Homework 2: HTML Forms and Simple Servlets

For this homework, you create a simple HTML form and servlet, and gain more practice using Subversion (svn).

You will do this homework as a team; however, each member of your team will be responsible for the completion of a particular task.

Step 1. Checkout a project skeleton

Checkout the **trunk/** subfolder of **movieclub/** in your team's svn repository.

This step is similar to last time; however, for all future homeworks, you will use this same skeleton. (Thus, you might never need to check it out again.) As before, each team member must perform this step, so that each has a working copy of the project.

Step 2. Create a web page

Each team member must choose one the web-page designs in the Tasks section (below) to build. All team members must do a different task. If your team has only 5 members, then ignore Task 6.

Each team member must create an HTML5 file in the **src/main/webapp/** directory, and name the file **task*X*.html** (case sensitive!) such that *X* is the number of the task (e.g., **task2.html** for Task 2).

The HTML file you create must have all features depicted in the design. Each design features an HTML form. Note that your page should use only basic HTML (no CSS, no JavaScript).

When you submit an HTML form, the form's target receives a set of name/value pairs that correspond to the user's input. Thus, you must make sure that your form properly specifies all those name/value pairs. To do so, you must set **name** and **value** attributes for your form elements.

Specifically, you must make sure that each **input** element (except submit buttons) and **option** element has a **name** attribute and the name defined is unique (with respect to the form). Your servlet will use these *names* to look up what *values* the user entered. You must also specify **value** attributes for each non-text input element. (Text-base elements send the user-entered text as their value.)

Also, you must fill in the **target** attribute of your **form** element as follows. If your HTML file is named **task*X*.html**, then enter the target as **task*X*.do**. Finally, make sure that your **form** element's **method** attribute is set to "post".

Hint: The http://www.w3schools.com/html/ pages are your friends—especially those on Tables, Forms, and Symbols.

Step 3. Create a servlet

You must create a servlet that corresponds to your HTML form. If your HTML file is named **task*X*.html**, then name your servlet class **Task*X*Servlet**, and set its URL pattern to **/task*X*.do**. Recall that URL patterns are set with the Java tag **WebServlet**. For example:

```
    @WebServlet("/task7.do")
    public class Task7Servlet …
```

Note that your form's **target** should match your URL pattern, except the **target** should not have the leading slash.

Your servlet must respond to HTTP POST requests with a proper HTML5 web page that prints out each of the *names* that your form submitted along with the user-entered *values* that correspond to that name. Hardcode the names that you're interested in into your servlet. (Do *not* make a general-purpose servlet that prints all name/value pairs associated with the request no matter what the names are.) Although I'm leaving the style of this page up to you, make it neat and easy to read. Also, provide a link at the end of the servlet-generated web page that takes you back to the form.

Step 4. Submit (by tagging) your team's submission

Attention! Before performing this step, you must make sure that all team members have committed their edits to the **trunk** in the repository.

Only one team member (the leader) performs the following.

First, you must fill out the **README.txt** file in your project's **trunk**. The file should list which team member performed each task (one team member per task).

To submit work in this course, you must tag it. Then, I will checkout the revision that you tagged and grade it. By tagging, you tell me that you are done, and this is the version you want me to grade.

The tag you must use for this homework is **hw2** (case sensitive, no spaces).

To tag the current revision of your trunk as **hw2**, do as follows:

1. Go to the **SVN Repository Exploring** perspective in Eclipse.
2. In the **SVN Repositories** view, find the **trunk** folder that you want to tag.
3. Right-click on the **trunk** folder, and click **Show History**. This should open the **History** view with a table listing the past commits to the **trunk**.
4. In the History table, right-click the newest revision (i.e., the one with the greatest revision number), and click **Tag from…** This should open a **Create Tag** dialog.
5. Enter **hw2** into the **Tag** field and optionally enter a log comment, then click **OK**. This should create the tag!

To verify that tagging was successful, open the following URL in a web browser (replacing *YOUR_TEAM* with the appropriate name):

https://utopia.cs.memphis.edu/course/comp7012-2013spring/teams/*YOUR_TEAM*/GameDen/tags/

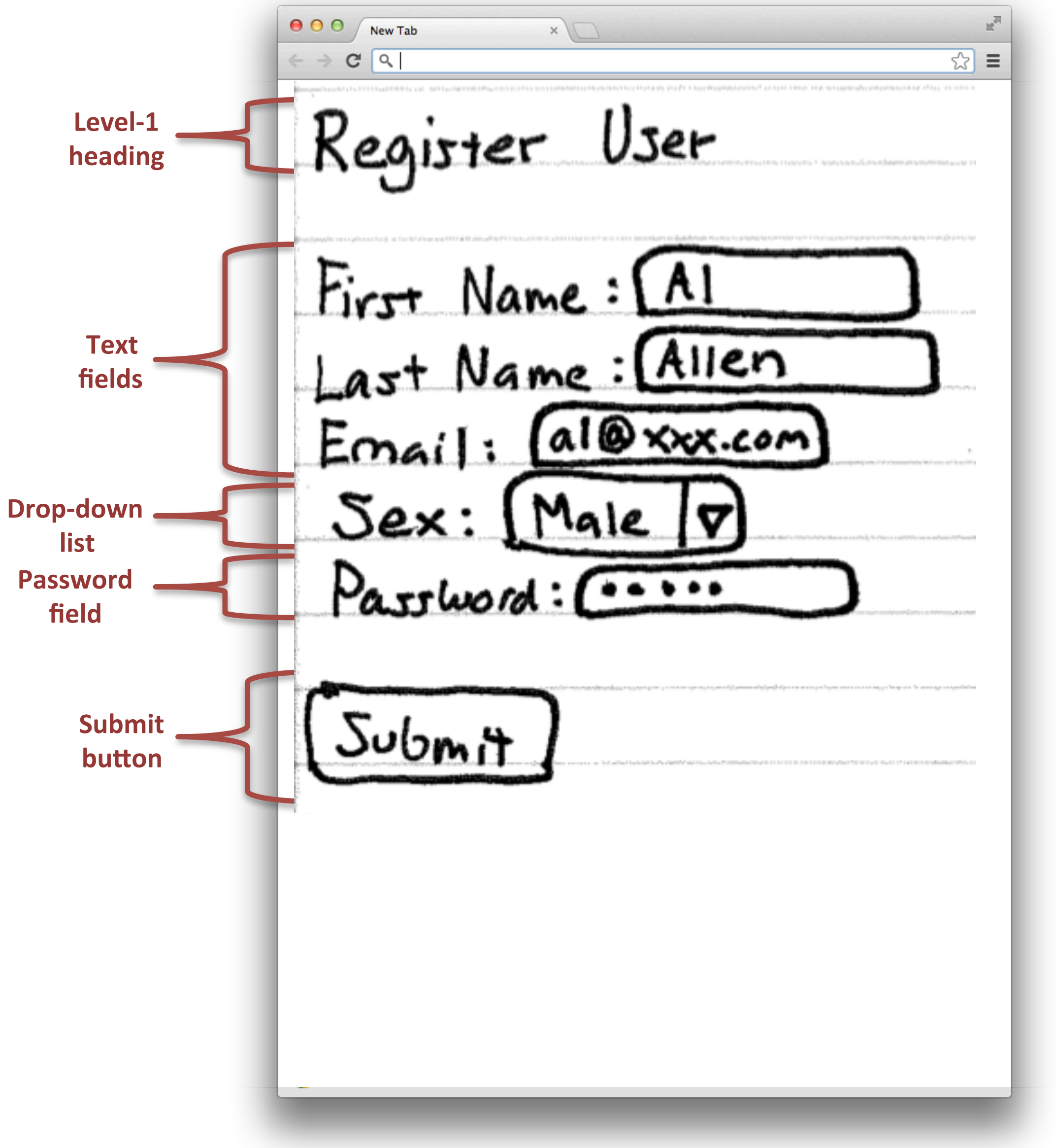
You should see an **hw2** folder, and within that folder should be the **src** folder along with the **.project**, **README.txt**, and **pom.xml** files. Everyone's HTML files should be in the **src/main/webapp/** folder, and their Java code should be in subdirectories of **src/main/java/**.

