Dev Day 23 Web Development Competition Round 1

Systems Limited is a Pakistani public technology company, involved in mortgage, apparel, retail and BPO services.In Systems Limited,There are multiple teams working on multiple projects. Therefore to manage and track all software development life cycle processes, System Limited wants to automate all these processes. For this purpose Systems Limited has hired you as a full stack web developer to create “Systems Project Management (SPM)” to solve this problem.

SPM consists of 2 types of users i) Manager ii) Developers.The functionalities of this system is different and it depends on the role of users.Below mentioned are the requirements of SPM given by Systems Limited.

Only create those functionalities and buttons which are mentioned in the documents,you can skip any other UI modules which are shown in the screenshots and not mentioned.
There are multiple tables of database Also, there are relations between them be careful while designing it , it depends on your understanding (you can use relational/non-relational databases but sql is recommended)

Basic Features:

Create an interactive UI and Database that covers the below requirements.

1. Create a sign up page for users which takes First Name, Last Name, Email, Password for creating an account and save it to Database.
2. Create a login page for user which takes Email,Password and authenticate it from Database to login their profiles
3. User can create the project with the minimalist details such as project name and a unique project key that should be auto generated and associated with the project. This user will be called administrator user for the created project.
4. User can add other users to the project with respect to their role by sending them project key through email. It’s mandatory to check that the recipient user should be registered in the system already otherwise email should not be send.
5. User can join a project by providing a project key.
6. Administrator user can add tasks in the project by providing the minimalist details like task name, nature of task (i.e. task or bug), start date and end date of task.
7. After creating the task, other members of the same project will be able to view the tasks.
8. Users can create an organization in SPM.
9. Work items are the primary means of the activities during a project which are handled by different team such as creating sub modules,Requirement gathering.
10. Work items can be created by any member of the project team and can be assigned to any other team member.
11. Work items can be organized into boards and provide a visual representation of the work items and their progress.
12. Work items can be easily moved between different columns on the board using drag and drop. For example, a

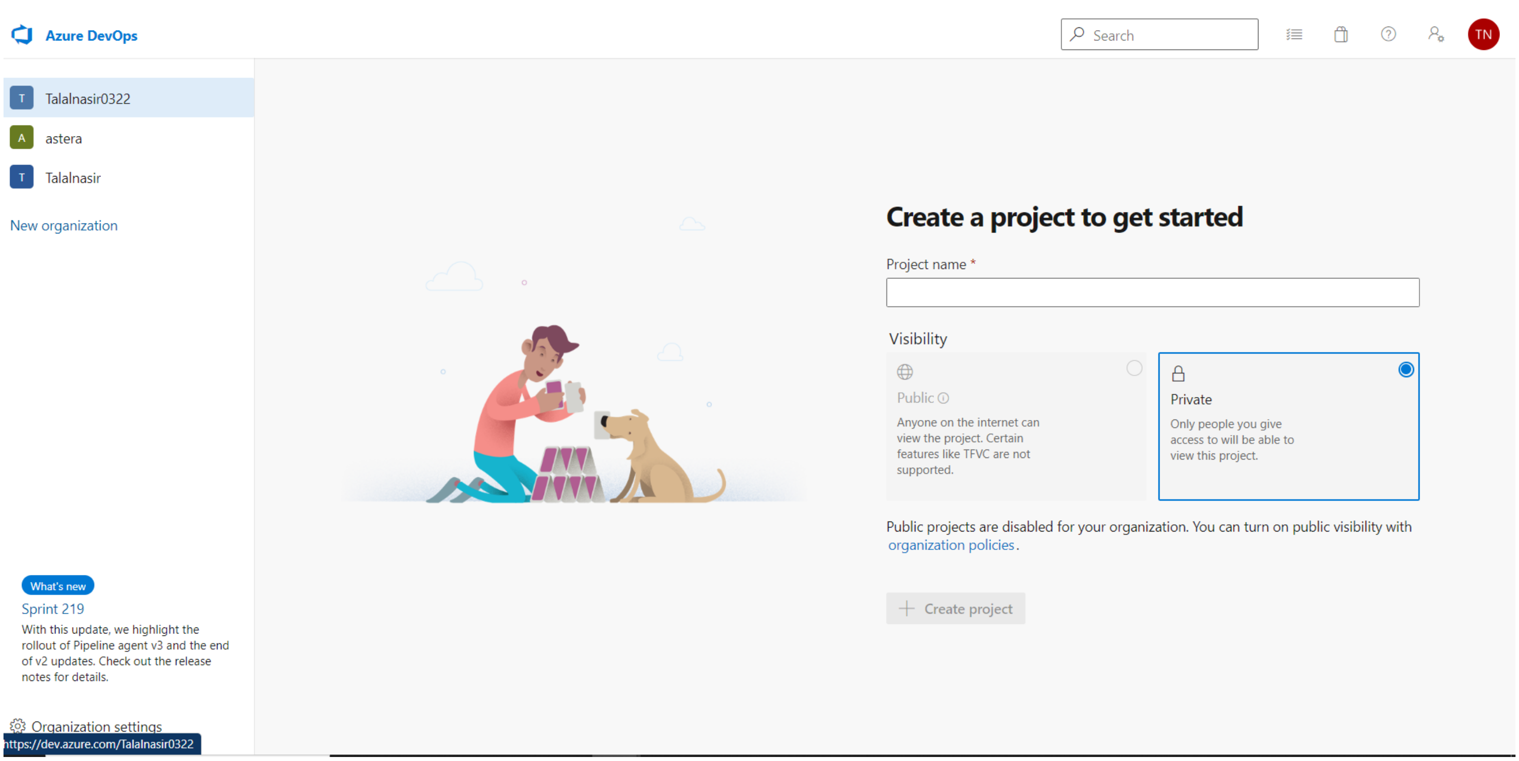
work item can be moved from the "To Do" column to the "In Progress" column when work has started on it.

