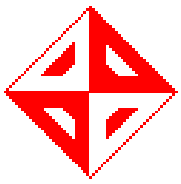
Middle East Technical University

Department of Computer Engineering



CENG491
Computer Engineering Design I
Requirement Analysis Report

CodeSchbeke Software Solutions



Sıla Arslan
Çağla Okutan
Hatice Kevser Sönmez
Bahar Pamuk
Ebru Doğan

FALL 2005

Contents

1 Introduction **1**

1.1 Purpose of the document 1
1.2 Abbreviations and Definitions 1
1.3 System Purpose 2
1.4 System Scope 2
1.5 References 2

2 Research **3**

2.1 Market Research 3
2.2 Technical Research 4
2.3 Survey 7
2.4 References 9

3 System Description and Modeling **10**

3.1 Functional Requirements 10
3.2 System Modeling 12
3.2.1 Data Flow Diagrams (DFD) 12
3.2.1.1. DFD Level 0 13
3.2.1.2. DFD Level I 13
3.2.1.3. DFD Level II 14
3.2.2 Use Case Diagrams 16

4 System Requirements **17**

4.1 Software Requirements 17
4.2 Hardware Requirements 18

5 Project Management Plan **18**

5.1 Team Organization 18
5.2 Process Model 18
5.3 Schedule 18
5.4 Risk Management Plan 19
5.4.1 Project Risks 19
5.4.2 Risk Table and RMMM 20

A Appendix **22**
A.1 Tool Comparison Table 22
A.2 Survey Questions 24
A.3 Data Dictionary 25
A.4 Gantt Chart 30

1 Introduction

1.1 Purpose of the document

This document is intended to describe the software to be developed by giving a detailed definition outlining its borders. The requirements of the user, the functionality of the system to satisfy these, main components of the system and their interactions will be described in detail. To construct a time plan is the other goal of this document.

1.2 Abbreviations and Definitions

- Abbreviations:

2D	: 2 Dimensional
3D	: 3 Dimensional
AI	: Artificial Intelligence
AVI	: Audio Video Interleave
CM	: Configuration Management
CMP	: Configuration Management Plan
DFD	: Data Flow Diagram
GNU	: Gnu's Not UNIX
GUI	: Graphical User Interface
JPEG	: Joint Photographic Experts Group
ODE	: Open Dynamics Engine
OpenGL	: Open Graphics Library
OSG	: OpenSceneGraph
RMMM	: Risk Mitigation Monitoring Management
.max	: 3D Studio Max modeling files extension

- Definitions:

Road Map	: The combination of roads with different/same characteristics.
TraffEdu Editor	: 2D Editor to design the traffic case with its environment.
TraffEdu Viewer	: 3D Viewer that runs the executable output of TraffEdu Editor.

1.3 System Purpose

The purpose of the system is to provide an educational tool that is to be used to design and animate possible cases occurring in vehicle traffic. The tool will assist anyone who wants to teach traffic by supplying a flexible and easy to use environment.

1.4 System Scope

Knowing traffic rules for people involving in traffic either being a pedestrian or a driver is a vital concept since most of the accidents, if not all, occurs due to violation of traffic rules. At this point people responsible for teaching traffic are loaded with big deal of charge. Since teaching and learning traffic from books is not satisfying, our system TraffEdu is intended to provide an environment that will ease the work of the people teaching traffic by enabling to construct a real traffic case implementation in 3D.

The system TraffEdu will consist of mainly two parts: one of which is a 2D editor (TraffEdu Editor) that will allow the user to prepare any traffic case which will include roads, road junctions, traffic signs and lights, curves, and vehicles of any kind. According to the specifications that the user will provide such as the behaviors of vehicles that do(not) obey traffic rules (e.g. (not) stopping at red light, exceeding speed limit, their path etc...), and environment characteristics, a file of a specific format will be produced. The functionality of second part (TraffEdu Viewer) of the system is to construct a 3D animation by using the data stored in this file. This animation will not be in the format of AVI file which does not allow user interaction. It will enable the user to change some parameters which will affect scenario and view of the case.

It is important to point out that the tool is not intended to teach how to drive to drivers but rather to ease teaching traffic rules.

1.5 References

- Project Proposal Report
- IEEE Guide for Developing System Requirements Specifications
- Pressman, R. S., *Software Engineering - A Practitioner's Approach*, 5th Edition
- Utah University Computer Science Department SRS outlines

2 Research

2.1 Market Research

In building up projects, market research is very important since it helps the developers to outline the borders of the system scope and its requirements. So to get down to the functional requirements of the system TraffEdu effectively, we made an extensive research in the area. Not only similar tools to TraffEdu system, but also the software packets that only support predisposed 2D/3D traffic cases are analyzed. Analyzing the market in such a wide perspective helped us to form and shape the most possible expected characteristics from TraffEdu. There are mainly three types of methodologies in education of traffic using computer and/or visualization (animation):

- Prerecorded traffic cases to video (restricts the trainer to predisposed cases)
- 2D traffic case animations (again restricts the trainer to predisposed cases but increases the interactivity using text based animated tests)
- Animations of user defined traffic accidents in 2D/3D (the most similar systems to TraffEdu).

The last group of tools providing the user to create his/her own traffic cases are usually used for analyzing traffic accidents and used by police departments.

Scene PD 2.0 [1], Easy Street Draw 3.0 [2], 3D Eyewitness 7.9 [3], Vista fx 2.0 [4], Crash Zone 7.0 [5] and MicroSurvey MapScenes Pro SP 2.0 [6] are the most similar systems to TraffEdu that are taken into consideration in detail. Among these, Scene PD 2.0 and Easy Street Draw 3.0 have support for only 2D drawing. They are also analyzed to get some creative ideas for TraffEdu 2D Editor. (For tools' specific characteristics refer to Appendix A.1).

We had also analyzed both traffic and driving simulation tools at the time of determination of our project scope and definition. (For related survey refer to literature survey section of proposal report [7])

The common characteristics of these tools are:

- The main characteristic that these tools have in common is their purpose; they all intend to ease the job of police departments to analyze the traffic accidents.
- They all have a 2D editor to ease the potential user's job; it helps the user to handle the traffic case environment s/he is designing.
- They all support various 2D objects e.g. cars, tracks, pedestrians, trees, buildings etc.

- The ones that have 3D functionality support various 3D models.
- The 3D animation and image construction tools support different camera views.
- The 3D animation tools don't have any feature for interactivity at the time of animation. Interactivity is vital for education tools because it gives the trainee the feeling of involving in the course. Since TraffEdu is an education tool, it shall have certain interactive features.
- They all support predisposed traffic cases and road maps.
- They all support saving and loading/opening file.
- They have no artificial intelligence properties. Every object behavior is defined by the user beforehand.
- They have support for usage of the tool via a help menu.

The different characteristics of these tools are:

- Some tools have a really exhaustive object database in that they give support for not only a car object but also its model and brand.
- Some tools give support for camera views either from inside or outside of vehicles.
- Some tools give support for different output files e.g. a JPEG format file in addition to an AVI format file.
- In addition to 3D animation, the 3D tools also support 2D animation from top view.

2.2 Technical Research

Technical research is needed in order to determine which software tools and libraries will be used in the project and how the interconnections between them can be setup. This is an important step in the development of the system and the libraries, engines and tools must be carefully determined such that no integration problem occurs in later steps. A number of researches are conducted through internet and via e-mail in order to gain knowledge and experience about the tools that will facilitate during the configuration of the TraffEdu system.

The researches we have made related to the technical issues can be sum up in six groups:

- **Researches related to rendering engine:**

  Rendering is a mechanism that provides the final 2D image of 3D object models, which is shown on the screen after several steps of mathematical calculations. Besides forming the images, these engines provide a way for an infinite loop that is needed for the 3D animation.