The Tasks

Below are the 6 web pages to be implemented by your team. In some of the pages, there are drop-down lists. Fill those lists as follows:

- Movie titles:
  - Caddyshack
  - Elysium
  - Pulp Fiction
  - The Godfather
  - Lawrence of Arabia
  - Jaws
  - Dr. Strangelove
  - Skyfall
  - Knocked Up
  - Donnie Darko
  - Beverly Hills Cop
  - 500 Days of Summer
- Genres:
  - Action
  - Comedy
  - Science Fiction
  - Horror
  - Drama
  - Documentary
  - Thriller
  - Romance
- Ratings:
  - G
  - PG
  - PG-13
  - R
  - NC-17
- Movie years: 1950–present (listed in reverse order)

Task 1: Register User



Register User

First Name : Al

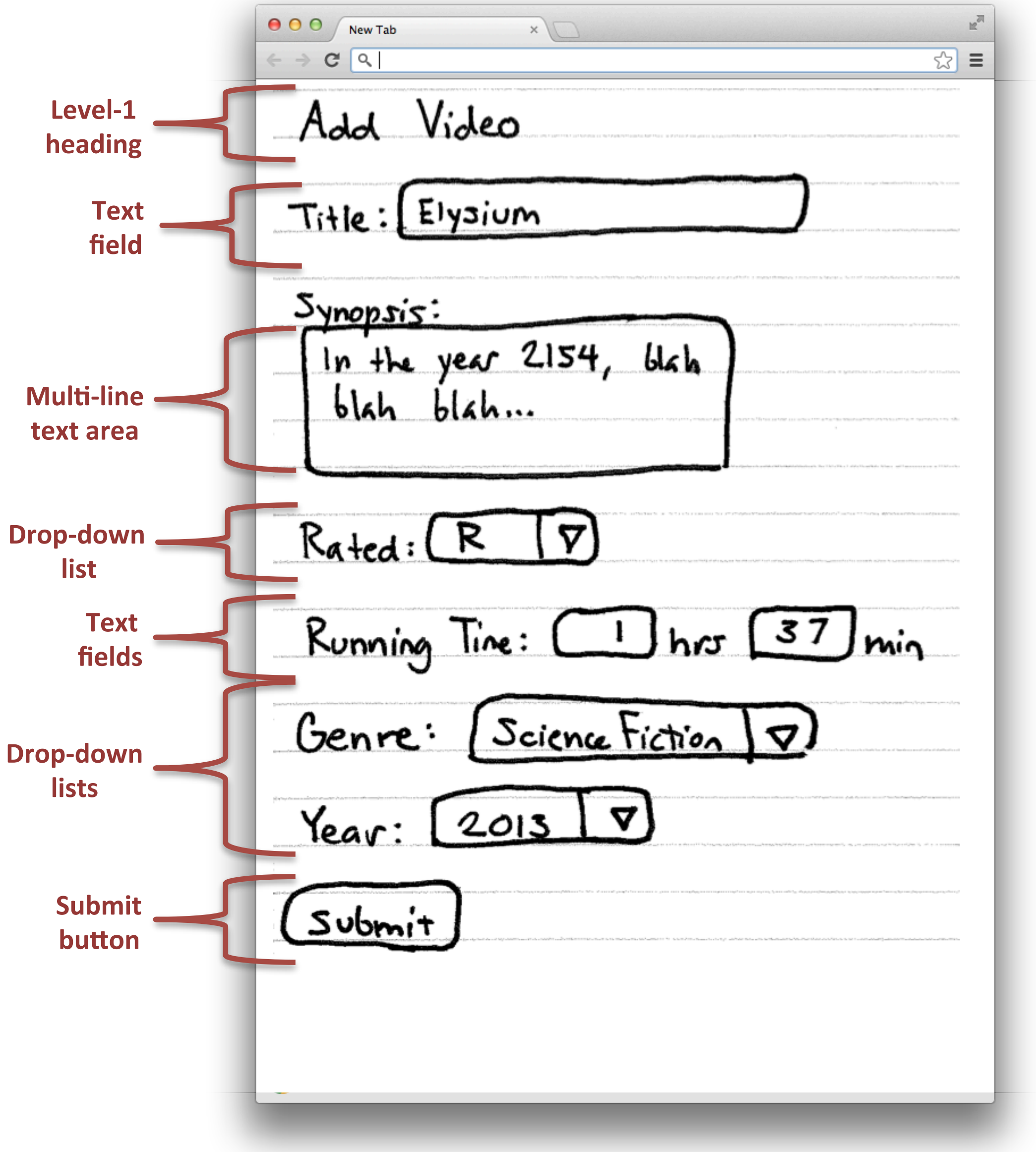
Last Name : Allen

Email : al@xxx.com

Sex : Male

Password : •••••

Task 2: Add Movie



Add Video

Title: Elysium

Synopsis:
In the year 2154, blah blah blah...

