
Bilkent University

Department of Computer Engineering

# Senior Design Project

*Emolyst*

High-Level Design Report

**Group Members**: Elif Kevser Arslan, Ali Bulut, Musab Erayman, Muammer Tan, Ömer Faruk Karakaya

**Supervisor:** Dr. Özcan Öztürk

**Jury Members:** Dr. Varol Akman and Dr. Mustafa Özdal

**Innovation Expert:** Barış Misman

High-Level Design Report

31.12.2018

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

# 1. Introduction

In these days, people are the fundamental interface of all businesses. People's emotion and behavior affect their vital decisions which direct the businesses. If machines can know what a person is feeling, incredible insights can be unlocked. While detecting and analyzing one's feelings and comprehending results in a smart way, computer science comes into play. With the help of trusted algorithms, collected data is processed and rational inferences can be made. Hence, the value for any business can be created in an incisive way. Concordantly, we are planning to bring this idea into our senior design project. We are planning to create an application called **Emolyst** that can detect emotion, gender, and age through the integration of several algorithms. Basically, real-time data that is collected through cameras will be analyzed. Later on, statistical results will be given to our clients to make them smart and profitable choices for their businesses. Since our application will process images for only real-time data analysis under the Code of Ethics, it certainly won't break the Law for Protection of Personal Data. When all the benefits of **Emolyst** are considered, it can be said that the "mainstreaming" of emotion, gender and age recognition of Emolyst will enable companies to engage it in a multitude of profitable uses, across industries.

## 1.1 Purpose of the System

Emolyst is a CRM (Customer Relationship Management) application for legal entities and it provides analysis of their customers based on their emotional status and age-gender categories. Emolyst uses and integrates face detection, emotion recognition, age, and gender detection algorithms. It processes these data to provide analysis to our clients about their customers. Mainly, Emolyst aims to improve customer services and quality standards of companies.

It is surely beyond doubt that customer ratings are valuable sources to understand their satisfaction and are critical for designing better customer experiences and recommendations. Conventionally most of the companies were using an inefficient way to collect data like creating surveys. Thus, we decided to create a program for helping companies to get real-time reliable data about their customers and increase their customer satisfaction.

One of the most innovative and useful functions of Emolyst is providing real-time analysis. It will analyze all customers' data in the given interval. While doing this, Emolyst will not keep or use any confidential data of people and will abide

by personal data protection law. Any of the faces will not be recognized or sold to third parties. Each company can only see the analysis of their customers.

## 1.2 Design Goals

### 1.2.1 Efficiency

- The system and response time should be as fast as possible since we are dealing with real-time data and users should gather the results of analysis in real-time.

### 1.2.2 Maintainability

- Since we are going to deal with lots of data accurately, we have to maintain and keep this data properly. To increase maintainability, we are going to use two types of databases, graph database and relational database.

### 1.2.3 Reliability

- Emolyst will provide vital analysis and information about their customers to its users so system has to be reliable and provide accurate results.

### 1.2.4 Security & Privacy

- Emolyst will have very important information about its users and their customers. Thus, we will guarantee our clients that we will not share any information with third parties.
- Since we are going to keep personal data of people, we will also abide Code of Ethics outlined by the National Society of Professional Engineers.
- Also Emolyst guarantees that it will not know the name or any other personal data of detected person.

### 1.2.5 Extensibility & Supportability

- Emolyst should be able to support updates and new features in an efficient way to be open for improvements and getting rid of deficiency.

- Therefore, our program should be developed in a way that makes extending its features and making updates easy.

### 1.2.6 Compatibility

- Since Emolyst can be used across industries and companies for different purposes, it should be compatible with other systems and web browsers.

### 1.2.7 Scalability

- Emolyst should be able to handle with large scale data while processing in real time.
- Since Emolyst get the data from the camera, system should be able to cope with many cameras at the same.

