Non-commercial project

CODENAME "OUR CITY"

Game Design Document

Table of Contents

SECTION I – GAME OVERVIEW

CONCEPT

THE BASIC IDEA

Our City (working title) is an arcade driving simulator in the open world with RPG elements. The whole city is located on the island. The game is inspired by GTA and Driver games concepts. The game will not have a multiplayer mode, only a singleplayer mode will be implemented. A player can only move around the city in his vehicle and will be completing a set of missions, which are connected to the story line. A player will collect game's currencies, to buy add-ons for his vehicle and other items. After all of the missions will be completed, the player will leave the city and the game will end.

PROCESSING

The Game will be processed in 3D; camera view will be fixed with a mouse controlled rotations. The island will be created as a set of squares, with building filled in necessary places. Around the island there will be water, which is going to work as a border for possible vehicle's positions.

THE ISLAND

Example of the Island's Plan

The player will have a free driving mode around the city, and he will contact with AI controlled vehicles. Most of AI controlled vehicles will respect the driving rules. Water around the island will be used as a border for allowable driving routes. The bridge that connects the island with the main land will be long enough, to disable possibility of crossing the water to the nearest land.

THE VEHICLE

The player will have his main vehicle that could be upgraded to more advanced versions. For some missions different vehicles will be used.

USER INTERFACE

MAIN MENU

Main menu consist of design elements and basic modes, including Singleplayer, Options and Exit.

HUD

In the bottom-left corner is the mini-sized map of surroundings, in the top-left corner is a minimized version of the communicating device, which can be rolled out. In the top-middle of the screen is a timer that usually appears in the missions. In the top-right corner of the screen is the oxygen bar, which is follow by notification shortcut. In the bottom-left corner is speedometer and tachometer.

GAME MODES

There will be two main game modes, including free driving mode, where the player can drive around the city. And a mission mode, where he has to do actions according to each mission. Missions will be implemented as checkpoints, races, etc.

SECTION II – GAMEPLAY AND MECHANICS

GAMEPLAY

GAME PROGRESSION

After the first cinematic trailer, where player will get basic ideas about Back Story and the World itself the training mission will start, where the player will get basic control skills. After that the free driving mode is setup, and the first mission will be available in some time. After the first mission is completed, free driving mode is setup again. This cycle will repeat enough times to satisfy the story line, and requirements for game ending, including completing all of the missions and updating main vehicle to the maximum level. Game ends with a cinematic trailer and credits.

MISSIONS STRUCTURE

Missions are accessible through the in-game menu (a HUD's component), and are available according to the Story Line. Missions are limited by the time and have checkpoints around the city, which should be passed. Some missions will have AI opponents.

MECHANICS

PHYSICS

The Game Physics is simulating real world physics. Most objects of the game are having attributes, such as position, weight, speed vector (which is representing the force). And according to them gravitation and speed is calculated. Each object has its own representing bound-box, according to which collisions are calculated, collisions are affecting speed vector, which affects position of the object.

MOVEMENT

There are 2 types of objects. Movable objects are representing vehicles mostly. These objects have speed vectors. Immovable objects don't have speed vectors and weight. Gravitation also has no need to be implemented. It's obvious, because a variable that represents immovable object's position is for reading only (can't be changed during the game execution).

OBJECTS

IMMOVABLE OBJECTS

Immovable objects are representing buildings, trees and other attributes of the world that normally don't have ability to move.

MOVING OBJECTS

There are vehicles (cars, trucks, buses, etc.), that are controlled by AI. There are boats with Water Watchers across the island. Their objective is to prevent citizens to swim out of the island. There are tanks or other military vehicles with Bridge Watches on the bridge. Their objective is to control in and out going movements of the city. Mostly they disable citizens to cross the bridge. And finally, there is a car, that's controlled by protagonist. Most of the game he is using this car to move across the island, but in some special missions he is driving other vehicles, such as ambulances, military, etc.

PICKING UP OBJECTS

Most common picking up object is oxygen, which could be loaded in car on oxygen station, but there exists a black market for trading oxygen as well. Also there are mission-related picking up objects.

ACTIONS

SWITCHES AND BUTTONS

W	-	Accelerate
S	-	Break / accelerate backwards
A	-	Turn left
D	-	Turn right
Space	-	Hand break
Escape	-	End game / Show main menu
Q	-	Show/Hide in-game menu
E	-	Action / Yes
M	-	Enlarge / Minimize map
Mouse	-	Controls camera views / Controls in-game menu (when it's showed)

Main menu is controlled by both keyboard arrows and mouse.

PICKING UP, CARRYING AND DROPPING

Oxygen could be picked up at the oxygen station, and is used as a resource for breathing. The amount is decreased each second and is limited by both game mode (easy, normal, hell) and car's maximum carrying amount, which can be upgraded during the game.

TALKING

There are very few possibilities of talking back to the game. All of them are optional and use mouse as input. There are incoming calls to player's talking device, which tell him about current available missions.

READING

Mission objectives are available as SMS/MMS/Maps at the communicating device, which is a part of the in-game menu.

