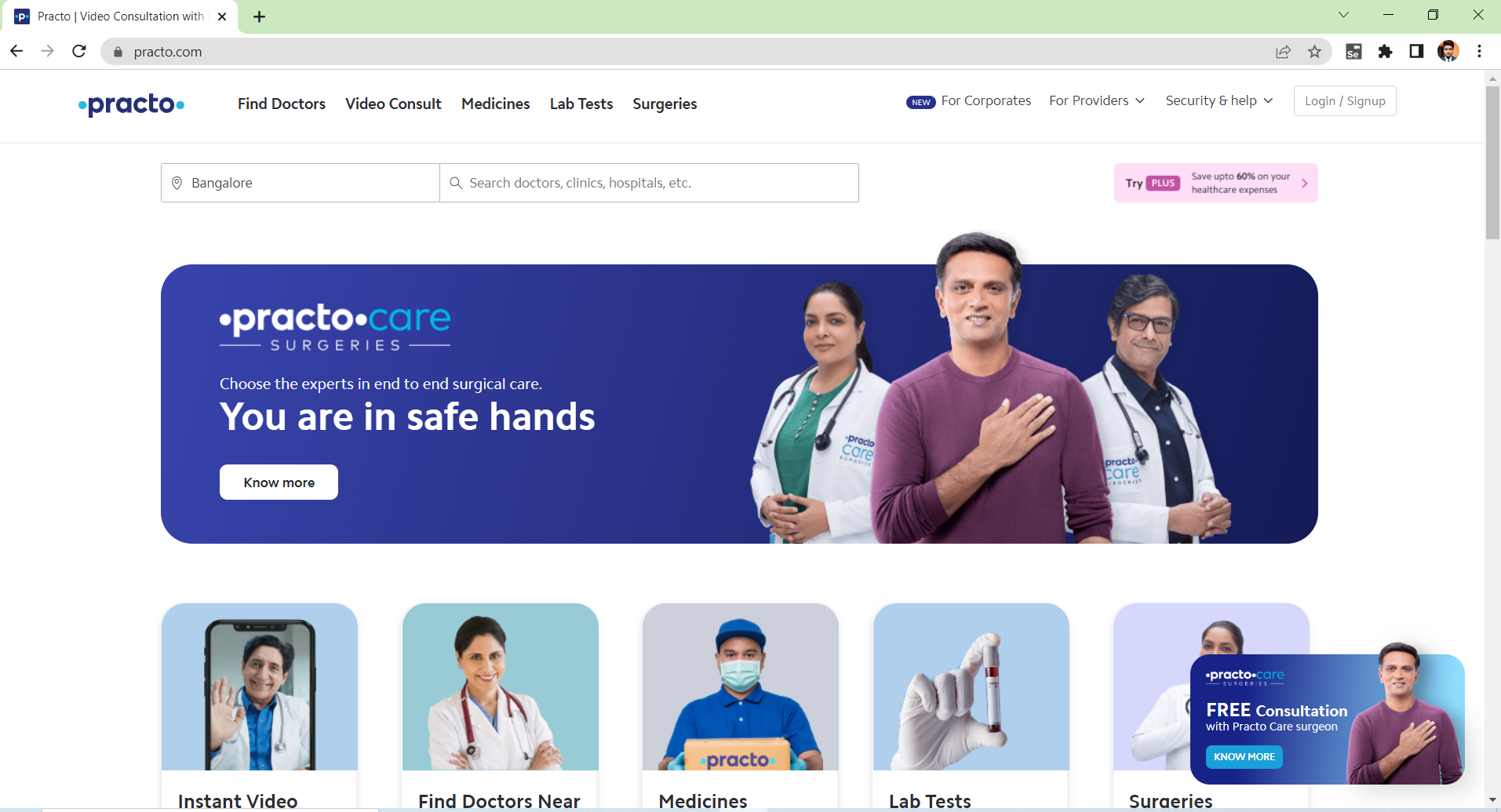
**Finding Hospitals**

**Group Members:**

* 2143044-Md Rajibul Islam
* 2143108-Priya Kumari
* 2142447-Abhishek Dharme
* 2142079-Shivang Singh
* 2142381-Jhilam Mondal

**Site on which work is done:-**

****

**Contents: -**

1.Introduction

          1.1: Scope

2.Source File

           2.1: Filename

           2.2: Source File Structure

           2.3: Indentation

           2.4: Name Convention

3.Exception Handling

4.Tools and Plugins

**Introduction**

For making the site reliable, scalable, accessible and sustainable    Quality Assurance (QA) team must adopt and maintain the coding standards which allows code readability and makes it easier to understand.

**Problem Statement: Finding Hospitals**

Get the hospital names   
1. Hospital near which is open 24\*7

2. Which has parking facility

3. rating should be more than 3.5

Site: “practo.com”

**Detailed Description:**

1. For Bangalore city, identify Hospitals that is Open 24/7, has Parking facility with rating more than 3.5 and display the hospital names.
2. In Diagnostics page, pick all the top cities name & store in a List and display the same.
3. Go to Corporate Wellness page, fill valid details, check if

“Schedule a demo” button is enabled.