  After the meeting with Veysi İşler [8] and Onur Soysal [9], the following rendering engines and platforms are considered mainly:

  - RenderWare Graphics SDK PC 3.7
  - Blender3D 2.37a
  - OpenSceneGraph 0.9.9

  Among the three, **RenderWare** supplies the most extensive functionalities including Graphics, Physics, AI and Audio [10]. It provides the following general functionalities such as: Object-Oriented Design, Plug-in Architecture, and Save/Load System. It has a built in editor which gives the user the opportunity to set some physic assets and preview them. The Physics engine of the platform provides basic physics, collision detection, and physics for rigid body. It also supplies 2D and 3D sound. The rendering functionalities it presents are quite advanced. It provides lightening, shadowing and texturing features [11]. Since RenderWare is a newly developed platform, its community is not broad enough to solve problems probable to be encountered and its features are mostly related to game development. Additionally RenderWare, supported for academic purposes, restricts the use of several functionalities.

  Blender is an open source software which provides 3D modeling, animation, rendering, post-production, interactive creation and playback that is available for all major operating systems with GNU Public License. Its key features are [12]:

  - Fully integrated creation suite, offering a broad range of essential tools for the creation of 3D content, including modeling, animation, rendering, video post production and game creation
  - Small executable size for easy distribution
  - Cross platform, with OpenGL based GUI, ready to use for all flavors of Windows, Linux, OSX, FreeBSD, Irix and Sun
  - High quality 3D architecture enabling fast and efficient creation work-flow
  - Free support channels via its web site
  - More than 250.000 people worldwide user community

  It supports importing and exporting 3D Studio Max models with the help of python scripts. As stated above, the software has a large community and documentation about the software is easily reachable from the site.

OpenSceneGraph is another open source 3D graphics toolkit which is a "cross platform graphics toolkit for the development of high performance graphics applications such as flight simulators, games, virtual reality and scientific visualization."[13]. General features the tool presents are Object-Oriented Design, Plug-in Architecture, and Save/Load System [11]. OSGEdit is an editor of scenes for OpenSceneGraph. It is a composer for a complex OSG scene into the application by importing the 3D models and editing them [14]. The tool provides basic physics, collision detection, and rigid body [11]. It also has an importer for files with .max extension. The OSG community and documentation is sufficient for getting any technical help.

Consequently among these three rendering tools we decided to prefer the open source ones since they provide more flexibility without putting any restrictions and also have a larger community that is required in case of any programming help is needed. Experimentations with OpenSceneGraph and Blender rendering engines are continuing and we have not yet decided which one to choose.

- **Researches related to modeling:**

  Modeling provides the creation of data structures used to represent objects and scenes. This process is different from rendering and generally is done by a help of special-purpose software. During the tool search for modeling, we conducted with Donald Newlands from company Newlands & Company, Inc. via mail [7]. After this mailing period, we decided to use predisposed models. We chose to do so in order not to waste time with modeling but rather to concentrate on fulfilling main requirements of TraffEdu.
  We mainly focused on 3D Studio Max [15] and Maya [16]. We chose to use 3D Studio Max models since it is possible to find importer plug-in for the two open source rendering engines we preferred as mentioned above. OpenSceneGraph and Blender also allow creation of models. Preferring 3D Studio Max models over rendering engines' own models is due to the large amount of 3D Studio Max models that can be found over internet.

- **Researches related to physics engine:**

  Physics engines are used to ensure realism in computer games. They apply Newtonian physics models to the objects to get a simulation that is much closer to the real life. In TraffEdu, physics engine will provide the necessary realistic behaviors of the vehicles that are in environment and also between the interactions of several vehicles such as collision. We decided to use Open Dynamics Engine 0.5 (ODE) as our physics engine because it is an open source library and very popular in game programming and also it provides many features for realistic simulation of objects [17]. It provides simulation of articulated rigid

body structures; it is possible to change the structure of the system during real-time simulation with ODE which will be used in the interactivity feature of TraffEdu Viewer. Also ODE has a built-in collision detection system that will be used in the crash scenes of the vehicles in 3D environment.

- **Development Environment:**

  The Operating System of the development environment will be Microsoft Windows XP Professional [18]. We will use C++ as the programming language and the development platform will be Microsoft Visual C++ 6.0. Choosing MS Windows is necessary due to problems that may occur with Linux graphic card drivers. All the rendering engines and physics engines mentioned above support Microsoft Visual Studio platform.

- **Texturing:**

  Texturing makes models more realistic by addition of some images. For creating images and texturing Adobe Photoshop will be used. It supplies a great variety of image editing features and easy to use interface. The texturing will be used in 3D view, and maybe applied on pedestrians, vehicles and buildings. Texturing may also be used in adding text fields to the 3D view that is supplied by the trainer during construction of the traffic case.

- **Audio:**

  The purpose of audio usage in TraffEdu is mainly to present a more effective way of demonstrating the 3D view. We will use predisposed audio files for our application but for the files that could not be found over internet we may use some audio processing tools.

2.3 Survey

User domain of TraffEdu system is mainly composed of traffic trainers since it is mainly intended to ease and facilitate the job of people teaching traffic. That's why, in the requirement analysis phase we preferred to conduct this survey with traffic trainers.

The aim of this survey is to look from the trainers' view, capture the points that can be overlooked and take their advice.

The survey is made with "Directorate of Traffic Control Department" and four traffic courses[1] interactively and is sent to other traffic courses outside Ankara

[1]Özakyıldız Sürücü Kursu ,Özkızılay Sürücü Kursu, Özler Sürücü Kursu, Yeni Binyıl Sürücü Kursu

via e-mail. Two of the courses[2] to which e-mails were sent, responded the questionnaire and we are in touch with both presently.

The survey questions are prepared so that familiarity of the user with computer, applied tools and technologies in traffic education, how much the issued tool will facilitate the trainers, which functionalities are absolutely essential are determined. Appendix A.2 can be referred for the questions of the survey.

Generally the trainers conducted, found TraffEdu system very beneficial and practical. Brief summary of what have been gathered from their answers is as follows:

- The trainers have insufficient knowledge of computer usage. This resulted in the need of providing detailed help menus as well as a prior training to end users.
- Most of the courses do not utilize computers in education due to the absence of resource.
- Most of the courses do not use simulators which became a popular device in traffic education in recent years.
- It is stated that, air conditions and road conditions are missing in simulation tools but it is essential that these properties exist in the simulation devices.
- The idea of preparing the maps and the traffic cases with their own creativity, sounds the trainers very charming. Because in this way they could express themselves clearer.
- One portion of the trainers preferred to view the road from the driver seat while the majority thinks that watching over the traffic from top view is more preferable.
- None of the trainer preferred to insert text or audio into the educational program. Because they thought that by this way, interactivity may be reduced during the lessons. However, in order to make the animation created by the trainer usable by the trainee on his/her own, the system will support this feature.
- By preparing the traffic cases in such a system, they thought it will become easier to control the flow of the lessons.
- The suggested ideas that are planned to take place in the system other than the properties that are asked are as follows: Drunk drivers, noise pollution, inconvenient parking.

[2]Etik Sürücü Kursu ve Motosiklet Okulu, Gelişim Sürücü Kursu

- Since most of the trainers do not have acquaintance with such kind of a tool, they could not imagine the graphical interface of this tool in their mind.