13. Work items can be viewed in a backlog, which is a prioritized list of work items that need to be completed. Backlogs can be customized to include various filters and sorting options.
14. Initially, the task has a "To Do" status, but the user who has been assigned this task or the administrator user can change its status to "In Progress" (means work is going on this task) "Open" (means work is not going on this task) "Done" (means the task is completed from the assigned user) "Closed" (means the administrator user has confirmed that the task is completed).
15. If the task status is changed to "Closed", no one except the administrative user can change its status back.
16. The administrative user can remove any other member from the project.
17. The administrative user should be able to view a report where they should be shown stats with respect to projects (i.e., hours spent on projects, tasks generated), tasks (i.e., hours spent on tasks, people assigned, etc.).
18. Upon successfully logging in, the user should be displayed all the projects they are working on and all the projects they worked on previously in tile format.

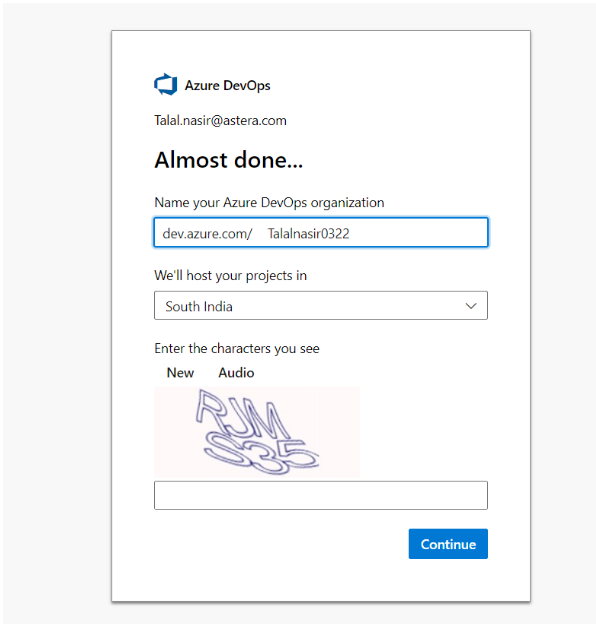
Home Page:

Example Screen Of Screen After Logging in is given Below:Fig(1)

