# Requirements

## Functional requirements

**Critical priority:**

1. As a project creator, I want to be able to create projects that contain requirements and tasks, such that we can track the progress of the project.
2. As a project creator, I want to be able to assign team members and their roles, such that they can work on projects.
3. As a product owner, I want to be able to add, remove and prioritize requirements(how to prioritize requirements, need to be specific), so that projects can be developed using an iterative approach.
4. As a scrum master, I want to be able to add tasks to the project, such that the rest of the team members know what tasks they have to work on.
5. As a team member, I want to be able to access the system without having to login, because the system can only be used by one team member at a time.

**High priority:**

1. As a product owner, I want to change who is responsible for a requirement….
2. As a scrum master, I want change who is responsible for a task….
3. As a product owner I want requirements to automatically get marked with “Ended” when all tasks for a requirement are done, so that I can see what requirements I should be testing.
4. As a product owner, I want requirements to contain an id, user stories in who, what, why template, estimated time, a deadline, who is responsible, status, total hours spent, such that I can easily get an overview of all relevant information for a requirement.
5. As a product owner, I want to be able to approve or reject requirements, such that the project can reach a finished state, and make sure it meets the customers needs.
6. As a scrum master, I want each task to contain all information (Requirement ID, task ID, title, time estimation, deadline, responsible team member, status, hours spent) such that I can easily get an overview of all relevant information for a requirement.
7. As a team member, I want to be able to register a total amount of hours to the system whenever a task has been finished, so that I can keep track of its progress and see our productivity in regards to how well we can estimate time for tasks.
8. As a scrum master, I want to mark the status of each task in the form of: started, not started, finished; such that team members know which tasks should be worked on.

**Low priority:**

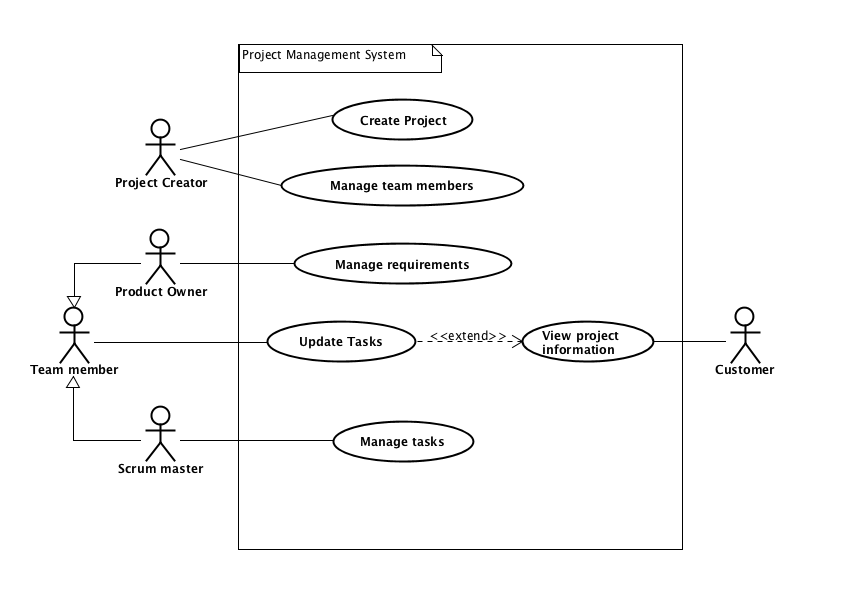
1. As a product owner, I want to be able to search information regarding the projects by ID, responsible team members, deadlines, such that I can have an overview of the progress on a current project.
2. As a project creator, I want to have the ability to change the roles of the team members, so they can work on more suitable tasks.
3. As a customer, I want the project’s description, requirements and their status, displayed on a website, such that I can track its progress.
4. As a project creator, I want to be able to add/remove new team members for a project in case the team is not able to cover all the tasks.
5. As a customer, I want my project’s information to be updated daily, such that I am well-informed on its progress.

