Bilkent University

Department of Computer Engineering

Senior Design Project

Project name: DAOS (Dealership Assistance and Optimization System)

High Level Design 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

December 22, 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

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?”

1.1.Purpose Of The System

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.

1.2.Design Goals

Usability: 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.
Reliability: DAOS will make critical suggestions to investors, which means that DAOS should be a reliable assistance.
Adaptability: DAOS will be a dynamic platform that should be in a continual change in order to reach the most updated data.
Availability: DAOS should be precise and available any time investors need suggestions or want to follow their existing investments.
Security: Investors and dealerships data should be secured.

1.3.Definitions, Acronyms, And Abbreviations

DAOS: Dealership Assistance and Optimization System
GIS: Geographical Information Systems
POI: Point of Interest
UI: User Interface

1.4.Overview

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.

2.Current Software Architecture

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 Software Architecture

3.1.Overview

The architecture of the proposed system is described in detail conforming to the following course. First the subsystems and their components, both the server and client application, are shown in diagrams and described in detail in the subsystem decomposition section. To provide a better understanding about our software and present the software architectures used. Then, the hardware-software mapping is presented to show which parts of our software is intended to work on which device and their communication. This section is followed by persistent data management explains our database the values we are keeping in the database and reason for making the explained selections. In access control and security part we are depicting how we are providing security for both inside of our system and transition from database to the client in our system and which partition of the data can be

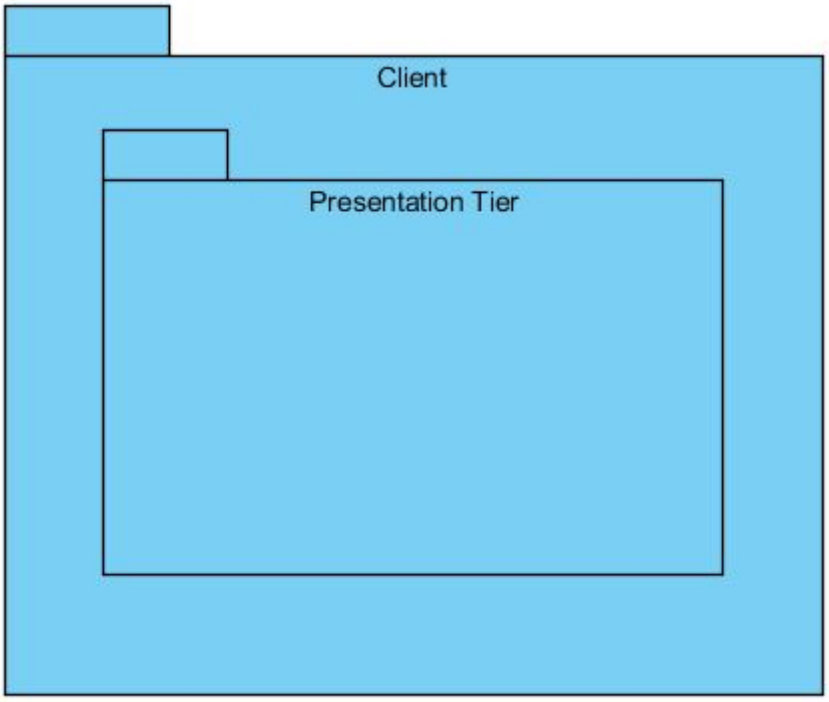
accessed by whom. Global software control explains the control flow of our software and behavior of our subsystems according to user input. At last boundary condition section describes how the software initializes terminates and the behavior of our system when exceptions occur.

