**SOFTWARE REQUIREMENTS SPECIFICATION**

**PROJECT: [Optimal Services Discovery for Visitors of Makkah (OSD)]**



**Document Control**

|  |  |  |  |
| --- | --- | --- | --- |
| Document ID | Version | Author/Editor | Date |
|  |  |  |  |

**TABLE OF CONTENTS**

[1. Description 4](#_Toc342728864)

[1.1. Business perspective 4](#_Toc342728865)

[1.2. Business Product/Service functions 4](#_Toc342728866)

[1.3. User classes and characteristics 4](#_Toc342728867)

[2. System Functions 5](#_Toc342728868)

[2.1. System function 1 5](#_Toc342728869)

[2.1.1. Identifier 5](#_Toc342728870)

[2.1.2. Description and priority 5](#_Toc342728871)

[2.1.3. Action/result 5](#_Toc342728872)

[2.1.4. Functional requirements 5](#_Toc342728873)

[3. Nonfunctional Requirements 7](#_Toc342728874)

[3.1. Performance requirements 7](#_Toc342728875)

[3.2. Safety requirements 7](#_Toc342728876)

[3.3. Security requirements 7](#_Toc342728877)

[3.4. Software quality attributes 7](#_Toc342728878)

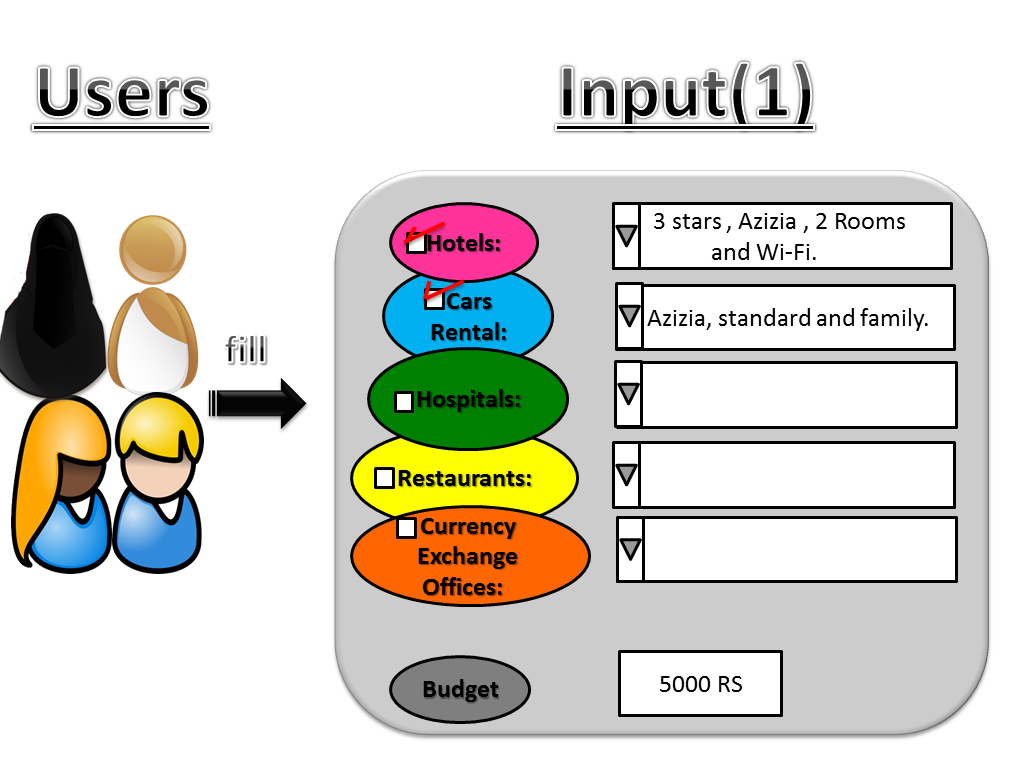
[3.5. Other Operational requirements 8](#_Toc342728879)

