Software Requirements Specification Document
Arabic Handwriting Forgery Detection

Sylvia Hani, Caroline Kamal, Diana Abdelnasser, Hatem Herzawy.
Supervised by: Dr.Sherin Moussa , Eng. Youssef Mohamed

November 2017

1 Introduction

1.1 Purpose of this document

The purpose of this Software Requirements Specification document is to outline the functional and non-functional requirements for Arabic handwritten Forgery detection documents. it will be build using MATLAB. It will be computer based application. Our target audience are the end users and the committee of our graduation projects for the year 2018.

1.2 Scope of this document

In this document, we are targeting the users who are in need of handwritten document verification like banks or business owners, who will be using our system to detect the forgery. We are under a time constraint of 9 months.

1.3 Overview

Increasing the accuracy rates of detecting forgery in Arabic handwritten documents. The user can know the authenticity of the handwriting, and detecting the forger when supplied with different writers data-set. The software will focus on document forgery detection specified Arabic handwriting style and font.

1.4 Business Context

Arabic handwritten forgery detection system is a recommended issue for important documents of companies. Detecting forgeries with higher accuracy than the current commercial systems, can give the users the trust to consume our system in their businesses.The idea of forgery software will be adopted in related businesses such as banks and companies that need secured contracts or any other kind of important documents.

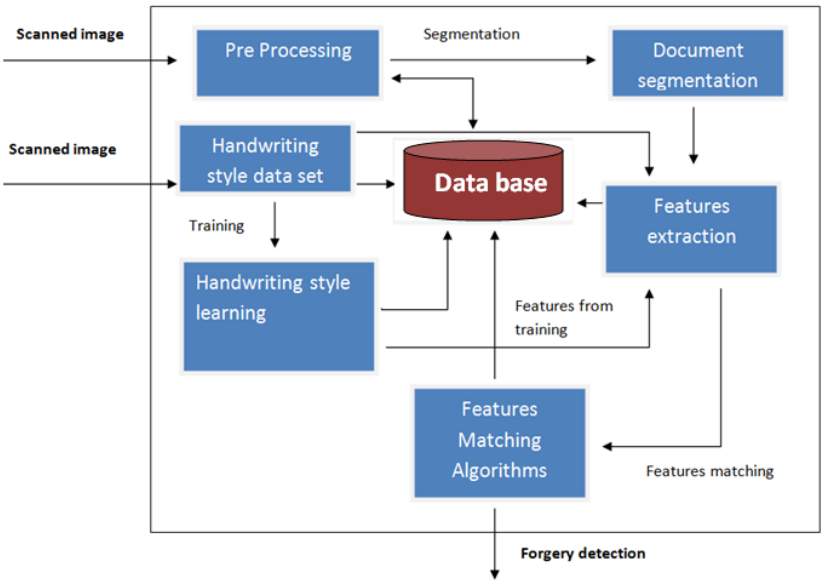


Figure 1: System overview

2 General Description

2.1 Product Functions

1. User can detect the authenticity of the document. whether it's forged or not.
2. User can have accurate detection of forged handwritten document.
3. User stores collective data sets.
4. System has different users.

2.2 Similar System Information

The similar system is writer recognition based on Edge-Hinge Distribution, Propose the detection of handwritten document by extracting different features of the style of handwriting. In order to apply those features, it uses some scanned images of characters as inputs, then introducing the required image doubting that it's forged to the features extracted to be matched with another images in the database based on different algorithms as Euclidean distance, Manhattan distance, distributive. Using IAM Handwriting Database, the system gets the image with the most matching extracted features measurements.This System is 98 percent accurate.

2.3 User Characteristics

2.3.1 User

1. Must have basic knowledge of using computers and web browsers.
2. Understanding of the System's functionality procedures

2.4 User Problem Statement

Improving system to have a system accurate enough to detect his own detailed style in the data-set to make the system trustful for the user.

2.5 User Objectives

The user objective by using our system is detecting:

1. The percentage of similarity and differences of the 2 images of handwritten document.
2. -The specific family that his own handwriting belongs.

2.6 General Constraints

1. The images has to be black and white scanned.
2. Images with 300 DPI.
3. Specific handwriting font size,style "Naskh", pen type.

