Software Design Tools and Diagrams

i) The diagrams that we are going to develop for the project development.

1) ER Diagram

ER Diagrams are developed to identify all the entities in the system, relationship between the entities and cardinality of their relationships. This diagram on the later stage helps to develop the database design.

2) Use Case Diagram

UML Use Case Diagrams can be used to describe the functionality of a system in a horizontal way. That is, rather than merely representing the details of individual features of your system, UCDs can be used to show all of its available functionality. Use case diagram gives more abstract view of the system.

3) Class Diagram

The class diagram is the main building block of object oriented modelling. It is used both for general conceptual modelling of the systematics of the application, and for detailed modelling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main objects, interactions in the application and the classes to be programmed. From the class diagram we can come to know about the detailed information related to the functionalities that we are going to implement and furthermore class diagrams can help us to modularize the system.

4) Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control. Activity diagrams are helpful in the development in the sense that they can help us to know each and every condition that is occurring and how to program it.

5) Sequence Diagram

A sequence diagram in a Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. It can be used to know the level of the coupling that is needed in the system and we can try to reduce it.

ii) Designing Tools that we are going to use during the development.

1) Rational Rose

Rational Rose is an object-oriented Unified Modeling Language (UML) software design tool intended for visual modeling and component construction of enterprise-level software applications. In much the same way a theatrical director blocks out a play, a software designer uses Rational Rose to visually create (model) the framework for an application by blocking out classes with actors (stick figures), use case elements (ovals), objects (rectangles) and messages/relationships (arrows) in a sequence diagram using drag-and-drop symbols. Rational Rose documents the diagram as it is being constructed and then generates code in the designer's choice of C++, Visual Basic, Java, Oracle8, Corba or Data Definition Language.

We are going to use Rational Rose tool for the creation of the UML diagrams listed above named Class Diagram, Use Case Diagram, Sequence Diagram and Activity Diagram.

2) MySQL WorkBench

We are going to use this tool for the ERD creation as it supports automatic generation of the ERD from the database. We would be developing rough ERD first and from that we are going to develop the database. From the database we are going to generate the ERD.