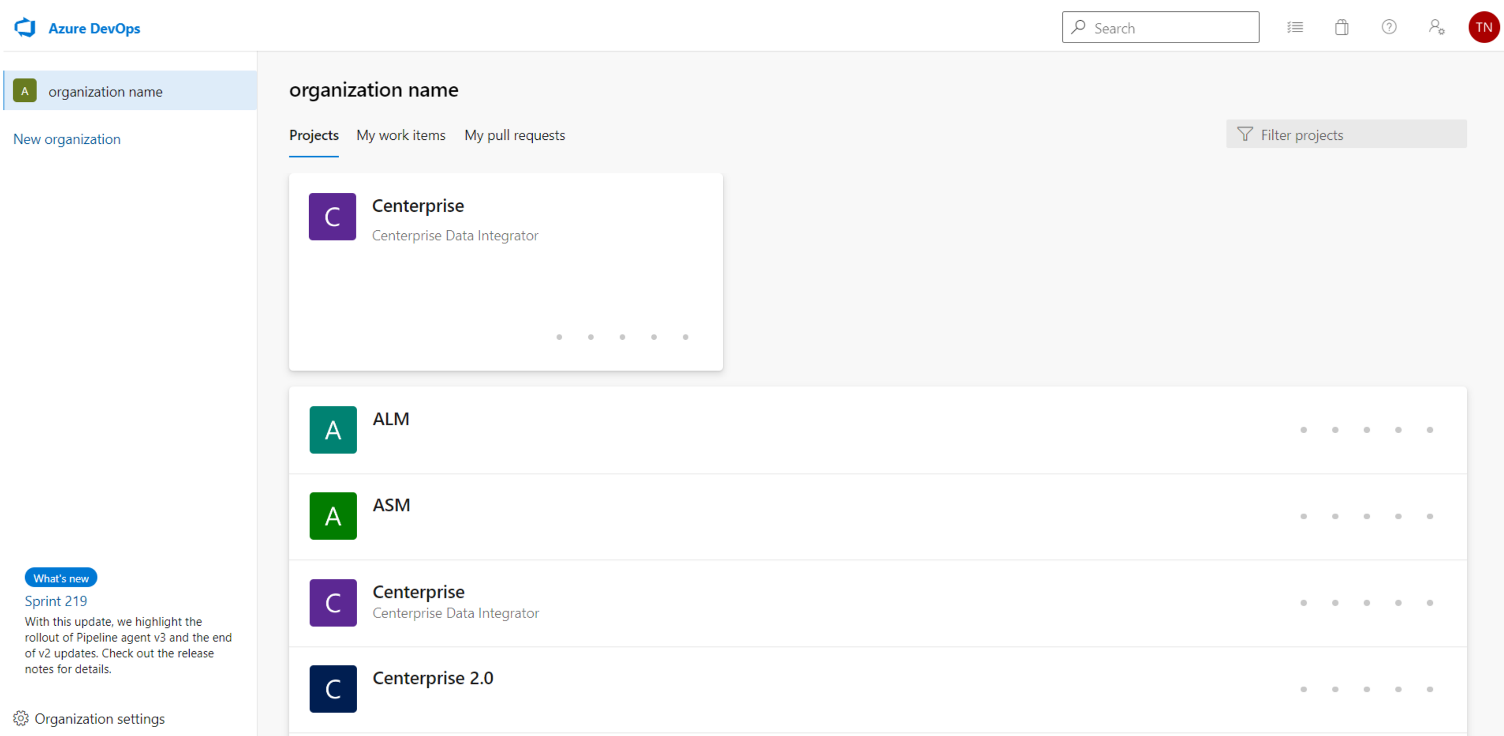
1) Create a sidebar which will show organization names.
2) Sidebar should have a “Creating new organization” Button to add new organizations.It will ask for Organization name,country and save it to the database along with user name.
3) Create a “Create a new project” button(modal) for the users that want to create a new project.which will ask for project name,project details and save it to the database.



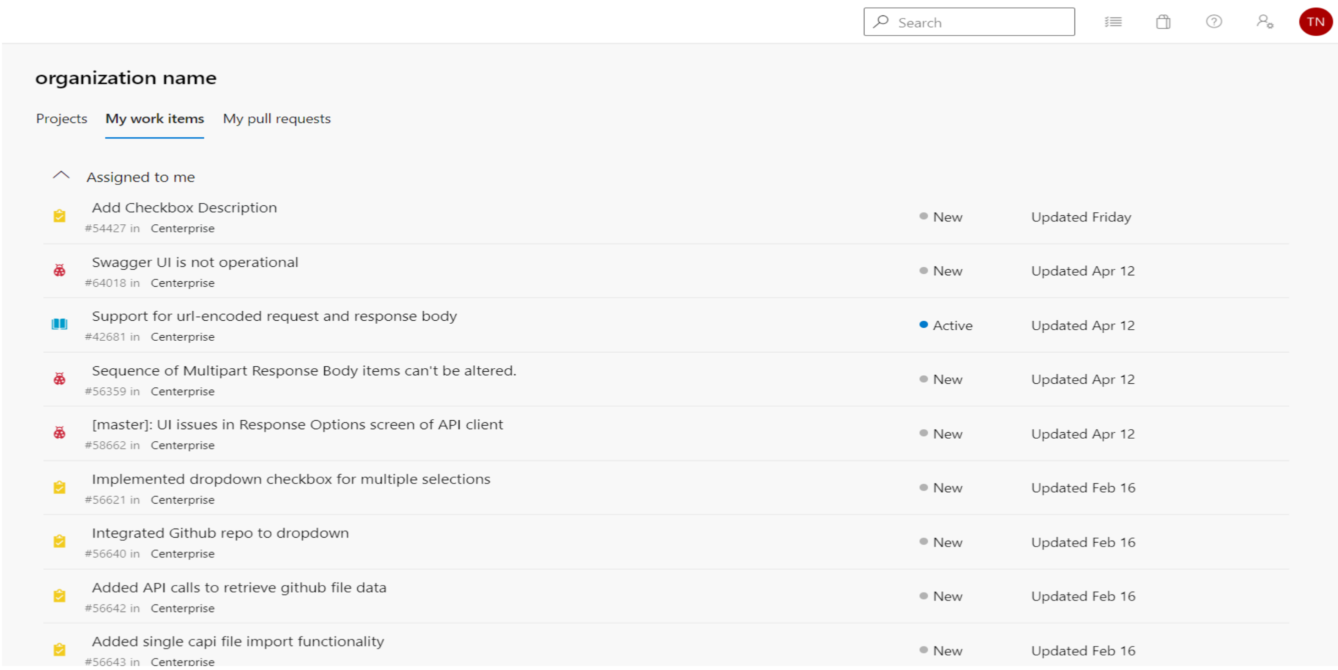
Creating an organization will be similar to this:



Fig(1.1) after creating project interface should be similar to this



Below Screenshots (Fig 1.2) shows the "My Work Item" tab where details of all working activities and their status assigned to the user are shown. Skip "My pull Request Button"

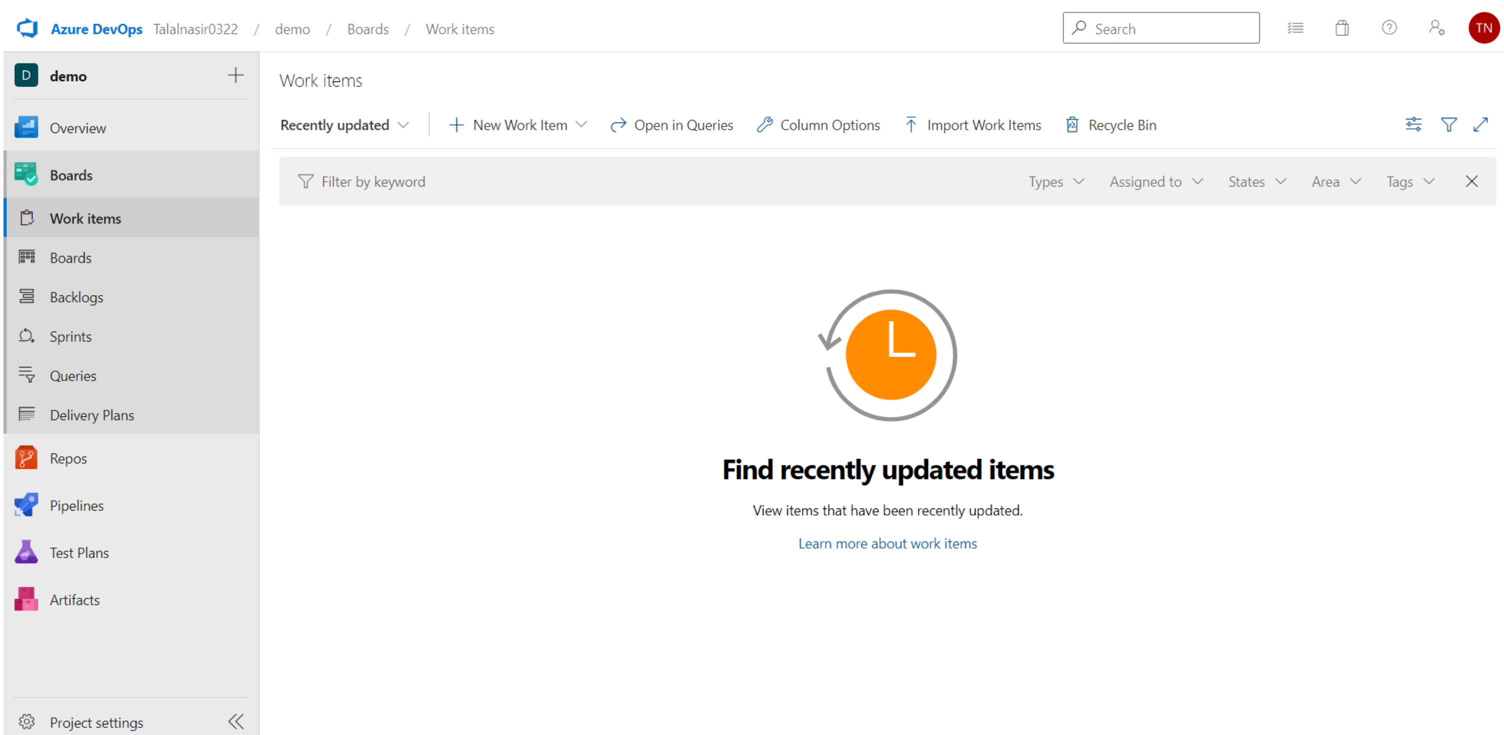

Search
TN
organization name
Projects
My work items
My pull requests
Assigned to me
Add Checkbox Description
#54427 in Centerprise
New
Updated Friday
Swagger UI is not operational
#64018 in Centerprise
New
Updated Apr 12
Support for url-encoded request and response body
#42681 in Centerprise
Active
Updated Apr 12
Sequence of Multipart Response Body items can't be altered.
#56359 in Centerprise
New
Updated Apr 12
[master]: UI issues in Response Options screen of API client
#58662 in Centerprise
New
Updated Apr 12
Implemented dropdown checkbox for multiple selections
#56621 in Centerprise
New
Updated Feb 16
Integrated Github repo to dropdown
#56640 in Centerprise
New
Updated Feb 16
Added API calls to retrieve github file data
#56642 in Centerprise
New
Updated Feb 16
Added single capi file import functionality
#56643 in Centerprise
New
Updated Feb 16

Project Page:

Fig(1.1) Explains home page, Now see “Centerprise” which is a project of an organization.On clicking “Centerprise” the page is directed to a new page similar to below screenshot (Fig 1.3) in which there is a sidebar having buttons of:

1) Overview: For seeing details of the project
2) Work Items: Each project can have multiple work items e.g there are multiple phases on project i.e Requirement Gathering, Developments, Development of sub parts,Integration and Devops, Testing e.t.c so these will be work items and they are assigned to a person. It should display ID,Title,Assign To,Status and Path of Work Item.(Fig 1.3)
3) Backlogs: Backlogs are activities that are created by the user, but these activities are not assigned to another person.
4) Delivery Items:

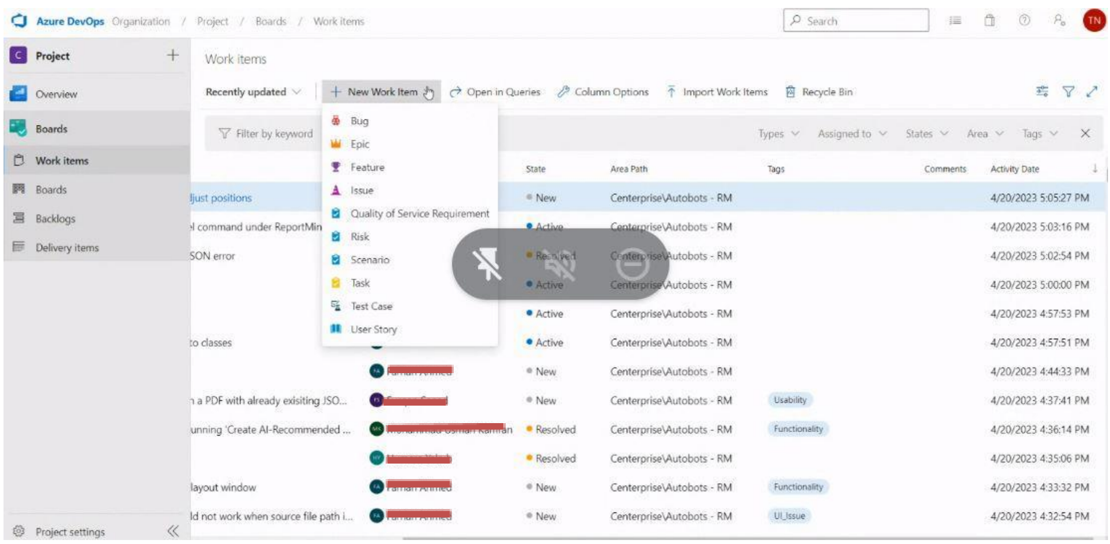
On clicking work item interface should be similar to this:



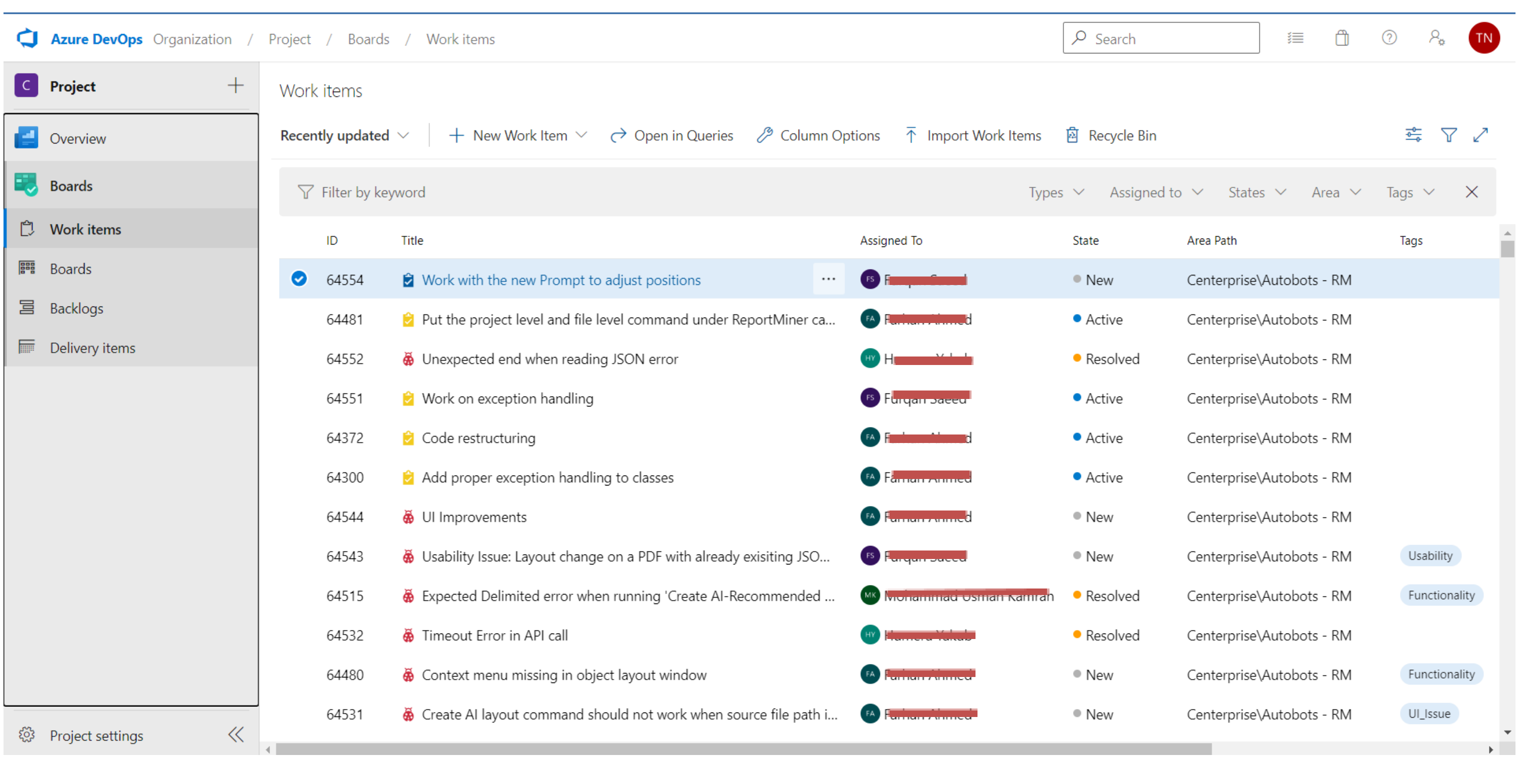
Create WorkItems:

1) "Creating Work Item" As mentioned above each project can have multiple work items for these work items there should be a "Create Work Item Button"(modal) where a user can give Title, Assign To, Status , type i.e Testing,Development etc and phase (Doing,Done) selection which will help in "Board" option, you can take an example from below fig(1.4)
2) Add "Delete Work Item" to delete work item.

   Note : Skip extra buttons e.g open in queue,column options,Import work items button.

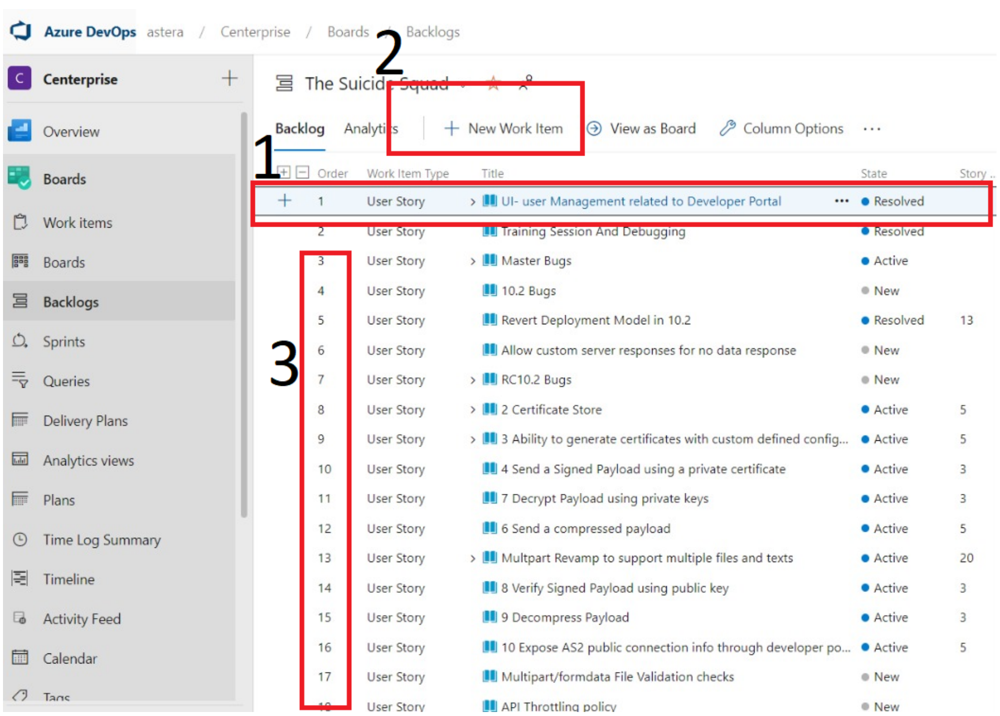


adding work items is given below



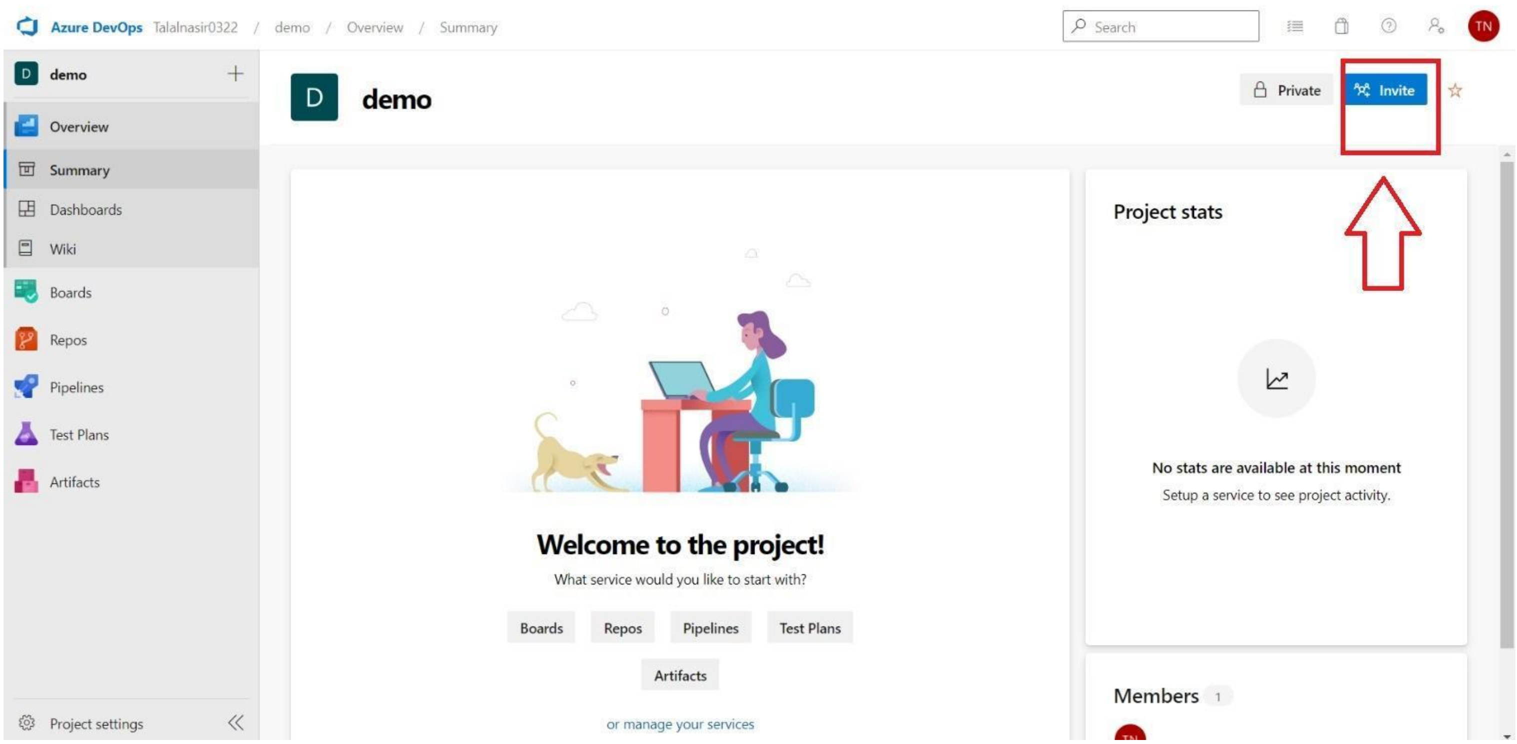
Product Backlog:

1. Develop a user story module that allows users to create and manage user stories in the backlog as shown in the screenshot below.
2. User stories should be created using the option "new work item".
3. User stories should be queued as they are added in the backlog where the user can update the status as per need.
4. When clicked on a particular user story the planning window should open which displays the timeline for the selected user story.



Add People To Project:

The last functionality of SPM is to invite people to project by using email. There will be a button "Invite People" on clicking this button a modal will open which asks for email, after submitting it will search that email in the database and add him to the project.

