Requirements Document

<Project name>

<Team Name>

# Business Requirements

## Introduction

## Statement of Problem or Need

*“The problem of [statement of problem] . . .”: What’s the goal or need of the audience; what’s the root cause problem your target market is trying to solve? What do stakeholders struggle with; what concerns do they have? Which detailed or lower-level problems are buried under bigger problems? Are the important issues called out?*

*“. . . affects [users and/or stakeholders] . . .” Who specifically is impacted? This list isn’t limited to just the problem-sufferer; it includes others also affected directly or indirectly.*

*“. . . the impact of which is [statement of issues, costs, or other impacts].” Lacking a solution (preferably yours), what are the impacts and outcomes suffered as a result of the business problem? (In other words, what’s the business driver; why may the company be compelled to solve the problem?)*

*“A successful solution would be to [important benefits that would successfully solve the problem].” What characteristics and qualities does an acceptable and/or compelling solution include? What would a (generic) solution do or have that addresses or resolves key issues and needs? A solution providing these benefits or capabilities would be just what the users need, with post-solution results or outcomes adding value.*

*A good example of a problem statement may be*

*The problem of customers smoking in our rooms affects other customers, who don’t appreciate the smoke and smell, and our housekeeping staff, who spend significantly more time cleaning smoking rooms versus nonsmoking ones, the impact of which is low customer satisfaction, reduced occupancy rates, and increased cleaning costs. A successful solution would be to eliminate smoking and smoking effects from our hotel rooms and readdress those impacts.*

*Source: McGoey 2013*

## Business Requirements

### List of stakeholders

Identify the main stakeholders

### Client

Identify your contact person who will represent one of the stakeholders.

### List of business requirements

List of business processes

### Quality requirements

List of non-functional requirements, i.e. security requirements and other e.g. quality requirements (e.g. availability, portability)

# Business Solution

## Options considered

* Option 1
* Option 2
* Option x

## Recommended Solution

### Solution statement

Solution position statement

*“For [target audience] . . .” State the target user, market, or customer by naming its role or characterizing its group.*

*“ . . . who [statement of the need or opportunity] . . .” Describe what the target is trying to achieve or do while it’s having the problem. Rather than state the problem directly, state the need your target audience will have the opportunity to meet.*

*“ . . . the [name of new product or business] is a [solution or business category] . . .” Set the context for the solution by portraying the kind of product, solution, or new business venture (if pursuing large opportunities) it is. State the category the solution belongs to so the audience members understand its relevance and relationship to their needs.*

*“ . . . that [statement of key benefit — the compelling reason to use the solution or do business with you].” Highlight what the solution offers that will motivate or most interest the audience. This piece is the must-have outcome of value — the persuasive benefit. Provide the reason why this solution is the right solution by stating how the solution meets the need.*

*“Unlike [primary competitive alternative] . . .” If the target didn’t have your solution, what solution(s) may it look for or consider instead? State the current environment or competitor products to beat, or internally, the unacceptable situation (what the company has now) to fix.*

*“ . . . our product [statement of primary differentiation].” Distinguish and sell the solution. Besides the fact that it meets the most critical need, what other interesting or attractive features are included? List key marketing points or functionality that addresses the problem statement. Establish why stakeholders would want or need this function. What sets it apart? What are the key differentiators?*

*Source: McGoey 2013*

### Main Features

List three main features

### Unique value proposition / Justification

What makes your recommended solution unique?

Why do you recommend this one?

# Solution Requirements

## Context Diagram

## User Roles

1. **Customer/User:**
   1. Responsibilities:
   * Browsing and searching for restaurants.
   * Viewing restaurant menus, ratings, and reviews.
   * Placing orders for food.
   * Managing their user profile, including adding payment methods and delivery addresses.
   * Rating and reviewing restaurants.
   * Adding and removing favorite restaurants.
   1. Permission
   * Access to browse restaurants and menus.
   * Access to order and pay for food.
   * Access to user profile management.
2. **Restaurant Owner/Partner:**
   1. Responsibilities:

* Listing their restaurant on the app.
* Managing their restaurant's menu items, prices, and descriptions.
* Receiving and processing incoming orders.
* Tracking order statuses.
* Responding to customer reviews.
  1. Permissions:
* Access to create and edit restaurant listings.
* Access to manage menu items and prices.
* Access to view and full fill incoming orders.
* Access to respond to customer reviews.
* Limited access to user data for order full fillment.

1. **Administrator/Support:**
   1. Responsibilities:

* Administering the platform.
* Handling customer support inquiries and issues.
* Managing restaurant listings, including adding, editing, and removing restaurants.
* Monitoring and addressing any fraudulent or inappropriate behaviour.
  1. Permissions:
* Full access to manage all aspects of the platform.
* Access to user data for support and administration purposes.
* Access to restaurant listings for moderation.

