Ooad project implementation using java(oo language)

Online car service management system

Sabhari p

2018103582

Contents

1. **Introduction**
2. **Requirements of the software**
3. **Programming language and concept used**
4. **Software explained**
5. **Conclusion**

**INTRODUCTION:**

We are living our world online. All the basic and needed utilities are available online. One step to make the customer ease of their car maintenance is to bring the car service booking and managing online. The busy people who hardly finds time to take care of their vehicle can use this function.

It will be very useful for a businessmen who doesn’t have time to manage their car. They can simply book their service and carry on with their work. Thus software act as an interface between the customer and service station.

If someone had moved to other city, then they find difficult to find new service stations. Thus they can use this software to find solution for it.

With these as our basic introduction we will jump in to look out more functionalities that the software must include.

**REQUIREMENTS OF THE SOFTWARE**

We define some of the basic requirements of the software.

* Software should have a login page for each users.
* It should also be provided with powerful database to store the login details.
* The login page must use the unique id for customers. We have registration number which is unique and lets use it.
* The password should be checked for each id.
* In case if the Registration number is not available in the database, then we should prompt for signup.
* Incase if the password is wrong, then we should prompt user about the wrong password.
* It should also be provided with signup options for first time users.
* Signup must get their details and store in the database.
* Once when the registration number and password is correct, it should be redirected to main page.
* In main page we should have options to use the software efficiently.
* Software should allow us to book service.
* It should also allow us to book for Towing and pickup services.
* There should be a notification tab to trace the users activities and to provide a short info.
* There should be a Service record tab to view all the services and its description.
* There should be an info about all the service stations available within the region.
* It should also provide customer care services.
* In addition we have an helper portion which makes our job much easier.

**PROGRAMMING LANGUAGE AND CONCEPT USED**

The programming language used here is **JAVA** .

**Java** is a [class-based](https://en.wikipedia.org/wiki/Class-based_programming), [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) [programming language](https://en.wikipedia.org/wiki/Programming_language) that is designed to have as few implementation [dependencies](https://en.wikipedia.org/wiki/Dependency_(computer_science)) as possible. It is a [general-purpose](https://en.wikipedia.org/wiki/General-purpose_language) programming language intended to let [application developers](https://en.wikipedia.org/wiki/Application_developer) *write once, run anywhere* (WORA), meaning that [compiled](https://en.wikipedia.org/wiki/Compiler) Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to [bytecode](https://en.wikipedia.org/wiki/Java_bytecode) that can run on any [Java virtual machine](https://en.wikipedia.org/wiki/Java_virtual_machine) (JVM) regardless of the underlying [computer architecture](https://en.wikipedia.org/wiki/Computer_architecture). The [syntax](https://en.wikipedia.org/wiki/Syntax_(programming_languages)) of [Java](https://en.wikipedia.org/wiki/Java_(software_platform)) is similar to [C](https://en.wikipedia.org/wiki/C_(programming_language)) and [C++](https://en.wikipedia.org/wiki/C%2B%2B), but has fewer [low-level](https://en.wikipedia.org/wiki/Low-level_programming_language) facilities than either of them. The Java runtime provides dynamic capabilities (such as reflection and runtime code modification) that are typically not available in traditional compiled languages.

**OBJECT ORIENTED PROGRAMMING** concept is used here.

Object-oriented programming (OOP) is a computer programming model that organizes software design around data, or [objects](https://searchapparchitecture.techtarget.com/definition/object), rather than functions and logic. An object can be defined as a data field that has unique attributes and behavior.

With OO concepts this software is carefully designed with required object with correct relations. Each page in our software is an class. The source codes are attached for further references.

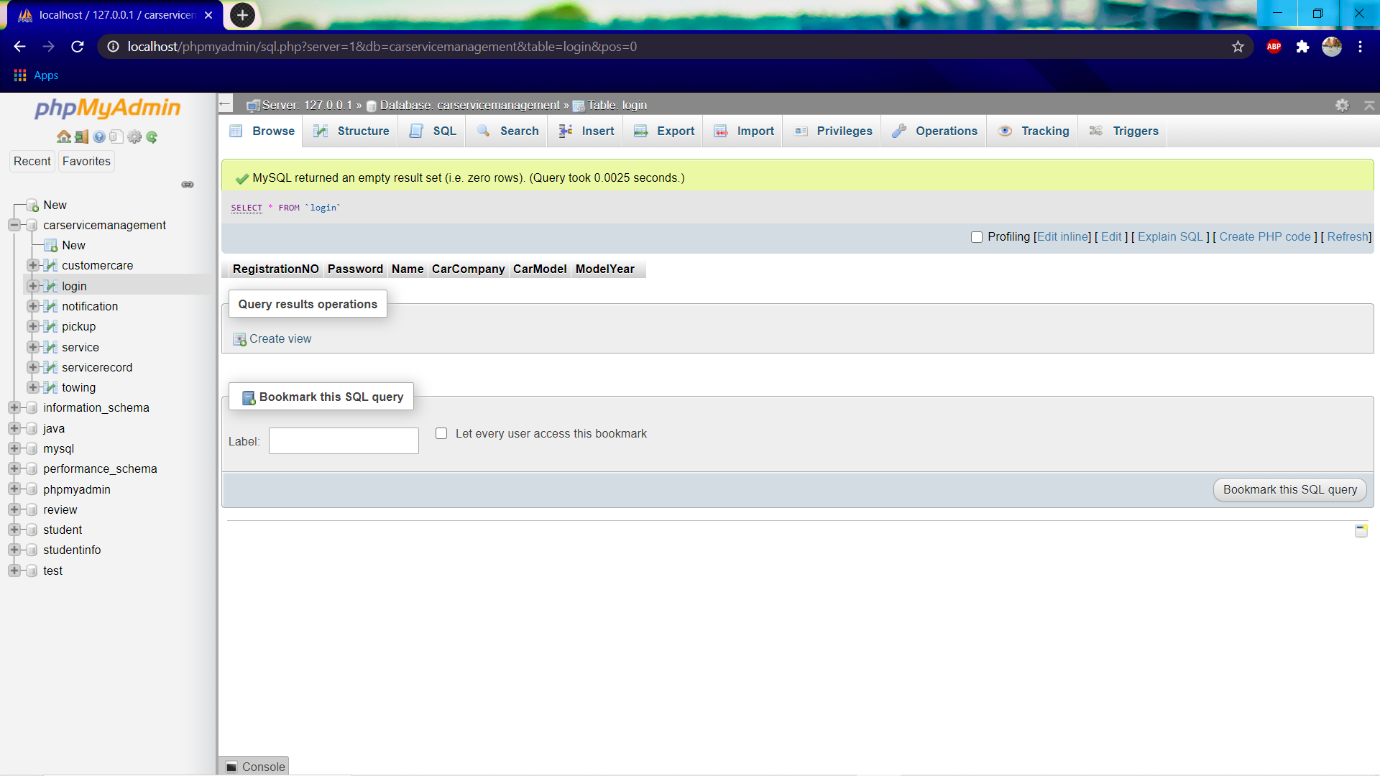
**SOFTWARE EXPLAINED**

Every software must have the login page. Our software also includes it. Lets view it



