

# SHARED APARTMENTS WEB APPLICATION

TEAM 8

---

## Software Requirements Specification

---

*Developers:*

Michael GRÜNIG

Sara PEETERS

Daniel ZILTENER

*Customer:*

Haidar OSMAN

Version	Date	Revision description
0.01	17/10/2014	First SRS in Latex
0.02		
...		
1.0		

November 11, 2014

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Purpose . . . . .	2
1.2	Stakeholders . . . . .	2
1.3	Definitions . . . . .	2
1.4	System goals . . . . .	2
1.5	System overview . . . . .	2
1.5.1	current state and location . . . . .	3
1.6	References . . . . .	3
<b>2</b>	<b>Specific requirements and definitions</b>	<b>3</b>
2.1	System element definitions . . . . .	3
2.1.1	Profile . . . . .	3
2.1.2	Ad . . . . .	4
2.2	Non-functional requirements . . . . .	4
<b>3</b>	<b>Overall description</b>	<b>5</b>
3.1	Use-case diagram . . . . .	5
3.2	Use cases . . . . .	5
3.2.1	Search for relevant ads . . . . .	5
3.2.2	See room/apartment information . . . . .	6
3.2.3	Bookmark ads . . . . .	7
3.2.4	Subscribe to search alert . . . . .	7
3.2.5	send enquiry . . . . .	8
3.2.6	Register for on-site visit . . . . .	9
3.2.7	Place ad . . . . .	9
3.2.8	Set visiting dates . . . . .	10
3.2.9	Manage ad . . . . .	11
3.2.10	Manage enquiries . . . . .	12
3.2.11	Manage on-site visits . . . . .	12
3.2.12	Compile list of promising candidates . . . . .	13
3.2.13	Log in to the website . . . . .	14
3.2.14	Manage user profile . . . . .	14
3.2.15	Manage users . . . . .	15
3.2.16	Manage ads . . . . .	15
3.3	Actor Characteristics . . . . .	16

# 1 Introduction

## 1.1 Purpose

This Software Requirements Specifications (SRS) are the main reference document concerning the Shared Apartments Web Application Rentr. Principle reading parties of the document are the customer (Haydar Osman), and Team 8: Michael Grnig, Sara Peeters and Daniel Ziltener, the developers.

Written in a language understandable for all parties, it aims at giving a complete overview of the project, as currently agreed on. To this end, terms acronyms and abbreviations required to properly interpret the SRS are declared under the Definitions subsection of this section. Furthermore this introductory section contains a quick overview of the project stakeholders, as well as the system under development. A complete list of references to further documents used in the project is also provided.

In the second section functional and non-functional requirements are enlisted. As for the functional requirements we provide a detailed definition list of the main components. How the components will interact is not described in this section. Therefore we refer to the use cases in section 3. We aim to keep this list complete, unambiguous and the requirements verifiable. Requirements will be ordered to priority.

The third section provides an in-depth overview of the system under development from a user point of view. Use cases are schematically represented and described in detail. Also user characteristics are analyzed and described in this section.

The developers commit to updating the document after each customer-meeting, assuring that it represents the project specifications in the best possible way at any time. The customer commits to report mistakes and unclarities in the document to the developers, so that these can be corrected.

## 1.2 Stakeholders

**The customer** is the ordering party in this contract. He will own and operate the web application after development.

**A Client** of the web application is a internet user who registers to use the services provided by the web application developed.

## 1.3 Definitions

*There are no definitions or abbreviations in this section yet. If any of the terms used in the document are unclear, please notify us, so it can be added to this section.*

## 1.4 System goals

To have the best features of the three pages listed:

wgzimmer.ch,

students.ch/wohnen

and tutti.ch/bern/immobilien/wg-zimmer.

## 1.5 System overview

The system consists of a client-side website and a server-side application containing all the logic, using Java EE as the platform, the Spring Framework for structure, Spring Data together with MySQL for data storage, JSF for webpage structure and Persona as authentication framework.

Users have to log in to the website using Persona and create a profile. Users can then create ads or search for ads. Searches can be stored as alerts, and the user gets notified whenever a new ad is placed matching the criteria. Ad placers can place dates for visits where other users can

subscribe. They can also manage visits and enquiries on the webpage. E-Mail-notifications are available. The administrator can modify and delete users and ads.

### 1.5.1 current state and location

After compiling the web-app can be found at <https://localhost:8443/>.

Currently implemented usecases are:

- As a user I want to place an ad for advertising a room in a shared apartment
- As a user I want to search for relevant ads in a specific area
- As a user I want to get exhaustive information about a room (e.g. pictures, location, roommate profiles, ..)

## 1.6 References

*There are no external references yet. They will be added to this section as needed.*

## 2 Specific requirements and definitions

### 2.1 System element definitions

#### 2.1.1 Profile

Each normal user has a profile. A profile consists of 2 parts: An obligatory part and an optional part. Each element contained in a user profile is either visible (V) to other users, or not ().

Obligatory profile:

- (V) Name
- (V) Surname
- () Email address
- () Password
- (V) Sex
- (V) Age
- () Email options for search alerts (always (default), daily digest, weekly digest, no email)
- () Email options for reactions on ads (always (default), daily digest, weekly digest, no email)
- () Email options for other occasions (always (default), daily digest, weekly digest, no email)

Optional profile:

- (V) profile picture
- (V) Description

### 2.1.2 Ad

Ads are an important component of the web application. Some, but not all elements of an ad are searchable (S), which means that a search can in some way select for this criterium.

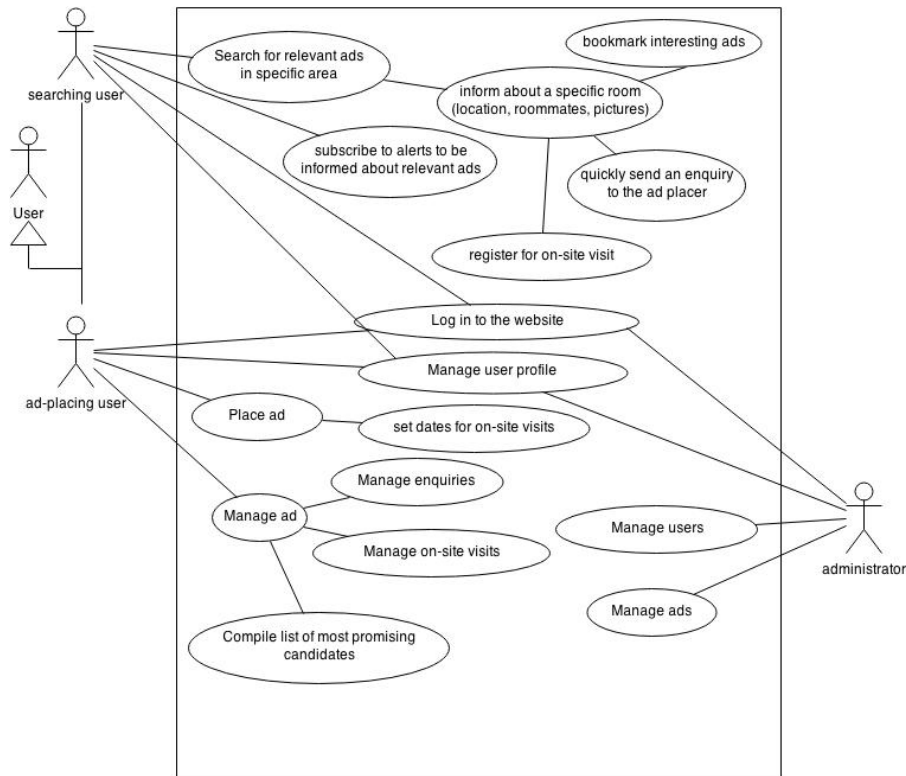
- () Title
- (S) Address  
Note: address consists of Street, number, ZIP-code and city. ZIP-code is searchable.
- (S) Category (shared apartment or apartment)
- (S) Price  
Note: Price is searchable via a from-to-field.
- (S) Room size  
Note: searchable via a from-to-field.
- (S) Free from Date  
Note: searchable by indicating a date and one of the options (strict, or earlier, or later).
- (S) Free until  
Note: can be unlimited, or a date is searchable by indicating no limit or a date and one of the options (strict, or earlier, or later).
- (S) Languages spoken
- (S) Keywords  
Note: Keywords help to personalize a search, without getting lost in the different terms that can be used for the same thing in a description. They are selected from a list. Some examples: Vegetarian/Vegan, Non-vegetarian, Quiet neighbourhood, Party-zone, Shared garden/Balcony, bike parking, staying weekends, eating/cooking together...
- () Pictures
- () Description
- (S) Roommates  
Note: Searchable on sex (all male, all female, no preference). Searchable on age via a from-to-field. Number of roommates is searchable via a from-to-field.
- () response method: enquiry and/or timeschedule
- () Deactivated

## 2.2 Non-functional requirements

- Secure login and data User should confirm registration through email link. Captcha is used in registration form. passwords must have a minimum length maximum number of login attempts from the same IP.
- Logging All site traffic is logged in different log files, documenting each user action.
- Responsive design The website adapts to mobile users (smartphones and tablets). The website will be tested in the latest Firefox and Chrome version. (On which mobile devices will we test our website)

## 3 Overall description

### 3.1 Use-case diagram



### 3.2 Use cases

#### 3.2.1 Search for relevant ads

##### Actors

Client; role: searching user.

##### Description

As a user, I want to search for relevant ads in a specific area.

##### Trigger

User starts filling in the search form.

##### Pre-conditions

User is logged in.

##### Post-conditions

User sees a list of relevant ads according to his search criteria Search is persisted to database.

##### Main scenario

1. User indicates he wants to search for shared apartments.
2. User enters search area via zip code or map position and radius.

3. User enters price range.
4. User enters range of number of roommates
5. User enters preferred sex of roommates, if any.
6. User hits search button.
7. User sees list of relevant search results.

#### **Alternative scenarios**

**1a.** User indicates he wants to search for normal apartments

1. Step 2 and 3 as in main scenario
2. User enters range for number of rooms
3. continue to step 6 of main scenario

**3a.** User doesn't want to enter further details and hits the quick search button.

1. continue to step 7 of main scenario

**7a.** After seeing the search results user wants to change his search criteria

1. User hits change search criteria button
2. User changes the desired criteria
3. continue to step 6 of main scenario

#### **Notes**

Other search criteria? eg.: free from..., pre-defined keywords: vegetarian, rural, shared garden,...

### **3.2.2 See room/apartment information**

#### **Actors**

Client; role: searching user.

#### **Description**

As a user, I want to get exhaustive information about a room.

#### **Trigger**

User clicks on interesting ad, either after doing a search, or after opening list of bookmarked ads.

#### **Pre-conditions**

User is logged in.

#### **Post-conditions**

User sees room/apartment information

#### **Main scenario**

1. User sees the following:
  - map with indication of apartment location
  - pictures of the apartment/room details (price, )
  - room/apartment description
  - link to roommates profiles
2. User can choose to sign up for an on-site visit, or to send an inquiry (depending on what the ad-placer allowed) or to bookmark the ad

3. User can go back to search results.

**Notes**

Do we want to show short user profiles of the roommates or just a link to the roommate profiles?

**3.2.3 Bookmark ads****Actors**

Client; role: searching user.

**Description**

As a user I want to bookmark interesting ads

**Trigger**

User clicks bookmark ad option when viewing an ad.

**Pre-conditions**

User is logged in.

**Post-conditions**

User gets a message that the ad is bookmarked.

List of bookmarked ads of user is updated.

In case of an error, or the user has already bookmarked this ad previously, a message stating this appears.

**Main scenario**

1. ad is bookmarked, message appears.

**Alternative scenarios**

**1a.** couldnt bookmark the ad because ad is already bookmarked.

1. ad already bookmarked message appears.

**1b.** User realises he has bookmarked the wrong ad

1. User goes to my bookmarks
2. User clicks delete button next to the bookmark that has to be deleted

**3.2.4 Subscribe to search alert****Actors**

Client; role: searching user.

**Description**

As a user I want to subscribe to alerts to be informed about relevant ads.

**Trigger**

User hits search-alert button at the end of the search form.

**Pre-conditions**

User is logged in.

User Has filled in at least some search criteria in the form.

**Post-conditions**

Search criteria are linked to user profile in the database.

User sees a success message.



### **Main scenario**

1. User clicks search-alert button after filling in search form, or after viewing the list of ads corresponding to his search.
2. User can review search criteria
3. User hits save button
4. User sees search successfully saved message

### **Alternative scenarios**

#### **2a.** User notices a mistake in search criteria

1. User hits edit search criteria
2. User corrects mistake
3. Continue with step 3

### **3.2.5 send enquiry**

#### **Actors**

Client; role: searching user.

#### **Description**

As a user I want to quickly send an inquiry to the ad-placer.

#### **Trigger**

User views an interesting ad, and clicks the send inquiry button.

#### **Pre-conditions**

User is logged in.

Ad placer has enabled searchers to send him a message.

#### **Post-conditions**

Message has been send by email to the ad placer.

Message is persisted to database.

User sees message send screen.

### **Main scenario**

1. User clicks on send inquiry button
2. User enters his message.
3. User clicks send button
4. User views succes message.

#### **Notes**

Does the user need to be able to edit his message after it has been send?

### 3.2.6 Register for on-site visit

#### Actors

Client; role: searching user.

#### Description

As a user I want to register for an on-site visit.

#### Trigger

User is viewing an ad and hits register for visit button.