2.4 References

[**1**] Scene PD version 2 by trancite logic systems
http://www.trancite.com/spd/

[**2**] Easy Street Drawing by trancite logic systems
http://www.trancite.com/esd/

[**3**] 3D EyeWitness Version 7.8 by DesignWare Inc.
http://www.designwareinc.com/3d_prod.htm

[**4**] Vista FX Version 2 by Visual Statement
http://www.visualstatement.com/default.aspx

[**5**] Crash Zone version 7.0 by Cad Zone:
http://www.cadzone.com/Crash_Zone/Crash_Zone.htm

[**6**] MicroSurvey MapScenes Pro SP 2(2002)
http://www.mapscenes.com/

[**7**] Literature Survey Section of CodeSchbeke Proposal Report

[**8**] Assoc.Prof.Dr. Veysi İşler
Office: B208
E-mail: isler@ceng.metu.edu.tr
url:http://www.ceng.metu.edu.tr/ isler
Phone: +90 312 210 5591
Fax: +90 312 210 1259
Address: Department of Computer Engineering
Middle East Technical University
Ankara, Turkey

[**9**] Onur Soysal
Research Assistant
Office: A301
E-mail: soysal@ceng.metu.edu.tr
url:http://www.ceng.metu.edu.tr/ soysal
Phone: +90 312 210 5545

Fax: +90 312 210 1259
Address: Department of Computer Engineering
Middle East Technical University
Ankara, Turkey

[**10**] RenderWare Graphics SDK
http://www.csl.com/

[**11**] DevMaster's Game and Graphics Engines Database
ttp://www.devmaster.net/engines

[**12**] Blender 3D Graphics Engine
http://www.blender3d.com/

[**13**] OpenSceneGraph 3D Graphics Engine
http://www.openscenegraph.org/

[**14**] OSGEditor for OpenSceneGraph
http://osgedit.sourceforge.net/

[**15**] Autodesk 3D Studio Max
http://usa.autodesk.com/adsk/servlet/index?id=5659302&siteID=123112

[**16**] Alias Maya
http://www.alias.com/glb/eng/productsservices/family_details.jsp?familyId=3900009

[**17**] Open Dynamics Engine
http://www.ode.org/

[**18**] Microsoft Windows XP Professional
http://www.microsoft.com/windowsxp/default.mspx

3 System Description and Modeling

3.1 Functional Requirements

The research period extremely helped us to construct the functional requirements of system TraffEdu. Most expected requirements are taken as the main requirements and others as extra features. The first eighteen are the ones that will

have higher priority with respect to remaining six requirements.

1. TraffEdu Editor shall provide the user to construct the demanded traffic cases and environment.

2. TraffEdu Editor shall support the user to create any road map s/he likes.

3. The system shall provide a database that includes sufficient objects (e.g. cars, trucks, traffic lights & signs and possibly some buildings) that facilitate to generate the scenario and necessary traffic cases.

4. TraffEdu Editor should enable the user to add/remove various objects from the 2D repository of the tool.

5. TraffEdu Editor shall provide the user to define the behaviors of the vehicles such as defining the path of the vehicle, behaving according to some of the traffic signs and traffic lights, obeying to or violating the passing precedence rules, exceeding or obeying the speed limit etc.

6. TraffEdu Editor shall provide the user to define traffic lights' synchronization.

7. TraffEdu Editor shall provide the user to define all these objects' behaviors according to timetable and/or loading the required knowledge of traffic rules to the objects.

8. TraffEdu Editor shall provide the user to add various texts and some audio (effects and a speech of the user)

9. TraffEdu Editor shall give the user to choose the hours of the day.

10. After s/he is done and satisfied with the traffic case produced, s/he will have the option to save this environment data in a file with a specific format either by saving the file edited or giving animate instruction to the system.

11. The user will be able to construct an executable file with animation instruction.

12. The system will provide a mechanism to translate the file created with the TraffEdu Editor to 3D animated scenes, using the file with the specific format.

13. TraffEdu Viewer shall provide realistic traffic situations by providing the vehicle movements closer to the real world. That is, the system shall have the property that the objects in 3D act synchronously and behave correctly in accordance with the traffic cases, generating a traffic scene close to real by using car physics.

14. The view angle of the camera during the animation of the scenario can be changed.

15. During the animation phase, it will be possible to change the behavior of a vehicle for demonstrating the results of some wrong action.

16. The system shall provide some sample traffic scenarios that helps the user for constructing of their own.

17. The system shall provide a way to watch the previously saved scenarios by running the executable files.

18. The system shall include a tutorial that demonstrates how to use the system in an understandable way, enriched by samples in a stepwise manner.

19. TraffEdu Editor may provide the user to use different type of roads while constructing road map (like unimproved or asphalt roads).

20. TraffEdu Editor may provide the user to design cases in different weather conditions.

21. The system may include additional objects of environment such as pedestrians, buildings, trees etc. to create more realistic environment.

22. The system may provide the flexibility to add new traffic signs and rules that will affect the behaviors of vehicles.

23. 3D models in the repository may be changed or new models may be added by the user.

24. The system shall have the ability to demonstrate the faults that drunk drivers make, the effect of noise pollution and inconvenient parking in traffic.

3.2 System Modeling

3.2.1 Data Flow Diagrams (DFD)

Due to the fact that TraffEdu system is mainly composed of two parts and there is considerably important information flow not only between two but also within these, DFD is chosen for modeling the system functionally. Appendix A.3 can be referred for detailed description of data elements.

3.2.1.1. DFD Level 0

The overall system along with the input data required and the output data produced is depicted in Figure 3.1. The system accepts mouse and keyboard inputs from the potential users and it provides audio and graphics outputs via speaker and monitor. 3D models, textures to apply, audio data, and images needed for the system are kept in a repository.

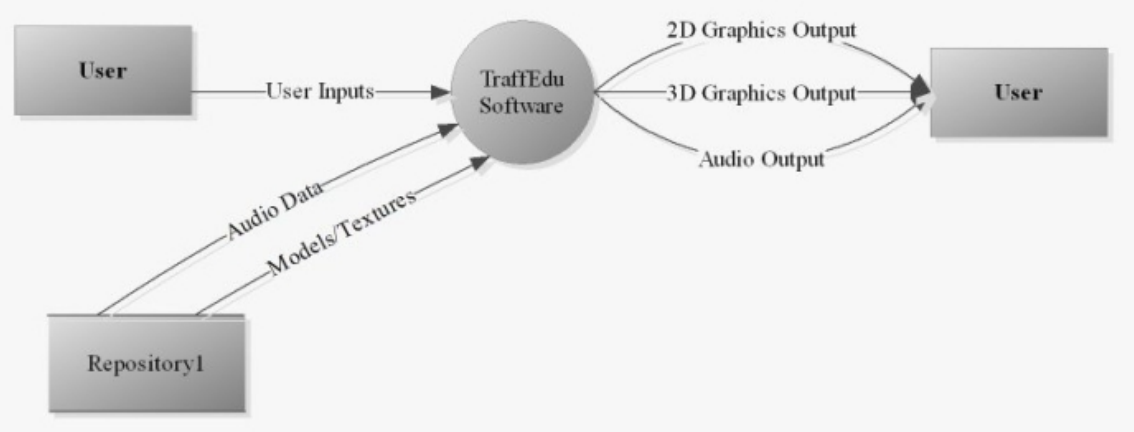


Figure 3.1: DFD Level 0

3.2.1.2. DFD Level I

Four main modules are found for TraffEdu system as shown in Figure 3.2 as DFD Level I. These modules are described below:

Event Handler Module: Data received from user firstly enters into Event Handler Module so that the module directs the data to 2D Editor, File Handler or Simulation Engine Modules according to the content of the input. That is to say this module meets the user inputs.

2D Editor Module: This module is the brain of TraffEdu Editor. It provides a GUI to the user while designing the road map and other environmental characteristics during constructing the traffic cases in 2D. The most important functionality of this module is to produce all case data interpreting the user inputs. 2D Editor Module interacts with File Handler Module by sending the produced case data and receiving extracted case data to open a pre-designed environment.

