SOFTWARE CONSTRUCTION

Final Project

ISTANBUL AYDIN UNIVERSITY
Florya Campus

Face Detection System

Version 1.0 approved

Prepared by:

Hasan Sidawi	**B1705.090059**
Samiullah Niazi	**B1605.090057**
Majed Bawarshi	**B1605.090072**
MHD Omar Albahra	**B1605.090065**

Supervisor: **Assoc. Prof. (Ph.D.) ILHAM HUSEYINOV**

Table of Contents

1. SHORT DESCRIPTION OF SCENARIO:

The person stands in front of the camera the system will recognize the face and give him access, otherwise it rejects.

2. USE CASE:

1. Scope:

Face Recognition System.

2. Primary Actor:

Camera.

3. Stakeholders and Interests:

- Visitors: People who are invited and want to enter the event.
- Manager: Main responsible person, who controls entrance failure and debug technical problems.
- Staff: Team working under the manager, to organize the event.
- Company: the company that owns the place wants to increase the security of its place and satisfy customer interest.
- Hosts (Inviters): People in charge of inserting invited people data into the database.

4. Preconditions:

- The camera is activated and the monitor is on.
- All Invited people data is stored.

5. Success Guarantee (Post-Conditions):

- The system records each visitor after they enter the event.
- Door Opens
- Monitor prints a “Welcome Message” for the recognized person.

6. Main success scenario:

1. The visitor arrives to the event.
2. The visitor stands in front of the camera that is connected to the system.
3. The system captures and detects the face in camera range.
4. The system extracts the identified face features, and compare it with the existed templates in the database.
5. When the system finds the scanned face in the database the system prints welcome to the screen and the door opens.
6. The system resets.

7. Extensions (or Alternative flows):

1. Captured face is not in the database:
    a) If face features do not match to the recorded data, then the system rejects.
    b) Prints that person is not recognized.
2. No enough details captured from face:

a) If the face is far from the camera, the system prints an error message.
b) If the face is partially covered by objects, the system prints an error message.

3. Focus failure:
   a) If the face is not still, the system prints an error message.

8. Special Requirements:

1. High-quality camera, that detects all details from 1 meter at most.
2. Monitor of size 22” inches.

3. OPERATION CONTRACTS:

Operation:isFaceDetected()
Cross References: Use Case: Face Recognition
Preconditions:
- There’s a recognition underway

PostCondition:
- It will return a boolean whether is the object face or not.
- It will make an association with FaceFeatures.

Operation:getEventInfo(d:Date, t:Time)
Cross References: Use Case: Face Recognition
Preconditions:
- There’s a recognition underway.

Postconditions:
- Information of the event will be returned (id, visitors).

Operation:getVisitors(eventID:Text)Cross References: Use Case: Face Recognition
Preconditions:
- There’s a recognition underway.

Postconditions:
- An array of Visitor that contains (ID, FullName, Gender, BirthDate) will be returned based on the event id from the database.

Operation:CompareToTemplate(x:Face, eventID:Text)
Cross References: Use Case: Face Recognition
Preconditions:
- There’s a recognition underway.

Postconditions:

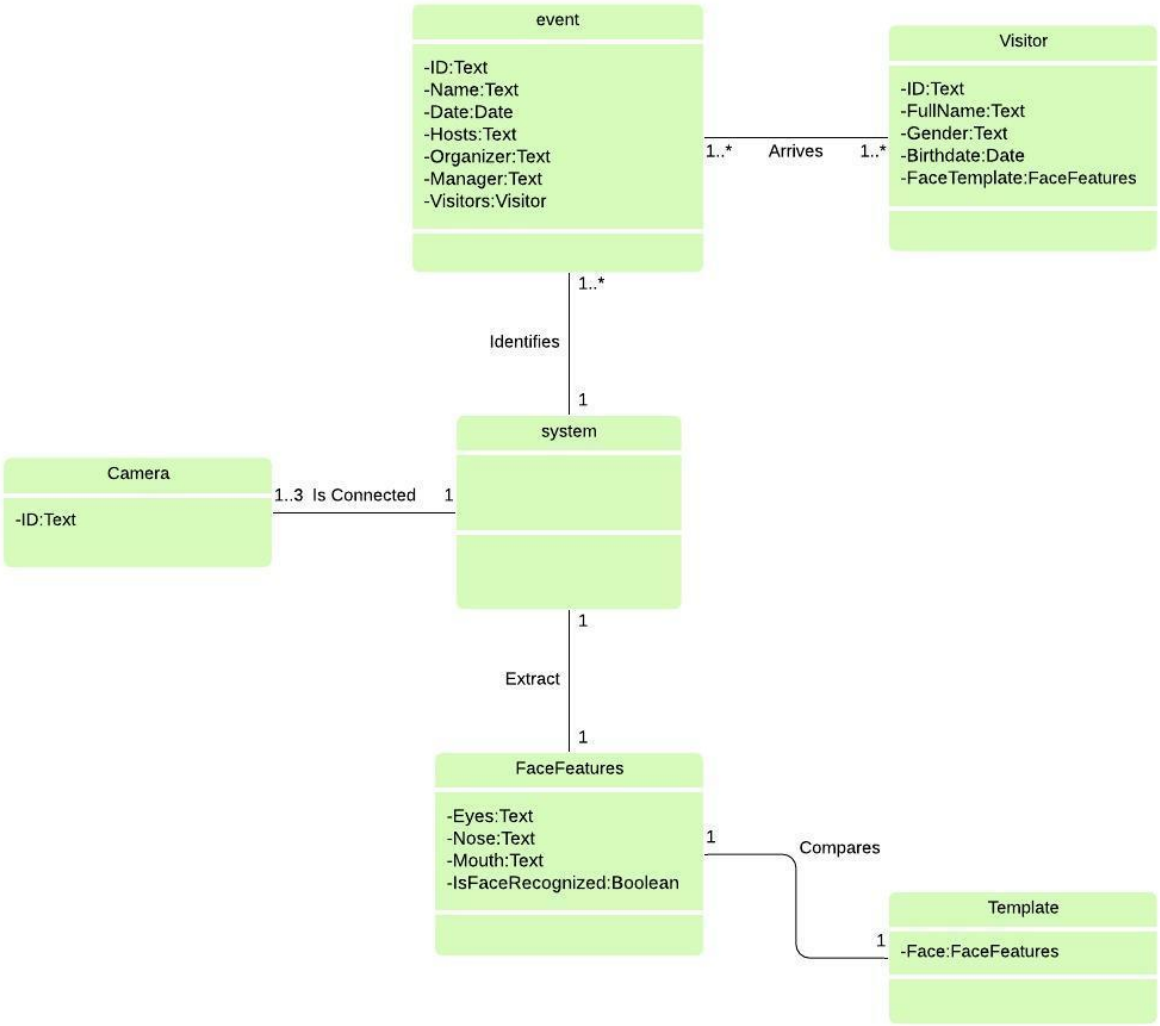
- The face will be compared with the faces in the database and a boolean value will be returned, to tell if the face is recognized (match with a face in the database, using the algorithm).

Operation: getFaceFeatures(imageInput)

Cross References: Use Cases:Face Recognition
Preconditions:There is a recognition underway.
Postconditions:

- System receives face features.
- FaceFeatures.eyes was set to extract.
- FaceFeatures.nose was set to extract.
- FaceFeatures.mouth was set to extract.
- Facefeatures was associated with the Template, based on eyes, nose and mouth match.

4. DOMAIN MODEL DIAGRAM (DEVELOPED BY GRAMMAR METHOD):



1. candidates of verbs or associations:

- Arrive
- Identify
- Detect
- Extract
- Compare
- Print
- Find
- Open
- Reset

2. candidates of classes:

- System
- Visitor
- Camera
- Event
- Face features
- Template

3. candidates of methods or operation contracts:

- isFaceDetected()
- getEventInfo(d:Date, t:Time)
- getVisitors(eventID:Text)
- CompareToTemplate(x:Face, eventID:Text)
- getFaceFeatures(imageInput)

4. candidates of attributes:

- ID
- FullName
- Date
- Gender
- Birthdate
- Face
- Manager
- Organizer
- Visitors
- Face
- Eyes
- Nose
- Mouth
- Hosts

