ACME EVENTS

DP – GROUP 21

Acme Events

Acme, Inc. is a holding that encompasses many companies worldwide, including Acme Event, Inc. Their business consists in helping managers to organize parties and rent his event hall, helping publicists to spread the publicity of the events and helping clients to learn about events and sign up for it.

The goal of this project is to develop a web information system that Acme Events, Inc. can use to run their business. This document provides an informal requirement specification. Ask your lecturers for clarifications and details, if necessary.

C-level requirements

Information requirements

1. The actors of the system are administrators, managers, and clients. For every actor, the system must store a name, an optional middle name, a surname, an optional photo, an email, an optional phone number, an optional address, and an arbitrary number of social profiles. The system must store the following data regarding such profiles: a nick, the name of the social network, a link to a profile in that social network. The system also stores the identification document number (DNI or similar) of every client.

2. Actors can exchange messages. For every message, the system must keep track of the sender, the recipient, the moment when it was sent, the subject, the body, its priority, and some optional tags. Priorities are HIGH, NEUTRAL, or LOW; no other values are expected. Every actor has the following message boxes: in box, out box, trash box, notification box and spam box. When an actor receives a message, it gets to the in box unless the system flags it as spam, in which case it gets to the spam box. When he or she sends a message to another user, it's saved to the out box. When an actor removes a message from a box other than trash box, it is moved to the trash box; when he or she removes it from the trash box, then it is actually removed from the system. The previous boxes are predefined and the actors must not be allowed to delete them, to change their names, or to move them. Actors are allowed to create new boxes that they can manage arbitrarily.

3. Managers have locals. For every local, the system must store a name, an address, a description and some optional pictures. When a local is created, it should be accepted by an administrator. Until the local is not accepted, events cannot be created in this local and he is not visible to the others actors.

4. Managers publish events. For every event, the system must store a ticker, the moment when it's published, a description, a price, the moments of starting and ending of the event, a status and the category and the local to which it belongs. Note the events that are saved in draft mode aren't visible to others actors until they are saved in final mode. The possible status are "AVAILABLE" and "CANCELLED".

5. The catalogue of categories is provided by the system. For every category, the system must store a title and its description.

6. The categories of events are organized into a hierarchy by the administrators. Every category belongs to a parent category, but the root category, which is called "CATEGORY".

7. A client may follow a local. When a client follow a local, he can see all his events.

8. Once an event is saved in final mode, the clients who follow the local receives a notification. The notification is received in the notification box.

9. Clients can publish opinions about an event (if he follows the local who organize the event), for each opinion, the system must store a title, a description and a score. Opinions cannot be removed or updated. A client can only publish opinions of events who are already finished.

Functional requirements

10. **An actor who is not authenticated must be able to:**
    1. Register to the system as a manager or a client.
    2. List all the locals that are in the system and see their information and the information of the manager, which includes his or her personal data plus his or her list of locals.

11. **An actor who is authenticated must be able to:**
    1. Do the same as an actor who is not authenticated, but register to the system.
    2. Edit his or her personal data.
    3. Exchange messages with other actors and manage them.
    4. Manage his or her message boxes, except for the system boxes.

12. **An actor who is authenticated as a manager must be able to:**
    1. Manage an arbitrary number of locals, which includes listing, showing, creating, updating, and deleting them. A local who have at least one event saved in final mode can't be deleted.
    2. Manage the events of their locals, which includes listing, creating, updating and deleting them. A manager may update or delete an event only if it's saved in draft mode. A manager can cancel an event who is saved in final mode.

13. **An actor who is authenticated as a client must be able to:**
    1. Follow or unfollow a local.

2. List the locals he follows and navigate to the events that are organized in that locals.
3. List the events that are organized in the locals he follows, ordered by date and grouped by status (passed, future, in course).
4. Filter the catalogue of events using the following filters: a single key word that must appear somewhere in its ticker, description; a category to which the event must belong; a range of prices; or a range of dates.
5. Publish an opinion about an event of a local how he is following.