ECONOMY

The game economics is built around 3 basic resources: money, tickets and plain oxygen itself. Car upgrades can be made for normal money, because they're shipped from outside world. Tickets are the main in-island currency, and are used for transactions between citizens of the island mostly. In order to load oxygen at the oxygen station, you need to use a ticket. If you don't have a ticket, you can load oxygen only at your home location. Plain oxygen is the resource at the black market and is traded for normal money; tickets are also traded at the black market.

SCREEN FLOW

SCREEN FLOW CHART

In the middle of the screen is a model of the car, the player is controlling. In the bottom-left corner is the mini-sized map of surroundings, in the top-left corner is a minimized version of the communicating device, which can be rolled out. In the top-middle of the screen is a timer that usually appears in the missions. In the top-right corner of the screen is the oxygen bar, which is follow by notification shortcut. In the bottom-left corner is speedometer and tachometer.

Mini-sized map is used for basic orientation; communication device is used for collecting information about available currencies, missions, objectives, etc. The timer shows up left time for missions, etc. Oxygen bar represents available oxygen, and turns red, when the level of oxygen is critical. Notification shortcut appears when there is an incoming call or message. Speedometer and tachometer represent speed and RPM (revolutions per minute) of the player's car.

SCREEN DESCRIPTIONS

MAIN MENU SCREEN

Main menu consists of design elements and basic modes, including Singleplayer, Options and Exit.

OPTIONS SCREEN

Options screen consists of few adjustable elements that will affect the game controls, graphics and sounds.

GAME OPTIONS

Game controls will change controlling of the game. There are 3 difficulty options - easy, hard and hell. Difficulty options can only be changed when the new game is created. They affect how fast the oxygen is used in car, how many tickets you get for each played hour and how much time you are getting for completing missions. Graphics options can be used to change the resolution. And Sound Options can change the volume of different parts of sound effects.

REPLAYING AND SAVING

Singleplayer mode will have 2 options: continue and new game. Continue will load the game to the latest saved point, on this account. New game will erase this data, and start the game from the beginning of the story. Saving will be available at the oxygen stations and home location. There is and automated saving after each mission is completed.

CHEATS

Cheats are not implemented yet.

SECTION III – STORY, SETTING AND CHARACTERS

STORY AND NARRATIVE

STORY CONCEPT

The "Virus V" was created as an unsuccessful experiment of the governmental project "Salvation" for creating anti-cancer drug. The development was taking place on the island, where mostly project related workers were living, but because the project was secret, there we're normal citizens as well, who didn't know about existence of this laboratory experiments on the island. From the very start the infection penetrates nervous system destabilizing motor functions and hence senility. The "Virus V" is distributed by breathing, and in order to prevent global epidemics, the island's been closed and the regime of quarantine was set up. The only terrestrial entrance to the island – the bridge is controlled by Special Forces, who are called "Bridge Watchers".

The governmental preparations allow people to live in their homes, which are really working as bunkers, with have its own stations for regenerating oxygen with anti-mutational ingredients, to prevent illness' mutation. Vehicles are adopted to be filled by this oxygen is special facilities, which are called oxygen stations.

The protagonist appears on the island with a feeling of amnesia, but as time goes by, he remembers most forgotten parts. He remembers that once he had a girlfriend, and that she left the island before it was closed. So he's making efforts to leave the island and meet her again.

When he's almost finished making necessary preparations to leave the island, he realizes that the Bridge Watchers had disappeared, and he rushes to the other side of the bridge.

He's feeling happy, but soon he realizes, that the Virus V escaped from the island and the infection became Global. But he still believes that he will find his girlfriend, and here the game ends.

PLOT ELEMENTS

Storyline starts at 2012. There is a party — Global Government.

Global Government on the UN annual meeting — petition (60 percent of world population signed).

More than enough to start a party in most of the countries. Global Government — Steve Jobs and others

Smart people, they just want to make the world better. GG starts Project Salvation — island in the

Open ocean — secret headquarters of the organization.

By 2020 GG is elected as a major party in 120 countries, majority of countries. GG rules the world

They've got a lot of money. Salvation has received major scientist, who are kidnapped and made to

2024 - A new cure for people to live 4ever, people are choosing immortality. Cure provided by

Religious parties are protesting against GG.

2025 death vaccine — first incidents after receiving vaccine — people are feeling weak and sick.

By 2026 Global epidemic is inevitable. The world is about to collapse. The game world story starts

Gray Cardinals concept.

GG — unite people and move the nations forward.

Salvation was started as the GG headquarters and the place where to hide from nuclear war.

