real race happened in our city.  The underlying motive for us to locate national architecture of Azerbaijan on the street make people who live in abroad feel nostalgic while playing game. Below, 3D models of the city shall be demonstrated:





Development and improvement process of model still continue. As it can be seen below model is combination of modern and national architecture of Azerbaijan.



Main menu will be displayed when player start game , which consist of Start Game, Settings, Controls, Quit Game, Story. Example game menu will be same as below mentioned. When player click the start game each time, player can choose three levels of game and unlocked cars to start new game. On settings mode, Player can modify audio, video, display on this screen. Further improvements shall be added as project growth.



Synopsis: Synopsis of game is players have to win every race in order to earn cash and points to unlock cars and levels, «A RACE ISN’T WON UNTIL IT’S OVER”.  In the game, options are provided to players to choose a car to start a race.

Game Play: This section will give brief information about how the player will play the game. The player control one of the cars to compete with other opponent computer players. In the beginning of the game, there will be a little bit easy levels but when  players pass levels, it will not be simple to finish the levels. The player uses keyboard to manage the car to turn left or right, move back and forward, to hit a break or accelerate.

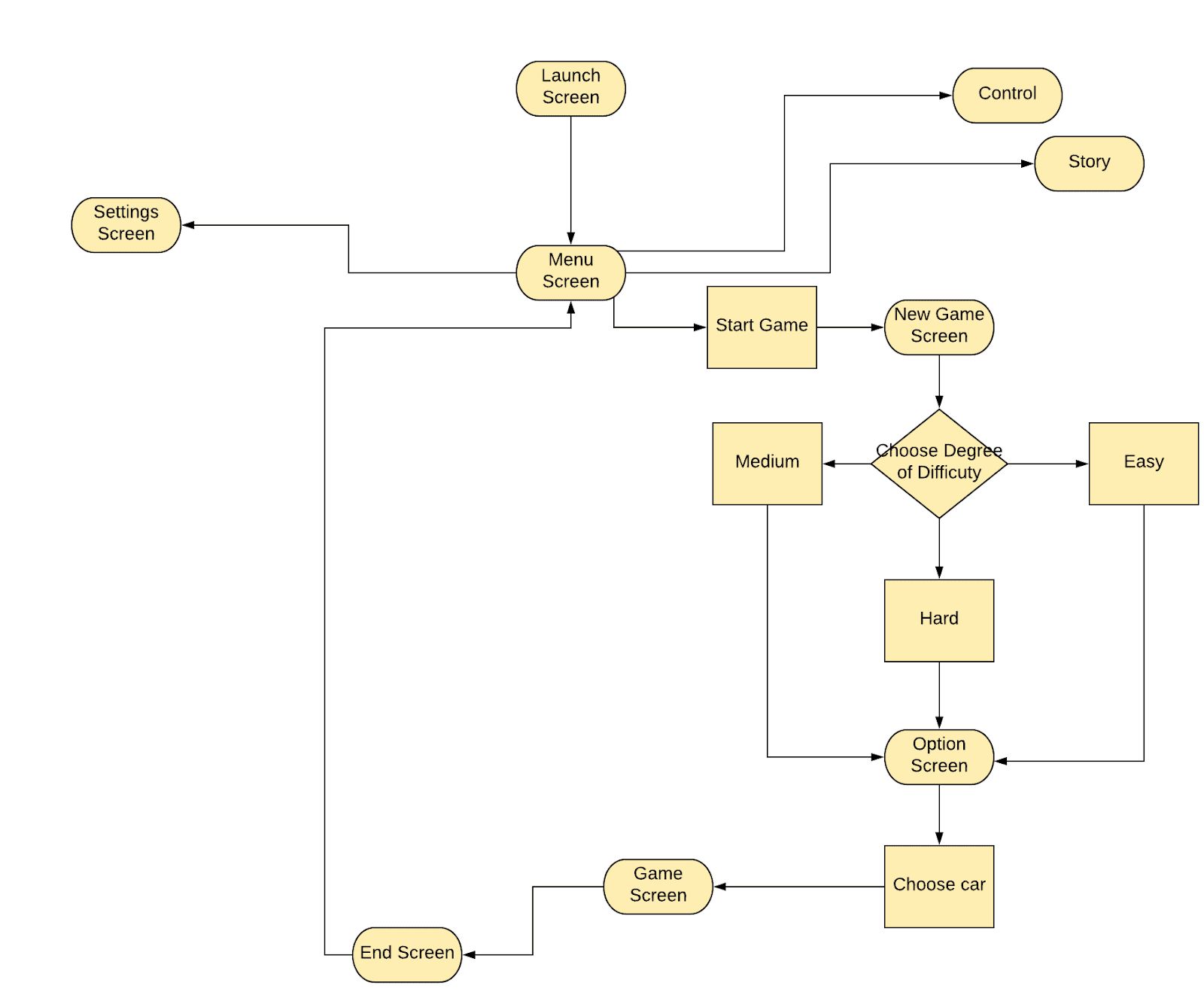
Player Control: The game will be played in computers and used keywords are W, S, R and moving symbols. W, S and R buttons are used for acceleration, brake and pause respectively. Moving symbols are used for moving to forward, backward, left and right.

