[2012]

|  |
| --- |
| [Team Assignment 01] | Team 02 – The HIT  *Giang Thị Hà Thanh – Trần Dũng Đạt – Nguyễn Trần Hồng Phúc - Huỳnh Chấn Huy - Tạ Quang Hiệp*  HIT-Big |



|  |  |
| --- | --- |
| Class K15T2 | SOFTWARE PROJECT MANAGEMENT PLAN ver 2.2 |

#### Information of document

|  |  |
| --- | --- |
| **Title** | **Software Project Management Plan** |
| **Author(s)** | All team |
| **Reviewer(s)** | All team |
| **Team name** | HIT Team |
| **Team members** | Thanh Giang, Hiep Ta, Phuc Nguyen, Giang Nguyen, Dat Tran, Huy Huynh |
| **Project mentors** | Ms. Huong Nguyen, Mr. Huyen Pham |
| **Editor** | Thanh Giang |
| **Type of report** | Software Project Management Plan |
| **Software used** | MS Word |

#### Document Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision Number** | **Date of Issue** | **Author(s)** | **Brief Description of Change** |
| 0.1 | 6/3/2011 | Team | Produce Draft ver |
| 0.2 | 7/3/2012 | Thanh Giang | Introduction, use-case diagrams and description |
| 0.3 | 9/3/2012 | Hiep Ta, Hong Phuc | Constrains and priority |
| 0.4 | 9/3/2012 | Dat Tran | Quality Attributes |
| 1.0 | 15/3/2012 | Thanh Giang | Consume for final submit |
| 2.0 | 12/7/2012 | Thanh Giang, | Fix for following the comment of mentor: introduction, Constrains and priority, use-case |
| 2.1 | 13/7/2012 | Dat Tran | Quality Attributes |
| 2.2 | 13/7/2012 | Hiep Ta, Trong Giang | Process, scope, Strategy |

Contents

[1. Introduction 4](#_Toc331108859)

[1.1. Purpose of this document 4](#_Toc331108860)

[1.2. Reference materials 4](#_Toc331108861)

[1.3. Definitions and acronyms 4](#_Toc331108862)

[1.4. Project Summary 4](#_Toc331108863)

[1.5. Project Scope and Objective 5](#_Toc331108864)

[2. Project organization 6](#_Toc331108865)

[2.1. Organizational structure 6](#_Toc331108866)

[2.2. Organizational Boundaries and Interfaces 6](#_Toc331108868)

[2.3. Roles and responsibilities 6](#_Toc331108869)

[3. Project Management 8](#_Toc331108870)

[3.1. Management Process 8](#_Toc331108871)

[3.2. Project strategy 9](#_Toc331108872)

[3.3. Top 10 Risks: 9](#_Toc331108873)

[3.4. Mile-stone: 10](#_Toc331108874)

[4. The architectural drivers 11](#_Toc331108875)

[4.1. High-level Functional Requirements 11](#_Toc331108876)

[4.1.1. Entity List 11](#_Toc331108877)

[4.1.2. Use-case List 11](#_Toc331108878)

[4.1.3. Use-case Diagrams and Use-case Descriptions 13](#_Toc331108879)

[4.2. Use-case Descriptions 19](#_Toc331108880)

[4.3. Constrains: 22](#_Toc331108881)

[4.3.1. Business Constrains: 22](#_Toc331108882)

[4.3.2. Technical Constrains: 23](#_Toc331108883)

[4.4. Quality Attributes 23](#_Toc331108884)

[4.4.1. Performance: 23](#_Toc331108885)

[4.4.2. Availability: 24](#_Toc331108886)

[4.5. Prioritization 24](#_Toc331108887)

[4.5.1. Team priority (Difficulty ranking scale) 24](#_Toc331108888)

[4.5.2. Stakeholder priority (Important scale) 25](#_Toc331108889)

[4.5.3. Priority scale 25](#_Toc331108890)

[4.5.4. Use Cases 25](#_Toc331108891)

[4.5.5. Quality Attribute Scenarios 26](#_Toc331108892)

[4.5.6. Constraints 26](#_Toc331108893)

[4.5.7. Technical Constraints 26](#_Toc331108894)

[4.5.8. Business Constraints 27](#_Toc331108895)

[-- The End -- 27](#_Toc331108896)

# Introduction

## Purpose of this document

This section of the Software Project Management Plan (SPMP) gives an overview of the business context and the architectural drivers with their impact on the project. It also contains the project deliverables, the summary of the schedule the SPMP. We also show the development strategy that we choose to show how this may affect the design of a system.

