



POLITECNICO
MILANO 1863

DREAM - Data-dRiven PrEdictive FArMing in Telangana

*Careddu Gianmario
La Greca Michele Carlo
Zoccheddu Sara*

Prof. Elisabetta Di Nitto

Acceptance Test Document

Version 1.0

12th January 2022

Deliverable: ATD

Title: Acceptance Test Document

Authors: Careddu Gianmario
La Greca Michele Carlo
Zoccheddu Sara

Version: 1.0

Date: 12th January 2022

Download page: <https://github.com/gccianmario/Zoccheddu-LaGreca-Careddu>

Copyright: Copyright © 2022, Careddu Gianmario, La Greca Michele Carlo,
Zoccheddu Sara – All rights reserved

TABLE OF CONTENTS

1. TESTED PROJECT	4
1.1. Authors	4
1.2. Repository	4
1.3. Documents considered	4
2. INSTALLATION SETUP	5
2.1. Overview	5
2.2. Setup of environment	5
2.2.1. Installing prerequisites	5
2.2.1.1. Installing Java	5
2.2.1.2. Installing Maven	6
2.2.2. Installing XAMPP and local database instances	6
2.2.3. Installing TomEE application server	7
2.3. Running	8
3. ACCEPTANCE TEST CASES	9
3.1. Developed features	9
3.2. Issues	10
4. DOCUMENT QUALITY	16
4.1. RASD	16
4.2. DD	16
4.3. IT	17
5. EFFORT SPENT	18
6. REFERENCES	19

1. TESTED PROJECT

1.1. Authors

- Alessandro Cecchetto, 10865806
- Mattia Siriani, 10571322
- Matteo Visotto, 10608623

1.2. Repository

<https://github.com/matteovisotto/SE2-CecchettoSirianiVisotto>

1.3. Documents considered

- RASD (Requirement analysis specification document)
- DD (Design document)
- IT (Implementation and testing document)

2. INSTALLATION SETUP

2.1. Overview

The installation of the system had several issues, generated especially by a not-enough clear documentation. However, thanks to the support and help of the authors of the system, the installation proceeded in a more or less good way.

Note that it is possible to directly visualize a deployed version of the system through: <https://forum.dreamplatform.it/>. This version however has limited functionalities: in order to test the others, it is necessary to access a local version. Below, the details of the installation of this version [3].

2.2. Setup of environment

2.2.1. Installing prerequisites

2.2.1.1. Installing Java

- Download Java SE JDK 15 (by search a provider such as OracleJDK or OpenJDK, and downloading the file)

- Extract content from the zip folder to your preferred location.

Then go to the environment variables.

In the User Section select Path variable and click on Edit.

Select New and type:

`C:\Users\<your-user>\<path-to-extracted-folder>\bin`, then save and exit. Open a cmd.exe and verify the Java version by typing the following command:

```
java -version
```

2.2.1.2. Installing Maven

- Download Maven framework version 8.0 (by search online and downloading the file)
- Extract to preferred location and repeat above steps.
- Verify your Maven version by typing the following command in a cmd.exe:

```
mvn --version
```

2.2.2. Installing XAMPP and local database instances

- Xampp is useful to deploy a local database to allow the system to work properly.
- Download Xampp and install it
- Open the control panel and activate Apache and Mysql by clicking on the respective buttons.
- Go to <http://localhost/phpmyadmin> and click on import
- It is necessary to import two different databases available at the following links: <https://github.com/matteovisotto/SE2-CecchettoSirianiVisotto>, inside the folder 'DeliveryFolder'
- After the database download, the first database that has to be imported is named 'dream_data.sql'. In PHP admin, after having clicked on import, it is necessary to upload the previously download sql file, and then click on Go in the right end side of the page
- Repeat the same procedure to import 'dream-forum.sql'

