

**Quiz Practice System**

**Software Requirement Specification**

**Project Code: QPS**

**Document Code: QPS- v1**

– Hanoi, July 2025 –

**Table of Contents**

[I. Record of Changes 3](#_Toc200613202)

[II. Software Requirement Specification 4](#_Toc200613203)

[1. Product Overview 4](#_Toc200613204)

[2. User Requirements 5](#_Toc200613205)

[2.1 Actors 5](#_Toc200613206)

[2.2 Use Cases 5](#_Toc200613207)

[3. Software Features 6](#_Toc200613208)

[3.1 Functional Overview 6](#_Toc200613209)

[3.2 <<Feature Name 1>> 8](#_Toc200613210)

[3.3 <<Feature Name 2>> 10](#_Toc200613211)

[4. Non-Functional Requirements 11](#_Toc200613212)

[4.1 External Interfaces 11](#_Toc200613213)

[4.2 Quality Attributes 11](#_Toc200613214)

[5. Requirement Appendix 12](#_Toc200613215)

[5.1 Business Rules 12](#_Toc200613216)

[5.2 Common Requirements 12](#_Toc200613217)

[5.3 Application Messages List 12](#_Toc200613218)

[5.4 Other Requirements… 13](#_Toc200613219)

# I. Record of Changes

|  |  |  |  |
| --- | --- | --- | --- |
| Date | A\* M, D | In charge | Change Description |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

\*A - Added M - Modified D - Deleted

# II. Software Requirement Specification

## 1. Product Overview

*[Gives the overall description about the product with some introduction and the context diagram. The context diagram presents the boundary and connections between the system you’re developing and everything else in the universe. This identifies external entities (or terminators – software, hardware, human components, and other systems) outside the system that interface to it in some way, as well as data, control, and material flows between the terminators and the system.]*

<<Sample: The Cafeteria Ordering System is a new software system that replaces the current manual and telephone processes for ordering and picking up meals in the Process Impact cafeteria. The context diagram below illustrates the external entities and system interfaces for release 1.0. The system is expected to evolve over several releases, ultimately connecting to the Internet ordering services for several local restaurants and to credit and debit card authorization services.



Figure 1-1: Cafeteria System overview

As described in the figure 1-2, the system will have the following components

* **Cafeteria Inventory System:**  this component is….
* **Payroll system:** …..

>>

## 2. User Requirements

### 2.1 Actors

*[An actor is a person (or sometimes another software system or a hardware device) that interacts with the system to perform a use case. Following are some questions you might ask to help user representatives identify actors*

* *Who (or what) is notified when something occurs within the system?*
* *Who (or what) provides information or services to the system?*
* *Who (or what) helps the system respond to and complete a task?*

*This part gives the description of system actors, you can follow the table form as below]*

|  |  |  |
| --- | --- | --- |
| **#** | **Actor** | **Description** |
| 1 | Administrator |  |
| 2 | Menu Manager |  |
| 3 | … |  |

### 2.2 Use Cases

*[A use case (UC) describes a sequence of interactions between a system and an external actor that results in the actor being able to achieve some outcome of value. The names of use cases are always written in the form of a verb followed by an object. Select strong, descriptive names to make it evident from the name that the use case will deliver something valuable for some user]*

#### 2.2.1 Diagram(s)

*[Provide the UC diagram(s) to show the actor-UCs and UC-UC relationships like the sample below. You can have multiple UC diagrams for the system]*

