DESCRIPTION OF THE SYSTEM:  
The main purpose of our project is to provide a useful system for creating, managing and coordinate some working group that consist of some user that have to realize a project. The administrator of the system or rather the boss of the company has his own administrator panel where he can add new user or create a new working group specifying also his dimension in terms of working user. The administrator will receive some application from the user for the team previously created and he can accept or discard them based on the number of works on which the user already work or other information. The user instead accessing is own panel can visualize on which job he/she is already doing and the suggested working group (not full) that it may want to join. A user can report the completion of his work for a team. If a user wants to join a new working group he/she can send an application to the administrator and wait for his/her acceptance.

USE CASE:

*USER:*

* A user can login to the system using a login with the credential given by his/her service in order to visualize the eventually work in progress or send a join team request.
* A user can see in the application all its work in progress with the information about the working group, it’s passed work and the not full working group suggested
* A user can mark his/her part of the work for a specific working group as complete
* A user can send an application for being part of a suggested working group
* A user that is the leader of a working group can see the application from other users to the working group that he/she led.
* A user that is the leader of a working group can accept or discard the application from other users to the working group that he/she led.

*ADMINISTRATOR:*

* An administrator can login to the system using a login service.
* An administrator can add a new user putting the personal information in a sing-up service obtaining the username that will allow them to use the application.
* An administrator can create a new team setting its name and its dimension (number of users allowed) and elect the leader of the new group.
* An administrator can see all the user of the application
* An administrator can delete a user from the system

*NON-FUNCTIONAL REQUIREMENT:*

* Quality: every time an error occurs the system will show an error message which explain the problem's reason.
* Response Time: the database chosen and used by the application is developed in order to get a low response time.