**Informatics Institute of Technology**

ECSI503 – Software Development Group Project

Software Requirment

Specification

* K.D. Isuru Amantha Siriwardena – 2014217 / w1560257
* H.A.L. Pradeep – 2014177 / w 1560247
* Salitha Lakshan Dassanayake – 2014261 / w1560206
* Andrew Eugene – 2014214 / w1560149

**Project summary**

You’re reading this now because you received proper education and overcame any disorders you had. But what if your growth was neglected while you had some disorder no one found out? That is why we plan to develop this application which is very user-friendly so even if you lack knowledge in children’s health this application will guide you through. This is much of a service rather than a commercial application. Parents tangled in their busy lifestyles will find this application a useful guide in keeping track of their children. Parents could get in touch with a doctor and get their question answered very easily. Just like browsing your social network daily you could visit our application and keep in touch with your bookmarked doctors and also learn facts you could’ve missed in the health field.

**Table of content**

|  |  |  |
| --- | --- | --- |
| Figure No | Description | Page no |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**List of figures**

|  |  |  |
| --- | --- | --- |
| Figure No | Description | Page no |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Content**

Assumption we made about the problem domain …………………………… 03

Context diagram …………………………………………………………....... 04

Use case diagram ……………………………………………………………. 05

Complete use case specification ………………………………………….…. 06

Sequence diagram …………………………………………………………… 07

Functional Requirement ……………………………………………………... 09

Class diagram ………………………………………………………………… 10

Class diagram description ……………………………………………………. 11

Web site ………………………………………………………………………. 12

**Assumptions we made about the problem domain and problem scope**

* Chapter overview: All the assumptions we have made to create this application scenario as well as the problem domain and the problem scope.

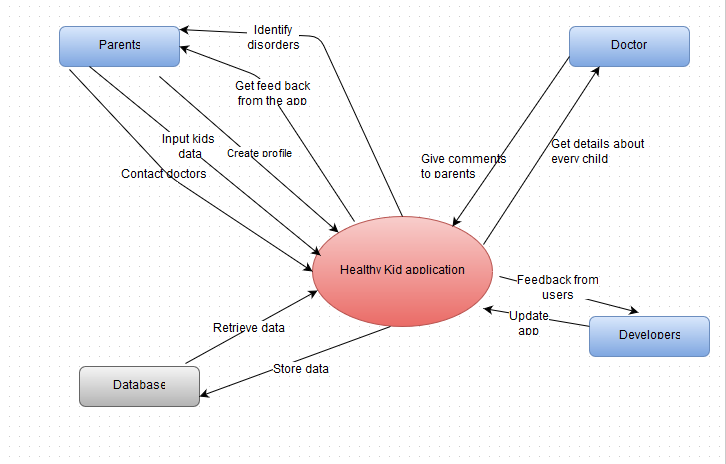
Assumptions can also be called ‘conjectures’ or simply ‘guesses’ regarding a particular subject. All assumptions made on our problem domain is important as it prepares us for the oncoming activities. Below are a few assumptions we recognized related to our problem domain.