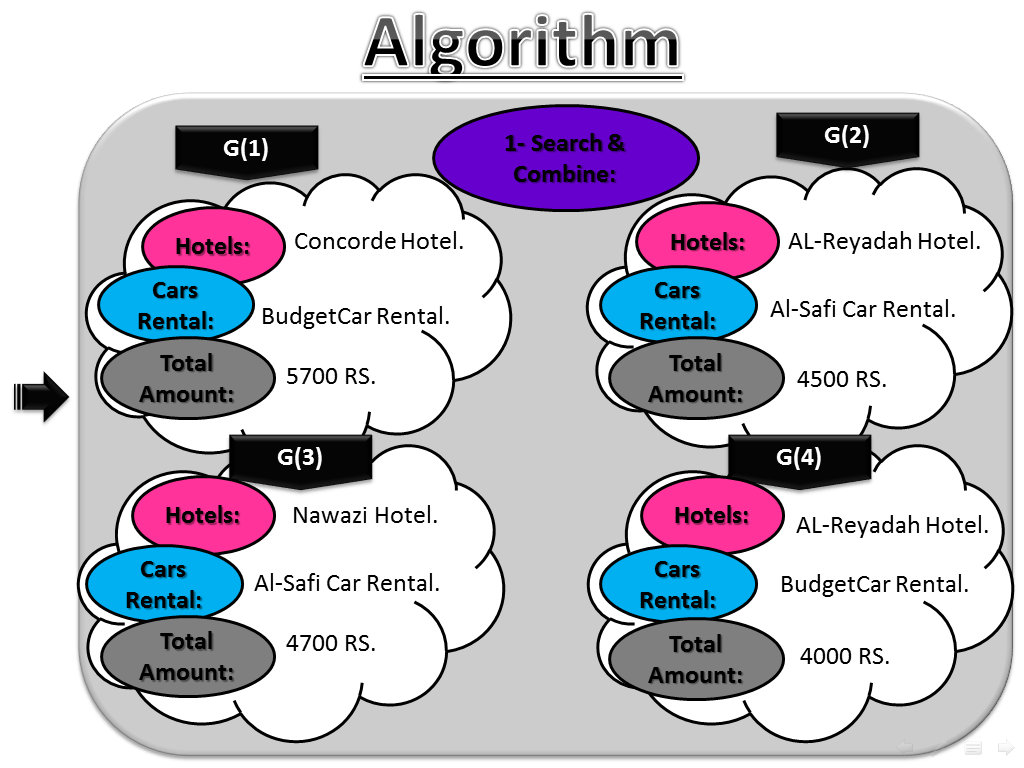
# Description

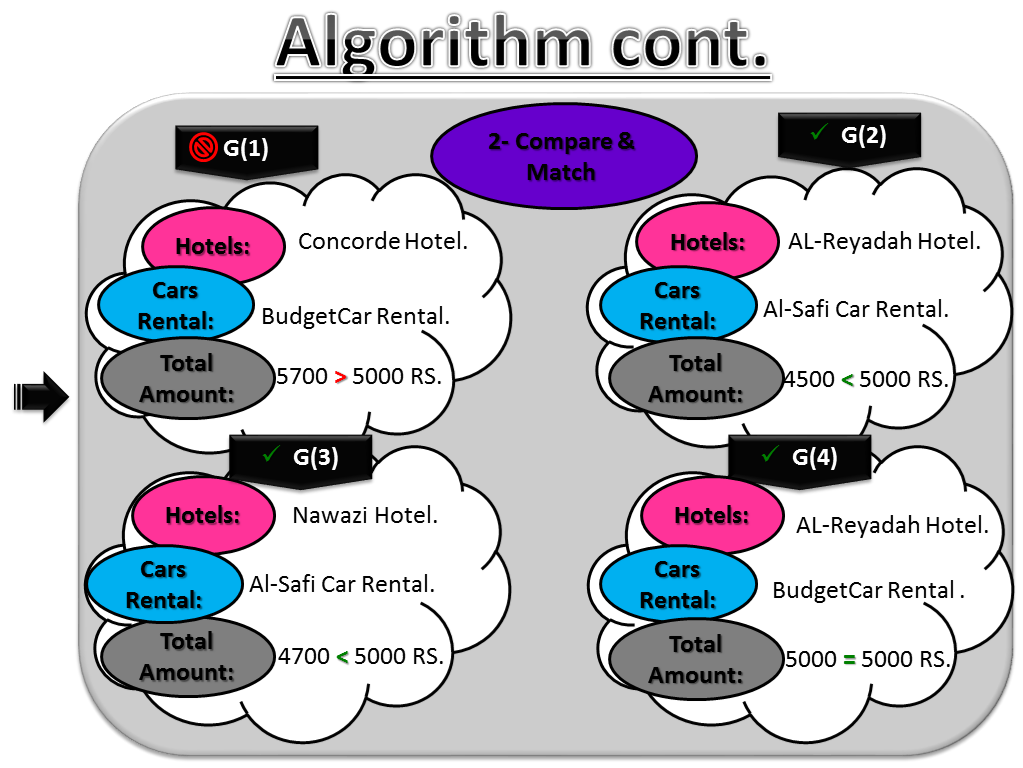
## Business perspective

Our project helps people find necessary visitor services including hotels and car rentals in Makkah using one system, saving visitors time and effort. This gives visitors greater opportunity to enjoy their experience visiting Makkah and explore more of the city and its offerings. There are many websites that can help visitors get the