

Project Requirements

Balz Aschwanden, Raoul Grossenbacher, Florian Jörg

October 22, 2014

Project Name: StudiHome

Team: 6

Costumer: Bledar Aga

Revision History

Version	Date	Revision Description
0.01	October 8. 2014	<ul style="list-style-type: none">• Added use cases and diagram• Changed user called <i>student</i> to <i>tenant</i>• Changed functional requirements for advertisements
0.02	October 22	<ul style="list-style-type: none">• Added minimal information requirement to use case <i>Generate Advertisement</i>• Added use case <i>Login</i>
0.03		
0.04		

1 Introduction

Purpose

- The aim of this project is to create a platform where rooms in shared flats and whole apartments can be rented and advertises.

Stakeholders

- StudiHome Inc. (CEO Bledar Aga)

References

- wgzimmer.ch
 - Communication between advertiser and tenant.
- students.ch
- tutti.ch
- immoscout24.ch
 - Search
 - List of favorites
 - Property alerts
- homegate.ch

2 Overall description

Actor characteristics

- Advertiser
 - Shared flat looking for room mate(s)
 - Landlord renting flat
- Tenant
 - Searching for room in shared flat
 - Searching for flat

3 Requirements

Functional requirements

- Login
 - Name
 - Valide mail address
- Advertisement
 - Rent

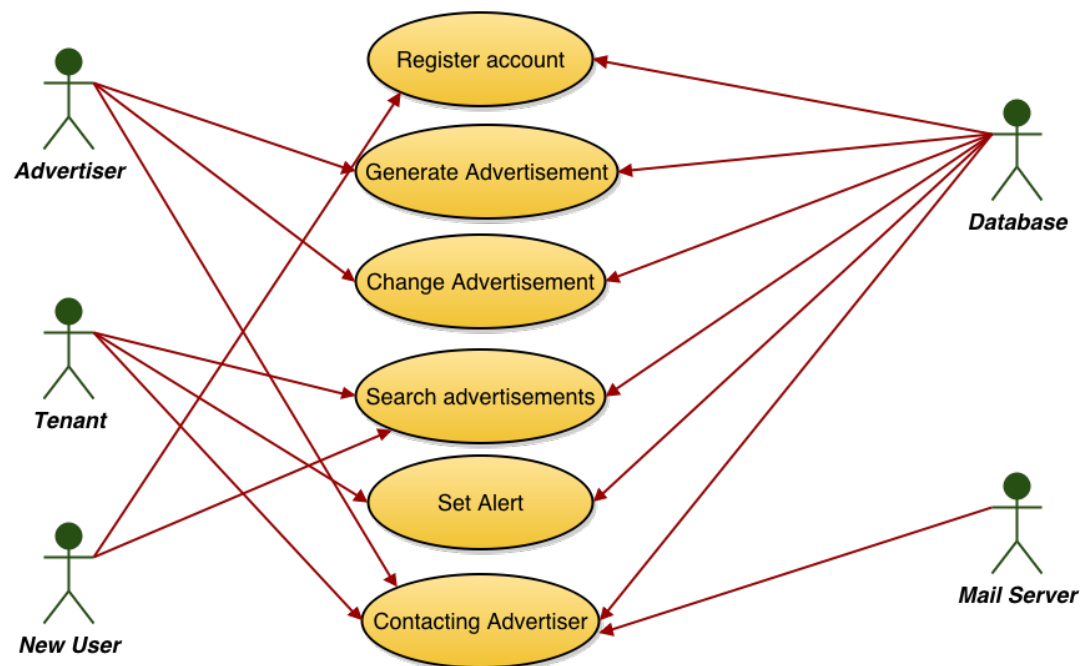
- Room/Apartment size
- Number of current inhabitants
- Address
- Expiration date (can be prolonged)
- Pictures
- Description (free text)
- Area map

Non-functional requirements

- Only one account per user
- No difference between tenant and advertiser account
- Usable on mobile devices and computers (autoscaling)

4 Use cases

Use case diagram



Login

- Actors

- User
- Description
 - Users have to have an account and have to be logged in in order to use advanced features like bookmarking, setting alerts and creating ads
- Trigger
 - User clicks the "login button"
 - User is not logged in and clicks on one of the advanced features
 - * Example: User wants to bookmark an ad after a search
- Pre-Condition
 - User is not logged in
- Post-Condition
 - User has an account
 - User is logged in
- Main Scenario
 - A new user comes to studihome and would like to create an ad
 - After clicking on the "create ad" button, he is asked to log in or to register an account
 - If he does not have an account, he can register one (see use case "Register account")
 - If he has an account, he can log in with his credential and is redirected to the appropriate page

Register account

- Actors
 - User
- Description
 - Every user has to have account and has to be logged in if he wants to use the more advanced features of our service.
- Trigger
 - Click the "create new account" button

- Pre-Condition
 - A (formally) valid email address has been entered in the form
 - A password has been entered
- Post-Condition
 - A new user has been generated
 - The user is brought to his home screen
 - Optional: The user receives a confirmation email
- Main Scenario
 - A new user comes to our web page. He does not yet have an account.
 - He enters his email and a password of his choice
 - A new account is created.

Generate Advertisement

- Actors
 - Advertiser
- Description
 - An advertiser wants to create an ad for his property and publish it, so interested tenants can access it.
- Trigger
 - The event is initialized by pressing the button "new advertisement"
- Pre-Condition
 - User is logged in
- Post-Condition
 - An advertisement object is created
 - * The minimal information requirements are
 - address
 - rent
 - room size
 - room number

- * The advertiser is free to give additional information if required
 - The ad is visible in "my ads" on the advertisers page
 - The ad is visible for other users and can be searched for
- Main Scenario
 - The advertiser clicks the appropriate button
 - A UI is opened where he can enter all the information he wants
 - He submits his ad by clicking the appropriate button
 - He is redirected to his home

Change Advertisement

- Actors
 - Advertiser
- Description
 - Advertisements have a definite lifespan and they might change. The advertiser can therefore edit his own advertisements, close them or keep them open for longer than normal.
- Trigger
 - Clicking the appropriate button in the web gui
- Pre-Condition
 - Advertisement is open
 - User is logged in
 - User owns the advertisement (Has created it himself)
- Post-Condition
 - Changes are uploaded and the ad is saved
 - Optional: Tenants having bookmarked this ad are informed about the change
- Main Scenario
 - A advertiser realizes, he has forgotten an important information in his advertisement or that it is about to be expired
 - He logges on to his account, finds the ad in question and clicks "edit"

- The edit view is opened and he can make his modifications and prolong the ad
- He submits the changed form and is brought back to his home screen

Search advertisements

- Actors
 - Tenant
 - New User (no account)
- Description
 - The basic search function in which users can search for advertisements.
- Trigger
 - The user clicks search in the main window
 - The user hits the enter button while the search window is active
- Pre-Condition
 - None
- Post-Condition
 - A list with all the ads matching the search criteria is shown
 - (The list is just the list off all open ads if no keywords have been entered.
- Main Scenario
 - A user finds StudiHome (our web service) and uses the search field.

Set Alert (Tenant)

- Actors
 - Tenant
- Description
 - The tenant wants to be informed if a new ad matching certain user defined criteria is published.
- Trigger

- An alarm is set in the search window
- Pre-Condition
 - One or more search keyword have been entered
 - The user is logged in
- Post-Condition
 - An alarm is set
 - The user will receive a message every time an ad matching his criteria is published
- Main Scenario
 - A logged in user is searching for an item but he is not happy with the result
 - He can set his current search queue as alarm
 - Every time a new item with these criteria is published, the user receives a message

Contacting Advertiser

- Actors
 - Advertiser
 - Tenant
- Description
 - Tenant is interested in advertisment and wants to contact advertiser to arrange visite.
- Trigger
 - Tenant clicks “interested button” an advertisement
- Pre-conditions
 - Tenant is logged in.
 - Advertisement is open (not closed or expired)
 - Tenant and Advertiser have valid email addresses
- Post-conditions
 - Interest is displayed in web app for advertiser
 - Advertiser receives notification email

- Advertisement is marked as “i’m interested”
- Main Scenario
 - Tenants clicks “interested” button
 - Advertiser receives notification

Advertiser invites tenant for visit

- Actors
 - Tenant
 - Advertiser
- Description
 - Advertiser wants to arrange visits with tenants who are interested in his advertisement
- Trigger
 - Advertiser receives notification (See other use case)
- Pre-Condition
 - Advertisement is open
 - Notification email was received
- Post-Condition
 - Tenant receives invitation email
 - message is marked as “invited” (Advertiser)
 - "Doodle" is generated
 - "Doodle" has expiration date
- Main Scenario
 - Advertiser receives notification email
 - Advertiser logs on and sees, who’s interested in his advertisement
 - Advertiser select message ...
 - ... and select time slots for visite
 - Click “send”
 - Notification email is send to tenant
 - Message is marked as “invited”

Advertiser writes message to tenant

- Actors
 - Advertiser
 - Tenant
- Description
 - Advertiser wants to send a message to an interested tenant
- Trigger
 - Advertiser receives notification (See other use case)
- Pre-Condition
 - Advertisement is open
 - Notification email was received
- Post-conditions
 - Tenant receives notification mail
 - Tenant receives message (in web app)
 - message is marked as “responded to” (Advertiser)
- Main Scenario
 - Advertiser receives notification email
 - Advertiser logs on to web app and sees, who’s interested in his advertisement
 - Select message
 - Writes new message
 - Click “send”
 - Email is send to tenant
 - Message is marked as “responded to”

Rank promising tenants (Advertiser)

- Actors
 - Advertiser
- Description

- The advertiser has an overview over all the tenants who are interested in one of his ads. He can arrange them to his liking, sorting them to by the most promising candidate.
- Trigger
 - Advertiser changes to the appropriate view and moves a tenant in his list to another position (Drag and Drop)
- Pre-Condition
 - Tenants show a serious interest in an advertisement (Click the appropriate button)
 - Advertisement is still open
 - Advertiser is logged in and in appropriate view
- Post-Condition
 - List order of interested tenants is updated
- Main Scenario
 - A advertiser receives multiple interests for one ad. He has a preference for his property and wants the list of interested tenants to reflect this.
 - He can move tenants in his list to other positions

Rank interesting advertisements (Tenant)

- Actors
 - Tenants
- Description
 - A tenant is interested in several offers. He has preferences and wants the list of his bookmarks to reflect that.
- Trigger
 - Dragging and dropping a bookmarked ad to another position
- Pre-Condition
 - Ad is open
 - Ad is bookmarked
 - Tenant is logged in

- Post-Condition
 - List order of bookmarked adds is updated
- Main Scenario
 - A tenant logs in for his first search
 - He bookmarks several ads
 - He sees a list with all his bookmarks
 - By dragging and dropping, he can arrange them according to his liking
 - The first few ads will show up on his main page

Tenant is no longer interested in ad

- Actors
 - Tenant
- Description
 - A tenant changes his opinion and an ad he bookmarked or even declared a serious interest in (contacted the advertiser) is no longer seen as desirable. He therefore wants to remove the ad from his bookmarks
- Trigger
 - Clicking the appropriate button
- Pre-Condition
 - Ad is bookmarked
 - Ad is open
 - User is logged in
- Post-Condition
 - Ad is no longer listed under bookmarked ads
 - If the tenant has contacted the advertiser, the second is notified about the change
- Main Scenario
 - A tenant has received more information about a bookmarked ad and decides that he is no longer interested in it.
 - He selects the ad.
 - He clicks the button that removed