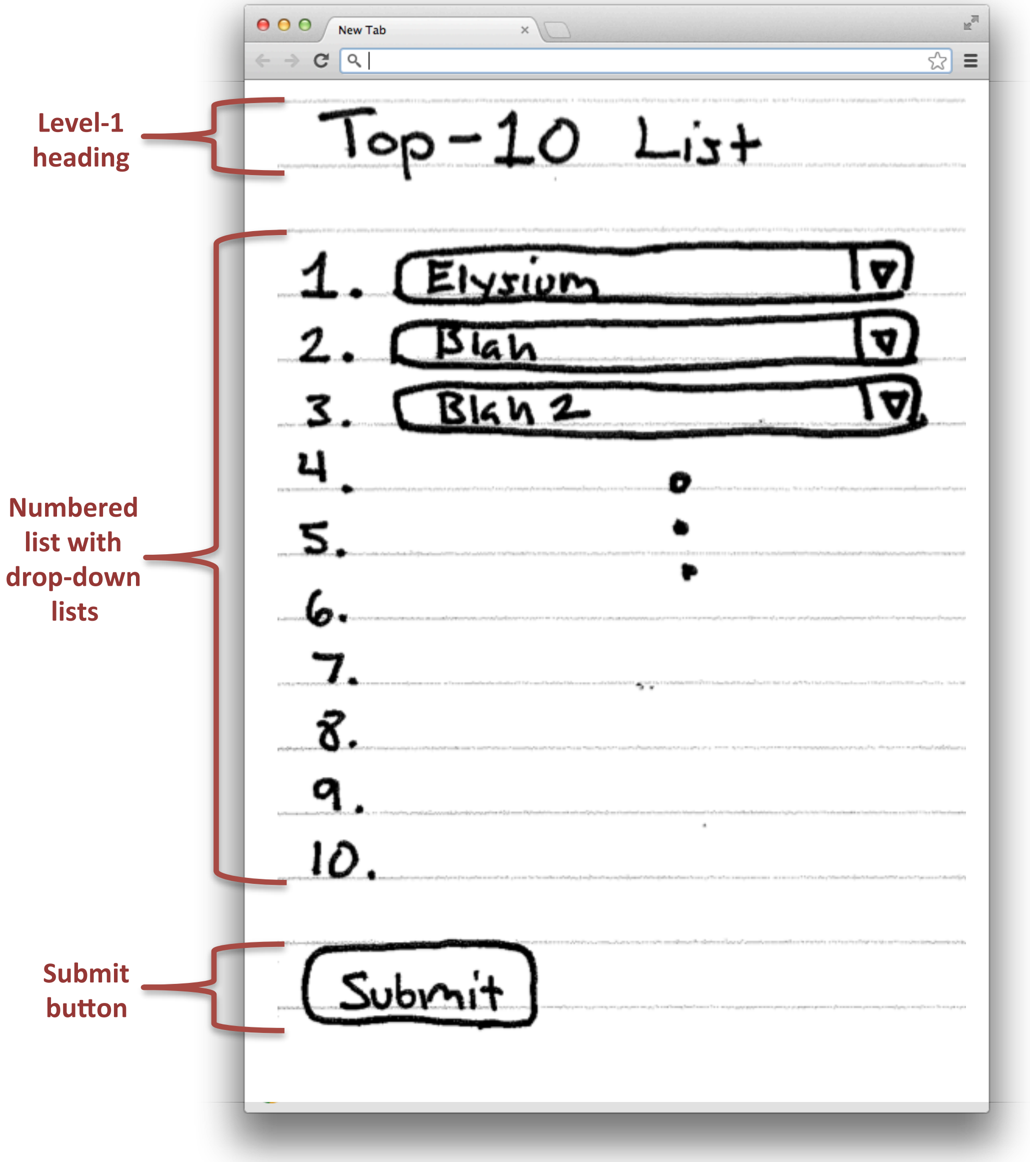
Rated: R

Running Time: 1 hrs 37 min

Genre: Science Fiction

Year: 2013

Task 3: Top-10 List