5. Nouns:

- system
- Visitor
- Person’s Face
- Event
- Face features
- Camera

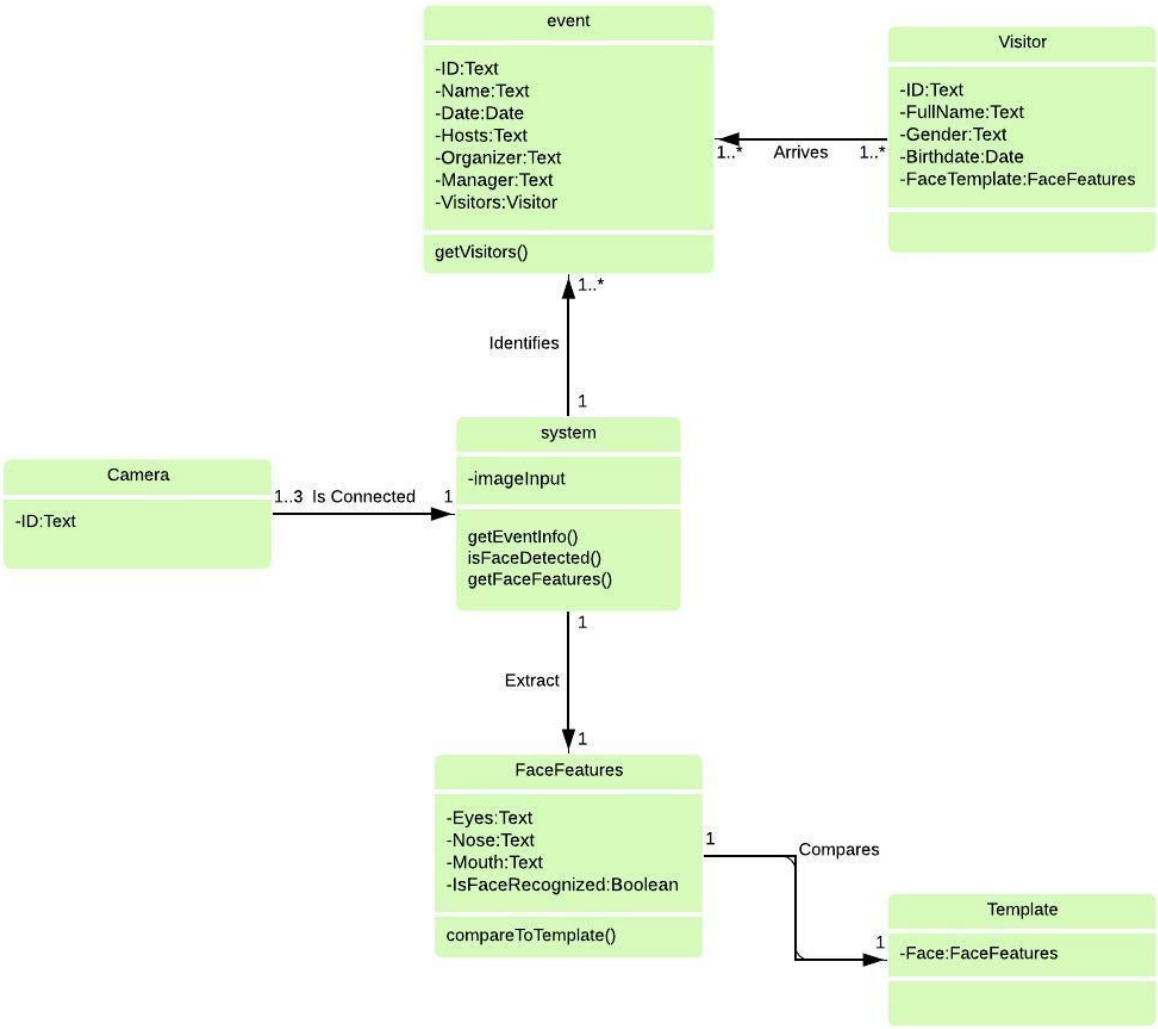
6. Attributes:

- Visitor:
  - ID
  - FullName
  - Gender
  - Birthdate
  - Face
- Face Features:
  - Eyes
  - Nose
  - Mouth
- Event:
  - ID
  - Name
  - Date
  - Hosts
  - Organizer
  - Manager
  - Visitors
- Template:
  - Face.
- Camera:
  - ID.
- System.

7. Verbs:

- Arrive
- Identifies
- Is connected
- Extracts
- Compares

5. CLASS DIAGRAM:



1. PATTERN:

a) EVENT IS CREATOR. IT CREATES VISITOR OBJECT.

b) SYSTEM IS THE MAIN CONTROLLER.

c) FACEFEATURES CLASS IS THE INFORMATION EXPERT.IT CALCULATES FACE FEATURES.

6. CODE OF THE CLASS DIAGRAM IN JAVA:

```java
package FaceRecognitionSystem;

import java.util.Date;

public class Visitor {

    private String ID;
    private String fullName;
    private String gender;
    private Date birthDay;
    private FaceFeatures faceTemplate;

}
```

```java
package FaceRecognitionSystem;

public class Template {

    private FaceFeatures face;

    public Boolean compareToTemplate(FaceFeatures face,
String eventID){

    }

}
```

```java
package FaceRecognitionSystem;

import java.awt.*;
import java.sql.Time;
import java.util.Date;

import FaceRecognitionSystem.Event;

public class System {

    Image imageInput;

    public static void main (String[] args){

        getEventInfo(Date d, Time t);
        isFaceDetected();
        getFaceFeatures(imageInput);

    }
    public Boolean isFaceDetected(){

    }
}
```

```java
package FaceRecognitionSystem;

import java.awt.*;

public class FaceFeatures {

    String eyes;
    String nose;
    String mouth;
    Boolean isFaceRecognized;

    public FaceFeatures FaceFeatures(Image imageInput){

        FaceFeatures fs = new FaceFeatures();
        return fs;
    }

}
```

```java
package FaceRecognitionSystem;

import com.sun.org.apache.bcel.internal.generic.Visitor;

import java.sql.Time;
import java.util.Date;

public class Event {

    private String ID;
    private String name;
    private Date date;
    private String host;
    private String orgnizers;
    private String manager;
    private Visitor[] visitors;

    public String getEventInfo(Date d, Time t){
        return this.ID;
    }

}
```

```java
package FaceRecognitionSystem;

public class Camera {

    String ID;

}
```

7. REFERENCES:

https://www.researchgate.net/publication/220566092_Face_Recognition_A_Literature_Survey

https://www.computer.org/csdl/trans/tp/1996/10/i1024.html

https://www.slideshare.net/dinakan1/07-contracts

https://www.lucidchart.com