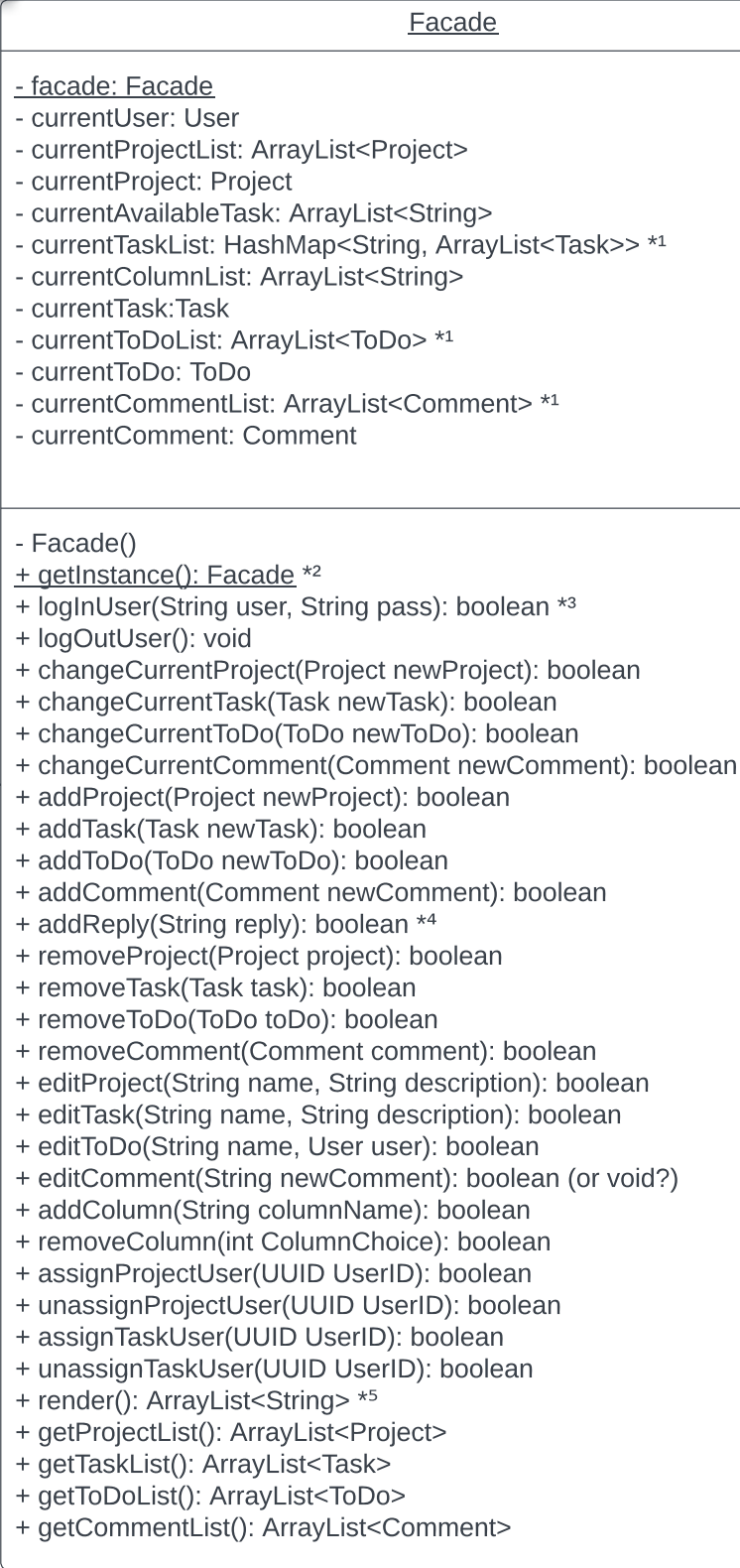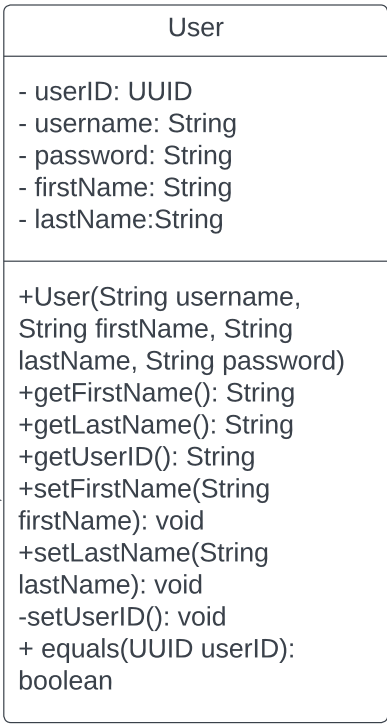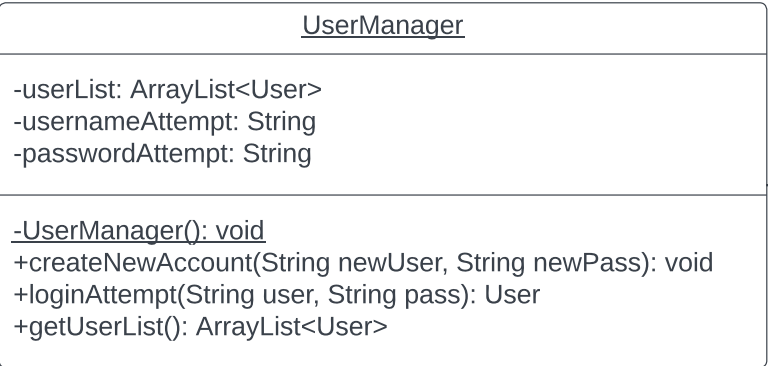**Notes (sticky):**

after ANY CHANGE, the DataWriter must be called!!!
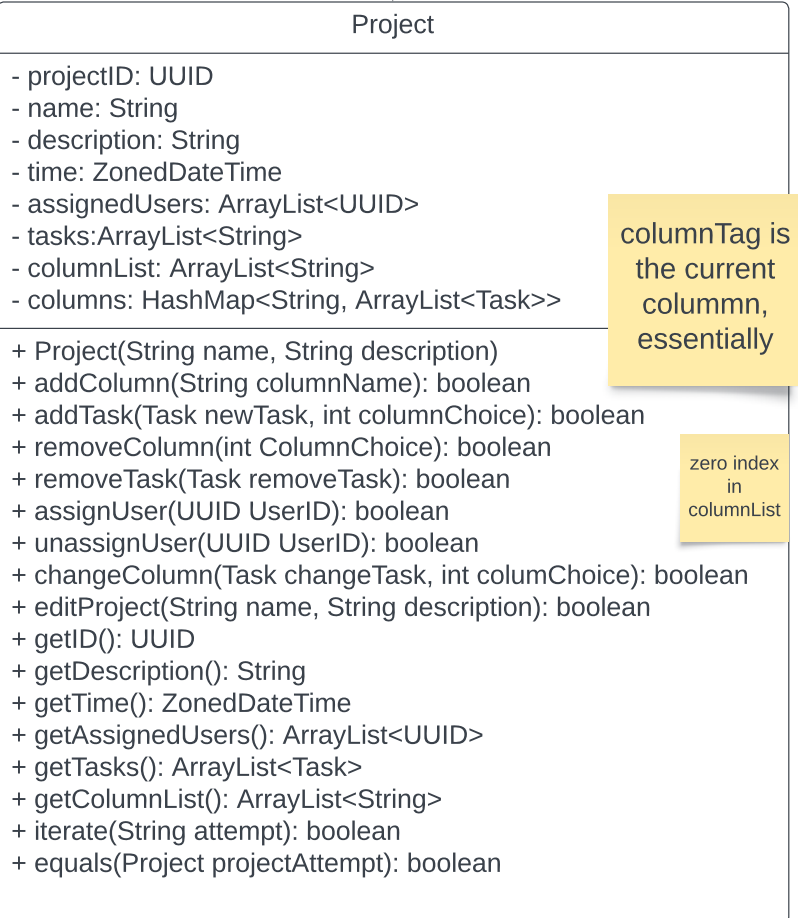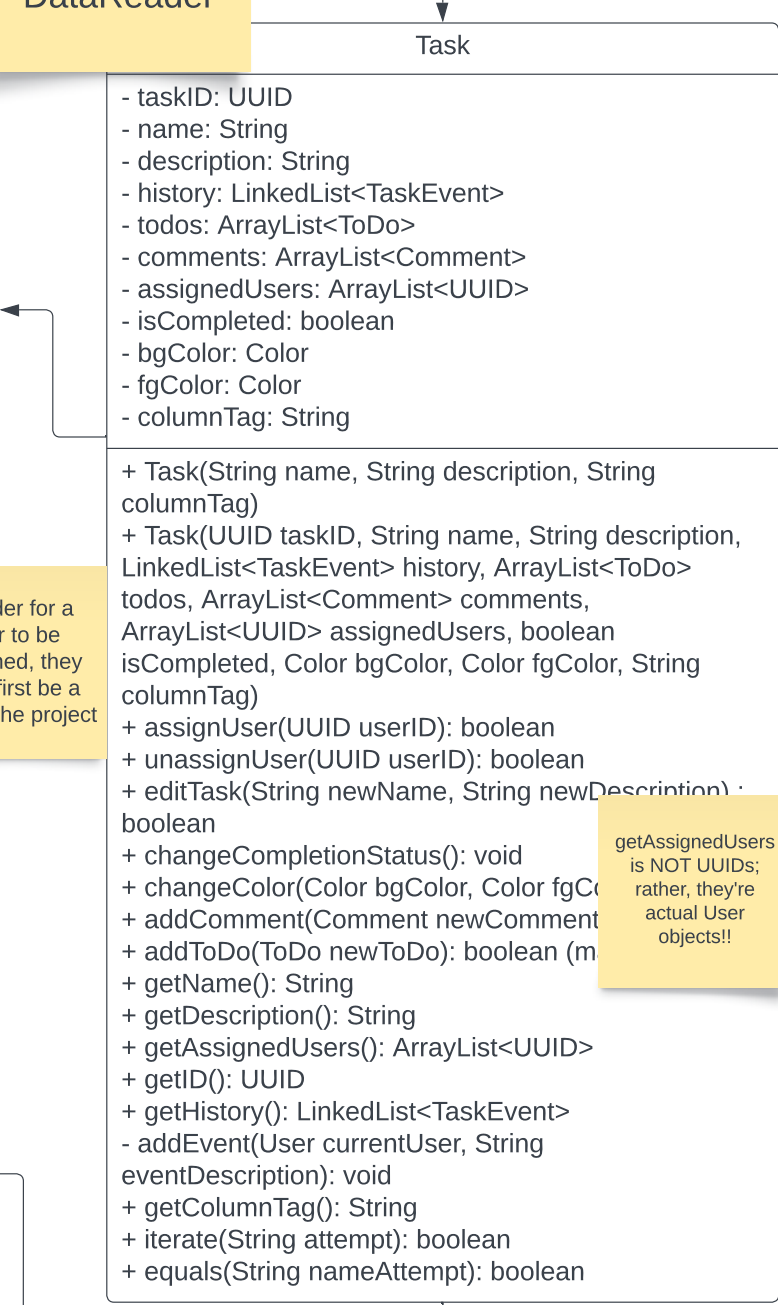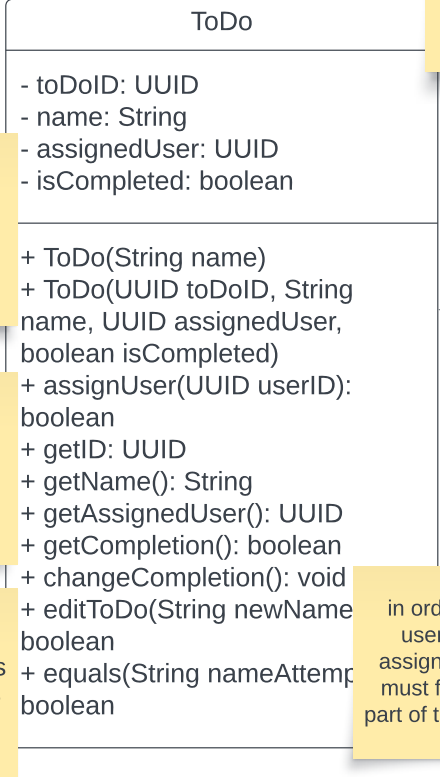
when a constructor is called: the calss containing the constructor call MUST first check for duplicates

