

Acme Hacker Rank

Acme, Inc. is a holding that encompasses many companies worldwide, including Acme Hacker Rank, Inc. Their business consists in helping information technology companies recruit talented hackers across the world.

The goal of this project is to develop a web information system that Acme Hacker Rank, 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**, **companies**, and **hackers**. For every actor, the system must store a name, one or more surnames, a VAT number, a valid **credit card**, an optional photo, an email, an optional phone number, and an optional address. The system must also store the commercial name of the companies.
2. 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.
3. 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.
4. Companies can publish **positions**. For every position, the system must store a title, a description, a deadline, the profile required, the skills required, the technologies required, and the salary offered. Every position is identified by means of a unique **ticker** that must match the following pattern: "XXXX-9999", where "XXXX" **are the first four letters in the commercial name of the company (Using "X" to pad shorter names)**, and "9999" **is a pseudo-random four-figure number**.
5. Companies can manage a database of **problems** that they can associate with their positions. For every problem, the system must store a title, a statement, an optional hint, and some attachments.
6. Hackers can apply for positions. For every **application** the system must store the application moment, a problem that the corresponding hacker must solve, his or her answer (which consists of a piece of text with explanations and a link to his or her code), and the moment when he or she submits it. An application may have the following statuses: PENDING, when it's created; SUBMITTED, when the corresponding hacker's submitted a solution to the problem; ACCEPTED, when the corresponding company's accepted it; REJECTED, when the corresponding company's rejected it.



Functional requirements

7. An actor who is not authenticated must be able to:
 1. Register to the system as a company or a hacker.
 2. List the positions available and navigate to the corresponding companies.
 3. List the companies available and navigate to the corresponding positions.
 4. Search for a position using a single key word that must be contained in its title, its description, its profile, its skills, its technologies, or the name of the corresponding company.
8. 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.
9. An actor who is authenticated as a company must be able to:
 1. Manage their positions, which includes listing, showing, creating, updating, and deleting them. Positions can be saved in draft mode; they are not available publicly until they are saved in final mode. Once a position is saved in final mode, it cannot be further edited, but it can be cancelled. A position cannot be saved in final mode unless there are at least two problems associated with it.
 2. Manage their database of problems, which includes listing, showing, creating, updating, and deleting them. Problems can be saved in draft mode; once they are saved in final mode, they cannot not be edited.
 3. Manage the applications to their positions, which includes listing them grouped by status, showing them, and updating them. Updating an application amounts to making a decision on it: an application whose status is SUBMITTED may change to status ACCEPTED or REJECTED.
10. An actor who is authenticated as a hacker must be able to:
 1. Manage his or her applications, which includes listing them grouped by status, showing them, creating them, and updating them. When an application is created, the system assigns an arbitrary problem to it (from the set of problems that have been registered for the corresponding position). Updating an application consists in submitting a solution to the corresponding problem (a piece of text with explanations and a link to the code), registering the submission moment, and changing the status to SUBMITTED.
11. An actor who is authenticated as an administrator must be able to:
 1. Create user accounts for new administrators.
 2. Display a dashboard with the following information:
 - The average, the minimum, the maximum, and the standard deviation of the number of positions per company.
 - The average, the minimum, the maximum, and the standard deviation of the number of applications per hacker.
 - The companies that have offered more positions.
 - The hackers who have made more applications.
 - The average, the minimum, the maximum, and the standard deviation of the salaries offered.
 - The best and the worst position in terms of salary.

Non-functional requirements

12. 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 displays.)

13. Photos and attachments are not required to be stored in the database, but links to external systems like Pinterest.com or GitHub.com, just to mention a couple of examples.
14. 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 Hacker Rank" by default); the banner shown at the header (it's the one available at <https://i.imgur.com/7b8lu4b.png> by default); the message that is shown on the welcome page ("Welcome to Acme hacker Rank! We're IT hacker's favourite job marketplace!" is the default welcome message in English; "¡Bienvenidos a Acme Hacker Rank! ¡Somos el mercado de trabajo favorito de los profesionales de las TICs!" is the default welcome message in Spanish); and the default country code in telephone numbers (it's "+34" by default).

B-level requirements

Information requirements

15. A hacker may register one or more curricula. Curricula consist of the following data: personal data, which includes a full name, a statement, a phone number, a GitHub profile, and a LinkedIn profile; position data, which includes the title, the description, the start date, and the optional end date of every position that a hacker has had; education data, which includes the degree, the institution, the mark, the start date, and the optional end date of every degree that a hacker has; and miscellaneous data, which is free text with optional attachments.
16. Hackers have finders that they can configure using the following search criteria: a single key word, a deadline, a minimum salary, and/or a maximum deadline. The contents of the finder are the positions that match those search criteria (the key word can be contained in the ticker, the title, the description, the skills, the technologies, or the profile); if a search criterion is not specified, then every position meets it.

Functional requirements

17. An actor who is authenticated as a hacker must be able to:
1. Manage his or her curricula, which includes listing, showing, creating, updating, and deleting them. When a hacker makes an application, he or she must select one of his or her curricula so that it's attached to the application. Note that attaching a curriculum makes a copy; the updates that a hacker performs on the original curriculum are not propagated to the applications to which he or she's attached a previous version.
 2. Manage his or her finder, which involves updating the search criteria, listing its contents, and clearing it.
18. 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 curricula per hacker.
 - The minimum, the maximum, the average, and the standard deviation of the number of results in the finders.
 - The ratio of empty versus non-empty finders.

Non-functional requirements

19. The results of a finder are cached for one hour by default. The administrator should be able to configure that period at will in order to adjust the performance of the system. The mini-

mum time's one hour and the maximum time's 24 hours. When a user requests to clear his or her finder, the system must re-compute its results immediately.

20. 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.

A-level requirements

Information requirements

21. The actors of the system can register their social **profiles**. The system must store the following data regarding them: a nick, the name of the social network, a link to a profile in that social network.
22. Every actor has a pool of messages. For every **message**, the system must keep track of the sender, the recipient, the moment when it was sent, the subject, its body, and some optional tags.

Functional requirements

23. An actor who is authenticated must be able to:
 1. Manage his or her social profiles, which includes listing, showing, creating, updating, and deleting them.
 2. Manage his or her messages, which includes listing them grouped by tag, showing them, sending a message to an actor, deleting a message that he or she's got. If a message is deleted and it doesn't have tag "DELETED" then it gets tag "DELETED", but it's not actually deleted from the system; if a message with tag "DELETED" is deleted, then it's actually removed from the system.
24. An actor who is authenticated as an administrator must be able to:
 1. Broadcast a notification message to the actors of the system. The message must have tag "SYSTEM" by default.
 2. Launch a process that flags the actors of the system as spammers or not-spammers. A user is considered to be a spammer if at least 10% of the messages that he or she's sent contain at least one spam word.
 3. **Ban** an actor with the spammer flag.
 4. **Unban** an actor who was banned previously.

Non-functional requirements

25. Wherever the profile of an actor is shown to an administrator, the system must show whether he or she is considered a spammer. Note that "N/A" must be shown in cases in which an actor has not been computed his or her spammer flag.
26. The default **list of spam words** includes "sex", "viagra", "cialis", "one million", "you've been selected", "Nigeria", and their corresponding Spanish translations.
27. The system must generate automatic notifications on the following events: an application changes its status, a new offer that matches a hacker's finder search criteria is published.