

Requirements Analysis Document

Application: 1000House

Date: 15/02/2018

Index

1. Introduction.....	3
1.1 Purpose of the system	3
1.2 Scope of the system	3
1.3 Objectives and success criteria of the project	3
1.4 Definitions, Acronyms, and abbreviations.....	3
2. System Description	4
2.1 Functional Requirements	4
2.1.1 General user.....	4
2.1.2 Guests.....	4
2.1.3 Hosts.....	4
2.1.4 Administrator.....	5
2.1.5 System.....	5
2.2 Non-functional Requirements	5
2.2.1 Usability.....	5
2.2.2 Reliability.....	5
2.2.3 Other Non-functional Requirements.....	5
3. Use Cases	6
3.1 Use Case diagram	6
3.2 Use case descriptions	7
3.2.1 Use Case Create an Offer.....	7
3.2.1 Use Case Buy an Offer.....	8
3.2.1 Use Case Approve an Offer.....	9
4. Mockups.....	10

1. Introduction

1.1 Purpose of the system

This application is designed to give the possibility to rent houses and, furthermore, to help people to find a good place for living or for holidays, depending on the purpose of each one.

Everyone who is registered can use it for both functions as a host or as a guest. The application is also useful for its users because it shows and manages the best offers with characteristics, so they can choose the offer which suites them better.

1.2 Scope of the system

- Allow registered users to login.
- Search renting offers.
- Let the guests buy and reserve offers.
- Let the hosts create offers
- Register the bought, booked and created offers
- Use an external system to manage the payment of the offers.
- Manage users' information.
- Allow an administrator to supervise all the created, bought and booked offers.

1.3 Objectives and success criteria of the project

The main objective is creating an application that manages the renting of houses for holiday or living purpose. The principal users of this application will be the guests, who want to buy/reserve and offer of a renting house, and the hosts, who rent the house to the guests.

Success of this software depends on the implementation of this main characteristics and the usability of the application by the users.

1.4 Definitions, Acronyms, and abbreviations

- **HOST:** A registered user who can rent houses for living and/or for holidays by creating offers.
- **GUEST:** A registered user who is able to book and buy hosts' offers.
- **HOLIDAYS PRICE:** The price of holidays offers, which includes the deposit and the price corresponding to the offered time.

- **LIVING PRICE:** The price of living offers which includes the deposit and the first month the guest is going to stay.
- **OFFER RECIPE:** A document that confirms a guest has reserved an offer with her/his identification.

2. System Description

2.1 Functional Requirements

2.1.1 General Users can:

- FR1: Search by offers, zip-codes and dates (beginning and ending) of the offers.
- FR2: Check and see offers but not to reserve, buy or comment them.
- FR3: See the description, price and dates of offers but not the comments.

2.1.2 Guests can:

- FR1: Login with her/his username and password. In her/his account the user has her/his name, surname and credit card number.
- FR2: Search by offers, zip-codes and dates (beginning and ending) of the offers.
- FR3: Search by the offers that have been bought or booked and by the numerical value of the comments of the offer.
- FR4: Check and see complete offers.
- FR5: Comment offers (although she/he has not booked or bought them) using text or a numerical value between 1 and 5.
- FR6: Book offers.
- FR7: Cancel reservations.
- FR8: Buy offers.
- FR9: Pay offers at the moment or before 5 days (if it is booked).
- FR10: Access to her/his register of bought and booked offers.
- FR11: Ask for changes in her/his profile information to the administrator.

2.1.3 Hosts can:

- FR1: Login.
- FR2: Create offers (holidays or living).
- FR3: Add location, date, price, description and other characteristics (like swimming pool) to offers.
- FR4: Choose the beginning day of the offer and the last day in case of a holidays offer, or the number of months to rent the house in case of living.
- FR5: Check her/his created offers and see the comments.

- FR6: Modify the offer if the administrator does not approve it.
- FR7: Access to her/his register of created offers.
- FR8: Ask for changes in her/his profile information to the administrator.

