

Preliminary Design

Requested by: Dr. Timoth Lederman
Professor of Computer Science
Siena College
Computer Science Department

oldGMCTrucks Mapper



Prepared by: Dustin Clark – Web Master
Devin Hurley – Lead Developer
Joshua Tomaszewski – Database Admin.
Chan Tran – Data Analyst
Zachary Witter – Team Leader

October 25, 2013

**oldGMCTrucks
Mapper
Preliminary Design**

1. Product Overview and Summary	5
2. User Case Narratives.....	5
2.1 non-oldGMCTrucks Forum Member.....	5
2.2 oldGMCTrucks Forum Member	5
2.4 System Administrator	6
3. UML Use Case Diagram.....	7
3.1 UML Use Case Diagram Legend	7
3.2 Use Case Diagram.....	8
4. UML Deployment Diagram	9
5. UML Activity Diagram	10
5.1 Activity Diagram Legend	10
5.2 Login Activity Diagram	11
5.3 Map Activity Diagram	12
5.4 Profile Activity Diagram	13
5.5 Request Account Activity Diagram	14
6. Website Map	15
6.1 Website Map Legend.....	15
6.2 Website Map Homepage	16
6.3 Website Map Forum Member Homepage.....	16
6.4 Website Map Member Homepage	17
6.5 Website Map Administrator Homepage	17
7. Dataflow Diagrams	18
7.1 Legend.....	18
7.2.1 Context Diagram.....	19
7.2.2 Level 0 Diagram.....	20
7.2.3 Level 1 - Confirm Account	21
7.2.4 Level 2 - Confirm Account - Email User.....	22
7.2.5 Level 2 – Confirm Account - Add User to Database	23
7.2.6 Level 2 – Confirm Account - Create Mapper Account.....	24
7.2.7 Level 1 - Log in	25
7.2.8 Level 1 - Modify Profile.....	26
7.2.9 Level 2 - Modify Profile - Create Profile	27
7.2.10 Level 2 - Modify Profile – Edit Profile.....	28
7.2.11 Level 2 - Modify Profile – Modify Shoutout.....	29
7.2.12 Level 1 - Modify Event.....	30
7.2.13 Level 2 – Modify Event - Create Event.....	31
7.2.14 Level 2 – Modify Event - Edit Event.....	32
7.2.15 Level 1 - Interact with Map	33
7.2.17 Level 1 - Request Account.....	34

7.2.18 Level 1 - Manage Map Data 35

8. Structure Diagram..... 36

8.1 Structure Diagrams..... 36

8.2.1 Context Diagram..... 36

8.2.2 Expansion of Login Module..... 36

8.2.3 Expansion of Modules Off of the Login Module 37

8.2.4 Expansion of Map Member Module 38

8.2.5 Expansion of Modify Own Profile Module 38

8.2.6 Expansion of Modify Event Module..... 39

8.2.7 Expansion of Directions Module..... 39

8.2.8 Expansion of Search Module 40

8.2.9 Expansion of System Administrators Module..... 40

8.2.10 Expansion of Confirm Account Request Module 41

9. Functional Requirements Inventory..... 42

9.1 General User Interface Functional Requirements 42

9.2 Random Actor 42

9.3 oldGMCTrucks Forum Member Actor 42

9.4 oldGMCTrucks Mapper Member Actor..... 43

9.5 oldGMCTrucks Mapper administrator 43

10. Non-Functional Requirements Inventory 43

11. Logical Data Dictionary 44

12. Prototypes 45

12.1 Admin Home page 45

12.2 Member Home Page..... 46

12.3 Profile Login Tab 47

12.4 Event Prototype 48

12.5 Create Event Prototype 49

12.6 Search Members 50

12.7 Search Members Options..... 51

12.8 Plan a Trip..... 52

12.9 Edit Shout-out Prototype..... 53

12.10 Edit Shout-out Privacy Prototype 54

13. Testing Plan 55

13.1 Overview & Strategy..... 55

13.2 Acceptance Test 55

13.3 Unit Tests 55

13.4 Test Cases..... 55

13.5 Exception Handling 55

14. Development and Production Environments 56

14.1	Development Environment.....	56
14.2	Operating Environment	57
14.3	Maintenance Environment.....	57
15.	Appendix A: Cross Reference Index.....	57
16.	Appendix B: Sources of Information	57
17.	Appendix C: Glossary of Terms.....	58
18.	Appendix D: Timeline (Gantt Chart).....	59

1. Product Overview and Summary

For many people who take joy and pride in their classic automobiles, it is difficult to find others to enjoy their cruising experience with. People with an interest in a more specific vehicle may even have a harder time finding people that understand their passion. Dr. Timoth Lederman, an active participant and administrator of an antique GMC truck forum, has expressed the interest in an easy to use web based mapping system for finding people with common interests in antique GMC trucks. The purpose of this system is to provide an intuitive map based interface that allows for GMC truck owners to arrange meetings, gatherings, and cruises with fellow forum members. Such a system would assist in the organization of social events, provide for cruising route suggestions, and even insure them that there would be someone around with the tools to help if they were to break down.

2. User Case Narratives

2.1 non-oldGMCTrucks Forum Member

This is a user who comes to our system from a link outside of the oldGMCTrucks.com forum. This user must create an account on the oldGMCTrucks.com site in order to be able to register for an account on the oldGMCTrucks Mapper system. Without an account on oldGMCTrucks.com, this user will be unable to access the map's information.

2.2 oldGMCTrucks Forum Member

The oldGMCTrucks Forum Member is a user who can create an account on our system. This is a user who has an account on the oldGMCTrucks.com forum site. The reason for this prerequisite is to prevent anyone from viewing the personal information stored on the map. This forum member will have access to information that was related to the link they clicked as well as information available on the map. They do not yet have a profile on the system and therefore cannot utilize all of the functionality of the oldGMCTrucks Mapper system.

2.3 oldGMCTrucks Mapper Member

This is a user who is a member of our system, the oldGMCTrucks Mapper. When this user reaches the site, they will be able to login if their cookie has expired, or they will be able to view all the information on the map if their cookie is still valid. The oldGMCTrucks Mapper Member will be able to perform maintenance on their own profile. This includes managing what information is visible in the user's "shout-out". The shout-out is a pin on a map which, when clicked, displays a little summary of that user's (or an event's) associated information. oldGMCTrucks Mapper Members also have the ability to create an event. Any oldGMCTrucks Mapper Member can edit any event on a map for the purpose of fixing incorrect information or updating information. oldGMCTrucks Mapper Members can also plan a trip using the oldGMCTrucks Mapper system.

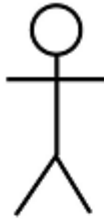
2.4 System Administrator

The System Administrator (admin) is responsible for the inspection of user data entered into the system as well as for maintaining users' data.

The admin is able to perform all the actions an oldGMCTrucks Mapper Member is able to. This includes editing his or her personal profile, editing event information and planning a trip. The admin also has privileges that an oldGMCTrucks Mapper Member does not. These include the ability to access all member information. The admin can modify any user's personal information, including information displayed in the shout-out.

3. UML Use Case Diagram

3.1 UML Use Case Diagram Legend



An actor is represented as a stick figure that is an internal or external entity that interacts with the system. The actor(s) of a system describes the internal and external entities.



The association of two objects is represented by a line. This describes the relationship and interaction between an actor and its use case.



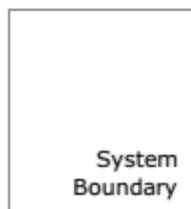
The includes association is represented by an arrow labeled <<Includes>>. This association describes that the base use case is dependent upon the included use case and cannot stand alone.



The extends association is represented by an open arrow labeled <<Extends>>. This association is dependant upon the base use case and adds another step to the use case.



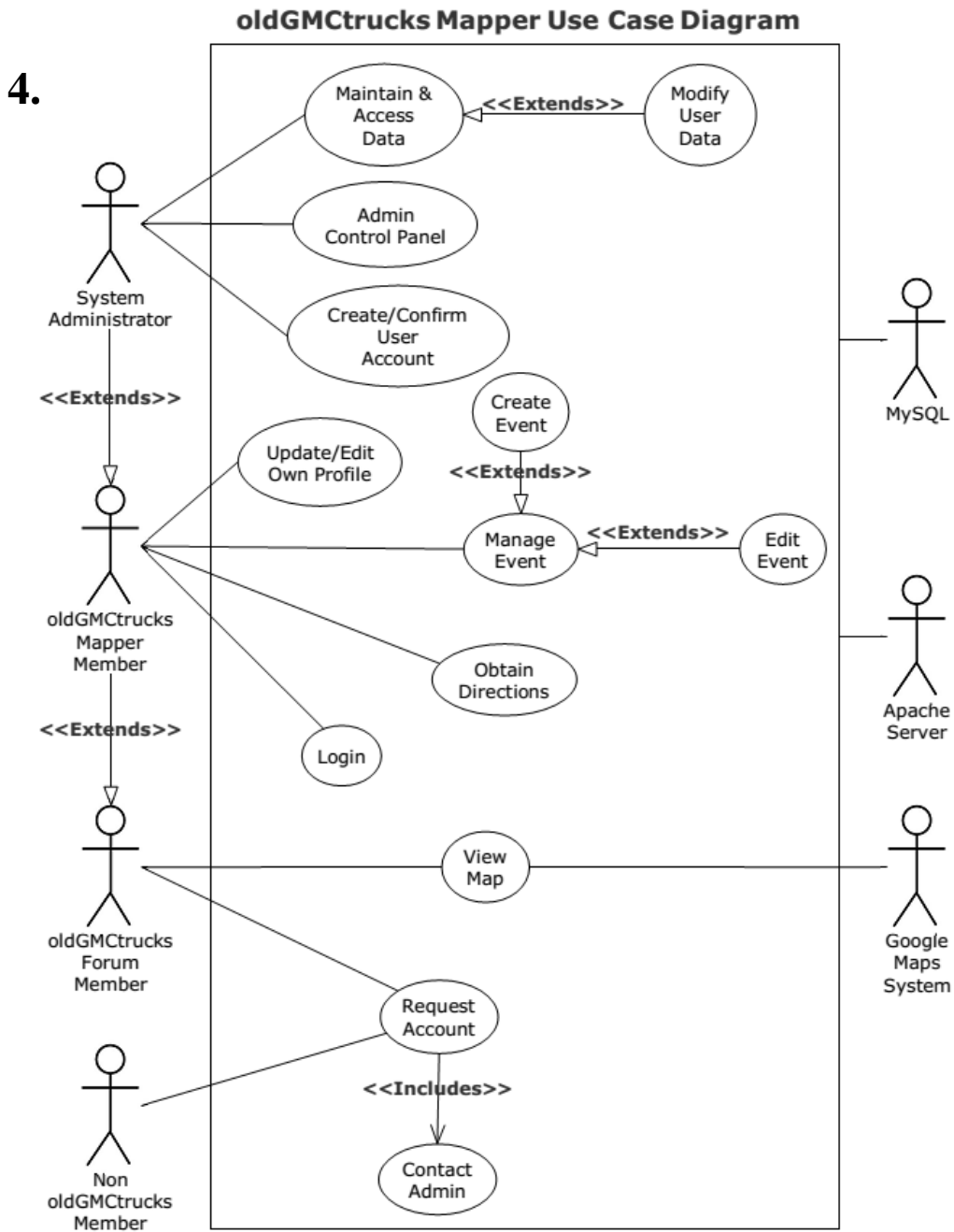
A use case is represented with an oval shaped object. There is often a name inside of the oval, giving the name of that function.



A system boundary is represented by a large rectangle, encapsulating the use cases inside. The actors remain outside of the system boundary but can interact with the use cases inside.

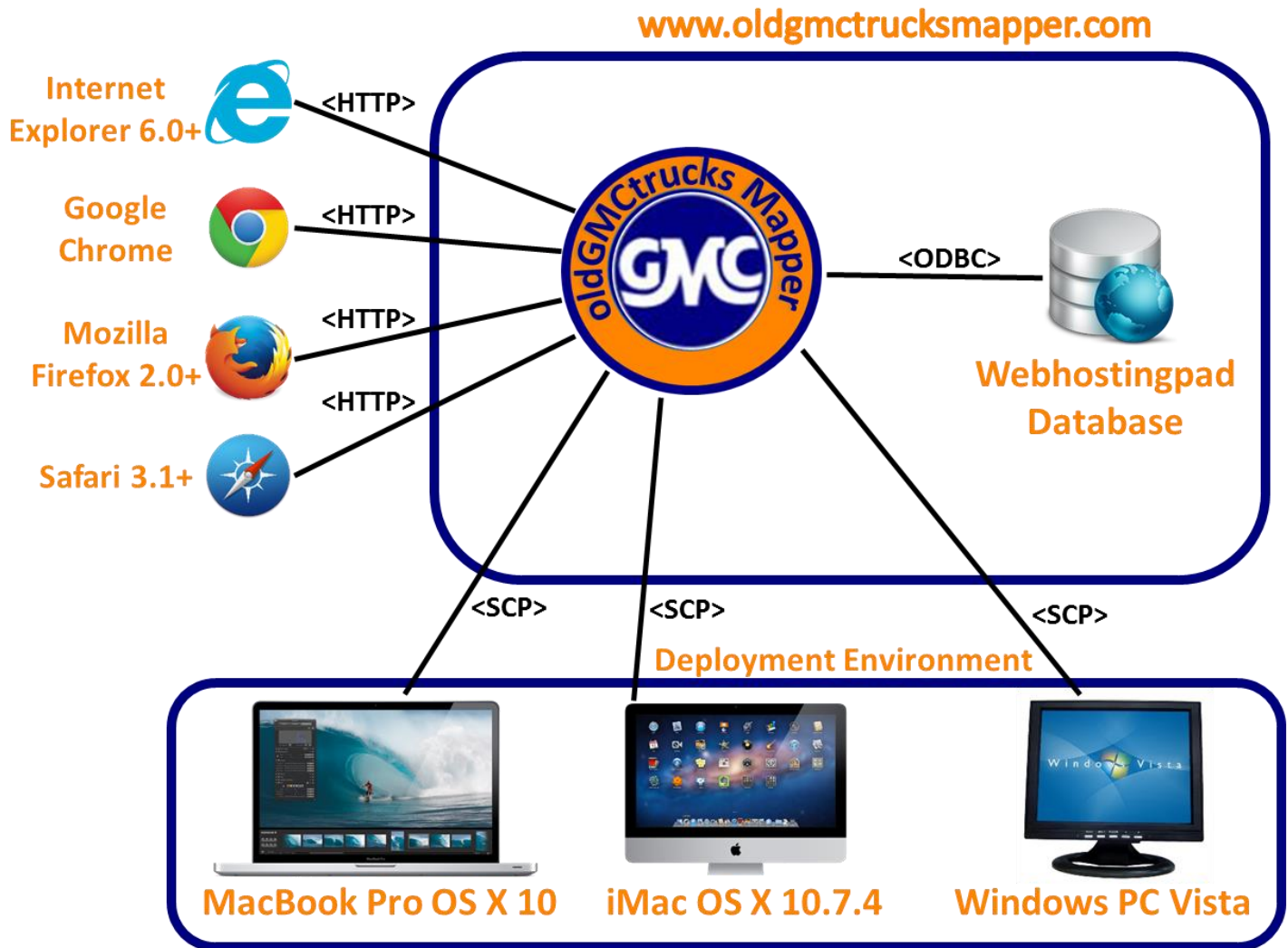
3.2 Use Case Diagram

A UML Use case Diagram represents how users interact with our system. Users (actors) interact with use cases, which are different functions of our system. Other actors interact with the system boundary, which means they interact with our entire system as a whole.



4. UML Deployment Diagram

A UML Deployment Diagram models the execution architecture of systems by representing each system as a node. Each node is connected by communication paths to develop a network of systems. The following diagram is a deployment diagram for oldGMTrucks Mapper.



5. UML Activity Diagram

A UML Activity Diagram graphically represents the flow and order of actions and activities for processes.

5.1 Activity Diagram Legend



Initial Node

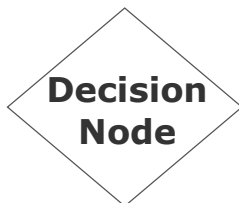
The initial node is a control node which indicates the start of the activity. It is possible to have more than one initial node.



An activity specifies the executions of behaviors. These behaviors can flow to and from other behaviors or to and from decision nodes.



The control flow is represented by an arrow. The control flow shows the relationship and flow of one activity to the next or between an activity and a decision node.



A decision node accepts an incoming edge and presents multiple outgoing edges based on the decision of that process. The decision will be based off of the text that is given by the node.

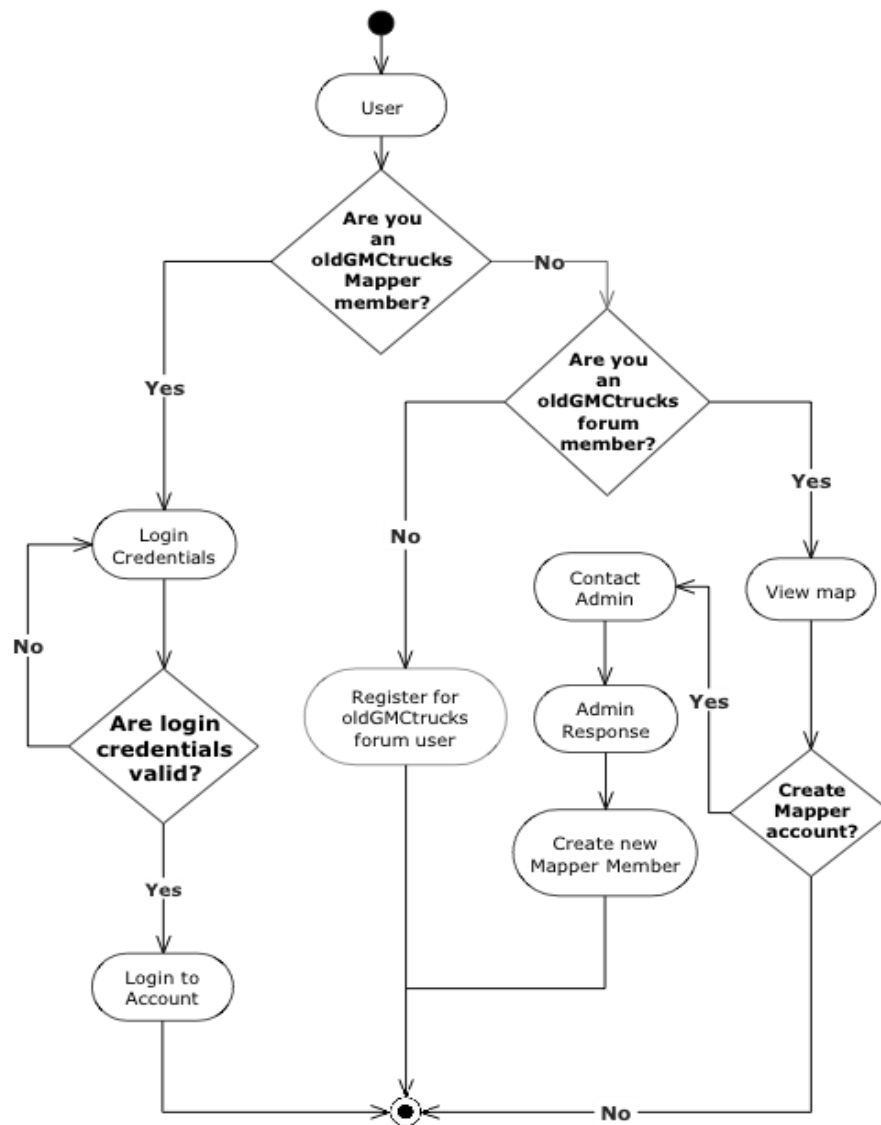


Final Node

The final node indicates an end to the activity diagram. All processes of that activity will end at the final node. There may be more than one final node.

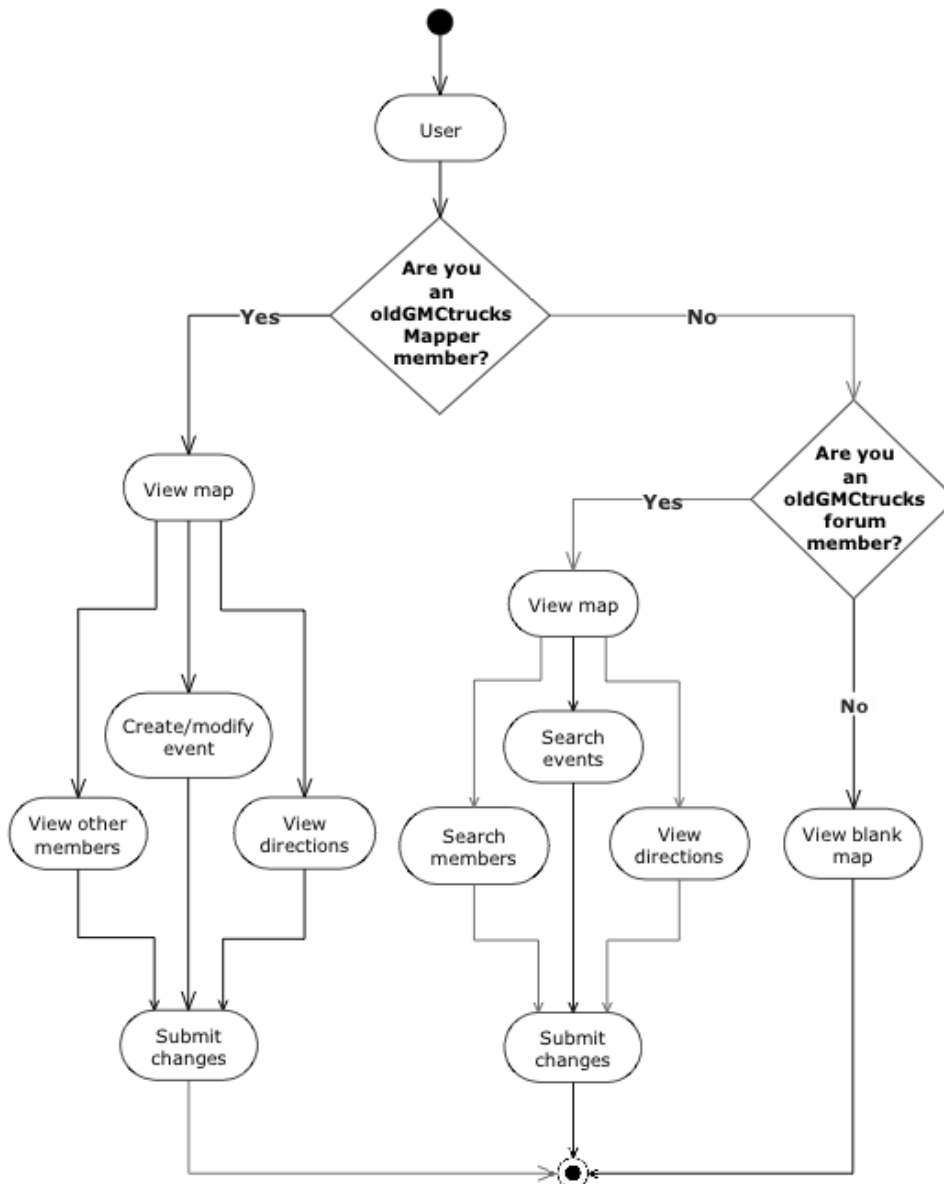
5.2 Login Activity Diagram

In this activity, the user will be prompted with a login menu. The user has the ability to input their login credentials but will not be able to log into the oldGMTrucks Mapper website if their credentials are not valid. If the user is not an oldGMTrucks Mapper member but is a member of the forum, they are able to view the map but will not have any functionality. The forum member will also have the ability to create a Mapper account if they so choose to. If the user is a random user, they will be prompted to register for an oldGMTrucks forum account.



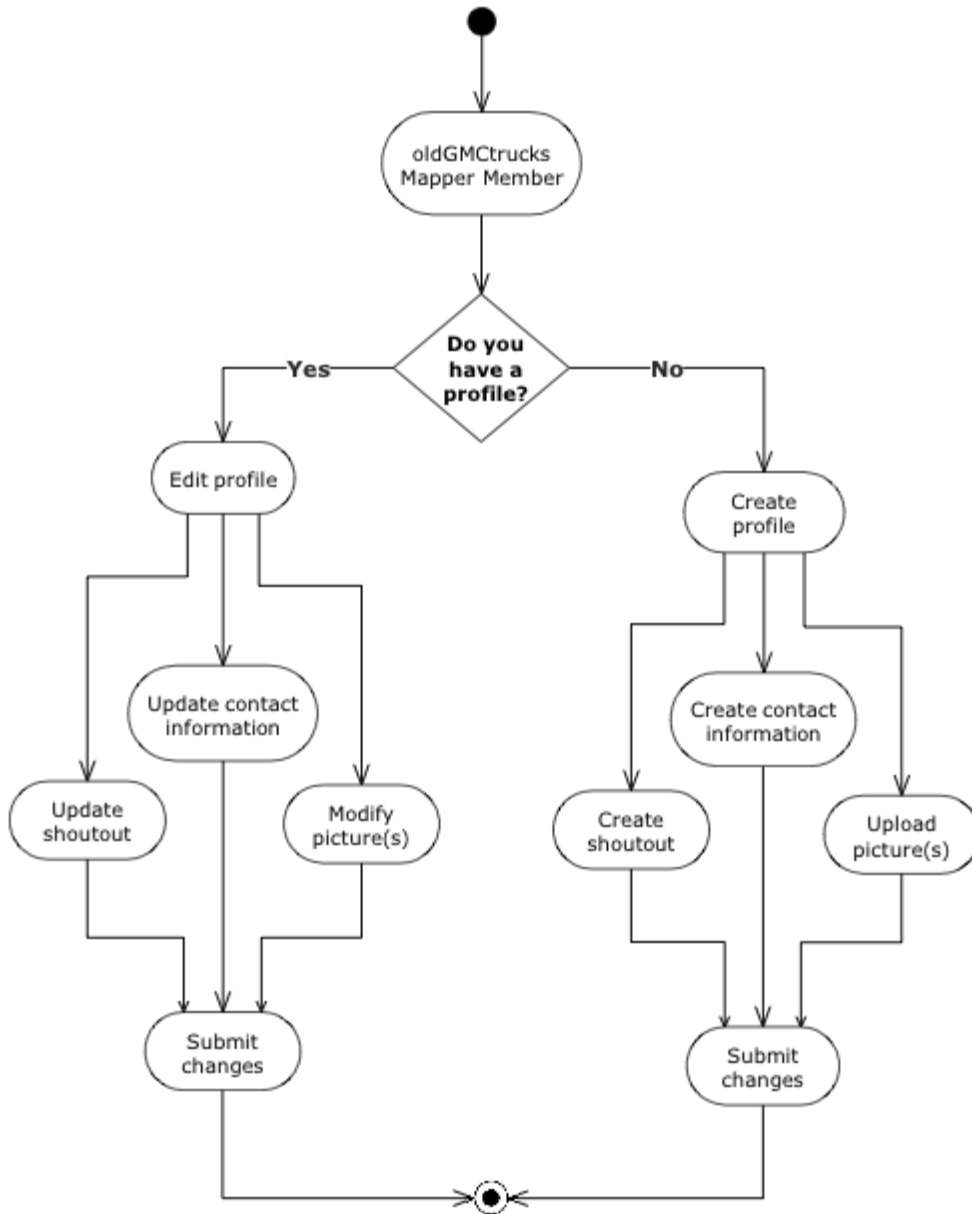
5.3 Map Activity Diagram

The map activity shows how the user will be able to interact with the map. If they have an oldGMTrucks Mapper account, they will have full functionality of the map and submit any changes that they would so choose. If the user is a forum member but does not have a Mapper account, they can only view the map as well as its contents. Any other random user will be able to view only a blank map with no contents or information.



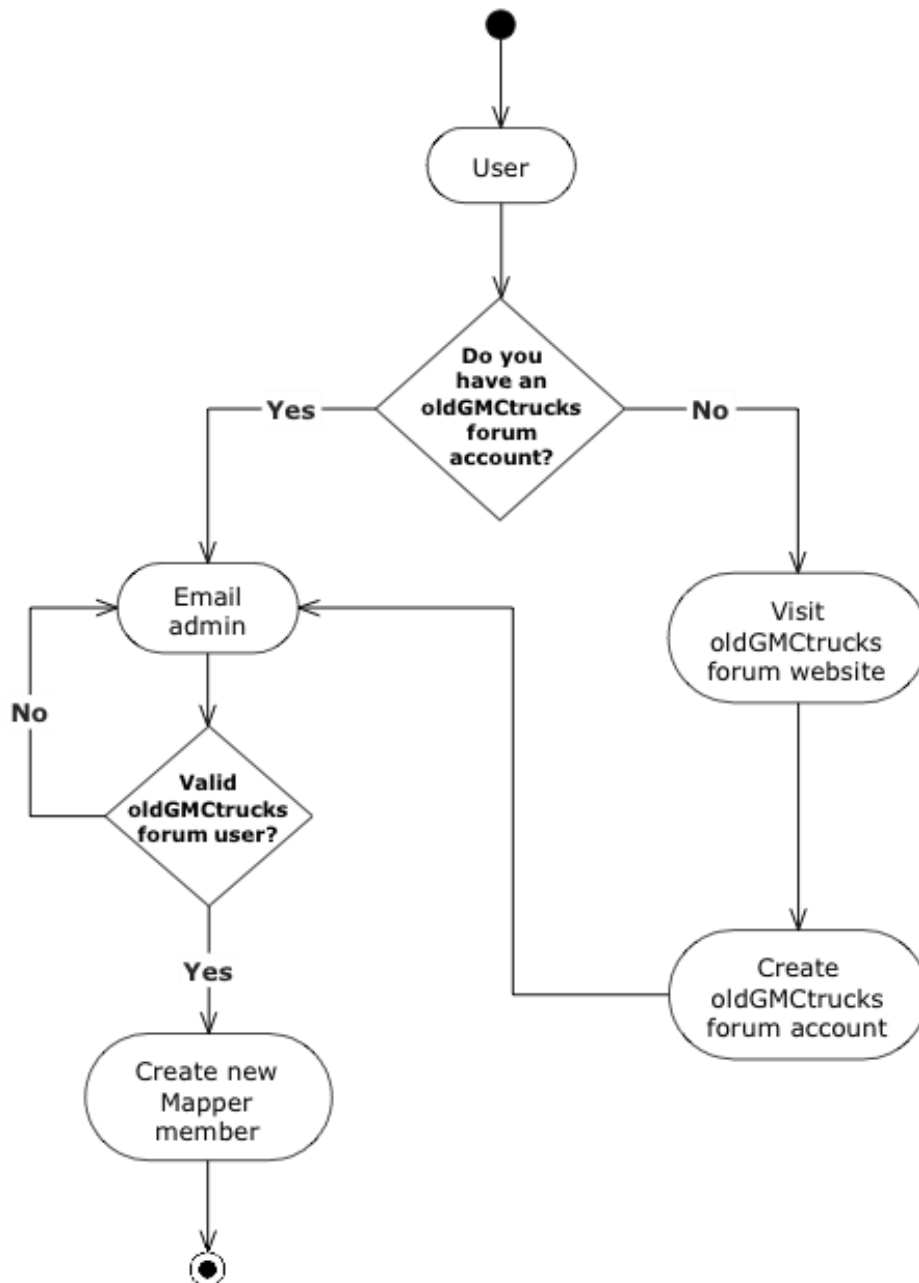
5.4 Profile Activity Diagram

In the profile activity, the oldGMCTrucks Mapper member has the ability to create or edit their own profile. If there is an existing profile, the member can modify any information that they so choose. If a profile has not been created yet, the Mapper member may add any information to their profile as they desire.



5.5 Request Account Activity Diagram

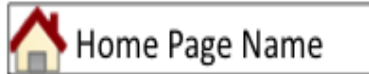
The request account activity will only apply to non-oldGMTrucks Mapper members. Existing forum users may choose whether or not they would like to create a Mapper account by emailing the administrator. A confirmation will be needed by the administrator in order to create a valid Mapper account. Other random users will have to create a forum account before requesting an account from the administrator.



6. Website Map

A website map is a graphical view showing the links between pages on a website.

6.1 Website Map Legend



Home Page

Home Page: The first page a user comes to when interacting with a webpage.



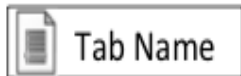
Web Page

Web Page Name: The name of the webpage the user is currently on.



Forum

Forum Name: Shows the name of a forum that has fields to be filled out by a user.



Tab

Tab Name: Shows the name of the tab visible on the current webpage.



Link

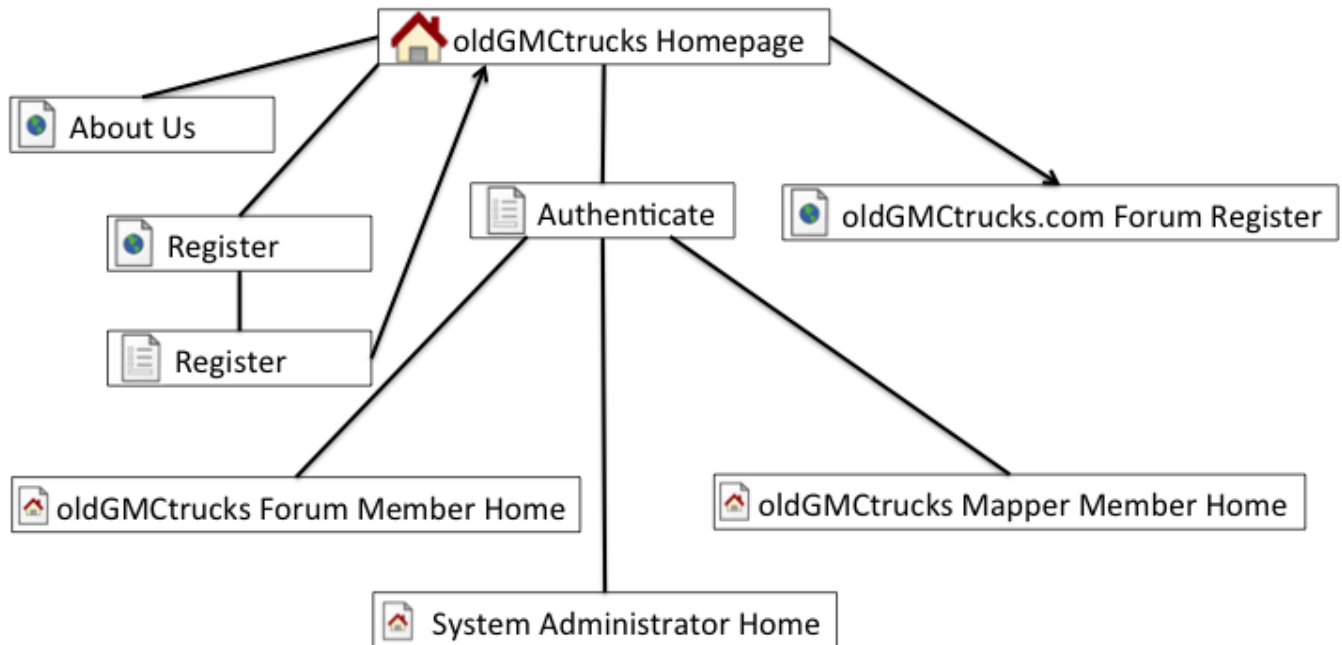
Link: Shows a link that is visible to a user via a hyperlink.



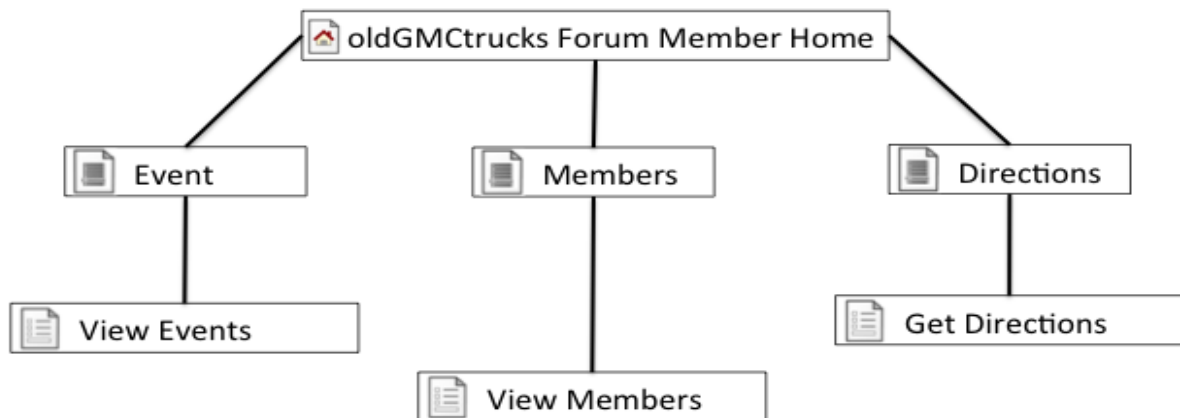
Redirect

Redirect: Shows a link that forcibly relocates a user to another webpage.

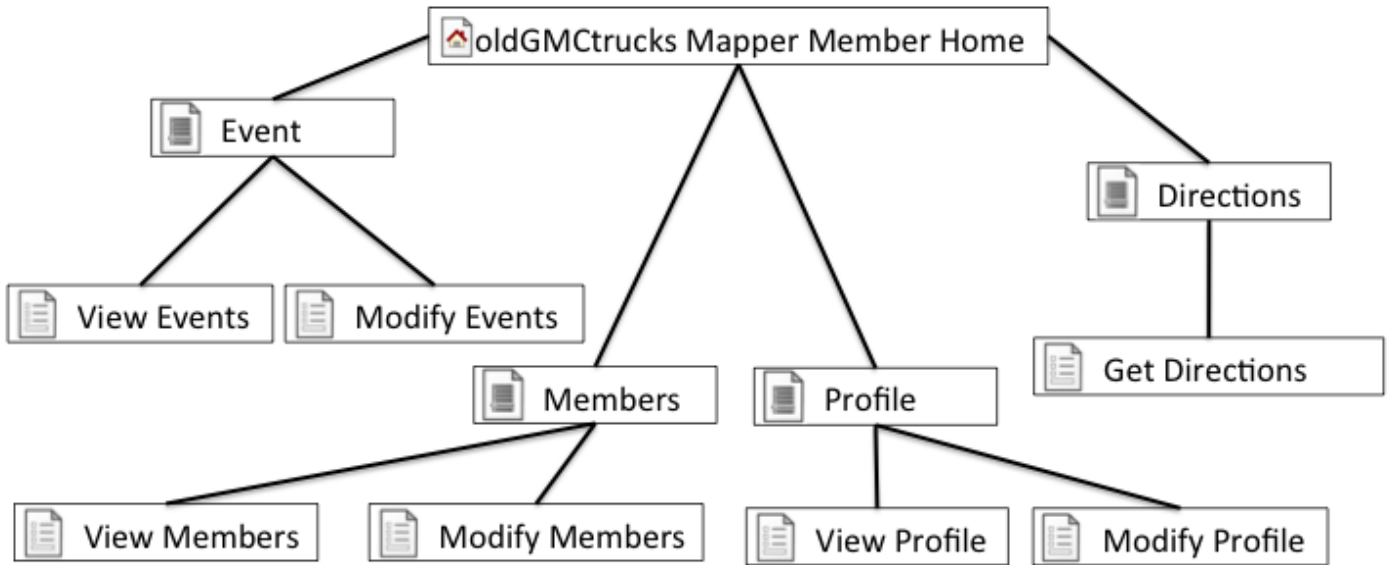
6.2 Website Map Homepage



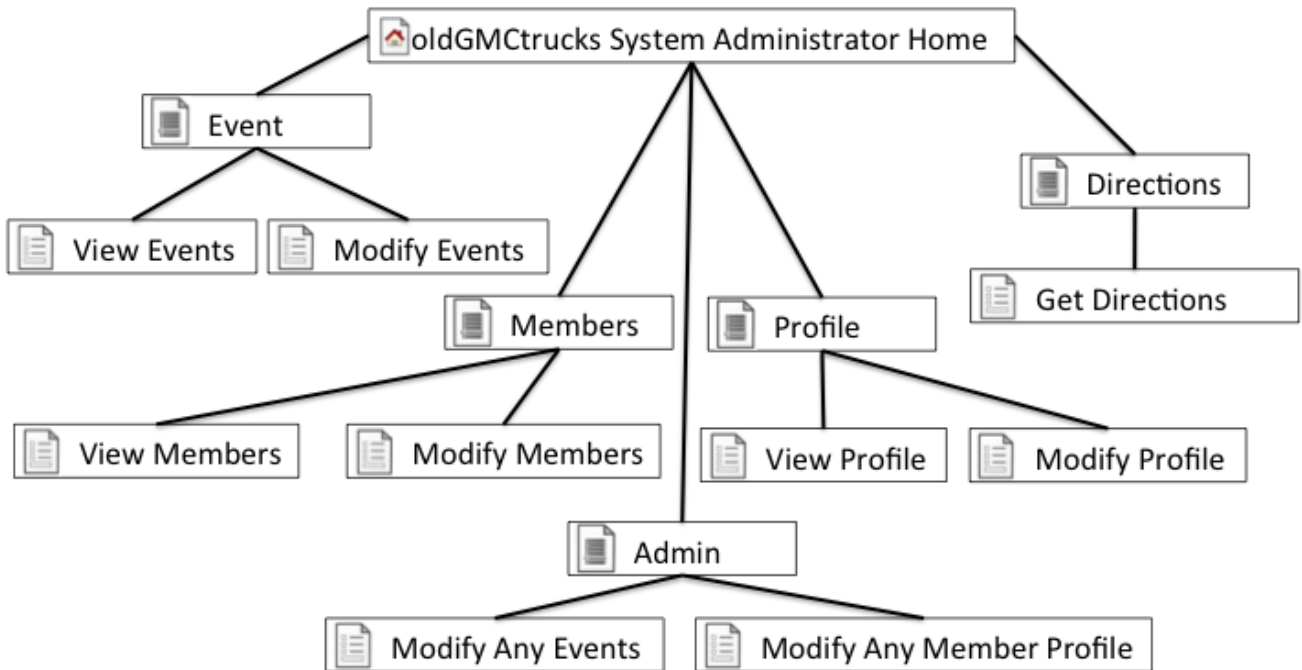
6.3 Website Map Forum Member Homepage



6.4 Website Map Member Homepage



6.5 Website Map Administrator Homepage



7. Dataflow Diagrams

Data flow diagrams show the movement of data throughout a system. The following diagrams show the flow of data within oldGMCTrucks Mapper. The data moving within oldGMCTrucks Mapper will allow users to create, edit, and browse profiles and event amount other actions a user can take. There are multiple levels, each breaking down the previous process into more detail.

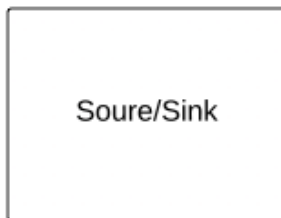
Context diagrams show the data flow between external entities and the system.

Level 0 diagrams show the main processes in the system and the data flow between the main processes and the entities.

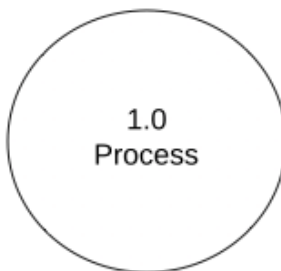
Level 1 diagrams go into more detail, expanding the parent processes shown in the level 0 diagram.

Level 2 diagrams go into even more detail, expanding the level 1 processes further.

7.1 Legend



Source or sinks are represented by a closed rectangle. A source or sink is either human or non-human where data originates from or ends up at. The name of the source or sink is written within the closed rectangle.



Processes are represented by a circle. A process is anything that performs an action on data. The process name is written inside the circle along with its process number.



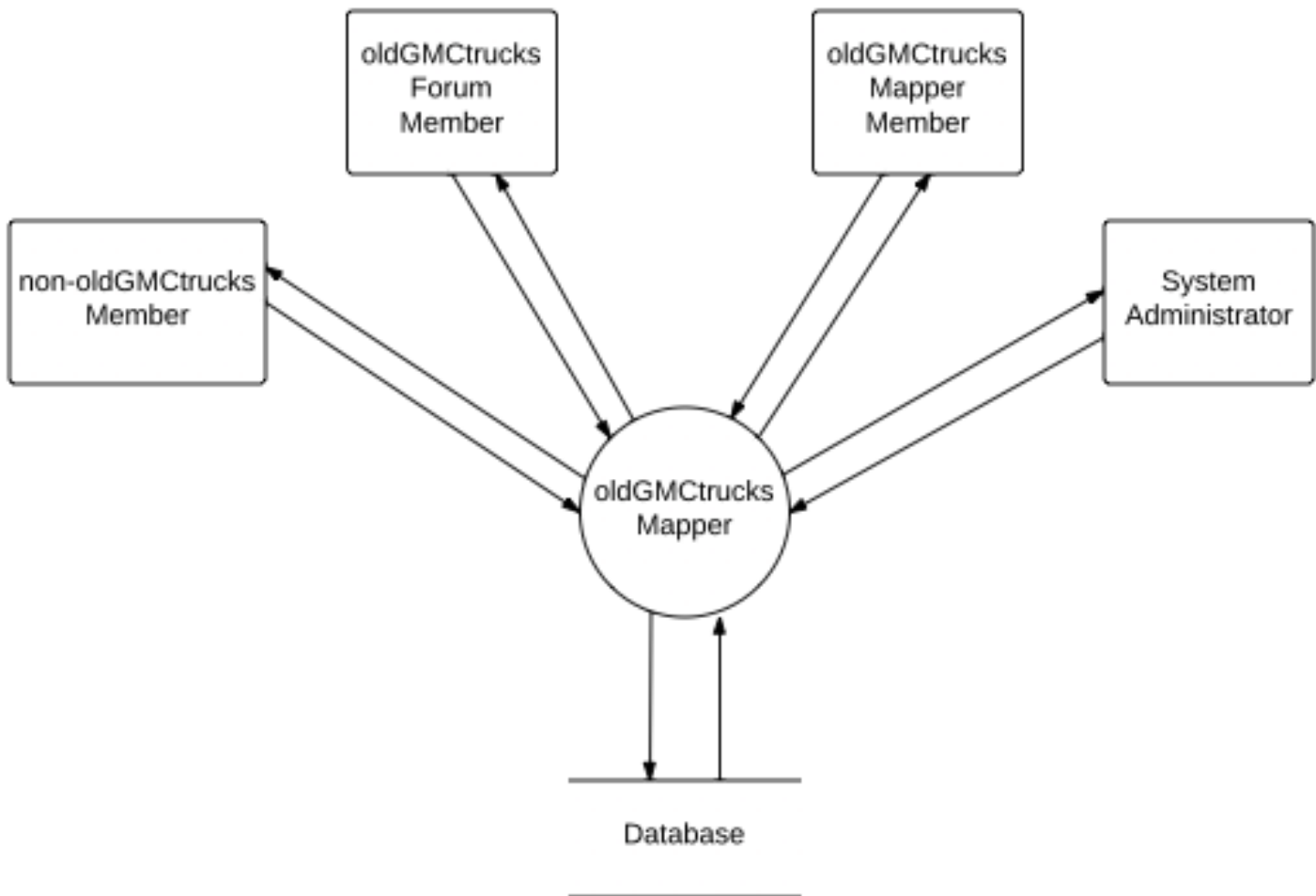
Data stores are represented by a rectangle that is open on both the right and left side. Data stores are physical or electronic locations where data is kept. The data store name is written inside the open rectangle.



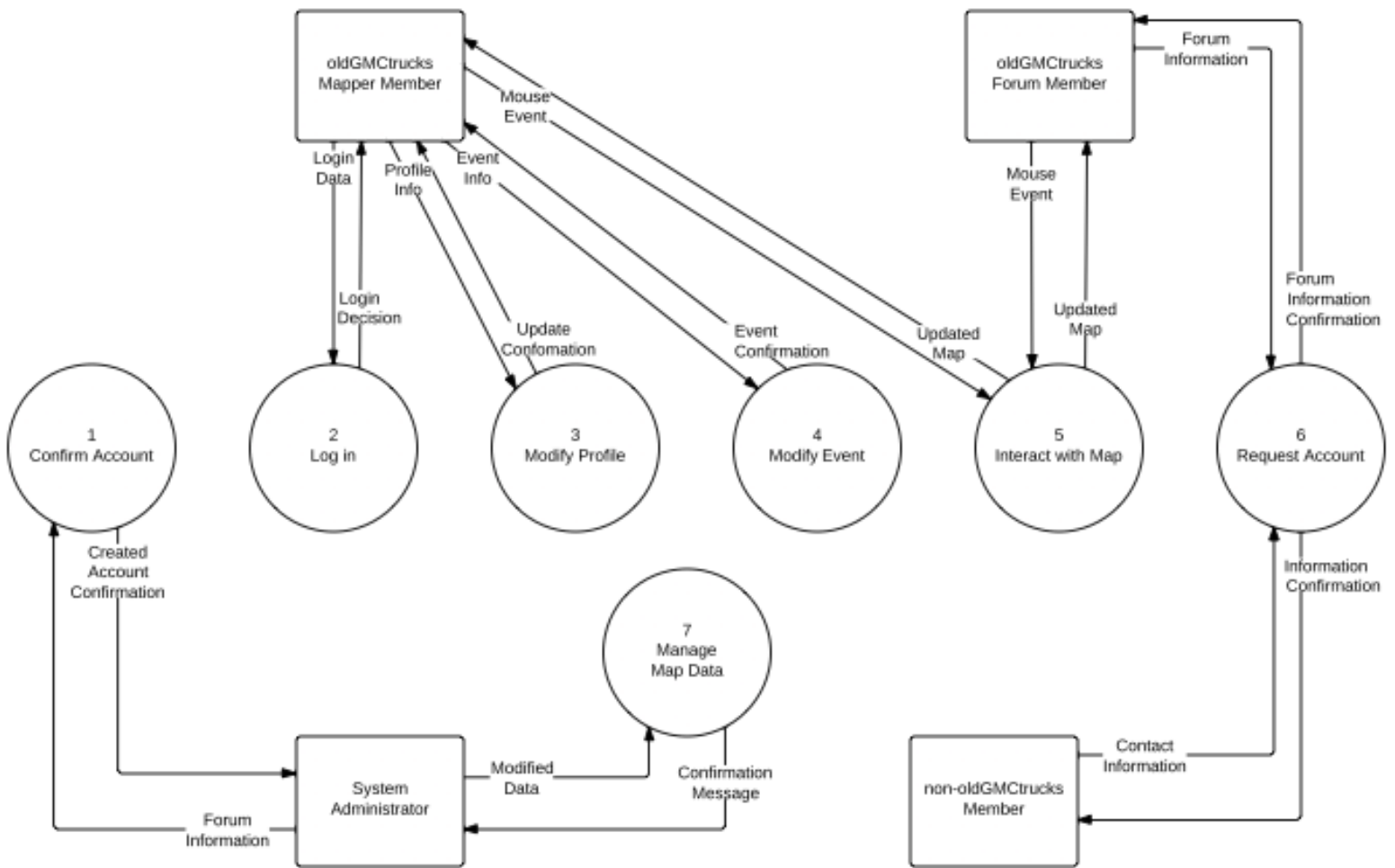
Data flows are represented by an arrow, the direction of the data flow is indicated by the direction of the arrowhead. A description of the data is written on the arrow.

7.2 Dataflow Diagrams

7.2.1 Context Diagram

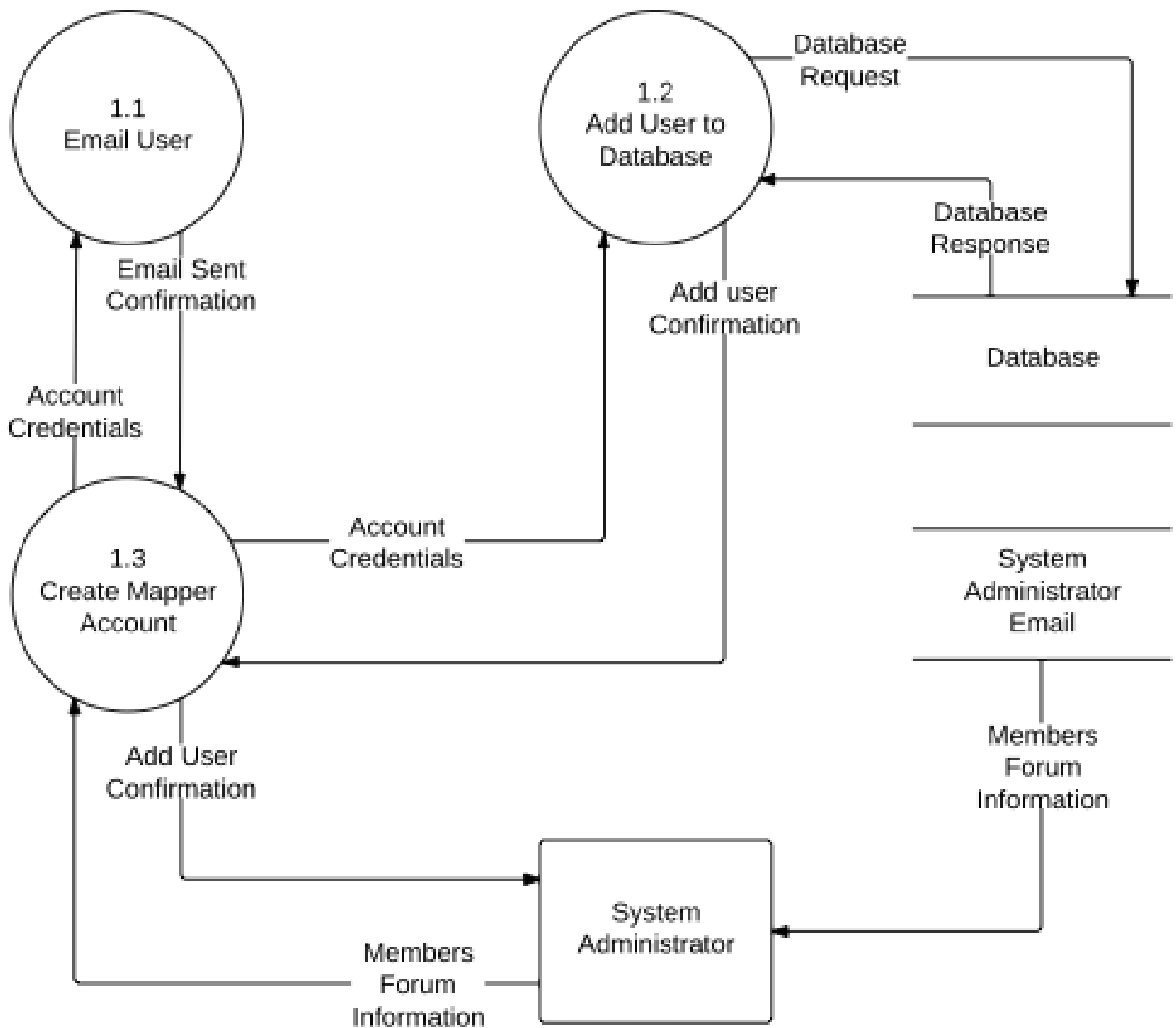


7.2.2 Level 0 Diagram

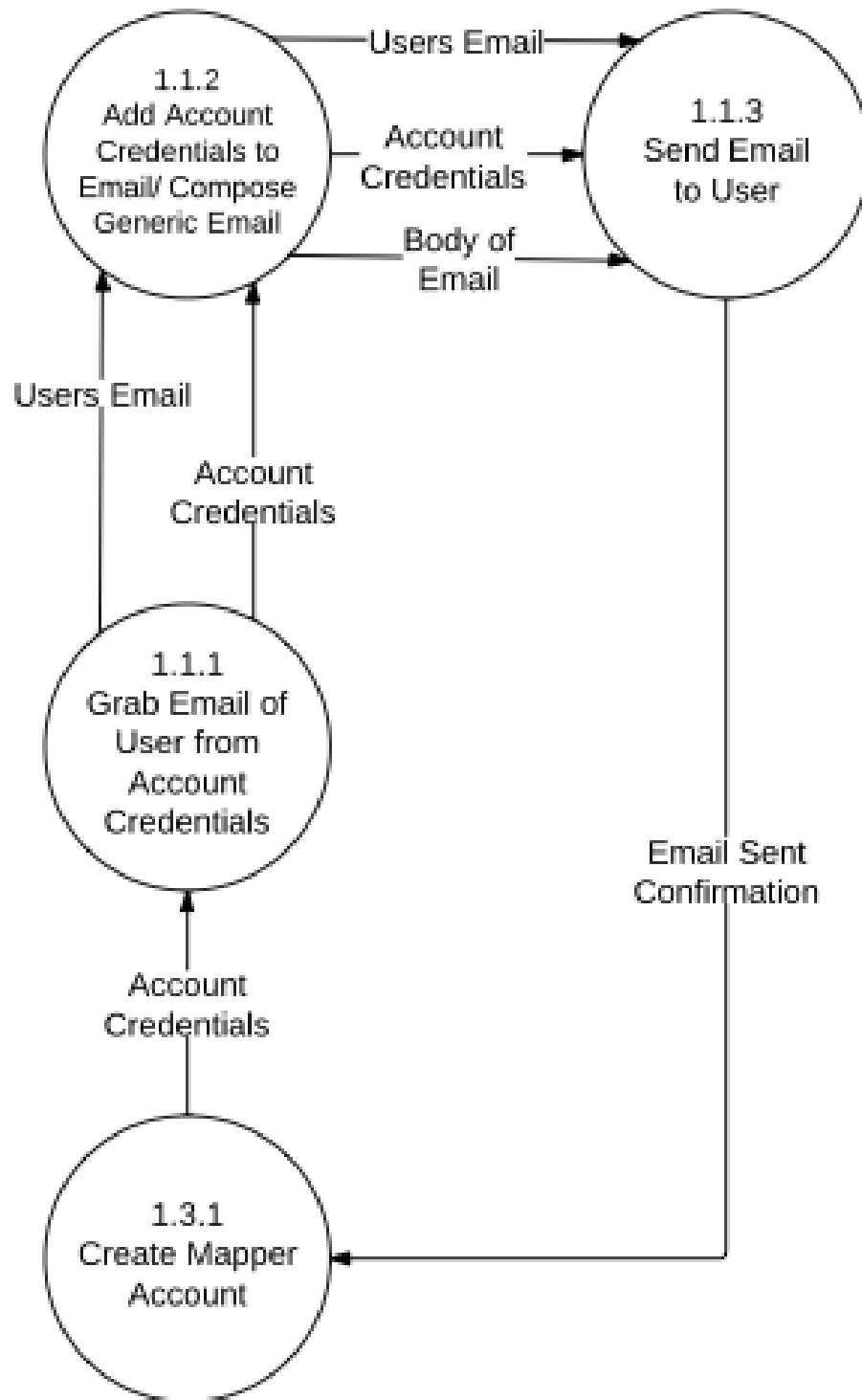


The System Administrator also acts as a oldGMTrucks Mapper Member and will have all the functionality of a oldGMTrucks Mapper Member

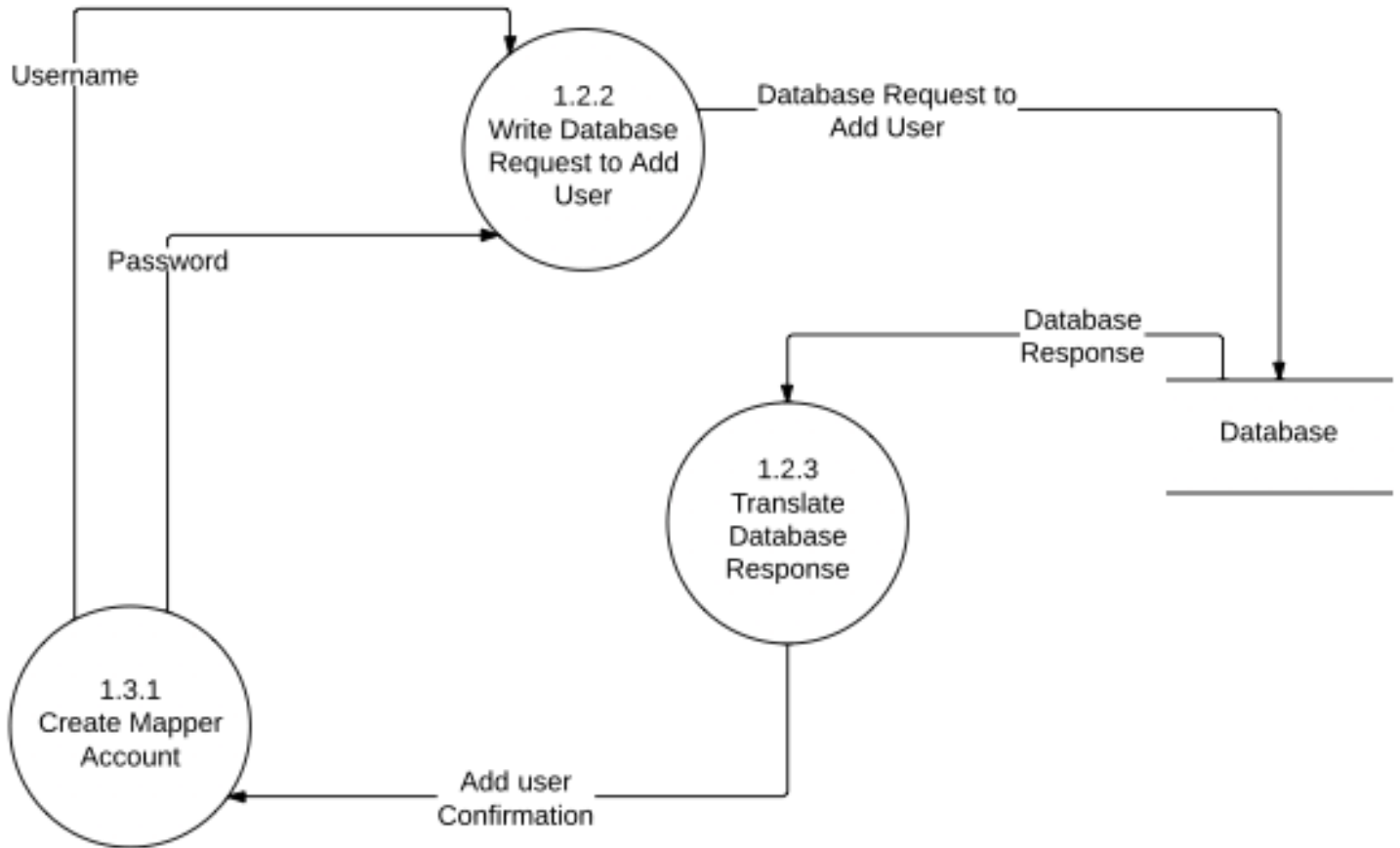
7.2.3 Level 1 - Confirm Account



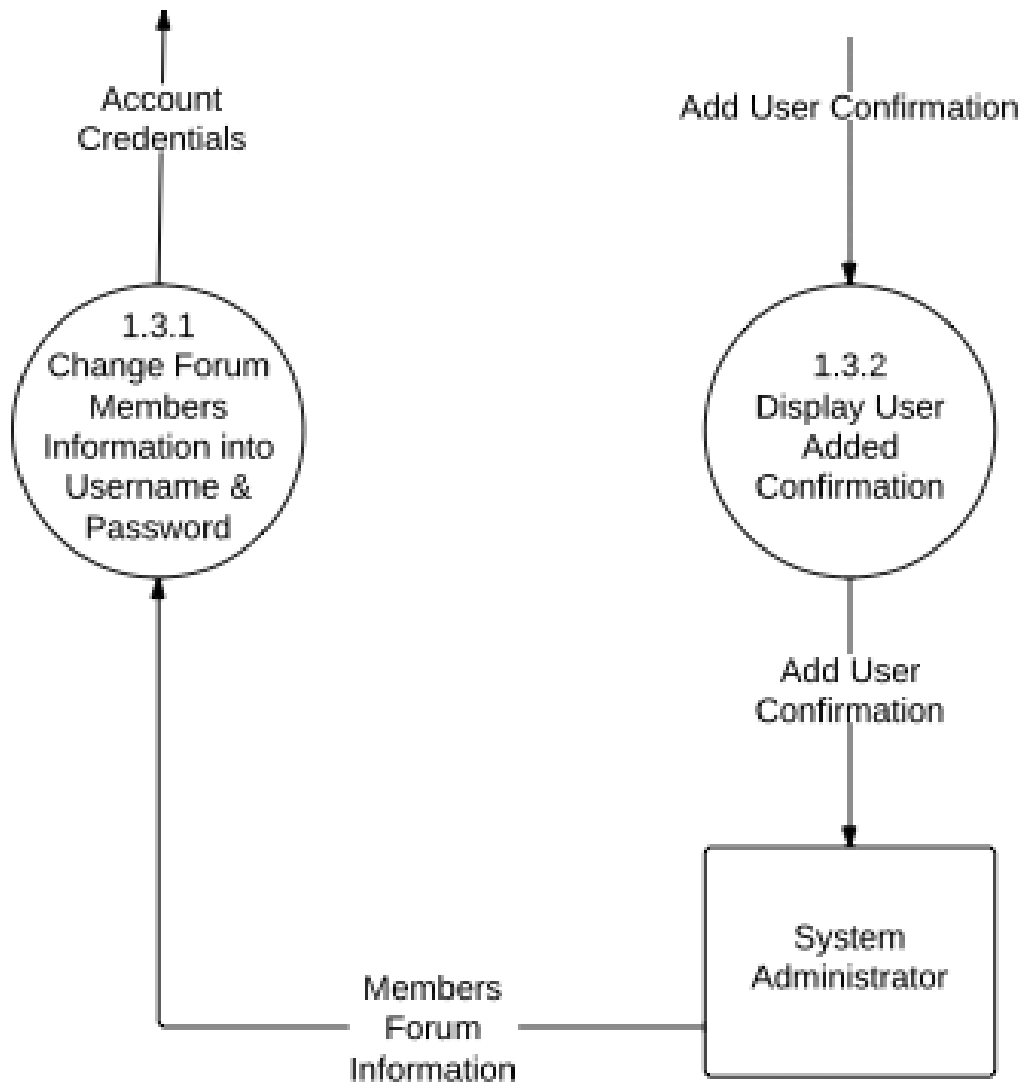
7.2.4 Level 2 - Confirm Account - Email User



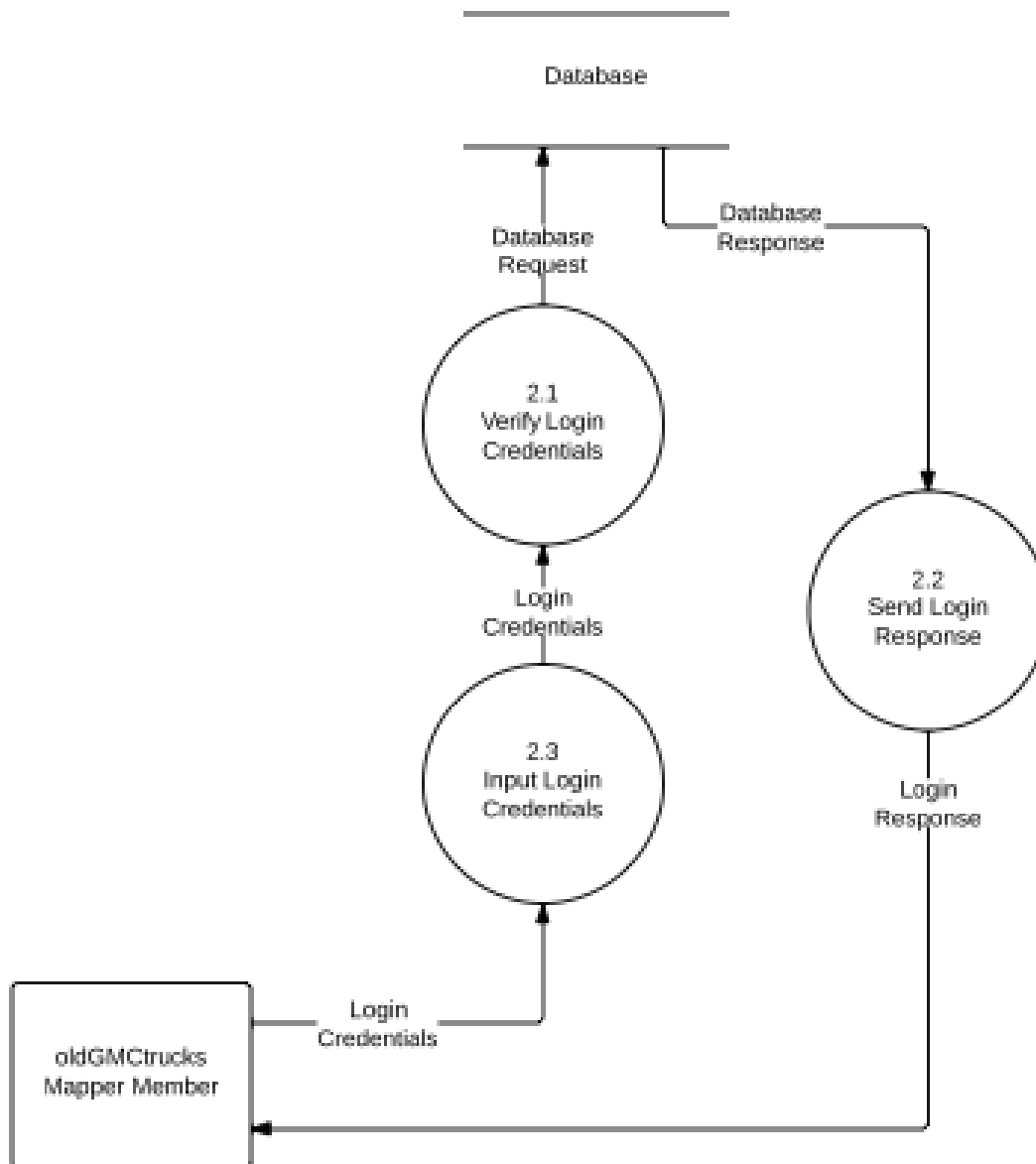
7.2.5 Level 2 – Confirm Account - Add User to Database



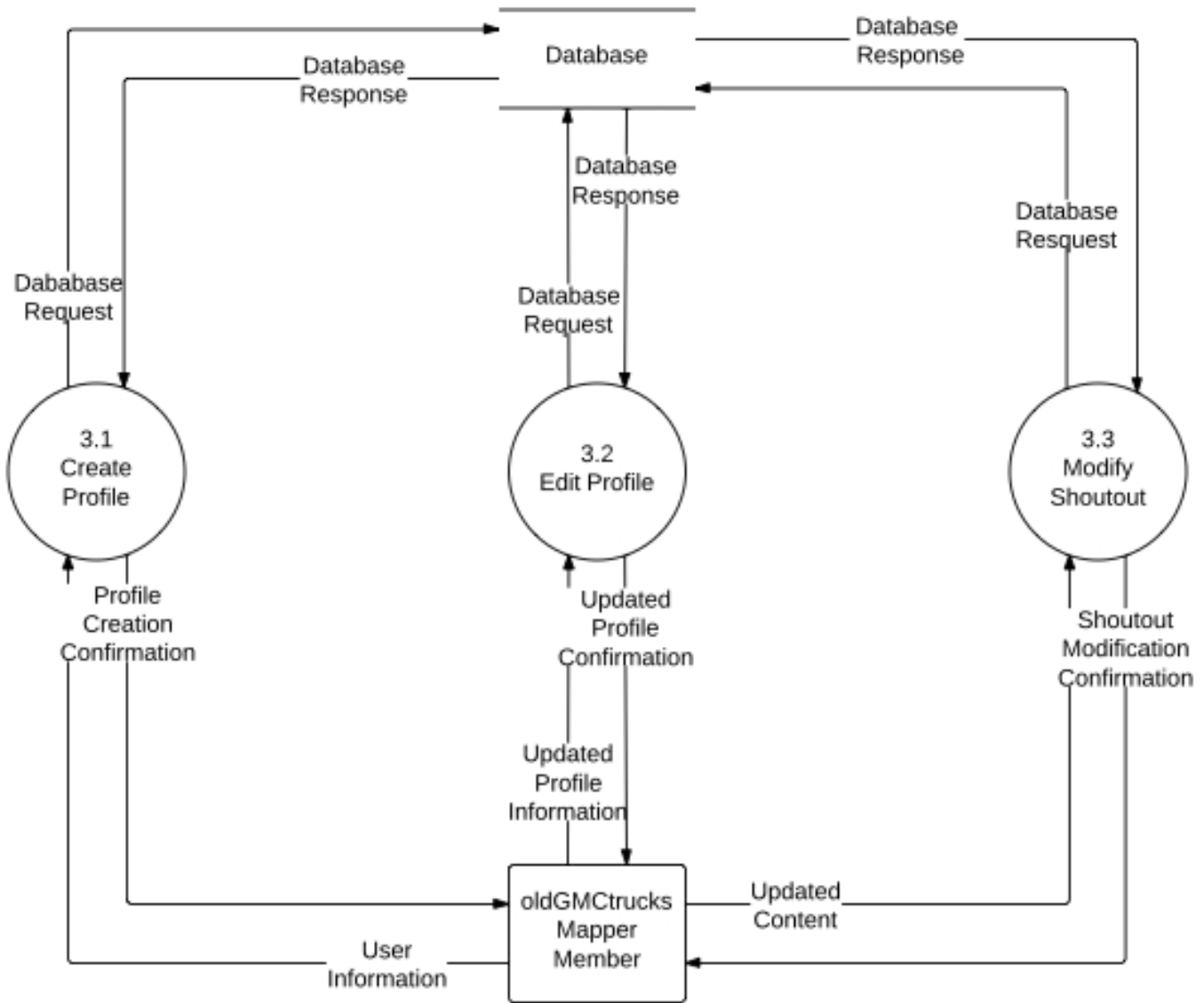
7.2.6 Level 2 – Confirm Account - Create Mapper Account



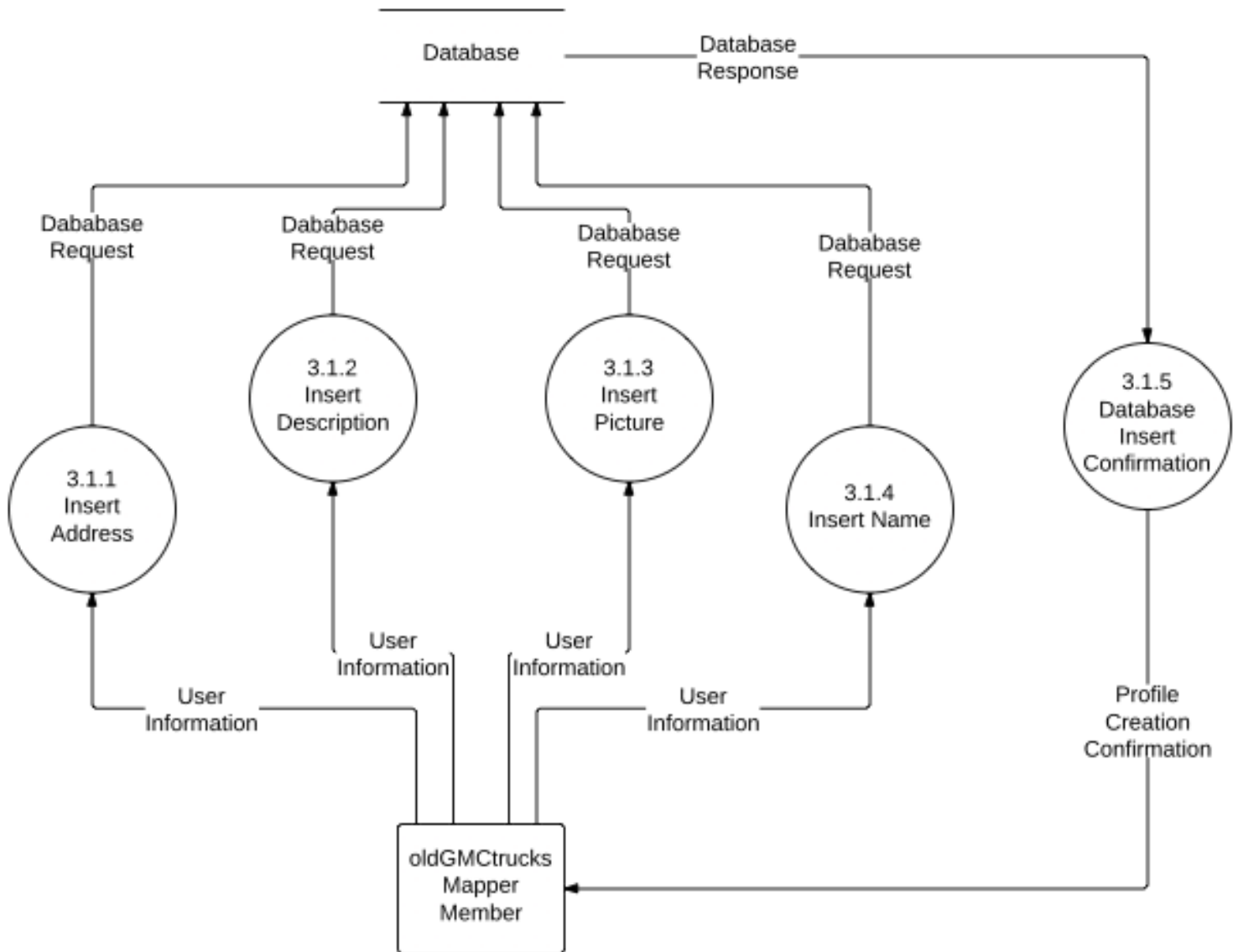
7.2.7 Level 1 - Log in



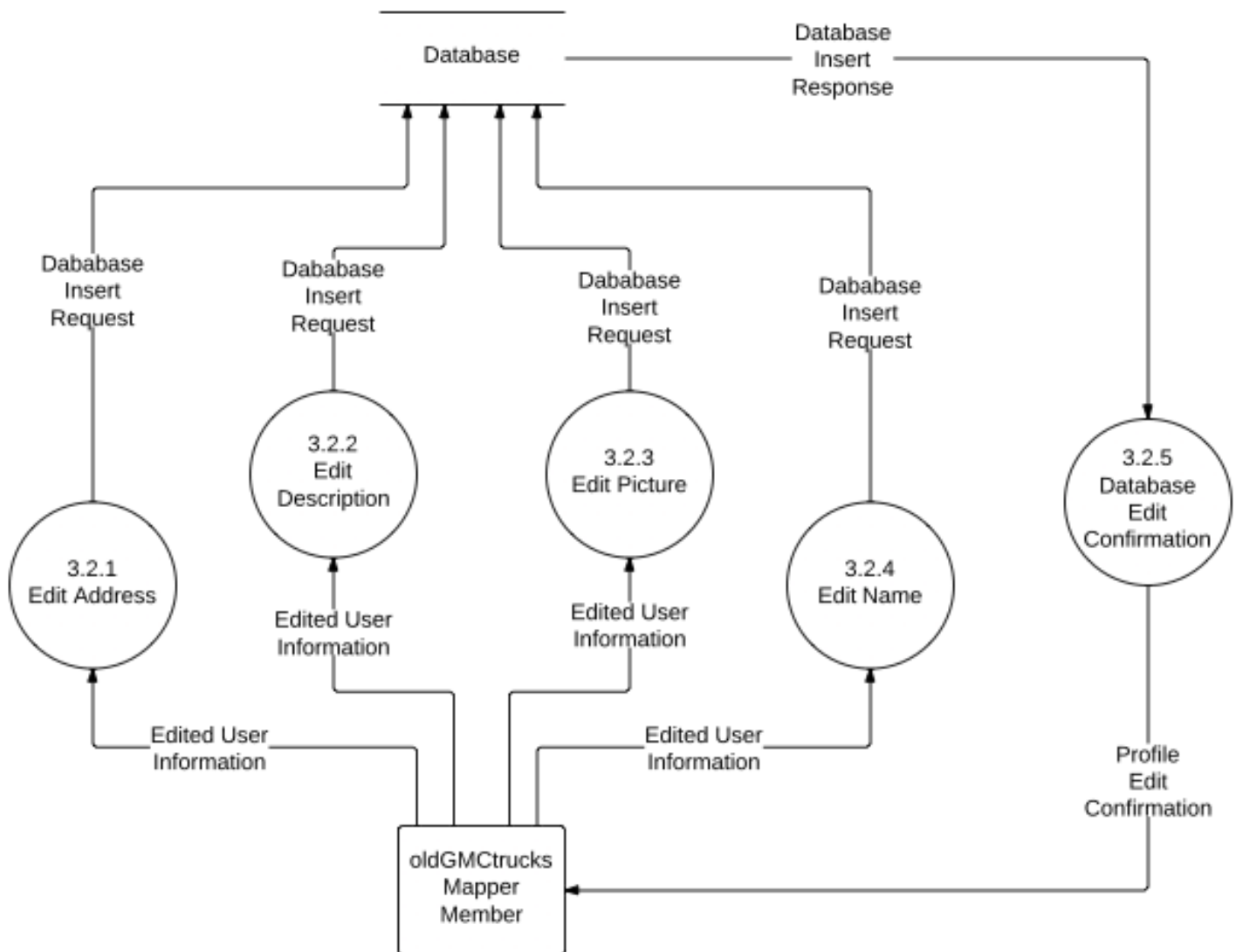
7.2.8 Level 1 - Modify Profile



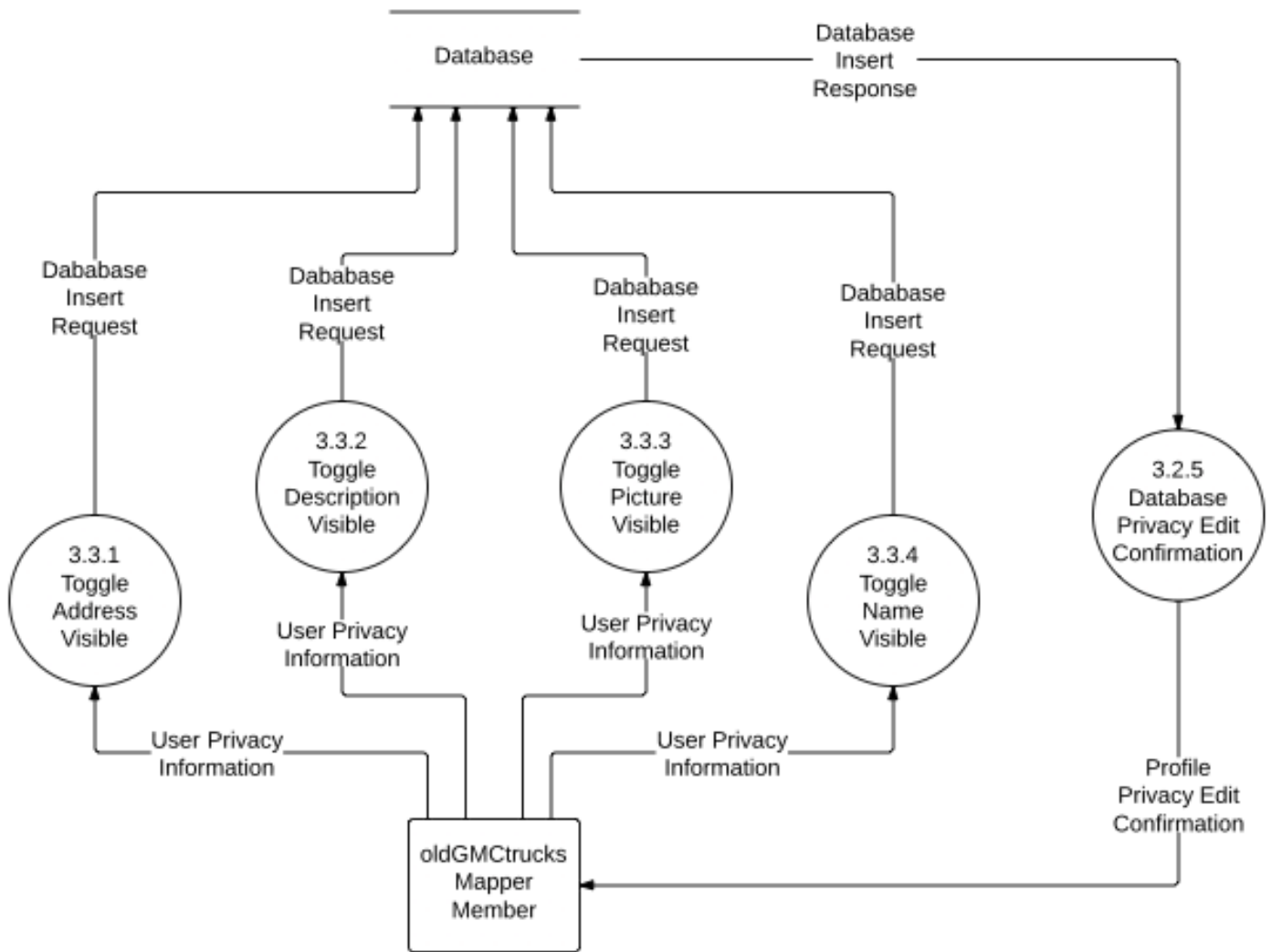
7.2.9 Level 2 - Modify Profile - Create Profile



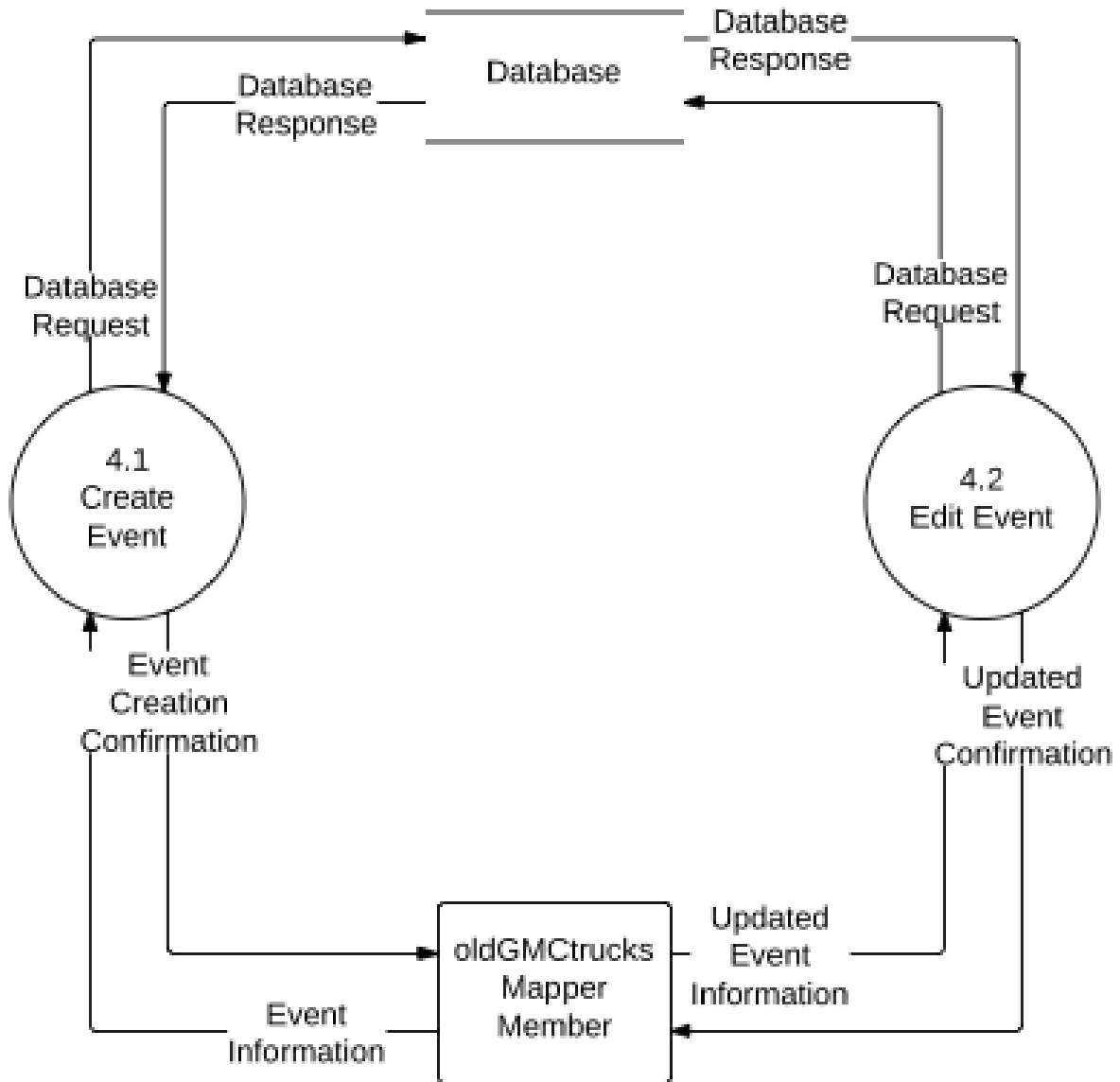
7.2.10 Level 2 - Modify Profile – Edit Profile



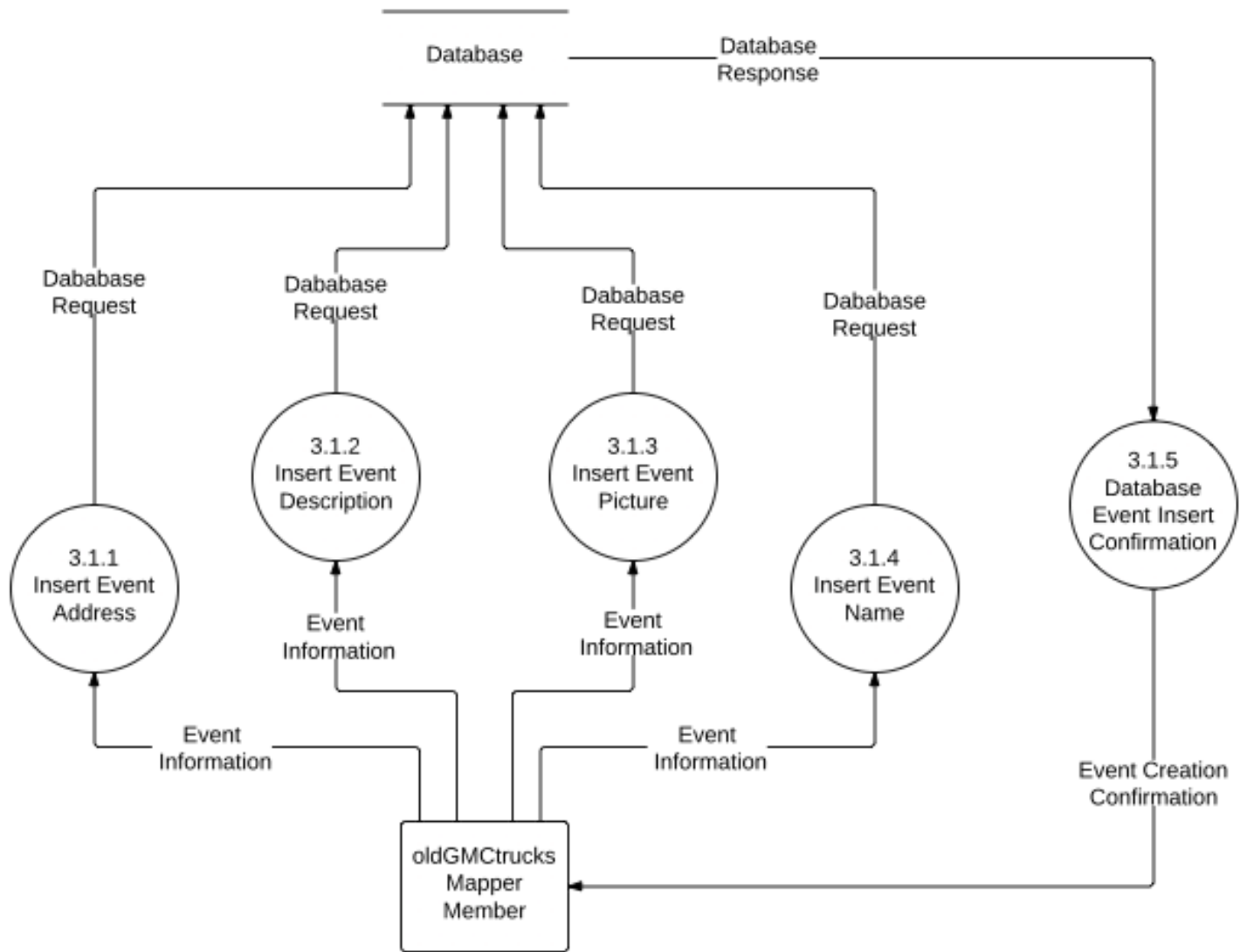
7.2.11 Level 2 - Modify Profile – Modify Shoutout



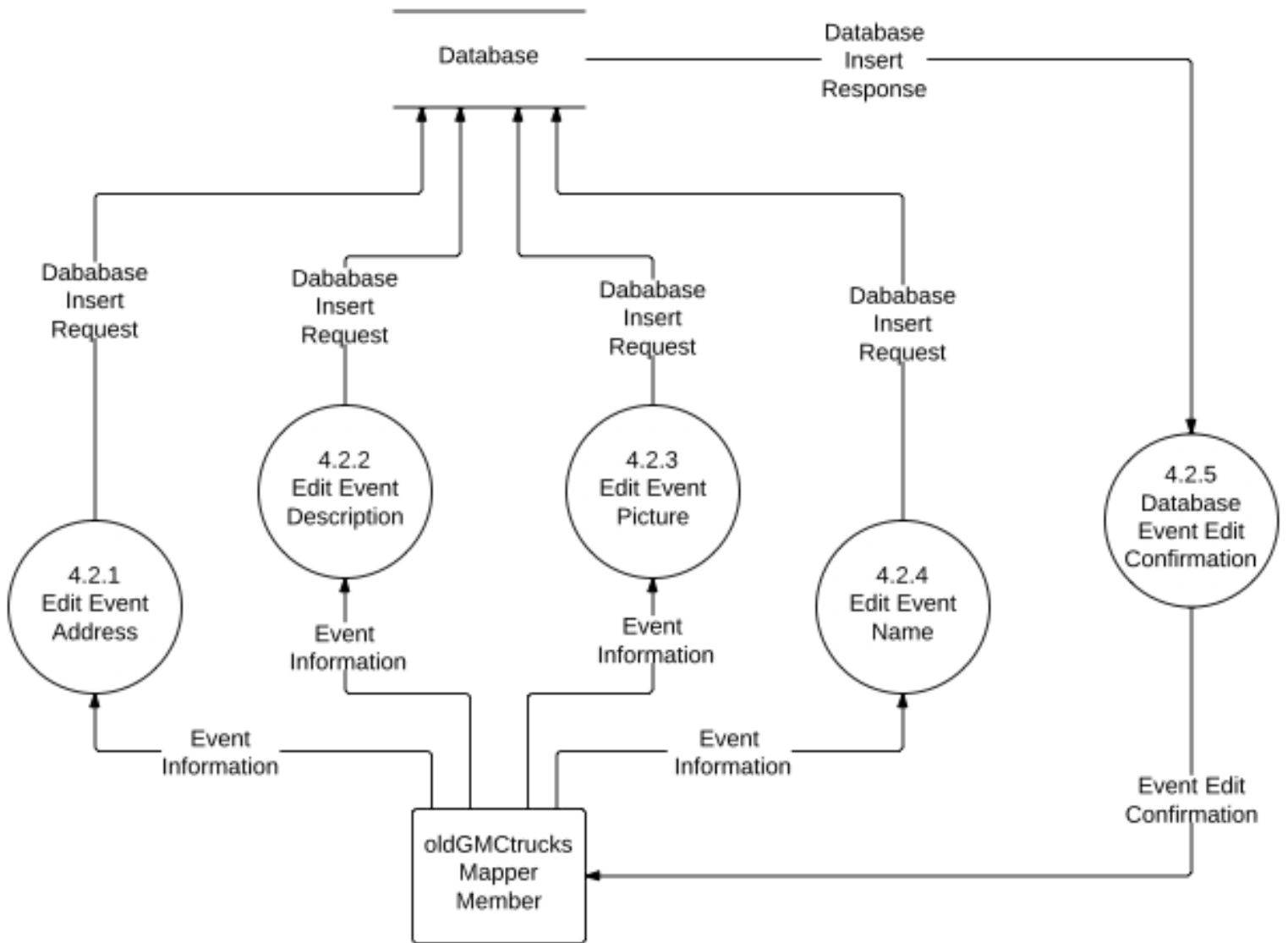
7.2.12 Level 1 - Modify Event



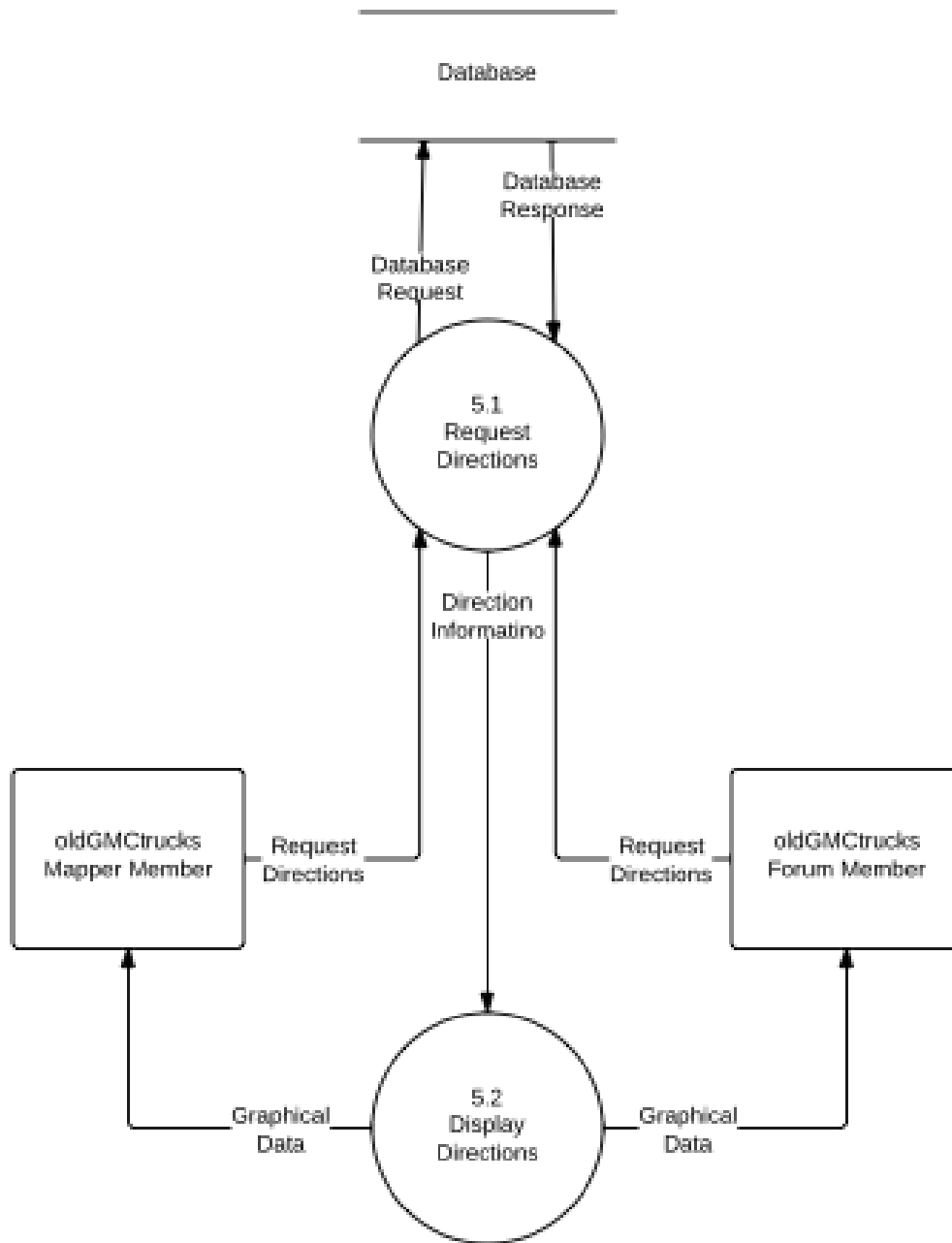
7.2.13 Level 2 – Modify Event - Create Event



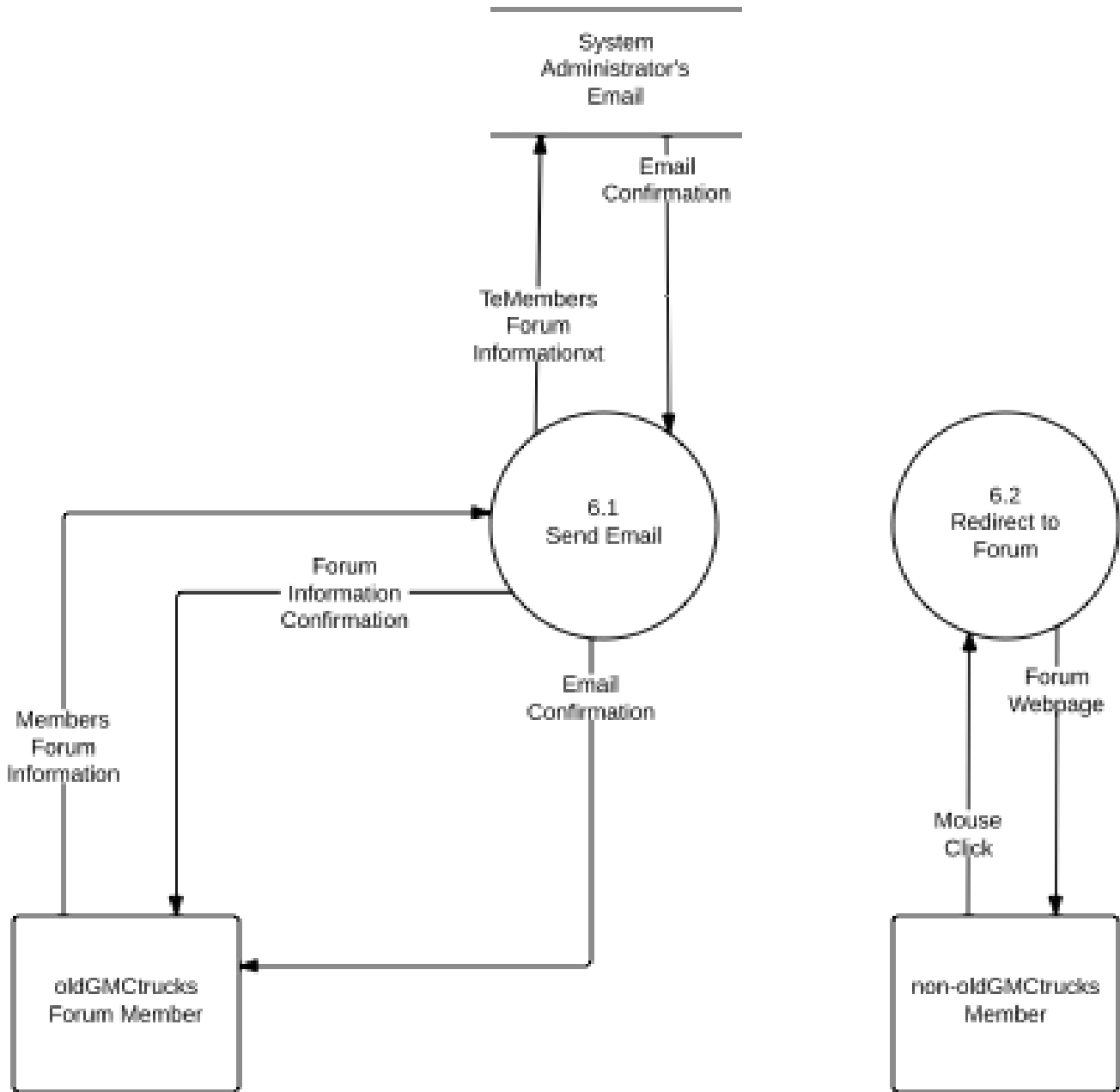
7.2.14 Level 2 – Modify Event - Edit Event



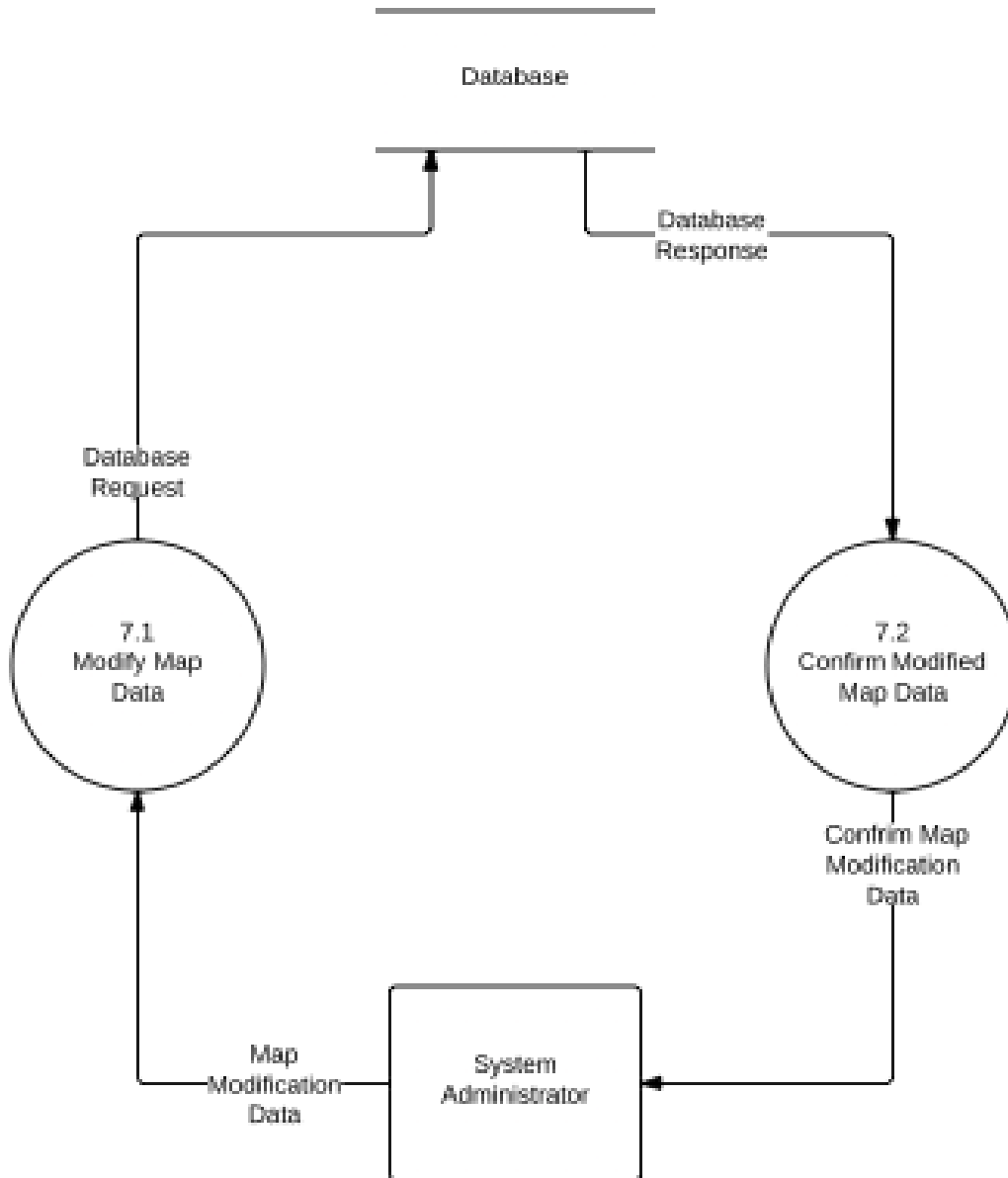
7.2.15 Level 1 - Interact with Map



7.2.17 Level 1 - Request Account



7.2.18 Level 1 - Manage Map Data

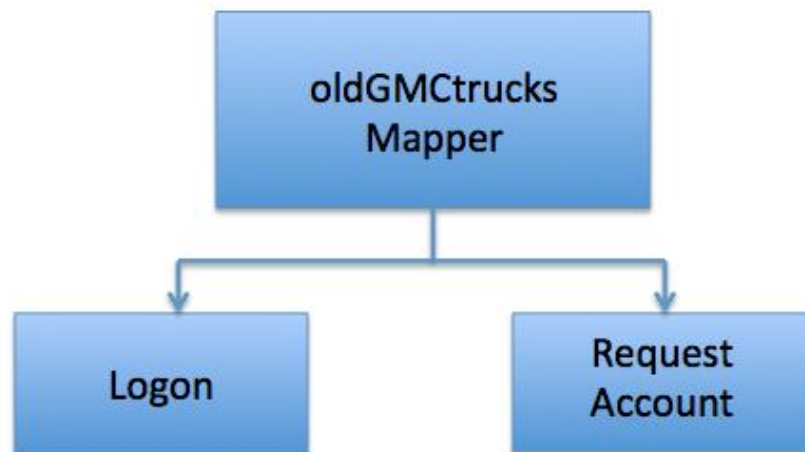


8. Structure Diagram

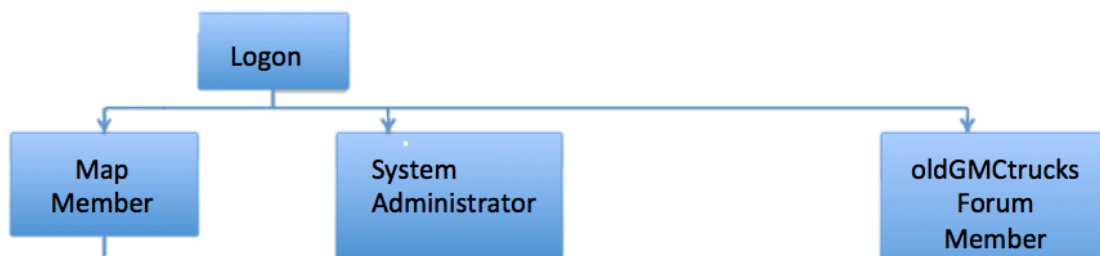
8.1 Structure Diagrams

A Structure Diagram shows the breakdown of oldGMTrucks Mapper at the lowest manageable levels. Each rectangular box represents a module while the arrows show the relationships between the modules.

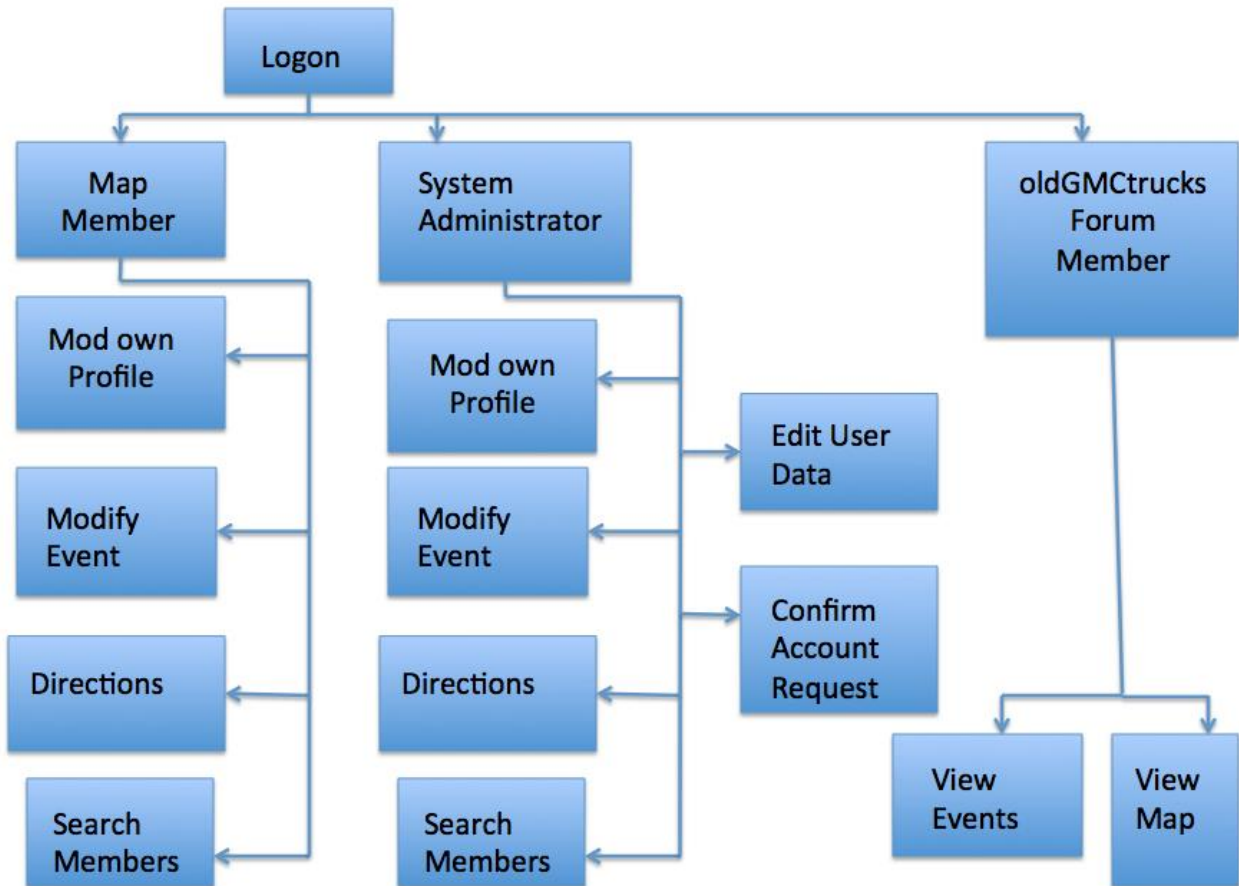
8.2.1 Context Diagram



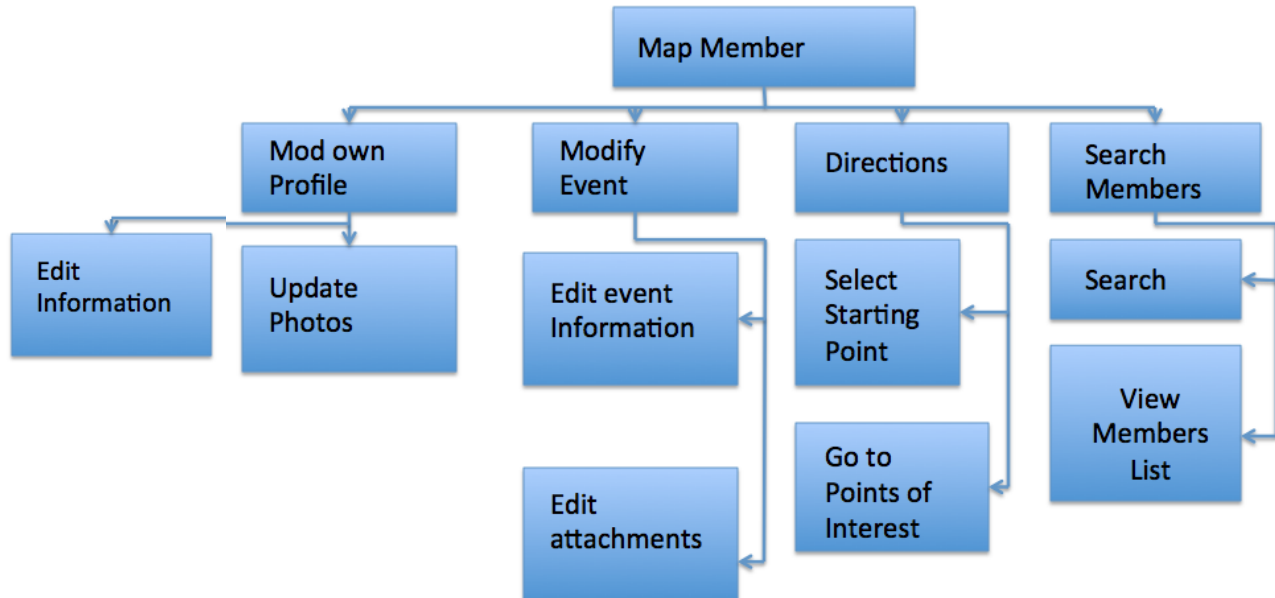
8.2.2 Expansion of Login Module



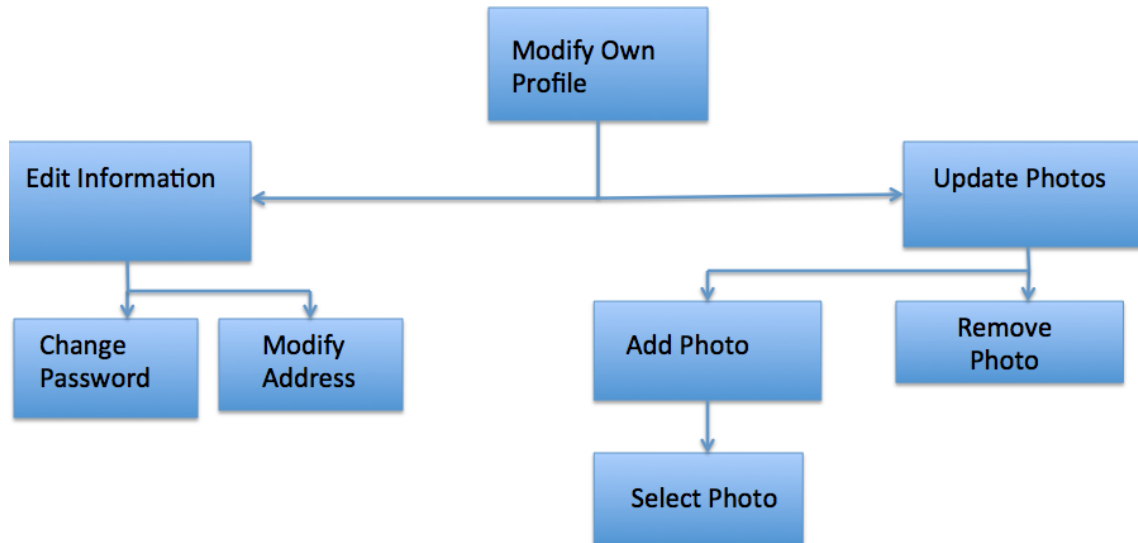
8.2.3 Expansion of Modules Off of the Login Module



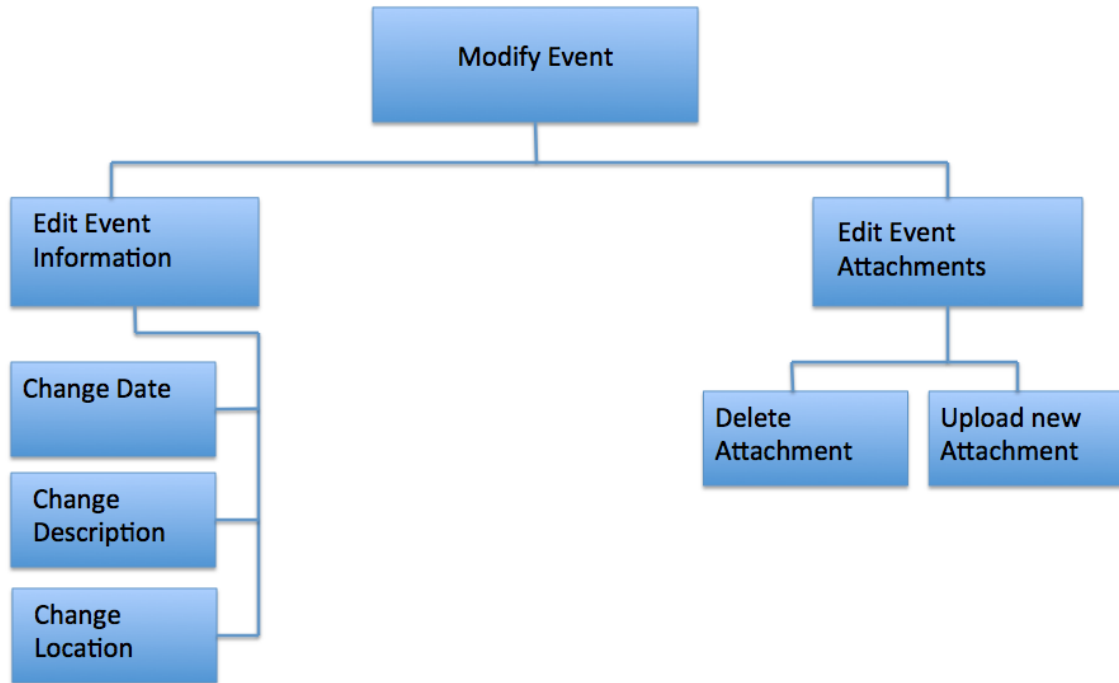
8.2.4 Expansion of Map Member Module



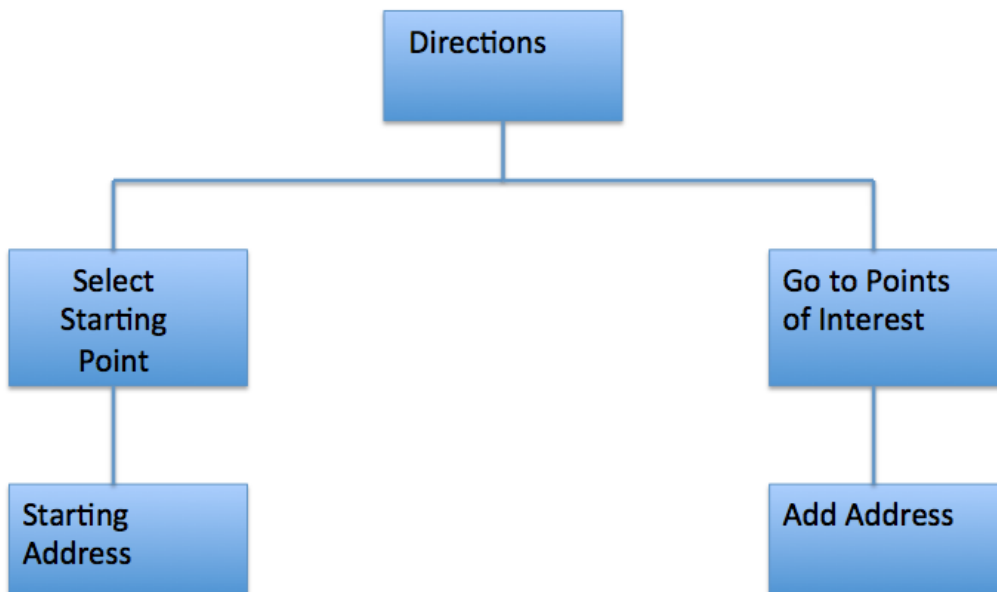
8.2.5 Expansion of Modify Own Profile Module



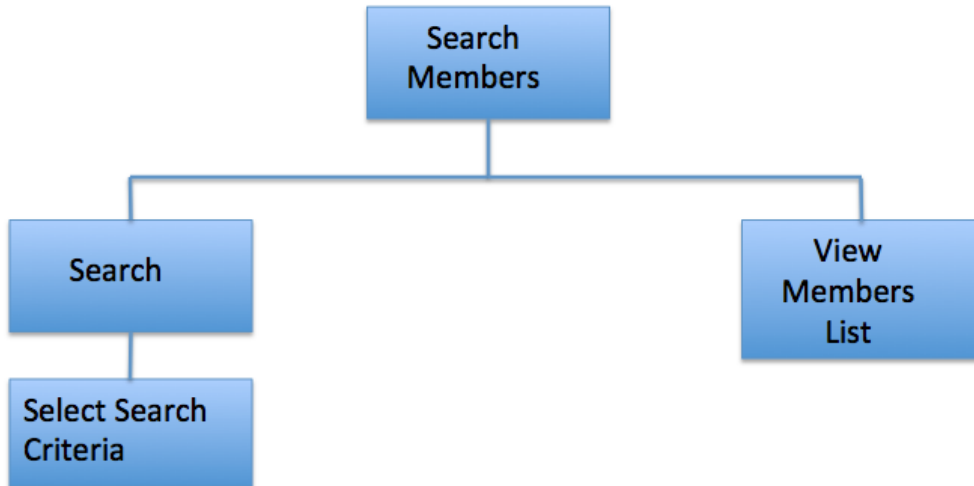
8.2.6 Expansion of Modify Event Module



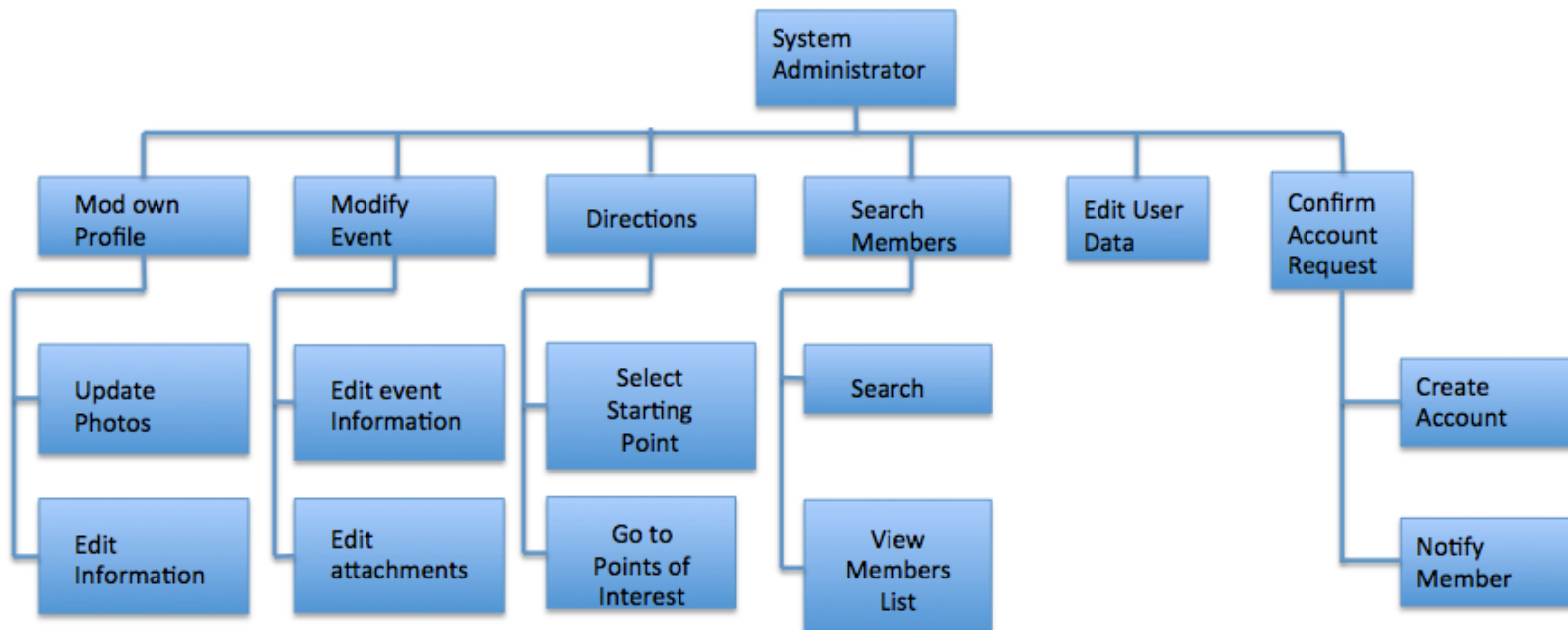
8.2.7 Expansion of Directions Module



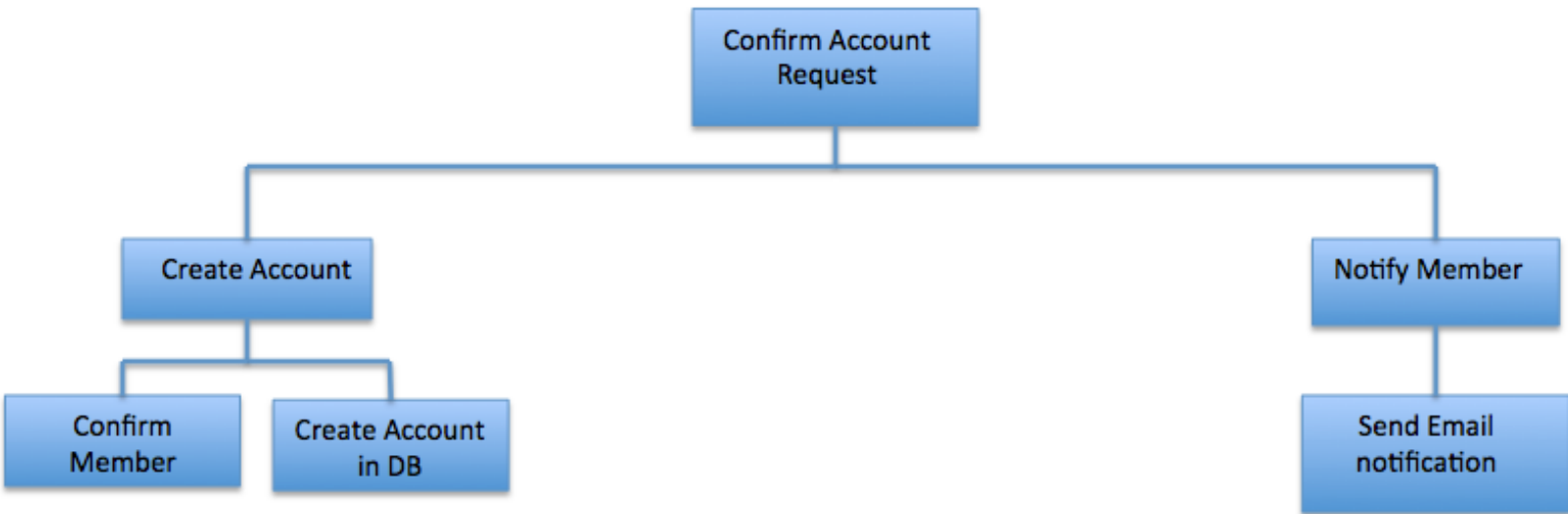
8.2.8 Expansion of Search Module



8.2.9 Expansion of System Administrators Module



8.2.10 Expansion of Confirm Account Request Module



9. Functional Requirements Inventory

The functional requirements inventory serves to define each function the system will provide for each different actor. Each function will include a set of inputs, the behavior (data transformation etc...) and outputs.

9.1 General User Interface Functional Requirements

- Map with pins representing member and event locations
 - Upon click, each pin displays with a “shoutout” containing member or event information
- At least two distinct pins for users and events
- Will have a slide-out right aligned panel with the following tabs
 - Members
 - Will have a search bar to search for members via the members’ first or last name first name, last name, username, or address.
 - Events
 - Directions
 - Will have the ability to create a trip between member’ locations and events
 - Profile
 - Admin
 - Will only exist for Administrators

9.2 Random Actor

- Will not be able to see anything. Redirected to create an account on the oldGMCTrucks forum.
- Will see blank map
- Will have the ability to sign up for an account on oldGMCTrucks forum.

9.3 oldGMCTrucks Forum Member Actor

- Will have the ability to access the site through the oldGMCTrucks forum.
- Will have the ability to view a map with user information on the map.
- Will have the ability to create an account on the oldGMCTrucks Mapper site.

9.4 oldGMCTrucks Mapper Member Actor

- Will have the ability to log in.
- Will have the ability to update information in their profile.
 - Update First name
 - Update Last name
 - Update Home address
 - Update Phone numbers
 - Update e-Mails
 - Update “about me”
 - Update Pictures
- Members can derive directions to an event or to another member’s location.
- Members can create events.
- Members can update any events.

9.5 oldGMCTrucks Mapper administrator

- Will have a profile with the same functionality as an oldGMCTrucks Mapper Member.
- Admin will also be able to modify any member’s data.
- Admin will be able to create user accounts.

10. Non-Functional Requirements Inventory

Non-functional requirements are non-specific behaviors of a system used to judge the overall quality of the operation of that system. For the oldGMCTrucks Mapper (the system), we have judged the following to be our non-functional requirements:

- 1) *The system will run efficiently on the following browsers.*
 - a. Internet Explorer 6.0+
 - b. Firefox 2.0+
 - c. Safari 3.1+
 - d. Chrome
- 2) *The system will be compliant with the Google Terms of Service.*
- 3) *Interactions with the system will be intuitive for non-tech savvy people.*
- 4) *The system will be well documented.*
- 5) *The system will be extensible.*
- 6) *The system will be reliable.*
- 7) *The system will be secure.*

Because non-functional requirements are non-specific behaviors, running acceptance tests on non-functional requirements is near impossible, and can really only be tested through user testing.

11. Logical Data Dictionary

The data dictionary is a repository that shows information about the data within the oldGMCTrucks Mapper system, including format, usage, and examples of correct and incorrect input.

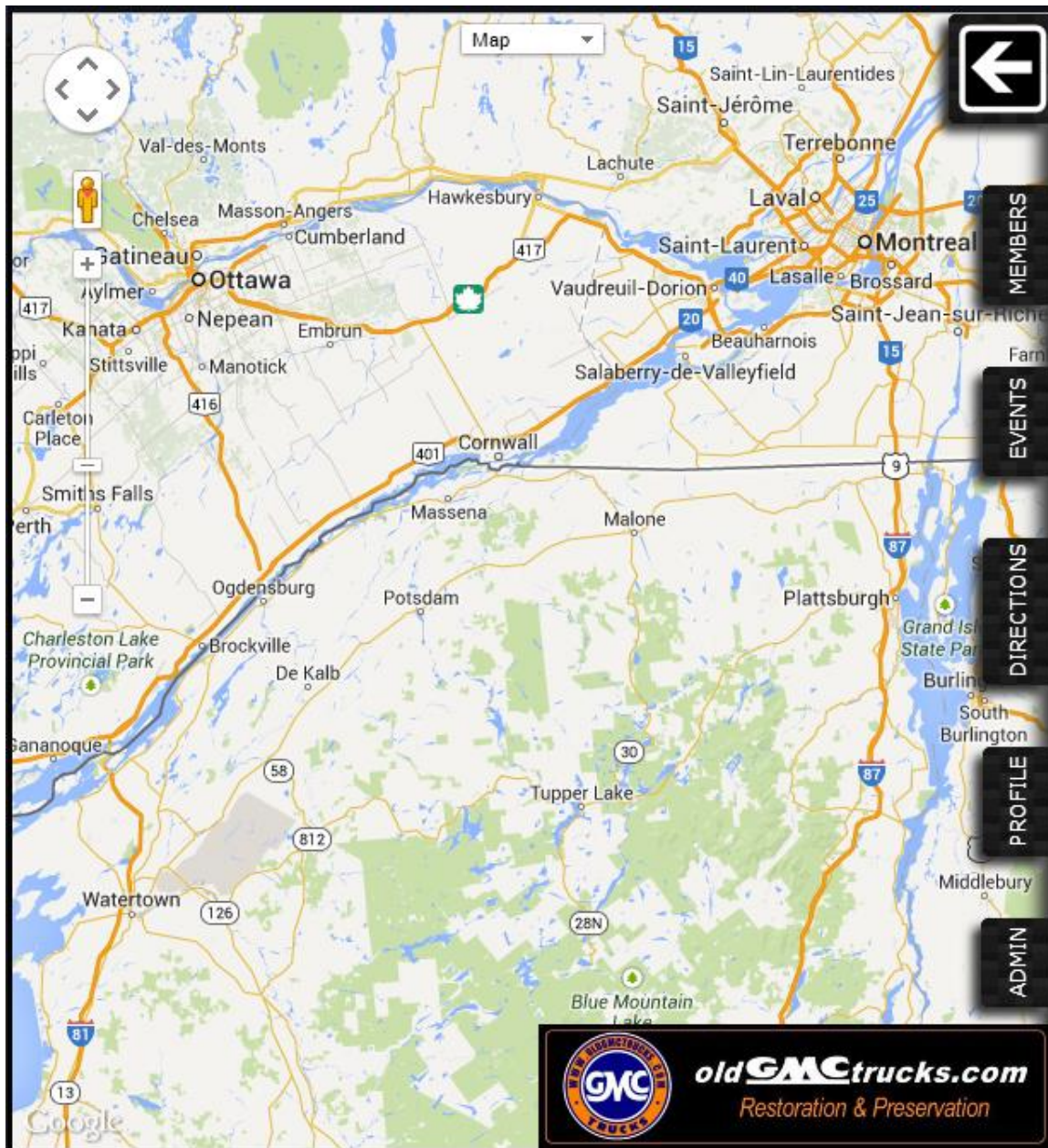
Please see the attached document to view the data dictionary. A Microsoft Excel file of the data dictionary is also available on the BitSize Solutions website.

12. Prototypes

To better show the design idea for the product consider the following pictures of various functions of our system.

12.1 Admin Home page

What the admin will see when the panel is hidden



12.2 Member Home Page

What the member sees when the panel is hidden



12.3 Profile Login Tab

What the anonymous user sees in regards to the panel



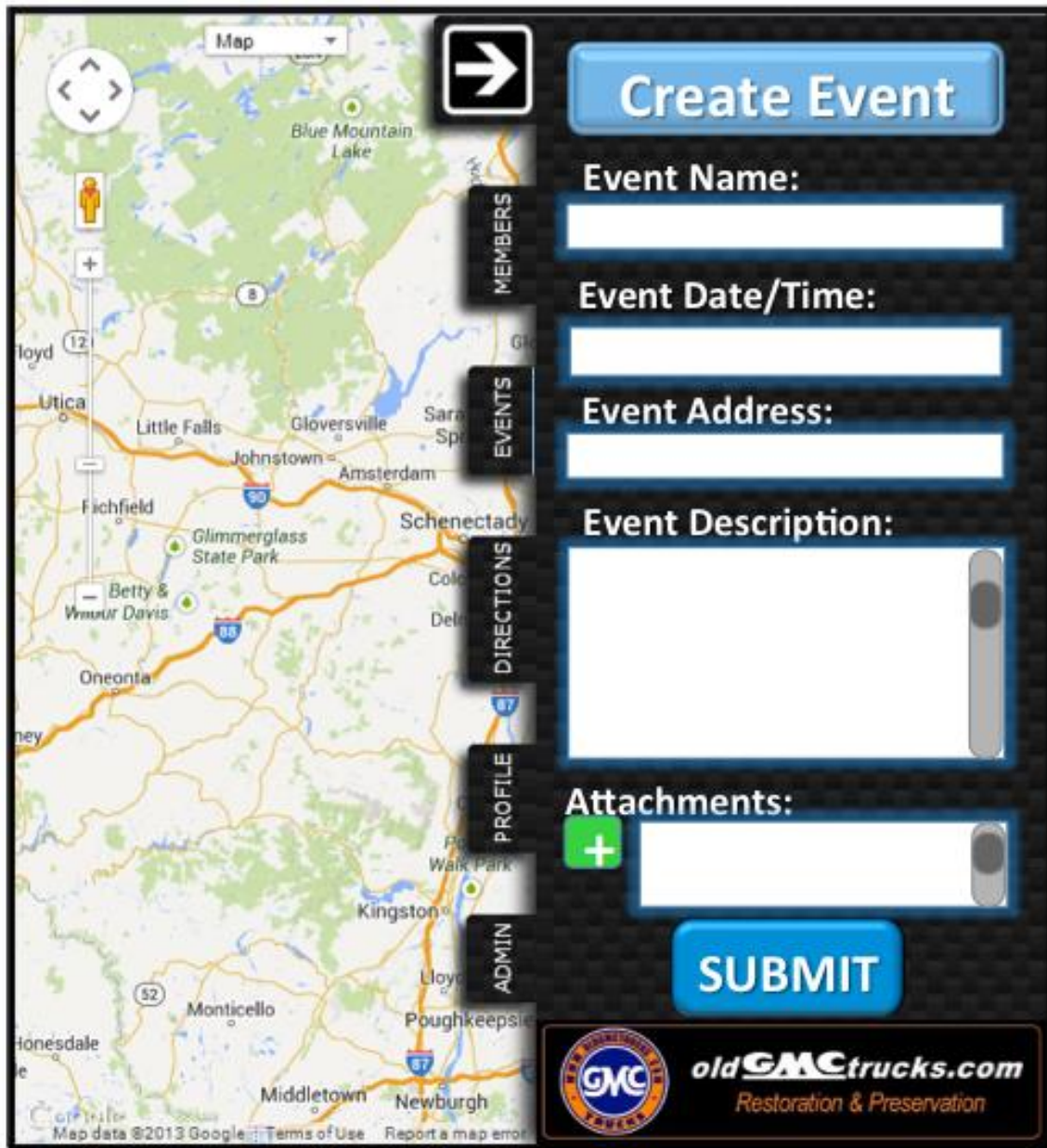
12.4 Event Prototype

What the event panel tab will look like



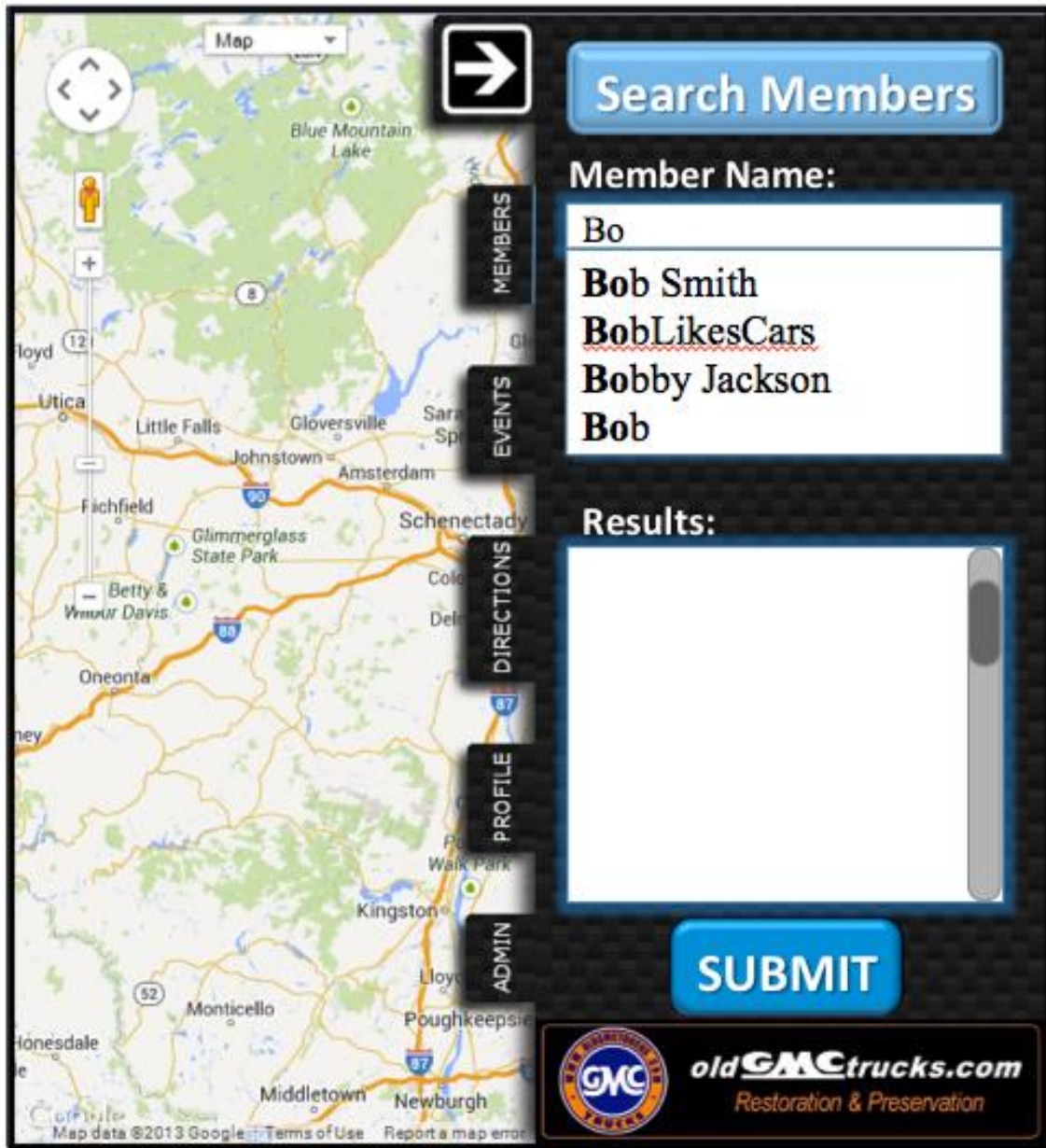
12.5 Create Event Prototype

What the event panel turns into when creating an event

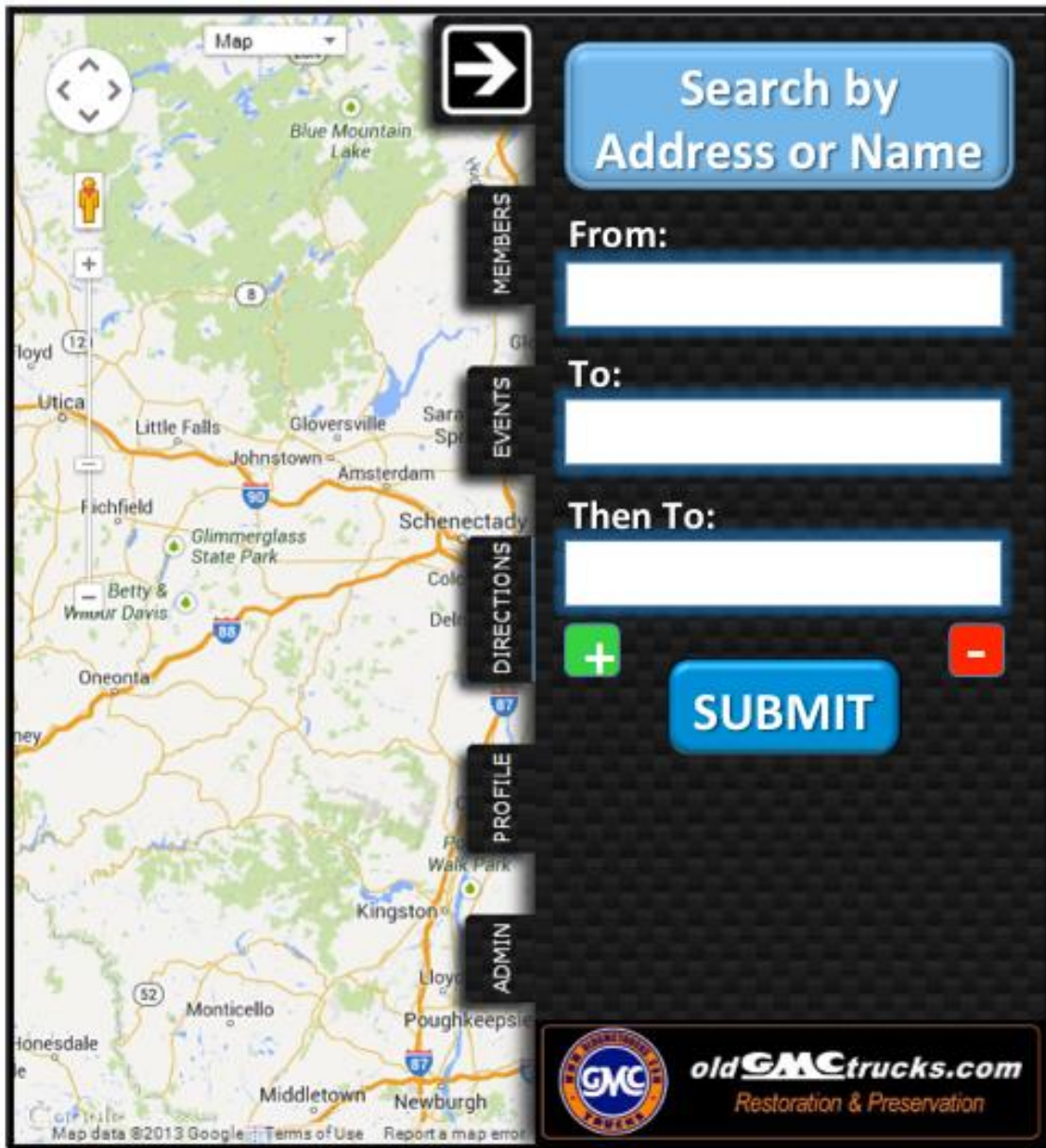


12.6 Search Members

What the member tab will look like

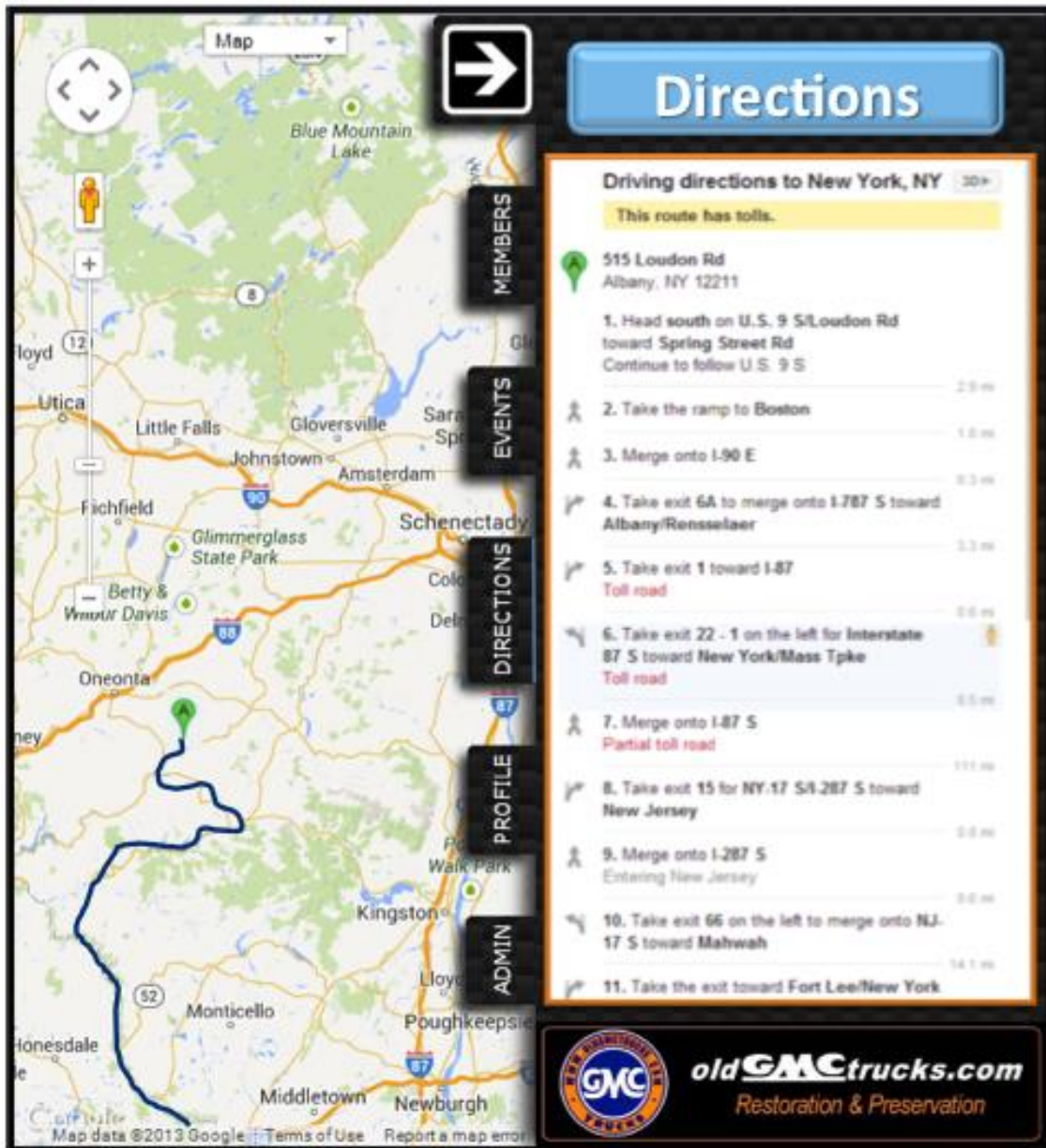


12.7 Search Members Options
Example of the directions tab



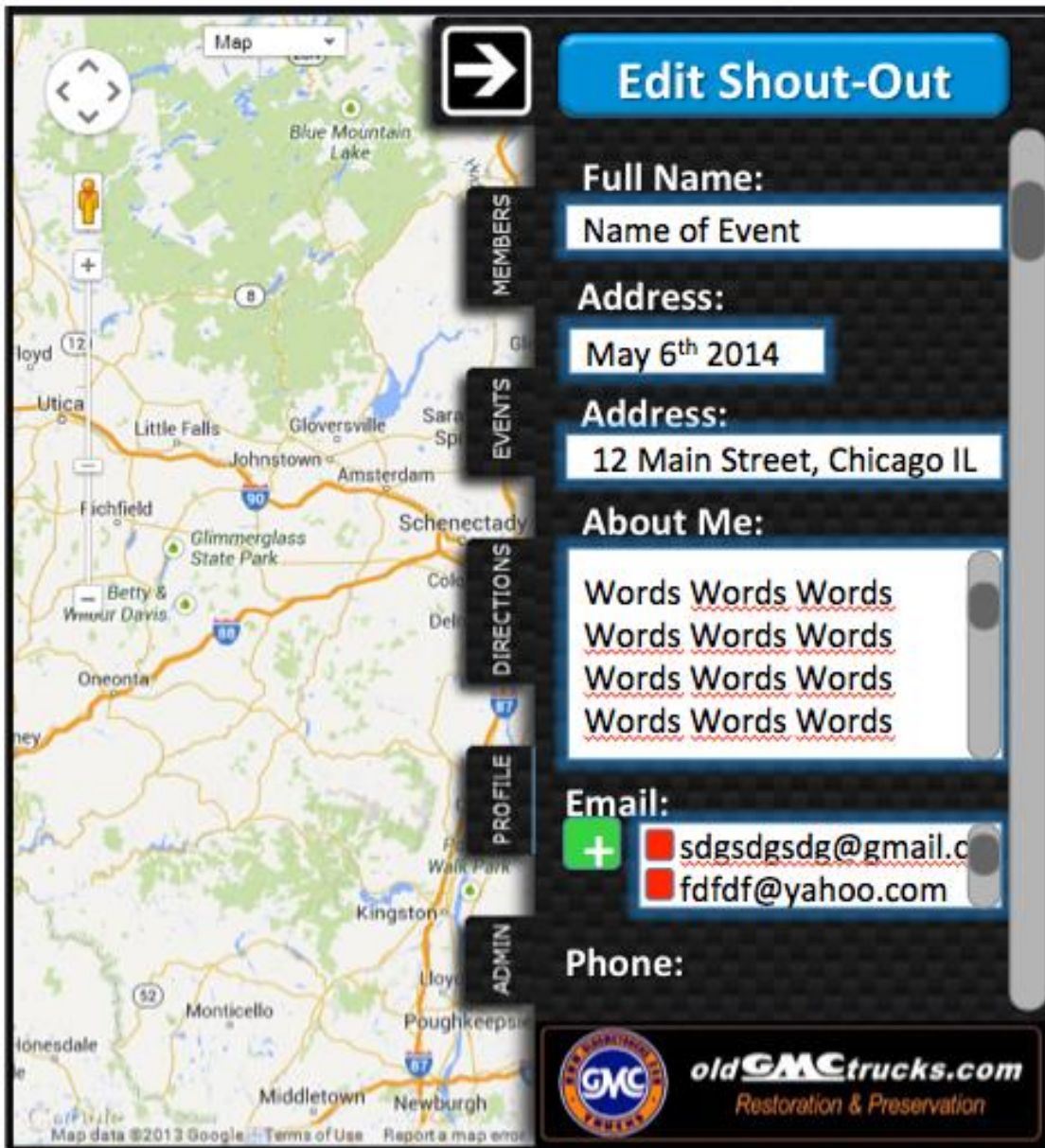
12.8 Plan a Trip

What a user sees after directions were entered



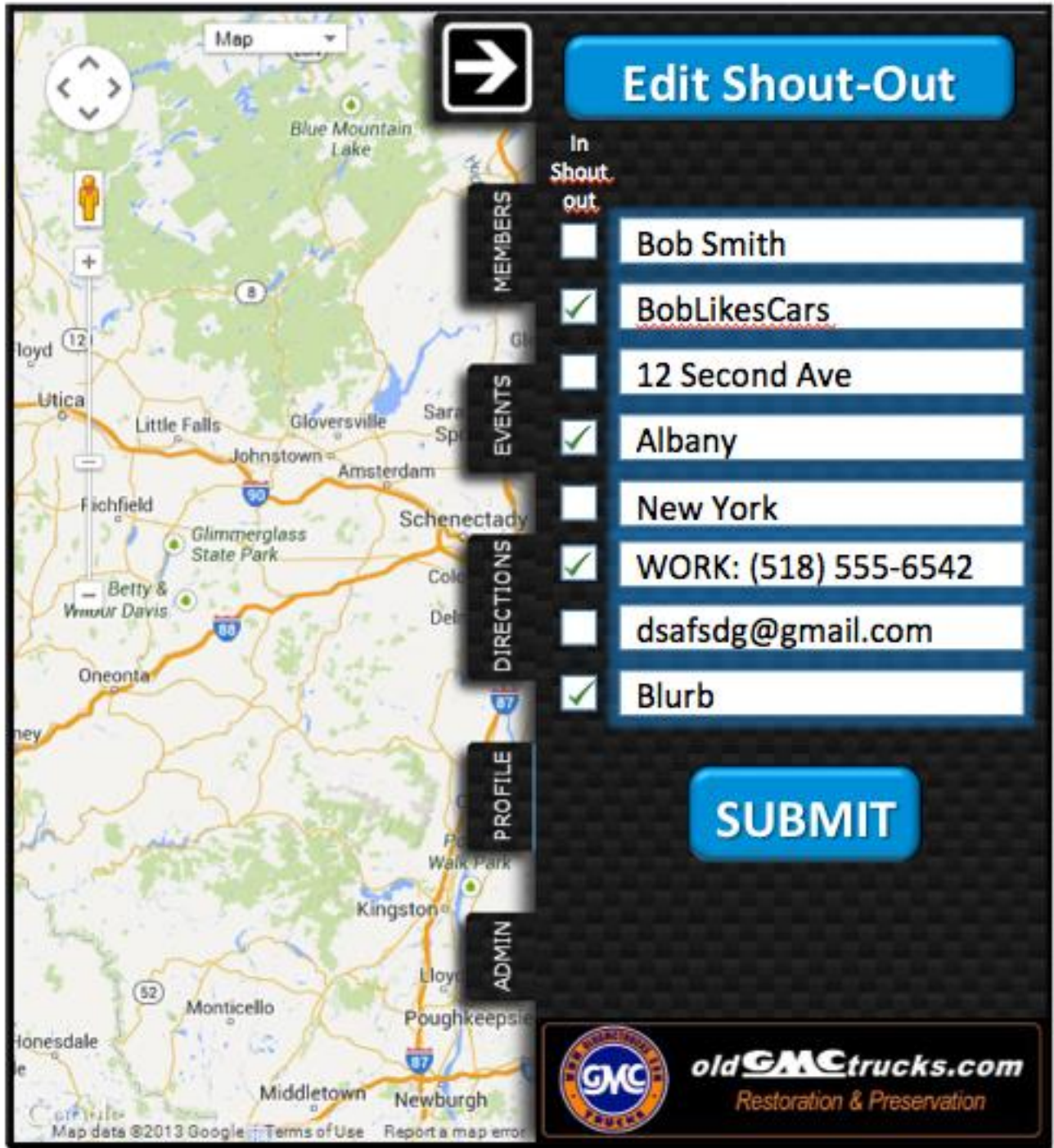
12.9 Edit Shout-out Prototype

What the user sees when they enter information in the shout-out



12.10 Edit Shout-out Privacy Prototype

What the user will see if they decide to change what other users can see in their shout-out



13. Testing Plan

13.1 Overview & Strategy

The oldGMCTrucks Mapper system will be tested on all four major browsers, including IE 6,7, and 8 as stated in the functional requirements. BitSize Solutions will test all the functional requirements of the oldGMCTrucks Mapper system. The functional requirements will be broken into modules and each module will be tested separately. Once a module is considered functional, the module will be tested via interactions with other modules. When this testing is complete, all the modules of the oldGMCTrucks Mapper system will be tested together as a whole system. The Detailed Design will document the testing plan in greater minutia.

BitSize Solutions will also test the non-functional requirements of the oldGMCTrucks Mapper system. The system will be tested for user-friendliness by reviewing new user interactions in the system. This will allow BitSize Solutions to examine how users go about performing tasks within the system.

13.2 Acceptance Test

The system will be considered acceptable under the auspices that the system will meet all of the functional and non-functional requirements. These are listed earlier in the document and are subject to modifications prior to the Acceptance Test.

13.3 Unit Tests

Unit tests will be tested separately. After each unit passes it's corresponding tests, BitSize solutions will run a test on the system that encompasses those units to make sure they interact appropriately. Please see the attached sheets for the unit tests for Login, View Map and Member List. The excel file containing the status of the unit tests is also available for download from the BitSize solutions website.

13.4 Test Cases

Each test case will be numbered and contain a description of; the unit involved, the action to be performed, the input entered by the user, the state before the test, and the expected result. Results of the tests will be recorded once testing has begun.

13.5 Exception Handling

BitSize Solutions will perform checks for all system exceptions and common user errors. System errors include database read and write errors, database connection errors and JavaScript errors. User errors are errors thrown by common invalid entries into the system. All errors will be handled to the furthest extent possible in oldGMCTrucks Mapper system. All errors conceived by BitSize Solutions will be

tested in the unit tests, which will be expanded as BitSize Solutions develops the system.

14. Development and Production Environments

14.1 Development Environment

Server

Operating System: CentOS 5.2, Kernal 2.6.18-92e15

Server Name: oraserv.cs.siena.edu

CPU: Intel Xeon 2.66 GHz

RAM: 8GB

Macintosh Computer

Operating System: OSX Version 10.7.4

Proc: Intel Core i5 @ 2.5GHz

Ram: 4GB 133 MHz DDR3

HDD: 500GB

Windows Computer

Operating System: Windows Vista

Proc: Intel Core 2 Duo E7500 @ 2.93 GHz

Ram: 4GB

HDD: 297GB

14.2 Operating Environment

The solution will be a web-based application by Web Hosting Pad. The web-based application will be accessible by oldGMCTrucks.com forum members.

14.3 Maintenance Environment

There should be minimal if any maintenance of the solution, however, if needed, the maintenance environment will include all the same software and hardware used in the Development Environment.

15. Appendix A: Cross Reference Index

oldGMCTrucks Mapper Context Diagram 5.2

oldGMCTrucks Mapper Level 0 Diagram 5.3

oldGMCTrucks Mapper Confirm Account Level 1 Diagram 5.4.1

oldGMCTrucks Mapper Login Level 1 Diagram 5.4.2

oldGMCTrucks Mapper Modify Profile Level 1 Diagram 5.4.3

oldGMCTrucks Mapper Modify Event Level 1 Diagram 5.4.4

oldGMCTrucks Mapper Interact with Map Level 1 Diagram 5.4.5

oldGMCTrucks Mapper Request Account Level 1 Diagram 5.4.6

oldGMCTrucks Mapper Manage Map Data Level 1 Diagram 5.4.7

16. Appendix B: Sources of Information

We derived the information in this document via various resources. Our largest source of information is Dr. Lederman. The structure of our document was built to the criteria our professor Dr. Lederman, specified in the grading scheme for the Requirements Specification. In addition, BitSize Solutions has consulted numerous Requirements Specification from previous years teams. Some other sources of information utilized in the creation of this document were the Google Maps API as well as various resources on the Internet pertaining to web development technologies.

17. Appendix C: Glossary of Terms

API – Application Programming Interface specifies how some software components should interact with each other.

Browser- Software application for retrieving, presenting and traversing information resources on the World Wide Web.

Extensible- a system design principle where the implementation takes future growth into consideration.

Gantt – a chart in which a series of horizontal lines shows the amount of work done or production completed in certain periods of time in relation to the amount planned for those periods.

HTML – HyperText Markup Language is the main markup language for creating web pages and other information that can be displayed in a web browser.

Input- Term denoting either an entrance or changes which are inserted into a system and which activate or modify a process.

Javascript – an interpreted computer programming language using client side web pages.

MySQL – the world's most widely used open-source relational database management system (RDBMS) that runs as a server providing multi-user access to a number of databases.

Shout-Out- Google Inc. has dubbed their onscreen pop up panels as “Shout-Outs”.

Web Hosting Pad – a web hosting site with support for server-side programming and front-facing web pages.

18. Appendix D: Timeline (Gantt Chart)

Task



A task is represented by a rectangle with rounded edges. The edges of the rounded rectangle represents the start and end date of a specific task.

Completeness Task



A completeness task is represented as a rounded rectangle with a portion that is filled in. This completeness task a visual representation of how much of a certain task has been completed. A fully shaded in task represents a completed task.

Milestone



Milestones are represented by a circle and represent a certain checkpoint in the project. A milestone is fully shaded if the previous task led to the completion of a certain checkpoint.

Dependency



Dependency is represented by an arrow from one task to another. A dependency arrow links two tasks, showing that the second task cannot be started without completing the first task fully.

Subtask



Subtasks are represented by a rectangle with pointed edges. Subtasks group many tasks together that are similar to each other. This is used to section similar tasks together, organizing the project for ease of reading.