## 1.3 Definitions, acronyms, and abbreviations

**DB:** Database
**API:** Application Programming Interface
**AWS:** Amazon Web Services
**GUI:** Graphical User Interface
**HW:** Hardware
**SW:** Software
**SAAS:** Software as a Service
**VPS:** Video Processing Subsystem

## 1.4 Overview

Emolyst is a web application which provides real-time customer analysis to its clients. It uses the information which is gathered from the integrated video input and processing device of the client. Emolyst uses face, age, gender and emotion detection algorithms to process the data. After processing the data with those algorithms, it tries to form and understandable and useful result by analyzing the data. Eventually, Emolyst provides the analysis of behaviors of the customers of its clients based on their age and gender using item-item association graph based on customers' emotions.

# 2. Current Software Architecture

According to our research, there are some applications that can detect age, gender and race. Also, emotion detection and face recognition programs are present at the market too. However, doing these detection and recognition at the same time in an integrated way is one of the unique features of Emolyst.

**KAIROS:** Kairos enables developers and businesses to easily build face recognition into their software products [1]. Its main features are the following:

- Face Detection:
  Kairos finds and tracks faces in any video or photo.

- Face Identification:
  It basically answers the “Who is this?” question and searches for face matches.

- Face Verification:
  It verifies users by searching for someone.

- Age Detection:
  Kairos detects age groups such that child, young-adult, adult or senior.

- Gender Detection:
  For each face found, it detects if it is female or male.

- Multi-face Detection:
  Kairos not only detects individuals but also can detect crowds and groups.

- Facial Features:
  It can detects facial features like eyes, mouth, eyebrows, nose and many more.

- Face Grouping:
  In order to search, sort and merge easily, it groups faces.

- Diversity Recognition:
  It understands the diversity of the human face.

**Microsoft Azure - Cognitive Services - Face API:** It detects and compares faces, organizes and groups images based on their similarities and identifies already tagged people. Also, it runs in the cloud or locally on-premises [2]. Its main features are the following:

- **Face Verification:**
  Microsoft Azure's Face API checks the probability that two faces belong to the same person. Later on, it returns a score of confidence that indicates the likelihood.

- **Face Detection:**
  For face detection, it uses face attributes that contain machine learning based predictions of facial features. For every face in the image, it detects age, gender, emotion, pose, smile and facial hair.

- **Emotion Recognition:**
  It returns the confidence across a set of emotions. It mainly detects anger, contempt, disgust, fear, happiness, neutral, sadness and surprise.

Despite the fact that these applications have similarities with our planned system, Emolyst will have improved, different and integrated features as best as we can. Firstly, both Kairos and Azure's Face API collect data and store them. However, in Emolyst we will not store personal data in order to not violate Code of Ethics, Law for Protection of Personal Data and abide personal privacy. To achieve this, Emolyst will carry real-time data analysis into effect in given time interval. Moreover, in Emolyst there will be more features since we are planning to integrate different algorithms for face, age, gender, race and emotion detection. Thus, this project will help industries and companies to improve their customers satisfaction levels and making profitable and smart choices.

As it is mentioned before, we are going to use face, gender, age and emotion detection algorithms. In this part, algorithms are going to be explained.

**OpenFace:** OpenFace is a Python and Torch implementation of face recognition with deep neural networks and is based on the CVPR 2015 paper FaceNet: A Unified Embedding for Face Recognition and Clustering. OpenFace has 97% accuracy when it is trained with nn4.small2.v1 model [3].

**yu4u/age-gender-estimation:** a Keras implementation of a CNN for estimating age and gender from a face image. In training, the IMDB-WIKI dataset is used [4].

**face-classification:** Real-time face detection and emotion/gender classification using fer2013/imdb datasets with a keras CNN model and openCV. face-classification has 96% accuracy of gender detection and 66% accuracy of emotion detection [5].

# 3. Proposed Software Architecture

In this section information about our system's high level design, especially our software architecture is provided.

