CHAPTER 4

DESIGN ASPECTS AND IMPLEMENTATION

In this section we are presenting in details about the design aspects and implementation of the PReWebD and PReWebN.

4.1 The Design Aspect

A Web application’s design is essentially its look and feel [71]. We have taken into account all the Web elements e.g. audience information, purpose and objective statement, domain information, Web specification and combine them to produce an arrangement for implementing PReWebD and PReWebN.

Both the applications have been developed by considering the profile of the Department of Electronics and Communication Technology (ECT), Gauhati University as the sample data. The CRUD operations are performed to generate the response. It is an operation in which data is created, retrieved, updated and deleted according to the user requirement. There are two types of account available for accessing PReWebD and PReWebN. The “USER ACCOUNT” which is available for the registered users that has limited access to PReWebD and PReWebN. The other one is the “ADMINISTRATIVE ACCOUNT” available for Administrator that has full access to the PReWebD and PReWebN. The basic working principle of PReWebD and PReWebN is shown in the flowchart of fig.4.1. The context diagram and first level Data Flow Diagram (DFD) [72] is shown in fig.4.2 and fig.4.3 respectively.

4.1.1 The Presentation of PReWebD and PReWebN

We have used the IDE to design, develop and implement the PReWebD and PReWebN. The use of IDE makes it easy for the designer to design because one can concentrate more on the BLo of the application. The CSS is used to layout the Web application to make it more users friendly and accessible. The layout is generated in a separate CSS file so that it can be linked to various pages of the application. This will
Fig. 4.1: The Flow chart for basic working principle for PReWebD and PReWebN
Fig. 4.2: The Context Diagram for PReWebD and PReWebN
Fig. 4.3: The First level DFD for PReWebD and PReWebN
make the layout design very easy, which can be changed without disturbing other pages of the applications.

4.1.2 The Framework

The Web tier has acted as an interface between the users and the BLo in PReWebD and PReWebN. As such, frameworks are used as a mechanism to transfer user data to the BL. We prepare a page layout common to many Web pages to make the layout and maintenance of the application easier. It is easier to design an application using MVC as well as layered structure. The MVC offers a clear separation of presentation with the BLo and the database. All the advantages of MVC, such as: modularity, anticipation of change, abstraction, low coupling, and high cohesion are used while developing the application. To achieve faster development, we use proven patterns and also reused the components.

4.1.3 The View Technology

A view is attached to the model and displays it in a form suitable for user interaction. In JSF framework, the view is represented by JSP while XHTML is used to represent the view in ASP.NET framework.

4.1.4 The BLo

It is a collection of classes which consist of different objects. The BLo objects are used to capture the data sent by the users and then processed accordingly. In the business tier, the objects have mainly performed to authenticate user while login, database insertion, updation, deletion, and search operation and the corresponding database tables are updated accordingly. Our main concern is how these objects may interact with each other and who can access and update them.

4.1.5 The Database Design

A good database design is one of the most important factors in the successful implementation of any database oriented system. The databases for PReWebD and PReWebN are designed by considering the fact to reduce the database anomalies e.g. redundancy, update anomalies, data inconsistency etc. In PReWebD, the database
connection is implemented using windows authentication. The authentication scheme and the connection string are configured in the Web.config file. We have written the ADO.NET code using vb.net to access the database. The database connection in the PReWebN is provided through the IDE services window.

Two sample database tables for PReWebD and PReWebN are shown in fig.4.4 and fig.4.5 respectively.

4.2 Technical Specifications

The technical specifications for PReWebD and PReWebN are given below:

4.2.1 Hardware Configuration

The hardware configuration for PReWebD and PReWebN is as given below:

PC: Intel® Pentium® Dual CPU E2200
Processor speed: @ 2.20 GHz
RAM: 1 GB
Memory space: 150 GB

4.2.2 Software Configuration

(i) For PReWebD

Web Server: IIS 5.1.
Operating system: Window XP Professional Service Pack 2.
Browser: Internet Explorer 6.1.
Network bandwidth: 128 kbps.

(ii) For PReWebN

Web Server: Apache Tomcat 6.0.18.
Database Server: MySQL 5.0.
Operating system: Window XP Professional Service Pack 2.
Software Platform: NetBeans 6.5.1.
Browser: Mozilla Firefox.
Network bandwidth: 128 kbps.
4.3 Implementation

The objective of our implementation is to combine the output from the design and planning process to create an effective Web. The implementation of PReWebD and PReWebN are prototype which is not released publicly, but available for analysis.

Two sample snapshots of the implemented web applications are shown in fig.4.6 and fig.4.7.

Next chapter of the thesis describes in details about the testing that has been performed on PReWebD and PReWebN.
Fig. 4.4: Sample data table for PReWebD

Fig. 4.5: Sample data table for PReWebN
Fig. 4.6: The snapshot of Insertion form for PReWebD

Fig. 4.7: The snapshot of inserting form for PReWebN