DP Hackathon: Acme-Patronage

Acme, Inc. is a holding that encompasses many companies worldwide. One of them is Acme Patronage, Inc., which is kind of a social network in which users can collaboratively finance projects.
The goal of this project is to develop a web information system to help Acme Patronage, Inc. manage their business. This document provides an informal description of their requirements; ask your lecturers for clarifications and details.

C-level requirements

Information requirements

1. The system must support two kind of actors, namely: users and admins. It must store the following information regarding them: their name, surname, birthday, email, an optional bio and an optional telephone (username and password are assumed).
2. Users can exchange messages. Every message must contain a header, a body and an optional link for an attachment (sender and recipient are assumed as part of the message).
3. A user can create projects, for which the system must store a title, a description, the moment of its creation, the economic goal, the minimum amount of money others can give them and the due date to fund the project. The creator is able to delete a project if there are no patronages. If any, it can only be marked as deleted.
4. Users can patronage projects. Every fund must store the amount of money, the date of the fund and a valid credit card with an expiration date after the project's due date. If the project is cancelled, these will be cancelled too.
5. The creator of the project can set a group of awards for each project. The system must store the name of the award, a short description and the amount of money necessary to unlock it. This unlock is cumulative, thus if a user funds a project with several awards already unlocked, he or she will receive those unlocked awards too.
6. The patronage will only be effective after the due date, thus all money won't be charged until the date is reached and only if the economic goal is achieved. If the economic goal is not reached, awards won't be delivered.
7. Projects can be organized using categories. Every category includes a name and a short description. These will be created by the admin.
8. The system must store the following information about credit cards: a holder name, a brand name, a number, an expiration month, an expiration year, and a CVV code, which is an integer between 100 and 999.

Functional requirements

9. An actor who is not authenticated must be able to:
    - 9.1. Register to the system as a user.
    - 9.2. List the projects with a future due date and browse them by categories.
    - 9.3. List the projects ordered by creation moment (new ones on top).
    - 9.4. List the awards of every project if they have any.
    - 9.5. List the categories of the system.
    - 9.6. List the users of the system and display their profile, which must include its personal data and the list of projects they have created and funded.
    - 9.7. Search for a project using a single keyword that must appear in its title or description.
10. An actor who is authenticated must be able to:
    - 10.1. Do the same as an actor who is not authenticated, but register to the system.
11. An actor who is authenticated as a user must be able to:
    - 11.1. Create a project. The user who has created a project is commonly referred as the creator.
    - 11.2. Edit a project as long as it's in draft mode. Once the project is in final mode, it cannot be modified and it's considered as published.
    - 11.3. Delete a project as long as it hasn't any patronages. If any, the project can only be cancelled.
    - 11.4. Manage the awards he or she has created. The user is able to create, edit and delete an award. Awards can be edited as long as its project is not published. However, further awards can be added to the project even if it's published.
    - 11.5. Fund or patronage a project of another user.
12. An actor who is authenticated as an administrator must be able to:
    - 12.1. Manage the categories, thus create, edit and delete a category. If a category is in use by any project it cannot be edited, only deleted. This removes the category from the project.
    - 12.2. Display a dashboard with the following information:
        - 12.2.1. The average and standard deviation of projects per user.
        - 12.2.2. The average and standard deviation of awards per project.
        - 12.2.3. The average and standard deviation of awards per user.
        - 12.2.4. The ratio of funded projects.
        - 12.2.5. The users who have, at least, a 10% more projects than the average.
        - 12.2.6. The users who have, at least, 10% more patronages than the average.
        - 12.2.7. The top-5 active projects (with a limit date not due) with more raised money.
        - 12.2.8. The top-5 due projects with more raised money.

