
System Requirements Specification Index

For

Online Medicine Shopping Application (InMemory)

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	View All medicine with medicine id and name.
2	Find a medicine by name and id.
3	See all details of medicine by id
4	Place medicine order by id.
5	Get the order info that was placed.
6	Book a doctor appointment
7	And get the doctor appointment info.

2. TEMPLATE CODE STRUCTURE

2.1 Package: OnlineMedicineShopping

Resources

Names	Resource	Remarks	Status
Package Structure/Project			
wwwroot	CSS, JS, Lib	Inside all these directory contains all styling logic code and javascripts file.	Already Implemented
controller	MedicineController	School controller handle all action method for respective user interface view(cshtml file)	Partially Implemented
Models	Error view model cs file	Contain all basic error definition	Already Implemented
Views	Medicine, Home, Shared	All View .cshtml file for user interface.	Already 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 CRUD 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			
DataGenerator	This cs class contain all dummy data for inMemory Operation	Create and initialize a basic data for application alive for demo.	Already Implemented
MedicineDbContext	Contain all business logic for data set and Dbset setting	Using this cs file performed all Data related settings operations.	

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.Entities

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

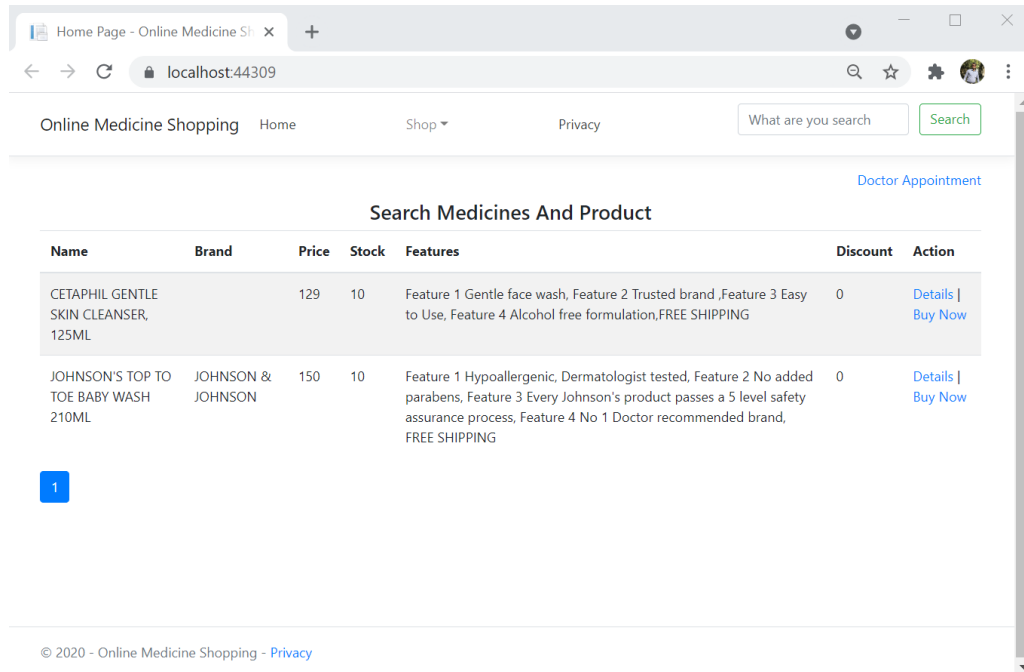
3.1 MedicineController

URL Exposed		Purpose								
/localhost:4431/ <table><tr><td>Http Method</td><td>GET</td></tr><tr><td>Parameters</td><td>Int(Id)</td></tr><tr><td>Return</td><td><IEnumerable<Medicine>></td></tr></table>		Http Method	GET	Parameters	Int(Id)	Return	<IEnumerable<Medicine>>	Show or Fetch all medicine and find		
Http Method	GET									
Parameters	Int(Id)									
Return	<IEnumerable<Medicine>>									
/Medicine/Details <table><tr><td>Http Method</td><td>GET</td></tr><tr><td>Parameter 1</td><td>int(MedicineId)</td></tr><tr><td>Return</td><td><Medicine></td></tr></table>		Http Method	GET	Parameter 1	int(MedicineId)	Return	<Medicine>	Find a medicine details by clicking details		
Http Method	GET									
Parameter 1	int(MedicineId)									
