

**Name** - Project Management System

**Team members** - Ankit Srivastava, Nikhil Sulegaon, Rohit Mehra

### Summary :-

Prorg is a web-based collaboration tool that organizes projects into boards. In one glance, Prorg tells you what's being worked on, who's working on what, and where something is in a process.

Imagine a white board, filled with lists of sticky notes, with each note as a task for you and your team. Now imagine that each of those sticky notes has photo-attachments, and a place to comment and collaborate with your teammates. Now imagine that you can take that whiteboard anywhere you go, and can access it from any computer through the web. That's Prorg!

### 1. Project Requirements:-

User Requirements					
S.No	Requirement ID	Description	Topic Area	Actor	Priority
1	UR01	User should be able to create a storyboard	Storyboard	Member User, Admin user	High
2	UR02	User should be able to add members to the storyboard	Storyboard	Member User, Admin user	High
3	UR03	User should be able to create/add/remove a swimlane	Swimlane	Member User, Admin user	High
4	UR04	User should be able to create/add/remove a card	Card	Member User, Admin user	High
5	UR05	User should be able to add label to a card	Label, Card	Member User, Admin user	Medium
6	UR06	User should be able to add/remove a task	Task, Card	Member User, Admin user	High
7	UR07	User should be able to move a card	Card	Member User, Admin user	High
8	UR08	User should be able to assign a card to another person	Card	Member User, Admin user	High
9	UR09	Admin user should be able to move card from	SwimLane	Admin User	High

		restricted swimlane			
10	UR10	User should receive notification when a card/swimlane is modified( provided user has access to following card/swimlanes)	Notification	Member User, Admin user, SMTP server	Medium

Business Requirements					
S.No	Requirement ID	Description	Topic Area	Actor	Priority
1		There is no business requirement			

Non Functional Requirements			
S.No	Requirement ID	Description	Priority
1	NFR01	Application can be accessed from all the browsers( Laptop/Desktop/Mobile/Tablet)	High
2	NFR02	Application can only be accessed by authorized users after successful login	High
3	NFR03	Adding a storyboard , swimlane or card should not take more than 2 seconds	High
4	NFR04	High Speed Connectivity will be needed to use the application	High
5	NFR05	Application should be able to scale itself depending on the number of users	High

## 2. Data storage:-

This project will be using MySQL RDBMS with an ORM used to map the tables of the database with the models of the backend.

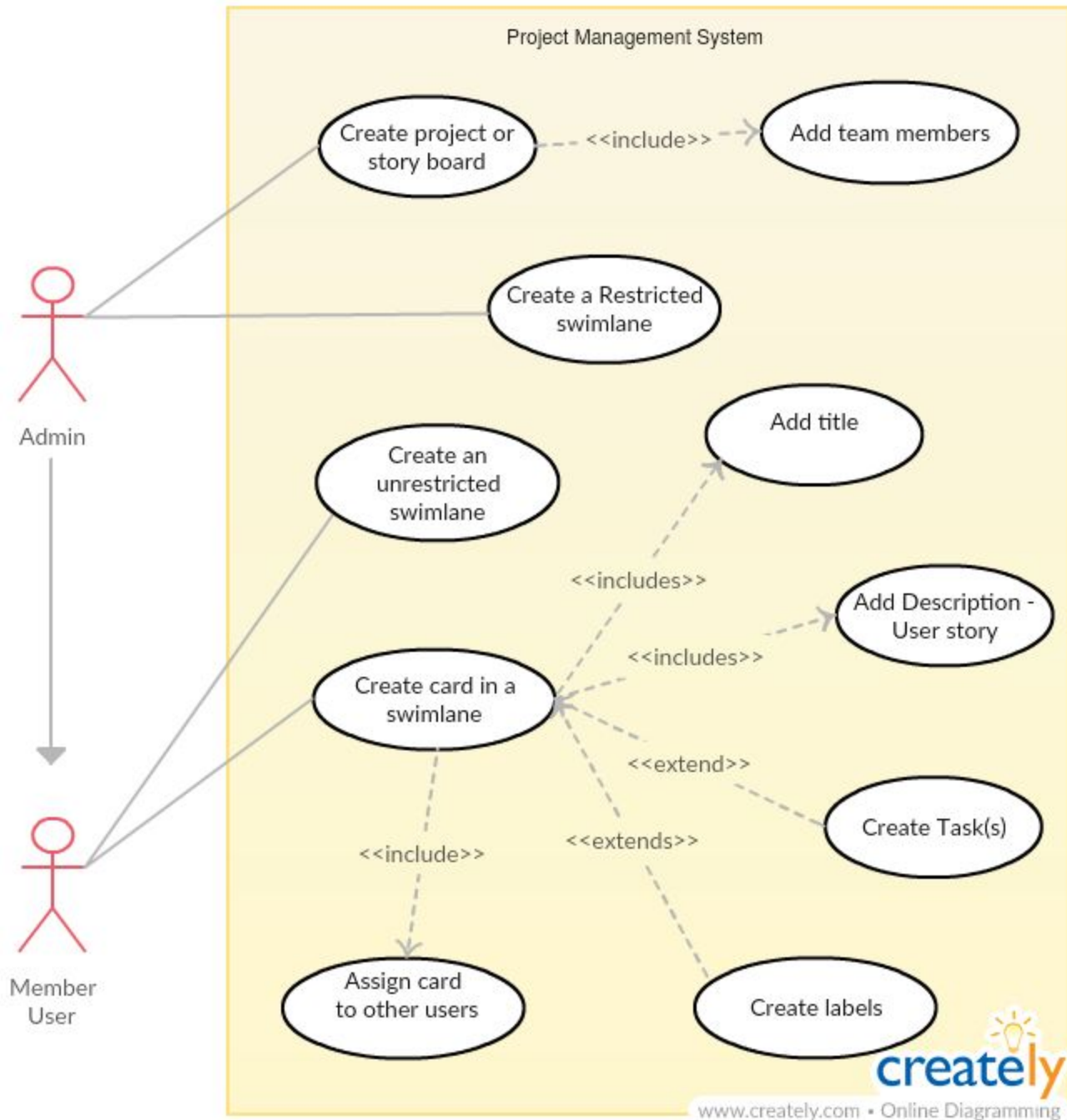
### 3. Use case:-

#### A. Use case Overview :-

Requirement ID: UR01, UR02, UR03, UR04, UR05, UR06, UR07, UR08

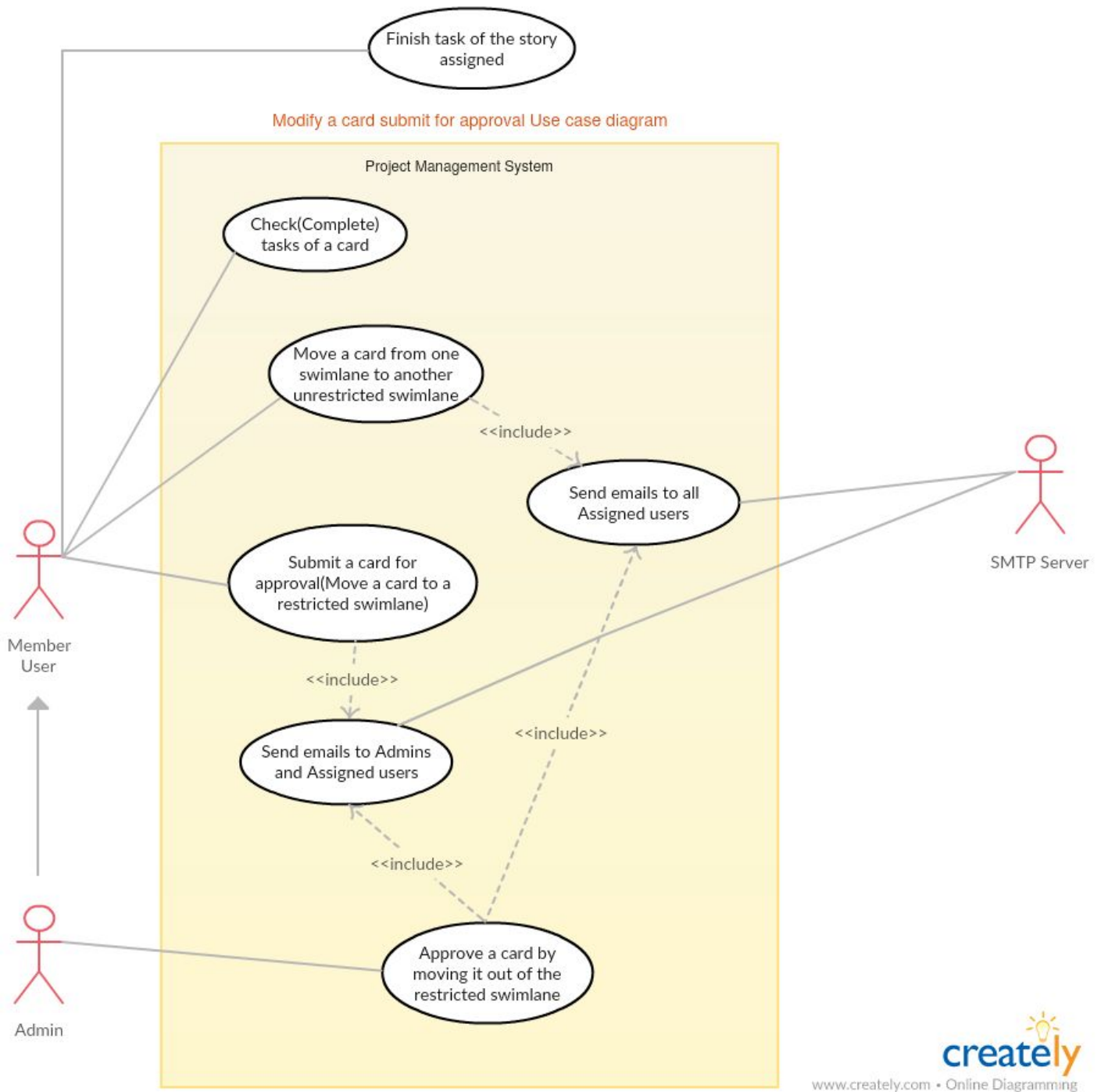
Use Case ID: #101

#### Create a new project and a card in it Use case diagram



Requirements ID: UR04, UR07, UR09, UR10

Use Case ID: #102



## B. Use case documents :-

## Use case document - #101

<b>Name of Use Case:</b>	Create a new card
<b>Created By:</b>	Project team #31
<b>Date Created:</b>	10/24/17
<b>Description:</b>	A registered user creates a new storyboard with a swimlane in it. A new card is then created in this swimlane by supplying the title, description, tasks and label.
<b>Actors:</b>	Member User and Admin
<b>Preconditions:</b>	<ol style="list-style-type: none"> <li>1. A team is working on a project and they want to effectively collaborate among themselves.</li> <li>2. This team has well defined set of goals it wants to achieve.</li> </ol>
<b>Postconditions:</b>	<ol style="list-style-type: none"> <li>1. The card's progress must be timely updated.</li> <li>2. Each member on the team should use this portal to know the overall progress of the project.</li> <li>3. Each member should also consult this portal before they start working on a part of the project.</li> </ol>
<b>Flow:</b>	<ol style="list-style-type: none"> <li>1. A user logs in by providing the right email and password.</li> <li>2. This registered user creates a new storyboard. This user becomes the Admin to the newly created storyboard.</li> <li>3. The Admin then grants a number of other registered users access to this storyboard, thus making them members of the storyboard.</li> <li>4. Creation of the storyboard involves creation of a default swimlane in the storyboard.</li> <li>5. A member then creates another swimlane - LaneTwo.</li> <li>6. A member then creates a card in LaneTwo.</li> <li>7. A card is created by supplying a Title, Description - which is more often than not a user story.</li> <li>8. The member can also create tasks which are a list of things to be done in order to complete a card.</li> <li>9. The member also add a label with a color code the card. Labels are used to categorization purposes within a swimlane.</li> <li>10. The user then saves the card.</li> </ol>

<b>Alternative Flows:</b>	In step 7, the user might not create tasks for each card. Tasks are not the only metric used to infer whether a card is complete or not. In such cases, a user may move a card to another lane whenever he/she deems it to be fit.
<b>Requirements:</b>	The following requirements must be met before execution of the use case <ul style="list-style-type: none"> <li>1. The user who logs into the system must be registered.</li> <li>2. There must be at least two users registered in the system.</li> </ul>

## Use case document - #102

<b>Name of Use Case:</b>	Update a card and submit it for approval
<b>Created By:</b>	Project team #31
<b>Date Created:</b>	10/24/17
<b>Description:</b>	A member user, finishes off the quantum of work denoted by a card, update it and moves it to a restricted lane. An Admin then review the work done and approve the card by moving it out of the restricted lane.
<b>Actors:</b>	Member User, Admin and SMTP server.
<b>Preconditions:</b>	<ul style="list-style-type: none"> <li>1. The user who is updating the card must be logged in to the system.</li> <li>2. User will only be able to view if the user has access to the storyboard.</li> </ul>
<b>Flow:</b>	<ul style="list-style-type: none"> <li>1. A member user finishes the work that he/she is assigned (this happens outside the system boundary).</li> <li>2. This user then updates the card to which he/she was assigned to. This updation can involve checking off all the tasks on the card.</li> <li>3. The member user then move this card from an unrestricted swimlane to a restricted swimlane. Thereby submitting the card for approval.</li> <li>4. The SMTP server then sends out emails to all the Admins and the other users assigned to the card. Notifying them that a card is submitted for approval.</li> <li>5. The Admin then reviews the work done as part of the card and then approves the card thus moving the card to the next unrestricted swimlane in the storyboard.</li> <li>6. On approval all the members assigned to the card and all the other Admins are notified of this approval via email. This is facilitated by the</li> </ul>

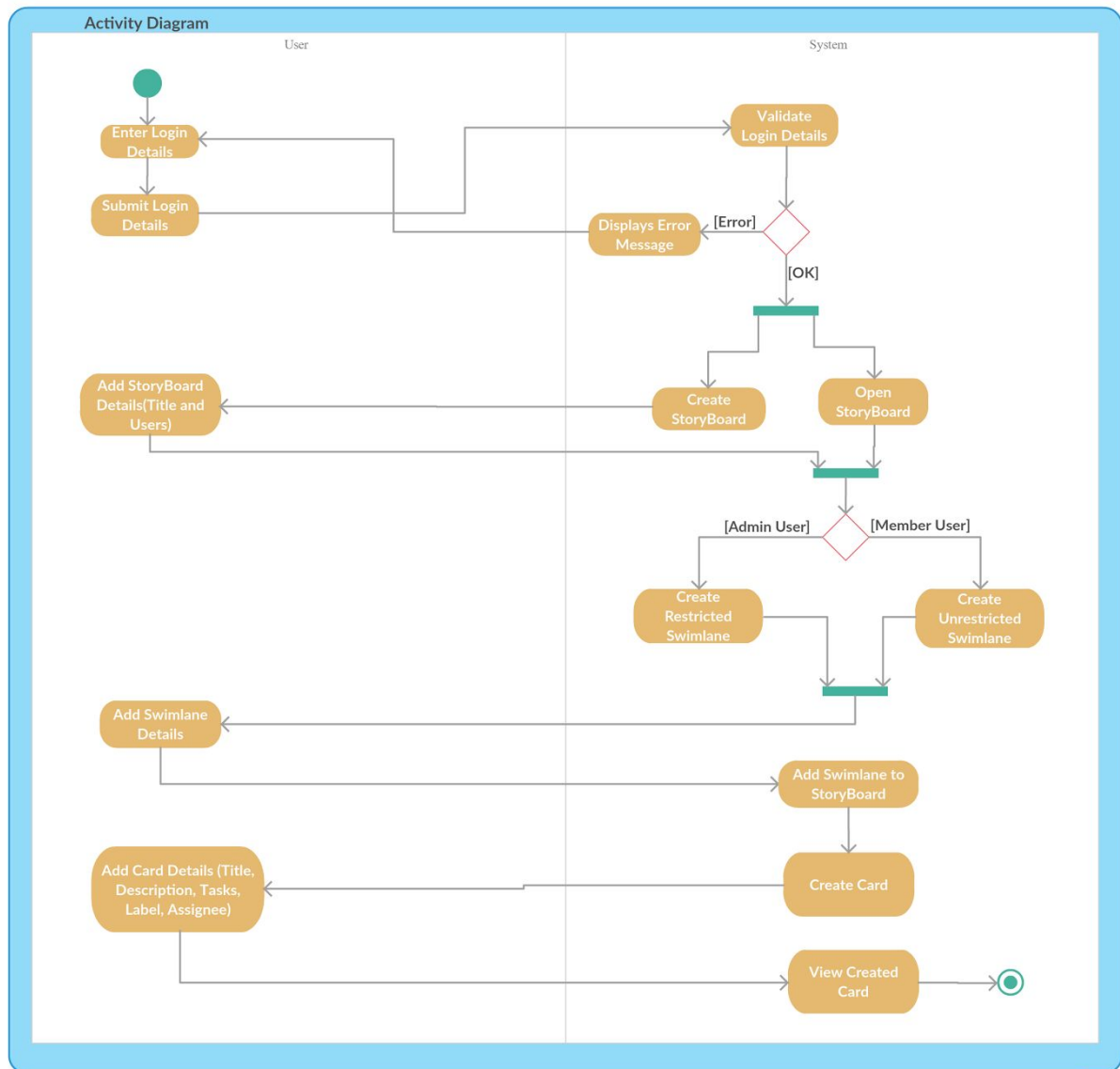
	SMTP server.
<b>Alternative Flows:</b>	In step 5, the Admin can choose to reject the card as well. If the admin rejects the card. The card moves back to the lane it came from. The SMTP server then notifies all the members assigned to the card and all other Admins.
<b>Exceptions:</b>	In step 5, a card after approval can move to a lane that happens to be restricted. In such cases all the admins are notified of another approval request. The card can then be moved out of this lane by approval or rejection. There could be multiple restricted lanes configured one after the other as well.
<b>Requirements:</b>	<p>The following requirements must be met before execution of the use case</p> <ol style="list-style-type: none"> <li>1. The Admin must actively monitor the portal and approve or reject a card that was submitted into a restricted lane.</li> <li>2. The user must move the card to a restricted lane so that the card can be reviewed</li> </ol>

## 4. Activity Diagram:-

Requirement ID: UR01, UR02, UR03, UR04, UR05, UR06, UR07, UR08

Use Case ID: #101

Use Case Short Description: A registered user creates a new storyboard with a swimlane in it. A new card is then created in this swimlane by supplying the title, description, tasks and label.

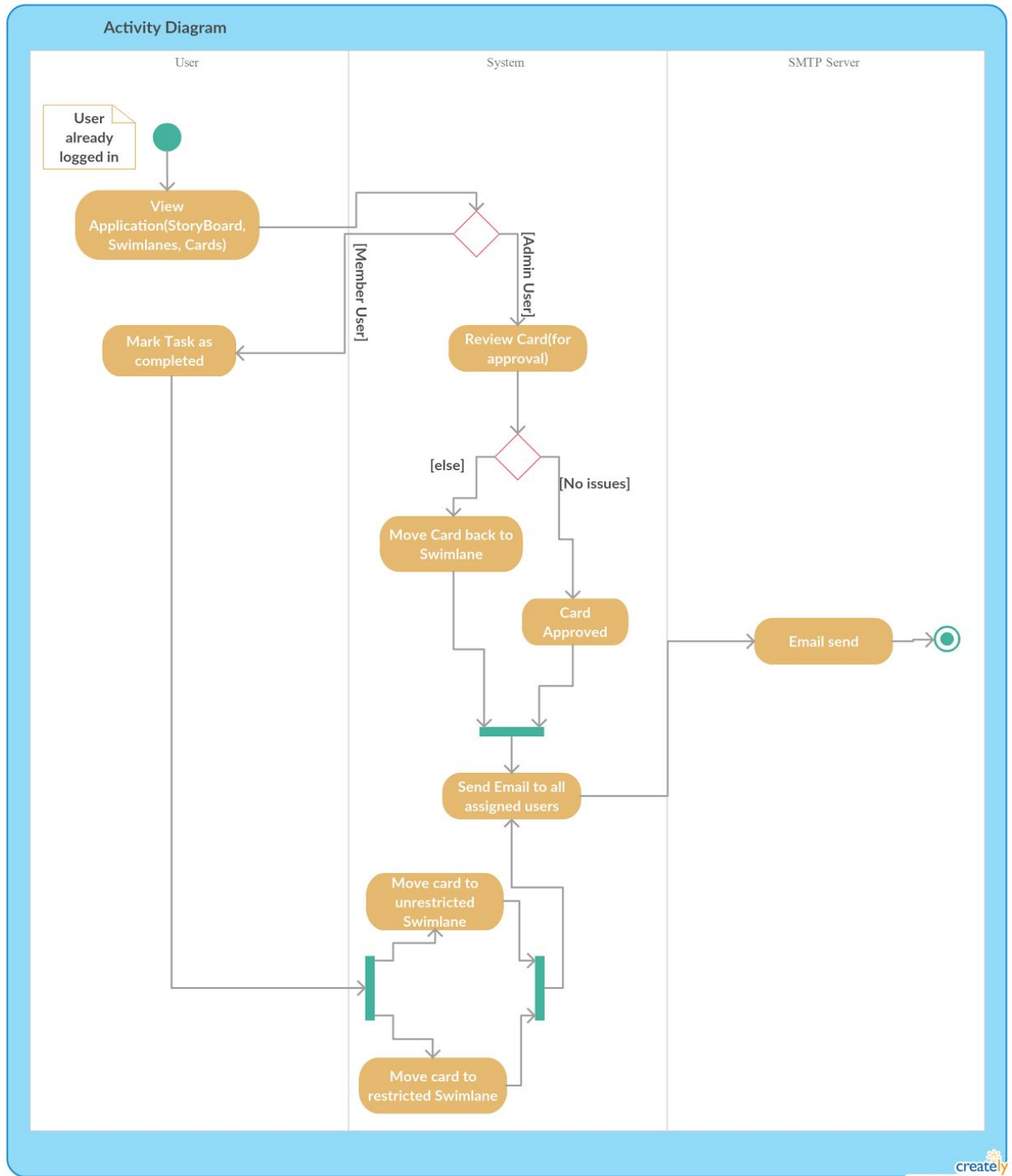




Requirement ID: UR04, UR07, UR09, UR10

Use Case ID: #102

Use Case Short Description: A member user, finishes off the quantum of work denoted by a card, update it and moves it to a restricted lane. An Admin then review the work done and approve the card by moving it out of the restricted lane.



## 5. UI Mockups:-

Browser window titled "Login" with address bar showing `https://some.domain.on.heroku`.

### Login to Prorg

Email:

Password:

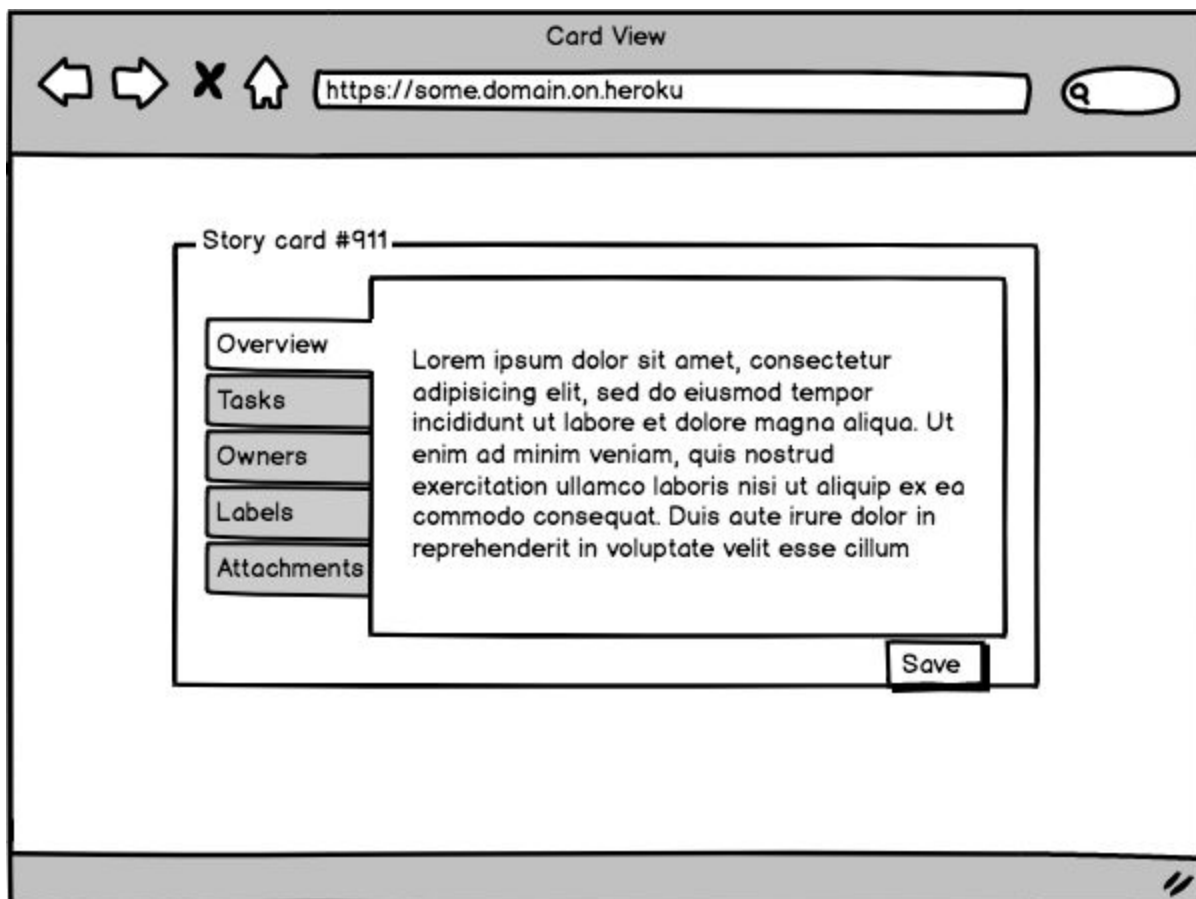
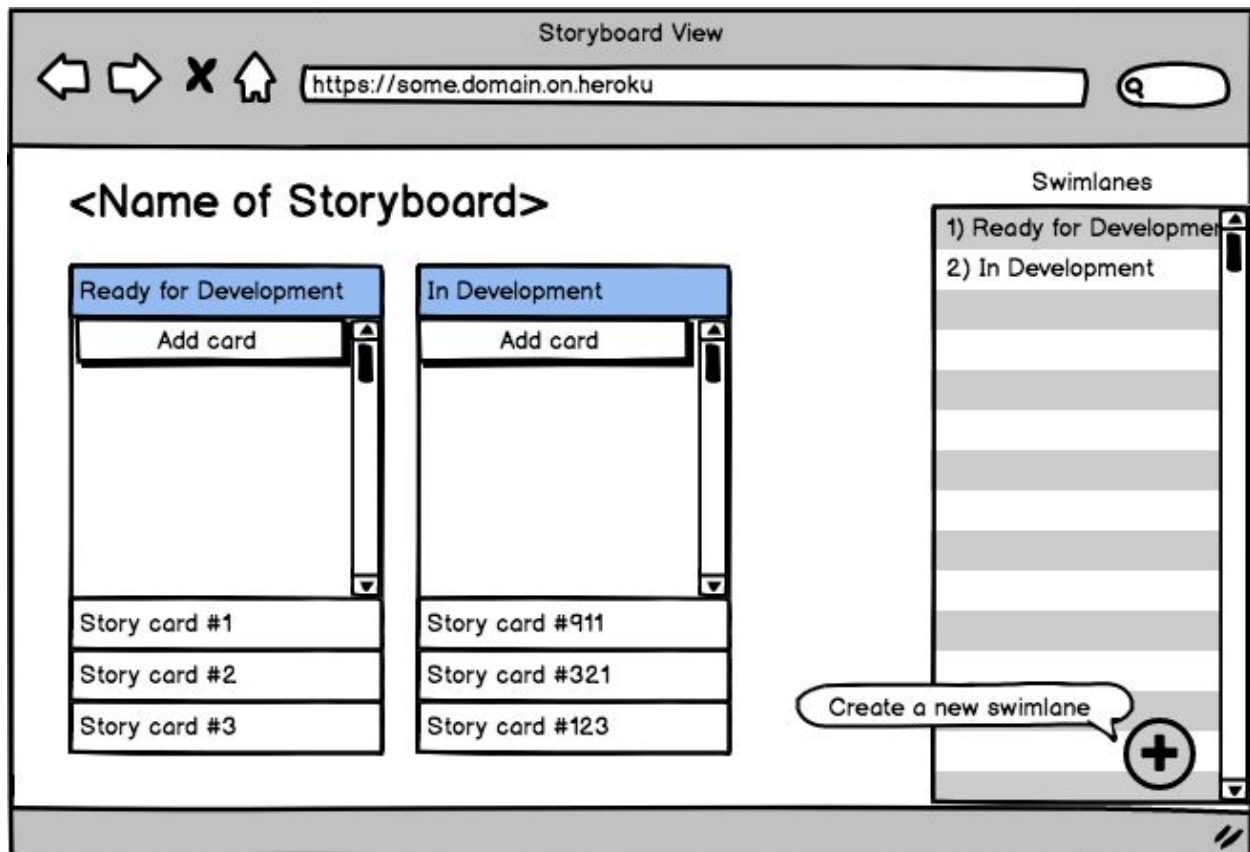
Browser window titled "Home page" with address bar showing `https://some.domain.on.heroku`.

