Craftsman Finder System

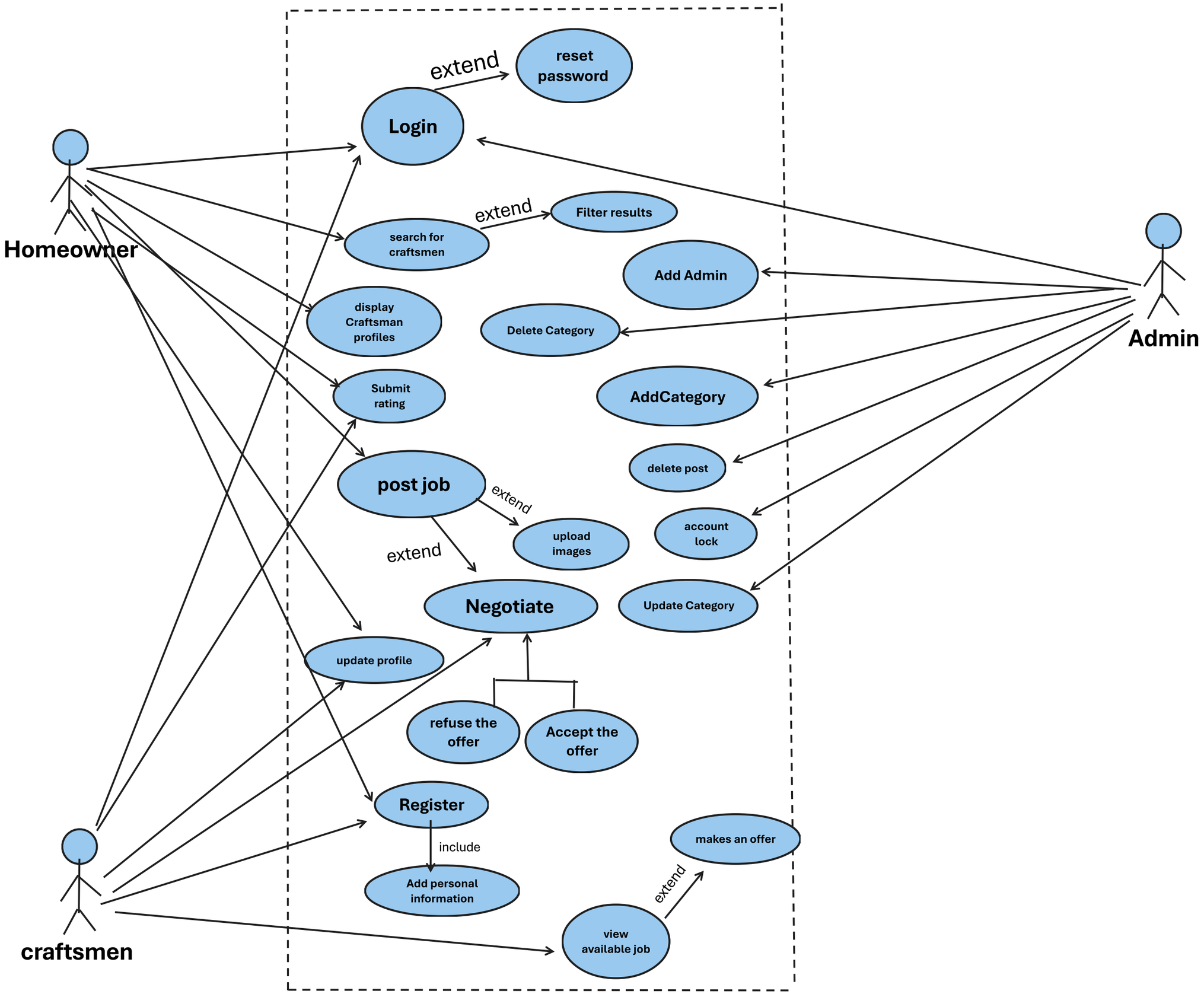
Functional Requirements

- Homeowners, officials and craftsmen have the ability to create an account (register).
- Homeowners will provide personal information, including name, email, address, and contact details.
- Users must log in using their registered credentials (email and password).
- Users should be able to reset their password in case they forget it.
- Homeowner’s update their profile, including contact information and address.
- Craftsmen can update their profile with new certifications, services offered, and service areas.
- Homeowners can search for craftsmen by service type (e.g., plumbing, electrical, carpentry) and location.
- Homeowners have the ability to filter search results by ratings, price range, availability and proximity.
- Homeowners have the ability to view detailed craftsman profiles, including reviews, ratings, testimonials and prices.
- Homeowners can post job requests with detailed descriptions, including the type of service, location, and desired completion date.
- Homeowners can upload images to provide craftsmen with a clearer idea of the work required.
- Homeowners can negotiate the offer..
- Craftsmen can negotiate the offer.
- Both Craftsmen and Homeowners can accept the offer.
- Craftsmen can provide Homeowners Ratings.
- Craftsmen can view available job listings and submit bids or quotes.
- After a job is completed, homeowners can rate craftsmen on a 5-star scale and leave detailed reviews.

- Reviews and ratings must be displayed on craftsman profiles and factor into search rankings.
- The admin has the ability to delete any evaluation or post that does not follow the guideline.
- The admin has the ability to create a lock for any account, whether a homeowner or a craftsman.
- Admin has the ability to do CRUD operation on categories
- The admin has the ability to add another admin.

