Acme BnB

Acme, Inc. is a holding that encompasses many companies worldwide. One of them is Acme Bed'n Breakfast , Inc., which is a system to find cheap accommodations. For short, Acme Bed'n Breakfast, Inc. is typically referred to as Acme BnB, Inc.

The goal of this project is to develop a web information system that Acme BnB, Inc. can use to support their business. This document provides an informal description of their requirements; ask your lecturers for clarifications and details.

C-level requirements

Information requirements

- The actors of the system can be either administrators, lessors, or tenants. For every actor, the system must store his or her name, surname, email, phone, and a picture. The phone number must take international prefixes into account.
- Lessors are people who own properties and wish to rent them. For every property, the system must store its name, its rate (in EURO per day), its description, and its address, plus a number of attributes. The pre-defined attributes are "country", "province", "state", "city", and "capacity", but others may be defined by the administrators.
- Tenants are people who are seeking for accommodation. Every tenant has a finder, which is kind of a search template in which he or she can specify the following data: the destination city; the minimum and the maximum price that he or she's willing to pay; a keyword that must appear somewhere in the name, the description, or the address of a property. Note that every field of a finder's optional, but the destination city.
- A tenant can make a request to book a property. He or she must indicate a check-in date and a check-out date, which must be at least one day after the check-in date, an indication on whether he or she's a smoker or not, and a valid credit card that will be charged the corresponding rate if the request's approved. Requests can be either PENDING, which means that it's been recorded but the corresponding lessor has neither approved nor denied it, ACCEPTED, if the corresponding lessor has accepted it, and DENIED, if the corresponding lessor has denied it.
- Every lessor who accepts a request is charged a one euro fee. Lessors must register a valid credit card before they can accept a request; prior to allowing a lessor to accept a request, the system must check that the credit card that he or she's registered's valid.

Functional requirements

- A user who is not authenticated must be able to:
    - Register to the system as a lessor or a tenant.
    - Browse the catalogue of properties, navigate to the corresponding lessors, and display their profiles.
- An actor who is authenticated must be able to:
    - Do the same as an actor who is not authenticated, but registering to the system.
    - Change the information in his or her profile.

- An actor who is authenticated as a lessor must be able to:
  - Manage his or her properties, which includes registering, listing, editing, and deleting them.
  - Approve or deny a request to book any of his or her properties.
  - List the requests to his or her properties.
  - See total fee amount that they have to pay.
- An actor who is authenticated as a tenant must be able to:
  - Configure his or her finder.
  - Browse the results of his or her finder.
  - Make a request for a property.
  - Browse his or her requests.
- An actor who is authenticated as an administrator must be able to:
  - Manage the attributes used to describe the properties, which includes listing, creating, modifying, and deleting them.
  - Display a dashboard with the following information:
    - The average number of accepted and denied requests per lessor.
    - The average number of accepted and denied requests per tenant.
    - The lessors who have approved more requests.
    - The lessors who have denied more requests.
    - The lessors who have more pending requests.
    - The tenants who have got more requests approved.
    - The tenants who have got more requests denied.
    - The tenants who have more pending requests.
    - The lessor/s whose ratio of requested versus approved request/s is the maximum or the minimum. (Ratios must be rounded to one decimal place).
    - The tenant/s whose ratio of requested versus approved request/s is the maximum or the minimum. (Ratios must be rounded to one decimal place).
    - The average, the minimum, and the maximum number of results per finder.

Non-functional requirements
- The system must be available in English and Spanish.
- 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; b) the requirement in LSSI regarding informing the Chamber of Commerce about your internet domain.
- The system must be as efficient as possible.
- Pictures are not required to be stored by the system, but referenced by means of their URLs.
- The results of a finder must be cached for at least one hour.
- The validity of a credit card must be checked as follows: a) its number must be run through Luhn's algorithm to checksum it; b) the expiry date must be confirmed to be at least seven days in future.
- Wherever a credit card is shown, it must be masked, i.e. only the leading and trailing four digits must be readable; the others must be displayed as asterisks.
- Wherever a catalogue of properties is shown, the user must be able to sort it according to the number of requests that the corresponding properties have got.