#### Pre-conditions

User is logged in.

Ad placer has opened some timeslots for on-site visits.

Not all timeslots are completely full yet.

User has not yet registered for a visit of this apartment

#### Post-conditions

User is registered for a time slot (in database)

Number of free places for timeslot is reduced by one.

User cannot register for other timeslots.

User has received a confirmation email.

#### Main scenario

1. User hits register for on-site visit button.
2. User sees open time slots and selects one.
3. User clicks register button

#### Alternative scenarios

**2a.** User sees that he cannot attend any of the free timeslots

1. User cancels registration
2. If ad placer has opened possibility to send inquiry, user can send an enquiry instead.

#### Notes

Should unregistering be possible?

### 3.2.7 Place ad

#### Actors

Client, role: ad-placing user

#### Description

As a ad-placing user I want to place an ad for the shared apartment.

#### Trigger

User triggers the My Ads button

#### Pre-conditions

Customer is logged in

#### Post-conditions

Ad saved in the data base.

User sees his ad and a confirmation

## **Main scenario**

1. User enters the My Ads page, which contains a list of the already placed ads.
2. User pushes the new button
3. User fills in the data relevant for the living community.
4. User uploads pictures.
5. User indicates he wants to allow interessants to send an enquiry message
6. User saves the data.
7. User sees the ad as it would appear on the website and clicks the confirm button.
8. User will be directed to the My Ads page as in 1.

## **Alternative scenarios**

**5a.** User wants to set timeslots for visits instead

1. Step 6 as in main scenario
2. User enters user case 9.

**6a.** User saved form with invalid data.

1. User gets an info of the incorrect insertion.
2. Use case resumes on step 3.

**7a.** User sees ad and realizes he wrote something wrong.

1. User pushes the Edit button
2. Use case resumes on step 3

## **Special requirements**

User can only place a certain number of ads. (otherwise abuse has to be assumed)

## **Notes**

Maximum size of ads per user?

### **3.2.8 Set visiting dates**

#### **Actors**

ad-placing user

#### **Description**

As ad-placing user I want to set dates for on-site visits so that searching users know when they can visit the flat.

#### **Trigger**

User has selected set timeslots where interessants can register for visits in the ad placing form and clicked Save

OR

User clickes add more timeslots button next to a previously placed ad in the My Ads page.

#### **Pre-conditions**

User is logged in.

User has placed an ad.

**Post-conditions**

on-site visit dates are linked to ad in database

Ad-placing user sees the new dates

**Main scenario**

1. User clicks button (see trigger).
2. User inserts the dates
3. User inserts time slots
4. User sets number of users per time slot
5. User clicks save time slots

**Alternative scenarios**

**1a.** User has not placed any ad.

1. User gets the information that no ad has placed yet.

**5a.** User inserts an invalid date.

1. User gets the information that the date is invalid.
2. Use case resumes on step 2

**3.2.9 Manage ad****Actors**

Ad-placing user

**Description**

The ad-placing user wants to manage his ads.

**Trigger**

User clicks my ads-Button.

**Pre-conditions**

User logged in

User has created an ad

**Post-conditions**

User sees changed ad

Changes are persisted to the database

**Main scenario**

1. Clicks my ads-Button.
2. Clicks on the ad he wants to edit
3. Changes ad content
4. Clicks the save button
5. Sees the changed ad

**Alternative scenarios**

**2a.** User wants to delete the ad

1. Clicks the delete button next to the ad.
2. User confirms the deletion
3. Ad is deleted and user is redirected to my ads page

### 3.2.10 Manage enquiries

**Actors**

Ad-placing user

**Description**

As a user I want to manage the enquiries.

**Trigger**

User clicks the enquiries button next to the ad in the my ads page.

**Pre-conditions**

User is logged in

Enquiries stack is not empty.

**Post-conditions****Main scenario**

1. User clicks the enquiries button.
2. User sees list of messages, including a sender profile picture a reply and delete button next to each message.
3. User reads message.
4. User double clicks on sender profile picture to see the complete profile of the candidate
5. User clicks back to go back to the enquiries
6. User clicks reply
7. User writes a reply message
8. User clicks send

**Alternative scenarios**

**6a.** User wants to delete the enquiry

1. User clicks delete button.

### 3.2.11 Manage on-site visits

**Actors**

Ad-placing user

**Description**

As a user I want to see how the timeslots I opened for flat visits are being filled up

**Trigger**

User clicks the visiting times button next to the ad in the my ads page.

