



# UNIVERSITÀ DEGLI STUDI DI MESSINA

## DIPARTMENT:

Scienze matematiche e informatiche, scienze fisiche e scienze  
della terra

## DEGREE COURSE:

INFORMATICA

---

## Programmazione II



**Idol Recognition System**

By: Tien M. Nguyen

Guided by: Professor Distefano Salvatore

---

Academic year 2017-2018

# Idol Recognition System

<b>I.</b>	<b>Ideal.....</b>	<b>4</b>
1.	<i>Ideal of this Application.....</i>	4
2.	<i>Some other application of this ideal.....</i>	4
<b>II.</b>	<b>Overall architecture and technology use.....</b>	<b>5</b>
1.	<i>Overall architecture.....</i>	5
a.	<i>Front-end.....</i>	5
b.	<i>Back-end .....</i>	5
c.	<i>Database.....</i>	5
d.	<i>Encryption of information .....</i>	5
<b>III.</b>	<b>Spring Framework .....</b>	<b>6</b>
1.	<i>What is Spring?.....</i>	6
2.	<i>Architecture, modules of Spring Framework .....</i>	7
3.	<i>Why Spring? .....</i>	7
4.	<i>Some other projects of Spring .....</i>	8
<b>IV.</b>	<b>Spring Boot .....</b>	<b>9</b>
1.	<i>What is Spring Boot?.....</i>	9
2.	<i>Why Spring Boot?.....</i>	9
<b>V.</b>	<b>MongoDB .....</b>	<b>10</b>
1.	<i>NoSQL .....</i>	10
2.	<i>MongoDB.....</i>	10
<b>VI.</b>	<b>Java Swing Framework .....</b>	<b>11</b>
1.	<i>What is Java Swing Framework?.....</i>	11
2.	<i>Why Java Swing?.....</i>	11
3.	<i>Container Class.....</i>	11
4.	<i>Java Swing Class Hierarchy.....</i>	11
<b>VII.</b>	<b>Microsoft Azure Cognitive Services .....</b>	<b>12</b>
1.	<i>What is Microsoft Azure Cognitive Services?.....</i>	12
2.	<i>Support services.....</i>	12
3.	<i>Why Microsoft Azure Cognitive Services?.....</i>	12
4.	<i>What will be used in this project?.....</i>	12
<b>VIII.</b>	<b>Bing Image Search API.....</b>	<b>13</b>
1.	<i>What is Bing Image Search API?.....</i>	13
2.	<i>Bing Image Search features.....</i>	13

3.	<i>Workflow</i> .....	13
4.	<i>What will be used in this project?</i> .....	13
<b>IX.</b>	<b>Azure Face Service</b> .....	<b>14</b>
1.	<i>What is Azure Face Service?</i> .....	14
2.	<i>Azure Face Service features</i> .....	14
3.	<i>What will be used in this project?</i> .....	14
<b>X.</b>	<b>Another technology</b> .....	<b>16</b>
1.	<i>RSA 2048</i> .....	16
2.	<i>AES 256</i> .....	16
3.	<i>MongoDB Java Drivers</i> .....	16
4.	<i>HttpClient and BSON for Java</i> .....	16
<b>XI.</b>	<b>Idol Recognition System</b> .....	<b>17</b>
1.	<i>What is Idol Recognition System?</i> .....	17
2.	<i>Idol Recognition System Structure</i> .....	17
<b>XII.</b>	<b>Idol Recognition System Structure</b> .....	<b>18</b>
1.	<i>Use Case</i> .....	18
2.	<i>Models</i> .....	18
3.	<i>Database Diagram</i> .....	18
4.	<i>Workflow</i> .....	19
5.	<i>Rest API list</i> .....	19
a.	For Client App .....	19
b.	For Admin App.....	19
<b>XIII.</b>	<b>Class Diagram</b> .....	<b>21</b>
1.	<i>What is Class Diagram?</i> .....	21
2.	<i>UML Class Notation</i> .....	21
3.	<i>Class Diagram for Idol Recognition Client</i> .....	22
4.	<i>Class Diagram for Idol Recognition Admin</i> .....	23
5.	<i>Class Diagram for Idol Recognition Server</i> .....	24
<b>XIV.</b>	<b>Sample Images</b> .....	<b>25</b>
1.	<i>Client Application</i> .....	25
2.	<i>Admin Application</i> .....	31

## I. Ideal

### 1. Ideal of this Application

Have you ever browsed social networking, see a meme that has the face of a celebrity, assuming that person is a very pretty person and you want to find more information about them? You go through a lot of comments to find someone commenting their name but there is no information, and everyone seems already know this person. So, you have no information what that character. What would you do in this situation?

From that idea, I decided to go to the Microsoft API to create a celebrity database, whenever I saw an image on the internet, I could find who they are, their specific information.

### 2. Some other application of this ideal

Find Professor information in a photo.

Find Student information from a security camera.

Find Someone information from a security camera.

...

## II. Overall architecture and technology use

### 1. Overall architecture

#### a. Front-end

Java using Java Swing framework.

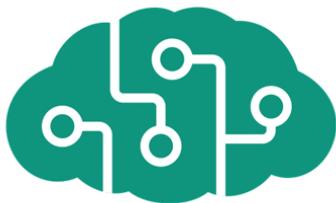


#### b. Back-end

Java using Spring Boot.



API Using: Microsoft Azure Cognitive Services.



 Microsoft  
Cognitive Services

#### c. Database

MongoDB



#### d. Encryption of information

All JSON files that are sent between clients and servers, in both directions, are encrypted using an asymmetric encryption technique (RSA 2048 and AES 256).

