Service Layer

UserService

- uf: UserFacade
- + ChangePassword(email: string, oldP: string, newP: string): string
- + ChangeEmail(oldEmail: string, passworrd: string, newEmail: string): string
- + Register(email: string, password: string): string
- + Login(email: string, password: string): string
- + Logout(email: string): string

Response

- + value: string
- + error: string

TaskService

- uf: UserFacade
- bf: BoardFacade
- + AddTask(email: string, boardName: string, title: string, description: string, due_date: dateTime): string
- + GetTask(email: string, boardName: string ,taskld: int): string
- + DeleteTask(email: string boardName: string, taskId: int): string
- + AdvanceTask(email: string, boardName: string, taskId: int, columnOrdinal: int): string
- + UpdateTaskeDueDate(email: string, boardName: string, taskId: int, newDueDate: DateTime): string
- + UpdateTaskTitle(email: string, boardName: string, taskId: int, newTilte: string): string
- + UpdateTaskDescription(email: string, boardName: string, taskId: int, newDescription: string): string
- + InProgressTasks(email:string): string

BoardService

- bf: BoardFacade
- uf: UserFacade
- column: List<List<Task>>
- + CreateBoard(email: string, boardName:string): string
- + DeleteBoard(email: string boardName: string): string
- + GetBoard(email: string boardName: string): Board
- + LimitNumTask(email: string, boardName: string,int columnOrdinal, limit: int): string
- + GetColumnLimit(email: string, boardName: string, columnOrdinal: int): string
- + GetColumnName(email: string, boardName: string, columnOrdinal: int): string

BoardToSend

- +BoardName: string
- columns: List<List<Task>>
- limitTaskInColumn2: int
- limitTaskInColumn1: int
- limitTaskInColumn0: int

UserToSend

+ Email: string

TaskToSend

+ Taskld: int

Business Layer UserFacade - Us: UserController - users: Dictionary<string, User> - history: IUserHistory - addeUser: bool - LoadData():void - DeleteData():void IsValidPassword(password: string): bool ~ IsValidEmail(password: string): bool ~ IsEmailExists(email: string): bool ~ getUser(email: string): bool + ChangeEmail(oldEmail: string, passworrd: string, newEmail: string): bool ~ ChangePassword(email: string, oldP: string, newP: string): bool ~ Register(email: string, password: string): User ~ IsLogin(email: string, password: string): User ~ Login(email: string, password: string): User ~ Logout(email: string): bool User - udto: UserDTO - email: string - password: string - loggedIn: bool ~ ChangePassword(oldP: string, newP: string): void ~ ChangeEmail(oldP: string, newP: string): void UserHistory - TasksInProgress: Dictionary<string, List<ITask>> - BoardsOfUsers: Dictionary<string, List<int>> - GetTasks(string email):List<ITask> - AddTaskInProgress(string email, ITask task): void - AddBoardToOwner(string email, Board board): void - AddNewEmailToBoards(string email): void

- DeleteTask(string email,int id): void

- DeleteData(): void

BoardFacade

- boards: Dictionary<string, Board>
- uf: UserFacade
- uh: UserHistory
- numberOfBoardTotal: int
- numberOfTaskTotal: int
- Et: ExceptionThrower
- Bc: BoardController
- + LoadData(): viod
- + DeleteData: void
- + GetBoard(email: string, boardName:string): Board
- + DeleteBoard(email: string boardName: string): string
- + CreateBoard(email: string, boardName:string): Board
- + GetUserBoards(email: string): List<int>
- + JoinBoard(email: string, boardId: int): void
- + GetBoard(boardId: int): Board
- + LeaveBoard(email: string, boardId: int): void
- + GetBoardName(boardId: int): string
- + TransferOwnership(string currentOwnerEmail, string newOwnerEmail, string boardName): string
- + AddTask(due_date: dateTime, title: string, description: string): Task
- + AssignTask(due date: dateTime, title: string, description: string): Task
- + DeleteTask(email: string boardName: string, taskId: int): string
- + GetTask(email: string, boardName: string,taskId: int): Task
- + LimitColumn(email: string, boardName: string,taskId: int): Task
- + GetColumnLimit(email: string, boardName: string, columnOrdinal: int): string
- ~ LimitNumTask(email: string, boardName: string, columnOrdinal: int, limitTaskInColumn: int): string
- + GetColumn(email: string, boardName: string, columnOrdinal: int): string
- + GetColumnName(email: string, boardName: string, columnOrdinal: int): string
- + AdvanceTask(email: string, boardName: string, taskId: int, columnOrdinal: int): string
- GetBoardsOfUser(string email):List<ITask>
- + GetInProgressList(email: string): List<ITask>
- ~ UpdateTaskTitle(newTilte: string, ,taskId: int): ITask
- ~ UpdateTaskDueDate(newDueDate: DateTime ,taskId: int): ITask
- ~ UpdateTaskDescription(newDescription: string, ,taskId: int): ITask
- + GetBoard(email: string boardName: string): Board
- ~ GetInProgressList(email: string): List<Task>



