Fabio Percivaldi

Yet Another Issue Tracker

1 agosto 2022

Overview

The goal of the project is to implement an Issue Tracker (Jira, Trello, etc) that implement the scrum framework and has some analytics capabilities (sprint performance, metrcs etc).

Timeline and Roadmap

The deadline for this project is february 2023, a first MVP must be released by the end of October containing at least the following features:

MVP Features

Using the Front End a User should be able to visualize the active sprint with his tickets. Tickets should be divided in column that can be customi

- Security
  - API are exposed using OAuth2 flow for communication between FE and BE
- Project
  - CRUD operation
  - project has default “To Do”, “In Progress”, “Done” column
- Sprint
  - CRUD operation
  - a sprint can have from 0 to N item
- Epic
  - CRUD operation
  - an epic can be assigned from 0 to N item
- Item
  - create new item inside a sprint or inside a generic backlog
  - assign an Epic to an Item
  - define the type of items (bug, task)
  - add description
- Board

    - visualize the board with the given columns for the project
    - visualize the Item inside the columns of the board
    - Item can be moved from a column to another
    - Item can be opened and modified

Optional Features to be shipped after MVP

- Security
    - the site is no longer available without previous registration/login
    - user can log in using a third party authentication mechanism
- Project
    - add custom column to a board
- Item
    - assign the item to a user
    - add attachments (files management)
    - add comments
    - add estimation for the Item
- Users
    - user receive notification when a new ticket is assigned
    - items can be filtered for a specific assignee
    - there is a viewer role that can only view item and add comments
    - there is an admin role that can create item and change the status
- Analytics
    - number of total ticket and closed ticket
    - average time for a ticket to be completed
    - average number of ticket closed per sprint

Technologies Requirements

Front End will be developed in **React**

Back End will be developed in **GO**

The Back End should be database independant, we only know that a **SQL** database will be used.

The Architecture will be hosted on **AWS** using EKS

The Architecture must be scalable and resilient

Notification must be handled with an async flow

Infrastructure must be deployable in different environment using IaC (**Terraform**)