Software Requirements Specification

For JCI Events Project

Version 1.3

Prepared by Karam Mustafa

JCI Damascus – JT Team

1\4\2022

Table of Contents

Contents

Revision History

Name	Date	Reason For Changes	Edited by	Version
JCI Events	1/4/2022	Initial	Karam Mustafa	1.0
JCI Events	5/4/2022	Review	Abdulrhman kujjah	1.1
JCI Events -users	8/4/2022	Add new column	Karam mustafa	1.2
JCI Events -users	14/4/2022	Add new use case	Abdulrhman kujjah	1.3

1. Introduction

JCI Events project is a project that within JCI software open source contributions program, this project use to organize the attendance of events for members of JCI Damascus

1.1 Purpose

The Purpose of this document is to describe the first release of JCI Events project; it is intended for developers and stakeholders as well.

1.2 Product Scope

The purpose of first release of system is to manage food menus, Tables and chairs and reservation of seating places by members at the inserted tables by the system.

1.3 References

All the requirement come from the local president of JCI Damascus (Shadii Azmeh) and local director of JCI technology (Karam Mustafa)

1.4 Product Perspective

The First release of system will consist of below parts:
1- Publish the project to Syrian open source platform
2- Table and chairs management
3- Report for the attendance
4- User Accounts
5- Menu management

1.5 Product Functions

- *The system admin should be able to generate a new user from the admin panel and generate the passwords.*
- *The system admin should be able to generate and edit and delete a new food menu*
- *The system admin should be able to generate and edit and view and delete a new food*
- *The system admin should be able to generate chairs automatically when each food table stored*

1.6   User Classes and Characteristics

There are two roles:
1.6.1   Super Admin (control everything, and it's a default user for the developers)
1.6.2   Admin (a person that add tables, menus, events, and members)
1.6.3   Member (a person who register his food and his place on table).

1.7   Assumptions and Dependencies

- system shows any interface of system for any USER depending on his time zone. The system converts the time from time zone of creator to time zone of user.

- In all view interface, the member cannot update or delete any info

- In all interfaces, system should display an helper text to help the member in using

2. External Interface Requirements

2.1 User Interfaces

2.1.1. User Management

2.2 Hardware Interfaces

This part is related with the host

2.3 Software Interfaces

2.3.1 Backend technologies: laravel php framework
2.3.2 Frontend technologies: blade engine from laravel framework and admin LTE package
2.3.3 Database: MySQl
2.3.4 Server: sub domain in shared host

2.4 Communications Interfaces

2.4.1 Standards: HTTPS connection

3.   System Features

3.1. User Management

3.1.1.   Description

In this use case, the super admin should be able to add a new admin, this admin can add the other management system such as menu and tables and member users.

3.1.2.   Functional Requirements

- FR#1: Add new admin
- FR#2: Edit admin
- FR#3: Delete admin
- FR#4: View admin

3.1.3.   Use Cases

1- UC#1: User Management

Use Case Name	User Management		
Version	**Description**	**Changed By**	**Date**
1.0	First Version	Abdulrahman	5/4/2022
Brief Description:	This use case is just a crud case to manage the admins		
Business Trigger:	A super admin inserts a new admin		
Preconditions:	The user login the system and has super admin and admin role		
Actor:	Super Admin		

Basic Flow:

1. **System display interface to add new admin for the system**
2. **The super admin enters the name**
3. **The super admin enters the email**
4. **The super admin enters the GSM**
5. **The super admin enters the password**
6. **The System will consider the role is [admin] only**

Assumptions: -	
Post Condition:	The use case ends.

Alternate Flow (AF2): select edit options.	
If the super admin select one admin record and choose the edit button	
Line	**System Actor Action**	**System Response**

	Actor press edit button	system display the admin record
	Actor edit the record info and press the save button	System save the new update in the database and response with success message.

Alternate Flow (AF2): select delete options.

If the super admin select one admin record and choose the delete button

Line	System Actor Action	System Response
	actor press delete button	System display confirmation message.
	Actor choose yes button	System will delete the admin record from the database

Alternate Flow (AF2): select view options.

If the super admin select one admin record and choose the edit button

Line	System Actor Action	System Response
	All records will displayed from the system in a table	All records will displayed from the system in a table

Alternate Flow (AF11): display error message.

Store the error message in the session.