## Reference materials

* Project Plan Outline IEEE Template
* Anthony J. Lattanze, 2008, Architecting Software Intensive Systems
* [Sandcastle] - Architectural Drivers Specification
* [Lat08] Lattanze, A. Architecting Software Intensive Systems: A Practitioners Handbook, New York, NY: Auerbach, 2008

## Definitions and acronyms

|  |  |
| --- | --- |
| Acronym | Definition |
| IEEE | Institute of Electrical and Electronics Engineers |
| CCB | Configuration Control Board |
| CM | Configuration Management |
| C&C | Component and Connector |
| DB | Database |
| POS | Point of Sale |
| LOC | Line of Code |
| OS | Operating System |
| SPMP | Software Project Management Plan |
| QA | Quality Assurance |
| SPMP | Software Project Management Plan |
| SRE | Software Risk Evaluation |
| SRS | Software Requirements Specification |
| TSP | Team Software Process for Education |

## Project Summary

Company A, a retail chain, has decided to develop a sales system (hereinafter, the system) in conjunction with its launch of a point service.

## Project Scope and Objective

HIT Team will develop POS System on web platform with following targets:

* Customer who have become point service members are issued point cards, which bear bar codes indicating their member numbers, and when they purchase products with cash, they are awarded points based on the amount of their purchase. The point award rate is fixed, and stored in the system as constant.
* Manage Products: For each product, its standard price, common to all stores, is set as a part of the product data.
* Manage Retail Store: Each store, however, can set and use its own actual retail price instead of the standard price during the limited period specified be each store. The actual retail price must be set in advance, and it cannot be charged in the middle of the specified period.
* Manage Types: Products are classified into product types such as food, general merchandise, etc. Not all stores carry every product type, and the range of product types carried is designated for each store.
* Analysis Statistic: Moreover, in addition to the sales operation, the system is also capable of performing the statistical analysis on the sales Bills of all stores in near real-time manner.The statistical analysis of the sales Bills is performed using the sales data stored at the head office server.

* 1. **Document's intended audience**

|  |  |
| --- | --- |
| **Intended Audience** | **Reading Suggestions** |
| **Project Manager** | Section 2 – The architectural Drivers: List functions showed by use-case diagrams and constrains to make the Project Manager has an overview. So he can have the estimates for the project.  Section 3 – The development Strategy |
| **Software Architecture and Designer** | Section 2 – The architectural Drivers: This section describes Use-case diagram and Use-case descriptions. It makes easily to design and develop the proposed system. |
| **Tester** | The Overview section and Use-case: they will help to make the test plan and write the acceptance test |

# Project organization



## Organizational structure

This section of the SPMP identifies the external entities structure to the project and their interaction with the project team and gives a brief description of each of the roles held by the team members, as well as internal project structure and roles and responsibilities for the project.



All members have specified areas of responsibility and everybody contributes equally to the project. There are six roles assigned to the six team members: Project Leader, Architect Manager, Biller, Planning Manager, and Support Manager.

All team members may provide input to all decisions that the team makes; however, greater weight will be given to the recommendations of the appropriate manager. If there is a problem, issues will be resolved within the team and all members will approve the final decision. The team members will change roles throughout the life of the project, and each member may continue to have more than one role.

## Organizational Boundaries and Interfaces

The team will meet weekly with the mentor to report progress and discuss changes and progress possible and discuss possible changes and amendments. Major changes will affect the important events or major changes will affect important events must be approved by the whole team. From these documents it will be important issues are all members agree.