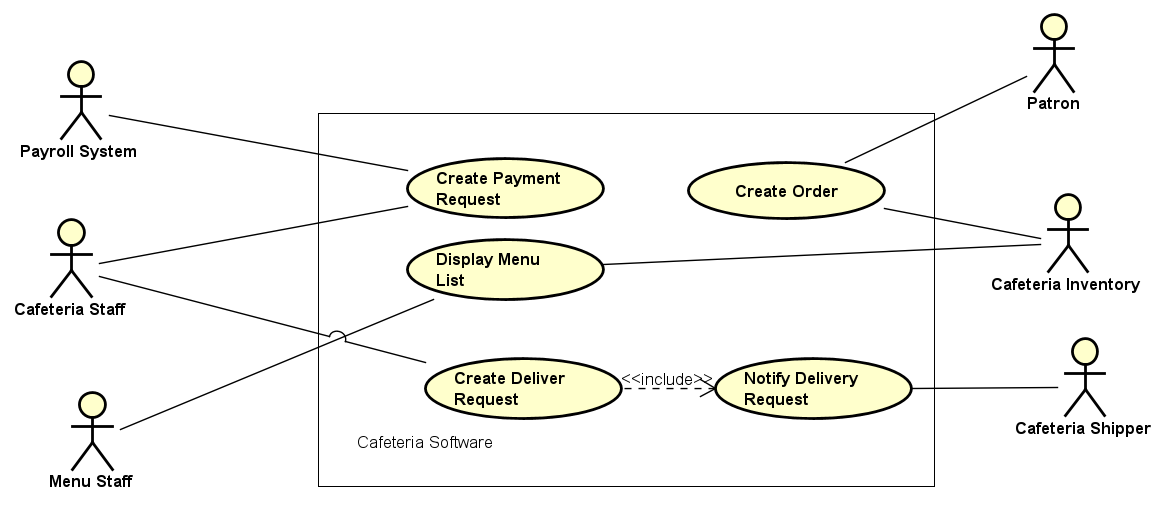
******

Figure 2-1: Use case diagram of Cafeteria system

#### 2.2.2 Descriptions

*This part describes the use cases & their main flow (the list of the user actions and corresponding system responses that will take place during execution of the use case under normal, expected conditions), you can follow the table form as below]*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Group function** | **Use Case** | **Actors** | **Use Case Description & Main Flow** |
| 01 | Order management | View Menu | Patron |  |
| 02 | Order a Meal | Patron |  |
| 03 |  | … |  |  |

## 3. Software Features

### 3.1 Functional Overview

*[Provide functionality overview of software system: screen flow, screen descriptions, system user roles, screen authorization, non-screen functions, ERD]*

#### 3.1.1 Screens Flow

*[This part shows the system screens and the relationship among screens. You can draw the Screens Flow for the system in the form of diagram as below. Please note that beside the normal flat screen, we might have the oval notation for pop-up screen (Import Order) or a screen with multiple information tab (Order Details), etc. You may also use text or background format for different visuality purpose]*



#### 3.1.2 Screen Descriptions

*[Provide the descriptions for the screens in the Screens Flow above]*

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **Screen** | **Description** |
| 1 | Order Meals | Create Order | <<Screen Brief description>> |
| 2 | Order Meals | Change Order |  |
| 3 | .. |  |  |

#### 3.1.3 Screen Authorization

*[Provide the system roles authorization to the system features (down to screens, and event to the screen activities if applicable) in the table form as below – replace Role1, Role2,… with your specific system user role names]*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **#** | **Screen** | **Role-Name1** | **Role-Name2** | **Role-Name3** | **…** |
| 1 | <<Screen Name1>> | X |  | X | X |
| 1.1 | <<Function name/ Action Name>> |  |  | X | X |
| 2 | Add New Product | X |  | X |  |
| 2.1 | Save New Production Info |  |  | X | X |
| 2.2 | Back to production list |  |  |  | X |
|  | Update Own Data |  |  |  | X |
|  | Update Managed Data |  |  |  | X |
|  | Delete Data |  |  |  |  |
|  | … |  |  |  |  |

#### 3.1.4 Non-Screen Functions

*[Provide the descriptions for the non-screen system functions, i.e batch/cron job, service, API, etc.]*

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **System Function** | **Description** |
| 1 | <<Feature Name>> | <<Function Name1>> | <<Function Name1 Description>> |
| 2 | … |  |  |

#### 3.1.5 Entity Relationship Diagram

*[Provide the entity relationship diagram and the entity descriptions in the table format as below]*