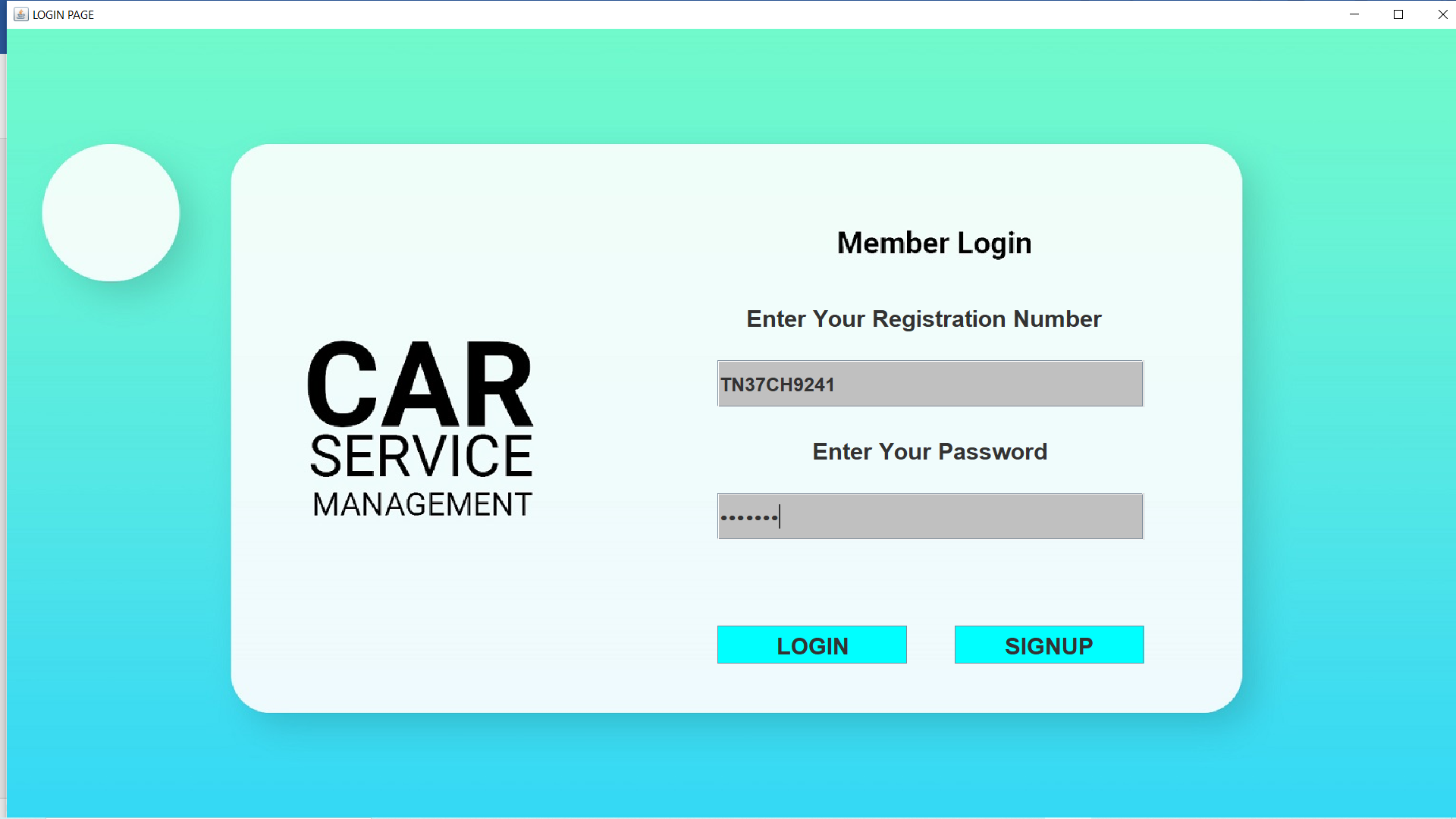
This is our login page with login and signup functions. Lets work with it.

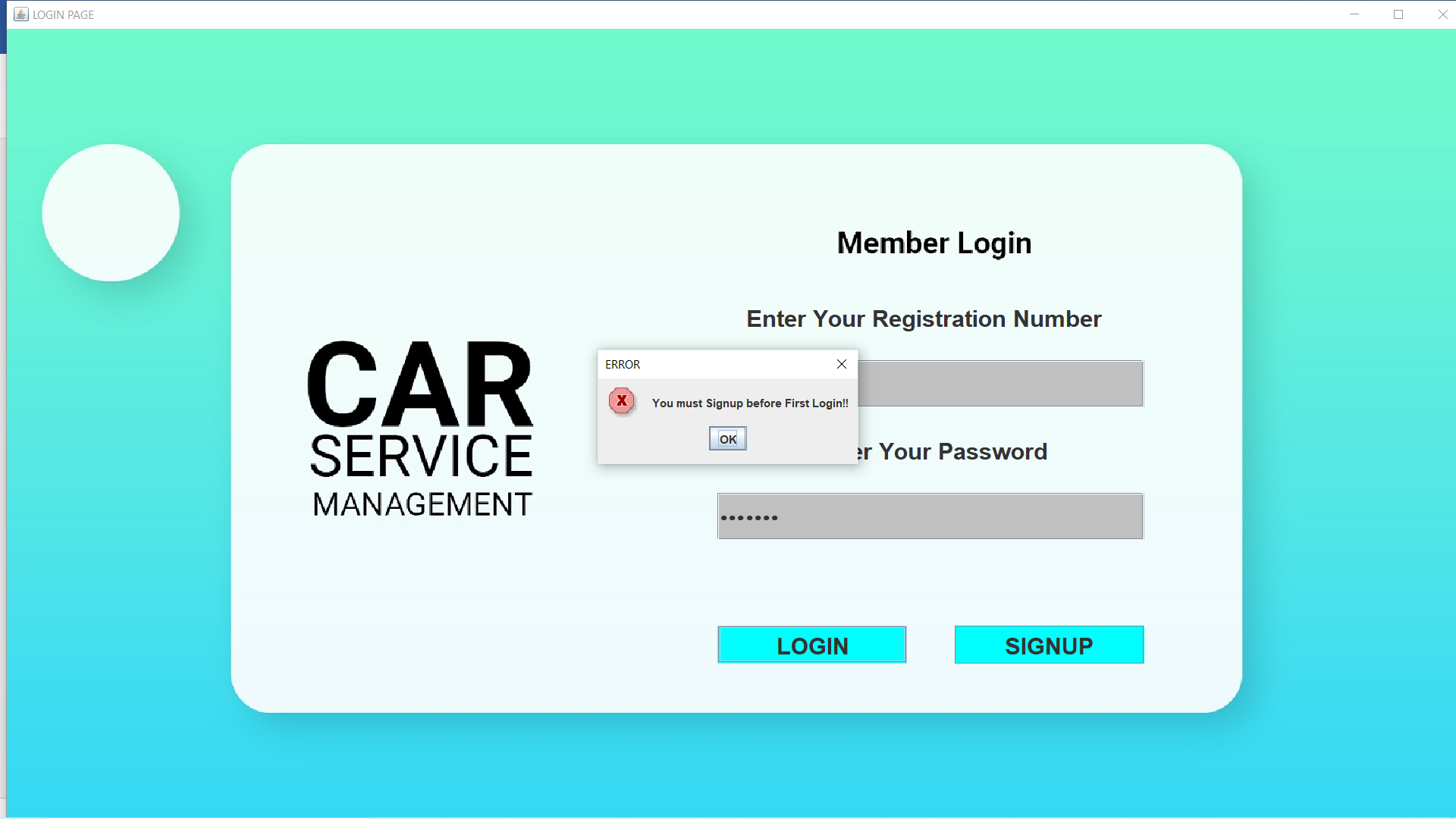
As a initial step we clear all the entries in the login database and work like a new software.



