1. **Introductio**n
   1. **Purpose and Intended Audience**

The purpose of this document is to describe the software requirements of the system. It will explain the purpose and features of the system, what the system will do, the constraints under which it must operate and how the system will react to the stimuli. It is meant to be used to maintain a shared understanding of the requirements between the developers and the clients of the system.

* 1. **Scope**

The system can only be controlled or manipulated by the clerks. The clerks will input the persons who will be assigned on a counter for the processing of the transactions.

* 1. **Definitions, Acronyms and Abbreviations**

|  |  |
| --- | --- |
| Term | Definition |
| Queue | A system in which the arrangement of items to be processed are systematically arranged |
| Queue Managemenent System(QMS) | A system in which the arrangement of clients are shown in chronological order in which the processing is in a First Come First Serve Basis |
| Constraints | something that limits or restricts someone or something |
| Enqueue | A process in which an incoming transaction is saved for it to be processed later |
| Output Window | A monitor where the information that should be visible in the client side is there. |
| Requeue | a system that controls the way information is shown to a computer user and the way the user is able to work with the computer |
| Log | A file that contains the overall activities/transactions in a day |
| Counter | A location in which a transaction in between a client and an organization is held and is being handled by a clerk |
| SRS | A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document. |
| System | a group of related parts that move or work together |
| Transaction | a communicative action or activity involving two parties or things that reciprocally affect or influence each other |
| Script | a file that contains the tasks to be done by a certain System |
| Clerk | a personnel who is responsible for handling of the transactions with the client |

* 1. **Overview of the Document**

The next chapter, the Overall Description section of this document gives an overview of the product perspective, functions, user characteristics, general development constraints and project assumptions and dependencies.

The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the requirements and the functionality of the product.

1. **Overall Description**
   1. **Product Perspective and context**
      1. **System Interfaces**

One Stop Shop Queue Management System requires that there should be a Output Window in which the following transactions after the currently processed transaction should be displayed. The system is designed for the clerks that will handle the existing queue of transactions that will be processed that will be passed to another client instance via file transfer of scripts.

* + 1. **User Interfaces**

The interfaces will involve Tables and Buttons. The Tables are where the clerks can manipulate and view the upcoming transactions. The Buttons will be used for the selection of action in which there are following controls.

Open .qms script to process a script to enqueue the transactions to be processed in a window

Error Messages will be generated when the clerk immediately close the program instance without clearing the queues and logs to ensure the security of the queues.

Dequeueing action in which the transaction that is currently finished is either to be passed to another QMS or to be cleared in a system as a done transaction.  
  
Locking action in which the system will be locked to ensure that the QMS’s that are left are secured from unauthorized control.

Saving of the queue as a .qms to be ready for transfer in another QMS program instance.

There will only be one type of user, but there’s more than one employee to fill in different work time shifts, each user will access the screens according to their account after entering their id and passwords. Standard screen format (fixed colors, fonts, background, the page layout, etc.) will be used throughout the interfaces.

The language of the user interfaces will be English.

* + 1. **Hardware and Software Interfaces**

**Hardware –**

The application can run in windows and requires no high system requirement to be able to run.

There should be another monitor in which the main monitor is for the clerk’s control in the system and the another is for the client’s viewing of the upcoming clients who will be next in the transaction

**Software –**

1. The clerk should log in the system for security purposes before accessing the application.

2. The clerk should communicate with the client for the information about the client

3. The employee should communicate with the other clerk to transfer the next transaction to the other clerk’s window.

4. The client can view the client whose transaction will be next.

5. The clerks’ actions should be saved in a log file.

6. The clerks should tell the client in where he will be going next after processing his transaction

* + 1. **Communication Interfaces**

The One Stop Shop Queue Management System doesn’t require internet connection and only installation of the application is required. The employee is the only one who interacts with the system.

* + 1. **Memory Constraints**

The system will not use any additional resources only the business computer, except for another extra monitor and has enough physical memory to run the application.

* + 1. **Operational Constraints**

The clerks are the one who are allow accessing the application. The application is a computer based. The logs and scripts are stored in the computer and can be access directly by the employees.

* 1. **Product Function**

The system lets the clerks’ manage the QMS by making enqueue scripts and logs and controlling the queue. Then the system stores the clerks’ actions into to a log file or a script.

* 1. **User Characteristics**

User has to have at least Window OS that supports .NET Framework 4.0 for administrating the system. Thus, applicable users are at least those who have an idea what is a system and how to run a system.