currentUser CANNOT be null!!

*1: should these be ArrayList<String> or of their respective type?
*2: this is a singleton, right? is this how you construct one properly?
*3: should this return a type user? maybe like "currentUser = loginUser(user, pass)"? doesn't make sense
*4: is this how they should add a reply, or should it be part of addComment?
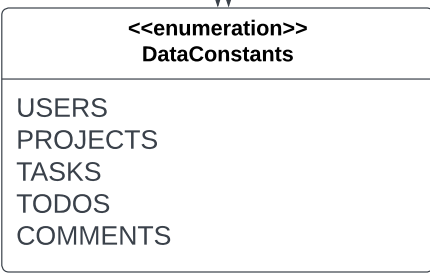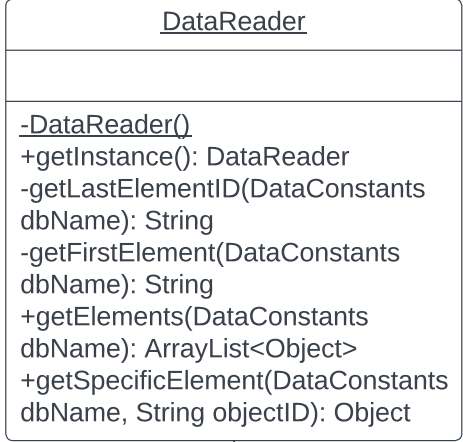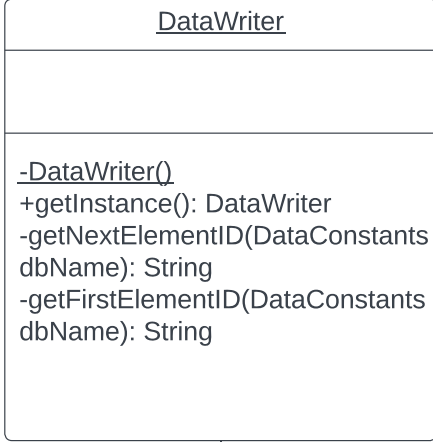*5: how does this exactly work?

if it contains a list, it has to call the DataReader

this is rene's expertise

assignedUser has to be looked up in database, can be null

all sddigned Users are also in the Task function

in order for a user to be assigned, they must first be a part of the project

getAssignedUsers is NOT UUIDs; rather, they're actual User objects!!

columnTag is the current columnn, essentially

zero index in columnList

User

- userID: UUID
- username: String
- password: String
- firstName: String
- lastName:String

+User(String username, String firstName, String lastName, String password)
+getFirstName(): String
+getLastName(): String
+getUserID(): String
+setFirstName(String firstName): void
+setLastName(String lastName): void
-setUserID(): void
+ equals(UUID userID): boolean

Facade

- facade: Facade
- currentUser: User
- currentProjectList: ArrayList<Project>
- currentProject: Project
- currentAvailableTask: ArrayList<String>
- currentTaskList: HashMap<String, ArrayList<Task>> *1
- currentColumnList: ArrayList<String>
- currentTask:Task
- currentToDoList: ArrayList<ToDo> *1
- currentToDo: ToDo
- currentCommentList: ArrayList<Comment> *1
- currentComment: Comment

- Facade()
+ getInstance(): Facade *2
+ logInUser(String user, String pass): boolean *3
+ logOutUser(): void
+ changeCurrentProject(Project newProject): boolean
+ changeCurrentTask(Task newTask): boolean
+ changeCurrentToDo(ToDo newToDo): boolean
+ changeCurrentComment(Comment newComment): boolean
+ addProject(Project newProject): boolean
+ addTask(Task newTask): boolean
+ addToDo(ToDo newToDo): boolean
+ addComment(Comment newComment): boolean
+ addReply(String reply): boolean *4
+ removeProject(Project project): boolean
+ removeTask(Task task): boolean
+ removeToDo(ToDo toDo): boolean
+ removeComment(Comment comment): boolean
+ editProject(String name, String description): boolean
+ editTask(String name, String description): boolean
+ editToDo(String name, User user): boolean
+ editComment(String newComment): boolean (or void?)
+ addColumn(String columnName): boolean
+ removeColumn(int ColumnChoice): boolean
+ assignProjectUser(UUID UserID): boolean
+ unassignProjectUser(UUID UserID): boolean
+ assignTaskUser(UUID UserID): boolean
+ unassignTaskUser(UUID UserID): boolean
+ render(): ArrayList<String> *5
+ getProjectList(): ArrayList<Project>
+ getTaskList(): ArrayList<Task>
+ getToDoList(): ArrayList<ToDo>
+ getCommentList(): ArrayList<Comment>

UserManager

-userList: ArrayList<User>
-usernameAttempt: String
-passwordAttempt: String

-UserManager(): void
+createNewAccount(String newUser, String newPass): void
+loginAttempt(String user, String pass): User
+getUserList(): ArrayList<User>

