An Apartment Booking System

by Alexandra Preuß

Supervised by: Dr. David Rossiter
Department of Computer Science and Engineering

technische universität dortmund

Fakultät Bio- und Chemieingenieurwesen

December 12, 2018

Table of Contents

1. Introduction 3

2. Comprehensive project outline 4

2.1. Acquisition of new tenants for apartments 4

2.1.1. The *‘Home’* tab 6

2.1.2. The *‘Apartment’* tab 8

2.2. Management of all involved parties in this system 10

2.2.1. The *‘Admin’* Page 10

2.3. Creation of a tenant account 12

2.3.1. Sessions 13

3. Results 15

4. Outlook 18

Table of Figures

Figure 1: Front-end to back-end flow. 3

Figure 2: Process flow from first accessing the website to the tenant account. 4

Figure 3: Registration and Login Page. 5

Figure 4: Landing Page. 7

Figure 5: Accessing 'Apartment' tab with no or invalid apartment id (options 1 & 2). .. 8

Figure 6: Typical way of displaying an apartment to the potential tenant. 9

Figure 7: Examples to add or update a record. 11

Figure 8: Illustration of administrative workspace. 11

Figure 9: Sample Tenant Account 12

Figure 10: First implementation to report an incident using the tenant account. 13

Figure 11: Example of a self-declaration form within the tenant account. 14

Figure 12: Dropdown div's applying JavaScript to be implemented in 'Home' and 'Apartment' webpages 16

Figure 13: Current design of displaying key data for each apartment. 16

1. Introduction

This project involves the design and implementation of a website for both the acquisition of new tenants of apartments as well as the corresponding property and facility management.

The project itself serves as an example to see how a website is set up, how to manage a database and consequently how to connect the front-end to the back-end. Thereby, the front-end refers to the website which is visited by a web user, while the back-end refers to the database created within phpMyAdmin and is a graphical user interface (GUI) for accessing a mysql database provided by the XAMPP server. The connection between the front-end and back-end as well as the relationship in between those is further displayed in Figure 1. The language which connects the *mysql database* to the website is PHP.

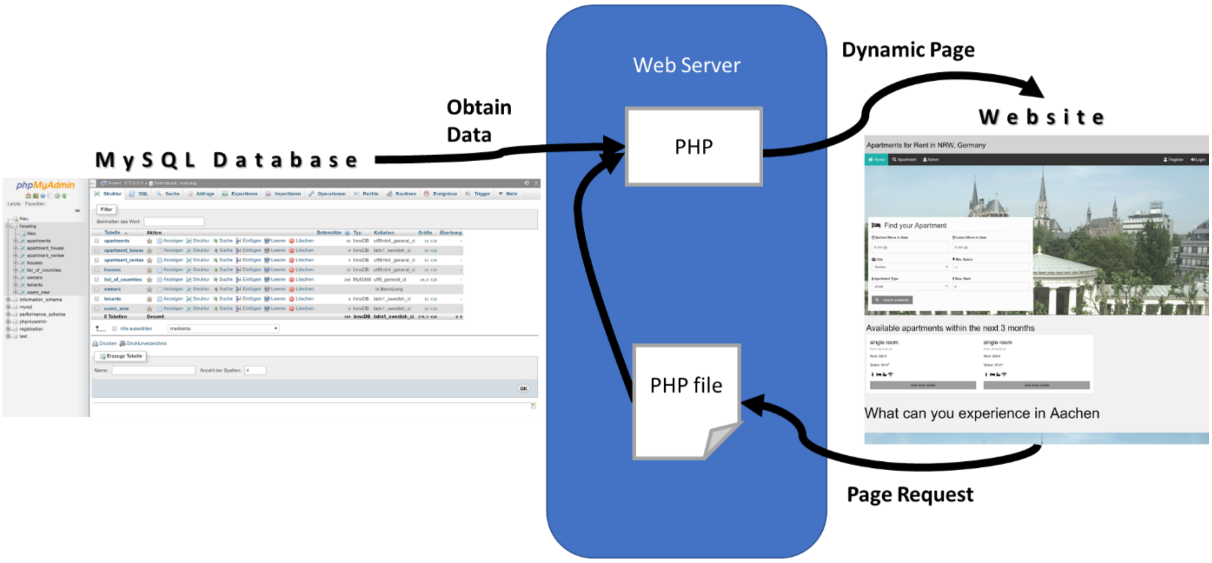

Web Server
Dynamic Page
Website
Obtain Data
MySQL Database
PHP
PHP file
Page Request
phpMyAdmin
Apartments for Rent in NRW, Germany
Find your Apartment
Available apartments within the next 3 months
What can you experience in Aachen

Figure 1: Front-end to back-end flow.

Front-end designing languages that are used in this project are primarily HTML, CSS, Bootstrap and some JavaScript coding.

2. Comprehensive project outline

2.1. Acquisition of new tenants for apartments

As this project will be used mainly within Europe, a few records in the largest searching machines for apartments will be placed redirecting interested people searching the web, further referred to as potential tenants (PT), to the newly developed homepage in this project.

The website offers different options for a PT to find a suitable apartment and get involved in the application process for one or more apartments. The options are captured in Figure 2 below.

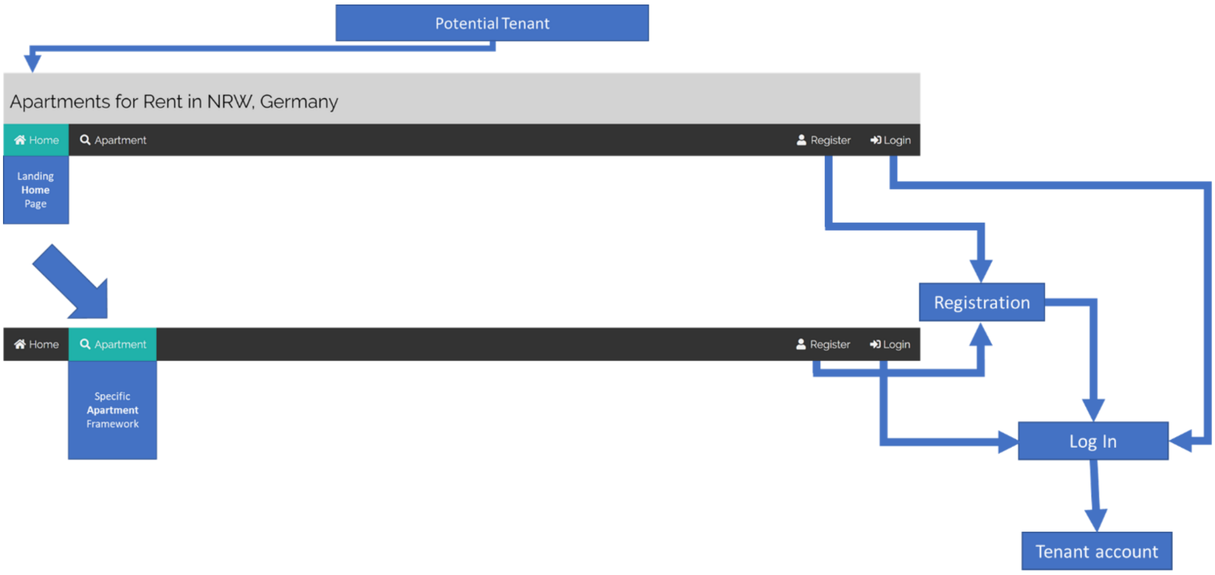


Figure 2: Process flow from first accessing the website to the tenant account.

At first, the PT is directed to the landing page, see Figure 4 for more details. At this landing page, the PT can either use a search engine to filter for a suitable apartment or view a selection of those apartments that will be vacant within the next three months. If interested, the PT can register and create an account through the links in the navigation bar such as ‘Registration’ and afterwards ‘Login’.

In case a PT has not found a suitable apartment yet, another and most likely way is that he or she checks out the apartments first and only register once a promising apartment has been found. There is a redirection to the registration page below each apartment. The larger arrow in Figure 2 indicates multiple options of displaying *apartment details* within a framework for which data is retrieved from the database.

That means that a PT is directed to the ‘*Apartment’* tab for any selected apartment and the only information which changes are the apartment details such as rent, space and so forth. Thus, the dynamic website will always depend on a specific apartment id. More information regarding this is provided in chapter 2.1.2 The ‘*Apartment’* tab.

Once logged in, the tenant can and will be asked to conduct the application process from within this account.

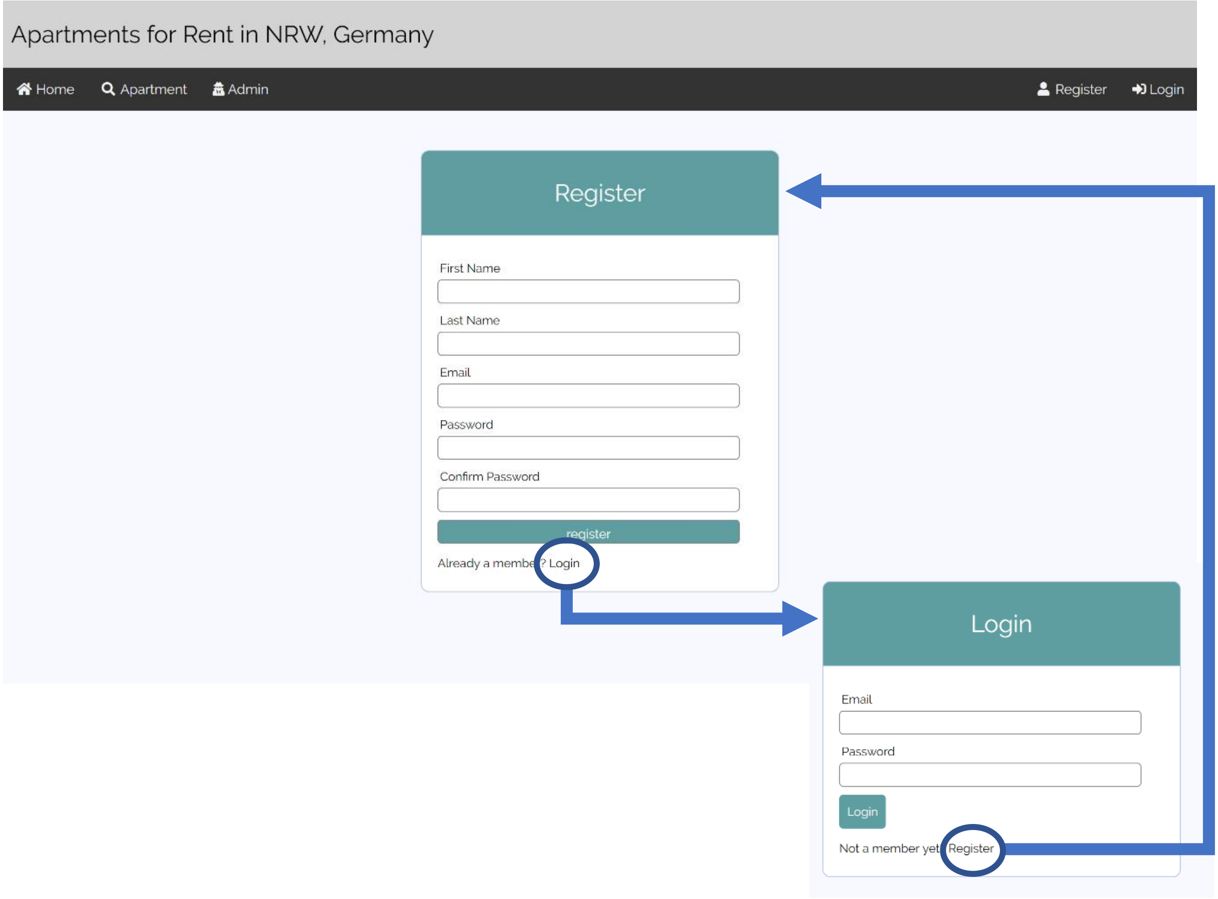


Figure 3: Registration and Login Page.

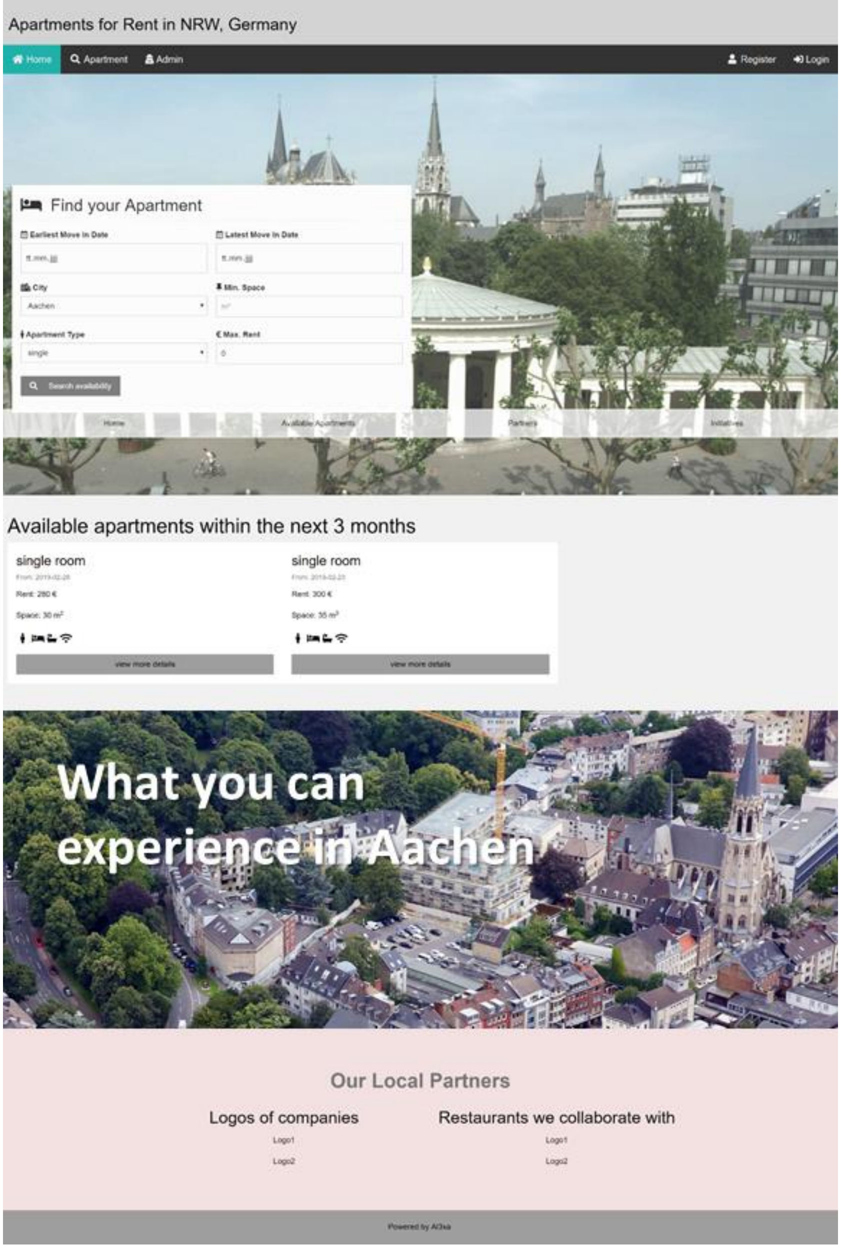
2.1.1. The *'Home'* tab

The landing page is the page with the '*Home'* tab in the navigation bar to which anyone will be redirected as explained in the Introduction. This *'Home'* page consists mainly of five divisions or so-called <div> sections which are:

- a navigation bar at the top end of the page
- a basic search machine to filter for suitable apartments
- a selection of upcoming available apartments
- an automatic slideshow to promote the corresponding cities and
- some general placeholders for local business advertisement respectively

The PT searching the web, however, is not to see the Admin tab in the navigation bar. A PT can only filter from the available apartments in the basic search field. The search options such as 'City' and 'Apartment Type' are directly connected to the database and therefore only show the options present within the database. For instance, if there are only three cities, in which apartments can be offered, then only those can be selected for the search in the dropdown menu.

Furthermore, when accessing this landing page, a selection of those apartments that will become vacant within the upcoming three months is displayed in the div below the searching machine. By default, those apartments are selected in the SQL query and executed by the PHP code. The PT can look for more details of each apartment when pressing any of the *'view more details'* buttons. These buttons include a link with a PHP loop retrieving the corresponding data from the database to display the relevant information for each apartment in the '*Apartment'* tab once clicked.



Find your Apartment

Earliest Move In Date: tt.mm.jjjj

Latest Move In Date: tt.mm.jjjj

City: Aachen

Min. Space: m²

Apartment Type: single

Max. Rent: 0

Available apartments within the next 3 months

single room
From: 2019-02-28
Rent: 280 €
Space: 30 m²

single room
From: 2019-02-23
Rent: 300 €
Space: 35 m²

Our Local Partners

Logos of companies
Logo1
Logo2

Restaurants we collaborate with
Logo1
Logo2

Powered by Al3xa

Figure 4: Landing Page.

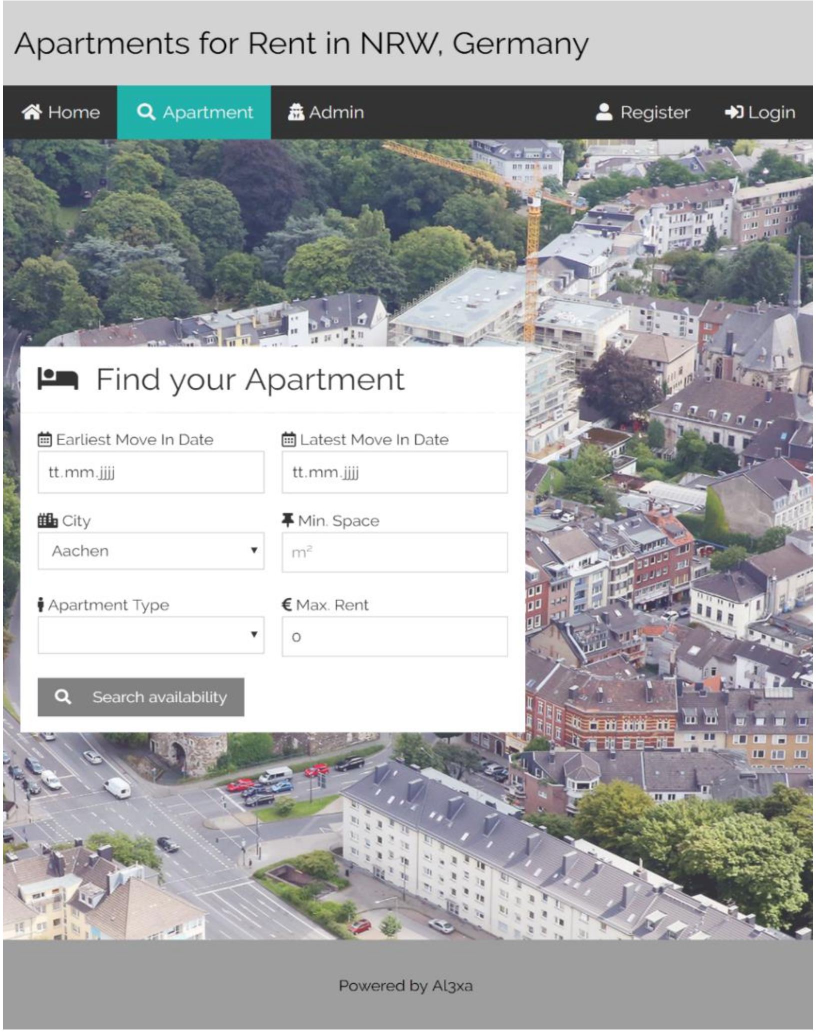
Furthermore, as displayed in Figure 4, there is one div supposed to be a placeholder for typical events, festivals, nightlife places and so forth that can be used for advertisement and help making this website more attractive to PTs.

Once this system is eventually running, the second last div indicates an opportunity to finance this website. The most important local partners can place their advertisement at this or also other spots on this website.

2.1.2. The *‘Apartment’* tab

The ‘*Apartment*’ tab in the top navigation bar includes a framework which retrieves specific data from the database. There are three options when accessing this ‘*Apartment*’ tab:

1. Click on the ‘*Apartment*’ tab itself; no apartment id is selected
2. Accessing the ‘*Apartment*’ tab by entering a faulty apartment id in the URL
3. Accessing the ‘*Apartment*’ tab by going through one of the generated buttons on the landing page; existing apartment id



Apartments for Rent in NRW, Germany

Find your Apartment

Earliest Move In Date: tt.mm.jjjj

Latest Move In Date: tt.mm.jjjj

City: Aachen

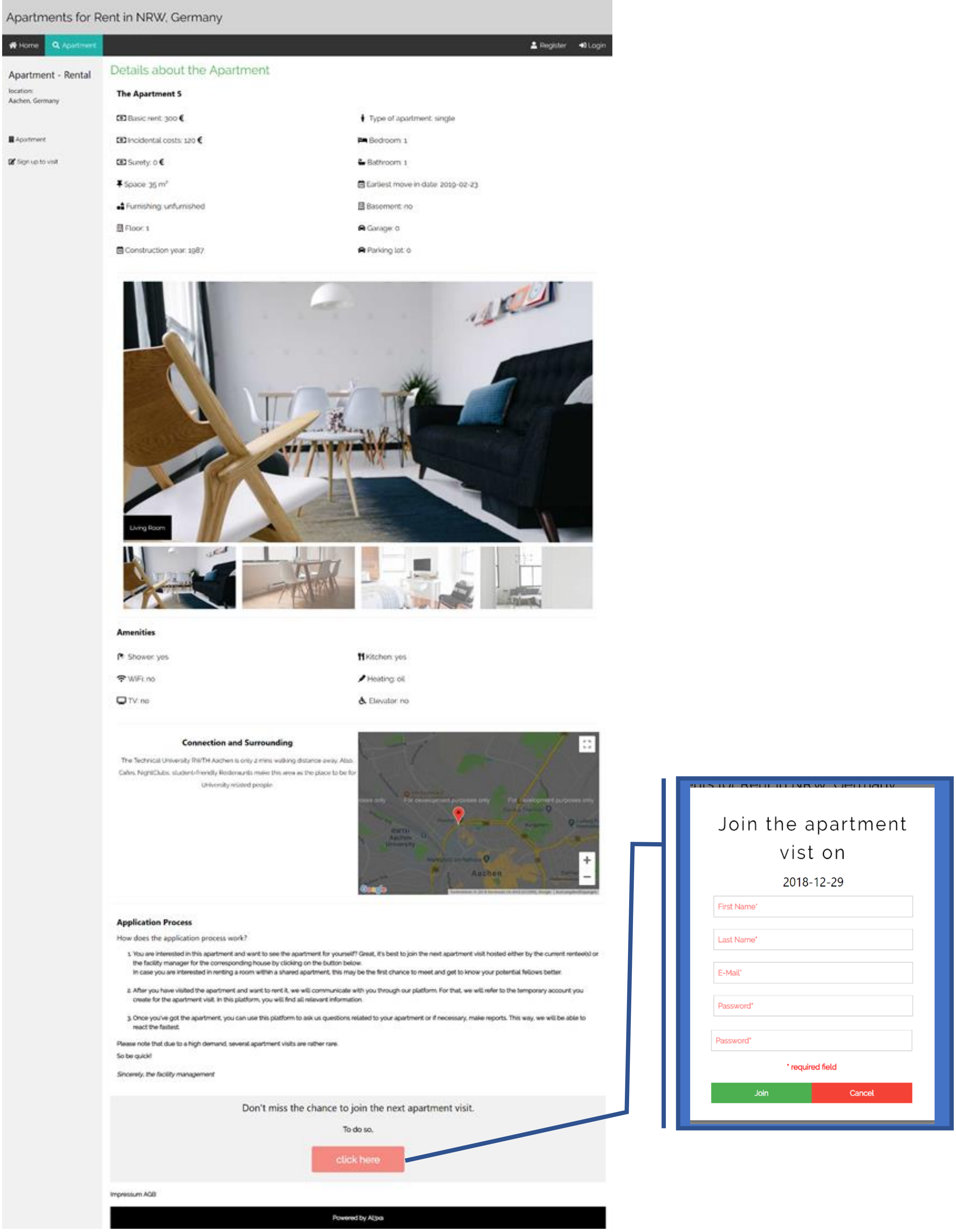
Min. Space: m²

Apartment Type:

Max. Rent: 0

Figure 5: Accessing ‘Apartment’ tab with no or invalid apartment id (options 1 & 2).

A typical design for accessing an apartment according to option 3, as in this case apartment with the apartment id no. 5, is displayed in Figure 6 on the next page.



Apartments for Rent in NRW, Germany

Apartment - Rental

location:
Aachen, Germany

Details about the Apartment

The Apartment 5

Basic rent: 300 €	Type of apartment: single
Incidental costs: 120 €	Bedroom: 1
Surety: 0 €	Bathroom: 1
Space: 35 m²	Earliest move in date: 2019-02-23
Furnishing: unfurnished	Basement: no
Floor: 1	Garage: 0
Construction year: 1987	Parking lot: 0

Amenities

Shower: yes	Kitchen: yes
WiFi: no	Heating: oil
TV: no	Elevator: no

Connection and Surrounding

The Technical University RWTH Aachen is only 2 mins walking distance away. Also, Cafes, NightClubs, student-friendly Restaurants make this area as the place to be for University related people.

Application Process

How does the application process work?

1. You are interested in this apartment and want to see the apartment for yourself? Great, it's best to join the next apartment visit hosted either by the current rentee(s) or the facility manager for the corresponding house by clicking on the button below.
   In case you are interested in renting a room within a shared apartment, this may be the first chance to meet and get to know your potential fellows better.
2. After you have visited the apartment and want to rent it, we will communicate with you through our platform. For that, we will refer to the temporary account you create for the apartment visit. In this platform, you will find all relevant information.
3. Once you've got the apartment, you can use this platform to ask us questions related to your apartment or if necessary, make reports. This way, we will be able to react the fastest.

Please note that due to a high demand, several apartment visits are rather rare.

So be quick!

Sincerely, the facility management

Don't miss the chance to join the next apartment visit.

To do so,

Impressum AGB

Powered by Allpa

Join the apartment visit on

2018-12-29

First Name*

Last Name*

E-Mail*

Password*

Password*

* required field

Figure 6: Typical way of displaying an apartment to the potential tenant.

2.2. Management of all involved parties in this system

Eventually, this system is meant to run fully automated and is supposed to be used by the following people:

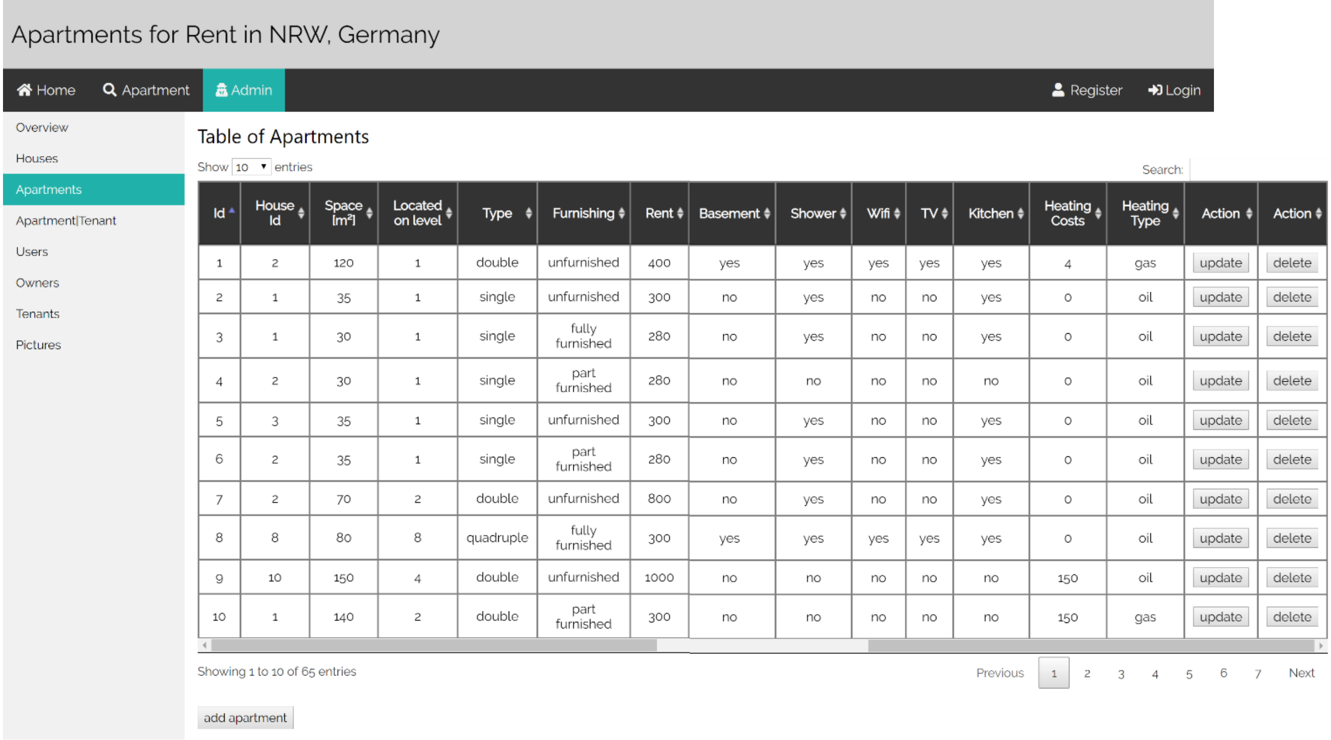
- **Owners:**
  The owners are basically providing the property and are only informed if relevant and financial issues arise
- **Managers:**
  The manager(s) takes care of the tenant acquisition process and oversees the daily business. The manager can *add* and *update* property and user information.
- **Admin:**
  The admin is in charge of assigning roles to the people in the system if needed. Also, the admin is granted the ability to *add*, *update* and *DELETE* property and user details.
- **Tenants:**
  The high demand/request of apartments by potential tenants is the reason why this project is worth to be implemented in first place. The scope of this project is to set up a working system to handle all tenants properly. However, in the future, this process should become much more user-friendly so that both the offering parties and searching parties can profit from this system.
- **Janitors (out of scope):**
  Janitors receive notifications and updates if needed
- **Third parties (out of scope):**
  Businesses, Clubs and special offers can be placed in this website to enhance local business and make profit.

2.2.1. The *'Admin'* Page

The Admin Page enables the admin to conduct four basic functions: The admin shall be able to *select*, *add* or *insert*, *update* and *delete* records from the website in the database. The basic records included in this project are records of houses, apartments, users, owners, tenants as well as pictures.

As of now in this system, users include the owners and tenants only. However in reality, users will include all people who are listed in chapter 2.2.

In general, the admin and the manager both see the overview of actions that are required. These may be to take action in handling some issues that are reported by the tenants or review the PT reactions who have completed and submitted a self-declaration form, see chapter 2.3. The manager should only be granted limited access such as to *add* or *update* records.



Apartments for Rent in NRW, Germany

Table of Apartments

Show 10 entries

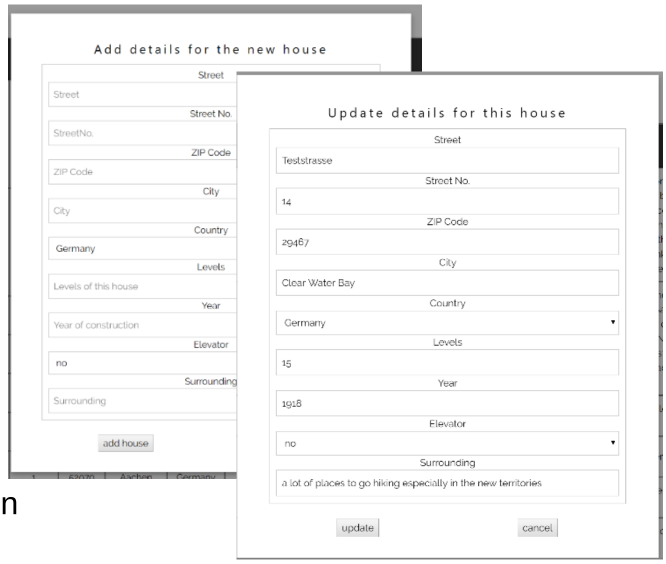
Search:

Id	House Id	Space [m²]	Located on level	Type	Furnishing	Rent	Basement	Shower	Wifi	TV	Kitchen	Heating Costs	Heating Type	Action	Action
1	2	120	1	double	unfurnished	400	yes	yes	yes	yes	yes	4	gas	update	delete
2	1	35	1	single	unfurnished	300	no	yes	no	no	yes	0	oil	update	delete
3	1	30	1	single	fully furnished	280	no	yes	no	no	yes	0	oil	update	delete
4	2	30	1	single	part furnished	280	no	no	no	no	no	0	oil	update	delete
5	3	35	1	single	unfurnished	300	no	yes	no	no	yes	0	oil	update	delete
6	2	35	1	single	part furnished	280	no	yes	no	no	yes	0	oil	update	delete
7	2	70	2	double	unfurnished	800	no	yes	no	no	yes	0	oil	update	delete
8	8	80	8	quadruple	fully furnished	300	yes	yes	yes	yes	yes	0	oil	update	delete
9	10	150	4	double	unfurnished	1000	no	no	no	no	no	150	oil	update	delete
10	1	140	2	double	part furnished	300	no	no	no	no	no	150	gas	update	delete

Showing 1 to 10 of 65 entries

Figure 8: Illustration of administrative workspace.

Within the '*Admin'* tab, it is possible to navigate throughout the different tables using the sidebar. Each table includes an *update* and *delete* button for each record as well as an *add* button below the table to insert a new record in the database, see Figure 7. An example of how this looks like is given in fig for the houses or fi for the apartment-house relationship table.



Add details for the new house

Street: Street
Street No.: StreetNo.
ZIP Code: ZIP Code
City: City
Country: Germany
Levels: Levels of this house
Year: Year of construction
Elevator: no
Surrounding: Surrounding

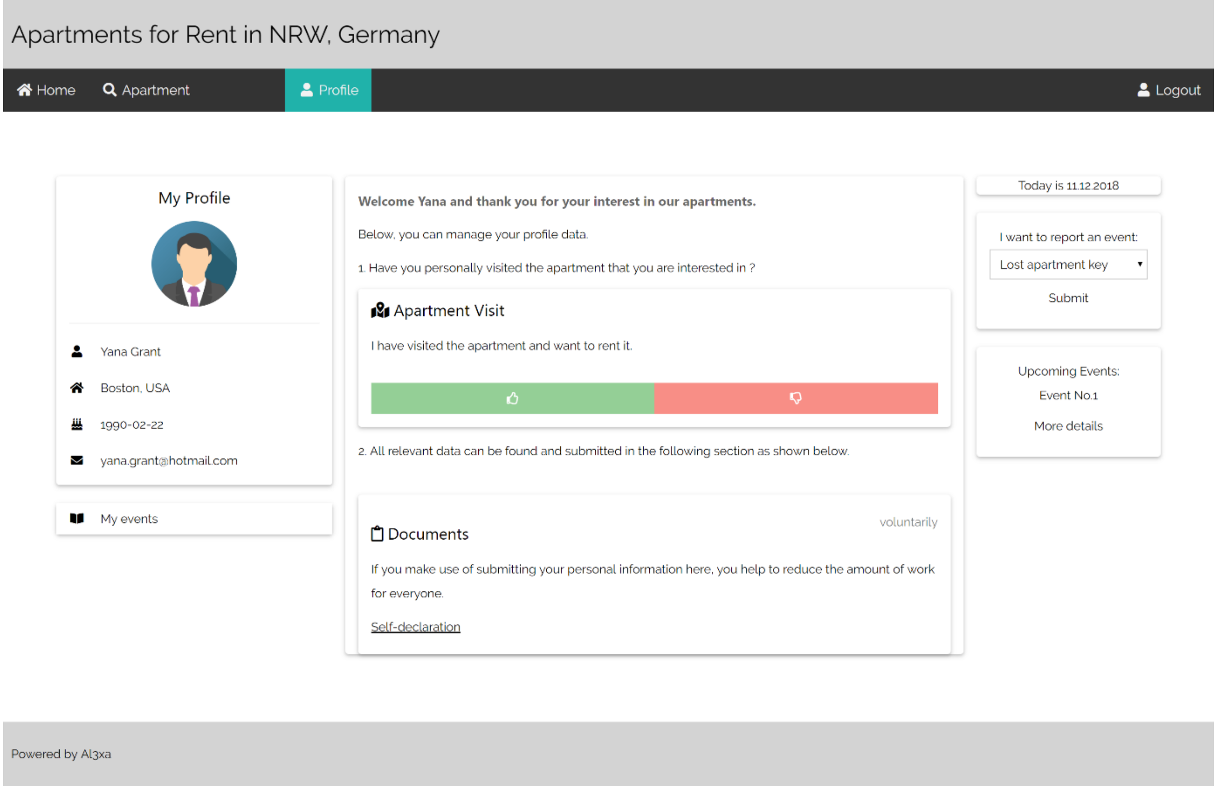
Update details for this house

Street: Teststrasse
Street No.: 14
ZIP Code: 29467
City: Clear Water Bay
Country: Germany
Levels: 15
Year: 1918
Elevator: no
Surrounding: a lot of places to go hiking especially in the new territories

Figure 7: Examples to add or update a record.

2.3. Creation of a tenant account

Once successfully logged in, the PT can review the data provided such as his or her first and last name as well as the email address. An example of how the interface currently looks like is shown in Figure 9 below.



Apartments for Rent in NRW, Germany

My Profile

Yana Grant
Boston, USA
1990-02-22
yana.grant@hotmail.com

My events

Welcome Yana and thank you for your interest in our apartments.

Below, you can manage your profile data.

1. Have you personally visited the apartment that you are interested in ?

Apartment Visit

I have visited the apartment and want to rent it.

2. All relevant data can be found and submitted in the following section as shown below.

Documents voluntarily

If you make use of submitting your personal information here, you help to reduce the amount of work for everyone.

Self-declaration

Today is 11.12.2018

I want to report an event:
Lost apartment key

Upcoming Events:
Event No.1

Powered by Al3xa

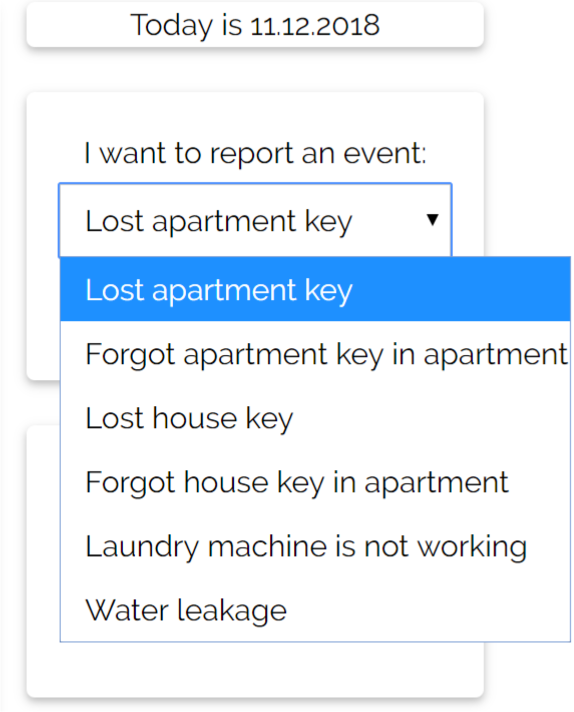
Figure 9: Sample Tenant Account.

On the left side, basic information including a PT's name, old address, birthday and email address will always be displayed. If the corresponding data is missing in the beginning, placeholders display the type of information.

The middle part includes the application process guideline with listed instructions as well as links to further pages. For instance, it would become a long list if the PT would be asked to fill out the self-declaration form in the front page of the Tenant account. Thus, this will take place on separate webpages, as shown in Figure 11.

The right side of the main account page will be more useful to tenants who already rent an apartment and need to report an incident. Six examples of possible and most common incidents are listed as quick dropdown options as shown in Figure 10.

However, it will be important that an additional option for free text is necessary as well and not implemented yet.



Today is 11.12.2018

I want to report an event: Lost apartment key

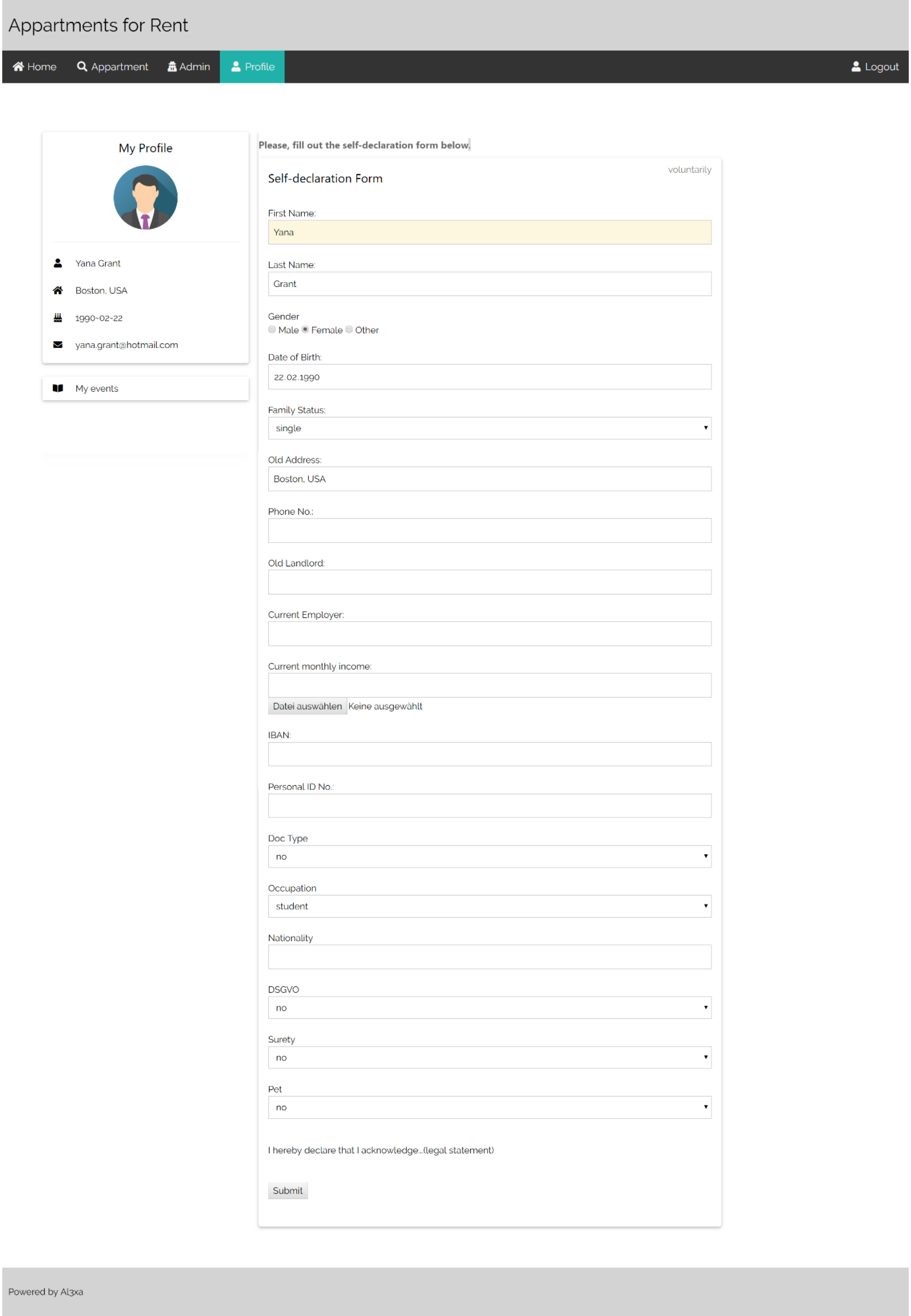
Lost apartment key
Forgot apartment key in apartment
Lost house key
Forgot house key in apartment
Laundry machine is not working
Water leakage

Figure 10: First implementation to report an incident using the tenant account.

2.3.1. Sessions

One essential part of getting this system to run is to implement the superglobal PHP variable SESSION allowing a user to login to the account and not lose connection when switching in between tabs of this website project system.

This implementation works in cases with smaller systems but not fully for all pages in this scope of this project so far.



My Profile

Yana Grant

Boston, USA

1990-02-22

yana.grant@hotmail.com

My events

Please, fill out the self-declaration form below.

Self-declaration Form

voluntarily

First Name: Yana

Last Name: Grant

Gender
○ Male ● Female ○ Other

Date of Birth: 22.02.1990

Family Status: single

Old Address: Boston, USA

Phone No.:

Old Landlord:

Current Employer:

Current monthly income:

Datei auswählen Keine ausgewählt

IBAN:

Personal ID No.:

Doc Type: no

Occupation: student

Nationality

DSGVO: no

Surety: no

Pet: no

I hereby declare that I acknowledge...(legal statement)

Submit

Powered by Al3xa

Figure 11: Example of a self-declaration form within the tenant account.

3. Results

When starting this project, there was no lack of ideas but rather a lack of knowledge which prevented this project from happening in earlier years.

The intention of this project was to build a fully automated system. That, however, worked out only partly.

Things that work:

- Setting up a basic website
    - Using HTML, CSS, Bootstrap, JavaScript
- Setting up a database with phpMyAdmin
    - Select, insert, update, delete
    - SQL queries to retrieve and update records connected to more than one table in the database and including PHP defined variables
    - Table structures: taking functional dependencies into consideration (3NF)
- Turning the website into a dynamic website applying PHP coding, e.g.
    - for apartment framework
    - for upcoming vacant apartments
- Creating accounts and sessions for users
    - Password encryption using *password_hash* function and SALT
    - not fully implemented on all webpages

Things that need to be implemented:

Style-related:

- Fixing the sidebar
    - ○ When scrolling, the sidebar should not move
- Fixing the navigation bar
    - ○ When scrolling down the page, the sticky navigation bar is meant to remain completely visible on the top of the page and not turn transparent when scrolled up
- Replacing registration and login page with a dropdown div including JavaScript
    - ○ Instead of keeping two individual pages for both registration and login, those two features should be implemented in the top navigation bar. An example of how it is supposed to appear is displayed in Figure 12.

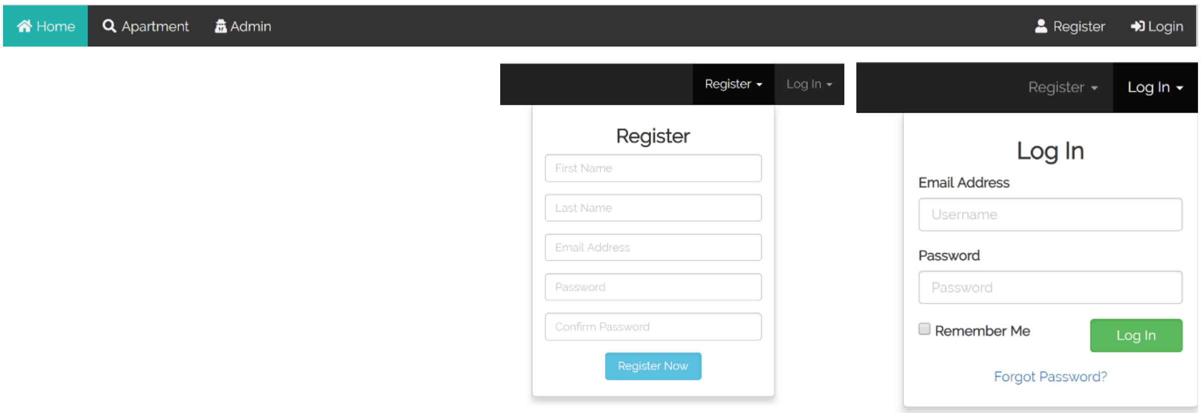
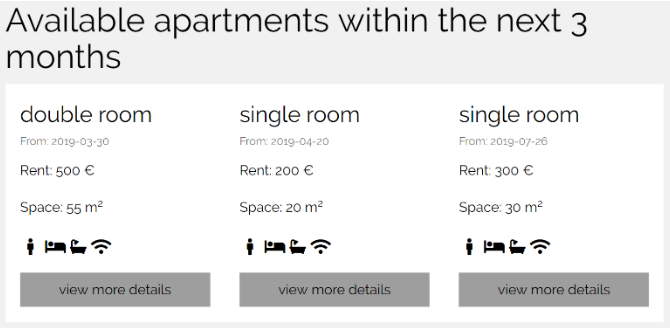


Figure 12: Dropdown div's applying JavaScript to be implemented in 'Home' and 'Apartment' webpages

- Add pictures to each div displaying the key data for each apartment



Available apartments within the next 3 months

double room
From: 2019-03-30
Rent: 500 €
Space: 55 m²
view more details

single room
From: 2019-04-20
Rent: 200 €
Space: 20 m²
view more details

single room
From: 2019-07-26
Rent: 300 €
Space: 30 m²
view more details

Figure 13: Current design of displaying key data for each apartment.

Functionality-related:

- Search engine implementation:
    - Finish writing all six SQL query options in PHP if Statements
- Figuring out why certain characters are not displayed properly in the tables that retrieve data from phpMyAdmin even though it is displayed correctly in the database
    - e.g. words including special characters such as ‘ß’ or ‘ä’,’ö’,‘ü’
- Testing the final website in other browsers than Chrome:
    - Safari
    - Firefox
    - Edge

All in all, this project helped a lot to gain basic insight of what is needed to proceed beyond the project’s deadline.

4. Outlook

This project needs to be further developed before it will become eligible to be used in reality.

Therefore, several steps still need to be performed:

- Sessions need to work throughout all accessible tabs
- Security need to be included
- Legal check by lawyers (e.g. imprint, terms and conditions)
- Setting language tags for further translations

Moreover, an application for tablets and smartphones is intended to support the administration process. Examples are:

- Allow a direct upload of pictures on the day an apartment is handed over to the tenant
- Improve the ease of use for tenants to directly make reports if necessary
- Check / schedule laundry machines directly
- Provide feedback possibilities