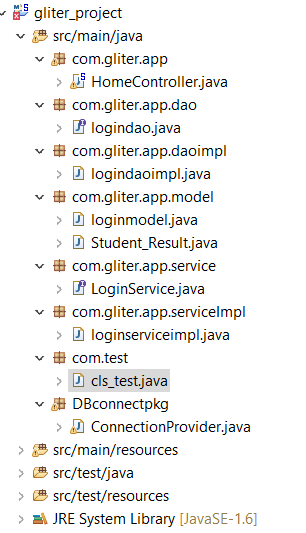
**Lets understand the souce code of this app :**

**Project Name :- gliter\_project**

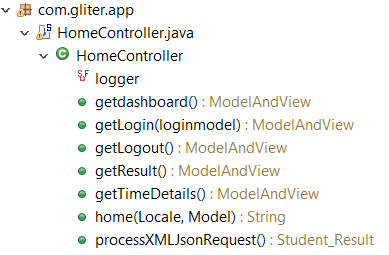
**Src folder :-**

In src folder such packages are available which is display like bellow **figure 1,**



Here some packages which is com.gliter.app,dao,daompl,service,serviceimpl,model,test,dbconnect.

In com.gliter.app has one class which is homecontroller.java .

 In homecontroller.java contains such methods which is mapping the resources as per request format here is define all the mapping in front controller which name is described in bellow figure.

1)home() method is used for display the default page

2)getdashboard () used for mapping the request while user hit the url(…/dashboard)then it’s dipaly dashboard page

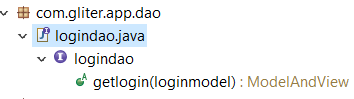
3) logout() used for mapping the request while user hit the url(…/logout)then it’s dipaly Login page and get back from the application.

4) processXMLJsonRequest() this method is used for getting the data from database and convert into the xml format

5)gettimedetails() is used for mapping the request(…/TaskTiming) diplay the line chart on screen after clicking right side manu of time taken in dashboard page.

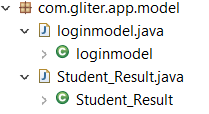
2) com.gliter.dao

Which is contailns one logindao interface which is one method which is parameterised. The fig is given bellow.

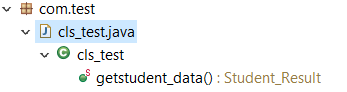


For the implementasion used logindaoimpl class which is under the com.gliter.daoimpl package here is control which view is bind like dashboard or failure page. For loginservice used as com.gliter.service consists a one abstract method and implements on serviceimpl pkg which is contain main logic like user name and password mached or not if result is true then bind dashboard else filure page.

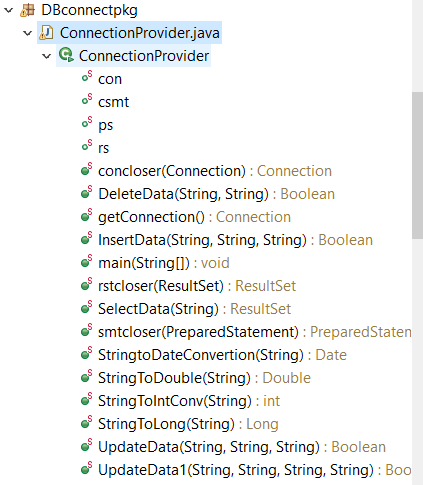
3) com.gliter.model in this package 2 classes which is student\_result.java and loginmodel.java classes as per bellow figure



4)com.test has one file which is cls\_test.java

In this class one static method which is getting the details from database. 