Non-functional Requirements

- The system must handle 20,000 users without a significant drop in performance.
- The platform should have an intuitive user interface, allowing users to navigate easily.
- The system should be responsive and accessible on a variety of devices (desktop, tablet, mobile).
- • Data backups should be performed every day to prevent data loss in the event of a system failure.
- The system should support easy integration of new features and services as needed.
- User data must be handled according to privacy best practices, and no data should be shared without explicit user consent.
- Sensitive data must be stored encrypted in the database.


Homeowner
Login
extend
reset password
search for craftsmen
extend
Filter results
display Craftsman profiles
Submit rating
post job
extend
upload images
extend
Negotiate
update profile
refuse the offer
Accept the offer
Register
include
Add personal information
craftsmen
view available job
extend
makes an offer
Admin
Add Admin
Delete Category
AddCategory
delete post
account lock
Update Category

1. User Registration and Login

- **Actors**: Homeowner, Craftsman, Admin
- **Preconditions**: User must not already have an account in the system.
- **Main Flow**:
    1. User selects "Create Account" option.
    2. User provides required information: personal details (name, email, address, and contact details) for homeowners; for craftsmen, includes additional information like certifications and services.
    3. System validates input and creates an account.
- **Alternate Flows**:

    - Invalid or incomplete input results in an error message and prompts for correction.

2. Login

- **Actors**: Homeowner, Craftsman, Admin
- **Preconditions**: User must have an existing account.
- **Main Flow**:
    1. User enters registered email and password.
    2. System verifies credentials and grants access if they are valid.
- **Alternate Flow**:

    - Incorrect credentials prompt an error message.
    - If a user has forgotten their password, they can select "Reset Password."

3. Reset Password

- **Actors**: Homeowner, Craftsman, Admin
- **Preconditions**: User has a registered email.
- **Main Flow**:

1. User selects "Forgot Password."
2. System prompts the user to enter their registered email.
3. System sends a password reset link to the email.
4. User clicks the link and enters a new password.
5. System confirms the password update.

- **Alternate Flow**:
  - Invalid email input prompts an error.

4. Profile Update

- **Actors**: Homeowner, Craftsman
- **Preconditions**: User is logged in.
- **Main Flow for Homeowners**:
  1. Homeowner accesses the profile settings.
  2. Homeowner updates contact details or address.
  3. System saves the changes.
- **Main Flow for Craftsmen**:
  1. Craftsman accesses profile settings.
  2. Craftsman updates certifications, services offered, or service area.
  3. System saves the changes.

5. Search for Craftsmen

- **Actors**: Homeowner
- **Preconditions**: Homeowner is logged in.
- **Main Flow**:
  1. Homeowner selects "Search for Craftsmen."
  2. Homeowner specifies service type and location.
  3. Homeowner applies filters for ratings, price range, availability, and proximity.
  4. System displays relevant craftsman profiles.
- **Alternate Flow**:
  - No matches found prompts the homeowner to adjust filters.

6. View Craftsman Profiles

- **Actors**: Homeowner
- **Preconditions**: Homeowner has performed a search.
- **Main Flow**:
    1. Homeowner selects a craftsman profile to view.
    2. System displays details: reviews, ratings, testimonials, certifications, and price range.
- **Alternate Flow**:
    - Inactive craftsman profile shows limited details or prompts an error.

7. Post Job Request

- **Actors**: Homeowner
- **Preconditions**: Homeowner is logged in.
- **Main Flow**:
    1. Homeowner selects "Post Job Request."
    2. Homeowner specifies service type, location, completion date, and uploads images if needed.
    3. System saves and displays the job request in the craftsman job listings.
- **Alternate Flow**:
    - Missing details prompt an error message.

8. Submit Bids and Negotiate

- **Actors**: Craftsman, Homeowner
- **Preconditions**: Homeowner has posted a job, and craftsman views job listings.
- **Main Flow**:
    1. Craftsman submits a bid for a job request.
    2. Homeowner and craftsman engage in negotiations.
    3. Either party can finalize or decline the offer.
- **Alternate Flow**:

- No agreement leads to the job request being open for other bids.