2.1.4 Administrators can:

- FR1: Login with a special id.
- FR2: Check and approve host's offers.
- FR3: Ask for changes in host's offers.
- FR4: Reject host's offers.
- FR5: Change registered users' profile information.
- FR6: Ban registered users.
- FR7: Remove expired reserves.

2.1.5 System can:

- FR1: Manage the user's payment.
- FR2: Charge the deposit and the final price.
- FR3: Give 2% to the company in case of holidays offer.
- FR4: Give 0.1% to the company in case of living.

2.2 Non-functional Requirements

2.2.1 Usability

The application must have an intuitive and simple interface so that the users, both registered and general users, can use it in an easy way making actions quickly, so they will have a quick and fast response. Furthermore, the app will be divided by different sections including searching, settings...

2.2.2 Reliability

The application must have a backup in order not to produce a data loss in case of several internal error.

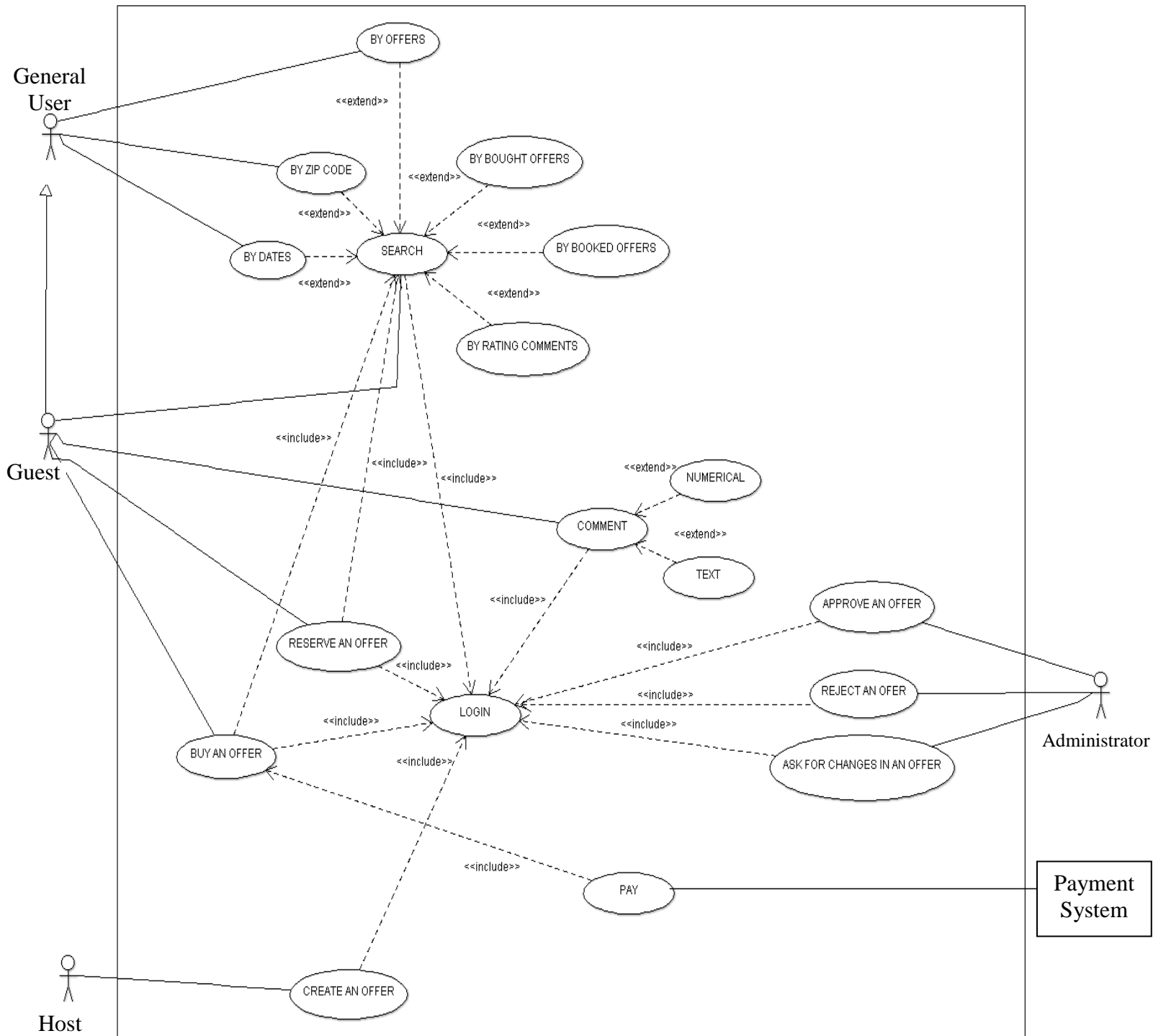
2.2.3 Other Non-functional Requirements

The application must ensure the protection of the data, keeping the security in the personal details and offers information.

Furthermore, another NFR should be compatible with Windows, Mac and Linux.

3. Use Cases

3.1 Use Case diagram



3.2 Use case descriptions

3.2.1 Use Case Create an Offer

Primary Actor: Host.

Stakeholders and Goals:

- **Host:** Rent a house for living or holidays to earn some money.
- **Guest:** Book and/or buy the offer for holidays or for living.
- **Administrator:** Reject or approve the offer.

Preconditions:

The host has to login with his/her username and password and click on the button “Create an Offer”.

Success guarantee (Post-conditions):

The offer is created with its characteristics and a notification is sent to the administrator in order to review it, asking for changes in case of rejection. In case of approval, the offer can be searched, booked and bought by guests.

Main Success Scenario:

1. The host selects “Create an Offer”.
2. The host writes a description of the offered house with its characteristics.
3. The host adds the location of the offered house.
4. The host selects a holidays offer.
5. The host adds the beginning and ending date of the offer.
6. The host adds the holidays price of the offer.
7. The host clicks on “Finish” and the administrator is notified.

Extensions (Alternative paths):

- 2a. The host selects a living offer.
 - 2a.1. The host adds the beginning day of the offer and the number of months the guest would can live.