SKETCHES AND SCHEMES
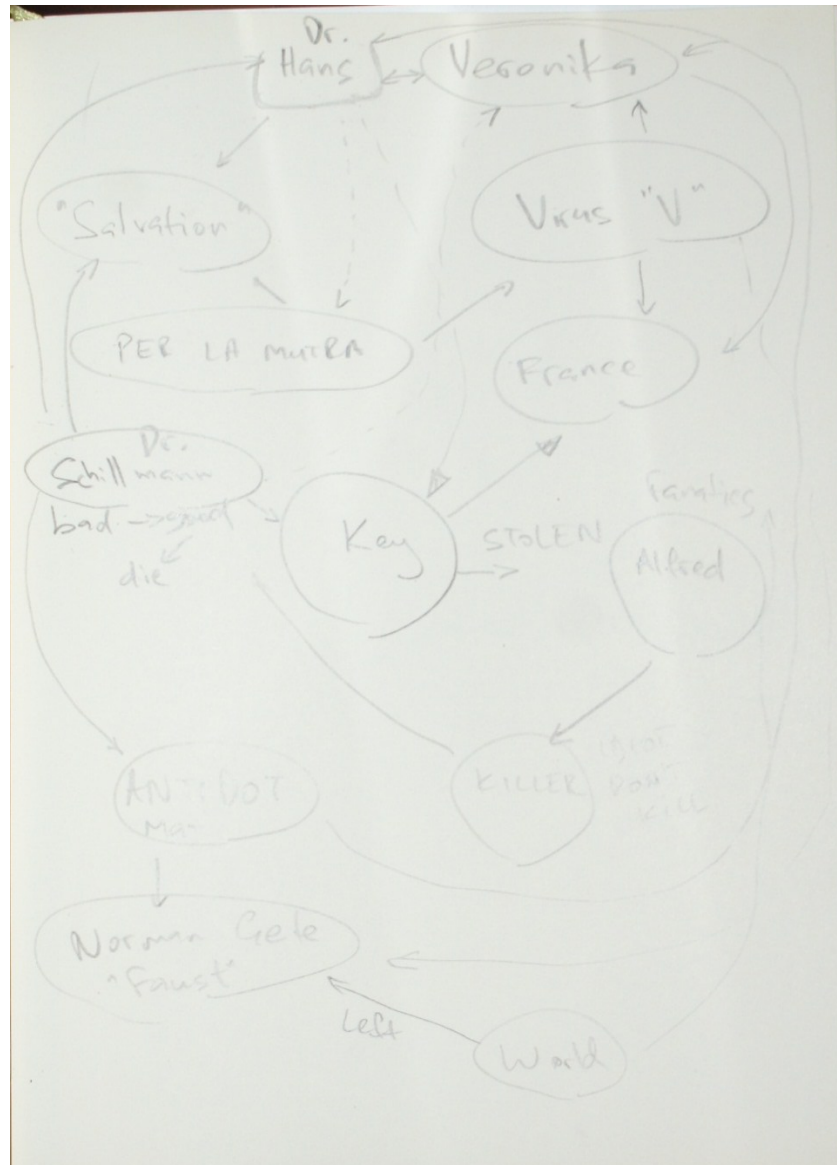
Game World
General look and feel of the world
ISLAND

General Description

Everything should be white — without bacteria, clean.

People's clothes black for the contrast

Every building has and access to the oxygen station, people are not willing to move around the

Streets coz of oxygen. People stay in the buildings most of their time. They go outside in cars.

Oxygen devices. Oxygen station is controlled by GG's. The main protagonist wake up and finds

Himself on the Salvation Island, he got amnesia, he doesn't remember what happened before. Hans

He is a doctor of Chemistry. He's participated in a project without knowing it, he did some research

However he never knew what had been the bigger picture.

PHYSICAL CHARACTERISTICS

Floating Island in the open ocean. Huge enough to satisfy stability.

CONNECTIONS TO OTHER AREA

Only secret underwater submarines, which are controlled by a GG.

Characters
Hans Christoph (The Protagonist)

BACK STORY

Hans Christoph — main hero. Protagonist.

God is merciful — Ioann — Johann — Hans

SECTION IV – MISSIONS

FREE RUN

SYNOPSIS

In this mission, the player can freely move around the map and explore its possibilities.

MAP

Currently available locations.

RACE

SYNOPSIS

In this mission is the matter pass a specified number of rounds in a closed circuit faster than the opponents.

OBJECTIVES

Being at the finish line as quickly as possible

MAP

One of predefined closed circuit.

TRAINING MISSION

SYNOPSIS

This mission is supposed to teach a player how to use controllers for driving around the city, basic HUD elements, key elements, etc.

ALMOST THERE

SYNOPSIS

Ugly wicked mission – player have opportunity to finally leave island, but is trap (or something) and it will twist other way (or something)

INTRODUCTORY MATERIAL (CUT SCENE)

Awesome way to get out of island appears before player, almost too easy.

OBJECTIVES

Get to the bridge.

