TABLE OF CONTENTS

1. INTRODUCTION

1.1. Project Definition

Nowadays, technology rises in an enormous way and accessing information becomes very easy. Importance of mobility accelerates technology towards developing smaller and demand meeting devices. Many people face with the dilemma of purchasing a powerful and immobile system or a less powerful and mobile system. If those mobile devices are provided with applications that meet at least the basic requirements of daily life, then that dilemma will disappear.

Mobile devices provide people with position independent information access. The most popular mobile devices are mobile phones; however, their independency is limited by the quality and availability of GSM operators. At this point, Global Positioning System (GPS) arrives on the scene and serves position information at anytime, anywhere. However, GPS itself does not produce high quality and detailed solutions. Thus, a Personal Digital Assistant (PDA) which has GPS and internet (wireless or GPRS) connection may be assumed as unrivalled among all mobile solutions. Because, PDA's computational power among the mobile devices and GPS' ability to serve accurate position are combined is this package.

We are planning to utilize this mobile power. We are going to serve a Mobile GIS application for people to search for the nearest cinema, concert hall, theatre, sport center and transportation to these social centers. The position of the user will be taken via a GPS connection and also request of the user about cultural activity will be gathered from the PDA. Then they will be sent to the server. After request information is processed, a result will be returned to the user with visual (map) data. Moreover, user will be guided to desired location with transportation alternatives. This Location Based Services (LBS) application of ours is named CoreAccess.

1.2. Project Scope and Goals

In this project, we aim the followings:

- to search for any cultural activity or place,
- to display most common cultural facilities to the customers (users),
- to donate the user with alternative results by the help of ontology in case of not finding exact matches to queries,
- to visualize the result with vector maps,
- to rotate/zoom in/zoom out/scroll the returned map,
- to give information about the name, address, place of the activity with some comments,
- to guide the user to the desired location with possible transportation alternatives,
- to make some estimations about the reaching time,
- to give multilingual support,
- to reach information anytime in anywhere,
- to provide the user with ease of use.

In this project, we have focused on these topics:

- to organize a complete project,
- to program the PDA in a user-friendly manner,
- to make PDA communicate with application server via Web Services,
- to generate ontology for cinema and transportation,
- to implement affective shortest path algorithms,
- to manipulate map data in GIS part.

2. PROCESS

2.1. Team Structure

Since we have worked together for a long time, we decided that Democratic Decentralized (DD) fits best to our project. Each member in the group has a talent in different areas. Therefore they can make some suggestions about their research and their area of specialization. At the end the suggestion has been voted and decision has been taken.

2.2. Process Model

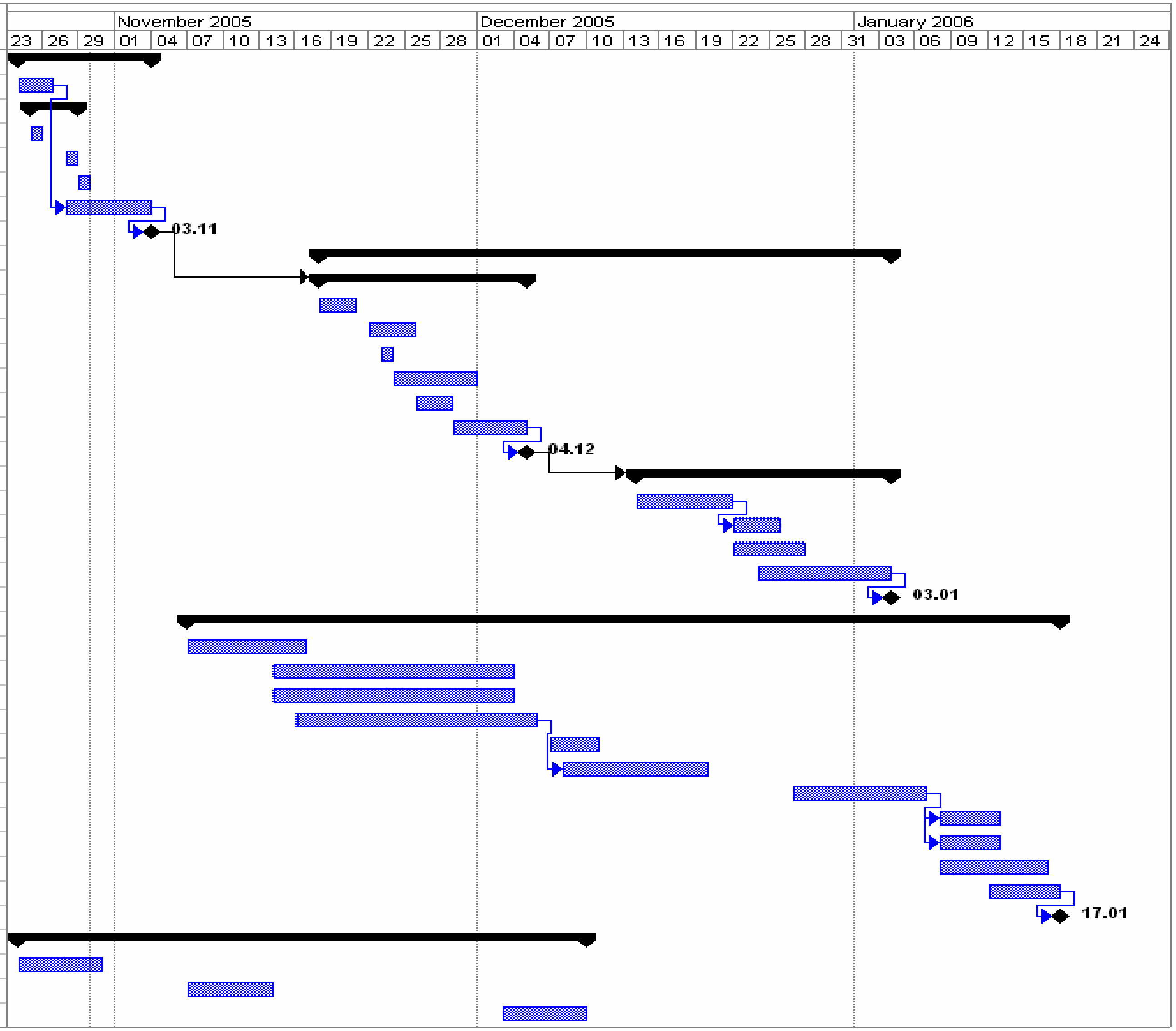
Our project has certain steps which are predefined. These steps are requirement analysis, initial design, detailed design, implementation and testing. Although deadlines of these steps are determined, we will release several prototypes in implementation stage and we have to consider their feedbacks. Thus, we may return back to some parts of detailed design according to the success of the prototypes. As a result, we decided that iterative process model best suits to our project.

3. PROJECT SCHEDULE

3.1. Gantt Chart

In the following gantt chart, only one semester, fall 2005, was considered.

ID		Task Name	Duration
1		**Analysis**	**11 days**
2		Template for Analysis Report	3 days
3		**Interview about GIS**	**4 days**
4		Talking with Veysi Isler	1 day
5		Talking with Netcad	1 day
6		Contacting with Ege University	1 day
7		Last version of Analysis Report	7 days
8		Milestone	0 days
9		**Design**	**46 days**
10		**Initial Design Report**	**17 days**
11		ER diagrams	3 days
12		Initial User Interfaces	4 days
13		Project schedule	1 day
14		Initial Component Level Design	7 days
15		Defining constraints	3 days
16		Forming the complete report	6 days
17		Milestone	0 days
18		**Detailed Design Report**	**20 days**
19		Research on multilinguality with .NET	8 days
20		Drawing Final User Interfaces	4 days
21		Detailed Component Level Design	6 days
22		Forming the complete report	10 days
23		Milestone	0 days
24		**Demo Preparation**	**68 days**
25		Example User interface that runs on the PDA	10 days
26		Research on GIS	20 days
27		Research on vector maps	20 days
28		Research on ontology (OWL, OWLS, OWL in:	20 days
29		Running a sample program on the real PDA	4 days
30		Studying tutorials on Protege	12 days
31		Study on web services	10 days
32		Implementation of a simple webservice caller	5 days
33		Implementation of a simple webservice provic	5 days
34		Initial Server side application	7 days
35		Demo Integration	4 days
36		Milestone	0 days
37		**Other**	**47 days**
38		Preparing web Site	7 days
39		Adding Forum to web site	7 days
40		Preparing presentations	7 days



4. PROJECT REQUIREMENTS

4.1. System Requirements

4.1.1. Hardware Requirements

Our hardware requirement decomposes into three main groups; first one is development side requirements, second one is server side requirements, and the last one is client side, which is mobile, hardware requirements.

Development – Side

For fast and comfortable development, approximate minimal requirements are as follows:

- 1.6 GHz CPU
- 512 MB DDRAM
- 30 GB HDD
- 16 MB Video Card
- Ethernet Card
- Internet Connection

Our own computers meet those minimal requirements, so we don't expect to face with a difficulty.

Server – Side

Server will act like a Web Server, it will always stay open and provide Web Services to several clients. It has to have a broadband internet connection and two HDDs which have huge capacity. Huge capacity is for various databases like maps, places, etc. SCSI HDD would be better for us, because SCSI standard is the best for servers. Those are designed for 24 hours of usage. Finally, two HDDs are for backup purposes, RAID 1 is the best choice. Requirements are as follows:

- AMD Athlon 64 3000+ 939 Venice CPU
- 1 GB DDRAM
- 2 x 80 GB SCSI HDD (RAID support)
- 32 MB Video Card (may be onboard also)
- Ethernet Card
- Broadband Internet Connection (high upload rate)

Client – Side

In fact, since we are planning to use Web Services and serve our application to every client, there may be several types of clients that have the ability to connect with web services such as PCs, workstations, laptops, etc. However in this part, we have to concentrate on the mobile clients PDAs, others can have any configuration. It has to have GPS connection, and we need a GPS receiver. Moreover, for connecting PDA to internet, it must have wireless connectivity (wi-fi 802.11b). Bluetooth is also necessary because connection between GPS receiver and PDA will be supplied via Bluetooth. Moreover, wireless connection is not available everywhere, so, also for GPRS connection, Bluetooth is the best standard between PDA and mobile phones that have a Bluetooth port. When there is no wireless connection in the area, both GPS receiver and mobile phone with a GPRS connection have to be connected to PDA. In that case, there seems a deadlock because of the number of the available ports. Infrared or some additional ports may be used in that case. Finally, expansion slot would be a beneficial property of PDA because this slot is designed for improving the restricted abilities of PDA. By the help of expansion slot, ROM may be increased or additional connection ports may be added. Finally, requirements are as follows:

- GPS receiver
- Mobile phone with GPRS and Bluetooth connectivity
- PDA that has the followings:
    - Wireless connection (wi-fi 802.11b)
    - Bluetooth connection
    - 300 MHz Processor
    - 56 MB RAM
    - 96 MB ROM
    - Expansion Slot (SD,MMC or CompactFlash)
    - TFT Display

4.1.2. Software Requirements

During the analysis, design, implementation and testing phases of the project, we will use several tools to carry out the project. These tools can be divided into two main groups; first one is documentation tools and the second is development tools.

Documentation Tools

Especially during the first term, we will be using several documentation tools. We have two opportunities for word processing; either Microsoft Office 2003 Word combined with Adobe Acrobat Professional, or OpenOffice 2.0 Writer. We usually use the first one.

We have chosen SmartDraw as the drawing tool because of its ease of use. We have some experiences from the previous terms also. In this phase of the project, we have drawn data flow, use case, entity relationship diagrams using SmartDraw. For drawing Gantt Chart, we preferred using Microsoft Project 2003.

Development Tools

Our development process branches into two parts; mobile device programming and server-side programming.

On the server-side programming, after a long research period, we decided to use Java 2 Platform Enterprise Edition (J2EE). In this critical choice, there are several reasons. First of all, we give great importance to platform independency. Java technologies are platform independent and they are very compatible with almost every Linux distribution, Windows, Solaris and several more platforms. Secondly, we have some experience in J2EE, especially in building Web Services, which can not be underestimated. Thirdly, there are very successful Integrated Development Environments (IDE) for Java. Eclipse is our favorite, however there is an IntelliJ fan in the team also. Since compatibility is provided by Java, we can use both of the IDEs at the same time. As a final reason, J2EE is very popular and at any time we feel we are in doubt, we are sure that we can find help on the internet in several ways: tutorials, forums, developers' web sites, etc.

For the development of Web Services, we will use two Apache projects: Tomcat and Axis. Tomcat is one of the most popular open source application servers and it is platform independent. Platform independency is important for us because we are planning to run our server on a Linux distribution. Axis is a very helpful tool for deploying Web Services easily. Deployment of Web Services is a bit stitching process, so this tool will be very beneficial for us.

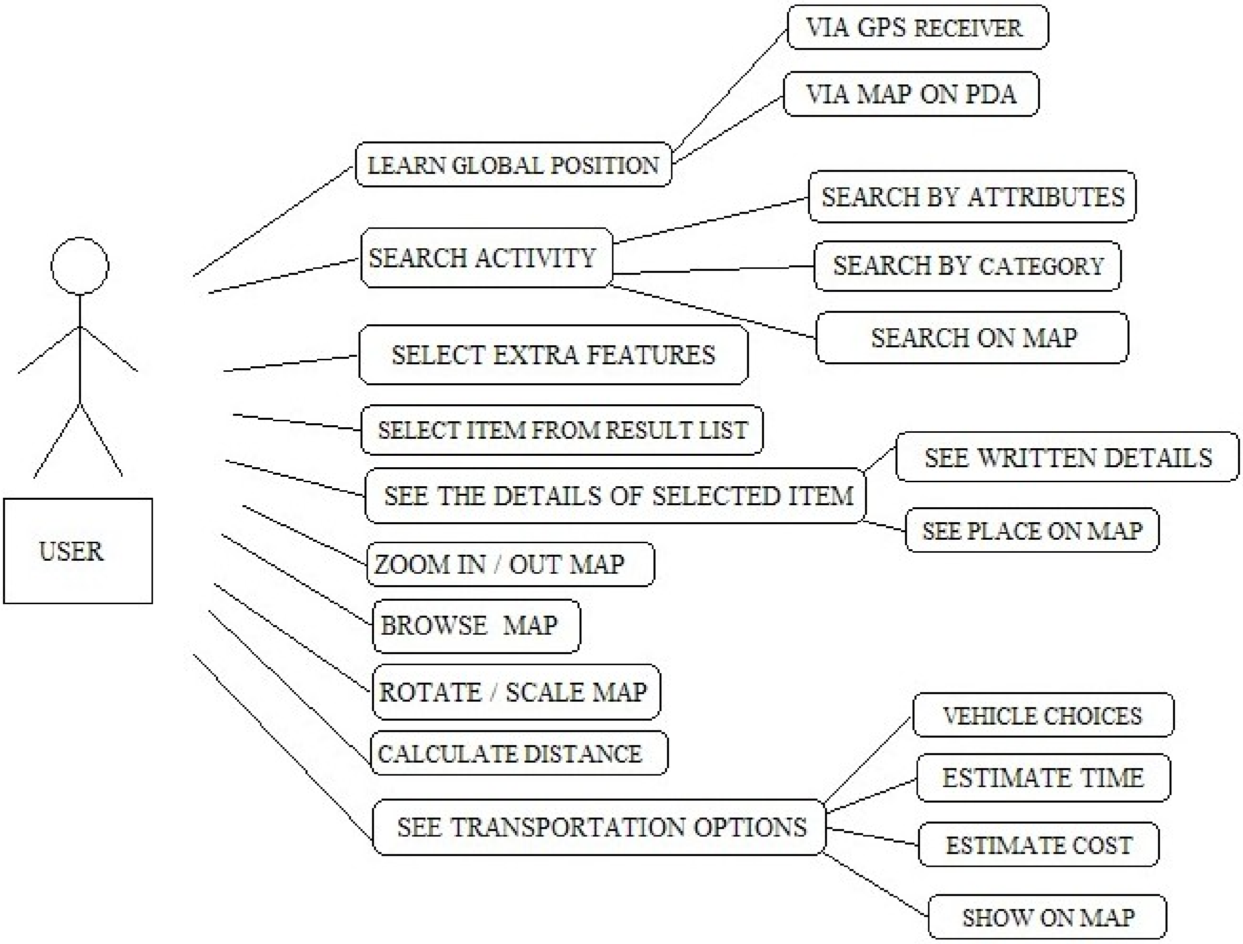
As a DBMS, we are planning to use Oracle 10g at the time. But we have not concluded to a strict decision yet. Most important factor here is the platform independency. So we have some strong alternatives such as MySQL and PostgreSQL.

For the ontology part, we will use Protégé. Protégé is a free, open source ontology editor developed by Stanford University. Protégé is based on Java and supports XML Schema, OWL and RDFs. It just meets our needs, because we are planning to use both Java and OWL.

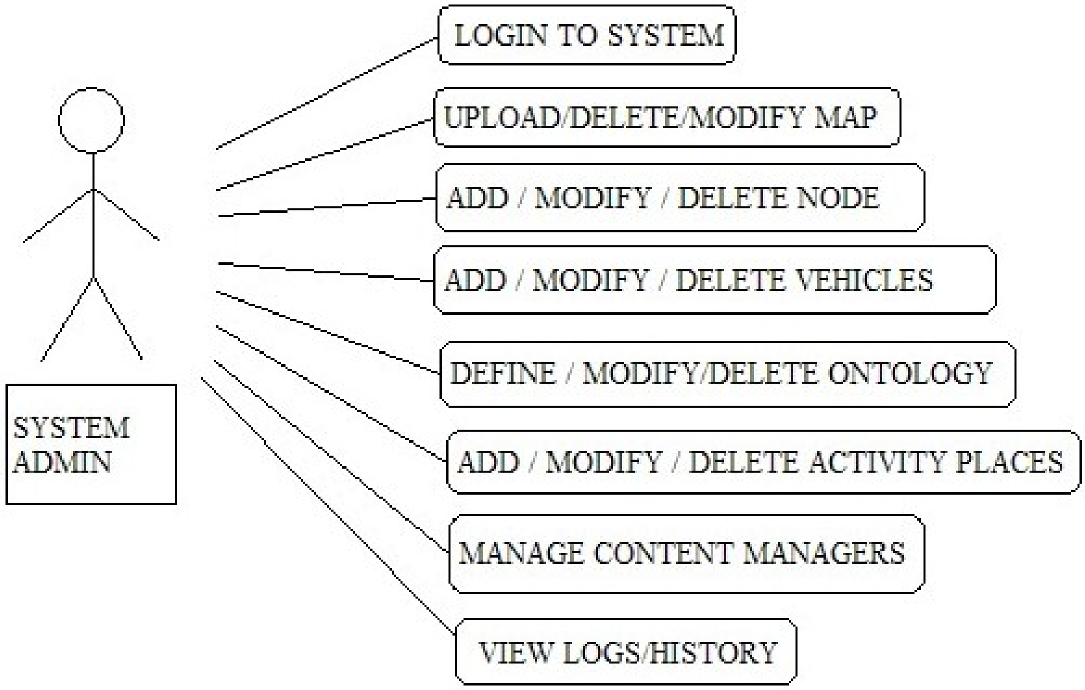
On the mobile device development part, we have decided to use .Net. We have made detailed research before this critical decision. In fact, we intended to use Java solutions for mobile devices also, however there comes the problem of finding an efficient Java Virtual Machine for PDA and a realistic Windows CE PDA simulator. Both of them caused severe problems so we decided to use .Net 2003 (may be .Net 2005, if we find the documentation enough after its release on 7 November 2005). Since we will serve our system via Web Services, we do not expect communication problems between J2EE and .Net.

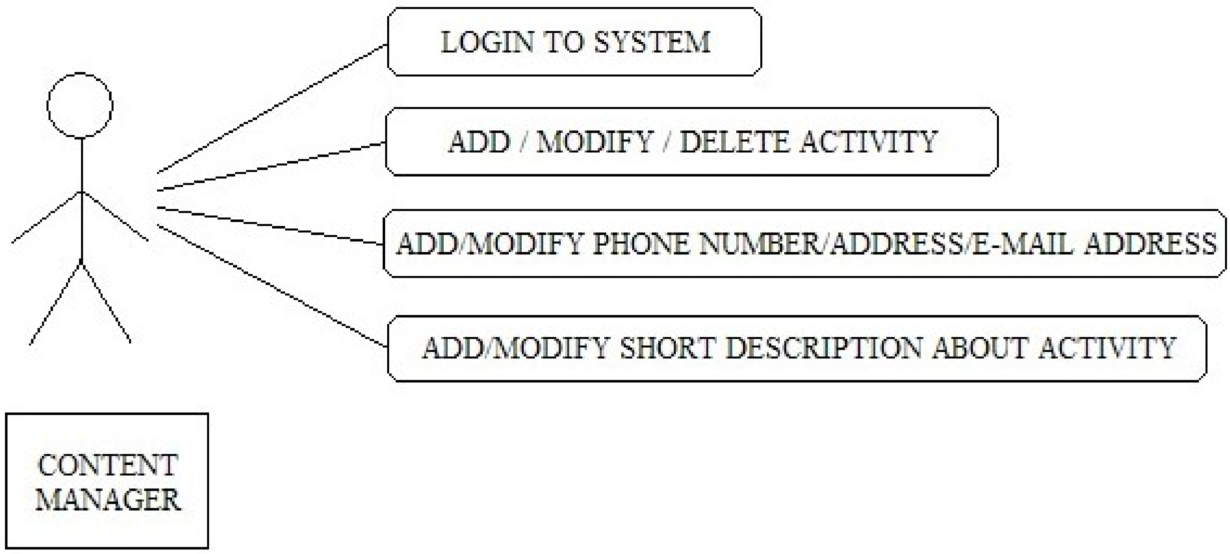
4.2. User Requirements

4.2.1. Use Case Diagrams



Use case diagram for user





Use case diagrams for system administrator and content manager

4.2.2. Functional Requirements

4.2.2.1. User Functionality

We have two different users for "CoreAccess". Main users are PDA users who run application from a PDA connected to GPS and internet. However, we add an extra feature for internet users. They can also reach application via internet from their web browsers. The followings are general functionalities:

- **Select Positioning Method:** User has two opportunities for determining global position. If there is a GPS receiver connected, then global position can be easily retrieved. However, if there is not a GPS receiver connected (web browser is in this case also) or user wants to select his position without GPS, he is able to select his global position from the map. First instance of map that is shown to user will not be detailed one, user will zoom in/out and browse the map as he likes. Then he will determine one final position that our program will assume that is his global position.

- **Search Activity:** User's global position is taken, now the search menu appears to the user. The application area of our project is social activities. These activities are cinema, theatre, music and sport. There are three different search options in this menu. First one is "search by category" option, second is "search by attributes" option and the last one is "search by browsing map" option. The results of all these three options are identical, they will return a result list. This is very important for modular programming. Graphical representation of this situation can be found in State Transition Diagrams.

  The details of these three menus are:

  - **Search by category:** If user wants to do an activity, but no matter the place, he/she can search for the appropriate places. The only necessity is to determine a category name between the cinema, theatre, music and sport categories. In fact, this option is for users who do not have any idea about what to do. For this reason, user may want to see the social activities around him. For instance, if a user in METU campus decides to go to cinema and select "cinema" category in this menu, our application will find the nearest cinemas around him. Results for this example may be first METU cinema (U3), then Bilkent and finally Tüze Armada. Moreover, he/she can view from the map.

  - **Search by attributes:** This option is for users who have at least some idea about what to do. User is able to enter his criterias for the activity he wants to do. This criterias may be;

    - Activity name,
    - Activity place,
    - Activity category,
    - Let activity date between preferred dates.

    User is able to specify none, one or many of these criterias. The number of entered criterias increases the detail level of search. Then, according to entered criterias, search is successfully done and results are shown to user as a list sorted according to smallest distance value.

  - **Search by browsing map:** This option is for users who do not want to use text based menus for searching. In this menu, according to users' global position, local map is shown to user. His position is signed on the map. He can browse, zoom in/out, rotate map and select the activity place. Moreover, if user knows the global position of the place he wants to go, he can write latitude and longitude values. Visible activity places may vary according to zoom level of the map. After deciding on the activity place from the map, following options will be the same as previous search methods.

- **Select extra features:** User can indicate some extra features which are predefined in the application of the place. For instance, user can request nonsmoking places with parking place and playground for his/her children. The extra features are thought as the followings (they can easily be extended in the future):

  - Nonsmoking/smoking
  - Car parking
  - Children playground

- **Select from Result List:** All three search options mentioned above come to this menu. The results are displayed to user as a list sorted from nearest to furthest. There may be more than one result, on the other side search may return empty list if even our ontology definitions fail to find a result. The activities and places are listed with their keywords in this menu. User can select one or more items from this menu. Next, he has two options on the selected items. User can either see the details of the activity and place in written form or see the places on the map. The details of these menus are explained in the following sentences.

- **Display Written Details:** This menu shows the details of the selected activity or place. The written details in this menu include:

  - Activity name,
  - Activity place,
  - Activity date and time,
  - Place address and phone,
  - Transportation options to that place,
  - Distance for the selected transportation method,
  - Traveling explanation for the selected transportation method,
  - Estimated time for the selected transportation method,
  - Estimated cost for the selected transportation method,
  - Extra properties of the place like having parking place, children playground, etc.

  User can easily go back to returned results menu and display another item's details. The transportation methods mentioned here may be "bus", "dolmuş", "metro", "tramway", "taxi" and finally "on foot". We have the path of each transportation method in our database. There may be cases when there is more than one possible way to reach activity place, or there may not be any public transportation vehicles available at that time to desired place and taxi would be the only solution. For all cases, “taxi” is the final option in CoreAccess.

- **Display Items on Map:** In this menu user is able to see his selections on the map. User may select more than one place in the previous selection menu. On the map, his global position and selected activity places will be shown with a placemark. The paths directed from his global position to activity places will also be drawn in different colors. Furthermore, user can make following operations on the map:

    - **Zoom in/out:** Map can be zoomed in / out.

    - **Rotate:** Map can be rotated either clockwise or counter clockwise.

    - **Browse Map:** User does not have to stick to the result map. He is able to browse the map by going upwards, downwards, left and right.

    - **Display Distance:** As in the previous menu, the distance of the paths may be shown on the map. This is a good option for comparing the distances of the paths. With the help of this property, user can manage his time efficiently.

    - **Time/Cost Estimation:** In the same manner as previous “Display Distance” option, estimated time and cost values may be shown on the map also. This property will increase the time and cost efficiency of the user’s choice.

User can easily go back to returned results menu and display another item’s details on map.

4.2.2.2. Content Manager Functionality

The duty of content manager is to change the contents of their company’s social activities. For instance, the content manager of a cinema can add the new films which are on screen. Content manager may be responsible of managing several places’ activities. The followings are the capabilities of the content manager’s functions:

- **Login:** A content manager has to login to the system first with his id and password. A content manager may have the ability to modify only one place’s activities or a number of places’ activities.

- **Add/Modify/Delete Activity:** If new activities are available, content manager has the responsibility to add new activities and move the past activities to history. There may be cases when content manager needs to delete or modify the activity. The properties of the activities need to be entered are:

    - Activity Name
    - Activity Time
    - Activity Place (hall)
    - Activity Cost
    - Activity Type

- **Add/Modify phone number/address/e-mail address:** Content manager can add or modify those important attributes of the activity place.

- **Add/Modify short description about the activity:** Content manager can add brief information about the activity. Thus, users can have some idea before attending any activity.

4.2.2.3. System Administrator Functionality

System Administrator has the highest level rights. He is able to do anything that user and content manager can do. Apart from those rights, he is responsible for uploading map data, ontology information, determining the relation between instances in the ontology. Functionalities of system administrator are explained below extensively:

- **Login:** In order to accomplish main activities, system administrator has to login to the system for security. This feature enables the protection of database contents of the application. Only system administrator has the right to modify and add the contents of the ontology and map data.

- **Upload/Delete/Modify Map:** In this functionality, system administrator can upload map, delete map, modify map. Some new areas that were not in the coverage area of CoreAccess can be added to extend the coverage area. In the same manner, some areas may be discarded or modified.

- **Add/Modify/Delete Node:** Nodes are very important in CoreAccess. All maps are processed as connected nodes. Activity places are special nodes. Apart from activity places, there are lots of nodes to describe roads. Vehicles' paths are constructed by series of connected nodes. As a result, its system administrator's responsibility to manage all nodes.

- **Add/Modify/Delete Vehicle:** System administrator can add, modify or delete a vehicle. As explained in the previous part, vehicles' paths are defined by connected nodes. System administrator can change the path of the vehicle by adding new nodes or removing existing nodes. Moreover, system administrator can manage the type (which may be one of "bus", "dolmuş", "metro", "tramway" or "taxi"), time_divider and cost_multiplier of the vehicle. time_divider and cost_multiplier are vehicle specific properties. These allow the system to estimate cost and duration of a path for selected transportation method.

- **Define/Modify/Delete Ontology:** System administrator can define different ontologies for activities and transportation methods. In the following topic, we will explain what ontology means for CoreAccess in detail.

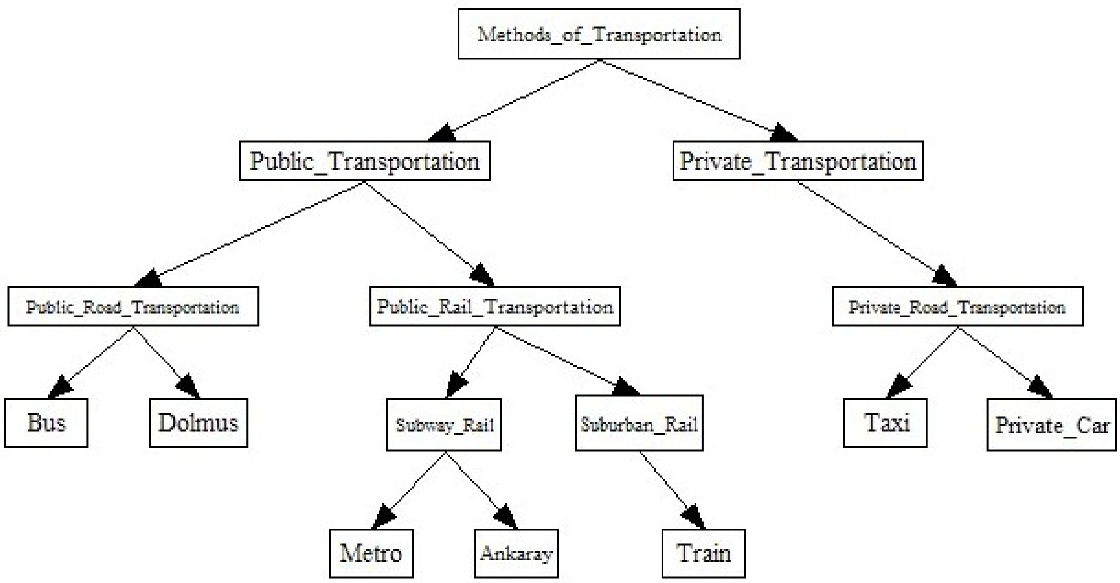
Details of Ontology Structure of CoreAccess:

CoreAccess will benefit from ontology in order to increase the number and improve the quality of results returned back to users. Description of ontology: *"Ontology as a branch of philosophy is the science of what is, of the kinds and structures of the objects, properties and relations in every area of reality"* [1]. Ontology tries to find the classification of entities, and model the sample space as close as possible to real world. In fact, ontology is just like the inheritance (multiple inheritance included) concept of object oriented designs. The relations between objects are supplied with inheritance (base classes, subclasses).

Most popular standards for constructing and sharing ontologies are Resource Description Framework (RDF) and Web Ontology Language (OWL). Both are XML based, so it can be said that they are human readable, however this is not the case. We have found a powerful ontology editor, namely Protégé. We have given brief information about it in the Software Requirements section.

There will be a semantic matching engine in CoreAccess and its duty will be to interpret defined ontologies and according to predefined rules, it will make decisions. By this way, CoreAccess will serve users alternative solutions and it will be hard to return a NULL list to users.

In CoreAccess, we are planning to integrate ontology to transportation, cinemas and theatres. Ontology for Transportation Systems itself is a huge research area and there are lots of papers and applications dealt with this subject. Main idea is to donate the user with exactly what he requests whenever possible, and serve some alternative transportation solutions when an exact match can not be found. Moreover, to inform the user, alternates can be given when there is a situation of exact match. As we have declared before, there will be several transportation methods (vehicles) in CoreAccess. To draw the picture in our minds, an example transportation ontology can be something like the following:



An example transportation ontology for Ankara

For cinemas and theatres, what we plan to do is less complicated. We will try to define editable ontology rules based on the subjects of movies or plays. These subjects may be “love, comedy, drama, action, love-comedy, documentary, science-fiction, etc.”. There will be more detailed subjects and ontology rule will be defined according to those. For example, if user searches for a “love-comedy” and there is no love-comedy movie on the scene, then user will have a chance to see “love” films and “comedy” films on the result list, because there did not occur an exact match and these are the strongest candidates that user may want to see. In this case, “love-comedy” is subclass of both “love” and “comedy”, an example of multiple inheritance. This alternate finding property will be done by semantic matching engine of CoreAccess.

Our research continues in ontology topic, we can add some new functionalities in the near future according to our research.

- **Add/Modify/Delete Activity Places:** Content manager can manage activities but its system administrator’s responsibility to manage activity places. The attributes that have to be filled are name, address, phone number, e-mail address, Global Position and Node of the activity place.

- **Manage Content Managers:** Its system administrator’s responsibility to add new content managers or delete old ones. System administrator gives user id and password to content managers and defines their abilities. Therefore, content managers have limited right to access and modify the database elements.

- **View Logs/History:** One of the most usable properties of CoreAccess is logging. CoreAccess logs all of the user activities. When user makes a search or requests details of any activity/place, CoreAccess stores this information to database. The attributes of the logs are: user ip, global position of the user, time and date, search details, requested activity’s details, etc. System administrator is able to see and print these logs any time. By this way, statistical data will be taken and managers and companies will be informed about the usage statistics.

4.2.3. Nonfunctional Requirements

4.2.3.1. Usability:

The usability of our product is very important for us. We are planning to develop a product to be user friendly for every one. Because, our product is too much social content oriented and users of the system may not have improved computer usage capabilities. The interfaces of our product on PDA will be very clear and understandable. Since the PDAs, on which we will implement our program, run on Windows, we do not think the users will encounter any problem while using it.

4.2.3.2. Portability:

We can truly say that our product will be portable, because during the development process we will always use platform independent tools. For instance, J2EE, TomCat and Axis are all open source platform independent technologies. Thus, server side of our program will work on any Linux distribution, Windows and all other operating systems that support these.

The only point that portability seems not possible is PDA side. Because, we have a PDA that runs Windows CE. However, with the help of platform independency of web service, it will not take much time for us to adapt our program to another platform. For example, we can design a user interface for Palm OS (in fact, there is no more Palm OS, thanks goes to Bill Gates!) that can communicate with our web service after a few weeks' work.

4.2.3.3. Reliability:

We are planning to develop our product so stable such that any minor problem will not cease the program. Moreover, we are going to do lots of tests after implementation to minimize the bugs on the program. We are planning to setup Bugzilla in order to report and trace the bugs effectively.

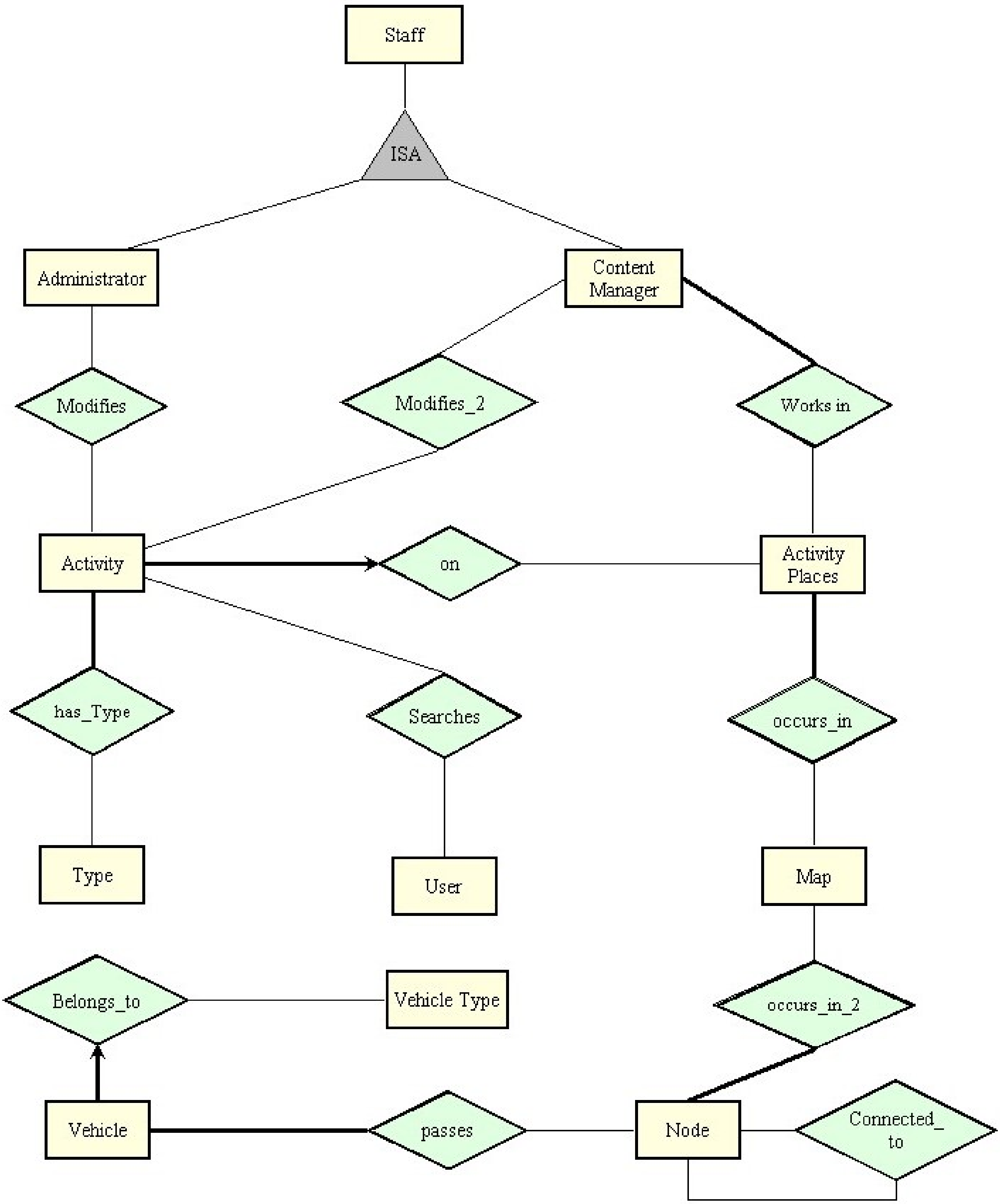
4.2.3.4. Security:

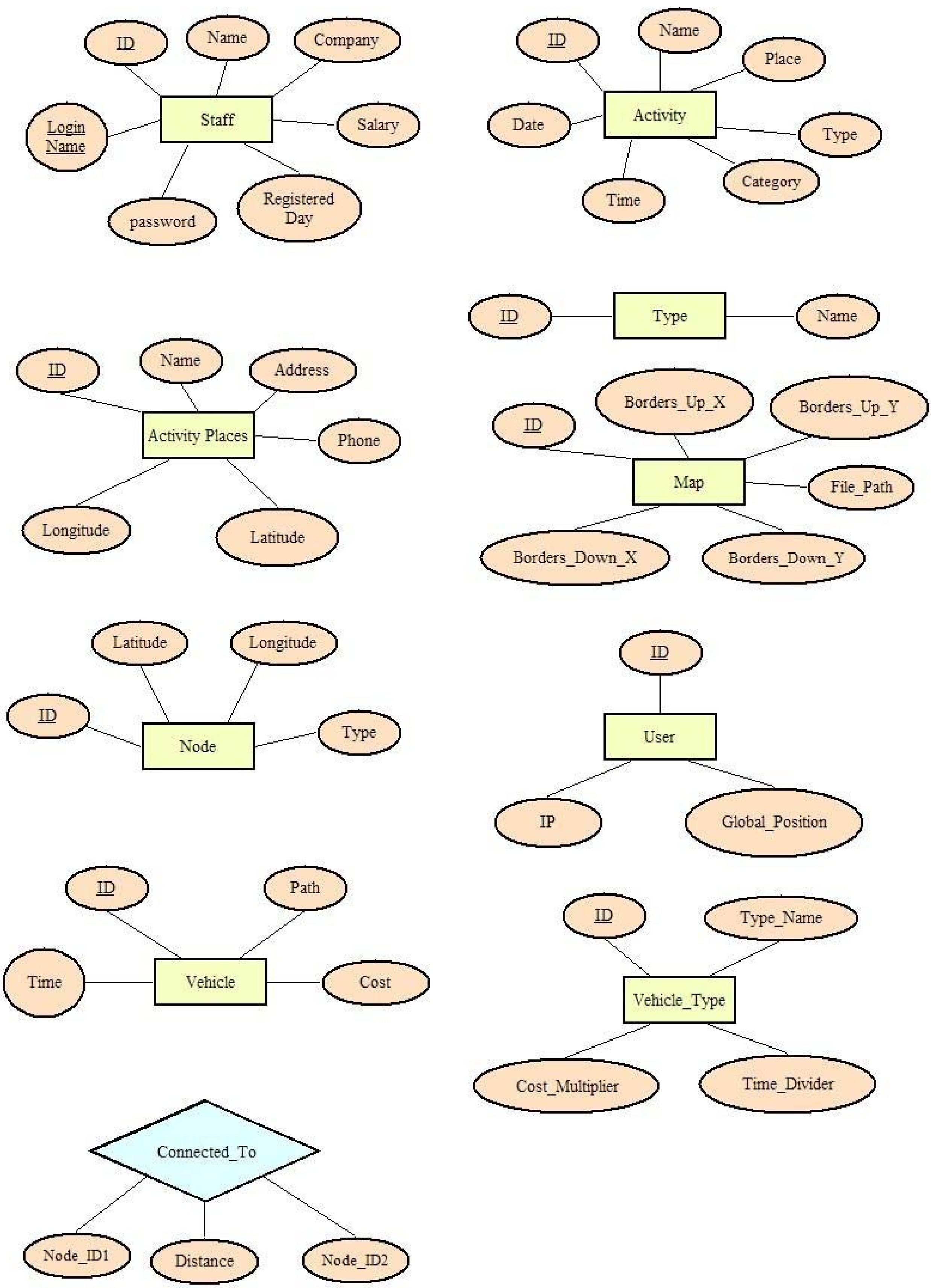
There are permissions to do critical accesses and jobs in CoreAccess. Only system administrators are allowed to do system wide processes and content managers can manage the data that they are allowed to do. Users are only users and they do not have any permission to do any changes to database. They can only use preference menus available in PDA side of CoreAccess.

5. MODELLING

5.1. Data Model

Entity – Relationship Diagram

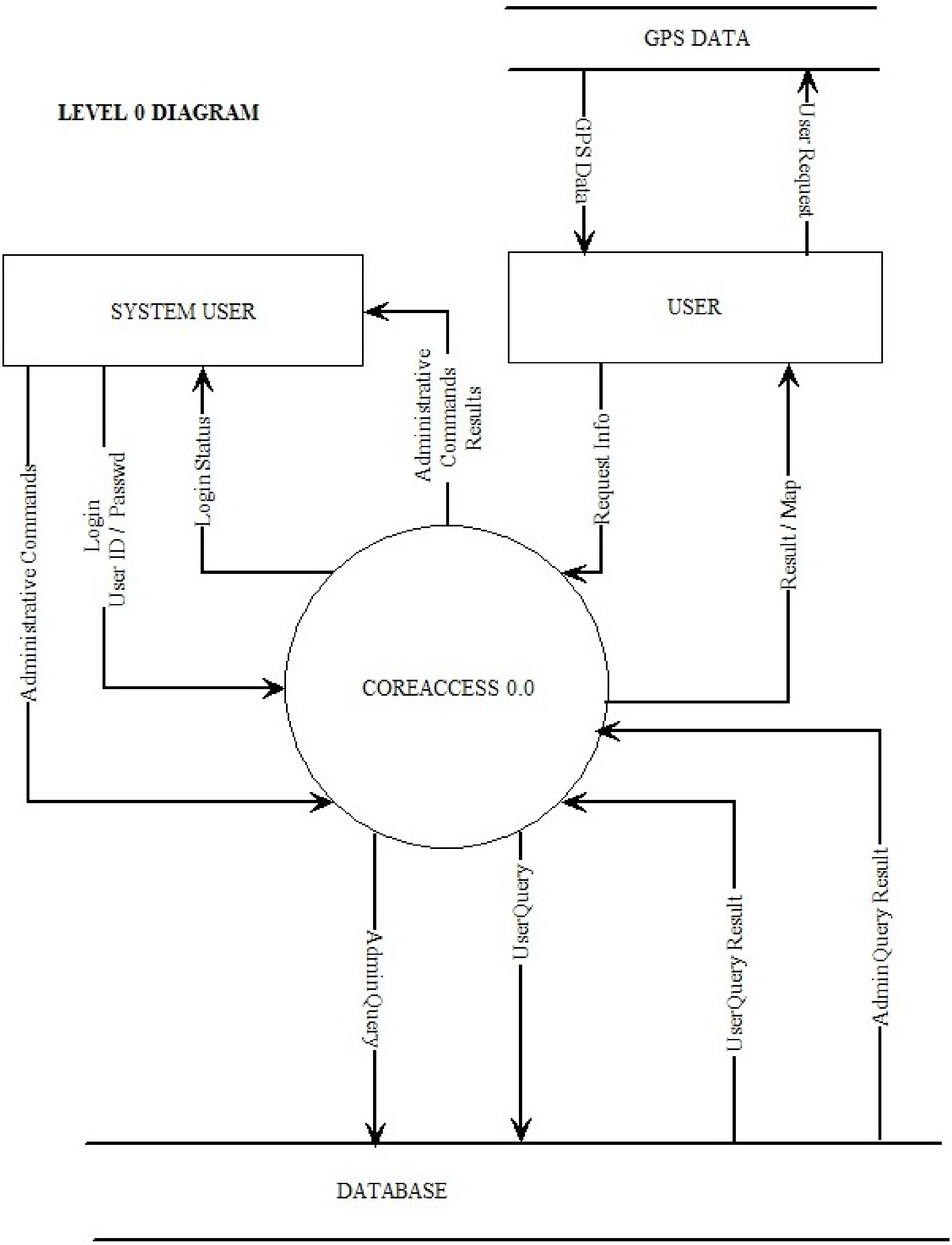



ID
Name
Company
Login Name
Staff
Salary
password
Registered Day
ID
Name
Place
Date
Activity
Type
Category
Time
ID
Type
Name
ID
Name
Address
Activity Places
Phone
Longitude
Latitude
Borders_Up_X
Borders_Up_Y
ID
Map
File_Path
Borders_Down_X
Borders_Down_Y
Latitude
Longitude
ID
Node
Type
ID
User
IP
Global_Position
ID
Path
Time
Vehicle
Cost
ID
Type_Name
Vehicle_Type
Cost_Multiplier
Time_Divider
Connected_To
Node_ID1
Distance
Node_ID2

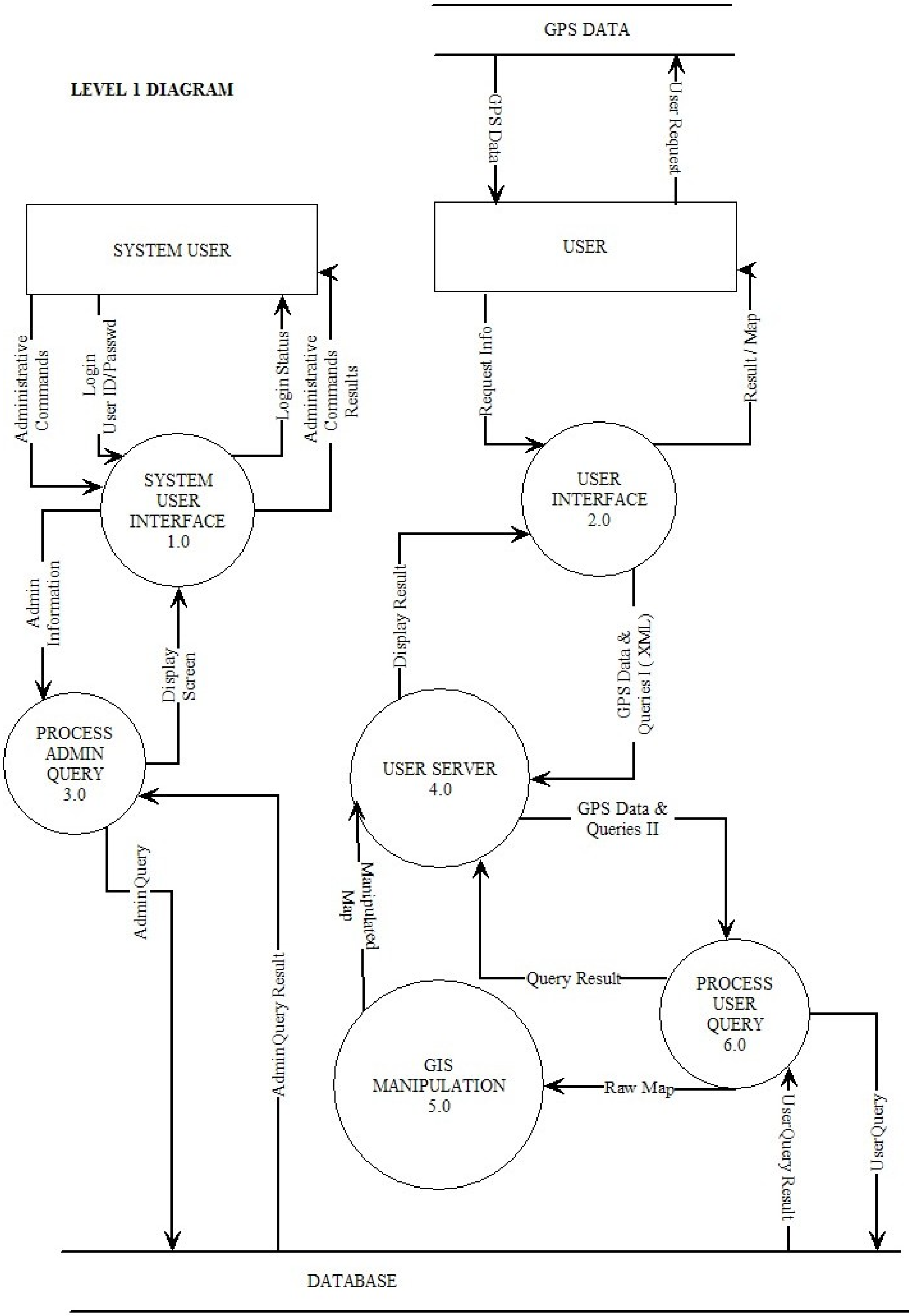
5.2. Functional Model

5.2.1. Data Flow Diagram

5.2.1.1. DFD Level0



5.2.1.2. DFD Level1



5.2.2. Data Dictionary

Name:	GPS Data
Alias:	User Coordinates
Where & How it is used:	USER *input*
Description:	" sends the GPS data (coordinates) to PDA user"

Name:	User Request
Alias:	GPS Device Signal
Where & How it is used:	USER *output*
Description:	" sends the request signal to GPS data from GPS device on User"

Name:	Request Info
Alias:	User demand
Where & How it is used:	USER *output* USER INTERFACE (2.0) *input*
Description:	" written request to the PDA by User "

Name:	Result /Map
Alias:	Visible result
Where & How it is used:	USER *input* USER INTERFACE (2.0) *output*
Description:	" appeared map and information for the user on the PDA "

Name:	Administrative commands
Alias:	System commands
Where & How it is used:	SYSTEM USER *output* SYSTEM USER INTERFACE (1.0) *input*
Description:	“written commands to interface by system user”

Name:	Login UserID/Password
Alias:	Submission info
Where & How it is used:	SYSTEM USER *output* SYSTEM USER INTERFACE (1.0) *input*
Description:	“ written information for accessing the system which controls the database”

Name:	Login Status
Alias:	User check
Where & How it is used:	SYSTEM USER *input* SYSTEM USER INTERFACE (1.0) *output*
Description:	“information about validity of the user ”

Name:	Administrative Commands Results
Alias:	System Returned Result
Where & How it is used:	SYSTEM USER *input* SYSTEM USER INTERFACE (1.0) *output*
Description:	“returned information to the System user whether the changes are done or not ”

Name:	Admin Information
Alias:	System Data
Where & How it is used:	PROCESS ADMIN QUERY (3.0) *input* SYSTEM USER INTERFACE (1.0) *output*
Description:	"gathered information from user interface which is written to the Browser by system user"

Name:	Display Screen
Alias:	Visible result
Where & How it is used:	PROCESS ADMIN QUERY (3.0) *output* SYSTEM USER INTERFACE (1.0) *input*
Description:	"generated result screen to the system user"

Name:	Display Result
Alias:	Result screen
Where & How it is used:	USER SERVER (4.0) *output* USER INTERFACE (1.0) *input*
Description:	" generated result screen to the user about his/her query "

Name:	GPS Data & Queries I(XML)
Alias:	Written data
Where & How it is used:	USER SERVER (4.0) *input* USER INTERFACE (1.0) *output*
Description:	"Gathered info from user interface"

Name:	Admin Query
Alias:	Generated query
Where & How it is used:	PROCESS ADMIN QUERY(3.0) *output*
Description:	“sends admin query to the database”

Name:	Admin query result
Alias:	Generated result
Where & How it is used:	PROCESS ADMIN QUERY(3.0) *input*
Description:	“recieves result of qurydrom database”

Name:	Manipulated Map
Alias:	Processed map data
Where & How it is used:	USER SERVER (4.0) *input* GIS MANIPULATION(5.0) *output*
Description:	“Processed map from raw state”

Name:	Raw map
Alias:	Initial map
Where & How it is used:	GIS MANIPULATION(5.0) *input* PROCESS USER QUERY(6.0) *output*
Description:	“the map after query execution , not processed”

Name:	Query Result
Alias:	Returned result
Where & How it is used:	USER SERVER (4.0) *input* PROCESS USER QUERY (6.0) *output*
Description:	“result after query execution”

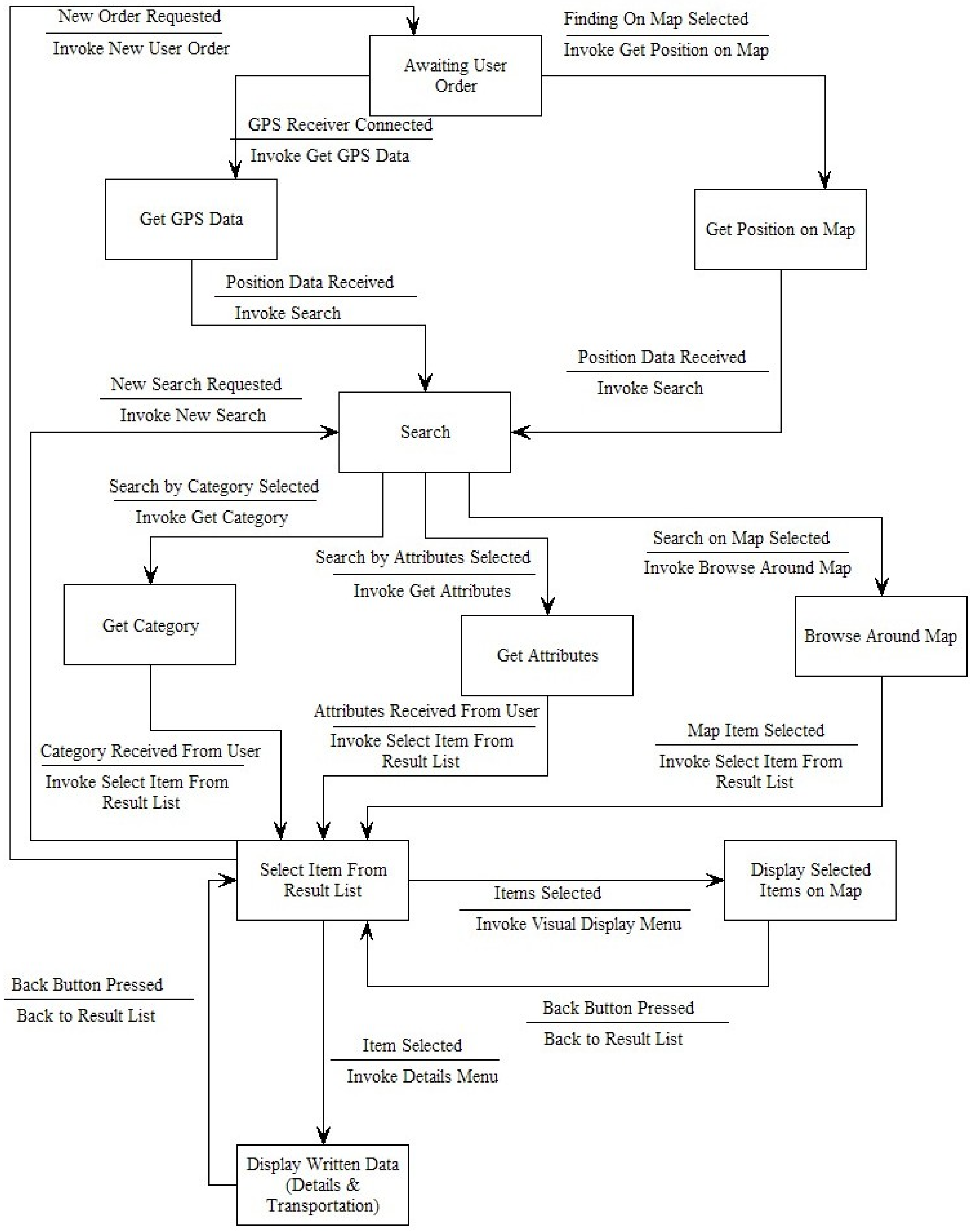
Name:	GPS Data & Queries II
Alias:	Whole request data
Where & How it is used:	USER SERVER (4.0) *output* PROCESS USER QUERY(6.0) *output*
Description:	“sent user request from server in XML format ”

Name:	UserQuery Result
Alias:	Terminal result
Where & How it is used:	PROCESS USER QUERY(6.0) *input*
Description:	“returned result from database to the server”

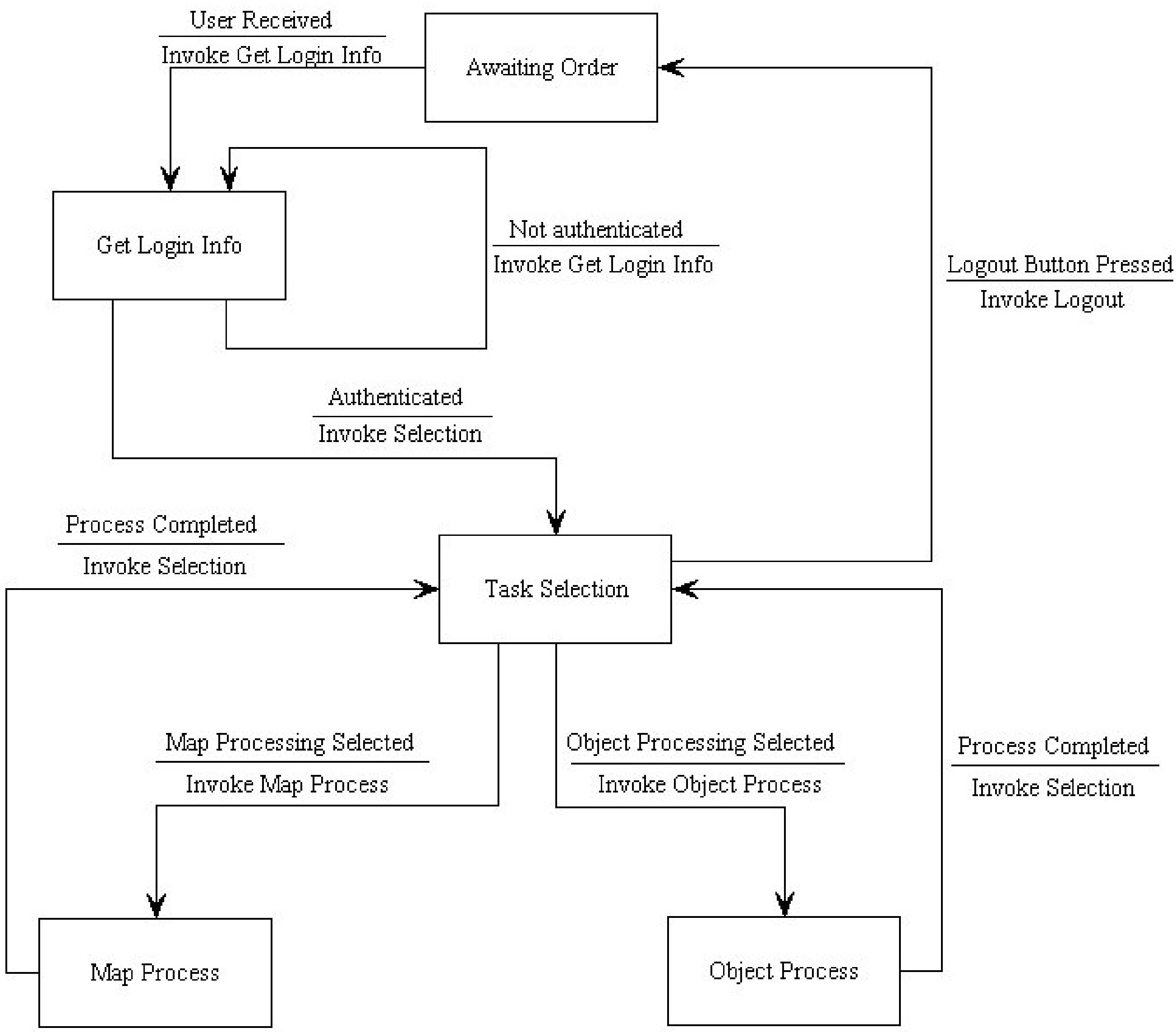
Name:	UserQuery
Alias:	Packed info
Where & How it is used:	PROCESS USER QUERY(6.0) *output*
Description:	“packed user request information, sent to the database ”

5.3. Behavioral Model

5.3.1. State Transition Diagram of User



5.3.2. State Transition Diagram of Administrator



6. MARKET RESEARCH

6.1. Literature Survey

6.1.1. Introduction

We have made a complete survey on the software that has been produced so far. In order to gain background knowledge, to ensure that the project is realizable, to infer useful ideas from both successful and unsuccessful projects and to broaden our imaginations, the market research was inevitable.

In the market, since there are not enough projects which are exactly related to our project proposal topic, we did market research on similar internet mobile GIS projects and on a project having similar infrastructure to our proposal. We divided the applications as Location Based Services and Vehicle Tracking System. Vehicle tracking system is a bit far from our project scope; however we had to see some real mobile GIS applications with qualified visual data. Vehicle Tracking System examples will not be discussed in this report, those are accessible in our “Market Research” report.

As Local Based Services, we further divided the group into two categories: Applications for tourism purposes and applications for social activities. WebPark and MacauMap are tourism oriented mobile GIS applications where Orange Guide, Turkcell NeNerede Services and Location Based Semantic Web Services Application for Mobile Environments from Ege University Computer Engineering Department are examples of social activity guides.

We constructed a table showing the properties of the Location Based Services software we have observed. Moreover, we added one more column for our project CoreAccess’ features.

The names in the tables refer to:

- WebPark: WebPark Project,
- MacauMap: MacauMap, Tourism Oriented Mobile GIS Application,
- Orange: Orange Switzerland LBS Project – Orange Guide,
- Ege: Location Based Semantic Web Services Application for Mobile Environments – Ege University Computer Engineering Department,
- NeNerede: Turkcell NeNerede Service,
- CoreAccess: Our project CoreAccess.

6.1.2. Comparison Table

Properties	**WebPark**	**MacauMap**	**Orange**	**Ege**	**NeNerede**	**CoreAccess**
Positioning method:	GPS	GPS	Cell ID	GPS	Cell ID	GPS
Project scope:	Tourism	Tourism	Social	Social	Social	Social
Mobile device type:	PDA	PDA	Mobile Phone	Mobile Phone	Mobile Phone	PDA
Mobile device software:	Windows CE	PalmOS	NA	Symbian	NA	Windows CE
Mobile device – server communication method:	Internet (wireless)	Internet (wireless)	WAP	Internet (GPRS)	SMS	Internet (wireless, else GPRS)
Multilanguage support	√	√	√			√
Supplying map data	√	√	√			√
Showing path to desired location on the map	√	√	√			√
Address description	√	√	√			√
Map navigation (zooming, panning)	√	√				√
Suggesting transportation methods	√	√	√			√
Guiding / giving information about the activity/place	√	√				√
Being accessible from anywhere via web services						√
Producing alternative results (via ontology)		√		√		√
Estimating time for_transportation methods	√		√			√
Supplying traffic situation			√			
Bookmarking/ writing comments	√					
Statistical Data			√			√

The items mentioned in the above table are usually general properties of the software. Most of them are expected from a well qualified LBS (Location Based Services) software. Among these features are, global positioning, efficient mobile device – server connection, supporting the result with clear visual data (vector maps), showing the path to desired location on the map.

Most admirable properties of all these are; address description for the path to desired location, supplying all transportation alternatives (metro, bus, tramway, taxi, etc.) and finally canalizing the user to less crowded places by reaching traffic situation. The first two are the properties that we have already written in the first draft of our proposal. But the last one is not possible without coming to an agreement with municipalities.

6.1.3. Results

We have gained valuable knowledge about our project area during the research period. We have seen what kind of applications is present in the market. While making research about products, we found some extreme information too. For example, we used to know that there is no problem with the coverage area of GPS, however we have learned that GPS can not retrieve the position information in closed areas (we mean indoor). In order to overcome this obstacle, there is another option: assisted GPS.

We have seen that there are tons of mobile GIS applications in the market, some of them also benefit from GPS technology in order to get position information. Most popular application in the market is vehicle tracing system as far as we have seen, but this topic is a bit far away from our project. Tourist guides are also popular; these applications aim to assist tourists in touristic places and we can conclude that they are closer to our project, in sake of the used devices and technologies such as GPS, mobile devices (PDA or mobile phones), internet (GPRS or wireless).

As a result of our literature survey, we could not find an exact matching of our project topic. Tourist guides that we have discussed are very successful examples of LBS software in the market, their structure is almost the same with our project. Only the application area is different, our product will serve to a more general segment of users. Apart from the difference in POIs (Points of interests), WebPark and MacauMap are suitable for being our reference projects.

In the social aspect, most successful and wide application is the Orange Guide; it encapsulates almost all of our topics. If only it had used GPS instead of Cell ID and it had some improved visual properties, Orange Guide would easily be our reference project. Turkcell NeNerede is close to Orange Guide but it lacks some major properties like visual data and transportation. Finally in this section, Ege University Computer Engineering Department arrives on the scene and does his job successfully with semantic matching engine and GPS. It lacks map data but developers hope to add this function soon. We believe we could gain valuable experience by getting in contact with them.

To conclude, our research will not stop here and we will always avoid bad properties of the applications in the market. While we are benefiting from nice and worthy properties, we will cover the lacks of our reference projects. We made our first step to get into contact with developers of our reference projects. For getting more information, you should see detailed “Market Research” document.

6.2. Interviews

6.2.1. Interview with Veysi İşler

We have talked with Veysi İşler about our project. He is also interested in GIS applications. He asked us which tool we are going to use in manipulating map data. However, we do not have a proper map data. He suggested some GIS companies which develop a GIS tool. He also recommended us to visit their website. This company is InterGraph and its tool is IntelliWire. We had a short interview with Veysi İşler. He said that he would try to make a contact with that company in the following days. He will inform us about the result.

6.2.2. Interview with NetCad

We sent an e-mail requesting an appointment within the week to NetCad in order to discuss whether they can assist us. We got a positive response on Wednesday which states that we can have a meeting on Friday afternoon. We had an interview with NetCad as a professional company. Here firstly a brief description of the company and some important projects of the company is given. Then, what we talked during interview is presented.

6.2.2.1. NetCad Company

NetCad Company has been established in 1987. Today, the company employs approximately 70 employees and works in software development, project management, technical support, and professional advisory. One of the main branches they provide in order to meet the needs of public and private sectors are GIS applications. They have an application, named NetCad, which has similar properties with our project. It is a web server on which GIS applications run. However, they do not have mobile device connected to server. Although mobile device is not connected to the server, they have an independent application named Pocket NetCad which runs on PDA. Pocket Netcad is one of the modules that runs on NetCad. On this PDA, a user can search on map and see the coordinates of the pointed location. Moreover, this system can work offline once the related map is downloaded. Below are some important application areas of this mobile solution.

- *Vehicle Tracking System:* With this module, the vehicles can be tracked and controlled on raster or vector maps.
- *Service Mobile Application:* This is again a kind of vehicle tracking system which provides the parents of the students with information about the position of the students through SMS.

6.2.2.2. Interview Results

The person we talked with was the technical services manager, Okan Sabancılar. At the beginning, he wanted us to describe our project. Our project description in the e-mail was not so detailed. At first sight, he thought us to develop an application only on the mobile device (PDA) and to use an already existing application for the server. We explained that we were going to implement both server side and mobile device side. After everything is clear, we asked our questions to him. The main points of the meeting are as follows:

- They showed us their Pocket NetCad. In the current implementation of the Pocket NetCad, a vector map is shown and detailed information of the pointed nodes is illustrated. For instance, x and y coordinates, name and address of the point are displayed.

- We asked them what kind of map they are using. The answer was both vector and raster map. Moreover, they support OpenGIS standards.

- Mr. Sabancılar demonstrated us a web application on the server. It is written with .Net. Some of the municipalities such as Söke and Bahçeşehir use this server. In the service application, directly the name of the searched item is entered and the places are returned as a list. When an item is chosen, it is shown on the map.

- It was also mentioned that they developed a system for cemetery in Kahramanmaraş. It works on a kiosk which displays the tombs of people to relative of them. The path of the tomb is illustrated and a print out can be taken. It is very user friendly interface, because this system addresses to everyone.

- We asked Mr. Sabancılar how they manipulate map data and which data type they are using. He said that he did not have enough detailed information about it. He promised us that he would arrange an appointment with software engineers working there.

Since we saw some real applications on both server and PDA, this meeting was beneficial for us. However, we could not get enough technical information since NetCad is a big company and it has different software development departments. Mr. Sabancılar wanted us to send a detailed e-mail explaining what information and support we request from their company. According to our e-mail, he will arrange an appointment with the related software developers in order us to be able to get satisfying technical information.

6.2.3. Interview with Potential Users

We made a small questionnaire with 15 students in dormitories and department of computer engineering. There are the questions and our commands about the given answers in the following:

Q1: Do you have any knowledge about GIS applications?
Comment: Generally most of them have no idea about GIS application. However, some of students in computer engineering department know about it and state that they do not see a real application.

Q2: How can you learn your position on the world if you are lost?
Comment: Most of them trust their mobile phones, but they feel desperate if the phone is out of service area.

Q3: What functionalities do you use if you are given a tool which gives you the exact coordinates of the objects on earth?
Comment: The most general answer was that they use that tool for finding their direction. Moreover, they search for learning any settlement area or roads.

Q4: Our group will develop an application on social activities. Is it difficult for you to learn about social activities?
Comment: Most of them use internet for searching social activity. However, they complain that the internet is not always with them.

Q5: If you are outside, how can you learn about a social activity or a place?
Comment: The participants state that they learn via mobile phone or ask someone else.

Q6: In our project, user can learn what social activities exist around him/her and how to go there via PDA. How would that help you?
Comment: Most of them are glad to hear that application will be available. They say that they can take advantage of this application in finding activities anywhere in anytime.

Q7: We are planning to inform the user with this application about the cinemas, sport and music event and also transportation information. If you are given such a PDA providing these properties, what would you expect?
Comment: They want some ease in transportation. Most of them desire to choose most suitable routine. Moreover, some participants want to see the public day of cinema.

7. SCENARIO

"CoreAccess" is completely a social project for human beings. In order to explain well, we wrote a sample scenario for this application.

The wheather is sunny and fine. Suppose, our user Olduz is outside and wants to go to a football game which is played between Fenerbahçe and Galatasaray. He only knows that it is in Ankara and today. However he does not know the exact place and time of the play. He desires to go that game very much. Thanks to CoreAccess application, he can learn the needed information from his PDA in which CoreAccess application is installed. He can search the activity place and time. He begins to write necessary information for search on PDA interface. He selects the search for activity name choice. Then a different screen appears. In that screen he select the sports activities option. A new screen is available for specific activity search. He writes the event name to the specified area and press the search button. If the written activity is in the database, the result is returned. The result contains the address, time, date, and name of the activity. He is happy with the result. Moreover, he can learn more information about the activity. The transportation choices are displayed in a map and the distance between him and the stadium. One of them is to go there with a taxi, the other is to go by bus and the other choice is to go there by a dolmuş. In the first choice it is illustrated that it costs 10 YTL and it takes only 20 minutes. The second choice costs 0.8 YTL and it takes 1 hour to arrive. The third choice costs 1.2 YTL and it takes 40 minutes. He looks at the time and it is not necessary to be hurry. Therefore he decides to go there with the bus.

After the enjoyable play, a heavy rain starts. However, Olduz does not want to go home. An idea appears in his mind. He has time to go to the cinema. But he does not know which film is worth to watch. Again CoreAccess helps him. He begins to use his PDA. This time he selects the search for activity type. He chooses the cinema activity. He fills the necessary fields on the PDA. He especially wants to go action film. Finally, so many results return with the name and place. He can learn the information about the movies by pressing on any of them. He decides the film and place. Then he views the map of transportation choices. He selects the appropriate one and he tracks that way. If Olduz had searched the film according to the name, ontology could have helped him if there is no cinema putting that film on the screen. Ontology shows all the action films.

As a result, CoreAccess provide user convenience in searching for social facilities. It does not matter where the user is. With the GPS data he can find the nearest activity to him. He can save time and money with CoreAccess.

8. CONCLUSION

Preparation of this report was a very beneficial process for our team. We did valuable brainstorming and made some points more clear. While discussing the details of CoreAccess, we also built some interfaces for the design report. We believe this analysis report will be helpful both for us and for the ones who are following the project.

9. REFERENCES

[1] Smith, B., 1999. *Ontology: Philosophical and Computational*. Unpublished manuscript, http://wings.buffalo.edu/philosophy/faculty/smith/articles/ontologies.htm