Project IAM description
EPITA 2016

Student: **Gervaise ALINA**

Supervisor: **Thomas BROUSSARD**

Contents

1. Subject analysis

1. Major features

The administrator of the database schema can create, modify, delete, and find Identities. There is no graphical user interface for this application but the user can use the command line.

2. Application Feasibility

The application can be used to authenticate and manage user for other applications.

3. Data description

The user will enter:

- Display name
- Birthday
- Password
- Email
- Actions to perform

4. Expected results

The application will display the created Identities. The user can view all identities when modifying or deleting.

5. Algorithms study

6. Scope of the application (limits, evolutions)

- The present project does not really allow Identity to connect to the database. The next step will be to grant privileges to each Identity and check the right to display the menu. Right now, only the root the schema can connect to the database.
- Other step will be to include a Graphical interface to the application.
- Managing sql jointures are not very easy especially between attributes and the other classes, another step will be to use a tool such as Hibernate or Spring
- The search feature also is not well advance and search methods need to be implemented
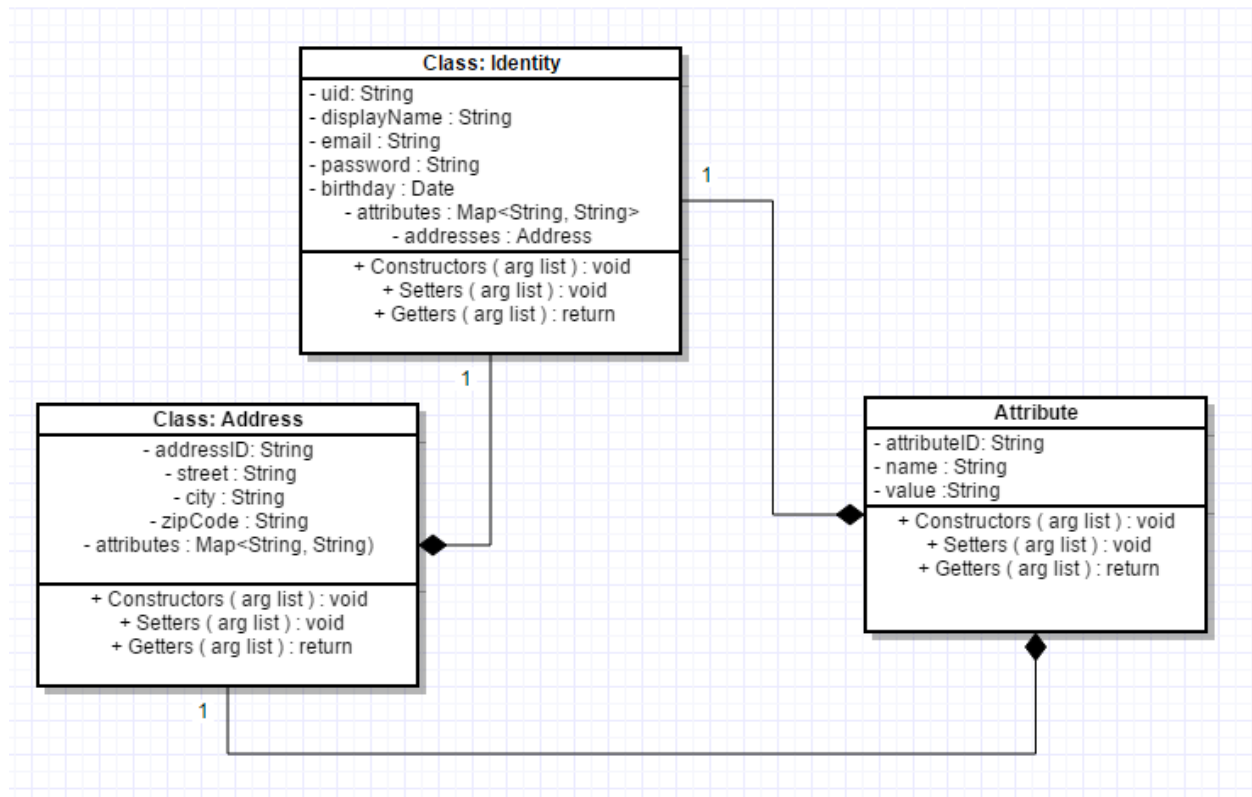- The Identity password is not yet secured

