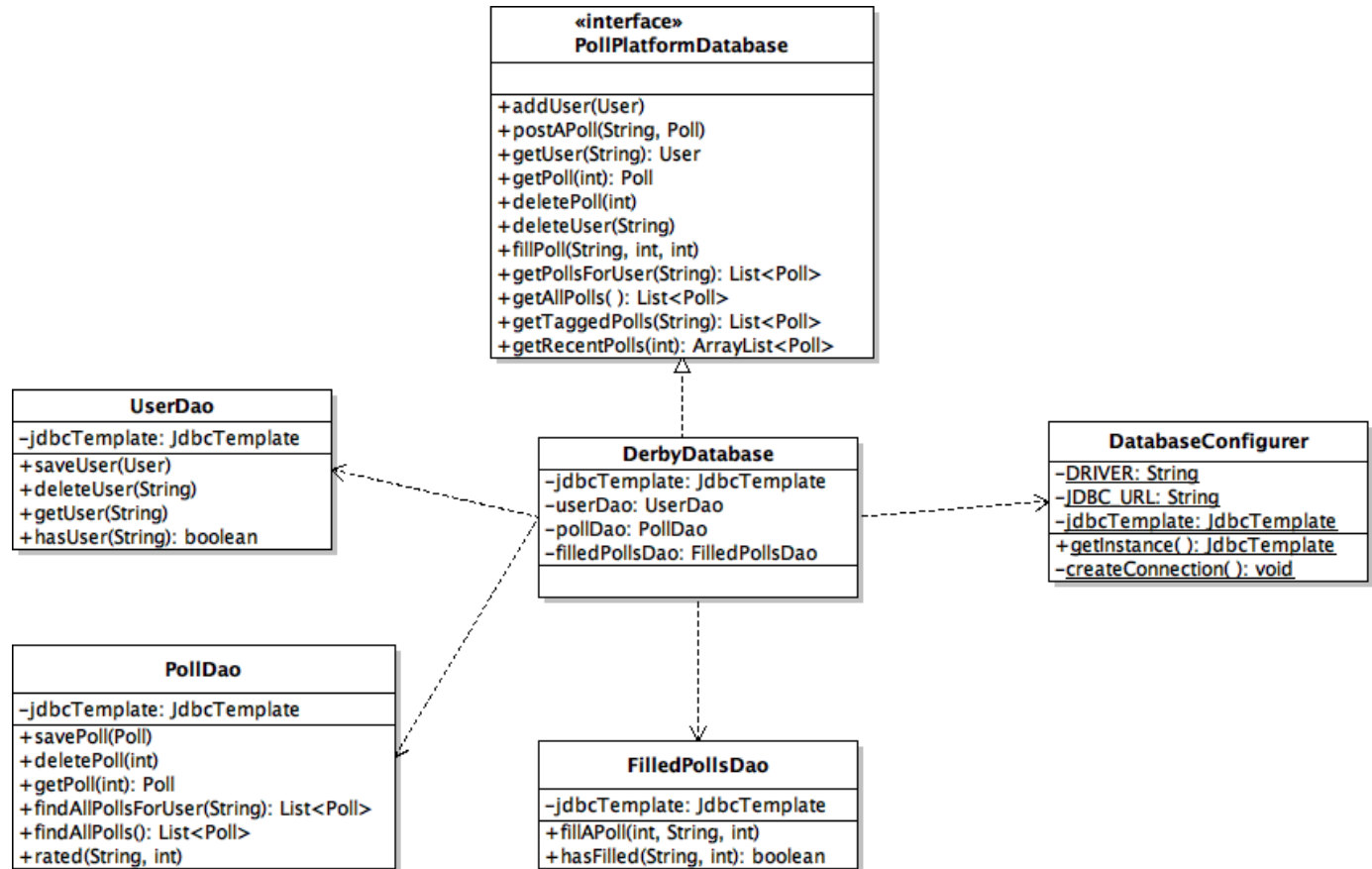


CIT 594 Project: Poll Platform

Description about design and data structures

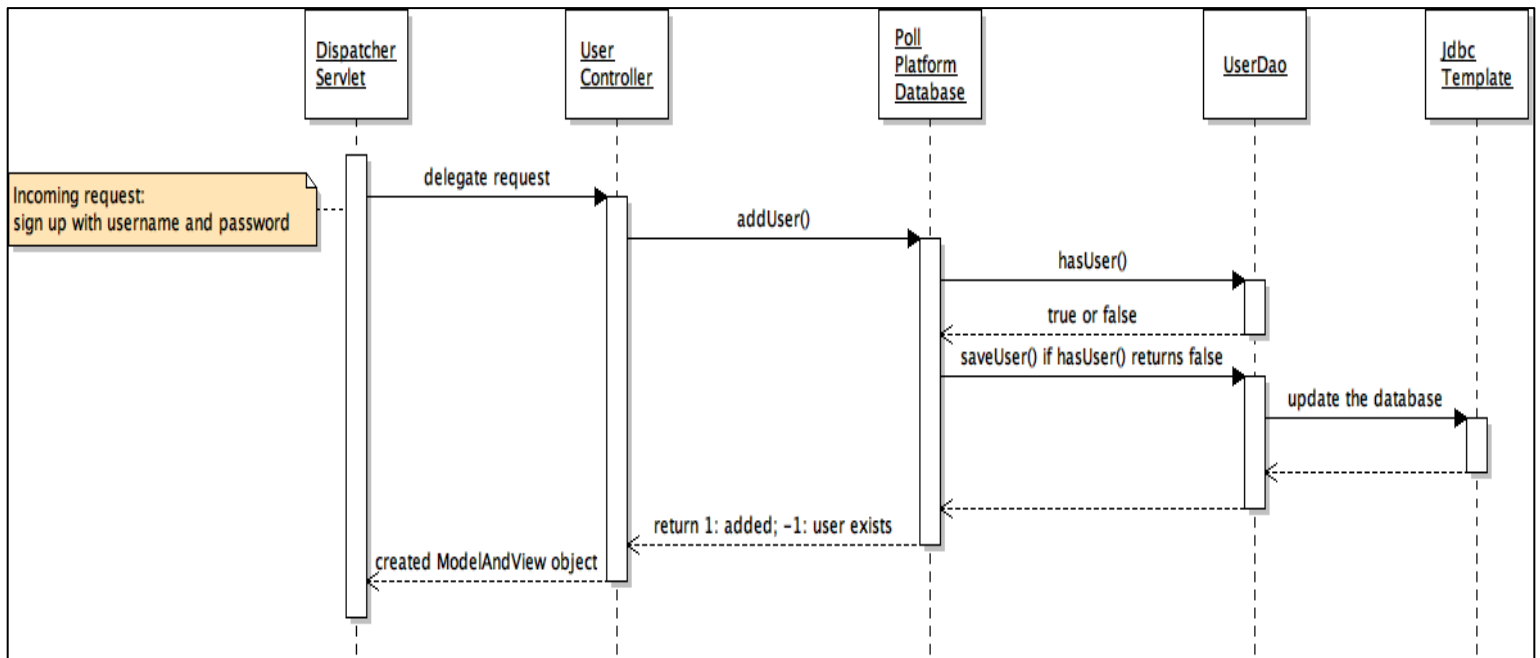
I. Formal Design

Class diagram for the package “com.database”:

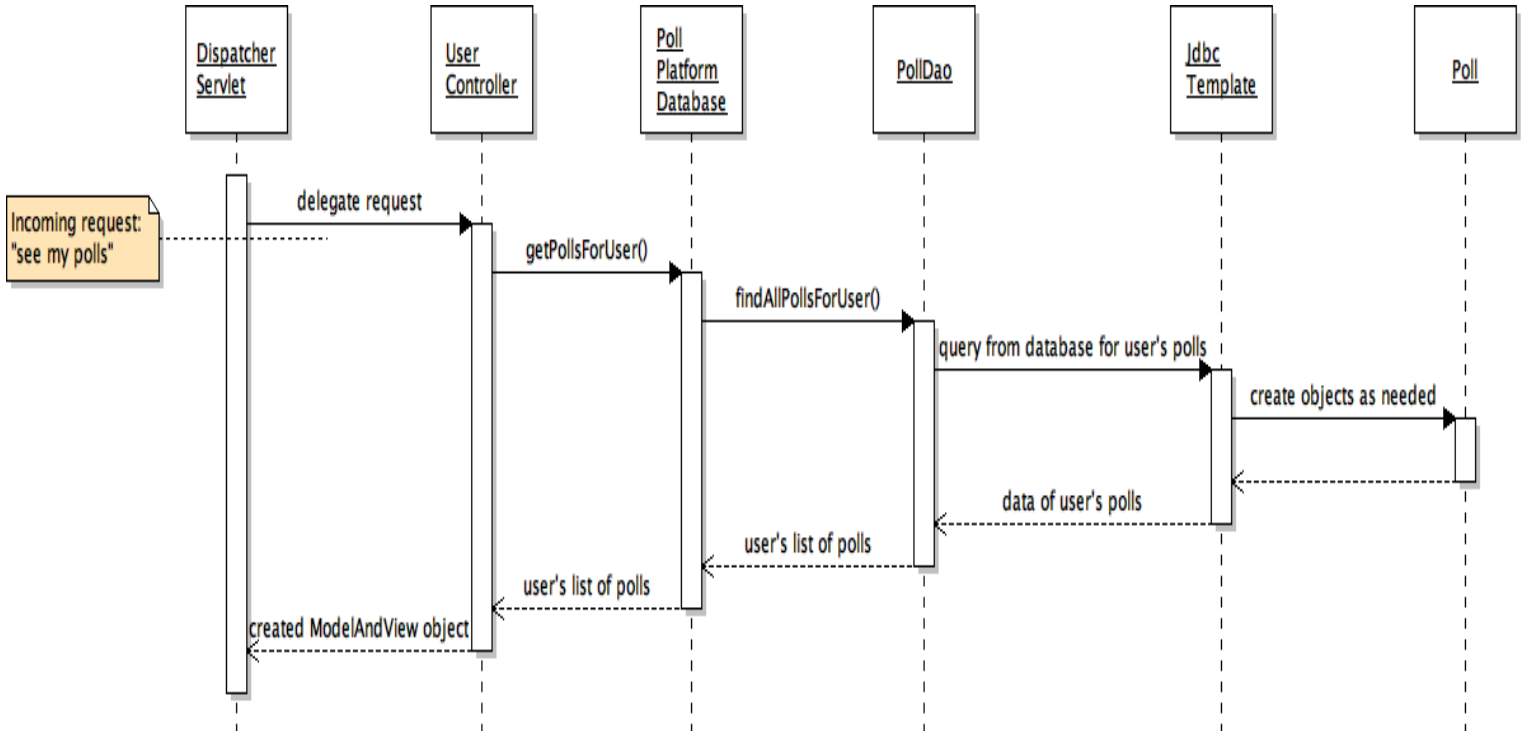


The package “com.database” contains all the java files that directly deal with the database, including configuring the database, accessing and updating the records, and carrying out certain algorithms on the data.

Sequence diagram when a user makes a “signing up” request:



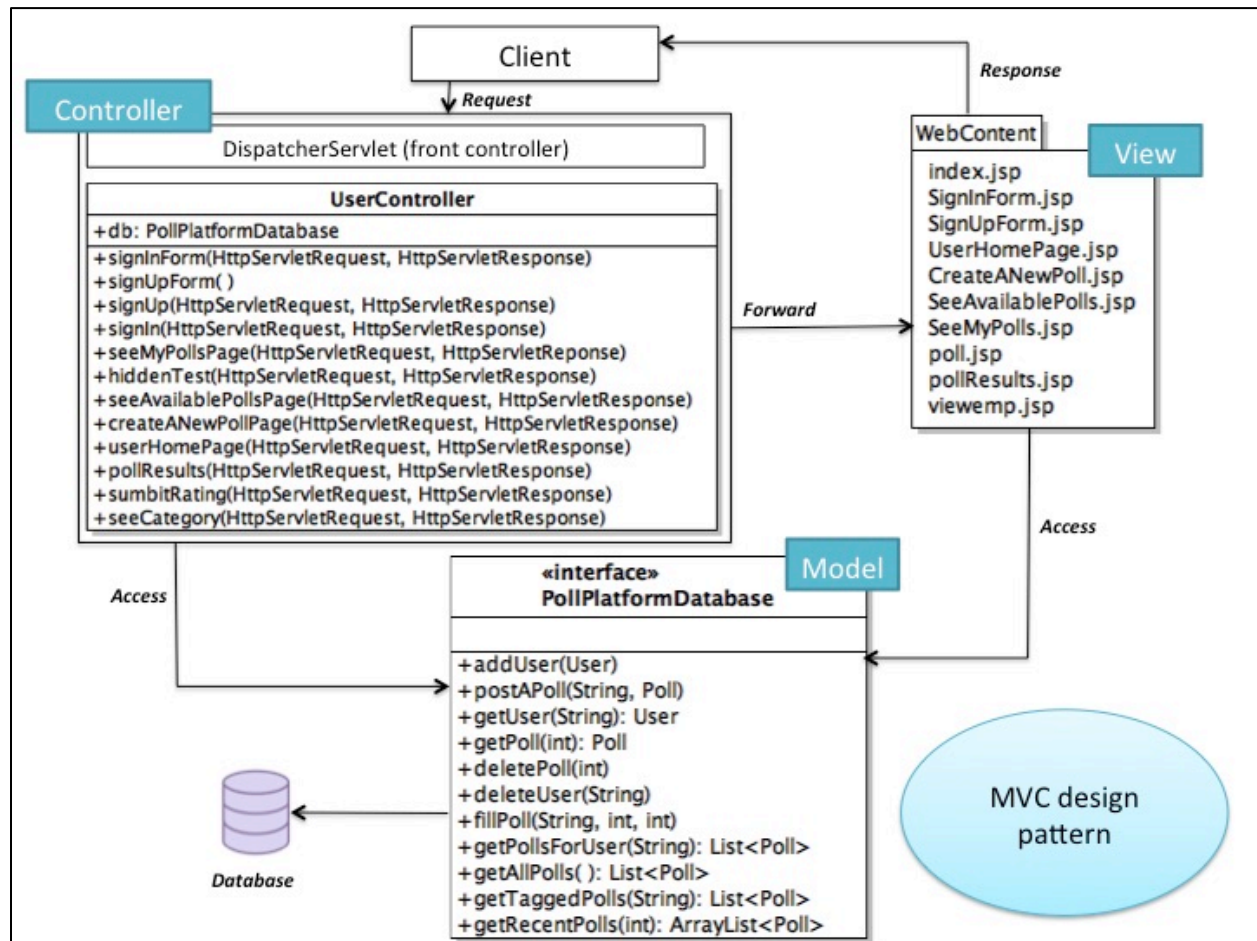
Sequence diagram when a user makes the request “see my polls”:



II. Design patterns

1. MVC design pattern:

The whole application adopts the MVC design pattern.



2. Façade design pattern:

The design of the package “com.database” follows a façade design pattern. As shown in the class diagram above, the interface “PollPlatformDatabase” wraps up the classes in the package and provides a unified interface to a set of common methods a user can call. The user can simply invoke the methods defined in the “PollPlatformDatabase” interface without worrying about the implementation details of these methods.

3. Singleton design pattern:

The DerbyDatabase class and DataConfigurer class in the “com.database” package follow the singleton design pattern. DerbyDatabase class ensures that the program has exactly one instance of DerbyDatabase, while DataConfigurer class ensures that the program has exactly one instance of JdbcTemplate.

III. Interface

The interface “PollPlatformDatabase” is defined for the user to interact with the database of the application. Users can call the methods declared in the interface like getAllPolls(), getRecentPolls(), postAPoll(), and makeRecommendations(). The DerbyDatabase class implements the interface.

IV. Data structures

1. List

The data structure list is used in PollDao class, DerbyDatabase class, and UserController class to get and pass lists of polls.

2. Array

An integer array is used to keep track of the counts for each poll’s ratings. Because each poll can be rated from 1 to 5, each of the five elements in the array records the count for the corresponding rating.

A string array “tagArray” is used to store the 8 tags used:

```
String[] tagArray = {"business", "science", "health", "sports", "arts", "entertainment", "life", "others"};
```

3. Priority queue

The data structure priority queue is used to sort the polls in the DerbyDatabase class. For example, to get the most recent polls, we create a max heap with the data structure, put all the polls in the max heap, and get the polls at the front of the max heap.

4. Hashmap

In the getRecommendations() method of the DerbyDatabase class, hashmap is used to map a tag with the counts of polls with this tag.

V. Algorithm

The application analyzes a user’s previous behavior and makes recommendations to him/her accordingly. The algorithm works as follows:

1. If the user does not post any poll, the application generates 5 most popular polls and recommends them to the user. The popularity of a poll is determined by how many people have rated it.
2. If the user has posted some polls, the polls’ tags are examined and the most used tags are computed. The program then gets the list of polls with the most used tag. From this list of polls, 5 most popular ones are recommended to the user.

VI. Advanced Java

1. Java web development by Java Spring
2. relational database – apache derby database