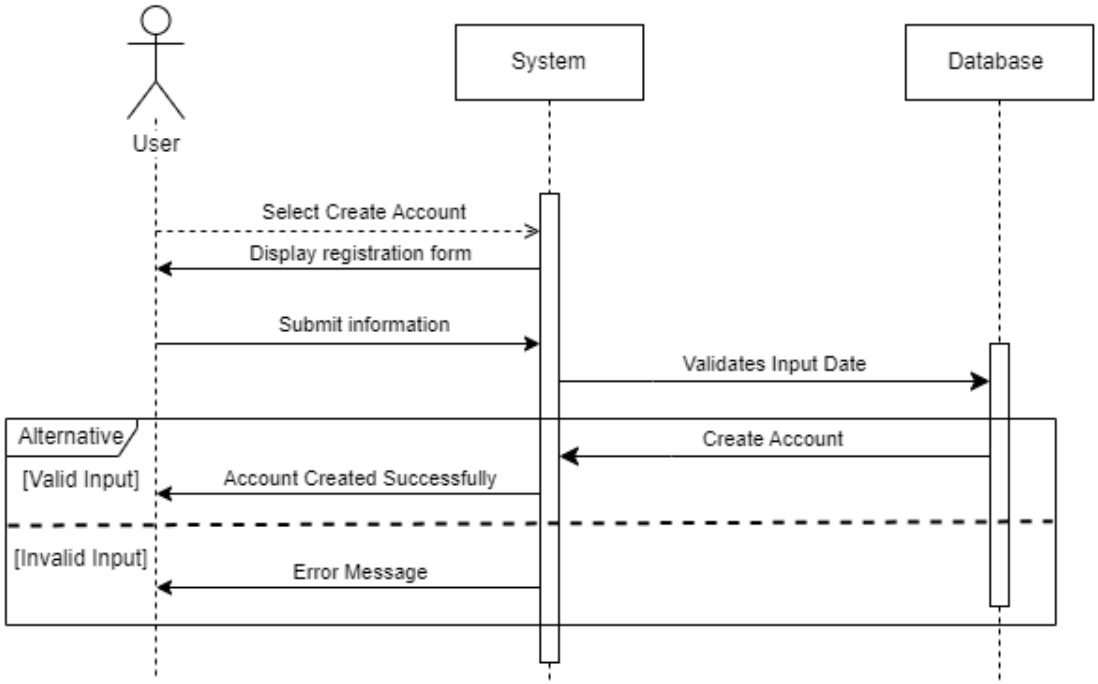
9. Rate and Review Craftsmen

- **Actors**: Homeowner
- **Preconditions**: Job request has been completed.
- **Main Flow**:
  1. Homeowner rates the craftsman on a 5-star scale.
  2. Homeowner leaves a review.
  3. System displays ratings and reviews on the craftsman's profile.
- **Alternate Flow**:
  - Inappropriate reviews may be flagged or removed by the admin.

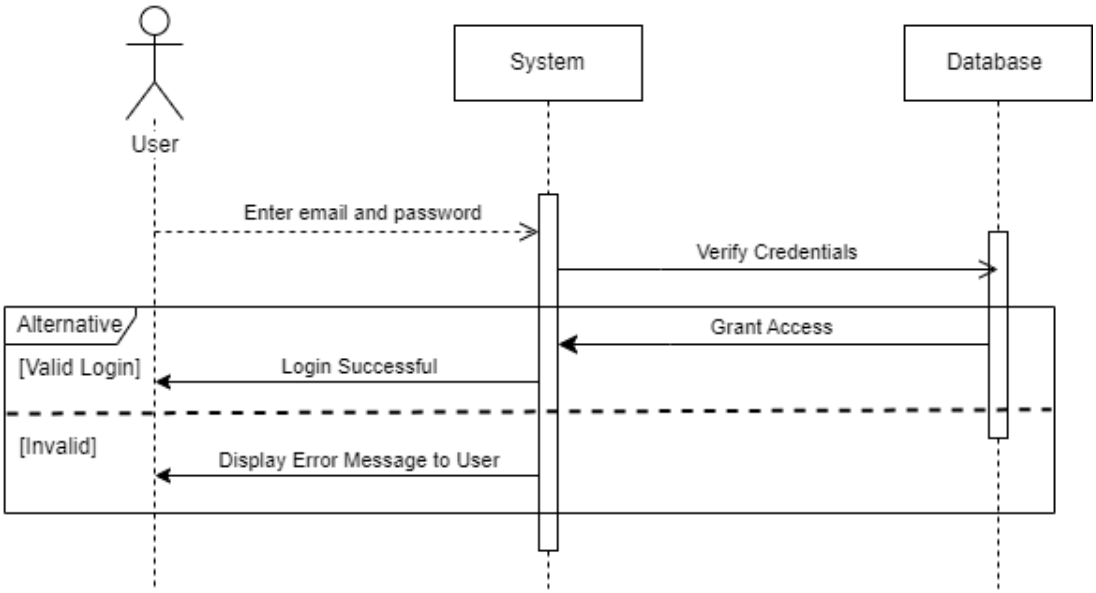
10. Admin Management of Content and Users

- **Actors**: Admin
- **Preconditions**: Admin is logged in with appropriate permissions.
- **Main Flows**:
  - **Delete Evaluations or Posts**:
    1. Admin reviews flagged evaluations or posts.
    2. Admin decides to retain or delete based on guidelines.
  - **Lock Accounts**:
    1. Admin identifies an account violating guidelines.
    2. Admin places a lock on the account, restricting access.
  - **CRUD on Categories**:
    1. Admin creates, updates, or deletes service categories as needed.
  - **Add New Admin**:
    1. Admin selects "Add New Admin."
    2. Admin enters details for the new admin and assigns permissions.