Game Over (Winning & Losing): In order to win the game, player have to be fast against to opponents. As players pass levels, they will earn point and money. Points will open levels and money will be used to buy unlocked cars. When player reach last level, all cars will be unlocked  and the game will finish.

**Game Matrix**

|  |  |
| --- | --- |
| **Player/Antagonistic elements** | **Game properties** |
| AI drivers | Opponents which will be controlled automatically by computer against to player. |
| Barriers | Surrounding the track and all players should avoid hitting the barriers . |

**Game Flow-Chart**



## System Requirements

When user purchases or install  game they have to make sure their computer support system requirements of game.Our game going to be PC oriented. Below are system requirements for game:

Minimum Requirements:

CPU:Intel i3-4170 @ 3.7Ghz / AMD FX 6300 or equivalent

RAM: 4GB RAM

HDD: 15 GB HD space

GPU:NVidia  GeForce GTX 650TI / GT 740 or AMD  Radeon R7 250x

DirectX: Version 11.0

Sound Card: DirectX Compatible

Operating System: Windows 7 64-Bit or later

Recommended Requirements:

CPU: Intel i7-6700k@ 4.0 GHz/AMD FX 9590  or equivalent

RAM: 16GB RAM

HDD: 25 GB HD space

GPU: NVidia GeForce  GTX 1080 or AMD Radeon RX 480

DirectX: Version 11.0

Sound Card: DirectX Compatible

Operating System: Windows 10 64 Bit

## Concerns and Alternatives

<If there are concerns about something technical, they should be stated here, along with any alternatives to the concern.>

## Resources

<The resources section lists applications and equipment that are acceptable for use in the development of this game. This begins to satisfy a legal challenge that developers must begin to be aware of.>

# Visual Content

<This is a section that lists technical requirements from those in concerned with the visual aspects of the game. All objects should be listed with their generic names.

* General
  + File Size Restrictions
  + File Format Type
  + File Quality Type
  + Visual Scale
* Player Elements
  + Type of States (Default, Damage, Destroyed, ect.)
  + Amount Animation Frames
* Heads Up Display (HUD)
  + Type Icons
  + States
  + Font Type
* Antagonistic Elements
  + Type of States (Default, Damage, Destroyed, ect.)
  + Amount Animation Frames
* Global Elements
  + Background/Texture/Tiles
  + Font Type

>

# Audio Content

<This is the section for organizing the audio content. It is very important to communicate with the audio designer before and while the audio content is being developed.

* General
  + File Size Restrictions
  + File Format Type
  + File Quality Type
* Player Elements
  + Type of Sound f/x
  + Device Vibration
* Antagonistic Elements
  + Type of Sound f/x
  + Device Vibration
* Global Elements
  + Ambient Music
* Splash Screens
  + Ambient Music
* Menus
  + Type of Sound f/x

>

# Programming Content

< The programming content section should help permit good collaboration with the programmer. The objective of this section (and task) is to try to organize and modulate as much as possible.

* General
  + Requirements
  + File Size Restrictions
  + File Format Type
  + Specify Coding Conventions
  + Language/Device Restrictions
  + Screen Type (Small, Medium, Large)
* Player Elements
  + Type of Event
* Antagonistic Elements
  + Type of Event
* Global Elements
  + Type of Event
* Splash Screens
  + Type of Event
* Menus
  + Type of Event
  + Type of Options

>

## Code Structure

<This is where an overview of how objects/functions/data interact, a list of what specified functions/routines do and a list of what order modules will be written.

**Bonus:** Get extra **10%** points for including Interaction Matrices[[1]](#footnote-2) and/or Class Diagrams.>

# References

<Insert here any document referred to in the document. An example might be articles or Web sites that you consulted during the literature search. This is not just a list of used materials, so do not forget to clearly MARK the exact points(s) of reference in the main text.>

1. Interaction matrix is a spreadsheet listing game objects on sides, and interactions that can occur between them during the game at intersections of rows and columns. [↑](#footnote-ref-2)