**Pre-conditions**

user logged in

**Post-conditions**

Visiting dates and registered users are shown

**Main scenario**

1. Clicks the visiting times button.

2. User sees a list of opened timeslots and registered candidates.
3. User can click on candidates names to see their full profile
4. User goes back to visiting times page

#### **Alternative scenarios**

**3a.** User can click add more timeslots if he sees that only little places are left

1. User enters user case 9

**3b.** User can change the number of allowed visitors for each timeslots as long as it does not go below the number of already registered candidates.

1. User changes the number
2. User clicks save changes button.

**3c.** User can delete a timeslot completely, sending already registered users a message that the timeslot is deleted and that they can contact the ad placer or register for another timeslot.

1. User clicks delete button next to timeslot.
2. User confirms deletion
3. An automatic message to already registered users is compiled.

### **3.2.12 Compile list of promising candidates**

#### **Actors**

Ad-placing user

#### **Description**

As a customer I want to compile a list of most compromising candidates.

#### **Trigger**

User clicks on Candidates button next to ad in My Ads page

#### **Pre-conditions**

User logged in

At least one candidate has registered for a visit or sent an enquiry.

#### **Post-conditions**

Changes are persisted to the database

#### **Main scenario**

1. User clicks on Candidates button
2. User sees a list of all candidates that have somehow reacted on the ad
3. User can click through to see full candidate profile
4. User check favorite box next to his favorite candidates
5. User can click compile favorite list at the top of the page to see a shortlist of favorite candidates.

### 3.2.13 Log in to the website

**Actors**

Ad-placing user, searching user, administrator

**Description**

The users want to have a profile with which to manage contact data and ads.

**Trigger**

Login is mandatory, so upon visiting the website, the user clicks the Login link.

**Pre-conditions**

None.

**Post-conditions**

User or administrator is logged in

**Main scenario**

1. Actor clicks the Login link.
2. The Persona login page opens and the user logs in.
3. The user is redirected to their profile page.

**Alternative scenarios**

**3a.** The user has logged in for the first time, and after redirect is asked to choose a user name (and, depending on requirements, further information).

### 3.2.14 Manage user profile

**Actors**

All users

**Description**

As a user, I want to be able to change my contact information, password, as well as manage my alerts, ads and favourites.

**Trigger**

User clicks the Profile link on the page.

**Pre-conditions**

The user is logged in.

**Post-conditions**

The data changed by the user is persisted to the DBMS.

The user is notified about the saved changes, or if it failed, why it did so.

**Main scenario**

1. Actor clicks the Profile link.
2. On the profile page, they can change their account data, ads, subscriptions and favourites.
3. Upon clicking Save the data gets stored if valid, and the user gets a notification about it on the profile page itself.

**Alternative scenarios**

**2a.** If the user deletes their account, they get redirected to the main page.

### 3.2.15 Manage users

#### Actors

The administrator.

#### Description

The administrator wants to be able to see all user data, and be able to delete and change users.

#### Trigger

The administrator clicks the Admin panel link next to the Profile link.

#### Pre-conditions

The user is logged in

The user is marked as administrator

#### Post-conditions

The changes are persisted to the database if they're valid

The admin gets notified about that

#### Main scenario

1. Administrator clicks on Admin panel link
2. The admin searches for a user using the admin panel search function
3. The admin selects a user from the user list
4. The user's profile page is displayed, where the admin can change the user's data and settings, as well as delete the user
5. After all changes are done, the admin leaves the page

### 3.2.16 Manage ads

#### Actors

The administrator

#### Description

As administrator, I want to be able to change ads on behalf of users, and delete ads for a number of reasons.

#### Trigger

The administrator clicks the Admin panel link next to the Profile link.

#### Pre-conditions

The user is logged in

The user is marked as administrator

#### Post-conditions

The changes are persisted to the database if they're valid

The admin gets notified about that

#### Main scenario

1. Admin clicks on Admin panel link
2. The admin first searches for a user (as described in scenario 16, Manage users), then selects an ad from that user's list of ads



3. The admin selects an ad from the ad list
4. The admin searches for an ad using the admin panel search function
5. On the ad management panel (the same as for the ad-placing user) the admin can change all data, as well as delete the ad
6. After all changes are done, the admin leaves the page

### **3.3 Actor Characteristics**

All users are logged in and have a minimal user profile (name, email address,...)

The clients of our website are people looking for a (shared) apartment in Switzerland. The main user group will be between aged 18 - 29 years old. We expect most but not all apartments to be in and around university cities. These users can place ads for (shared) apartments or search the ads. We expect these group of users to be the largest group of actors.

Another actor group is the administrator group. This is a small group of 1 to 5 people, who can view all information about users and ads. Members of this group are permitted by our customer.