3 Functional Requirements

3.1 Class 1

3.1.1 ID: 1

- Title : Manage User
- Description : The Admin manage all the users and can add,modify,delete,search and list users.
- Input : First Name, Last Name,User Name, Email, Password, Telephone, Gender, Job.
- Action : Checks if all fields are filled and if so the data is entered in a new record in the database accordingly.
- Output : Confirmation message or error message if something went wrong upon validating the fields.
- Pre-condition : None.

- Post-condition : If Add so the database will be updated with the new user ,if modify the user so database will edit the user information ,if the admin delete the user so it will be deleted from the database etc.. .

3.1.2 ID: 2

- Title: User Log-in.
- Description: The User logs into his account.
- Input: User's Email and password.
- Action: Checks if all fields are filled and compares data entered to that in the database records.
- Output: The homepage is previewed(User's account) and Log-in successful message or error message upon validating the fields.
- Pre-condition: User is already registered in the database.
- Post-condition: Redirected to the User's account page.

3.1.3 ID: 3

- Title: Segmentation.
- Description: Documents are being extracted from documents to lines, from lines to words, from words to characters.
- Input: Image name.
- Action: Segmentation features are applied to the image.
- Output: Documents have been segmented to lines, words and characters and stored to the database.
- Pre-condition: Desired image already stored in the database.
- Post-condition: segmented documents are to be stored in the database.

3.1.4 ID: 4

- Title: Matching.
- Description: Features of same document to matched with another features to detect the accuracy between them.
- Input: Image name.
- Action: Features of an image to another are to be retrieved from the database to be matched and the result is stored again in the database

- Output: none.
- Pre-condition: Desired features already stored in the database.
- Post-condition: None.

3.1.5 ID: 5

- Title: Manage document
- Description: The user can add, delete, list or update a document of a specific writer
- Input: Data set of Images
- Action: Add/Delete/List or Update a document.
- Output: Confirmation notification with the action approved
- Pre-condition: For the delete/update actions there must be an exist document.
- Post-condition: Data set is successfully added/deleted/listed or updated

3.1.6 ID: 6

- Title: User's data Encryption
- Description: The system must encrypt the user's data
- Input: User's data like ID, password
- Action: Encrypting of the user's data
- Output: User's data encrypted
- Pre-condition: There must an existing a valid user's data
- Post-condition: User's data successfully encrypted

3.1.7 ID: 7

- Title: User's data decryption
- Description: The system must decrypt the user's data
- Input: User's data encrypted like ID, password
- Action: Decryption of the user's data
- Output: User's data decrypted
- Pre-condition: There must be exist a valid encrypted user's data
- Post-condition: User's data successfully decrypted

3.1.8 ID: 8

- Title: Document's path Encryption
- Description: The system must encrypt the path of the document against any hacking or ear dropping.
- Input: Document ID, Document path.
- Action: Encryption of the document path.
- Output: document path is encrypted.
- Pre-condition: at least one Document path stored n the database.
- Post-condition: Document's path successfully encrypted and stored in the database.

3.1.9 ID: 8

- Title: Document's path Decryption
- Description: The system must decrypt the path of the document after encryption when restored from the database by one of the authorized admins or users.
- Input: Document ID, Document path.
- Action: Decryption of the document path.
- Output: document path is decrypted.
- Pre-condition: at least one Document path stored n the database.
- Post-condition: Document's path successfully decrypted and retrieved from the database.

3.1.10 ID: 10

- Title: Document Feature extraction
- Description: The user can extract the document features of a specific writer
- Input: Image document
- Action: Document‘s features are being extracted
- Output: List of features values extracted
- Pre-condition: The document must be scanned as an image
- Post-condition: None

3.1.11 ID: 11

- Title: Data set training.
- Description: The system trains a group of images "data-set" that belongs to the same writer based on Neural networks algorithm to be able to identify the writer's handwriting in the future.
- Input: Array of Image documents
- Action: images are being trained to be identified as a range of handwriting that belongs to the same writer.
- Output: values that represent the range between the handwriting's owner.
- Pre-condition: at least one images stored in the database.
- Post-condition: none.