### III. Spring Framework



**SPRING**  
Framework

#### 1. What is Spring?

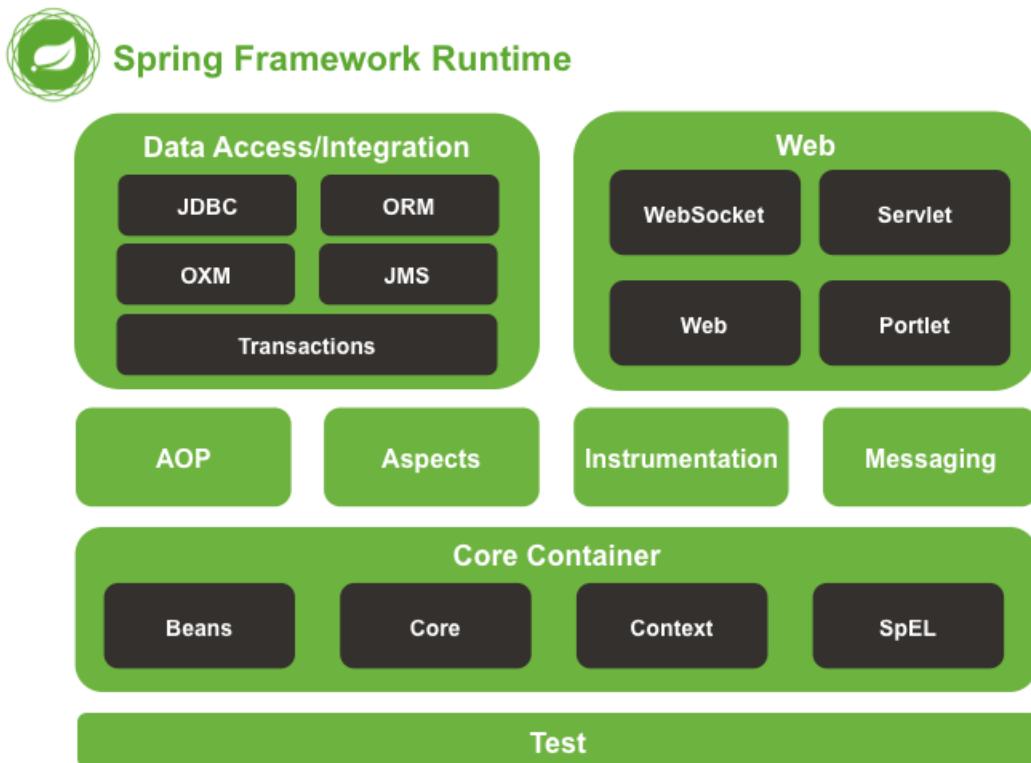
The Spring Framework is an application framework and inversion of control container for the Java platform. The framework's core features can be used by any Java application, but there are extensions for building web applications on top of the Java EE (Enterprise Edition) platform. Although the framework does not impose any specific programming model, it has become popular in the Java community as an addition to the Enterprise JavaBeans (EJB) model.

- The Spring Framework is open source.
- Spring is light and transparent (light: small in size, the basic version is only about 2MB; transparent: works transparently with programmers)
- Spring Framework is built on two main design principles: Dependency Injection and Aspect-Oriented Programming.

## 2. Architecture, modules of Spring Framework

Spring is divided into many different modules, depending on the purpose of application development that we use one of those modules.

Below is the overall architecture of the Spring Framework.



- Test: This layer provides testing support with JUnit and TestNG.
- Core Container: Includes spring core, beans, context, and expression language (EL) modules
  - Spring core, bean provide IOC and Dependency Injection.
  - Spring Context supports internationalization, Java EE features like EJB, JMX.
  - Expression Language is extended from Expression Language in JSP. It provides support for setting / getting values, improved methods for accessing collections, indexes, logical operators, etc.
- AOP, Aspects and Instrumentation: These modules support Aspect-Oriented Programming, supporting integration with AspectJ.
- Data Access / Integration: This group includes JDBC, ORM, OXM, JMS and Transaction module. These modules provide the ability to communicate with the database.
- Web: Also known as Spring MVC This group includes Web, Web-Servlet,... supports creating web applications.

## 3. Why Spring?

- Spring allows developers to use POJOs. Using POJOs helps you not have to work with EJB, applications, running streams, configurations ... much simpler.

- Spring is organized in a modular fashion. The number of packages and classes is quite large, but you only need to care about what you need and do not care about the rest.
- Spring supports the use of many technologies such as ORM Framework, logging frameworks, JEE, scheduling libraries (Quartz and JDK timer) ...
- Spring's Web module is designed according to MVC model, so it provides full of features to help replace other web frameworks like Struts.
- ...

#### 4. Some other projects of Spring

Also based on the basic design principles of spring core. Spring also develops many subprojects such as:

- Spring MVC
  - Spring MVC is designed for building web-based applications.
- Spring Security
  - Provides authentication and authorization mechanisms for your application.
- Spring Boot
  - Spring Boot is a framework that helps us develop and run applications quickly.
- Spring Batch
  - This project makes it easy to create schedules and scheduling for batch jobs.
- Spring Social
  - This project will connect your application with third-party APIs of Facebook, Twitter, Linkedin ... (eg login with facebook, google + ...)
- Spring IO
- Spring Cloud
- Spring Mobile
- Spring for Android
- Spring Session

## IV. Spring Boot



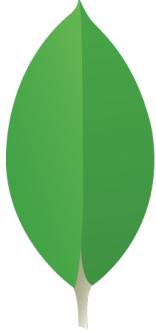
### 1. What is Spring Boot?

- Spring Boot is a Java language development project in the Spring framework ecosystem. It helps our developers simplify the process of programming an application with Spring, focusing only on developing the business for the application.
- Spring Boot is built on top of the Spring framework, and it comes with many dependencies that can be plugged into the Spring application. Some examples are Spring Kafka, Spring LDAP, Spring Web Services, and Spring Security. However, developers have to configure each building brick themselves using a lot of XML configuration files or annotations.

### 2. Why Spring Boot?

- Autoconfiguration: Developers can automatically configure their Spring application. However, Spring Boot is also capable of changing the configuration based on the dependencies you list. For example, when you list “MySQL” as a dependency, it will configure your Spring application with the “MySQL connector” included. And if you want to add a custom configuration, you can create a class that overrides the default configuration for your “MySQL connector”.
- Standalone: There’s no need to deploy your application to a web server. You simply enter the run command to start the application.
- Opinionated: On the official page, we find that Spring Boot decides for you which defaults to use for the configuration. Also, it decides which packages to install for the dependencies you require. For example, if you include the Spring Boot starter “pom” for “JPA”, it will autoconfigure an in-memory database, a hibernate entity manager, and a simple data source. This is an example of an opinionated default configuration that you can override. While some developers might feel this is too opinionated, Spring Boot’s opinionated setup helps developers to get started quickly on their projects.

## V. MongoDB



# mongoDB®

### 1. NoSQL

- NoSQL is an open source database and is abbreviated by: None-Relational SQL or there is often called Not-Only SQL.
- NoSQL is developed on Javascript Framework with JSON data type and key and value data type.
- NoSQL was born as a patch for the shortcomings and shortcomings and limitations of RDBMS (Relational Database Management System) in terms of speed, features, Ability of extension, ...
- With NoSQL you can extend data without worrying about things like creating foreign keys, primary keys, checking constraints, etc.
- NoSQL ignores the integrity of data and transactions in exchange for fast performance and scalability.
- NoSQL is used in many companies and large corporations, for example, FaceBook uses Cassandra developed by FaceBook, Google developed and used BigTable, ...

### 2. MongoDB

- MongoDB is an open source database management system, is a NoSql database and is used by millions of people.
- MongoDB is a document-oriented database, the data is stored in JSON-style document instead of tables like relational database so the query will be very fast.
- With relational databases we have the concept of tables, relational databases (such as MySQL or SQL Server ...) use tables to store data, with MongoDB we will use the concept of collection instead of tables.
- Compared to RDBMS, in MongoDB collection corresponds to table, while document will correspond to row, MongoDB will use documents instead of row in RDBMS.
- Collections in MongoDB are structured very flexibly, allowing archived data not to follow a certain structure.
- Related information is stored together for quick query access through the MongoDB query language

## VI. Java Swing Framework

### 1. What is Java Swing Framework?

- Swing is a GUI widget toolkit for Java. It is part of Oracle's Java Foundation Classes (JFC) – an API for providing a graphical user interface (GUI) for Java programs.
- Swing was developed to provide a more sophisticated set of GUI components than the earlier Abstract Window Toolkit (AWT). Swing provides a look and feels that emulates the look and feel of several platforms, and also supports a pluggable look and feel that allows applications to have a look and feel unrelated to the underlying platform. It has more powerful and flexible components than AWT. In addition to familiar components such as buttons, checkboxes and labels, Swing provides several advanced components such as tabbed panel, scroll panes, trees, tables, and lists.

