



Create a project management system that holds the following data:

- Support multiple projects. Each project must have a unique name.
- For each project have a backlog with list of tasks.
- Each task should have the following information:
 - o Type: Story or Bug
 - o Title: string
 - o Description: string
 - o Assignee: string
 - o Priority: Low, Normal, High, Critical
 - o Status: To Do, In Progress, Ready for Test, Done
 - o Estimate: number
 - o Created At: Date and Time when the item was added to the project
 - o <any other properties that you think are relevant for each task>

Create a desktop Electron-based application that has the following functionalities:

1. Show all the available projects and allow the user to switch between them.
2. Show all the tasks in the currently selected project in a list. Each task in the list must have the title and the status shown.
3. When a task in the list is selected, open it in a new page and show all the available information for it.
4. In the page where concrete task is shown, allow editing all the information. Use UI elements suitable for each property. Also, allow getting back to the main page of the project. When an information of the specific task is changed, it should be persisted in a way, so the next time the application is opened, it will show the last changes.
5. Have a board where you have all the statuses of the items as columns and each task is placed in the column based on its status.
6. Add search functionality in the page with all tasks list, which allows the user to search in the title and description of the tasks. The list must show only the tasks that match the provided search criteria.
7. Add preview functionality in the page with all tasks list – instead of opening each item in a separate page, have an easy way to preview all the information of an item and edit it without changing the current page.
8. Add history for each of the tasks – the history must contain all the changes made to the tasks and the date and time when the change was applied. The history must be visible when you open concrete task.
9. Add logo for each of the projects. Allow file upload to set the logo of concrete project.
10. Add 3 unit tests for the functionalities you think are most important in the app

Technical requirements:

- Use a front-end framework to build the UI. The UI must be responsive.

- You must use Electron and the app should be able to run on multiple OSes (Windows, macOS, Linux).
- You must have a server, separate from the Electron application, and use it to load and persist the data.
- The server should use another technology, preferably .NET
- The application must be available in GitHub or GitLab repository

Preferably, you should not use database in your implementation.

If you are unable to implement all the functionalities, add a short description explaining how you would implement the ones for which you were unable to finish.