4 Interface Requirements

4.1 User Interfaces

4.1.1 GUI

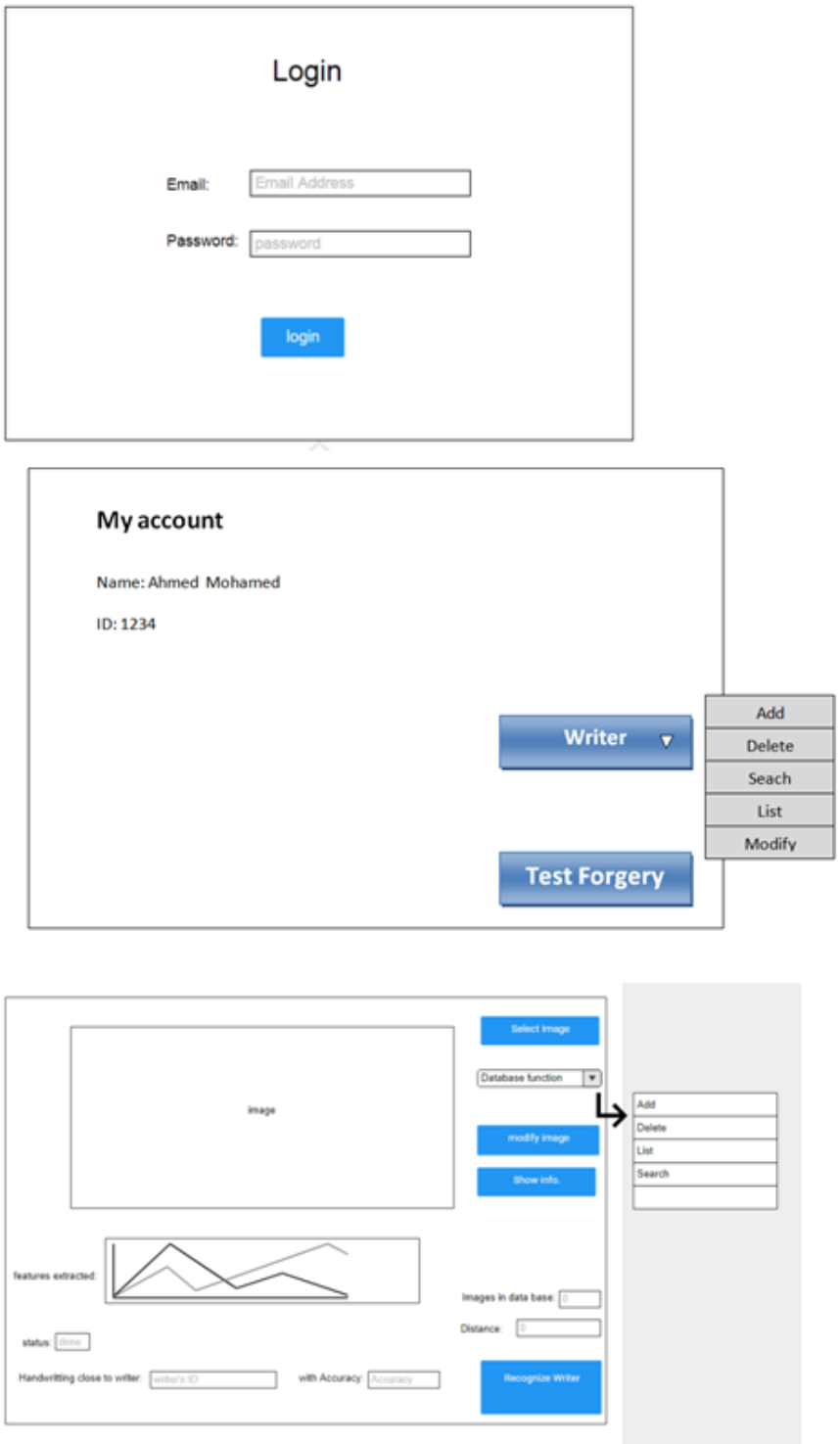


Figure 2: Main user interface

4.1.2 API

4.2 Hardware Interfaces

The system uses Scanner for image collection, stored in IAM database, and a screen to display he output on.

5 Performance Requirements

The system uses Scanner that 2400 x 4800 DPI DPI, duty scanning cycle 26,000 scan/min. The hard disk storage requirement depends on the company using the system.

