HUMAN BODY TRACKING SYSTEM

Software Requirements Specification

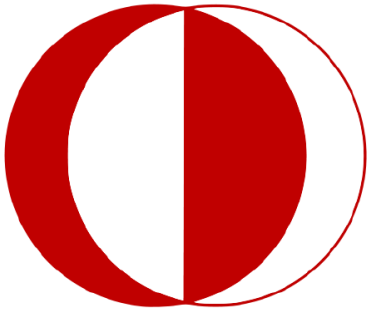
30.11.2014

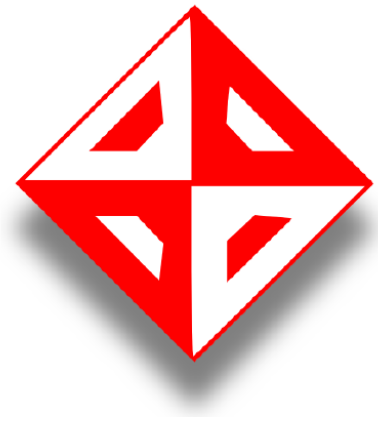
Zehra Deniz Çelik

Burak Araz

Cem Aydın

Yalçın Savrun





Revision History

Date	Revision	Comment
30.11.2014	1.0	Created

Preface

This document includes the system requirements of the Human Body Tracking System. The document is prepared according to the

[1]IEEE Std 830–1998

[2]IEEE Std 1233a-1998.

The document gives a complete description of specifications and requirements of the Human Body Tracking System. The first section explains purpose and scope of the document, gives definitions, acronyms, and abbreviations used throughout the document.

Overall description is given on the second section of the document. Product perspective, product functions, constraints, assumptions and dependencies subsections explain overall description of the document in detail. The third section discuss about specific requirements which contains interface requirements, functional and non-functional requirements.

Data model and description is given on the fourth section of the document. Class diagrams are also given in this section.

The fifth section explains behavior of the system by giving state transition diagrams.

Final part of the document gives planning of team structure, basic schedule, process model and conclusion.

Contents

1. Introduction

1.1 Problem Definition

In today's world, companies are trying to explore new and productive market strategies. Large part of these ideas is the most client-interactive ones. This means that, companies are trying to trace customers to see why they purchase this product, and why not. And according to this, they arrange their market strategies. As engineers, we can develop some solutions that are needed to assist people that have such concerns.

1.2 Purpose

The purpose of creating this document is to provide the users with a complete scope of all functionalities, specifications and behavior of the system. Some interfaces will explain system requirements. Moreover, how the user manages the system and interacts with it is explained in a detailed manner in use case diagrams. Class diagram is used to show the object classes and relations between each class in the system. Also, state diagram shows system states as nodes and events as arcs between these nodes. When an event occurs, the system moves from one state to another.

This SRS document is addressed to those type of audience:

- Audience who wants to be informed about the system and use the product of this project.
- Software developers, designers, testers and analysts who will take part in this project.

1.3 Scope

Human Body Tracking System aims to track human bodies in retailer environments in order to generate a heat map which shows usage weights of different tracks on a retailer environment. In other words, system calculates places how much time their customers spend on different tracks and show this information on a heat map. Then managers and store managers can display these heat maps from the web application that will be implemented in the scope of this project. According to this information, they can change their sales strategies. They are also capable of arranging the time interval of heat maps. Hence they can follow the success or failure of their strategies by comparing different heat maps resulting from different time intervals.

1.4 Definitions, Acronyms and Abbreviations

Definitions of all terms, acronyms and abbreviations are listed in the below table;

TERM , ABBREVIATION, ACRONYM	**DEFINITION**
Heat Map	Result of the system data in graphical representation. Data is measured with individual colors on the heat map.
Brand	Brand is the right-holder client, the company that applies the product on its branch stores.
Interface	Boundary across between separate components of system that provides information exchanging.
State Transition Diagram	Diagram that shows conversions of the system states and events between them.
Use Case Diagram	Diagram that shows interactions of system and its environment between each other
Class Diagram	It shows the interactions between a system and its environment
Store Manager	The person who is responsible for branch store.
Manager	The person who is responsible for all branch store and brand.
Admin	The person who is responsible for whole system.
Criteria	The attributes offered by the system to filter the heat map according to time intervals.
Background Subtraction	It is a technique in the fields of image processing and computer vision.

1.5. Existing Solutions & Literature Survey

1.5.1 Existing Solutions

Some companies and universities are working on this problem to innovate new aspects about this problem and there are some products which are available in the market.

For instance Alex Leykin who has project named with “Tracking and Activity Analysis in Retail Environments” (2005), tried to track customers in a store and performs a number of activity analysis tasks based on the output from the tracker. Figure 1 shows the low – level processing components of the system according to the Alex Leykin’s reports. For more information the website of the project can be visited. [3]

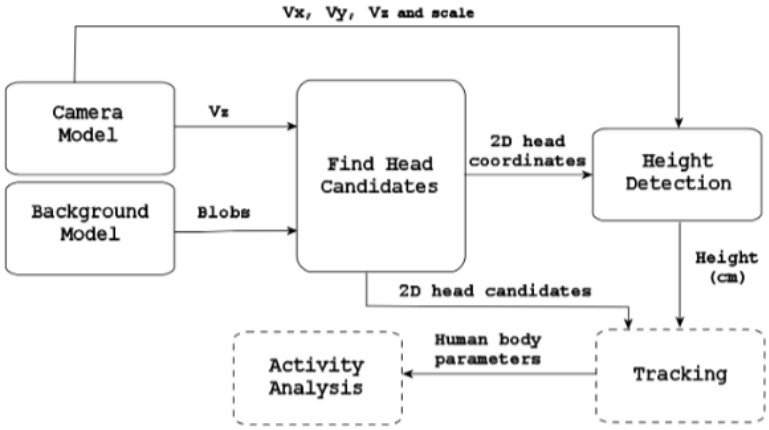


Figure 1

The other product is “Method and system for detecting and tracking shopping carts from videos” (2009) which is invented by Hankyu Moon, Rajeev Sharma and Namsoon Jung. The aim of this project is detecting and tracking shopping cart from video images in retail environment. [4]

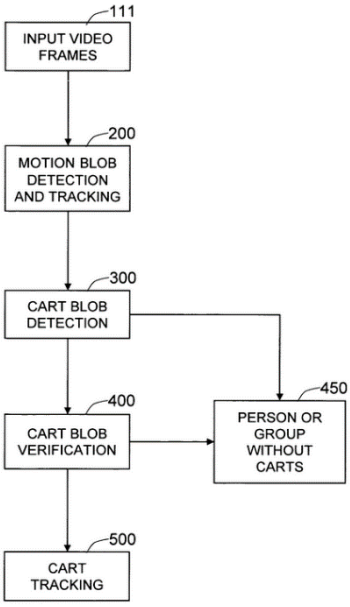


Figure 2

Figure 2 shows the components of the system according to the report of the inventors. For more information the website of the project can be visited.

1.5.2 Literature Survey

During implementation phase, the first step that we should do is background subtraction on digitized frames. For this purpose, three popular background subtraction algorithms, Mixture of Gaussian, ViBe and Mean-Median filtering have been investigated in detail in following subsections. According to given advantages and disadvantages of these algorithms, it is decided to use Mixture of Gaussian algorithm for background subtraction. Literature survey of the other steps will be introduced in the upcoming versions of this document.

1.5.2.1 Mixture of Gaussian

In Mixture of Gaussian model, the values of each pixel is modeled by a mixture of K Gaussian distributions (K = 3 to 5). At each iteration weights are calculated according to Gaussian distributions. These weights show the time ratios that those colors stay in the scene. If the pixels do not correspond to any background Gaussians, it becomes foreground. Then these foreground pixels are

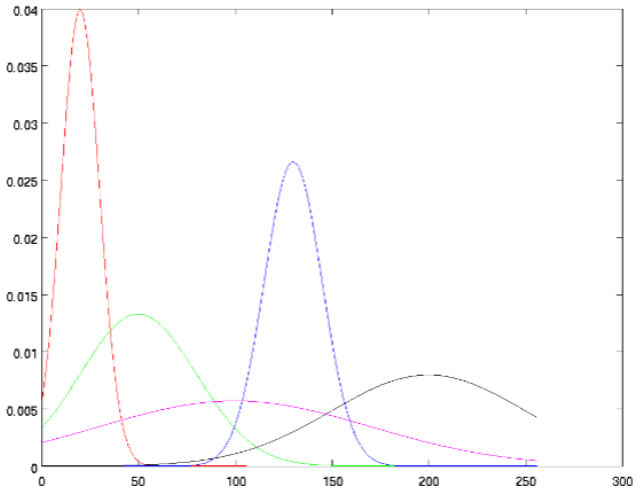


Figure 3

combined by using 2D connected component.

If the K is 5 and it is assumed that the images are grey scale. Then the history of a pixel is like in the Figure 3.