14. An actor who is authenticated as an administrator must be able to:
1. Create user accounts for new administrators.
2. Manage the catalogue of categories, which includes listing, showing, creating, up-dating, and deleting them. Note that categories evolve independently from events, which means that they can be created, modified, or deleted independently from whether they are referenced from an event or not.
3. List the locals that aren't accepted yet and accept or refuse them. If the local is refused, a reason must be given by the administrator.
4. Broadcast a message to all of the actors of the system.
5. Display a dashboard with the following information:
   - The average, the minimum, the maximum, and the standard deviation of the number of locals per manager.
   - The average, the minimum, the maximum, and the standard deviation of the number of events per local.
   - The average, the minimum, the maximum, and the standard deviation of the maximum price of the events.
   - The average, the minimum, the maximum, and the standard deviation of the number of events per client.
   - The ratio of planned events.
   - The ratio of cancelled events.
   - The listing of managers who have at least 10% or more events with cancelled status.

Non-functional requirements

15. The system must be available in English and Spanish. (Unless otherwise stated, the data are not required to be available in several languages, only the messages that the system dis-plays.)

16. The system must be easy to customise at run time. The customisation includes, but is not limited to: the name of the system (it's "Acme Events" by default); the banner shown at the header; the message that is shown on the welcome page ("Welcome to Acme Events! The best place to make and go to events." is the default welcome message in English; "¡Bienvenidos a Acme Events! El major sitio para publicitar y asistir a eventos." is the default welcome message in Spanish); a list of spam words (it's "sex", "viagra", "cialis", "one million", "you've been selected", "Nigeria", "sexo", "un

millón", and "ha sido seleccionado" by default); the default country code in telephone numbers (it's "+34"by default); the default list of credit card makes (it's "VISA", "MASTER", "DINNERS", and "AMEX" by default).

17. The system must provide the following leaf categories by default: MODERNO/MODERN, POPULAR/POPULAR, REGGEATON/ REGGEATON, CLASICO/CLASSIC, ELECTRO/ELECTRO, DEEP/DEEP, TECH/TECH, TRAP/TRAP, RAP/RAP, ORQUESTA/ORQUEST, PIANO/PIANO, VIOLÍN/VIOLIN, GUITARRA/GUITAR, ROCK/ROCK, POP/POP. Their names must be displayed appropriately in Spanish or English depending on the language that the principal's selected. The categories must be grouped appropriately into higher-level categories.

18. Photos are not required to be stored in the database, but links to external systems like Pin-terest.com or Flickr.com, just to mention a couple of examples.

19. When an actor gets a message that contains a spam word, it must be stored in the spam box instead of the input box.

20. Every time that an event is saved in final mode or change its status, the system must send a message to both the corresponding manager and the clients who follows the local who the event is organized. The message must be written in both English and Spanish.

21. Events must be shown according to the following colour scheme: events that are AVAILABLE must be shown in green; events that are CANCELLED must be shown in red.

22. Tickers must adhere to the following pattern: "yymmdd-xxxxxx", where "yymmdd" refers to the year, month, and day when the corresponding entity is registered, and "xxxxxx" to a random uppercase alpha-numeric string. No two entities may have the same ticker since it's assumed to be a unique external identifier.

23. Phone numbers should adhere to the following patterns: "+CC (AC) PN", "+CC PN", or "PN": "+CC" denotes a country code in range "+1" up to "+999", "(AC)" denotes an area code in range "(1)" up to "(999)", and "PN" denotes a number that must have at least four digits. Phone numbers with pattern "PN" must be added automatically a default country, which is a parameter that can be changed by administrators. Note that phone numbers should adhere to the previous patterns, but they are not required to. Whenever a phone number that does not match this pattern is entered, the system must ask for confirmation; if the user confirms the number, it then must be stored.

24. Email addresses must adhere to any of the following patterns: "identifier@domain", "alias <identifier@domain>"; administrators may have email addresses of the form "identifier@", or "alias <identifier@>". The identifier is an alpha-numeric string, the domain is a sequence of alpha-numeric strings that are separated by dots, and the alias is a sequence of alpha-numeric strings that are separated by spaces.

B-level requirements

Information requirements

25. Clients can register their curricula. Every curriculum has a ticker, a personal record, some education records, some professional records and some miscellaneous records.

26. A personal record consists of the full name of a client, a photo of him or her, his or her email, his or her phone number, and a URL to his or her LinkedIn profile.

27. An education record consists of the title of a diploma, the period during which the client was studying, the institution that awarded the diploma, an optional link to an attachment, and some optional comments. Note that an education record may refer to a period that hasn't finished yet if the handy worker's still studying.