* 1. **General Development**

The system can handle a single queue and can store information and reservations in the logs or script. The clerks’ can access the application easily by logging in.

The application has no problem in storing the information in the log file.

* 1. **Project Assumptions and Dependencies**

The application is expected to run in any computer that has enough memory to handle the processes. It can also handle multiple request and services of the clerks’ in an efficient and effective way.

The application can run in any operating systems that the business manager desires to use. It can also handle multiple tasks with no lags.

1. **Specific Requirements**
   1. **External Interfaces**
      1. **Hardware and Software Interface Specifications**

The application can be accessed only by the clerk using their respective employee ID and password. The log-in window is serves as the first interface of the application before going to the main interface.

After the employee logged in, QMS control form will appear. It has menus namely: File that has a submenu of Open .qms file, Save As .qms, Save As .log, and Exit. It has Buttons namely Request Transaction, Dequeue, Requeue. This also shows the queues of transactions to be done and the currently processed transaction.

The Transaction Request form is where the client’s information and the needs are located. The information from here will be stored in the queue of the system like clients’ name, and his order.

The Output Window form shows the currently processed client and his transaction, along with the upcoming transactions to be processed later.

Windows XP, Windows 7, Windows 8 are the only operating system that can install and run the application. Enough memory is required for running this system. No third party application is needed to be able to run the system.

The internet connection is not required and only need to install the application.

* + 1. **Detailed Description of input and output**

The information gathered in Request Reservation form will be stored in the database of the system. Reservation List is where the employee can view what is the available date for reservation by accessing the system’s database.

The Official Receipt form will show the clients’ payment and is ready to be printed out.

MS Access will serve as the system’s database that stores the information fetched in forms.

* 1. **Functional Requirements**

**•** Saves a back-up data for every transaction and other important details such as modification of the system.

• Gives a complete display of information on every transaction made.

• Displays clear list of reservation time and date for efficient monitoring.

• Issues a printed receipt to a client with a transaction number similar to the transaction number saved in the systems database.

* + 1. **Use Case Diagram**

**Records client request for**

**Reservation**

**Access account**

**Employee Maintains Casket**

**Maintains Service**

**Prints voucher**

* + 1. **Use Case Report**

**1. Use Case Name: Employee Log In**

**1.1 Brief Description:** The employee must log in first to create, edit and cancel a reservation for the client.

**2. Flow of Events**

**2.1 Basic Flow**

1. Employee Logs On.

The employee will enter his/her employee ID and password in order to have an access in the system.

**2.2 Alternative Flow**

2.2.1*Wrong Information*

At the employee log in, the system will determine that the employee ID or password are not valid, an error message is displayed. The system will load again the log in page.

2.2.2 *Quit*

The system allows the employee to quit at any time during the use case. The use case ends.

**Scenarios:** The Employee Log In: Basic Flow

Wrong Information: Basic Flow, Log In

Quit Before Completion: Basic Flow, Quit

**2. Use Case Name: Employee Access Account**

**1.1 Brief Description:** The interface of the system has a account menu where you can edit your password and can log out the account.

**2. Flow of Events**

**2.1 Basic Flow of Events**

1. Selects "Change Password".

The employee can change password of their accounts or log out their account.

2. Selects "Log Out".

The employee logs off the system. The system will load the log in page.

**2.2 Alternative Flow of Events**

2.2.1 *Quit*

The system allows the employee to quit at any time during the use case. The use case ends.

**Scenario:**

Quit Before Completion: Basic Flow, Quit

**3. Use Case Name: Request Reservation**

**1.1 Brief Description:** The system will load a reservation request for client reservation and a list of function.

**2. Flow of Events**

**2.1 Basic Flow**

1. Employee Selects "Add".

The employee chooses to Add". The system will load the reservation request page for client.

2. Employee selects "Edit".

The employee chooses "Edit Reservation". The system will load the edit reservation request page.

3. Employee selects "Cancel".

The employee chooses "Cancel". The system will load the cancel reservation request page.

4. Employee Selects "Save".

The system will save the given information and the request form. A message "Successfully save!” will be displayed.

5. Employee Selects a "Type of Casket".

The employee chooses "type of Casket". The system will load the type of casket available in another page.

6. Employee Selects "Services Offered".

The employee chooses "Service". The system will load the type of services available in another page.

7. Employee Selects “Payment".

The employee chooses “Payment". The system will load the mode of payment page.

**2.2 Alternative Flow**

2.2.1 *Incomplete Information*

At the reservation request, the system will determines if the information is incomplete, an error message is displayed, the will load again the reservation request page.

