**GROUP 6**

**Problem statement**: Select, Design and Plan installation of database for test platform.

**1.Selection of Database:**

We have Selected **PostgreSQL** which is an advanced, enterprise class open source **relational database.**

Relational databases are used to organize data and identify relationships between key data points. They make it easy to sort and find information, which helps organizations make business decisions more efficiently and minimize costs. They work well with structured data. It supports many powerful features .For example:

**Server-based**

Because Postgres is usually on a server in the cloud, like on Amazon or Google, any number of users or apps can connect to it at once and perform operations.

**Data types**

Postgres has a richer set of column data types than SQLite. Some notable examples of column types that Postgres has but SQLite doesn't are:

* JSON - store JSON arrays with ability to query against them
* MONEY - makes it easier to work with time series data, like that of stocks
* date and timestamp - ability to index and sort by dates and times, also useful for time series data
* inet/cidr - store IP addresses, which is useful for some web apps

**Full text search**

Postgres can easily store vector representations of text you're storing and allow superfast queries on it. This is handy for things like autocompleting search fields in websites, as well as data science projects using natural language processing.

**2.Planning of installation of Database**

**Setting up a Postgres server locally**

1.Go to the downloads page on <https://www.postgresql.org/download/> and select Windows in the list.

2.On the next page, click the link that says "Download the installer"

3.On the next page, click "Download" in the Windows column of the latest version

4.Run the .exe to install Postgres - just use all the defaults for installation, but skip launching StackBuilder at the end.

Now, the Postgres application should be running. Now we can create a server and database.

### **Open pgAdmin**

With your Postgres installation came a user interface called **pgAdmin** that lets you view your servers in a front-end application. Search Windows for **pgAdmin** and launch it. It should open a new browser tab on your localhost.

When you first launch pgAdmin it'll open a new window, but it'll also add an icon to your toolbar. If you ever close your window and need to open a new one, just right-click the blue elephant icon and click "New pgAdmin Window".

### 

### **Make a Server**

Let's give our new server a name of "local", use "localhost" as the Host name, and create an easy password (make sure to "save password"). Click Save and you should see a new server pop into the tree on the left.

### **Make a Database**

Now that our server is created, in the tree expand the new "local" server, then right-click "Databases", hover over "Create", and select "Database". Under the "Database" field give it a name. We are using the name "test\_platform" for this database.

In the tree, expand "Databases" and you should see the name of your new database. When we create models and store data we'll be able to use the pgAdmin interface to view our tables and data like an Excel workbook and run SQL queries to explore and debug our project.

**DATABASE DESIGN**

