**Assignment 1**

**COMP8772 Web-Based Systems Development**

**Prepared by:**

|  |  |
| --- | --- |
| **FAN** | **Name** |
| **kim0861** | **Kyuhyun Kim** |
| **lin0385** | **Sen Lin** |
| **lee1761** | **Hwanjoo Lee** |
| **ocba0001** | **Sherica Fabienne Ocbania** |

**Introduction**

Building a user-friendly webpage is vital to provide up-to-date information for end users. Every information should be reviewed and updated when new contents are included on the website so that visitors will be able to view refreshed information. In the point of companies, managing all contents on the page, such as adding, updating, or deleting them must be performed to manage their current state. Based on the essential requirements, this report will discuss the website design and functionalities that are developed to give reasonable user experiences.

1. **Sitemap**

The website contains three major functionalities – User administration, Product administration, and Management reporting. Each actor (Manager and Administrator) can access specific functions under their parent functionalities, depending on their roles. User authentication is verified on the login page, and anyone without authorisation will not be allowed to access the page. Once their identification is checked, actors can travel to the page, containing a side bar menu so that they can choose one of the options in the menu. Administrators are able to manage all users while managers can manage all products, and they can only access functionalities being responsible for. Apart from administrators, however, managers can view a report of all changes made by any users in the page as a table in the date order with the most recent at the top of it. Figure 1 represents the graphical flow of the sitemap applied in the design of the web site.