### 2. Why Java Swing?

- Platform Independent
- Follows MVC
- Supported by NetBeans
- Supports pluggable look and feel
- Lightweight

### 3. Container Class

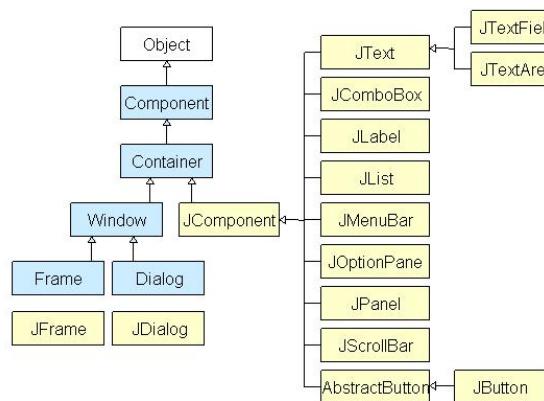
Any class which has other components in it is called as a container class. For building GUI applications at least one container class is necessary.

Following are the three types of container classes:

- Panel – It is used to organize components on to a window
- Frame – A fully functioning window with icons and titles
- Dialog – It is like a pop up window but not fully functional like the frame

### 4. Java Swing Class Hierarchy

All the components in swing like JButton, JComboBox, JList, JLabel are inherited from the JComponent class which can be added to the container classes. Containers are the windows like frame and dialog boxes. Basic swing components are the building blocks of any gui application. Methods like setLayout override the default layout in each container. Containers like JFrame and JDialog can only add a component to itself. Following are a few components with examples to understand how we can use them.



## VII. Microsoft Azure Cognitive Services



### 1. What is Microsoft Azure Cognitive Services?

- Cognitive Services bring AI within reach of every developer—without requiring machine-learning expertise. All it takes is an API call to embed the ability to see, hear, speak, search, understand, and accelerate decision-making into your apps.
- Azure Cognitive Services is a set of APIs, SDKs and container images that enables developers to integrate ready-made AI directly into their applications. Azure Cognitive Services contains a broad set of capabilities including text analytics; facial detection, speech and vision recognition; natural language understanding, and more.

### 2. Support services

Microsoft Azure Cognitive Services supports:

- Decision: Make smarter decisions faster.
- Language: Extract meaning from unstructured text.
- Speech: Integrate speech processing into apps and services.
- Vision: Identify and analyze content within images, videos, and digital ink.
- Web search: Find what you're looking for from the world-wide-web.

### 3. Why Microsoft Azure Cognitive Services?

- Apply AI to more scenarios with the most comprehensive portfolio of domain-specific AI capabilities on the market.
- Build confidently with the first AI services to achieve human parity in computer vision, speech, and language.
- Deploy Cognitive Services anywhere from the cloud to the edge with containers.
- No machine-learning expertise required.

### 4. What will be used in this project?

In this project, I use 2 services of Cognitive Services:

- Vision
  - Azure Face Service
- Web search
  - Bing Image Search API

## VIII. Bing Image Search API

### 1. What is Bing Image Search API?

The Bing Image Search API enables us to use Bing's image search capabilities in our application. By sending search queries to the API, we can get high-quality images similar to [bing.com/images](http://bing.com/images).

[Documents](#) for using Bing Image Search API in Java.

### 2. Bing Image Search features

- Suggest search terms in real-time.
- Filter and restrict image results.
- Crop, resize, and display thumbnails.
- Pivot & expand user search queries.
- Get trending images.

### 3. Workflow

The Bing Image Search API is a RESTful web service, making it easy to call from any programming language that can make HTTP requests and parse JSON.

- Create a Cognitive Services API account with access to the Bing Search APIs.
- Send a request to the API, with a valid search query.
- Process the API response by parsing the returned JSON message.

### 4. What will be used in this project?

For this project, Bing Image Search API acts as a data source for machine to learn from. While input is Idol's name, the output is the images of the face of that idol. We will need about 20-30 images for each idol.

## IX. Azure Face Service

### 1. What is Azure Face Service?

The Azure Cognitive Services Face service provides algorithms that detect, recognize, and analyze human faces in images. The ability to process human face information is important in many different software scenarios. Example scenarios include security, natural user interface, image content analysis and management, mobile apps, and robotics.

### 2. Azure Face Service features

The Face service provides several different functions which are each outlined in the following sections.

- Face detection: The Face service detects human faces in an image and returns the rectangle coordinates of their locations. Optionally, face detection can extract a series of face-related attributes. Examples are head pose, gender, age, emotion, facial hair, and glasses.
- Face verification: The Verify API does an authentication against two detected faces or from one detected face to one-person object. Practically, it evaluates whether two faces belong to the same person.
- Find similar faces: The Find Similar API compares a target face with a set of candidate faces to find a smaller set of faces that look similar to the target face. Two working modes, matchPerson and matchFace, are supported. The matchPerson mode returns similar faces after it filters for the same person by using the Verify API. The matchFace mode ignores the same-person filter. It returns a list of similar candidate faces that might or might not belong to the same person.
- Face grouping: The Group API divides a set of unknown faces into several groups based on similarity. Each group is a disjoint proper subset of the original set of faces. All of the faces in a group are likely to belong to the same person. There can be several different groups for a single person. The groups are differentiated by another factor, such as expression, for example.
- Person identification: The Identify API is used to identify a detected face against a database of people. This feature might be useful for automatic image tagging in photo management software. You create the database in advance, and you can edit it over time.

### 3. What will be used in this project?

The face recognition mechanism consists of 2 processes: Face Detection and Face Recognition. These two processes correspond to the two Microsoft Cognitive API APIs: Detect and Identity.

- We need the detect function, input is an image URL, returns the face ID.

- Output example:

```
[  
  {  
    "faceId": "c5c24a82-6845-4031-9d5d-978df9175426"  
  }  
]
```

- Once we have the face ID in the image (faceID), we use these ids as inputs for the identity function. The API compares the faces, giving the ID of the person with the closest face (in the candidates array).

- Output example:

```
[
  {
    "faceId": "c5c24a82-6845-4031-9d5d-978df9175426",
    "candidates": [
      {
        "personId": "25985303-c537-4467-b41d-bdb45cd95ca1",
        "confidence": 0.92
      }
    ]
  }
]
```

- After obtaining the ID of the person, we find the corresponding idol with the person ID from the Database.