## Non-functional Requirements

1. As a product owner, I want the GUI implemented with Java/JavaFx, such that it would be easy to modify.
2. As a customer, I would like to be able to access project information using Google Chrome (version 86.0.4240.193, release date 2020-11-10) Mozilla Firefox (version Firefox 82, release date 2020-10-20) Safari (version Safari 14.0, release date 2020-9-16). So that I will be able to find information from multiple devices.

# 

# Use Case Diagram



# 

# 

# Use Case Descriptions

| **Use case** | **Create Project** |
| --- | --- |
| **Summary** | Creating a project in which the customer can see information |
| **Actor** | Project Creator |
| **Precondition** | Analysis should be accepted by the customer and all requirements and tasks should be specified |
| **Postcondition** | A project is created, therefore, requirements, tasks and other work can be completed |
| **Base sequence** | 1. Open system 2. Create a project 3. Choose the name of the project 4. In case requirements and tasks are to be added, go to step 5, otherwise go to step 9 5. Add requirement 6. Add task for the requirement 7. If necessary, repeat step 6, otherwise go to step 8 8. If multiple requirements should be added, go to step 5 9. Save project |
| **Exception sequence** |  |
| **Note** | This use case covers requirements 1, 2 |

| **Use case** | **Manage Team Members** |
| --- | --- |
| **Summary** | Adding/removing Team Members or assigning (new) roles to those |
| **Actor** | Project Creator |
| **Precondition** | A project has to be created |
| **Postcondition** | The project is better organized so that there are no disruptions to completing the requirements for it |
| **Base sequence** | 1. Open system 2. Search for a project 3. Add Team Members 4. In case Team Members are to be removed, go to step 10 5. Assign roles to each Team Member 6. In case of changing roles of a Team Member, go to step 14 7. In case of unpredicted delays or reasons that interfere with getting the tasks done, redo either step 3 or 4 8. Save project 9. Start working on the project 10. Remove Team Member 11. If necessary, repeat step 10, otherwise go to step 12 12. Save changes 13. Go to step 5 14. Choose Team Member 15. Change role 16. If required, repeat step 14, otherwise go to step 17 17. Save changes 18. Go to step 7 |
| **Exception sequence** |  |
| **Note** | This use case covers requirements 4, 5, 16, 21  If there is no project created, then there can not be any management of the Team Members done. |

| **Use case** | **Manage requirements** |
| --- | --- |
| **Summary** | Add or remove requirements, such that the project is developed using an iterative approach. |
| **Actor** | Product Owner |
| **Precondition** | A project should already be created in the system. |
| **Postcondition** | The requirements are now updated. |
| **Base sequence** | 1. Open System. 2. Open Project. 3. Open List of Requirements. 4. Add or Remove Requirement/s. 5. Give Requirement/s an ID, estimated time, user stories, deadline, who is responsible for working on them, status and total time spent. 6. If all tasks for requirements are complete, the requirement is marked as ‘ended’. 7. Change Priority of Requirements if need be. 8. Approve or reject requirements if need be. 9. Save List of Requirements. 10. Save Project. |
| **Exception sequence** |  |
| **Note** | This use case covers requirements 3, 6, 7, 8 |

Split up the base sequence into more parts = step 4 - Add/remove requirements

| **Use case** | **Update tasks** |
| --- | --- |
| **Summary** | Update tasks that a team member has worked on, and find project information. |
| **Actor** | Team member |
| **Precondition** | A project has to be created in the system. |
| **Postcondition** | Tasks have been updated, and team members have found required information. |
| **Base sequence** | 1. Open system 2. Search for a project 3. If project not found go back to step 2 4. Open the project 5. For updating tasks go to 5 6. Finding information about a project go to 10 7. Open tasks 8. Open task you need to update 9. Enter amount of hours spent and what date 10. Save task 11. If more tasks need to be updated go to step 2 12. Read information about project 13. To find information about requirements go to step 11 14. To find information about tasks go to step 13 15. Open requirements 16. Open specific requirement you are interested in 17. To find information about a different project go to step 2 18. Open tasks 19. Open specific task you are interested in 20. To find information about a different project go to step 2 |
| **Exception sequence** |  |
| **Note** | If there are no projects in the system, team members will not be able to do anything in the system.  This use case covers requirements 5, 10 |

