

Product Catalogue Application

The client is an international development agency that promotes the right to each individual to enjoy a life of health and equal opportunities. It helps in monitoring population data for policies and programs to safe birth, reduce poverty, fatal infections and diseases and to ensure healthy world.

The Vision

The objective of the Product Catalogue application is to enhance and increase capacity building by making product information more accessible to internal and external users. It addresses the need to provide product related information from system to users in a read-only, well presented fashion. The data related to various product parameters and vendors and their mutual relationship is made available by the Product Catalogue Application. The vision was also to make online catalogue with advanced search options and to filter out the necessary information about products and vendors, and creating technical foundations for future extensibility.

Application Overview

The client wanted a Product Catalogue Application with product and vendor search, data retrieval, user group configuration according to their access rights. The application provides capability to view details of the product, and information related to product that is vendor specific.

The application consists of three components as follows -

- **Product search Component**
 - Key drivers shaping this component are:
 - Free text based search
 - Search based on various parameters
 - Ability to represent product list in paged format
 - Ability to export data to Excel or PDF
 - Custom views of information based on the user group and the corresponding configuration for that user group
- **Vendor listing component**
 - Key drivers shaping this component are:
 - Ability to display vendor list in paged format.

- Free text based search
- **Access configuration component**
 - Key drivers shaping this component are:
 - Ability to select/deselect all data fields for a particular type of view or download.
 - Ability to select different data fields for each user group.
- **Help component**
 - Key drivers shaping this component are:
 - Static HTML pages.
 - Index Based Search.

Technologies and Tools

Liferay: The application was developed as a portlet in Liferay Portal 6.0.6 Community Edition running on Tomcat 6. The Liferay portal server served as a host for the Product Catalogue application.

The portlets in Liferay was developed using JSF Prime Faces as UI frameworks. Its advantages are as follows.

- Simplicity and Performance
- Ease of use
- Rich built in features
- Lightweight
- Best performance when used with Liferay as compared to other JSF frameworks
- The development team encountered many problems with ICE Faces in the past

Jasper: Jasper is an open source Java reporting tool which is easy to use. Jasper reports were used for export functionality.

Dependency Injection Technology: The primary purpose of the dependency injection pattern is to allow selection among multiple implementations of a given dependency interface at runtime, or via configuration files, instead of at compile time. Spring 3.0 was used as the dependency injection technology because of the following reasons -

- Spring technology is easy to implement and it is quiet stable.
- e-Zest has extensive experience of using Spring applications in past.

Hibernate 3.2: Hibernate was used as the ORM framework for catalogue application as it is an open source and the development team has extensive experience in using Hibernate. Also it is the most widely used ORM framework with Java applications

Informix Database: Informix Database was used to provide read only access to end users.

Challenges and Solutions

The following challenges were faced by the team during the Catalogue portlets development.

- To connect all the sources of information to each other and increase the efficiency in updating multiple sources.
- Group-wise division of the users to provide access to the Product Catalogue information which is relevant to them.

The approached solutions to overcome these challenges were as follows.

