

IAM System Technical Specification

By Zexin Lin

Content

1. Subject description	2
2. Subject analysis	2
2.1. Major features	2
2.2. Application feasibility	3
2.3. Data description	3
2.4. Expected result	4
2.5. Scope of the application	4
3. Conception	4
3.1. Data structure	4
3.2. Global application flow	5
3.3. Global schema and major features schema	6
4. Commented screenshots	8
4.1 Login	8
4.2 Homepage	8
4.3 Display	9
4.4 Search	9
4.5 Add field	9
4.6 Create	10
4.7 Modify	10

1. Subject description

The IAM Project is mainly aimed at managing users of an Information System. The functions of the application should cover the following:

- 1) Create, delete, modify and search for (by multiple criteria) identity information.
- 2) Persist user data in a database
- 3) Provide a user-friendly interface, which in my case is a website.
- 4) Be robust, capable of good performance

2. Subject analysis

2.1. Major features

- 1) Manage authentication for the users

Query from the database to get the corresponding password for the current login, then check with the password submitted by the user. The security mechanism should be the main concern.

- 2) Manage the public common attributes for all identity

The system should be able to load the attribute list from the database first. Users can view, add and delete the field from the table, which provide the convenience for users to manage the information dynamically.

- 3) Manage the private attribute for certain identity

User can customize the extra information for any user.

- 4) Create an identity

Insert a new identity to the table holding identity information in the database

- 5) Delete an identity

Delete an existing identity from database with specified id

- 6) Update an identity

Update identity information

7) Search for identities by multiple criteria

Users can specify multiple criteria according to the current attribute list to search for related identities.

2.2. Application feasibility

Strength:

- 1) Any cooperation nowadays can't survive without a reliable and efficient Information Management System.
- 2) All the features can easily be achieved.

Weakness:

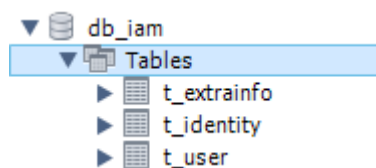
- 1) How to make the system robust and capable of high-performance remains a challenge.
- 2) How to guarantee the security of the system required professional skill on the related field.

2.3. Data description

The system mainly persists two kinds of data, **User** who manage and **Identity** who are managed.

Identity
-uid : String -displayName :String -email : String -extraInfo: Map<String, String>
+...getter(), setter()

MySQL database is used to implement data access and persist.



Database db_iam consist of three tables:

t_user: Manage the user data including username, password and some security-related fields.

	uid	userName	password	secret
▶	1	admin	admin	NULL
*	NULL	NULL	NULL	NULL

t_identity: Manage public identity information.

	uid	displayName	email	FirstName	LastName
▶	1	jack	31213@163.com	zx	l
	2	Amy	test		

t_extrainfo: Manage private identity information owned by certain user

	uid	attrName	attrValue
--	-----	----------	-----------

2.4. Expected result

- 1) Define the entity model
- 2) Implement reliable DAO interface to persist data in the database
- 3) Implement main features mentions above using MVC design pattern.
- 4) Provide user-friendly interface
- 5) Make the system robust and capable of high-performance

2.5. Scope of the application

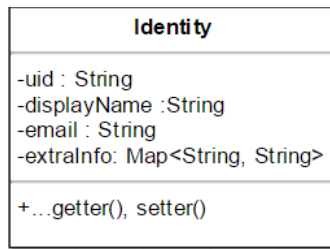
This application can fulfill the basic functions for the identity management, covering user authentication, access and modification of identity information dynamically.

Limits: Can't hold big data volume, no sorting and searching algorithm used.

3. Conception

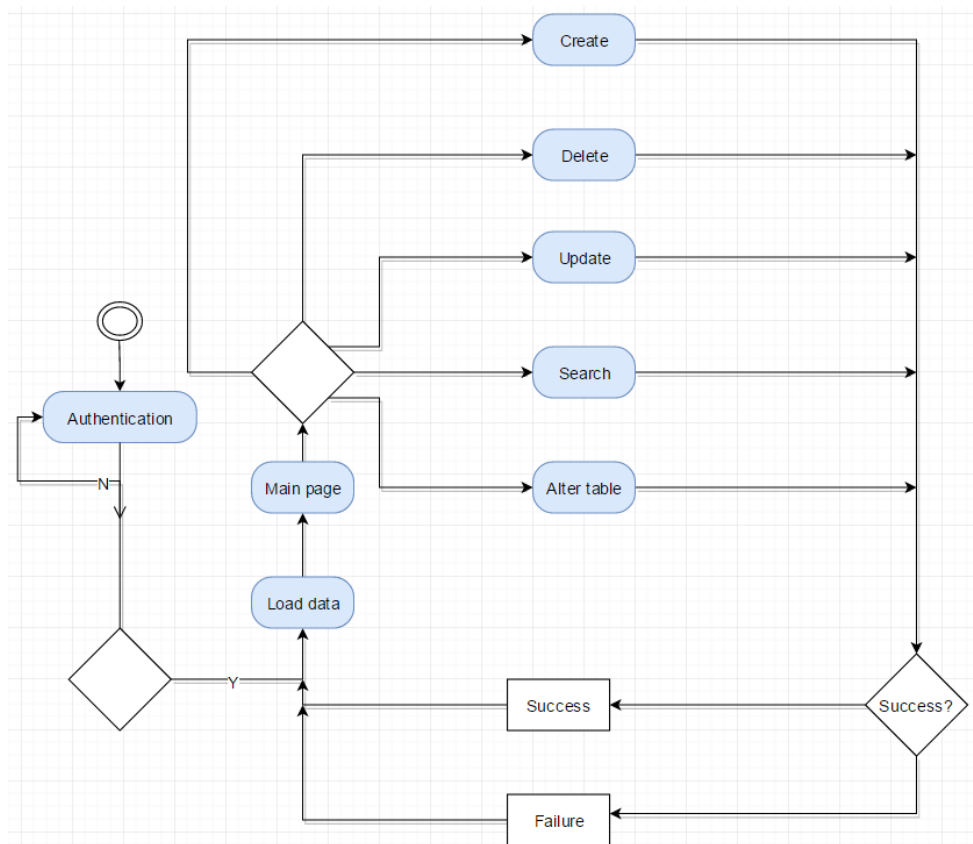
3.1. Data structure

Only the Identity class is defined.

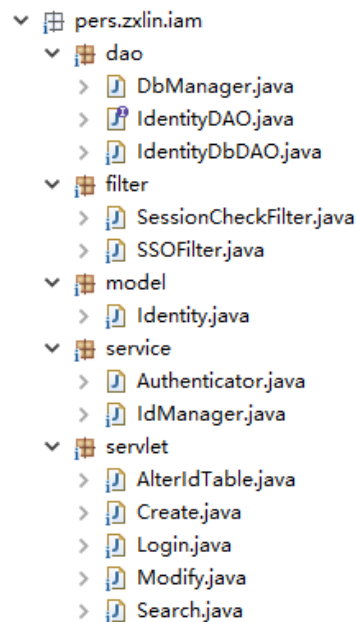


Within the Identity class, the Map data structure is used to contain all attribute to value pairs which cooperates with the dynamic modification of identity attributes.