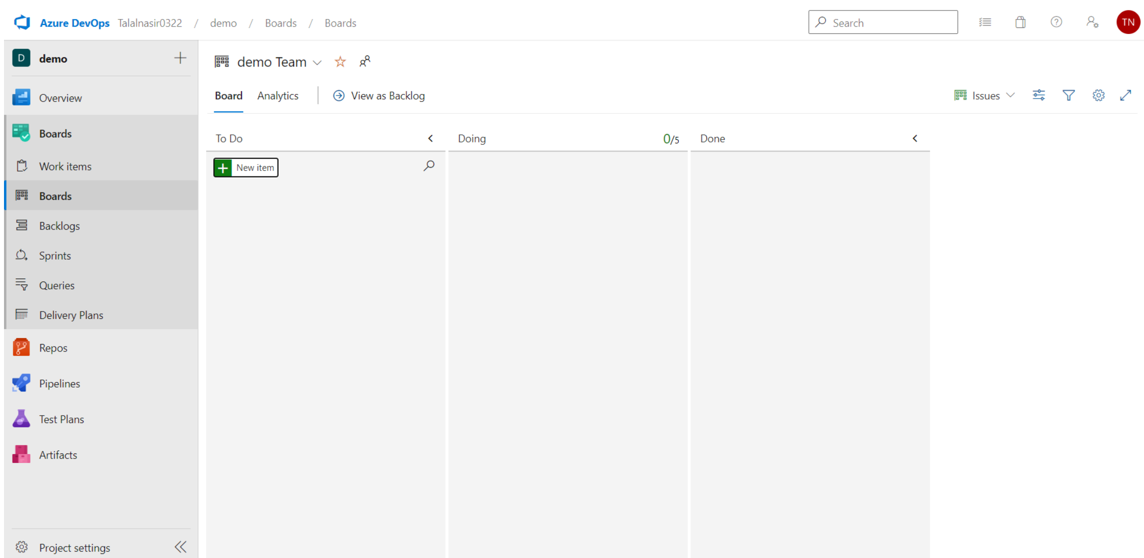


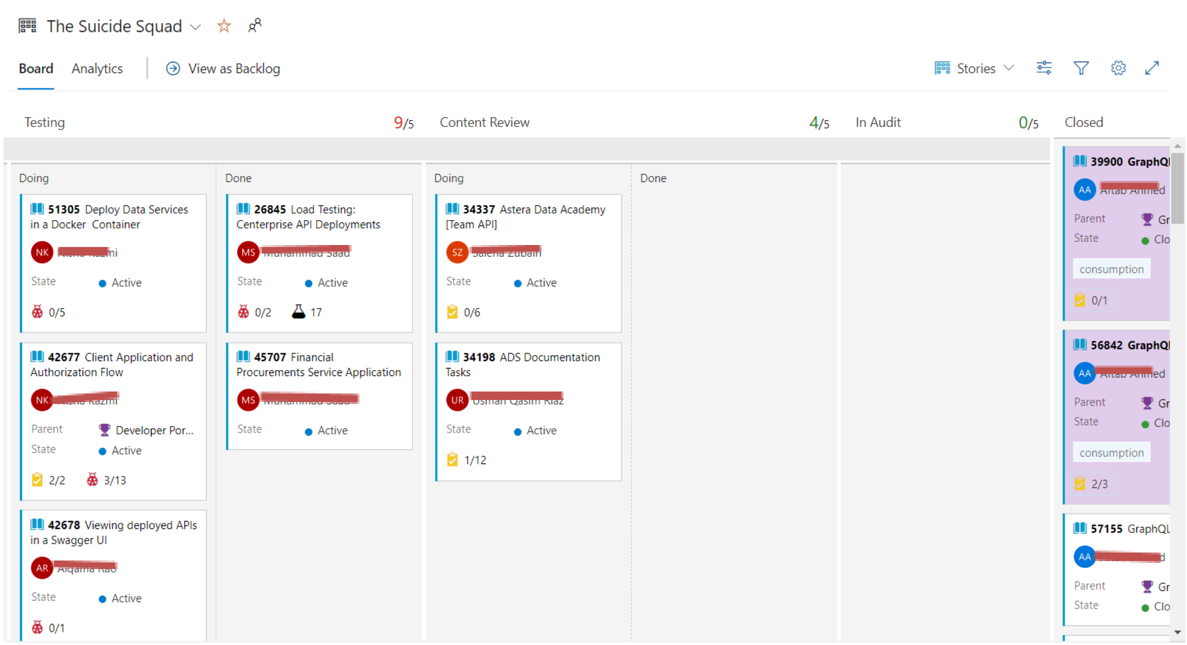
Note: These pictures are just for the reference and do not depict the exact requirements/user stories mentioned above. You can have a different UI/UX as per the requirements.

Boards Tab (Bonus) :

Another functionality is "Board" check (Fig 1.3) sidebar ,

1) Create a Drag and Drop, Board interface in which ToDo,Doing,Done Table is present.
2) ToDo column has a "New Item" button, on clicking a modal will be open which will give all work items of the project.
3) Users have to select from these work items on toDo,Doing,Done phase.
4) Board interface should be like (Fig 1.6) in Doing and Done phases has workitem of that project.Doing activities has workitem of whose phase in Doing while creating work item.
5) Also, Done activities have work items of those whose phase is Done.
6) Users should be able to drag and drop from doing and doing phases and vice versa.





Important Notes:

- You are not allowed to use any cms, wordpress, wix or any other site builder
- You are not allowed to use paid version of any library/framework/script
- In case of any cheating or plagiarism, team will be disqualified immediately
- You can email any queries to k191105@nu.edu.pk or k200346@nu.edu.pk with the Subject Name as "DevDay 23 Web Dev Competition".

Submission:

- You have to create two folders. One for the frontend and one for the backend. Put your code in these folders respectively.
- Push these folders into a single repository over GitHub
- Make a 5 minute video of your project in which you will be presenting the application you developed. You need to present all features that you implemented into your application and explain any choices or decisions that you made during the development.
- Once you have done all the above things, submit the GitHub repository URL, and video URL into the form provided to you on Google Classroom before the deadline.
- Make sure all links being provided by you are publicly accessible and not broken.

Duration: 48 Hours

Start Time: 28-04-2023 6 pm

Finish Time: 30-04-2023 6 pm

Teams have the creative freedom and they will be judged on the following criteria:

1. Requirements are satisfied
2. Responsiveness
3. Security
4. Design patterns
5. Their Own creativity
6. Output