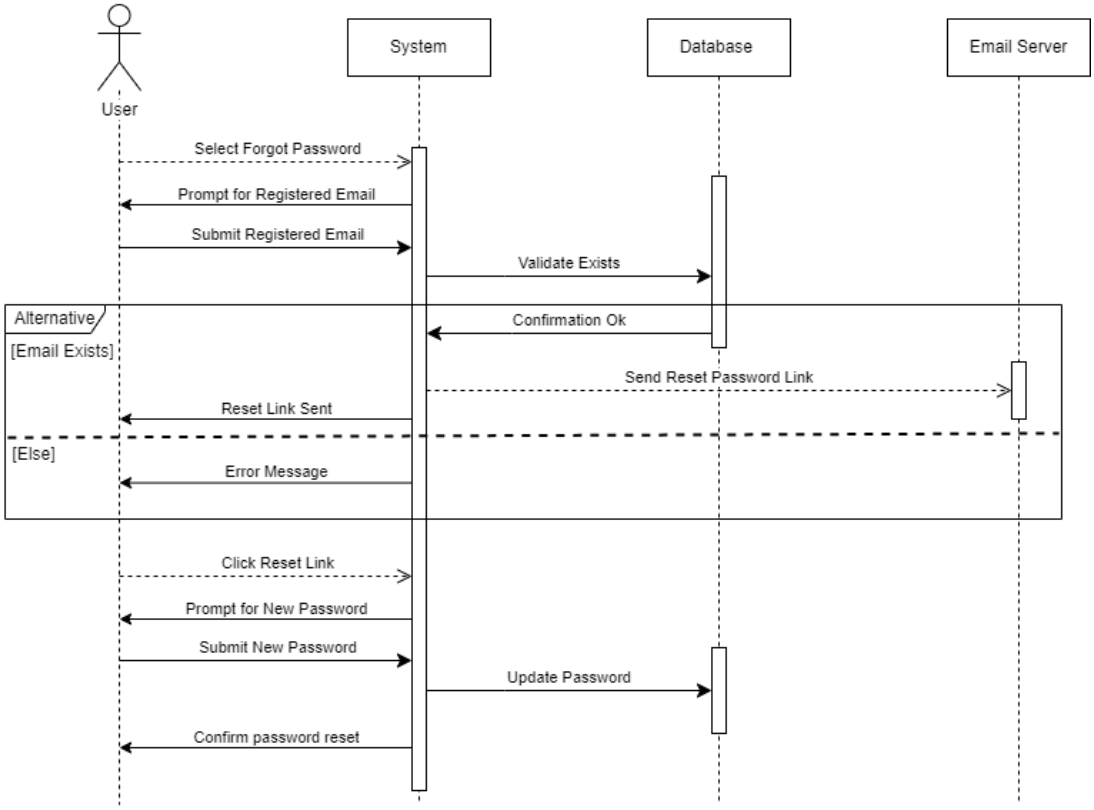
Registration



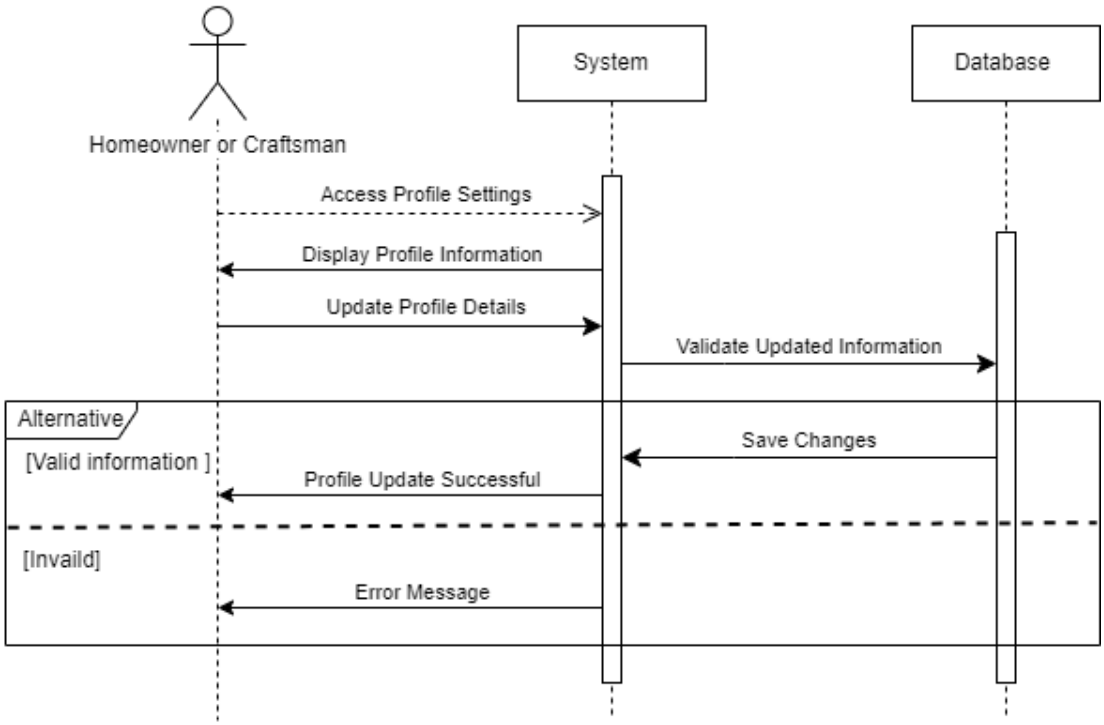
login



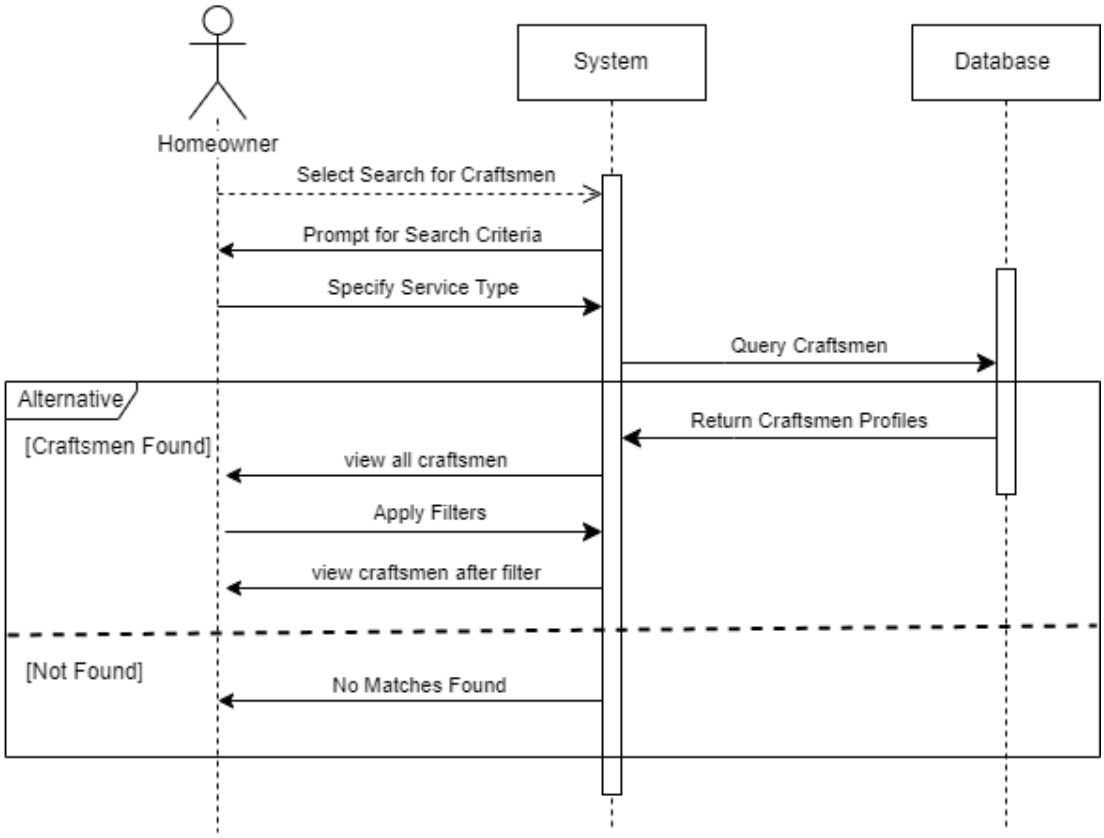
Reset Password



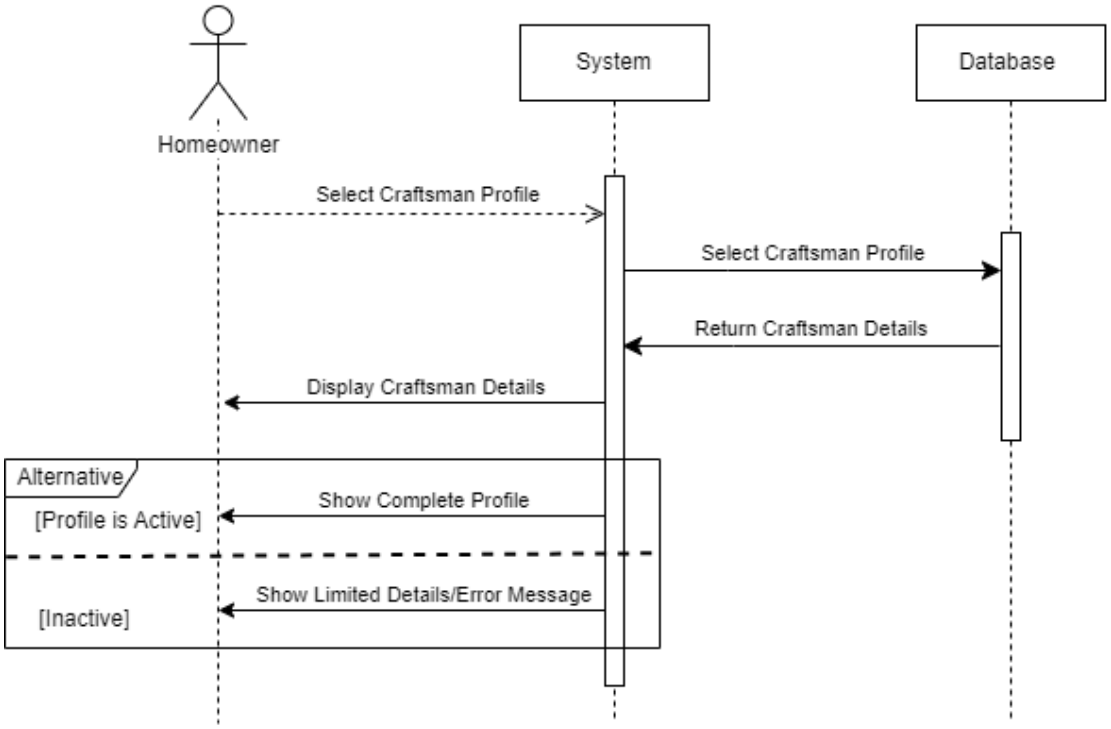
Profile Update



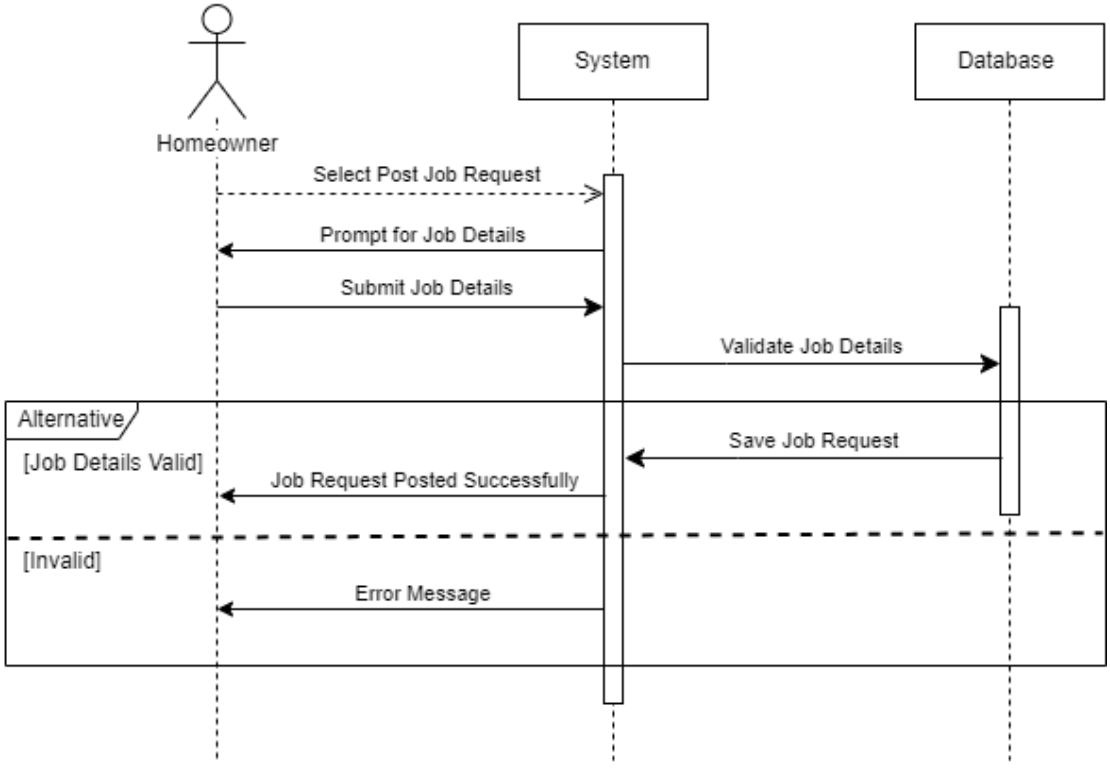
Search for Craftsmen



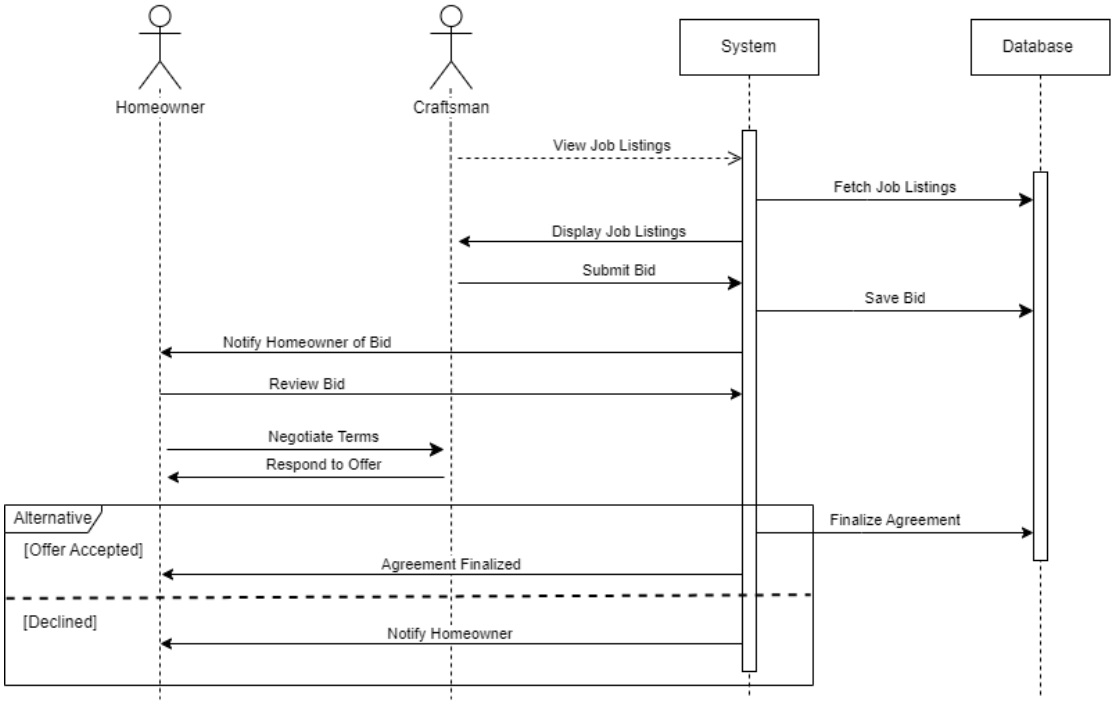
View Craftsman Profiles



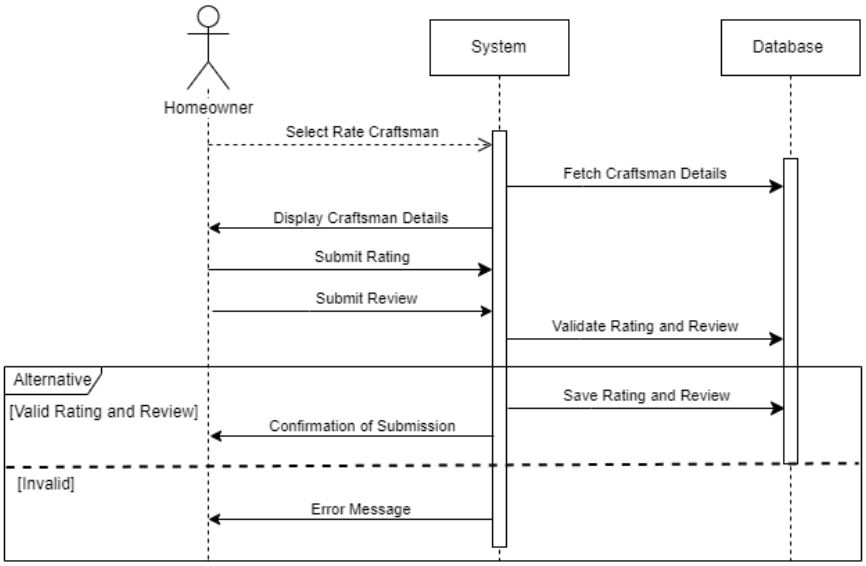
Post Job Request