| **Use case** | **View project information** |
| --- | --- |
| **Summary** | Customer finds information regarding the ordered project. |
| **Actor** | Customer |
| **Precondition** | Work has started on the customers ordered project. |
| **Postcondition** | Customer knows about progress on the ordered project. |
| **Base sequence** | 1. Enter provided web address 2. Search for specific project 3. Wrong search term, repeat step 2. 4. Open project 5. Read project information 6. For information about a requirement go to step 5 7. Open requirements 8. Open specific requirement 9. To find more information about the project go to step 3 10. To find information about another project go to step 2 |
| **Exception sequence** |  |
| **Note** | Website has to be regularly updated by the development team in order for the customer to see the latest information regarding their projects.  This use case covers requirements 12, 14, 16 |

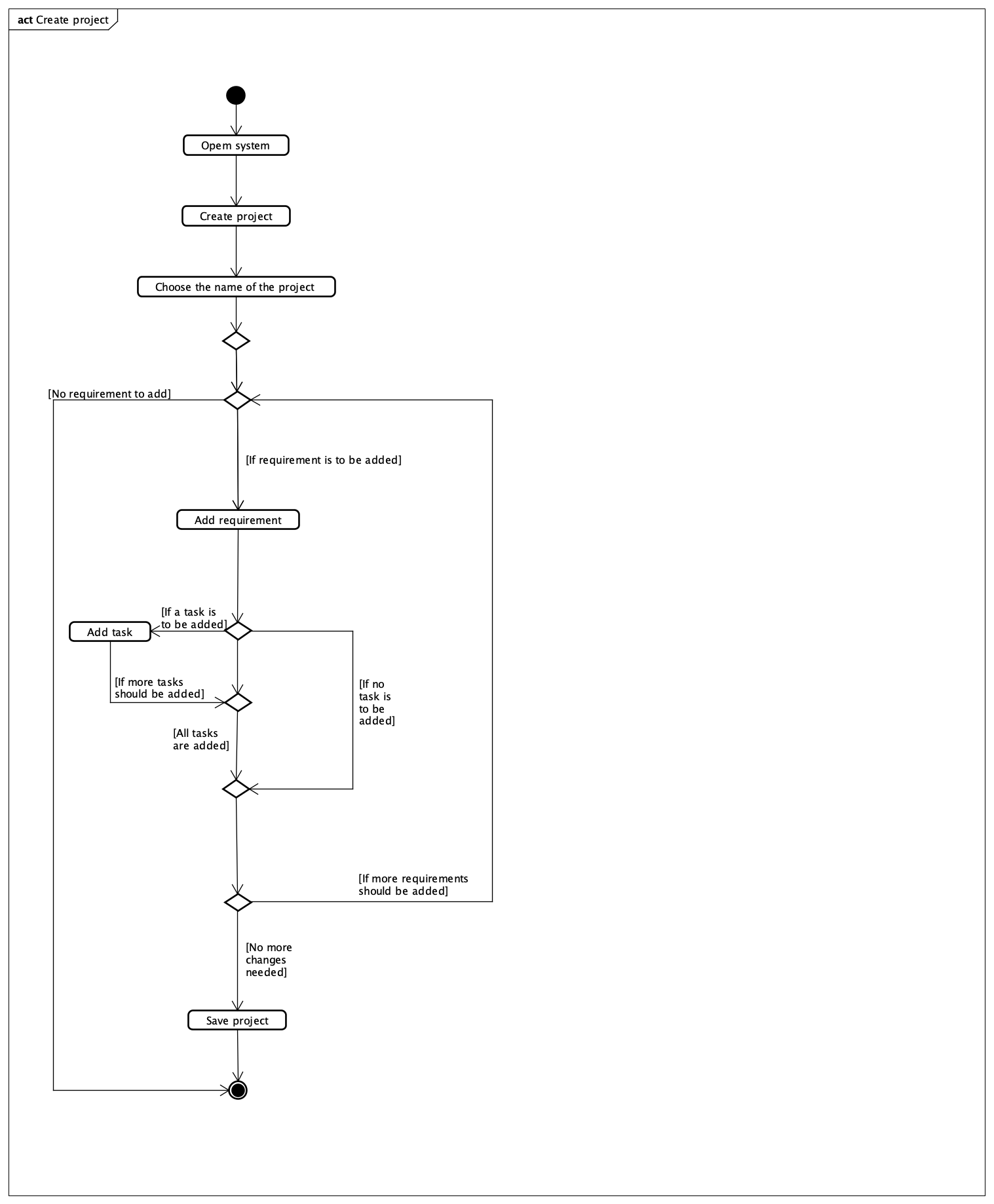
| **Use case** | **Manage Tasks** |
| --- | --- |
| **Summary** | Adding tasks to the project and documenting the status of each task in form of - “Started”, “Not started”, “Finished”. Also, reporting the status of the tasks further to the Product owner. |
| **Actor** | Scrum master |
| **Precondition** | Project requirements have to be identified and project with assigned team members created. |
| **Postcondition** | A task has been added and later on updated from “Not started” status to "Started" status or “Finished” status. A sent documentation to the Product owner about the status information for further approval/verification. |
| **Base sequence** | 1. Start the system 2. Search for the requirements by their ID to display necessary information like description, time estimation, deadlines, responsible team members. 3. In case of a new project - Add task/-s, if not, then skip this step and go to step 4. 4. Follow-up and check the status of each specific task and requirement by using a search option like in step 2. 5. Update the status of the project by marking each task:  * If the work process on the task has begun, then mark it as “Started”. * If the work process on the task has not begun yet, then mark it as “Not started”. * If the work process on the task has been finished, then mark it as “Finished”  1. Save changes. |
| **Exception sequence** |  |
| **Note** | This use case covers requirements 4, 9, 11. |