**Entities Description**

|  |  |  |
| --- | --- | --- |
| **#** | **Entity** | **Description** |
| 1 | User |  |
| 2 | Meal |  |
| 3 | Meal Subscription |  |
| 4 | … |  |

#### 3.1.6 Entity Details

*[Phần này Mô tả chi tiết các thực thể để thấy rõ các thực thể có các thuộc tính nào, thuộc tính nào là thuộc tính khóa…]*

1. **Product**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **#** | **Attribute name** | **PK** | **Type** | **Mandatory** | **Description** |
| 1 | Product Number | x | String | Yes |  |
| 2 | Name |  | String | Yes |  |
| 3 |  |  |  |  |  |
| 4 | … |  |  |  |  |

### 3.2 <<Feature Name 1>>

#### 3.2.1 <<FID - Function Name 1 / UC-ID Use Case Name >>

*[A function can be a screen or a non-screen function (listed in the part 3.1.5 above). In this part, you need to provide the details on the related function, focus on mentioning below information.*

*]*

***3.2.1.1 Screen Mock-up***

*<<Provide basic (concept) design of screen –include screen mock-up and screen definition>>*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Login** |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  | User name |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  | Password |  |  |  |  |  |
|  |  | Login | | |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

**Figure 3-1**: Screen Design of Login

**Table 3-1: Screen Definition**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Field Name | Type | Mandatory | Max Length | Description |
| 1 | User Name | Text | Yes | 20 |  |
| 2 | Password | Text | Yes | 20 | Display "\*" instead of clear character |
| 3 | Login | Button |  |  | Go to home page |
| 4 | Close | Button |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

***3.2.1.2 Use Case Description***

***<Provide detailed description for Use case>***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case ID** | | **UC01** | **Use Case Name** | | **Login** | |
| **Author** | | **HungNV** | **Version** | **1.0** | **Date** | **11/05/2013** |
| **Actor** | | LMS User | | | | |
| **Description** | | The function allows a user to be able to login in the software when he/she have registered an account and his/her account is still active. | | | | |
| **Precondition** | | PRE-01:  PRE-02: | | | | |
| **Trigger** | | TRG-01: | | | | |
| **Post-Condition** | | POS-01: | | | | |
| **Main flows** | | | | | | |
| ***Step*** | ***Actor*** | ***Action*** | | | | |
| *1* | User | Type URL into location field of internet browser and then press enter | | | | |
| *2* | LMS | Display Login screen with the following fields:  - User Name  - Password  - Login and Close button | | | | |
| *3* | User | Enter user name & password into User Name & Password fields on the Login screen, then click on Login button. | | | | |
| *4* | LMS | Validate the entered user name & password and then display Home screen | | | | |
|  |  |  | | | | |

**Alternative flows**

|  |  |  |
| --- | --- | --- |
| **AT1** | At step 2 in the main flows**,** if there is an internal error in the system, | |
| **Sub step** | **Actor** | **Action** |
| 2.1 | LMS | Display "Error" page with message "Internal System Error, please contact with administrator" |
|  |  |  |
|  |  |  |

***Business Rules***

|  |  |
| --- | --- |
| **#** | **Rule Description** |
| *BR-01* |  |
| *BR-02* |  |

#### 3.2.2 <<Function Name 2>>

…

### 3.3 <<Feature Name 2>>

…

## 4. Non-Functional Requirements

### 4.1 External Interfaces

*[This section provides information to ensure that the system will communicate properly with users and with external hardware or software/system elements.]*

### 4.2 Quality Attributes

*[List all the required system characteristics (quality attributes) specification. Some of the possible attributes are provided with the guide/descriptions are mentioned here]*

#### 4.2.1 Usability

*[This section includes all those requirements that affect usability. For example, specify the required training time for a normal users and a power user to become productive at particular operations specify measurable task times for typical tasks or base the new system’s usability requirements on other systems that the users know and like specify requirement to conform to common usability standards, such as IBM’s CUA standards Microsoft’s GUI standards]*