6 Design Constraints

6.1 Hardware Limitations

The system highest resolution is 2400 x 4800 DPI, speed for 10 x 15 cm (600 DPI) is about 20 sec maximum

7 Other non-functional attributes

7.1 Performance

- Name : System must be interactive and maintains it's speed.
- Type : Performance requirement.
- Date : 2/6/2018
- Description : System should be very interactive and reduce any delays that occurs in the system, such as feature extraction process or matching process, because it's processed at real time.
- Priority : 10/10

7.2 Security

- Name : Password Encryption, System log
- Type : Security requirement
- Date : 2/6/2018
- Description : the system should encrypt the user's password so no one can ear drop into their documents, System should keep track of Customer's behavior when using the System.

- passwords : to ensure the security of the user verification.
- Priority : 10/10

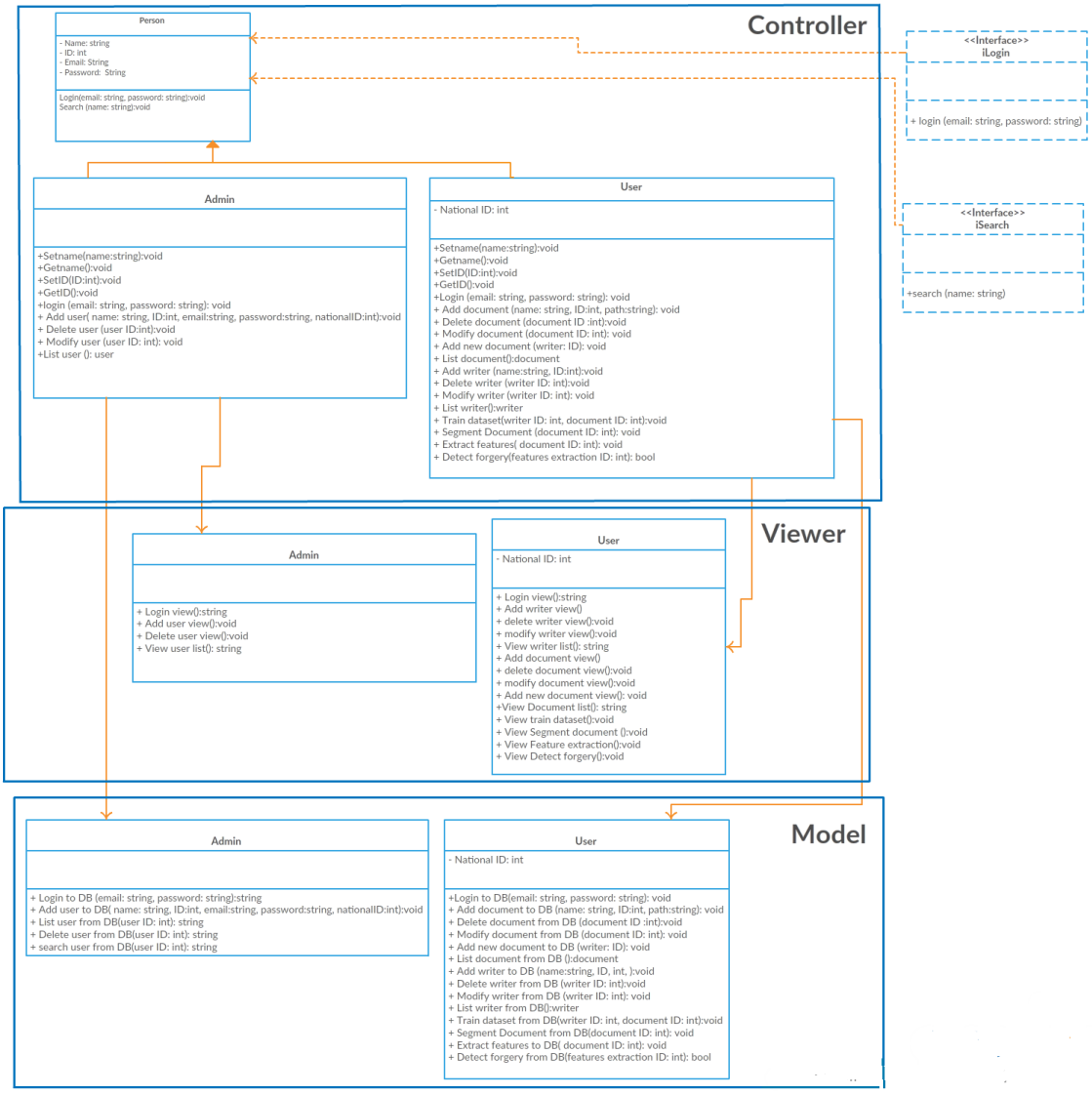
7.3 Maintainability

- Name : ease of system maintainability
- Type : Maintainability
- Date : 2/6/2018
- Description : Ensure the system ease of maintainability through the implementation of MVC Design patterns, Any change in the System design won't harm the System's whole architecture.
- Priority : 10/10

7.4 Reliability

- Name : Insurance of system reliability
- Type : Reliability requirement
- Date : 2/6/2018
- Description : The system should be reliable and trust worthy to the User, as the system would do well training of each individual writer's data-set to be able to know their personal style and then by many features matching algorithm; higher accuracy detection would occur.
- Priority : 10/10

8 Preliminary Object-Oriented Domain Analysis


Controller
Person
- Name: string
- ID: int
- Email: String
- Password: String
Login(email: string, password: string):void
Search (name: string):void
Admin
+Setname(name:string):void
+Getname():void
+SetID(ID:int):void
+GetID():void
+login (email: string, password: string): void
+ Add user(name: string, ID:int, email:string, password:string, nationalID:int):void
+ Delete user (user ID:int):void
+ Modify user (user ID: int): void
+List user (): user
User
- National ID: int
+Setname(name:string):void
+Getname():void
+SetID(ID:int):void