2.2.3. Installing TomEE application server

- Download Apache TomEE 8.0.9 (by searching online and downloading the file)
- Extract the file in a specific folder
- Go inside this directory and then in directory conf, and open the file tomee.xml
- Before the last tag `</tomEE>` it is necessary to insert:

```
<Resource id="DreamForumDB" type="DataSource">
    JdbcDriver com.mysql.cj.jdbc.Driver
    JdbcUrl jdbc:mysql://127.0.0.1:3306/dream_forum
    Username root
    Password
    JtaManaged true
</Resource>
<Resource id="DreamDataDB" type="DataSource">
    JdbcDriver com.mysql.cj.jdbc.Driver
    JdbcUrl jdbc:mysql://127.0.0.1:3306/dream_data
    Username root
    Password
</Resource>
```

Note that it is possible to change the privileges of the database, by modifying the passwords of the 3 root users (in php admin), but then they must be changed also i the tomee.xml file, by replacing the empty string with the new password

- A library has to be added to the lib folder of the TomEE folder. This library has been provided by the authors of the code.

2.3. Running

- To run the server, it is necessary to compile all the architecture. However, due to several issues found during this phase, two .war files containing the already compiled instance of the server have been provided by the authors of the system.
- To run those files, it is necessary to put them inside the tomEE folder, specifically inside the webapps folder of the installing directory of TomEE.
- The file has to be named ROOT.war and any folder already present in the folder named ROOT has to be deleted.
- After this operation, it is necessary to open a cmd.exe inside the directory bin of the TomEE folder, and execute the command:
`startup.bat`
- By going to <http://localhost:8080>, it is possible to access to the local server
- Since there are two .war files, one related to forum functionalities and one related to policy maker functionalities, to run each .war file it is necessary to delete the already one present in the webapps folder, and copy the other one in the same folder, renaming it as ROOT.war. Note that each .war file must be runned alone, without any other ROOT.war file in the same directory [3].

3. ACCEPTANCE TEST CASES

3.1. Developed features

In the RASD document a comprehensive list of product functions is not provided, and in the IT document a list of developed features is missing. We have decided to use all use cases as a base to understand which are the features that the system should have.

Through reverse engineering of deployed artifacts we have built the following list, reporting what is developed and what generates issues.

Note: In the RASD document there is an inconsistency in the name given to the actors of the system, as we reported in the next section “Document quality”. Basically, it seems that a user is intended as a registered client that is not a policy maker and the policy maker is a registered client in quality of policy maker. For clarity, when we want to refer to any of the registered actors we use “clients” [1].

Functionality	Implemented	Issues
Filter data (visitors)	No	-
Download dataset (visitors)	No	-
Navigate the forum and download a document (visitors)	Only navigation	-
Sign up user (users)	Yes	-
Login user (users)	Yes	Yes
Publish a post (users)	Yes	Yes
Modify a post (users)	Yes	Yes
Sign up user (policy makers)	Yes	-
Login user (policy makers)	Yes	Yes
Publish a post (policy makers)	Yes	Yes
Modify a post (policy makers)	Yes	Yes
Delete a post (policy makers)	Yes	-

Create a new discussion (policy makers)	Yes	-
Delete a discussion (policy makers)	Yes	-
Confirm pending posts (policy makers)	Yes	Yes
Decline pending posts (policy makers)	Yes	-
Recalculate new deviance (policy makers)	Not Clear	-
Modify a data source (administrator)	No	-
Remove a data source (administrator)	No	-
Add new data source (administrators)	No	-

