

High Level Architecture Document V1.0



Contents

[Application Scope](#)

[High Level Architecture Diagram](#)

[Application Activities](#)

[Class Diagrams](#)

[User Login Class Diagram](#)

[Sequence Diagram for user login](#)

[Frameworks to be used](#)

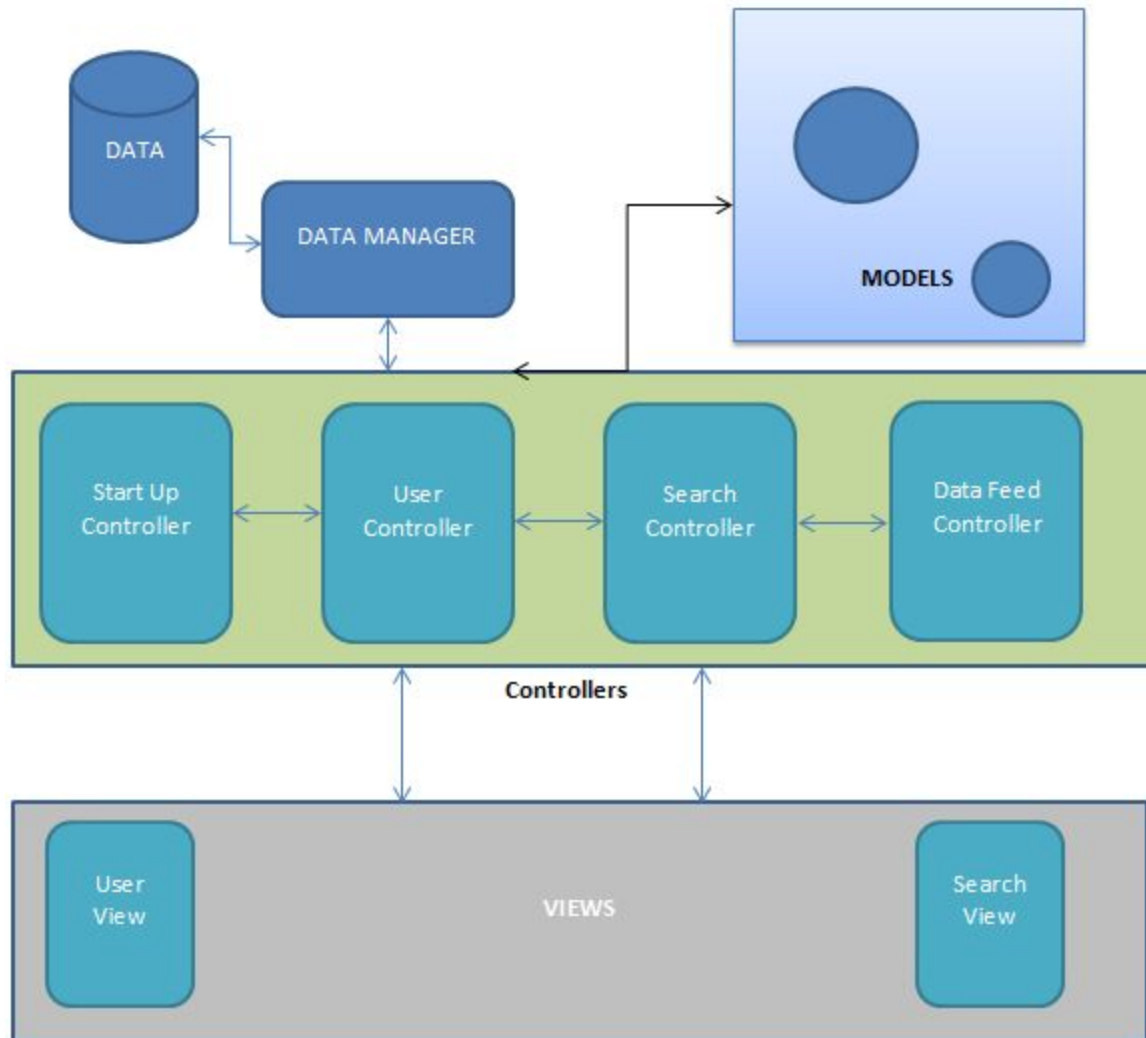
[Critical Issues](#)

