Architecture Design

For KSU – Purchasing Contracts Management System
(KSU-PCMS)

Version 1.0

Submitted in partial fulfillment of the
Masters of Software Engineering Degree.

Arthi Subramanian
CIS 895 – MSE Project
Department of Computing and Information Sciences
Kansas State University

Committee Members
Dr. Daniel Andresen
Dr. Scott De Loach
Dr. Mitchell Neilsen
# TABLE OF CONTENTS

1. Introduction .................................................................................................................. 3

2. Architecture of the System .......................................................................................... 3

   2.1. Presentation Tier ..................................................................................................... 4

   2.2. Business Tier .......................................................................................................... 5

      2.2.1. Class Descriptions .......................................................................................... 5

      2.2.2. Sequence Diagrams ......................................................................................... 8

      2.2.3. Data Tier (Database) ....................................................................................... 11

   2.3. References ............................................................................................................ 12
1. Introduction
The primary purpose of this document is to provide an architectural design for the purchasing contracts management system. The design will show the presentation tier, the business tier that composes the class and sequence diagrams, and the data tier. Each class will have a brief description about its purpose.

2. Architecture
The following diagram represents the three tier architecture that consists of the presentation tier, the business tier and the data tier. The presentation tier contains the UI (User Interface) elements of the site. This layer is responsible for displaying the contents to the users. The business tier receives requests from the presentation tier and based on the business logic it contains, the result is returned to the presentation tier. When the business tier requests for data, the data tier is responsible for sending the stored application’s data.

Figure 1: Three Tier Architecture
2.1. Presentation Tier

The .NET IDE is used in creating web forms that represents the presentation tier for the purchasing contracts management system. The following table represents the ASP.NET web forms for the clients and their respective purpose in the system.

<table>
<thead>
<tr>
<th>Web Forms</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login.aspx</td>
<td>The default welcome page that allows the procurement officer to enter his login credentials.</td>
</tr>
<tr>
<td>AddorEdit.aspx</td>
<td>Here the procurement officer decides whether he has to add a new contract or edit an existing contract.</td>
</tr>
<tr>
<td>AddContract.aspx</td>
<td>If the procurement officer had decided to add a new contract he is navigated to this page.</td>
</tr>
<tr>
<td>UpdateContract.aspx</td>
<td>If the procurement officer had made a choice to update an existing contract this page appears.</td>
</tr>
</tbody>
</table>

The following table below shows ASP.NET Web forms for the end user of the system.

<table>
<thead>
<tr>
<th>Web Forms</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contracts.aspx</td>
<td>The default page that contains the search textbox and search filters. On performing a valid search the page contains a grid that populates the search results.</td>
</tr>
</tbody>
</table>
2.2. Business Tier

2.2.1. Class Descriptions

2.2.1.1. Administrator

This class will handle the actions that are performed by the procurement officer. The procurement officer of the purchasing department plays the role of an administrator in our system. The administrator has an unique email and password that allows them to
login to the system. The class includes methods such as login( ) that is called when the procurement officer logs onto the system. Once the procurement officer enters his valid credentials the verifylogin() method is called that accepts the email and password as its input and returns a Boolean value based on if the login is successful or not. Before performing a change to an existing contract, the searchcontracts( ) method is first called and a search for the contract to be updated is made.

2.2.1.2. KSU Departments

```
+searchcontracts()
```

The KSU departments class will handle primarily one action of searching for contracts that is indicated by the searchcontracts( ) method. This method will look for contracts in the system based on the contract number/contract title/vendor name/procurement officer/any other keyword.

2.2.1.3. Users

```
+search()
```

This class will handle all user actions. The User class is the super class of the KSU departments class and the Procurement Officer class. It includes a method search() that is used to search for contracts in the system. This method is called when a search field is entered, search filter set and the “search” button I clicked.