MAP

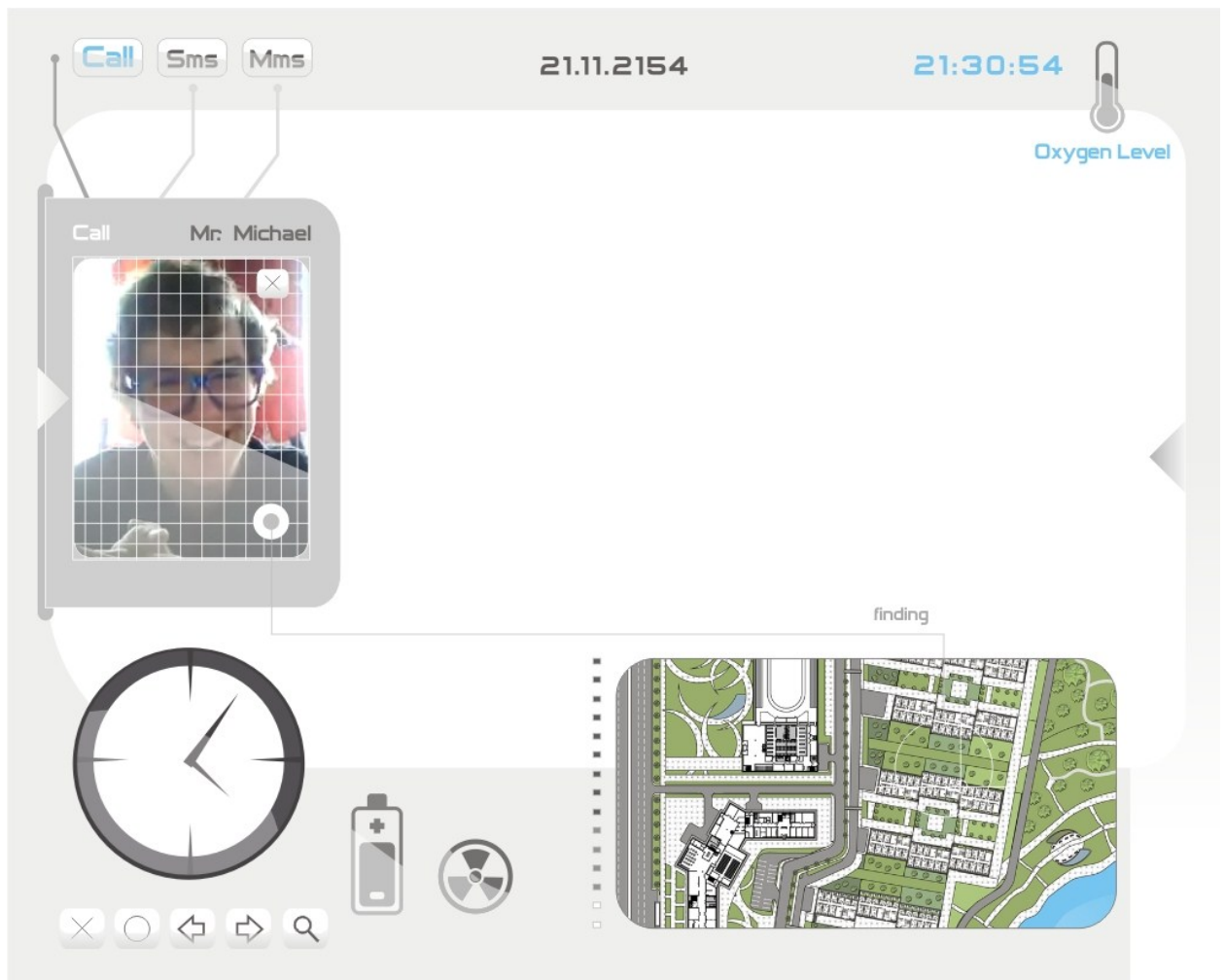Player starts furthest away of bridge as possible.

MISSION WALKTHROUGH

Player is driving towards the bridge but suddenly he's stopped.
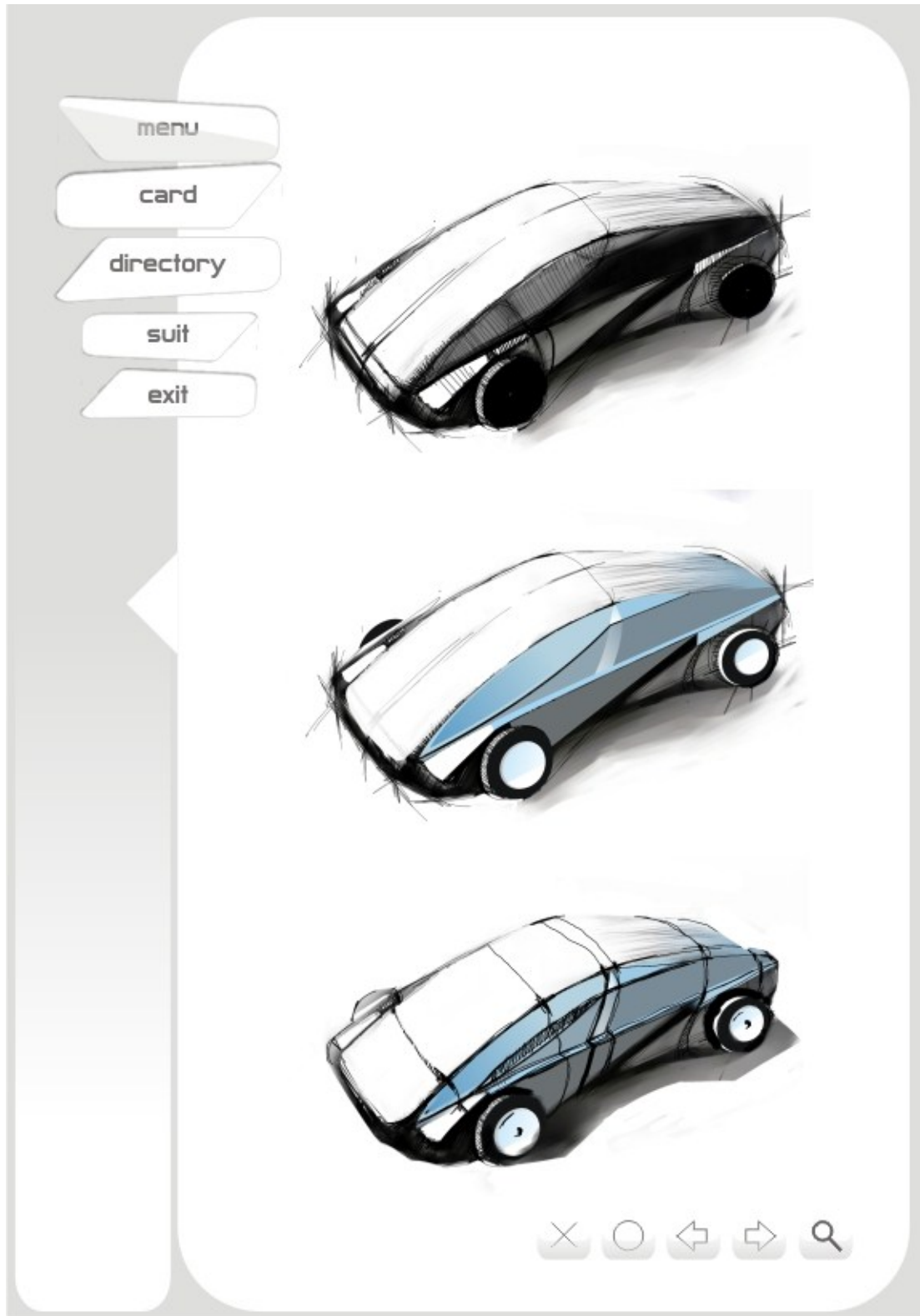
SECTION V – INTERFACE