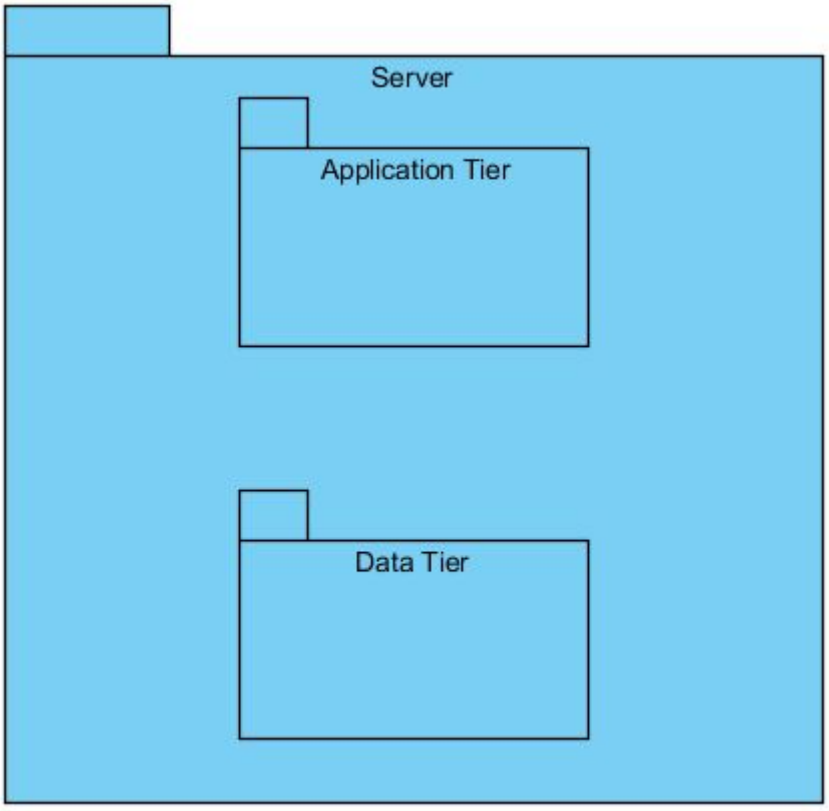
3.2.Subsystem Decomposition

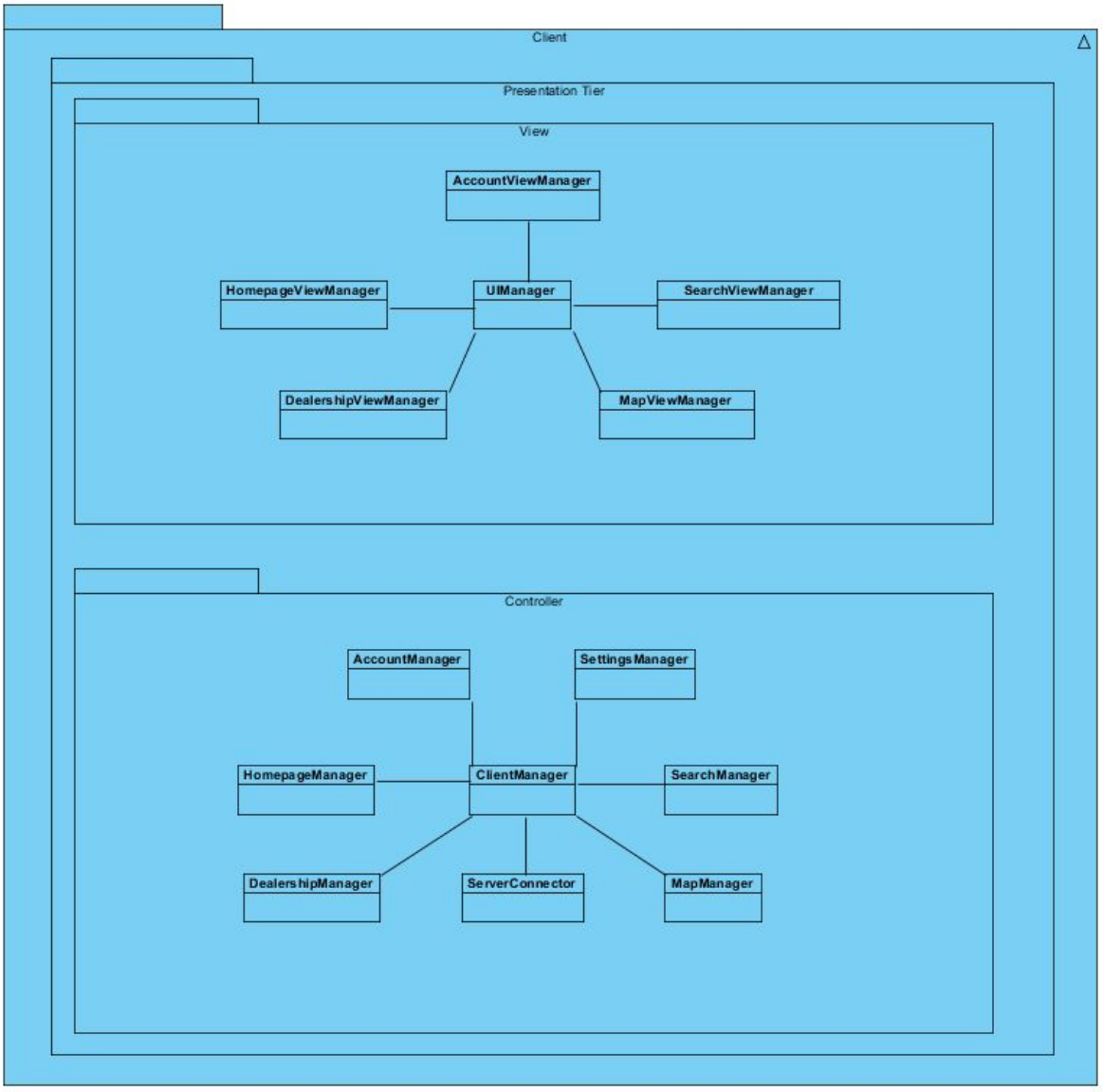
DAOS mainly follows a Client/Server architecture where we designate the server to handle all queries and calculations. The client side provides the interface to our user to communicate, make queries and changes to the server and display the response from the server. Client will be a single executable on windows which present all of our functionalities to the users. Many clients will connect to our one server which is an apache HTTP server working with a MySQL database solution. This way we aim to provide smooth transitions on our interface to our user provided by the client and all the processing load is on the server. Maintenance and performance increase of the server can be made centrally so scalability is ensured with this architecture model.