File Handler Module: This module supplies two main functionalities. First of which is, packaging data taken from 2D Editor Module into a file of a specific format. When Event Handler sends save request to this module, it sends case data request to 2D Editor Module so that 2D Editor returns required information to be packaged. Second one is extracting required data from saved file. This functionality supplies the data flow from repositories to 2D Editor and Simulation Engine Modules

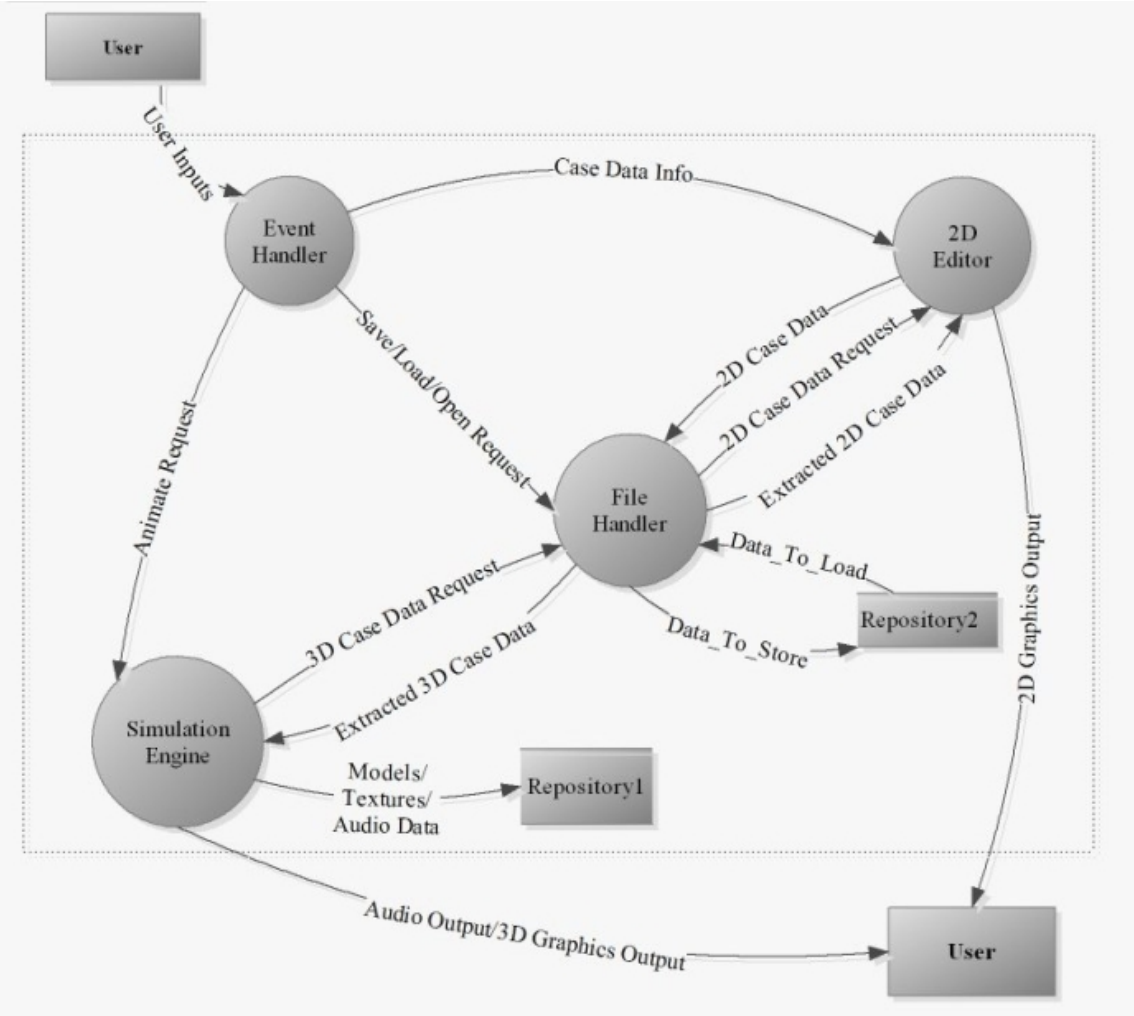


Figure 3.2: DFD Level I

so that required data is gathered. Similarly, File Handler Module extracts required case data and sends to Simulation Engine Module in response to a request from that module.

Simulation Engine Module: This module is the brain of TraffEdu Viewer that implements all 3D functionalities. Animation request that comes from Event Handler is handled by this Simulation Engine Module. It also has an interaction with File Handler at the time of translation from 2D data to 3D animation. Further details of this module will be explained in 3.2.1.3.

3.2.1.3. DFD Level II

Inner structures and the data flows of Simulation Engine Module are shown

in Figure 3.3. This module consists of five sub-modules described following the diagram.

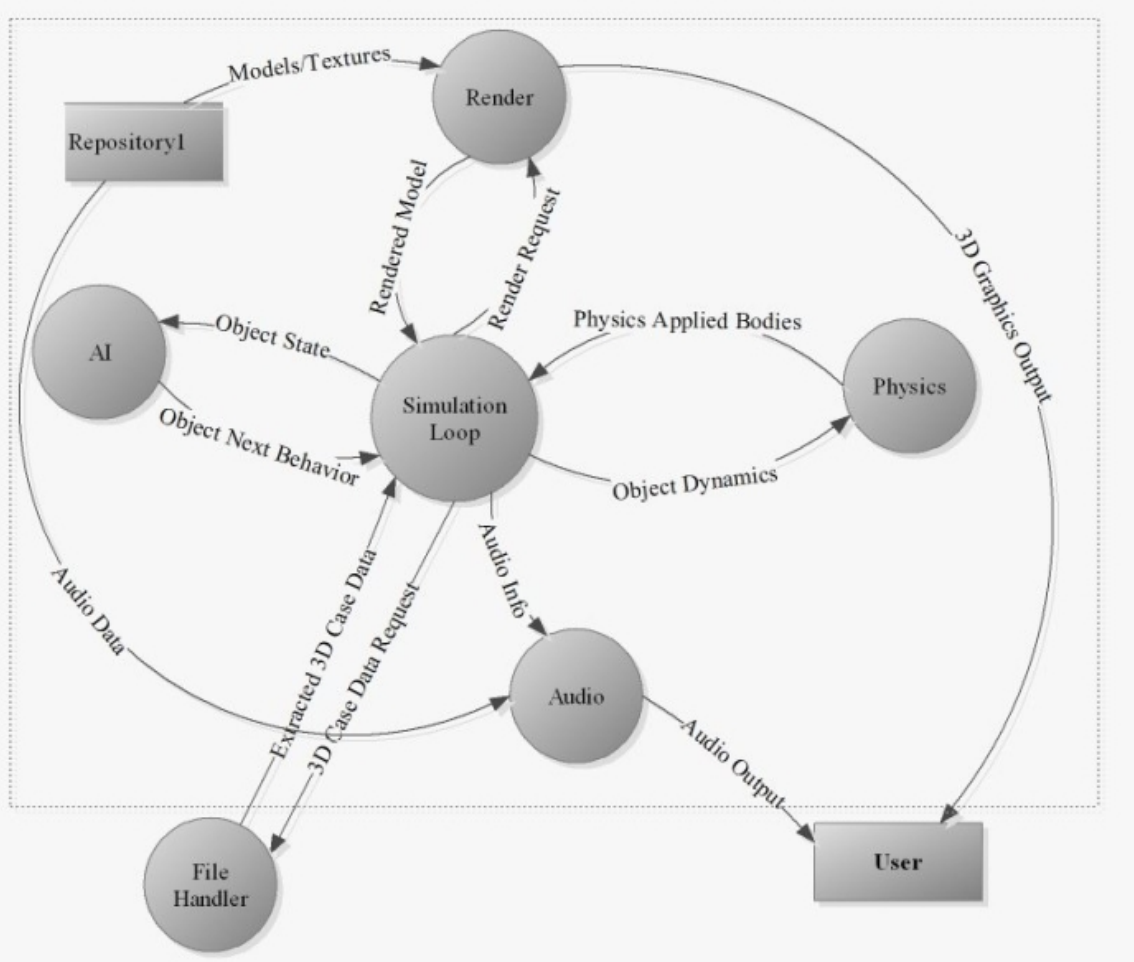


Figure 3.3: DFD Level II of Simulation Engine Module

Render Module: Graphical output displayed to the potential user as a 3D animation will be rendered via this module. Rendering will be done on pre-prepared models. Texturing and lightning effects will be added by this module. It will be activated by render request from Simulation Loop Module and it responds with the rendered model.

