CSC 581 [PROJECT REPORT – 4]

I.   PROJECT TOPIC

Handyman Service Request Application

II.   GROUP DETAILS

Group Number: Group – 1

Group Members:
1. Ananya Kakumanu
2. Christian Quintero
3. Cole Allen
4. Lokesh Purohit
5. Malika Hafiza Pasha
6. Sajeel Mohammed Abdul
7. Shoaibuddin Mohammed

III.   PROJECT DESCRIPTION

A handyman can sign up for the app and advertise their skills. Depending on the service needed, a customer may decide to hire a handyman. A repair estimate can be sent to the customer when they provide images of the assignment. The client has the choice to take use of the service and make an appointment. The service, cost, and appointment for the repair will all be communicated to the client. Payment may be made by cash or card upon job completion.

IV.   ACTORS

A. PRIMARY ACTORS

A user or system component that interacts with the system primarily to accomplish a particular objective is considered the primary actor. In this project the primary actors are:

1. Handyman
   - Signs up to offer services
   - Provides details about the services offered
   - Receives service requests and provides repair quotations
   - Confirms and completes repair appointments

2. Customer
   - Signs up to request handyman services
   - Provides details about the task and upload photos
   - Receives repair quotations
   - Accepts the service and schedules an appointment
   - Makes payments for the services using a credit card

3. Admin
   - Manages and monitors the platform
   - Ensures smooth functioning of the app
   - Resolves disputes or issues if needed

4. External Database
   - Holds user information, service requests, quotes, messages, images, etc

5. Third Party Payment Handler
   - Processes payment/deposit requests and returns confirmation/denial and receipt information

B. SECONDARY ACTORS

A user or system that helps the primary actor to achieve their objectives or is impacted by the use case but is not a direct beneficiary is referred to as a secondary actor. For this project the secondary actors are:

- Building Manager/Inhabitant
  - May request service but not actually use app since they do not own the property
  - Will be involved in showing handyman where problem is
  - Will be involved in confirming handyman's work was acceptable
- Security Guard/Clerk/Desk Attendant
  - May be involved in directing handyman to proper apartment
  - May be involved in letting handyman into building or complex

V. FUNCTIONAL REQUIREMENTS

The specific functionality and features that the handyman service request app must have to satisfy the needs of its consumers are outlined in the functional requirements.
For this project the functional requirements are as follows:

A. **User Registration and Authentication**
   ● Customers and handymen who provide the required information can create accounts.
   ● Email verification for account activation should be supported by the app.
   ● Users ought to be able to securely log in with their credentials.

B. **Handyman Profile**
   ● The ability to create and administer an account should be available to handymen.
   ● The ability to list the services provided, the hourly prices, the service locations, and the availability.
   ● Upload your profile photo and any necessary credentials.

C. **Service Request Creation**
   ● Customers can create service requests, detailing the task and, if necessary, adding images.
   ● Indicate the appointment's desired time and date.
   ● Choose the required service type from a list that has been predetermined.

D. **Service Quotation**
   ● A handyman can browse work information and images and accept service requests.
   ● Based on the information given, provide a repair estimate.
   ● Provide the customer with the quotation for evaluation.

E. **Appointment Management**
   ● Customers can evaluate repair quotes they have received and select a handyman based on the quotes they have received.
   ● Verify the service and arrange a time to meet with the chosen handyman.
   ● Appointments should be confirmed and alerts sent to the handyman.

F. **Payment Processing**
   ● Customers should be able to safely use their credit card information to pay for the service.
   ● Manage payment confirmation and offer receipts for payments.

G. **Notification System**
   ● Establish a system of alerts to inform users of service requests, estimates, appointment confirmations, and reminders.

H. **Feedback and Ratings**
   ● Encourage clients to comment and rate the work that has been done.
   ● Show the average ratings given to each handyman based on client comments.

I. Search and Filters
- Make it possible for customers to look for handymen based on the services they offer, their location, and their ratings.
- Use filters to enhance search results.

