



Bilkent University

Department of Computer Engineering

Senior Design Project

Project name: DAOS (Dealership Assistance and Optimization System)

Analysis Report

Metehan Kesekler, Kerem Tuna Özlü, Mustafa Said Sarı, Sarp Tuğberk Topallar

Supervisor: Uğur Doğrusöz

Jury Members: M.Mustafa Özdal and Mehmet Koyutürk

Progress Report
November 6, 2017

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

1. Introduction	3
2. Current System	4
3. Proposed system	4
3.1 Overview	4
3.2 Functional Requirements	5
3.3 Nonfunctional Requirements	5
3.4 Pseudo requirements	6
3.5 System models	6
3.5.1 Scenarios	6
3.5.1.1 Scenario 1	6
3.5.1.2 Scenario 2	6
3.5.1.3 Scenario 3	7
3.5.1.4 Scenario 4	7
3.5.1.5 Scenario 5	8
3.5.1.6 Scenario 6	8
3.5.1.7 Scenario 7	9
3.5.1.8 Scenario 8	9
3.5.1.9 Scenario 9	9
3.5.1.10 Scenario 10	10
3.5.1.11 Scenario 11	10
3.5.1.12 Scenario 12	11
3.5.1.13 Scenario 13	11
3.5.1.14 Scenario 14	12
3.5.1.15 Scenario 15	12
3.5.1.16 Scenario 16	13
3.5.1.17 Scenario 17	13
3.5.1.18 Scenario 18	14
3.5.2 Use case model	15
3.5.2.1 Create a New Account Use Case	16
3.5.2.2 Login Use Case	17

3.5.2.3 View Homepage Use Case	18
3.5.2.4 Suggest Area for a New Branch Use Case	18
3.5.2.5 Suggest Area for Termination of a Branch Use Case	19
3.5.2.6 View Dealership Statistics Use Case	19
3.5.2.7 Add new Branch Use Case	20
3.5.2.8 Remove a Branch Use Case	21
3.5.2.9 View Dealership Map Use Case	22
3.5.2.10 View Contact Info of a Branch Owner Use Case	23
3.5.2.11 Forgot Password Use Case	23
3.5.3 Object and Class Model	25
3.5.3.1 Person	25
3.5.3.2 Company	26
3.5.3.3 Dealership	26
3.5.3.4 Region	26
3.5.3.5 AccountManager	26
3.5.3.6 DataManager	26
3.5.3.7 Assistance	27
3.5.3.8 Performance	27
3.5.4 Dynamic models	27
3.5.4.1 Sequence Diagrams	27
3.5.4.2 Activity Diagram	33
3.5.5 User Interface - Navigational Paths and Screen Mock-ups	34
4. Glossary	39
5. References	39

1. Introduction

When dealerships are given by companies to third parties, two of the most essential concerns are legality and profitability for such a dealership. Both of these issues can be addressed by selecting the appropriate location for the dealership. Finding such proper locations manually is very troublesome, because one must obtain information about the legal restrictions and regulations about the area. These restrictions and regulations may differ from province to province, even sometimes municipalities within the same province enforce different regulations. In addition to that, profitability potential is very hard to measure and may be affected by many factors such as weather, time of the day, seasonal activities of the locals, shopping demographic of the area etc. Even if these potential factors are successfully be identified initially, they may also change later on. Luckily, all these problems are dependent of the location of the dealership and using a geographic information system (GIS) can be helpful to solve all of them.

The issue of legality is very area sensitive and different constraints must be met for different types of businesses. For example, liquor and tobacco shops cannot be nearer than 100 meters to schools, dorms or sanctuaries by law. Therefore, having such sensitive places as points of interest (POI) in DAOS and display them on the actual map would certainly be very beneficial for companies trying to pick up the right venue for the regulatory assessment.

When the profit potential of a dealership is considered, the area can be manually inspected. But manual inspection can be made for a limited amount of time and therefore may not be enough to cover the overall trend for the area. Sending a team to different cities around the country for field surveys is also very costly. DAOS will present the data about spending potential of the area such as number of residents in the area and the foot traffic of the streets. Data provided by DAOS can be interpreted to become aware of the trends about the area even it could be used for making suggestions about possible business opportunities since the information about the area is accumulated in the database.

After the dealership becomes operational, its performance must be continuously observed. Because, the habits of the customers are not static and businesses are very sensitive to competition. So, a very profitable location for a dealership may not be still profitable so monitoring the operations of that dealership can be very important for making a decision about allocating more resources there or closing it entirely. Monitoring the sales of different branches only provides information about quantity, but considering branch sales according to their local shopper profile will give their actual performance metric for evaluation.

Our solution, DAOS with its enhanced user interface, interactive and color coded maps and broad capabilities about different types of businesses, provides the answers for the following questions “Can this dealership be opened here?”, “Should this dealership be opened here?” and “Is this dealership still profitable?”

2. Current System

There is not any current system for dealership assistance and performance review. Companies may use similar systems designed by their IT department but any of them is public.

3. Proposed system

Details about the proposed system such as descriptions, requirements and models are presented in this section.

3.1 Overview

Purpose of the project is to develop a special tool based on geographic information system and sales database to provide recommendations to companies for selecting the best dealership locations and optimizing the sales performance and profits.

Selecting the right location for a new dealership which will comply with regulatory constraints and at the same time generate and sustain maximum profitability is a costly and long process by using only conventional tools and field visits. When a company wants to expand their branches their expansion policy is very important, for the companies that are careful and selective about their decisions they will spend time and money to find an optimal location for adding it to their network. Instead of spending money and time for different professionals to handle legal, regional or monitoring requirements iteratively we combine this operations in one very easily understandable UI to provide the chance of handling them all in a single place and on a single time whether for expansion of their brand or auditing their profitability of their continuing service. This process allow us to provide the most carefully calculated information to our customers even recommending new opportunities to them.

3.2 Functional Requirements

- Name, owner's information, open addresses, sales information and geo-codes of dealerships will be held in dealership database.
- Foot traffic, prize of dealerships per meter square, statistical information and POIs(Point of Interests) will be held in region database.
- By using our sophisticated algorithm, performance information for existing dealerships and optimal location for potential dealerships will be created.
- In accordance of performance information, colors will be assigned to cities and counties in order to sort out the good from the bad.
- Performance information in a good user interface will be provided to user.
- An optimal area will be suggested to the investor if a new dealership is being planned to be opened.
- An exact dealership will be suggested to the investor if an existing dealerships is being planned to be closed or contract to be cancelled.
- Ease of use of the colored map will provide a good user experience.

3.3 Nonfunctional Requirements

- User interface and ease of use is important in order to prevent confusion of the users because DAOS is willing to reach any kind of user from dealership owner to highly educated executives of investor company.
- DAOS will make critical suggestions to investors, which means that DAOS should be a reliable assistance.
- DAOS will be a dynamic platform that should be in a continual change in order to reach the most updated data.
- DAOS should be precise and available any time investors need suggestions or want to follow their existing investments.
- Investors and dealerships data should be secured.

3.4 Pseudo requirements

- Collaboration tool for this project will be GitHub and our informative webpage will be <https://daosproject.github.io>
- Our program will be a desktop utility that prioritizes client server architecture.
- MySQL - MariaDB database will be used to store user and geographical data.
- Server side will be implemented C++ and Python.
- Desktop client will be implemented on C++.
- Besides from user provided data our system will use information provided by Google Maps, Yandex, Solvoyo, PlusOneMinusOne, Neozeka, Capgemini, etc. via their interfaces.

3.5 System models

3.5.1 Scenarios

3.5.1.1 Scenario 1

Use Case Name: CreateAccount for Eric

Actors: Eric

Entry Conditions:

- Eric starts the program and navigates to 'Create Account'

Exit Condition:

- Sign-up successful, redirected to homepage.

Events:

1. Eric enters his username
2. Eric enters his password
3. Eric enters his contact information (phone, email)
4. Eric fills the information fields about the specifics of the business
5. Eric clicks on 'Create Account'
6. Account is created and credentials are saved to our database

3.5.1.2 Scenario 2

Use Case Name: CreateAccount for Eric

Actors: Eric

Entry Conditions:

- Eric starts the program and navigates to 'Create Account'

Exit Condition:

- Sign-up declined, error message displayed.

Events:

1. Eric enters his username
2. Eric enters his password
3. Eric enters his contact information (phone, email)
4. Eric fills the information fields about the specifics of the business
5. Eric clicks on 'Create Account'
6. Error message displayed. Account is not created.

3.5.1.3 Scenario 3

Use Case Name: Login for Eric

Actors: Eric

Entry Conditions:

- Eric is on login screen

Exit Condition:

- Log-in successful, redirected to homepage.

Events:

1. Eric enters his Username
2. Eric enters his Password
3. Eric Clicks on 'Login' button
4. Credentials are matched with database
5. Eric is granted access to homepage

3.5.1.4 Scenario 4

Use Case Name: Login for Eric

Actors: Eric

Entry Conditions:

- Eric is on login screen

Exit Condition:

- Log-in declined, error message displayed.

