

CPSC 304 Project Cover Page

Milestone #: 1

Date: 07/18/2023

Group Number: 22

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Selin Uz	55505440	z6s4s	ecselinuz2@gmail.com
Guzide Irmak Bayir	11179330	w1q0i	irmakbayir@gmail.com
Huseyin Kutluay Cakadur	47261367	p6n2b	hk.cakadur@hotmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

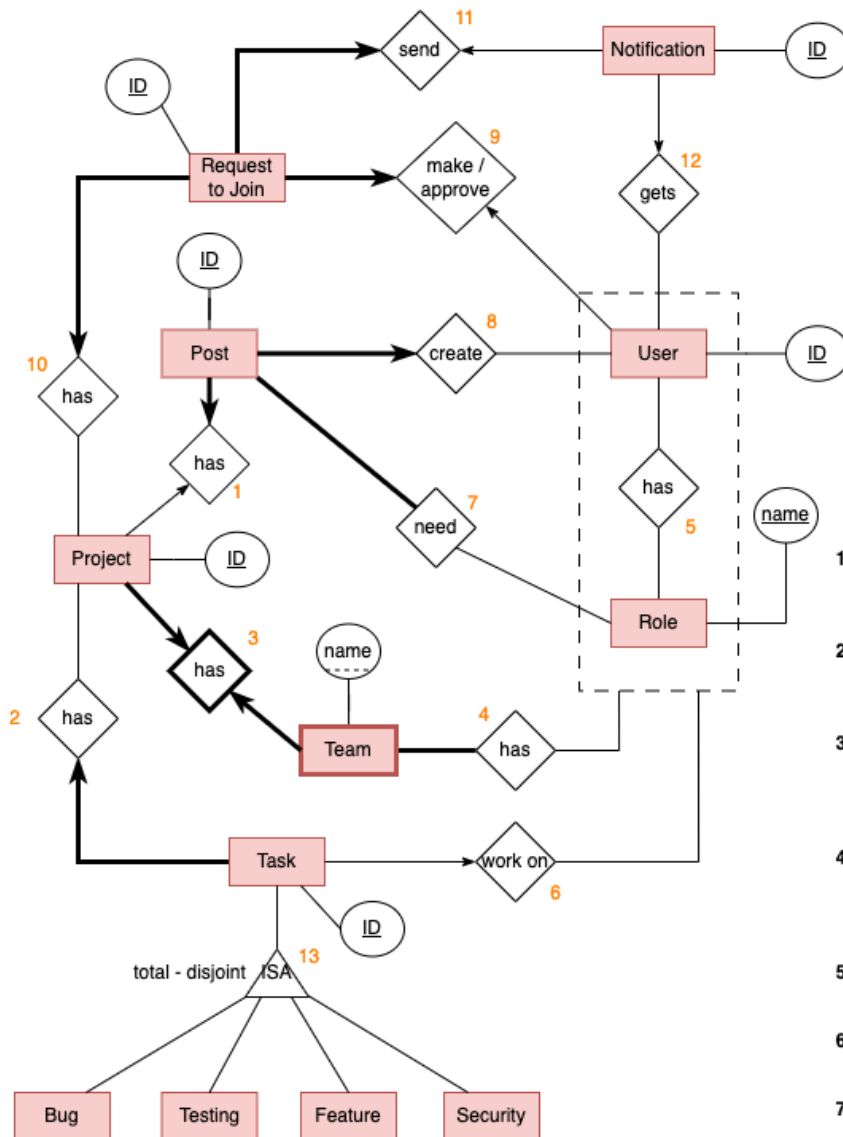
In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia.

Project Description

The project aims to develop a collaborative platform for IT projects. The platform will facilitate the formation of project teams, project idea sharing, and task tracking. The target users are individuals seeking teammates for their project ideas and those looking to join existing projects.

The database will model various aspects of the domain, including user profiles (including their skills, password and bio), project ideas (presented as posts), teams, collaboration requests, assigned tasks and notifications. The application will enable users to manage their profiles and create/share posts. Users can also request to join existing projects through the post, project owners receive a notification and will have the authority to approve or reject these requests. Upon approval, the users will be assigned specific roles and tasks based on their preferences and skills. Additionally, the database will track the progress of projects and maintain relevant task information such as bug fixes, features, tests, and security tasks, along with their respective statuses.

We will be using PHP as the programming language and MySQL as the database management system. As the project progresses and further understanding is gained through tutorials, we might use a framework.



1. Each project should be individually posted but it might not have a post and be saved in drafts. You must have a project to create a post.
2. Projects have zero to many number of tasks to do and each task is associated with only one project. Users are not able to create tasks without projects.
3. A team belongs to a project and cannot be identified without the project because users are not able to create teams without projects. Each project will have a specific team that is working on it.
4. Each team will have a number of users with specific roles, and there must at least be one person on the team for the project to be valid. Users can be in 0 to many teams.
5. Users can have zero to many roles in different projects, and roles are not specific to users.
6. A user with a role works on a task and a task is either unassigned or assigned to one person.
7. Each post must be seeking for at least one role for it to be active, but it can need many as well.
8. A post always has one author, but a user can create zero to many posts as needed.
9. Users can make zero or one request to join a project to a specific role. Each request must be associated with one user. Each request is approved by one user (specifically the project lead).
10. Projects can have zero to many requests to join but each request to join is only associated with only one project.
11. Each request to join sends one notification. A notification doesn't need to be from a request to join but if it is, it should only include one request.
12. Notifications are specific to user and a user can get zero to many notifications.
13. A task must be either a bug, testing, feature or security. Different task types will have different mandatory components to fill.

Entities	Attributes
Post	<u>ID</u> , date
Project	<u>ID</u> , name, description, status
Team	name(partial key)
User	<u>ID</u> , password, bio
Role	<u>name</u>
Task	<u>ID</u> , description, status
- Bug	severity, steps_to_reproduce
- Testing	environment, type, coverage
- Feature	acceptance_criteria
- Security	cvss, rating
Request to Join	<u>ID</u> , date, status, applied_role
Notification	<u>ID</u>