**Design and Development of an Integrated Enterprise System (IESWeb: Pronounced "iceWeb")**

**Bina Ramamurthy**

**CSE4/587 Information Structures** **Due Date: 4/3/2003 by mid-night.**

Consider the scenario described in Chapter 3 "Defining the Enterprise Architecture", page 81 onwards in your textbook. It is a comprehensive system involving many processes and functional modules: customer front end (sales only through e-commerce site), billing, order processing, purchasing, shipping, customer relations management (primarily through email about the status of their order), inventory control (text book calls it warehouse management) and accounting. See the figure 3.8, page 82. You are required to design and implement IESWeb to support this structure for any application domain of your choice: plant nursery, hardware store, etc. Your first task is to choose the domain you are familiar with so that you can analyze it and obtain the requirements easily. You will implement the complete solution using Java 2 Enterprise Edition (J2EE). Each one of you will have to establish a development environment for you to work on the project. Once the development is completed I plan to deploy all the working projects on a production environment that my server will host as a demo for the whole world.

1. Study and understand the enterprise integration scenario described in your text Chapter 3. Also Chapters 6-11 will provide a good reading for analyzing and deciding on the needs of the IESWeb.

2. Decide on the application model. For example, Java Server Pages (JSP) front end, EJB middle tier, Oracle database as the data tier. Or you may want servlet front end. Or you may want a DB2 database. But EJB middle layer and J2EE is mandatory for this project. Draw the block of the application model.

3. Since J2EE is the server environment for the project, download a copy of the J2EE SDK, and the accompanying tutorial and set up the development environment in your computer. See the J2EE tutorial at http://www.hi.is/pub/cs/2002-03/hv1/j2eetutorial Go through the Preface, Overview and Getting Started sections.

4. Decide the users of the system and develop the use case diagrams. This will provide the requirements for the user interfaces and the functionality. Design the user interfaces and implement the web module and test it separately on the J2EE environment. Lets call this module IESWebWAR. (WAR stands for web archive).

5. Analyze the data requirements for the IESWeb. Create an Entity-Relationship (ER) diagram to show the various data needs and dependencies. Create a SQL file to define the needed tables, and insert the data into table. Establish the IESWeb database.

6. Define on the processes, rules, entities, and transactions by object-oriented component-based design using one or more class diagrams. Implement these as business logic components with appropriate type of beans (session: state, stateful, entity, and message). You may need helper classes to support the beans. Test them individually. Assemble the

application using these beans. Call the application IESWebApp. It may have many JARs representing the beans. You may have to rename it when you submit it for the production environment since we want distinct name identifying the business domain you are dealing with in the application. We may also rename by using group numbers. We will discuss the details further during the lecture.

7. Connect the web module and the database module with the application logic module using Java Naming and Directory Interface (JNDI) methods.

8. You may use Java Mail for mailing people involved in the system and Java Messaging for communication between applications. For example, exceptions need to be handled using mail and messages in this project.

9. Deploy the application using appropriate security features, context settings, filters and other resource settings. This can be done from the deploytool as well as using scripts such as j2eeadmin provided with the reference implementation.

10. For the report (documentation) details follow the Project1 handout.