2.2.1.4. Session Manager

```
+getAdminDetails(email: String): Set(Administrator)
```


The session manager class is responsible for maintaining the user information such as the email of the procurement officer for a particular session. The getAdminDetails( ) method is called when the administrator information needs to be displayed.

2.2.1.5. Contracts

![Contracts Diagram]

This class comprises of contracts with attributes such as the contract number, contract title, procurement officer, vendor name, start date and expiry date, ApprovalRequired, Link etc.

The getcontractdetails( ) method is called when the user enters a search criteria and the system retrieves the contract details and the master contract file on click of the contract fields. The addcontracts(c:Contract) is called when the user chooses to add a new contract to the system. This method requires the contract details as an input. On the other hand when the user chooses to update existing contract details, the contract details are taken as an input and the changes are made.

2.2.1.6. Search Facade

![Search Facade Diagram]
This class primarily contains methods that are used to retrieve the search results based on the options chosen by the user. The getbycontractnumber(contractNumber:int) is called when the user enters search data, sets the search filter to “contract number” and hits the search button. Likewise the getbycontractTitle(contractTitle:string), getbyvendornname(vendorName:string), getbyprocurofficer(procurOfficer:string), getbykeyword(Keys:String) are called when a search criteria is entered and the search filter is set to contractTitle, vendorname, procurement officer and keyword respectively.

2.2.1.7. Keyword

<table>
<thead>
<tr>
<th>Keyword</th>
</tr>
</thead>
<tbody>
<tr>
<td>-Keywords : Set(String)</td>
</tr>
<tr>
<td>+insertKeywords(Keys:Set(String)): Boolean</td>
</tr>
</tbody>
</table>

The keyword class contains a private method that inserts keywords as a set of strings for the file to be uploaded

2.2.1.8. Recommendation Set

<table>
<thead>
<tr>
<th>Recommendation Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>-ContractNumber : Integer</td>
</tr>
<tr>
<td>-getrecomm(ContractNumber : Integer) : contractset</td>
</tr>
</tbody>
</table>

When an end user performs a search based on a particular contract number that does not exist in the system, the getrecomm() method is invoked. This method further returns a set of contracts whose contract number is the closest match to the number searched by the user.

2.2.2. Sequence Diagrams

2.2.2.1 Procurement Officer –Login to the system
In order to add or edit contracts the procurement officer needs to login to the system. The officer will also enter his/her email and password and logins to the system. On successful login the procurement officer is redirected to the AddorEdit.aspx page where he can perform the desired operation.
The above diagram represents the sequence of activities that the KSU departments go through to perform a search for contract details. The search keyword is entered and a search filter is set from 5 categories: contract number, contract title, vendor, procurement officer, keywords. The entered keyword is matched against the contracts table and the matched set is returned back to the user through the contracts.aspx page.

The following diagram represents the sequence of activities that are carried out by the procurement officer in order to insert/edit a contract. The procurement officer enters the contracts details and if the contract number does not exist, the contract is added to the system.

**Figure: Add contract**

The following diagram represents the edit contracts scenario. When the procurement officer chooses to edit an existing contract, the updatecontract() method is called, he first has to search for the required contract number and if the contract exists further edit and save it. If no search results are returned then a message “No results found” is displayed to the user.
2.2.3. Data Tier

The system database has the following structure of tables:

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contracts</td>
<td>Contains the contract related information such as the contract number, contract title, vendor information, procurement officer etc</td>
</tr>
<tr>
<td>Departments</td>
<td>Contains the details of the departments such as the department name and department ID that are provide approval for certain contracts.</td>
</tr>
<tr>
<td>Administrators</td>
<td>Contain the login details of the administrators.</td>
</tr>
<tr>
<td>Vendors</td>
<td>Contain the vendor information such as the vendor name, Item etc</td>
</tr>
</tbody>
</table>
References:


- Sequence Diagram in UML, By Mandar Chitnis, Pravin Tiwari, & Lakshmi Ananthamurthy

- Donald Bell UML basics: An introduction to the Unified Modeling Language