3.2. Issues

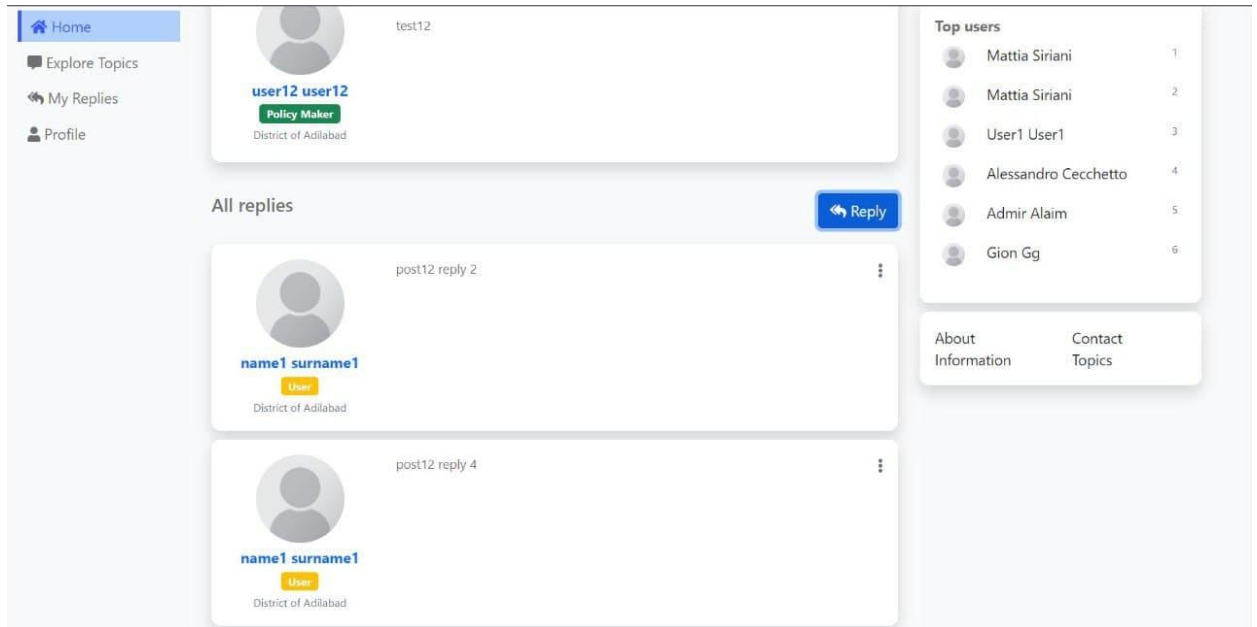
- **Login** (both users and policy makers)

After the first log in procedure, the log out is not executed correctly. In fact, the user or the policy maker's session is kept open, and cannot be closed using the frontend, forcing the clients to stay logged in and preventing new log in actions. It is not clear how the session is managed and automatically closed.

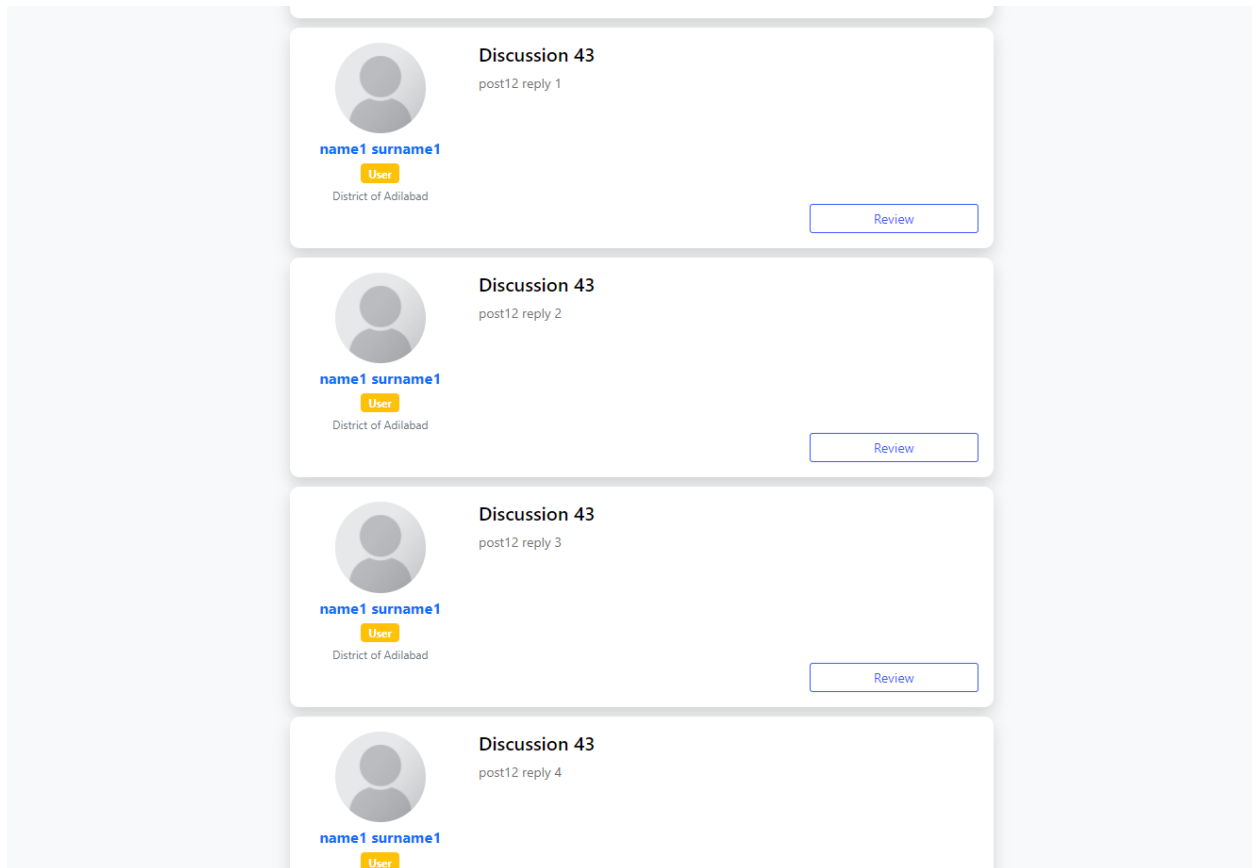
- **Publish a post** (users) and **Confirm pending posts** (policy makers)

