
System Requirements Specification Index

For

Online Medicine Shopping Application (MS SQL)

Version 4.0

IIHT Pvt. Ltd.

IIHT Ltd, No: 15, 2nd Floor, Sri Lakshmi Complex, Off MG Road, Near SBI LHO,
Bangalore, Karnataka – 560001, India
fullstack@iiht.com

ONLINE MEDICINE SHOPPING APPLICATION

System Requirements Specification

1. BUSINESS-REQUIREMENT:

1.1 PROBLEM STATEMENT:

The purpose of this application is to allow you to place a medicine order online, search medicine by name, add See medicine stock and list of medicine and more functions.

1.2 FOLLOWING IS THE REQUIREMENT SPECIFICATION:

	Online Medicine Shopping Application
Medicine Controller	
1	Find a medicine by id.
2	See all details of medicine by id
3	Place medicine order by id.
4	Get the order info that was placed.
5	Book a doctor appointment
6	And get the doctor appointment info.

2. TEMPLATE CODE STRUCTURE

2.1 Package: OnlineMedicineShopping

Resources

Names	Resource	Remarks	Status
Package Structure/Project			
controller	MedicineController	Medicine controller handle all action method for crud operations.	Partially Implemented

2.2 Package: OnlineMedicineShopping.BusinessLayer

Resources

Names	Resource	Remarks	Status
Package Structure			
Interfaces	Interface directory contain all interface for Services class	Inside this directory contains all business logic code and CURD Operation Logic method.	Already Implemented
Services, Repository	IMedicineServices, MedicineServices, IMedicineRepository, MedicineRepository.cs file for Method and business logic	Using this all cs file performed all CRUD operation.	Partially Implemented
ViewModels	Cs file for represent all view entities	All view entities setting class	Already Implemented

2.3 Package: OnlineMedicineShopping.DataLayer

Resources

Names	Resource	Remarks	Status
Package Structure			
MedicineDbContext	Contain all business logic for data set and Dbset setting	Using this cs file performed all Data related settings operations.	Already Implemented

2.4 Package: OnlineMedicineShopping.Entities

Resources

Names	Resource	Remarks	Status
Package Structure			
Entities Class	ApplicationUser, Appointment, Category, Doctor, Medicine, MedicineOrder, Class List cs file	Contain all entities property for application	Already Implemented

2.5 Package: OnlineMedicineShopping.Test

Resources

Note : - OnlineMedicineShopping.Test contains all test class and test case functions for code evaluation.

3. REST ENDPOINTS

Rest End-points to be exposed in the controller along with method details for the same to be created

3.1 MedicineController

URL Exposed		Purpose
/get-medicine-details		Find a medicine details by clicking details
Http Method	GET	
Parameter 1	int(MedicineId)	
Return	<Medicine>	
/place-order		Place an order of medicine
Http Method	GET	
Parameter 1	ApplicationUser user	
Parameter 2	Int(MedicineId)	
Return	<ApplicationUser>	
/get-order-info		Get order info of user
Http Method	GET	
Parameter 1	Int (userId)	
Return	<MedicineOrder>	

/book-appointment		Book an appointment with doctor
Http Method	POST	
Parameter 1	Appointment appointment	
Return	<Appointment>	
/appointment-details-by-id		Get details of doctor appointment
Http Method	GET	
Parameter 1	Int(appointmentId)	
Return	<Appointment>	

4. ASSUMPTIONS, DEPENDENCIES, RISKS / CONSTRAINTS

4.1 Common Constraints:

- Following validation constraints are to be added
- All the value for Appoint details: must not be null and min of 4 chars
- All the Medicine value: must not be null and min of 3 chars
- All the User: must not be null and min of 3 chars.
- All the Categories: must not be null.
- All the Doctor: must not be null.
- For all receiving Url parameter, validation check must be done and must throw custom exception if data is invalid
- Must not go and touch the test resources, as they will be used for Auto-Evaluation.

5. EXECUTION STEPS TO FOLLOW

1. All actions like build, compile, running application, running test cases will be through Command Terminal.
2. To open the command terminal the test takers need to go to the Application menu (Three horizontal lines at left top) Terminal → New Terminal.
3. On command prompt, cd into your project folder (**cd <Your-Project-folder>**).
4. (OnlineMedicineShopping/**sqlcmd -S localhost -U sa -P pass@word1**)
 - To create database from terminal -
 1. **Create Database OnlineMedicineShopping_Db**
 2. **Go**
5. Steps to Apply Migration(Code first approach):
 - Press **Ctrl+C** to get back to command prompt
 - Run following command to apply migration- (OnlineMedicineShopping/**dotnet-ef database update**)
6. To check whether migrations are applied from terminal:
(OnlineMedicineShopping/**sqlcmd -S localhost -U sa -P pass@word1**)
 - 1> **Use OnlineMedicineShopping_Db**
 - 2> **Go**
 - 1> **Select * From __EFMigrationsHistory**
 - 2> **Go**
7. To build your project use command:
(OnlineMedicineShopping /**dotnet build**)
8. To launch your application, Run the following command to run the application:
(OnlineMedicineShopping/**dotnet run**)
9. This editor Auto Save the code.
10. To test any Restful application, the last option on the left panel of IDE, you can find ThunderClient, which is the lightweight equivalent of POSTMAN.
11. To test any UI based application the second last option on the left panel of IDE, you can find Browser Preview, where you can launch the application.

12. To run the test cases in CMD, Run the following command to test the application:
(You can run this command multiple times to identify the test case status,
and refactor code to make maximum test cases passed before final submission)
(OnlineMedicineShopping.Test/**dotnet test --logger "console;verbosity=detailed"**).
13. If you want to exit(login) and continue the coding later anytime (using Save & Exit
option on Assessment Landing Page) then you need to use CTRL+Shift+B -
command compulsorily on code IDE. This will push or save the updated contents in
the internal git/repository. Else the code will not be available in the next login.
14. These are time bound assessments the timer would stop if you logout and while
logging in back using the same credentials the timer would resume from the same
time it was stopped from the previous login.
15. You need to use CTRL+Shift+B - command compulsorily on code IDE, before final
submission as well. This will push or save the updated contents in the internal
git/repository, and will be used to evaluate the code quality.
-