- An Idol in database has the data like this:

```
_id: ObjectId("5d544e91378f8408b2871bca")
personId: "25ebba90-2ba7-4591-93b2-7c048ce5a26d"
> persistedFaceIds: Array
  name: "Bella Hadid"
  userData: "3"
```

## X. Another technology

### 1. RSA 2048

RSA is named for the MIT scientists (Rivest, Shamir, and Adleman) who first described it in 1977. It is an asymmetric algorithm that uses a publicly known key for encryption, but requires a different key, known only to the intended recipient, for decryption. In this system, appropriately called public key cryptography (PKC), the public key is the product of multiplying two huge prime numbers together. Only that product, 1024, 2048, or 4096 bits in length, is made public.

### 2. AES 256

AES (Advanced Encryption Standard) has become the encryption algorithm of choice for governments, financial institutions, and security-conscious enterprises around the world. The U.S. National Security Agency (NSA) uses it to protect the country's "top secret" information.

The AES algorithm successively applies a series of mathematical transformations to each 128-bit block of data. Because the computational requirements of this approach are low, AES can be used with consumer computing devices such as laptops and smartphones, as well as for quickly encrypting large amounts of data. For example, the IBM z14 mainframe series uses AES to enable pervasive encryption in which all the data in the entire system, whether at rest or in transit, is encrypted.

AES is a symmetric algorithm which uses the same 128, 192, or 256 bits key for both encryption and decryption (the security of an AES system increases exponentially with key length). With even a 128-bit key, the task of cracking AES by checking each of the  $2^{128}$  possible key values (a "brute force" attack) is so computationally intensive that even the fastest supercomputer would require, on average, more than 100 trillion years to do it. In fact, AES has never been cracked, and based on current technological trends, is expected to remain secure for years to come.

### 3. MongoDB Java Drivers

The official MongoDB Java Drivers providing both synchronous and asynchronous interaction with MongoDB.

### 4. HttpClient and BSON for Java

Used to make HTTP requests and parse JSON.

## XI. Idol Recognition System

### 1. What is Idol Recognition System?

In an ideal state (when the idol database is big enough), this system will be the place where:

- Users can access and use the feature to find names of people in photos, view history of identification and delete history.
- Meanwhile, administrators can manage the user list and the list of idols in the database (view, delete, edit).

### 2. Idol Recognition System Structure

This project use:

- Java Swing as Front-end application for client and admin.
- Spring with Spring Boot as Back-end application for server.
- MongoDB as Database.

## XII. Idol Recognition System Structure

### 1. Use Case



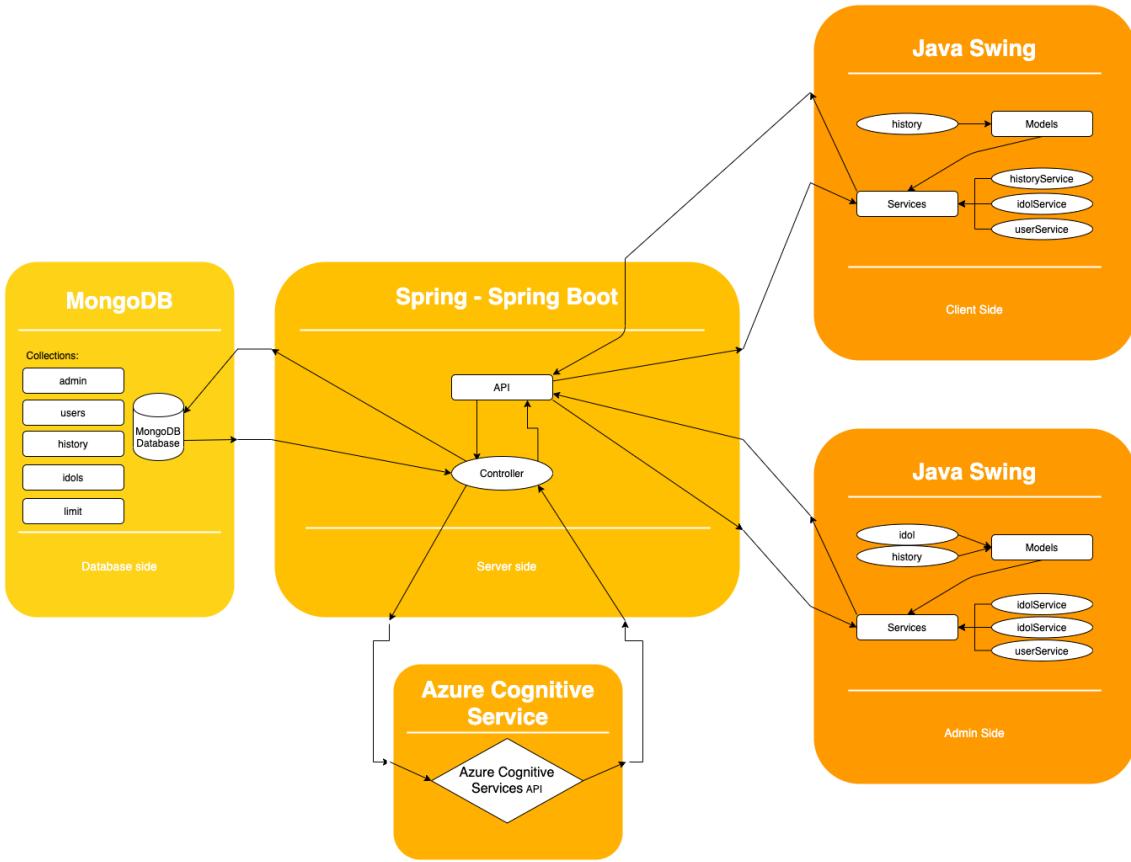
### 2. Models

idol	user	history
id String name String userData String personID String	id String name String surname String email String password String token String accountType String	id String token String url String idolRecognized String success Boolean visible Boolean date Date

### 3. Database Diagram

admin	idol	history
_id ObjectId name String surname String email String password String token String accountType String	_id ObjectId name String userData String personID String persistedFaceIds Array	_id ObjectId token String url String idolRecognized String success Boolean visible Boolean date Date
user	limit	
_id ObjectId name String surname String email String password String token String accountType String	_id ObjectId token String limit Int32	

## 4. Workflow



## 5. Rest API list

### a. For Client App

- @PostMapping("/checkUser")
  - public boolean userAuthentication(@RequestBody String dataRaw)
- @PostMapping("/createUser")
  - public boolean createUser(@RequestBody String dataRaw)
- @PostMapping("/changePassword")
  - public boolean changeUserPassword(@RequestBody String dataRaw)
- @PostMapping("/getUserInfo")
  - public String findUserInfo(@RequestBody String dataRaw)
- @PostMapping("/findIdol")
  - public String getIdol(@RequestBody String dataRaw)
- @PostMapping("/getAllHistory")
  - public String getAllHistory(@RequestBody String dataRaw)
- @PostMapping("/deleteHistory")
  - public void deleteAllHistories(@RequestBody String dataRaw)

### b. For Admin App

- @PostMapping("/checkAdmin")
  - public boolean adminAuthentication(@RequestBody String dataRaw)
- @PostMapping("/createAdmin")