When a user tries to publish a post in a discussion, the post is correctly sent to policy makers for review, however if the same user tries to post repeatedly under the same discussion the posts are alternatingly displayed directly in the frontend, even if they are under review. For instance if a user tries to post 4 posts under a discussion, all the 4 posts are sent to policy makers for review but the 2nd and 4th post are shown in the frontend.

User's point of view after publication of "post 12 reply 1", "post 12 reply 2", "post 12 reply 3", "post 12 reply 4": the posts "post 12 reply 2", "post 12 reply 4" are directly posted without the review, even after a page refresh the situation doesn't change.



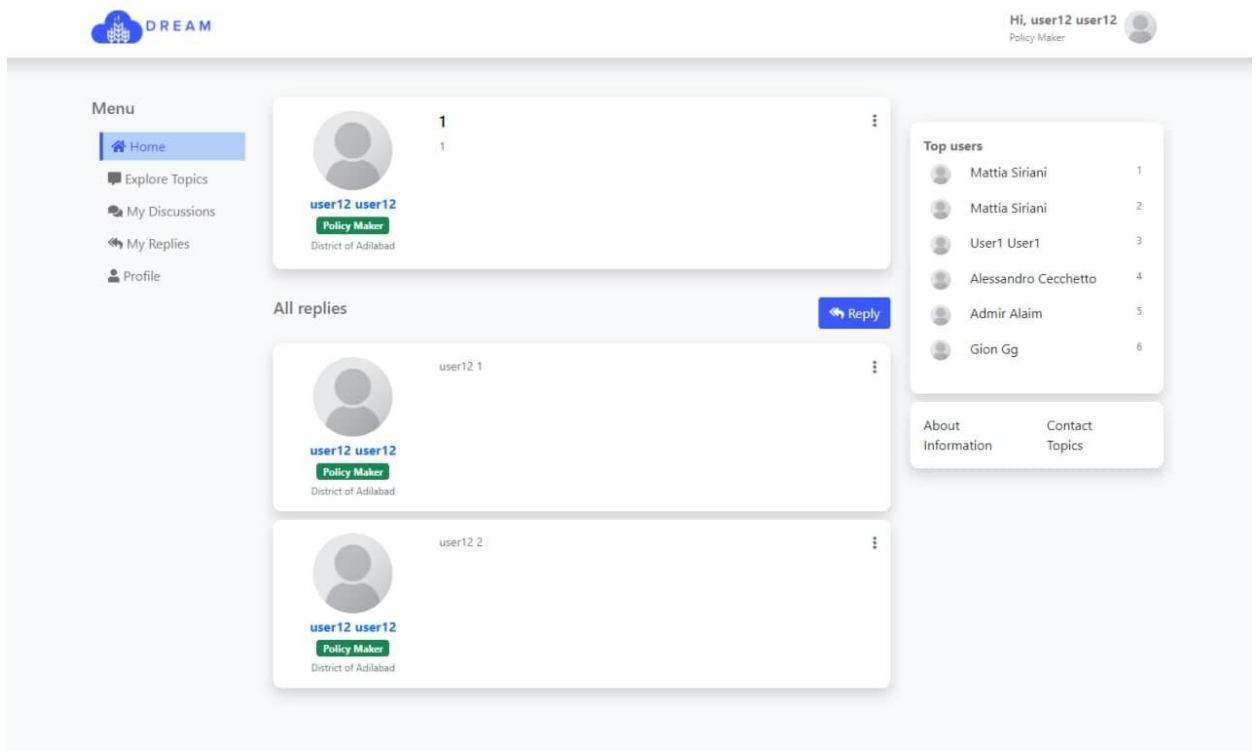
Policy maker's point of view: all the posts are marked as they need the review.