Events:

1. User enters Username
2. User enters Password
3. User Clicks 'Login' button
4. Credentials are not matched with database
5. Login denied and error message displayed

3.5.1.5 Scenario 5

Use Case Name: ViewHomepage for Eric

Actors: Eric

Entry Conditions:

- Eric Successfully logged in
- Eric Successfully created an account

Exit Condition:

- Eric navigates to another screen

Events:

1. Eric is greeted via the homepage after a successful login
2. Color Coded map and navigation buttons are displayed to Eric

3.5.1.6 Scenario 6

Use Case Name: SuggestOpen for Eric

Actors: Eric

Entry Conditions:

- Eric Clicks 'New Dealership' button

Exit Condition:

- Eric clicks 'Create New Dealership'

Events:

1. Eric clicks on 'create new dealership'
2. Eric selects a specific area
3. Screen including suggestions from the system is displayed to Eric
4. Eric examines the provided suggestions
5. Eric selects a suggestion
6. Detailed information about selected suggestion is displayed to Eric

3.5.1.7 Scenario 7

Use Case Name: SuggestOpen for Eric

Actors: Eric

Entry Conditions:

- Eric Clicks 'New Dealership' button

Exit Condition:

- Eric cancels

Events:

1. Eric clicks on 'create new dealership'
2. Eric cancels and returns back to homepage.

3.5.1.8 Scenario 8

Use Case Name: SuggestClose for Eric

Actors: Eric

Entry Conditions:

- Eric clicks on 'View Dealerships'

Exit Condition:

- Eric clicks 'Terminate Dealership'

Events:

1. Eric navigates to list of the dealerships via clicking 'View Dealerships'
2. Existing dealerships are displayed to Eric
3. Worst performing dealership is listed at the top
4. Eric clicks onto the dealership to see the details
5. Eric clicks 'Terminate Dealership'
6. Dealership is terminated
7. Termination data is sent to the database
8. Database is updated

3.5.1.9 Scenario 9

Use Case Name: SuggestClose for Eric

Actors: Eric

Entry Conditions:

- Eric clicks on 'View Dealerships'

Exit Condition:

- Eric cancels

Events:

1. User navigates to list of the dealerships via clicking 'View Dealerships'
2. Eric cancels and returns back to homepage.

3.5.1.10 Scenario 10

Use Case Name: ViewDealershipStatus for Eric

Actors: Eric

Entry Conditions:

- Eric clicked on a dealership within the 'view dealerships' menu

Exit Condition:

- Eric pressed Back button

Events:

1. Eric clicked on a dealership
2. Dealership statistics are retrieved from the database
3. Pulled data is presented to Eric
4. Rank of the dealership is also displayed
5. Map with a marker on the dealership location and coverage radius is also displayed to Eric
6. Eric pressed 'Back' button

3.5.1.11 Scenario 11

Use Case Name: ViewDealershipStatus for Eric

Actors: Eric

Entry Conditions:

- Eric clicked on a dealership within the 'view dealerships' menu

Exit Condition:

- Eric navigated to another screen

Events:

1. Eric clicked on a dealership.
2. Dealership statistics are retrieved from the database
3. Pulled data is presented to Eric
4. Rank of the dealership also displayed
5. Eric navigates another screen

3.5.1.12 Scenario 12

Use Case Name: AddDealership for Eric

Actors: Eric

Entry Conditions:

- Eric clicked on 'New Dealership'

Exit Condition:

- Eric clicked 'Add Dealership' after providing the dealership info

Events:

1. Eric clicked 'New Dealership' button
2. Eric filtered an area
3. Eric is presented with the map which shows the POI's and existing dealerships with their coverage radii
4. Eric selects a location on the map
5. Location is checked for legality
6. Location performance estimations are presented to Eric
7. Eric selects the location
8. Eric enters the information about the dealership
9. Eric clicks 'Add Dealership' button
10. Information is sent to the database
11. Database is updated

3.5.1.13 Scenario 13

Use Case Name: AddDealership for Eric

Actors: Eric

Entry Conditions:

- Eric clicked on 'New Dealership'

Exit Condition:

- Eric clicked 'Add Dealership' after suggested dealership is clicked

Events:

1. Eric clicked 'New Dealership' button
2. Eric clicks one of the recommended area suggestions
3. Eric filtered a specific region to list all the dealerships
4. Eric is presented with the map which shows the POI's and existing dealerships with their coverage radii

5. Eric selects a location on the map
6. Location is checked for legality
7. Location performance estimations are presented to Eric
8. Eric selects the location
9. Eric enters the information about the dealership
10. Eric clicks 'Add Dealership' button
11. Information is sent to the database
12. Database is updated

3.5.1.14 Scenario 14

Use Case Name: RemoveDealership for Eric

Actors: Eric

Entry Conditions:

- Eric selects a branch from the list to see the details

Exit Condition:

- Eric clicks 'Terminate Branch' after selecting a dealership

Events:

1. Eric selects 'See Branches'
2. Eric filters an area
3. All the dealerships on the area are listed
4. Eric selects one of them
5. Detailed information and the map view is displayed to Eric
6. Eric clicks 'Terminate Branch'
7. Confirmation dialog is displayed
8. Database updated

3.5.1.15 Scenario 15

Use Case Name: ViewDealershipMap for Eric

Actors: Eric

Entry Conditions: Eric selected a dealership from the list

Exit Condition:

- Eric navigates to another menu

Events:

1. Eric navigate to 'See Branches'
2. Eric narrows down the area
3. A list of dealerships residing in the area is provided to Eric

4. Eric selects one of the dealerships
5. Eric sees the dealership in detail and the small form of the map view
6. Eric clicks on the map view
7. Map enlarges
8. Map displays the relevant POI's and the dealerships to Eric
9. Eric inspects and interacts with the map
10. Eric presses Esc or Back to navigate back

3.5.1.16 Scenario 16

Use Case Name: ViewOwner for Eric

Actors: Eric

Entry Conditions:

- Eric clicks on 'see owner info'

Exit Condition:

- Eric Pressed Esc or Back button

Events:

1. Eric navigates to list of the dealerships via clicking 'View Dealerships'
2. Existing dealerships are displayed to Eric
3. Eric clicks onto a dealership to see the details
4. Eric clicks 'see owner info'
5. Eric is presented with the information of the owner or the responsible for that particular dealership

3.5.1.17 Scenario 17

Use Case Name: ForgotPassword for Eric

Actors: Eric

Entry Conditions:

- Eric clicked on 'forgot password'

Exit Condition:

- New password is set

Events:

1. Eric clicks on the 'forgot password'
2. Email address of Eric is requested
3. Eric enters the email address

4. Email address is confirmed with the one stored in the database while registering first time
5. If the address is correct a password reset email is sent to Eric
6. Eric sets a new password
7. Database updated

3.5.1.18 Scenario 18

Use Case Name: ForgotPassword for Eric

Actors: Eric

Entry Conditions:

- Eric clicked on 'forgot password'

Exit Condition:

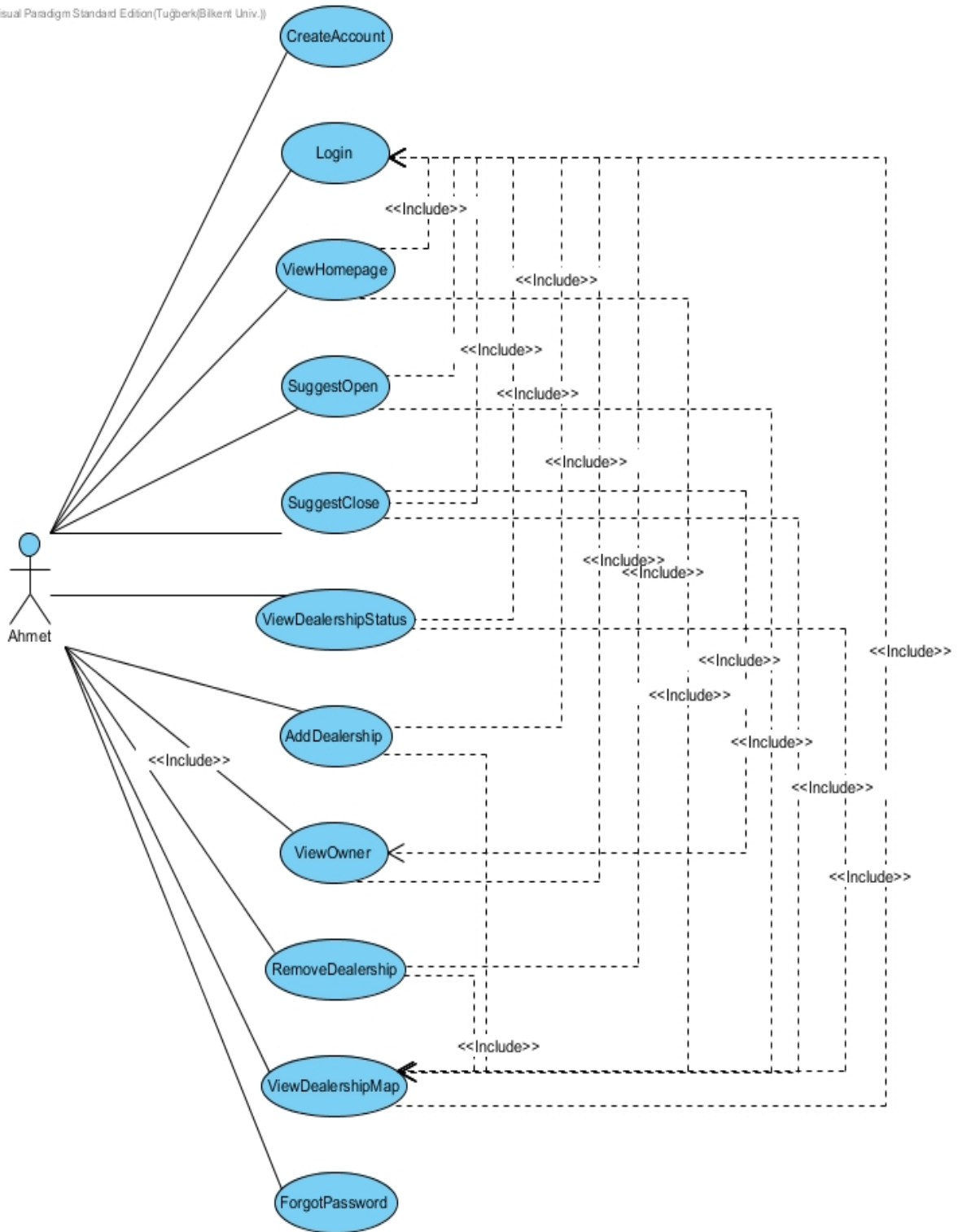
- New password setting is failed

Events:

1. User clicks on the 'forgot password'
2. Email address of the user is requested
3. User enters the email address
4. Email address is not confirmed
5. New password setting is failed, an error message is displayed to Eric.
6. Eric is asked to enter another email address.

