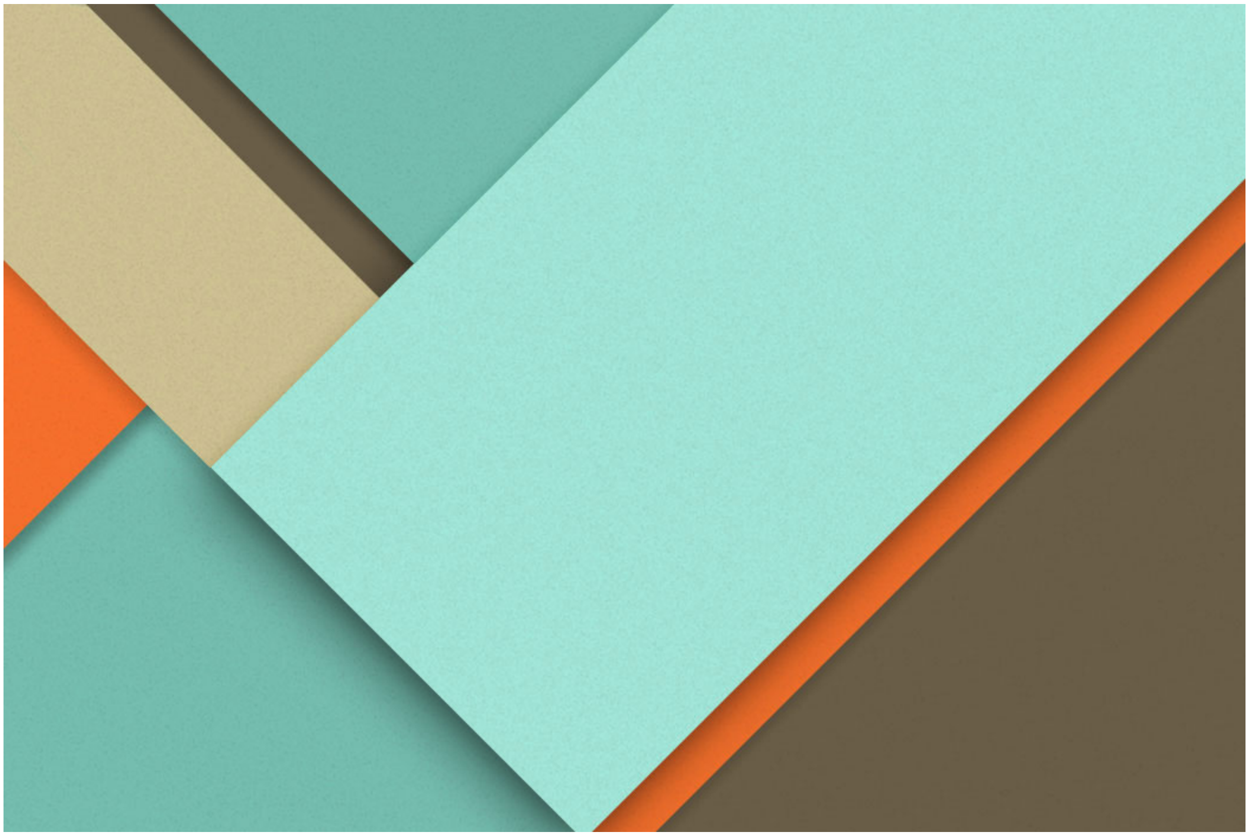


E-Plan

08.11.2015

Database Applications Project

CS 387

Utkarsh Kumar	130050022
Anand Bhoraskar	130050025
Pranjal Khare	130050028

Introduction

Today's world is a world of specialization. All professionals work in teams with each individual highly accomplished in his/her own field. Collaboration is the way to go. And any real life problem must be tackled by a team. In such a situation, coordination is paramount. Our application will aid such by doing the coordination part for them.

Our application can be used by anyone who is working in a team, be it students, researchers, IT professionals, consulting professionals, finance professionals… well almost anyone. It can also be put to use in event management to keep a close eye on the exact progress so far.