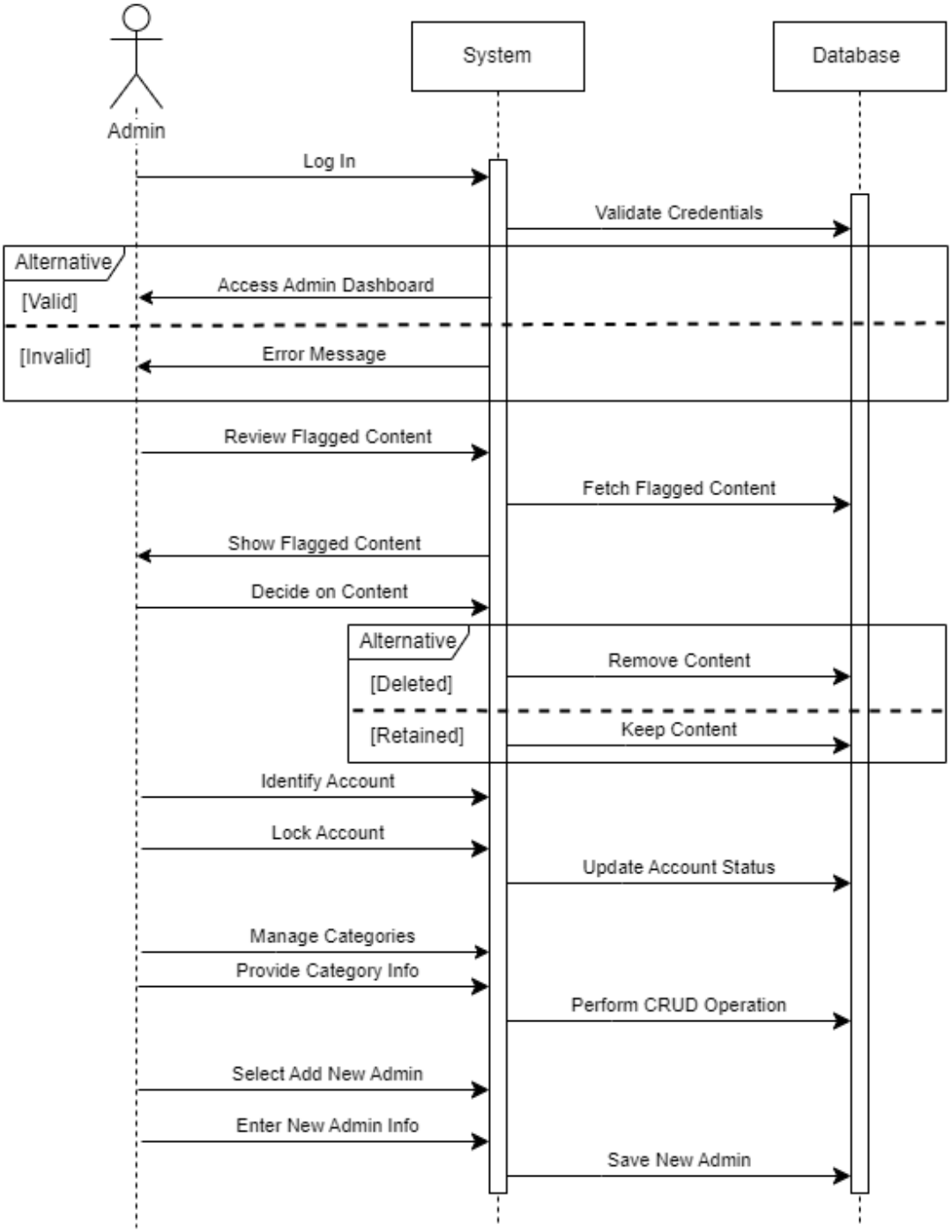
Submit Bids and Negotiate

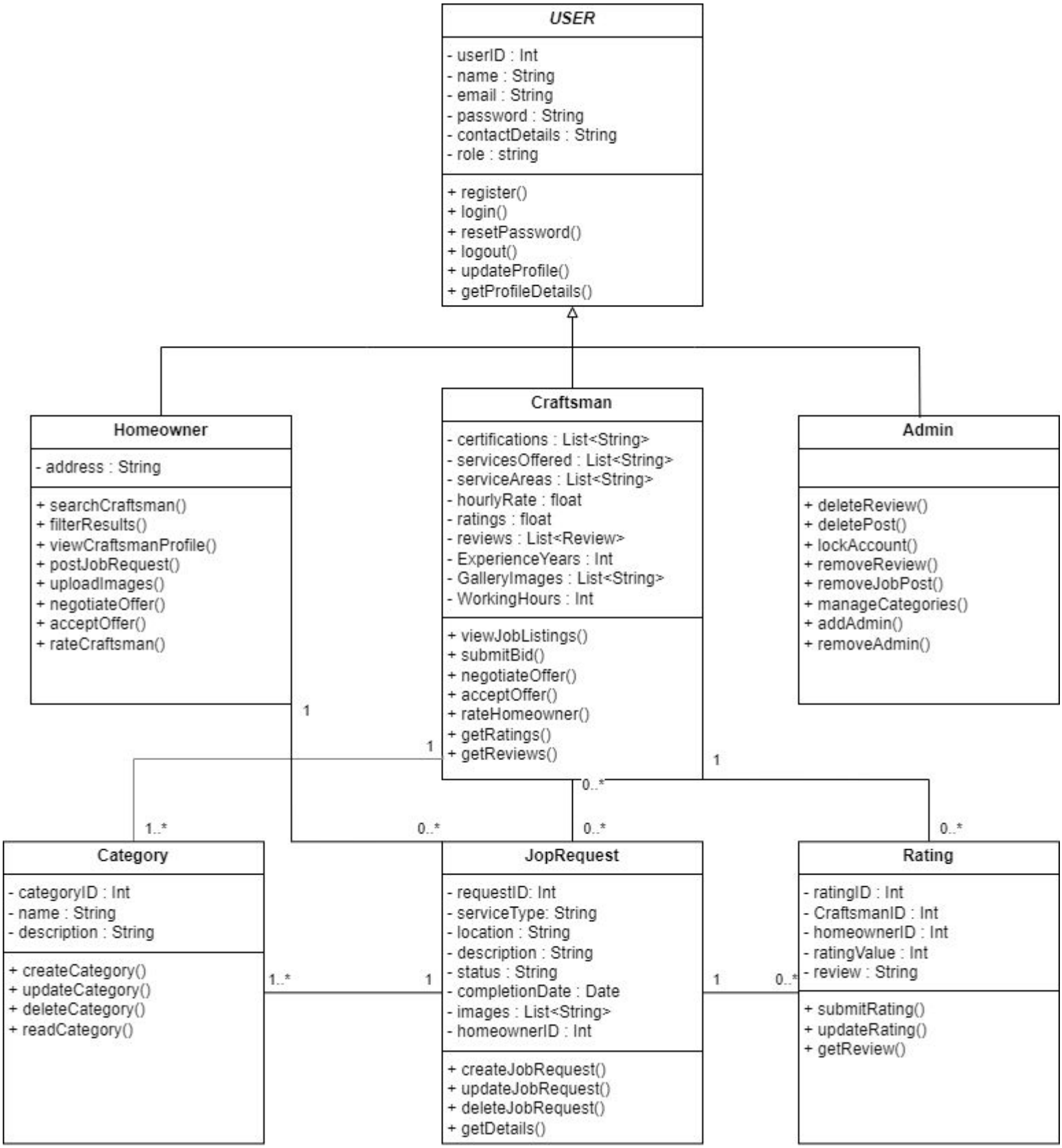


Rate and Review Craftsmen



Admin Management of Content and Users




USER
- userID : Int
- name : String
- email : String
- password : String
- contactDetails : String
- role : string
+ register()
+ login()
+ resetPassword()
+ logout()
+ updateProfile()
+ getProfileDetails()
Homeowner
- address : String