VISUAL SYSTEM

HUD - WHAT CONTROLS

Menus

SECTION VI – MUSIC

Currently collaborating with Larie1 (http://funkshuiradio.com/) on game's soundtrack. All tracks are going to be composed and arranged for this game specifically, to match our general feeling of the game.

What's know now is that different locations of the island will have specific themes, music style is ambient mostly.

SECTION VII – ARTIFICIAL INTELLIGENCE

Artificial intelligence will be directly applied to the control of the vehicle model. Each vehicle will therefore be accessed via the same API as the vehicle controlled by the player. Simply speaking, the AI will determine whether to accelerate, or turn left.

MAP

Map will be represented by a graph, where vertices are the places where it is necessary to respond in a certain way depending on the type of vertices. Vertices are not only a crossroad, but also a traffic sign, or some other significant points on the map. Edges of the graph will be individually routes, by which the vehicle can go, direction of edges determines the direction of transport.

OPPONENT AI

Opponent will be to find a route uses a modified A * algorithm, which will also consider the player's route, which will try to predict and therefore tactically cross and block.

When passing by the ideal route will help to correct the deviation estimate distance from objects to aid decision-making route maps and will choose the most ideal solution to a concrete situation.

PASSIVE AI

Passive AI will use methods Racing Lines, which will be corrected by the active response to the surroundings, ie the surrounding traffic and road signs. It will therefore adhere to speed limits on the crossroads of priority and more. Racing Lines, then sketched the route in advance, which followed the vehicle, these are already part of the map, will be divided into sub-parts, always from the intersection to intersection. The choice to follow another route will be by chance.

COLLISION DETECTION

Vehicles will help detect direct collision with beams, similar to that of contemporary real-world vehicle parking assistant.

FRIENDLY CHARACTERS

Friendly characters will use the same AI as opponents, in key situations, of course, otherwise preserved - for example, will block opponent car, instead of a player.

SUPPORT AI

PATH FINDING

Searching for the optimal route, which the player gets from the current position to the Objective will be used modify A* algorithm, as well as searching the way for a player's opponents.

SECTION VIII – TECHNICAL

TARGET HARDWARE

The Game is going to be run on both PC with installed Microsoft Windows 7 and XBOX.

DEVELOPMENT HARDWARE AND SOFTWARE

The Game is developed on PCs running different versions of Microsoft Windows.

For coding, modeling and compiling Microsoft Visual Studio 2010 Ultimate is used.

For creating 3D Models we are using Autodesk Maya 2011, Softimage 2012 and Google Sketch-Up.

For versioning source codes is used SVN, for ticket and bug tracking we are using Trac. For messaging, exchanging and editing text files we are using Google Docs, Skype and ICQ.

For formatting and publishing documents Microsoft Office 2010 is used.

