BrewFrog: A User Interface Facilitating Efficient Product Differentiation

Colton Shoenberger
College of Computing & Informatics
Drexel University
Philadelphia, PA, USA
cs3585@drexel.com

ABSTRACT

The rise of the internet has facilitated increased information availability, which has significantly transformed our economy. Mass-marketing a small number of products for a broad audience is no longer as effective, as consumers search for products that meet their specific and unique needs. As a result, product arrays are now more diversified than ever before, and marketers target smaller segments of the broader population. As a consumer faced with more products than ever before, the value of being able to quickly determine what option best fits their needs is at a premium. Meanwhile, navigability is still one of the top user issues for web interfaces, which suggests an area of opportunity. BrewFrog is a system designed to highlight a user interface that can be used for facilitating efficient product differentiation.

1 Introduction

Throughout the past several decades, the rise of the internet has facilitated increased information availability, which has fundamentally transformed our economy and altered the way we shop. With readily available information, consumers can now better than ever find the best prices and make determinations as to what products are the best fit for their specific needs. In response to this trend, theories have emerged as to how firms might market their products more successfully to well-informed consumers. Mass-marketing a small number of products that meet the needs of a broad audience is no longer as effective. In the place of mass-marketing, the practice of *resonance marketing* is emerging, described by Kauffman et al. (2009) as "the strategy of developing products that produce the strongest favorable responses among targeted segments of the consumer population" [10].

What has emerged is a market flooded with heterogeneous products. This trend has been termed *hyperdifferentiation*, defined by Clemons et al. (2006) as the situation where "firms can produce almost anything that appeals to consumers and they can manage the complexity of the increasingly diverse product portfolios that result" [5].

As consumers are faced with more and more product options, being able to access information quickly and efficiently in order to differentiate products is more valuable than ever. BrewFrog is a project motivated by the desire to design a web application (app) user interface (UI) that facilitates efficient product comparison. While the choice of using the beer industry admittedly originated from my love of craft beer, the choice is in fact an apt fit for this project because, as noted by Clemons et al. (2006), beer is a highly differentiable product [5].

Thus far we have considered the economic context motivating our project research. Next, we will turn to a brief background of research in HCI addressing how to design web apps that facilitate efficient information retrieval.

2 Background

Difficult navigation is a perennial user experience issue, and for decades there has been research and debate on how to best design websites that make it easy for users to find the information they seek. Throughout the 1990s, many attempted to prescribe design guidelines for developers to follow in order to create more usable sites; examples include de Souza and Bevan (1990), Comber (1995), Borges et al. (1996) and Nielsen (1999) [3, 4, 7, 8]. While some suggestions were straightforward to implement—e.g. use headlines to aid scannability, keep pages short to reduce download times—others were more esoteric—e.g. maintain consistency, support users' goals and behaviors.

Unfortunately, there was much inconsistency between the various web design guidelines, and following the turn of the century, there was still a lack of general agreement about which guidelines were correct. Moreover, Ivory et al. (2001) lamented that many of these guidelines lacked any empirical evidence to support their claims [9]. In their study, Ivory et al. employed statistical techniques to evaluate websites judged for the 2000 Webby awards, which were assigned ratings by a panel of over 100 judges deemed to be web "experts". They assessed thirteen "superficial", quantifiable metrics of the sites (e.g. word count, body text percentage, graphics count, color count, etc.) to determine the degree to which they could predict whether a site would receive a good or bad score. In general, their results were able to predict if a site would receive a high score with 65% accuracy; however, if they constrained their predictions to pages within specific categories (e.g. education, community, finance, etc.), their prediction accuracy increased to 80%. While a few of the metrics they assessed had consistent effects across the board—e.g. high graphics count almost always correlated with a worse score, whereas high color count almost always correlated with a higher score—for the most part these metrics depended significantly on the category of the website. The authors conclude that there is simply no one path to good design for all websites, and they suggest that we look to refine methodologies for designing

sites within specific categories, as differing user needs call for different designs.

Thus far we have looked at navigation and information retrieval from a usability and design standpoint. From a different angle, some researchers have considered the problem from a cognitive perspective. Belkin (1980) notes that the issue of information retrieval exists not only in the properties of systems, but also in the *anomalous states of knowledge* of users [2]. In other words, various users may come to a website with different information goals. While some information goals may be conscious—i.e. realized by the user—others may be visceral—i.e. actual but unexpressed. Moreover, whereas in some cases the user may know exactly what is necessary to satisfy their information need, in other cases the user may not know what information would satisfy their need. A clear consequence of this observation in terms of design is that providing a search bar may not be enough to lead a user to their information need, not strictly for faults in the search algorithms, but because the user may not know what to search for in order to satisfy their need.

Another cognitive perspective significant to issues of navigability is Information Foraging Theory, which, as described by Chi et al. (2003), "predicts the information gathering behavior of users in an information environment" [6]. Researchers in this field have discovered that we can predict and simulate what links users with a specific task will click when browsing a site. This research has led to the notion of *information scent*, which represents the user's perception of the value and cost of clicking a link without fully knowing what information is available "on the other side". Research in this field has shown that there are predictable limits to how far people will click on sites to achieve their goal.

From this research, we are presented with many challenges for our system. As previously stated, our goal is to design a web app UI that facilitates efficient product comparison, and product comparison is fundamentally a problem of navigability and information retrieval. Knowing that our users may come to our site with diverse information needs and anomalous states of knowledge, we must provide them with efficient ways of sorting through available data to achieve their goals. In the design of BrewFrog, I attempt to implement menus and filtering criteria that simplify navigability and efficiently elicit and satisfy information needs. Going forward, I draw from Read et al. [1] for guidance on menu design, as well as some of the more practical design guidelines mentioned earlier.

3 BrewFrog – System Description

Our system begins with a small SQL (SQLite) database table (`'Beer'`) of 17 beers. Each beer is represented with a unique ID number (`uid`), image file, name, brewery, style, ABV (Alcohol by Volume), IBU (International Bitterness Units), and calories value. It is worth acknowledging that this is a very simple list of attributes for describing a beer, and for the most part these attributes are objective (with "style" being the only possible exception). I will later reflect on some consequences of this data representation and suggest ways to extend the system for greater sophistication. For now, let us continue our description of the system.

Each beer is visualized in the UI as a Bootstrap "card". The top of each card is an image of the beer. Beneath the image, the beer's name is used as the card title. Beneath the card title is the card body featuring a list of key-value descriptors of the beer (i.e. style, ABV, IBUs, calories). At the bottom of the card body is a "Favorite" button that the user can click to add the beer to a list of their favorite beers. The home page of the site provides the view for the cards. The cards are limited to displaying three at a time, and pagination links beneath the cards are available to move back and forward through the available cards.

When the home page is first loaded, a JavaScript function (`loadBeers()`) is called to load the data required to construct the cards for the page. This function uses jQuery to pass an HTTP GET request along to the back end. The back end of the project utilizes the Flask (Python) web application framework. The GET request is sent to an application route that is responsible for querying the database and returning the data in JavaScript Object Notation (JSON). When the data is retrieved, it is sent to another JavaScript function (`buildCards()`) that constructs the cards to be displayed on the screen.

When the home page is first loaded, the initial query of the database will return all beers in the data table, as the user has not yet provided any criteria for filtering results (though again, only three will initially be displayed due to pagination). The next step in the implementation is to provide the user with efficient ways to filter through the data. To achieve this, we provide the user with a list of attributes by which they can filter their results. This list exists in a collapsible sidebar on the left-hand side of the screen. Each attribute that can be filtered has a drop-down toggle that, when expanded, reveals form fields for specifying desired values. Our simple implementation includes the ABV, IBUs, and calories attributes. For each attribute, the user can specify minimum and maximum value.

