Finance Web Vendor Portal - Group 13 - Requirements Document

|  |  |  |
| --- | --- | --- |
| Role | Name | Contact |
| Group 13 | Nicholas Lawrence | Nicholas.Lawrence@student.curtin.edu.au |
| Group 13 | Chen Bi | Chen.Bi@student.curtin.edu.au |
| Group 13 | Yan Tsz Cheng | YanTsz.Cheng@student.curtin.edu.au |
|  |  |  |
| Curtin Supervisor | Hannes Herrmann | Hannes.Herrmann@curtin.edu.au |
|  |  |  |
| Client | Mia Kelly | Mia.Kelly@curtin.edu.au |
| Client | Scott Martinson | S.Martinson@curtin.edu.au |

# Introduction

Nicholas Lawrence, Chen Bi and Yan Tsz Cheng will be designing a website for the Curtin Financial Team. The website will assist the vendors of Curtin with queries relating to their invoices in order to reduce the amount of traffic faced by telephone. Johannes Herrmann will be the supervisor for the duration of this project.

# System

The website will be written in C# targeting .NET 4.5. The front end will utilise HTML and CSS for format and styling and JavaScript will be used to extend front end functionality. The portal will be developed against the ASP.NET MVC as this is required by CITS for portals which are to be integrated with their systems.

The website will utilise a Microsoft SQL Server database as this is the type of database maintained by CITS (who will be responsible for the website after its implementation.

All of the code which is produced, along with all documentation, meeting minutes and any other important information will be stored in a Bitbucket repository provided to us by Johannes Herrmann and will be published under a license of Johannes' preference.

We will use Trello for tracking the allocation and completion of tasks. Hours spent on a given task will be recorded against the relevant card in Trello.

# System Requirements

## 1.1 Functional Requirements

1. Users and administrators must be able to update their own password. This can be managed through a "Manage Profile" page or through a "Forgot your Password" link.  
     
   The "Forgot your Password" link will send a reset password email to the registered email address for the account holder.  
     
   In the event that a user has registered with an incorrect email address, they will need to contact an administrator in order to correct their details to reset their password. The administrator can also reset the user's password manually to help them access the system.
2. Identification of a vendor at their initial registration will occur through the use of a vendor ID and temporary password. After this initial log in, the vendor will be able to log into their account through the use of their vendor ID and a password they choose themselves during initial log in.
3. Administrators must be able to add, delete, and modify any user or administrator, where “modify” means changing a password, or changing other stored details such as the e-mail address. Users and administrators are identified by unique usernames.
4. A single administrator account will be created initially, with an administrator able to create other administrator accounts. User accounts are only added through the application system, based on information stored in the database.
5. The system core must update the database when the DSA makes any modifications to its data (the only component of the database the system core will have access to will be a user table determining user accounts and registered information).
6. The system core must provide an API through which the DSA can define and show any necessary user interfaces. Functions will be provided to allow the DSA access to the information from the views.
7. A user must be able to view a list of all of their invoices and their status.
8. A user must be able to select an invoice in order to view the specific details related to the status of that invoice (Department the invoice is currently with, contact number, email address, status within department).
9. Web pages must be responsive to mobile devices.

### 1.2 Access Requirements

1. A running instance of the system must be accessible from any device with a web browser so long as the browser meets the requirements of the following browsers:
   1. Chrome 52.0
   2. Firefox 47.0
   3. Internet Explorer 11.0
   4. Safari 9.0
2. The system core must ensure that information and functionality provided by the system is only accessible while the user or administrator is logged in. Users and administrators log in via username and password.
3. The system core must ensure that information and functionality intended only for administrators is not provided to non-administrative users.

### 1.3 Security Requirements

1. The system core must sanitise all database inputs.

### 1.4 Platform Requirements

1. Server platform must be supported by CITS.

## 2.1 Non-Functional Requirements

1. All code will adhere to the Department of Computing’s Coding Standard.
2. All code will meet the requirements from CITS for code to be used for Curtin finance since it will eventually be hosted by CITS.
3. All passwords will meet Curtin standards.

## 3.1 Constraints

1. All software required for the running, use and maintenance of the system must be either available freely (open-source, free-ware, *etc*.) or must be approved in writing by both the project supervisor and the client areas. The database used must be the version of Oracle for which Curtin has a license.
2. The front-end must function on low-end machines, such as those found in the older laboratories in Curtin. The developers may assume that appropriate software (*e.g.*, Java) has been updated to the latest version.

## 4.1 User Acceptance

1. The following must be presented to the representative(s) of the client areas before the due date for the second half of the project, with signed receipts being included in the final submission:
   1. The user manual for that area.
   2. The technical manual.
   3. A copy of any software required.
   4. A presentation of the functionality for the front-end users.
   5. A presentation of the functionality of the front-end for administrators.