Database Design

A sample database can be created by using the following scripts:

DDL_drop.sql:	Drops all the tables
DDL.sql:	Creates new Tables
InsertValues.sql:	Inserts Values into the Tables

There are a total of 7 tables in our databases.

A. teams

This table stores the Team Name and its Leader

B. users

This table stores all the basic information about a user from its username, password to his phone number and mailing address.

C. tasks

This table stores all the details regarding a task i.e. name, description, assigned_by, supertask (if any), deadline, etc.

D. resources

This table stores resources file in the form of BYTEA and its filename.

E. teamAssign

This table stores the mapping between team_id and user_id of the user that lies in that team. A single team can have multiple users as well as a single user can have multiple teams.

F. taskAssign

This table stores the mapping between task_id and user_id of the user to whom the task is assigned to. A single task can be assigned to multiple users as well as a single user can have multiple tasks.

G. resourceAssign

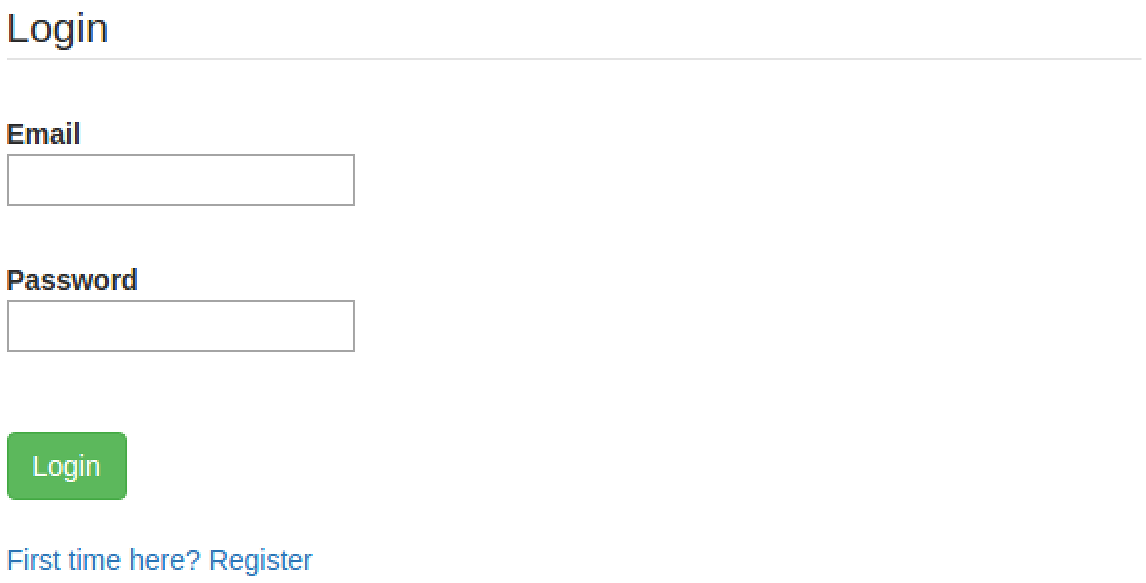
This tables stores the mapping between resource_id and task_id of the task on which the resource is shared. A single resource can be shared among multiple tasks and a single task can have multiple resources.

