MID SEM II

ADVANCE JAVA

SET E

Submitted by: VATSAL PAREKH

1810DMBCSE03326

1) **Provide the database schema (tables and relations) for this project.**

This project we just had to collect the data of the patients which are tested for covid - 19 and entry the data whether they are tested positive or negative so this can be done using one single table TestData and if the project is further extended we can increase the number of tables and create a bigger and better schema like for those patients who have been tested are alive or not so a table for those patients who died and another table for those patients who recovered from the virus and headed back to home also a login table can be used to allow access to only registered users basically making it a hospital management system but in this project only the task of updating the test results on the patients registered tests is there so only a single table is used here.

TESTDATA	
UID	VAR , PK
NAME	VAR ,NOTNULL
DOB	DATE
MOBILENO	LONG
GENDER	VAR
TESTRESULT	VAR
SAMPLE_RECEIVING_TIME	DATE

COVID_DATA

2) Describe all the servlet classes and their functionalities that you need to build this project.

This project is simply about collecting and storing data of the peoples who have had a test of this corona virus and after the test results are out then updating the result of a the respective patient with respective result each patient is uniquely identified by a unique id. In this project there are basically two tasks performed by the application first is to create or insert data of test Subject secondly to update the test results of the test subject whether positive or negative.

To create this web application servlet and jsp is used and in servlet HttpServletRequest and HttpServletResponse are used to fetch data from the jsp form and PrintWriter is used to print the data onto the pages through response objects

In this project to perform these two tasks there are two servlet classes used one is insertRecord.java and other is updatetest.java.

insertRecord.java
in this servlet the information of the test subject is fetched from the form which is in our js page by the httpServlet request object and then each parameter is stored in a string variable then the first step to database connectivity is performed using forName() method of class Class and providing the driver for mysql , jdbc, after that a connection object is created and connection is established using the URL database name and user name and password of the mysql, the a statement is created using the PreparedStatement and query is passed into the arguments the values are passed into query which were fetched by the request object, the unique id is generated by combining the mobile number and date of birth of the test subject and shown as a message to user after the query is successfully executed or if there is a exception then to avoid this the whole code is placed inside a try-catch block. After query is executed and result being printed using response object and PrintWriter class the established connection now is terminated.

```
package com.covid19record;

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
```

```
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class insertRecord extends HttpServlet {
    public void doPost(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        String name = request.getParameter("userName");
        String mob = request.getParameter("mobno");
        String dob = request.getParameter("dob");
        String gender = request.getParameter("gender");


        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con = DriverManager.getConnection(
                    "jdbc:mysql://localhost:3306/coviddata", "root", "1009");

            String id=dob+mob;
            String def="Negative";
            PreparedStatement ps = con
                    .prepareStatement("insert into TESTDATA
values(?,?,?,?,?,?,now())");
            ps.setString(1, id);
            ps.setString(2, name);
            ps.setString(3, mob);
            ps.setString(4, dob);
            ps.setString(5, gender);
            ps.setString(6,def);

            int i = ps.executeUpdate();
            if (i > 0)
            {
                  out.print("<html><body><center>");
                out.print("Your Record Has been Saved Successfully");
                  out.print("<br/><br/>");
                  out.print("Your Unique ID is :"+id);
                  out.print("</center></body></html>");
            }

        } catch (Exception e2) {
            System.out.println(e2);
        }

        out.close();
    }

}
```

Updatetest.java

In this servlet the pre-inserted information of the test subject is fetched by using the unique id and if the record is found then the update query is applied on the table and the test result of the patient is updated whether positive or negative according to the form filled by user after the query successfully applied a message of update successful is displayed and if record is not found the a message of record not found is shown. All this is done within a try catch block where connection is created query is executed and connection is terminated according to the steps to perform database management system as same as the first class.

```
package com.covid19record;

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class updatetest extends HttpServlet {
    public void doPost(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        String uid = request.getParameter("uid");
        String res = request.getParameter("result");



        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con = DriverManager.getConnection(
                    "jdbc:mysql://localhost:3306/coviddata", "root", "1009");


            PreparedStatement ps = con
                    .prepareStatement("select from TESTDATA where uid=?");
            ps.setString(1, uid);


            int i = ps.executeUpdate();
```

```
            if (i > 0)
            {
            PreparedStatement ps2=con.prepareStatement("update TESTDATA set
TestResult=? where uid= ?");
            ps2.setString(1,res);
            ps2.setString(2, uid);
            int j=ps2.executeUpdate();
            if(j>0)
                  out.print("UPDATED SUCCESSFULLY");
            }
            else {
                  out.print("Record not Found !!!");
            }

        } catch (Exception e2) {
            System.out.println(e2);
        }

        out.close();
    }

}
```

3) **Describe all the non-servlet java classes and their functionalities that you will use in servlet classes.**

   In this project the non servlet classes which are used are used for database connectivity and for executing queries we'll see them below.