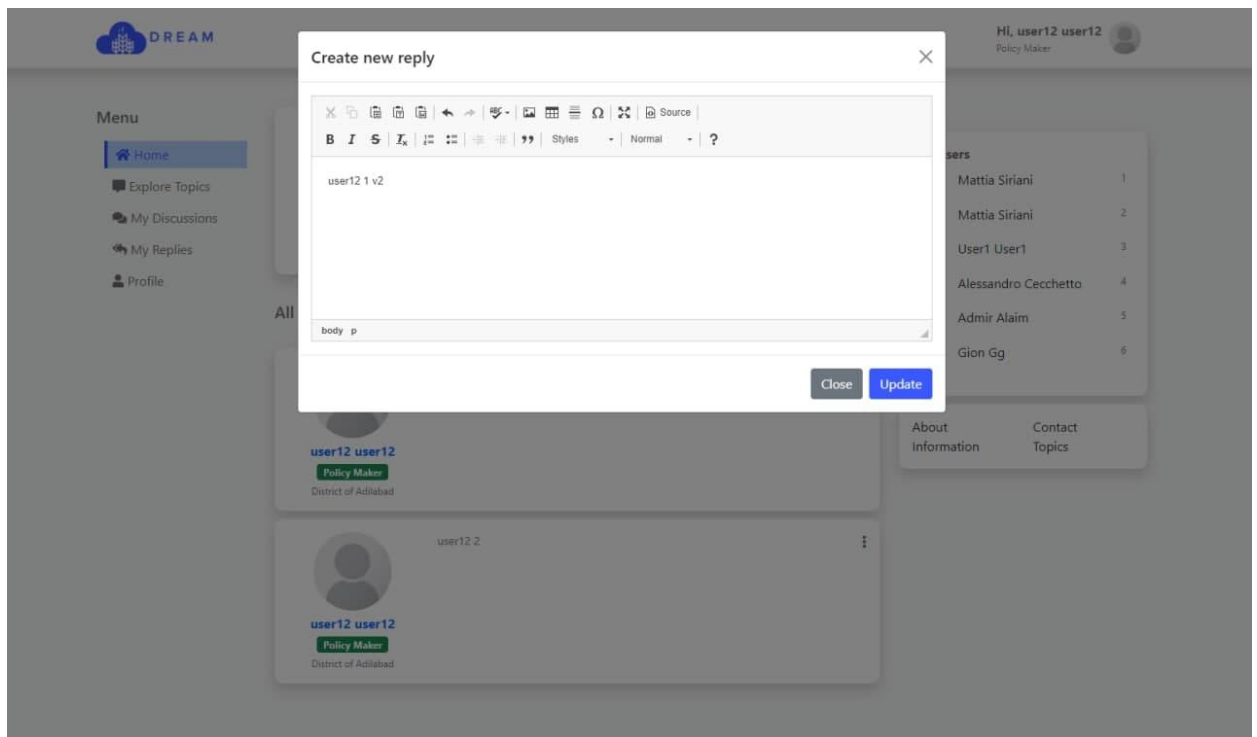
- **Modify a post** (both users and policy makers)

After a post modification by any client, if the client tries to create a new post the most recently modified post is overwritten by the new post. After this bug the system enters in a loop where every new attempt to create a post overwrites the same previously modified post.