3.5.2 Use Case Model

Visual Paradigm Standard Edition(Tugberk/Bilkent Univ.)



Create a New Account: User can create a new account which will be affiliated with a company on the database.

Login: A registered user can sign-in with correct credentials.

View Homepage: The homepage and the majority of the navigation buttons are displayed here.

Suggest Area for a New Branch: A new branch location to be pinned on the map with the possible coverage radius is presented to the user.

Suggest Area for Termination of a Branch: The worst performing branch is displayed to the user according to the areal information.

View Dealership Statistics: The sales and regional statistics about a specific dealership or all the dealerships in a region.

Add new Branch: A new branch can be added by entering the information about the branch manually or using the suggestion function of the system.

Remove a Branch: When a branch is closed this is the use case for updating the database or when the company user oversees the active branches the worst performing dealership is notified as a termination suggestion.

View Dealership Map: A map with POI's and branches with their coverage radii are displayed.

View Contact Info of a Branch Owner: Information about the owner or the responsible for that specific branch is displayed to company user.

Forgot Password: A company user can restore account via register email in case of a forgotten password.

3.5.2.1 Create a New Account Use Case

Use Case Name: CreateAccount

Actors

- Person

Entry Conditions

- User starts the program and navigates to 'Create Account'

Exit Conditions

- Sign-up successful, redirected to homepage.
- Sign-up declined, error message displayed.

Main Flow of Events

7. User provides a username
8. User provides a password
9. User enters the contact information (phone, email)
10. User fills the information fields about the specifics of the business
11. Click 'Create Account'
12. Account is created and credentials are saved to our database

Alternative Flow of Events

- A. One of the fields is missing
 - 5a. DAOS highlights the missing parts to user
 - 5b. User completes the missing areas
 - 5c. User Clicks 'Create Account'
 - 5d. Account is created and credentials are saved to our database

3.5.2.2 Login Use Case

Use Case Name: Login

Actors:

- Person

Entry Conditions:

- User is on login screen

Exit Conditions:

- Log-in successful, redirected to homepage.
- Log-in declined, error message displayed.

Main Flow of Events:

6. User enters Username
7. User enters Password
8. User Clicks 'Login' button
9. Credentials are matched with database
10. User is granted access to homepage

Alternative Flow of Events:

- A. Credentials are wrong
 - 4a. DAOS asks again for the correct credentials

- 4b. User provides the correct authentication information
- 4c. User Clicks 'Login'
- 4d. Credentials are matched with database
- 4e. User is granted access to homepage

3.5.2.3 View Homepage Use Case

Use Case Name: ViewHomepage

Actors:

- Person

Entry Conditions:

- Successful Login use case
- Successful CreateAccount use case

Exit Conditions:

- User navigates to another screen

Main Flow of Events:

3. User is greeted via the homepage after a successful login
4. Color Coded map and navigation buttons are displayed to user

Alternative Flow of Events:

- A. User clicks 'Homepage' button from another screen
 - 1a. User is greeted via the homepage after a successful login
 - 1b. Color Coded map and navigation buttons are displayed to user

3.5.2.4 Suggest Area for a New Branch Use Case

Use Case Name: SuggestOpen

Actors:

- Person

Entry Conditions:

- User Clicks 'New Dealership' button

Exit Conditions:

- User clicks 'Create New Dealership'
- User cancels

Main Flow of Events:

7. User clicks on 'create new dealership'
8. User selects a specific area
9. Screen including suggestions from the system is displayed
10. User examines the provided suggestions

11. User selects a suggestion
12. Detailed information about selected suggestion is displayed

Alternative Flow of Events:

- A. User skips 2.
 - 2a. User is greeted via the homepage after a successful login
 - 2b. Color Coded map and navigation buttons are displayed to user

3.5.2.5 Suggest Area for Termination of a Branch Use Case

Use Case Name: SuggestClose

Actors:

- Person

Entry Conditions:

- User clicks on 'View Dealerships'

Exit Conditions:

- User clicks 'Terminate Dealership'
- User cancels

Main Flow of Events:

9. User navigates to list of the dealerships via clicking 'View Dealerships'
10. Existing dealerships are displayed to the user
11. Worst performing dealership is listed at the top
12. User clicks onto the dealership to see the details
13. User clicks 'Terminate Dealership'
14. Dealership is terminated
15. Termination data is sent to the database
16. Database is updated

Alternative Flow of Events:

- A. User clicks 'see owner info'
 - 4a. Contact information about the dealership owner is pulled from the database
 - 4b. Contact information is presented to the user

3.5.2.6 View Dealership Statistics Use Case

Use Case Name: ViewDealershipStatus

Actors:

- Person

Entry Conditions:

- Clicked on a dealership within the 'view dealerships' menu

Exit Conditions:

- Back button is pressed
- Navigated to another screen

Main Flow of Events:

7. User clicked on a dealership
8. Dealership statistics are retrieved from the database
9. Pulled data is presented to the user
10. Rank of the dealership is also displayed
11. Map with a marker on the dealership location and coverage radius is also displayed

Alternative Flow of Events:

- A. User clicks 'see owner info'
 - 1a. User filtered a specific region to list all the dealerships
 - 1b. User clicked on a dealership
 - 1c. Dealership statistics are retrieved from the database
 - 1d. Pulled data is presented to the user
 - 1e. Rank of the dealership is also displayed
 - 1f. Map with a marker on the dealership location and coverage radius is also displayed

3.5.2.7 Add new Branch Use Case

Use Case Name: AddDealership

Actors:

- Person

Entry Conditions:

- 'New Dealership' clicked

Exit Conditions:

- 'Add Dealership' clicked after providing the dealership info
- 'Add Dealership' clicked after suggested dealership is clicked

Main Flow of Events:

12. User clicked 'New Dealership' button
13. User filtered an area
14. User is presented with the map which shows the POI's and existing dealerships with their coverage radii
15. User selects a location on the map
16. Location is checked for legality

17. Location performance estimations are presented to the user
18. User selects the location
19. User enters the information about the dealership
20. User clicks 'Add Dealership' button
21. Information is sent to the database
22. Database is updated

Alternative Flow of Events:

- A. User clicks one of the recommended area suggestions
 - 2a. User filtered a specific region to list all the dealerships
 - 2b. User is presented with the map which shows the POI's and existing dealerships with their coverage radii
 - 2c. User selects a location on the map
 - 2d. Location is checked for legality
 - 2e. Location performance estimations are presented to the user
 - 2f. User selects the location
 - 2g. User enters the information about the dealership
 - 2h. User clicks 'Add Dealership' button
 - 2i. Information is sent to the database
 - 2j. Database is updated

3.5.2.8 Remove a Branch Use Case

Use Case Name: RemoveDealership

Actors:

- Person

Entry Conditions:

- User selects a branch from the list to see the details

Exit Conditions:

- User clicks 'Terminate Branch' after selecting a dealership

Main Flow of Events:

9. User selects 'See Branches'
10. User filters an area
11. All the dealerships on the area are listed
12. User selects one of them
13. Detailed information and the map view is displayed to the user
14. User clicks 'Terminate Branch'
15. Confirmation dialog is displayed
16. Database updated

Alternative Flow of Events:

- A. User clicks 'See Branches'
 - 2a. User selects the worst performing dealership which is listed at the top
 - 2b. Detailed information and the map view is displayed to the user
 - 2c. User clicks 'Terminate Branch'
 - 2d. Confirmation dialog is displayed
 - 2e. Database updated

3.5.2.9 View Dealership Map Use Case

Use Case Name: ViewDealershipMap

Actors:

- Person

Entry Conditions:

- A dealership is selected from the list

Exit Conditions:

- Navigate to another menu

Main Flow of Events:

11. User navigate to 'See Branches'
12. User narrows down the area
13. A list of dealerships residing in the area is provided to the user
14. User selects one of the dealerships
15. User sees the dealership in detail and the small form of the map view
16. User clicks on the map view
17. Map enlarges
18. Map displays the relevant POI's and the dealerships to user
19. User inspects and interacts with the map
20. User presses esc or back to navigate back

Alternative Flow of Events:

- A. User navigates 'new dealership'
 - 1a. User is presented with a list of suggestions
 - 1b. User picks one of the suggestions
 - 1c. Selected area is displayed on the map with details like POI's, suggested location's foot traffic coverage
 - 1d. User presses esc or back to navigate back

3.5.2.10 View Contact Info of a Branch Owner Use Case

Use Case Name: ViewOwner

Actors:

- Person

Entry Conditions:

- Clicked 'see owner info'

Exit Conditions:

- Pressed esc or back

Main Flow of Events:

6. User navigates to list of the dealerships via clicking 'View Dealerships'
7. Existing dealerships are displayed to the user
8. User clicks onto a dealership to see the details
9. User clicks 'see owner info'
10. User is presented with the information of the owner or the responsible for that particular dealership

3.5.2.11 Forgot Password Use Case

Use Case Name: ForgotPassword

Actors:

- Person

Entry Conditions:

- 'forgot password' is clicked

Exit Conditions:

- New password is set
- New password setting is failed

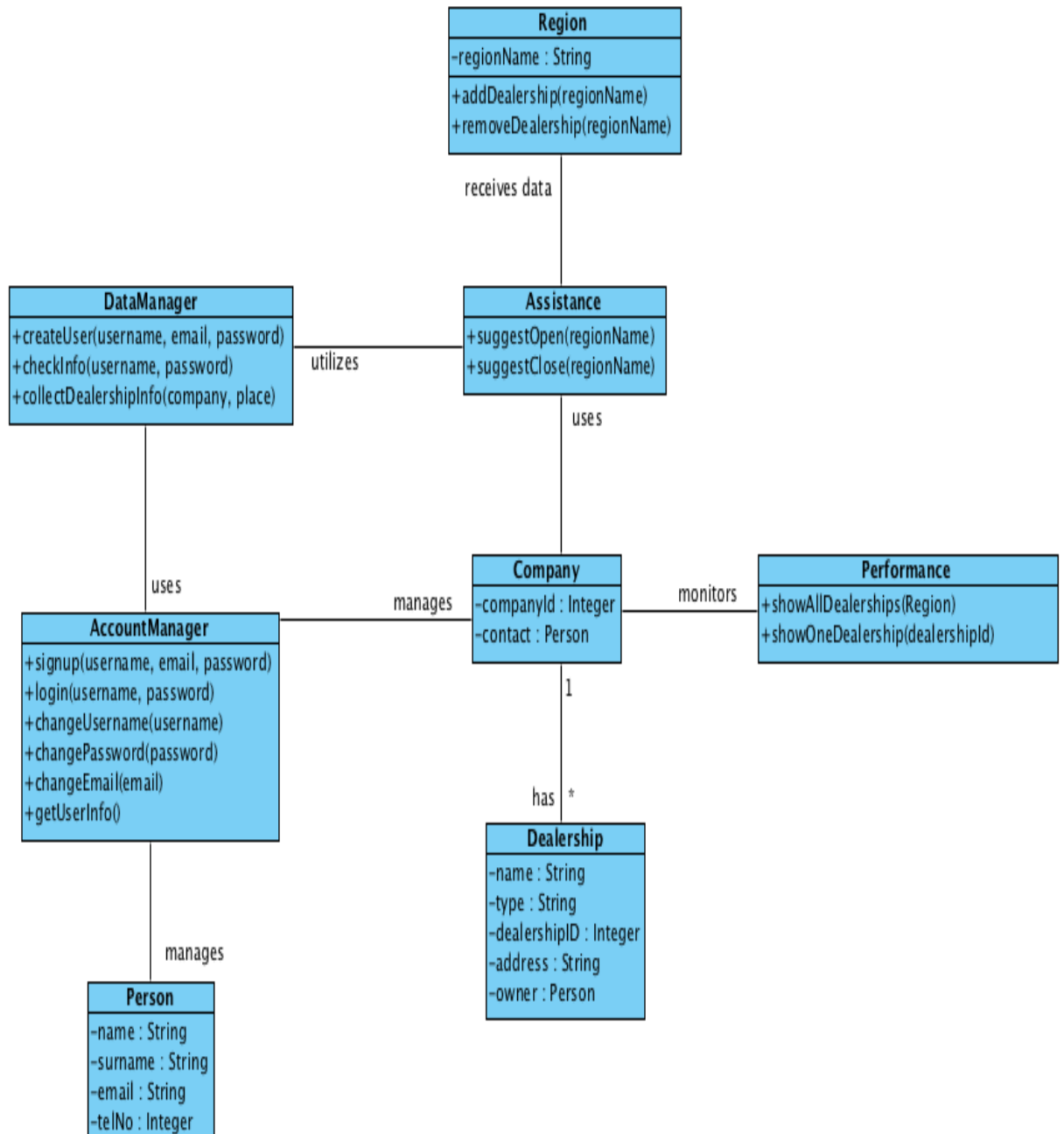
Main Flow of Events:

8. User clicks on the 'forgot password'
9. Email address of the user is requested
10. User enters the email address
11. Email address is confirmed with the one stored in the database while registering first time
12. If the address is correct a password reset email is sent to user
13. User sets a new password
14. Database updated

Alternative Flow of Events:

- A. Credentials are wrong
 - 4a. User provides a wrong email address
 - 4b. Email address is checked with the database and refused
 - 4c. User prompted to enter a correct email address
 - 4d. Correct email is provided by the user
 - 4e. If the address is correct a password reset email is sent to user
 - 4f. User sets a new password
 - 4g. Database updated

3.5.3 Object and Class Model



3.5.3.1 Person

Person class has attributes of a person who is going to use DAOS. Name, surname, email and phone number will be asked while a person creating an account. Moreover, a dealership owner will be considered as a person.

3.5.3.2 Company

Company is the user of DAOS. Company uses a Person object's information to login. Company has many dealerships and Company has authorization to reach performance of its dealerships all around the region. Company can also use Assistance utilization.

3.5.3.3 Dealership

A dealership has dealershipID, name, type and address attributes. Each dealership has an owner as a Person. While a company can have many dealerships, a dealership can only have one company. Dealership info will be stored in a database in order to be used in Performance and Assistance service.

3.5.3.4 Region

All regions has a name and will be categorized in the database. Three different types of regions will appear, Country, City and Country. Region class will be used in Assistance and Performance classes while suggesting a place for a new dealership or showing the performance of existing dealerships.

3.5.3.5 AccountManager

AccountManager class is the account management system of DAOS. It has methods to provide Company users signup, login and change their information for their account.

3.5.3.6 DataManager

DataManager class creates a new Company for the database and manages both Company's and Dealership's information.

3.5.3.7 Assistance

Assistance class has two functions. suggestOpen will take a Region as a parameter and will suggest an optimized area for a new dealership location, which a dealership will be the most profitable, in the specified Region. suggestClose will also take a Region as a parameter and will suggest a specific dealership that will be most efficient and profitable if closed.

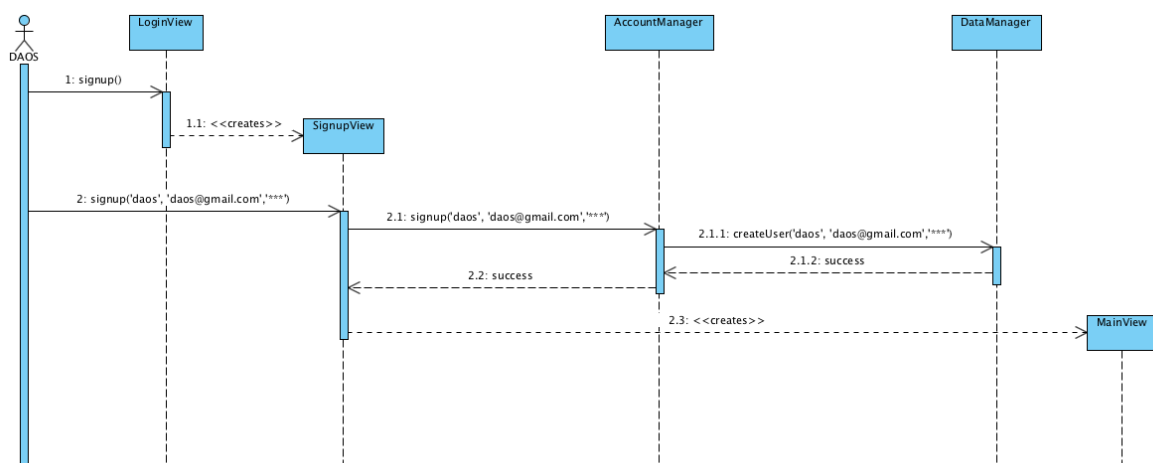
3.5.3.8 Performance

Performance class will show general performance of its Dealerships to the Company. Company will be able to see all of the dealerships' performances in the specified region, a specific dealership's performance and average performance in an area.

3.5.4 Dynamic models

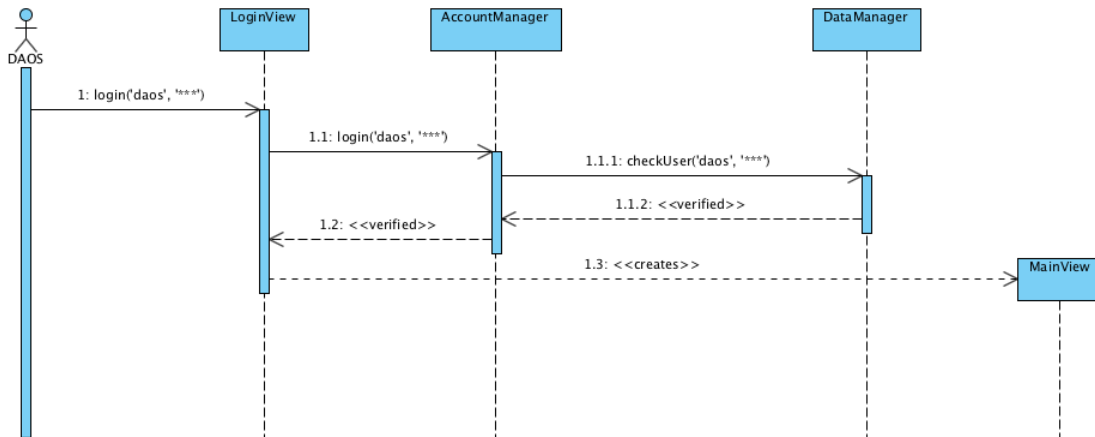
3.5.4.1 Sequence Diagrams

Create Account Sequence Diagram



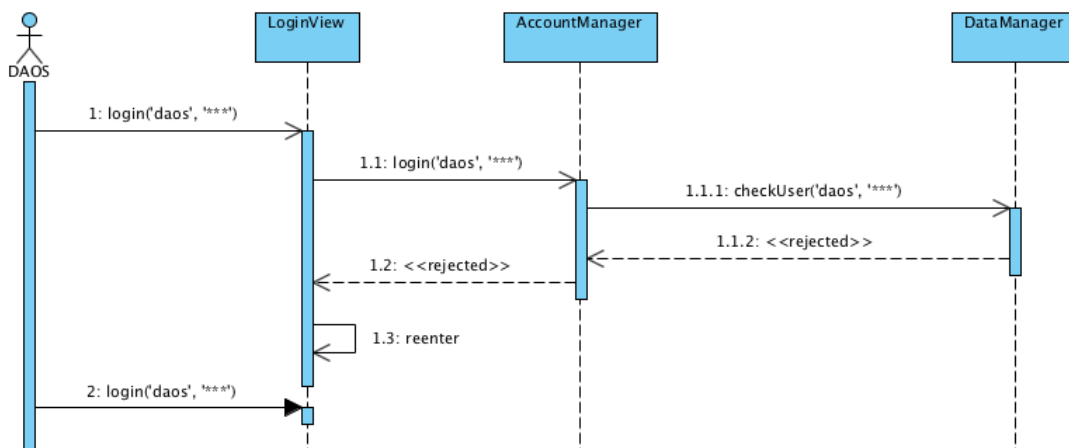
This sequence diagram shows the situation when a user wants to create an account and do it without any problem. First LoginView presented to the user. Then user will be redirected to SignupView in order to complete signup process. User will enter username, email and password information. These information will be passed to AccountManager and from there to the DataManager that will create a user with received data. Success response will forward back till the SignupView and then it creates a MainView in order to display to the user.