AI Module: Apart from the characteristics the user attaches to the objects while designing the traffic case in TraffEdu Editor, the system should deduce additional characteristics. This will reduce the burden on the user by imposing reasoning ability to objects in the system so that they behave realistically in designed scenarios. Objects' current states are transmitted to AI Module from Simulation Loop

Module and next state of the objects is returned.

Physics Module: This module is used for simulating vehicles in TraffEdu System so that the resulting animation becomes more realistic. This module receives vehicle features (like speed, mass, etc) and some environmental properties like friction from Simulation Engine and responds with the physics that is required for realistic animation applied to the vehicles.

Audio Module: According to the request coming from Simulation Loop Module, Audio Module transfer audio data from the repository to the user.

Simulation Loop: This module, as deduced implicitly from above, is the center of Simulation Engine Module. It provides the synchronization and an infinite loop for generating 3D animation. It decides to whom and when to transfer data.

3.2.2 Use Case Diagrams

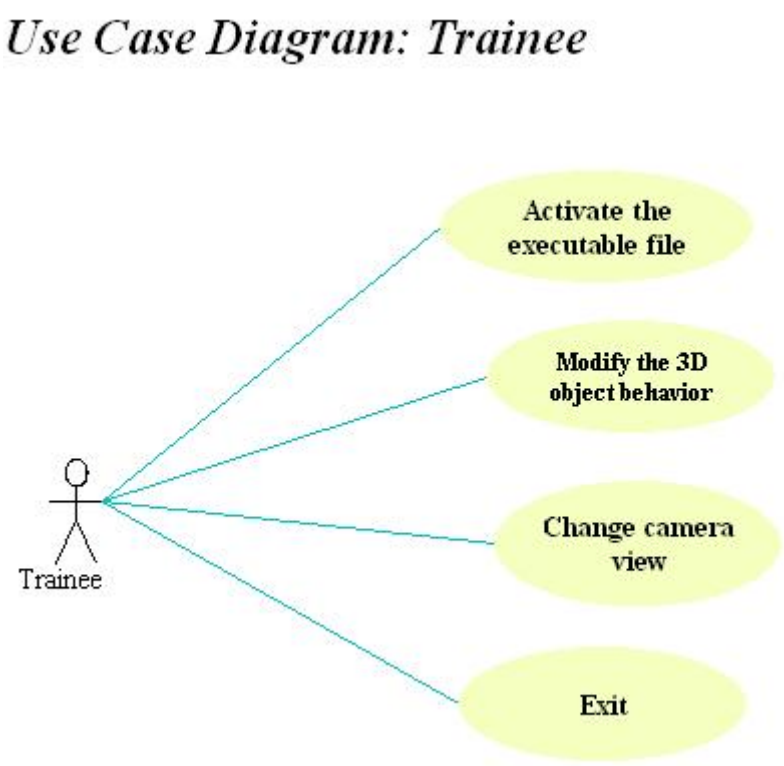


Figure 3.4: Use case Diagram: Trainee

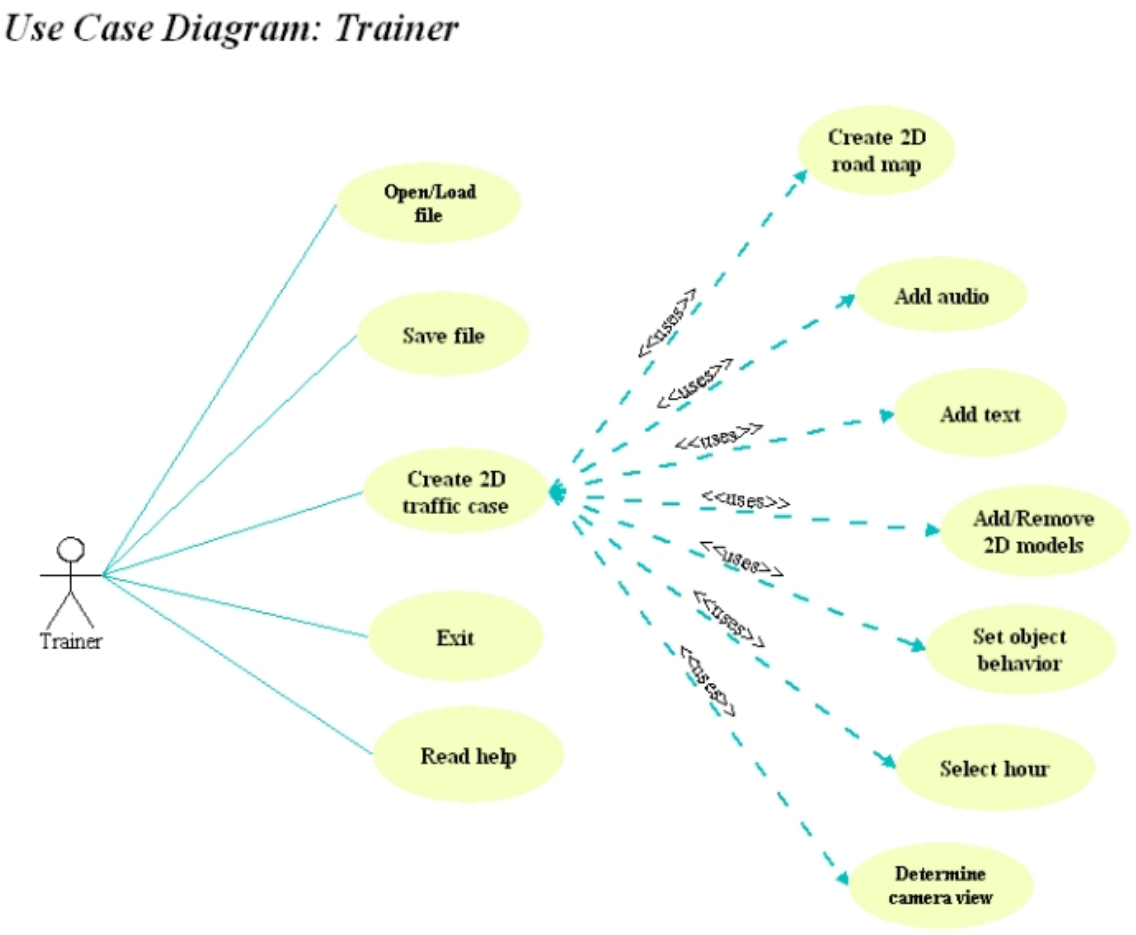


Figure 3.5: Use Case Diagram: Trainer

4 System Requirements

4.1 Software Requirements

TraffEdu System requires Microsoft Windows 98/2000/NT/XP Operating System on the user side.

On the developer side the requirements are listed below:

- Microsoft Windows XP Professional
- Microsoft Visual C++ 6.0
- OpenSceneGraph 0.9.9 / Blender3D 2.37a
- ODE 0.5

- 3D Studio Max
- Adobe Photoshop
- LaTeX

4.2 Hardware Requirements

TraffEdu System requires the following hardware platform at least on the user side:

Processor: Pentium III
RAM: 64 MB
Video display: 32MB OpenGL supported graphic card

Development of TraffEdu software requires the following hardware platform on developer side:

Processor: Pentium III
RAM: 128 MB
Video display: 32MB OpenGL supported graphic card

5 Project Management Plan

5.1 Team Organization

To encourage brainstorming and consideration of alternative solutions, we decided the structure of our team to be Democratic Decentralized (DD). Since we do not have a permanent leader, we tried to fit tasks to the capabilities and motivation of the people available, so for each task a leader is appointed.