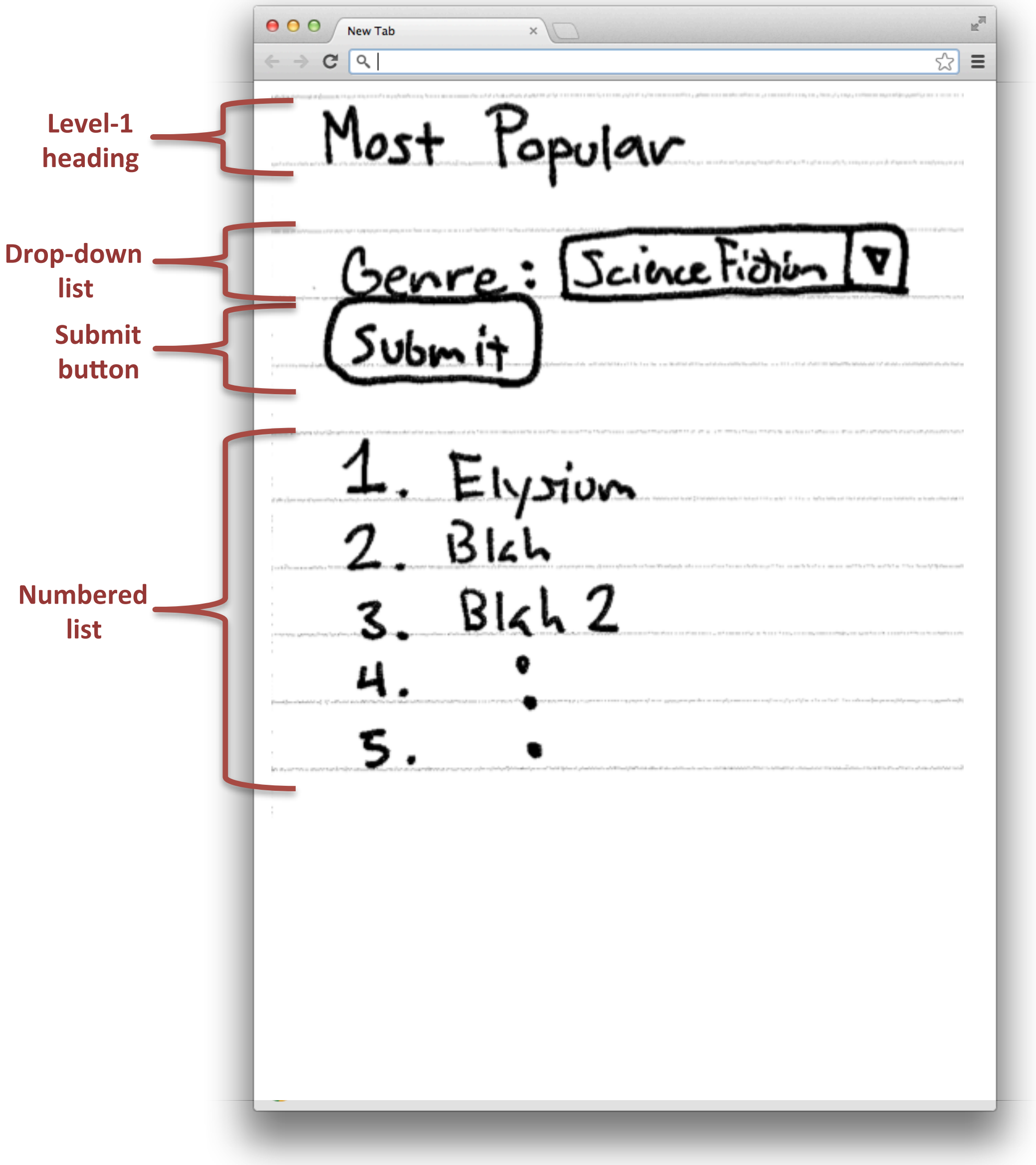


Level-1 heading

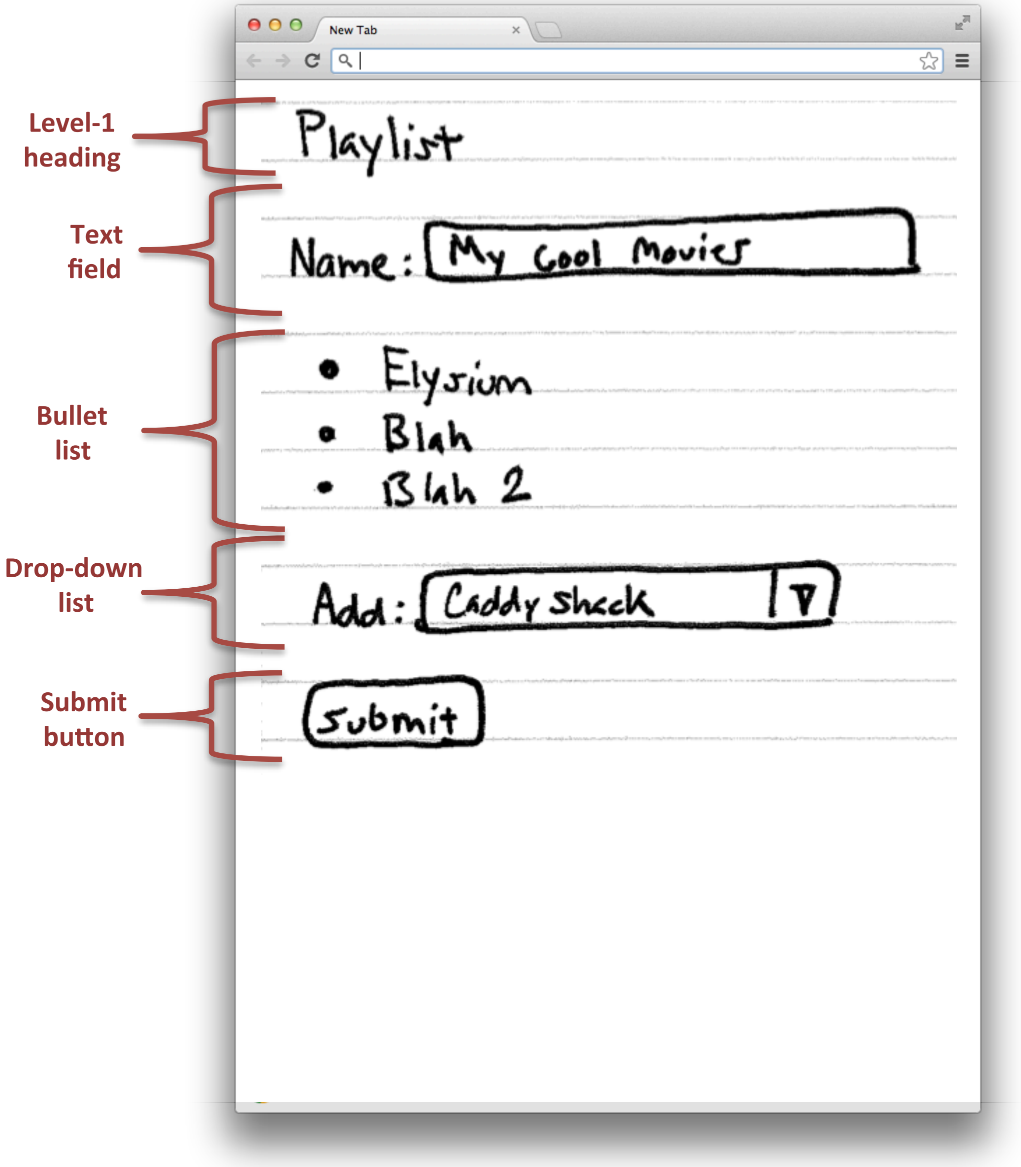
Numbered list with drop-down lists

Submit button