Login Success Sequence Diagram



This sequence diagram shows the situation when a user wants to login to the application. First user sees login view which is named LoginView. User enters the required information that are username and password and click to the login button. The information will be passed to the AccountManager and from there to DataManager that will check whether entered information exist or not. In this case, DataManager will return a verification message to the AccountManager and after it has verification message, it will pass it to the LoginView and it creates MainView of the application in order to display to the user.

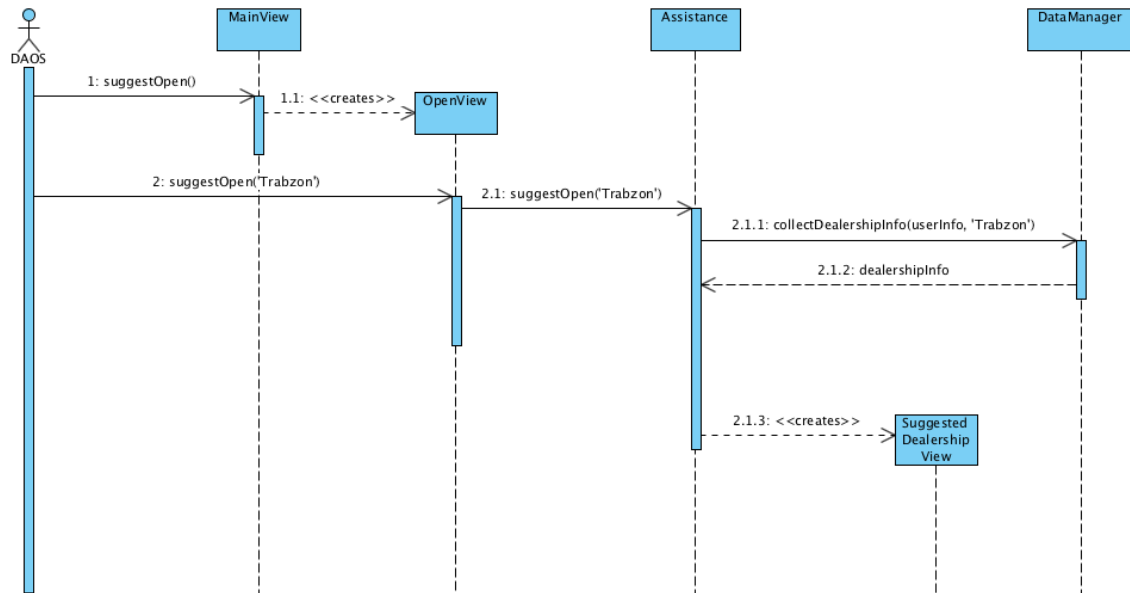
Login Fail Sequence Diagram



This sequence diagram shows the situation when a user tries to login to the application with wrong data. User enter username and password in the LoginView. As in the previous scenario will be forward to the AccountManager and from there to the DataManager.

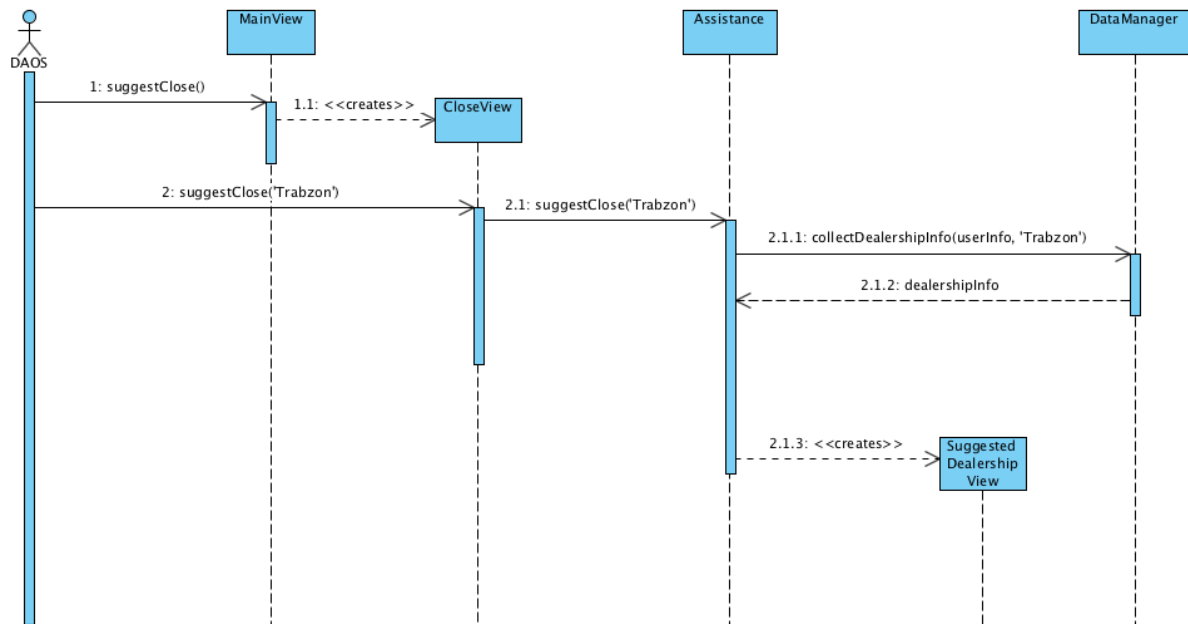
DataManager checks entered data and return a negative response to AccountManager informing it that there is no user with entered information. AccountManager forwards this message to the LoginView that recreates itself to ask the user again for the right information.

New Dealership Suggest Sequence Diagram



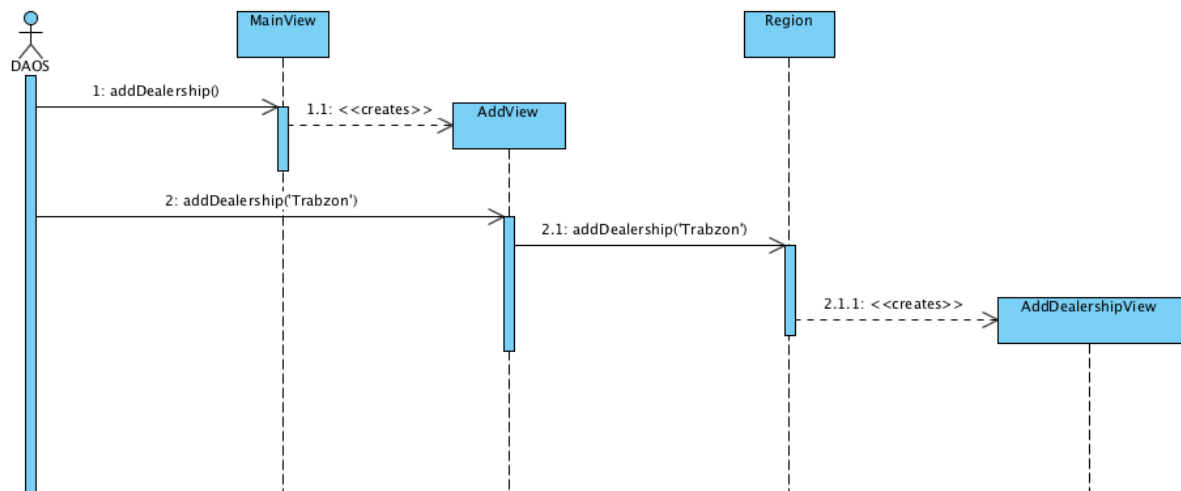
This sequence diagram describes the situation when a user decides to get new dealership suggestion from system. For this, user first interacts with the home page screen of the application named MainView. User interacts with OpenView where he can choose a region to limit the results for a specific region. User specifications will be passed to the Assistance and it asks DataManager information of user's dealerships. With the received information and other crucial information related to dealerships that hold in databases will be processed and Assistance suggests a place for a new dealership through SuggestedDealershipView.

Close Dealership Suggest Sequence Diagram



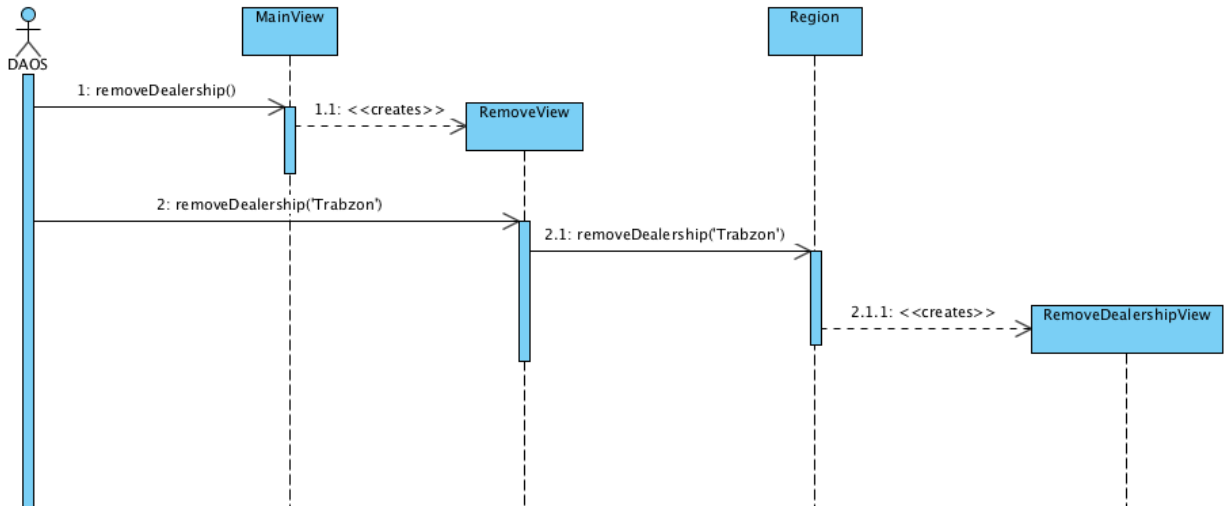
This sequence diagram describes the situation when a user decides to get suggestion from system to close dealership. For this, user first interact with home page screen of the application named MainView. User interacting with CloseView where he can choose region to limits the results for specific region. Users specifications will be passed to the Assistance and it ask DataManager information of user's dealerships. With the received information and other crucial information related dealerships that hold in databases will be processed and Assistance suggest most proper dealership to close through SuggestedDealershipView.