information they want, but as these system focus on individual services (hotel, car rental, hospital, restaurant and currency exchange office) like Booking website, searching for these services becomes time-consuming and wastes visitors’ time. To solve this problem and meet visitors’ needs, we will provide one system for visitors to find every service needed.

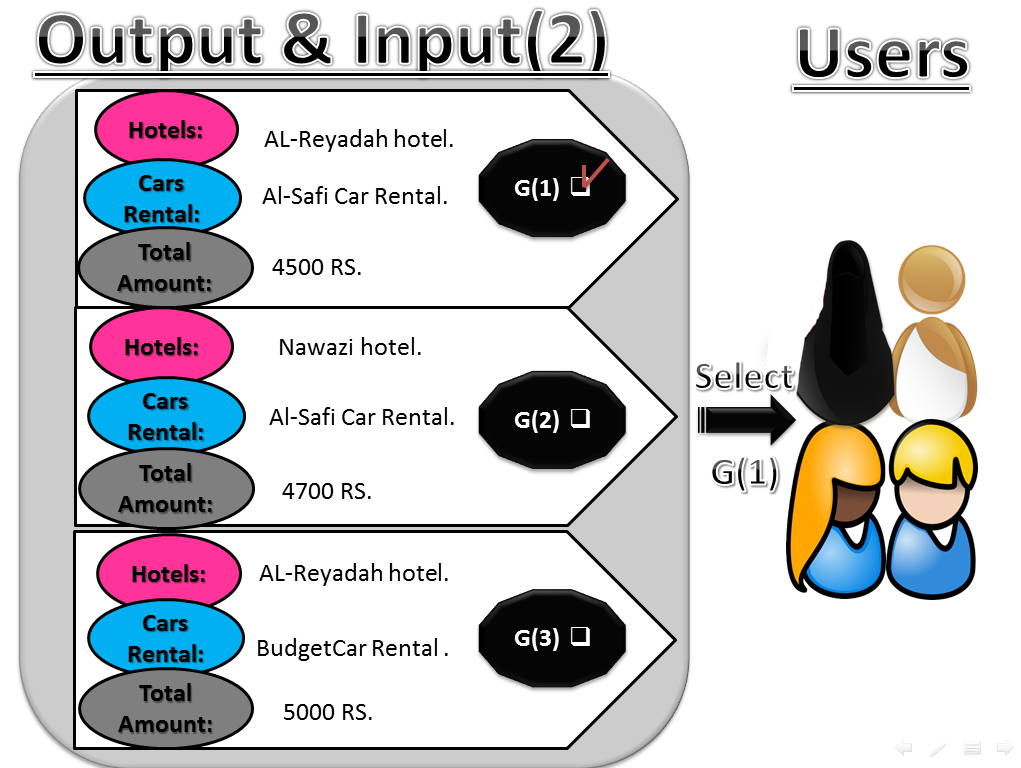
**Sketch describe the System Work (Simple Example):**