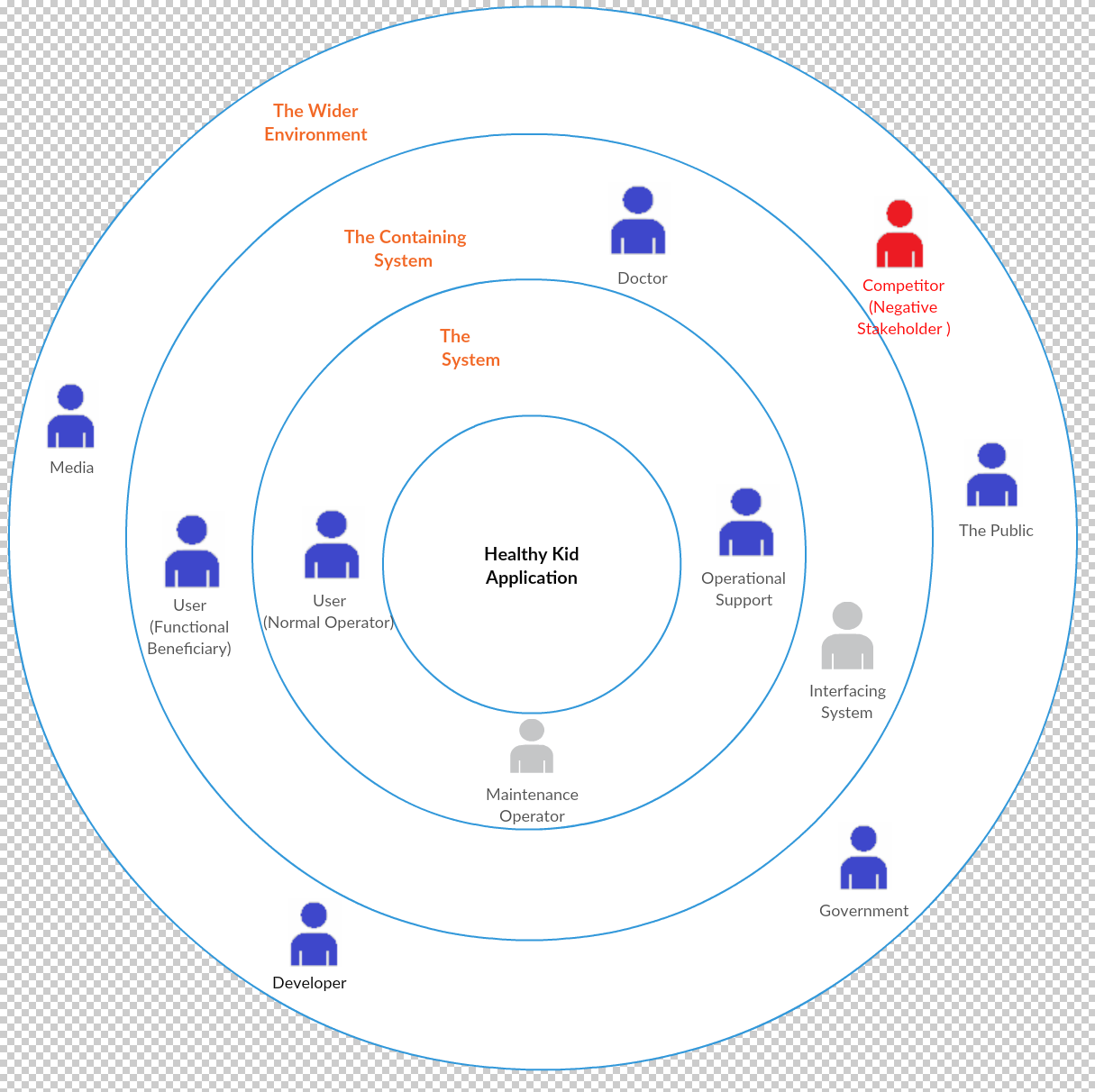
One assumption we could come across is that ‘Users acting as knowledgeable in using a smart phone.’ To justify this:Modern generation is extremely capable of handling a mobile phone. However, when it comes to the older generation the question yet remains as to whether they could manage. The exposure given by the younger generation to the older generation is quite high thereby having the chance of knowledge being shared. Since our application has a target on health volunteers (who are usually at the age of 20’s) it is provable that a smart phone could be used. Apart from that another assumption is ‘Health workers giving feedback to parents’ questions properly. So this can be justified since questions are openly available to be answered there is a chance of it being answered within a short period. Health workers and volunteers will have the ability to respond to these questions and notifications will indicate health workers when a question is asked which will speed up the process. Yet another assumption is that ‘Parents being able accurately identify children disorders’. The application will provide precise details of identifying a disorder. Rather a step-by-step process of recognition. Thereby allowing any user who could read and manage a user-friendly interface the ability to identify any disorders within their child. Other assumptions would be one such as health workers appearing as knowledgeable in maintaining online health records of children. Since most health workers are of mid age technology might not be their best interest. However the application is coded into a very user-friendly interface where even a beginner could understand. Other than that we assumed a few more points:

* Most users being new parents
* Users having more favor towards technology
* Health workers agreeing to support needing parents online
* Parents providing accurate details regarding their children for identifying any disorders
* Ability to afford a windows smart phone
* Chapter summary: All the assumptions we have made so far.

**Context Diagram**

* Chapter overview: Context diagram and onion model of our system.





* Chapter summary: Demonstrate a sketch of the system’s context diagram and the onion model.

**Use Case Diagram**

* Chapter Overview: Use case diagram of the system which shows the actors, use cases, associations between them and the boundary. Then the use case description about each use cases.