Client has only presentation tier which consists of a View subsystem and a Controller subsystem. View is mainly responsible for UI and display. The user consistently interacts with the view subsystem and this subsystem is responsible for transmitting the inputs to the controller and provide the user with a seamless to use interface. All subcomponents of the view subsystem is connected to a main component UIManager and every compartment interacts with another via UIManager. The controller subsystem is responsible for supervising the incoming commands from the view and transmit them via ServerConnector comportment. The main compartment in the controller is ClientManager and all the other ones interact with ClientManager then ClientManager preprocessing is complete the instructions to our server is made by ServerController which can work bidirectional which means both the user input and the result created by our server passes through this route.

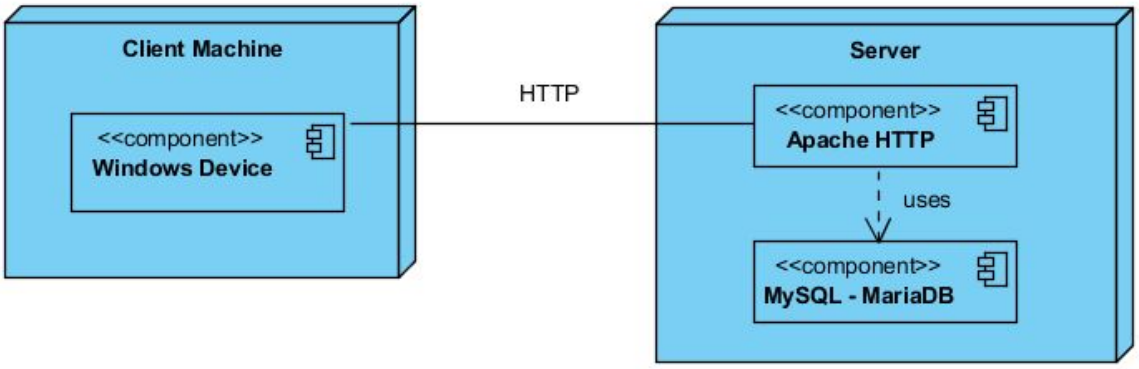
Server has two tiers, Application tier and Data tier. Application tier is responsible for all the processing which will be performed by DAOS the inputs come into application tier with ClientConnectionManager (which itself communicates with ServerController in the Controller in the Client side). This is the component where the input or query comes then be processed or the result of the query's final destination in the server side. Application tier also communicates to data tier which all of our information is stored in the database for apply the logic of the DAOS and if necessary make the updates on the database.