DEVELOPMENT PROCEDURES AND STANDARDS

GAME ENGINE

Microsoft XNA 4

Game engine puts to the use open-source physical library JibLibX.
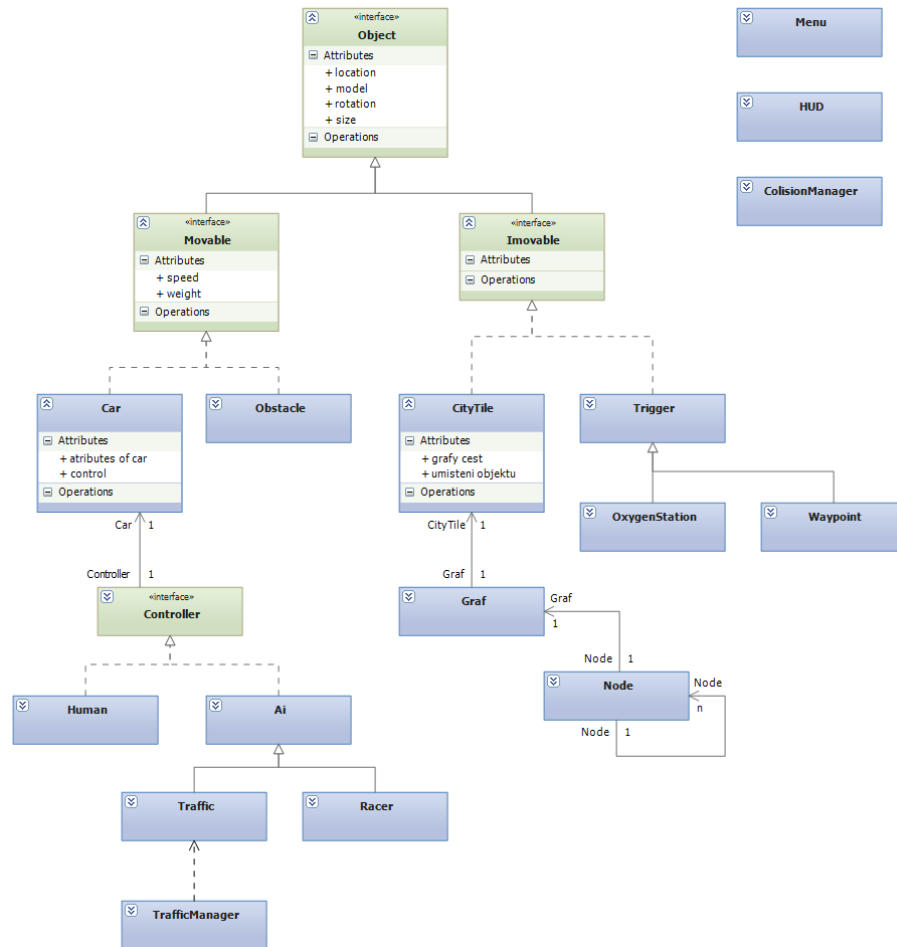
All in-game objects will share a common Object interface. It contains basic information about objects such as location, model, size, rotation, etc. It also contains common abstract methods for drawing and position manipulation to be implemented accordingly in descendant classes.

All objects are divided between movable and immovable with their corresponding interfaces originating from Object interface.

Typical examples of movable objects are vehicles and obstacles. Immovable objects are map tiles and trigger objects.

MOVABLE OBJECTS

VEHICLES

Vehicles use Dependency injection pattern to determine it's behavior. For this purpose the Controller interface is used. Every vehicle can be controlled either by human player or AI.

OBSTACLES

Obstacles are only subject to physics engine without any specific functionality added.

MAP TILES

Whole map consists of multiple sized tiles. (Preferably with their size being multiples of the base size.)

Map tiles contain information about building placement and path graphs for AI. There are AI graphs for traffic and race opponents.

TRIGGERS

Triggers play the role of "checkpoints". They can be either with or without visible model. They are used for triggering events used for game progress.

GAME STATES
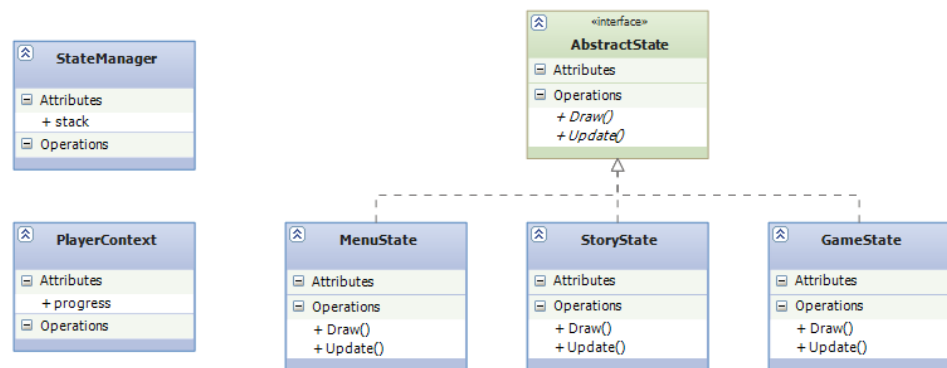
The menus, cut scenes, races, etc. are different states represented by their corresponding classes that contain functionality for correct state execution.