**A complete use case specification**

* In this use case diagram, we have drawn 3 actors which are parent, doctor and the database. Since we are developing our mobile application for the children, we can’t let them to input the relevant details. Therefore, on behalf of the child, mother will input the data. That’s why mother is acting a major role in the system. We are hoping to store all the details in the Microsoft Azure cloud, so the database will play another major role in the system. Our system will always deal with the database and retrieve the data. Finally, our other main actor is the doctor. If the parent would like to contact the doctor, within one click they can connect with the doctor and will be able to find some solution regarding their problem.
* We have created the create profile use case because it will allow to user to input the data and then create a profile about the child. Application doesn’t make any sense without the profiles. Every child must have an each and dedicated profile to use this application. Enter the validate data is necessary in this use case.
* In the check development milestone use case, it is generalized with 4 more use cases which are identifying hearing problems, identifying visual problems, identifying speaking problems and identifying behavior problems. Those are the main 4 features in this application. Check development milestone use case should always connect with doctor and the database because the database will do the analyzing part and other stuff.
* Access learning tools only valid for Pro version. It will enable an alphabet with sounds which will really helpful to the child.
* Access child health development use case is another important thing in our application. Parent must include data such as height, weight accurately to get the maximum benefits out of it.
* View profile is another use case which is important to user to maintain the user profile. By using it user can easily edit the profile.
* The login doctor side use case represents the doctor’s login to the system.

|  |  |
| --- | --- |
| Name | Access timeline |
| Brief description | This use case will show the access to the time line. That means the main menu of the user interface |
| Actors | Parent |
| Pre-conditions | Parent must logged into the system |
| Basic flow | Parent login to the system. Then parent will get the timeline. From there parent can add journal entries, add photos and etc. |
| Alternative flow |  |
| Post-conditions | The system shall save all the details |

|  |  |
| --- | --- |
| Name | Create a profile |
| Brief description | This use case represents the creation of the user profiles. If the parent have more than one baby, multiple accounts can be created. |
| Actors | Parent, database |
| Pre-conditions | Should know the basic details about the baby |
| Basic flow | Create a baby’s profile by entering the basic details of the baby such as weight, height, name etc. and click and save it in the cloud services. |
| Alternative flow |  |
| Post-conditions | System must save all the details in cloud |

|  |  |
| --- | --- |
| Name | Login doctor side |
| Brief description | This is represents the user login of the doctor. This is not a mobile application, but a desktop application |
| Actors | Doctor, database |
| Pre-conditions |  |
| Basic flow | By providing necessary details doctor can create and login to the system. |
| Alternative flow |  |
| Post-conditions | System should save the all details and provide the necessary details about the babies from the cloud. |

|  |  |
| --- | --- |
| Name | Child development milestone |
| Brief description | This is a one of the important use case. All the major features are indicates here. |
| Actors | Parent, doctor, database |
| Pre-conditions | User must logged into the system |
| Basic flow | First user has to log into the system. Then by him/her desires choose what to identify in the baby. Then it will show the proper guideline. |
| Alternative flow |  |
| Post-conditions | Save all the data |

|  |  |
| --- | --- |
| Name | Access child health development card |
| Brief description | Some of the major important use case among others. It will show the details about the child health development card |
| Actors | Parent, database, doctor |
| Pre-conditions | Parent must be logged into the system |
| Basic flow | After login into the application, parent should include the all details regarding the child’s health care development process. Then all the details will backed up in the cloud. |
| Alternative flow |  |
| Post-conditions | Data should save in the cloud services. |

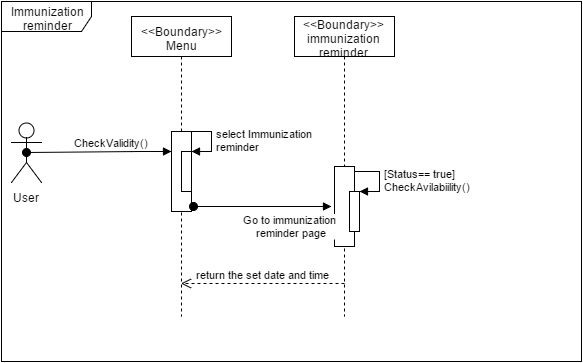
|  |  |
| --- | --- |
| Name | View profile |
| Brief description | View profile and edit profile |
| Actors | Parent, database |
| Pre-conditions | Logged to the system |
| Basic flow | First of all user must logged into the system. Then if you want to modify the details of the child, go to view profile and modify the relevant data. |
| Alternative flow |  |
| Post-conditions | Update the data |

* Chapter summary: Completed use case diagram and description about each use cases.