3.2. Global application flow

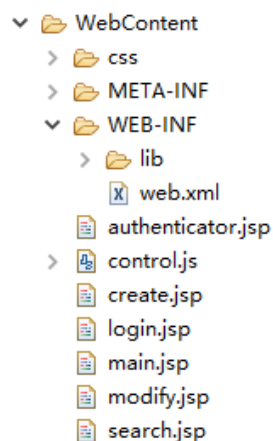


3.3. Global schema and major features schema



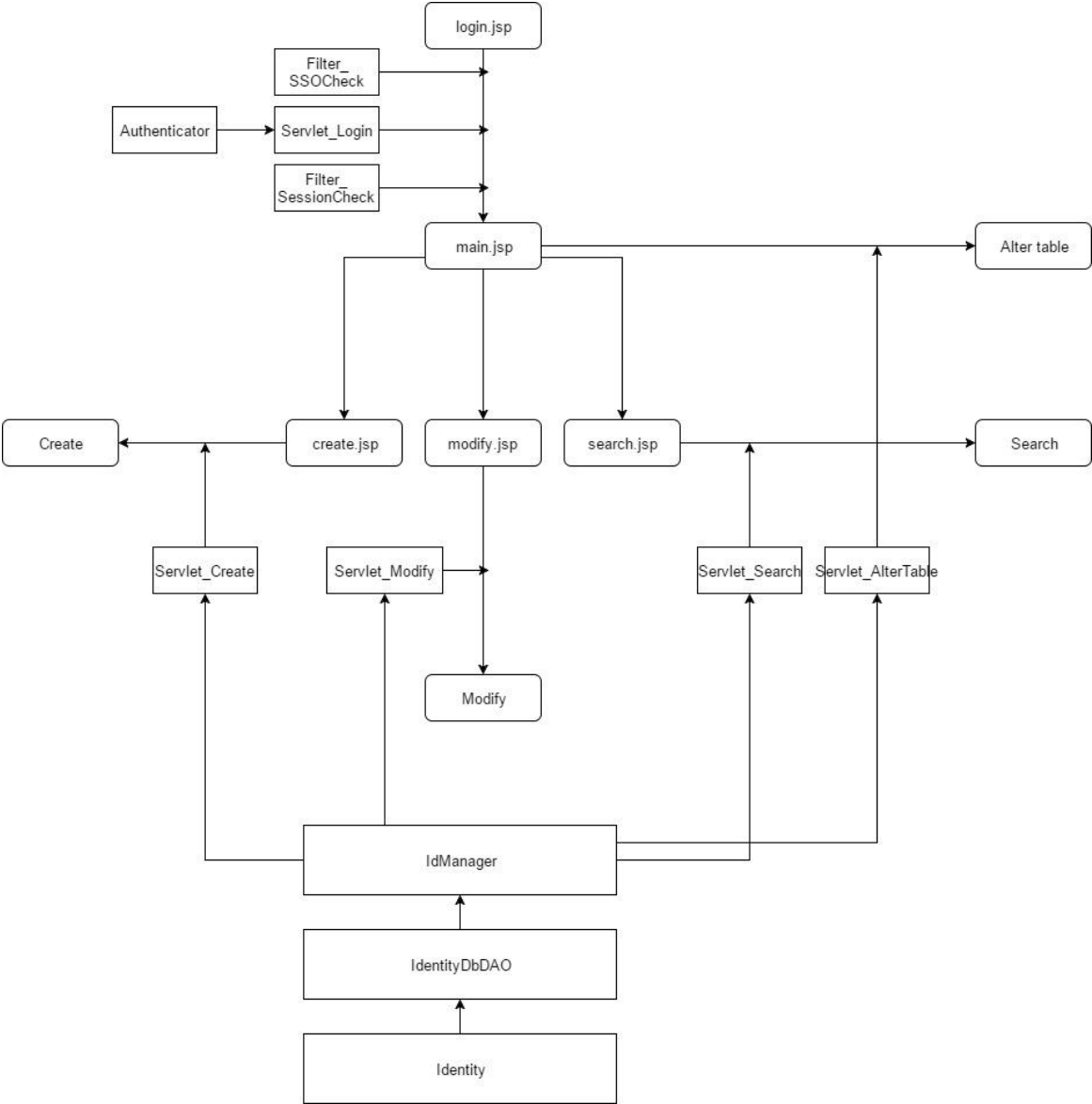
Java Package structure for backend:

- 1) Model Package defines entity class Identity to be maintained.
- 2) DAO Package regulates the IdentityDAO interface for accessing and motifying Identity class, and also contains the implementation based on MySQL database.
- 3) Service Package implements main features using DAO interface, and provides interface for UI layer to invoke.
- 4) Servlet Package include several servlets in charge of different functions. Filter Package requests from user.