- public boolean createAdmin(@RequestBody String dataRaw)
- @GetMapping("/getAllUsers")
  - public String getAllUsers()
- @PutMapping("/editUser")
  - public String editUser(@RequestBody String dataRaw)
- @PutMapping("/deleteUser")
  - public boolean deleteUser(@RequestBody String dataRaw)
- @GetMapping("/getAllIdols")
  - public String getAllIdols()
- @PostMapping("/indexNewIdol")
  - public boolean indexAddNewIdol(@RequestBody String dataRaw)
- @PutMapping("/editIdol")
  - public String editIdol(@RequestBody String dataRaw)
- @PutMapping("/deleteIdol")
  - public boolean deleteIdol(@RequestBody String dataRaw)

## XIII. Class Diagram

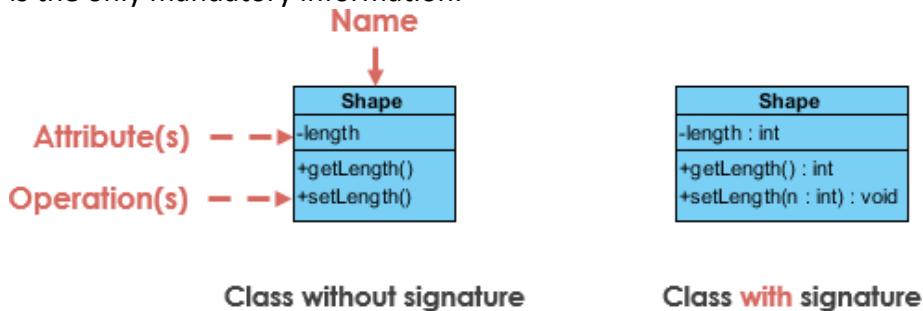
### 1. What is Class Diagram?

The UML Class diagram is a graphical notation used to construct and visualize object-oriented systems. A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's:

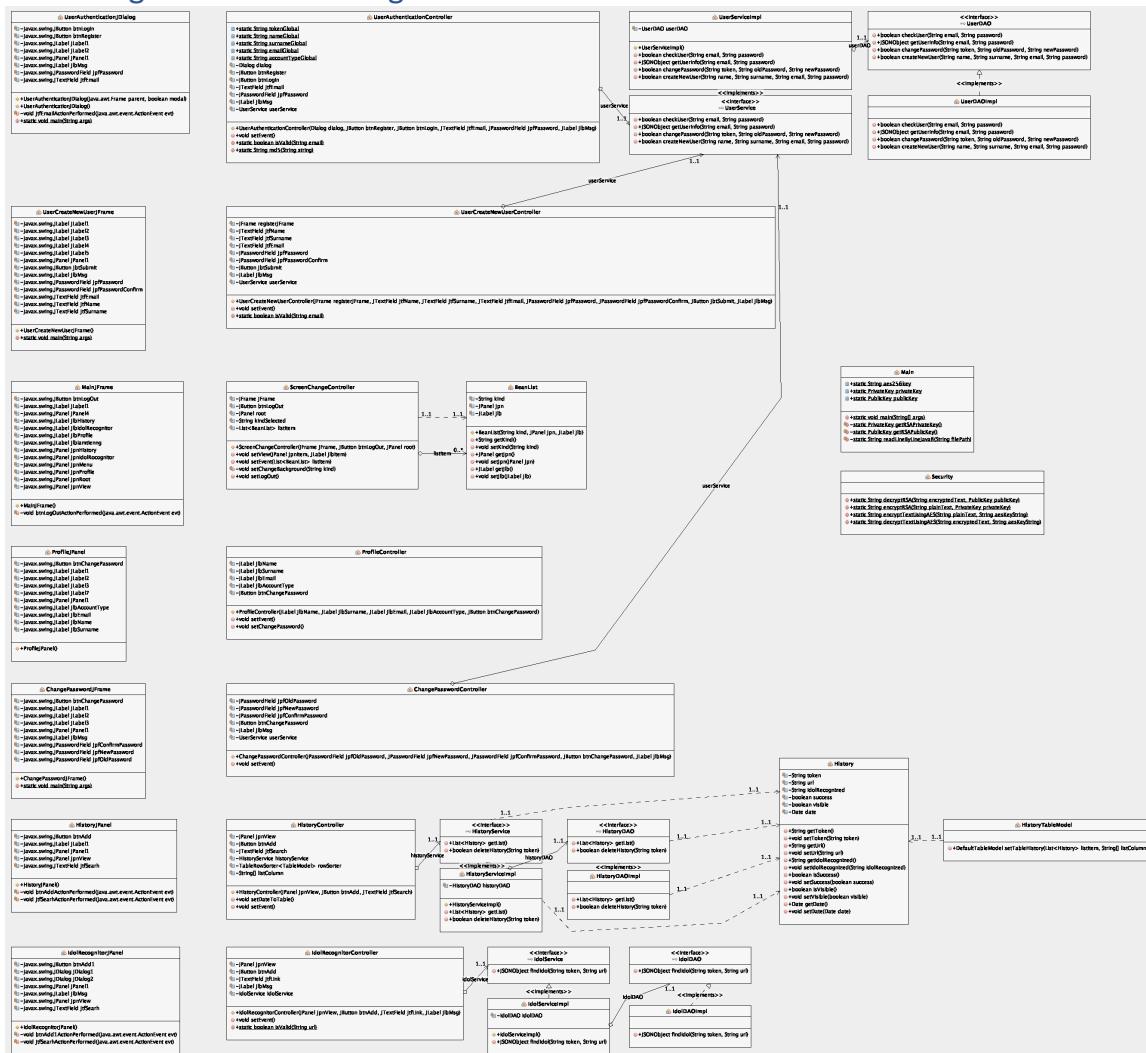
- classes
- their attributes
- operations (or methods)
- and the relationships among objects

### 2. UML Class Notation

A class represent a concept which encapsulates state (attributes) and behavior (operations). Each attribute has a type. Each operation has a signature. The class name is the only mandatory information.