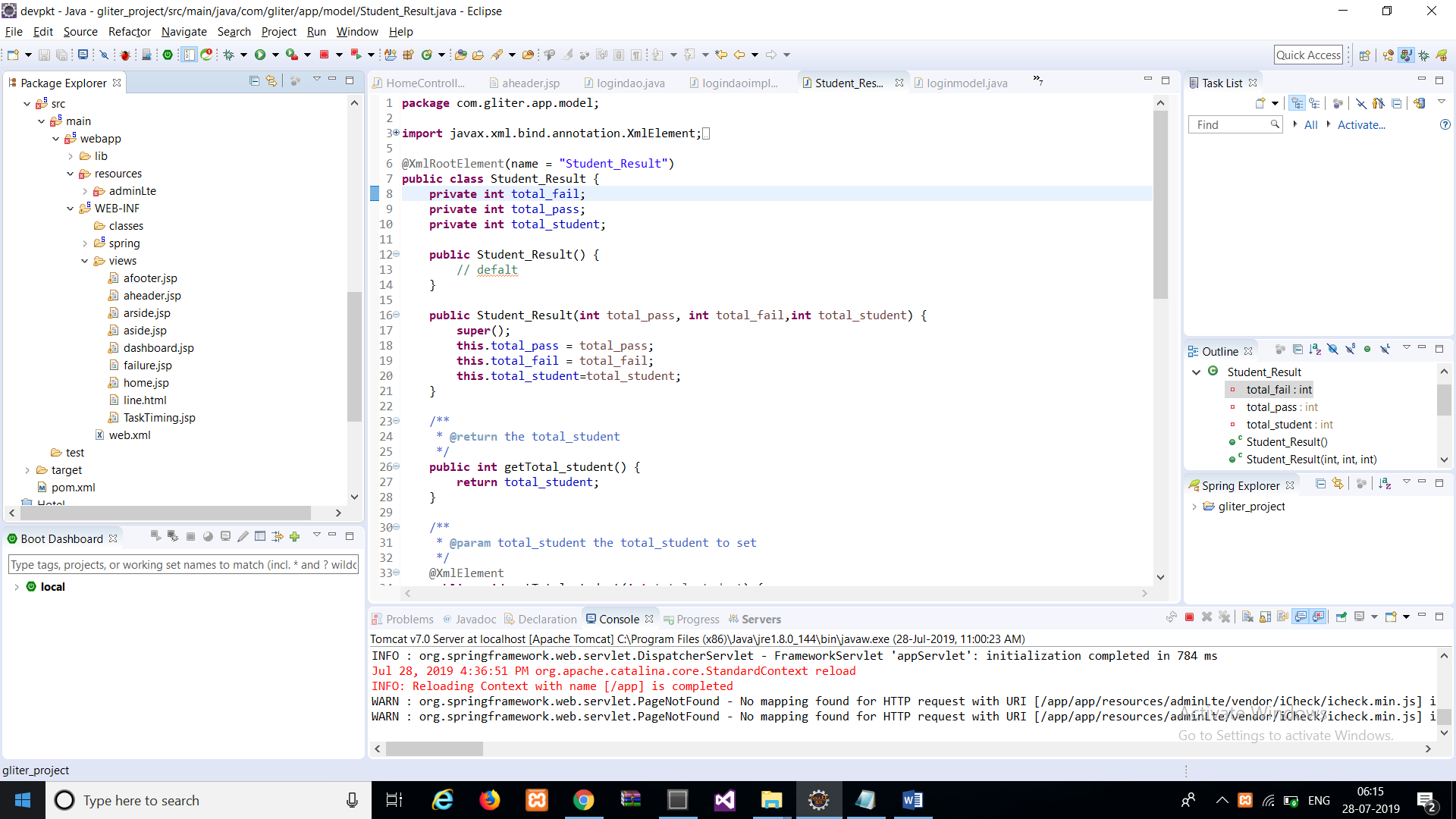
5) DBconnect package has one class connectionprovider.java which contains verious stathic methods this methods like bellow figure.



This methods are used for getting the connection from database.getting the result and return as result set to the source object basicaly it’s a userdefine utility class. Wich support convertion,getting data from database ,insert,update,delete,get current system date,string to date etc.

**For view** used in theis project template (Admin Lte Template).

Just take a look of bellow figure.

****

In resources folder it contains adminlte folder which contains all css and jquery plugins.

Under the web-inf folder Views folder contains all the view pages like above figure.

1)dashboard,Tasktime,failure,login,aheader,aside,arside,afooter.jsp

i)template pages:- aheader,aside,arside,afooter.jsp

ii) Bodys :- dashboard,Tasktime

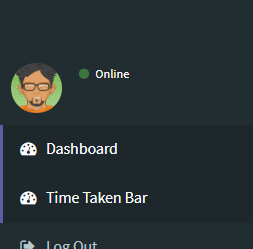
Description :-page has devided into 5 parts bcouse this code are constant to all pages but body is defer so all static functionality divided into parts like.(This is also called code reutilization and user wants to change them then only chage into three files like (header,footer,aside )pages.)

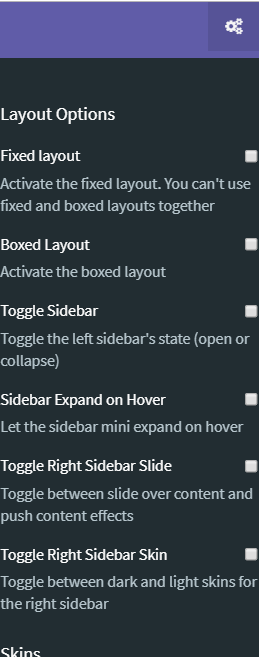
**Header.jsp** : - this page contains the design of header page like bellow figure



Which contains all css source file links.

**Aside.jsp**:- this is ysed for all manues to display like bellow.



**Rside.jsp** :- used for the thems related setting while click on gears icone on headers right side it’s shows like

**Footer.jsp** :- this is diplay bottom of the page and also contains all jquery links. Have a look of bellow fig.



**Dashboard.jsp** : - this page contains partial pages.

Like header,asides body,footer.

In dashboard use for display a pie chart using some js file and css file after parsing the xml data and pass this data to the js files it’s shows the pie chart.

For parsing the xml data using the jstl tag library.

**Timetaken.jsp** :- used for displaying the barchart on the page

**Failure.jsp** :- if any error ocured or result is false then it’s displayed.

**Login.jsp** :- this page is used for authentication after authentication enter into the application.