Detailed Application Design and Implementation

I. JSP's

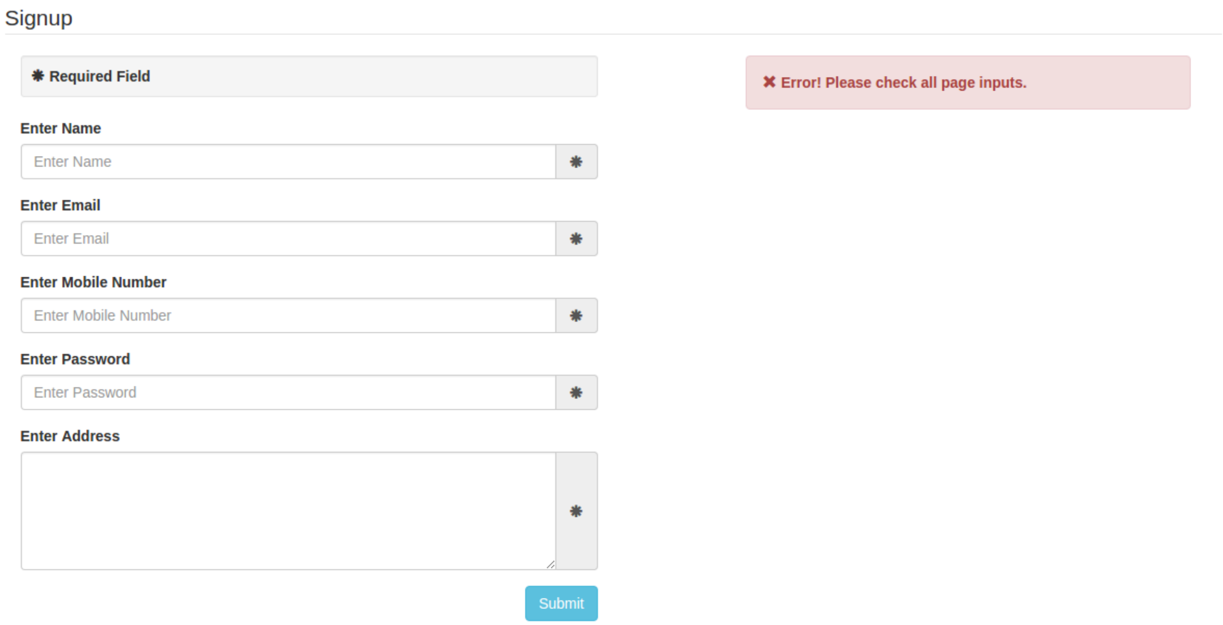
A. Login

This JSP is used for making the simple login interface and also provides a link to user registration form.



B. Signup

This JSP is used for making the form for registering a new user. All the fields are marked as required and also undergo integrity checks like correct email format etc.



Signup

✱ Required Field

✖ Error! Please check all page inputs.

Enter Name

Enter Name ✱

Enter Email

Enter Email ✱

Enter Mobile Number

Enter Mobile Number ✱

Enter Password

Enter Password ✱

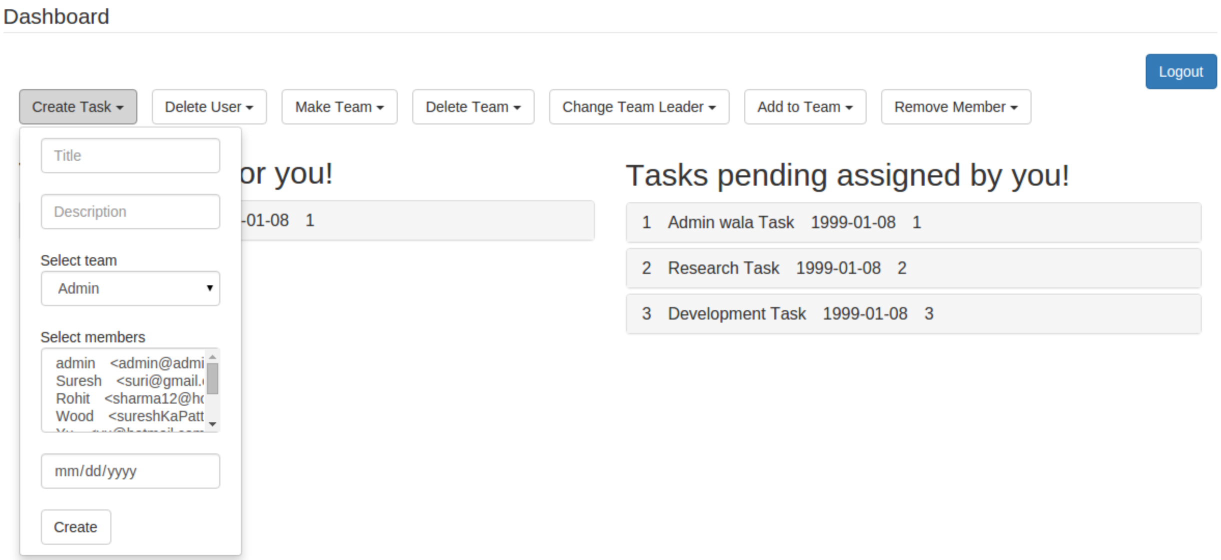
Enter Address

✱

Submit

C. Dashboard

This is where the user will have all the functionalities depending on whether he\she is Admin or normal User.