## 3.1 Overview

In this section, Emolyst's software architecture is explained in detail. The first section includes the subsystem decomposition. Secondly, HW/SW mapping is explained. Then, in the following, persistent data management, Emolyst's access control and security, global software control and boundary conditions are explained in detail lastly.

## 3.2 Subsystem Decomposition

Emolyst will be composed of two different module. One of them is Video Processing Subsystem which will be directly connected to camera input and this module will process video streaming data and creates gender, age, emotion, face feature data on which analysis will be made.

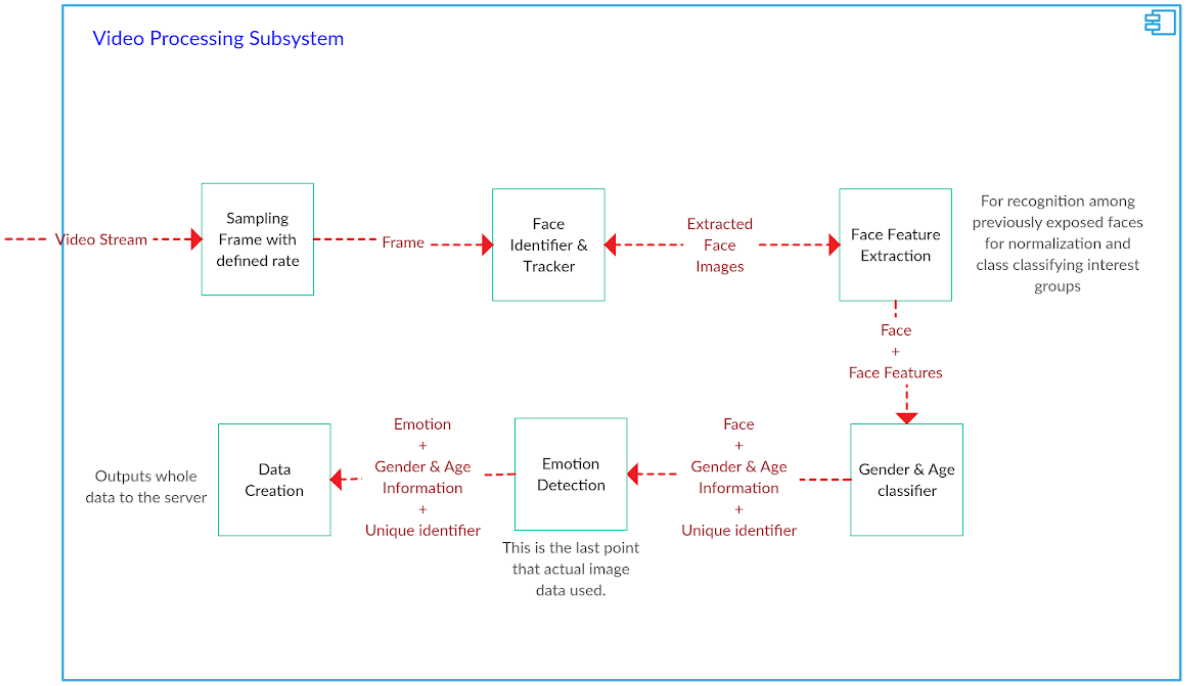


Figure 1 - Video processing Subsystem Decomposition

Data Analysis module will be working on centralized server working in Saas model. Every Video Processing Subsystem will subscribe to Data Analysis module with their unique id which identifies the item that corresponding video processing subsystem creates data for, and with their group id which identifies group that corresponding video input belongs. For example in an exhibition that Emolyst system is setted up each and where there are VPS's on each stand. Those VPSs will subscribe the Data Analysis Subsystem with unique id and group id which represent this specific exhibition.