+GetID():void
+Login (email: string, password: string): void
+ Add document (name: string, ID:int, path:string): void
+ Delete document (document ID :int):void
+ Modify document (document ID: int): void
+ Add new document (writer: ID): void
+ List document():document
+ Add writer (name:string, ID:int):void
+ Delete writer (writer ID: int):void
+ Modify writer (writer ID: int): void
+ List writer():writer
+ Train dataset(writer ID: int, document ID: int):void
+ Segment Document (document ID: int): void
+ Extract features(document ID: int): void
+ Detect forgery(features extraction ID: int): bool
<<Interface>>
iLogin
+ login (email: string, password: string)
<<Interface>>
iSearch
+search (name: string)
Viewer
Admin
+ Login view():string
+ Add user view():void
+ Delete user view():void
+ View user list(): string
User
- National ID: int
+ Login view():string
+ Add writer view()
+ delete writer view():void
+ modify writer view():void
+ View writer list(): string
+ Add document view()
+ delete document view():void
+ modify document view():void
+ Add new document view(): void
+View Document list(): string
+ View train dataset():void
+ View Segment document ():void
+ View Feature extraction():void
+ View Detect forgery():void
Model
Admin
+ Login to DB (email: string, password: string):string
+ Add user to DB(name: string, ID:int, email:string, password:string, nationalID:int):void
+ List user from DB(user ID: int): string
+ Delete user from DB(user ID: int): string
+ search user from DB(user ID: int): string
User
- National ID: int
+Login to DB(email: string, password: string): void
+ Add document to DB (name: string, ID:int, path:string): void
+ Delete document from DB (document ID :int):void
+ Modify document from DB (document ID: int): void
+ Add new document to DB (writer: ID): void
+ List document from DB ():document
+ Add writer to DB (name:string, ID, int,):void
+ Delete writer from DB (writer ID: int):void
+ Modify writer from DB (writer ID: int): void
+ List writer from DB():writer
+ Train dataset from DB(writer ID: int, document ID: int):void
+ Segment Document from DB(document ID: int): void
+ Extract features to DB(document ID: int): void
+ Detect forgery from DB(features extraction ID: int): bool

