Háskólinn í Reykjavík

Greining og hönnun hugbúnaðar
Kennari: Marta Kristín Lárusdóttir

Greining og hönnun hugbúnaðar
Problem Solving Assignment 5

22.09.2024
Haustönn

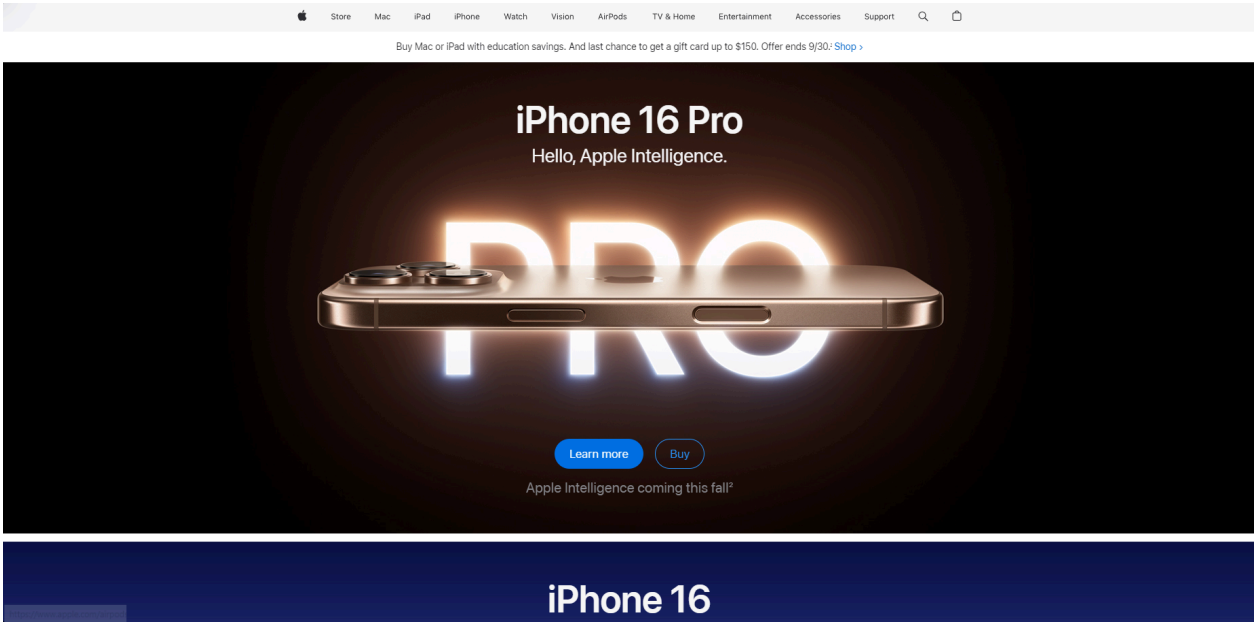
Bjarni Hrafnkelsson

Visibility

The first design guideline we're going to explore is Visibility. Visibility is about showing the user what is available within the software. It means to have the most used features easily accessible. Think about why the user is on a particular page and show him the options he would like. If necessary options or information is hidden can lead to confusion for the user about what he should do.

