Group Diary

Meeting Minutes #1

Date: 27/3/2015

Time:

Agenda

* Meeting with new advisor for new project

Minutes

* Today we met up with our new advisor to talk about the possible projects that we can work on. The previous project seems too difficult due to the lack of expertise in the game development field. We listed a few possible projects that we are interested in and capable of doing from a list of projects. Two projects that stood out was Body Systems and Car systems, both bring distributed systems project.
* We decided to take on the Car Systems project.

Action Item

* Thinking of possible ideas/ways for implementation of software

Meeting Minutes #2

Date: 30/3/2015

Agenda

* Discussion on the project, Car Systems

Minutes

* We discussed how the system would look in terms of its interface. Buttons which surrounds a car represents different parts of the cars, and detailed description is given for each parts.
* For later parts of the project, as we simulate the processes in a car, for example refuelling the car, different parts of the car which are required for the process are shown and a simple animation of sort will show how the process is done.
* The implementation of a quiz to test the users on his knowledge on cars could be implemented in the system.
* Other than learning the processes in a car, the software also aims to teach the users things such as what each dashboard symbol represents and much more.

Agenda

* Learning of Visual C++
* Requirements
* Presentation

Meeting Minutes #3

Date: 2/4/2015

Agenda

* Review on presentation and prototype

Minutes

* Dyalan and Elle showed the prototype they have been working on to Luke. Reconfirmation on what program should be used was also directed towards Luke. Visual C++ and Qt were both being considered for the development of the project. Much more learning would be required for using Qt compared to Visual C++. We decided to stick with Visual C++ for now.
* Kapil then showed Luke the presentation that he has been working on and modified from the previous project. Discussions were brought up on things that might require changing. This allow for further understanding of the project in hand for all of us as a group and how we will soon approach it.

Action Item

* Continue development of prototype
* Presentation and requirements

Meeting Minutes #4

Agenda

* Meeting up with advisor to discuss what to do from this point onwards.

Minutes

* Elle and Dyalan clarified on the resolution to be used for the program. Pictures to be used for the interior of the car could possibly be found on the manufacturer’s website.
* For the quiz, a different car model might be used to prevent users from memorizing parts of the car to answer the quiz.
* The software should have multiregional support, different components of the car is named differently in different countries (e.g boot and trunk). There may also be a difference in which side the driver drives on and should be included.
* We then discussed the deliverables needed for the end of the semester, which includes design documents such as class diagrams and use cases, data dictionary, a short script on how the threading for the simulation would be implemented, test methodology (black box and white box testing), assumptions that can be made, user acceptance testing, convention naming variables(?), design choices, load testing.
* If possible we would try to complete the first phase of the software and also a framework for the quiz.
* We then discussed on how implementation of the infographic section can be done, highlighting different components of the car when a name is clicked on. This process might involve photoshopping a lot of photos. Simple but tedious and time consuming.
* Allocation of work, Wee will work on use case diagram, Elle and Dyalan on the user manual, Kapil on documentation.

Action Item

* Use case diagram and description (Wee)
* User manual (Elle and Dyalan)
* Documentation (Kapil)

Meeting Minutes #5

Agenda

* Meeting up with client to for a brainstorming session

Minutes

* We first discuss about the quiz section of the software, addition of an instruction page that will be displayed to the user upon clicking “Quiz” from the main menu which allows the user to be able to prepare before actually taking the quiz; instead of a submit button to confirm submission of answers by the user, we can use next buttons (clicked using mouse) to go the next question and in addition to that we will have boxes at the bottom part of the quiz which will represent number of the question that can be clicked to jump to different questions (different colours to represent different situations, different colour for questions that are not answered, different colour for questions that are answered and different colour for the question the user is currently in); upon submission, the program will have to check if the user has answered all the questions; if the user decides to leave mid-way through a quiz, an abort button will be available for the user to click and it will bring the user back to the main menu; the button to exit the program on the top right should be hidden so that the user will not be able to exit that way (this is to prevent any dynamic memory initialized in the system from being lost); for the review of the quiz, the user will be able to view all the questions that have been answered, for wrongly answered questions, the box that represents the question that is answered wrongly will be coloured differently so that the user will be available to click and review the questions that he/she has answered incorrectly by clicking on it; we should try to make the system register keyboard inputs by the user so that he can navigate through the questions and also at the same time answer questions; we will also have 10 drag and drop questions in addition to the 10 MCQ; the order of the answer for a question will change; we will also have parameterized questions (context is the same, the meaning of the question is the same but expressed in different words, the answer for the question will also be the same);
* The screen size should not be full screen due to the different screen sizes people will have so for now we have decided that the screen size would be set to 1024 \* 768.
* For the infographics part, to register a mouse click of the user on a particular component of the car can either be done by placing a square box on that particular component, or checking the mouse click coordinates against the placement of the component on the screen (hard to express it in words) (quiz interface for drag and drop will use the similar interface); for the infographics part, there will be buttons up, down, left and right, starting from the default view, clicking up will zoom in one layer and by clicking left and right you change views.
* For the videos part, we can use an embedded player or allow the user to open up their local media player instead of going through via Youtube as it will require internet connection. For the embedded player, we will try to darken down the back and put the video layout on top.

Action item

* Use case description (Wee)
* Sequence diagram and communication diagram (Dyalan)
* User manual (Elle)
* Documentation (Kapil)

Meeting Minutes #6

Agenda

* Another brainstorming session

Minutes

* Kapil confirmed the documentation needed for the final submission of the first semester and talked briefly on the website. Things to finish up include class diagram, data dictionary and state diagrams (There is actually more than this).
* We started talking a bit on the sequence and communication diagram for the fuel simulation. We discussed on how there would be a normal simulation of the car upon clicking on any simulation.
* As we discussed, we come to an agreement that there would be a persistent UI for the simulation part, where a dashboard would be placed permanently on screen, and any change in data would change accordingly on the dashboard. Things that will remain persistent on the screen would include the dashboard, the start and stop engine button, the speed bar and also the incline (terrain, increase angle [probably needs better wording]) bar.
* We also discussed on the simulation of the fuel system. For example, if the user wishes to fill petrol into his car, he/she would need to reduce the speed of the car and then turning off the engine (probably the off button) before being able to fill his/her car.
* The speed of the car will increase gradually when being displayed on the dashboard (instead from 0 – 40 instantly)
* A normal flow of the system when being used by the user started from the menu, clicks the simulation, the system shows the persistent interfaces (dashboard, speed bar, incline bar, start and stop engine), user starts the engine, speed increases, dashboard updates accordingly. User now wants to try how the increased incline would affect the speed of the car and the rate of fuel consumption, user would be able to see the change in speed on the dashboard but to view the rate of fuel consumption, user clicks on “Fuel system” and would be able to see in details the changes that occur in the fuel system.

Action Item

* Use case (Wee)
* Sequnce and comm diagram (dyalan)
* User manual (Elle)
* Documentation (Kapil)