2.2.2 *Invalid Information*

In filling up the information needed, if the system determines the given information is not valid, an error message is displayed. The system will load the reservation request page.

2.2.3 *Forget to "Save"*

In reservation request, the employee selects "type of Casket" or "Service", if the reservation has not been save, an error message is displayed, the system will reload the reservation request page.

2.2.4 *Quit*

The system allows the employee to quit at any time during the use case. The use case ends.

**Scenario:**

Quit Before Completion: Basic Flow, Quit

**4. Use Case Name: Maintaining Casket**

**1.1 Brief Description:** The system will display a list of the available types of casket that the employee can maintain through edit, add and delete if it is no longer available.

**2. Flow of Events**

**2.1 Basic Flow of Events**

1**.** Employee Choose any Type of Casket.

The employee will select the type of casket is needed to maintain.

2. Employee Selects “Save”.

The employee selects “Save”. The system will save the new maintained service.

**2.2 Alternative Flow of Events**

2.2.1 *Quit*

The system allows the employee to quit at any time during the use case. The use case ends.

**Scenario:**

Quit Before Completion: Basic Flow, Quit

**5. Use Case Name: Maintaining Service**

**1.1 Brief Description:** The system will display a list of the available services that the employee can maintain through edit, add and delete if it is no longer available.

**2. Flow of Events**

**2.1 Basic Flow of Events**

1**.** Employee Choose any Type of Service.

The employee will select the type of service that is needed to maintain.

2. Employee Selects “Save”.

The employee selects “Save”. The system will save the new maintained service.

**2.2 Alternative Flow of Events**

2.2.1 *Quit*

The system allows the employee to quit at any time during the use case. The use case ends.

**Scenario:**

Quit Before Completion: Basic Flow, Quit

**6. Use Case Name: Payment**

**1.1 Brief Description:** The system will load the Payment page and let the client choose what mode of payment they want to acquire.

**2. Flow of Events**

**2.1 Basic Flow**

1. The Client Choose “Cheque”.

The employee selects “Cheque”. The cheque will be enabled and the client will input cheque no. and bank’s name.

2. The Client Choose “Cash”.

The employee selects “Cash”. The Client can choose if full or partial payment and the system will display the amount to be paid by the client.

3. Employee Selects “Ok”.

The system will load the official receipt and the Employee can print the receipt. The system will load the transaction page after.

**2.2 Alternative Flow**

2.2.1 *Quit*

The system allows the employee to quit at any time during the use case. The use case ends.

**Scenario:**

Quit Before Completion: Basic Flow, Quit

**2.1 Performance Requirements**

The system should give response to 95% of the requests in less than 15 seconds. The system should support the user actions.