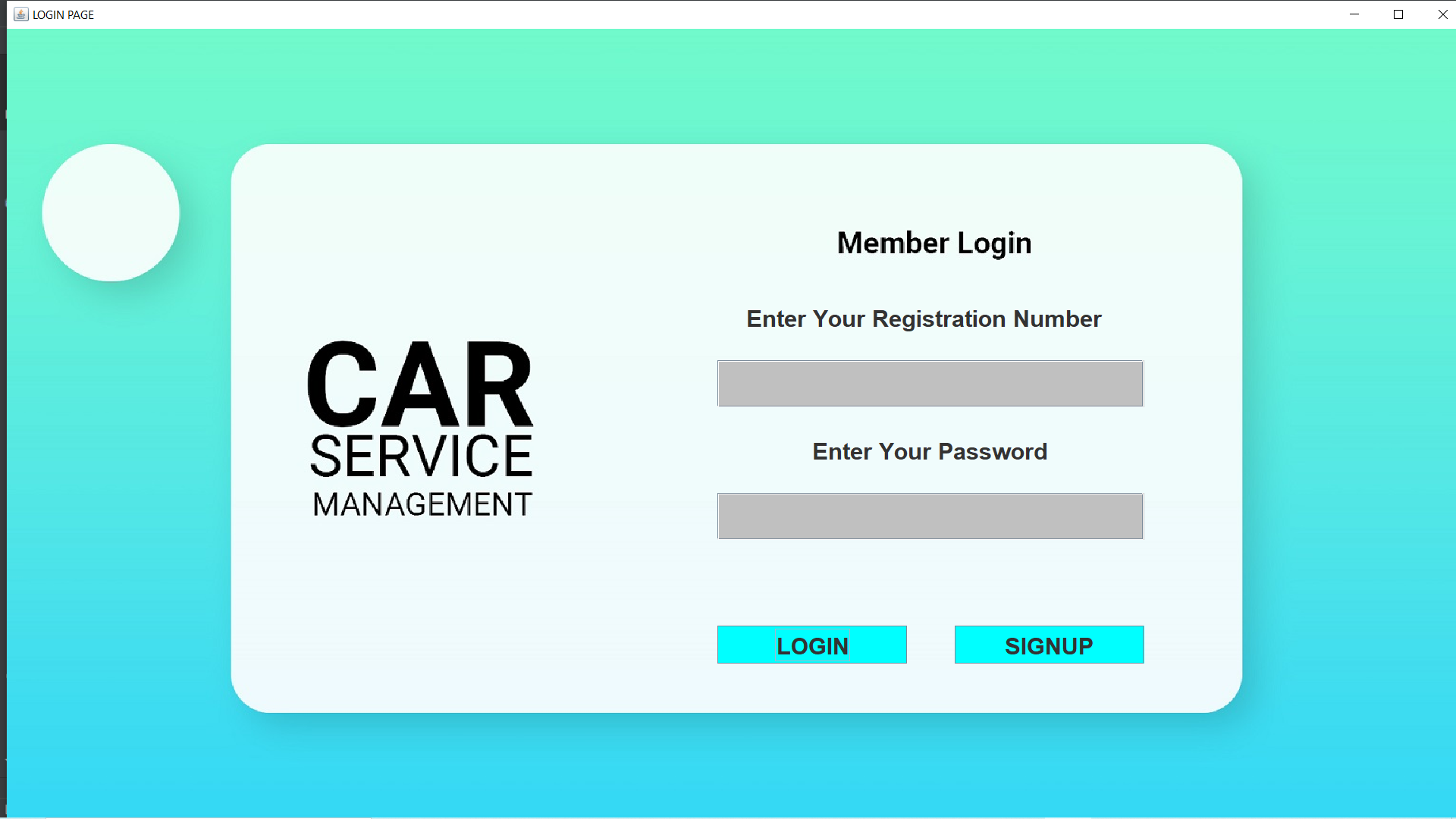
We can see from the above picture that the login database is empty and we work with it as a new software.

Lets now try to login the software and check what we get as a result.

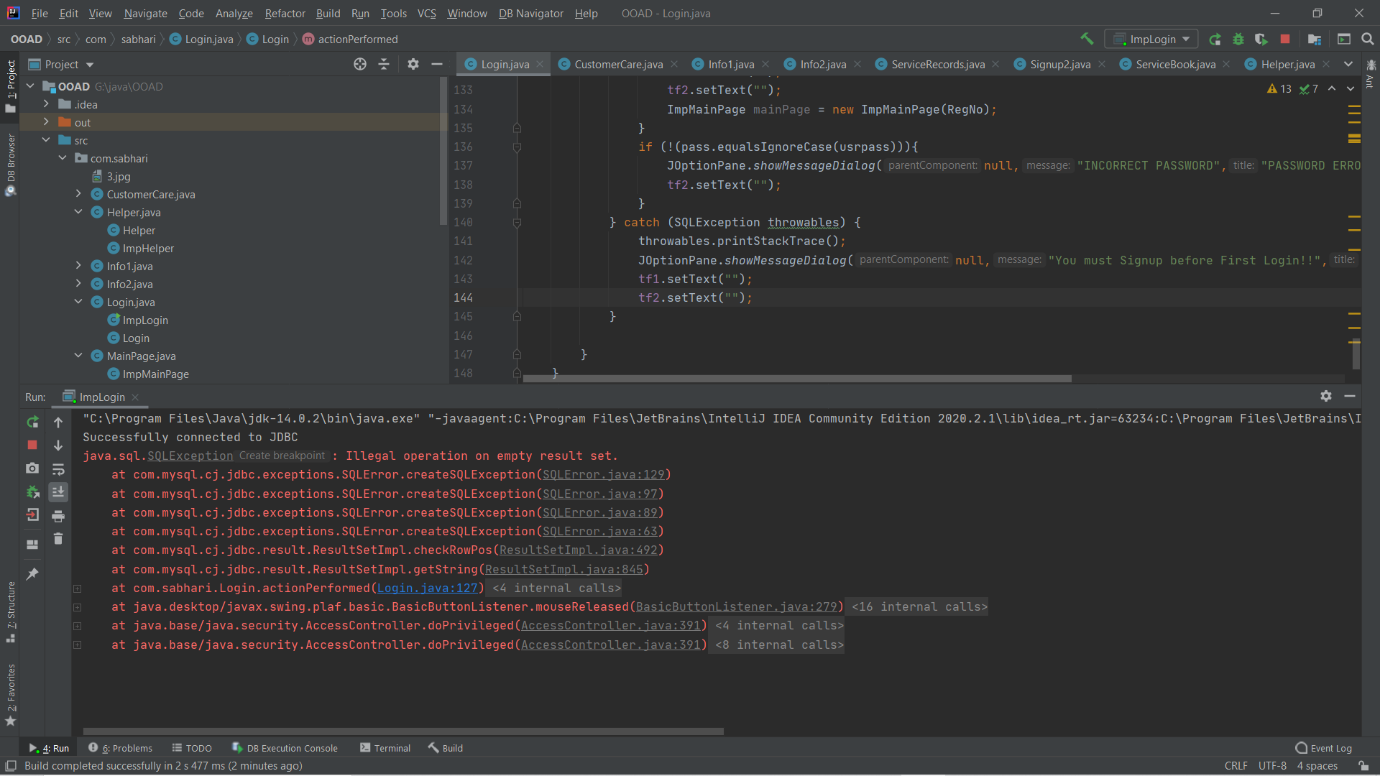
We have entered an registration number and password which is not yet signedup. Let us click login to see what happens.