## Roles and responsibilities

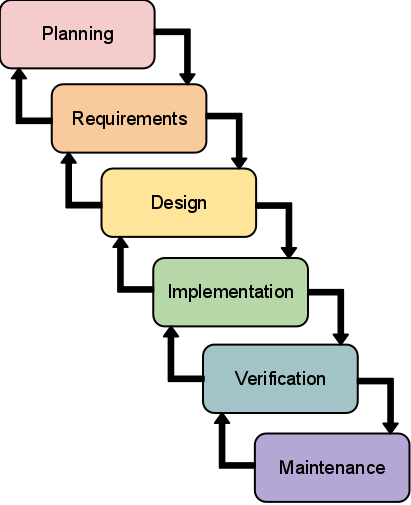
|  |  |
| --- | --- |
| Roles | Responsibilities |
| Managing engineer  (Thanh Giang) | * Coordinating the overall system design and development effort. * Responsible for the success (or failure) of the design team * Listen to other members of the design team * Plan, coordinate, track, and direct the overall activities of the design team * Creating and maintaining the programmatic plans and schedules in both the period of uncertainty and the period of certainty |
| Support engineer  (Phuc Nguyen) | * Setting up and maintaining the design team’s support tools and environments * Responsible for the system or product infrastructure or environment * Play a key role in the design of the system from a physical perspective |
| Chief architect  (Huy Huynh) | * Responsible for overall system design * Work with all of the other members of the design team to coordinate the system design * Provide enormous value throughout the system or product life cycle in managing change and evolution * Provide enormous value throughout the system or product life cycle in managing change and evolution |
| Requirements engineer  (Giang Nguyen) | * Leads the effort to gather and document the architectural drivers. * Help to manage the change and evolution of the architectural drivers * Serve as the primary customer liaison * Assist the quality engineer in coordinating architecture design review and in defining “black box” system or product tests |
| Chief scientist  (Dat Tran) | * Responsible for coordinating the planning, tracking, and documentation of experiments that are used to refine the architecture design * Focuses inwardly on technological issues that could impact the architecture * Assists the architect with detailed technical issues concerning architectural design. * Assists the quality engineer in the architectural design reviews and in the development of “clear box” tests |
| Quality process engineer  (Hiep Ta) | * Ensures that ACDM and other defined processes are followed as prescribed to ascertain project quality goals are met. * Responsible for coordinating architecture design reviews as well as product test development, planning, and execution. * Work with the requirements engineer and the chief scientist to coordinate the architecture design reviews and in planning product or system tests * Responsible for capturing, documenting, and tracking architectural issues uncovered during architectural evaluation * Work with the team to establish the processes for configuration management, defect tracking, and so forth that the design team uses |
| Production engineers  (HIT team) | * Focus on detailed design, implementation of the architectural elements, and integration of the elements to compose the system |

# Project Management

## Management Process

To complete this project, HIT team decided to use Waterfall model to develop Sales System of a retail chain using a loyalty card point system, because:

* It is linear and therefore very easy to be implemented, to understand and manage.
* Required amount of resources are minimal.
* Documentation is produced at every stage of waterfall model development.
* After every major stage of coding, testing is done to check whether the code running is correct or not.
* It’s easy to apply.
* Suitable for this project (small projects and requirements have not changed much).
* Output document of each stage are being built and more fully.

****

***Waterfall model***

## Project strategy

#### Within 6 week, Team decided:

Choose basic function to deliver to customer:

* Manage system
* Manage Customer
* Manage Product
* Manage Categories
* Manage Retail Store
* Manage Bill
* Log in
* Log out
* Change password

Because those functions are important function, easy to development and must have to operation Sales System of a retail chain using a loyalty card point system

Defer statistic function because this function do not so important, requirement about this function still not clear and hard to development

## Top 10 Risks:

|  |  |  |
| --- | --- | --- |
| Risk | Priority | Mitigation |
| Estimate the time is too far from real | **High** | Based on historical data, try to really precise estimates and keep actual not go too far for plan. |
| Customer change requirements or addition requirement | **Medium** | Actively persuading customers when changes occur and attract customers from the initial design of the first request. |
| Unskillful members-programming skill | **Medium** | Organizations working in small groups support each other in to cultivate programming skills. |
| Developing the wrong user interface | **Low** | Read carefully user requirement and organizations regular meeting with the customer to more understand. |
| Process is applied that is not really appropriate. | **Low** | Modify the process for appropriate of team and assign roles and responsibilities |
| Difficulty in module integrations with each other. |  | Apply continue integrations system to manage code integrations. Set up an effective SCM(software configuration management) early in project. |
| Choose wrong architecture drivers for the system | **High** | Discuss with team |