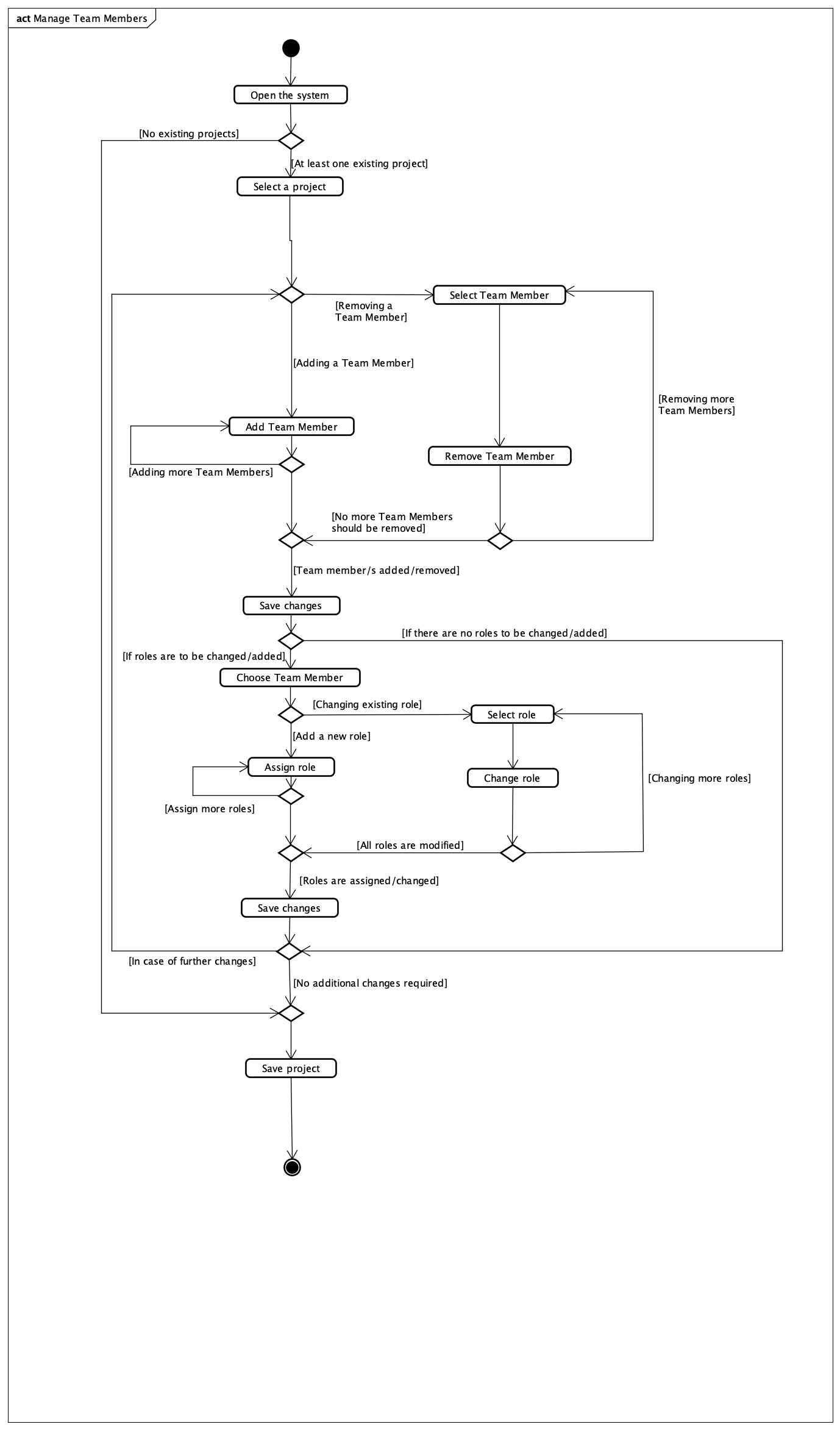
Relation between requirements and use cases