**Screen (1):** User fill specifications and budget in the form.

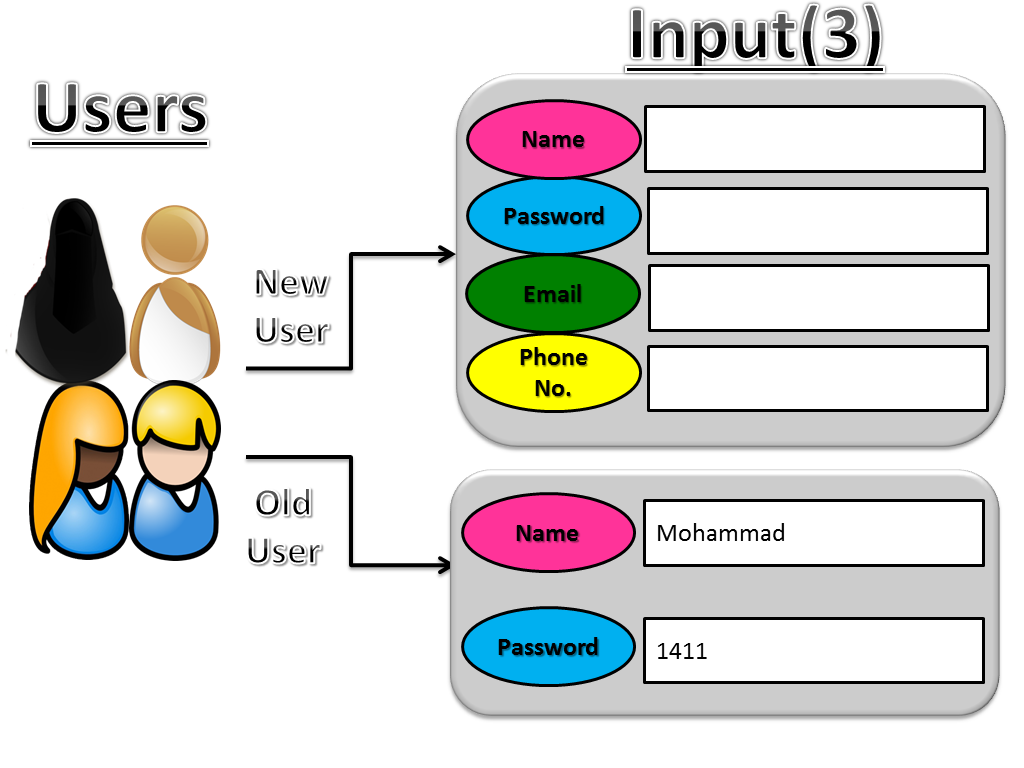
**Screen (2):** Shows how the algorithm works step (1) : search & combine.

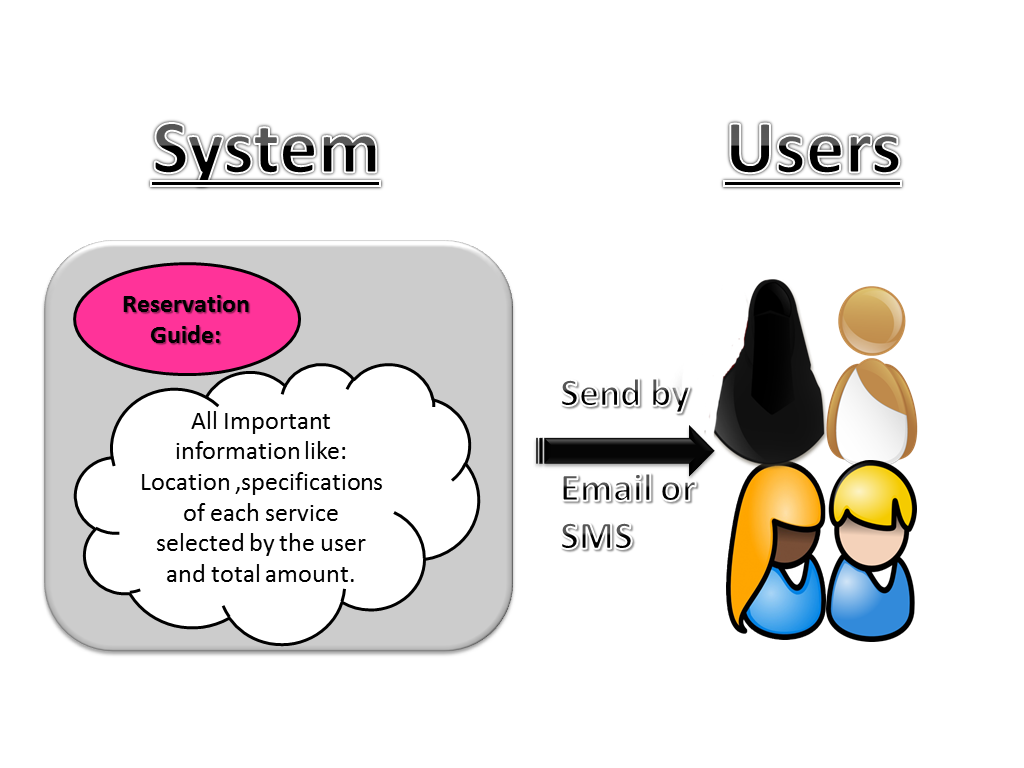
**Screen (3) :** Shows how the algorithm works step (2) : compare & match the total amount with user budget.

