Backend – Rest API Server

**Classes:**

Issue -

* Variables
  + ID
  + Title
  + Summary
  + IssueLevel
  + CreatedBy
  + CreatedDateTime
* Functions
  + Initialise
  + Return\_Details
  + Update\_Details

**Variables:**

Database = Array of Class Issue

**Methods:**

**POST – To create new issue.**

Request body

{

‘title’ : string (title),

‘summary’: string (summary),

‘issueLevel’: float (0-5)

‘createdBy’: string (username),

‘createdDateTime’: int(time since epoch)

}

Function would check the current number of issues and create a new unique ID, create an new Issue from class using the values from the request body and the ID. This issue would then be added to the database.

Logs

This method would return the issue’s ID.

**GET – To read an existing issue.**

Query parameter

Id = int (id)

Function would look up an issue from the database using the query ID return the entire issue class as a JSON.

If Issue with ID does not exist return Error.

**PUT – To update an existing issue.**

Request body

{

‘id: int(id)

‘title’ : string (title),

‘summary’: string (summary),

‘issueLevel’: float (0-5)

‘createdBy’: string (username),

‘createdDateTime’: int(time since epoch)

}

Function to check the ID of the request against the database.

If the ID exists, update the Issue details with the new details from the request body. Return success message.

If the ID does not exist, return error.

**DELETE – To remove an existing issue.**

Query parameter

Id = int (id)

Checks if Issue with ID exists in database.

If Issue exists – replace reference with Null – return error.

If Issue does not exist – return error.