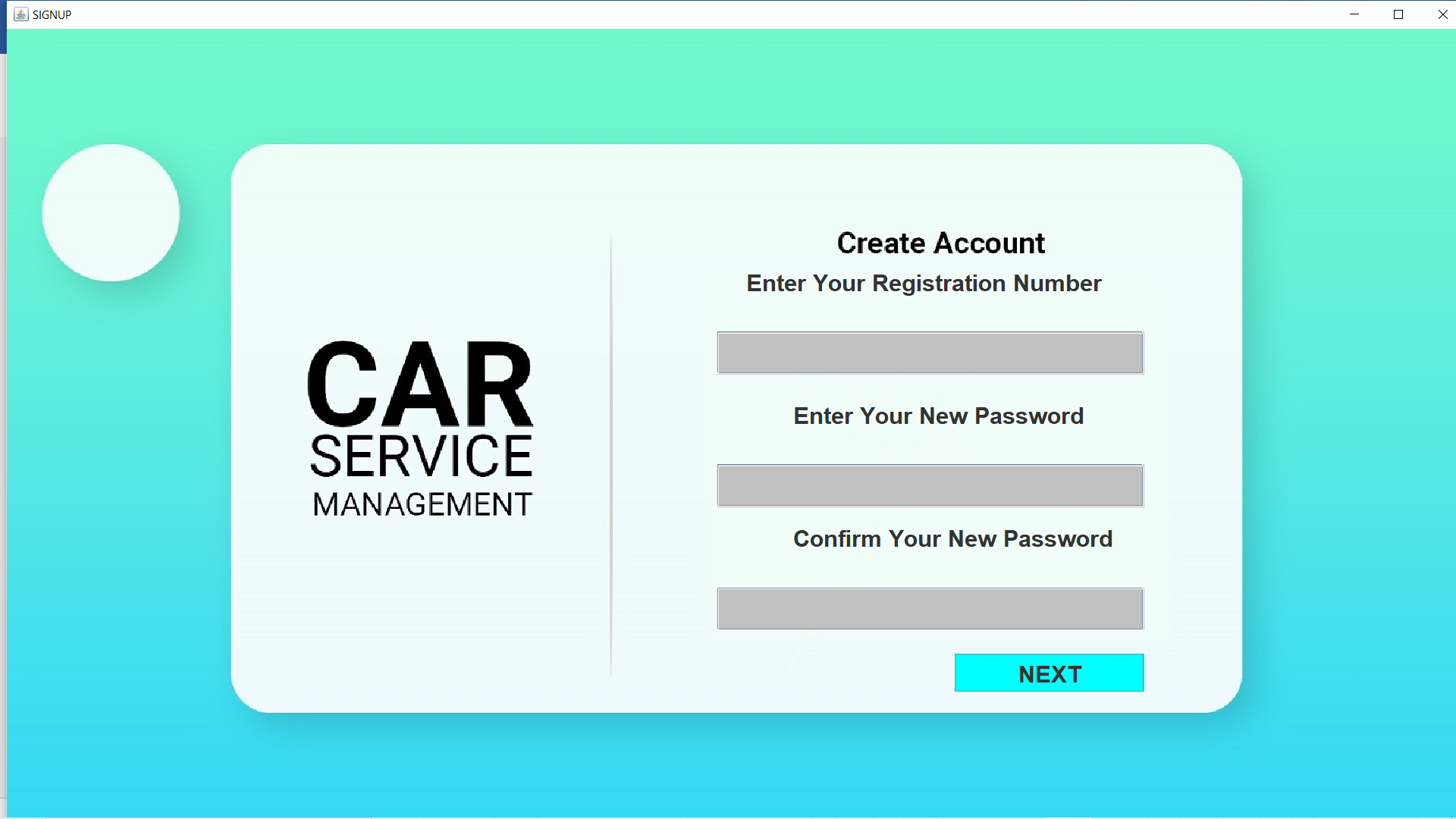


As expected the software prompts me to first sign up before login as the login data is not present in our database.

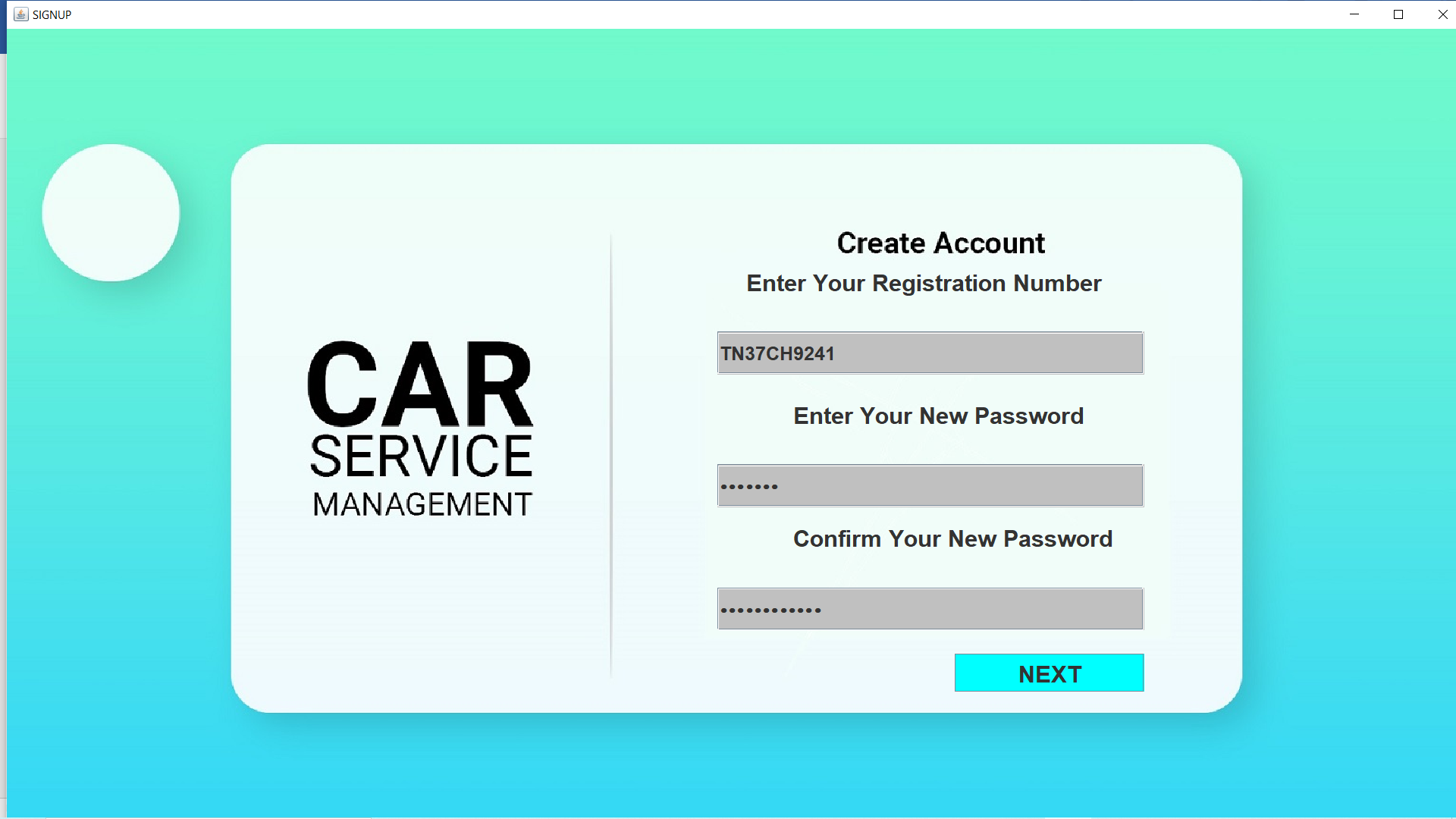


As I press the ok button, the entries in the fields are cleared and set ready for a signup or a login process. In our case we should go with sign up. So I need to click signup button below.

 We get a error of performing operation on an empty set. It is the required error and now we will move on to sign up option.

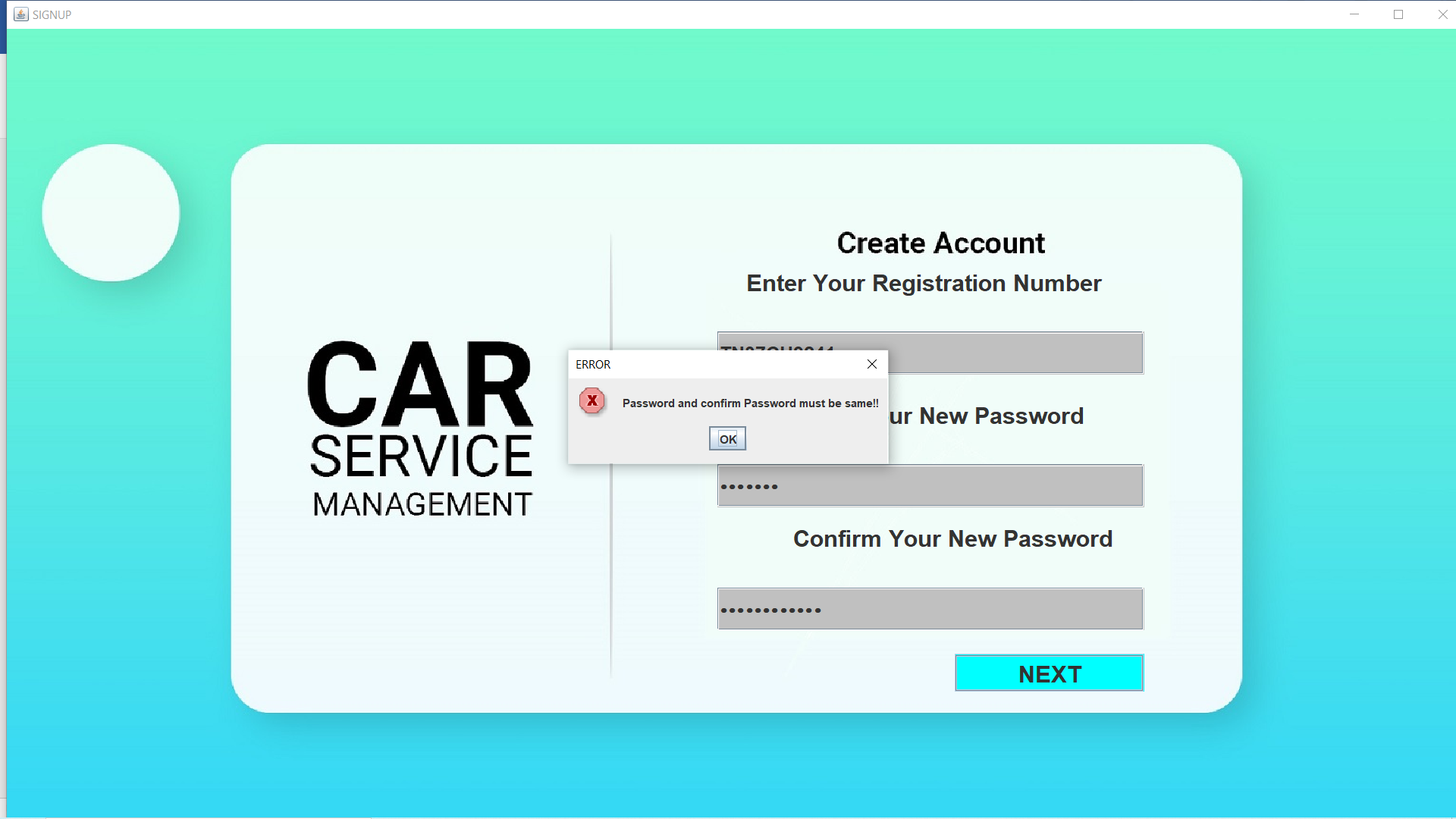
 When we press the signup button the following window appears.