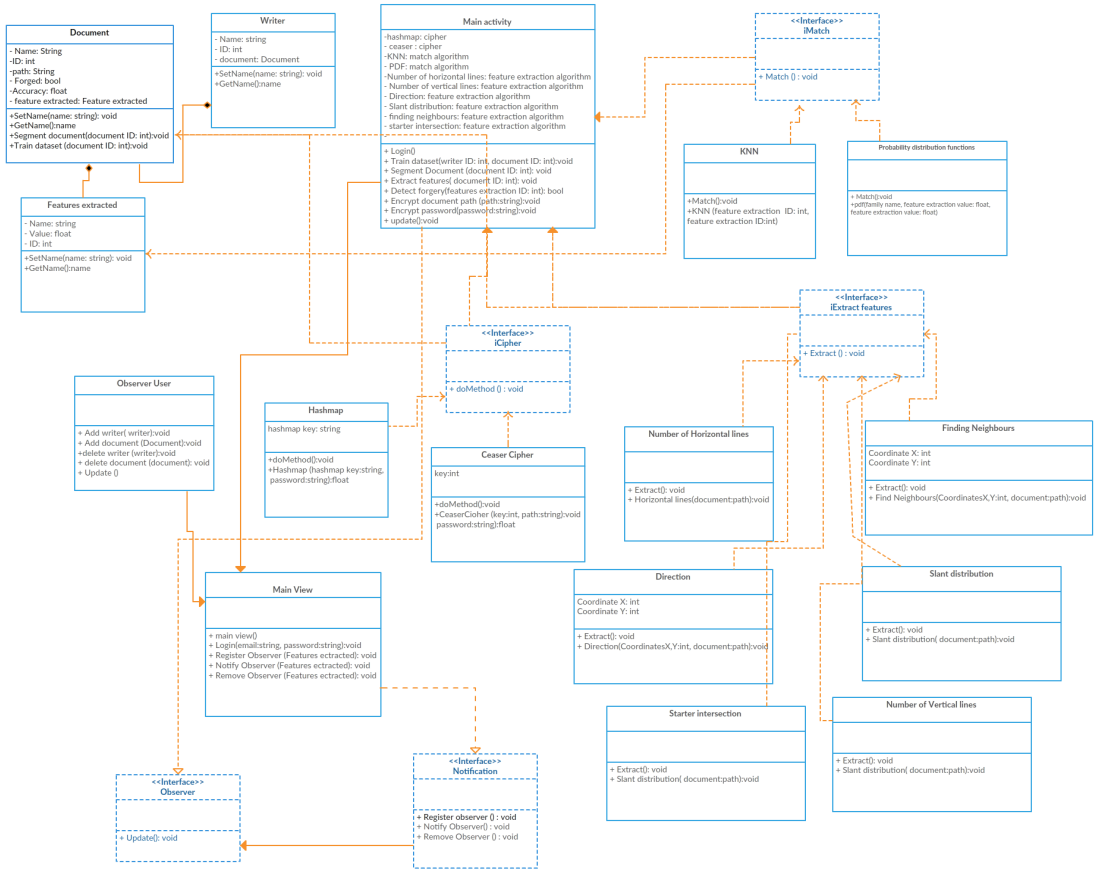


Figure 3: Primitive class diagram

8.1 Class descriptions

8.1.1 Class name

- Concrete: Person
- Concrete: Admin
- Concrete: User
- Concrete: Writer
- Concrete: Image
- Concrete: Feature