5.2 Process Model

Since we have a progression through analysis, design, coding, testing and maintenance phases and system requirements are well understood, linear sequential model is chosen as our process model.

5.3 Schedule

For project schedule refer to Gantt chart at Appendix A.4.

5.4 Risk Management Plan

5.4.1 Project Risks

Being aware of the potential dangers that may cause our project team fail to succeed in this project, a risk management strategy has been developed. Risks that are special to our case have been identified and management strategies are formed according to these risks. Below are the potential risks gathered under seven main categories:

- Team Management (TM)
  Lack of Roles and Responsibilities: Since our team does not consist of professionals and this project is not commercial, there exists lack of sanction. This makes it possible that one member may not meet what is expected from her by not fulfilling her responsibilities.

- Project Parameters (PP)
  Large Project Size: The project scope and all the functionalities the system will supply are determined by the team. Due to lack of experience, the project size may be underestimated. That is to say, the team may have been undertook burden that is hard if not impossible to realize.
  Hardware Constraints: The end product will run on a single platform. This restriction may dissatisfy the user.

- Customer Characteristics (CC)
  Inabilities of the Customer: As it is stated in section 2.3 of this document, end users of our system have insufficient computer knowledge. This constitutes risk both during the development and after deployment of the product by complicating the communication with the user, ending with incomplete and unclear requirements.

- Development Process Risk (DPR)
  Lack of Quality Assurance Approach: Due to lack of policies and standards, end product may not be of high quality.
  Lack of Configuration Management Plan: If no CMP exists, any kind of data loss and data confusion may reveal.
  Lack of Early Identification of Defects: Auditing is crucial for a project to succeed in desired time and desired quality. Professionals in the area give approximately sixty percent of effort for identification of defects and verification process. Unfortunately, due to time and employee deficiency, insufficient auditing constitutes risk for our project.

- Employee Risks (ER)
  Low Application Experience: Not only computer graphics but also the tools

and the software that will be used is new to all group members. Lack of experience on these will certainly decelerate the process.

- Development Environment Risks (DER)
  Insufficiency of Resources and Lack of Quality of Tools: For the development environment we are restricted with the laboratories of our department. Moreover, the tools and software that we are planning to use may introduce integration problems.

5.4.2 Risk Table and RMMM

Risk Table

Potential Risks	**Category**	**Probability**	**Impact**
Lack of Roles and Responsibilities	TM	% 10	3
Large Project Size	PP	% 20	3
Hardware Constraints	PP	% 20	1
Inabilities of the Customer	CC	% 60	3
Lack of Quality Assurance Approach	DPR	% 20	3
Lack of Configuration Management Plan	DPR	% 20	4
Lack of Early Identification of Defects	DPR	% 25	4
Low Application Experience	ER	% 80	3
Insufficiency of Resources, Lack of Quality of Tools	DER	% 40	2

Impact Values:
1. Negligible
2. Marginal
3. Critical
4. Catastrophic

RMMM

Risks	Mitigation	Monitoring	Management
Lack of Roles and Responsibilities	Construct a 'Team Agreement Report', Regular meetings, Assign a reviewer to every member	Detect the deficiency in tasks	Assign more task
Large Project Size	Intend to implement reasonable amount of functionality	Check the Gantt Chart	Remove some functionalities
Hardware Constraints	Choose appropriate hardware	Detect system failure	Provide necessary hardware
Inabilities of the Customer	Regular meetings with the potential customer	Detect the increase in misunderstandings of the customer	Prepare detailed training plan for the user about product usage
Lack of Quality Assurance Approach	Prepare our own policies and standards	Examine user satisfaction	Refine standards
Lack of Configuration Management Plan	Use open source CM tools, Assign a configuration manager,Back up data	Detect loss of data and inconsistency in data	Construct new CMP
Lack of Early Identification of Defects	Include verification as a process in every phase of the project	Record the time spend on correction of the products	Use modules pre-developed to fill the gap
Low Application Experience	Thorough search and pre-work on tools	Detect the time required to complete a task	Ask for assistance from experienced people
Insufficiency of Resources, Lack of Quality of Tools	Pre-work and experiments on tools to be used	Record the effort spend on integration and usage of outer software	Choose another technology that will cause the least loss

A Appendix

A.1 Tool Comparison Table

Functionalities/Tools	**Scene PD 2**	**Easy Street Draw 3.0**
Primitive 2D objects (E.g. cars,tracks, lorries, trees, traffic lights etc.)	Support for primitive objects in traffic area and lots of objects related to other specialized areas (E.g.architecture, military, electronics, marine etc.)	Support for primitive objects in traffic area but not much support for other specialized areas
Self road creation/ template road modification	Self road creation is not supported, restricted modification to templates	Self road creation is not supported, restricted modification to templates
Object Addition, Modification and Creation	Additions supported in a drag and drop manner, insertions can be done via importing image files (JPEG , BMP, TIF)	Additions supported in a drag and drop manner
Drawing Template Support	Available	Available
Save/Load Options	Available for both a self-created object and also traffic case file produced	Available for traffic case file produced

Functionalities/Tools	3D Eyewitness 7.9	Crash Zone Version 7.0	MapScenes Pro SP	Vista FX 2
2D Drawing Editor	All tools taken into consideration has a 2D Drawing editor part			
Environmental Design				
2D model objects (E.g. cars, tracks, lorries, trees, traffic lights etc.)	Support for primitive objects in traffic area	Support for primitive objects in traffic area and lots of objects related to other specialized areas	Support for primitive objects in traffic area and little support for other specialized areas	Extensive database (e.g. a car object with its brand and model)
Self Road Creation and template road modification	Supports self creation and also has templates that can be modified via a checklist	Supports high modification and self creation	Self creation and additional ramps support	Highest flexibility in self creating roads
Object Addition, Modification and Creation	By Modeler part of tool any object can be created and inserted, object additions to scene are done in a drag and drop manner	Additions are done in a drag and drop manner	Additions are done in a more complex way	Additions are done in a drag and drop manner
Image and Animation Support				
2D Simulation	Unavailable	Unavailable	Unavailable	Available
Object Control and Interactivity	-	-	-	Object control beforehand, no interactivity at runtime
Transition from 2D to 3D images	Available via screenshots	Available	Available	Available
File Support				
2D file save/load/open	Available	Available	Available	Available
Output file formats supported	BMP, JPEG	AutoCAD file format	BMP	BMP, JPEG
Help Support	Available	Available	Available	Available
Template Support	Available	Available	Available	Available
3D Viewing Editor				
Environmental Design				
3D Model Support	Available, every 2D model has a 3D model version	Available	Available	Available
Animation and Image Support	Available	Unavailable via the tool, but can be done using other presentation tools	Unavailable	Available
Camera View Support (perspective, pan, zoom)	Available from any point	Available from any point	Available from any point	Available from any point
Object Control and Interactivity	Available via 3DMovieMaker part of the tool, movie created by a sequence of screenshots	Available	Available	Available, Object control done beforehand at 2DEditor
Physics (closeness to real world)	At low level, especially problems at gravity issues	At medium level	Good at gravity, other properties are inapplicable	Excellent
AI Support	Unavailable	Unavailable	Unavailable	Unavailable
Rendering (lightning, shadowing, textures)	Shadowing and lightning at medium level, texturing supported	Only texturing	Available	Available
Audio Support	Unavailable	Unavailable	Unavailable	Unavailable
File Support	Available	Available	Available	Available
3D file save/load	Available	Available	Available	Available
Output file formats supported	AVI and MOV formats for animation	-	-	-
Help Support	Available	Available	Available	Available
Template Traffic Cases Support	Available	Available	Available	Available

A.2 Survey Questions

1. Do you know how to use computer? How much you need to use computers in your daily life?
2. In the lectures at your course, do you benefit from computers?
3. Is the education in your course supported with simulator or animation tools? If it is, how much beneficial are these tools in your opinion?
4. What properties are missing or what properties are essential to exist in those simulator tools?
5. Which one do you choose in education supported with computers?
   - the education of traffic rules on a road in a predisposed map
   - the education of traffic rules on a road in a map that is newly constructed or previously constructed and saved by the trainer himself
6. Which one is more illustrative when describing various cases in traffic?
   - viewing the road from the driver seat
   - watching traffic from top view
   - both of these
7. In an educational traffic tool could you put the items below that are essential for the system in priority order?
   - adding new traffic cases
   - explaining the conditions occurred in the current traffic case by inserting audio or texts
   - realistic vision
   - examining the trainees by various tests
   - adding new traffic signs
   - adjusting traffic density, air conditions, road conditions, velocity of the vehicles
8. How much do you think that designing your own traffic cases is beneficial?
9. Are there any other properties that can be essential to exist in this tool? If there is, what are they?
10. How should the screen that you are going to interact with be designed so that you can use with ease?