Task 4: Most Popular Movies



Task 5: Movie Playlist



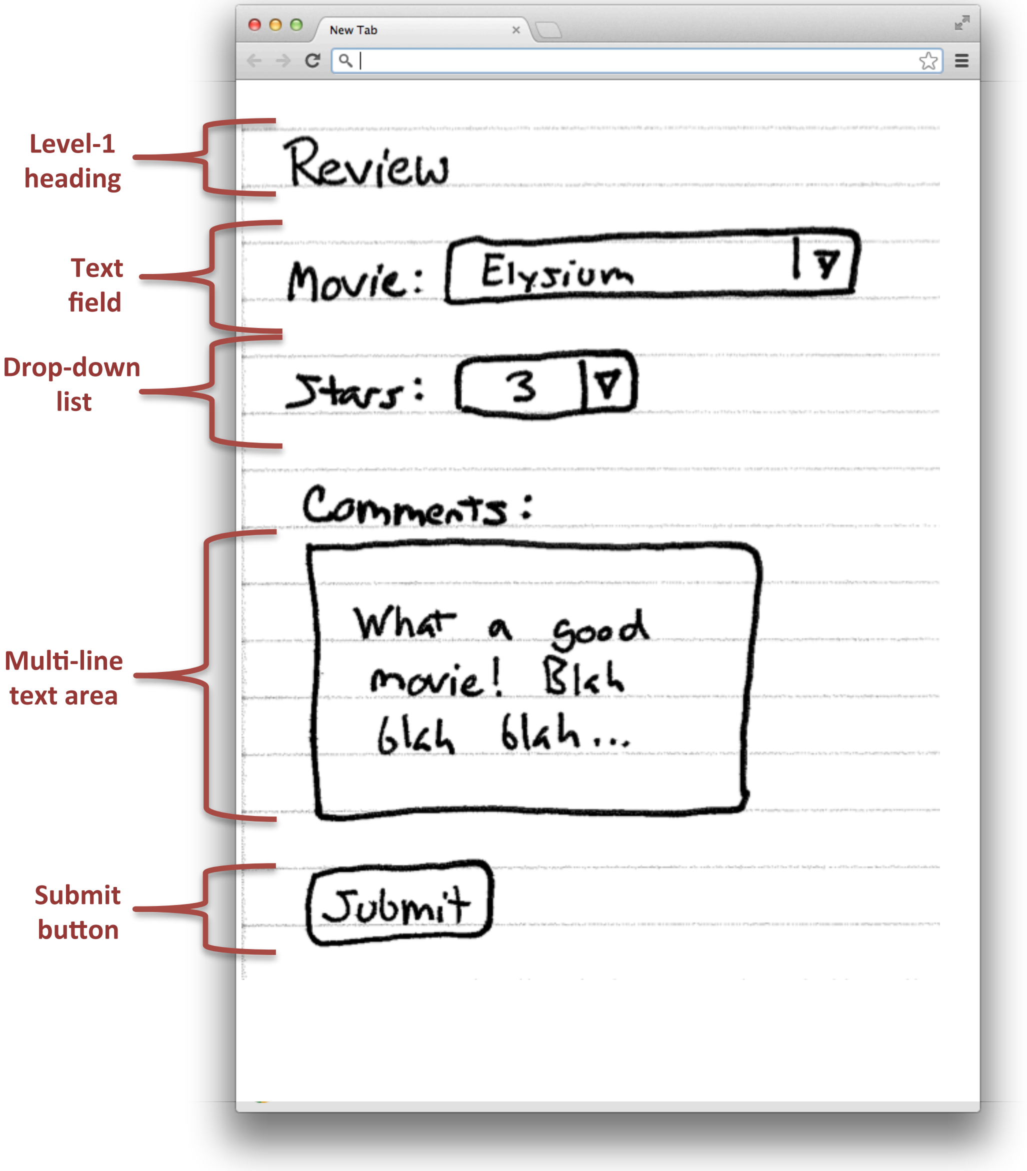
Playlist

Name: My Cool Movies

- Elysium
- Blah
- Blah 2

Add: Caddy Shack ▼

Task 6: Movie Review



Review

Movie: Elysium

Stars: 3

Comments:

What a good movie! Blah blah blah...