Line	System Actor Action	System Response
10		**If** the system cannot save the info, it displays error message **in the header of the page**

2- *UC#2: User Registration*

Use Case Name	User Management			
Version 1.0	**Description** First Version		**Changed By** Abdulrahman	**Date** 14/4/2022
Brief Description:	This use case is just an interface to manage the member registration			
Business Trigger:	Member will use the register interface			
Preconditions:	Member opened the registration interface			
Actor:	Member			

Basic Flow:

1. **System display interface to Register user in the system**

2. **The Member enters the name**

3. **The Member enters the email**

4. **The Member enters the GSM**

5. **The Member enters the password**	
6. **The Member confirm the password**	
7. **The System will consider the role is [member] only**	

Assumptions: -	
Post Condition:	1- If member is new then insert his/her information in user's database and send confirmation email. 2- If member is not new (already registered) and is not verified by using email confirmation then resend confirmation email and show interface to check email inbox to verify the email. 3- If member is not new (already registered) and he/she verified his/her email by clicking on the verification link that received on his/her email then show error (member already registered and verified).

3- UC#3: Email Confirmation

Use Case Name	User Management		
Version	**Description**	**Changed By**	**Date**
1.0	First Version	Abdulrahman	14/4/2022
Brief Description:	This use case is just an email will be sent to the member's email		
Business Trigger:	New member has been registered successfully and still not verified		
Preconditions:	Member should open his/her email and check inbox from JCI		
Actor:	Member		

Basic Flow:	
1. **Member check his/her email inbox**	
2. **The Member opens the verification email.**	
3. **The Member clicks on verification button in the body of the email**	

Assumptions: -	
Post Condition:	1- If verification link is still active and less than expiration date then the system will change the status of member to verified 2- If verification link is not active then the system resend a confirmation email to the member's email.

3.1.4. Business Rules

Business Rules	
BR ID	**BR Description**
1	Add a new admin
2	The email must be unique
3	Must be determine the role

3.1.5. Data Dictionary

Field	Type	M|O	Default value	notes
Name	String	M	Null	
GSM	String	M	Null	Mobile number
Email	String	M	Null	Must be unique
Password	String	M	Null	Must be hashed
Role	String	M	Null	Must be in [admin, member]
Verified_at	Datetime	O	Null	
isApproved	Tiny int	M	0	

3.2. Food Event Management

3.2.1. Description

In this use case the admin must be able to create a new event and do the crud operation on it, this use case use to create food event, this event has a menu and tables and each member will be able to choose the table and his food.

3.2.2. Functional Requirements

- FR#1: Add new admin
- FR#2: Edit admin
- FR#3: Delete admin
- FR#4: View admin

3.2.3. Use Case

3.2.3.1.   *UC#1: Food Event Management*

Use Case Name	Food Event Management		
Version	**Description**	**Changed By**	**Date**
1.0	First Version	Karam	3/4/2022
Brief Description:	This use case is just a crud case to manage the food events		
Business Trigger:	An admin add a new food event		
Preconditions:	The user login the system and has super admin and admin role		
Actor:	Super Admin, Admin		

Basic Flow:

4. **System display interface to add a new food event for the system**

5. **The actor enters the title**

6. **The actor enters the description**

7. **The actor enters the cover**

8. **The actor adds start date**

9. **The actor adds end data**

Assumptions: -	
Post Condition:	The use case ends.

Alternate Flow (AF2): select view option.	
If the actor enter to the use case tab	
Line	**System Actor Action**	**System Response**

Line	System Actor Action	System Response
	All records will displayed from the system in a table	All records will displayed from the system in a table

Alternate Flow (AF2): select edit option.	
If actor select one admin record and choose the edit button	

Line	System Actor Action	System Response
	Actor press the edit button	The system check if the event still exist and update the info

Alternate Flow (AF2): select delete options.	
If the super admin select one admin record and choose the edit button	

Line	System Actor Action	System Response

	Actor choose an active food event record to delete it	System check if this food events active and there is any member has registered on it by check if the member has chosen any food or chairs that related to this food event
		System return error message if the food event is active.
	Actor choose an active food event record to delete it	System delete the food event if there are no members have registered on it

3.2.4   Business Rules

Business Rules	
BR ID	**BR Description**
1	Add a new food event
2	The event food must not be deleted if there are any members choose from its info any table of food from the menu.
3	The end date should be after the start date
4	The start date must be bigger than the created date.

3.2.5   Data Dictionary

Field	Type	M|O	Default value	notes
title	String	M	Null	
description	String	M	Null	
cover	String	M	Null	Must be image
Start date	timestamp	M	Null	Must be bigger that created date
End date	timestamp	M	Null	Must be after the start date
Is active	Boolean	M	False	

3.3 Menu Management

3.3.1 Description:

let's suppose we have a food list and as a member, the system should be able to allow me choose foods from the menu list, I login and choose the food from the menu that relative with a custom event, so this use case descript the event management from the admin and registration by user.