## Business Domain Model

Entity Relationship Diagram

1. **Entities:**
   * **User**
     + Attributes:
       1. User ID (Primary Key)
       2. First Name
       3. Last Name
       4. Email Address
       5. Phone Number
       6. Address
       7. Payment Methods
   * **Restaurant**
     + Attributes:
       1. Restaurant ID (Primary Key)
       2. Name
       3. Description
       4. Location
       5. Owner
     + Relationships:
       1. One-to-Many with Menu Item (A restaurant can have many menu items)
   * **Menu Item**
     + Attributes:
       1. Menu Item ID (Primary Key)
       2. Name
       3. Description
       4. Price
     + Relationships:
       1. Many-to-One with Restaurant (Many menu items belong to one restaurant)
   * **Order**
     + Attributes:
       1. Order ID (Primary Key)
       2. User (Foreign Key)
       3. Restaurant (Foreign Key)
       4. Order Date
       5. Status (e.g., Processing, Delivered)
       6. Total Amount
       7. Delivery Address
       8. Payment Method
     + Relationships:
       1. Many-to-One with User (Many orders can belong to one user)
       2. Many-to-One with Restaurant (Many orders can belong to one restaurant)
   * **Review**
     + Attributes:
       1. Review ID (Primary Key)
       2. User (Foreign Key)
       3. Restaurant (Foreign Key)
       4. Rating
       5. Comment
     + Relationships:
       1. Many-to-One with User (Many reviews can belong to one user)
       2. Many-to-One with Restaurant (Many reviews can belong to one restaurant)
   * **Favorite Restaurant**
     + Attributes:
       1. User (Foreign Key)
       2. Restaurant (Foreign Key)
2. **Business Rules and Logic:**
   * + - * Users can register, log in, and manage their profiles, including adding payment methods and favorite restaurants.
         * Restaurants can list their menus, which consist of multiple menu items.
         * Users can place orders at restaurants, specifying delivery addresses and payment methods.
         * Users can rate and review restaurants they've ordered from.
         * Users can add and remove restaurants from their list of favorite restaurants.
         * Restaurants can manage their listings, including menus, reviews, and orders, through an admin dashboard.

Description or Definition of each entity

## User Stories

**3.4.1 User Registration**

I want to be able to register as a new user with my:

* First Name
* Last Name
* Email Address
* Phone Number
* Address
* Payment Information
* Password

So that I can start writing reviews of eateries and restaurants.

**3.4.2 Login and account**

As a registered user, I want to be able to access my account and old reviews by logging in with my email and password.

I want to be able to modify my information, read my reviews, and track my activity on the site from my profile page.

**3.4.3 Restaurant Owner Registration**

I want to be able to register my restaurant and offer information about it, such as its name, address, and menu.

**3.4.4 Restaurant Details**

In order to assess a restaurant's quality, I as a user want to see the average rating based on other customer ratings as well as their menu and products.

**3.4.5 Leave a Review**

As a logged-in user, I want to write a review for a restaurant, including rating it and adding comments, to share my dining experience

**3.4.6 Place order**

As a logged-in user, I want to place an order from a restaurant.

**3.4.7 Add restaurant to favorite list**

As a logged-in user, I want to be able to add restaurants to my favorite list and browse it.

## Non Functional Requirements

### Security requirements

**3.5.1.1 Data Encryption**

The application should use https to encrypt data transmitted between the user's browser and the server in order to prevent interception of sensitive data, such as login credentials and user reviews.

**3.5.1.2 Authentication**

Use secure authentication methods, such as salting and password hashing, to prevent unauthorized access to user account information.

**3.5.1.3 Authorization**

Make sure that users can only access the information and features that are available to them based on their roles and permissions by enforcing proper authorization checks.

**3.5.1.4 Database Backup**

To ensure data recovery in the event of system failures or data breaches, the application's database should be backed up daily or weekly.

**3.5.1.4 Users Privacy**

Provide users with access to their personal data and the option to delete their accounts in order to comply with data protection laws.

### Other quality requirements

**3.5.2.1 Performance**

Establish performance benchmarks, such as response times for various operations (like restaurant searching or user profile loading), and make sure the application satisfies these benchmarks even during peak usage.

**3.5.2.2 Scalability**

Make sure the application can scale horizontally (by adding more servers) or vertically (by upgrading server resources) to handle an increasing number of users, restaurants, and reviews.

**3.5.2.3 Usability**

To make sure the user interface is simple and easy to use, conduct usability tests and user testing, and based on the results, make the necessary adjustments.

**3.5.2.4 Accessibility**

Make sure the web application complies with accessibility standards like the WCAG (web content accessibility guidelines) and is usable by people with disabilities.

**3.5.2.5 Maintenance**

To make the codebase maintainable by future developers and to promote collaboration within your development team, enforce coding standards and documentation best practices.

**3.5.2.6 Availability**

Make sure the application is highly available, reliable, and provides redundancy in case of server failures to reduce downtime.

**3.5.2.7 Testing**

Establish testing specifications, such as unit tests, integration tests, and user acceptance tests, to guarantee the accuracy and dependability of the application.

**3.5.2.8 Scalability**

Define the scalability of the application as the user base expands. Provisions for load balancing, database sharing, and caching techniques may be included in this.

# Scope

Indicate which requirements you plan to include in the first two iterations. Note that this may change as you advance through the project.

## Iteration 1

List user stories

List non-functional requirements

## Iteration 2

List user stories

List non-functional requirements

## Backlog

List explicitely what is not included in the first three iterations

List user stories

List non-functional requirements