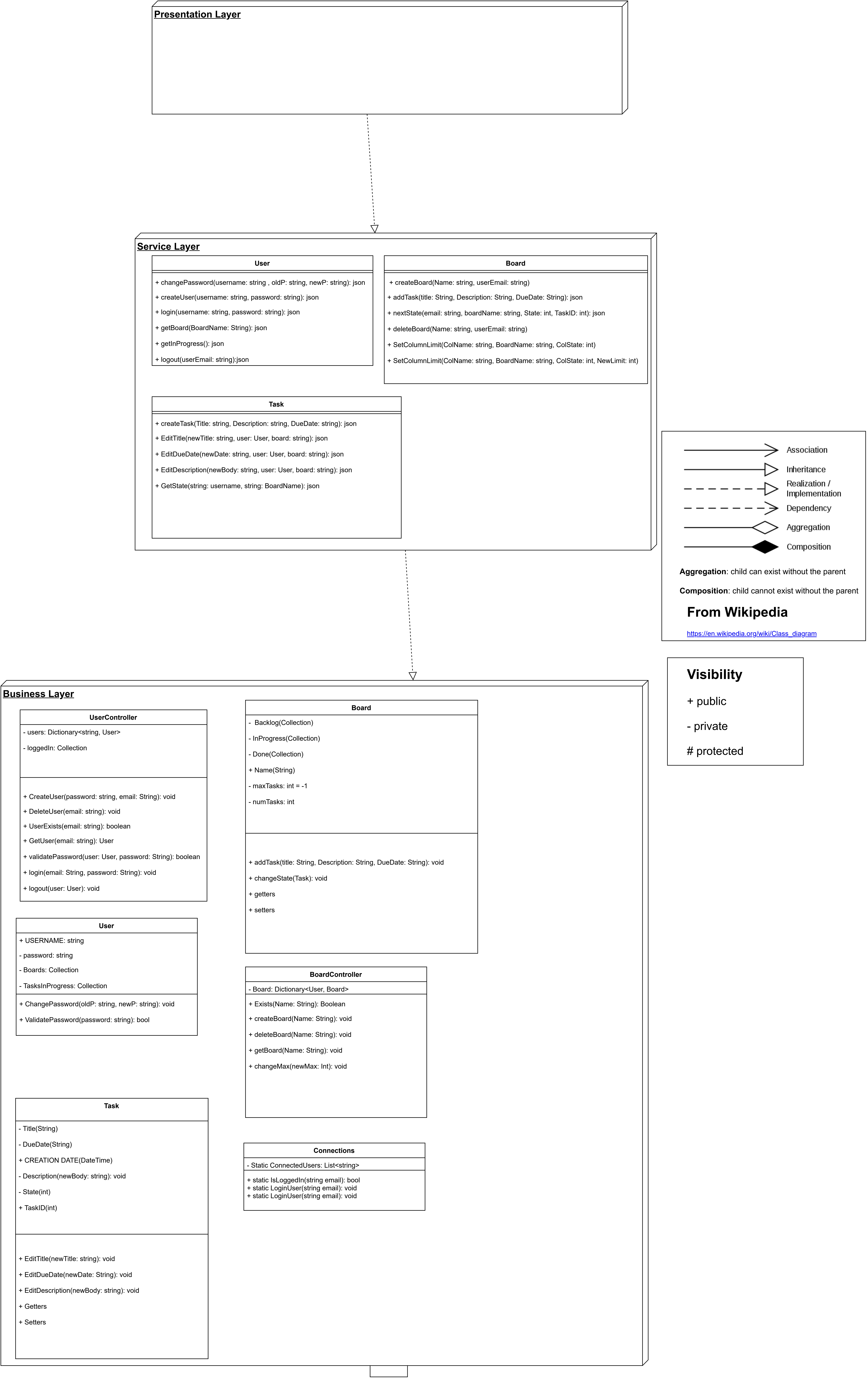

Presentation Layer

Service Layer

User

+ changePassword(username: string , oldP: string, newP: string): json
+ createUser(username: string, password: string): json
+ login(username: string, password: string): json
+ getBoard(BoardName: String): json
+ getInProgress(): json
+ logout(userEmail: string):json

Board

+ createBoard(Name: string, userEmail: string)
+ addTask(title: String, Description: String, DueDate: String): json
+ nextState(email: string, boardName: string, State: int, TaskID: int): json
+ deleteBoard(Name: string, userEmail: string)
+ SetColumnLimit(ColName: string, BoardName: string, ColState: int)
+ SetColumnLimit(ColName: string, BoardName: string, ColState: int, NewLimit: int)

Task

+ createTask(Title: string, Description: string, DueDate: string): json
+ EditTitle(newTitle: string, user: User, board: string): json
+ EditDueDate(newDate: string, user: User, board: string): json
+ EditDescription(newBody: string, user: User, board: string): json
+ GetState(string: username, string: BoardName): json

Business Layer

UserController

- users: Dictionary<string, User>
- loggedIn: Collection

+ CreateUser(password: string, email: String): void
+ DeleteUser(email: string): void
+ UserExists(email: string): boolean
+ GetUser(email: string): User
+ validatePassword(user: User, password: String): boolean
+ login(email: String, password: String): void
+ logout(user: User): void

Board

- Backlog(Collection)
- InProgress(Collection)
- Done(Collection)
+ Name(String)
- maxTasks: int = -1
- numTasks: int

+ addTask(title: String, Description: String, DueDate: String): void
+ changeState(Task): void
+ getters
+ setters

User

+ USERNAME: string
- password: string
- Boards: Collection
- TasksInProgress: Collection

+ ChangePassword(oldP: string, newP: string): void
+ ValidatePassword(password: string): bool

BoardController

- Board: Dictionary<User, Board>

+ Exists(Name: String): Boolean
+ createBoard(Name: String): void
+ deleteBoard(Name: String): void
+ getBoard(Name: String): void
+ changeMax(newMax: Int): void

Task

- Title(String)
- DueDate(String)
+ CREATION DATE(DateTime)
- Description(newBody: string): void
- State(int)
+ TaskID(int)

+ EditTitle(newTitle: string): void
+ EditDueDate(newDate: String): void
+ EditDescription(newBody: string): void
+ Getters
+ Setters

Connections

- Static ConnectedUsers: List<string>

+ static IsLoggedIn(string email): bool
+ static LoginUser(string email): void
+ static LoginUser(string email): void

Aggregation: child can exist without the parent

Composition: child cannot exist without the parent

From Wikipedia

https://en.wikipedia.org/wiki/Class_diagram

Visibility

+ public

- private

protected