A.3 Data Dictionary

Name	User Inputs
Aliases	None
Where used / How used	User (output) Event Handler(input)
Description	User Inputs = [keyboard Inputs — mouse Inputs] keyboard inputs = [characters — numbers — control keys] characters = [A — B — C —...— Z] number = [0 — 1 — 2 —...— 9] control keys = [Esc — Tab — Enter — Back Space — Caps Lock — arrows] mouse inputs = [left mouse click — right mouse click]

Name	Case Data Info
Aliases	None
Where used / How used	Event Handler (output) 2D Editor (input)
Description	Case Data Info = "all the information that the user supply needed to construct the demanded traffic cases and the environment "

Name	Animate Request
Aliases	None
Where used / How used	Event Handler (output) Simulation Loop (input)
Description	Animate Request = "request to animate a predisposed traffic case in TraffEdu Viewer"

Name	Save/Load/Open Request
Aliases	None
Where used / How used	Event Handler (output) File Handler (input)
Description	Save/Load/Open Request = [save request — load request — open request] save request = "request to save the current prepared environment in TraffEdu Editor" load request = "request to load pre-prepared environment in TraffEdu Editor" open request = "request to open new environment in TraffEdu Editor"

Name	2D Case Data
Aliases	None
Where used / How used	2D Editor(output) File Handler (input)
Description	2D Case Data = "all data taken from TraffEdu Editor representing the constructed traffic case which will be used in TraffEdu Viewer"

Name	2D Case Data Request/Extracted 2D Case Data
Aliases	None
Where used / How used	File Handler(output) 2D Editor(input)
Description	2D Case Data Request/Extracted 2D Case Data = [2D case data request — extracted 2D case data] 2D case data request = "request to send all data representing the current constructed traffic case in TraffEdu Editor" extracted 2D case data = "data extracted from saved file to be used in constructing TraffEdu Editor environment"

Name	Data_To_Store
Aliases	Data_To_Load
Where used / How used	File Handler(output) Repository2(input)
Description	Data_To_Store = "Special format file to be stored in Repository2"

Name	Data_To_Load
Aliases	Data_To_Store
Where used / How used	Repository2(output) File Handler(input)
Description	Data_To_Load = "Special format file to be taken from Repository2"

Name	2D Graphics Output
Aliases	None
Where used / How used	2D Editor(output) User(input)
Description	2D Graphics Output = “graphs, figures, shapes, text images,GUI seen by the monitor”

Name	3D Case Data Request
Aliases	None
Where used / How used	Simulation Loop(output) File Handler(input)
Description	3D Case Data Request = “request for receiving the 3D case data representing the predisposed traffic case”

Name	Extracted 3D Case Data
Aliases	None
Where used / How used	File Handler(output) Simulation Loop(input)
Description	Extracted 3D Case Data = “3D case data to construct the 3D animation in TraffEdu Viewer”

Name	Audio Output
Aliases	None
Where used / How used	Audio(output) User(input)
Description	Audio Output = “sound which can be heard by the user’s speaker”

Name	3D Graphics Output
Aliases	None
Where used / How used	Render(output) User(input)
Description	3D Graphics Output = “all the graphics output for the animation seen by the monitor”

Name	Models/Textures
Aliases	None
Where used / How used	Repository1(output) Render(input)
Description	Models/Textures = [models — textures] models = “created models which are used for preparing an animation in TraffEdu Viewer” textures = “color and material properties applied to a model”

Name	Audio Data
Aliases	None
Where used / How used	Repository1(output) Audio(input)
Description	Audio Data = “audio files for the TraffEdu Viewer”

Name	Audio Info
Aliases	None
Where used / How used	Simulation Loop(output) Audio(input)
Description	Audio Info = “information about the audio data to be used”

Name	Render Request
Aliases	None
Where used / How used	Simulation Loop(output) Render(input)
Description	Render Request = “request to render the specified 3D model”

Name	Rendered Model
Aliases	None
Where used / How used	Render(output) Simulation Loop(input)
Description	Rendered Model = “rendered form of specified 3D model”

Name	Object State
Aliases	None
Where used / How used	Simulation Loop(output) AI(input)
Description	Object State = “a collection of characteristics that can be used to define the status of an object (an issue, an event, or a condition)”

Name	Object Next Behavior
Aliases	None
Where used / How used	AI(output) Simulation Loop(input)
Description	Object Next Behavior = “behavior of the object to the current situation”

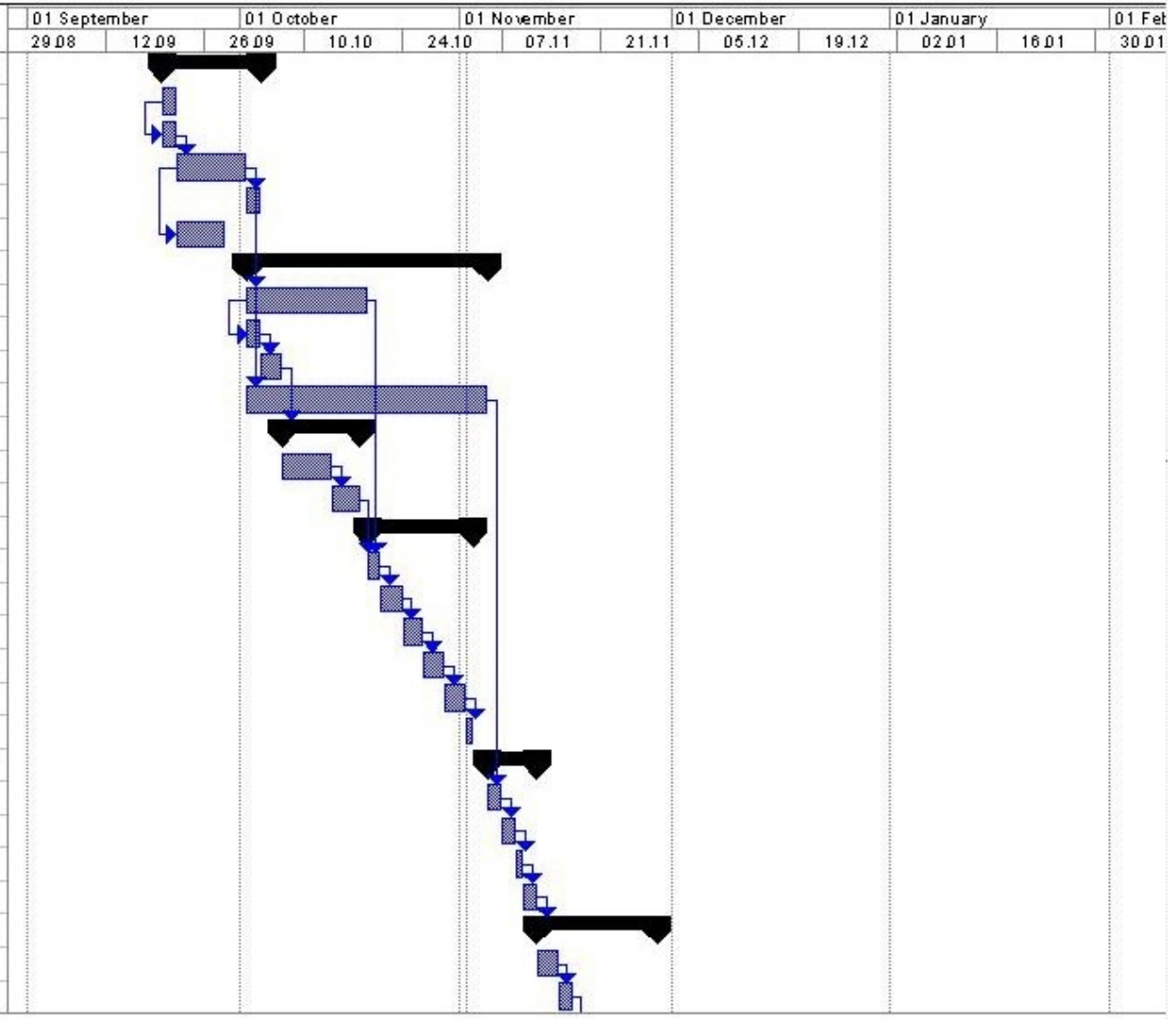
Name	Object Dynamics
Aliases	None
Where used / How used	Simulation Loop(output) Physics(input)
Description	Object Dynamics = “variables determining an object like mass, velocity, friction, etc...”

