IBM

Easy to Setup Org Directory – EZYORG

Minor Project

INTRODUCTION

The purpose of this document is to define scope and requirements of a tool to setup Organization Directory - EZYORG for a software house. They wanted to build an easy to setup organization module that boot straps the implementation of final software at the customer site. Currently, the execution time gets wasted uploading data for masters, finding discrepancies and fixing them before customer verifies the end result. All this leads to lots of project hours becoming unproductive while waiting for the inputs or receiving half-baked information etc.

Implementing this module at the customer site before the rest of the software is delivered will ensure the customer's team enters data and in the process they get trained in setting up of the organization directory. The tool developed should be maintainable. This will primarily come in handy for workflow applications that heavily rely on organization directory.

This document is the primary input to the development team to architect a solution for this project.

System Users

Application Admin and HR department of the customer will primarily use the Easy to setup Organization Directory, EZYORG.

Assumptions

1. EZYORG will be integrated with the existing customer's intranet and therefore it will leverage the existing Intranet's authentication mechanism.

2. To simplify development, we will assume a very lean setup of organization.

REQUIREMENTS

EZYORG will be a web-based re-usable component to be integrated with business workflow applications developed and implemented by the Software house. This tool will operate as an independent module on the customer's Intranet giving access to only the HR team for entering and viewing the Organization Master Data.

Landing Page

EZYORG's landing page is a tabbed page, where tabs are ordered as Step 1, 2,3 to facilitate the setup of organization directory.

Step - 1 Location

An organization can operate from more than one location, thus a list of locations is the first step to setting up the organization directory. Each location has an address, contact numbers and Head of Location. There could be many types of location requirements e.g. companies could have offices in multiple countries, cities, branches within a city. Such type of organization setup will maintain all these types of data separately and relate them in hierarchical manner. E.g. a record for City will

have a reference of which country this city belongs to. Similarly, a record of a Branch will maintain which city it is operating from. There could be multiple branches within a city like we find for banks, insurance companies, mobile offices etc.

For the purpose of this project, we are considering a linear structure of a location will be of type 'City'.

The screen will display 2 options for the user to select a) Add New b) View or Modify.

'**Add New**' option displays a form for capturing data Location id, Location Name, Address1, Address2, Contact Number, Head of Location and Start Date. The head of location information if not available at the time of this record is being created, can be entered later using modify option.

'**View or Modify**' option will display the existing location records in an alphabetical order. User can perform actions such as Delete or Update using the links provided in front of each record.

Deleting a location requires validation of all the data for that location is removed first before confirming the delete action.

Update action for a location will display the selected location record in an editable format; the user can make changes and click on Update button to save changes in the existing record.

Step -2 Department
The screen will display 2 options for the user to select i.e. a) Add New b) View or Modify.

'**Add New**' option displays a form for capturing data such as Department id, Department Name and Head of Department (HOD). The HOD can be updated later using Modify option once the employee master is setup.

'**View or Modify**' option will display the existing department records in an alphabetical order. User can perform actions such as Delete or Update using the links provided in front of each record.

Deleting a department requires validation of all the data for that department is removed first before confirming the delete action. E.g. Employees who have this department stored in the master data, the selected department's master record cannot be removed. This type of situation impacts the referential integrity of a database.

Update of department shall open the selected record in an editable format; the user can make changes and click on Update to save changes in the existing record.

In real world technical jargon Organization Unit (OU) is a term used for a department or a function in an organization. In a mature organization model, OU will be a nested structure i.e. it will have levels like OU Level 0 = Company, OU Level 1 = Functions, OU Level 2 = Departments, OU Level 3 = Sub Departments

Step – 3 Employee

The UI and navigation works in an identical manner for the Employee tab as mentioned for step 1 and step 2.

The form to capture employee information will have the following fields:

Employee Id, First Name, Department (Selection using picker of departments stored in the master), Administrative Reporting Manager (list of employees should be displayed with type ahead), Functional Reporting Manager to be selected using picker.

As a concept, functional reporting of an employee is to his/her own department's senior employee, could be an HOD or a Manager level role. The word functional here implies Job related reporting.

The administrative reporting is normally to the same person as functional reporting. In some cases, when employee is operating from a location where his/her Functional reporting manager is not available on regular basis, the employee will have administrative reporting to that location's head. The word Administrative here implies local office related duties such as following office time or leave discipline. In case leave is required then administrative reporting manager has to be informed.

```
A user-friendly interface needs to be developed to ensure
smooth usage of the system.
```

DEVELOPMENT ENVIRONMENT

EZYORG will be developed as a web application using Java/JSP and DB2 database. Eclipse will be used as the IDE for the same. You may consider using a JavaScript framework like Prototype/Scriptaculous /jQuery.