Functional requirements:

1. The application shall have three types of users: unregistered citizens, registered citizens, and administrator.
2. Unregistered citizens can only register in the application, providing a name, an ID and a password.
3. The name must be unique in the application.
4. The ID must be unique in the application and valid (nine characters).
5. Registered citizens can log in to the application providing their names or IDs and their passwords.
6. A registered citizen can create a group becoming its representative.
7. The group shall have a unique name in the application.
8. A representative of a group can create an empty group inside it.
9. A registered citizen can join a group if he/she does not represent it, does not belong to it, a parent or a child group.
10. A registered citizen can leave a group.
11. A registered citizen can propose a project individually becoming its proponent.
12. A registered citizen can propose a project as a representative of a group becoming this group its proponent.
13. The projects shall have a title (max. 25 characters), description (max. 500 characters), requested budget, granted budget and proponent.
14. There can be infrastructural and social projects.
15. Infrastructural projects shall have a graphical scheme and a list of neighborhoods.
16. Social projects shall have a social group (max. 50 characters) and national or international aim.
17. The proponent of a project (citizen or group) automatically votes it.
18. A registered citizen can individually support a project.
19. The support of a citizen to a project sums one vote.
20. A representative can give the support of his/her groups to a project.
21. The support of a group sums as many votes as members have it.
22. Every time a project receives new votes, the support of this project shall be updated.
23. Every time the support of a project is updated, the duplicated votes shall be removed.
24. A registered citizen can subscribe to a project.
25. A citizen subscribed to a project shall receive notifications about its conversion to fundable, acceptance or rejection, and expiration.
26. The proponent of a project automatically subscribes to this project.
27. A registered citizen can read the projects that he/she has voted and can ask for popularity reports of them.
28. A representative of a group can read the projects the group has voted and can ask for popularity reports of them.
29. The popularity of a project is the number of votes that has received, considering individual and group votes and removing duplicated ones.
30. A registered citizen can read the groups that he/she belongs to and can ask for an affinity report between two of them.
31. The affinity among two groups A and B is computed as the number of projects of A supported by B plus the number of projects of B supported by A divided by the total number of projects.
32. A proponent can request funding for his/her project when it has certain support.

33. A proponent can read if the funding has been accepted or rejected.
34. An external application will receive the projects and answer if they are accepted (with the granted budget) or rejected.
35. The administrator of the application shall be unique.
36. The administrator shall log in to the application with a predefined name and password.
37. The administrator can accept or reject a project giving a reason of 50 characters.
38. If the project is rejected, it does not appear in the app.
39. If the project is accepted, it can be voted.
40. The administrator can ban and unban registered citizens.
41. Banned citizens cannot create groups, create projects and vote projects.
42. The vote of banned citizens shall not be counted.
43. The administrator can set a threshold of votes required to fund the projects.
44. The application shall give a unique ID and date to every project.
45. The application shall remove the project rejected by the administrator.
46. The application shall show the rejected projects with the reason to the proponent in the next access.
47. The application shall mark a project as fundable when it reaches the threshold of votes.
48. The proponent of a project can ask for funding and the project can still be voted.
49. A project can expire when it does not receive votes for 30 days.
50. Expired projects cannot receive new votes and cannot be funded.
51. The application shall show expired projects.
52. The application shall manage exceptions to show errors when a funding request is not answered due to connection failure or wrong data.
53. Wrong data can be an empty field or a negative budget.

Non-functional requirements:

1. The application shall work on a desktop computer.
2. The application shall work on one operating system: Windows, Linux, Mac…
3. The application shall be developed on Java/Swing.
4. The application shall consider an unlimited number of users (1,000 for tests).
5. The application shall consider an unlimited number of projects and groups (100 for tests).
6. The application shall be persistent: data must be saved when closing it and loaded when opening it again.
7. The application shall manage simulated dates to facilitate tests.