 - 2a.2. The host adds the living price of the offer.

Special Requirements:

- A quick response (<0.5 sec) and update.
- An immediate notification to the administrator.
- A quick answer from the administrator in case he/she asks for some changes to create it.

Frequency:

Quite high, several times in a day.

Open Issues:

- Internationalization(i18n): the app should be able to show the description and characteristics of the offered house in several different languages so that it can be more intuitive.

3.2.2 Use Case Buy an Offer

Primary Actor: Guest.

Stakeholders and Goals:

- **Guest:** To buy an offer (holidays or living).
- **Host:** Sells his offer and gets the money.
- **Administrator:** Modifies accounts in case of errors.
- **System:** Manages the payment process.

Preconditions:

The guest has to login and be in the initial page, where the search options are. He/she searches an offer which suits him/her best by clicking on the searching button and adding his/her preferences. Different offers will appear on the screen.

Success guarantee (Post-conditions):

The guest buys the offer, the system manages the payment and keeps a x% of it for the company, depending on the type of offer, giving the rest to the host.

Main Success Scenario:

1. The guest selects a holiday offer.
2. The guest scrolls down checking the house information, including the date.
3. The guest decides to buy the offer, so she/he clicks on the buy button.
4. The system reads the credit card information of the user and proceeds to the payment.
5. The page confirms the payment, adding the offer to the bought offers of the user, and to the sold offers of the host, sending a message to this one.

Extensions (Alternative paths):

- 1.a The guest selects a living offer.
 - 1.a.1 The page redirects the user to the payment screen, managed by an external system, and confirms the payment of the living price.
- 3.a The offer has already been bought/reserved
 - 3.a.1 The system informs the user he can't buy the offer
 - 3.a.2 The user goes back to search another offer.
- 4.a The credit card information is incorrect
 - 4.a.1 The system informs the user of the error and bans her/him.
 - 4.a.2 A message is sent to the administrator to inform him/her that he/she has to check the user information and modify it
 - 4.a.3 The user can't use his account until the administrator lets her/him

Special Requirements:

- Quick response during the payment process (<0.5 s).
- Ensure the protection and security of the user data during the payment.

