User - name : String - projects : Project - tasks : vector<lssue> - sprints : vector<Sprint> + getName(): String + setName(string U) : Void + printTasks(): Void + numTasks(): Int + printProject(): Void + setProject(Project P) : Void + setIssue(Issue P): Void + printlssues(): Void + setIssue(Issue P) : Void + printlssues(): Void + addUser2Issue(User U, Issue I): Int + isNewUser(String S) : Bool TeamMember ProjectLead - name : String - name : String + getName(): String + getName(): String + setName(string U) : Void + setName(string U) : Void + printTasks(): Void + printTasks(): Void + numTasks(): Int + numTasks(): Int

Proiect - title : String - toDo : vector<lssue> - inProgress : vector<Sprint> - workDone : vector<Issue> - projectOwner : User - projectLeads : vector<User> - teamMembers : vector<User> + getTitle(): String + setTitle(string T): Void + getToDo(): vector<lssue> + addToDo(Issue I): Void + getToDo(): vector<Issue> + getInProg(): vector<Sprint> + addInProg(Sprint S): Void + addWorkDone(Sprint) : Void + getWorkDone(): vector<lssue> + addToDo(Issue I): Void + getToDo(): vector<lssue> + addInProg(Sprint S): Void

Sprint - iobs : vector<lssue> - timeFrame : Int + removeDoneJobs(): Void + addJob(Issue S): Void + getTimeF(): Int + setTimeF(int N) : Void + numJobs(): Int Issue - iD : Int - type : IssueType - reporter : User - assignee : User - priority : Int - status : StatusType - createdDate : String - comments : String - description : String - timeFrame : Int - priority : Int + getTimeF : Int + setTimeF (int F): void

StatusType Open InProgress Done Overdue IssueType UserStory Task Build Test Debug Documentation