#### 4.2.2 Reliability

*[Requirements for reliability of the system should be specified here. Some suggestions follow:*

*Availability—specify the percentage of time available ( xx.xx%), hours of use, maintenance access, degraded mode operations, and so on.*

*Mean Time Between Failures (MTBF) — this is usually specified in hours, but it could also be specified in terms of days, months or years.*

*Mean Time To Repair (MTTR)—how long is the system allowed to be out of operation after it has failed?*

*Accuracy—specifies precision (resolution) and accuracy (by some known standard) that is required in the system’s output.*

*Maximum Bugs or Defect Rate—usually expressed in terms of bugs per thousand lines of code (bugs/KLOC) or bugs per function-point( bugs/function-point).*

*Bugs or Defect Rate—categorized in terms of minor, significant, and critical bugs: the requirement(s) must define what is meant by a “critical” bug; for example, complete loss of data or a complete inability to use certain parts of the system’s functionality.]*

#### 4.2.3 Performance

*[The system’s performance characteristics are outlined in this section. Include specific response times. Where applicable, reference related Use Cases by name.*

*Response time for a transaction (average, maximum)*

*Throughput, for example, transactions per second*

*Capacity, for example, the number of customers or transactions the system can accommodate*

*Resource utilization, such as memory, disk, communications, and so forth.]*

#### 4.2.4 …

## 5. Requirement Appendix

*[Provide business rules, common requirements, or other extra requirements information here]*

### 5.1 Business Rules

*[Provide common business rules that you must follow. The information can be provided in the table format as the sample below]*

|  |  |
| --- | --- |
| ID | Rule Definition |
| BR-01 | Delivery time windows are 15 minutes, beginning on each quarter hour. |
| BR-02 | Deliveries must be completed between 10:00 A.M. and 2:00 P.M. local time, inclusive. |
| BR-03 | All meals in a single order must be delivered to the same location. |
| BR-04 | All meals in a single order must be paid for by using the same payment method. |
| BR-11 | If an order is to be delivered, the patron must pay by payroll deduction. |
| BR-12 | Order price is calculated as the sum of each food item price times the quantity of that food item ordered, plus applicable sales tax, plus a delivery charge if a meal is delivered outside the free delivery zone. |
| BR-24 | Only cafeteria employees who are designated as Menu Managers by the Cafeteria Manager can create, modify, or delete cafeteria menus. |
| BR-33 | Network transmissions that involve financial information or personally identifiable information require 256-bit encryption. |
| BR-86 | Only regular employees can register for payroll deduction for any company purchase. |
| BR-88 | An employee can register for payroll deduction payment of cafeteria meals if no more than 40 percent of his gross pay is currently being deducted for other reasons. |

### 5.2 Common Requirements

*[Fill all the common requirements here..]*

### 5.3 Application Messages List

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Message code** | **Message Type** | **Context** | **Content** |
| 1 | MSG01 | In line | There is not any search result | *No search results.* |
| 2 | MSG02 | In red, under the text box | Input-required fields are empty | *The \* field is required.* |
| 3 | MSG03 | Toast message | Updating asset(s) information successfully | *Update asset(s) successfully.* |
| 4 | MSG04 | Toast message | Adding new asset successfully | *Add asset successfully.* |
| 5 | MSG05 | Toast message | Confirming email of asset hand-over is sent successfully | *A confirmation email has been sent to {email\_address}.* |
| 6 | MSG06 | Toast message | Resetting asset information successfully | *Return asset(s) successfully.* |
| 7 | MSG07 | Toast message | Deleting asset information successfully | *Delete asset(s) successfully.* |
| 8 | MSG08 | In red, under the text box | Input value length > max length | *Exceed max length of {max\_length}.* |
| 9 | MSG09 | In line | Username or password is not correct when clicking sign-in | *Incorrrect user name or password. Please check again.* |

### 5.4 Other Requirements…