**3.4 Database Requirements**

**3.4.1 Entity-Relationship Diagram**

PAYMENT

ReceiptNo

RequestNo

ReceiptDate

REQUEST

RequestNo

RequestDate

ClientFirstName

ClientMiddleName

ClientLastName

DeceaseFirstName

DeceaseMiddleName

DeceaseLastName

Street

City

Zip

ContactNo

EmailAddress

CasketCode

ServiceNo

DateofInterment

PartialAmountPaid

EmployeeId

EMPLOYEE

EmployeeId

FirstName

MiddleNameLastNaMe

Position

ContactNo

**Has**

**Assign**

**Has**

**Has**

**Has**

SERVICE

ServiceNo

ServiceDescription

ServiceAmount

CASKET

CasketCode

CasketType

CasketAmount

CHECK

ReceiptNo

CheckNo

BankName

**3.4.2 Data Dictionary**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Casket | | | | |
| Field Name | **Data Type** | **Field Size** | **Description** | **Other Information** |
| CasketCode | Text/Number | 5 | Identify the casket chosen | Primary Key Field  Required |
| CasketType | Text/String | 150 | Describe the type of casket | Required |
| CasketAmount | Currency |  | Shows the amount of the casket | Format: "Php"#,##0.00;"(Php"#,##0.00)  Required  Decimal Places: Auto |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Employee | | | | |
| Field Name | **Data Type** | **Field Size** | **Description** | **Other Information** |
| EmployeeID | Text/Number | 5 | Identify the employee in order to log-in in the system | Primary Key Field  Required |
| FirstName | Text/String | 20 | Supports the identity of the employee | Required |
| MiddleName | Text/String | 20 | Supports the identity of the employee | Required |
| LastName | Text/String | 20 | Supports the identity of the employee | Required |
| Position | Text/String | 20 | Identify the position of the employee in the company/business | Required |
| ContactNo | Number | 11 | Supports the identity of the employee | Required |
| Password | Number/Text | 255 | Supports the identity of the employee | Required |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Service | | | | |
| Field Name | **Data Type** | **Field Size** | **Description** | **Other Information** |
| ServiceNo | Number | 5 | Identify the service chosen | Primary Key Field  Required |
| ServiceDescription | Text/String | 255 | Describes the services offered of the business | Required |
| ServiceAmount | Currency |  | Shows the amount of the services offered | Format: Php"#,##0.00;"(Php"#,##0.00)  Required |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Payment | | | | |
| Field Name | **Data Type** | **Field Size** | **Description** | **Other Information** |
| ReceiptNo | Number | 10 | Shows the receipt number of the client | Primary Key Field  Required |
| RequestNo | Number | 10 | Shows the request number of the client | Required |
| ReceiptDate | Date/Time |  | Determines the date of the transaction | Format :MM/DD/YY  Required |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Cheque | | | | |
| Field Name | **Data Type** | **Field Size** | **Description** | **Other Information** |
| ReceiptNo | Number | 10 | Shows the receipt number of the client | Required |
| ChequeNo | Number | 12 | Shows the cheque number of the client | Required |
| BankName | Text/String | 25 | Determines the bank name where the client gets the money in order to pay the bill from the funeral | Required |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Request | | | | |
| Field Name | **Data Type** | **Field Size** | **Description** | **Other Information** |
| RequestNo | Number | 5 | Shows the request number of the client | Primary Key Field  Required |
| RequestDate | Date/Time |  | Shows the date of transaction | Format: MM/DD/YY  Required |
| ClientFirstName | Text/String | 20 | Supports the identity of the client | Required |
| ClientMiddleName | Text/String | 20 | Supports the identity of the client | Required |
| ClientLastName | Text/String | 20 | Supports the identity of the client | Required |
| DeceasedFirstName | Text/String | 20 | Supports the identity of the client | Required |
| DeceasedMiddleName | Text/String | 20 | Supports the identity of the client | Required |
| DeceasedLastName | Text/String | 20 | Supports the identity of the client | Required |
| Street | Text/String | 20 | Supports the identity of the client | Required |
| City | Text/String | 20 | Supports the identity of the client | Required |
| Zip | Number | 4 | Supports the identity of the client | Required |
| ContactNo | Number | 11 | Supports the identity of the client | Required |
| EmailAddress | Number/Text | 30 | Supports the identity if the client | Required |
| CasketCode | Number | 5 | Identify the casket chosen | Required |
| ServiceNo | Number | 5 | Identify the service chosen | Required |
| DateOfInternment | Date/Time |  | Identify the date of internment of the deceased person | Required |

* 1. **Object Model**
     1. **Per Use-Case Activity Diagram**

1. **Employee Log In**

Quit

Log In

Incorrect

Account not found

Correct

1. **Employee Access Account**

QUIT

Log out

Change Password

1. **Request Reservation**

Add

Edit

Cancel

Services Offered

Save

Type of Casket

Quit

Error message

Error message

Incomplete Information/Invalid information

Forget to save

**4. Maintaining Casket**

Type of Casket

Save

Quit

**5. Maintaining Service**

Type of Service

Save

Quit

**6. Payment**

Cash

Type of Payment

Quit

Cheque

Total Payment

Bank name

Cheque Numeber

* + 1. **Per Use-Case class diagram**

Employee

EmpID

EmpFname

EmpMname

EmpLname

EmpAddress

EmpAge

EmpPass

Save()

Employee Log In

EmpID

EmpPass

Log In()

Use Case: Employee Log In

Logs -in

1..1

1..1

Employee Access Account

Employee

EmpID

EmpPass

Login()

Account

Password

LogOut() Save()

Edit()

1..1

1..1

Access

**Use Case: Request Reservation**

Request Reservation

**.**

**.**

**.**

Save() Add() Edit() Cancel()

Service

.

.

.

Save()Add() Edit() Cancel()

Type of Casket

.

.

.

Save()Add() Edit() Cancel()

Employee

**.**

**.**

**.**

Save()

Deceased Information

.

.

.

Save() Add() Edit() Cancel()

Client Information.

.

.

.

Save()Add() Edit() Cancel()

Mode of Payment

.

.

.