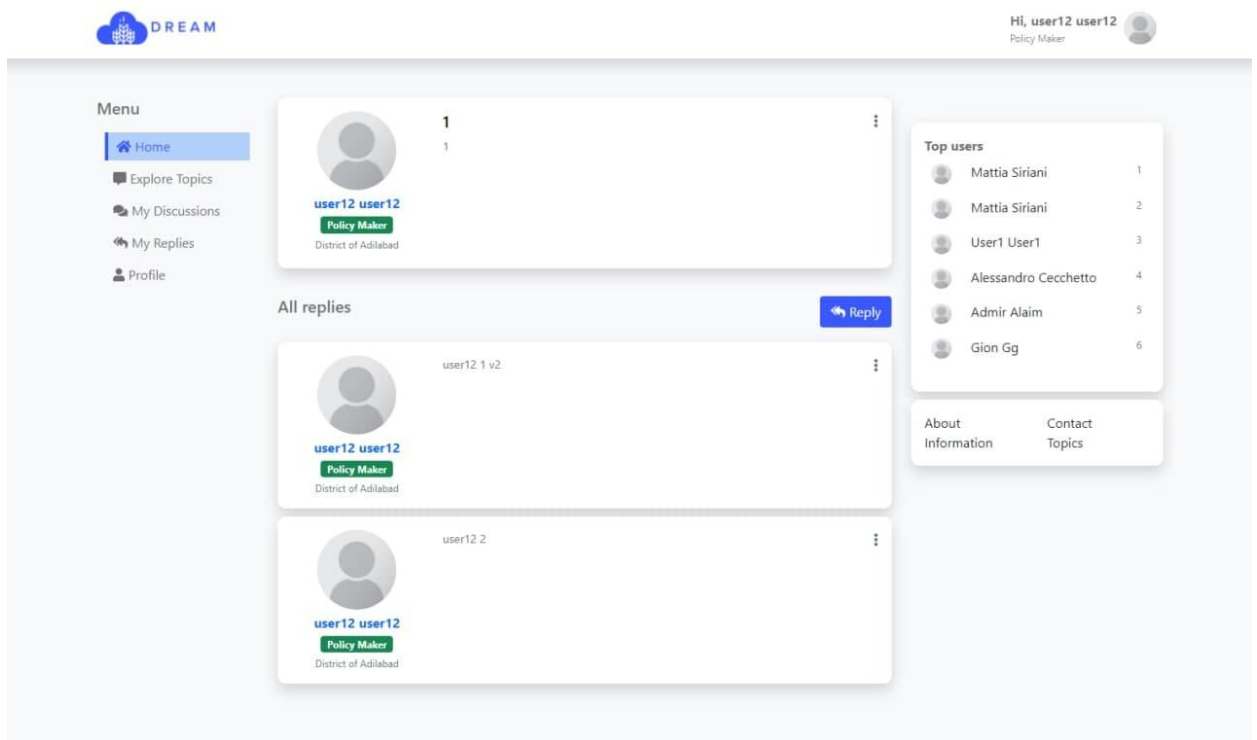
Post “user 12 1” and “user 12 2” are shown