Non-functional requirements

13. The system must be available in English and Spanish.
14. The system will be run in Spain, so it must comply with the Spanish regulations except for the following ones: a) the requirement in LOPD regarding keeping files and communications secure and confidential (unless you opt to earn an A+, in which case you're requested to use the HTTPS protocol where appropriate); b) the requirement in LSSI regarding informing the Chamber of Commerce about your internet domain.
15. The system must be as efficient as possible.
16. Pictures are not required to be stored by the system, but their URLs to external storage systems like Flickr.com or Pinterest.com.

B-level requirements

Information requirements

17. There is a new kind of actor: corporation. A corporation can support projects by means of sponsorships.
18. The system must store the following data regarding sponsorships: a URL to a banner and a link to an additional info page. A sponsorship can only be added to a project which hasn't reach its economic goal.
19. Users can post comments in several entities, such as projects, awards and announcements. The system must store the comment's text, the moment it was posted and a optional rating from 1 to 5. A comment can be deleted by the administrator or the user who posted it.
20. Creators can post announcements about their projects. The system must record the moment when an announcement is made, plus a title and a description.

Functional requirements

21. An actor who is not authenticated must be able to:
    - 21.1. Register to the system as a corporation.
    - 21.2. List the corporations.
    - 21.3. List the sponsorships of a project.
    - 21.4. List the comments of a project.
    - 21.5. List the comments of an award.
    - 21.6. List the comments of an announcement.
    - 21.7. List the announcements related to a project.
22. An actor who is authenticated must be able to:
    - 22.1. Do the same as an actor who is not authenticated, but register to the system.

23. An actor who is authenticated as a user must be able to:
    23.1. Post an announcement of a project he or she has created. Announcements cannot be edited or deleted.
    23.2. Display a stream of announcements regarding the projects that he or she has funded.
    23.3. Post a comment in a project, even if he or she hasn't funded it.
    23.4. Post a comment in an award, even if he or she hasn't funded it.
    23.5. Post a comment in an announcement, only if he or she has funded the related project.
24. An actor who is authenticated as a corporation must be able to:
    24.1. Manage its sponsorships, thus create, edit and delete them.
25. An actor who is authenticated as an administrator must be able to:
    25.1. Delete a comment if he or she finds it inappropriate.
    25.2. Display a dashboard with the following information:
        25.2.1. The average and standard deviation of sponsorships per corporation.
        25.2.2. The average and standard deviation of sponsorships per project.
        25.2.3. The average and standard deviation of comments per project.
        25.2.4. The projects that have, at least, a 10% more sponsorships than the average.
        25.2.5. The corporations that have, at least, a 10% more sponsorships than the average.

Non-functional requirements

26. Whenever a project is displayed, a random sponsorship must be selected and its banner shown, if any.

A-level requirements

Information requirements

27. There is a new kind of actor: moderators. A moderator takes care of the reports filled by other users.
28. Users can fill a report about a project. A report is a complaint about any aspect of the project. Moderators will investigate and decide if the complaint is legit. If so, it will be accepted and automatically delivered to the administrator. The system must store the reason why the report was accepted.
29. If a legit complaint has been filled against a project, the administrator will be able to delete it if he or she consider it inappropriate. This will remove its patronages.

Functional requirements

30. An actor who is not authenticated must be able to:
    30.1. Register to the system as a moderator.
31. An actor who is authenticated as a user must be able to:
    31.1. Fill a complaint about any project.
32. An actor who is authenticated as a moderator must be able to:
    32.1. Accept or reject a complaint.
33. An actor who is authenticated as a administrator must be able to:
    33.1. Ban or unban a user if he or she has created a project with a legit complaint.
    33.2. Delete a project with a legit complaint.
    33.3. Display a dashboard with the following information:
        33.3.1. The average and standard deviation of complaints per project.
        33.3.2. The ratio of complaints per project.
        33.3.3. The ratio of complaints per user.
        33.3.4. The ratio of legit complaints.
        33.3.5. The average and standard deviation of banned users.
        33.3.6. The ratio of banned users.

Non-functional requirements

Intentionally blank.

A+ - level requirements

34. Configure a continuos integration system and deliver a report which explains how to set up the environment and an example of use.