# Welcome to Prorg!

No Storyboards available. Please create one!

~~~~~

Create a new storyboard



Card View

<https://some.domain.on.heroku>

Story card #911

Overview

Tasks

Owners

Labels

Attachments

☒ Some task you need to do

☐ Some other task that you don't like

☒ Another one!

☐ Yet another one! ARGH!

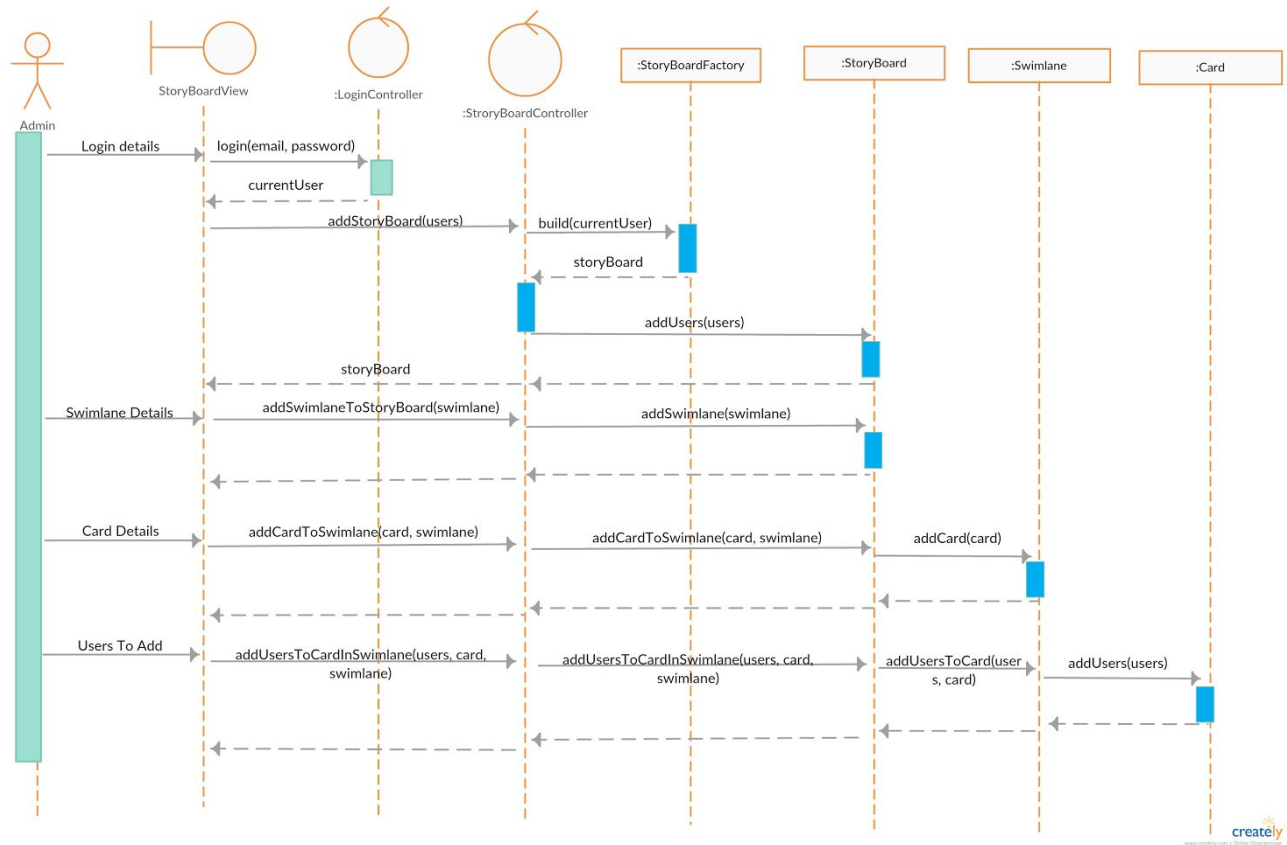
Save

## 6. User Interactions:-

Requirement ID: UR01, UR02, UR03, UR04, UR05, UR06, UR07, UR08

Use Case ID: #101

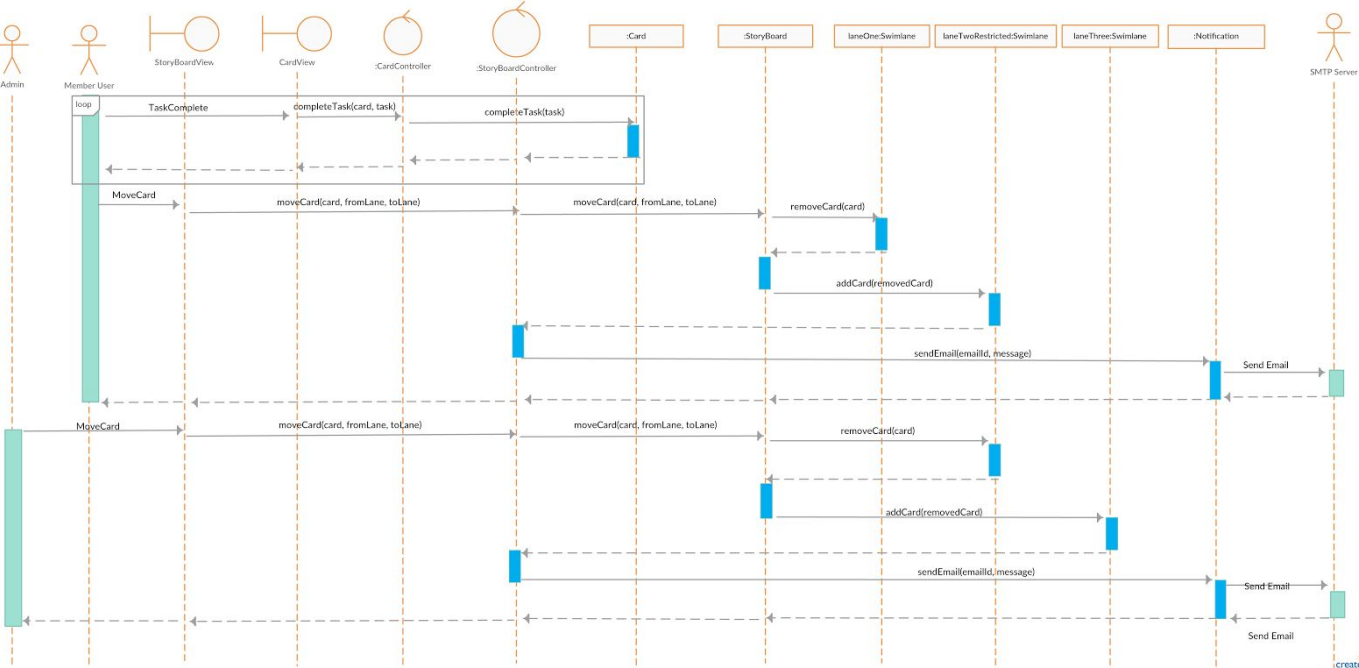
Use Case Short Description: A registered user creates a new storyboard with a swimlane in it. A new card is then created in this swimlane by supplying the title, description, tasks and label.



Requirement ID: UR04, UR07, UR09, UR10

Use Case ID: #102

**Use Case Short Description:** A member user, finishes off the quantum of work denoted by a card, update it and moves it to a restricted lane. An Admin then review the work done and approve the card by moving it out of the restricted lane.



## 7. Class Diagram:-

Class Diagram