Return	<Medicine>									
1/Medicine/Placeorder <table><tr><td>Http Method</td><td>GET</td></tr><tr><td>Parameter 1</td><td>ApplicationUser user</td></tr><tr><td>Parameter 2</td><td>Int(MedicineId)</td></tr><tr><td>Return</td><td><ApplicationUser></td></tr></table>		Http Method	GET	Parameter 1	ApplicationUser user	Parameter 2	Int(MedicineId)	Return	<ApplicationUser>	Place an order of medicine
Http Method	GET									
Parameter 1	ApplicationUser user									
Parameter 2	Int(MedicineId)									
Return	<ApplicationUser>									
/Medicine/OrderInfo/orderId <table><tr><td>Http Method</td><td>GET</td></tr><tr><td>Parameter 1</td><td>Int (orderId)</td></tr><tr><td>Return</td><td><MedicineOrder></td></tr></table>		Http Method	GET	Parameter 1	Int (orderId)	Return	<MedicineOrder>	Get order info of medicine details		
Http Method	GET									
Parameter 1	Int (orderId)									
Return	<MedicineOrder>									
/Medicine/DoctorAppointment <table><tr><td>Http Method</td><td>POST</td></tr><tr><td>Parameter 1</td><td>Appointment appointment</td></tr><tr><td>Return</td><td><Appointment></td></tr></table>		Http Method	POST	Parameter 1	Appointment appointment	Return	<Appointment>	Book an appointment with doctor		
Http Method	POST									
Parameter 1	Appointment appointment									
Return	<Appointment>									
/Medicine/AppointmentInfo/Id <table><tr><td>Http Method</td><td>GET</td></tr><tr><td>Parameter 1</td><td>Int(Id)</td></tr><tr><td>Return</td><td><Appointment></td></tr></table>		Http Method	GET	Parameter 1	Int(Id)	Return	<Appointment>	Get details of doctor appointment		
Http Method	GET									
Parameter 1	Int(Id)									
Return	<Appointment>									

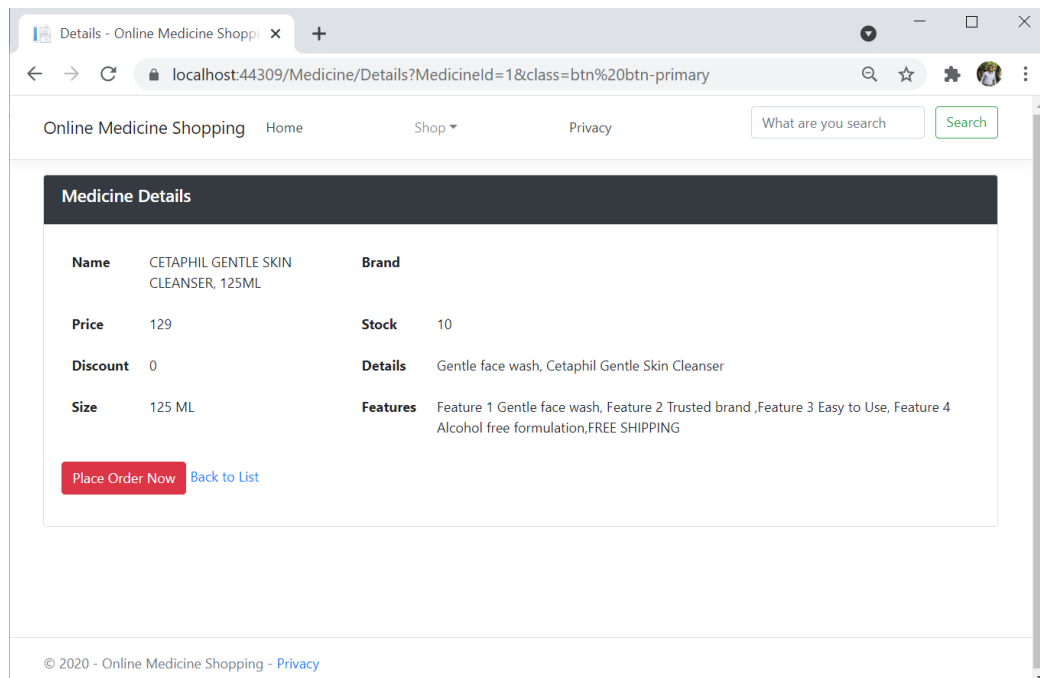
4. WIREFRAME

1. Following wireframe need to consider to develop the UI.

Home/Index Page



Medicine Details Page



Place Order Page

Placeorder - Online Medicine Shopping

localhost:44309/Medicine/Placeorder?MedicineId=2

Online Medicine Shopping Home Shop Privacy What are you search Search

Place Order now

Name	Email Address
<input type="text"/>	<input type="text"/>
Mobile	Pin Code
<input type="text"/>	<input type="text"/>
HouseNo, Building Name	Road Name, Area, Colony
<input type="text"/>	<input type="text"/>
City	State
<input type="text"/>	<input type="text"/>

