Homework 7: Search Application Usability

Due ***Friday*** 3/13 @ 4pm on Courseworks. (note extended deadline)

This assignment is the equivalent of a midterm in other classes – it displays cumulative knowledge of the things learned in all the lectures in the first half of the class. In this class, we think the best way to show mastery of the material is not to take a written test for 2 hours, but to spend a week implementing a project.

Consider this assignment a take-home midterm and treat it with the same urgency as an in-class written midterm. However, you are still allowed to google things, come to office hours, ask for friends to look at your code, and seek other forms of help. Even people who are masters of this material need to google things and to ask for help. However, if you have mastered the material, you will be able to do this assignment within the time frame, including managing other midterms you may have. I recommend starting early.

If you request an extension on this assignment – the normal rules apply. Wednesday 3/18 at 4pm is the latest it can be considered. Additionally, the staff will not hold office hours or answer piazza questions after 3/13. It's spring break –all the staff on a beach without access to the internet.

What to submit:

- 2 uploads titled:
  - Hw7.zip: A Flask project containing:
    - server.py
    - templates/ (and the HTML templates you need)
    - static/ (and any static files you need)
  - hw7_video.(mp4/mov/other video file type)
    - a video of your site where you demonstrated each of the features

Part 1 – Usable Functionality:

1. **Menu/Navigation**.
    a. For consistency, all the templates should be rendered with a shared template that contains a navbar.
    b. The navbar should contain:
        i. A home link (at the "/" route)
        ii. A text box to enter a search query and a "go" button (at the "/search" route). When the user presses enter on the search bar it should also "go".
        iii. A create link (at the "/create" route)
2. **Home**. The home link should render at the "/".
    a. It should contain a one sentence summary of the mission of the site. This mission should make it clear who the intended user is and what specific goal it helps them achieve.
    b. It should show the latest 10 entries added to the database to entice the viewer to click on something and start exploring.
    c. Each of the 10 entries should be formatted as a Bootstrap Card that contains an image and the title of the item. If there is some other essential field, it can show that too, but it should not show all the data fields – it's meant to be a summary.
    d. When you click the image, it should take you to the page for viewing the item.
3. **Search**. When the user presses "go" on the search link (or presses enter), it should search for the items and return a list of all matching results.
    a. **Flexibility**. The query must do substring matching that is not case sensitive on the title and one other text field.
    b. **Feedback**. In addition to returning the results, the page must say how many results there are. If there are zero results, you don't need to do anything other than say there are zero results.
    c. **Feedback**. When you present the results to the user, the bit that matches the substring must be easy to scan for, according to gestalt principles.
4. **State/Options/Transitions**. On the template for creating a new database item, you will still have input boxes for all the fields the user must input. In addition:
    a. **Error Detection**. When creating a new database entry, there must be error handling on all the fields. If the field must be a number, then ensure it is a number. At the very least, you can check that the field in not blank (remember to trim the text to test if it's blank). Design the error feedback so that it directs the user's attention to the right place to correct the error.
    b. **Transitions**. After the user presses "submit" and the data successfully submits, allow the user to either view the item or enter a new item.
        i. At the top of the page it should say, "New item successfully created." With a button or link that says "see it here" (or words to that effect). This links to a page for viewing the item.
        ii. Additionally, the input boxes should clear and the focus should be placed on the first text box so the user is ready to submit another item.
5. **State/Options/Transitions**. There will no longer be a separate /edit/<id> route. Editing will now be done in /view/<id>

    a. For each field that can be edited, create a small edit icon next to it. (at least two fields must be editable – including one that is involves changes the text)
    b. **State Change**. When the user presses the “edit” icon the field to be edited, it must immediately turn editable with a “submit” and “discard changes” option.
    c. **Options.** The chosen text must disappear, and in its place, there should be a textbox or text input with the text they way to edit, with the focus in the input field.
    d. **Transitions**. After the users presses “submit” or “discard changes” the page should go back to how it looked when they were viewing it (and not editing it).
    e. Note: If your “update” was to add a review to a list, you don’t need to populate the textbox with any text. You may call it “add review” instead of “edit” if you like.

6. **User control and freedom (Undo)**.
    a. The user should no longer be able to delete entire database items from the search page.
    b. Instead, in the /view/<id> route, in addition to making at least two field editable, you must make it possible for the user to delete any item they want from the field that contains a list.
    c. Once they delete, it, a button should appear in it’s place called “undo delete”
    d. On the back end, the item should not actually delete from the database, instead it should have a field called “mark_as_deleted” which starts as false and you update it to be true.
    e. When you display items in that list, you only display the items where “mark_as_deleted” = false. (This is like the Gmail “delete” that we discussed in class.
7. **Accessibility**. All media must have alt text. In the video you will, open the Inspect page tool of the browser and click on a few media items to and show that there is alt text.

Part 2 - Graphic Design

There are now 5 distinct states the user can be in:

1. Home
2. Search results
3. View item
4. Update item
5. Create item.

In each state, there should be good information hierarchy using all the principles taught in class, including: 7 tools for visually indicating importance, information scent (good labels), and insights about color, text and gestalt. It doesn't have to be fancy, but it should pass the squint test (where the most important things is visible) and it should have good conceptual grouping – things related to each other should be close together. Use color last! We will take off points for gratuitous use of color where other design heuristics should be used instead.

In your video, show a user doing the following **in this order**:

1. Coming to site and clicking on a card to randomly explore the data.
2. Typing in the search bar for something they are looking for, using only a substring of the thing they want. (for example, typing “big” when they want “The Big Lebowski”)
3. This should match multiple results and present them to the user.
4. The user clicks on the one they want.
5. The user views the item.
6. They see something they want to update, enter the update state and press save. We should see the update reflected on the page.
7. Go to a new browser tab and show that the update has indeed taken place on the other page (this shows us that the update is on the server side, not just the client)
8. They then try to update something else, but press “discard changes instead of change”. We should see that the update was not done.
9. Delete a list item from the view/<id> page. Go to the url in another browser tab to show us that it is deleted for the other user as well.
10. Go back to the original browser tab and press undo. Reload the page in the second browser tab to demonstrate that the data has also re-appeared on this page.
11. Click the create button and create a new item. The data in the fields must be realistic. It can’t just be garbage text or placeholder text.
12. Show the user performing an error in creation and getting the error preventing message.
13. They correct the error, then submit it.
14. They click the button that says “see it here”
15. After viewing the item, they click on the home button on the navbar. This takes them to the home screen where they see their new entry first on the home page (because it was the item most recently created).
16. Use the inspector tools of the browser and hover over a few image elements to show there is alt text. Two media elements is enough.

Your video should be short and sweet. If it’s longer than 5 minutes, you’re doing something wrong. 2-3 minutes should be plenty.

You don’t need to do voice over, but you can if you want. It usually takes me 3 tries to get a good recording where I actually remember to do all the things. Give yourself some time to do this. Usually an hour is enough.

In general, you have some creative license for what you name buttons and other bits of text on the UI. As long it is clear from the video what the function is, it’s fine.

We prefer you use a screen recorder (Quicktime is great for this), but you can also take a video with your phone.