Application Scope

This application enables users for the following two functionalities

1. To Create a new account
2. To Login
3. Change Password
4. Search Coupons
5. Display Search Results

High Level Architecture Diagram

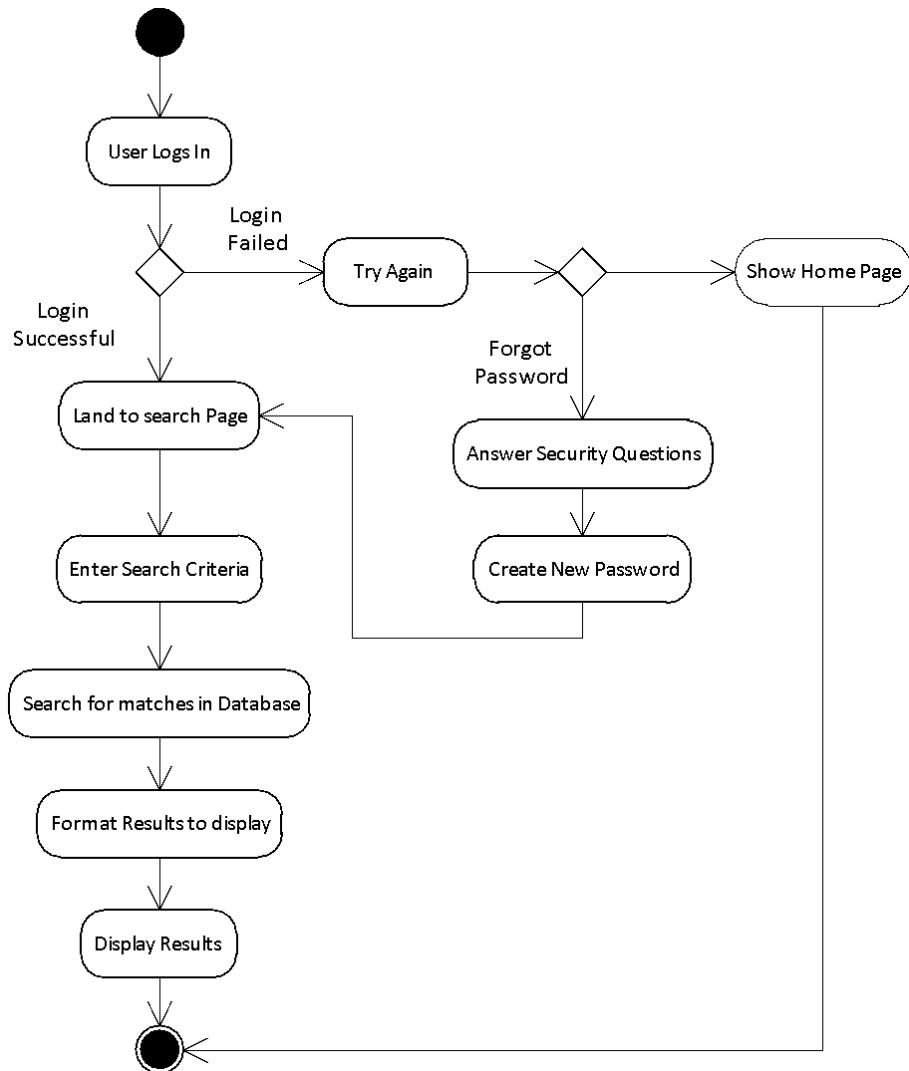


The system is designed to use the concept of Model View Controller pattern. This enables the system to keep the partitions separate. It has a set of controllers and views which works around models to provide the functionalities required by the system. The separation of concerns is maintained to

promote the maintainability of the system. Also the data feed controller is designed to fetch the data in the data base from some other data center.

There may be more views added in the system as per the requirements. This diagram shows the views which are necessary for the system to work.