Technology and Data Variations List:

- Option to make a quick-buy from the selection screen.
- Option to print the offer recipe in PDF and to send it to the e-mail.

Frequency:

More than a hundred per minute

Open Issues:

- Allow the user to introduce his credit card number manually if his information is incorrect.

3.2.3 Use Case Approve an Offer

Primary Actor: Administrator

Stakeholders and Goals:

- Administrator: Checks a new offer created by a host and approves, denies or asks for changes in it.
- Host: Wants to have his/her offer approved

Preconditions:

The administrator logs in and goes to the notification window, where a new offer has appeared (created by a host).

Success guarantee (Post-conditions):

The offer has no mistakes and it is properly created, so the administrator approves it.

Main Success Scenario:

1. The administrator clicks on the offer notification.
2. The administrator reads the description and checks the location.
3. The administrator verifies that the prices, both the holidays and the living price, are correct.
4. The administrator clicks on “Approve offer” and a notification is sent to the respective host.

Extensions (Alternative paths):

- 4.a. The administrator decides that something is wrong and clicks on “*Deny offer*” and a notification is sent to the host.
- 4.b. The administrator finds some minor mistakes and clicks on “*Ask for changes*” and a notification is sent to the host.

Special Requirements:

- A quick response (<0.5 sec).
- An immediate notification to the host with the details.

Technology and Data Variations List:

- The option that the location of the house is checked by 1000Maps.
- Improve it by adding a software which checks the proper vocabulary.

Frequency:

Very high, several times in an hour.

4. Mockups



When you open the application, you are a General User so you can do his/her FR and you must click on *Sign in* to do the rest of FR depending on the role you have.

Here you the General User can see different offers. He can click on one to see its characteristics.

He can search something different and look for the best offers.



When the General User tries to click on *Reserve*, *Buy* or *Comment*, an alert is opened. He/she can choose between signing in or canceling it.



If the user wants to login in order to rent, book... He/she must click on *Sign in* with the username and password.



The image shows a web browser window titled "1000House". The main heading is "Welcome To 1000House!" in green. Below this, there are two input fields: "Username" and "Password". To the right of the "Username" field is a question mark icon. Below the input fields is a checkbox labeled "Remember me" and a green "Login" button. At the bottom, there are four links: "Home", "Terms&Conditions", "Security", and "Contact us".

If the user is a guest, he/she can now reserve, buy or comment an offer.