J. Messaging System
- Establish a messaging system that enables handymen and clients to communicate about service specifics, appointment scheduling, etc.

K. Profile Management
- Users should have the opportunity to amend their profiles, change availability, and update contact information as well as service specifics.

L. Admin Dashboard
- Offer a dashboard for administrators to manage users, resolve conflicts, keep an eye on activity, and produce reports.

M. Appointments History
- Keep a record of all handymen and client appointments, including past and future appointments.

VI. USE CASES WITH FLOWS

1. Register User
   - Description: This use case describes the process of registering a user (*handyman* or *customer*) with the app. The details of the user must be saved to the *external database*, and the registered user must be signed in to access the app's services.

   - Basic Flow
     1. The user is prompted to sign up or sign in
     2. The user selects "sign up" and is prompted to enter an email address. (A1)
     3. The existence of the email address is checked by the external database to verify its validity
     4. On acceptance of the email address, the user inputs their personal information, such as name, sex, age, address, payment information, and password
     5. The user is prompted to enter their password again to confirm their new password
     6. If the confirmation password matches the initial password, the user's account is created successfully

7. User is then returned to the sign-in/sign-up menu
- Alternate Flow
    i. <A1 – If the user selects "sign in", they are prompted for their email and password. If the latter matches the information saved in the database, the user is granted access to their account. If the user is signing in for the first time, they are directed to their profile and prompted to (optionally) fill it out. (A2) When done with the app, they may sign out by clicking the relevant button. A dialog box then appears with a brief snippet of the appointments pending with their timing, with a question asking confirmation of the decision to sign out? On selecting "YES", the user is signed out, taking them back to the sign in/sign up menu.>
    ii. <A2 – The user selects different fields like profile picture, name, address, age, payment information. On selecting a specific field, the user is allowed to update the information in that field. After the changes are made, the user selects the "SAVE" button at the bottom. On clicking on "SAVE", a dialog box appears to seek confirmation of the desire in making the change in information. On clicking "YES", the new information is fed to the external database, to replace the previous information, where it is saved in that user profile. The user is taken back to the previous screen (main edit profile page) where all the profile fields are displayed that can be chosen to edit information.>

2. Request Service
    - Description: This use case describes the process of a *customer* requesting a service from a *handyman*. The *customer* must first select a particular *handyman*, then send a request to the latter; requests are saved to the *external database*.

    - Basic Flow:
        1. A customer navigates to their dashboard(A1) and clicks the "Create Service Request" button.
        2. The customer inputs a description of the requested service and selects the desired appointment date and time. (A2)
        3. The system displays all the available handymen with their ratings and other information
        4. The customer selects a preferred handyman and writes a service request with a text explanation, optionally accompanied by images.
        5. Upon the customer clicking "Send", the service request is saved to the external database and sent to the selected handyman. Both the handyman and the customer can view the service request at any time. (A3)

- Alternative Flow:
  - i. <A1 – The user browses through handymen and finds one with the qualifications that they desire. The customer then starts a service request from that handyman's profile.>
  - ii. <A2 – If the customer has preferences or does not see any handymen that they would like to choose, they can set filters for the list of displayed handymen to narrow down the results to best fit their criteria.>
  - iii. <A3 – The customer or handyman may cancel a request if they desire.>

3. Service Response
   - Description: This use case describes the process of the ***handyman*** viewing a service request from a ***customer*** and giving the ***customer*** a quote based on that service request, and the ***customer*** accepting the service quote and setting an appointment time.