28. A professional record consists of the name of the company for which a client was working, the corresponding period of time, the role played, an optional link to an attachment, and some optional comments. Note that a professional record may refer to a period that hasn't finished yet.

29. A miscellaneous record consists of a title, an optional link to an attachment, and some optional comments.

30. Managers write job offers for their locals. For every job offer the system must store a ticker, a title, a description, some requisites and a status. The possible status are "OPEN" and "CLOSED".

31. Clients can apply for a job offer. For each application, the system must store a description, the moment when the application is submitted and some optional comments. To apply, the client must have at least a curriculum. When an application is submitted, the curriculum of the client is copied and linked to the application.

32. They are a new kind of actor, publiciters.

33. Publiciters can apply to a local. For every application, the system must store an optional commentary, the moment when the application is submitted, an optional moment when the application is rejected and an optional moment when the application is accepted.

Functional requirements

34. **An actor who is authenticated as a manager must be able to:**
    1. Manage his or her job offers, which includes listing, showing and creating them.
    2. Close an open job offer.

3. Manage the list of applications of publiciters and accept or reject it.
4. List the applications of job offers and see the personal information of the clients and their curriculum.
5. Close an application for a job offer.

35. An actor who is authenticated as a client must be able to:
1. Manage his or her curriculum, which includes showing, updating, deleting and creating them.
2. Manage his or her personal records, which includes listing, showing, updating, deleting and creating them.
3. Manage his or her education records, which includes listing, showing, updating, deleting and creating them.
4. Manage his or her miscellaneous records, which includes listing, showing, updating, deleting and creating them.
5. Manage his or her professional records, which includes listing, showing, updating, deleting and creating them.
6. Apply for an open job offer. A curriculum is required to apply for a job offer.
7. List the application for job offers that he has done.
8. List the publiciters of a local and see his personal data and social profiles.

36. An actor who is authenticated as a publiciter must be able to:
1. List the locals of the system.
2. Apply to a local.
3. List her locals and their events.

37. An actor who is authenticated as an administrator must be able to:
1. Display a listing of suspicious actors. An actor is considered suspicious if he or she publishes some data that includes spam words.
2. Ban an actor who is considered suspicious, which means that his or her user account is deactivated.
3. Unban an actor, which means that his or her user account is reactivated.
4. Launch a process who update the scores of the events and locals.
5. Display a dashboard with the following information:
   - The minimum, the maximum, the average, and the standard deviation of the number of job offers per local.
   - The minimum, the maximum, the average, and the standard deviation of the number of job offers per manager.
   - The top-three locals in terms of job offers.
   - The top-three locals in terms of publiciters

Non-functional requirements

38. The maximum number of results that a finder returns is 10 by default. The administrator should be able to change this parameter in order to adjust the performance of the system. The absolute maximum is 100 results.

39. Attachments are not required to be stored by the system, but their URLs to external storage systems like Drive.com or Dropbox.com, to mention a few examples.

40. Everywhere an event is showed, the median of the score must be showed.

41. Everywhere a local is showed, the median of the scores of his events must be showed.

A-level requirements

Information requirements

42. Publiciters can write offers for an event. For every offer, the system must store its title, a description and some optional pictures.

43. Clients can apply for an offer, for every application the system must store a unique ticker and the moment when the application is submitted.

Functional requirements

44. An actor who is not authenticated must be able to:
   1. Register to the system as a sponsor.

45. An actor who is authenticated as a publiciter must be able to:
   1. Manage his offers, who includes listing and creating.

46. An actor who is authenticated as a client must be able to:
   1. List the offers of an event an apply to it.
   2. List his offers and show their tickers.

47. An actor who is authenticated as an administrator must be able to:
   1. Display a dashboard with the following information:
       - The minimum, the maximum, the average, and the standard deviation of the number of offers per event.
       - The minimum, the maximum, the average, and the standard deviation of the number of offers per publiciter.
       - The top-three publiciters in terms of offers.

Non-functional requirements

48. Tickers must adhere to the following pattern: "yymmdd-xxxxxx", where "yymmdd" refers to the year, month, and day when the corresponding entity is registered, and "xxxxxx" to a random uppercase alpha-numeric string. No two entities may have the same ticker since it's assumed to be a unique external identifier.