Service Layer

BoardService

-ilog:log

- +LimitColumn(userEmail,creatorEmail, boardName, columnOrdinal, limit, userController):Response
- + GetColumnLimit(userEmail,creatorEmail, boardName, columnOrdinal, userController):Response<int>
- +GetColumnName(userEmail,creatorEmail, boardName, columnOrdinal, boardController):Response<string>
- + AddTask(userEmail,creatorEmail, boardName, title, description, dueDate, boardController):Response<Task>
- +UpdateTaskDueDate(userEmail,creatorEmail, boardName, columnOrdinal, taskId, dueDate, userController):Response
- +UpdateTaskTitle(userEmail,creatorEmail, boardName, columnOrdinal, taskId, title, userController):Response
- +UpdateTaskDescription(userEmail,creatorEmail, boardName, columnOrdinal, taskId, description, userController):Response
- +AdvanceTask(userEmail,creatorEmail, boardName, columnOrdinal, taskId, boardController):Response
- +GetColumn(userEmail, creatorEmail, boardName, columnOrdinal, userController):Response<lList<Task>>
- +AddBoard(userEmail,creatorEmail, name, userController):Response
- +RemoveBoard(userEmail,creatorEmail, name, userController):Response
- + InProgressTasks(userEmail,creatorEmail, userController):Response<IList<Task>>

+ load Data (Board Controller, user Controller): Response

- +deletData(BoardController,userController):Response
- + joinBoard(userEmail,creatorEmail, boardName, boardController):Response
- + getBoardsName(userEmail, boardController):Response<IList<string>>
- + assignTask(userEmail.creatorEmail.boardName.boardController. columnOridinal. taskId.enailAssignee):Response

Service

-BoardController:BoardController

- -UserController:UserController
- -BoardService:BoardService
- -UserService:UserService
- -ilog:log
- +LoadData():Response
- +DeleteData():Response
- + Login(email, password):Response<User>
- +Register(userEmail, password):Response
- +Logout(email):Response
- +LimitColumn(userEmail,creatorEmail, boardName, columnOrdinal, limit):Response
- +GetColumnLimit(userEmail,creatorEmail, boardName, columnOrdinal):Response<int>
- + Get Column Name (user Email, creator Email, board Name, column Ordinal): Response < string > 100 cm of the column Name (user Email, creator Email, board Name, column Ordinal): Response < 100 cm of the column Name (user Email, creator Email, board Name, column Ordinal): Response < 100 cm of the column Name (user Email, creator Email, board Name, column Ordinal): Response < 100 cm of the column Name (user Email, creator Email, board Name, column Ordinal): Response < 100 cm of the column Name (user Email, creator Email, board Name, column Name (user Email, creator Emai
- +AddTask(userEmail,creatorEmail, boardName, title, description, dueDate):Response<Task>
- +UpdateTaskDueDate(userEmail,creatorEmail, boardName, columnOrdinal, taskId, dueDate):Response
- +UpdateTaskTitle(userEmail,creatorEmail, boardName, columnOrdinal, taskId, title):Response
- +UpdateTaskDescription(userEmail,creatorEmail, boardName, columnOrdinal, taskId, description)
- +AdvanceTask(userEmail,creatorEmail, boardName, columnOrdinal, taskId):Response
- +AddBoard(userEmail, name):Response
- +RemoveBoard(userEmail,creatorEmail, name):Response
- +Response<IList<Task>> InProgressTasks(userEmail)
- + join Board (user Email, creator Email, board Name): Response
- + assign Task (user Email, creator Email, board Name, column Ordinal, task Id, email Assignee): Response to the contract of the contract of
- +getBoardName(userEmail):Response <iList<string>>

ResponseT

- +FromError(msg):Response<T>
- +FromValue(T value):Response<T>
- +base(msg):Response(value, msg)

Response

- +ErrorOccured:BOOLEAN
- +Response(string msg)
- +Response()

UserService

-ilog:log

- + Register(email, password, userController):Response
- + Login(email, password, userController):Response<User>
- + Logout(email, userController):Response
- + deleteData(userController):Response
- + loadData(userController):Response

Labe