   - Basic Flow
     1. The handyman receives a service request, proceeds to check the description of the service required, appointment date and time requested, and location distance, and sends a service quotation.(A1)
     2. On the customer's acceptance of the quotation(A2), the system asks for payment confirmation.
     3. The customer fills in the payment information to complete the transaction setup(A3), in which the system confirms the service appointment to both the customer and the handyman.(A4)

   - Alternative Flow:
     - i. <A1 – The handyman may instead opt to deny the service request.>
     - ii. <A2 – The customer may instead opt to deny the quotation or message the handyman to discuss pricing.>
     - iii. <A3 – If the customer plans to instead pay by cash or check, they must note this here to continue without putting in online payment information; the handyman then has the option to confirm or deny payment method acceptability, and thus confirm the appointment.>
     - iv. <A4 – The customer and the handyman both have the option to review appointment details at any time and cancel the appointment if unforeseen circumstances or changes in plans require the service to be postponed or canceled.>

4. Payment
   - Description: This use case describes the process of the ***customer*** paying the ***handyman*** for their services. When creating or editing their accounts, the

customer and the **handyman** both have the option to put in their payment information for current or future purchases or payments of services. After payment information is successfully updated, it gets updated in the **external database**. When completing a service request, the **payment handler** will handle the transaction using the saved payment information of both parties.