3.3.2 Functional requirement

- FR#1: Add new menu
- FR#2: View the menu
- FR#3: Delete the menu
- FR#4: Edit the menu
- FR#5: Authenticated member view the active menu
- FR#6: Choose the food as a logged in member

3.3.3 Use Case

3.3.3.1 UC#1: Menu Management by admin.

Use Case Name	Menu Management			
Version	**Description**		**Changed By**	**Date**
1.0	First Version		Karam Mustafa	4/4/2022
Brief Description:	This use case is just a crud case to manage the menu			
Business Trigger:	An admin add a new menu			
Preconditions:	The user login the system and has super admin and admin role			
Actor:	Admin			
Basic Flow:				
10. This use case is separated is handle from two table, the first table is the menu and the second one is the food items				
11. The actor chooses to create a new menu				
12. The actor enters the menu title				
13. The actor enters a note if exists.				
14. The actor choose whatever this menu belongs to which event and theses event should be active events				
Assumptions: -				
Post Condition:	The use case ends.			

Alternate Flow (AF2): select delete options.
If the super admin select one admin record and choose the delete button

Line	System Actor Action	System Response
	Actor choose an active food menu record to delete it	System check if this food menu active and there is any member has registered on it by check if the member has chosen any food or chairs that related to this food menu
		System return error message if the food menu is active.
	Actor choose an active food menu record to delete it	System delete the food event if there are no members have registered on it

Alternate Flow (AF2): select edit options.		
If the super admin select one admin record and choose the edit button		
Line	System Actor Action	System Response
	Actor press edit button	System is updated the title only

Alternate Flow (AF2): select view options.		
If the super admin select one admin record and choose the edit button		
Line	System Actor Action	System Response
	View the data in the table	

3.2.3.2.   UC#2: Add Food items to a menu.

Use Case Name	Menu Management		
Version	**Description**	**Changed By**	**Date**
1.0	First Version	Karam Mustafa	4/4/2022
Brief Description:	Add new food items to an event.		
Business Trigger:	An admin add a new food items		
Preconditions:	The user login the system and has super admin and admin role		
Actor:	Admin		
Basic Flow:			
15. The actor press on menu card			
16. The actor press on add new item button			
17. The actor enters the title			
18. The actor enters the description			
19. The actor enters the menu id			
Assumptions: -			
Post Condition:	The use case ends.		

Alternate Flow (AF2): select delete options.		
If the super admin select one admin record and choose the edit button		
Line	**System Actor Action**	**System Response**
	Actor press delete button	System check if this item is not selected by any user, then it delete it, otherwise system return error message.

3.2.3.3.   UC#3: Member choose food item

Use Case Name	Member choose food item			
Version	**Description**		**Changed By**	**Date**
1.0	First Version		Abdulrahman	5/4/2022
Brief Description:	An member login and choose food items.			
Business Trigger:	An member select food items within max plates (n)			
Preconditions:	Authenticated member			
Actor:	Member			
Basic Flow:				
20. The actor logged in				
21. The actor goes into active event				
22. The actor goes into menu				
23. The actor selects whatever he wants from list (with limitation less than max plates(n))				
24. The actor presses on save button				
Assumptions: -				
Post Condition:	The use case ends.			

3.2.3.4.   UC#4: view selected food items

Use Case Name	view selected food items			
Version	**Description**		**Changed By**	**Date**
1.0	First Version		Karam Mustafa	4/4/2022
Brief Description:	An admin wants to view the food items inside a menu and view each item and the count of users that choose this item			
Business Trigger:	Ad admin view the food items			
Preconditions:	Authenticated admin			
Actor:	Admin			
Basic Flow:				
25. The actor logged in				
26. The actor goes into active event				

27. The actor goes into menu	
28. The actor goes into food menu items	
29. The actor view each food item with the count of the users that choose this item with the user details	
Assumptions: -	
Post Condition:	The use case ends.

3.3.4. Business Rules

Business Rules	
BR ID	**BR Description**
1	Admin mustn't update the food items if there is any member has selected this food item
2	The interfaces must not show un active events
3	The user can select more than one food items, so the relation is many to many between the food items and members
4	The member cannot edit his selection after submit it.

3.3.5. Data Dictionary

3.3.5.1. Menu table

Field	Type	M|O	Default value	notes
title	String	M	-	
Event id	Foreign key	M	-	
Max_plates	int	M	null	
Notes	Text	O	null	

3.3.5.2. Food Items table

Field	Type	M|O	Default value	notes
title	String	M	-	
description	text	M	-	
Menu id	Foreign key	M		

3.3.5.3. Food items member table

Field	Type	M|O	Default value	notes
Food item id	Foreign key	M	-	
User id	Foreign key	M	-	

3.4. Food Table Management

3.4.1. Description

This use case is representing the member chair selection, the admin is insert a new food table and choose the chairs number on this table, and the authenticated member login and choose the table and the chair, and see the people on this table.

