CS-639 Building User Interfaces, Fall 2020, Professor Mutlu

Javascript β (2 Points)

Empathy-Driven Redesign Using Component Libraries

GitHub Classroom Starter Code for JavaScript β

In this assignment, you will practice design thinking and visual design methods to improve upon Badger Bank. You will complete this assignment in three parts.

In the **first** part, you will practice using the "empathy" method (the think-aloud protocol) that we have discussed in class as a method that will give you the most bang for the buck in terms of understanding user needs, preferences, and behavior. Specifically, in the think aloud, you will (1) identify users who represent your target group of users, (2) identify/develop tasks that represent the functioning of the target system, (3) observe users as they perform the tasks, and (4) analyze your data to develop design insight.

In the **second** part, you will practice the ideation and visual design principles you have learned in class to turn your design insight in the first part into a redesign of the Badger Bank website you have created, expressed in a layout sketch.

Finally, in the **third** part of the assignment, you will implement the design you created in the second part using the Bootstrap component library. This three part assignment will take you through user research to ideation and from sketching to implementation of user-facing elements. Follow the instructions below to complete the assignment.

Part 1: Think-aloud

(0.1 Points) **Step 1. Identify users.** Who uses online banking? Describe below the characteristics of this user group, identify one person (a roommate, a friend, or a family member) who might be willing to take part in your user research, and ask the person for their interest/availability. (If the person you identified is in this class, it is acceptable to swap roles.)

Those people who are young, educated and like to try new things use online banking.

(0.1 Points) **Step 2. Develop tasks.** Study the new Badger Bank application, Javascript **β**, to develop a set of tasks that users might perform with the system. Rank your list in terms of importance (simultaneously considering impact, frequency, prevalence) and identify the top three tasks. Describe each task in 1–2 sentences in a way that your users can understand.

Note: As banking involves private and sensitive data, be sure that your tasks do not require your user to reveal private information (e.g., login credentials, bank balance) to you or anything else that may make them uncomfortable. Your tasks can focus on the general visual and click-through navigation of the site and interact with personal information in a very limited sense.

---

1. QuickPay with Zelle. This is used for private transactions. The task is the most frequently used by the user in online bank and also popular in student.
2. Digital Assistant. You can search for help by talking with the digital assistance. There are some guidelines in the window that is very convenient.
3. Deposit Checks. Check is one of the most common payment methods. You can just can the check then the system will automatically deposit money to you account.

(0.2 Points) **Step 3. Perform think-aloud.** Hold a think-aloud session with your representative user (e.g., over Zoom using screen sharing) *on a real banking website*. First describe to your user how the think-aloud will work (refer to the reading and class notes), describe the tasks one at a time (answer any questions you might have), and ask them to perform each task while they say out loud what they are thinking.

> **Pro tip:** If you see your user performing the task but not speaking, probe them by asking what they are thinking or reminding them that they should be describing.

As you observe your user performing the tasks, take notes (using the other sheet) of important actions, problems they encounter, confusions they might voice, and so on. For anything that stands out, after each task, ask your user why they did that or said that. Your observations and notes will form your data. Include your data below.

Note: If your tasks involve entering or reviewing personal information, you can ask your user to stop screen sharing while they perform these actions in order to preserve their privacy.

---

1. The homepage has very limited functions, if the user wants to find some specific functions he should to look the inner page of the online bank, for example, viewing the statement.
2. Most of the homepage is occupied by the advertisement of bank promotion, and many task menus are concealed in the inner page and not easy to find.
3. After signing in user's account , only three main tasks are in the account page.
4. The user always used Zelle for transaction, but the direct menu has both "pay" and "transfer". It makes him little confused.
5. The user thought the few people will concern the news of the bank on the homepage, but it takes up a lot of space.
6. The user tried to use the automatic location service of his cellphone to find a local branch, but the website only support ZIP search.
7. The user tried to have online chat with customer service representatives, but my design did not include such a capability.
8. The user wanted to compare different types of checking accounts in a table, in terms of interest rates, fees, minimum requirements to waive service fees, but my design did not include such a feature.

(0.2 Points) **Step 4. Create insight.** In your data (e.g., notes), highlight where you saw significant breakdowns in functioning, need for better functioning, or user preferences that would require an alternative design. Make a list of your findings as design recommendations.