Dashboard

Logout

Create Task ▾ Delete User ▾ Make Team ▾ Delete Team ▾ Change Team Leader ▾ Add to Team ▾ Remove Member ▾

Title

Description

Select team

Admin ▾

Select members

admin <admin@admi
Suresh <suri@gmail.
Rohit <sharma12@hc
Wood <sureshKaPatt

mm/dd/yyyy

Create

or you!

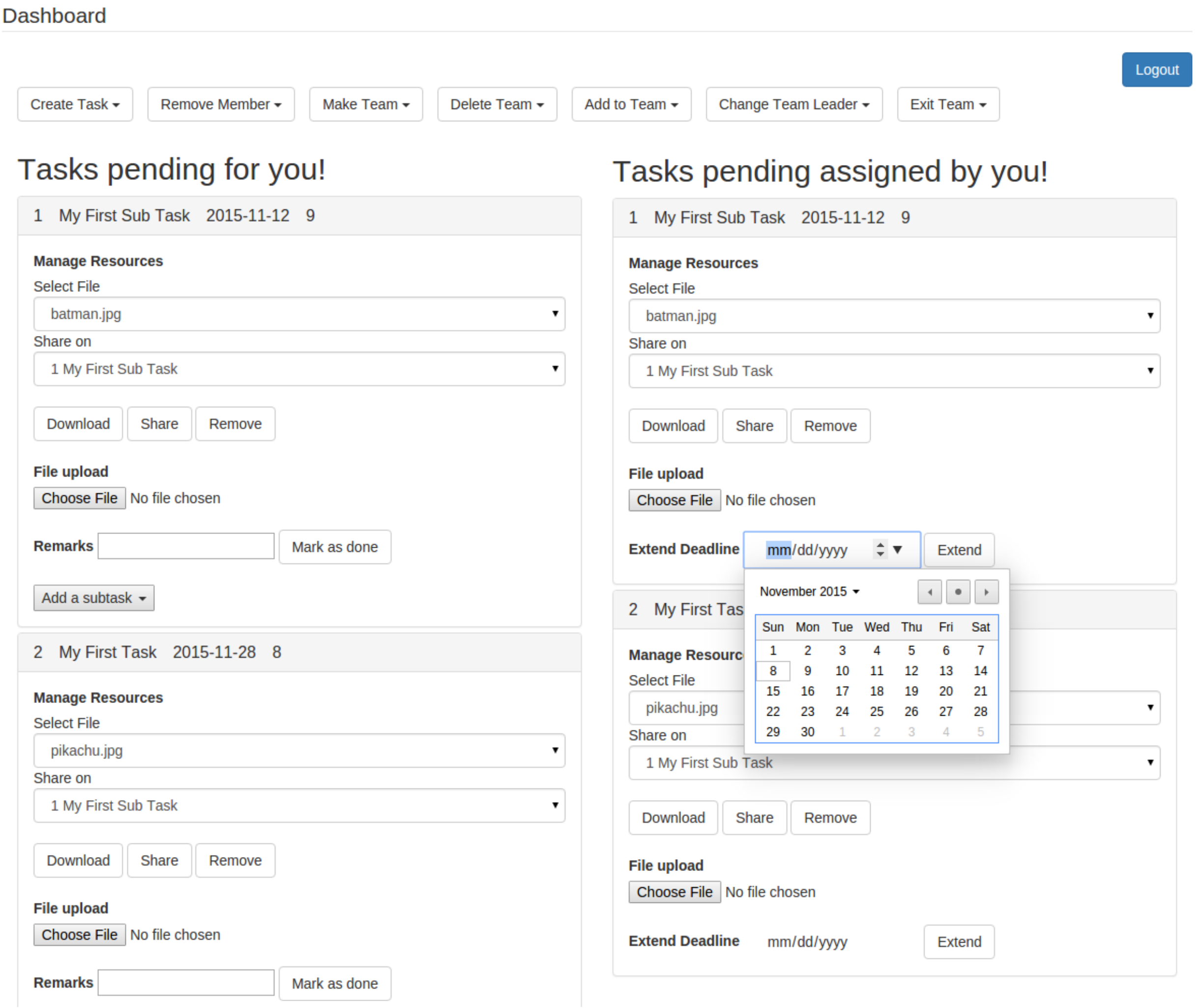
-01-08 1

Tasks pending assigned by you!

1 Admin wala Task 1999-01-08 1

2 Research Task 1999-01-08 2

3 Development Task 1999-01-08 3



Dashboard

Tasks pending for you!

1 My First Sub Task 2015-11-12 9

Manage Resources

Select File: batman.jpg

Share on: 1 My First Sub Task

File upload

No file chosen

Remarks

2 My First Task 2015-11-28 8

Manage Resources

Select File: pikachu.jpg

Share on: 1 My First Sub Task

File upload

No file chosen

Remarks

Tasks pending assigned by you!

1 My First Sub Task 2015-11-12 9

Manage Resources

Select File: batman.jpg

Share on: 1 My First Sub Task

File upload

No file chosen

Extend Deadline: mm/dd/yyyy

November 2015

Sun	Mon	Tue	Wed	Thu	Fri	Sat
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5

2 My First Tas

Manage Resourc

Select File: pikachu.jpg

Share on: 1 My First Sub Task

File upload

No file chosen

Extend Deadline: mm/dd/yyyy

II. Servlets