**Screen (4):** User select one group from their 3 groups.

****

**Screen (5):** User registers their information if new user or login if old user .

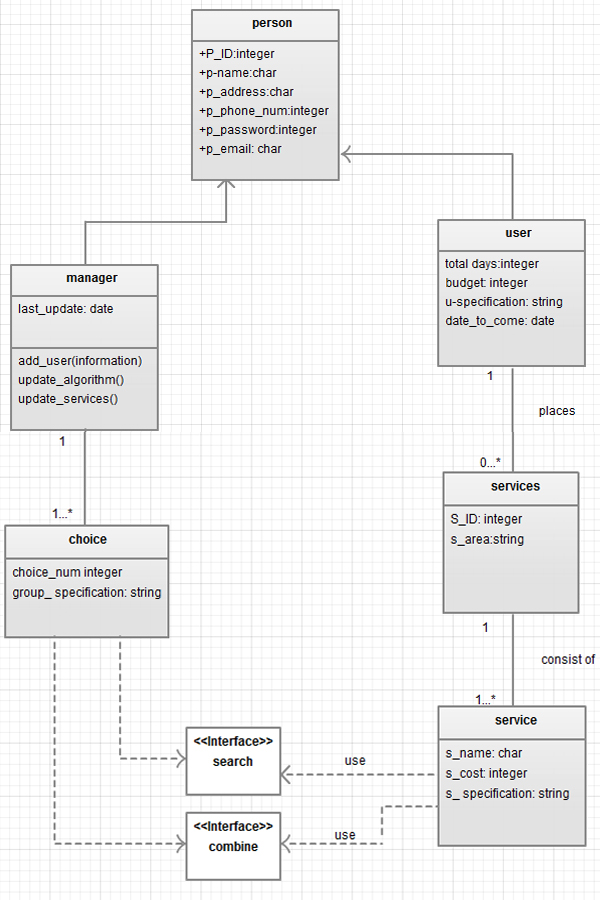
****

**Screen (6):** In the end the system send to user reservation guide by email or SMS.

## Business Product/Service functions

Our goal is to create a system that gives the visitors the best combination of services. We will use the mathematical method to create a new algorithm that obtains the visitors' requirements and gives them the best combination of required services.

The novelty in our idea is to get a collection of services that will be searched in reduced time. Additionally, we will provide users with extra information about services in the region as well, such as which one is the nearest the hotel, restaurant, ATM, hospital and so on.

**Class diagram **

## User classes and characteristics

**USE CASE DESCRIPTION:**

**Brief Description for Each Use Case:**

1. **User Specifications:**

When the user installs the system correctly and runs it, the user needs to fill the form that has one field for each service (user need to choose at least two services) and one field for the budget. Then the system takes these required specifications and saves them.

On the manager side, the manager can change the services form because there may be times where we need to add or change services attributes.

1. **Mathematical Algorithm:**

When the system sends the user specifications to the algorithm, the algorithm starts a search for each service individually to find the three or four best combinations of services depending on user specifications. The algorithm then saves these groups of services before displaying them to the user.

The manager can update some parts of the algorithm to improve the work of the algorithm.