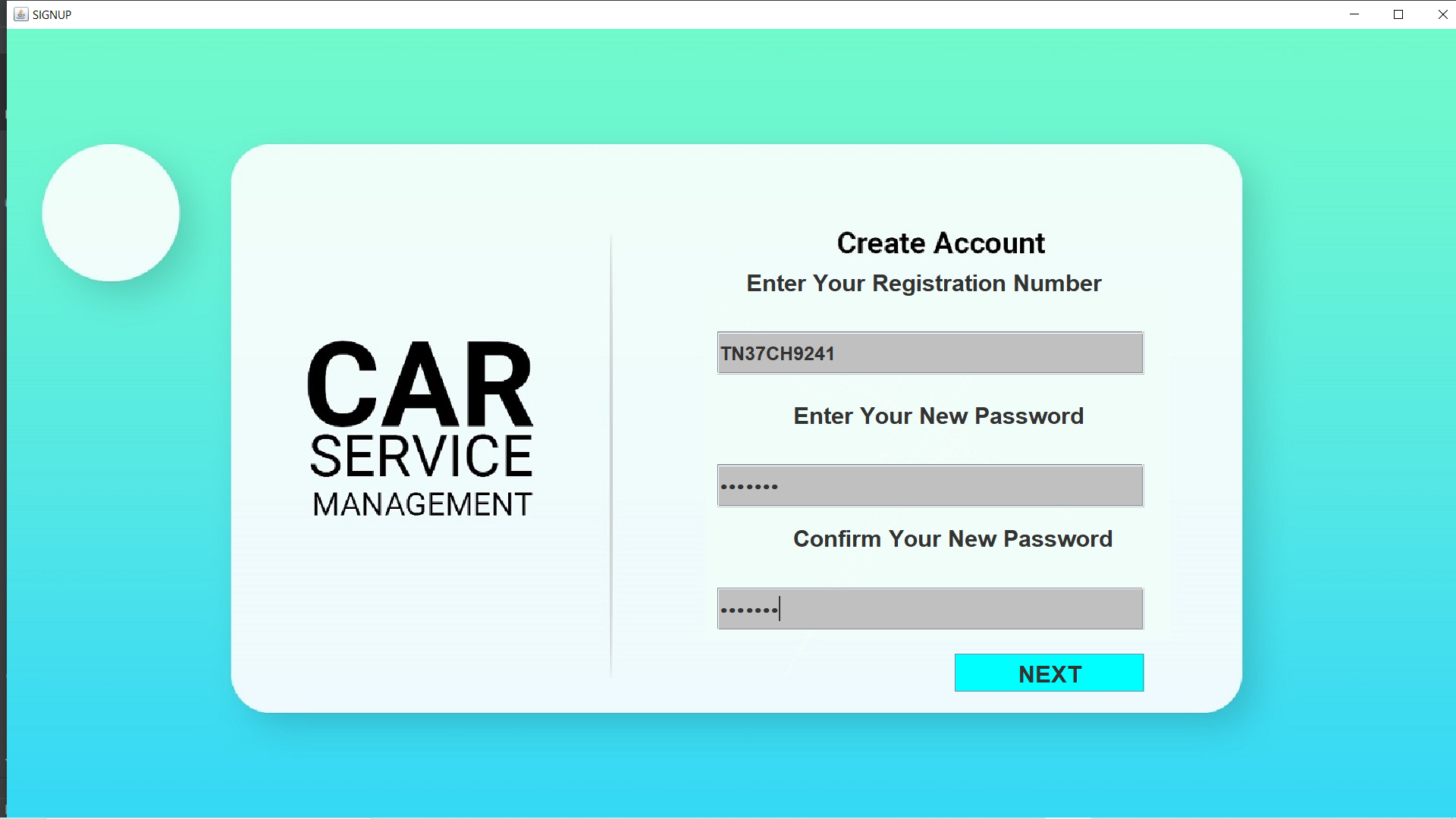
We now going to enter the data into the signup form.



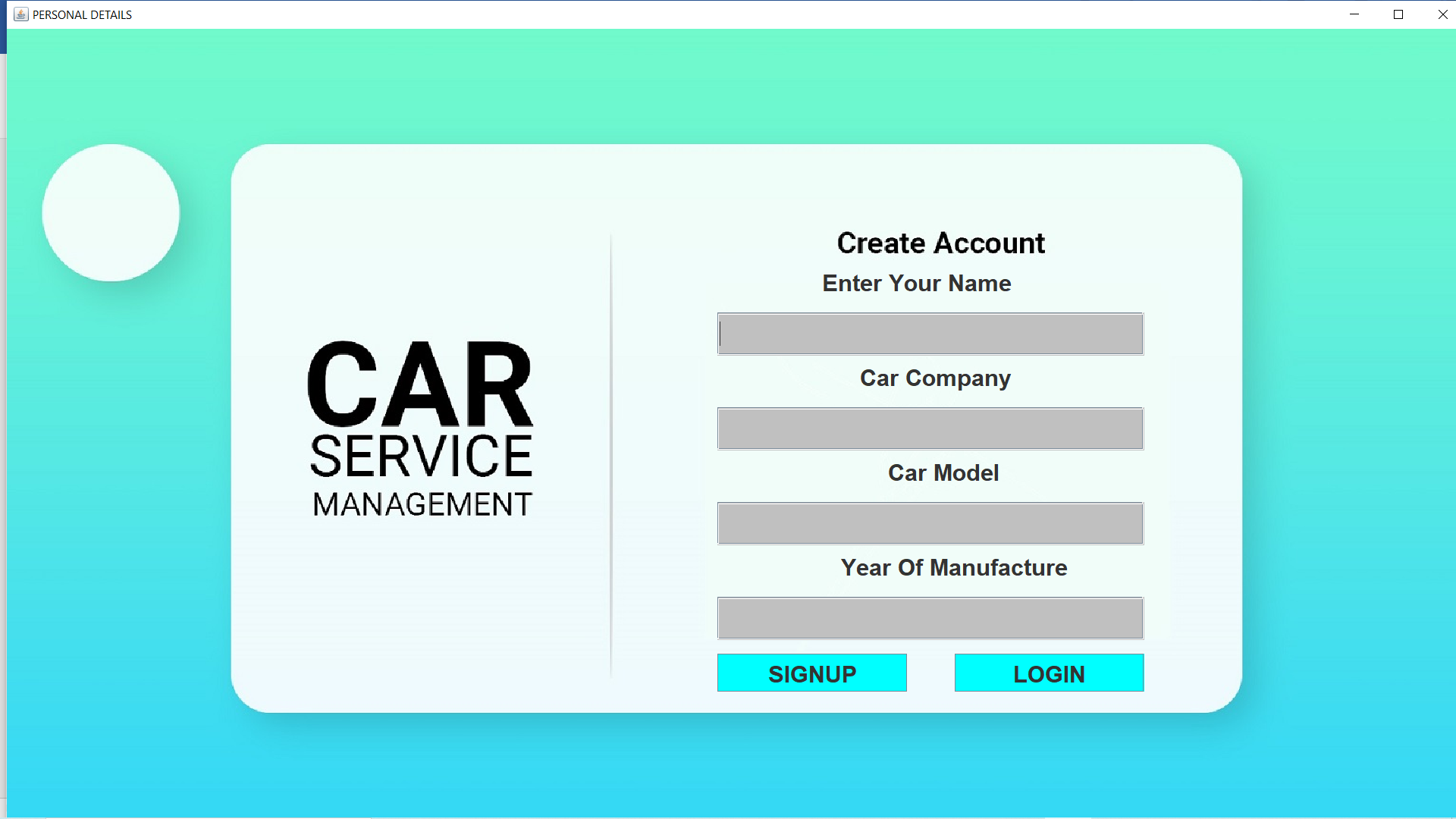
Here we enter the signup data, suppose we enter the new password and confirm password as different. We need to get the error message lets see the error message.