Application Activities

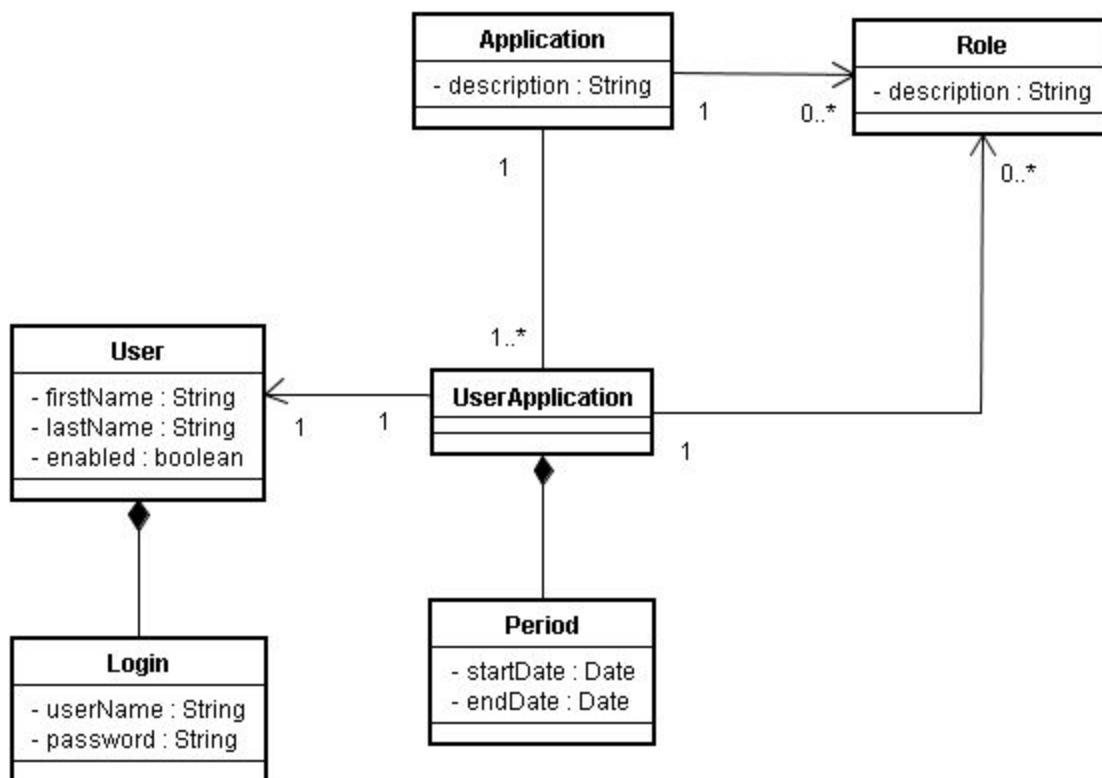


Activity Diagram

This Activity diagram assumes that the use is already registered and there is relevant data in the database.

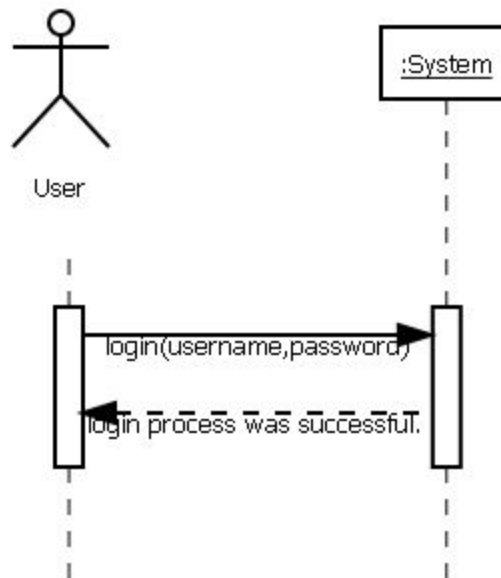
Class Diagrams

User Login Class Diagram



Each user will have a set of properties as specified in the class diagram given above. The role will be used to identify the type of the user. The user may be an Admin or a business user. When user deletes its records the system will mark it as disabled and will not delete the user from the system. This will help in the audit and in keeping a track of the users coming back to system.

Sequence Diagram for user login



The user controller will provide the service to check the username and password of the user attempting to login into the system. If the user is validated then the service returns true other it returns false.

Frameworks to be used

For ORM – Hibernate 3.3

For User Interface – JSP, HTML, Javascript, css

Controllers and Models – J2SE 1.5 with Spring

Server – Tomcat 5.x

Database – Mysql 5.x

IDE to be used: Eclipse 3.x

All of the above given frameworks and tools are free to use. So there is no additional cost involved in the development process.

Critical Issues

Some critical issues identified are provided in as under

- **High Number of search results**

Solution: Server side pagination will be used to eliminate this issue. The views will make user of the pagination technique so that the user can browse the fetched results. The result set will maintain the state using the cursor provided by the database. The number of results viewed by the user can be changed and the pagination will adapt to the new changed value automatically.

- **Memory Issues + Performance**

Solution: The system will make use of advance references provided by the programming language to keep the primary memory usage to minimum. For Performance SLA performance testing will be done to fine tune it.

- **Association of Tags**

Solution: The tags will be associated for each coupon and the coupon entities will contains a collection of these Tag Ids. This all mapping will be done in the ORM Meta-data. The controller will enable the user to make the conditions like including, NOT, OR etc.