# Birzeit University

# Faculty of Engineering and Information Technology

# Computer Science Department

# Comp 334, Web Applications and Technology

# Project Assignment

## IMPORTANT NOTES

- You should not in any case use any external web development tool. Any try to use such development tool will get you a mark of **zero**.
- Submission is only through your domain that we created for you.
- Since you are submitting to your domain at BZU, it is your responsibility to make sure that your scripts such as file upload, sending email, etc will work properly on BZU server. Make sure that you install your scripts before submission date so that you have some time to fix any issues.

## INTRODUCTION

Online team management applications are essential web tools that help in managing and coordinating projects and daily tasks among team members. Through these applications, tasks are available online and thus, can be accessed, shared and collaborated from anywhere at work, at home, at school, etc.

You are required to create a dynamic web application for team management using HTML, CSS, PHP and MySQL database. The main focus of this web application is to enable team members to assign tasks to and track tasks with other team members. Your application should be simple to use with consistent look and feel among all web pages.

## SYSTEM MAIN FEATURES

Your web application should provide the following main functional features:

1. <u>Team member registration</u>: before team members can join your team, they should register and create their account. After creating their account, they can login and start using the system, i.e., assign, view, update tasks, etc. This also means that only **authenticated** members can use your web application.

2. <u>Task management:</u> through this feature, team members should be able to:
    a. Assign new tasks to themselves or other members. The same task cannot be assigned to more than one member.
    b. View their own tasks and update them, for example changing the **status** of a task (pending→ active→ finished)
    c. When team members login to web site, the system displays their daily tasks. However, they can also search for tasks according to date, status, and assigned members.
    d. Since those tasks will be displayed in a list, every task will have a different background color depending on its status.
    e. A team member can view (read only) other team members tasks.
    f. Since that tasks lists will be displayed in tables, you should provide the functionality so that tasks can be **sorted** according to date, priority and status.

## TEAM MEMBER INFORMATION

The following are the information that the system should record for each team member when registering:

Note all member information are **required.**

1. Name: first and last name
2. Username: this is user email
3. Password.
4. Profile picture.
5. Join date.

## TASK INFORMATION

For each task, the system should record the following information:

1. Task title, **required**
2. Task description: **optional**
3. Start date: **required**
4. Due/end date: **required**
5. Priority 1-3, 1= low and 3 = high priority, default is 1. **optional**
6. Assigned-to member: the member that the task is assigned to, **required.**
7. Assigning member: the name of member how assigned the tasks

## DETAILED SYSTEM REQUIREMENTS

- When member logs into web site, he/she are presented with their daily tasks, that is, the tasks that are **due** on that date.
- Tasks' status can be explained as follows:
  - When tasks is first assigned the system sets it as **pending.**
  - When user starts working on a task, he/she should change its status into **Active.**
  - When user finishes his/her task, he/she changes its status into **Finished.**
  - If task passes its due date and still not finished, the system automatically sets it to **Late.**
- The system should provide a special search page that enables members to search and view tasks according to:
  - Dates
  - Priority
  - Status: view late, active, finished or pending tasks.
  - Member name: view tasks assigned to another members.
    - View late or active etc tasks of other members
- The system should display the profile photo (in small size of course) next to each task for the member how assigned the task.
- Whenever a new tasks is assigned to one member, the system send an email notification for that member notifying him/her that a new task is assigned to him/her.
- All application pages should be authenticated (only logged users can access them)
- For each table displaying a list of records in your web application, the number of displayed records  = 10, and user can view the other record by choosing page2, page3, etc feature at end of each table.
- Your page design should be consistent, containing header, navigation and footer sections. However, these sections are dynamically generated by your application and **not static.**
- The first page to load when user first access your web site is the login page.
- When user makes successful login, the system displays the main page with daily tasks.
- All pages contains:
  - Navigation section on the left which contains:
    - Home page link: this returns the user to the main page which displays the daily tasks.
    - Search page link.
    - Add new task link.
    - Late tasks link: view late tasks
    - Pending tasks link: view pending tasks
    - Active tasks link: view active tasks.
  - Header section containing:

- Your logo
- A name for your web application
- About us page link.
- Contact us page link
- And log out link.
  - o Footer section: smaller logo and copy write text.
  - o Main Content section: this section differs from one page to another, i.e., all pages have same header, navigation and footer sections and differ in the content section. By default when user first login, the content section displays a list of daily tasks, i.e, the tasks for that date.
- pages' structure should be created using CSS, not tables.
- The tasks list, which is a table, should contain task title, sender photo, receiver photo, priority and special background color for its status.
- User can update/his/her profile information.
- The task list should also contain a link to open full task details in another page and enable user to change task data such as its status.
- In the user forms, all fields that are required should have special background color. User CSS classes for this option.
- System should mark input field with special background color if user forgets to enter required field. System should also display appropriate error message near that error input field.
- The pages' authentication mechanism should be done using **sessions**
- Since that tables have a default row limit of **10,** user can customize this option. So for example one user can choose to set the limit=5. So this new value should be stored in **cookies**

## CSS REQUIREMENTS

- Make sure all your CSS rules are in external files
- Make sure you use the CSS rules that are given to you in lecture ONLY!
- The input fields should change background color when get focus.

## BONUS REQUIREMENTS

You will get bonus marks for doing the following:

- In the navigation section, add a number next to "pending", "active", "late", etc to indicate to user how many tasks are in that category.

- Add a feature to calculate who is the best team member, i.e., who is the member that does the best job so far. We leave it to you on figuring out how to calculate this, but is should be convincing.

## SUBMISSION

Submission instructions

The Deadline for the Project Submission due on Tuesday 11/05/2016 At 12:00 noon, you should submit to your domain on Ritaj. The upload ftp service will be closed on Tuesday 11/05/2016 at 15:00. But the deadline is 12:00 noon.

*In addition to your submission to the CShost you have to send me your project to Ritaj by replaying to the this message. Also the deadline is also* 11/05/2016.

What you have to send to Ritaj.

Your project should be compressed, the compressed file should be extracted to Folder name stdID (where std is your name and ID is your student ID). Make sure when your extract the compressed server to the localhost and type the following URL  http://localhost/webprojects/stdID it works and all the links are working fine.

The folder should have the following:-

- All the php scripts, , HTML, CSS, images, ..etc.

- The database Schema exported as SQL, the file name should be"dbschema_yourNumber.sql".

- Index.html