Inside our web.xml file we have defined a mapping for the following two servlets and also stated login.jsp as welcome page of our app.

A. Login

This servlet mainly deals with redirection between various JSP's.

Whenever a user performs some action eventually he has to get through this servlet which will redirect him to Dashboard or Login screen depending on which action he performed, whether he is already logged in or his session has expired and needs to login again, etc

Login and Logout buttons directly lead to this servlet.

This will perform the authentication while logging in and also deletes the session when logging out.

B. Submit

This servlet is called whenever user tries to use any functionalities inside the Dashboard.

When the user tries to register he is also redirected to this servlet.

Inside this servlet we have performed various Integrity Checks on the input provided by the user before passing them to the Java functions which performs queries. Also if there is any error in the input it also provides a proper error message to the user.

This servlet will ultimately redirect to the Login servlet which will then redirect us to either Dashboard or Login screen based on our state.

III. Classes

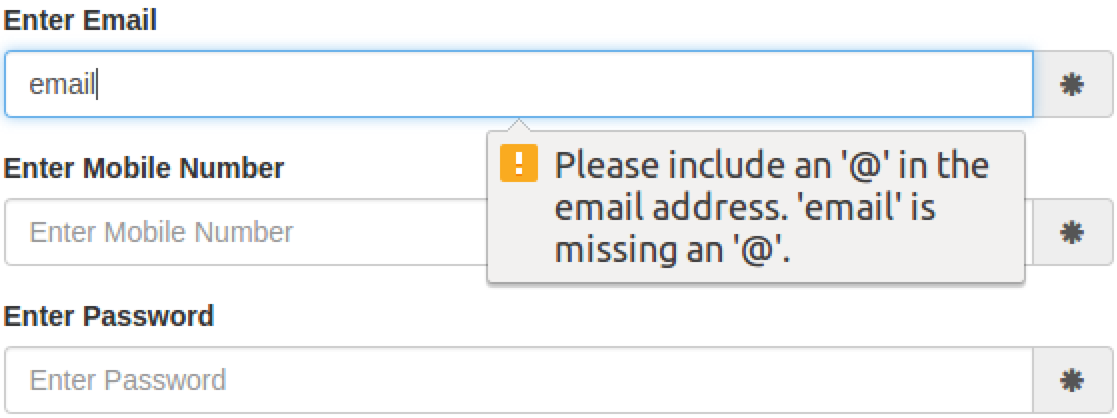
There are a total of 4 classes : **User**, **Task**, **Team** and **Resource**.

They all contain various functions which provide us with all the queries needed for our project. We have called them from inside the JSP's to provide us with data for displaying in our forms and also from our servlets to check, update or retrieve information which the user has asked for.

IV. Integrity Checks

A. Forms

All forms are provided with required field tags in order to prevent null values from passing into the servlets.



Enter Email

email

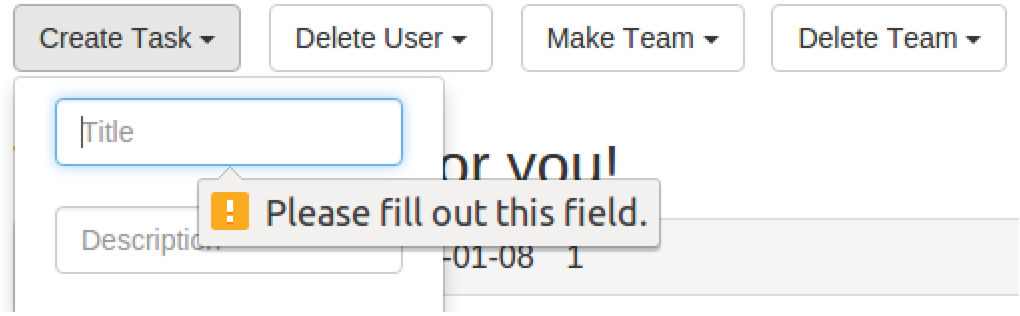
Please include an '@' in the email address. 'email' is missing an '@'.

Enter Mobile Number

Enter Mobile Number

Enter Password

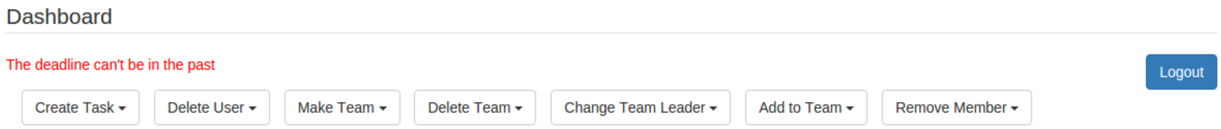
Enter Password



Please fill out this field.

B. Dates

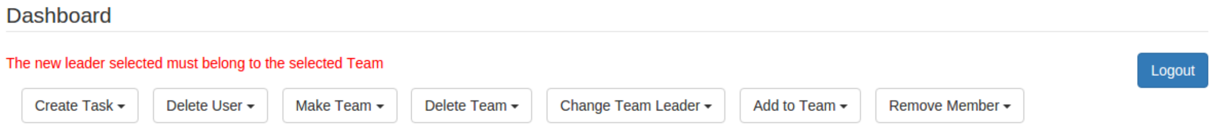
All the date inputs are checked with the present date to prevent the user from changing deadline to some date in the past.



The deadline can't be in the past

C. Team Check

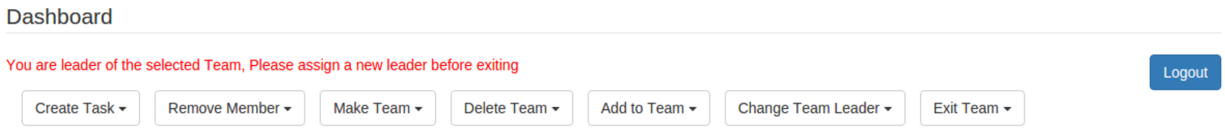
Whenever we select a user and a team we always check whether the user is indeed in the team or not.



The new leader selected must belong to the selected Team

D. Leader Check

Whenever we try to remove a leader or we are the leader and try to exit a group then we get an error message. Also if we are the only member of a group (also the leader) then we can’t exit and will have to delete the group in order to exit. Hence a team can’t exist without any leader (member).



You are leader of the selected Team, Please assign a new leader before exiting

References

http://getbootstrap.com/

http://www.w3schools.com/

Also took JDBC reference from sample JDBCLab given during the lab.