- The fee that lessors have to pay must be configurable; it's 1.00 € by default, but administrators must be able to change it easily. Note that changing the fee must not have any effects on the total fee that a lessor has to pay prior to the change.

B-level requirements

Information requirements
- The system has a new kind of actor: auditors. For every auditor, the system must store the name of the company for which he or she works.
- Auditors produce audits regarding properties. For every audit, the system must store the moment when it is written and the corresponding text plus an arbitrary number of attachments. An auditor is not allowed to write more than one audit per property.
- Some actors are allowed to post comments. For every comment, the system must store a title, the moment when it's posted, a piece of text and a number of stars (in range zero – five).

Functional requirements
- An actor who is authenticated must be able to:
    o Browse the catalogue of properties, navigate and display their audits, if any.
- An actor who is authenticated as a lessor must be able to:
    o Post comments to his own profile or to the profile of a tenant who has requested any of his or her properties.
- An actor who is authenticated as a tenant must be able to:
    o Post comments to his own profile or to the profile of any lessor whose properties he or she has requested.
- An actor who is authenticated as an auditor must be able to:
    o Write an audit regarding any of the properties that have been registered in the system.
- An actor who is authenticated as an administrator must be able to:
    o Register a new actor as an auditor.
    o Display a dashboard with the following information:
        ▪ The minimum, the average, and the maximum number of audits that the properties have got.
        ▪ A listing in which the attributes are sorted in descending order regarding the number of times they have been used to describe a property.
    o Display a dashboard with the following information regarding every lessor:
        ▪ A listing with his or her properties sorted according to the number of audits that they have got.
        ▪ A listing with his or her properties sorted according to the number of requests that they have got.
        ▪ A listing with his or her properties sorted according to the number of approved requests that they have got.
        ▪ A listing with his or her properties sorted according to the number of denied requests that they have got.

- A listing with his or her properties sorted according to the number of pending requests that they have got.

Non-functional requirements

- The system's not requested to store any attachments, only the URLs that allow to download them from external storage services like consigna.us.es, dropbox.com, or mega.com.
- Auditors must be allowed to save drafts of their audits. An audit must not be displayed while it's in draft mode, only when it's declared final. Audits can be modified or deleted as long as they are drafts. It's mandatory that the system asks for confirmation before making an audit final.
- Whenever the profile of a lessor or a tenant is shown, it must include the comments that other lessors or tenants have posted to it.
- The requirements regarding a lessor posting comments to a tenant's profile and vice versa are quite volatile. It's expected that it changes in forthcoming releases of the project so that actors can post comments to other entities, e.g., properties or audits.

A-level requirements

Information requirements

- Actors may register their social identities. For every social identity, the system must store a nick, the name of a social network, and the URL of his or her profile in that social network.
- The system can issue invoices to tenants. For every invoice, it must store the moment when it's issued, Acme BnB's VAT number, some information about the tenant, the details, the total amount due (in EURO), and the credit card that was used to pay it.

Functional requirements

- An actor who is authenticated must be able to:
  - Manage his or social identities, which includes creating, listing, modifying, and deleting them.
- An actor who is authenticated as a tenant must be able to:
  - Request the system to issue an invoice regarding a property that he or she's booked.
- An actor who is authenticated as an administrator must be able to:
  - Display a dashboard with the following information:
    - The minimum, the average, and the maximum number of social identities per actor.
    - The minimum, the average, and the maximum number of invoices issued to the tenants.
    - The total amount of money due in the invoices issued by the system.
    - The average number of requests for properties that have at least an audit record versus the average number of requests for properties that do not have any audits.

Non-functional requirements

- Whenever an actor's profile is shown, it must include the corresponding listing of social identities, if any.