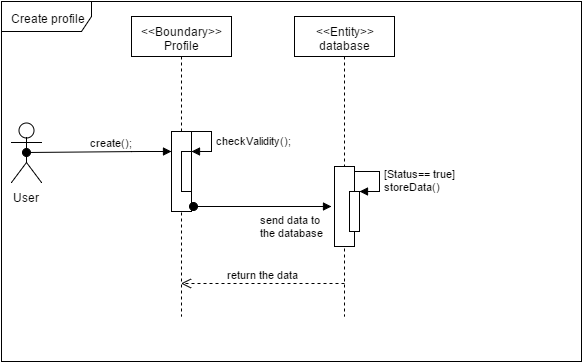
**Sequence Diagram**

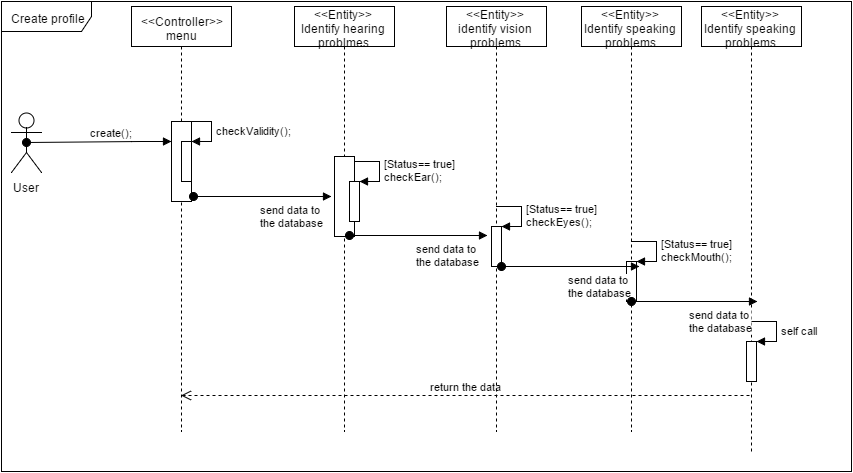
* Chapter Overview: A sequence diagram for the key use cases and a short description.

1. Immunization reminder – This sequence diagram will show about the immunization reminder. Actor is the user. User must login to the system first through a validation system. Then user will get the main menu. From the main menu you can select the immunization reminder option. User can select this option over and over again. Then it will prompt the immunization page and from there, you can set the reminder by setting a specific time and a date. After that it will return to back to the main page.



1. Create user profiles – This diagram will show the creation process of the user profiles. Again user has to log to the system first. Then by providing necessary details user has to create the profile for their baby. Then the all the data will send to the cloud service via the internet. If the validation process return true, then the data will backed up in the cloud service.



1. Check development milestones – This is the important sequence diagram among others. First of all user has to create a profile and log in to the system. In that process system will check the validity. Then system will send the updated data to the database based on the user inputs. As an example if the user choose to identify the hearing problems, system will give the particular guideline and get the inputs from the user and send the data to the cloud services analysis them. This can apply to the other life lines as well.

* Chapter summary: Sequence diagrams for key components and a small description.

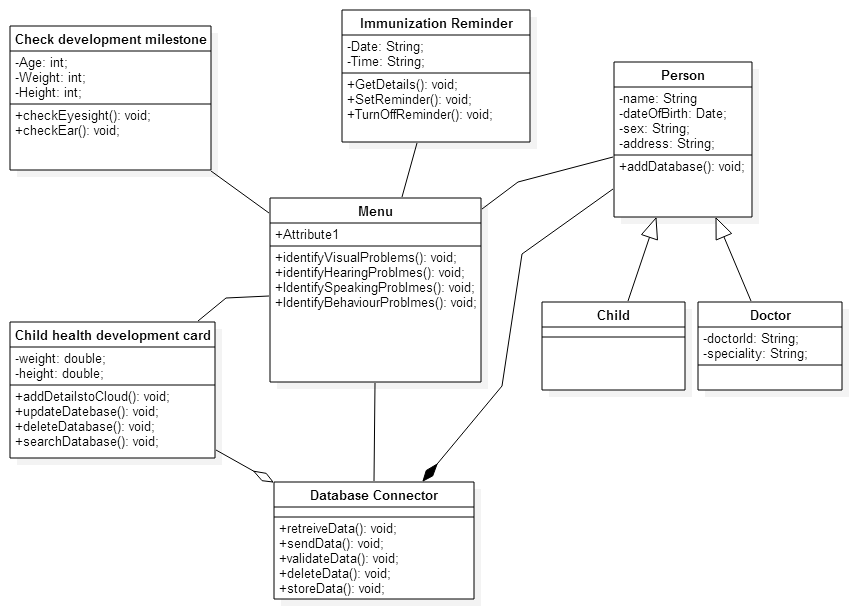
**Functional System Requirement**

* Chapter Overview: Functional system requirements which will important to the system.