An on-line K-means is used to update the Gaussians. If a new pixel value is matched to one of the existing Gaussians, then the Gaussian will be updated like this:

$$\mu_{i,t+1} = (1-\rho)\mu_{i,t} + \rho X_{t+1}$$

and

$$\sigma^2_{i,t+1} = (1-\rho)\sigma^2_{i,t} + \rho(X_{t+1} - \mu_{i,t+1})^2$$

If a new pixel value is not match to one of the Gaussians distributions, then the least probably distribution will be replaced with a new one. This new pixel has a high variance and a low prior weight.

After these calculations, we need to define a background according to the K distributions Gaussian Model. It is important that the background pixels have least variance and high weights.[1]

Advantages

- Threshold is not same for each pixel.
- These pixel-wise thresholds are chosen by time.
- Ensures fast recovery.
- This algorithm overcomes the dynamic backgrounds.

Disadvantages

- Initializing the Gaussian is important.
- Cannot deal with the drastic changes.

1.5.2.2 ViBe

Pixel model and classification process: The algorithm builds a pixel model by just looking at real observed pixel values. The current value of the pixel is compared to its closest samples within the collection of samples.

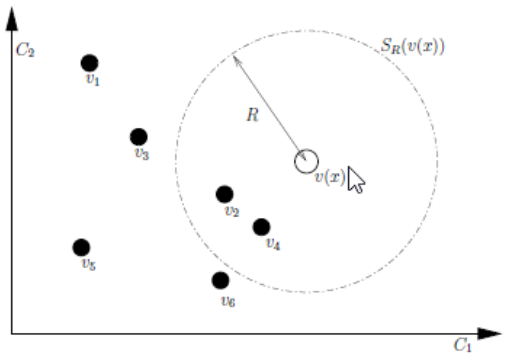


Figure 4 - Comparison of a pixel value with a set of samples in a two dimensional Euclidean color space (C1, C2)

This comparison process can be explained by Figure 4. In a given Euclidean color space, v(x) stands for the current value of the pixel located at x in an image. M(x) = {v_1, v_2,…, v_N} is consists of background samples values(v_i) that are taken from previous frames. In order to determine a pixel value v(x) from its related set of samples M(x), it is compared by the closest values from a subset of M(x) that are covered by a sphere S_R(v(x)). This sphere is centered at v(x) and has a radius R. If more than or equal to n_{thresh} values from M(x) are inside the sphere, v(x) is a background value, otherwise not.

Background model initialization from a single frame: ViBe initializes the background from a single frame. When a frame is obtained, it populates the pixel models with values found in the spatial neighborhood of each pixel. The model is initialized as follows; $M^0(x) = \{v^0(y \mid y \in N_G(x))\}$ where 0 corresponds to first frame, N_G corresponds to spatial neighborhood of the pixel at x and y corresponds to locations that chosen randomly according to a uniform law.

Updating the background model over time: The ViBe update method incorporates three important components;

Memoryless Update Policy: Which value is discarded from the model M(x) is determined randomly. Every value in M(x) has equal probability of being discarded after a fixed time. The past has not any effect in this update process.

Random Time Subsampling: When a pixel value is determined as the background, again a random process determines whether this value is used to update the corresponding pixel model or not.

Updating the neighboring pixels' background values: If the current pixel value v(x) is determined as the background, a pixel from the neighborhood of x is randomly chosen and its background value is updated. [5]

Advantages

- Very fast background model initialization.
- Overcomes the dynamic backgrounds.

Disadvantages

- It is patented and its full version can be used only by paying some licensing fee.
- Background model may include 'ghost' for some time

1.5.2.2 Mean-Median

Mean Filter: Background is the mean of the previous n frames.

$$B(x,y,t) = \frac{1}{n}\sum_{i=0}^{n-1} I(x,y,t-i)$$

$$|\, I(x,y,t) - B(x,y,t)| > T_h$$

Figure 5 - Result when $T_h > 20$

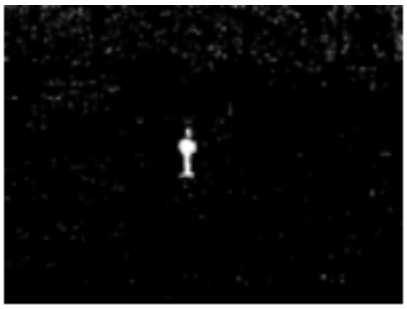


Figure 6 - Result when $T_h > 50$

Median Filter: Background is the median of the previous n frames.

$$B(x,y,t) = median\,\{I(x,y,t-i)\}$$

$$|\, I(x,y,t) - B(x,y,t)| > T_h$$

Advantages:

- Easy to implement and use.
- Pretty fast
- Background models are not constant they change in time.

Disadvantages:

- Accuracy of frame differencing depends on object speed and frame rate.
- High memory requirements.
- There is one global threshold, for all pixels in the image (T_h). [1]

1.6 References

[1] Background Subtraction (2009) Retrieve from http://www.cs.utexas.edu/~grauman/courses/fall2009/slides/lecture9_background.pdf

[2] OpenCV Documentation Retrieve from http://docs.opencv.org/2.4.7/index.html

[3] Tracking and Activity Analysis in Retail Environments Technical Report 620 (2005) Retrieved from http://www.cs.indiana.edu/ftp/techreports/TR620.pdf

[4] United States Patent (2012) Retrieve from https://drive.google.com/viewerng/viewer?url=patentimages.storage.googleapis.com%2Fpdfs%2FUS8325982.pdf

[5] ViBe: A universal background subtraction algorithm for video squences (2011) Retrieve from http://www.vibeinmotion.com/Portals/0/Documents/Barnich2011ViBe.pdf

1.7 Overview

This document was written with recommended IEEE standards (IEEE Std 830-1998). In the following chapters, the information will be given in the order of overall description, specific requirements, data model and description, behavioral model and description and planning. Use case, class, block and state transition diagrams will be used to give more precise explanations.

2. Overall Description

This section of the report will try to give a general understanding about the project by explaining the factors that affect the product and its requirements.

2.1 Product Perspective

As mentioned earlier, the product will be called Human Body Tracking System for Retail Environments. The product aims to collect data from the stores by a number of cameras and represent this information as a heat map to the users of the system. The product is designed as independent and totally self-contained, so it is not designed as a component of a bigger system.

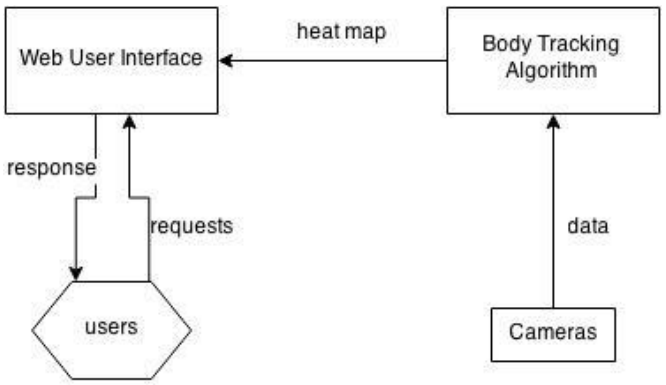


Figure 7

Figure 7 shows that the system is designed as mainly two parts. First part includes collecting data and give meaning to them by using an algorithmic logic. In this point the “algorithmic logic” term must be enlarged to define the system better. Track algorithms collect data from the cameras by the help of the frame grabber component, which will be used in the system, and then after applying the algorithm some meaningful information will be present. This information involves how long a blob (tracked objects are humans for this case) stayed in which place of a store. A heat map will be generated from that information.

Second part of the system consists of showing this information to the users of the system. A web user interface is used for this part. By logging into the system with their credentials, desired information will be shown to the users.

2.1.1. System Interfaces

Because the system is totally independent and is not related to any other product, it has not any interface matching the system. It can be said that the only interfaces are Java library for web application and OpenCV library for human tracking.

2.1.2. User Interfaces

In this section, user interfaces of the system will be explained in detail with some draft displays.

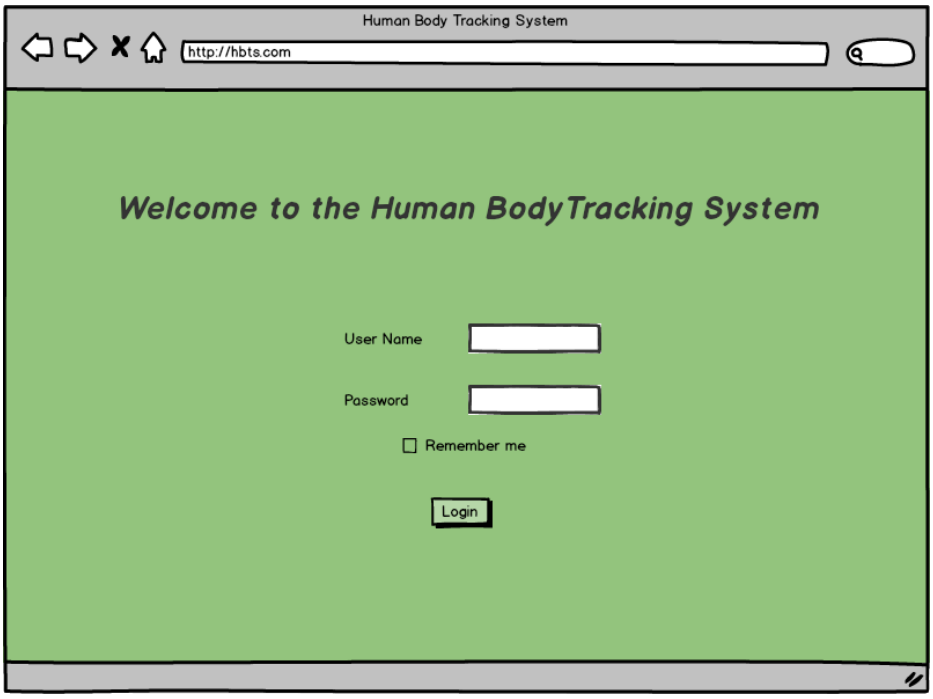


Figure 8 – Login Page

Figure 8 shows login page of the system and all kinds of users must login first so that they can use the functionalities of the system. If the 'Remember me' option is checked by the user, username and password information are remembered by the browser for the next time. After user clicks the 'Login' button, he/she is directed to the home page if entered username and password exist.