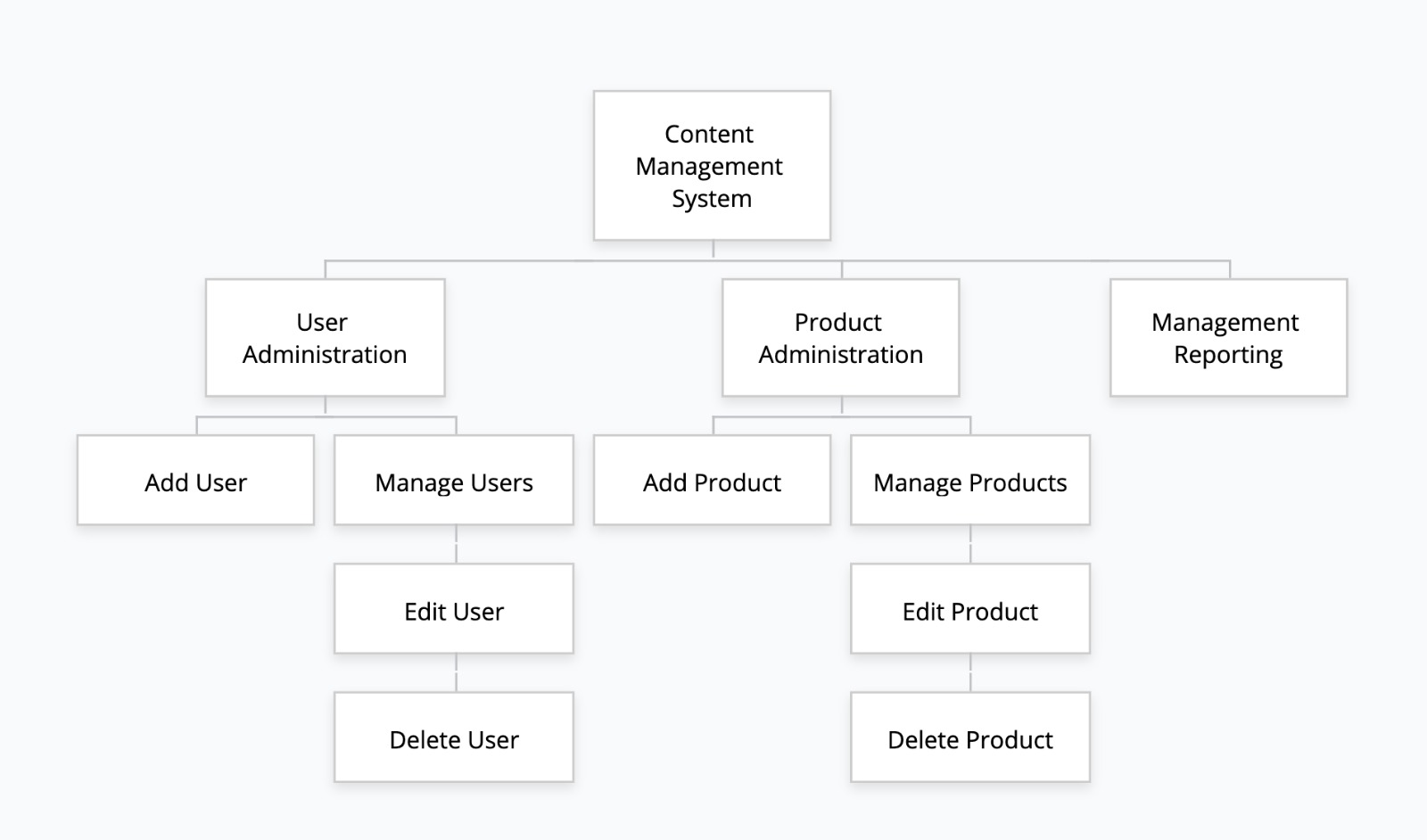


Figure 1. A sitemap for the website

1. **Technique description**

The major tools that are used for designing the website is HTML, CSS, PHP and MySQL. HTML is included in a few PHP files, where style specifications, defined in ‘style.css’ for each element is applied. Within PHP, phpMyAdmin supports the administration of MySQL, in which manages all functions of databases, and any SQL statements can be executed. In the database, three tables exist from the queried SQL command. Figure 2 shows the table information in the ‘cms’ database on the page of phpMyAdmin.

텍스트이(가) 표시된 사진

자동 생성된 설명

Figure 2. A screenshot for the ‘cms’ database on the phpMyAdmin

1. **Shared use cases**

The website verifies the authentication of all users when they wish to access the page whereas the identification is not checked when logging out. Descriptions of shared use cases for all users are stated in the table 1 and 2, and figure 3 and 4 show the design for the log in page and a dashboard.

|  |  |
| --- | --- |
| Use case | Log in |
| Actor | Manager, Administrator |
| Description | All users can log in to the page, then system verifies authentication information. Once it is checked, the system travels to the page of a dashboard. |

Table 1. First shared use case

|  |  |
| --- | --- |
| Use case | Log out |
| Actor | Manager, Administrator |
| Description | All users can log in from the page, then system receives its request and travels back to the log in page. |

Table 2. Second shared use case

텍스트, 하늘, 실외, 군중이(가) 표시된 사진

자동 생성된 설명

Figure 3. Screenshot for the log in page

**텍스트이(가) 표시된 사진

자동 생성된 설명**

Figure 4. Screenshot for the dashboard

1. **Use cases - manager**

|  |  |
| --- | --- |
| Use case | Add new products |
| Actor | Manager |
| Description | A manager can add new products and the system receives his/her request, as well as verifies authentication information. Once it is checked, the system travels to the page to add information about new products. A manager types required contents – name, price and amount and saves them. The system then moves back to the page of a list of all products. |

Table 3. First use case for managers

|  |  |
| --- | --- |
| Use case | Edit current products |
| Actor | Manager |
| Description | A manager can update products that are in the product list and the system receives his/her request, as well as verifies authentication information. Once it is checked, the system travels to the page to edit information about products. A manager types of new contents – name, price and amount and saves them. The system then moves back to the page of a list of all products. |

Table 4. Second use case for managers

|  |  |
| --- | --- |
| Use case | Delete products |
| Actor | Manager |
| Description | A manager can remove products that are in the product list and the system receives his/her request, as well as verifies authentication information. Once it is checked, the system gets rid of its information from the list. The system then moves back to the page of a list of all products. |

Table 5. Third use case for managers

|  |  |
| --- | --- |
| Use case | View management log |
| Actor | Manager |
| Description | A manager can view a report that contains all changes made by him/her. The system receives his/her request, as well as verifies authentication information. Once it is checked, the system displays a table of all previous changes in date order with the most recent at the top of the list. |

Table 6. Fourth use case for managers

1. **Use cases - administrator**

There are three main functionalities for administrators and summaries for these use cases are followed in table 7, 8, and 9, as well as corresponding screenshots available.

