Application Task – Guestbook

**Technologies used:**

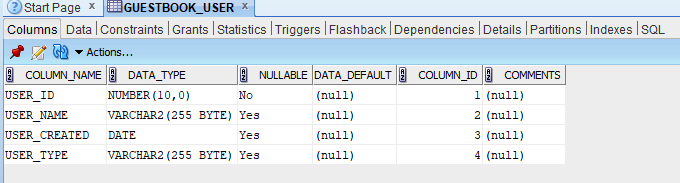
* Backend Framework – Spring Boot, Spring MVC
* Database – MySQL
* Database connection - JPA
* Junit
* Git

**Database Tables and its significance:**

There are 3 tables created namely,

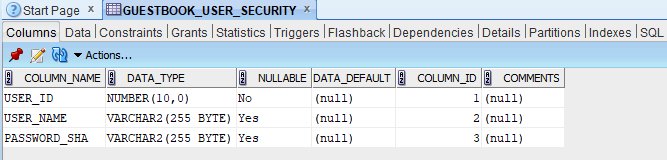
* GUESTBOOK\_USER

Significance of this table is to store username and userType (viz. Admin/ Guest) and the user created Date.



* GUESTBOOK\_USER\_SECURITY

This table stores the username, which is logged in along with its password in encrypted format.

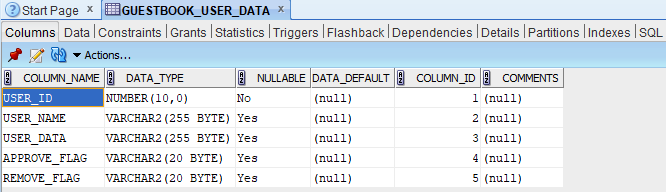


* GUESTBOOK\_USER\_DATA

This table resides of Data (USER\_DATA), which is entered by logged in valid user. Along with that it consist of approve and remove flags (viz. APPROVE\_FLAG and REMOVE\_FLAG).

Approve and remove flag values are “Y” and “N” respectively when user is logged in and enter some data.

Only Admin user has authority to change the value of these flags.



**Database Scripts:**

**Method Explanation:**

GuestBookMain is the main class from where application is get started on port 9000. I mentioned only one controller class (viz. UserFormController) in this application, which is having three different web service calls. Below is the significance of each one,

1. **/services/saveUser:**

This web service expecting username, password and userType from the front end. These three parameters are coming in UserModel where validation from server side also has been written for username and Password.

This web service performs two task as, one, when the user comes first time it takes its username and password and store it in respective tables. Two, If the user is already exist, it keep that user (that means not getting duplicate value of username) and return user is already exist.

This is the request, which is coming from front end,



This is the response, when new user created successfully…



And this is a response when user is already exist,



1. **/services/saveUserData:**

This web service expecting userName, userData, approveFlag and removeFlag from the front end. In this web service two operations are happening.

One, If the user is “Guest” user, he can only able to enter the data. Along with that from front-end approveFlag and removeFlag will come as value as “Y” and “N” respectively by default.

Two, If the user is “Admin” user, he will get all the list of Guest user along with data entered by them and Id, approveFlag as well removeFlag.

From the front-end “Admin” user have button along with userData i.e Approve/Decline button and Remove Button.

When user click Approve/ Decline button approveFlag will change to “N”/ “Y” with respective with operation and as per Approved/ Declined shown in front of user Data.

When Remove Button clicked, Guest User cannot able to see data. In this removeFlag value changes to “Y”.

Request From front-End,



Response of the Admin user,



1. **/services/approveRemoveUserData:**

This web service expecting userName, userData, approveFlag, removeFlag and userId from the front end.

This web service is used to update approve or remove flags on the basis of inputs provided from front-end. In this approve/remove flag updated by checking an userId belongs to that userData.