DataWriter

-DataWriter()
+getInstance(): DataWriter
-getNextElementID(DataConstants dbName): String
-getFirstElementID(DataConstants dbName): String

DataReader

-DataReader()
+getInstance(): DataReader
-getLastElementID(DataConstants dbName): String
-getFirstElement(DataConstants dbName): String
+getElements(DataConstants dbName): ArrayList<Object>
+getSpecificElement(DataConstants dbName, String objectID): Object

---

<<enumeration>> DataConstants

USERS
PROJECTS
TASKS
TODOS
COMMENTS

ToDo

- toDoID: UUID
- name: String
- assignedUser: UUID
- isCompleted: boolean

+ ToDo(String name)
+ ToDo(UUID toDoID, String name, UUID assignedUser, boolean isCompleted)
+ assignUser(UUID userID): boolean
+ getID: UUID
+ getName(): String
+ getAssignedUser(): UUID
+ getCompletion(): boolean
+ changeCompletion(): void
+ editToDo(String newName): boolean
+ equals(String nameAttempt): boolean

(note near ToDo: int based on error codes)

Task

- taskID: UUID
- name: String
- description: String
- history: LinkedList<TaskEvent>
- todos: ArrayList<ToDo>
- comments: ArrayList<Comment>
- assignedUsers: ArrayList<UUID>
- isCompleted: boolean
- bgColor: Color
- fgColor: Color
- columnTag: String

+ Task(String name, String description, String columnTag)
+ Task(UUID taskID, String name, String description, LinkedList<TaskEvent> history, ArrayList<ToDo> todos, ArrayList<Comment> comments, ArrayList<UUID> assignedUsers, boolean isCompleted, Color bgColor, Color fgColor, String columnTag)
+ assignUser(UUID userID): boolean
+ unassignUser(UUID userID): boolean
+ editTask(String newName, String newDescription): boolean
+ changeCompletionStatus(): void
+ changeColor(Color bgColor, Color fgColor): void
+ addComment(Comment newComment): ...
+ addToDo(ToDo newToDo): boolean (m...)
+ getName(): String
+ getDescription(): String
+ getAssignedUsers(): ArrayList<UUID>
+ getID(): UUID
+ getHistory(): LinkedList<TaskEvent>
- addEvent(User currentUser, String eventDescription): void
+ getColumnTag(): String
+ iterate(String attempt): boolean
+ equals(String nameAttempt): boolean

Project

- projectID: UUID
- name: String
- description: String
- time: ZonedDateTime
- assignedUsers: ArrayList<UUID>
- tasks:ArrayList<String>
- columnList: ArrayList<String>
- columns: HashMap<String, ArrayList<Task>>

+ Project(String name, String description)
+ addColumn(String columnName): boolean
+ addTask(Task newTask, int columnChoice): boolean
+ removeColumn(int ColumnChoice): boolean
+ removeTask(Task removeTask): boolean
+ assignUser(UUID UserID): boolean
+ unassignUser(UUID UserID): boolean
+ changeColumn(Task changeTask, int columnChoice): boolean
+ editProject(String name, String description): boolean
+ getID(): UUID
+ getDescription(): String
+ getTime(): ZonedDateTime
+ getAssignedUsers(): ArrayList<UUID>
+ getTasks(): ArrayList<Task>
+ getColumnList(): ArrayList<String>
+ iterate(String attempt): boolean
+ equals(Project projectAttempt): boolean

Comment

- commentID: UUID
- commentData: String
- author: UUID
- replies: ArrayList<Comment>
- time: ZonedDateTime (?)
- commentBackEdge: UUID

+ Comment(User currentUser, String commentData, Comment backEdge)
+ Comment(UUID commentID, String commentData, UUID author, ArrayList<Comment> replies, ZonedDateTime time, UUID commentBackEdge)
+ editComment(String newComment): boolean
+ getCommentData(): String
+ getAuthor(): UUID
+ getBackEdge(): UUID
+ getTime: ZonedDateTime
+ addReply(String reply, UUID author): boolean
+ getReplies() : ArrayList<Comment>
+ getID(): UUID
+ equals(UUID commentAttempt): boolean

TaskEvent

-eventTime: ZonedDateTime
-eventName: String
-involvedUser: User

+TaskEvent(String eventName, User involvedUser)