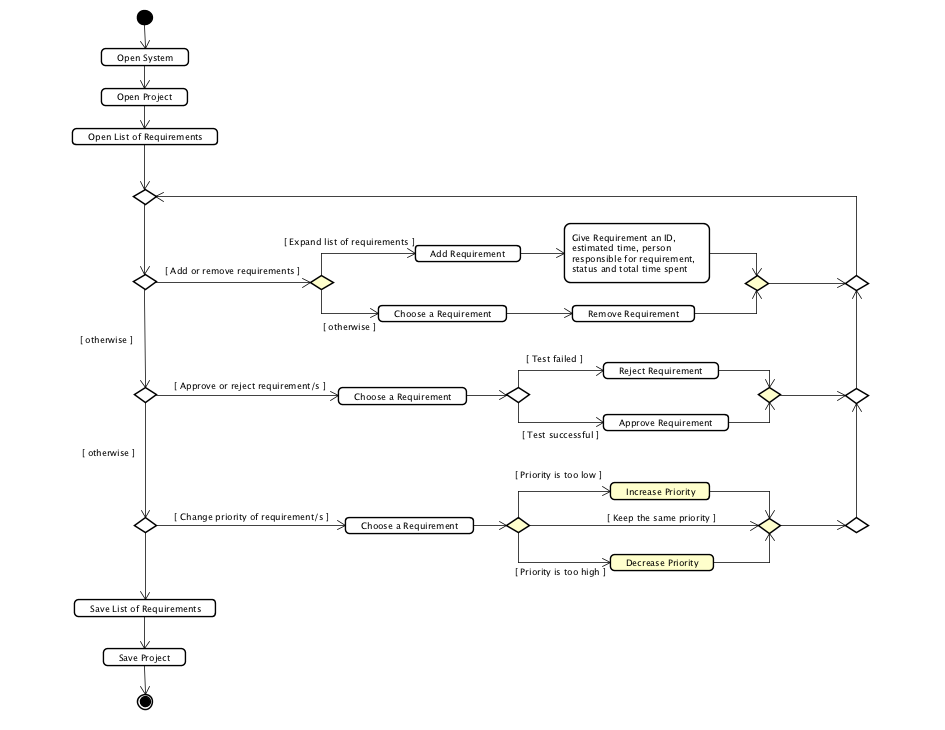
| **Use case** | **Covered requirements** |
| --- | --- |
| *Create project* | 1 |
| *Manage team members* | 2, 13, 15 |
| *Manage requirements* | 3, 6, 7, 8 |
| *Update tasks* | 5, 10 |
| *View project information* | 12, 14, 16 |
| *Manage tasks* | 4, 9, 11. |

