UserDao _username: string _password: string _userld: string Singleton Pattern <<Enumeration>> UserState NOTLOGGEDIN FileHandler LOGGEDIN _instance: FileHandler() Null Object Pattern AbstractUser 4 + GetInstance(): FileHandler username: string NullUser 0...1 ReadFile(_pathToFile: string): any _password: string + ReadArrayFile(_pathToFile: string): any[] _userld: string + ReadObjectFile(_pathToFile: string): any + WriteFile(_pathToFile: string, _dataToWrite: any): void TextAdventureDataBase _textAdventures: TextAdventure[] users: User[] User currentUser: User _userState: UserState Dao Pattern + ShowFunctionalities(): Promise<void> Singleton Pattern TextAdventure - ShowRegisteredFunctionalities(): Promise<void> _title: string TextAdventureDao ConsoleHandling - HandleLoggedInAnswer(_answer: string): Promise<void> _title: string x: number _instance: ConsoleHandling() - HandleAnswer(_answer: string): Promise<void> x number _y: number + GetInstance(): ConsoleHandling - SearchTextAdventureByTitle(): Promise<void> 0...n _map: Location[] _y: number + Question(_question: string): Promise<string> ListAllTextAdventures(): Promise<void> startX: number _map: Location[] + ShowPossibilities(_showPossibilities: string[], _question: string): Promise<string> - Login(): Promise<void> startY: number _startX: number + PrintInput(_input: string): void - Register(): Promise<void> _textAdventureId: string _startY: number + CloseConsole(): void CreateTextAdventureMap(): Promise<void> _creatorUserId: string textAdventureId: string - PlayTextAdventure(_textAdventure: TextAdventure): Promis e < void > _playCounter: number _creatorUserId: string - ShowStatistics(): void _playCounter: number _turnCounter: number - GoNext(): Promise<void> turnCounter: number + PlayTextAdventure(): Promise<void> - PrintMap(_north?: Location, _east?: Location, _south?: Location, _west?: Location): void 0...n Location _x: number _y: number

Dao Pattern

title: strina