When a user chooses one group of services, the system will save the user’s choice and display the total cost of this group to the user.

1. **User Choice:**

When user choose one group of services then the system will save the user choice and display the total amount of this group to the user.

1. **Register Operation:**

When the system takes the user’s consent, the system then asks the user to enter their information if this is the first time s/he has used our system. Otherwise, users may update their information.

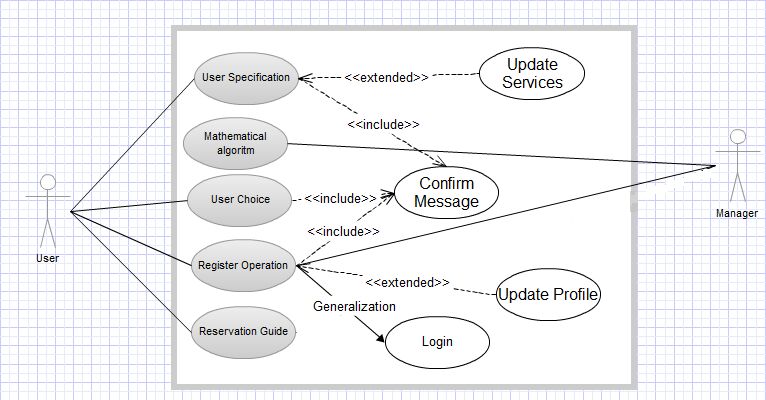
The manager can update the database when needed.

1. **Reservation Guide:**

When saving the user information finished by the system, the system sends a reserve guide for each service to the user via email or SMS.

USE CASE DIAGRAM:

Identify external and internal factors that affect the system such as the user and manager, used to collect system requirements (a mathematical algorithm, register operation).



# System Functions

## User Specifications

### Identifier

*REQ\_FUNC\_1.0.*

### Description and priority

When the user installs the system correctly and runs it, the system takes the required specifications from the user and saves them. On the manager side, the manager can change the services form. This function conducted as high priority feature.

### Action/result

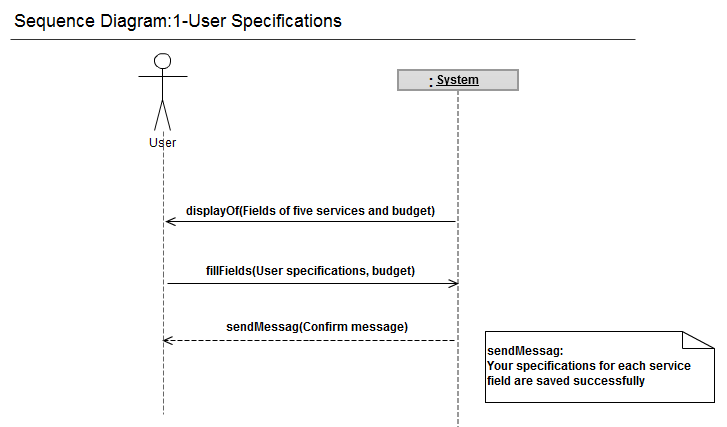
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Event | Trigger | Source | Use Case | Response | Destination |
| Install  system | Start system | User | User specifications | 1-Take the required specifications from user  2- Save these specifications | System  Manager |

### Functional requirements

REQ\_FUNC\_1.1. : When the user installs the system correctly and runs it, the user shall be able to fill the form that has one field for each service (user need to choose at least two services) and one field for the budget.

REQ\_FUNC\_1.2. :The system shall be able to take the required user specifications and saves them.

REQ\_FUNC\_1.3. :The manager shall be able to change the services form because there may be times where we need to add or change services attributes.



## Mathematical Algorithm

### Identifier

*REQ\_FUNC\_2.0.*

### Description and priority

System receives user specifications and apply some sort of algorithm to find three or four best combinations of services depending on user specifications. This function conducted as high priority function.