Class :-

Java provides a class with name Class **in java.lang package. Instances of the class Class represent classes and interfaces in a running Java application. The primitive Java types (boolean, byte, char, short, int, long, float, and double), and the keyword *void* are also represented as Class objects. It has no public constructor. Class objects are constructed automatically by the Java Virtual Machine(JVM). It is a final class, so we cannot extend it.The Class class methods are widely used in Reflection API.**

Connnection Interface:-

A Connection is the session between java application and database. The Connection interface is a factory of Statement, PreparedStatement, and DatabaseMetaData i.e. object of Connection can be used to get the object of Statement and DatabaseMetaData. The Connection interface provide many methods for transaction management like commit(), rollback() etc.

PreparedStatement:-

PreparedStatement is a class in java.sql package and allows Java programmer to execute SQL queries by using JDBC package. You can get PreparedStatement object by calling connection.prepareStatement() method.SQL queries passed to this method goes to Database for pre-compilation if JDBC driver supports it. If it doesn't than pre-compilation occurs when you execute prepared queries. Prepared Statement queries are pre-compiled on database and there access plan will be reused to execute further queries which allows them to execute much quicker than normal queries generated by Statement object.

Exception :-

The class Exception and its subclasses are a form of Throwable that indicates conditions that a reasonable application might want to catch.

The class Exception and any subclasses that are not also subclasses of Runtime Exception are *checked exceptions*. Checked exceptions need to be declared in a method or constructor's throws clause if they can be thrown by the execution of the method or constructor and propagate outside the method or constructor boundary.

4) Describe all the major challenges that you may face to build this project

#USER INTERFACE AND USER EXPERIENCE

Think a decade ago, the web was a completely different place. Smartphones don't exist.Simpler and customer oriented web application are highly expected now. Sometimes it's the small UI elements that make the biggest impact. In the era of Smartphones, websites should be responsive enough on the smaller screens. If your web applications frustrate or confuse users, then it is difficult to maintain your customer's loyalty for your website. Website navigation is another part often neglected by developers. Intuitive navigation creates a better user experience for the website visitor. Intuitive navigation is leading your audience to the information they are looking without a learning curve. And when the navigation is intuitive, visitors can find out information without any pain, creating a flawless experience preventing them from visiting the competitors.

#SCALABILITY

Scalability is neither performance nor it's about making good use of computing power and bandwidth. It's about load balancing between the servers, hence, when the load increases (i.e. more traffic on the page) additional servers can be added to balance it. You should not just throw all the load on a single server but you should design the software such that it can work on a cluster of servers. Service-oriented architecture (SOA) can help in improving scalability when more and more servers are added. SOA gives you the flexibility to change easily. Service oriented architecture is a design where application components provide services to other components through the communication protocol, basically over a network.

#PERFORMANCE

Generally, it is accepted that website speed has the major importance for a successful website. When your business is online every second counts. Slow web applications are a failure. As a result, customers abscond your website thus, damaging your revenue as well as reputation. It is said that think about performance first before developing the web application. Some of the performance issues are Poorly written code, Un-Optimized Databases, Unmanaged Growth of data, Traffic spikes, Poor load distribution, Default configuration, Troublesome third party services, etc. A content distribution network (CDN) is globally distributed network of proxy servers deployed in multiple data centres. It means instead of using a single web server for the website, use a network of servers. Some of the benefits of CDN are that the requests on the server will be routed to different servers balancing the traffic, the files are divided on different CDNs so there will be no queuing and wait for downloading different files like images, videos, text, etc.

#KNOWLEDGE OF FRAMEWORK AND PLATFORMS

Frameworks are the kick start for development languages: they boost performance, offer libraries of coding and extend capabilities, so developers need not do hand-coding web applications from the ground up. Frameworks offer features like models, APIs, snippets of code and other elements to develop dynamic web applications. Some of the frameworks have a rigid approach to development and some are flexible. Common examples of web frameworks are PHP, ASP.Net, Ruby on Rails and J2EE. Web platforms provide client libraries build on existing frameworks required to develop a web application or website. A new functionality can be added via external API. Developers and small business owners should

have a clear understanding of their company needs related to website and application development. Information delivery and online presence would require a simple web platform such as WordPress or Squarespace but a selling product requires an e-commerce platform such as Magento, Shopify. WooCommerce or BigCommerce). While choosing the perfect platform one should also consider technical skills, learning curve, pricing, customization options and analytics.

#SECURITY

In the midst of design and user experience, web app security is often neglected. But security should be considered throughout the software development life cycle, especially when the application is dealing with the vital information such as payment details, contact information, and confidential data. There are many things to consider when it comes to web application security such as denial of service attacks, the safety of user data, database malfunctioning, unauthorized access to restricted parts of the website, etc. Some of the security threats are Cross-Site Scripting, Phishing, Cross-Site Request Forgery, Shell Injection, Session Hijacking, SQL Injection, Buffer Overflow, etc. The website should be carefully coded to be safe against these security concerns.

Web development can be deliberately difficult as it involves achieving a final product which should be pleasing, builds the brand and is technically up to date with sound visuals.

#TIME TO DEVELOP THE APPLICATION

The project was given with a very small time to be prepared so time was the biggest factor in the problems.

5) **Build this application and submit the java source file and database .sql file**

https://github.com/VatsalParekhgit/Covid-19-Tests-Set-E.git