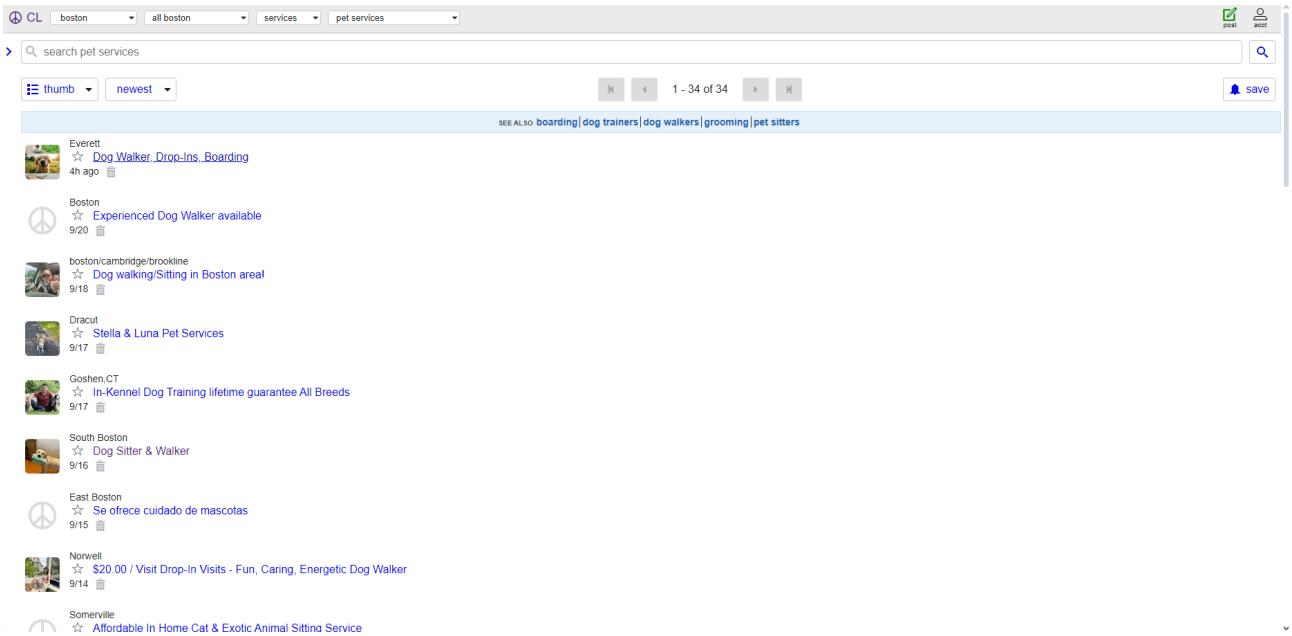
Apple

A website that does this well is the Apple.com website. They have recognized that users that visit their website are most likely looking for their most recent product. Consequently, their most recent product is the first thing on their page with an option to buy. Going further down there are other products. If you´re looking for a specific option, they´re all displayed at the bar at the top, although not very flattering, you can find every product and their features without scrolling or thinking much about it.

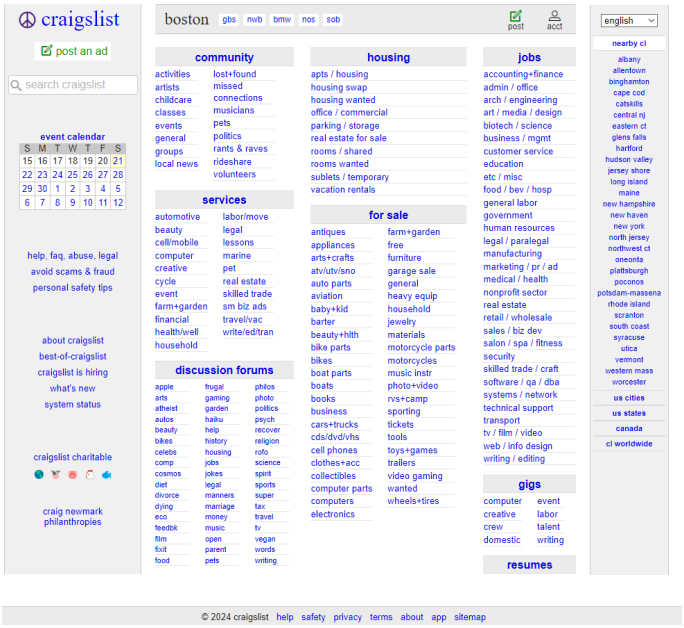


Craigslist

Craigslist is an example of a website that does not implement this very well. To start off, the website itself is not very appealing. It is only made of links that are blue and don´t fit into the theme of the website. The front page does not make it easy for the user to navigate. It has a lot of text and although they have split them into categories, they are very broad or ultra specific. Making the website a lot to read at the start.



When going into a category, very little information is shown. You can see the title that people set and a number that is not explained, f.x. 18/20, 4/9.
Users browsing for f.x. Possible dog sitters will have to click on every link and browse every profile since there is little to none information without going to every link.
Additionally, for people looking for something that is not on the front page, it will be very tricky if you don't know the exact keywords to be looking for.