- ○ Basic Flow
    1. Once a job is completed to the customer's satisfaction, the customer and handyman are both given a prompt to confirm payment. (A1)
    2. When the customer and handyman have both confirmed payment, stored payment information is passed along to the external payment handler; payment is then charged to the customer, deposited to the handyman, and confirmed as completed. (A2)
    3. Once payment is confirmed as completed, the transaction is closed and logged in the database. Receipts are given to both users.
    4. Users are returned to their main menus.
- ○ Alternate Flow
    i. <A1 – If a user has not entered any payment information yet, they are given the option to either enter and save payment details to their account, or skip to exchanging cash payment instead.>
    ii. <A2 – If payment information is not correct, error messages are displayed to both users, and the offending user(s) is given the option to correct their financial information. If this is not possible, options are given to both users to exchange cash payment instead and confirm payment without using the third party payment system.>
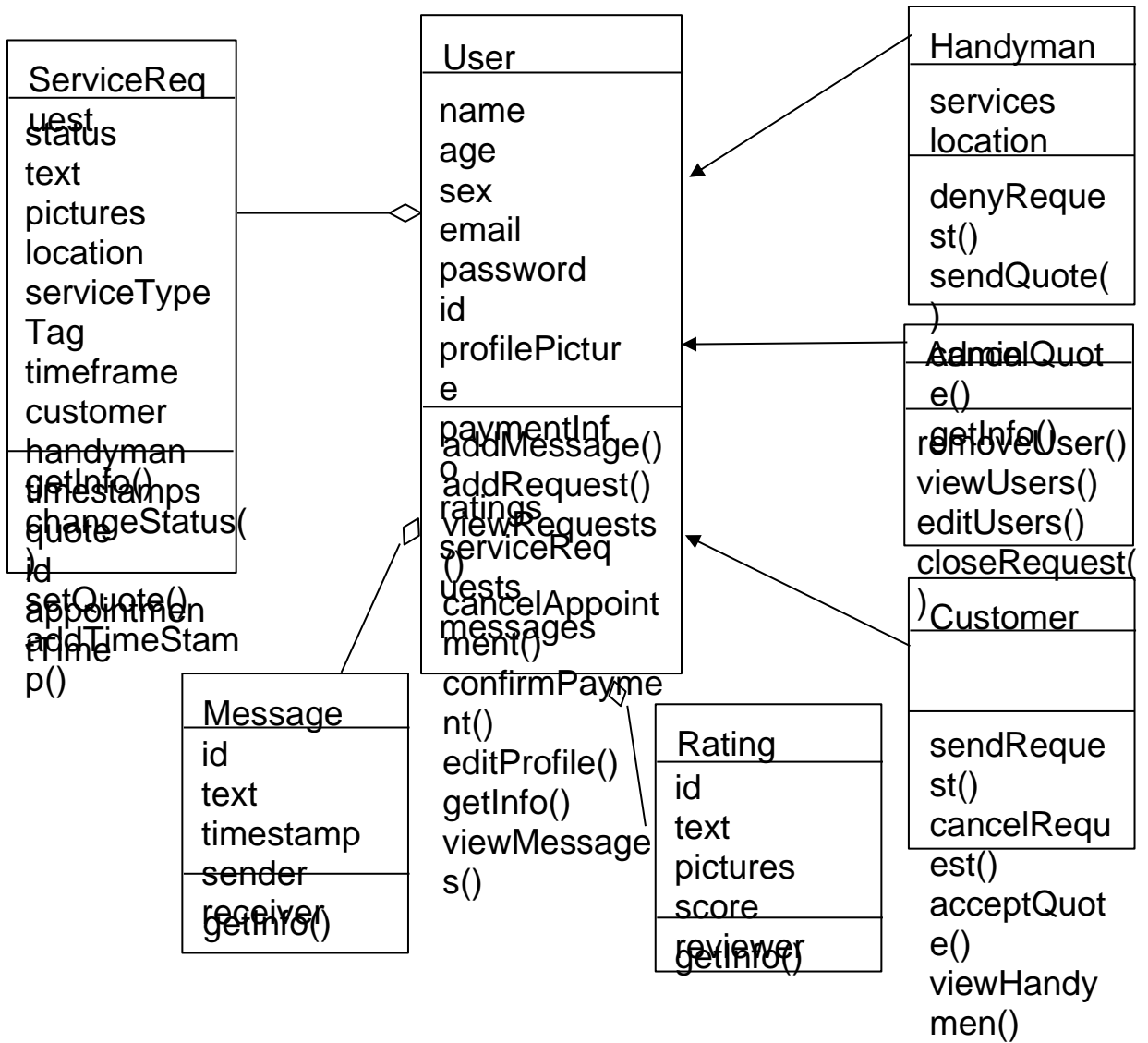
5. Message User
    - ○ Description: This use case describes the process of an **admin, customer** or **handyman** sending a message to another user. Messages can be used to communicate about service requests, appointments, or other topics. The **external database** also holds these messages.

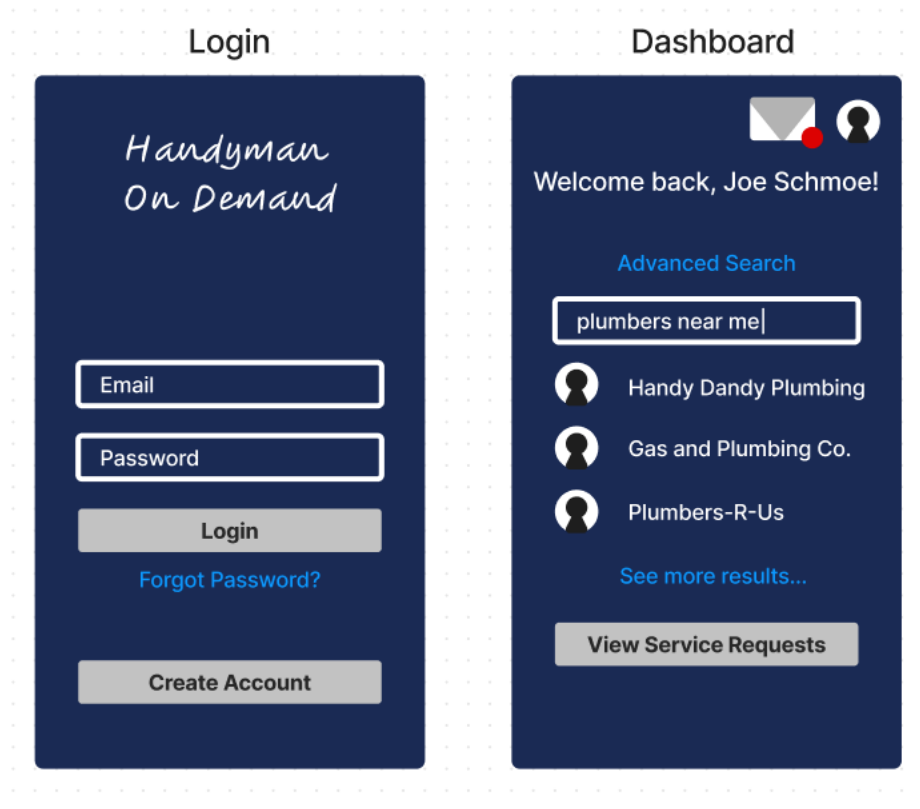   - ○ Basic Flow

        1. A user (must be signed in) navigates to their inbox.
        2. The logged in user selects the recipient they want to communicate with (could be from a list of previous conversations or by searching for a user).
        3. User composes the message in the provided text box or editor.
        4. User clicks 'Send' or equivalent button to send the message to the corresponding recipient.

     5. System saves and sends the message. (A1)

     6. System updates the external database with the new message.

     7. The receiving user gets a notification about the new message. (A2)

- Alternate Flow
  - i. <A1 – If the recipient has blocked the sender, the message will not be delivered. Instead, a notification will be sent to the sender telling them that the message cannot be delivered. Admins cannot be blocked.>
  - ii. <A2 – If the recipient has their notifications turned off, they will not get any notifications about new messages until they open the app.>

6. Manage User
   - Description: This use case describes the process of an **admin** viewing and making privileged actions with regards to **customers** and **handymen**.
   - Basic Flow

     1. An admin signs into the app via their own special portal using admin credentials.
     2. The admin is presented with a special menu (admin dashboard), where they can see admin notifications, messages, and a filterable list of users.
     3. Admin selects a particular user, either by searching or browsing.
     4. Admin selects the option "Remove User"
     5. Admin is prompted to confirm that they are sure they want to remove this user.
     6. Admin is prompted for their admin credentials.
     7. Admin is prompted for a reason as to remove the user; they have the option to upload images along with text.
     8. Admin is asked credentials a second time upon clicking "Confirm Removal".
     9. All pending appointments with the given user are canceled and notifications sent to relevant users. The user is removed from the system. An email is sent to both admin and user as a record of removal. Finally, the removal reason and any associated images are stored in the database.

VII.   <u>USE CASE DIAGRAM</u>

ServiceReq

ServiceReq uest

status
text
pictures
location
serviceType
Tag
timeframe
customer
handyman
timestamps
quote
id
appointmen
tTime

getInfo()
changeStatus(
)
setQuote()
addTimeStam
p()

User

name
age
sex
email
password
id
profilePictur
e

paymentInf
o
ratings
serviceReq
uests
messages

addMessage()
addRequest()
viewRequests
()
cancelAppoint
ment()
confirmPayme
nt()
editProfile()
getInfo()
viewMessage
s()

Handyman

services
location

denyReque
st()
sendQuote(
)
acceptQuot
e()

Admin

removeUser()
viewUsers()
editUsers()
closeRequest(
)

getInfo()

Customer

sendReque
st()
cancelRequ
est()
acceptQuot
e()
viewHandy
men()

Message

id
text
timestamp
sender
receiver

getInfo()

Rating

id
text
pictures
score
reviewer

getInfo()

VIII.   UI SKETCHES

Login

Handyman
On Demand

Email

Password

Login

Forgot Password?

Create Account

Dashboard

Welcome back, Joe Schmoe!

Advanced Search

plumbers near me

Handy Dandy Plumbing

Gas and Plumbing Co.

Plumbers-R-Us

See more results...

View Service Requests

Editing Profile

Update Profile Details!