1. The System shall be able to issue immunization reminders to users.
2. The System shall allow users to upload child development data.
3. The System shall allow users to retrieve child development data.
4. The System shall allow users to identify children’s hearing problems
5. The System shall provide suggestions to users on hearing disorders.
6. The System shall allow health workers to maintain a digital child health development record.
7. The System shall allow users to identify children’s vision problems.
8. The System shall allow users to identify children’s speaking problems.
9. The System shall warn users regarding unusual children behavior.
10. The System shall allow users to ask questions related to children’s health.
11. The System shall educate parents of food and its nutrition.
12. The System shall allow parents to view the English alphabet in colorful font.
13. The System shall prompt incoming FAQ messages from doctors.
14. The System shall allow users to purchase full version of application
15. The System shall allow users to respond incoming FAQ messages.
16. The System shall allow health workers to conduct screening.
17. The System shall provide nutrition levels of children to health workers.
18. The System shall prompt health workers for weight and height when nutrition level is requested.
19. System shall be able to prompt users with random tips regarding child health
20. System shall war users regarding identified disorders
21. System shall record all developmental records input by users
22. System shall allow users to search health tips categorized by disease

**Class diagram**

* Chapter Overview: A completed class diagram for the system and a description about it.



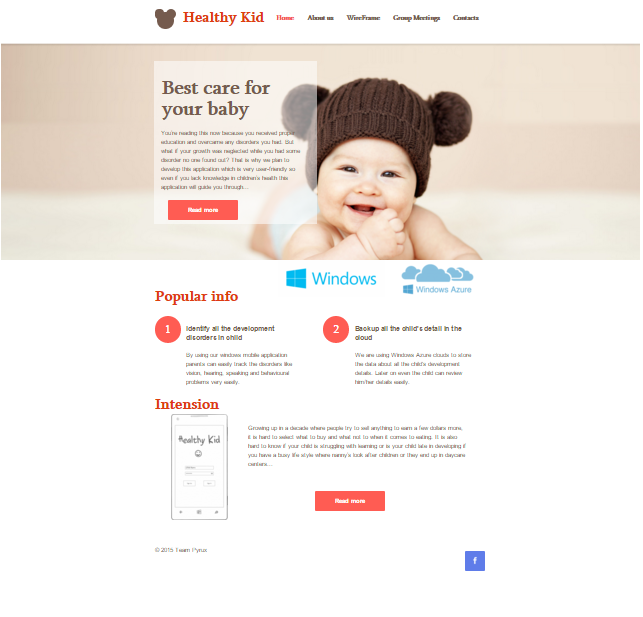
**Class diagram description**

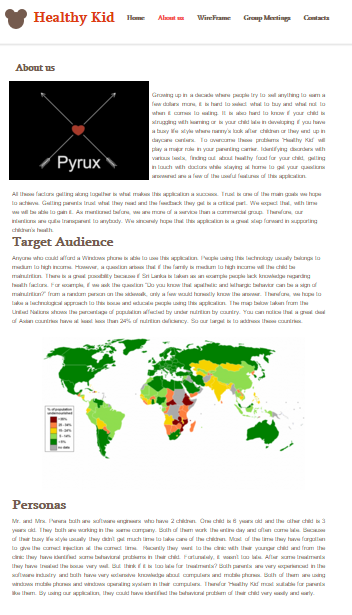
* If we consider the class diagram according to the ECB model, Menu class will be the controller. Menu class is the core of the app and it will control all the features in the application. It has 4 methods such as identify visual problems, identify hearing problems, identify behavior problems and so on. Other all four class will associate with the menu class. That’s why we have created association connection between other class diagrams.
* In this class diagram we have defined the person class a super class and also child and doctor as the sub classes. The connection between the sub class and super classes have shown by a generalization connection. Child and doctor automatically inherited with super class attributes. Apart from that doctor class has special attributes such as doctor ID and specialty.
* The immunization reminder class set and update the reminders about the immunizations. Attributes such as date, time will pass as the parameters.
* The other important class is database connector class. There is composition connection between the person and database controller. Person doesn’t exist without database in our assumption.
* Child health development class is also important. The attributes such as weight, height must input by the user and based on those measurements, child health card will be updated and it will show the updated chart. And also it will update the database as well.
* Check development milestone is the most important class in the scenario. It has methods like check eye sight, check ear. And it will be having association with the menu class.
* Chapter summary: Completed class diagram and a short description describing all of the classes.