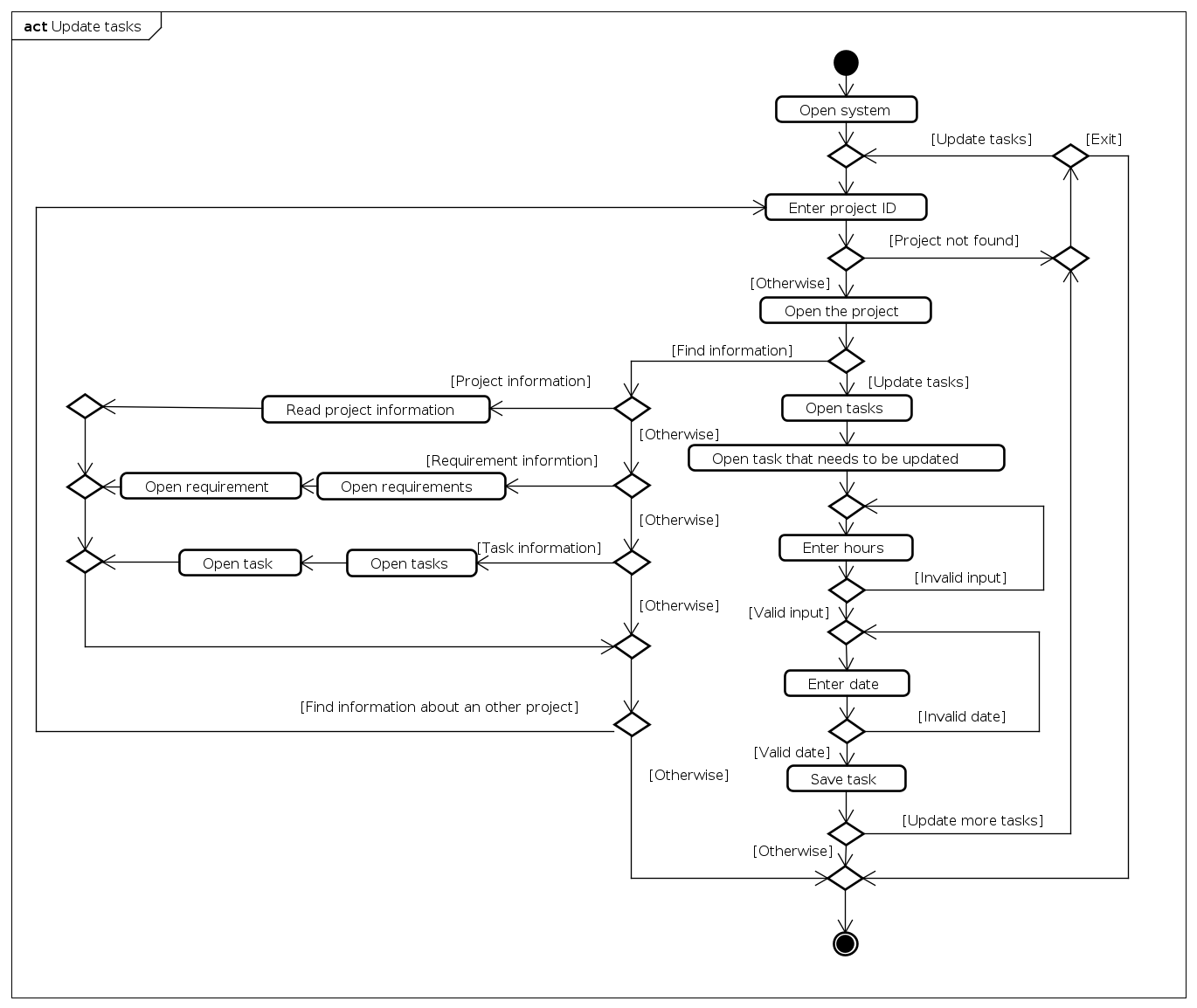
| **Requirement** | **Related use case** |
| --- | --- |
| **1** | Create project (step 2) |
| **2** | Create project (step 3) |
| **3** | Manage requirements (step 4) |
| **4** | Manage tasks (step 3) |
| **5** | Update tasks (step 1) |
| **6** | Manage requirements (step 5) |
| **7** | Manage requirements (step 4a) |
| **8** | Manage requirements (step 7) |
| **9** | Manage tasks (step 2 and 4) |
| **10** | Update tasks (step 7) |
| **11** | Manage tasks (step 5) |
| **12** | View project information (step 2) |
| **13** | Manage team members (step 5) |
| **14** | View project information (step 3) |
| **15** | Manage team members (step 3, 10) |
| **16** | View project information (step 3) |