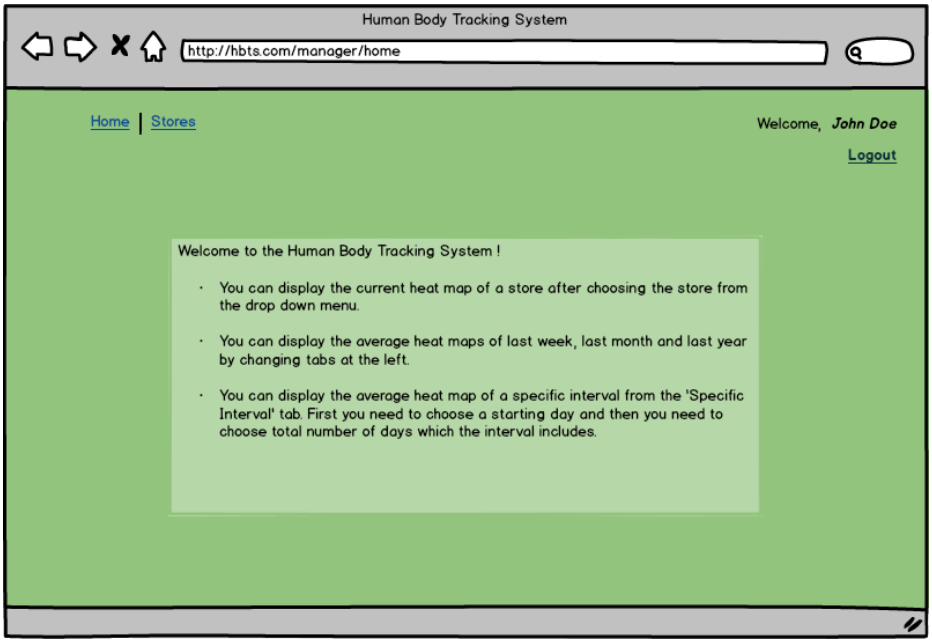


Figure 9 – Home page of Manager

Above figure shows the home page of manager and it includes some instructions about usage of the system. At the top right corner of the page, there is a welcome message with the name and surname of the manager. Manager can logout from the system by clicking 'Logout' that is under the welcome message. Same situation is valid for the other user types. When 'Home' is chosen from the navigation bar, user is directed to this page. The other option 'Stores' from the navigation bar provides to display heat maps of available stores.

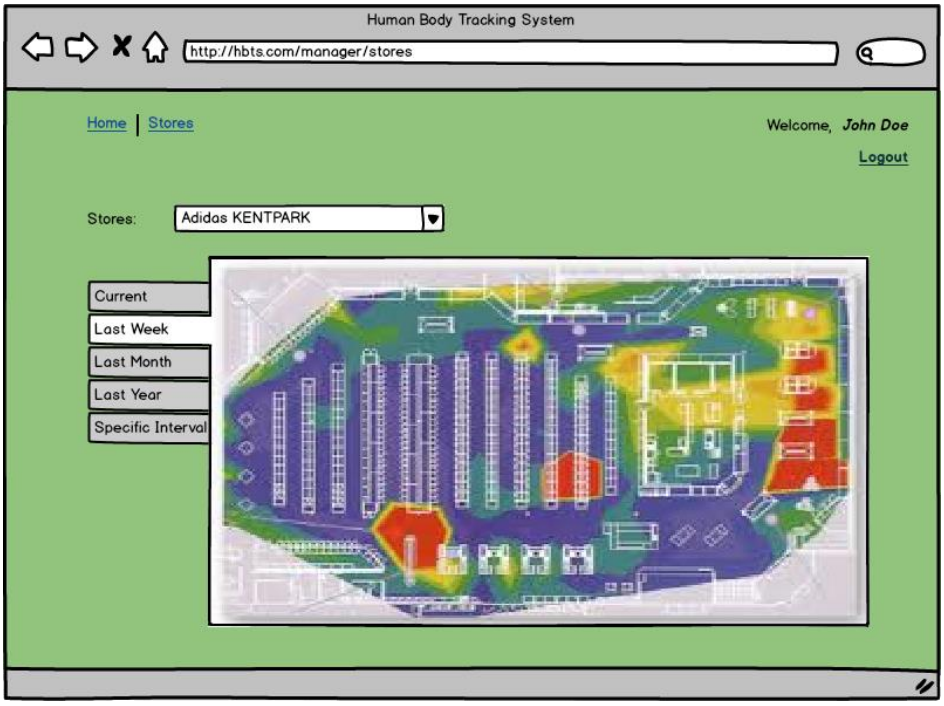


Figure 10 – Stores page of Manager when 'Last Week' tab is active

First of all, manager must choose the store whose heat maps are wanted to be displayed and he/she uses drop down list for this purpose. At the moment he/she chose the store, current heat map is displayed in the space of the page. User can display different heat maps corresponding to different time intervals by clicking tabs at the left. As mentioned before, current heat map of the store is displayed by default. Figure 10 shows the heat map when 'Last Week' tab is active. While 'Current', 'Last Week', 'Last Month' and 'Last Year' tabs directly open heat map of that time interval, user should enter two information first in 'Specific Interval'. Below figure shows corresponding page when 'Specific Interval' tab is active.

Figure 11 - Stores page of Manager when 'Specific Interval' tab is active

User should first choose the beginning date of the interval from the date time picker and then enter the total number of days that the interval includes. After these two operations, when the ‘Display’ button is clicked, average heat map that encompasses whole interval appears. For example, heat map in the Figure 11 shows average heat map of an interval that begins with 15.09.2014 and ends with 30/09/2014.

Figure 12 - Heat map page of Store Manager

If a store manager logins to the system, a home page which is almost same with page at Figure 9 welcomes the user. There is ‘Heat Map’ option in the navigation bar instead of ‘Stores’ and instructions are prepared according to capabilities of this user type. When the user clicks the ‘Heat Map’, same tabs of manager are appeared on the left. The only difference is that there is not any drop down list for selection of store. (Figure 12)

If an admin logins to the system, page at the Figure 13 is displayed. Navigation bar contains ‘Brands’, ‘Stores’ and ‘New’ options. The home page of admin shows all brands that were registered to the system. Brands are listed in a table and pagination can be used to navigate between pages of table. The table includes the name of the brand, the number of stores in the system that the brand has and name of the manager of the brand. Admin can delete or can update a brand by clicking corresponding buttons in the last column.