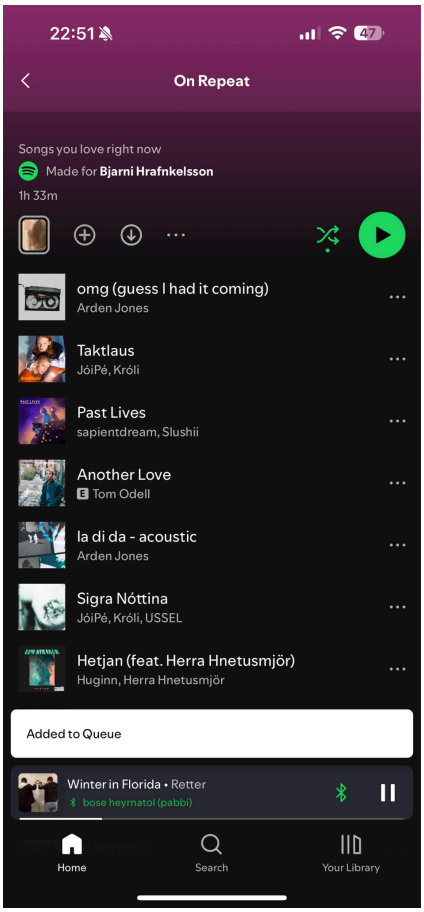


Feedback

Feedback is letting the user know when their input is received. Accomplishing that can be as simple as a button changing color when pressed. When the user doesn't know when their inputs have been received can lead to them pressing the same button multiple times, unsure of if their request has gone through.