## Mile-stone:

|  |  |  |  |
| --- | --- | --- | --- |
| ID |  | Task | Mile-stone |
| 1 | RE | Propose | 10-3-2012 |
| Concept Of Operation | 13-3-2012 |
| SRS, Test plan | 17-3-2012 |
| 2 | Design | Architect Design | 24-3-2012 |
| Detail Design | 28-3-2012 |
| 3 | Implement | Manage System | 29-3-2012 |
| Manage Customers | 30-3-2012 |
| Manage Products | 1-4-2012 |
| Manage Categories | 2-4-2012 |
| Manage Retail Store | 3-4-2012 |
| Manage Bills | 4-4-2012 |
| Product - Version 1 | 5-4-2012 |
| 4 | Test | Unit test | 12-4-2012 |
| System test | 16-4-2012 |
| 5 | Delivery | Final project | 20-4-2012 |

# The architectural drivers



## High-level Functional Requirements

### Entity List

|  |  |
| --- | --- |
| Actor | Description |
| Administrator | Responsible for manage user of the system such as: Create new, assign authorize. He can also sync information between |
| Staff | Responsible for manage information of products, categories, customer, retail stores and they can statistic sales by many criterion |
| Cashier | Responsible for check bills (Bills) |
| User | Includes Staff, cashier and admin: they can use basic function like: log in, logout, change password |

### Use-case List

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID | Use Case Name | | Description |
| System Management | | | |
| UC\_SM01 | Add New User | | Allow Admin to add new user into the system |
| UC\_SM02 | Search/ View User List | | Allow Admin to search user of the system with name, job… and the system will display in the user list |
| UC\_SM03 | View User Log | |  |
| UC\_SM04 | View User Detail Information | | Allow Admin to View User Detail Information |
| UC\_SM05 | Update User Information | | Allow Admin to Update User Information and Assign Authorize |
| UC\_SM06 | Sync Information | | Allow Admin Sync all information between Head Office and Retail Store at the beginning of everyday |
| Product Management | | | |
| UC\_PM01 | Add New Product | | Allow Staff to add new product into the system |
| UC\_PM02 | Search/ View Product List | | Allow Staff to search product of the system with name, type… and the system will display in the product list |
| UC\_PM03 | View Product Detail Information | | Allow Staff to View Product Detail Information |
| UC\_PM05 | Update Product Information | | Allow Staff to Update Product Information |
| Bill Management | |  | |
| UC\_RM01 | Add New Bill | | Allow Cashier to add new bill into the system |
| UC\_RM02 | Search/ View Bill List | | Allow Cashier to search bill of the system with date, customer name… and the system will display in the product list |
| UC\_RM03 | View Bill Detail Information | | Allow Cashier to View Bill Detail Information |
| UC\_RM04 | Print Bill | | Allow Cashier to Print Bill |
| Retail Stores Management | | | |
| UC\_RSM01 | Add New Store | | Allow Manager to add new retail store into the system. They may configure what types will be sold in that store |
| UC\_RSM02 | Search/ View Store List | | Allow Manager to search store of the system with name… and the system will display in the product list |
| UC\_RSM03 | View Store Detail Information | | Allow Manager to View Store Detail Information |
| UC\_RSM04 | Update Store Information | | Allow Manager to Update Store Information |
| Category Management | | | |
| UC\_CM01 | Add New Category | | Allow Staff to add new type of product into the system |
| UC\_CM02 | Search/ View Category List | | Allow Staff to search product of the system with name… and the system will display in the type list |
| UC\_CM03 | View Category Detail Information | | Allow Staff to View Type Detail Information |
| UC\_CM04 | Update Category Information | | Allow Staff to Update Type Information |
| Customer Management | | | |  | Allow Staff to add new member into the system |
| UC\_C01 | Add New Customer | | Allow Staff to add new member into the system |
| UC\_C02 | Search/ View Customer List | | Allow Staff to search Member of the system with name, ID… and the system will display in the type list |
| UC\_C03 | View Customer Detail Information | | Allow Staff to View Member Detail Information |
| UC\_C04 | Update Customer Information | | Allow Staff to Update Member Information |
| UC\_C05 | View Customer Point Log | | Allow Staff View Member Point Log |
| UC\_S01 | Statistic Information | | Allow Manager to have the Analysis Statistic from sale of month, year… |

### Use-case Diagrams and Use-case Descriptions

***Note Table:***





**Level 1 – Retail Sales System**



**Level 2 - System Management**



**Level 2 - Customer Management**



**Level 2 - Product Management**



**Level 2 - Category Management**



**Level 2 - Bill Management**



**Level 2 - Retail Stores Management**



## Use-case Descriptions

|  |  |
| --- | --- |
| **Use Case Title:** Add New Store | **Use Case ID:** UC\_RSM01 |
| **General Use Case Description:** This use case help the manager add new sale store. This function can also split products of retail stores that are consumed, as well as regulate the product price. | |
| **Entities Involved:** Staff | |
| **Preconditions:**   * Staff is assigned authorized to use this function * Staff has chosen Retail Stores Management function * The system is available | |
| **Primary Use Case Flow of Events:**   1. Staff choose “Add new store” button 2. Program displays Add new store interface 3. Staff fills in Store name text field 4. Staff fills in Address text field 5. Staff chooses a Category 6. Program displays Products which has in that Category 7. Staff chooses a Product 8. Program add the product which user just has chosen to the Product List with the standard Price 9. Staff can edit the Price base on how much the store want to sell that product. 10. Then Staff choose specific period for the prize of that product: from <day> to <day> 11. Finally, Staff can write some information about this retail store: what type of food they sale… 12. Staff clicks "OK". 13. Program shows message "Are you sure you want to create new retail store’s information as above?" 14. Staff clicks "OK" to confirm 15. The system stores the retail store’s information that has just been created and returned to retail store management interface 16. End Use-case | |
| **Primary Use Case Post Conditions:**  Successful: Create new successful. The system stores the retail store’s information that has just been created and returned to retail store management interface  Fail: Failed to create new | |
| **Alternate Use Case #1 Flow of Events:**  There is a store with already name (starting from step 11 of the main flow)   1. Program shows message "Are you sure you want to create new retail store’s information as above?" 2. User clicks "OK" to confirm 3. Program check in database and found that it has a store with already name 4. Program display warning dialog “There is a store with already name. Please fill in again” 5. End Use-case | |
| **Alternate Use Case #1 Post Events:** | |

|  |  |
| --- | --- |
| **Use Case Title:** Add New Bill | **Use Case ID:** UC\_RM01 |
| **General Use Case Description:** This use case helps cashiers who work at retail stores make bill-paying for customers. Cashier can use barcode reader or directly enter the product code and product number by keyboard, then use the payment function to save to database and printed out bills for customers. | |
| **Entities Involved:** Cashier | |
| **Preconditions:**   * User is assigned authorized to use this function * User has chosen Bill Management function * The system is available. | |
| **Primary Use Case Flow of Events:**   1. User choose “Add new Bill” button 2. Program displays Add new store interface 3. User fills in Bar Code text field 4. User choose “Add” button 5. The system load information from database to the “Product List” table on the interface. 6. User fills in Quantity text field 7. User scan or fill in customer Loyalty card ID 8. The system load information from database and show customer name with their point on the interface 9. If customer want to use their point to pay bill, User choose “Type to pay bill” 10. There are three ways to pay bill:     1. Pay by cash: User does not fill in “Point” text field. The system auto shows all the money customer has to pay in “Cash” text field. Cashier type the amount of money that they received from customer, system will show the Spare Cash.     2. Pay by point: User fills in “Point” text field if user has enough point to pay that bill, the system auto shows “0 VND” in “Cash” text field.     3. Pay by point and cash: User fills in “Point” text field, the system shows money that customer still has to pay with that bill in “Cash” text field. 11. User choose “Pay by Cash” 12. User choose Pay bill button 13. The system stores the Bill’s information that has just been created and prints the bill. Then the program returns to Bill management interface 14. End Use-case | |
| **Primary Use Case Post Conditions:**  Successful: Create new successful. The system stores the Bill’s information that has just been created and prints the bill. Then the program returns to Bill management interface  Fail: Can’t print the bill | |
| **Alternate Use Case #1 Flow of Events:**  The bar code is invalid (starting from step 5 of the main flow)   1. Program loads information from database and shows message “The bar code is invalid” 2. User clicks "OK" to confirm 3. Program allows user to retype the bar code 4. End Use-case | |
| **Alternate Use Case #1 Post Events:** | |