Add Dealership Sequence Diagram



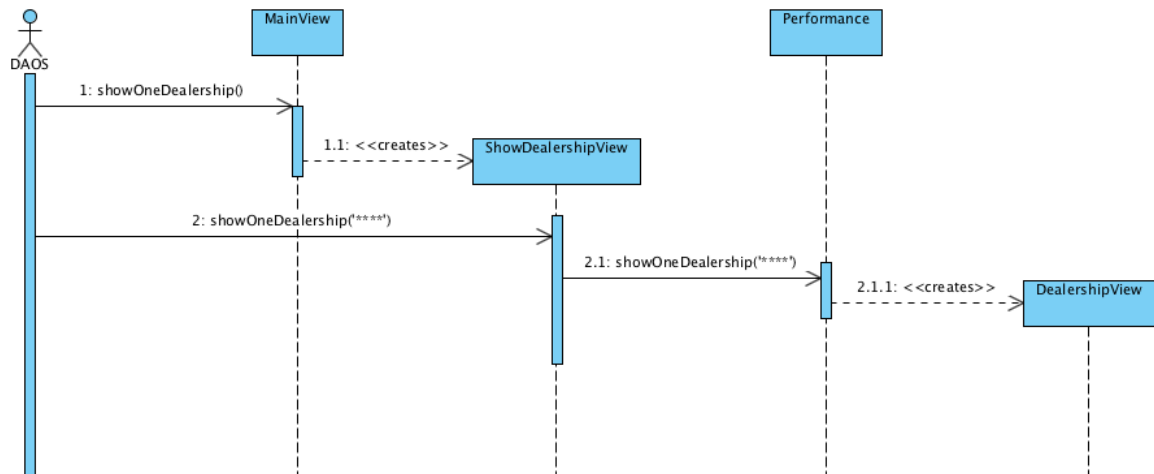
This sequence diagram describes the situation when a user decides to add new dealership to the system. For this, user first interact with home page screen of the application named MainView. After clicking add dealership, user interacting with AddView where he can choose region. Then Region creates AddDealershipView and user enters information of new dealership to the system through AddDealershipView.

Remove Dealership Sequence Diagram



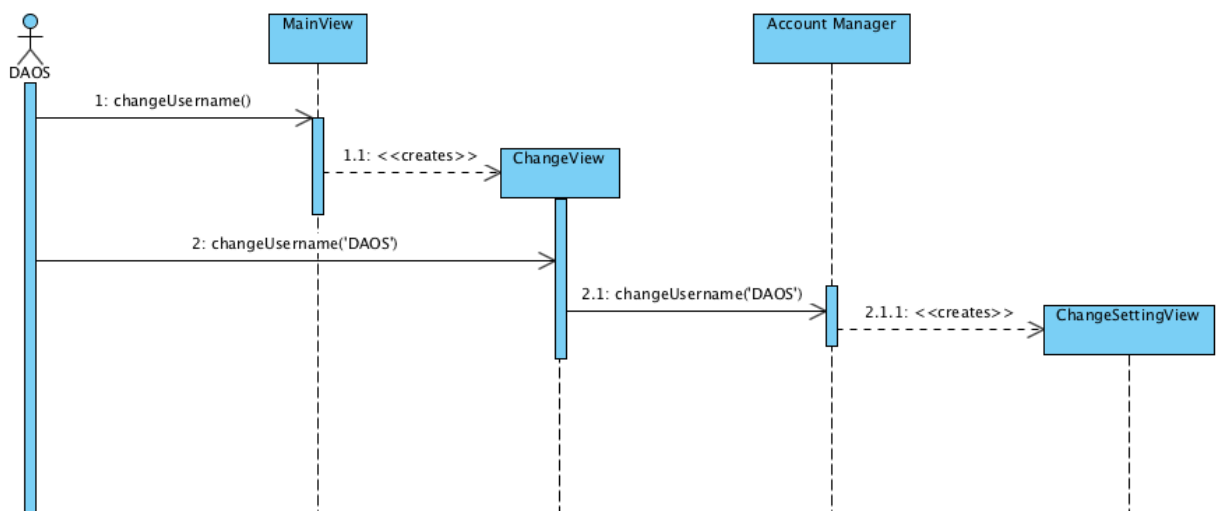
This sequence diagram describes the situation when a user decides to remove new dealership from the system. For this, user first interact with home page screen of the application named MainView. After clicking add dealership, user interacting with RemoveView where he can choose area. Then Region creates RemoveDealershipView and user deletes information of dealership that user wants to remove through RemoveDealershipView.

Show One Dealership Sequence Diagram



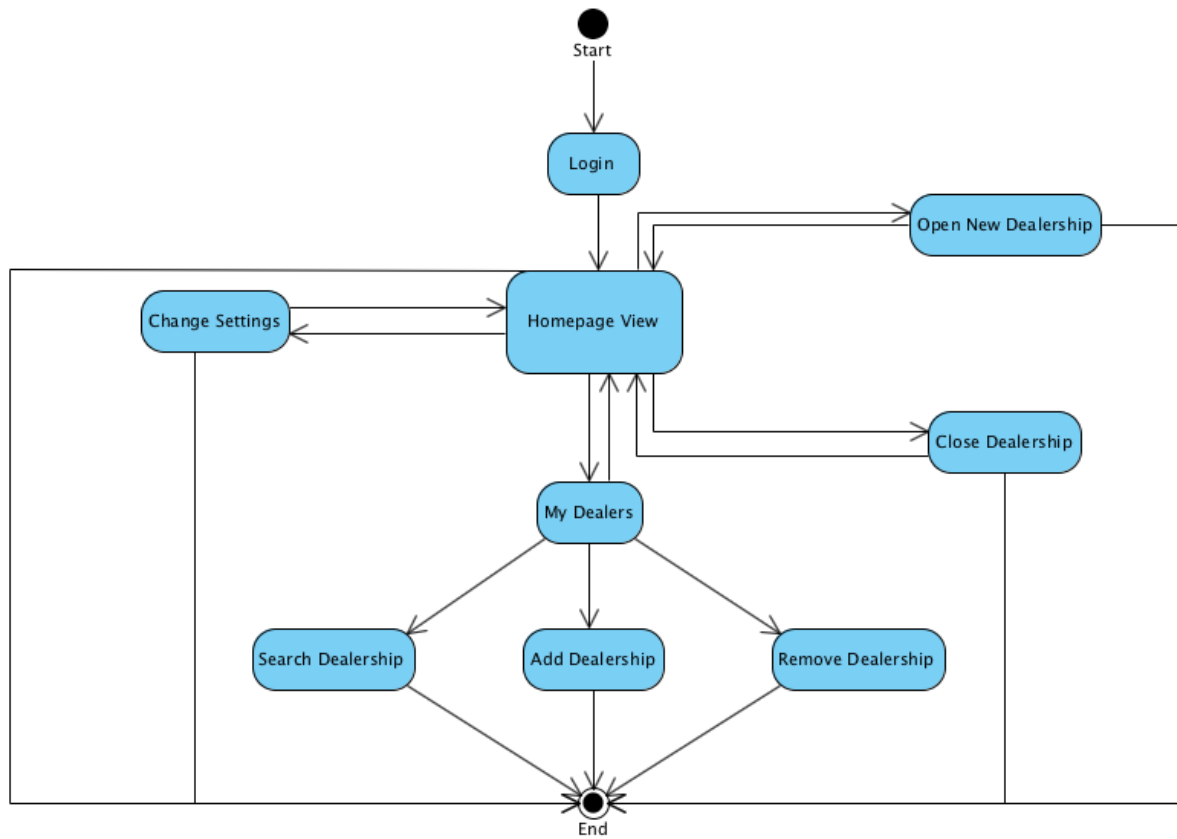
This sequence diagram describes the situation when a user decides to see one dealership's information. For this, user first interact with home page screen of the application named MainView. After entering id of dealership to the ShowDealershipView, Performance creates DealershipView and user can examine the information of dealership that he or she wants to see through DealershipView.

Change Username Sequence Diagram



This sequence diagram describes the situation when a user wants to change his or her username. For this, user first interact with home page screen of the application named MainView. User click change settings and first enters his or her current username then enters new username through ChangeSettingView.

