# Small group project self-assessment

**Team name**: Team Leopard

The table below contains a list of epics mentioned in the assignment. For each epic that the team produced features for, add a brief explanation outlining how the user can access and use it. You have been provided with some code that incorporates some of the features you need (though you can extend it). This self-assessment will be used to help the marker figure out how to access your system, so try to keep it clear and concise.

Do review the marking scheme for the small group project to understand how the project is assessed. You are reminded that the assignment is **not** to try and implement every single epic.

|  |  |
| --- | --- |
| **Epic** | **Implementation** |
| Epic 1  The web application should have user authentication and authorisation mechanisms. Users should be able to create accounts, log in, and manage their profiles. | *The features in this cell were provided in the scaffolding project.*   * *A home screen gives users the option to sign-up or login (URL/)* * *Users without account, who are not logged in, can sign up as new users (URL/sign\_up)* * *Users with an account can log in (URL/log\_in)* * *Logged in users can log out (URL/log\_out)* * *Logged in users can change their password (URL/password)* * *Logged in users can change their profile (URL/profile)* * *After completing an account related task, logged in users are redirected to a currently blank dashboard (URL/dashboard)* * *If a not-logged in user tries to access pages requiring login, they are redirected to the login page, and then to the page they were after following successful login.* * *A menu is available on all pages while users are logged in.* |
| No further implementation |
| Epic 2  Users can form teams, and invite team members to their team. | * *Logged in users are given the option to create a new team from the create team button on dashboard (URL/dashboard)* * *The create team page allows users to create a team by entering the team's name & description (URL/team\_creation)* * *Users who have created a team are redirected to a page where they enter a comma separated list to invite members to their team (URL/add\_members/<team\_id>)* * *Users who are leaders can add members from their team page as well (URL/team\_page/<team\_id>)* * *Users who are the leader of their team can remove members from the team page* * *Users who are not the team leader are able to leave team from the team page* * *Users who are the leader of their team can delete their team from the team page* * *Users can accept or decline invites to a team from dashboard* * *Users are redirected to the team page once they accept an invite* |
| Epic 3  Each user can create tasks, assign tasks to other team members, and set due dates. | * *Users can access a button that leads to a task to create a form (URL/create\_task/<team\_id>/) from a team page.* * *Users who are not members of a team cannot create tasks.* * *In the create task form users can create a task with a title, description, due date and priority level. Users are also required to assign these tasks to a member of the team they’re creating it in.* * *Once a task is created any user who the task is assigned to, along with the task creator, can edit the task (URL/edit\_task/<task\_id>/) by clicking on the edit task hyperlink.* * *The creator of the task is able to delete it (URL/delete\_task/<task\_id>/) in the edit task page.* * *All members of the team the task is set in can view its details by clicking on the title or a specific view task hyperlink.* * *If a user who is not part of the team the task is being created for attempts to create, edit, or view one they will be redirected to the dashboard.* |
| Epic 4  Users provide a dashboard to view and manage assigned tasks, and team tasks. | * *Created a dedicated dashboard (URL/dashboard) for users to view and manage assigned tasks and team tasks* * *Designed an intuitive and user-friendly interface to display key information at a glance* * *Users are able to view list of assigned tasks on dashboard* * *Users are able to access a section on the dashboard that displays tasks assigned to their team* * *Implemented a detailed view for each task, displaying relevant information such as due date, priority, and assigned team members* * *Ensure dashboard is responsive and accessible across various devices/screen sizes* |
| Epic 5  Enable users to search, order and filter tasks. These could be based on name, completion status, priority, due date ranges, assigned developer, or team/project. | * *Users are able to search, order and filter tasks by using the search bar, ordering and filtering options on the dashboard, the task search (URL/task\_search) page and on the team page.* * *In all search areas users can search for tasks based on words in their title and descriptions, order tasks in alphabetical order or by closest due date.* * *In the dashboard and task search page users can filter tasks by what team they belong to (if the user is a member of multiple teams).* * *In the team page users can filter tasks based on which user they are assigned to.* |
| Epic 6  Add a priority system for tasks, allowing users to assign priority levels and filter tasks based on priority. Include automated reminders for high-priority tasks nearing their due dates. | * *Upon task creation or while editing, users can label their tasks as either: low, high or medium priority or leave the field blank.* * *If a task is labelled as “high” priority and its due date is in 3 days or less, the task will be highlighted in red* |
| Epic 7  Introduce task dependencies, enabling users to link tasks and establish dependencies between them. Ensure that the application adjusts task timelines based on dependencies. | Not implemented |
| Epic 8  Integrate a time tracking feature for tasks, allowing users to log time spent on each task. Provide summary reports of time spent on tasks over different periods. | Not implemented |
| Epic 9  Implement an activity log that records user actions within the application, providing an audit trail for accountability. | * *Team leaders are able to access the audit log for their team by clicking on the audit log (URL/audit\_log/<team\_id>/) button in the team page once the first task has been creates.* * *The audit log details the 20 most recent activities relating to tasks in the team (task creations, edits, and deletions).* * *The audit log details what has been changed when a task is edited.* |
| Epic 10  Add a notification system (within the constraints of a Django) to alert users about task assignments and upcoming due dates. | Not implemented |
| Epic 11  Introduce gamification elements to motivate users, such as achievement badges, leaderboards, or a points system based on task completion. | Not implemented |