Client
Presentation Tier


Server
Application Tier
Data Tier


Client
Presentation Tier
View
AccountViewManager
HomepageViewManager
UIManager
SearchViewManager
DealershipViewManager
MapViewManager
Controller
AccountManager
SettingsManager
HomepageManager
ClientManager
SearchManager
DealershipManager
ServerConnector
MapManager

3.3.Hardware / Software Mapping



DAOS requires an internet connection in order to connect to the database. Client machine will interact with the server using HTTP requests. When a user opens the program on their computer it will be connected to the MySQL database which is in Apache server.

3.4.Persistent Data Management

Time is one of the most important things for companies. We need to provide information as soon as possible. Some of our objects such as person and dealership will not be changed if there is no need to change information about dealerships or the person using DAOS. Until the user change these informations, adds or removes dealership, the data remains. If the user searches dealerships, it needs to be shown quickly. We chose to store our data in MySQL - MariaDB, we believe it will satisfy our needs.

3.5.Access Control And Security

DAOS is working on Windows device and it is not accessible from external users. In order to access system, registered user has to enter his information. When user try to login, system checks the corresponding data in database system. According to information stored in database, system allows or denies the user to enter the system. If system denies user, system stays in the login page and asks login information again. Therefore, user accounts are secured by users' login information. Also, in order to provide safe and secure system, the information of clients will be stored encrypted in database.

3.6.Global Software Control

In our project, server-client architecture is favored and system decomposed into two main subsystems which are client subsystem and server subsystem. Our platform work based on multi-layer architecture. Also, our system has an event-driven software control. When user wants to make changes in his account such as registering new account or change account information, the system updates the account of the user accordingly. When client add or remove a dealership, system updates the database and notify subsystems to make sure future recommendations will be more accurate. When clients are in the system, they contact with main server and it will be connected with the database.

3.7.Boundary Conditions

Initialization

User can access DAOS by installing it to their Windows device. User needs to create an account to login. Also in order to login, users need to enter their username and password. If entered information matches an entry in database, that user will be logged in and the homepage of DAOS will be displayed. If entered information does not match, user will not be logged in and asked to login information again. In addition, the application retrieves real-time data to perform its functionalities thus it requires an internet connection.

Termination

Users are able to terminate the application by logging out from system. If user clicks logout button when an update is being performed, system wait it to finish. Instant termination is allowed during all other processes.