**Appendix**

Our main web page URL: <http://healthykid.tk/>

1. Home Page



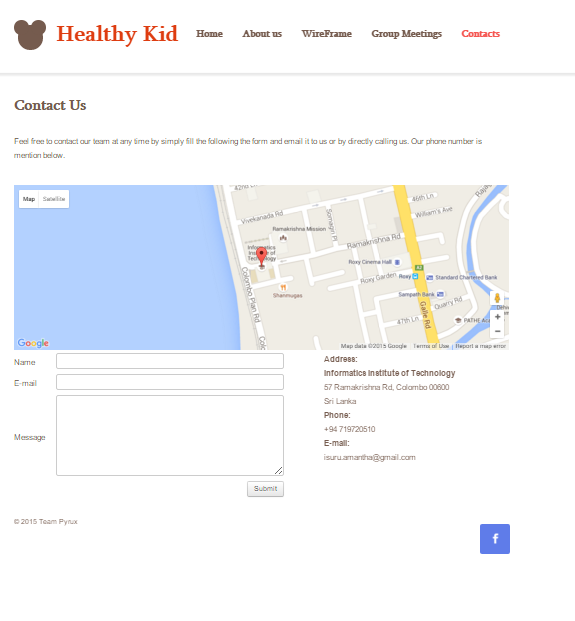
1. About us page
2. Wire Frame page



1. Group meeting Page



1. Contact Us Page



ECSC503 Software Development Group Project

Group Contract Template

**(to be signed by all members of the group)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Full Name and ID (please type) | | | Contribution  (1-100%) | Signature |
| Surname | First Name | ID |  | |
| Siriwardena | Isuru | 2014217 | 100% |  |
| Hatharasinghe | Pradeep | 2014177 | 100% |  |
| Lakshan | Salitha | 2014261 | 100% |  |
| Eugene | Andrew | 2014214 | 100% |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Notes:**

1. **Indicate the contribution of each member of the group towards the final output allocating percentages from 1% to 100%.**
2. **Practical contribution of individual members - 1% = unreliable, unwilling to undertake boring tasks, missed deadlines, no practical contribution to the final output, etc, and 100% = reliable and willing in all designated tasks, completed all designated tasks towards final output**
3. Example 1**: you may decide that each member of the group has contributed equally 100%, in which case you write 100% against each name.**

Example 2**: you may decide that “student A” has contributed only 50% and all other students equally 100%, in which case you write 50% against student A and 100% against each other name.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Coursework element | What is assessed | Assessment Criteria | Marks Allocated | Student’s Marks |
| Report  (80 Marks) | Assumptions you made about the problem domain and problem scope | Clarity of the assumptions and their relevance | 5 |  |
| A Context Diagram of your system | Quality of the Diagram and its correctness | 5 |  |
| A Use Case Diagram of your system | Quality of the Diagram, its completeness | 10 |  |
| A complete Use Case specification for all the use cases that are shown in your use case diagram | Completeness and Correctness of Use Case Specifications | 15 |  |
| A Sequence Diagram for the key Use Case specification and its short description | Quality of the Diagram, its completeness, and quality of descriptions | 5 |  |
| A list of all functional requirements | Quality of the written requirements | 5 |  |
| A Class Diagram including class attributes and class associations (i.e. domain model) | Quality of the Diagram, its completeness and its relevance to the sequence diagram | 15 |  |
| Description of the reasons to include each class in the design and how it fits into the design | Clarity of the description of the classes and reasons to include each of them | 10 |  |
| Written presentation skills & references | Quality of Academic Writing, Logic of the Written Presentation, Quality of Referencing | 10 |  |
| Oral Group Presentation of the Analysis and Design | Oral presentation skills | Quality of Slides  Logic of the Presentation  Time Management  Oral Presentation Skills | 10 |  |
| Project WWW site | Project WWW site management (A Blog) | Consistent Web site maintenance, WWW site content (minutes, summaries of group meetings, Diagrams and other material reflecting your progress with tutorial tasks) | 10 |  |
|  | Total possible marks 100 | | |  |