At the bottom of the sidebar is a "Submit" button. Once the user has specified their desired values in the form fields, they can click the submit button to update the list of beers displayed on the page. Clicking the submit button sends an HTTP POST request to refresh the page template, but this time it passes the desired parameters along to the database query. The new results are returned and the page template refreshes with the filtered results.

As is standard in modern web design, across the top of the page is a fixed-position navigation bar (navbar), and at the very left of the navbar is an image of the BrewFrog logo that, when clicked, redirects the user to the home page. If the user is on the home page, the next element to the right of the BrewFrog logo is a button to toggle the collapsible sidebar. This is important because the sidebar interface is only useful to the user while they are specifying their desired parameters for filtering results. While the user is just browsing through beers, the sidebar takes up unnecessary space in the view—especially on mobile devices—and thus the user will want to collapse it. On pages other than the home page, the sidebar will not appear, and thus neither will the button to toggle its visibility in the navbar.

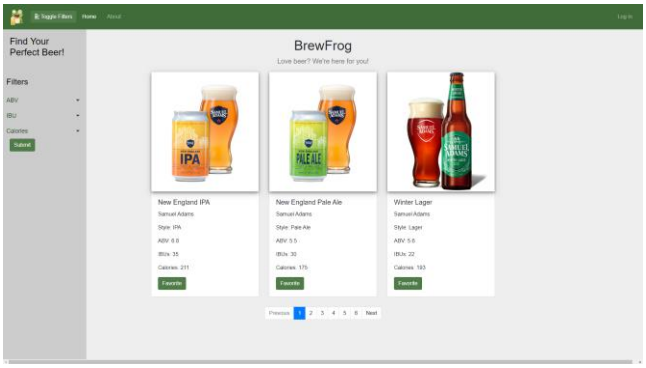

Figure 1: BrewFrog Home Page - Desktop View

The next two links along the left-hand side of the navbar are a "Home" link that directs to the homepage, and an "About" link that directs to a page that provides an informative description about BrewFrog. To the very right of the navbar is a "Login" link that directs the user to a page where they can log in to their account. If they do not have an account, they can click a "Create an account" link on the login page where they can create an account by providing their name, desired username, and password. On smaller viewports such as mobile devices, the "Home", "About", and "Login" links will not initially display; instead, a navbar toggler icon will display that the user can click to reveal a drop-down list that contains theses links.

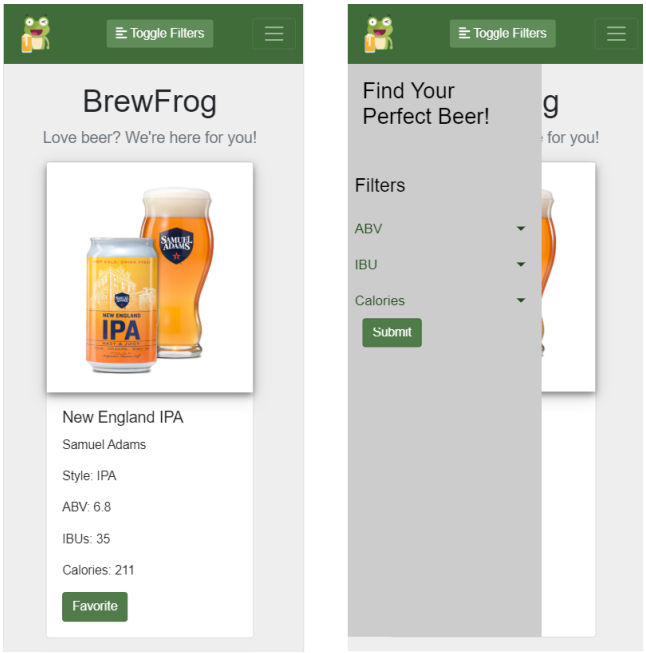

Figure 2: BrewFrog Home Page - Mobile View

The login and create user pages interact with a table of users in the database (`'Users'`). Users are represented with a unique ID number (`uid`), name, username, and encrypted password (for security, actual passwords are not stored in the database). The `UNIQUE` constraint is placed on the username so that two users cannot create an account with the same username. If the user attempts to create an account with an unavailable username or attempts to sign into an account with an incorrect combination of username and password, these pages will display an error message to inform the user of the issue.