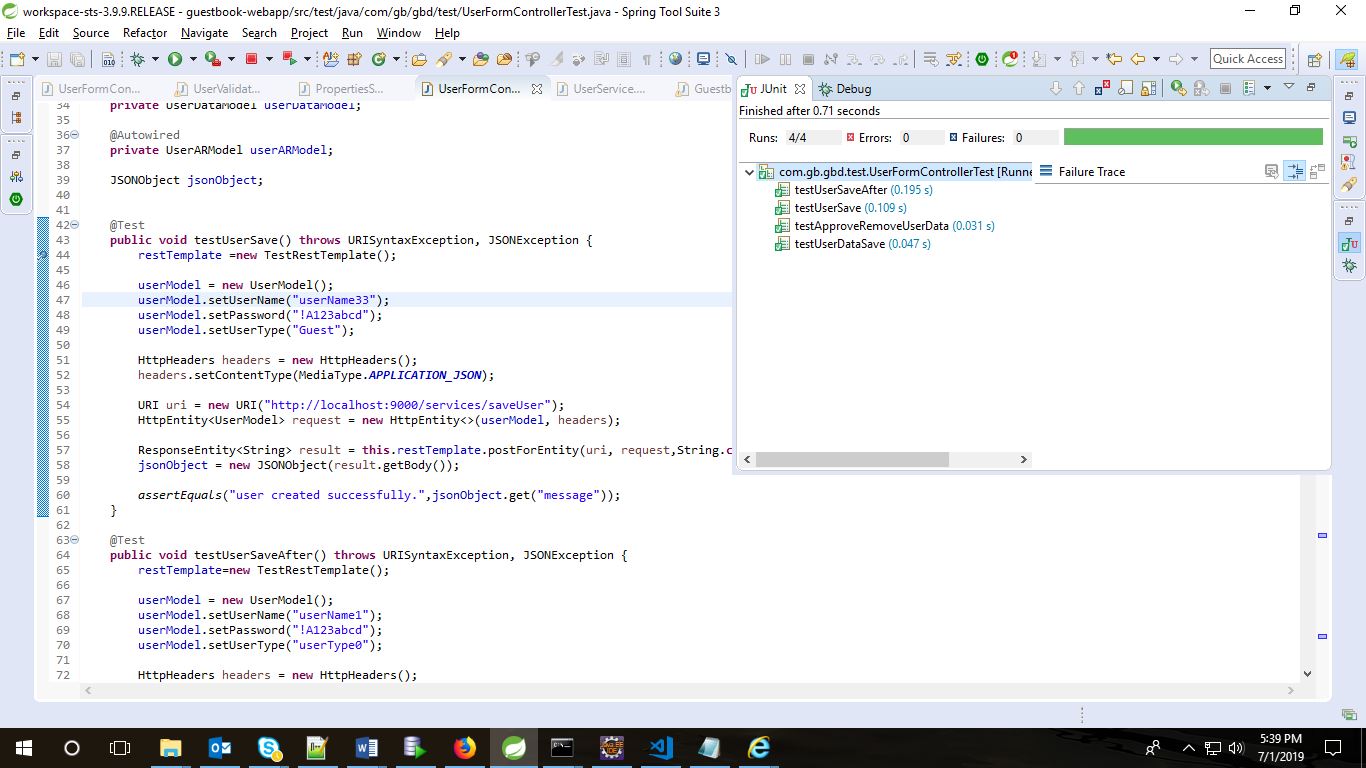
Request:



**JUNIT:**

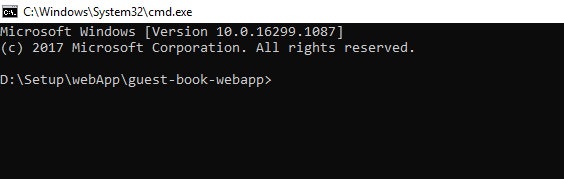
Following are the Test covered for each functionality,

* **testUserSave():** This test case is for checking that the user which is logged in is saved in database successfully or not.
* **testUserSaveAfter():** This test case is to check user is existing or new one.
* **testUserDataSave():** This test case is for data entered by logged in user ie saved in database successfully or not.
* **testApproveRemoveUserData():** This test case is for testing updated approve or remove flags getting updated in respective table or not.

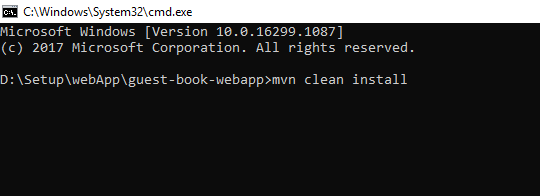
**JUNIT Evidence:**

**How to Execute Application:**

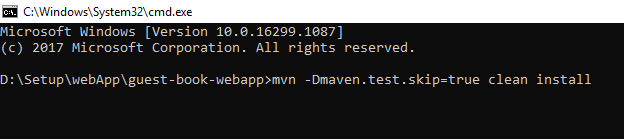
1. Save application on your local machine and go till the path /guest-book-webapp in command prompt.



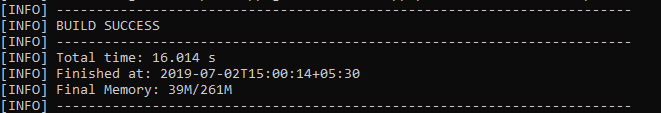
1. Run commans as “ mvn clean install” which will take a while in processing as whole required libraries get downloaded.



1. To avoid test case to check run “” this command.



After successful execution of any of this command, you will get following message as,



1. After BUILD SUCCESS, “guestbook-webapp-0.0.1.jar” will be created. After successful jar creation run below command as,

“java -jar <path up to jar>\guestbook-webapp-0.0.1.jar”

1. With this application get deployed on tomcat server.