3.5.4.2 Activity Diagram

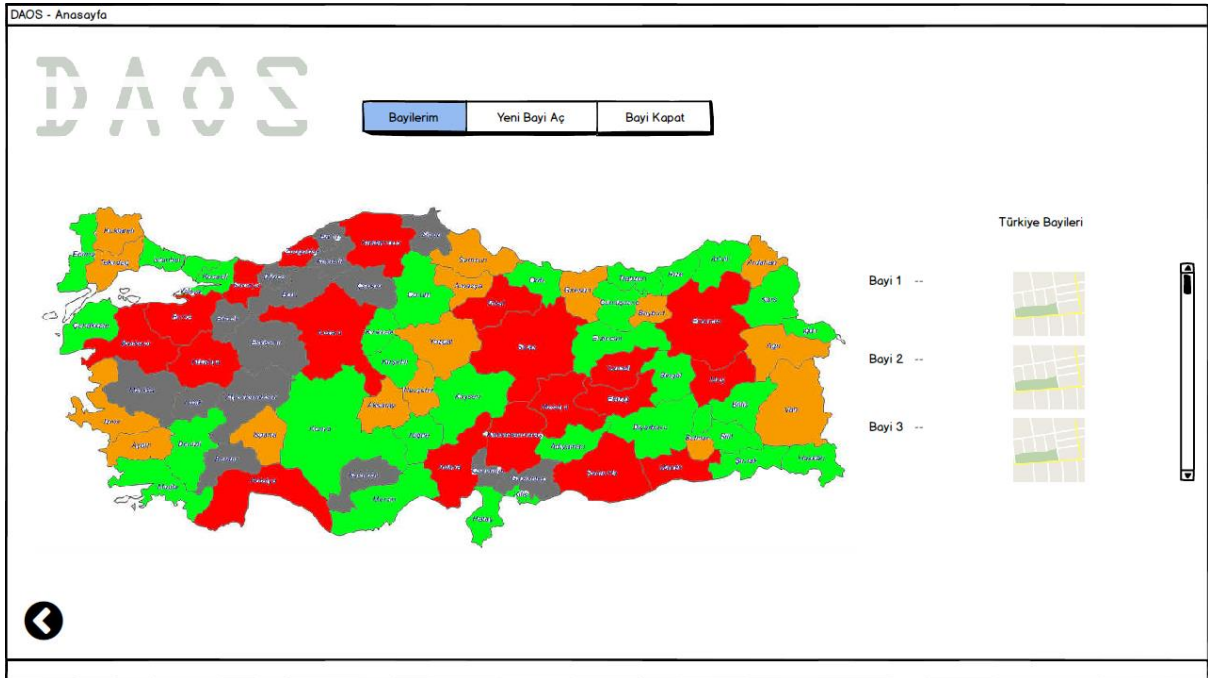


The activity flow starts when the user logs into the system. The user first sees his homepage. User can change his username, password and email through change settings. He or she can also search, add and remove dealership after clicking My Dealers. The user can also take suggestion for new place to open new dealership or dealership to close via clicking Open New Dealership and Close Dealership. User can logout from any screen.

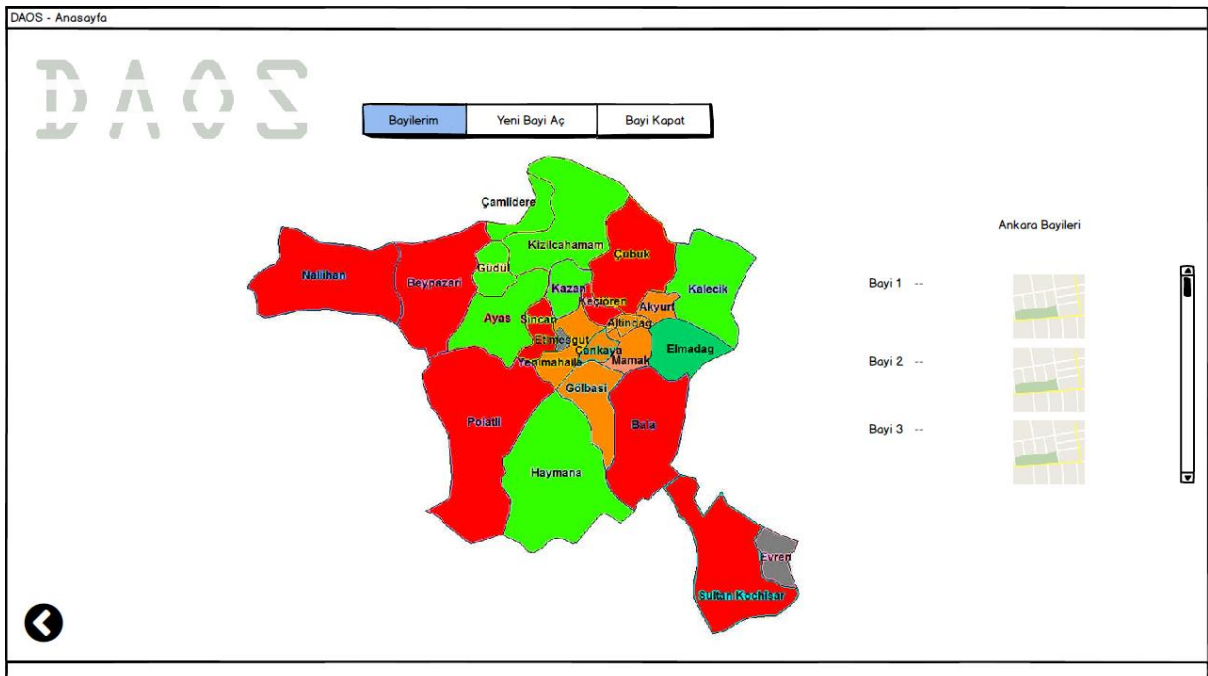
3.5.5 User Interface - Navigational Paths and Screen Mock-ups

Login screen of the program. User enters username and password to log in. User can choose Turkish or English for the language of the program. There is also a “Forgot my password” button.

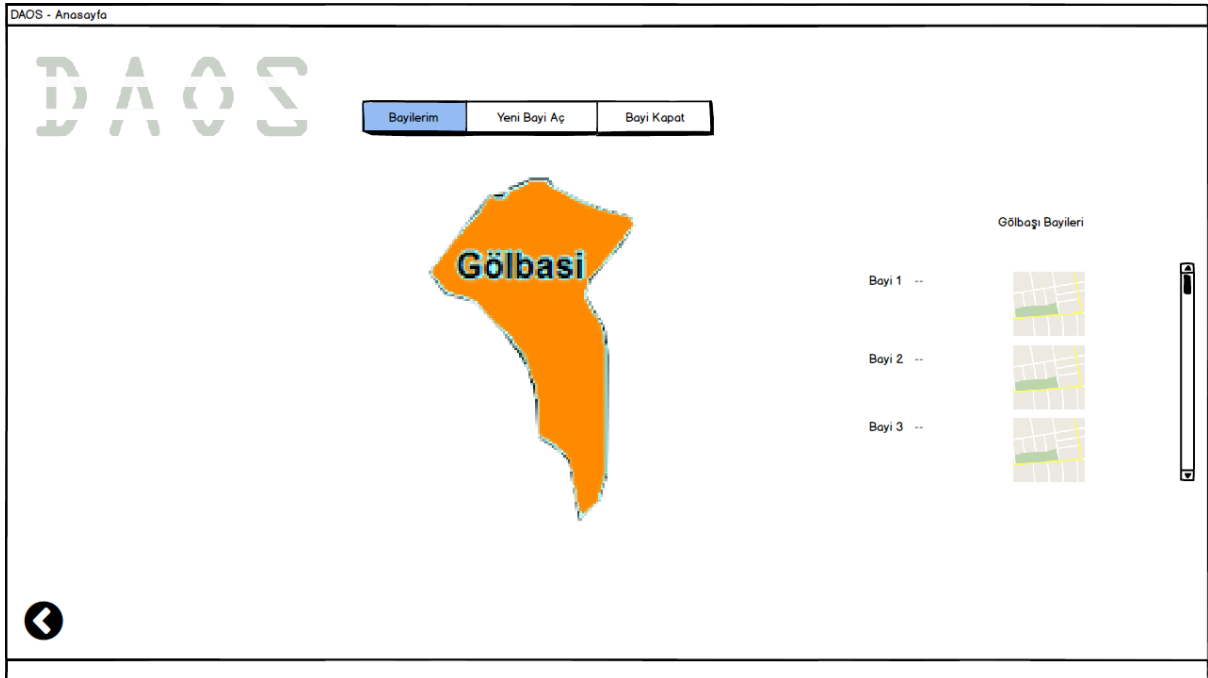
When user clicks “Forgot my password” this screen appears. User can enter his e-mail and get his forgotten password.



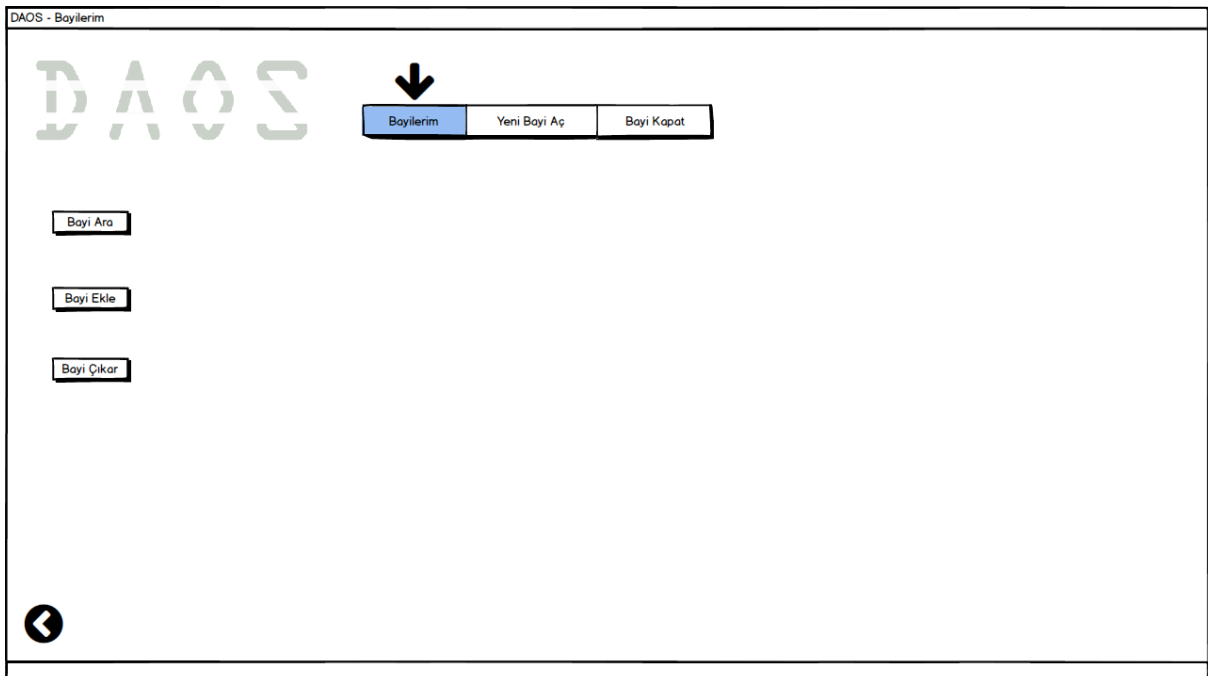
Homepage of the program. When user logins all dealerships he has will be shown.