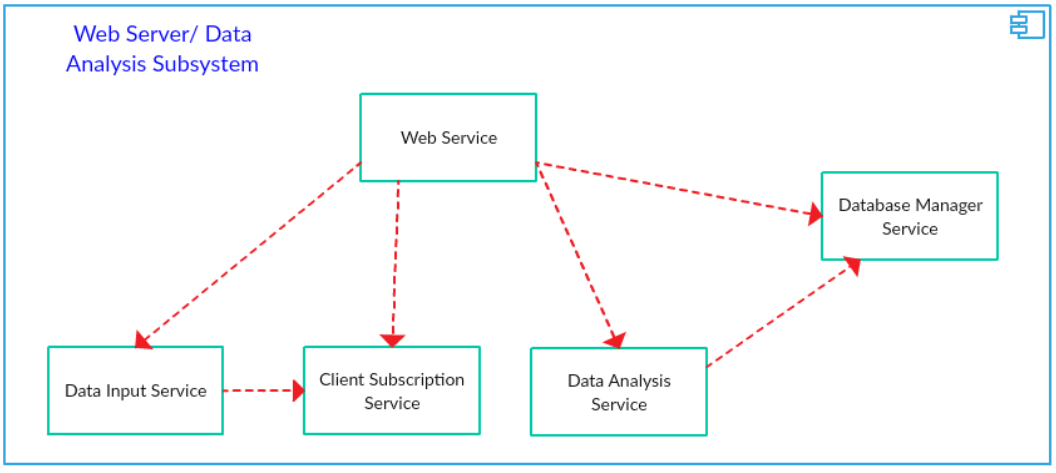


Figure 2 - Data Analysis Subsytem

## 3.3 Hardware/Software mapping

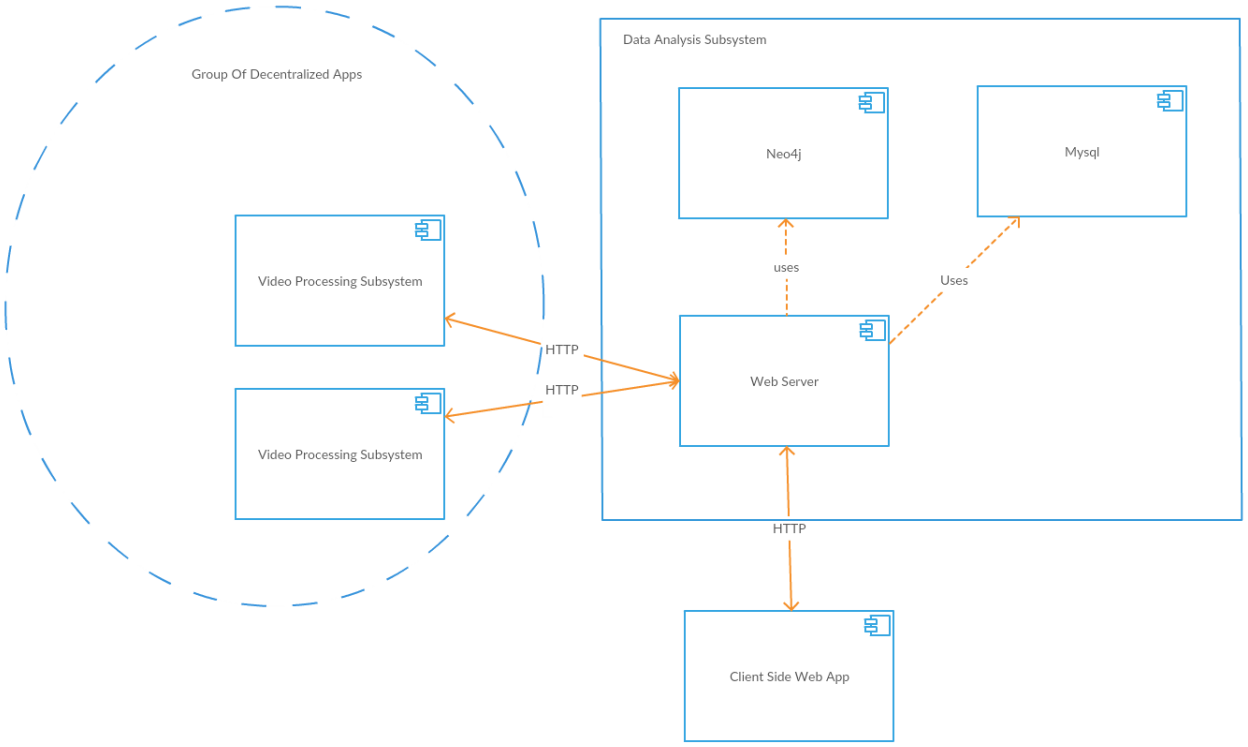


Figure 3 - Hardware Software Mapping

Emolyst will have distributed architecture on video capturing and video processing units. Videos captured by cameras are processed right away on the machines that are connected to cameras. This machines will be single board computers like Raspberry Pi[6] or Orange Pi[7]. With this approach only necessary data will be transferred over network to server. For communication between Video Processing Subsystem and Data Web Server inside Data Analysis Subsystem we will be using data streaming platforms like Apache Kafka[8] or AWS Kinesis[9].

On the server side pre processed data is analyzed, several aggregations will be made such as listing items by time spent reviewing it and user - item interactions will be generated as hypergraphs. The visualisation of the graphs will be made at the web client side.

## 3.4 Persistent Data Management

In Emolyst some data must be persistent, have a lifecycle beyond one execution. Therefore we need to store persistent objects in a DB. In the server side of Emolyst there will be two different data storages. One of them will be Neo4j[10] which is a graph DB to store relations and interactions of humans and items as a graph, and the other one will be mysql[11] which will be used to store extracted features from the video processing and also user, admin, subscription records will be stored here .

## 3.5 Access Control and Security

In Emolyst, we have one type of user which can be said “admin”. Anyone who wants to use our system can use the Emolyst. Users must sign up to the system with their email and password. Having signing up to the system, users have kind of admin authorization. Each user can see and edit their camera configurations. Each camera will have unique ID and each camera will be available only one user. Each user can see the processed data which was initially taken from their configured cameras and evaluate the data. The

In the aspect of security, server will be secure against security attacks.For example our DBs are protected against SQL injections to maintain reliability and truthness of the data in the system. One user’s data will be invisible to another user. Also any non user can not see any data via the web application. These are all the things that are thought during the implementation and provided with ensuring “secure programming” during implementation.