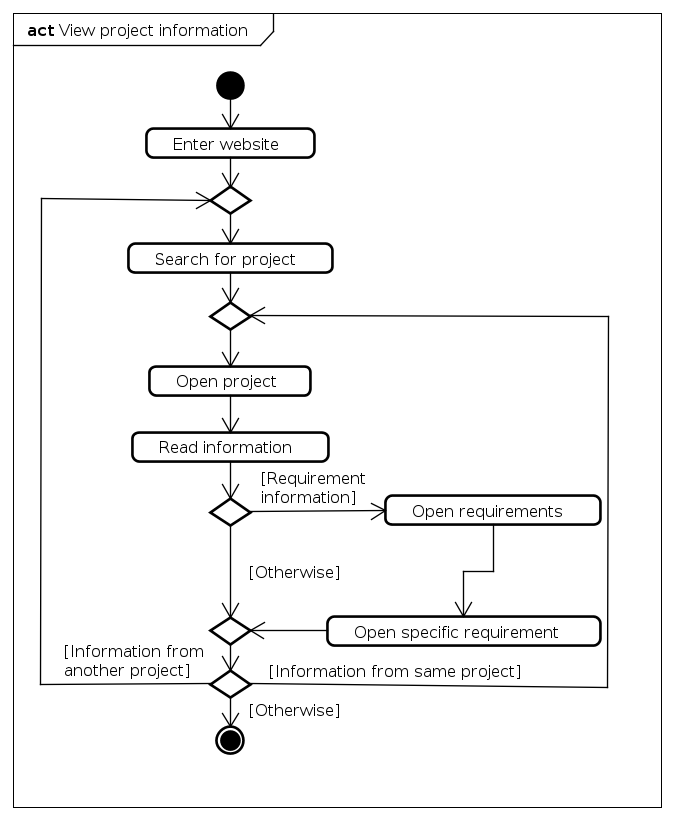
Activity diagrams

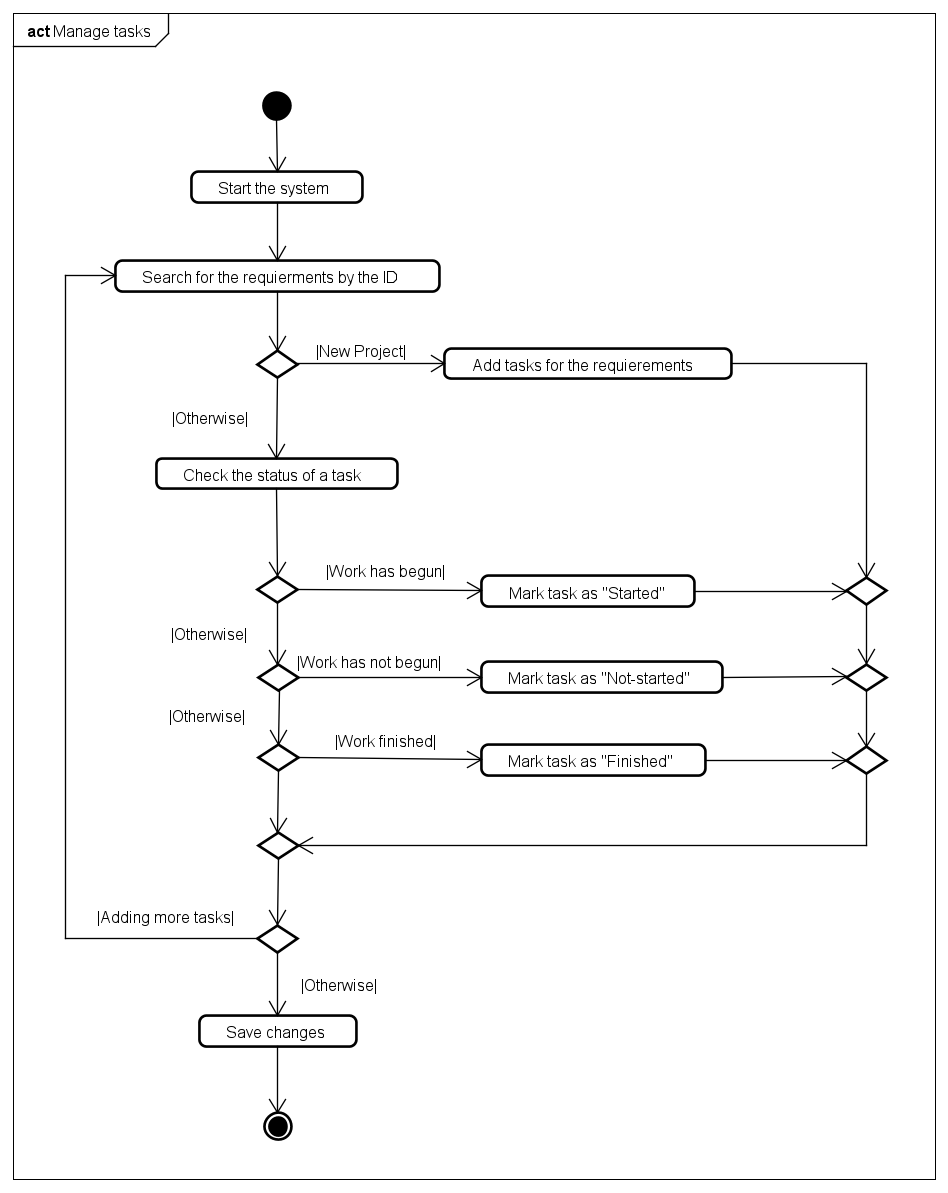












Domain model