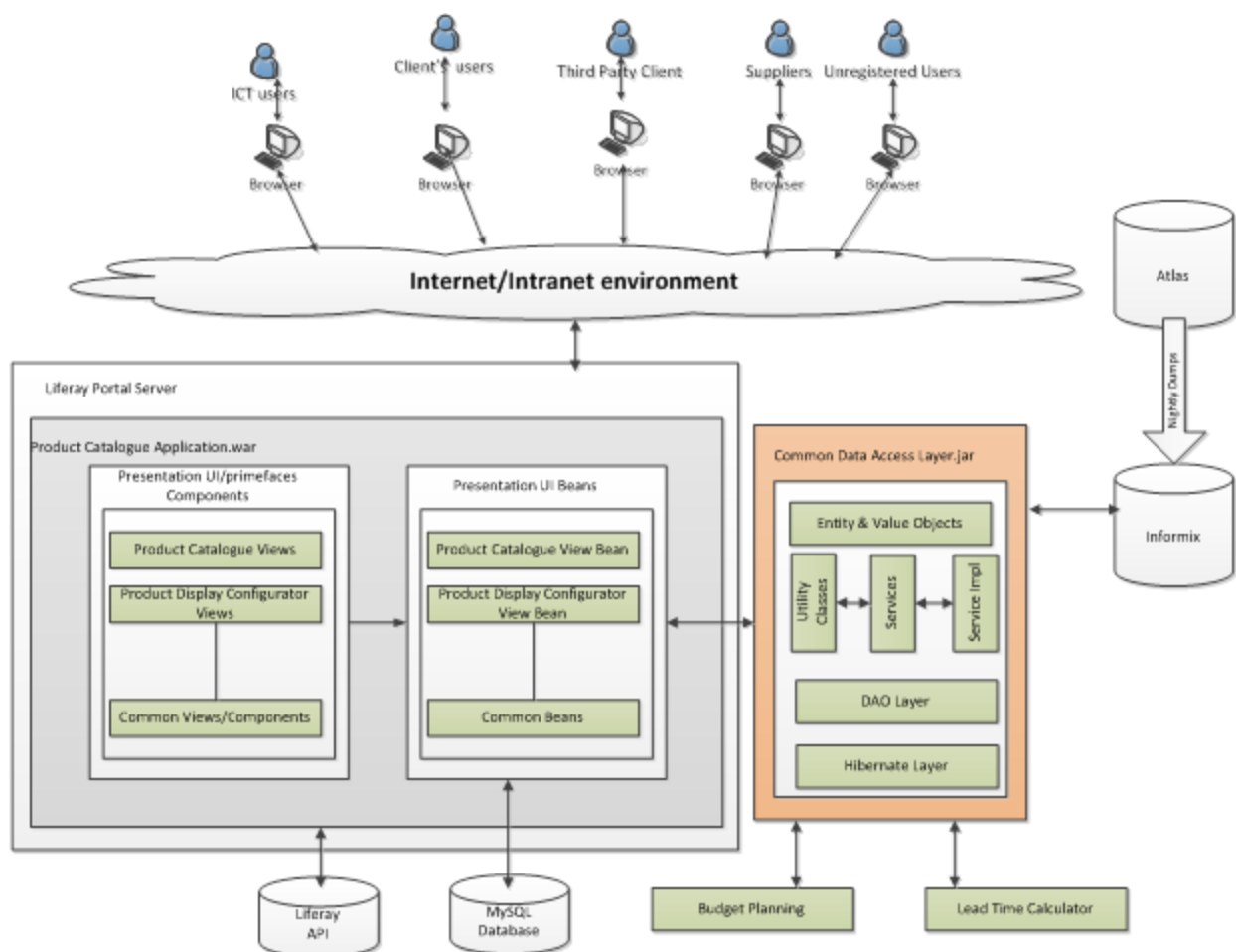
- Hibernate and spring technology was used to increase efficiency and throughput.
- Configuration module was developed to give group wise access.

Solution Architecture

The Product Catalogue was developed as a portlets inside the Liferay portal server which was deployed by the client. There were two major components of the application as follows -

- Product Catalogue web part
- Product Catalogue data access part

The data access layer is reusable provided in the form of an independent JAR so that it can be used by Liferay as well as non Liferay applications. The Product Catalogue web part will access the common data access layer to fetch the information related to various products and vendors. The high level architecture can be as follows.



The architecture of the Product Catalogue is simple to understand and is layered in nature. As depicted in the diagram above, the system would be accessed by five categories of users:

- ICT Users
- Other client's Users
- Third Party clients
- Suppliers
- Unregistered users

The Product Catalogue application was logically divided into two tiers. The presentation tier and the business logic tier. The UI component were developed using JSF technology and PrimeFaces. The business logic tier will be developed using Spring and Hibernate technologies. The Product Catalogue application assumes that the user authentication is done by some external system and the user group information or user role information would be available via the standard Liferay APIs. The development team used the standard Liferay APIs to access the required information. The Product Catalogue application will save the data related to configuration of user groups in a different database to where the product information is stored. Since Liferay uses a database for its own purpose, it is logical to reuse the same database for the product configuration purpose. Common data access layer JAR was developed using Spring and Hibernate technologies. The UI component and the data access layer will communicate with each other using the publically exposed API of the common data access layer. The common data access layer is the only module that interacts with the Informix database.

Business Benefits

- The application provided a single source of information which is consistent with advanced search options.
- Group-wise access to information made it easy for get the required information about a product or vendor.
- As the main business website portal was in Liferay we have given the client a unified solution by portlet development.
- As the application is an online application printing and distribution efforts dropped down to zero.

Bottom Line

The application proved to be an efficient solution to client to enhance and increase capacity building by making product and vendor information more accessible to internal and external users. As it is a single source of information the consistency and efficiencies in updating multiple sources has increased.