Name: Ralph C|

Number: 212-133-1342

Address: 15 Main St, Newton, CA

ZIP Code: 91103

Edit Financial Info

Save Profile Changes

Creating Request

Select a Date: 11/30/2023

Select a Time Slot: 2pm-3pm

Bath Fitting   Add
*4.58(12.3k reviews)
Installation of mirrors and showers
Starts at $2k │ 13 hrs
View Details

Basin and Sink   Add
*4.79(87.1k reviews)
Pipe leakage and wash basin repairs
Starts at $180 │ 1 hr 30 mins
View Details

See more results...

Send Service Requests

Service Response

Total Service Request: 12

Bath Fitting
Description: 2 Broken mirrors and
replacement of 2 showers
View Details
Quotation Sent

Basin and Sink
Description: Waste pipe leakage and
wash basin installation
Appointment Date: 11/30/2023
Appointment Time: 2.05pm
Location: Long Beach
Distance: 30 mins away

Enter Quotation for Service Request: $2.2k

See more requests...

IX.   LIST OF CLASSES

- User: This class represents any user of the system, including customers, handymen, and admins.
- Handyman: This class represents a handyman who provides services to customers through the system.
- Admin: This class represents an administrator of the system, who has special privileges such as viewing and managing user accounts.
- Customer: This class represents a customer who requests services from handymen through the system.
- ServiceRequest: This class represents a request from a customer for a service from a handyman.
- Rating: This class represents ratings assigned to the user after every service request is completed.
- Message: This class represents a single message sent from one user to another.
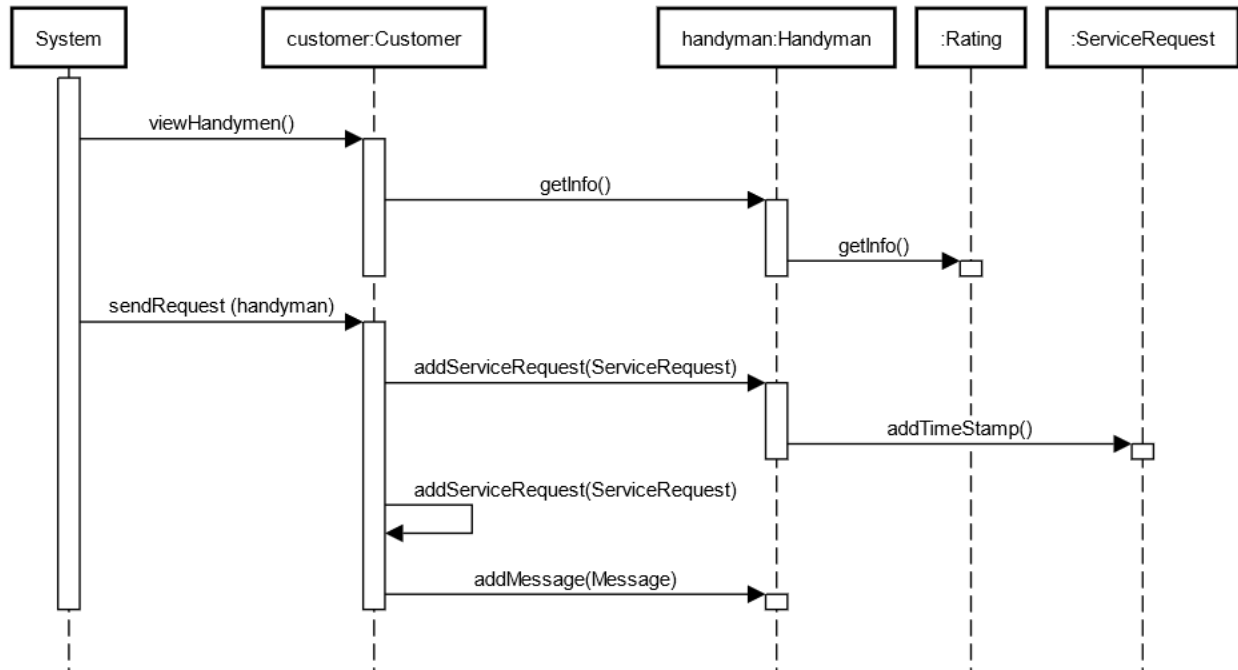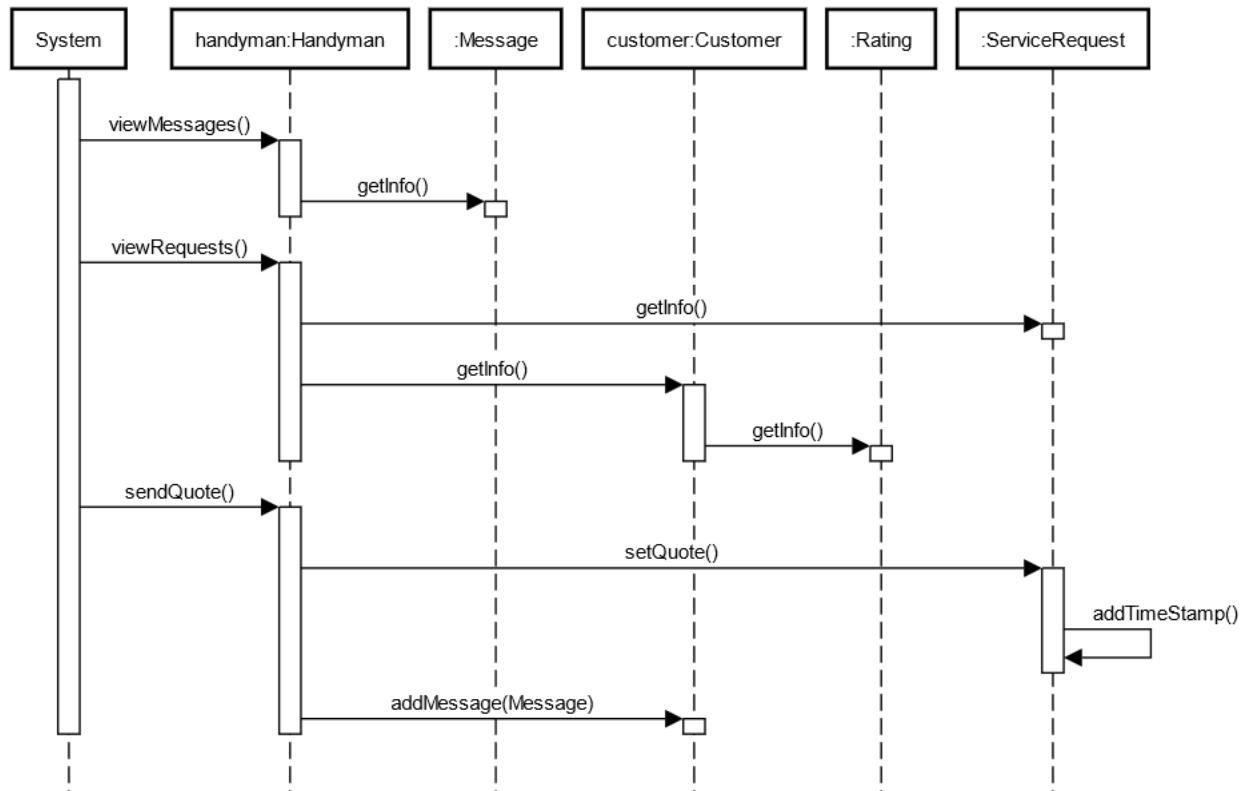
X.   **CLASS DIAGRAM**

Notes:
1. Admin and Customer intentionally have no unique attributes.
2. Database and payment methods are assumed to be handled by library methods and thus no classes are presented for them here.

XI. SEQUENCE DIAGRAMS

Send Service Request

Response to Service Request

Accept Quote

System	customer:Customer	:Message	handyman:Handyman	:ServiceRequest

viewMessages()

getInfo()

viewRequests()

getInfo()

acceptQuote(ServiceRequest)

addMessage(Message)

changeStatus()

addTimeStamp()