+ searchCraftsman()
+ filterResults()
+ viewCraftsmanProfile()
+ postJobRequest()
+ uploadImages()
+ negotiateOffer()
+ acceptOffer()
+ rateCraftsman()
Craftsman
- certifications : List<String>
- servicesOffered : List<String>
- serviceAreas : List<String>
- hourlyRate : float
- ratings : float
- reviews : List<Review>
- ExperienceYears : Int
- GalleryImages : List<String>
- WorkingHours : Int
+ viewJobListings()
+ submitBid()
+ negotiateOffer()
+ acceptOffer()
+ rateHomeowner()
+ getRatings()
+ getReviews()
Admin
+ deleteReview()
+ deletePost()
+ lockAccount()
+ removeReview()
+ removeJobPost()
+ manageCategories()
+ addAdmin()
+ removeAdmin()
1
1
1
0..*
1..*
0..*
0..*
0..*
Category
- categoryID : Int
- name : String
- description : String
+ createCategory()
+ updateCategory()
+ deleteCategory()
+ readCategory()
JopRequest
- requestID: Int
- serviceType: String
- location : String
- description : String
- status : String
- completionDate : Date
- images : List<String>
- homeownerID : Int
+ createJobRequest()
+ updateJobRequest()
+ deleteJobRequest()
+ getDetails()
Rating
- ratingID : Int
- CraftsmanID : Int
- homeownerID : Int
- ratingValue : Int
- review : String
+ submitRating()
+ updateRating()
+ getReview()
1..*
1
1
0..*