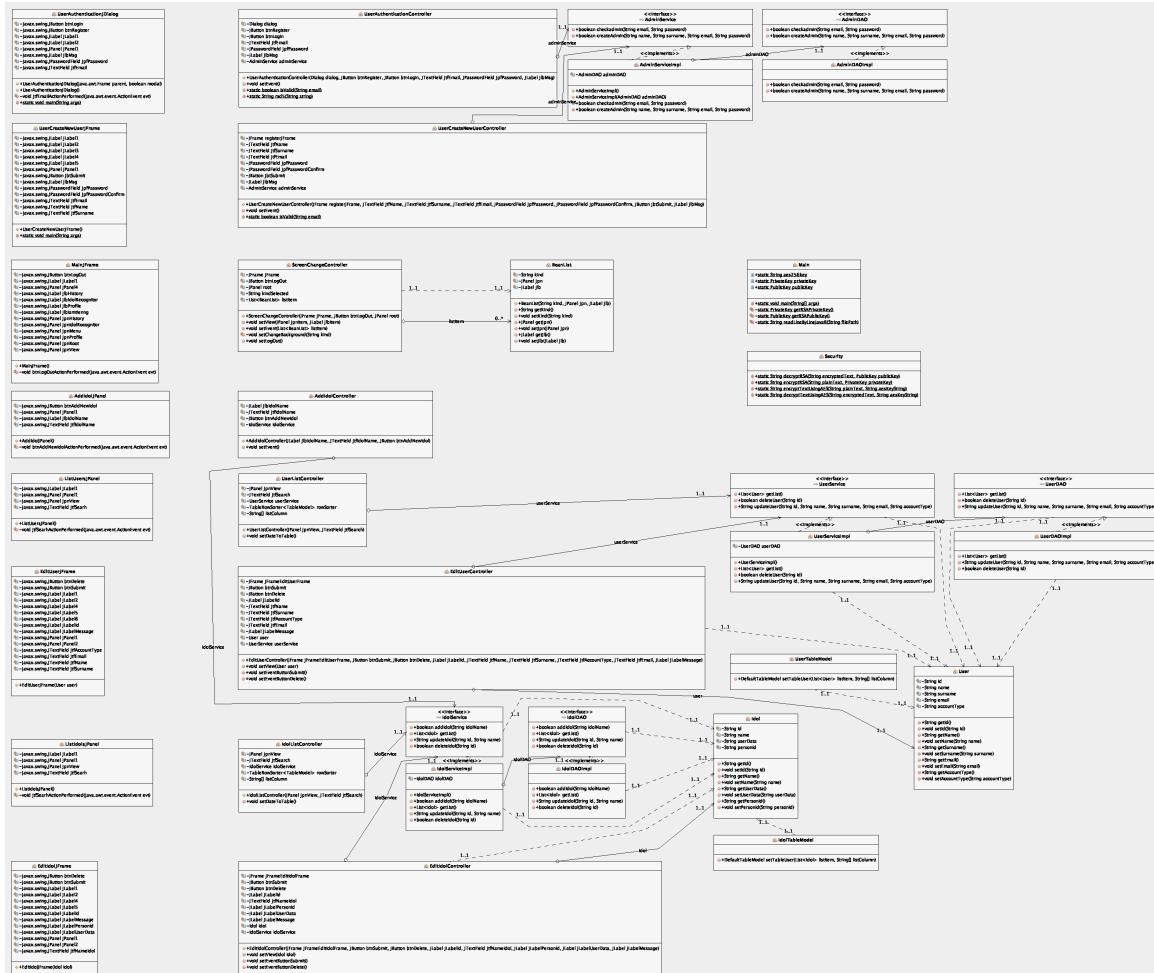


### 3. Class Diagram for Idol Recognition Client



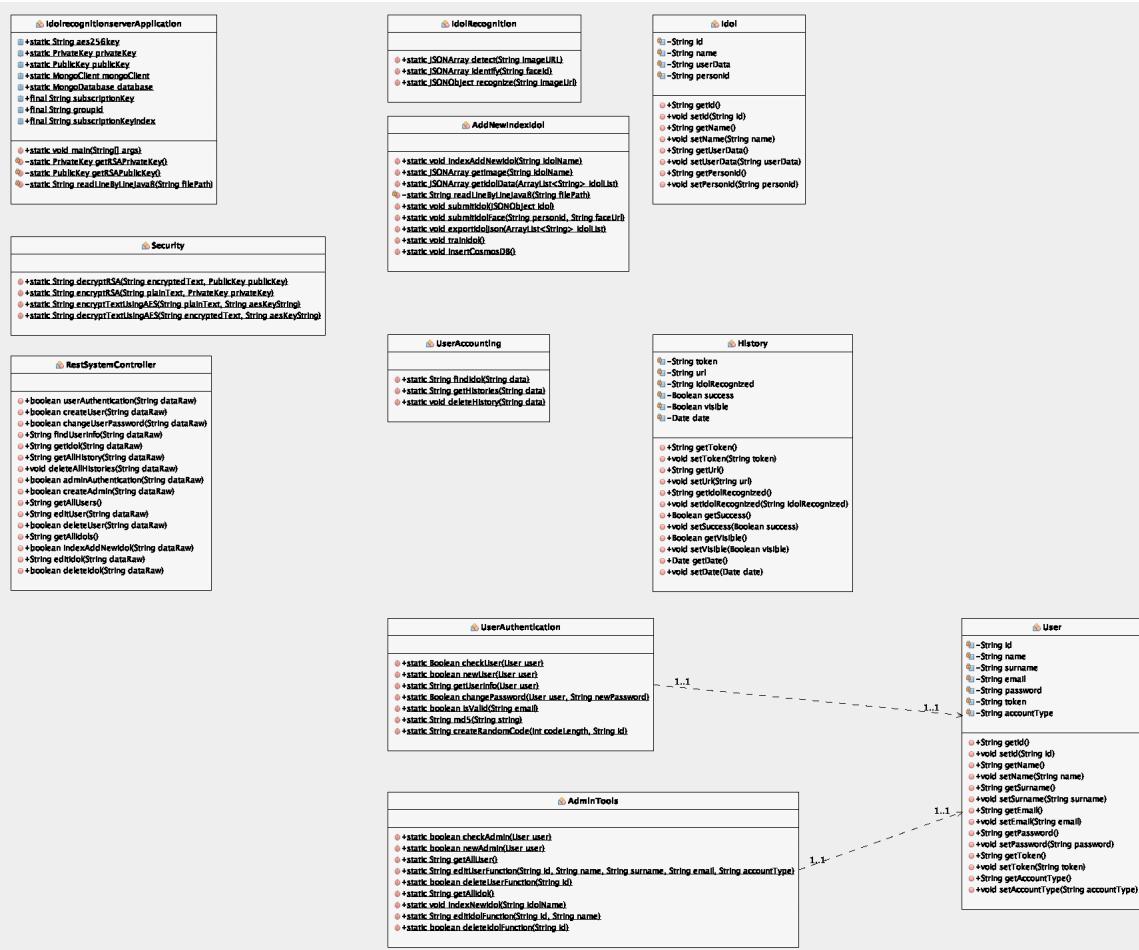
For detail, the diagram appears in UMLDiagramsClient project.

#### 4. Class Diagram for Idol Recognition Admin



For detail, the diagram appears in UMLDiagramsAdmin project.

## 5. Class Diagram for Idol Recognition Server



For detail, the diagram appears in UMLDiagramsServer project.