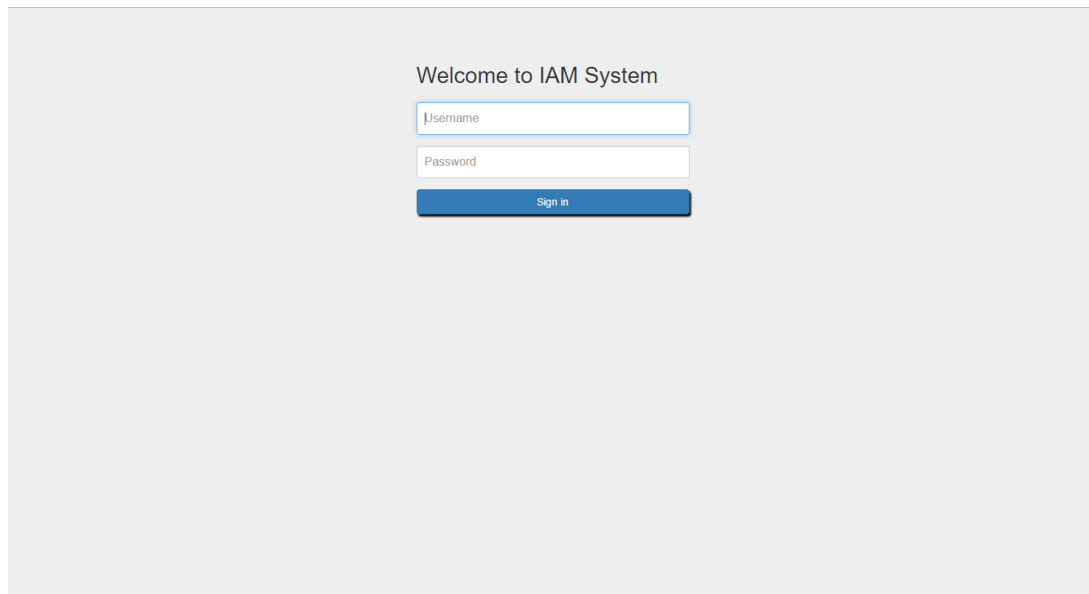
Web content: —

Following schema shows the structure of the application



4. Commented screenshots

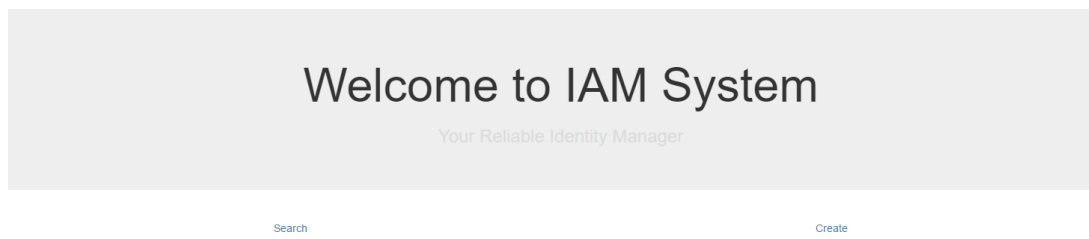
4.1 Login



A screenshot of a login page for the IAM System. The page has a light gray background. In the center, there is a white rectangular area containing the text "Welcome to IAM System" at the top. Below this text are two input fields: the first is labeled "Username" and the second is labeled "Password". Below the password field is a blue button with the text "Sign in" in white.

4.2 Homepage

Contains two tab panels in charge of Search and Create module. ,



4.3 Display

Display identity information

Search

Create

Search For Identity

Add New Attribute For The Whole

#	uid	displayName	email	FirstName	LastName
	1	jack	31213@163.com	zx	l
	2	Amy	test		
	3	test3	test@fddg.com		
	12	testt	test		
	13	testttt	test		
	14	testwfd	test		
	15	test8	test		
	16	rrr	test		
	39	qqqq	13	21	1
	40	tweqwrq	eqveqr	qve	qve
	41	qveewqeqw	eqvqve	qveqve	eqvqvevqe
	42	eqveewqveew	eqveqr	sda	dsa
	43	testtt			
	44	ttt			
	46	smalljack	sdaad@gmail.com	a	b

Modify

4.4 Search

Search for identities using the criteria filled in the form

Search

Create

Search For Identity

Add New Attribute For The Whole

Set your search criteria

uid:

uid

displayName:

displayName

email:

email

FirstName:

FirstName

LastName:

LastName

Search

Add New Attribute For The Whole

4.5 Add field

Add a new field for all identities

Search

Create

Search For Identity

Add New Attribute For The Whole

Enter the new field name

Add

4.6 Create

Create a new identity using the information filled in the form

Search

Create

Create New Identity

Fill in the information

displayName

displayName

email

email

FirstName

FirstName

LastName

LastName

Create

4.7 Modify

User can add a new private attribute, update identity information, delete current identity in this page.

Modify Identity Information

uid

46

displayName

smalljack

email

sdaad@gmail.com

FirstName

a

LastName

b

If you want to add new attribute:

Enter New Attribute Name

Enter New Attribute Value

Add

Update

Delete

Cancel