**1.1: Scope:**

The practices are targeted towards code readability and maintenance of coding standards while maintaining standard Indentation and naming conventions which demonstrate the knowledge of professionalism.

* Writing Java code that is easy to maintain and easy to demonstrate.
* Increase code usability.

**Key Automation Scope**

* Handling multiple browser windows, drop down, search option.
* Navigating back to home page.
* Extract multiple options items & store in collections.
* Filling form (in different objects in web page).
* Capture submission message.

**Source File**

The header should be followed by the package and import statements and then the documentation comments exactly in the following sequence and indented to the same level.

**2.1: Filename**

The source file name consists of the case-sensitive name of   the top-level class it contains, plus the .java extension.

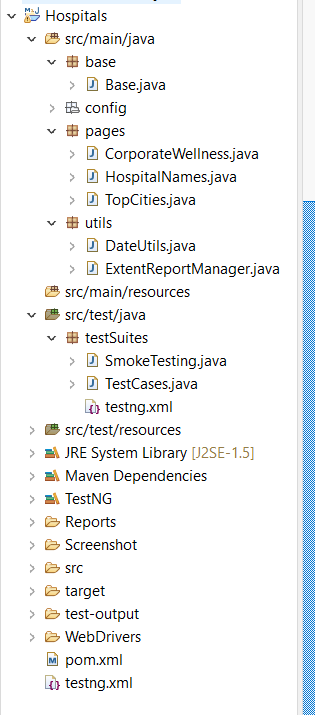
**2.2: Source File Structure**

            A source file consists of, in order:

            ●Package Name

            ●Imports

            ●Class definition



**2.3: Indentation**

* There must be a space after giving a comma between two function arguments.
* Proper indentation should be there at the beginning and end of each block in the program.
* All braces should start from a new line and the code following the end of braces should also start from a new line.

**2.4: Naming Conventions**

* Local variables and Global variables should be named using

camel case lettering starting with a small letter(e.g., localData).

* Constant Data should be named using upper case letters only. (e.g., CONSTDATA).
* Avoid use of digits for naming variables.
* The names of functions should be written in camel case starting
* Use descriptive names for variables and methods.

**3: Exception Handling**

You must follow the below Java coding guidelines for implementing effective exception handling.

1-Always write a catch block for handling exceptions.

2-Make sure to add a logging message or the stack trace in the catch block.

3-Avoid catching the general exception and have a specific exception.

4-The clean-up code should be added in the <finally> block.

5-This provides a single location for the clean-up, and it is guaranteed to run.

**4: Tools and Plugins**

1. **Selenium with Java**

* Selenium is the first thing that comes to mind when one is planning to automate the testing of web application. [Selenium](https://www.browserstack.com/selenium) is a beneficial tool because it is not only open source but also a portable software testing framework for web applications that support multiple languages like Java, C#, Ruby, Python. Choosing the right language depends on the application under test, the supporting community, available [test automation frameworks](https://www.browserstack.com/guide/best-test-automation-frameworks), usability, elegance, and of course, seamless build integration.
* Java is a popular programming language. As per StackOverflow, it is the third most popular back-end technology after JavaScript and SQL.
* A good community of developers to create documentation and resolve issues has helped Java to become the most preferred language among the application developers. Thus, writing selenium test cases using Java has multiple benefits:
* Selenium supports Java. So, testers can leverage the active community of contributors and detailed documentation to write test cases
* Programs written in Java are faster than other popular languages like Python
* Java is more widely used in commercial applications as compared to other programming languages like Python and hence integrating Selenium tests it easier.

1. **Maven**

* Maven's primary goal is to allow a developer to comprehend the complete state of a development effort in the shortest period. To attain this goal, Maven deals with several areas of concern:
* Making the build process easy
* Providing a uniform build system
* Providing quality project information
* Encouraging better development practices

1. **TestNG**

TestNG is an open-source test automation framework for Java. It is developed on the same lines of JUnit and NUnit. Few advanced and useful features provided by TestNG makes it a more robust framework compared to its peers. The NG in TestNG stands for 'Next Generation'.

1. **Apache POI**

Apache POI is a popular API that allows programmers to create, modify, and display MS Office files using Java programs. It is an open-source library developed and distributed by Apache Software Foundation to design or modify Microsoft Office files using Java program.

1. **POM**

A Project Object Model or POM is the fundamental unit of work in Maven. It is an XML file that contains information about the project and configuration details used by Maven to build the project. It contains default values for most projects. Examples for this is the build directory, which is target; the source directory, which is src/main/java; the test source directory, which is src/test/java; and so on. When executing a task or goal, Maven looks for the POM in the current directory. It reads the POM, gets the needed configuration information, then executes the goal.

Some of the configuration that can be specified in the POM are the project dependencies, the plugins or goals that can be executed, the build profiles, and so on. Other information such as the project version, description, developers, mailing lists and such can also be specified.

1. **Extent Reports**

Extent Reports is an open-source reporting library useful for test automation. It can be easily integrated with major testing frameworks like JUnit, NUnit, TestNG, etc. These reports are HTML documents that depict results as pie charts. They also allow the generation of custom logs, snapshots, and other customized details.

Once an automated test script runs successfully, testers need to generate a test execution report. While TestNG does provide a default report, they do not provide the details