---

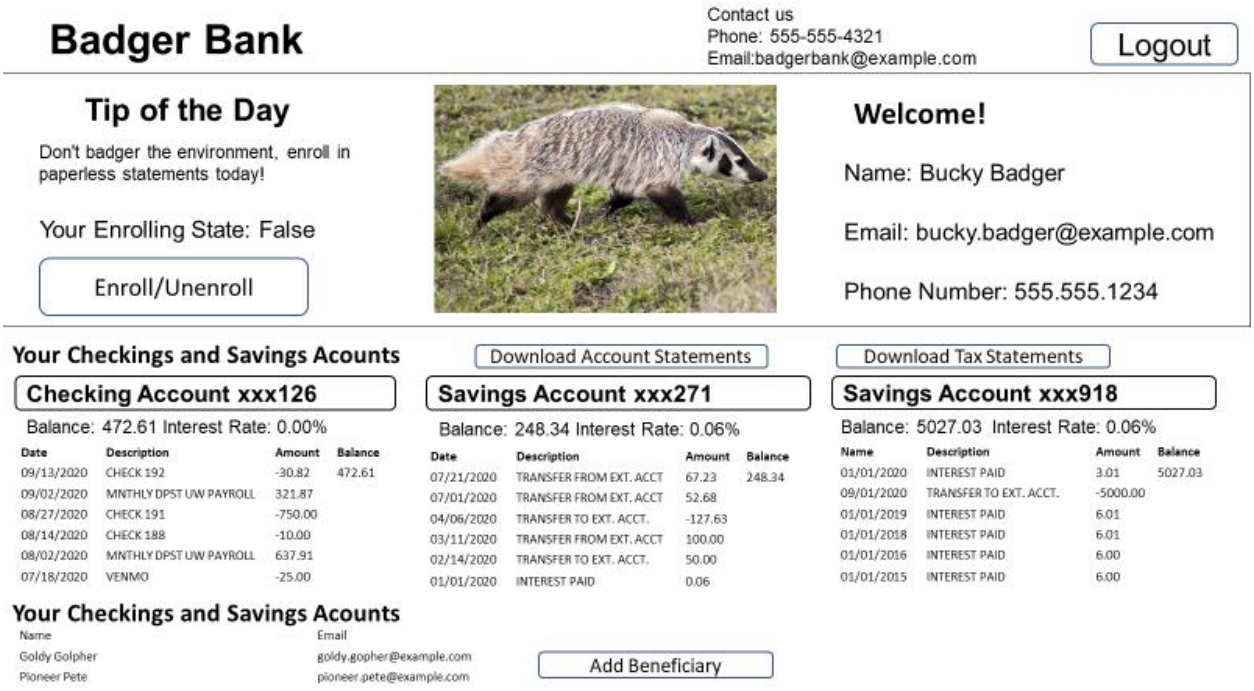
1. Reduce the space of advertisements and news, make more the tasks can be found easily.
2. Add some announcement on the webpage.
3. Present more task menus on the account page.
4. Add some most frequently tasks as the focal point of the homepage
5. Add automatic location service.
6. Add online chat function.

Part 2: Design

(0.4 Points) **Step 1. Visual Design.** Next, using knowledge of banking operations from the think-aloud, you will redesign JavaScript **β** with the goal of improving its visual design. First consider what elements you must place on the canvas. All elements should have a function. If the original design included elements with no function or use, you can exclude them in your redesign. Next, consider the design principles, and ask yourself, "how can I direct user attention appropriately," "do I need to create contrast between elements," "how do I achieve unity," and so on. You will need to go back and forth between the elements and principles. For example, you must determine what user attention should be directed to, such as a product photo, a button, or a paragraph of text, in order to place the focal point to it. Similarly, you should

think about the composition of elements to create an appropriate level of balance on the page. Your redesign should be in the form of a digitally- or hand-drawn wireframe with annotations that justify the use of the elements and principles.

