|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Class  Method  Requirement | MySystem | | | | | | | Project | | | | | | | | | |
| update(g:  Group) | update( p: Project) | update(  u:  User) | get  Instance() | logIn  (ni:String, ps: String) | read  () | write  () | update(g:  Group) | update(u:  User) | askFor  Money  Grante() | deletePro  ject() | checkNumberOf  Votes  () | vote  User  () | vote  Group  () | un  vote  Pro  ject() | send  Exter  nal  Asocciation() | is  Expired() |
| The user must be registered |  |  |  |  | X |  |  |  |  |  |  |  |  |  |  |  |  |
| Only the administrator can ban or unban |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Only the administrator can reject a petition to join the desktop app |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| A person can just register once |  |  |  |  | X |  |  |  |  |  |  |  |  |  |  |  |  |
| Users can only see the groups  other users are in |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Groups are created by users. Then are managers |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Only the manager of a group can create a subgroup |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| A user can be manager of several groups |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Users are let to be in several subgroups if they are siblings |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Class  Method  Requirement | MySystem | | | | | | | Project | | | | | | | | | |
| update(g:  Group) | update( p: Project) | update(  u:  User) | get  Instance() | logIn  (ni:String, ps: String) | read  () | write  () | update(g:  Group) | update(u:  User) | askFor  Money  Grante() | deletePro  ject() | checkNumberOf  Votes  () | vote  User  () | vote  Group  () | un  vote  Pro  ject() | send  Exter  nal  Asocciation() | is  Expired() |
| Users can unsubscribe from a group |  |  |  |  |  |  |  | X | X |  |  |  |  |  | X |  |  |
| If a user wants to receive a notification he will have to turn it on |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| A user receives a notification if a project created by a group he is in, is approved or rejected |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| A manager of a group will always receive notifications |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| A project is created by a user |  | X |  |  |  |  |  |  | X |  |  |  | X |  |  |  |  |
| The system adds an ID to the project |  | X |  |  |  |  |  |  |  |  |  |  |  |  |  | X | X |
| The administrator can reject or approve a project |  | X |  |  |  |  |  |  |  |  |  |  |  |  |  |  | X |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Class  Method  Requirement | MySystem | | | | | | | Project | | | | | | | | | |
| update(g:  Group) | update( p: Project) | update(  u:  User) | get  Instance() | logIn  (ni:String, ps: String) | read  () | write  () | update(g:  Group) | update(u:  User) | askFor  Money  Grante() | deletePro  ject() | checkNumberOf  Votes  () | vote  User  () | vote  Group  () | un  vote  Pro  ject() | send  Exter  nal  Asocciation() | is  Expired() |
| When accepted, the administrator should set the minimum number of votes |  | X |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| A project expires in 30 days since it is approved |  | X |  |  |  |  |  |  |  |  |  |  |  |  |  |  | X |
| The project manager submits the project for founding |  | X |  |  |  |  |  |  |  |  |  |  |  |  |  | X |  |
| Votes can be done as an individual |  |  |  |  |  |  |  |  | X |  |  |  | X |  |  |  |  |
| Votes can be done as a manager of a group |  |  |  |  |  |  |  | X |  |  |  |  |  | X |  |  |  |
| If a user votes as a manager, he must specify which group he is voting for |  |  |  |  |  |  |  | X | X |  |  |  | X | X |  |  |  |
| If a user votes as an individual, he cannot unvote |  |  |  |  |  |  |  |  | X |  |  |  | X |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Class  Method  Requirement | MySystem | | | | | | | Project | | | | | | | | | |
| update(g:  Group) | update( p: Project) | update(  u:  User) | get  Instance() | logIn  (ni:String, ps: String) | read  () | write  () | update(g:  Group) | update(u:  User) | askFor  Money  Grante() | deletePro  ject() | checkNumberOf  Votes  () | vote  User  () | vote  Group  () | un  vote  Pro  ject() | send  Exter  nal  Asocciation() | is  Expired() |
| If a user is in a group and he unsubscribes from it, and that group voted on a project, his vote will be removed |  |  |  |  |  |  |  | X | X |  |  |  |  |  | X |  |  |
| If a user wants to see the number of votes that a project has, he needs to have voted that  project |  |  |  |  |  |  |  |  | X |  |  | X | X |  |  |  |  |
| If a user gets banned, its votes are removed | X | X | X |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| It exists popularity reports |  |  |  |  |  |  |  |  |  |  |  | X |  |  |  |  |  |
| It exists affinity reports |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |