Bilkent University

Department of Computer Engineering

Senior Design Project

Brandlyser: Web Based Analyzes Systems

Analysis Report

Ali Şengül, Ferdi Aslan, Sera Büyükgöz, Tunahan Kunt

Supervisor: Hakan Ferhatosmanoğlu

Jury Members: Erman Ayday and Özgür Ulusoy

Web Page: www.tunahankuntt.wix.com/brandlyser

Analysis Report
Nov 20, 2015

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

Contents

List of Figures

Analysis Report

Brandlyser: Web Based analyzes Systems,

1 Introduction

In today's world, gathering information on internet before doing an activity becomes popular and reliable. Almost every brand in the industry need to be informed about their reputation. However, every day people complain different parts of their life to the world through internet such as social media, blogs, web sites etc. They complain about where they go, the quality of the product that they buy, their feelings and opinion about almost every topic. They also comments each other's complaints. That creates a huge amount of data on the internet. That huge amount of information awaits and ignores because of mess on the internet. Since that data is not classified or listed. To reach information about reputation of brand, that data need to be collected and classified according to the specific topic to be use in the industry.

Automobile industry in one of the topic that people mostly talk about on the internet. Brandlyser aims for gathering information about that brand and that brand's model via using Twitter and Facebook posts, using comments under the commercial videos and related other videos in YouTube, posts on the some popular Blogs and some complaints Blogs. Program synthesis that data then gives analyzes with statistic of market place of that brand on the internet.

2 Current System

The current analyze systems are unfortunately not taking advantage of power of web based public comments fully. Most of them are just taking into account the real sales figures. However, it is not fully efficient solution to analyze for 2015. This is why; following sales figures is a long process so it carries out an analysis after a long period.

On the other hand, some of the analyze tools are using web based comments but they are not effective for comparison of brands. Some examples of them.

RapidMiner: RapidMiner is a web based predictive tool for both community and professional. It do semantic web analysis both for sentences and paragraphs from finance to technology; business to art. RapidMiner's core functionality is it speaks of characteristics of product. However, it does not analyze each sentence. It just categorizes the sentences of paragraphs. Also it does not have a comparison feature.

Viki Core: In the Content Oriented business Viki Core is very powerful tool for understanding of text, dialogs and comments [1]. It extracts and understands information from natural language and outputs converted into XML and JSON. However as RapidMiner it does not have a comparison feature.

Brandyzer: Brandyzer is presenting tools to brands about what is speaking about them or their rivals. Brandyzer's customers can follow their advertisement campaigns' effects, they can follow their rivals' enterprises, they can make search with real time, they can focus on special networks, they can make comparison between past and present, they can select a time span and they can create archive for the results.

Radian6: Radian6 enables its customers to track, monitor and react to comments and questions according to demography, trends and targets. By Radian6 you can reach various sources from Twitter, Facebook, YouTube, blogs, news and more. Radia6 is one of the most popular analyze tools in the world.

Boomsonar: Boomsonar is a social media monitoring tool which provides social media following, measurement, and reporting and online prestige management to its customers. It can provide localized results for every country. It uses their own

SmartAlgorithm and it offers deep segmentation and categories for all the website results. Figure 1 shows user interface panel of Boomsonar application.

Figure 1: BoomSonar User Interface

SentoApp: Sento made sematic analysis in real time for twitter. It uses only twitter and keyword for analysis. It lets user to filter results according to companies, brands, products, place, etc. It also can detect emotion in conversation. However, sento can only understand the way of twitter talks [2].

Even if each application has different features and from some ways they all have powerful functions, the main lack of all of them is they do not have a superlative analyze function for comparison of brands. Therefore, we believe that our application will be more long wearing than them

3 Proposed System

3.1 Overview

Brandlyser is a web - based application that helps user to analyze brand values and comparisons through the view of customers on the Internet.

Brandlyser has a brand description screen. In that screen, Brands products, industries, mottos and their competitors are entered manually. These creates keywords. Then according to keywords, data are gathered from different web pages on the Internet. Also that data are classified as their types, such as social media data, commercial data, blogs data etc.

During collecting of data, texts automatically enters our system which will design and analyze these texts. Our algorithm fundamentally will do sentimental analyses and will utilize superlatives in the text for comparison of brands. Other part of the algorithm will do text mining and sentimental analyses to find user opinions about topic and will turn them into a statistical chart.

Finally, Brandlyser will give result of these analyses in a dash box. Dash box, will be a page which is divided into smaller boxes and every box gives a unique part of web site analyze in a general form. To illustrate that, one box gives result of analysis about YouTube commercials data and other box gives the result of analysis on the social media data. Dash box also contains a general analysis which are combined with all little boxes.

After clicking that small box in the dash box, the new page will be opened and that gives more detailed analyze report of that web page. The detailed analysis includes people's opinion ratio of specific attributes of that brand such as, for an automobile company that attributes are price, comfort, performance etc. We will seek ways to incorporate topic modelling algorithm to achieve this. If there are any, they are listed at the bottom of the page with name of "Top Talked Car Models". After clicked the name of the model, it directed to the other page. That page will

give more specified analysis about mostly said model of that company and statistic about that model.

User also can ask any request to the admin by using “Ask Request” button which is exist at the bottom of Dash Box Page. That button opens Ask Request Page. In that page, user can send their request about their analysis and their wishes to admin directly. Admin reads that request in Request Page and change requests status when they apply that request.

3.2 Functional Requirements

- Brandlyser will have two levels of access: client level access and administrator level access.
- Client level access provides required reports to customers.
- Brandlyser will have two separate interfaces for client level access and administrator level access.
- Administrator level access provides activating or deactivating of any module for any customer.
- Administrator level access can enter and save information about new customer.
- Administrator level access can create new customer/User.
- Client can ask for a new request to Admin.
- Admin can answer and apply new request.
- Admin update status of request when apply client’s request.
- Brandlyser needs a login system for authorization. It will be doing by password.
- Brandlyser will provide different modules for different customers.
- Brandlyser will be a web-based application.
- Brandlyser will use YouTube, Social Medias APIs and many web sites for fetching data.
- Machine learning algorithms will be used for decreasing error rate of analyzes.

- Topic modelling algorithm will be used for determine topics of texts.
- Superlatives will be used in a reliable algorithm and catching superlatives in texts will give result us about comparison between brands.
- Brandlyser collects processed data and by classifying this data, it will show a proportional popularity of brands.
- Brandlyser will show the statistics of analyzed data as a graphic and chart.
- If clients desired to have their rivals' data, it will be fetched, analyzed and reported.
- Job scheduler will be used for constant fetching, analyzing and reporting processes.
- System will close clients' accounts when they do not make payment on time.
- Turkish and English languages will be supported in the first version of our software.

3.3 Nonfunctional Requirements

- The system will consist of user-friendly interface for usability of Brandlyser. Elegant design of Brandlyser one of the most important features of our software.
- Brandlyser one of the most important features of our software.
- Brandlyser needs an internet access to work correctly.
- Brandlyser will be reliable.
- Control mechanism of Brandlyser will have low response time and fast transition between pages.

3.4 Pseudo Requirements

- Implementation of Brandlyser will be done in Java for back-end development (analyze of data, API connections etc.)
- Some of the graphic objects will be designed using Adobe Photoshop CS4.
- As Brandlyser will be web based application, our application will be compatible with most of the operating systems.

3.5 System Models

3.5.1 Scenarios

3.5.1.1 Saves Description of Brand

Actor:	**Admin**
Entry Condition:	Admin has entered the website and logged into the system.
Exit Condition:	Admin saves information on the page.
Main Flow of Event:	1. Admin opens Brand Detailed Specifications Page. 2. Admin enters Brand's Name, Motto, Rivals Keywords and Models. 3. Admin selects requested features. 4. Admin selects web sites which uses for analysis. 5. Admin clicks save button and saved entered information about Brand.
Alternative Flow of Event:	- Admin can fill boxes in any order. - All boxes do not need to fill in the same time. - Already saved event can be accessible and modified again.

3.5.1.2 Sending Data to Algorithm

Actor:	**Admin**
Entry Condition:	Admin has entered the website and logged into the system.
Exit Condition:	Admin starts search.
Main Flow of Event:	1. Admin opens Brand Detailed Specifications Page. 2. Admin filled all boxes. 3. Admin clicks save button to save new information. 4. Admin clicks search button and starts search.
Alternative Flow of Event:	- Admin can open half-filled page and finalized that information then can do event 3 and 4.

3.5.1.3 Register User

Actor:	**Admin**
Entry Condition:	Admin has entered the website and logged into the system. Admin also need to enter some information about Brand to the Brand Detailed Specifications Page.
Exit Condition:	User added to registration system.
Main Flow of Event:	**1.** Admin opens Registration Page. **2.** Admin filled Company's name. **3.** Admin gives username and password for user. **4.** Admin clicks Create User button for adding user to the registration system.

3.5.1.4 Ask Request

Actor:	**User**
Entry Condition:	User has entered the website and logged into the system. User also need to opens DashBox Page.
Exit Condition:	User send request.
Main Flow of Event:	**1.** User need to click Ask Request Button at the bottom of the page. **2.** User transferred to the Ask Request Page. **3.** User need to filled form in the Ask Request Page. **4.** User need to click Send Button to send request.

3.5.1.5 Answer Request

Actor:	**Admin**
Entry Condition:	Admin has entered the website and logged into the system.
Exit Condition:	Admin do modification according to the request.
Main Flow of Event:	**1.** Admin opens Request Page. **2.** Admin reads requests to the specific customer. **3.** Admin goes Brand Detailed Specifications Page and add features which is specified in the request. **4.** Admin come back and ticks status of request in Request Page.
Alternative Flow of Event:	- Admin can read request and do modification in different times so ticks specify to modification be done.

3.5.1.6 Users Usage

Actor:	**User**
Entry Condition:	User has entered the website and logged into the system.
Main Flow of Event:	**1.** User opens Dashbox Page. **2.** User looks their analysis. **3.** User clicks each boxes to reach detailed analysis of box's web sites.
Alternative Flow of Event:	- User can clicks Ask Request button at the main Dashbox Page to ask a request.

3.5.2 Use Case Model

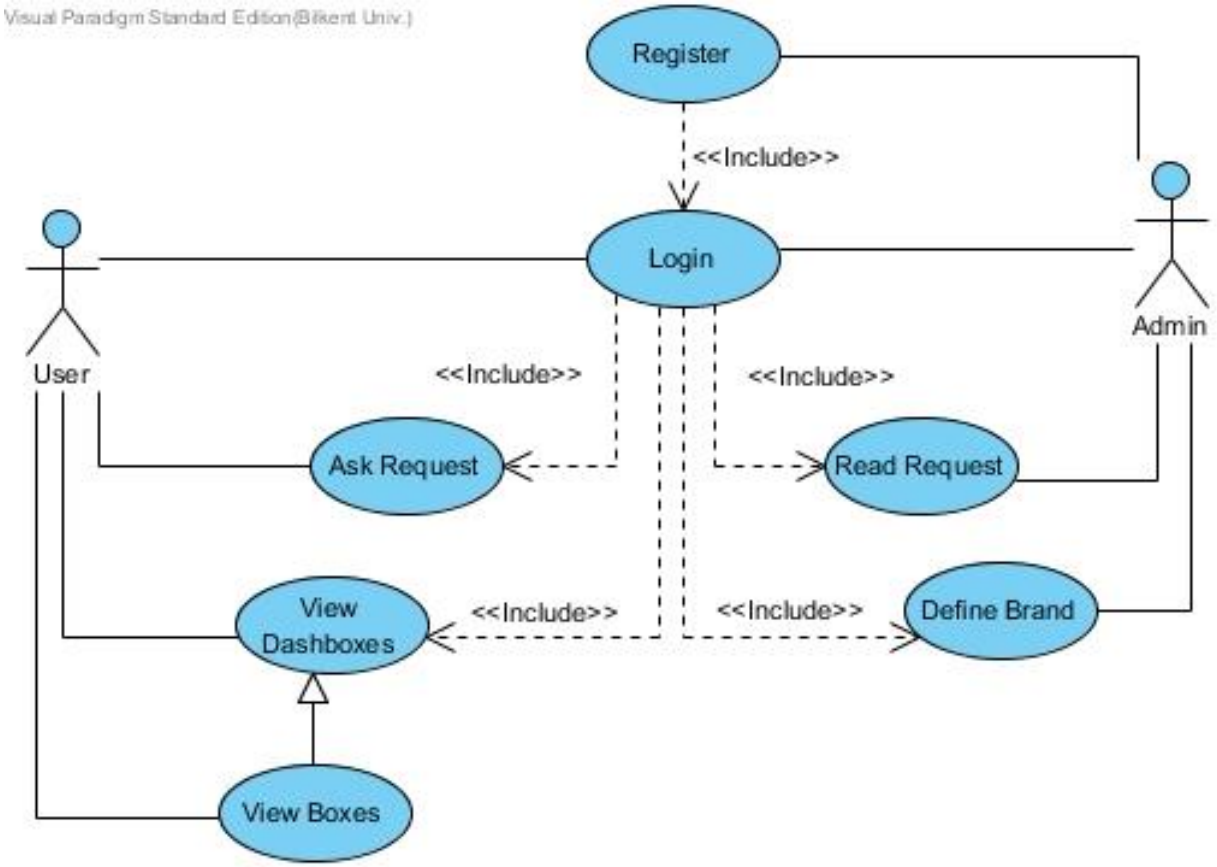

Figure 2: Use Case Model of the system

3.5.3 Object and Class Models

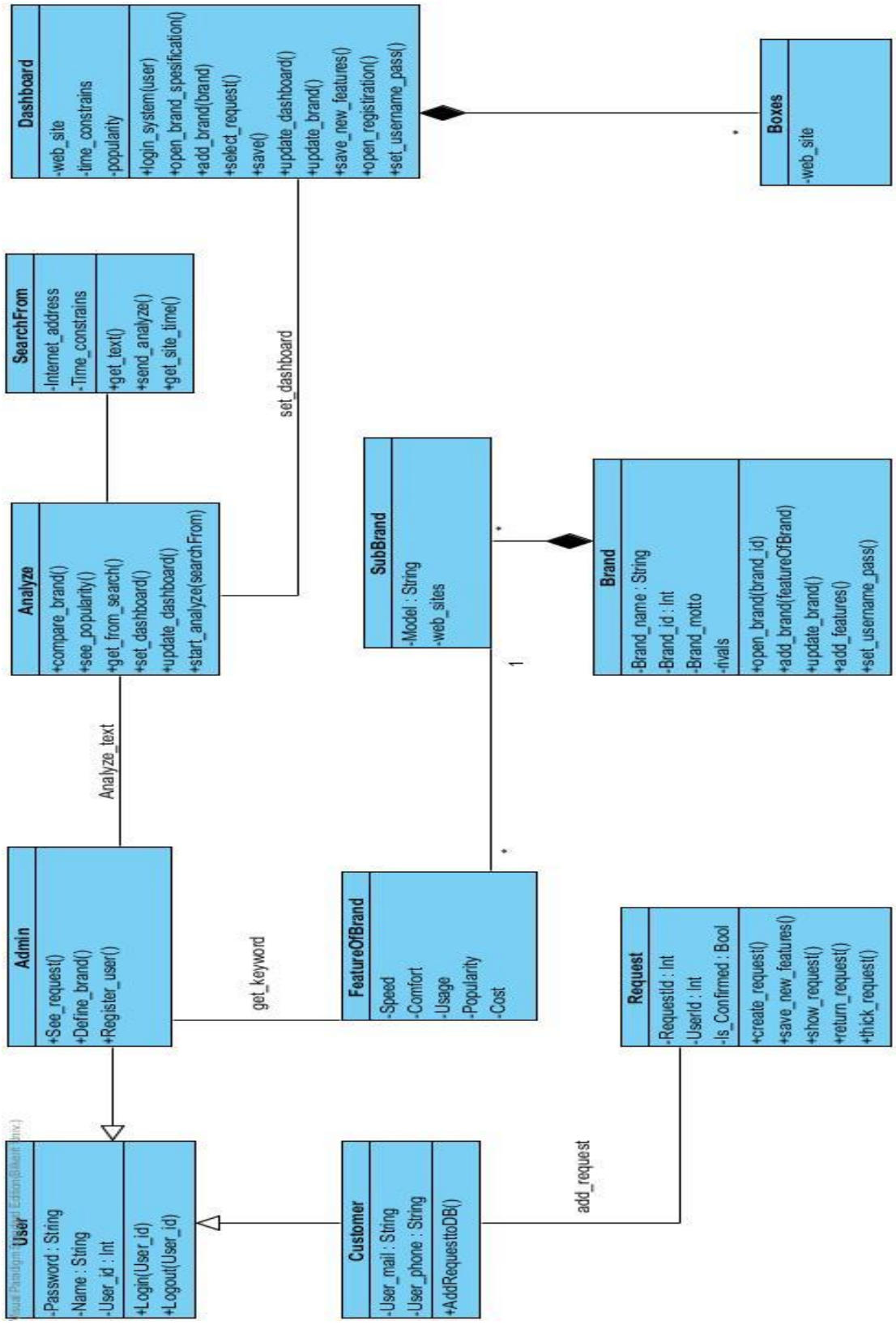

Figure 3: Class Diagram of Brandlyser

3.5.4 Dynamic Models

3.5.4.1 Activity Diagram

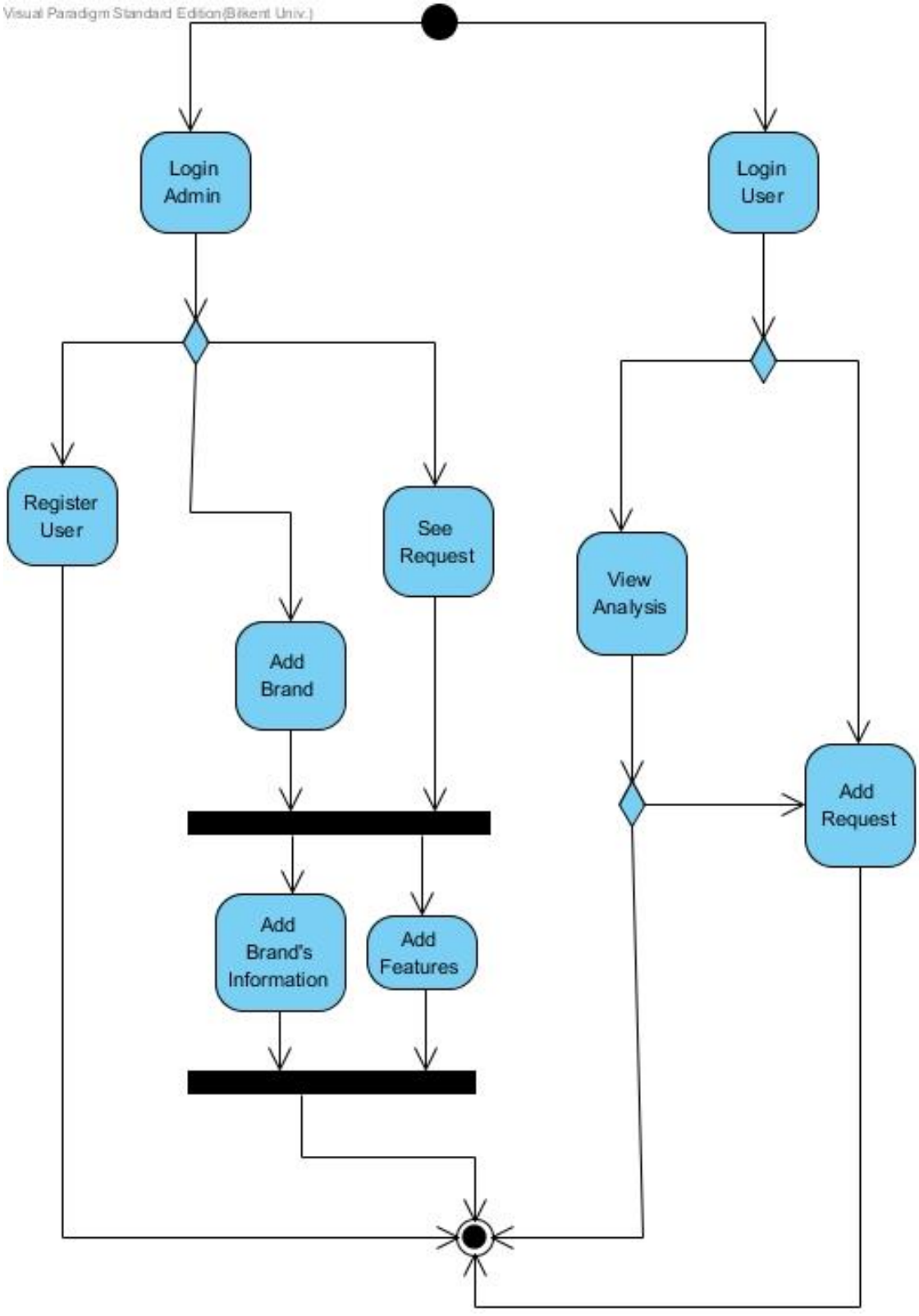

Figure 4: Activity Diagram of System

3.5.4.2 Sequence Diagrams

3.5.4.2.1 Saves Description of Brand

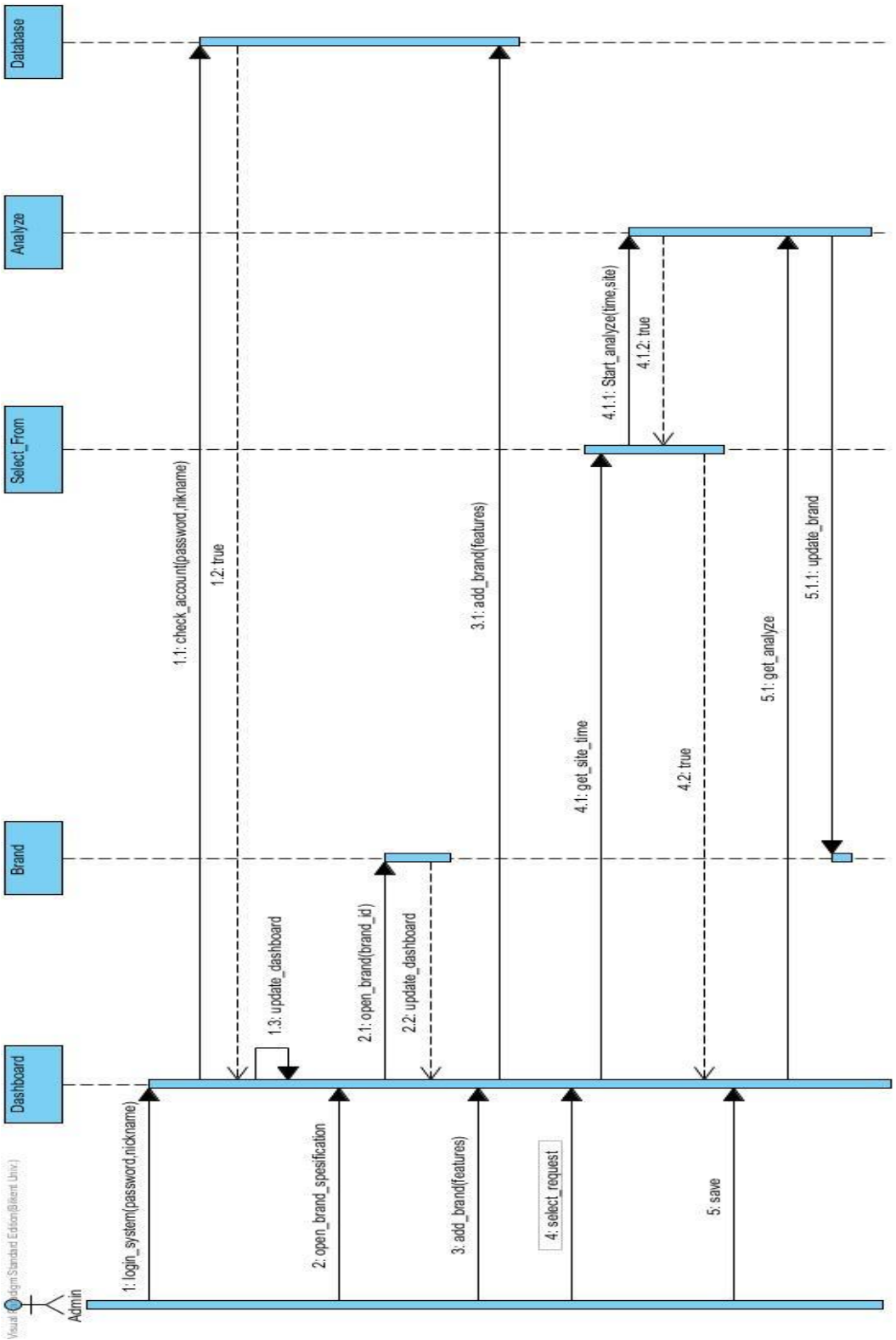

Figure 5: Sequence Diagram of save Brand description

3.5.4.2.2 Sending Data to Algorithm

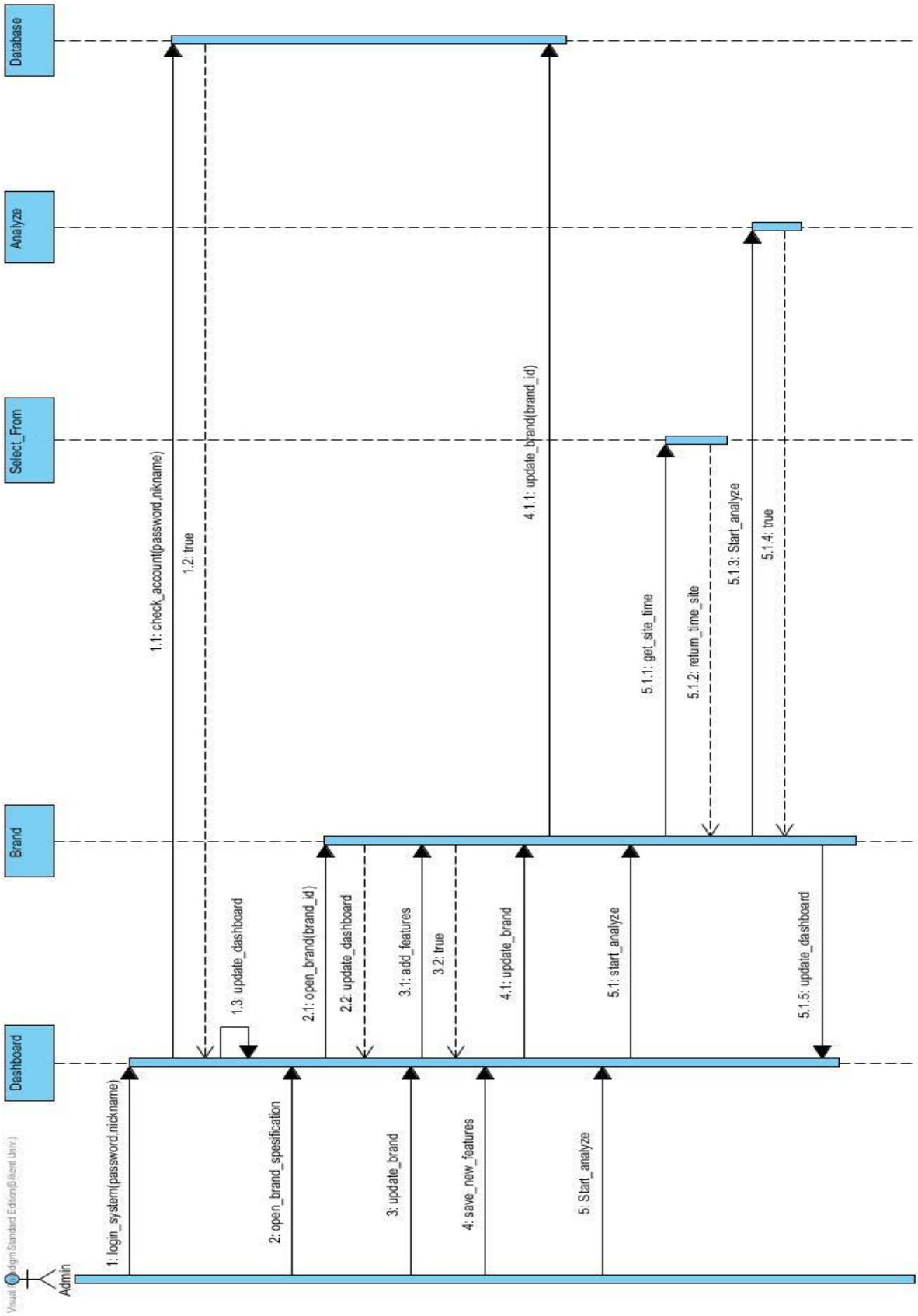

Figure 6: Sequence Diagram of sending Data to Algorithm

3.5.4.2.3 Register User

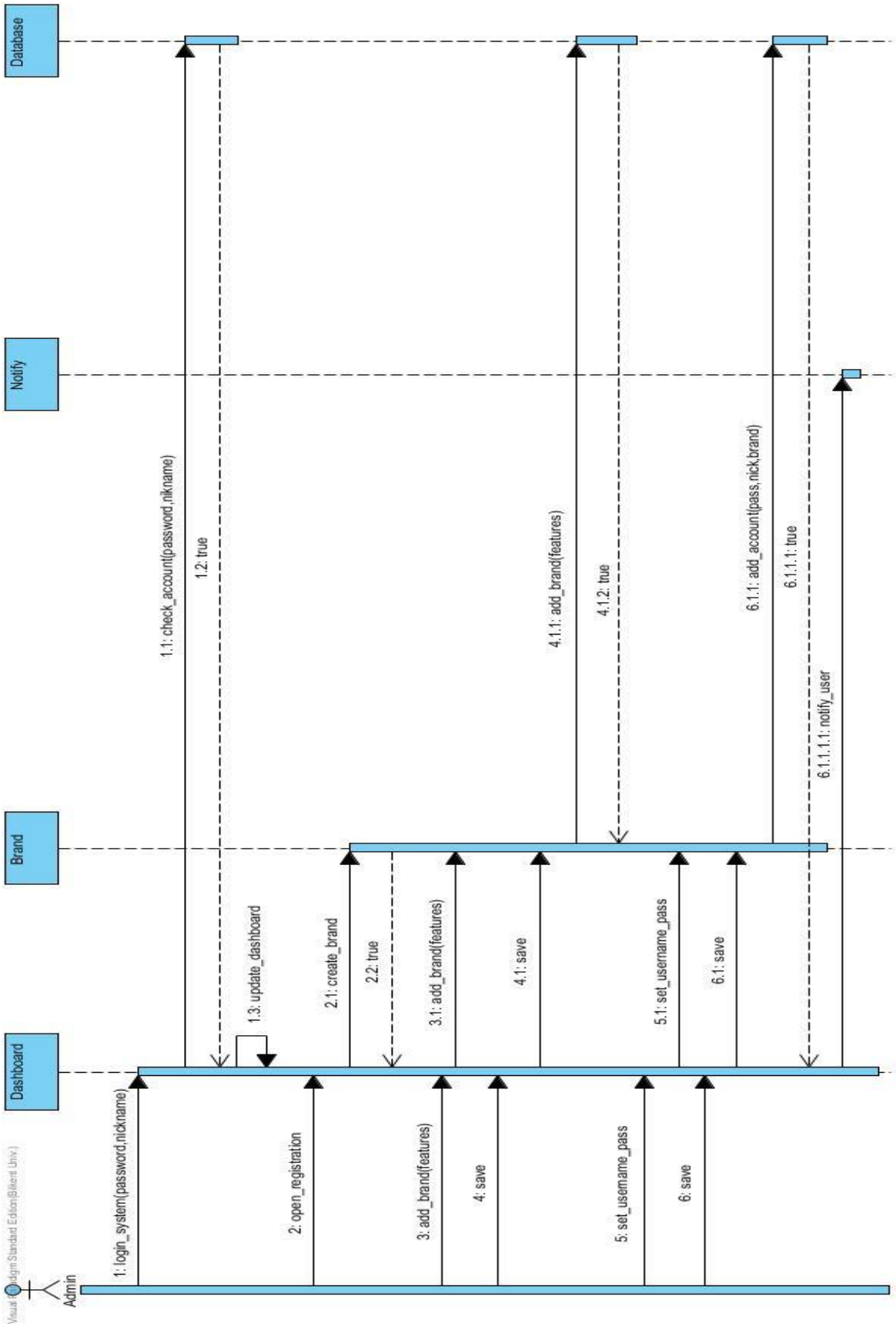

Figure 7: Sequence Diagram of Register User

3.5.4.2.4 Add Request

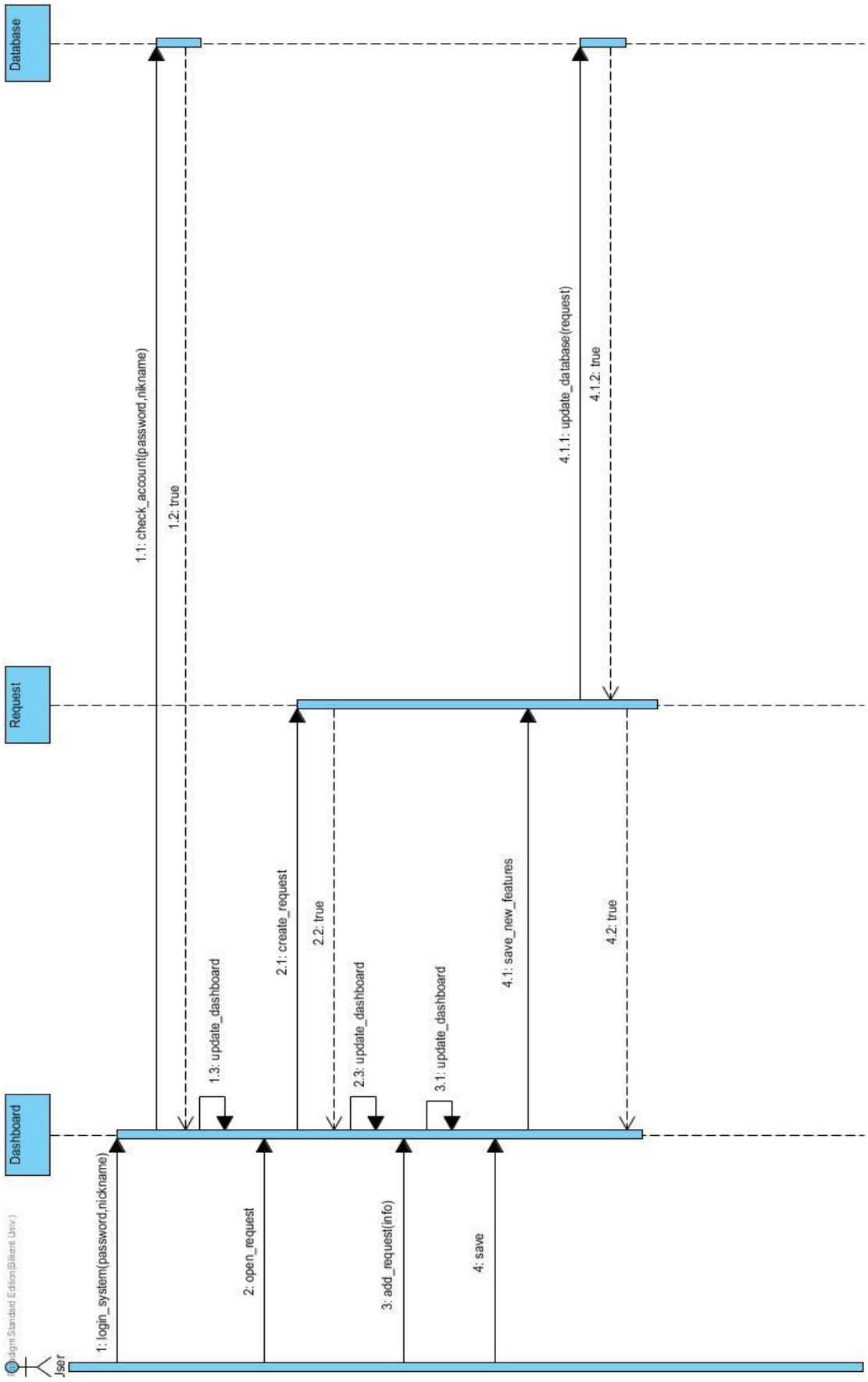

Figure 8: Sequence Diagram of Add Request

3.5.4.2.5 Answer Request

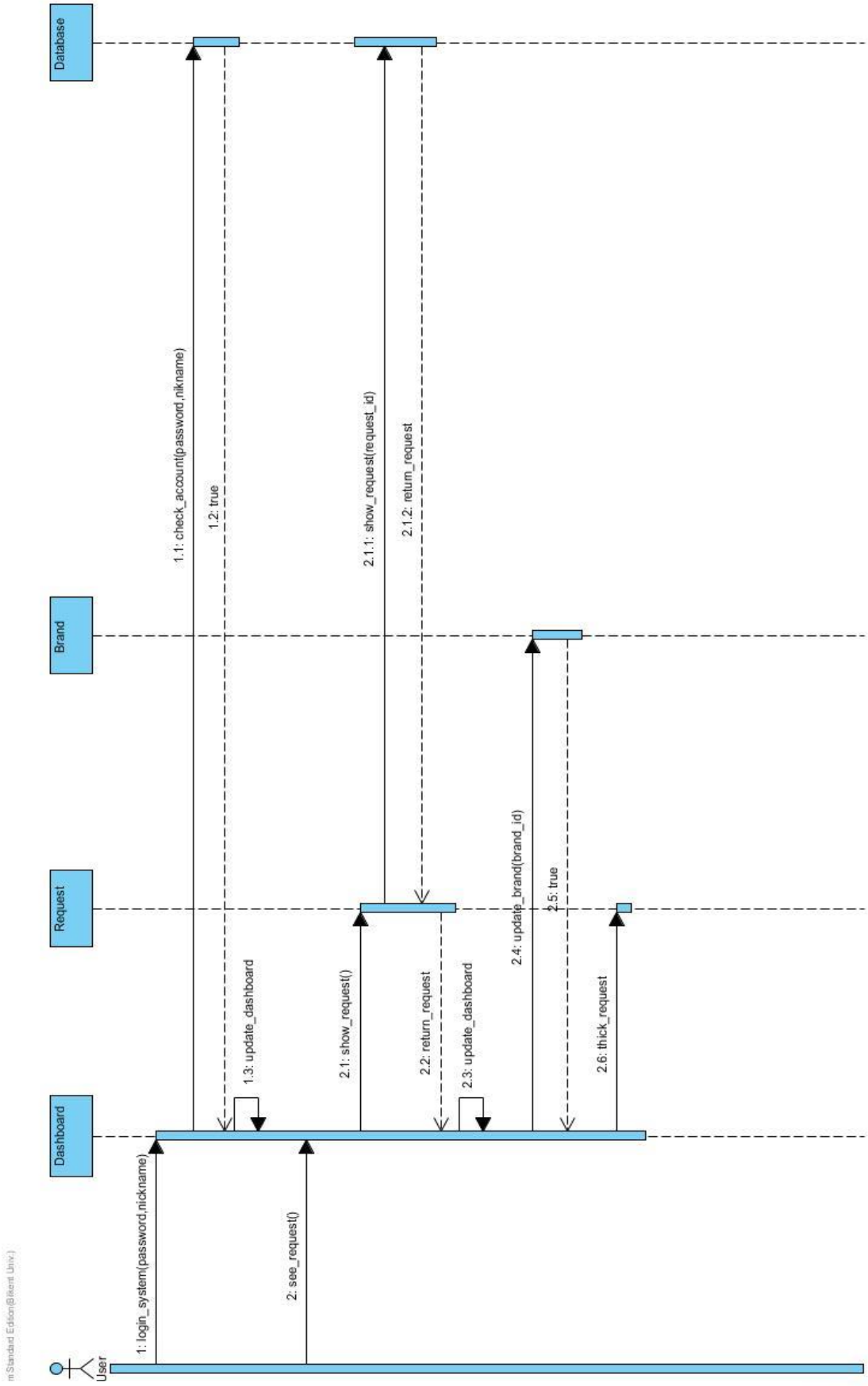

Figure 9: Sequence Diagram of Answer Request

3.5.4.2.6 User Usage

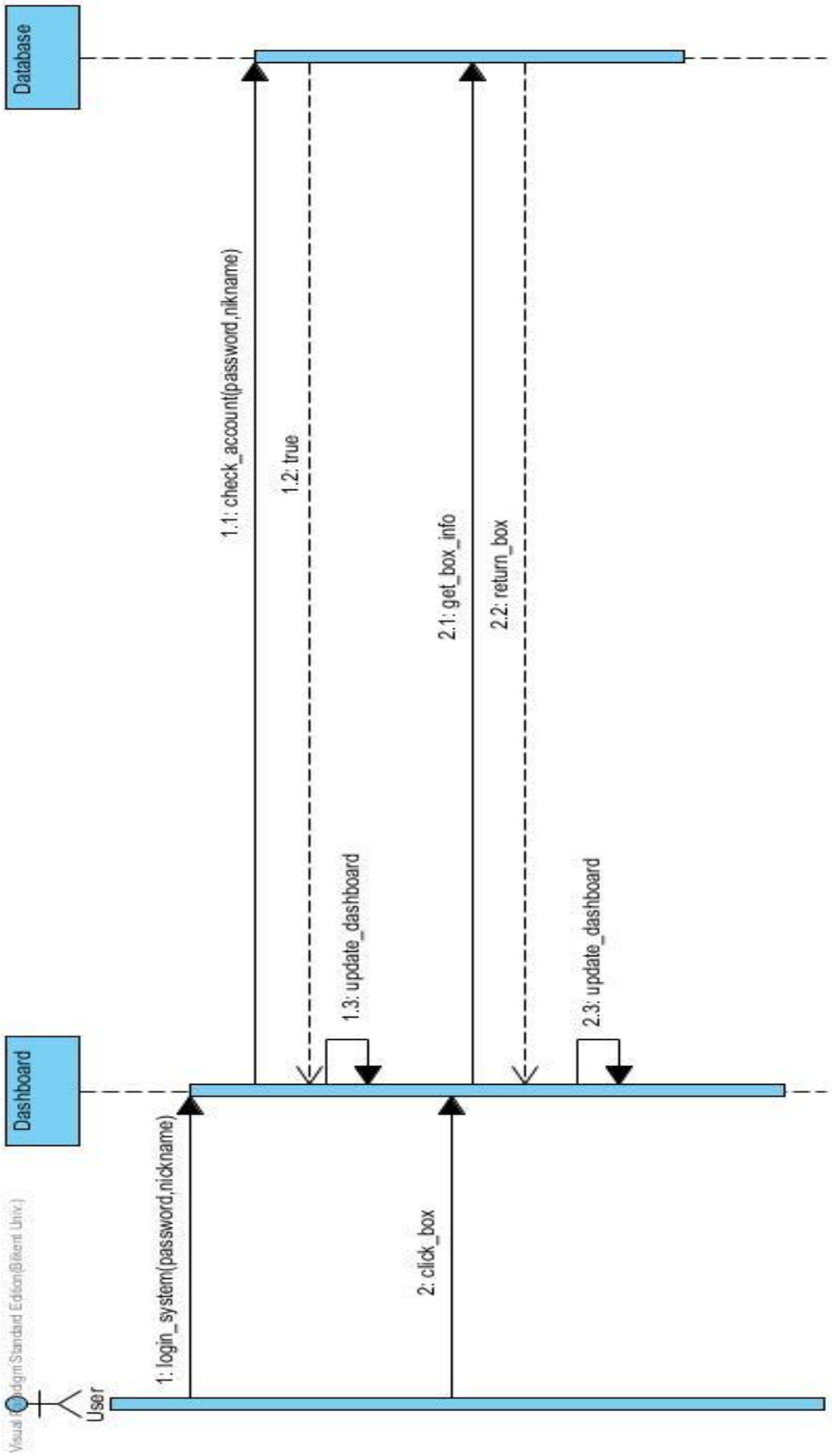

Figure 10: Sequence Diagram of User Usage

3.5.5 User Interface

3.5.5.1 Login Page

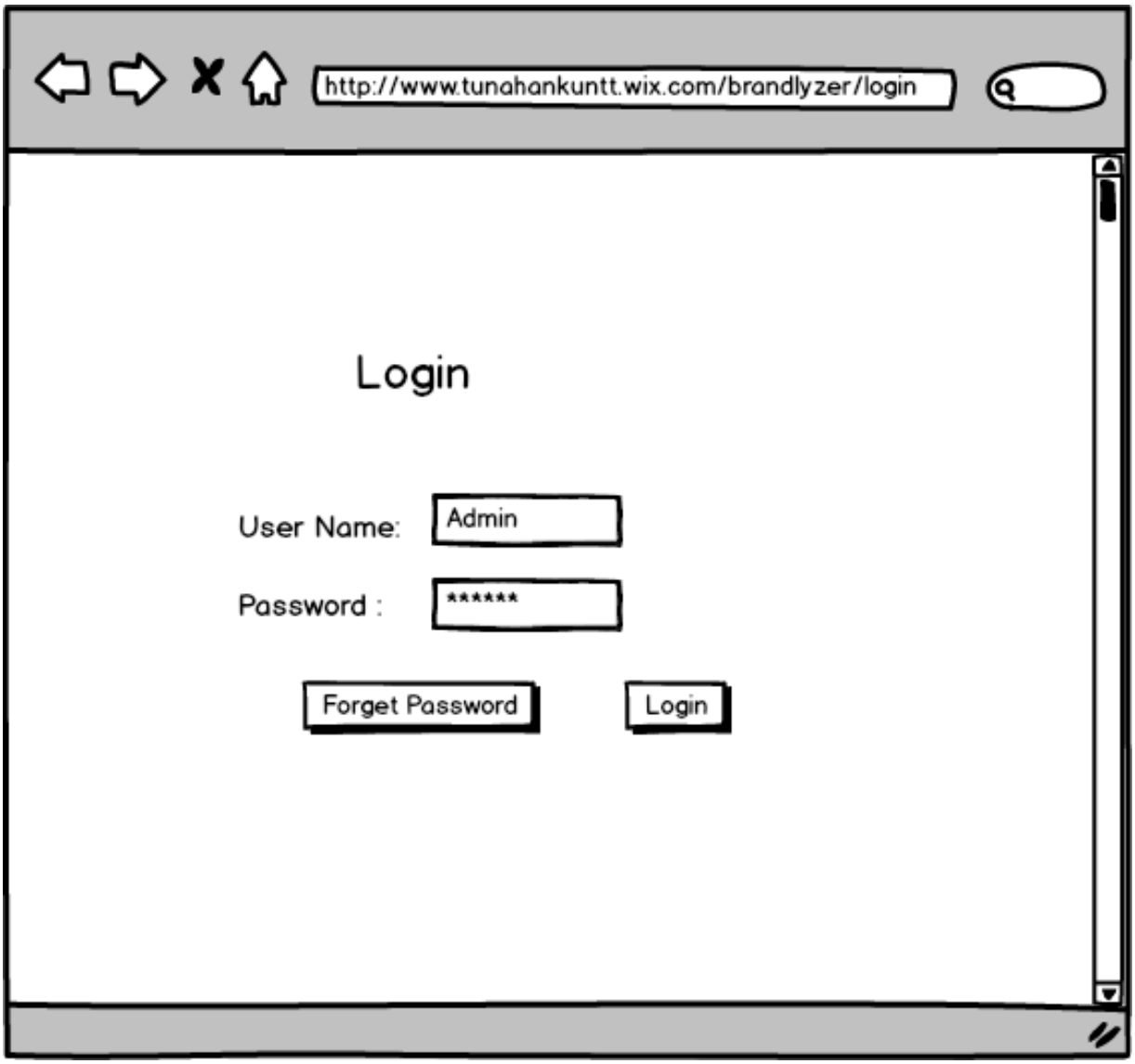

Figure 11: Mock-up of Login Page

3.5.5.2 Registration Page

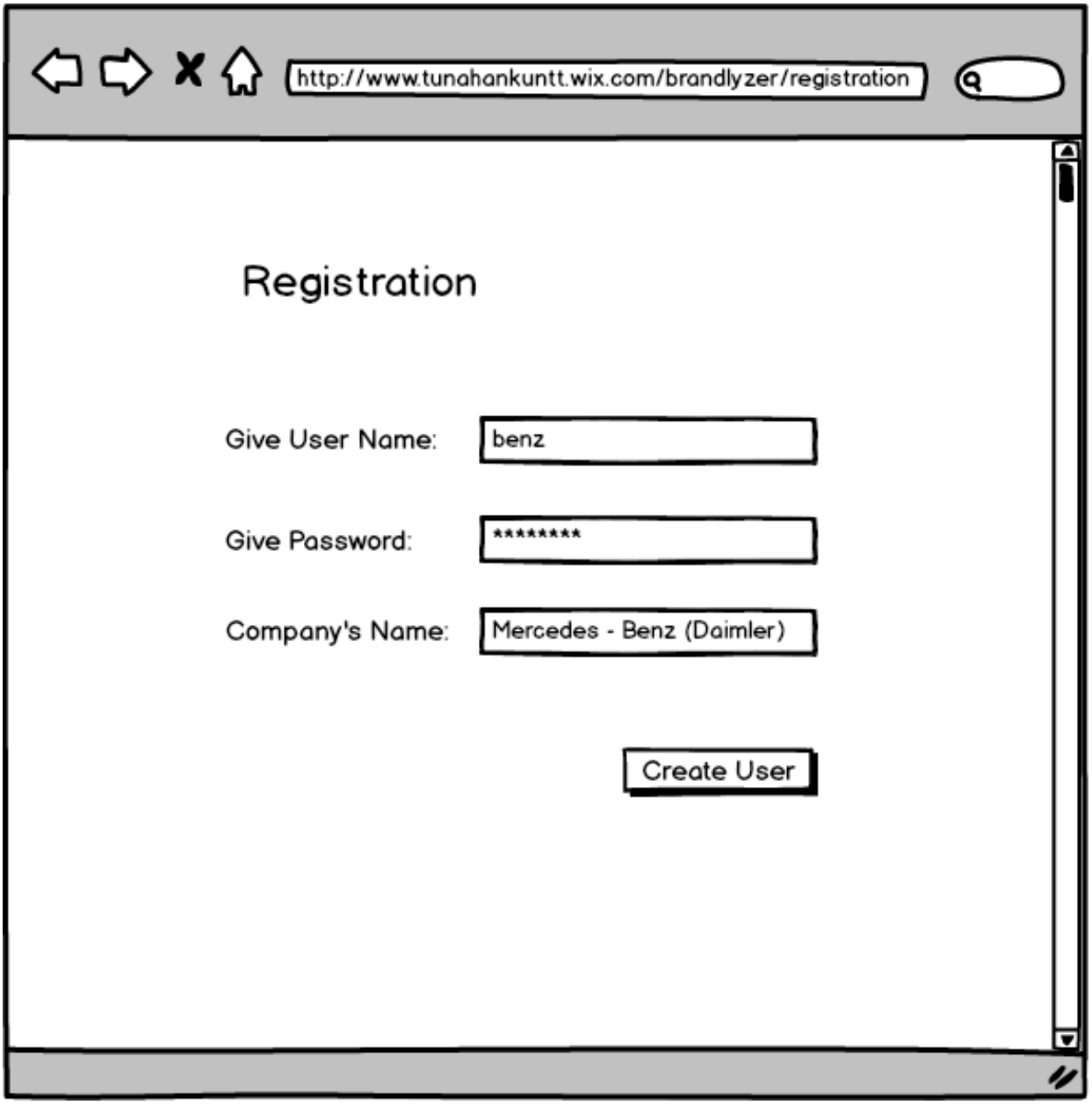

Figure 12: Mock-up of Registration Page

3.5.5.3 Ask Request Page

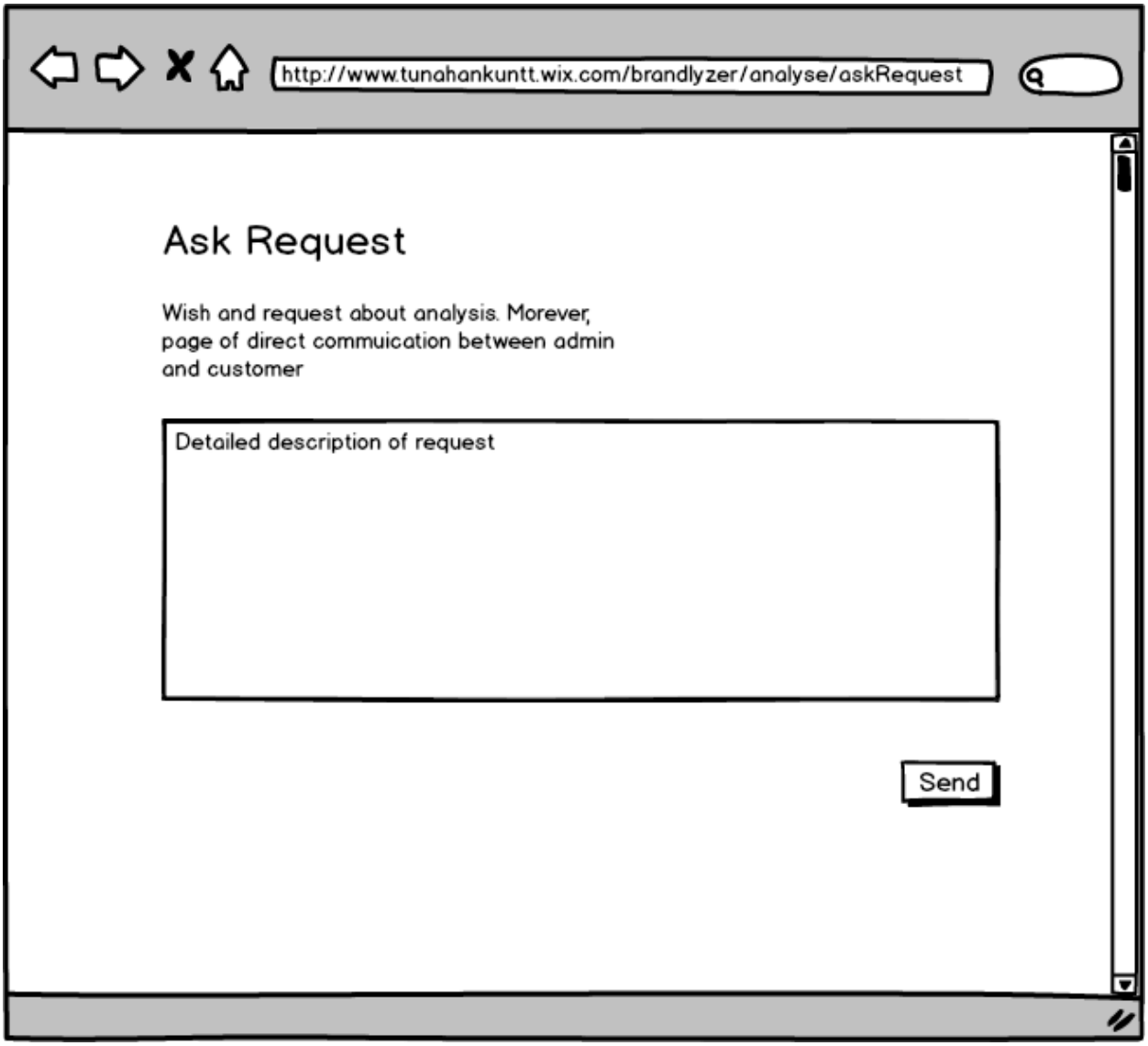

Figure 13: Mock – Up of Ask Request Page

3.5.5.4 Read Request Page

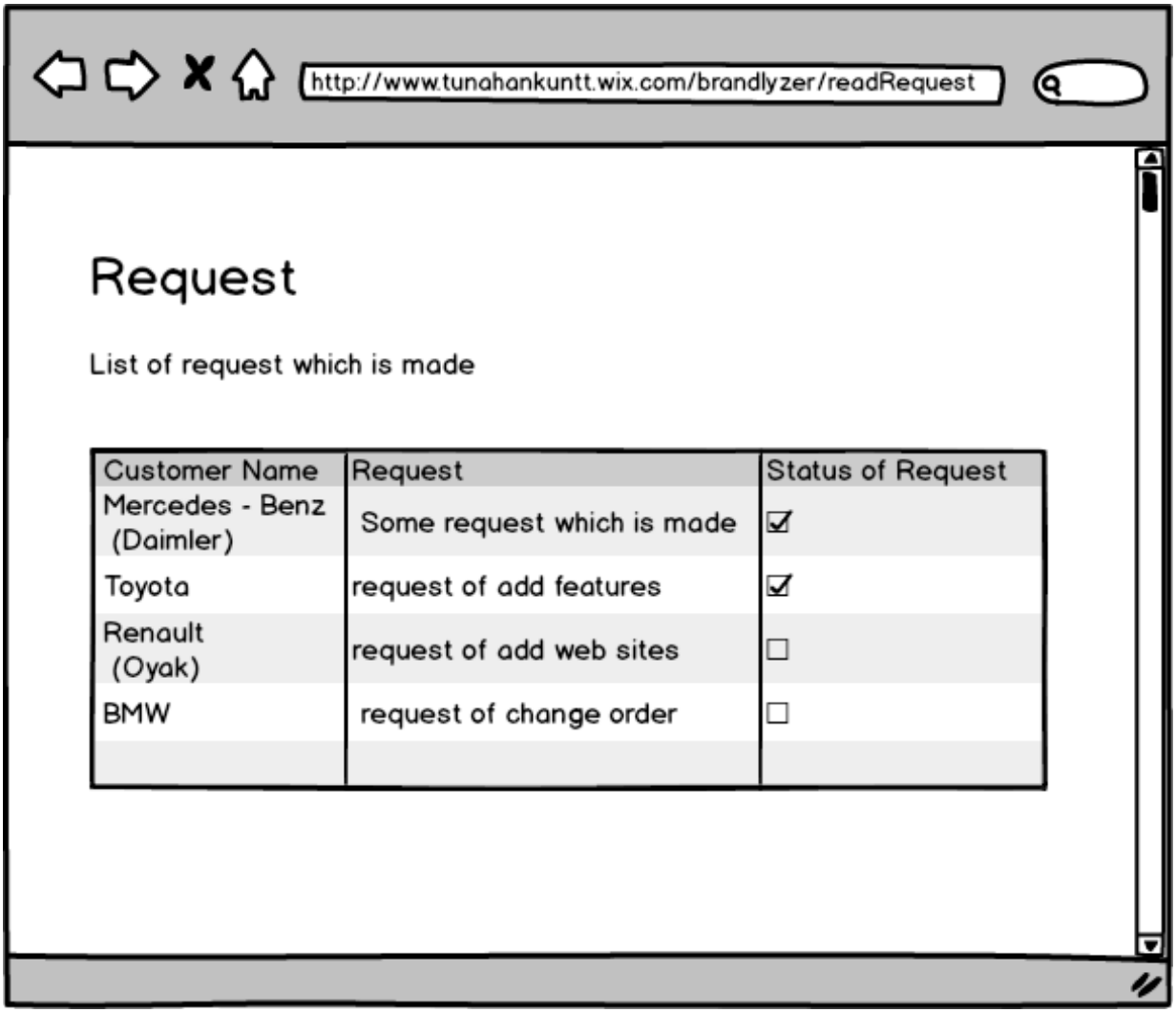

Figure 14: Mock – Up of Request Page which admins tracks requests

3.5.5.5 Brand Detailed Specification Page

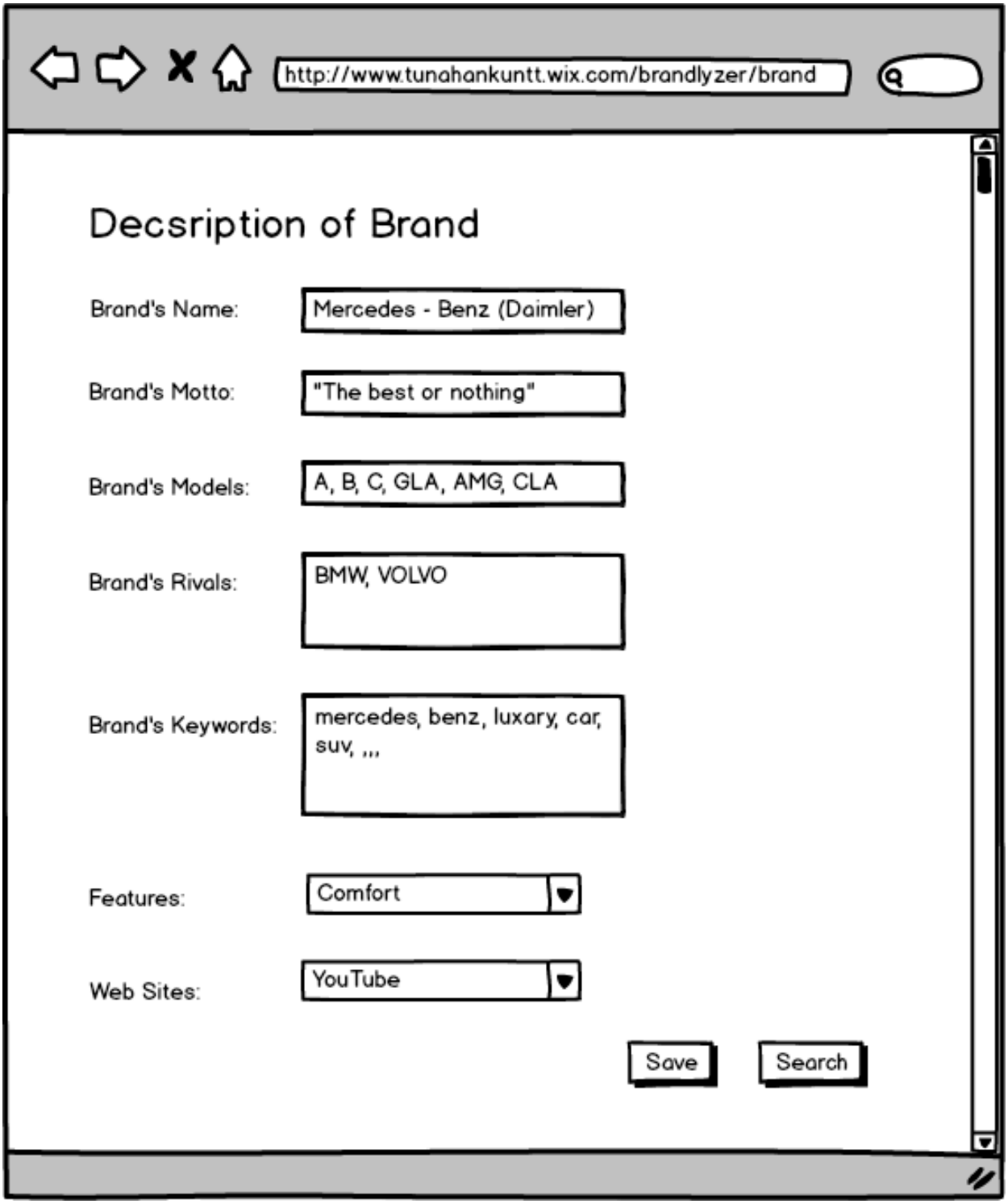

Figure 15: Mock – Up of Description of Brand Page which Brands are detailed specifically

3.5.5.6 Dash Box Page

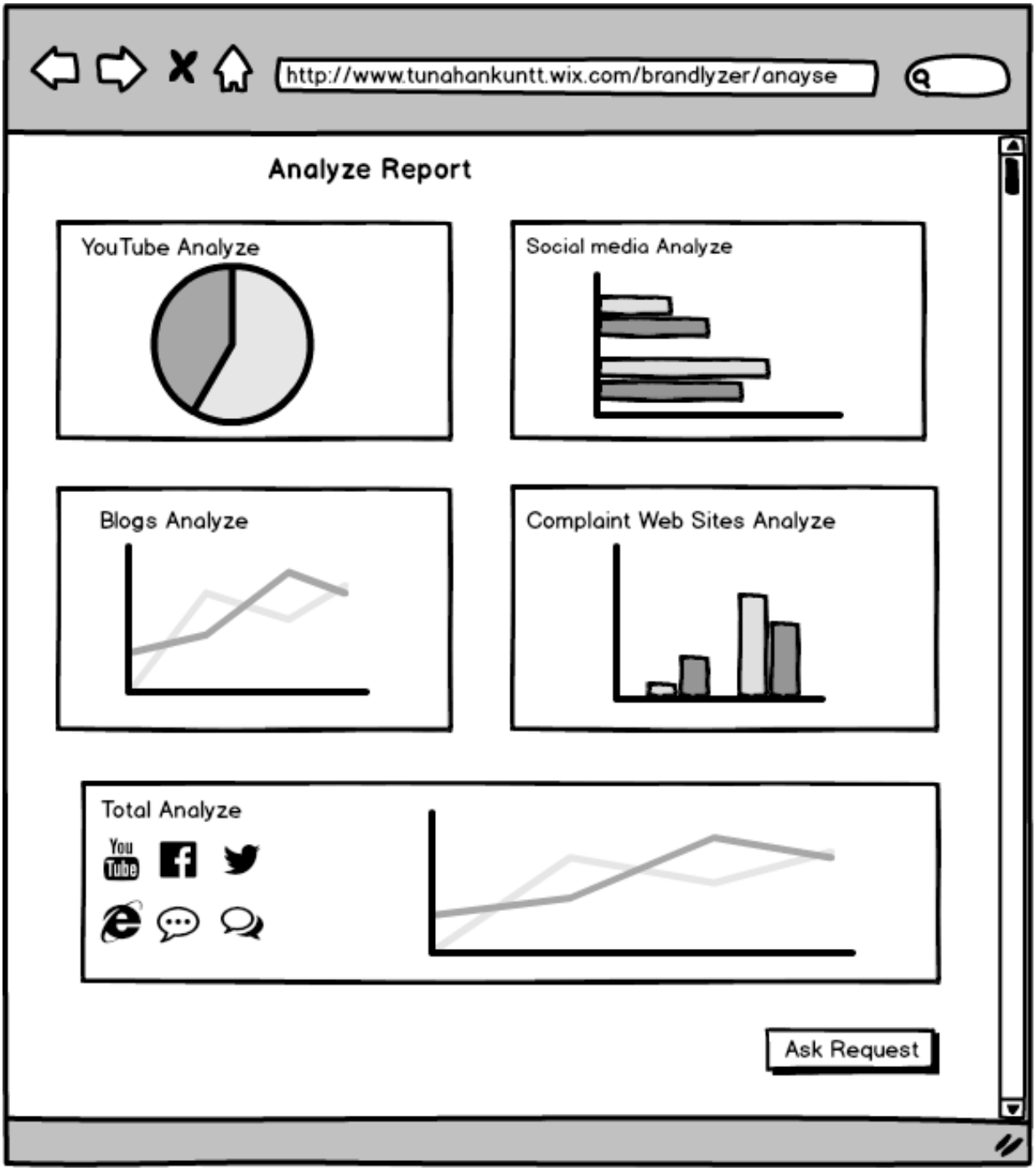

Figure 16: Mock – Up of Dash box which is Analysis Report.

3.5.5.7 Detail of Boxes in Dashbox Page

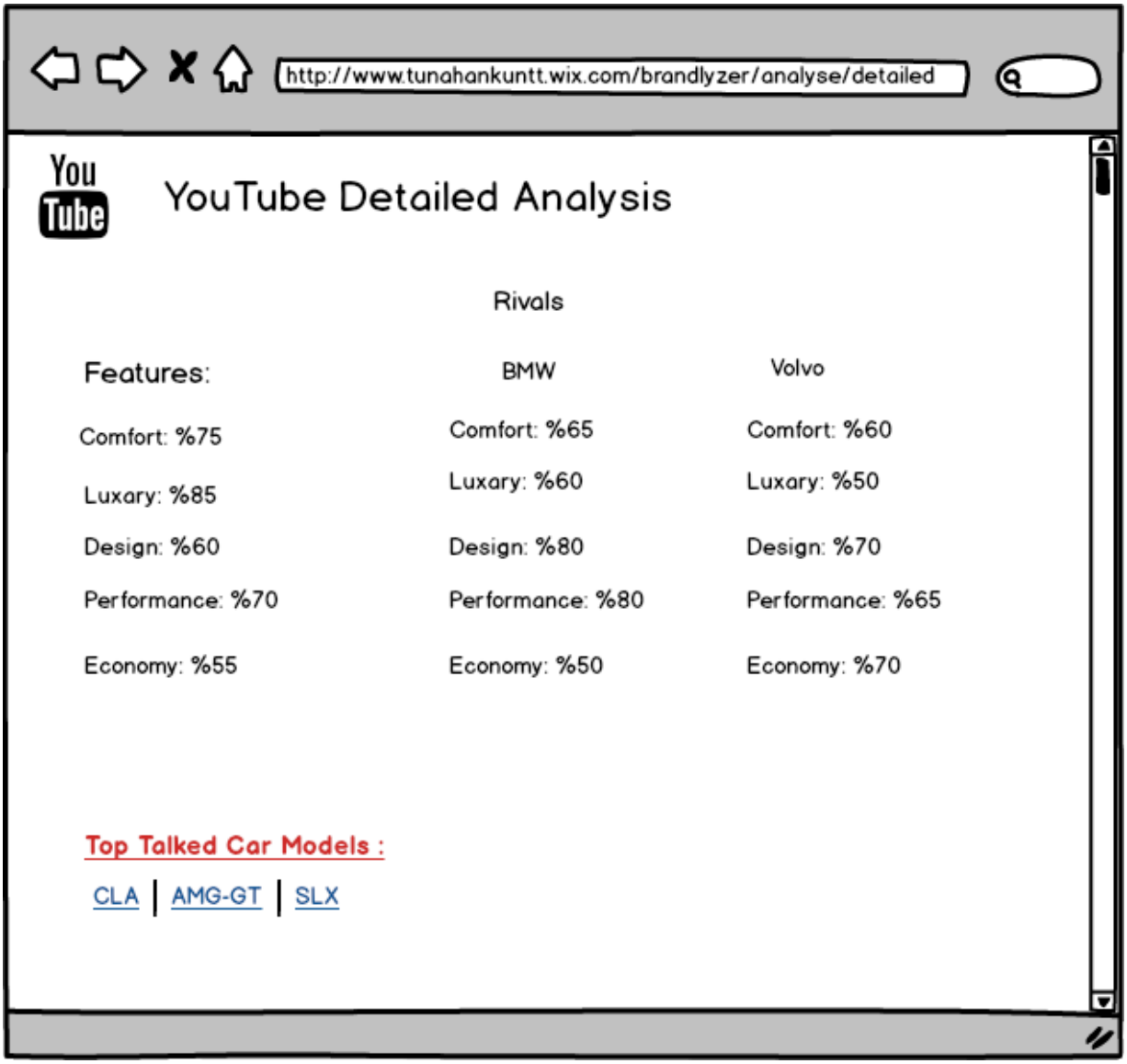

Figure 17: Mock – Up of Detailed Analysis which is detailed analysis of specific boxes.

4 References

[1] Softwareevolution.it, 'VIKI Core API - semantic analysis api', 2015. [Online]. Available:http://www.softwareevolution.it/en/products/viki-core-api.html. [Accessed: 31- Oct- 2015].

[2] "Sento | Thousands of Tweets Analyzed for Sentiment and Meaning in the Blink of an Eye," *Sento App*. [Online]. Available at: http://sentoapp.com/#features. [Accessed: 2015].