CS449- Human Computer Interaction
Assignment-1 Design Diary
Uğur Öztunç
28176

HCI Design Problems on Spotify Desktop application:

The design problem that I consistently encounter in the Spotify Desktop application belong to its interface and functionality. My primary task within this interactive application is to enjoy music while simultaneously keeping track of my friends' music activity, also accessing detailed information about the currently playing song and its artist. The issue I face is directly related to the interfaces associated with these features, which underwent significant changes in a recent update. Before continuing, if you are not familiar with the Spotify Desktop, it is recommended that you familiarize yourself with the interface components that I will talk about by looking at the screenshots in the Figma link that I provided at the very end of this document. Prior to this update, the application was behaving in the following manner: Upon clicking the 'expand' button, located on small album art picture on the bottom left of the screen, the album art would expand on the same location, making it larger while still positioned at the bottom left of the screen. Furthermore, toggling the 'Friend Activity' button on the top right, to the left of the user profile icon, reveals a new pane on the right of the screen, allows users to see his/her friends' listening activity. With the update, a new feature 'Now Playing View' was introduced. This feature is presented as a pane that opens on the right side of the screen, just like the friend activity pane, offering details about the current song and artist, along with the expanded album art of the current track. To open the 'Now Playing View' pane, a toggleable button is located at the bottom right of the screen.

The problem arises in the continued existence of the 'expand' button on the bottom left. Instead of the old behaviour I explained, clicking this button now opens the 'Now Playing View' pane. The first issue is the misleading nature of the 'expand' button, which confuses users who expect the album art to enlarge, as it did in previous versions. While the desired functionality of viewing a larger album art retained within the 'Now Playing View' pane, even a user who is entirely new to the interface would expect an action to occur in the area of the screen where the 'expand' button is located, since the button is in shape of up arrow. This can be associated with the concepts of "informative feedback" discussed by Ben Shneiderman in "Designing the User Interface: Strategies for Effective Human-Computer Interaction", which emphasizes the importance of providing clear and immediate feedback to users regarding the results of actions (2016). In this case, the fact that clicking a button labelled 'expand' and shaped as up arrow on the bottom left of the screen results in the opening of a pane on the right side of the screen shows a clear design problem, as it is quite opposite to what user is expecting as feedback.

The second problem relates to the conflict between the 'Friend Activity' and 'Now Playing View' panes. When 'Friend Activity' is opened on the right, subsequently opening the 'Now Playing View' pane causes it to replace the 'Friend Activity' pane, or vice-versa. This behaviour is expected and normal; however, an issue arises when user closes the lastly opened pane, resulting in the unexpected closure of the entire right panel. This issue can be related to Shneiderman's principle of "Prevent Errors," where it's essential to design the system in a way that minimizes the occurrence of errors, whether they are

major or minor (2016). In this case, the design leads to unexpected actions that can be considered errors from the user's perspective.

Another principle stated under the title of “Golden Rules of Interface Design” is about allowing users to easily reverse their actions. Important motivation for doing such is “encouraging exploration of unfamiliar options”, as Shneiderman suggests (2016). At this point, the third issue concerns the placement of the toggleable buttons for opening ‘Friend Activity’ and ‘Now Playing View’ panes. Typically, users expect to find buttons that toggle related features in similar locations on the interface. However, in this case, the buttons are not placed in similar locations: 'Friend Activity' button is located at the top right, while the 'Now Playing View' toggle button is positioned at the bottom right. This discrepancy in button placements can lead user to get confused about interface and, also might frustrate user when he/she would like to switch panes rapidly, which clearly violates the principle of “Permit easy reversal of actions”, by rendering to explore different options difficult and not intuitive.

To illustrate the impact these issues, I can describe a common use scenario that I experience and suffer from every day. While enjoying music, I open ‘Friend Activity’ pane using the button at the top right to see friends’ listening activity. When I want to view album art larger, following my familiarity with the old versions, I click the small ‘expand’ button, unintentionally opening the ‘Now Playing View’ pane on the right side of the screen. Immediately, I attempt to close the pane by clicking the actual ‘Now Playing View’ button, since it replaced ‘Friend Activity’ pane. However, this action inadvertently results in the closure of the 'Friend Activity' pane it replaced. As a result, I need to move my mouse to the very top right of the screen and click on the 'Friend Activity' button once more to undo the action. These repeated actions to view a larger album art, and friends’ activity leads me to frustration about interface. The nature of these issues in causing confusion, frustration, and unexpected behaviour directly aligns with HCI principles of user feedback, error prevention, and intuitive design, highlighting why these are HCI problems that should be addressed.

Possible Solutions:

In order to eliminate this design problem, two solutions can be developed. The first solution addresses the issue of confusing pane closure behaviour. To align with Shneiderman's (2016) principles of "Prevent Errors" and "Design Dialogs to Yield Closure," the panes would be stacked upon each other as they are opened. When a user closes one pane, the last opened pane must remain visible after the closure. This change ensures that the user can quickly switch between panes and close them while others remain open. By doing so, we reduce the likelihood of unexpected closures, in accordance with the principles of minimizing errors and providing clear user feedback.

Alongside this solution, the second possible solution requires repositioning interface components to better satisfy Shneiderman's (2016) principle of "Strive for Consistency." In this redesign, the 'expand' button, which is a source of confusion, would be entirely removed. The 'Friend Activity' button, positioned at the top right, would be relocated to just left of the 'Now Playing View' button. This modification not only adheres to the principle of consistency by placing buttons that result in similar behaviours in similar places, but it also aligns with the principle of "Permit Easy Reversal of Actions.", as user will be able to switch between panes without too much effort, especially when they accidentally open the wrong panel. Users, both novice and experienced, will find it easier to switch between panes without excessive effort and confusion, and ultimately these changes will lead a more intuitive and user-friendly experience on Spotify Desktop interface. I have provided an example interface which can be accessible from the Figma link below.

Figma Links

Spotify Desktop interface and the problem:

https://www.figma.com/proto/MIFwFQiVnCiGT9QY8u3IhI/CS449_Design_Diary_1?type=design&node-id=4-74&t=PVt6OpsBFVeBMkHq-0&scaling=min-zoom&page-id=0%3A1

Improved version for 2nd solution:

https://www.figma.com/proto/VGCKHcH3g48q7DBxOfSvC9/CS449_Design_Diary_2?type=design&node-id=1-3&t=PVt6OpsBFVeBMkHq-0&scaling=min-zoom&page-id=0%3A1

References

Shneiderman, B. et.al. (2016). Guidelines, Principles and Theories. pp. 81-120. Designing the User Interface: Strategies for Effective Human-Computer Interaction, 6th Edition