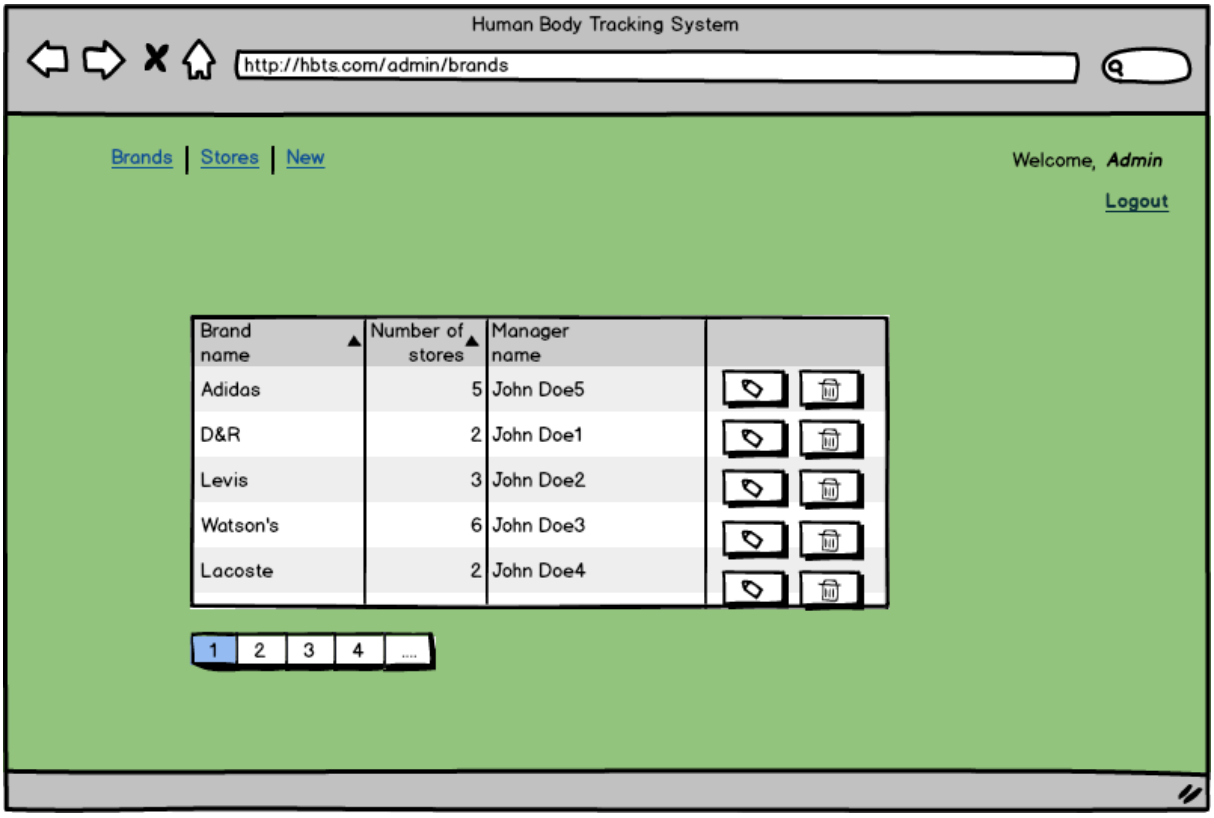


Figure 13 - The brand page of admin

In addition, admin can click items in the number of stores and manager name columns. If an item from manager name column is clicked, detailed information about that manager is displayed like in Figure 14. This figure shows the case that 'John Doe5' is clicked from the Figure 13.

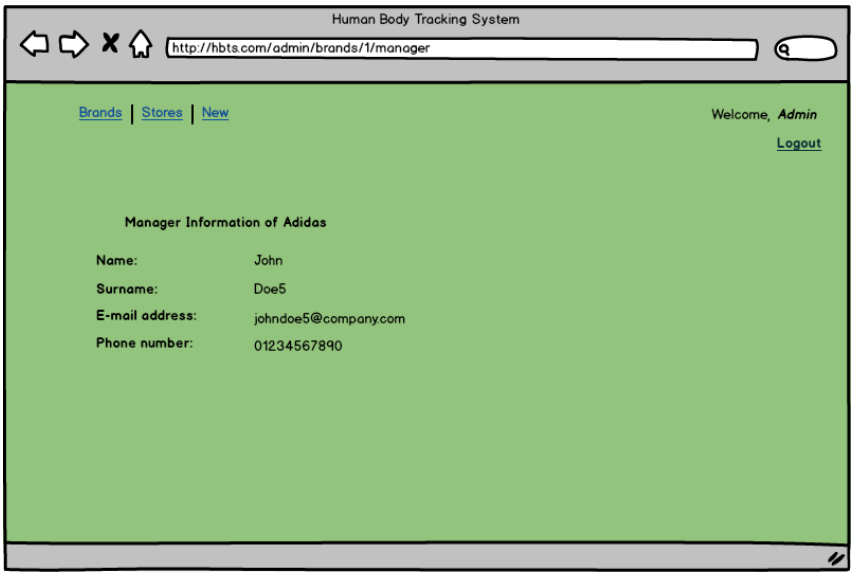


Figure 14 - Manager details page of admin

If an item from number of stores is clicked, new page is appeared containing a table of stores. For example, if '5' is clicked from the first row in Figure 13, below page is opened. It includes all stores of selected brand and manager names of these stores. Admin can delete

or edit a store from options in the last column of table. In addition detailed information about store manager can be displayed by clicking the name of store manager like Figure 14.

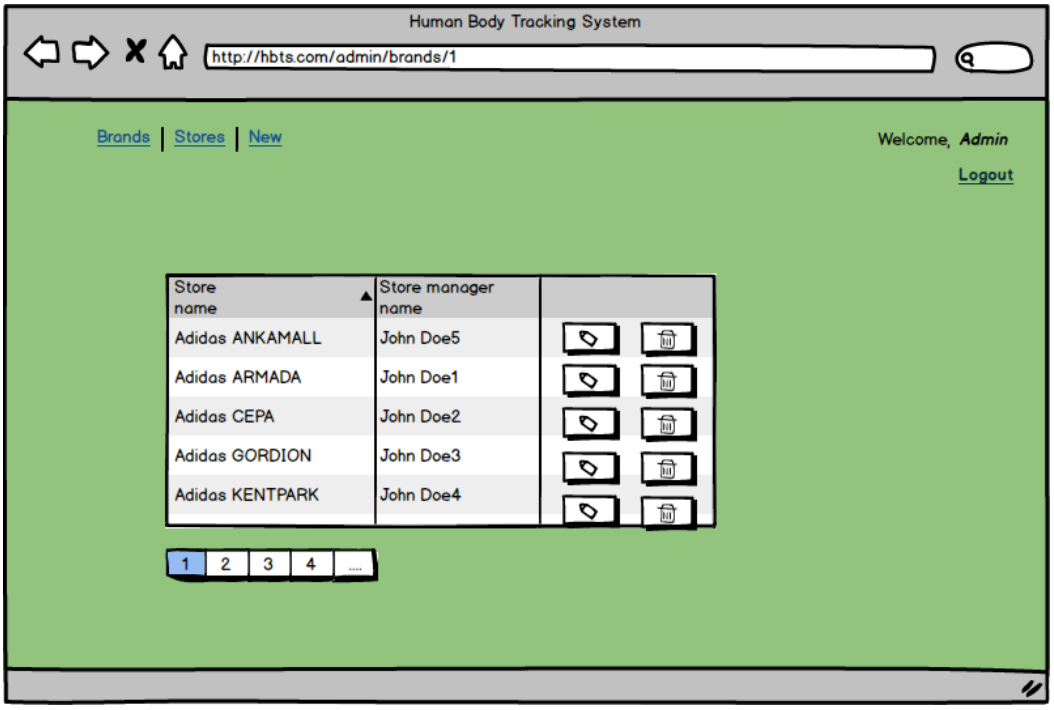


Figure 15 - Stores page of admin

Admin can display heat maps of all stores in the system from 'Stores' option in the navigation bar of admin page. This page is almost same with page at Figure 11.

Figure 16 - New option in the navigation bar

'New' option in the navigation bar can be expanded like in the Figure 16. Admin can add new brand and a new store from options in the figure. Because pages of these options seem like each other, only new brand page will be explained in this document.

When Admin clicks 'Brand' from these options, below page is displayed. There is one text box for the name of new brand. The other text boxes are for information about manager that is responsible for this new brand. Name, surname, e-mail address and phone number of the manager must be entered to text boxes respectively. After admin clicks the 'Add' button new information is saved.