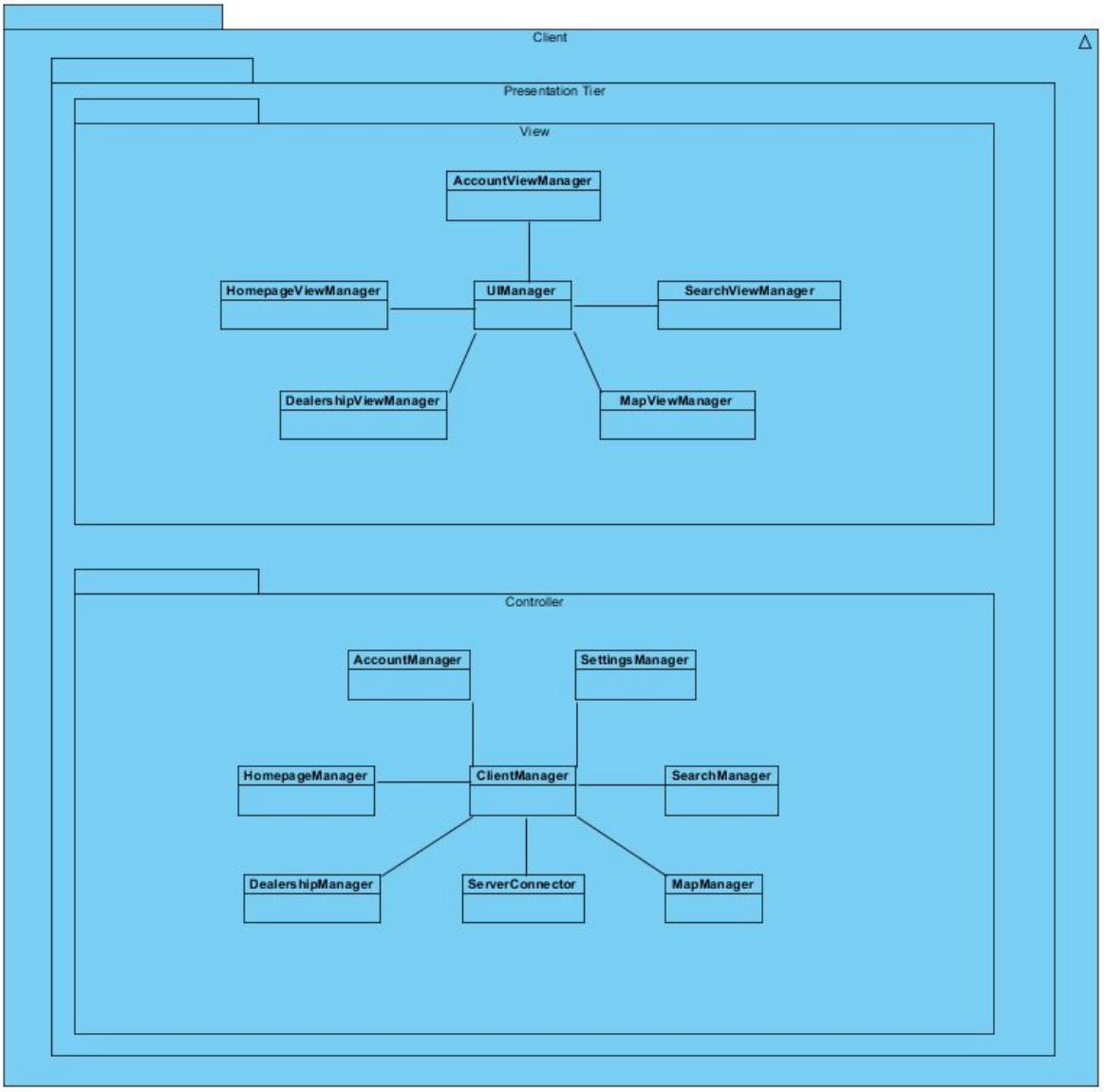
Exceptions Handling

Failure of the application can occur when there is no internet connection. In this case, the real-time data retrieval for dealerships will not occur and suggestions for dealerships cannot be done. The failure occur if there is a connection error during an operation which involves editing the database.

4.Subsystem Services

This part of the report analyzes the subsystems of our system and describes the services they provide in detail.

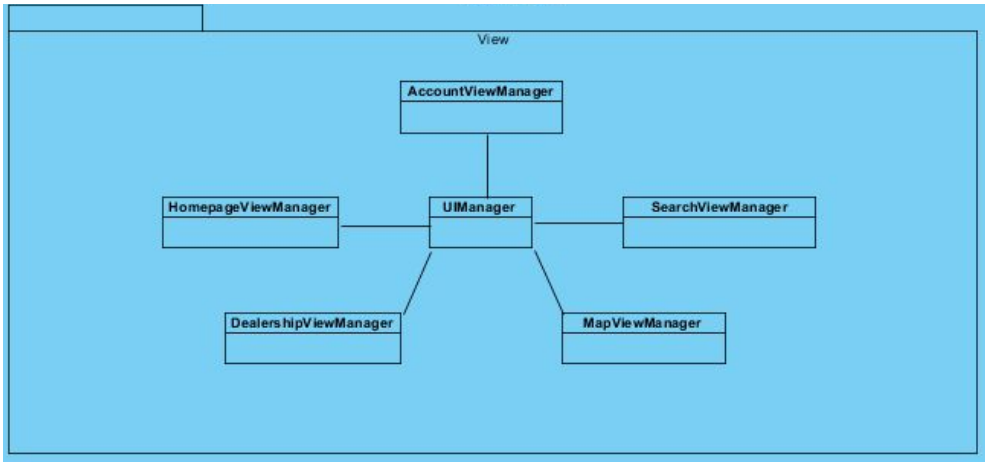
4.1.Client



The client corresponds to the desktop application of our system. The client is the presentation layer of our system. The user creates an account or logins to the system