Emolyst also ensures data privacy of the users. Any user information is not used shared or by other users or third parties. Also faces that are recognized by the program not will be shared by any user of the Emolyst, is not stored anywhere and is not by seen any human being, only processed features such as gender, age range, emotion is provided to the user. Emolyst ensure anonymity of the people and their faces.

## 3.6 Global Software Control

In the design of Emolyst we decided to follow decentralized approach for Video Processing Subsystem. Each video input will be connected to single board computers like Raspberry Pi or Orange Pi and they will be processed on this single board computers on run time. Only processed data will be streamed to Server. Decentralized approach has 2 main advantages for our system.

- Network Cost: Only processed data will be transported over network rather than streaming whole video to centralized server.
- Privacy & Security: Streaming video data that contains face images may be violation of privacy of people and securing these data over network transmission is also difficult. With distributed approach face images processed locally and no network transmission or storage is needed for face images.

For the Data Analysis subsystem we followed centralized approach. Each video Processing Subsystem will subscribe with unique id and group id. With all data gathered from several VPS's cumulative analysis will take place on the Data Analysis Subsystem. Those analysis are :

- For each item average expression, change of the expression, time spend looking at this specific item among different age groups and different genders.
- Hypergraph creation and visualization in order to discover similarities between items and segmentation of people.

## 3.7 Boundary conditions

### 3.7.1 Initialization

First of all, users of Emolyst should sign up to the system with their email address and password if they don't have already an account via Web application interface.

Registered users need to login to the system in order to use the Emolyst and see the visualization of the data with their email and password via Web application interface.