### Action/result

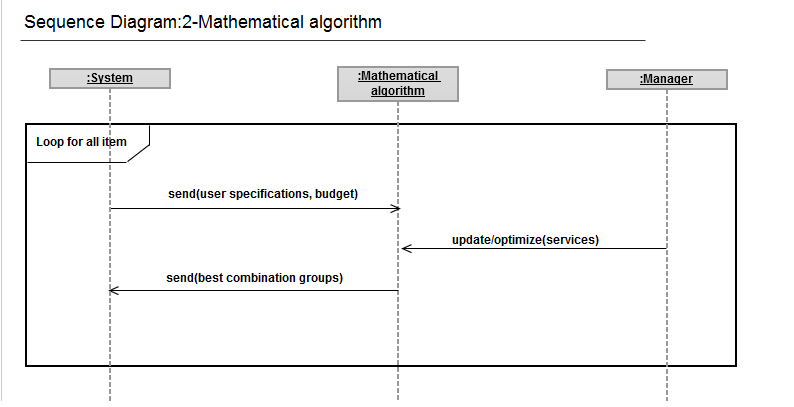
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Event | Trigger | Source | Use Case | Response | Destination |
| Send the user  specifications | User  specifications | System | Mathematical algorithm | 1-Start search about  each service  2- Save result of all best group combination services  3- Give the user the best combination of services | User  Manager |

### Functional requirements

REQ\_FUNC\_2.1. : When the system sends the user specifications to the algorithm, the algorithm shall be able to start a search for each service individually to find the three or four best combinations of services depending on user specifications.

REQ\_FUNC\_2.2. : The algorithm shall be able to saves these groups of services before displaying them to the user.

REQ\_FUNC\_2.3. :The manager shall be able to update some parts of the algorithm to improve the work of the algorithm.



## User Choice

### Identifier

*REQ\_FUNC\_3.0.*

### Description and priority

When user choose one group of services then the system will save the user choice and display the total amount of this group to the user. This function conducted as high priority function.

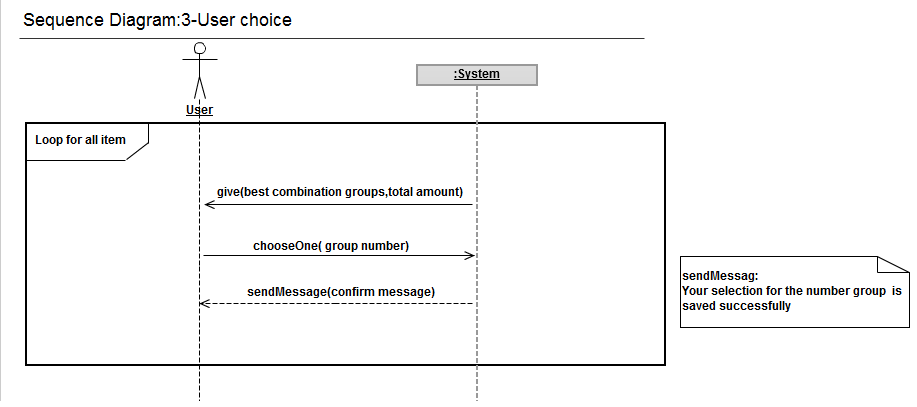
### Action/result

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Event | Trigger | Source | Use Case | Response | Destination |
| User chooses one group of services | Choice number | User | User choice | 1-Save the user choice  2-Display the total amount of this group | System |

### Functional requirements

REQ\_FUNC\_3.1. :The user shall be able to choose one group of services offered by the system.

REQ\_FUNC\_3.2. : The system shall be able to save the user’s choice and display the total cost of this group to the user.



## Register Operation

### Identifier

*REQ\_FUNC\_4.0.*

### Description and priority

When the system takes the user’s consent, the system then asks the user for their information or update them if already registered. This function conducted as medium priority function.