Once a user is successfully logged in, the "Login" link in the navbar is replaced with a "My Account" dropdown. The dropdown contains two new links: a "My Favorites" link, which directs the user to a page that provides a view of the beers they have added to their favorites list, and a "Logout" link, which logs the user out of their session. As previously mentioned, each card on the homepage includes a "favorite" button. When the user is not logged in and clicks this button, they will be redirected to the login page. If the user is logged in, clicking this button will trigger a JavaScript function (`favoriteBeer()`) that effectively adds the beer to their list of favorites if it is not already in the list. More precisely, the function inserts the user's unique ID number (`user_id`) along with the unique ID number of the chosen beer (`beer_id`) into a third database table (`'Favorites'`). `User_id` and `beer_id` are both foreign keys referencing the unique ID numbers of the `Users` and `Beer` tables, and the combination of the two values has a Unique constraint to prevent the user from adding the same beer to their favorites more than once. When the user opens the "My Favorites" page, the `Favorites` table is queried returns and all rows where `user_ID` matches the user's `uid` number. Then, each `beer_id` returned is matched to the `uid` of a beer in the `Beer` table, and cards are built for each beer to be displayed in the view. These cards are identical to those on the home page, except that the "Favorite" button is replaced with an "Unfavorite" button. When the "Unfavorite" button is clicked, the appropriate row in the favorites table is deleted, and the page template is refreshed so the selected beer is removed from the favorites list.

This about sums up our overview of the BrewFrog system implementation. We now turn to a discussion of some of the benefits the system offers and share some ways in which the system could be meaningfully expanded for greater sophistication.

4 Discussion

As previously stated, the motivation behind BrewFrog is to assist consumers with differentiating products (specifically, beers). To this end, what I consider to be the most important aspect of the system is the sidebar that provides filters for the dataset. Earlier, we discussed the issue of users who may not be able to articulate their exact information need, or not knowing what information is required to satisfy their information need. The list of filters is critical in suggesting to the user ways in which they can narrow the list of available data to pinpoint precisely what they are looking for.

At present, the sidebar is admittedly rather unsophisticated. The only attributes that can currently be filtered are ABV, IBUs, and calories. These attributes are all quantifiable, and they all function similarly (i.e. they all offer "minimum" and "maximum" values). To extend the system's capabilities, a planned improvement for a future release is to allow users to filter by the "brewery" and "style" attributes as well.

Earlier we remarked that the attributes chosen to represent the beer were kept simple, and that they were mostly objective descriptions of the beer. It is worth noting that there are other attributes that would be important to a user who is seriously interested in differentiating beers would care to know, such as aroma, appearance, flavor, mouthfeel, and ingredients. These attributes (except for ingredients) are notably more subjective than our current list of attributes, and certain decisions would have to be made with care about how to offer filters based on these criteria.

First, decisions would have to be made about what values to offer as filters within each category. Take style, for example, since it already exists in our implementation. What style options do we want to offer? We want to be careful not to offer too many options such that the user becomes overwhelmed, but we also want to offer enough options so that a beer may accurately be described by an option. Should "Wheat Ale", "White Ale", "Brown Ale", and "Pale Ale" all be separate filters? Should they all be lumped into one "Ale" filter? Is "Pale Ale" distinct enough to warrant its own filter, whereas the other three can be lumped together? These are practical questions that need to be addressed. A practical solution might be to offer a general filter ("Ale"), that, when selected, offers a list of more specific sub-filters ("Wheat Ale", "White Ale", etc.).