## Constrains:

### Business Constrains:

|  |  |  |
| --- | --- | --- |
| Consideration | ID | Business Constraints. |
| Schedule limitations. | BC01 | Begin at 2/4/2012 to 5/7/2012 |
| Organizational restrictions and demands. | BC02 | One team with 6 members. Each team member must have effort time about thirty hours per week for this |
| Loyalty Point | BC03 | Plus Point will be save in the system and just can be used on the next day |

### Technical Constrains:

|  |  |  |
| --- | --- | --- |
| Consideration. | ID | Technical Constraints |
| Commercial hardware or software products. | TC01 | Bar code readers  Computers  SQL Server Database |
| Tools and methods. | TC02 | Visual Studio 2010, SQL Server Management Studio 2008.  .Net Framework |
| Protocols, interfaces, standards. | TC03 | TCP/IP protocol |
| Computer languages(s) | TC04 | C# |
| System | TC05 | There is a Head Office server and POS terminals at retail stores |

## Quality Attributes

### Performance:

This is sales system; we must always ensure that customer service was not delayed. Intervals such as scan the point card, bar code scanning, displaying results... will be done in the shortest time possible to avoid the dissatisfy customer. Things that need attention are: increased client response time, reduced throughput, and server resource over utilization. Ensure that you structure the application in an appropriate way and deploy it onto a system or systems that provide sufficient resources.

**Example of Performance Attribute:** Users initiate 100 transactions per minute at 100 POS terminal under overload operations, and these transactions are processed with an average latency of two seconds.

|  |  |
| --- | --- |
| Portion of scenario | Possible values |
| Source | User |
| Stimulus | 100 transactions per minute at 100 POS terminal |
| Artifact | System |
| Environment | Overload operation |
| Response | These transactions are processed |
| Response Measure | 2 seconds |

### Availability:

This system has two servers located at the head office server and POS terminal, if the failure occurs at head office server or the network; we need the system still available and working normal at the POS terminal.

**Example of Availability Attribute**: The temporary failure of the head office server, the POS terminal will be still working securely. The POST terminals can carry out the sales operation efficiently using locally stored data as much as possible.

|  |  |
| --- | --- |
| Portion of scenario | Possible values |
| Source | Head office database |
| Stimulus | Temporary failure of the head office server |
| Artifact | System |
| Environment | Normal operation |
| Response | To be still working securely and carry out the sales operation efficiently using locally stored data. |
| Response Measure | No down time |

## Prioritization

### Team priority (Difficulty ranking scale)

The difficulty scale has been defined on the basis of complexity and effort. Complexity is defined as how difficult the design of a solution is and whether the team has previous experience in designing or implementing such a design. Both measures, for complexity and effort, are relative to each other.

|  |  |
| --- | --- |
| Difficulty (numeric) | Description |
| 1 | High complexity and large amount of effort required |
| 2 | High complexity or large amount of effort required |
| 3 | Moderate complexity and medium amount of effort required |

### Stakeholder priority (Important scale)

The important scale has been defined on the basis of expectation and influence of stakeholder needs. It also describes the importance of quality attribute that follow in view of stakeholders.

|  |  |
| --- | --- |
| Important (numeric) | Description |
| 1 | High expectation and high influence |
| 2 | High expectation or high influence |
| 3 | Moderate expectation and medium influence |

### Priority scale

|  |  |  |
| --- | --- | --- |
| Priority (numeric) | Priority (name) | Description |
| 1 | Must Have | Must be present in the end product at all costs. |
| 2 | Nice to Have | Customer would greatly appreciate implementation of these features. |
| 3 | If There’s Time | Consider if customer deems them important enough. |