All states are managed by StateManager. Every time a state has to be changed. For example player chooses to start game from the menu. The State manager will push the MenuState to stack, thereby pausing it's execution and create Game state as the current one.
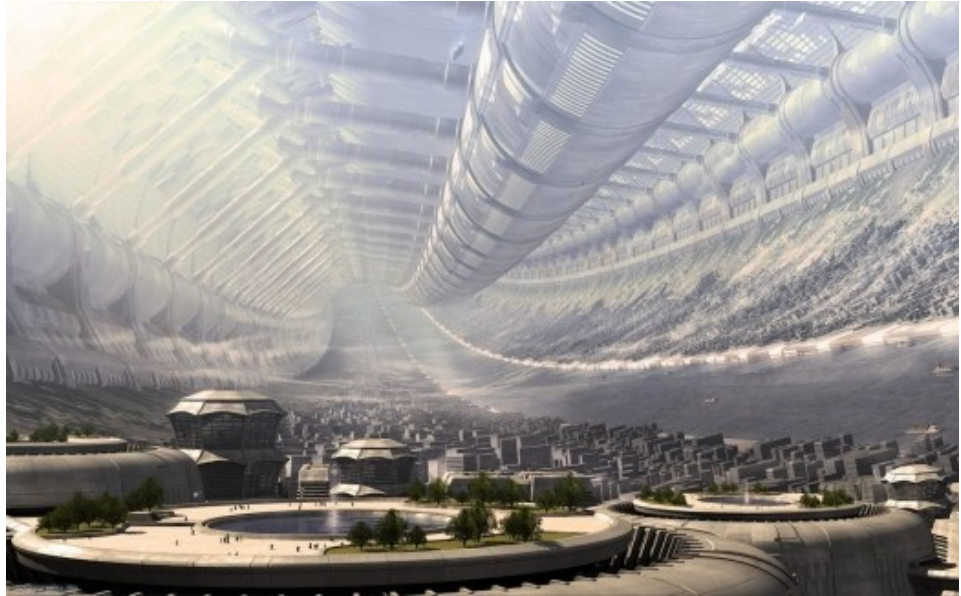
PLAYER CONTEXT

Player context class contains all information about Player's game progress. Based on its content each Game state will render corresponding part of the story.