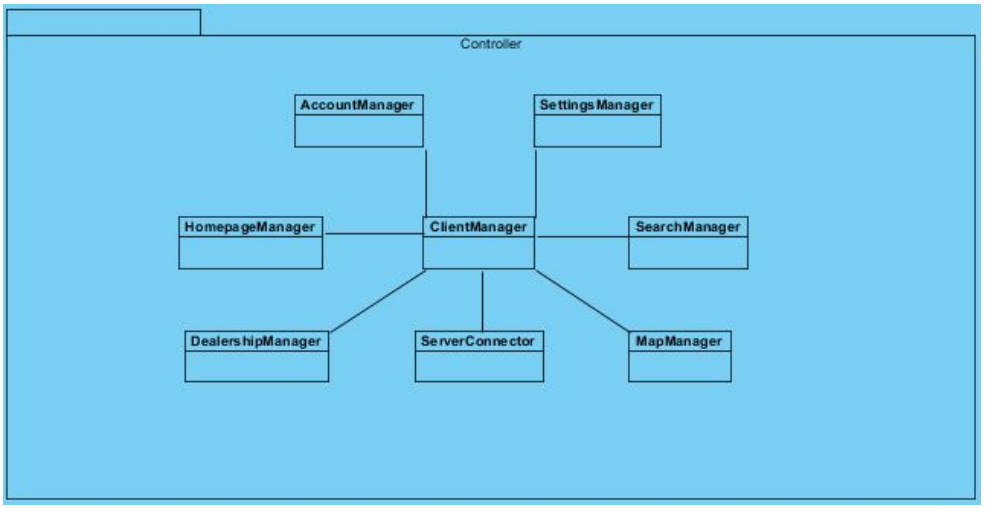
via the client. The client requests login access from the server. The client manages the account of the user, the preferences and searching. The user specifies the filters to see the current or past performance of the dealerships.
Client subsystem includes Presentation Tier and the Presentation Tier has View and Controller subsystems. View subsystem is responsible for all the user interface operations. Controller subsystem manages the interaction between the client and the server and controls the operations within the client. It collects the data from the application and sends it to server. It also requests the required data such as performance report or suggestion requests.

4.1.1.View Subsystem



UIManager: Main User Interface class that manages other UI classes.
AccountViewManager: Class for arranging the account view.
SearchViewManager: Class that manages dealership and region searching and also showing the filters.
MapViewManager: Responsible for map views.
DealershipViewManager: Class that is responsible for showing filtered and/or sorted dealerships.
HomepageViewManager: Class that presents all the components of the users' homepages.

4.1.2.Controller Subsystem



ClientManager: Main manager class that manages all operations of other controller classes.

AccountManager: Responsible class for account information.

SettingsManager: Class that will change the settings by the users' preferences.

SearchManager: Class that is responsible for filtering, searching and sorting.