Save() Assess()

1..1

1..1

Request

1..1

1..1

Log in

Record

1..1

1..1

1..1

1..1

1..1

1..1

Save

1..1

1..1

1..1

Save

**Use Case: Maintaining Casket**

Employee

.

.

.

Save()

Casket

.

.

.

Save()Add() Edit() Cancel()

1..1

1..1

Maintain

1..1

1..1

Maintain

1..1

1..1

Maintain

1..1

1..1

Maintain

1..1

1..1

Maintain

**Use Case: Maintaining Service**

Employee

.

.

.

Save()

Service

.

.

.

Save()Add() Edit() Cancel()

1..1

1..1

Maintain

**Use Case: Payment**

Request

.

.

.

Save() Add() Edit() Cancel()

Voucher

.

.

.

Save() Print() Cancel()

Employee

.

.

.

Save()

Mode of Payment

.

.

.

Save()Assess() Cancel()

1..1

1..1

Save

1..1

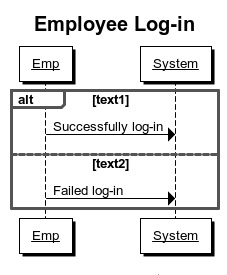
1..1

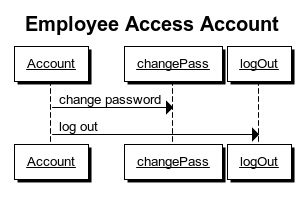
1..1

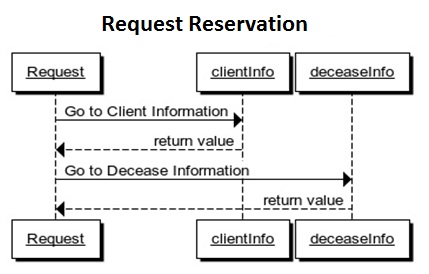
Save

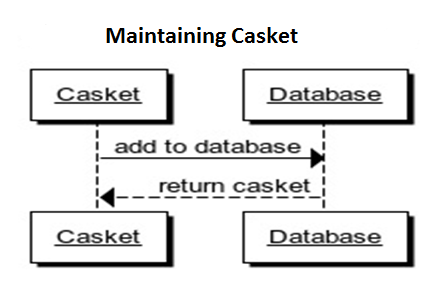
1..1

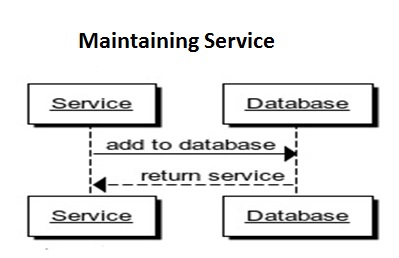
Record

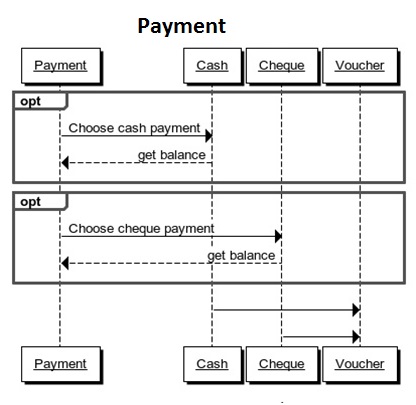
**3.5.3 Per UC communication diagram / sequence diagram**

****

****

****

****

****

* 1. **Software System Attributes**

**Reliability**

The system has to operate with no issues’ during the transaction hours. The number of defect should not exceed 10 per function.

**Security**

The system has an authorization mechanism for users to identify their personal profiles. Therefore, different users will have different authorization levels to access the data.

**Maintainability**

The system can meet the changing requirements easily, since the infrastructure of the system would not need major changes. The requirements of the software while evolving will be met by just adding new sub-functions.

**Availability by 24/7**

Users can access the system any time of the day.

**Response time not over 3 seconds**

Response time is assured not to last longer than 3 seconds, because of its nature that runs as an executable file and not web based.

1. **Supporting Details**
   1. **Appendices**
   2. **Index**

**C S**

Casket page 1 Software page 3

Communication Interfaces page 3 System Interfaces page 2

**D U**

Data Dictionary page 11-14 Use Case Diagram page 6

Use Case Report page 6

**E** User Characteristics page 3

Entity-Relationship Diagram page 10

External Interfaces page 4  **V**

Voucher page 1

**F**

Functional Requirements page 5

**H**

Hardware page 2

**I**

IEEE page 1

**M**

Memory Constraints page 3

**O**

Operational Constraints page 3

**P**

Product Function page 3