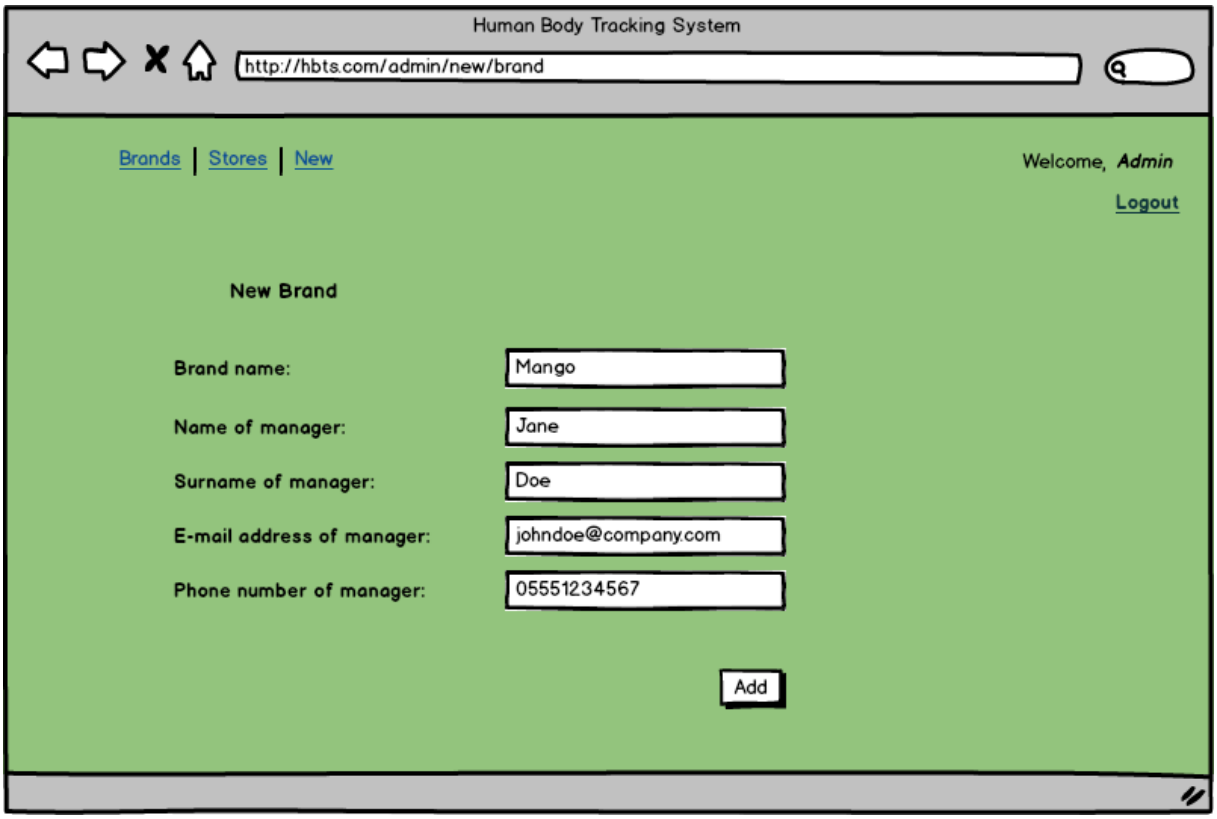


Figure 17 - New brand page of admin

System will give appropriate errors and warnings according to user interactions. When a text box is left empty, a red error message *'This field cannot be left empty*' will be appeared at right of the text box. When admin is adding a new brand or store, if duplicate value occurs, an error message '*There is already a brand with this name!*' will be appeared on the screen. In addition, while a brand or a store is deleting, a warning message '*Are you sure about deleting this brand*?' will be appeared.

2.1.3. Hardware Interfaces

Surveillance cameras that will be installed in retailer environments play an important role from the hardware interfaces point of view. There are two types of cameras which are considered to be used. First one is analog camera. Analog cameras send the recorded video or captured images in analog format via a coax cable or an UTP cabling to a Digital Video Recorder (DVR). DVR is a kind of analog to digital converter and it provides to video be digitized. At this point, video can be stored on hard drives. With today's technology, all DVRs are coming with a LAN/WAN setup so can be accessed remotely through internet. Thus, although the video is transmitted from the analog cameras in an analog format, recorded

video is still available over the network. The other one is IP (Internet Protocol) camera. They transfer videos with digital format over an IP network. A video from the IP camera is streamed straight from the camera to the network video recorder (NVR). Video footage is stored on these NVRs and is protected by RAID technology.

Analog cameras and their equipment have lower costs than IP cameras. Cost difference between them may be significant with changing design preferences. On the other hand, IP cameras provide high resolution videos. While analog cameras produce 0.4 mega pixel videos, IP cameras can offer a resolution ranging from 1 mega pixels to 29 mega pixels. According to these advantages and disadvantages the camera selection may change through time.

2.1.4. Software Interfaces

Open Source Computer Vision (OpenCV), Version 2.4.7., will be used in this project in order to benefit from its wide libraries about video capturing, background subtraction, image filtering etc. OpenCV is an open source computer vision and machine learning software library and it includes more than 2500 optimized algorithms. It is completely free for both commercial and non-commercial use. [2]

Eclipse Platform, Version 3.8.1, will be preferred as a development environment. It gives a useful development environment containing OpenCV especially on Linux machines. Eclipse CDT should also be installed on development computers, because project will be implemented by C++. Eclipse CDT provides a functional C and C++ environments for Eclipse IDE.

An Intel toolset will be used to enhance some features of the project. The toolset is called Intel Parallel Studio XE 2015. This toolset consists of: C/C++ compiler, thread library (Intel TBB (Threading Building Blocks)), thread design and prototyping (Intel Advisor), memory and thread debugging (Intel Inspector) and performance profiler (Intel VTune Amplifier). C/C++ compiler and Intel TBB will be frequently used during the project. They can be integrated with Eclipse IDE on Linux.

Database side of the project will be carried out with MySQL 5.5. It is an open-source relational database management system (RDBMS).

2.1.5 Communication Interfaces

Communication interface that stems from the connection between host DBMS and Human Body Tracking System is provided by TCP/IP protocol.

2.1.6 Memory

The system required to work on a minimum of 4 GB primary memory. Memory usage of the system should not exceed 60% of the overall system. All memory tests should be done in order to avoid memory leaks in the system.

2.1.7 Operations

The operations are explained in User Interfaces section (2.1.2) in detail. So, it will not be mentioned again here.

2.1.8. Site Adoption Requirements

As this system is supposed to be installed and used in a physical environment, it is important to think about some specifications. Camera angles and lightning is crucial for a system like this in order to work on correct information.

For the web-interface part which clients are going to use, a web browser and an internet connection is enough.

2.2. Product Functions

Human Body Tracking System consists of three user types. Those user types are managers, store managers and admins.

Manager user type is the managers of a brand. In some parts of the SRS, this user type is also referred as brand manager. Brand managers are able to see all the stores of a brand. Figure 18 shows that they can choose a store of their brand and display its heat map by choosing the desired date intervals.

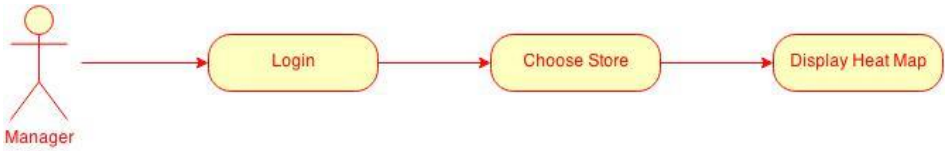


Figure 18 – Use Case Diagram of Manager

Store managers are described as a manager of a single store of a brand. Therefore they are only able to display the heat map of their store as shown in figure 19.

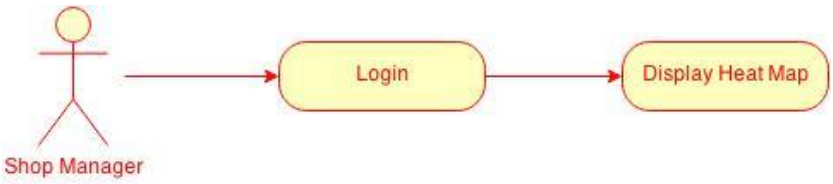


Figure 19 – Use Case Diagram of Store Manager

Admin user type are responsible for system administration. They have duties like handling information about brands. If a brand bought the product and after required hardware installation admin must add that brand to the system with the required brand manager information. Also admins can do delete and edit operations about the brands as shown in figure 20.

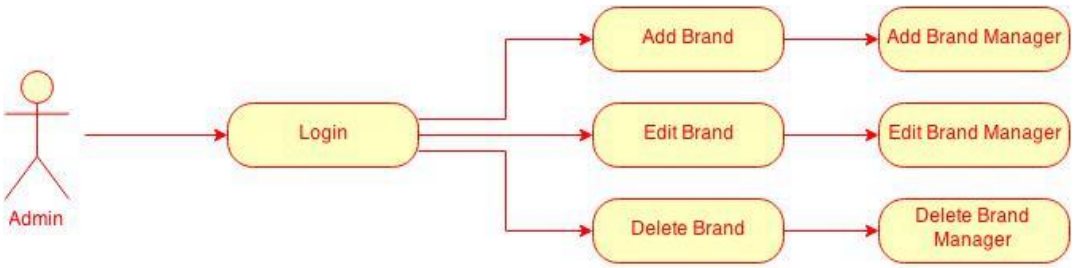


Figure 20 – Use Case Diagram of Admin - 1

Admins are also responsible for adding, editing and deleting store information about related brand as shown in figure 21.

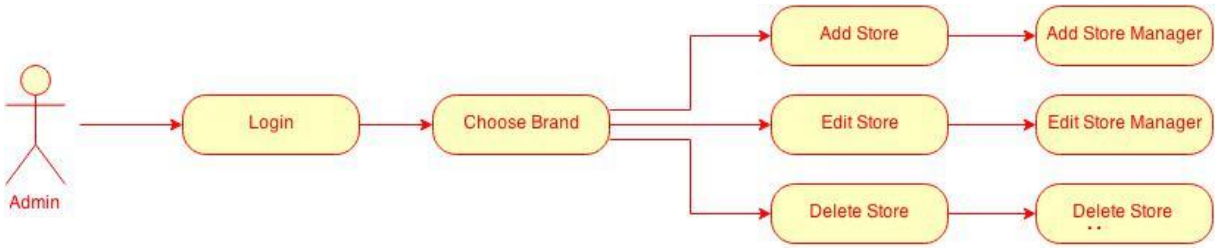


Figure 21 – Use Case Diagram of Admin - 2

Admins can display the heat map of all the stores of all the brands in the system as shown in figure 22.

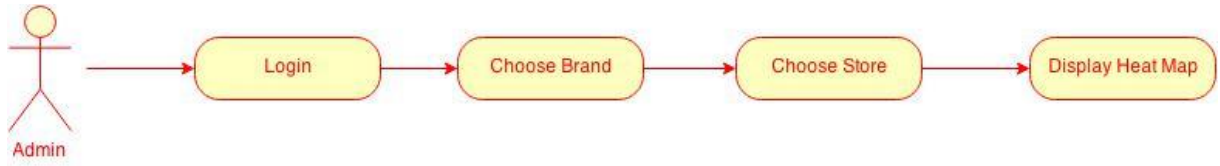


Figure 22 – Use Case Diagram of Admin - 3

All user types can logout from the system whenever they want to.

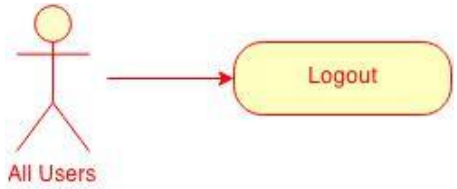


Figure 23 – Use Case Diagram of All Users

2.3. Constraints

Hardware limitations may occur as a constraint for this system as cameras are the main gatherer of data. Data flow from cameras to the system may be a constraint.

The system is using cameras in retail environments. Privacy of people must be considered as an important issue and should not be violated while creating and using the product.

2.4. Assumptions and Dependencies

This SRS is written from the assumption that Mixture of Gaussian algorithm will be used for the background subtraction algorithm. If there will be any changes in the implementation phase, it will be stated in the revisions of SRS.

The web user interface is assumed to be worked at browsers.

If better results will be encountered in other usages of hardware, the system may change the usage of the hardware. (For example: used cameras in the retail environments)

3 Specific Requirements

3.1 Interface Requirements

The systems interfaces were explained in detail in Section 2.1.2 Overall Description.

3.2 Functional Requirements

This subsection contains each major software function.

No	Functionality	Short Description
1	Login	Login to the system according to the user id and password.
2	Logout	Logout from the system.
3	Filter/Sort Heat Map	Filtering the heat map.
4	View Brand	View the brand information.
5	View Branch Store	View the branch store information.
6	Add Brand	Add new brand to the system.
7	Update Brand	Update the branch information.
8	Delete Brand	Delete brand in the system.
9	Add Branch Store	Add branch store to the brand
10	Update Branch Store	Update the branch store information.
11	Delete Branch Store	Delete one of the branch from the brands' page

3.2.1 'Login' Use Case

Use Case ID	HUMANTRACKINGSYSTEM.UC.1

Use Case Name	LOGIN
Description	This use case logins the admin, manager and the store manager to the system.
Actors	Admin, manager and store manager are able to login to the system.
Preconditions	Admin, manager and store manager need userID and password to login to the system.
Trigger	This use case is initiated when the user fills the id and password fills and enter the 'Login' button.
Basic Flow	Step 1: Admin, manager and the store manager opens the HUMANTRACKING System homepage on the web browser. Step 2: Users enter the login button after filling necessary blanks.
Alternate Flow	-
Exception Flow	If the user enters the wrong id or password, system gives error message.
Post Conditions	-

3.2.1.1 Functional Requirement 1.1

Only admin, manager and store manager are able to login the system with their own id and password.

3.2.1.2 Functional Requirement 1.2

Each user is differentiated in terms of authority after login to the system with the specific id and user password.

3.2.1.3 Functional Requirement 1.3

If the user check the ‘Remember me’ button, system must fill the blanks after the user filled it once.

3.2.2 ‘Logout’ Use Case

Use Case ID	HUMANTRACKINGSYSTEM.UC.2
Use Case Name	LOGOUT
Description	This use case logouts the admin, manager and the store manager from the system.
Actors	Admin, manager and store manager are able to logout from the system.
Preconditions	Admin, manager and store manager need to login the system before enter the logout.
Trigger	This use case is initiated when the user clicks the logout button.
Basic Flow	Step 1: User opens the HUMANTRACKING System homepage on the web browser. Step 2: User clicks the ‘Login’ button. Step 3: User clicks the ‘Logout’ button.
Alternate Flow	-
Exception Flow	-
Post Conditions	-

3.2.2.1 Functional Requirement 2.1

The user shall login the system to be able to logout from the system.

3.2.2.2 Functional Requirement 2.2

After the user logout, the system must record the changes to the database.

3.2.3 'Filter/Sort Heat Map' Use Case

Use Case ID	HUMANTRACKINGSYSTEM.UC.3
Use Case Name	Filter/Sort Heat Map
Description	This use case describes the event when a user filters the heat map according to some filter options.
Actors	Users who have the ability to login the system, can use this function.
Preconditions	User must choose a criteria before he/she tries to filter the map.
Trigger	This use case is initiated when the user selects the criteria of his/her filter/sort and clicks on the button of filter.
Basic Flow	Step 1: User opens the HUMANTRACKING System homepage on the web browser. Step 2: User selects one of the criteria, which is located in left of the heat map. Step 3: System shows the heat map according to the filter option.
Alternate Flow	Step 1: User opens the HUMANTRACKING System homepage on the web browser. Step 2: User selects specific interval criteria. Step 3: User fills the date interval and the total number of days. Step 4: User clicks the 'Display' button to see the heat map
Exception Flow	If system couldn't find any heat map depends on criteria, it returns error message and recommend users to change the criteria.
Post Conditions	-

3.2.3.1 Functional Requirement 3.1

User shall set the date field which has to be numeric values.

3.2.3.2 Functional Requirement 3.2

System shall contain heat map which satisfies the criteria.

3.2.3.3 Functional Requirement 3.3

System shall produce a result at least one of the criteria is entered.

3.2.3.4 Functional Requirement 3.4

User shall fill the necessary blank when selects the 'Specific Interval' criteria.

3.2.4 'View Brand' Use Case

Use Case ID	HUMANTRACKINGSYSTEM.UC.4
Use Case Name	VIEW BRAND
Description	This use case shows the brands' features such as manager name, e-mail address and the phone number.
Actors	Only admin can view the brand in the system.
Preconditions	The user shall login with the admin id and password.
Trigger	This use case is initiated when the admin selects one of the brands from the 'Brand Name' column.
Basic Flow	Step 1: User opens the HUMANTRACKING System homepage on the web browser. Step 2: User enters the "Login" button with the id and the password. Step 3: User selects one of the brands from the 'Brand name' column.

Alternate Flow	-
Exception Flow	-
Post Conditions	-

3.2.4.1 Functional Requirement 4.1

Only admin shall view brand in the system.

3.2.4.2 Functional Requirement 4.2

System shall keep the brands information in the database.

3.2.5 'View Branch Store' Use Case

Use Case ID	HUMANTRACKINGSYSTEM.UC.5
Use Case Name	View Branch Store
Description	This use case shows the branch stores' features such as manager name, e-mail address and the phone number.
Actors	Only admin can view the branch store in the system.
Preconditions	The user must login to the system with the admin id or manager id.
Trigger	This use case is initiated when the admin selects one of the branch store.
Basic Flow	Step 1: User opens the HUMANTRACKING System homepage on the web browser. Step 2: User enters the 'Login' button with id and password. Step 3: User selects one of the brands' store number from the 'Number of stores' column.
Alternate Flow	-
Exception Flow	-

Post Conditions	-

3.2.5.1 Functional Requirement 5.1

The user login to the system as an admin.

3.2.5.2 Functional Requirement 5.2

System shall keep the branch store information in the database.

3.2.6 'Add Brand' Use Case

Use Case ID	HUMANTRACKINGSYSTEM.UC.6
Use Case Name	ADD BRAND
Description	This use case adds the brand to the system.
Actors	Only admin can add brand to the system.
Preconditions	Admin need to fill the necessary blanks to add the brand.
Trigger	This use case is initiated when the admin enters the 'New' from the top navigating bar.
Basic Flow	Step 1: User opens the HUMANTRACKING System homepage on the web browser. Step 2: User enters the "Login" button with the id and the password. Step 3: User enters the 'New' button. Step 4: User selects the 'Brand' button. Step 5: User fills the necessary blanks. Step 6: User enters the 'Add' button.
Alternate Flow	-
Exception Flow	If the system find the same brand name, it returns error

	messages.
Post Conditions	-

3.2.6.1 Functional Requirement 6.1

Only admin shall add brand to the system.

3.2.6.2 Functional Requirement 6.2

Admin shall fill all the blanks to add brands.

3.2.6.3 Functional Requirement 6.3

The system shall put the brands to the database when the admin enters the brand to the system.

3.2.6.4 Functional Requirement 6.4

The system shall check the new brand with the old brands to prevent replicate data.

3.2.7 'Update Brand' Use Case

Use Case ID	HUMANTRACKINGSYSTEM.UC.7
Use Case Name	UPDATE BRAND
Description	This use case updates the brand in the system.
Actors	Only admin can update brand in the system.
Preconditions	Admin need to choose one of the brand to update.
Trigger	This use case is initiated when the admin enters the update.
Basic Flow	Step 1: User opens the HUMANTRACKING System homepage on the web browser. Step 2: User enters the "Login" button with the id and the password. Step 3: User enters the update figure which is located at the

	same row with the desired brand to update.
Alternate Flow	-
Exception Flow	-
Post Conditions	-

3.2.7.1 Functional Requirement 7.1

Only admin shall update brand in the system.

3.2.7.2 Functional Requirement 7.2

The system shall save the changes to database after update action finished.

3.2.8 'Delete Brand' Use Case

Use Case ID	HUMANTRACKINGSYSTEM.UC.8
Use Case Name	DELETE BRAND
Description	This use case deletes the brand from the system.
Actors	Only admin can delete brand from the system.
Preconditions	Admin need to choose one of the brand to delete.
Trigger	This use case is initiated when the admin enters the delete.
Basic Flow	Step 1: User opens the HUMANTRACKING System homepage on the web browser. Step 2: User enters the "Login" button with the id and the password. Step 3: User enters the delete figure which is located at the same row with the desired brand to delete..
Alternate Flow	-

Exception Flow	-
Post Conditions	-

3.2.8.1 Functional Requirement 8.1

Only admin shall delete brand from the system.

3.2.8.2 Functional Requirement 8.2

The system shall save the changes to database after delete action finished.

3.2.9 'Add Branch Store' Use Case

Use Case ID	HUMANTRACKINGSYSTEM.UC.9
Use Case Name	ADD BRANCH STORE
Description	This use case adds the branch store to the system.
Actors	Only admin can add branch to the system.
Preconditions	Admin need to fill the necessary blanks to add the store.
Trigger	This use case is initiated when the admin enters the new store.
Basic Flow	Step 1: User opens the HUMANTRACKING System homepage on the web browser. Step 2: User enters the "Login" button with the id and the password. Step 3: User enters the 'New' button. Step 4: User selects the 'Store' button. Step 5: User fills the necessary blanks. Step 6: User enters the 'Add' button.
Alternate Flow	-

Exception Flow	If the system find the same store name, it returns error messages.
Post Conditions	-

3.2.9.1 Functional Requirement 9.1

Only admin shall add store to the system.

3.2.9.2 Functional Requirement 9.2

Admin shall fill the all blanks to add branch store.

3.2.9.3 Functional Requirement 9.3

The system shall put the stores to the database when the admin enters the store to the system.

3.2.9.4 Functional Requirement 9.4

The system shall check the new store with the olds to prevent replicate data.

3.2.10 'Update Branch Store' Use Case

Use Case ID	HUMANTRACKINGSYSTEM.UC.10
Use Case Name	UPDATE BRACH STORE
Description	This use case updates the branch store in the system.
Actors	Only admin can update branch store in the system.
Preconditions	Admin need to choose one of the branch store to update.
Trigger	This use case is initiated when the admin enters the update.
Basic Flow	Step 1: User opens the HUMANTRACKING System homepage on the web browser. Step 2: User enters the "Login" button with the id and the password.

	Step 3: User selects one of the number in the 'Number of stores'. Step 4: User enters the update figure which is located at the same row with the desired branch store to update.
Alternate Flow	-
Exception Flow	-
Post Conditions	-

3.2.10.1 Functional Requirement 10.1

Only admin shall update brand in the system.

3.2.10.2 Functional Requirement 10.2

The system shall save the changes to database after update action finished.

3.2.11 'Delete Branch Store' Use Case

Use Case ID	HUMANTRACKINGSYSTEM.UC.11
Use Case Name	DELETE BRANCH STORE
Description	This use case deletes the branch store from the system.
Actors	Only admin can delete branch store from the system.
Preconditions	Admin need to choose one of the branch store to delete.
Trigger	This use case is initiated when the admin selects and delete the store.
Basic Flow	Step 1: User opens the HUMANTRACKING System homepage on the web browser. Step 2: User enters the "Login" button with the id and the password.

	Step 3: User selects one of the number in the 'Number of stores'. Step 4: User enters the delete figure which is located at the same row with the desired branch store to delete.
Alternate Flow	-
Exception Flow	-
Post Conditions	-

3.2.11.1 Functional Requirement 11.1

Only admin shall delete branch store in the system.

3.2.11.2 Functional Requirement 11.2

System shall save the changes to the database.

3.3 Non-Functional Requirements

3.3.1 Performance requirements

- Any number of simultaneous users will be able to use the system.

- System works in real time environment and must give the response in 5 seconds.

- Admins must occupy the system for updating at most 10 seconds.

3.3.2 Design Constraints

3.3.2.1 Standards

- IEEE standards will be used for the documentation.

- UML standards will be used for the diagrams in this project.

3.3.2.2 Reliability

- System produces heat map with the 80/85 accuracy

- Passes between cameras work fine with 95%.

3.3.2.3 Maintainability

- In order to establish maintainability, all documentations about the software should be very detailed and understandable, and they should prepare in IEEE standard 830-1998.

- The system is always checked in the field by the team and if any bugs occur in the system, the team solves it and provides maintenance.

3.3.2.4 Portability

The project should be developed by using common technologies and tools, therefore users can access the system from all common operating systems and web browsers with the specific id and password.

3.3.2.5 Security

- The users, who have not admin privileges cannot add, delete or update any information about managers, shop managers and stores.

- The users, who have not manager privileges, cannot see the other branch stores information.

- The users, who have not any id or password, cannot login and use this system.

4. Data Description

In this part of the SRS, the data objects that will be handled by the user are described. While describing, class diagram and data dictionary will be given in order to visualize data objects clearly.

4.1 Data Objects

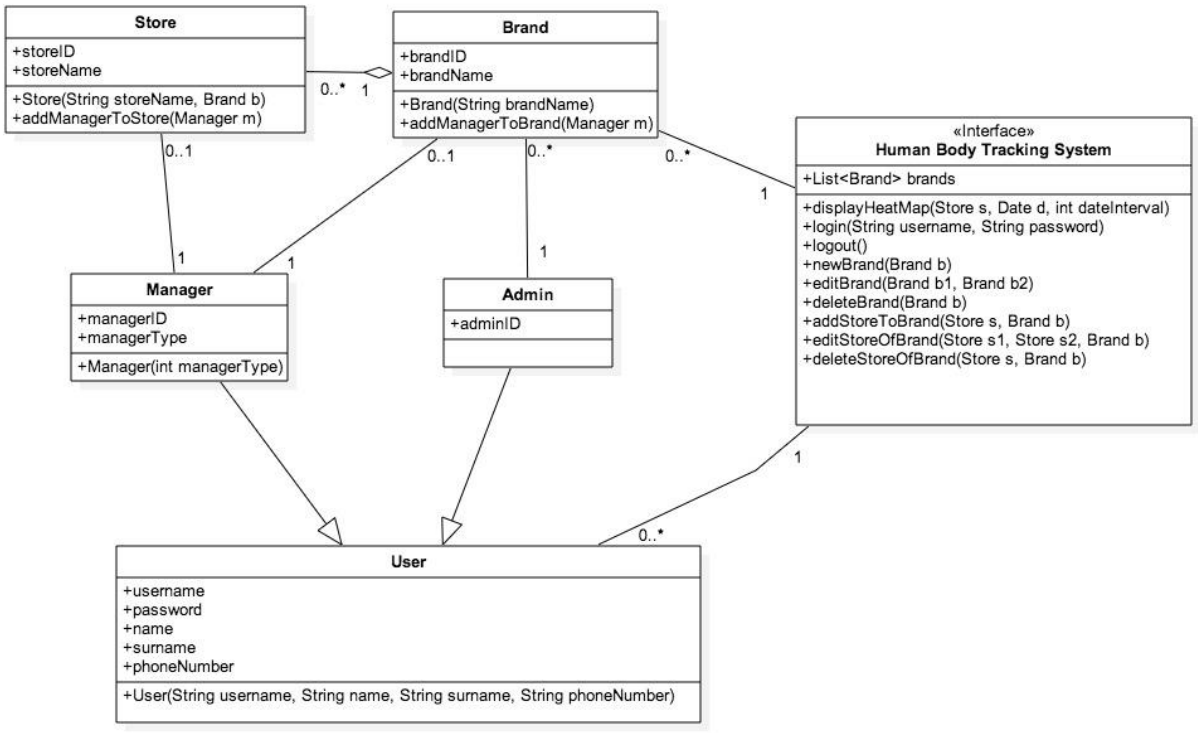


Figure 24 – Class Diagram

The class diagram is provided in the above figure. It is also seen in that class diagram that there are 6 main data objects in the project.

Human Body Tracking System is used as an abstract class. This class is considered as the user interface where users of the systems will try do the operations by calling the methods under this class. The operations a user can do in this class is logging in to the system (login function), displaying the heat map of a selected store with selected date intervals (displayHeatMap() function) and logging out of the system (logout function). Also admin operations like manipulating brand information are done in this class. (Other methods of the class) The attribute List<Brand> brands is present in this class in order to have the information of the brands that are using the system. Those brands are added to the system by the admin.

Brand is the data object that holds the information about a brand. Brands have ID's that are generated by the system and brand names. Because of the relationship between Brand and Manager data objects, all brands have a manager. It can also be said that all brands are owned by the admin. This expression corresponds to admins have access over all brands so they can do operations about them in the system.