A second decision that would have to be addressed is who decides what filters are associated with a beer. "Mouthfeel" may be a better example here than "Style", as surprisingly scientific beer style guides and taxonomies of styles in fact exist [11]. Who determines mouthfeel? An obvious choice would be the brewer, though another option would be to leave it to the vote of the system's users. The latter decision would of course entail more complexity in the system, though it does suggest another meaningful way of making the system more sophisticated, which would be to allow users to "rate" beers, and allowing users to filter through results based on user ratings.

In conclusion, while there are many ways that the BrewFrog system could be extended for greater sophistication, the system offers a reliable starting point for a user interface designed to achieve product differentiation. Offering filters for narrowing data is an efficient way to aid with navigability, as filters greatly assist users with achieving their information needs. The UI model provided here could easily be extended into product domains other than beer, only requiring the list of filter options be amended to the needs of the chosen product domain accordingly.

REFERENCES

[1] A. Read, A. Tarrell and A. Fruhling, "Exploring User Preference for the Dashboard Menu Design," 2009. *42nd Hawaii International Conference on System Sciences*, Big Island, HI, 2009, pp. 1-10. DOI: 10.1109/HICSS.2009.213

[2] Belkin, Nicholas. (1980). Anomalous States Of Knowledge As A Basis For Information Retrieval. *Canadian Journal of Information Science*. 133-143.

[3] Comber, T 1995, 'Building usable web pages: an HCI perspective', in A Ellis & R Debreceny (eds), *AusWeb95, innovation and diversity: the World Wide Web in Australia, proceedings of AusWeb95, the first Australian World Wide Web Conference*, Ballina, NSW, Norsearch Ltd, Lismore, NSW, pp. 119-124.

[4] De Souza, Flavio & Bevan, Nigel. (1990). The use of guidelines in menu interface design: Evaluation of a draft standard. *Proc. of INTERACT '90*. 435-440.

[5] E. K. Clemons, Guodong Gao and L. M. Hitt, "When Online Reviews Meet Hyperdifferentiation: A Study of Craft Beer Industry," *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06)*, Kauia, HI, USA, 2006, pp. 116c-116c. DOI: 10.1109/HICSS.2006.534

[6] Ed H. Chi, Adam Rosien, Gesara Supattanasiri, Amanda Williams, Christiaan Royer, Celia Chow, Erica Robles, Brinda Dalal, Julie Chen, and Steve Cousins. 2003. The bloodhound project: automating discovery of web usability issues using the InfoScentπ simulator. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '03)*. Association for Computing Machinery, New York, NY, USA, 505–512. DOI: 10.1145/642611.642699

[7] Jakob Nielsen. 1999. User interface directions for the Web. In *Communications of the ACM,* 42, 1 (January 1999), 65–72. DOI: .1145/291469.291470

[8] Jośe A. Borges, Israel Morales, and Néstor J. Rodríguez. 1996. Guidelines for designing usable World Wide Web pages. In *Conference Companion on Human Factors in Computing Systems (CHI '96)*. Association for Computing Machinery, New York, NY, USA, 277–278. DOI: 10.1145/257089.257320

[9] Melody Y. Ivory, Rashmi R. Sinha, and Marti A. Hearst. 2001. Empirically validated web page design metrics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '01)*. Association for Computing Machinery, New York, NY, USA, 53–60. DOI: 10.1145/365024.365035

[10] R. J. Kauffman, Ting Li, E. van Heck and P. Vervest, "Consumer Informedness and Hyperdifferentiation: An Empirical Test of the 'Trading Down' and 'Trading Out' Hypotheses," 2009 42nd Hawaii International Conference on System Sciences, Big Island, HI, 2009, pp. 1-10. DOI: 10.1109/HICSS.2009.130

[11] 2020 Brewers Association Beer Style Guidelines. (2020, February 21). Retrieved from https://s3-us-west-2.amazonaws.com/brewersassoc/wp-content/uploads/2020/03/BA-beer-style-guidelines-2020.pdf