|  |  |
| --- | --- |
| Use case | Add users |
| Actor | Administrator |
| Description | An administrator can add new users and the system receives his/her request, as well as verifies authentication information. Once it is checked, the system travels to the page to add information about new users. An administrator types required contents – ‘Username’, ‘Password’, and selects his/her role. The system then moves back to the page of a list of all users including newly added ones. |

Table 7. First use case for administrators

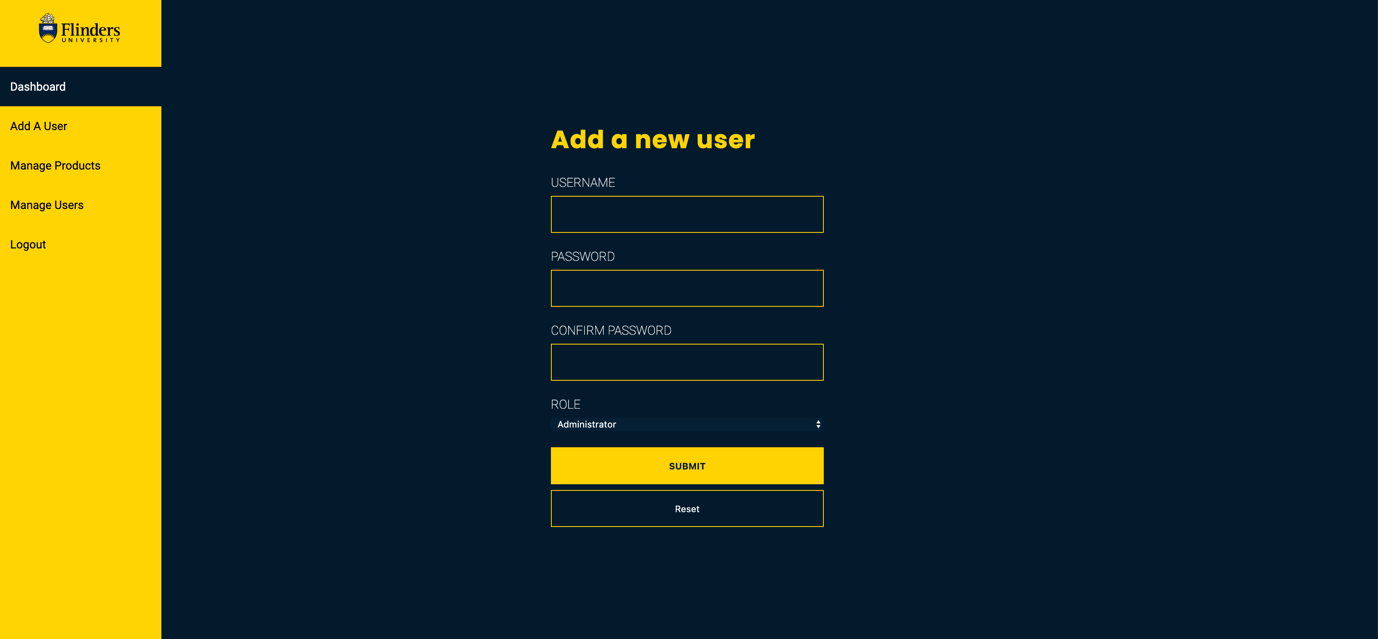


Figure X. Screenshot for the ‘add user’ use case

|  |  |
| --- | --- |
| Use case | Edit users |
| Actor | Administrator |
| Description | An administrator can update current a user and the system receives his/her request, as well as verifies authentication information. Once it is checked, the system travels to the page to edit information about users. An administrator types of new Username and changes his/her role if needed. The system then moves back to the page of a list of all users including newly edited ones. |

Table 8. Second use case for administrators

|  |  |
| --- | --- |
| Use case | Delete users |
| Actor | Administrator |
| Description | An administrator can delete a user and the system receives his/her request, as well as verifies authentication information. Once it is checked, the system removes the information from the list of users. The system then moves back to the page of a list of all users without deleted user’s information. |

Table 9. Third use case for administrators