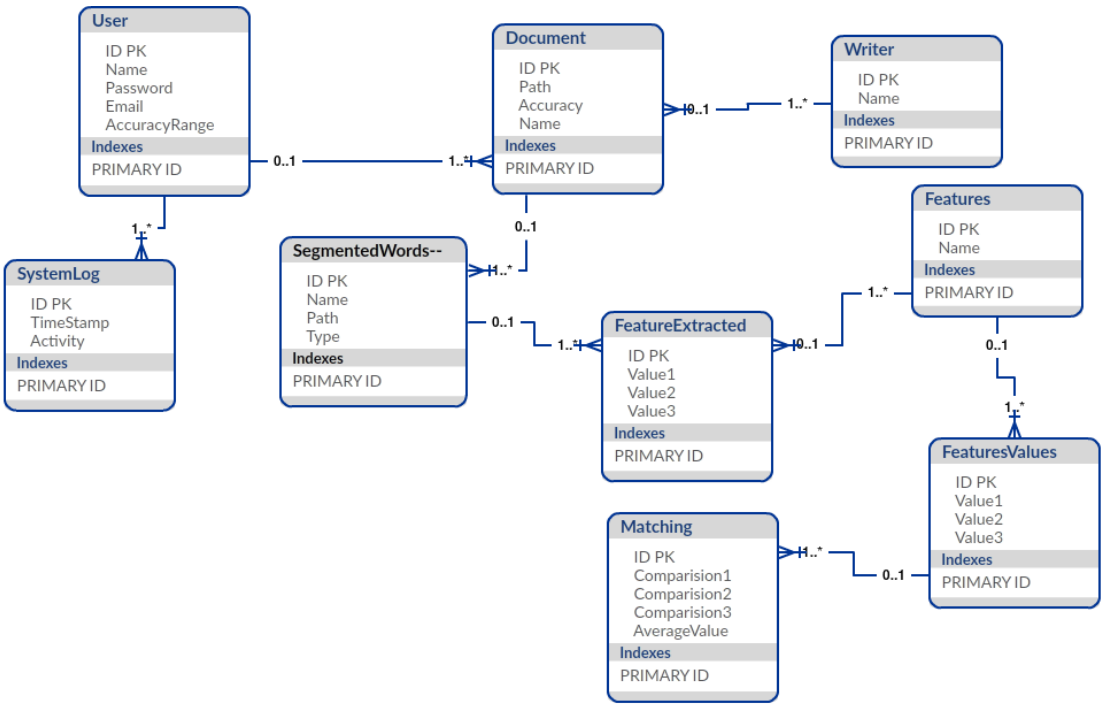


Figure 4: Primitive Database Diagram

8.1.2 List of Super-classes

Class: Person

8.1.3 List of Sub-classes

Class: Admin Class: User

8.1.4 Purpose

- Person : this is the parent class of all the users that may use the system.
- Admin : This class it the user with the highest privilege in the system which has the ability to manipulate the Customer.
- User : Is the main user of the system that will use the main function of the system which is the forgery detection or writer identification.
- Image : The class that contains the main attributes of an image of a document.
- Feature: The class that define each feature to e extracted.

9 Operational Scenarios

Here will be illustrated some of the scenarios that are shown in the system's use case diagram.