### Use Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID | Use Case Name | Important Level | Difficulty Level | Priority |
| System Management | | | |  |
| UC\_SM01 | Add New User | 1 | 2 | 1 |
| UC\_SM02 | Search/ View User List | 2 | 3 | 2 |
| UC\_SM03 | View User Log | 3 | 1 | 3 |
| UC\_SM04 | View User Detail Information | 2 | 3 | 3 |
| UC\_SM05 | Assign Authorize | 1 | 1 | 1 |
| UC\_SM06 | Update User Information | 2 | 2 | 2 |
| UC\_SM07 | Sync Information | 1 | 1 | 1 |
| Product Management | | | |  |
| UC\_PM01 | Add New Product | 1 | 2 | 1 |
| UC\_PM02 | Search/ View Product List | 2 | 3 | 2 |
| UC\_PM03 | View Product Detail Information | 2 | 3 | 2 |
| UC\_PM04 | Choose Category | 2 | 3 | 2 |
| UC\_PM05 | Update Product Information | 2 | 2 | 2 |
| Bill Management | | | |  |
| UC\_RM01 | Add New Bill | 1 | 1 | 1 |
| UC\_RM02 | Search/ View Bill List | 1 | 2 | 2 |
| UC\_RM03 | View Bill Detail Information | 1 | 2 | 2 |
| UC\_RM04 | Print Bill | 1 | 2 | 1 |
| Retail Stores Management | | | |  |
| UC\_RSM01 | Add New Store | 1 | 2 | 2 |
| UC\_RSM02 | Search/ View Store List | 2 | 3 | 2 |
| UC\_RSM03 | View Store Detail Information | 2 | 3 | 2 |
| UC\_RSM04 | Update Store Information | 2 | 2 | 2 |
| Category Management | | | |  |
| UC\_CM01 | Add New Category | 1 | 2 | 1 |
| UC\_CM02 | Search/ View Category List | 2 | 3 | 2 |
| UC\_CM03 | View Category Detail Information | 2 | 3 | 2 |
| UC\_CM04 | Update Category Information | 2 | 2 | 2 |
| Customer Management | | | |  |
| UC\_C01 | Add New Customer | 1 | 2 | 1 |
| UC\_C02 | Search/ View Customer List | 2 | 3 | 2 |
| UC\_C03 | View Customer Detail Information | 2 | 3 | 2 |
| UC\_C04 | Update Customer Information | 2 | 2 | 2 |
| UC\_C05 | View Customer Point Log | 1 | 2 | 1 |
| UC\_S01 | Statistic Information | 3 | 1 | 2 |

### Quality Attribute Scenarios

|  |  |  |  |
| --- | --- | --- | --- |
| ID & Title | Stakeholder priority | Difficulty ranking | Priority |
| QAS1 Sale product when database at Head Office crash | 3 | 3 | 3 |
| QAS2 Save Bill to the database | 1 | 2 | 1 |
| QAS3 Statistic bill | 2 | 2 | 2 |
| QAS4 Scan member Loyal | 1 | 2 | 1 |
| QAS5 Block unauthorized access | 1 | 1 | 1 |
| QAS6 Authority | 1 | 1 | 1 |

### Constraints

Constraints have not been prioritized in terms of importance because by definition they are of the highest priority. We have however prioritized them by difficulty, as judged by the development team.

### Technical Constraints

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Consideration | Difficulty ranking | Priority |
| TC01 | **Commercial hardware or software products.** | 1 | 1 |
| TC02 | **Tools and methods.** | 1 | 1 |
| TC03 | **Protocols, interfaces, standards.** | 2 | 2 |
| TC04 | **Computer languages(s)** | 1 | 1 |
| TC05 | **System** | 2 | 1 |

### Business Constraints

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Consideration | Difficulty ranking | Priority |
| BC01 | **Schedule limitations.** | 2 | 1 |
| BC02 | **Organizational restrictions and demands.** | 3 | 2 |
| BC03 | **Loyalty Point** | 1 | 1 |

## -- The End --