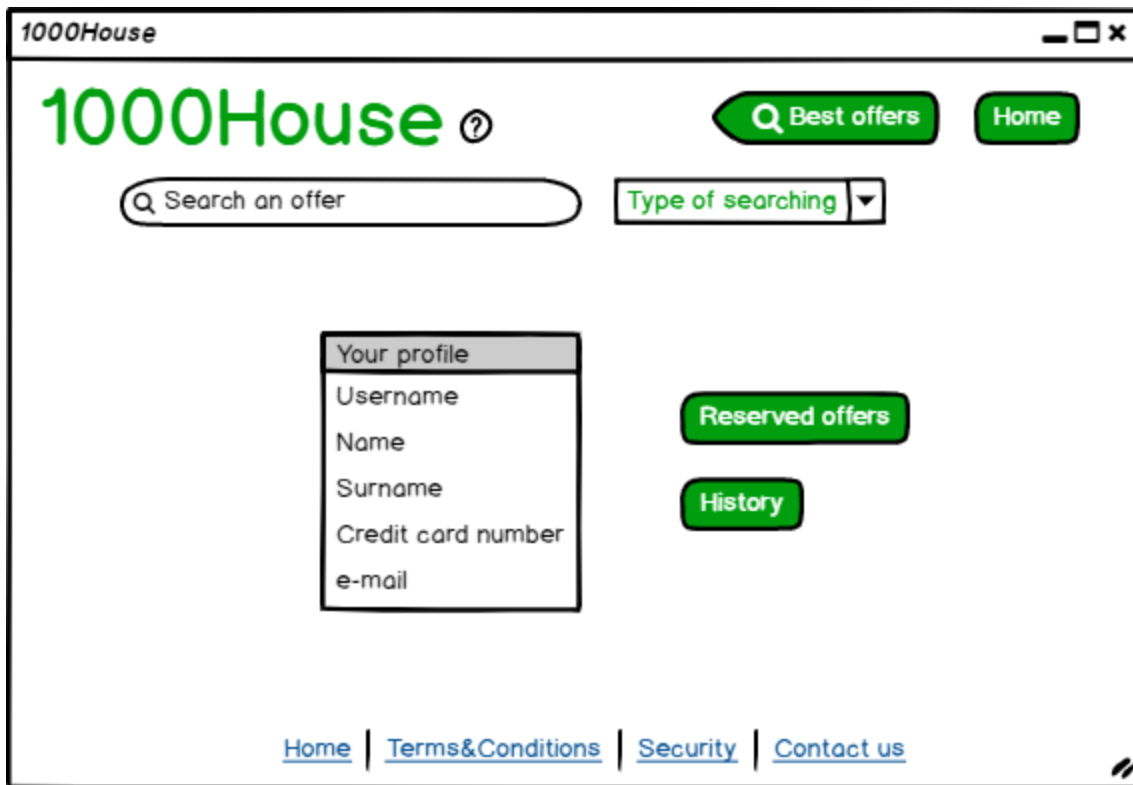


The image shows a web browser window titled "1000House" displaying a guest interface. At the top, there is a search bar with the text "Search an offer" and a dropdown menu labeled "Type of searching". To the right of the search bar are two buttons: "Best offers" and "Profile". Below the search bar, there is a section titled "Holidays Off" with placeholder text. A white alert box is overlaid on the page, containing the text "Alert" and "Sorry! This offer is already reserved", with a "Close" button. Below the alert box, there is a section with two reviews: "mirotic: 4/5" and "balodis: 1/5". To the right of the reviews, there is a price tag "57€" and two buttons: "Reserve" and "Buy". At the bottom, there are four links: "Home", "Terms&Conditions", "Security", and "Contact us".



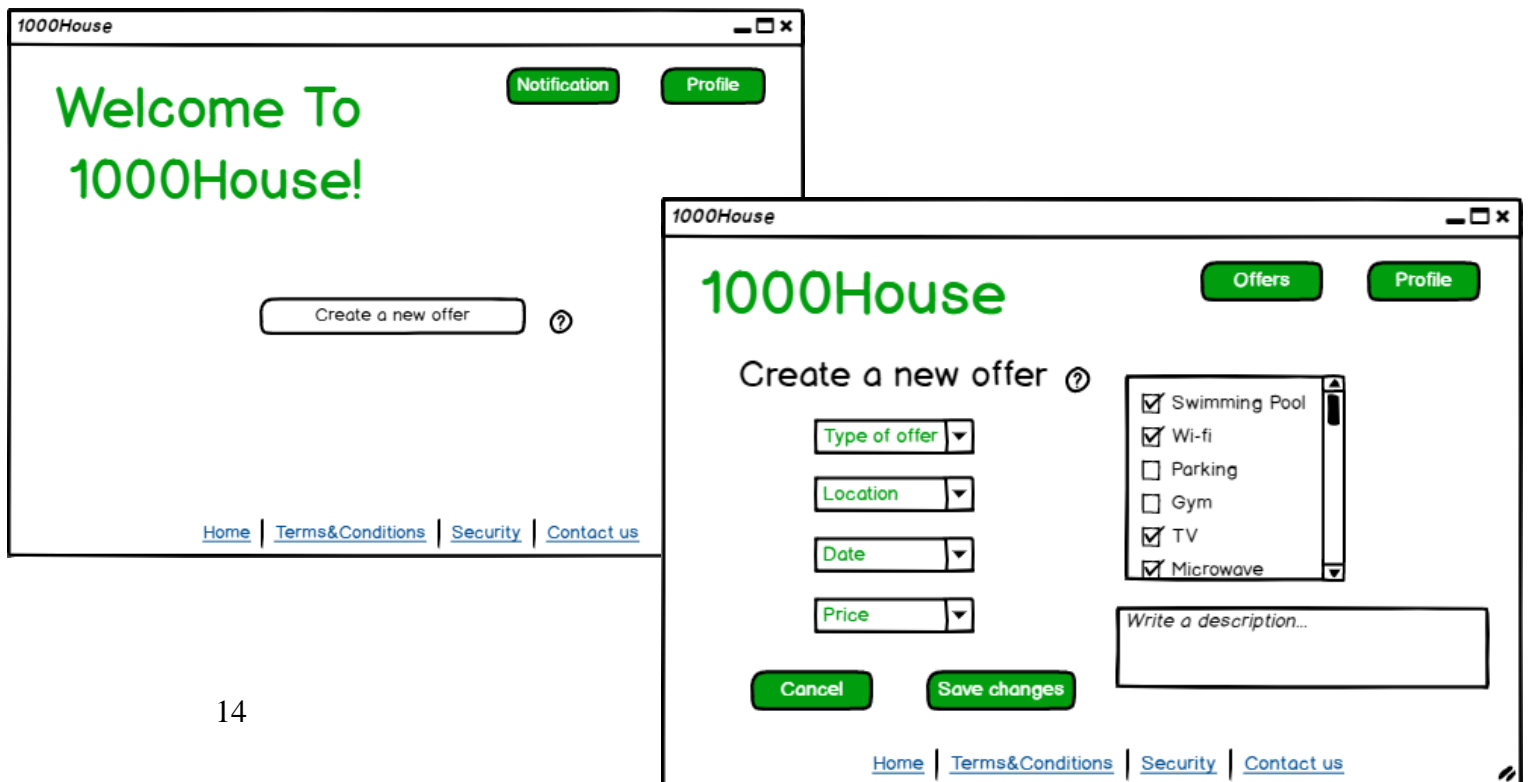
When the user wants to comment he can choose the numerical value or the letter value.

When the user clicks on *Profile* he/she can see his/her information.



The screenshot shows the '1000House' user profile page. At the top, there's a header with the '1000House' logo, a search bar labeled 'Search an offer', and buttons for 'Best offers' and 'Home'. Below the header, a dropdown menu titled 'Type of searching' is visible. The main content area features a 'Your profile' section with a list of fields: Username, Name, Surname, Credit card number, and e-mail. To the right of this list are two buttons: 'Reserved offers' and 'History'. At the bottom, there's a navigation bar with links: Home, Terms&Conditions, Security, and Contact us.

When the user is a Host he can create a new offer and access to the history.



The screenshot shows the '1000House' host interface. It consists of two overlapping windows. The top window is a 'Welcome To 1000House!' screen with a 'Create a new offer' button and a 'Notification' button. The bottom window is the 'Create a new offer' form. It has a header with the '1000House' logo and buttons for 'Offers' and 'Profile'. The form includes fields for 'Type of offer', 'Location', 'Date', and 'Price'. To the right of these fields is a list of amenities with checkboxes: Swimming Pool, Wi-fi, Parking, Gym, TV, and Microwave. Below the amenities list is a text area labeled 'Write a description...'. At the bottom of the form are 'Cancel' and 'Save changes' buttons. A navigation bar at the very bottom contains links: Home, Terms&Conditions, Security, and Contact us.

Finally, when the user is an Administrator, he/she can check the offers in order to approve or reject them (or ask for changes) and check users.