## XIV. Sample Images

### 1. Client Application

Login

Dialog



Register

Dialog

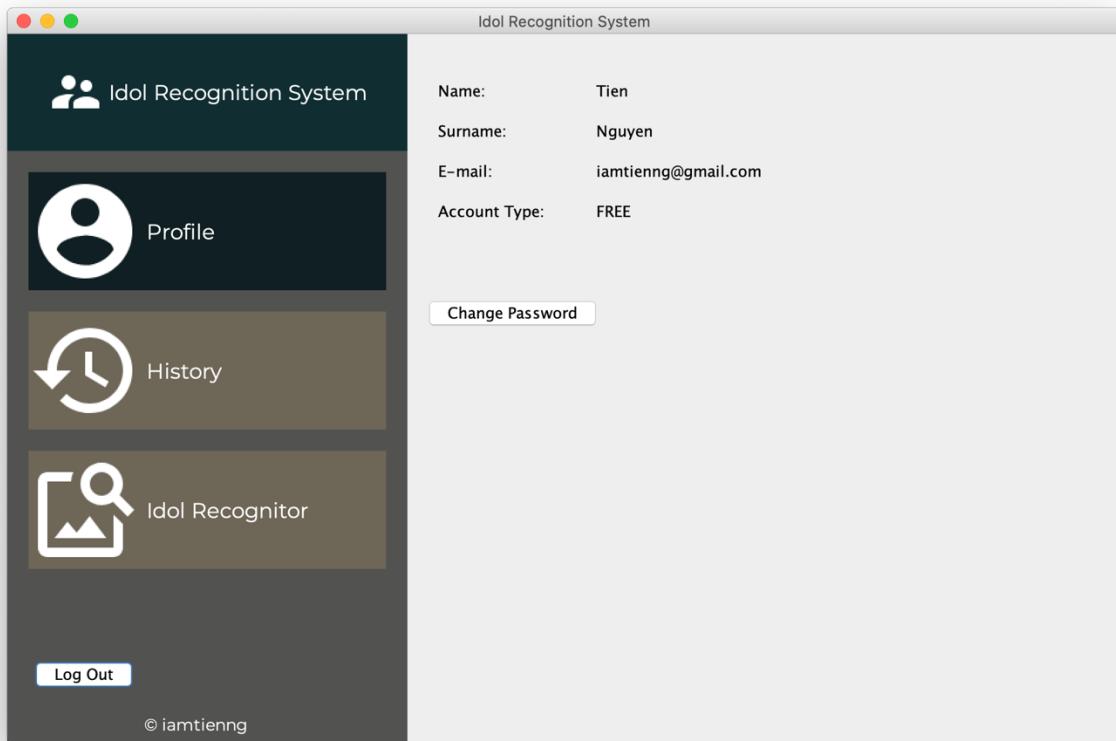
A screenshot of a 'Register New User' dialog box. The window has a title bar with three standard OS X window controls (red, yellow, green) and a title 'Register New User'. Inside, there are five input fields: 'Name' (Tien), 'Surname' (Nguyen), 'E-mail' (iamtienng@gmail.com), 'Password' (represented by seven black dots), and 'Confirm Password' (also represented by seven black dots). A 'Submit' button is located at the bottom right of the form area.

Name	Tien
Surname	Nguyen
E-mail	iamtienng@gmail.com
Password	•••••••
Confirm Password	•••••••

Submit

## Profile

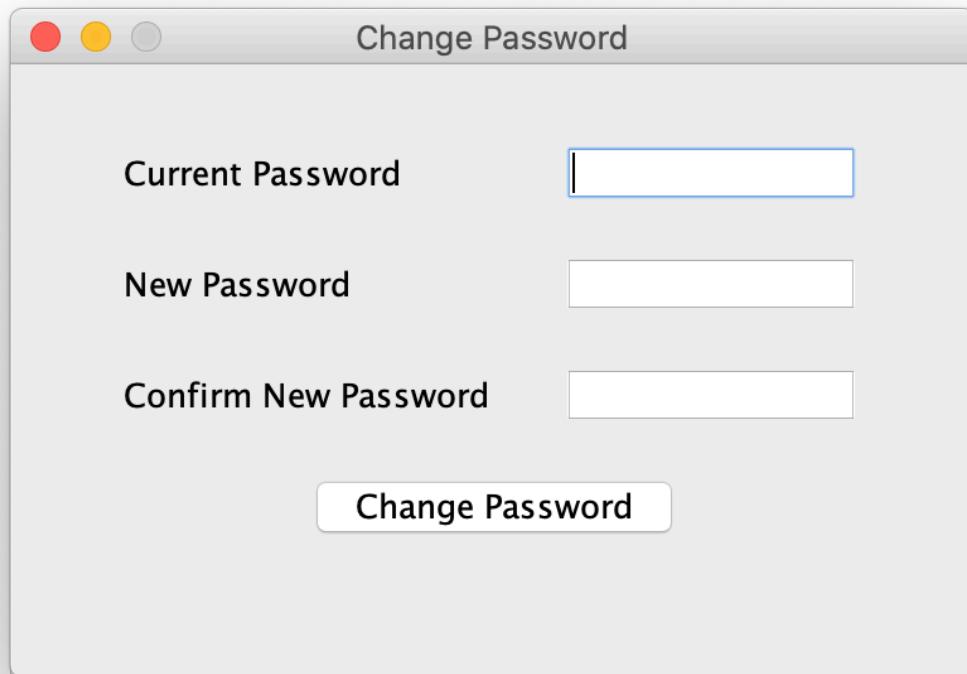
## Page



Change

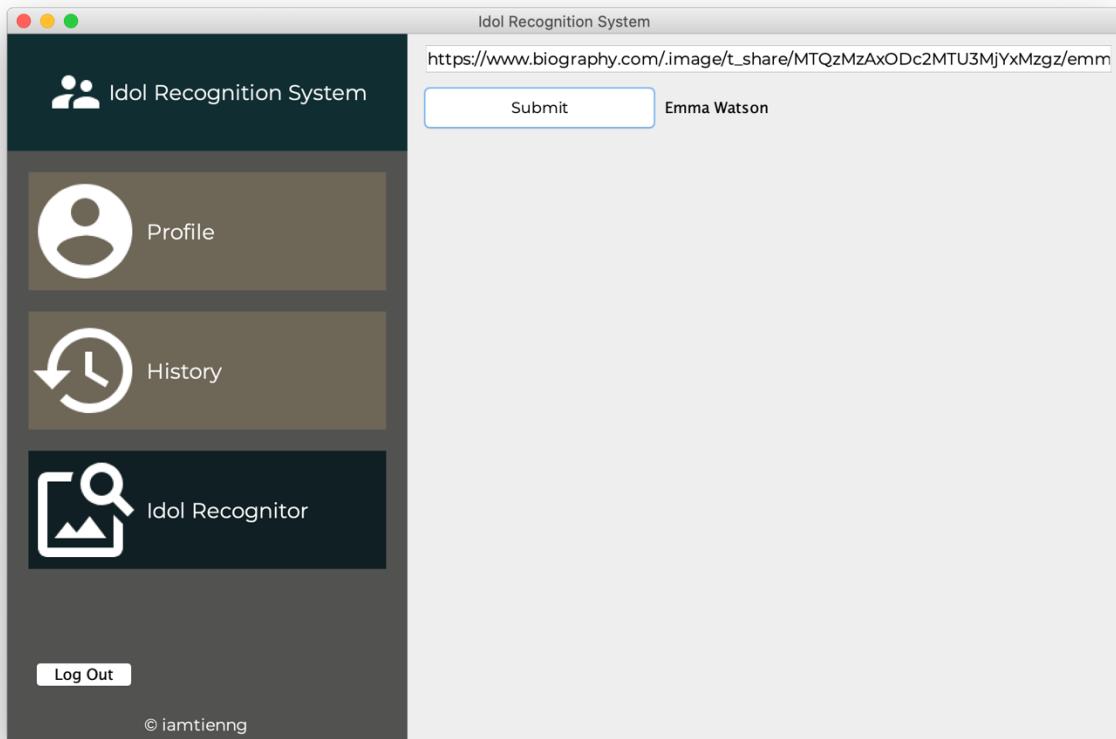
Password

Dialog



Idol

Recognizer



The screenshot shows a Mac OS X style application window titled "Idol Recognition System". On the left is a sidebar with three items: "Profile" (with a user icon), "History" (with a circular arrow icon), and "Idol Recognitor" (with a magnifying glass icon). At the bottom of the sidebar are "Log Out" and the copyright notice "© iamtiennng". The main content area is titled "Idol Recognition System" and contains a search bar with placeholder "Search:" and a "Delete History" button. Below is a table with columns "Serial", "Link", "Date", and "Idol Name". Two entries are listed:

Serial	Link	Date	Idol Name
1	<a href="https://www.biography.com...">https://www.biography.com...</a>	Thu Jul 02 02:09:56 CEST ...	Emma Watson
2	<a href="https://i.ytimg.com/vi/wfu9...">https://i.ytimg.com/vi/wfu9...</a>	Thu Jul 02 02:11:10 CEST ...	Emma Watson

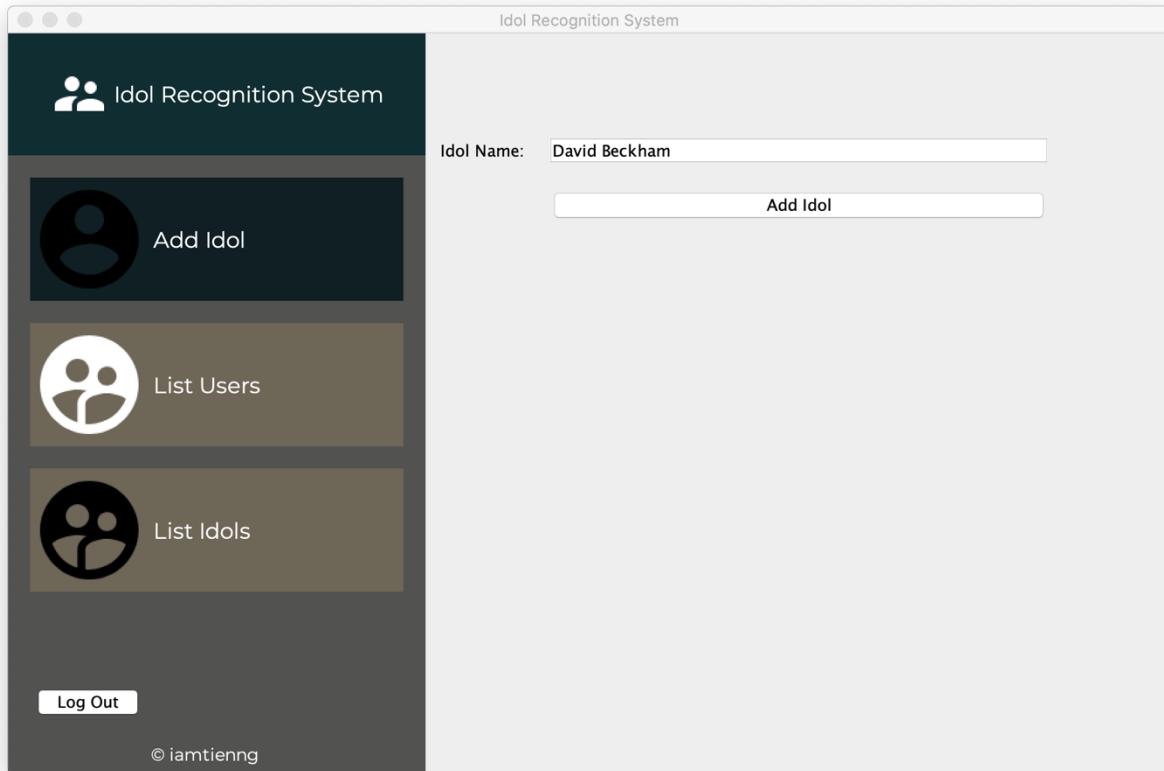
## 2. Admin Application

Add

New

Idol

Page



Users

List

Page

The screenshot shows a Mac OS X style application window titled "Idol Recognition System". The window has a dark header bar with the title and standard window controls. The main area is divided into two sections: a sidebar on the left and a table view on the right.

**Sidebar (Left):**

- Add Idol:** Icon of a person, labeled "Add Idol".
- List Users:** Icon of two people, labeled "List Users".
- List Idols:** Icon of two people, labeled "List Idols".
- Log Out:** A button labeled "Log Out".

**Table View (Right):**

A search bar at the top of the table says "Search: [ ]". The table has a header row with columns: **Serial**, **Name**, **Surname**, **E-mail**, and **Account Type**.

Serial	Name	Surname	E-mail	Account Type
5efd2...	Tien	Nguyen	iamtiennng@gmail.com	free
5efd2...	Ronald	Quinn	ronaldaquinn@exam...	free

At the bottom left of the sidebar, it says "© iamtiennng".

The screenshot shows a window titled "User Information" with a title bar featuring standard Mac OS X red, yellow, and green buttons. Below the title bar is a toolbar with two buttons: "Delete User" and "Save Data". The main content area is titled "Idol Information" in bold black font. It contains five input fields with the following data:

ID:	5efd25865dd623ccba33e9cf
Name:	Tien
Surname:	Nguyen
Email:	iamtienng@gmail.com
Account Type:	free

Idols

List

Page

The screenshot shows a desktop application window titled "Idol Recognition System". On the left side, there is a sidebar with three main items: "Add Idol" (with a person icon), "List Users" (with a people icon), and "List Idols" (with a person icon). At the bottom of the sidebar is a "Log Out" button. The right side of the window contains a table titled "User DATA" with columns for "ID", "Name", "User DATA", and "PersonID". The table lists nine entries, each consisting of a partial ID, a name, a numeric value under "User DATA", and a long alphanumeric string under "PersonID". A search bar labeled "Search:" is located at the top of the main content area.

ID	Name	User DATA	PersonID
5d54...	Kylie Jenner	5	3c74d18c-d001-4233-8e...
5d54...	Nicole Scherzinger	6	2fcb76e3-7c12-4ce0-869...
5d54...	Margot Robbie	10	521bb3c6-5d47-400a-9b...
5d54...	Gigi Hadid	2	3dd5530b-9463-4522-b...
5d54...	Kim Kardashian	8	98df3d49-81e1-4905-b2...
5d54...	Emma Watson	1	e7d31531-27d2-48ad-9c...
5d54...	Kendall Jenner	4	0843b240-e9c7-4599-ba...
5d54...	Bella Hadid	3	25ebba90-2ba7-4591-93...
5d54...	Jourdan Dunn	9	24467d90-78a7-48c4-8e...

Idol Information

Delete Idol      Save Data

**Idol Information**

ID:	5d544e91378f8408b2871bcd
Name:	Kylie Jenner
Person ID:	3c74d18c-d001-4233-8e2c-c89e3da006af
User Data:	5