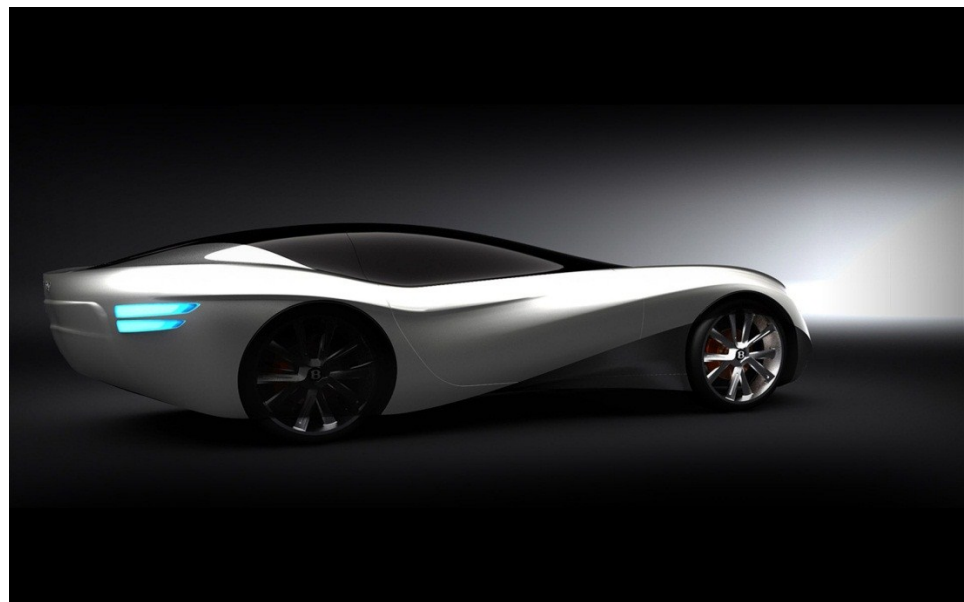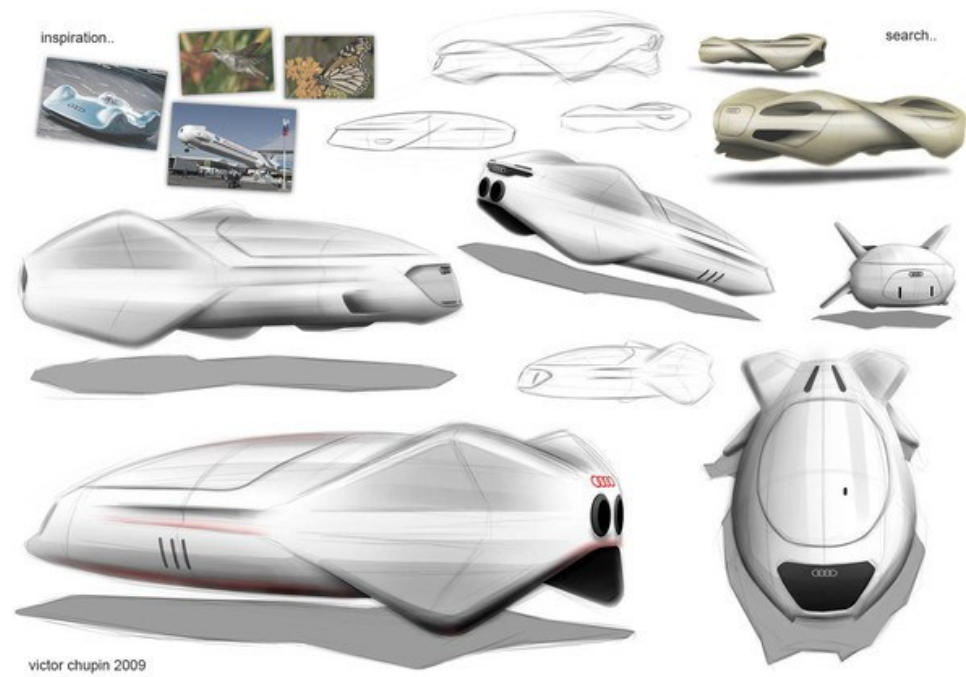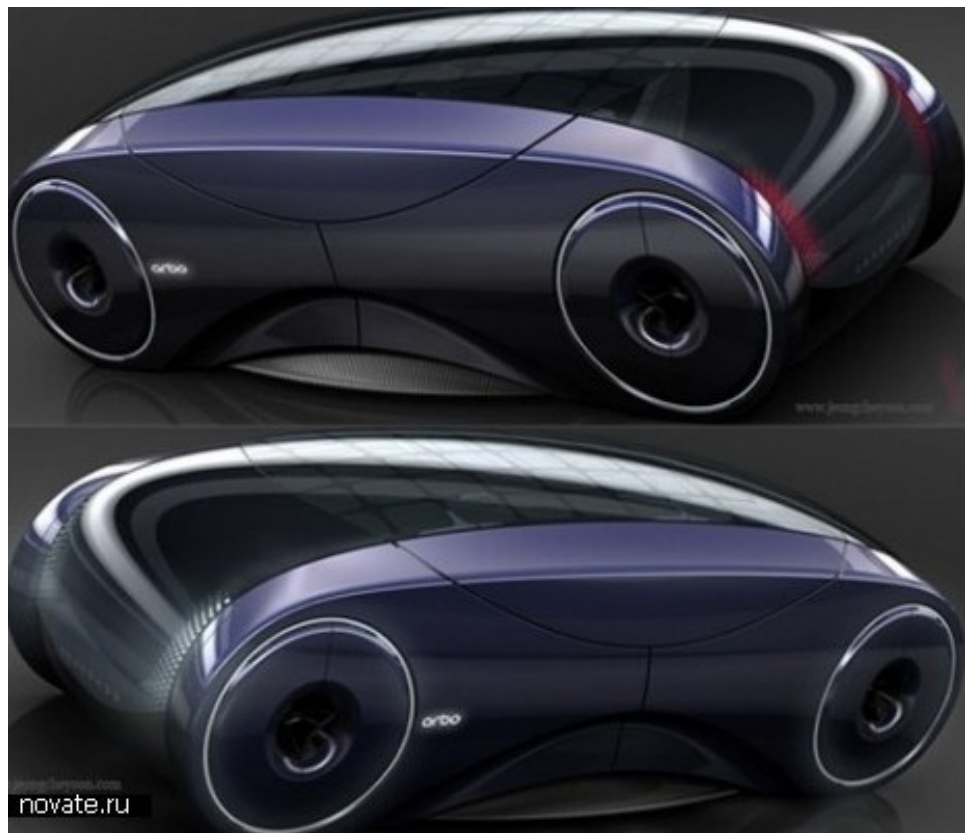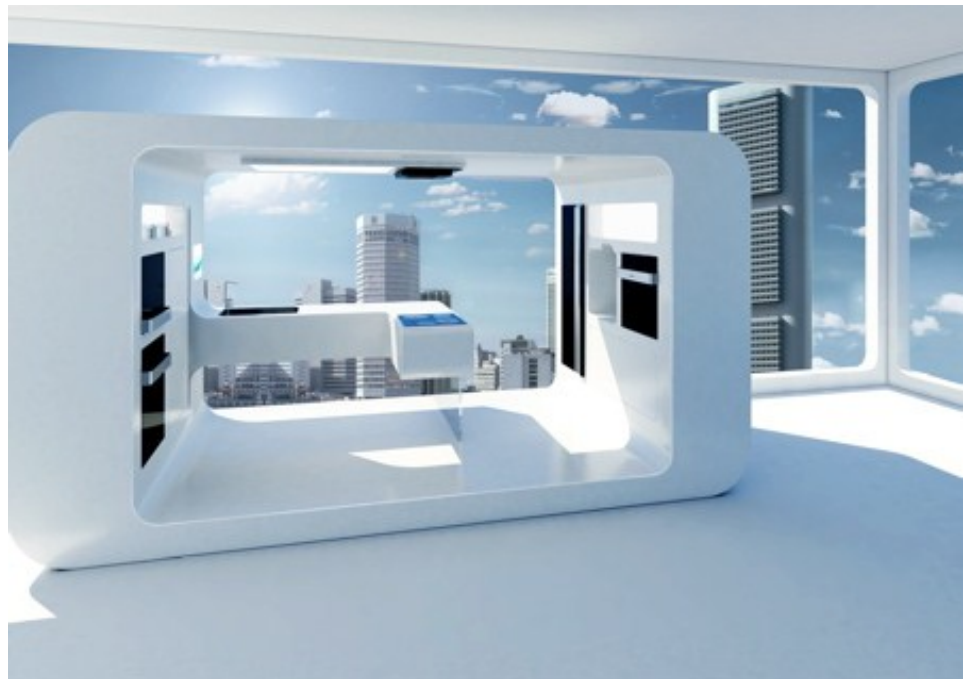# SECTION IX – GAME ART

CONCEPT ART

ISLAND LOOKS CONCEPT

Car Concepts

victor chupin 2009

GADGETS AND LIFESTYLE

Prototypes