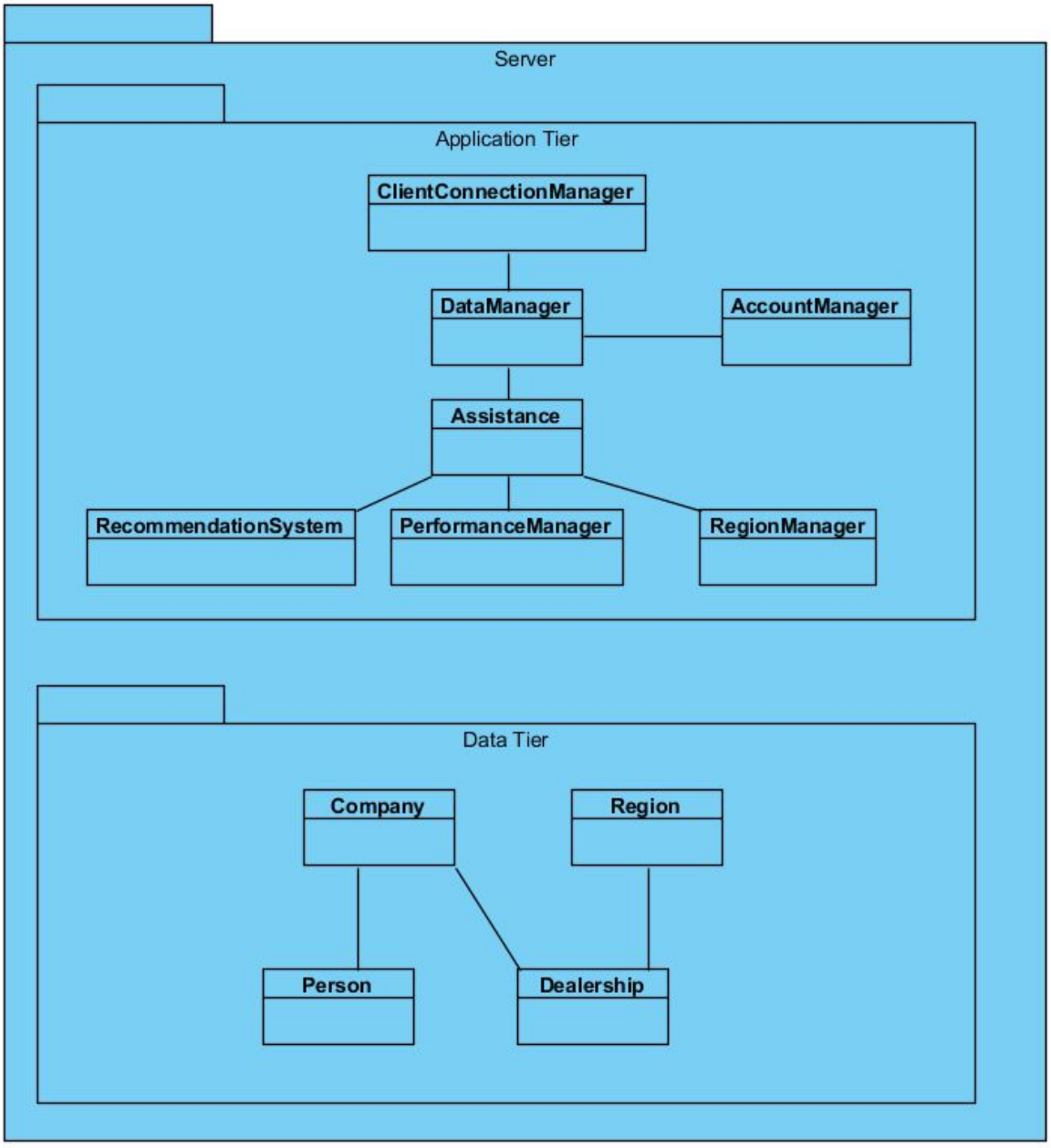
MapManager: Class that is changing the map according to performance and new dealerships.

ServerConnector: Class for handling interaction between the client and the server.

DealershipManager: Class that is responsible of details of dealerships.

HomepageManager: Class that decides the information on the homepage.