[Place Order](#) [Back To Product](#)

© 2020 - Online Medicine Shopping - [Privacy](#)

Purchase or OrderInfo Page

OrderInfo - Online Medicine Shopping

localhost:44309/Medicine/OrderInfo?userId=2

Online Medicine Shopping Home Shop Privacy What are you search Search

Purchase Order Info

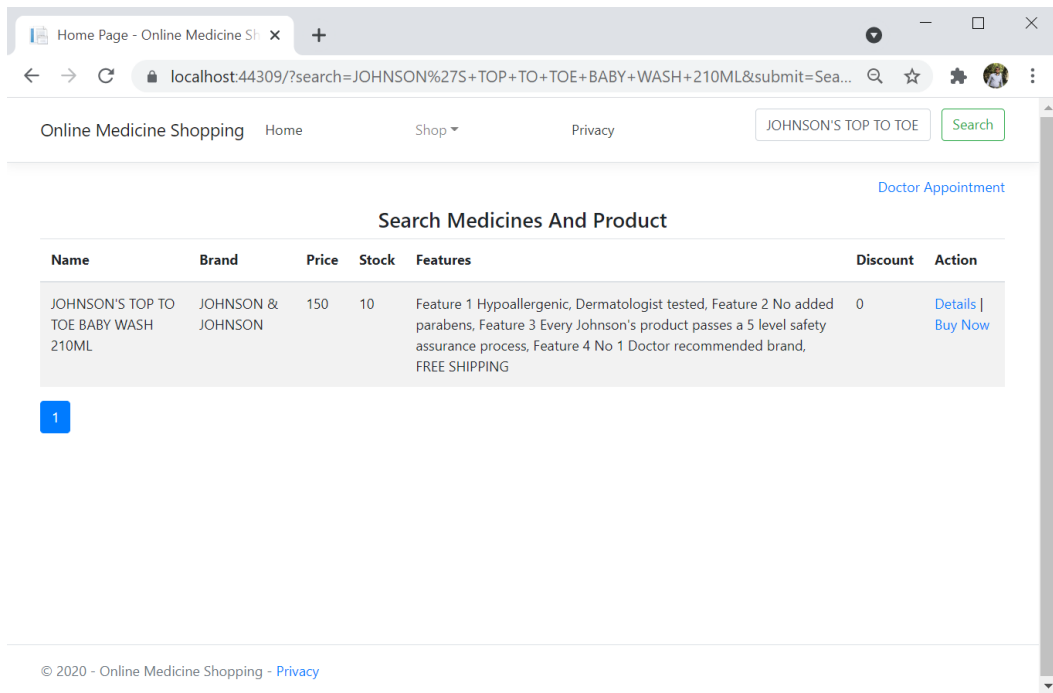
Thanku for shopping with us your order will be processed....

[Shop Again](#)

OrderId	MedicineId	UserId	Action
2	2	2	Product Details

© 2020 - Online Medicine Shopping - [Privacy](#)