Name	Physics Applied Bodies
Aliases	None
Where used / How used	Physics(output) Simulation Loop(input)
Description	Physics Applied Bodies = “predicted effects on an object under different conditions”

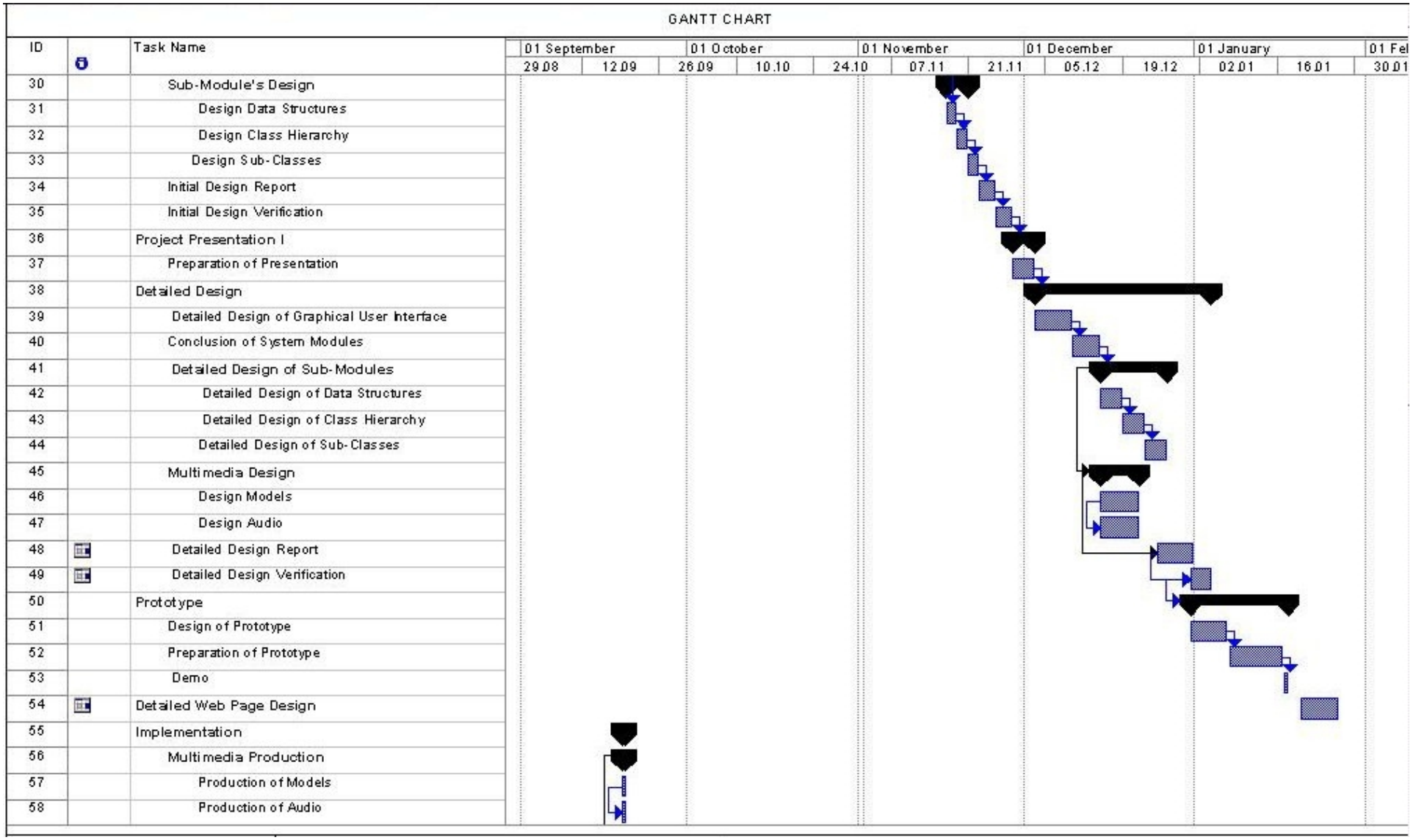
A.4 Gantt Chart

GANTT CHART

ID		Task Name
1		Team Construction
2		Team Organization
3		Team Agreement Report Preparation
4		Problem Definition
5		Proposal Report Preparation
6		Initial Web Page Design
7		Research and Survey
8		Market Search
9		Survey Conduct and Interviews
10		Literature Survey Report Preparation
11		Technical Research
12		Scope
13		Scope Outlining
14		System Main Functionalities Determination
15		Requirement Analysis
16		Conclusion on Scope
17		Conclusion on System Functionalities
18		Initial Modularization of System
19		Review on Requirements
20		Requirement Analysis Report
21		Requirement Analysis Verification
22		Development Tools
23		Further Research on Software
24		Conclusion on Software
25		Installation of all Software
26		Experimentation of all Software
27		Initial Design
28		Design Graphical User Interface
29		Detailed Modularization of System



GANTT CHART


ID
Task Name
01 September
01 October
01 November
01 December
01 January
01 Fel
29.08
12.09
26.09
10.10
24.10
07.11
21.11
05.12
19.12
02.01
16.01
30.01
30 Sub-Module's Design
31 Design Data Structures
32 Design Class Hierarchy
33 Design Sub-Classes
34 Initial Design Report
35 Initial Design Verification
36 Project Presentation I
37 Preparation of Presentation
38 Detailed Design
39 Detailed Design of Graphical User Interface
40 Conclusion of System Modules
41 Detailed Design of Sub-Modules
42 Detailed Design of Data Structures
43 Detailed Design of Class Hierarchy
44 Detailed Design of Sub-Classes
45 Multimedia Design
46 Design Models
47 Design Audio
48 Detailed Design Report
49 Detailed Design Verification
50 Prototype
51 Design of Prototype
52 Preparation of Prototype
53 Demo
54 Detailed Web Page Design
55 Implementation
56 Multimedia Production
57 Production of Models
58 Production of Audio

GANTT CHART

ID	Task Name
59	Implementation of Sub-Modules
60	Integration of Modules
61	Verification
62	Test Case Design
63	Test Case Implementation
64	Run Modular Tests
65	Run System Integration Test
66	Project Presentation II
67	Preparation of Presentation
68	Documentation
69	Preparation of User Manuals