4.2.Server

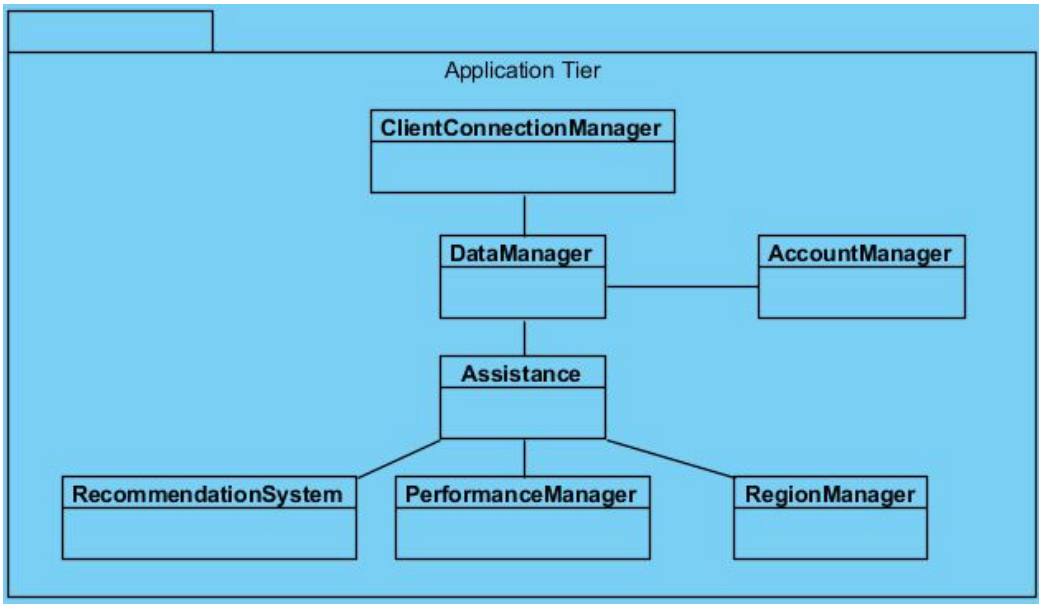


Server is the part of our system where the crucial operations such as showing performance and dealership location suggestions are handled. The client sends data of the user to server. Server contains all data of all users. Furthermore, server is responsible from collecting dealership and region details from Google Maps. Server continuously analyzes the performance data to improve the recommendation system. Server has crucial functions such as showing performance details and suggesting a location for a new dealership. When the

user requests a new suggestion, the client sends the filters for the suggestion to server. The server creates the best possible suggestion and sends it back to the client.

Server includes two layers, Application Tier and Data Tier. Application Tier is the main operational layer that handles all performance and suggestion functions. Application Tier also interacts with the client to respond to the requests of the clients. Data Tier includes the Database Management Subsystem. This subsystem represents the database where the region, dealership and account information are stored.

4.2.1.Application Tier



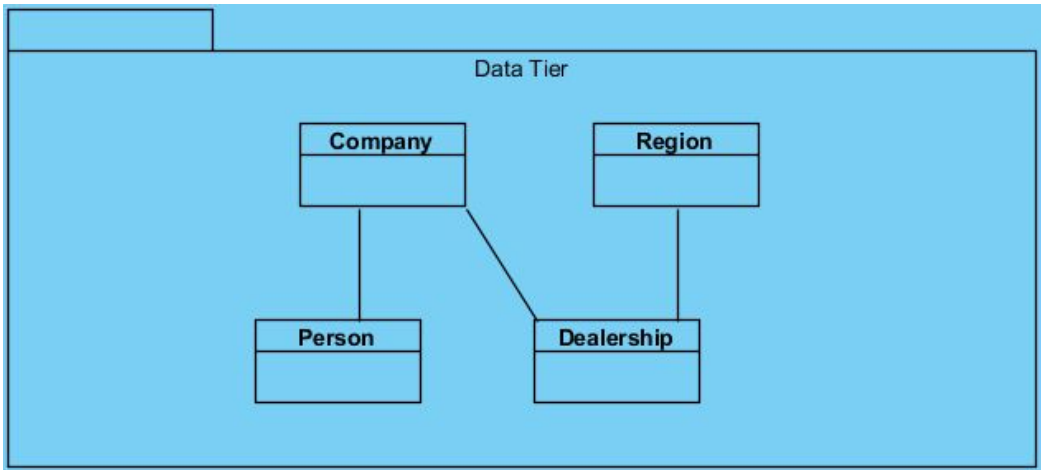
ClientConnectionManager: Every client is using a determined area in our server. ClientConnectionManager's job is to separate all the users and keep their information safe from others.

DataManager: Manager class responsible from obtaining necessary data from the Data Tier. All the data about region, company and dealerships is kept in the database. DataManager collects the data required for the current operation. It also manages data update operations.

AccountManager: Class that manages the accounts of companies. All data of the user accounts are handled by this class. AccountManager interacts with the client and the Data Tier to manage account operations.

Assistance: Class that is responsible for assisting the user. Assistance class analyzes the performance of dealerships and remarks positive or negative events.
RecommendationSystem: Responsible class for DAOS main feature, recommendation. RecommendationSystem will calculate the best possible location for a new dealership.
PerformanceManager: PerformanceManager is responsible for calculating the performance of the dealerships.
RegionManager: Class that is managing the regions. Regions are very crucial part of the system because of the dealerships' nature of getting affected by the environmental changes.

4.2.1.Data Tier



Company: Company is the main customer of DAOS. Companies' information will be held in Company.
Person: Contact person of a company will be held in Person.
Region: Crucial data of a region such as POIs will be held in Region.
Dealership: Dealerships' information will be held in Dealership.

5.Glossary

DAOS: Dealership Assistance and Optimization System
GIS: Geographical Information Systems
POI: Point of Interest

6.References