Developing Fabflix (2)

UCI - CS122B
Jamshid Esmaelnezhad
April 12th, 2016

Outline

- How to make sure all requests go through authentication first?
  - Using filters in Tomcat
- Implementing the “single movie information page”
- Implementing the search feature for movie titles and director name
- How to add the pagination feature to search
- How to implement the shopping cart
- How to use cURL

Using filters

- What is the filter chain?
- What is the execution order of the servlet with respect to the filter chain?
  - What if we use forward from the filter chain? (Try it.)
- How to add a filter in Tomcat?
  - Add the class
  - Add the filter mapping tags to deployment description (web.xml)
- What are the use cases of a filter chain?
  - Authorization
  - Logging
  - etc.
- Read this tutorial, this one, and this one.

Single Movie Information Page

★ Let’s use our in-class example...

Implementing a Search Box for Fabflix

- What should we include in the request?
- How would the backend modules look like?
  - What should go to the Controller codes? What should go to the Model?
  - Will this be easy to add the search functionality to other places in future?
- Side lessons:
  - Including another JSP file. Also read about the View Helper design pattern
  - How to map a URL to a JSP file
  - Showing error messages.

Pagination

- What new parameters do we need in the request?
- Changes to the backend module: OFFSET and LIMIT clauses in SQL.
- How can we use caching to improve the performance here?
  - What can we cache?
  - How can we separate the cache logic from the main search module?

Shopping Cart

★ Just another example of a *state* of the user

Using cURL

- What is it?
  - Tutorial
- Use the following command to install cURL - on Ubuntu:
  - `sudo apt-get install curl`
- Usage example:
  - `curl http://54.200.164.182:8080/MiniFabflix/search?keywords=Spider+Ma`