We get the error as password and confirm password must be same. It is the required error and we get it as expected.

 We now enter the correct password and confirm password to proceede with.

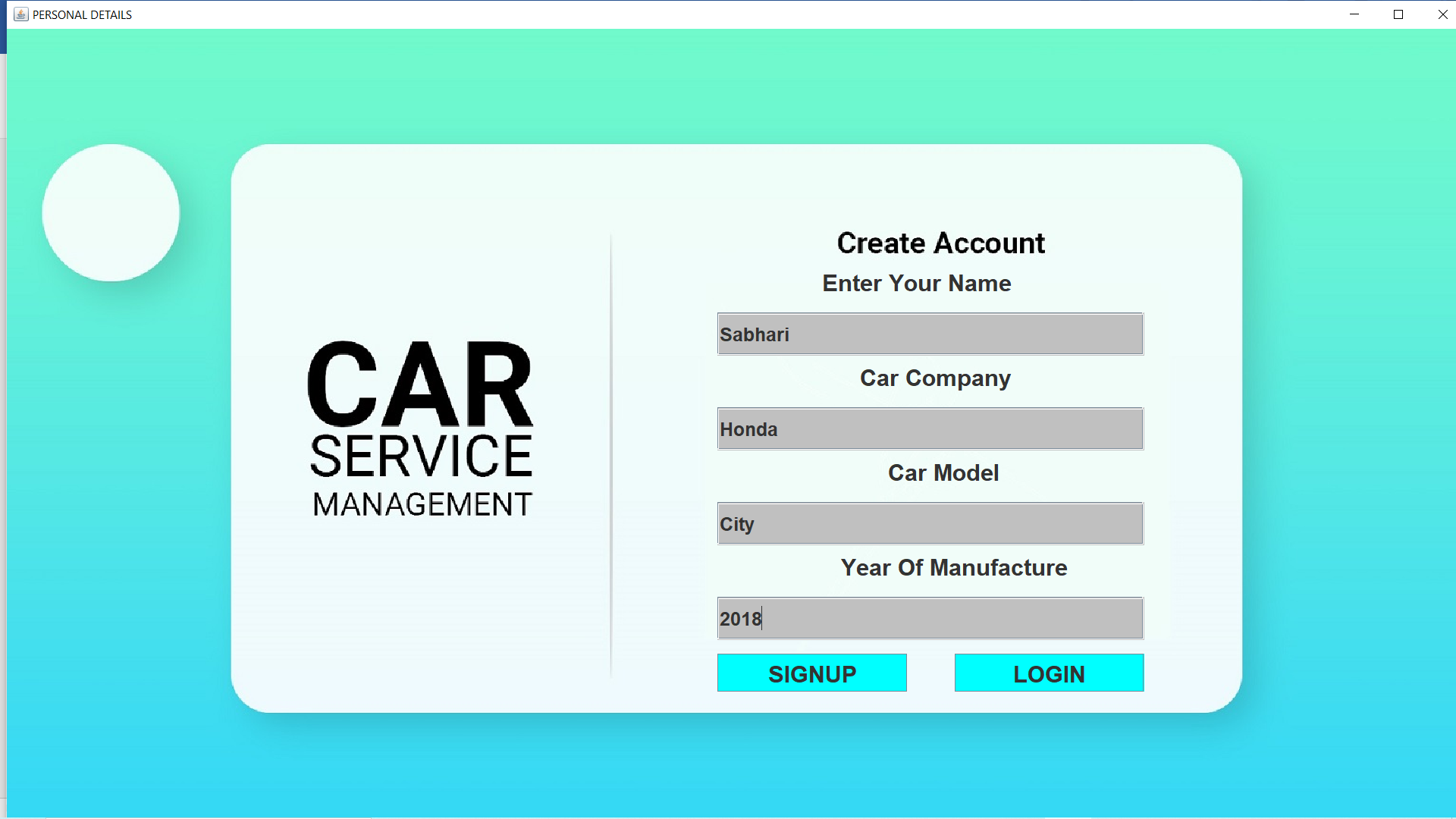


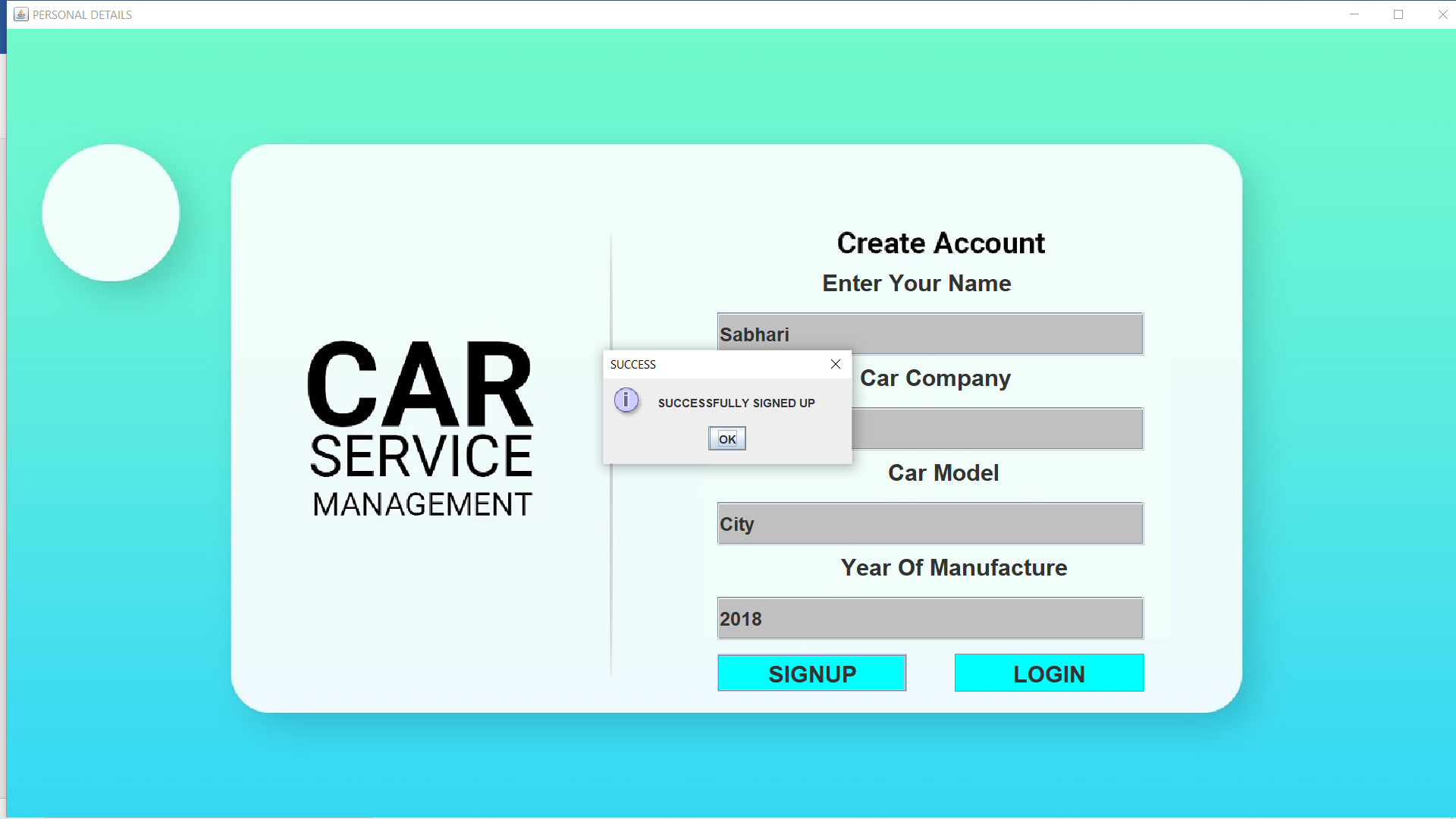
We now press the next button, the password and confirm password are checked and it is correct so the following window appears.