### Action/result

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Event | Trigger | Source | Use Case | Response | Destination |
| Ask the user to enter their information if this is the first time s/he has used our system,  or update their information | User information | System | Register  Operation | 1-Take the user’s information  2-Save the user’s information  with the choice | User  Manager |

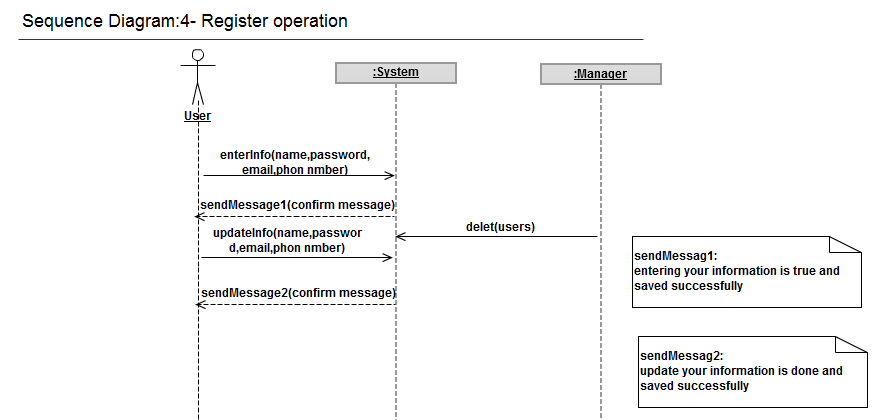
### Functional requirements

REQ\_FUNC\_4.1. :The user shall be able to take the user information if it is the first time s/he has used the system.

REQ\_FUNC\_4.2. : The user shall be able to update the user information if s/he has used the system.

REQ\_FUNC\_4.3. : The manager shall be able to *update the database when needed.*

.



## Reservation Guide

### Identifier

*REQ\_FUNC\_5.0.*

### Description and priority

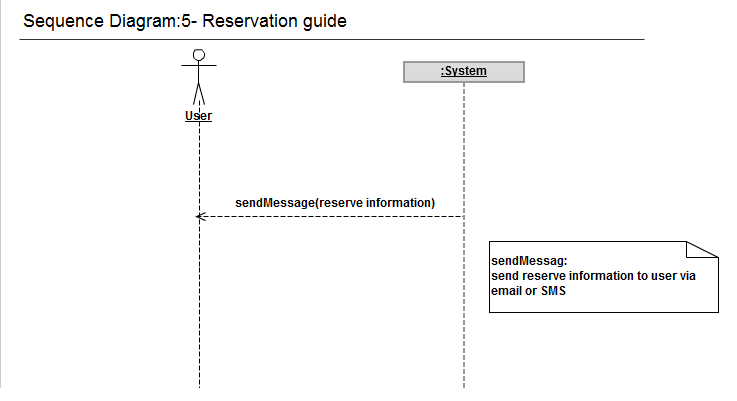
When saving the user information finished by the system, the system sends a reserve information to the user.This function conducted as medium priority function.

### Action/result

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Event | Trigger | Source | Use Case | Response | Destination |
| Give the Reserve information to the user | Reserve guide | System | Reservation guide | Send reserve information to the user | User |

### Functional requirements

REQ\_FUNC\_5.1. : The system sends a reserve guide for each service to the user via email or SMS.

****

# Nonfunctional Requirements

## Performance requirements

* Time: a short response time will complete users’ requirements and give them the best solution.
* Quality: we will fulfill user requests completely if able, or we will suggest the closest services available for them. Therefore, the user will be satisfied with our services.

## Safety requirements

* Availability:

It will be available free to all persons on the internet.

* Reliability:

System reliability depends not only on the reliabilities of the components in the system but also on their interactions.

1. There is a service with real information in it.
2. User will get the best combination of services, get results and at the end, choose the services.
3. We will provide the user the companies in Makkah, that provide the best services.
4. We must update the available information to ensure the validity and accuracy of the information.

## Security requirements

1. Users give us their information. This information will stored in a database; then, we will use this information to send the user the services.
2. We will keep this information safe from spyware or theft.
3. We will provide integrity by not allowing anyone to modify the information and only allowing the user modification access.

## Software quality attributes

* Maintainability:

Our system needs to be updated frequently, about every three or four months to:

* Add new services or meet new requirements.
* Correct defects or their cause.
* Make future maintenance easier.
* Cope with a changed environment.
* Portability:

Our project will be implemented as a system installed in computers.

## Other Operational requirements

* Database Requirements:

The database should be very extensive because different information is stored in the database, such as:

* User information, including name, email, etc.
* Information about the services and most common companies that offer the best services, including name, type of service, the cost of the service, etc.

We will use SQL database or access system.

* Domain Requirements:

Specify the system work in which environment that we need in our system.