3.4.2. Functional Requirements

- FR#1: View table and chairs
- FR#2: Insert a new table
- FR#3: Edit Table
- FR#4: Delete table
- FR#5: Authenticated member view the tables
- FR#6: Authenticated member can choose table and received a chair

3.4.3. Use case

3.4.3.1. UC#1: Food Table from admin panel

Use Case Name	Food Table Management

Version	Description		Changed By	Date
1.0	First Version		Karam Mustafa	5/4/2022
Brief Description:	This use case is just a crud case to manage the food table			
Business Trigger:	An admin add a new food table			
Preconditions:	The user login the system and has super admin and admin role			
Actor:	Admin			

Basic Flow:

30. This use case is separated is handle from two table, the first table is the food table and the second one is the chairs

31. The admin chooses to create a new food table

32. The admin enters the menu title

33. The admin chooses whatever this food table belongs to which event and theses event should be active events

34. The actor chooses the numbers of chairs

35. The actor press save button

    35.1.   System should save chairs records depending on chairs count that actor choose

    35.2.   Example: user type food table 1 and chairs count is 10, now system should generate one record for food table in database, and 10 records for chairs with food table id

Assumptions: -	
Post Condition:	The use case ends.

Alternate Flow (AF2): select view options.	
If the super admin press in food table tab	
Line	**System Actor Action**	**System Response**

Line	**System Actor Action**	**System Response**
	All records will displayed from the system in a table	All records will displayed from the system in a table

Alternate Flow (AF2): select edit options.	
If the super admin select one admin record and choose the edit button	

Line	**System Actor Action**	**System Response**
	An actor press edit button	System check if this food table there is any member has registered on it by check if the member has chosen any chair that related with this food table or chairs that related to this food table

		System return error if the process cannot be handled
		System save the update if the process is allowed

Alternate Flow (AF2): select delete options.

If the super admin select one admin record and choose the delete button

Line	System Actor Action	System Response
	An actor press delete button	System check if this food table there is any member has registered on it by check if the member has chosen any chair that related with this food table or chairs that related to this food table
		System return error if the process cannot be handled
		System delete the record if the process is allowed

3.4.3.2. *UC#2: Member view the tables and each table with the members on it*

Use Case Name	**Member view the tables and each table with the members on it**		
Version	**Description**	**Changed By**	**Date**
1.0	First Version	Karam Mustafa	5/4/2022
Brief Description:	This use case is for the displaying the tables from the client side		
Business Trigger:	An Member view the tables		
Preconditions:	Authenticated member		
Actor:	Member		

Basic Flow:

36. Actor logging

37. Actor choose tables card

38. Actor view the tables as cards and each card should display the title, count of available chairs and member names on this table.

Assumptions: -

Post Condition:	The use case ends.

3.4.3.3. *UC#3: Member choose a chair from an available food table*

Use Case Name	**Member choose a chair from an available food table**

Version	Description		Changed By	Date
1.0	First Version		Karam Mustafa	5/4/2022
Brief Description:	This use case to choose a chair from available table.			
Business Trigger:	An Member choose a chair			
Preconditions:	Authenticated member			
Actor:	Member			
Basic Flow:				
39. Actor logging				
40. Actor choose tables card				
41. Actor view the tables as cards and each card should display the title, count of available chairs and member names on this table.				
42. After that the actor decide a table to set on it, and press the select a chair (should be in the bottom of table card)				
43. After that system should save the actor id on this chair record				
Assumptions: -				
Post Condition:	The use case ends.			

3.4.4. Business Rules

Business Rules	
BR ID	**BR Description**
1	Admin isn't able to update the food table if there is any member has select this food item
2	The interfaces must not show un active food tables (food table related with un active event)
3	The use can only select one table to seat
4	The member can't update his info after he received a chairs

3.4.5. Data Dictionary

3.4.5.1. *Food table*

Field	Type	M|O	Default value	notes
Event id	Foreign key	M	-	

title	Foreign key	M	-	
Chairs count				

3.4.5.2.   *Chair table*

Field	Type	M|O	Default value	notes
User id	Foreign key	M	-	
Food table id	Foreign key	M	-	

4. Nonfunctional Requirements

4.1. Availability

- System should available 24/7

4.2. Security

Appendix A: Glossary

Here we list all the terminology that is used throughout the document: