Jorge Blanco Rey
Ángel Casanova Bienzobas
S.Xiao Fernández Marín

jorge.blancor@estudiante.uam.es
angel.casanova@estudiante.uam.es
sofiax.fernandez@estudiante.uam.es

1. Introduction

1.1 Purpose of the system

The application "Citicide" allows citizens to create and vote Infrastructure and Social projects, so they can be part of the decisions which are made to improve their countries and solve international problems. Citizens, in the following lines called users, can vote individually and be part of different groups for supporting all together a project.
This is a very useful application, since it provides users the freedom to propose and vote different projects, and by this way they have the feeling of being part of the decisions that are made around their lives.

1.2 Scope of the system

The application "Citicide" should be able to hold on online a lot of people from all around the world.
It should be capable of letting the users create as many groups as they want, but not letting them to be at a subgroup if they are already on a group that is above or below that one. The user must be registered if he wants to access the platform. Also, he can log in and log out as many times as he wants but if he fails on entering the password 5 times, he will need to change it.

Desktop App. No networking, etc. Ok!, etc. Try to reduce the scope of the system

1.3 Objectives and success criteria of the project

The success of the application depends upon meeting the following core set of objectives:

- Users must have an easy experience for signing up.
- When logging in, users should be capable to create/join any group, not for subgroups (as explained before).
- Apart from groups, this program also aims to generate real projects that will be send to the external organization, asking for a welfare fund.
- As we believe in a democratic application, we want that everyone can vote for a project, also a group owner can vote in name of the whole group.

Our main goal is to create a steady system, in which the future enhance tasks would be developed in the best way, also we desire to create a system capable to support different modules including a GUI, where all possible users would feel comfortable.

In order to do so, we will test each module separately and at the end we are going to do a deep unitary test for all the application, with all those tests we will ensure the proper functioning of the whole system.

* try to

Avoid ambiguous objectives (easy, ~~fast~~ etc)

Jorge Blanco Rey | jorge.blancor@estudiante.uam.es
Ángel Casanova Bienzobas | angel.casanova@estudiante.uam.es
S.Xiao Fernández Marín | sofiax.fernandez@estudiante.uam.es

1.4 Definitions, Acronyms, and abbreviations

Administrator Person who controls what happens in the application. He is the responsible of admitting or rejecting new projects and determine how many votes does they need to be sent to the external organization. He can also ban users giving them a reason and unban them as well.

User Person that has created an account on "Citicide". There are two types of users:
- *Manager*: Creator of a group.
- *Proponent*: Creator of a project.

Ban Action that the administrator can take on a user that has not behave properly in agreement with the rules of the application.

Group A collective of users that has a topic in common. A user can subscribe a group if it is of his interest, as well as he can unsubscribe. A manager of a group can create a project. That project will be supported by all the members of the group. The manager can also vote for a project that he thinks the group will be in favor of.

Subgroup Groups inside other groups. If a user wants to be part of more than one subgroup, he can only pick subgroups that are on the same depth level.

Project An idea that a user or a group has, and they want to accomplish it. There are two types of projects:
- *Infrastructure*: Construction that citizens want to be built in a certain district.
- *Social project*: Proposal that can be national or international, for helping a specific society group.

Votes Numbers of users in favor of that project for taking it as a proposal for an external organization.

Reports There are two types of reports:
- *Popularity Reports*: Contains the number of votes that a project has. It can only be seen by the users who have already voted.
- *Affinity Reports*: Number of common projects that a group has with another one. This report is calculated by doing the number of people that are in each group and support the projects founded by the other group, and the other way around. Then divide this number by the total of users in both groups. (N12+N21)/(N1+N2)
  N12: People who are in group N1 and support projects of group N2.
  N21: People who are in group N2 and support projects of group N1.
  N1: Total number of users in group N1.
  N2: Total number of users in group N2.

Jorge Blanco Rey | jorge.blancor@estudiante.uam.es
Ángel Casanova Bienzobas | angel.casanova@estudiante.uam.es
S.Xiao Fernández Marín | sofiax.fernandez@estudiante.uam.es

2. System Description

2.1 Functional Requirements

- The user should be registered to enter the platform. If he is not, he can only register.
- The only person who can ban a user or approves projects requested by other users is the administrator of the application.
- The system should be able to register a person just once.
- The application must hide the information of the users. Users can only see the groups other users are in.
- Groups are done by a user. Once a user has created a group, he becomes the manager of the group. He, and only he, can create another subgroup (there is no limit for how many subgroups can be created).
- A user can be the manager of several groups (there is no limit).
- There is no chat in the group.
- It should be capable of letting the users create as many groups as they want, but not letting them to be at a subgroup if they are already on a group that is above that one.
- The manager of a group cannot delete another user from the group. The users can only unsubscribe from them.
- If a manager of a group gets banned nothing happens. He will still be in their group description, because he could be unbanned in the future.
- When a project created from a group or from a user gets approved or rejected you get a notification. The manager will always receive the notifications. The members of the group should turn them on if they want.
- The user who creates a project becomes the proponent.
- Users must create projects that will help the community.
- There are two types of projects: Infrastructure and Social projects.
- Votes can be individual or done by a manager of a group. If it is done as a manager, he must specify which group is he voting for.
- Once a user votes on a project as an individual or as manager, he cannot remove the vote.
- If a user is in a group and the manager votes for a project, he can leave the group for removing his vote.
- If a user wants to see the number of votes that a project has, he needs to have voted that project.
- If a user gets banned, its votes are removed from all the projects where he has performed this action.
-There are two types of reports: Popularity reports, can only be seen if the user is supporting the project, the report contains only the number of votes; and Affinity Reports, are the number of common projects supported by groups, only users who belongs to both groups can see them.
- The user can search projects (the owner can check the status with the ID once it is founded) and groups by name.

Jorge Blanco Rey | jorge.blancor@estudiante.uam.es
Ángel Casanova Bienzobas | angel.casanova@estudiante.uam.es
S.Xiao Fernández Marín | sofiax.fernandez@estudiante.uam.es

2.2 Non-functional Requirements

- **Operational:** Saving the information of the application in a server: Every 30 seconds.
- **Security:** A user can log in and log out as many times as he wants, but if he fails on entering the password 5 times on a row, he will need to change it.
- **Resources:** Memory (Database): 100 Terabyte.
- **Performance:** Response time for every button: less than 0,5 ms.

- Maximum number of users: 1,95 billion. (too big for this app!:)
- Text: Must be clear and visible from at least 30cm. ✓
- Registration: The user must provide an ID, their ID number (with 9 characters) and a password. Functional
- Users can have a state. Functional.
- Users only can see the IDs of other users, but not their ID number or password. Functional.
- Group (public): It must have a name, the manager of that group and can have a description. Functional
- Projects:
  · Infrastructure: It must have a title of 25 char maximum (there can be projects with the same name), a description with 500 maximum, district, a drawing, the minimum number of votes, amount of money requested and the amount of money granted after having the minimum votes. And if the project has been rejected or accepted. Functional.
  · Social Projects: Specify if they are national or international, title of 25 chars max, a description of 500 characters max, the group of people the project is addressed to, with a maximum of 50 characters, the minimum number of votes, amount of money requested and the amount of money granted after having the minimum votes. And if the project has been rejected or accepted. Functional.
- The minimum number of votes for approving a project is set by the administrator. After that it cannot be changed. Functional.
- The state of a project can be checked by the proponent with its ID (unique). Functional.
- If a project is not approved by the administrator, he can give a reason using 50 characters. Functional
- If a project does not have the minimum number of votes to get founded for 30 days since it is approved by the administrator, it gets expired. Functional.
- The minimum number of votes of a project can only be seen by the manager of a group or individual users, once they have vote it. Functional.
- When a project receives the minimum number of votes, the representative of the project submits it to an external organization, the proponent receives an ID and with it he can check the state of the project (rejected, approved, …). Functional.
- Votes are dynamic, if a group is supporting a project and it is added a new member it must be summed a vote, and the other way around, if someone unsubscribes or it is banned (same approach for individual votes), its vote is removed. Functional
- The Affinity Reports are calculated by doing the number of people that are in each group and support the projects founded by the other group, and the other way around. Then divide this number by the total number of users in both groups. Functional