2. Conception

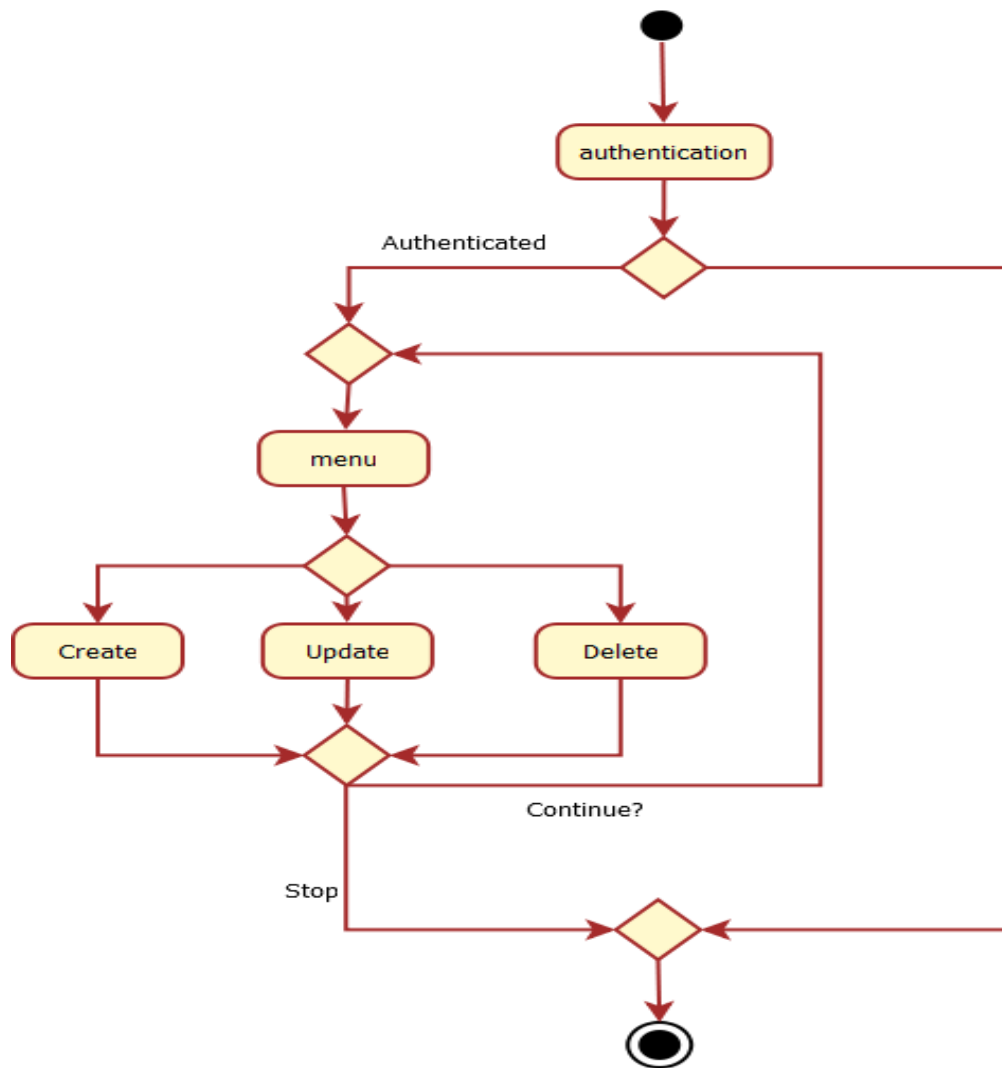The application contains the following packages:

- Business layer
- Config package
- Datamodel layer
- Exceptions packages
- Gui layer: Not implemented at this time
- Launcher: Where the main file is ConsoleLauncher
- Javadoc
- Test
- Sql

1. Chosen algorithm

2. Data structures

```
                    Class: Identity
          - uid: String
          - displayName : String
          - email : String
          - password : String
          - birthday : Date
                - attributes : Map<String, String>
                    - addresses : Address

              + Constructors ( arg list ) : void
                + Setters ( arg list ) : void
                + Getters ( arg list ) : return
```

```
        Class: Address
      - addressID: String
        - street : String
        - city : String
      - zipCode : String
    - attributes : Map<String, String)

      + Constructors ( arg list ) : void
        + Setters ( arg list ) : void
        + Getters ( arg list ) : return
```

```
              Attribute
    - attributeID: String
    - name : String
    - value :String

      + Constructors ( arg list ) : void
        + Setters ( arg list ) : void
        + Getters ( arg list ) : return
```
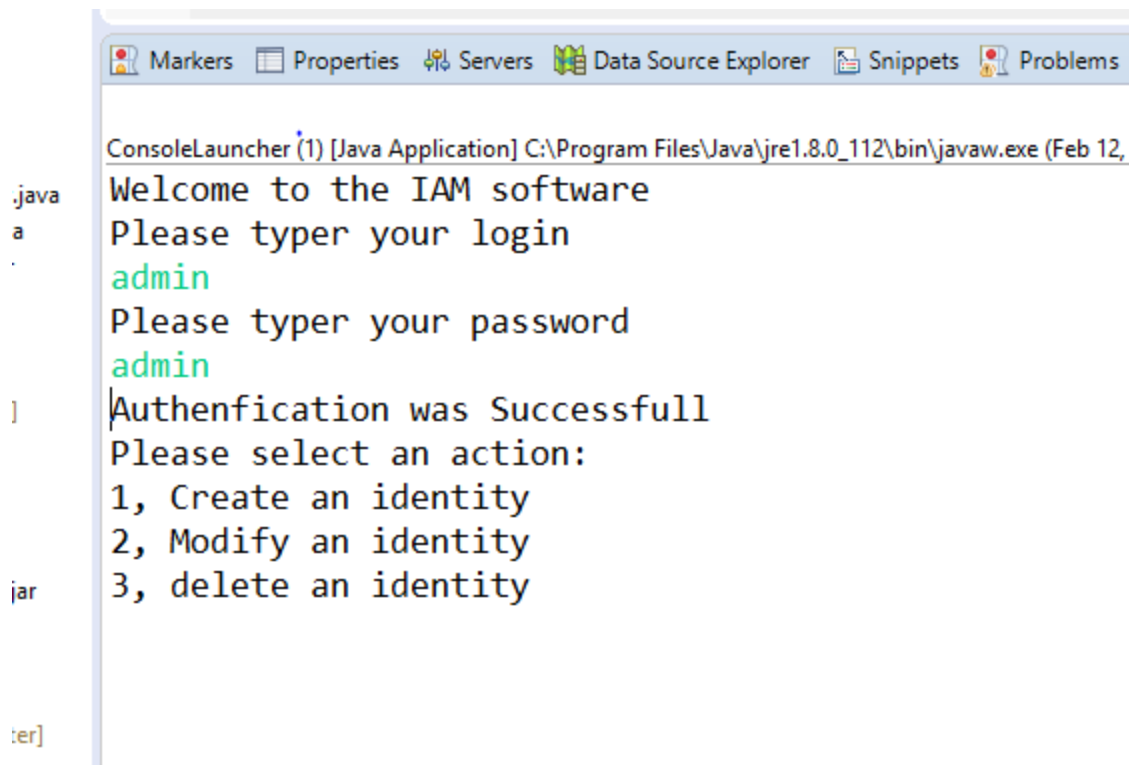
3. Global application flow

4. Global schema and major features schema

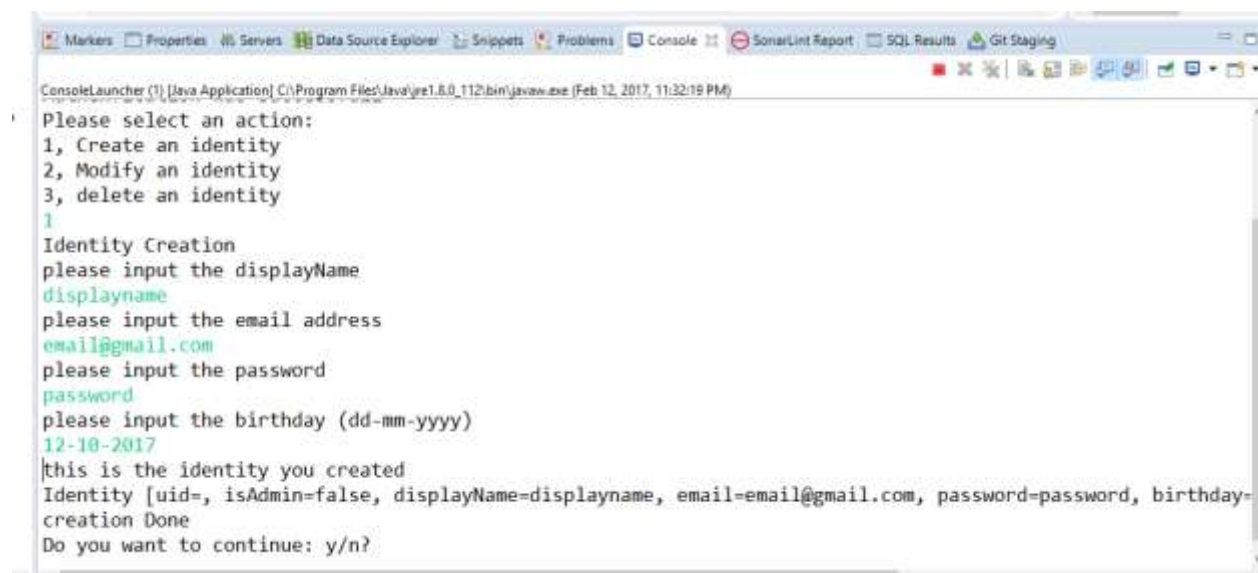### 3. Console operations description

1. Authenticate

- The user launches the application
- The application requests the login
- The user enters a login name
- The application requests the password
- The user enters the password
- The application return a successful message in case the user is an admin

2. Create operation

- The application prompt the user to select an action.
- The user types "1"
- The application prompts the user to enter all basic attributes of the Identities
- The user enters values
- The application creates an Identity
- The application prompt the user to continue or the leave the application

3. Update operation

- The application prompt the user to select an action .
- The user types "2"
- The application presents a list of identity created by their ID
- The user enters select one ID
- The application displays a menu of Identity attributes to modify
- The user select an attribute
- The application prompts the user to enter a value for that attribute
- The user enters a value for the selected attributes
- Once the user select the quit option
- The application displayed the edited Identity
- The application prompt the user to continue or the leave the application

4. Delete operation

- The application prompt the user to select an action .
- The user types "3"
- The application presents a list of identity created by their ID
- The user enters select one ID
- The application confirm that the user wants to delete the Identity with ID
- The user select "y"
- The application delete the identity
- The application prompt the user to continue or the leave the application

4. Configuration instructions

1. Setup the database:

- Download apache derby client

- Create an instance of derby name it instance name =IAM with username="admin"

 and password="admin".

- run /db-derby-10.13.1.1-bin/bin/startNetworkServer

- Go to iam-core/src/fr/epita/iam/config/config.properties

 edit user value with the user name you used to create your derby schema

Do the same with password, the name if instance if you used your own values

- Go to iam-core/sql run create_tables.sql on your instance IAM

2. Run the application:

- Go to iam-core/src/fr/epita/iam/launcher execute ConsoleLauncher.java

3. Developers:

- Go to iam_core/javadoc for documentation

- Go to iam-core/test to run advanced test. The console application does not allows the user to add other attributes and addresses on identity but by running

these tests you can see how to use the advanced options.

5. Commented Screenshots

No screen shot is available .We have not yet implement the Graphical User Interface

6. Bibliography

- www.stackoverflow.com