**Writeup for Capstone:**

**Name: Nileena James.**

Github link: <https://github.com/niljames/Capstone-Project.git>

Question:

DESCRIPTION

Create a dynamic and responsive .Net+Angular e-healthcare web application to allow users to purchase medicines for different categories.

**Background of the problem statement**

ABC Healthcare is a Mumbai-based pharma company which sells medicines throughout India as per the demand it receives from multiple hospitals and clinics. It has been operating in India for the last 15 years.

However, in recent years since all industries are coming online to boost their sales and cover larger market space, ABC Healthcare missed the chance and as a result their sales have been dropping for the last 2 years. To overcome this and bring sales back on track they have decided to bring their service online to allow their customers to purchase medicines through their web application.

You are hired as one of the Full Stack .Net developers and have been asked to develop the web application. The management team has provided you the requirements and their business model so that you can easily arrange different components of the application.

**Explanation:**

**I am unable to deploy it in azure as I don’t have enough credits/balance in my azure account. I have practiced for the practice session as well as for phase-4 projects. Other than that, everything else is implemented.**

**![A picture containing text, indoor, screenshot

Description automatically generated]()Fig 1: Homepage before login.**

**![A screenshot of a computer

Description automatically generated with low confidence]() Fig 2.1: Signup modal.**

**![Graphical user interface, application

Description automatically generated]() Fig 2.2: Login as user.**

**![A close-up of a game board

Description automatically generated with low confidence]() Fig 3: Homepage after login.**

![Graphical user interface

Description automatically generated]() **Fig 4: Different categories of the medicine displayed.**

**![Graphical user interface, application, website

Description automatically generated]() Fig 5: Medicines under liquid medicine category.**

**![Graphical user interface, website

Description automatically generated]() Fig 6: Medicine details on click.**

**Graphical user interface, application

Description automatically generated Fig 7: Adding a medicine to cart.**

**Graphical user interface, application

Description automatically generatedFig 8: Changing the quantity of the medicine in cart.**

**![Graphical user interface, application

Description automatically generated]()Fig 9: Payment gateway.**

**![Graphical user interface, application

Description automatically generated]() Fig 10: On successful payment.**

**Graphical user interface, application, Teams

Description automatically generatedFig 11: Token generation on successful payment.**

**![A picture containing text, indoor, screenshot

Description automatically generated]()Fig 12: Login as admin.**

**Graphical user interface

Description automatically generated with medium confidenceFig 12: Admin main page.**

**![Table

Description automatically generated]()Fig 13: Admin user dashboard.**

**![Graphical user interface, table

Description automatically generated]() Fig 14: Admin medicine category dashboard.**

**![Graphical user interface, website

Description automatically generated]() Fig 14.1: Admin medicine dashboard.**

**![Graphical user interface, application, PowerPoint

Description automatically generated]() Fig 14.2: Admin medicine dashboard.**

**Graphical user interface, text, application

Description automatically generated Fig 15.1: api for users.**

**Graphical user interface, text, application

Description automatically generatedFig 15.2: api for Medicine Category.**

**![Text

Description automatically generated]() Fig 15.3: api for Medicine.**

**![Graphical user interface, application, table

Description automatically generated]()**

**Fig 16.1: User table in the database.**

**![Table

Description automatically generated]()**

**Fig 16.1: Medicine Category table in the database.**

**![Graphical user interface, text

Description automatically generated]() Fig 16.1: Medicine table in the database.**

* **I have used angular to make the front-end, SSMS for the database and ASP.NET for the api calls.**
* **Have created the respective web api controllers for users, medicine category as well as medicine.**
* **Can login as admin with credentials** [**admin@gmail.com**](mailto:admin@gmail.com) **and “admin” as the email and password respectively to access the admin dashboards.**
* **Can perform all the crud operations on all the dashboards on all the tables.**
* **As user, the user can view the different categories. Choose the required category and add the required medicine into the cart and pay for them smoothly.**
* **It’s also responsive in mobile devices.**
* **As for the automation testing, I have created two test files. One for the admin test cases and one for the user test cases.**