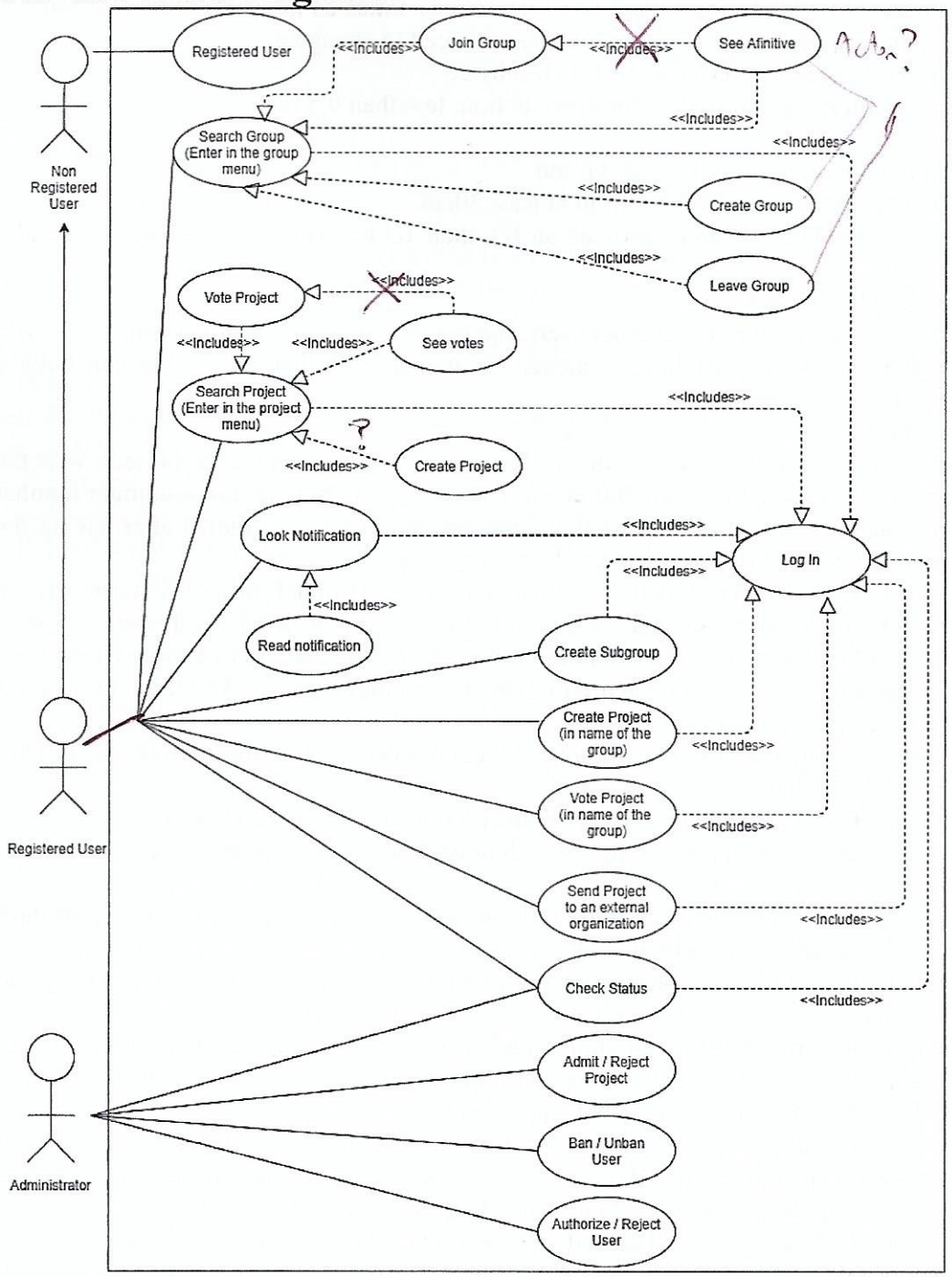
Few non-functional, and most are functional, actually

Jorge Blanco Rey
Ángel Casanova Bienzobas
S.Xiao Fernández Marín

jorge.blancor@estudiante.uam.es
angel.casanova@estudiante.uam.es
sofiax.fernandez@estudiante.uam.es

3. Use Cases

3.1 Use Case diagram



3.2 Use case descriptions

Use Case 3.2.1: Search Group.

Primary actor: Registered User.

Stakeholders and goals:

- **Registered User**: To enter in the "Group" menu interface, where he can make all the actions related with a group, as individual or if he is a group owner, he can access to all the functions of the group, taking those actions in name of the whole group.
- **Administrator**: He can check the behavior of the groups to decide whether ban a user or not.

Preconditions: The user is identified by the log in use case.

Success Guarantee (Postconditions): The user could display all the groups that he belongs to, also he could search a new group to log in and access any possible action related with groups management through this menu.

Main Success Scenario:

1. The user selects the "Groups" tab.
2. The user selects a group (if any) or search a new one.
3. The user can join the group.
4. The user displays the information of the group.
5. The user may look the Affinity Report associated with this group.
6. The user can leave the group.

Extensions (Alternative paths):

I. The user joins a new group.
   a) The system decides if the user can enter in the group or not.
   b) The system displays a new group in the current groups list.

II. The user leaves a group.
   a) The system erases a group from the current group list.

III. The user creates a new group.
   a) The user fulfills the information about the group.
   b) The system promotes the user from register user to group owner.
   c) The group is included in the data base and it is ready to anyone that wants to join in.

IV. The user can see the affinity between two groups (only if he belongs to both groups).
   a) The system displays the affinity coefficient or an error.

Technology and Data Variations List:
- Option to search new groups.
- Option to display data about the groups that the user belongs to.
- Option to display the Affinity Report.
- Option to join/leave groups.

Frequency:
- Very high, in order of hundreds per current users.

Open Issues:
- In a future version the group may be expanded, so it is possible to append new extensions to this main menu.

Use Case 3.2.2: Search Project.

Primary actor: Registered User.

Stakeholders and goals:
- **Registered User**: To enter in the "Project" menu interface, where he can make all the actions related with a project, as individual, or if he is the proponent of a project, he can send the project to an external organization, then he can check the status of the project by an ID given by the this organization.
- **Administrator:** Must accept or reject a new project, when it is created, also he must decide the minimum number of votes that the project needs in order to be send to the organization.

Preconditions: The user is identified by the log in use case.

Success Guarantee (Postconditions): The user could display all the projects that he has voted, also he could search new project to vote and access any possible action related with project management through this menu (such as see the vote of the project).

Main Success Scenario:
1. The user selects the "Projects" tab.
2. The user selects any project (if any) or search a new one.
3. The user can vote for a project.
4. The user displays the information of the project.
5. The user may look the project report (number of votes) if he has vote for this project as individual.
6. The user can create a new project.

different use cases

Extensions (Alternative paths):

I. The user votes for a project.
   a) The system updates the number of votes.

II. The user creates a new project.
   a) The user fulfills the information about the project.
   b) The system sends a message to an administrator to check the project.
   c) The system promotes the user from register user to project owner.
   d) The project is included in the data base and it is ready for anyone that wants to vote for it.

III. The user can see number of votes of the project if he has voted for the it as an individual.
   a) The system displays the number of votes or an error message.

Technology and Data Variations List:

- Option to search new projects.
- Option to display data about the project that the user has vote for.
- Option to display the project report.
- Option to vote a project.

Frequency:

- Median, in order of tens per current users.

Open Issues:

- In a future version the group may be expanded, so it is possible to append new extensions to this main menu.

Use Case 3.2.3: Create a Project (in name of a group).

Primary actor: Registered User.

Stakeholders and goals:

- **Register User**: He must be a project owner, if not, an error pop-up will show a warning to the user. If he is a group owner, he will enter in the "Project" menu interface, where he can make all the actions related with a project, as a collective leader and also he can send the project to an external organization, then he can check the status of the project by an ID given by this organization, as a normal user.
- **Administrator:** Must accept or reject a new project, when it is created, he must also decide the minimum number of votes that the project needs in order to be sent to the organization.

Preconditions: The user is identified by the log in use case, and he must be the owner of at least one group.

Success Guarantee (Postconditions): The user creates a project, in name of a group that he owns, so the project will have the name of the group and will start with a number of votes equal to the number of users that the group has.

Main Success Scenario:

1. The group owner presses the create project button.
2. The group owner selects the "as group owner" option.
3. The group owner fulfills the information about the project.
4. The group owner waits for the administrator to accept the project.

Extensions (Alternative paths):

- In this case, they are not any extension to the use case.

Technology and Data Variations List:

- Option to create a project as collective.

Frequency:

- Very low, in order of one quarter per current users.

Open Issues:

- We do not think on future improvements in this use case.