

Observers

I used the observer pattern with the Logout Action. The LogOutObserver was the subject. The SimpleNotificationObserver was the observer. The subject belongs to the Presenter layer, and the observer belongs to the Model Layer

Generics

I use generics in BackgroundTaskHandler. The generic type T can only be an object that implements the ServiceObserver interface or implements an interface that extends the ServiceObserver.

Template

The AuthenticateTask uses the template method pattern

```
@Override
protected void runTask() {
    Pair<User, AuthToken> loginResult = runAuthenticationTask();
    This is the part that is deferred to subclass ^^^^^^^^
    authenticatedUser = loginResult.getFirst();
    authToken = loginResult.getSecond();
}

protected abstract Pair<User, AuthToken> runAuthenticationTask();
```

runAuthenticationTask() is deferred to the class's subclass. The function is implemented in LoginTask and RegisterTask.