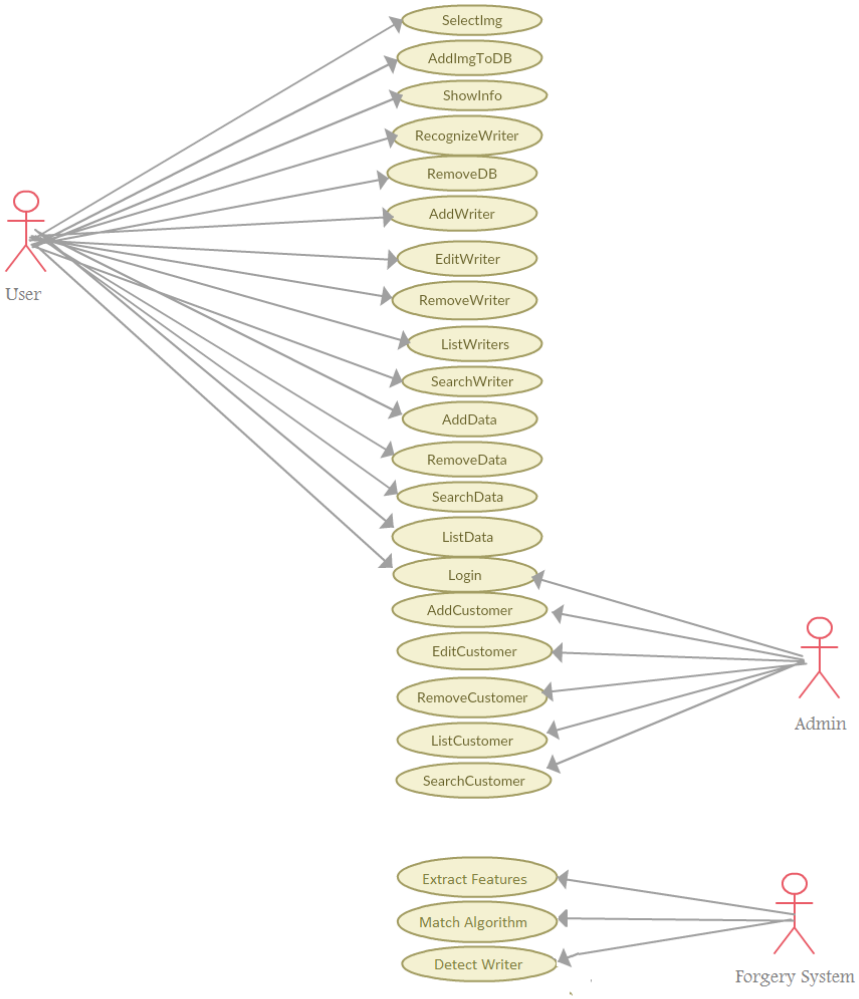


Figure 5: Use Case diagram

9.1 User scenario

The user included in this scenario is the admin. The admin is in control of manipulating the users in the system and this includes:

1. adding a user
2. Editing information of users.
3. Removing users
4. Listing all users
5. searching for a specific user

9.2 Scenario 2

The User who control the inputs in database:

1. Selecting adding to database.
2. Remove from database.
3. Add remove search list data set.
4. Add remove edit list Writer.

9.3 Scenario 3

Detecting and classifying forgery handwriting:

These actions are done by the system that the system detect the feature extraction of the handwriting image and classify the matching algorithm then the last step recognition of the writer that the output of the system says that if the writer write this document or not.

10 Preliminary Schedule Adjusted

11 Preliminary Budget Adjusted

1- HP Scanjet 200 Flatbed Photo Scanner - 1,420 EG

12 Appendices

12.1 Definitions, Acronyms, Abbreviations

– MATLAB : (matrix laboratory) is a multi-paradigm numerical computing environment, plotting of functions and data, implementation of algorithms, creation of user interfaces, including Java libraries.

Figure 6: Time Schedule

- Genetic Algorithms : is a method for solving both constrained and unconstrained optimization problems based on a natural selection process that mimics biological evolution.
- The IAM Handwriting Database : contains forms of handwritten English text which can be used to train and test handwritten text recognizes and to perform writer identification and verification experiments.

12.2 Collected material

- Arabic Alphabet Data-set.
- Arabic text Data-set.
- Arabic most common used words Data-set.

13 Requirements Matrix

14 Appendecies

1. KNN : K-Nearest Neighbor classifier.
2. HMM : Hidden Markov Model classifier.
3. SVM : Support Vector Machine classifier.
4. Haar- Cascade classifier.
5. PDF : Probability Distribution functions.

6. ANN : Artificial Neural Notworks.

References

[1] Vlad Atanasiu,Laurence Likforman-Sulem,Nicole, Vincent.Writer Retrieval - Exploration of a Novel Biometric Scenario Using Perceptual Features Derived from Script Orientation.Beijing, China.03 November 2011

[2] Marius Bulacu,Lambert Schomaker,Louis Vuurpijl.Writer Identification Using Edge-Based Directional Features .AI Institute, Groningen University The Netherlands.August 03 - 06, 2003 .

[3] Hamdi Boukamcha.Writer Recognition System V2 : Simple and Effective Source Code For Write Recognition Based Edge-Hinge Distribution.Tunisia.Dec 13, 2014.

[4] Dinesh Dileep, A feature extraction technique based on character geometry for character recognition.

[5]] Lei Hu, Richard Zanibbi "HMM-Based Recognition of Online Handwritten Mathematical Symbols Using Segmental K-means Initialization and A Modified Pen-up/down Feature". In: Department of Computer Science Rochester Institute of Technology Rochester, USA, 2011.

[6] Hans-Jürgen Winkler, "HMM-BASED HANDWRITTEN SYMBOL RECOGNITION USING ON-LINE AND OFF-LINE FEATURES" In: Institute for Human-Machine-Communication Munich University of Technology Arcisstr. 21, 80290 Munich, Germany, May 1996.

[7] MUSTAFA BERKAY YILMAZ, "OFFLINE SIGNATURE VERIFICATION WITH USER-BASED AND GLOBAL CLASSIFIERS OF LOCAL FEATURES". In: Submitted to the Graduate School of Engineering and Natural Sciences in partial fulfillment of the requirements for the degree of Doctor of Philosophy Sabancı University. February 2015