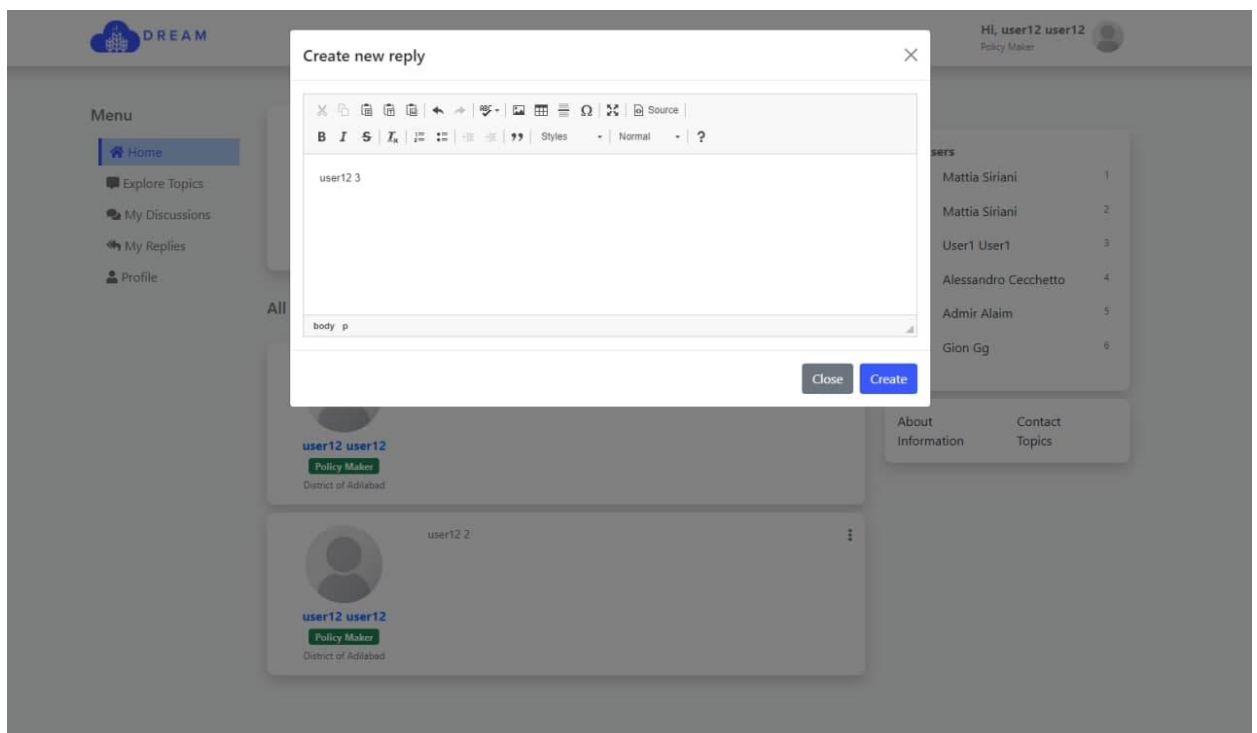
Post “user 12 1” is modified in “user 12 1 v2”