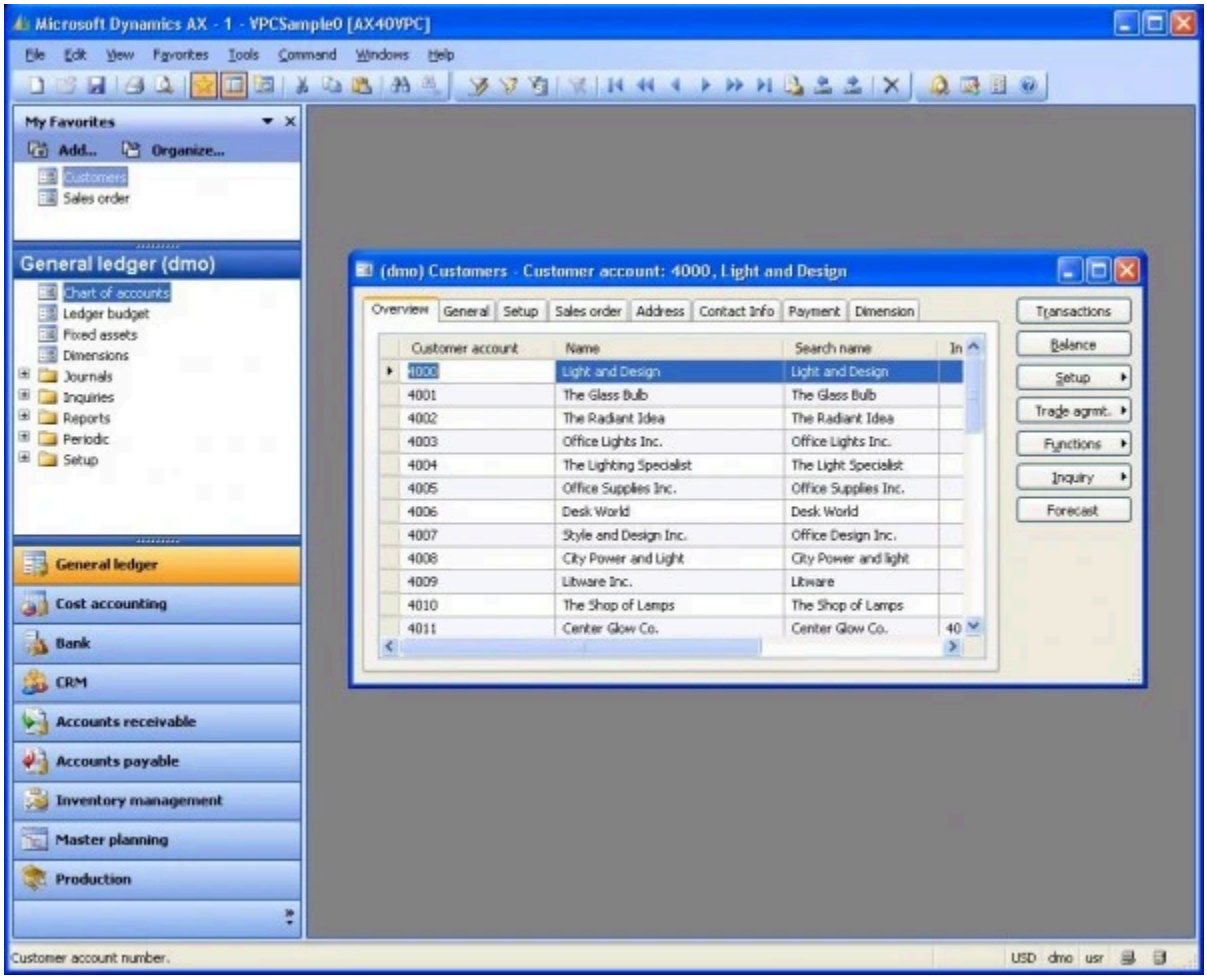
Spotify

One example of a good feedback design is Spotify. They've implemented this very well. When skipping a song, the name of the song will swipe to the left as you slide across the screen, signifying that you're skipping that song and the name of the song that will play is shown, before you take your finger off the screen. Furthermore, when adding a song to the queue, you will get a little notification at the bottom of your screen saying that the song has been added to the queue. Everything inside that app makes it clear to you that it's receiving your input.



Axapta

Axapta is a program that has zero to none feedback in their design. I have experience with this program through work and it has been a pain to use. But for the feedback section, it is very minimal. Specifically, when moving items between stores, there is no confirmation of when you have submitted your transfer request, and when it finishes the request, which can take a while, it does not tell you in any way. The only way to know is to see the transfer list and see if your request is still there.



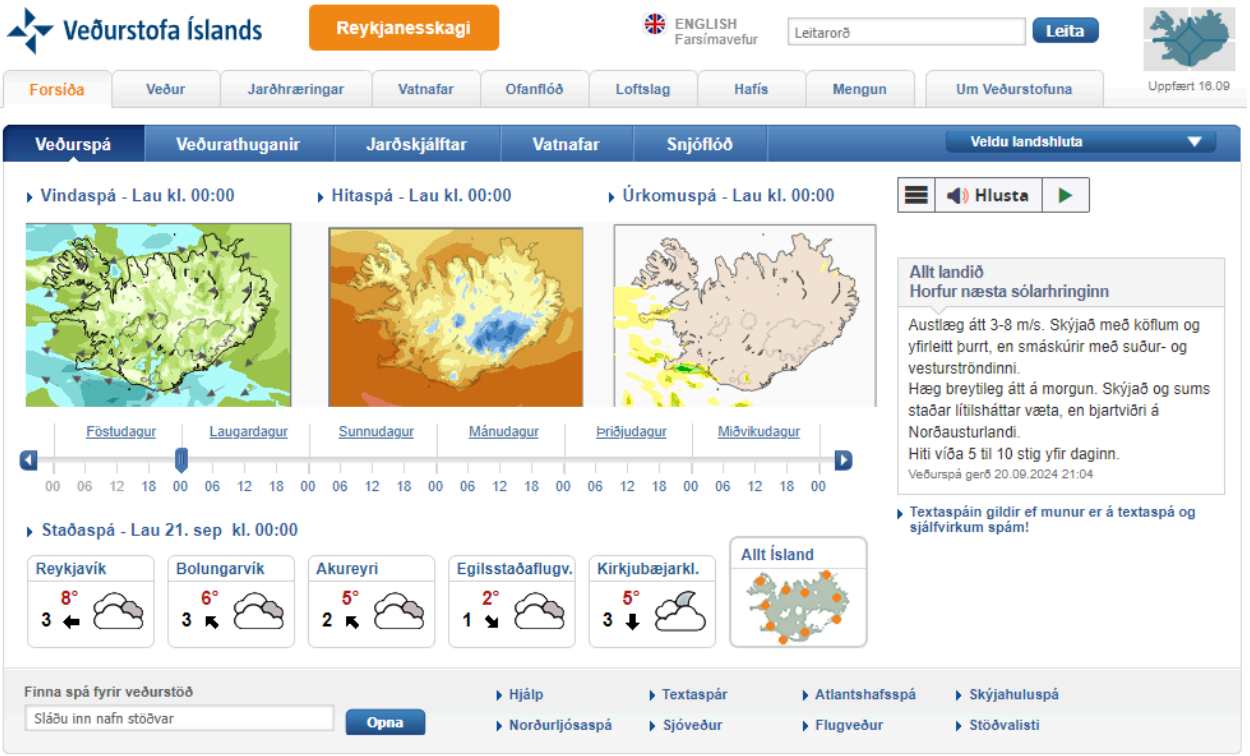
Problems that could arise from that is that because there is no confirmation that you finished processing your transfer, it happens that we do it twice or we don't know if it has already been done or not.