Search Medicine result page

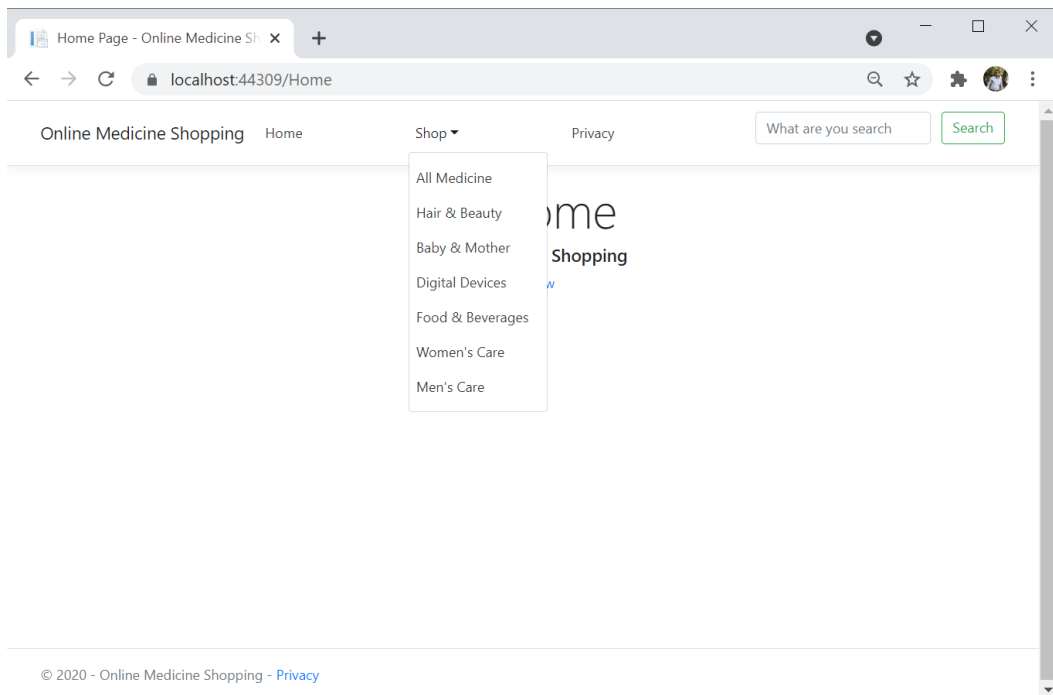


The screenshot shows a web browser window with the URL `localhost:44309/?search=JOHNSON%27S+TOP+TO+TOE+BABY+WASH+210ML&submit=Sea...`. The page title is "Online Medicine Shopping". The navigation bar includes "Home", "Shop", and "Privacy" links, along with a search bar containing "JOHNSON'S TOP TO TOE" and a "Search" button. A "Doctor Appointment" link is visible in the top right. The main content area is titled "Search Medicines And Product" and displays a table with the following data:

Name	Brand	Price	Stock	Features	Discount	Action
JOHNSON'S TOP TO TOE BABY WASH 210ML	JOHNSON & JOHNSON	150	10	Feature 1 Hypoallergenic, Dermatologist tested, Feature 2 No added parabens, Feature 3 Every Johnson's product passes a 5 level safety assurance process, Feature 4 No 1 Doctor recommended brand, FREE SHIPPING	0	Details Buy Now

A pagination control shows "1" in a blue box. The footer contains the text "© 2020 - Online Medicine Shopping - [Privacy](#)".

MenuBar Options



The screenshot shows the same web browser window with the URL `localhost:44309/Home`. The navigation bar includes "Home", "Shop", and "Privacy" links, along with a search bar containing "What are you search" and a "Search" button. A dropdown menu is open under the "Shop" link, displaying the following options:

- All Medicine
- Hair & Beauty
- Baby & Mother
- Digital Devices
- Food & Beverages
- Women's Care
- Men's Care

The footer contains the text "© 2020 - Online Medicine Shopping - [Privacy](#)".

Book Doctor Appointment Page

Appointment - Online Medicine Shopping

localhost:44309/Medicine/DoctorAppointment?class=btn%20btn-danger

Online Medicine Shopping Home Shop Privacy What are you search Search

Book Appointment now

Patient Name	Doctor Name
<input type="text" value="Uma"/>	<input type="text" value="Rajnish Ranjan"/>
Appointment Date	Symtoms
<input type="text" value="24-09-2021"/>	<input type="text" value="Ill"/>
Patient Age	Remark
<input type="text" value="23"/>	<input type="text" value="Body Pain"/>

[Book Appointment](#) [Back To Home](#)

© 2020 - Online Medicine Shopping - [Privacy](#)

Appointment Details Page

Details - Online Medicine Shopping

localhost:44309/Medicine/AppointmentInfo?AppointmentId=1

Online Medicine Shopping Home Shop Privacy What are you search Search

Appintment Details

Keep your details while meeting with doctor

Patient Name	Uma	Doctor Name	Rajnish Ranjan
Appointment Date	24-09-2021 00:00:00	Patient Age	23
Symtoms	Ill	Remark	Body Pain
Appointment Number	1		

[Back to Home](#)

Thanks for using Online Medical Shopping Appointment

© 2020 - Online Medicine Shopping - [Privacy](#)

5. ASSUMPTIONS, DEPENDENCIES, RISKS / CONSTRAINTS

5.1 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.
- **Database used is Embedded InMemory with pre-written entities data and value.**

6. 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. To build your project use command:
(OnlineMedicineShopping / **dotnet build**)
5. To launch your application, Run the following command to run the application:
(OnlineMedicineShopping/ **dotnet run**)
6. This editor Auto Saves the code.
7. 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.

8. 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.
 9. To run the test cases in CMD, Run the following command to test the application:
(OnlineMedicineShopping / **dotnet test --logger "console;verbosity=detailed"**)
(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)
 10. If you want to exit(logout) 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.
 11. 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 logout.
 12. 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.
-