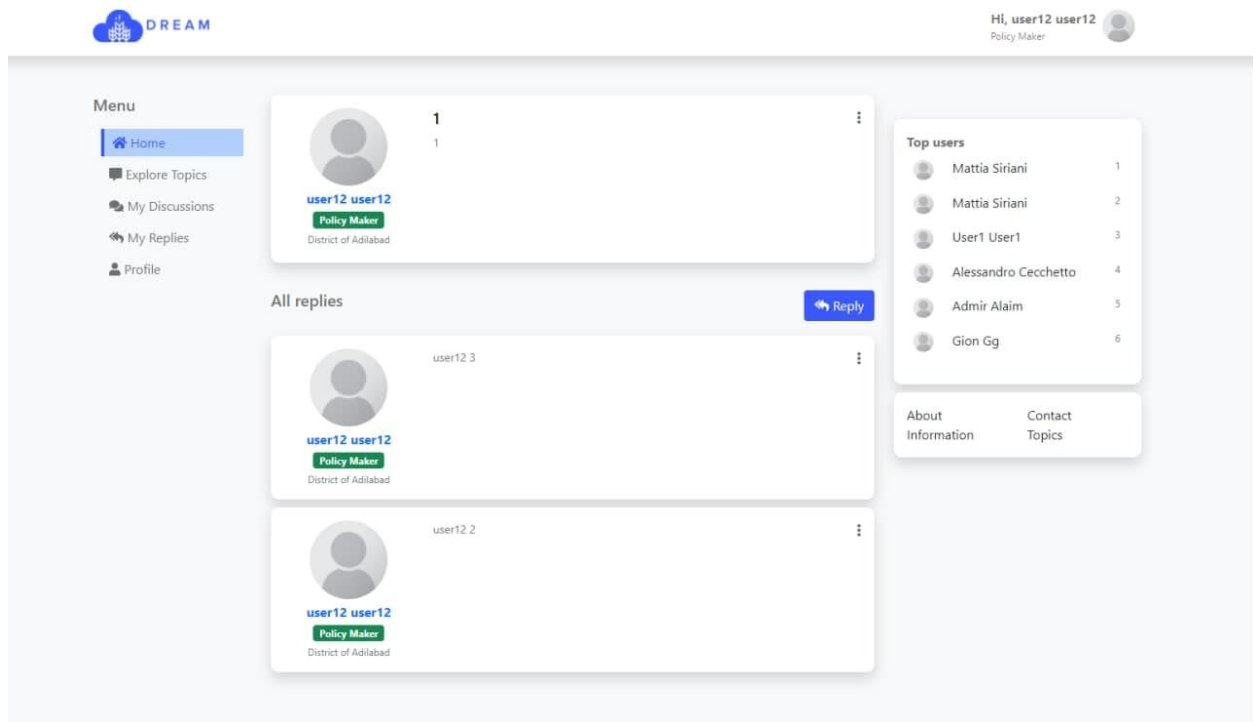
The modification correctly shows up



Attempt to insert a new a new post “user 12 3”



The modified post “user 12 v2” is overwritten and becomes “user 12 3”



4. DOCUMENT QUALITY

4.1. RASD

World phenomena seems confused with product functions; also, goals are not general and refer to some specific features that the system is required to have. There is not a clear definition of what the various users are: in the definition section, users, visitors, farmers, policy makers, moderators and administrators are defined but in the actors paragraph only user, policy makers, visitors and administrators are nominated.

In the class diagram only user, and policy maker as user's extension, are defined. The figure of the agronomist present in the specification document is never mentioned. The class diagram lacks clarity, some relations are without verbs and multiplicity.

The product functions section is bare and incomplete, and a lot of what are supposed to be product functions are presented in an obscure way inside the domain assumptions and world phenomena.

Domain assumptions are not appropriate, they represent a broad description of what the user can do using the system. In section 2.4.2 some implementation details are presented, but they are not supposed to be in the RASD document.

In section 3.5.3 the lower bound for availability is set to 90% for the forum: this is a questionable decision since in the worst case can imply unavailability of the forum for up to 35 days.

The formal analysis with alloy is presented without any comment or clarification in what has been proved using the formal analysis, but at least some comments inside the code are provided [1].

4.2. DD

The DD is clear and touches and clarifies all the working principles of the system that are missed in the RASD. A good description of the data managed by the system is provided together with a detailed site map [2].

4.3. IT

The implemented functionalities are not listed, and it is never said what has been developed. This would have been important to check how the requirements stated in the RASD have been met in the implementation, as well as to test those functionalities. The test coverage results are not commented, and reading the numbers it looks like that the majority of the code has not been tested. How to manage the DB dump (and its existence) is never mentioned: we had to figure it out. Also no mock credentials are given and not even instructions on how to access the system. During the setup it is not clear what is mandatory and what not, windows and linux setup information are mixed together.

The whole tutorial is imprecise and some important steps are omitted. In the final part of the document, big snippets of code are presented without explanation of what they are used for.

This document fails on its purpose, the developed system after its reading is still a black box [3].

5. EFFORT SPENT

	Total time
Careddu Gianmario	9h
La Greca Michele Carlo	9h
Zoccheddu Sara	9h

6. REFERENCES

1. https://github.com/matteovisotto/SE2-CecchettoSirianiVisotto/blob/main/DeliveryFolder/RASD_v1.2.pdf
2. https://github.com/matteovisotto/SE2-CecchettoSirianiVisotto/blob/main/DeliveryFolder/DD_v1.1.pdf
3. https://github.com/matteovisotto/SE2-CecchettoSirianiVisotto/blob/main/DeliveryFolder/IT_v1.0.pdf