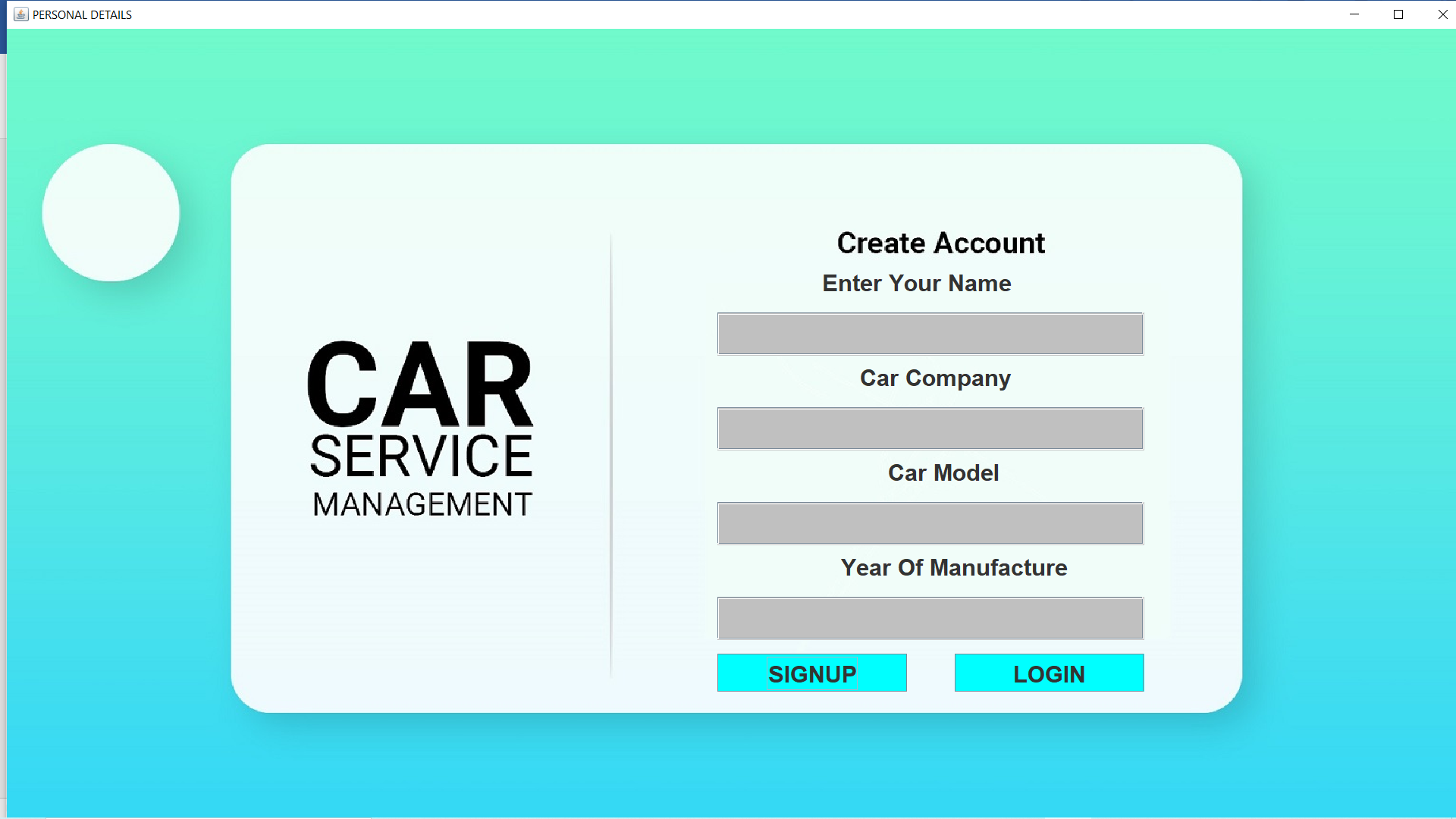
We now enter into the personal detail field of sign in. This will hold few personal detail.

Enter the required data. I had entered the data as below.

Now we press the sign up button to sign up the new user. Lets see the response from our database.

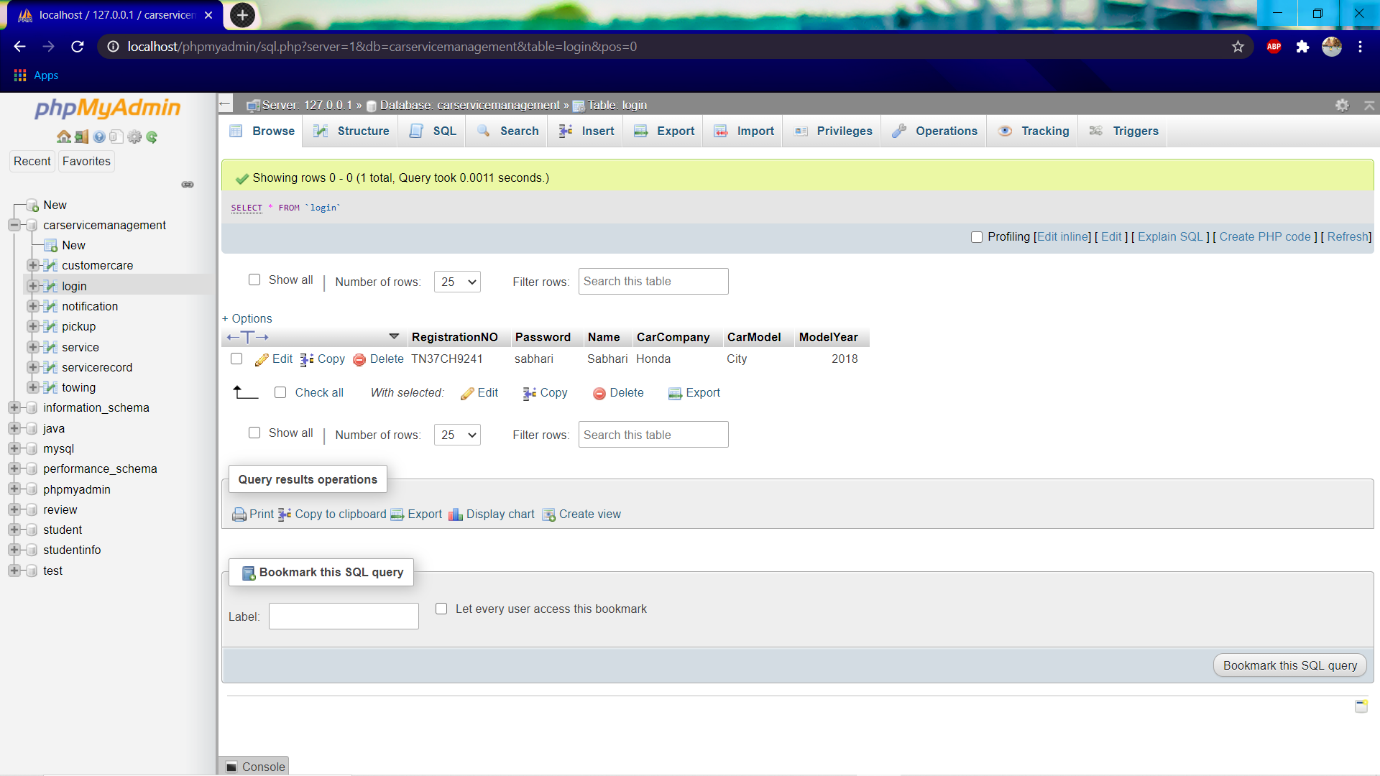


We get the response as successfully signed up from the server. Now we press ok button. The window appears as below