Store is another data objects that is connected to the Brand data object by an aggregation relationship. Stores have a brand and brands may have many stores. Stores also have a generated storeID, storeName and a manager.

As it was already stated in the previous chapters of the SRS, Human Body Tracking System has three kinds of user types. They are admins, brand managers and store managers. Detailed information about user types was given in the section 2.2. In order to have a clearer object description, not keeping unnecessary and repetitive information about users in a possible usage of a database, an abstract data object called User is defined. This data object is a generalisation of the three user types in the system. The information which are hold in common for all three user types are defined in this User data object. Those informations are username (which is defined in the system as the e-mail address of the user), name, surname, phone number and a password. Password is generated by the system and delivered to the user by an email sent by the system. Security is an important issue here as email confirmation is used as a security check. Manager data object consists of two user types: brand managers and store managers. managerType attribute in this class is holding the information about if the manager is a store manager or a brand manager. Admin data object is for the admin user type.

4.2 Data Dictionary

Classes	Description	Attributes
HUMAN BODY TRACKING SYSTEM	Abstract interface class. User operations are done in this class.	List<Brand> brands
USER	Base user class. Other user types are inherited from this class	username, password, name, surname, phone number
ADMIN	Admin user type class	admin ID

MANAGER	Manager user type class. Representing both managers and store managers	manager ID, manager type
BRAND	Brand that bought the service of the product	brand ID, brand name
STORE	Store of a brand where the product is installed and used	store ID, store name

Relations	Description	Entities involved
COMPOSITION	Brands have stores and when a brand is destroyed all stores of that brand are also destroyed	BRAND - STORE (1 to N)
GENERALIZATION	Admin class is generated from the User class	USER - ADMIN
GENERALIZATION	Manager class is generated from the User class	USER - MANAGER
ASSOCIATION	Human Body Tracking System may have numerous Users.	HUMAN BODY TRACKING SYSTEM - USER (1 to N)
ASSOCIATION	Human Body Tracking System may have numerous Brands.	HUMAN BODY TRACKING SYSTEM - BRAND (1 to N)
ASSOCIATION	All stores must have one manager. Managers does not have to manage a store.	STORE - MANAGER (0,1 to 1)
ASSOCIATION	All brands must have one manager. Managers does not have to manage a brand.	BRAND - MANAGER (0,1 to 1)
ASSOCIATION	Admins have access to all brands.	ADMIN - BRAND (1 to N)

5. Behavioral Model and Description

This section will try to give a general perspective to the readers about how the system will behave. State transition diagram will be showed for this purpose.

5.1. Description for software behavior

Human Body Tracking System welcomes you with a login screen. All user types must login to the system in order to use it. When the username and the password was entered, the system checks those credentials and if there was a successful match, it directs the user to the corresponding page.

Admin user type has privileges over other user types. It has the ability to do add, edit, delete operations. Admins are also able to display the heat maps of all brands. This ability was given to the admins for the case that they had to detect any abnormalities in the system for the other users.

Manager user type can only display heat maps of which they have access to see it. If the manager is a brand manager, he/she has a wider access for heat maps as he/she can see all the heat maps of a brand's store by choosing a store. A store manager can only see his/ her store.

All user types can logout from the system whenever they like.

5.2 State Transition Diagram

The below figure is the state transition diagram of the system. The behavior of the system can be seen in this diagram. It can be seen in the diagram that user type checks are important aspect of this system as it is resulting in different behaviors of the system.

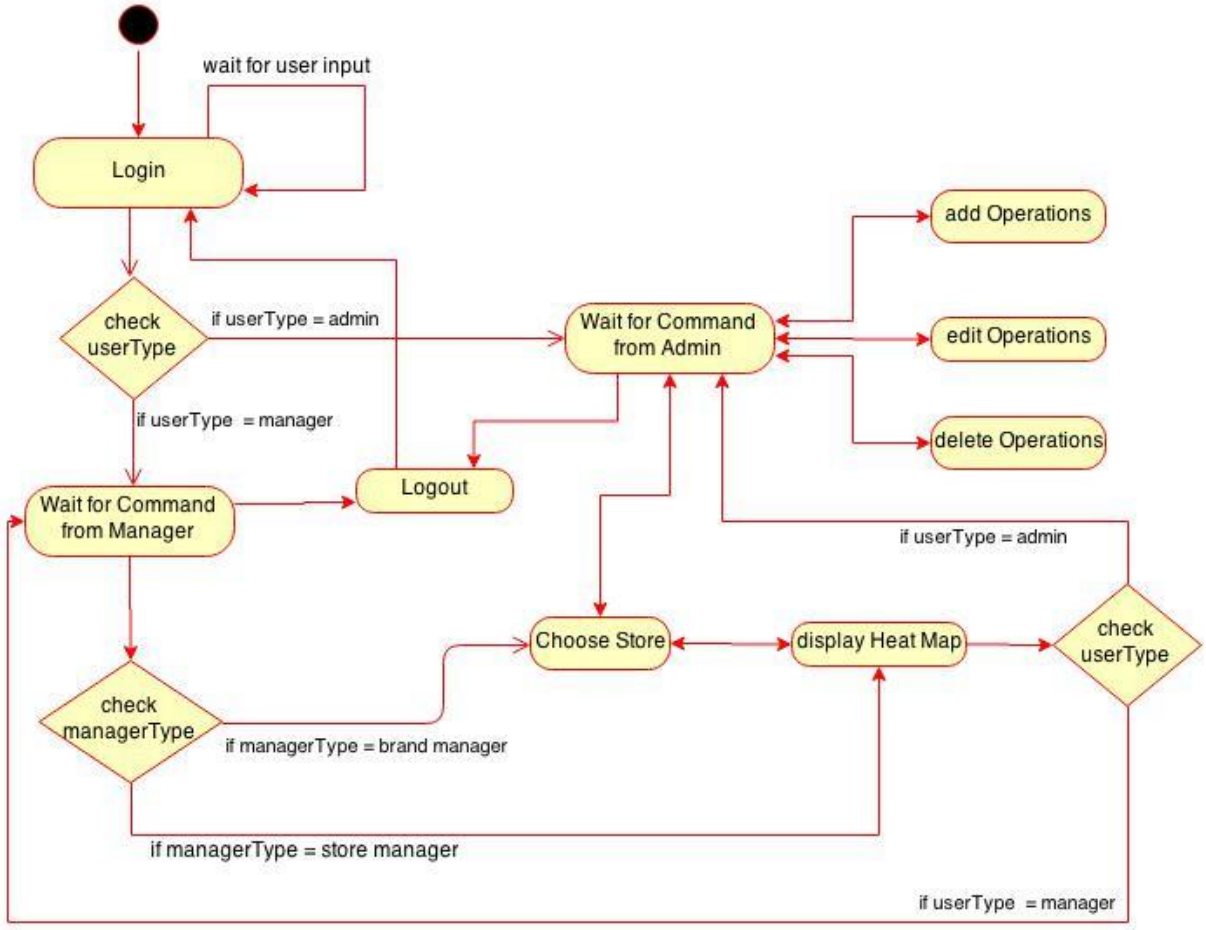


Figure 25 – State Transition Diagram

6.Planning

6.1 Team Structure

Our team consists of 4 senior students of METU Computer Engineering department:

1) Burak ARAZ
2) Cem AYDIN
3) Yalçın SAVRUN
4) Zehra Deniz ÇELİK

Project process will be proceeding in control of all of us. Estimated project plan includes 4 main work activities. We classified the major work activities in our process as below:

- Literature survey

- Project planning and reporting

- Design and Implementation of web user interface

- Implementation of track algorithm

- Integration and testing

Those activities will be dividing some organizational units that consist of at least two of us who take a role, in order to taking more clear steps.

We prefer to share the total effort equally over the members. The table below shows the distribution of roles and responsibilities:

Activity/Person	Burak ARAZ	Cem AYDIN	Yalçın SAVRUN	Zehra Deniz ÇELİK
Literature Survey	X	X	X	X
Project Planning/Report	X	X	X	X
Design and implementation of web user interface			X	X
Implementation of track algorithm	X	X		
Testing & Integration	X	X	X	X

6.2 Estimation (Basic Schedule)

- SDD report will be written by the end of December 2014.

- There are two revision planned for this SRS Document. (January 2015 - March 2015)

- Project planning will be finalized by the end of 15th of December and the team will start the implementation phase.

6.3 Process Model

The process schedule that the department provides us is organized according to the agile process model. We are taking our steps according to this process model, and work in a period of 3-week sprints.

7. Conclusion

To sum up this product may be helpful for companies which are looking for useful marketing strategies for their retail environments. Human tracking project will give a definite tips about customers' intend and their needs. Furthermore, web interface of the system will help managers or data analysts of the company to see how the system is working in clear and detailed way.

This document includes detailed information about architectural design. By the interaction with the stakeholders of the system, design viewpoints are shown in detail with their related UML diagrams. Technologies used in development and design of the project have also identified.

8 Supporting Information

8.1 Index