1-Functional User Requirements

Function	Function Requirement Description
Add Food Items	Admin can add food items details like name, Photo, Description/ Ingredients, Price, Category etc.
View / Edit (Price, Enable/Disable) / Delete Items	Admin can view, edit the food items details as well as Enable/Disable food items according to season or availability
View Orders	Admin can track live orders and Status Update for particular food delivery
View Transactions	View previous transaction by date search and total amount at end.
View Users	Admin can view the users' details which are given during registration.
Register	If the user doesn't have an account so he should register. He has to register with essential details for food ordering system.
Login	User have to login with their credentials to access food ordering system like (username, password).
Menu List	The user can view the menu which has different food items with respective categories viewable to user.
Item Details	User can view food items details and buy the product by doing online payment.
View Cart	User can view their cart details, delete food items from cart, update quantity etc.
View Orders and Status	User can keep track of their orders with delivery status.

2-Nun-Functional User Requirements

User Interfaces.	There is interface for Web that is easy to use.
Performance.	The application is dependent on internet connection so strong or medium stable internet connection is required.
Operational & Environmental	The system will be used in a standard office environment, except that high levels of background noise may occur.
Reliability & Availability.	24/7 application is available also customer service available for being in touch with users.
Usability.	Using simple interface. Application does not require any technical knowledge and can be accessed by anyone with any device.
Security.	The system has a big security and no one can access the user data except the admin.

3- Functional System Requirements

Function	Function Requirement Description
Add Food Items	Admin can add food items details like name(name of food item exist under the food item photo), Photo(admin can add food items photo which attract the attention of the user), Description/ Ingredients(admin can add description to the application), Price(admin can add the food items price), Category etc.
View / Edit (Price, Enable/Disable) / Delete Items	Admin can view, edit the food items details as well as Enable/Disable food items according to season or availability.
View Orders	Admin can track live orders and Status Update for particular food delivery. So as not to have any defects during live orders and if that happened he shall solve this defect quickly.
View Transactions	Admin can view previous transaction by date search and total amount at end.
View Users	Admin can view the user's details which are given during registration.
Register	If the user doesn't have an account so he should register. User have to register with essential details like (FirstName, LastName, Email, Password, confirm password, phone number, Address).
login	If user is already registered so he have to login by using his username and password .User have to login with their credentials to access food ordering system.
Menu List	The user can view the menu which has different food items with respective categories.
Item Details	User can view food items details and buy the product by doing online payment.
View Cart	User can view their cart details, delete food items from cart, update quantity etc.
View Orders and Status	User can keep track of their orders with delivery status using the Web APIs.

4. Non-Functional System Requirements

User Interfaces.	-There is interface for Web that is easy to use, The user interface shall be implemented using any tool or software like (HTML, php). Achieved in the project by making it easy to use, does not require any technical skills.
Performance.	-Performance is how fast does the system return results? How much will this performance change with higher workloads? 1- The application is dependent on internet connection so strong or medium stable internet connection is required. 2- The Performance is dependent on hardware components of the customer. 3-The system shall produce the schedule within 3 seconds of the user's request. 4-The system shall calculate a guest's bill in 2 seconds.

	5-The system shall handle up to 100 users simultaneously. 6-The system shall, on average, operate without failure for 2 months.
Operational & Environmental.	1-The system will be used in a standard office environment, except that high levels of background noise may occur. 2-The system will use three colors in addition to many photos.
Reliability & Availability.	Reliability is how often does the system experience critical failures? And how much time is it available to users against downtimes. 1- In case of failure the system shall return to normal after a maximum of 5 minutes. 2-24/7 application is available also customer service available for being in touch with users. 3-The rate of failure occurrence is (16.6%) where the failure shall occurred in average every 2 months.
Usability.	Usability is how easy is it for a customer and user to use the system? 1-Using simple interface. 2-Application does not require any technical knowledge and can be accessed by anyone with any device. 3-The application shall provide a uniform look and feel between all the pages. 4-The application shall provide multi-language support. 5-The application staff shall be able to use all the system functions after 4 hours of training. After this training, the average number of errors made by experienced users shall not exceed 2 per hour of system use.
Security.	1-Admin who has the right to access to data & functionality continuously. 2-No one can access to data except the admin by entering the password.

A textual description of a use case in the online ordering food system domain

Identifier and name	The user order food online .
initiator	The user .
Goal	The user buy an order and the order arrives to him .
precondition	The user must have registered before using the system .

Postcondition	The user paid for his order and receive the order .
Assumptions	The user not registered to the system and not able to login (see step 1).

Main success scenario :

1 . the user registers to be able to log into the system .

2 . the user view food items details and add items to his cart then review his cart details before sending his order .

3 . the user pays for order and sends his order .

4 . the admin receive for order and view transactions and users .

5 . the user receives a confirmation and can keep track his order , then receive his order .

Extensions :

1.a . the user can not log in the system because he did not register before or enter wrong data .

2.a . the user want to change his order before sending the order .

2.a.1 . the user update the quantity , add items or delete items .

4.a . according to season and availability the admin want to change the items .

4.a.1 . the admin add new food items, edit/delete food items, Enable/Disable the food items .

4.a.2 . the admin change the price or make it (Enable /Disable) .