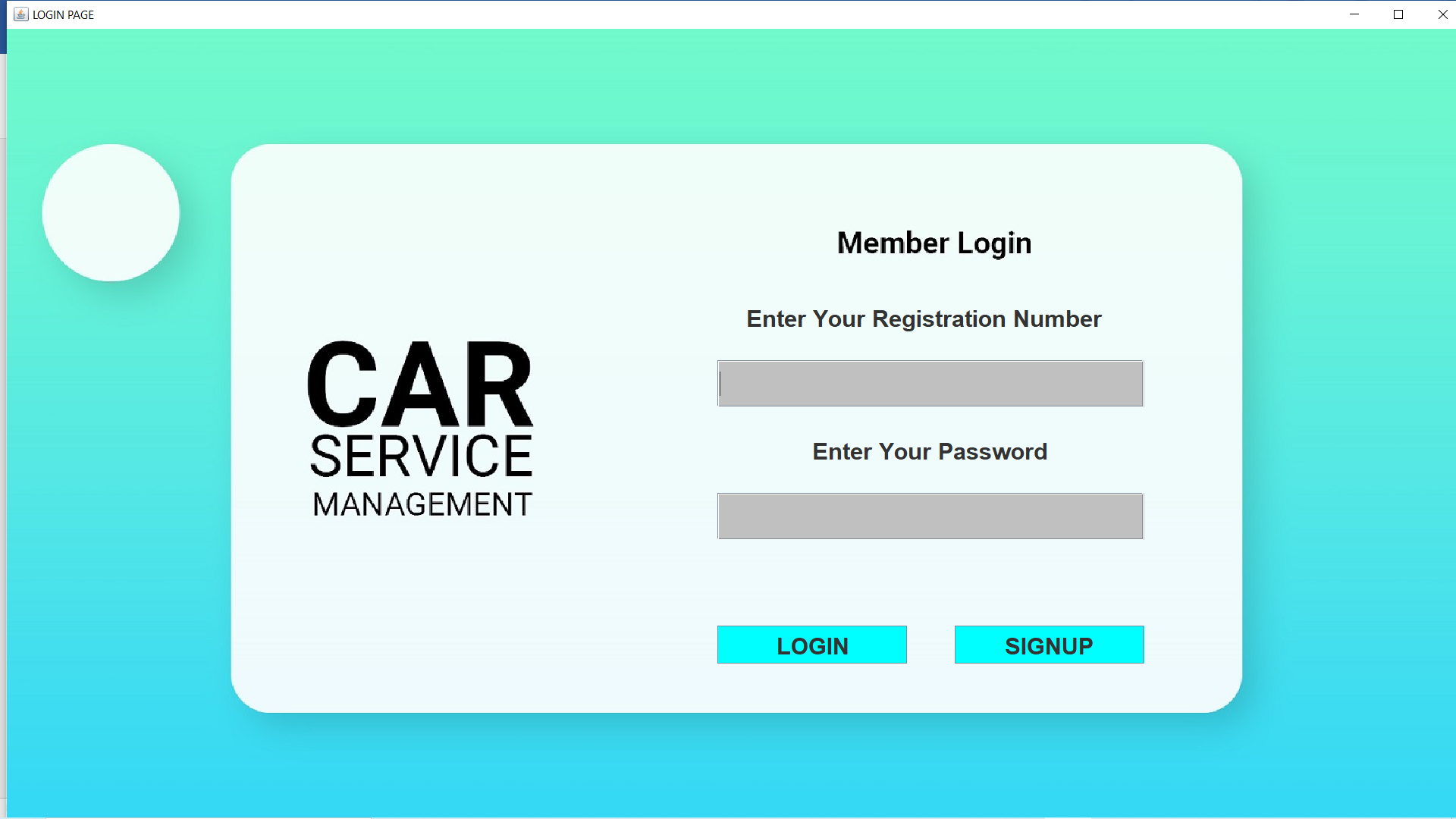


The text fields are cleared and kept ready for the next request. As expected the next request is to press the login button to continue with login process.

Let us now check the database for the data.

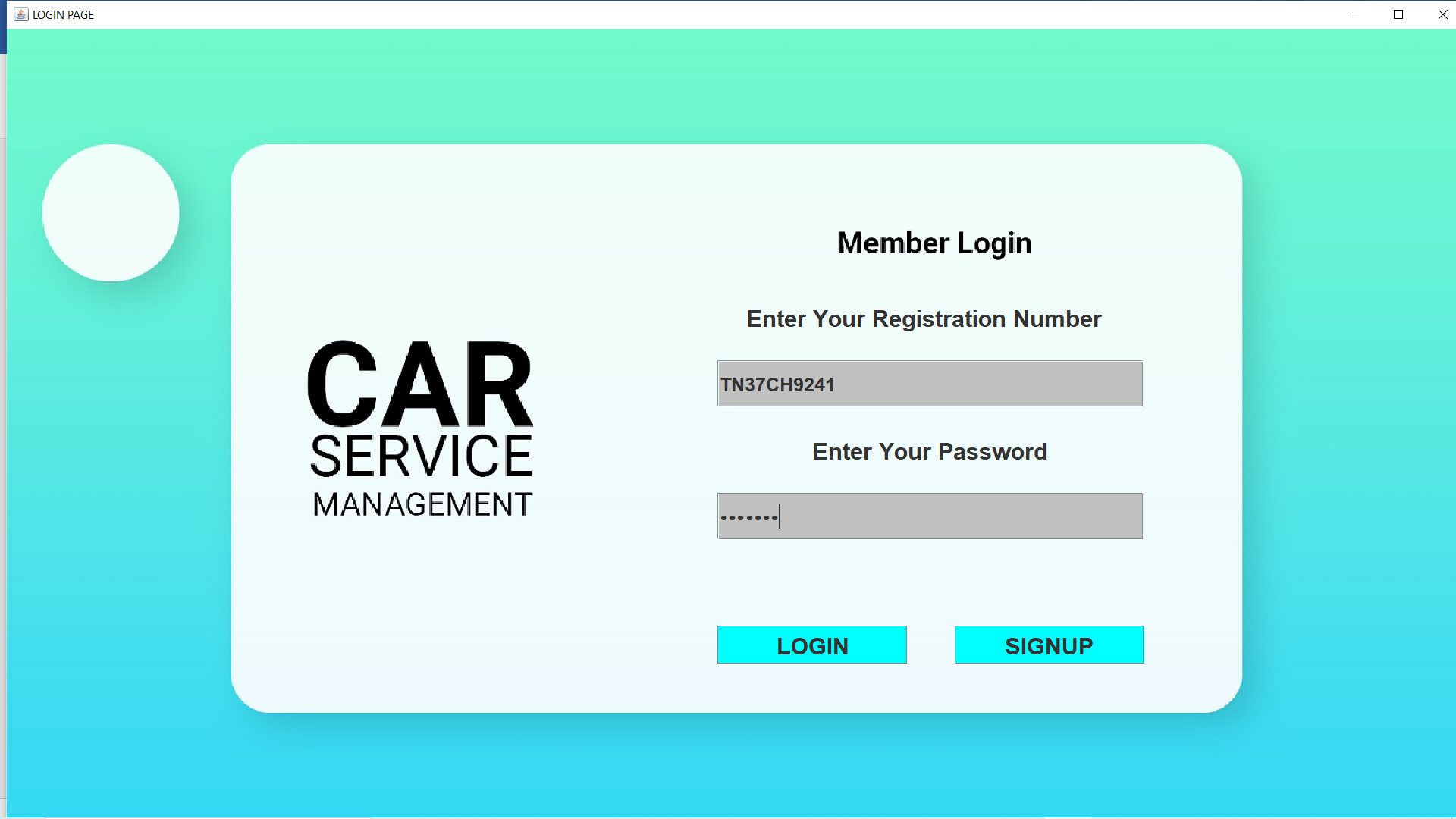
Our signup record was successfully entered into the database. Now we can login with our signup details. Lets get back to the login page and get use of it.

On pressing the login button, we get back to the login page again. Lets click login now.



Now we are back to login page and enter signup details here.

We enter the following details as shown



Now we are going to press the login button.



We get the message as correct registration number and password. Lets press ok button.



On successful login details , we get directed to the main page of our software. The above picture shows the main home page. We have several tabs which satisfies our requirments. We should also note the welcome note at the top. We get the welcome note as welcome and the customer name. This is provided to check whether the user has entered into the correct account. We can also see the car details which is used for checking the correct login entry. Lets now go through each and every tabs to meet our requirments.