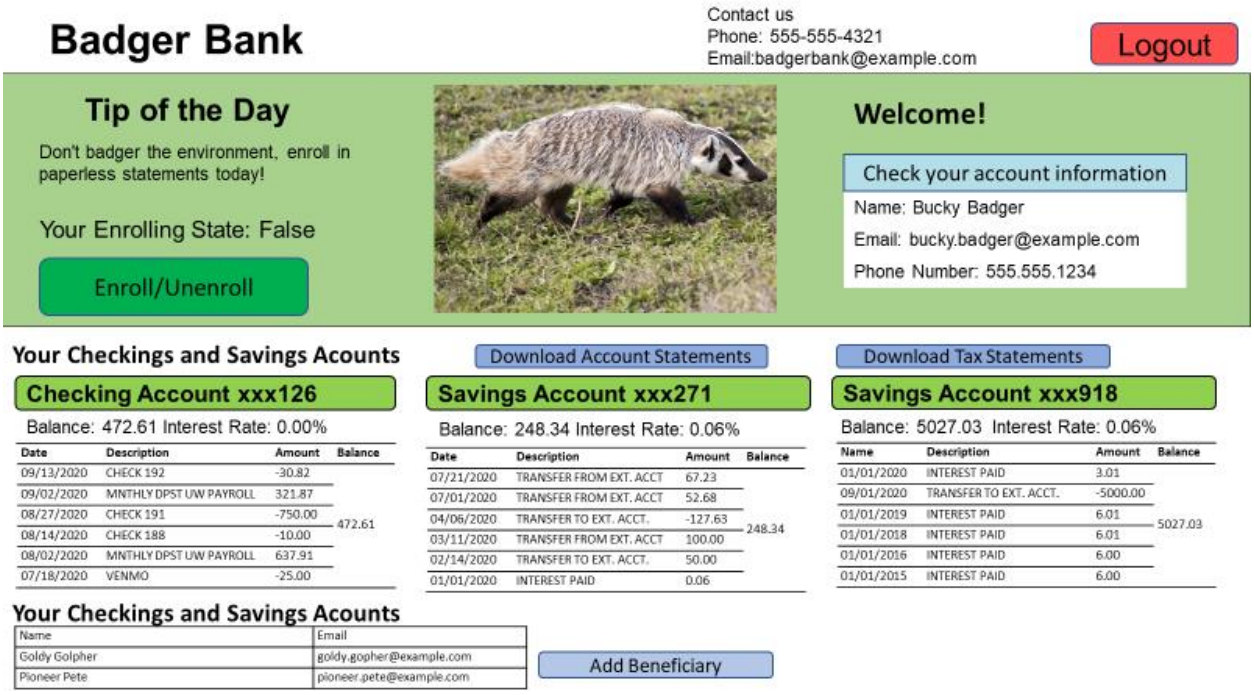
<describe-your-designs-here>
I use the elements including space, line, shape, size in my visual design. Between the head and middle part with picture and the three account tables, the space is arranged to make the whole page comfortable. I use a rectangle box which consists of lines to contain the "tip of today", picture of badger and account info. Also, different size of buttons with rounded rectangle shape are design to build a hierarchy style. For example, the "Enroll/Unenroll" button with large size to attract user of attention, hoping more users can attend this activity. This can be viewed as a focal point. Three account information tables are aligned in arow with same shape and style keep the balance view. Moreover, such design can create a unity scene .



(0.2 Points) **Step 2. Specify color and type choices.** Finally, determine what color palette your redesign should follow. How many colors/shades will you use? Why will you use these colors? Keep design principles in mind when you are choosing colors. For example, using contrasting colors, you can create contrast and manage user attention. Additionally, determine what category of typeface and what font you will use, whether or not you will use multiple fonts on the page, and how you will parameterize each font. You may review the fonts in your computer's fonts folder (on Windows, go to "My Computer > Control Panel > Fonts"

and select “View > Details;” on the Mac use the Font Book app) or the Google Fonts collection to give you ideas and get familiar with available typefaces.

<describe-your-choices-here>
Only three kinds of colors are applied in the design. Since the picture has green background, I choose the similar background color in the middle box. The text color is mainly black. Different buttons are set as different. “Enroll/Unenroll” button with green color implies this is an environment friendly activity, meanwhile, it matches the background of the box. All download buttons are set as blue as the bule button represents information button in bootstrap. The logout button is red for the sake of easy find. I use contrast color of “balance” and the its value to make sure users can see that at a glance, and the same consideration was taken in the checking account information. Text font are all set as “Arial” because I think this kind of style looks gentle and same font make the page clean and integrate.



Part 3: Implementation

(0.2 Points) **Step 1. Inspect component library elements.** In this step, you will inspect the Bootstrap component library to see how you can realize the layout, visual-element, and color/type choices you have made in the previous part using the component library. You are not expected to change the library components to exactly match your design choices, but to identify which component elements might best

meet your design goals. Below, copy the design and the choices you generated in Part 2 and annotate them to describe which components from the library you will use to accomplish your design goals.

<include-your-annotated-design/choices-here>
I use the bootstrap button to set the button style. Most are set as the default size expect for “Enroll/Unenroll” button. The larger size of “Enroll/Unenroll” button can attract more people to pay attention to the activity and green represent environment friendly. The account table is added with “.table-hover” class so that users can located one item obviously and will not be misled by other items. The beneficiary table is included with “.table-bordered” class for an direct view. To be more obvious, I decorate the button and contact information with bootstrap glyph icons. Bootstrap panel is applied for present user’s account information giving an impression that your information is written in a panel and now we give it to you to check.

(0.6 Points) **Step 2. Implement your redesign.** The last step of this part will involve implementing the layout and components you had identified in the previous step. You will include Bootstrap in your project and use it to implement your design. You do not have to implement new *functionality*; focus on implementing your *design*.

Your deliverable will be a completed version of this document, attached to the canvas assignment as a PDF, and the GitHub Classroom repository name and latest commit hash.

<include-screenshot(s)-of-implementation>