For all of these user needs to have internet connection.

### 3.7.2 Termination

User can log out from the system via logout button in the Web application. If user did not logout, the system will log the user out after the session expired.

### 3.7.3 Failure

Failures may occur in 4 cases.

- We must make our server always be up. If server is down user cannot get the visualized data even though visualization is done at client side because the data is retrieved from the DB located in the host server.
- Second one occurs when user disconnects from the network. If user loses his/her internet connection web application will not response.
- Third one occurs when cameras are broken or failed.
- The other one occur when video processing component(it may be Raspberry Pi) has lost the connection to the our server in AWS cloud.

# 4. Subsystem services

## 4.1 Video Processing Subsystem

**Face Feature Extractor:** This class is responsible for extracting face features in an image.

**Face Identifier & Tracker:** This class is responsible for detecting and identifying faces in the frame. Having identifying, it tracks the identified faces.

**Gender Detector:** This class is responsible for detecting gender of the person in the frame.

**Age Detector:** This class estimates the age of the person with respect to extracted features.

**Emotion Detector:** Responsibility of this class is to detect the emotional state of the person in the frame.

**Data Creator:** This class gets all related results from gender, age and emotion detector classes and creates data from these results.

## 4.2 Data Analysis Subsystem

**Data Input Service:** This service is responsible for reading stream data coming from VPS's that are subscribed.

**Client Subscription Service:** This service is responsible for subscription of clients to data analysis module. VPS units subscribes server via this system with their unique id and group id.

**Data Analysis Service:** This service is responsible for analysis of the data.

**Database Manager Service:** This service is responsible for DB management. It controls the data flow in the subsystem between DB and all other services.

# 5. Glossary

**AWS:** Amazon Web Services. It is widely used secure cloud services platform.

**Raspberry Pi:** It is a low cost credit sized efficient computer widely used in embedded systems.

**CVPR:** Computer Vision and Pattern Recognition.

**SaaS:** It stands for at least three different "as-a-service" offerings and its expansion is Software as a service.

# 6. References

[1] “What is Kairos?”, *Kairos.com,* 2018. [Online]. Available: https://www.kairos.com/features. [Accessed: 29- Dec- 2018].

[2] “Face API”, *azure.microsoft.com,* 2018. [Online]. Available: https://azure.microsoft.com/en-us/services/cognitive-services/face/. [Accessed: 29- Dec- 2018].

[3] “OpenFace”, *cmusatyalab.github.io,* 2018. [Online]. Available: https://cmusatyalab.github.io/openface/#news. [Accessed: 30- Dec- 2018].

[4] “Age Gender Estimation”, *github.com,* 2018. [Online]. Available: https://github.com/yu4u/age-gender-estimation. [Accessed: 30- Dec- 2018].

[5] “Face Classification”, *github.com,* 2018. [Online]. Available: https://github.com/oarriaga/face_classification. [Accessed: 30- Dec- 2018].

[6] “Raspberry Pi”, *raspberrypi.org,* 2018. [Online]. Available: https://www.raspberrypi.org. [Accessed: 30- Dec- 2018].

[7] “Orange Pi”, *orangepi.org,* 2018. [Online]. Available: http://www.orangepi.org. [Accessed: 30- Dec- 2018].

[8] “Apache Kafka”, *kafka.apache.org.* [Online]. Available: https://kafka.apache.org. [Accessed: 30- Dec- 2018].

[9] “Amazon Kinesis”, *amazon.com,* 2018. [Online]. Available: https://kafka.apache.org. [Accessed: 30- Dec- 2018].

[10] “Neo4j - The Fastest Path To Graph Success”, *neo4j.com,* 2018. [Online]. Available: https://neo4j.com. [Accessed: 30- Dec- 2018].

[11] “MySQL”, *mysql.com,* 2018. [Online]. Available: https://www.mysql.com. [Accessed: 30- Dec- 2018].