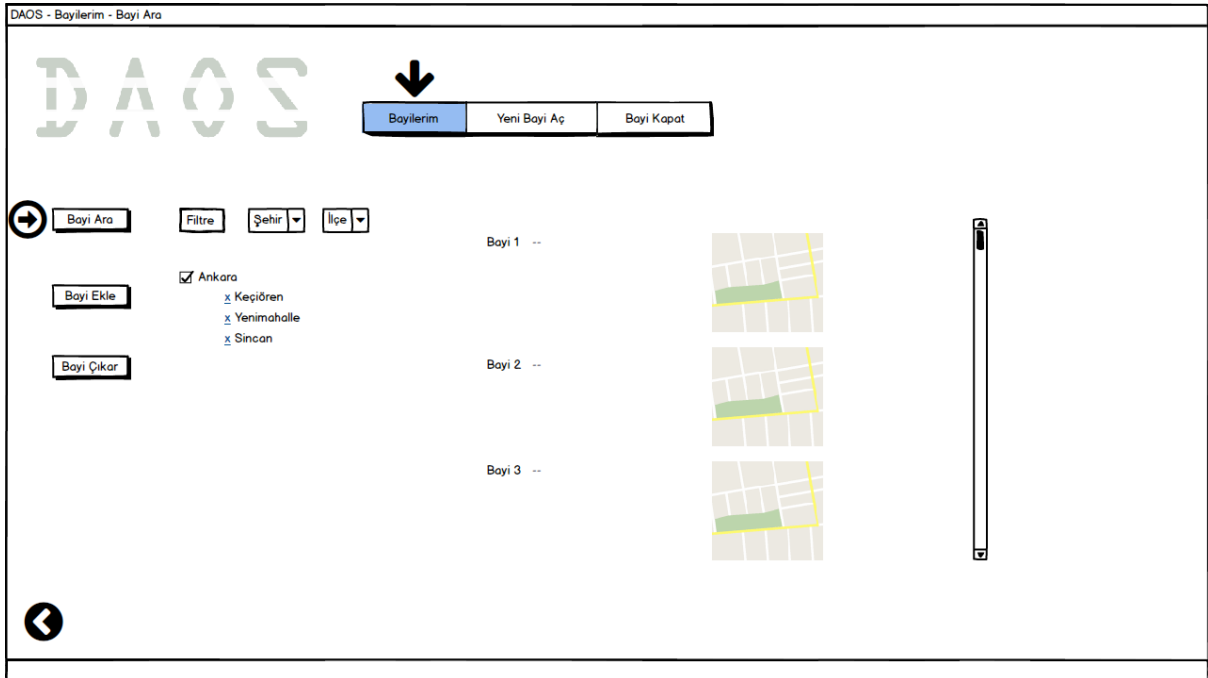
When user clicks on a city, the city map will be shown besides the dealerships user has in that city.



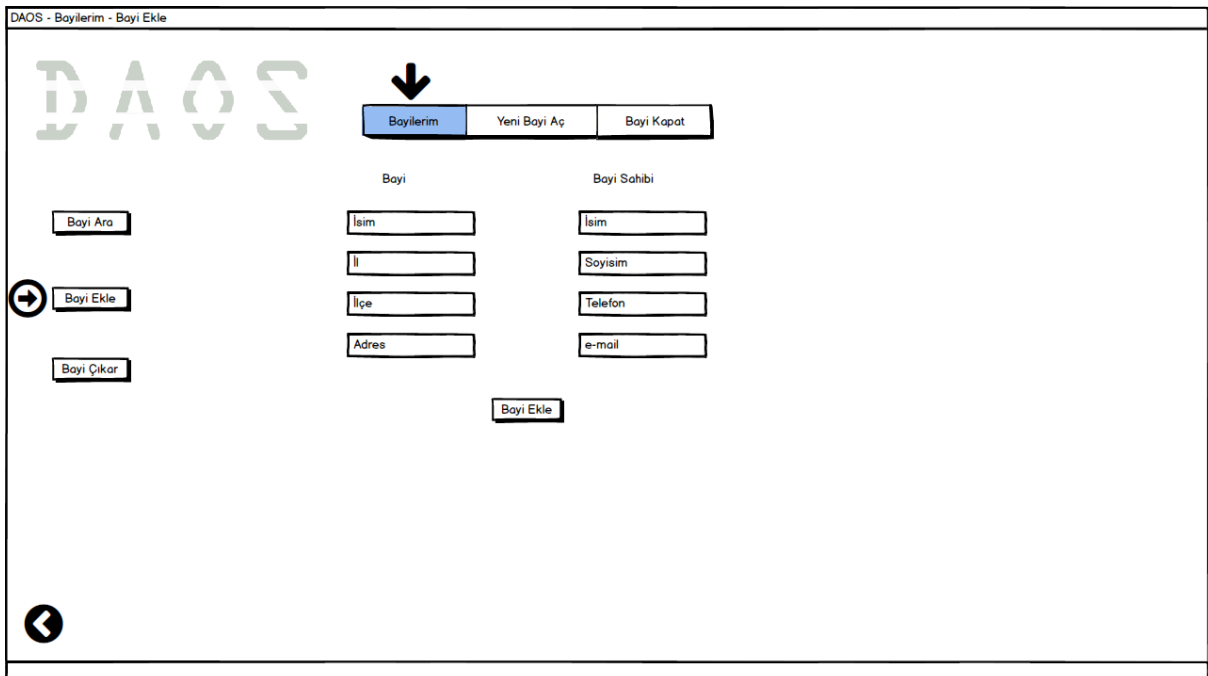
When user clicks on a county, the county map will be shown besides the dealerships user has in that county.



When user clicks “My Dealerships”, there are 3 buttons; “Search Dealership”, “Add Dealership”, “Remove Dealership”.



When user clicks “Search Dealership” he will be able to filter city and county to see all dealerships he has on those areas.



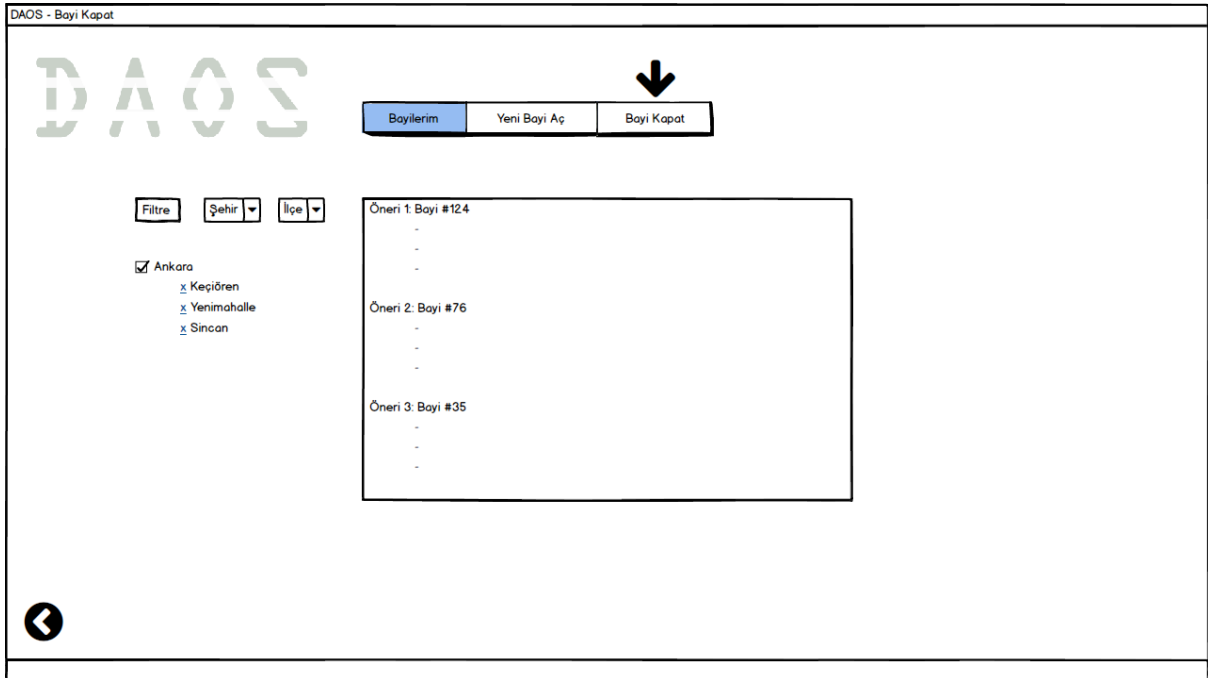
When user clicks “Add Dealership” he will be able to enter the necessary information for the dealership and its owner to add a dealership.



When user clicks “Remove Dealership” he will be able to filter the dealerships and remove it.



When user clicks “Open Dealership” on the homepage he can filter the areas he wants to open a new dealership and the program suggests some number of places to be opened and show information about why the company should open a dealership such as average cost and profit for the company.



When user clicks “Close Dealership” on the homepage he can filter the areas he wants to close a dealership and the program suggests some number of places to be closed and show information about why company should close them.

4. Glossary

DAOS: Dealership Assistance and Optimization System

GIS: Geographical Information Systems

POI: Point of Interest

5. References