Constraints

Constraints are options that are not given to a user to prevent unwanted mistakes. Wanting to set an appointment and having a constraint that does not allow you to book them in the past, is an example of a good constraint.

Vedur.is

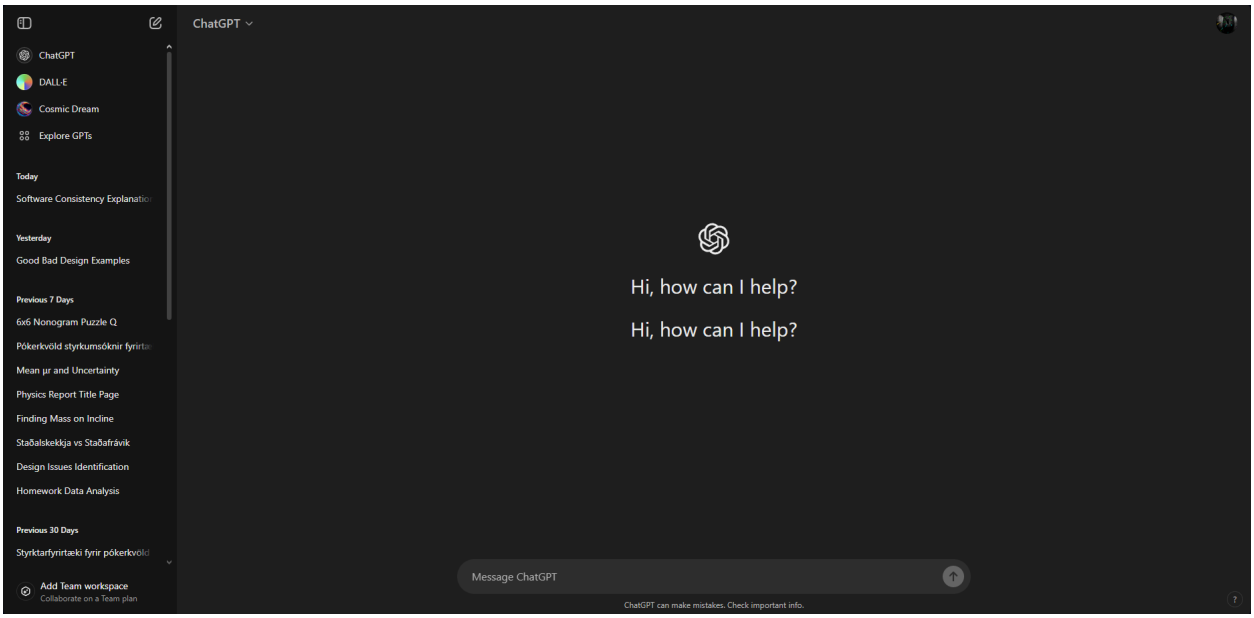
Vedur.is does this well. They can not predict the weather far into the future so instead of allowing the user to insert a date, they have made them choose a pre-selected date that they have information on. That way it is shown to the user what days are available in their database and prevents mistakes from happening.



ChatGPT

Although not in the traditional sense of a constraint, I wanted to include ChatGPT. But usually an example of a bad constraint is allowing the user to do something that will lead to a mistake or an incorrect result. ChatGPT is a different example of a bad implementation of a constraint. When having an artificial intelligence for the general public it is important to make sure there are constraints to what you can ask the bot.

A popular example of how not having constraints on artificial intelligence can lead to disastrous outcomes is showing a picture of available material and asking him how you can make a bomb with those materials.

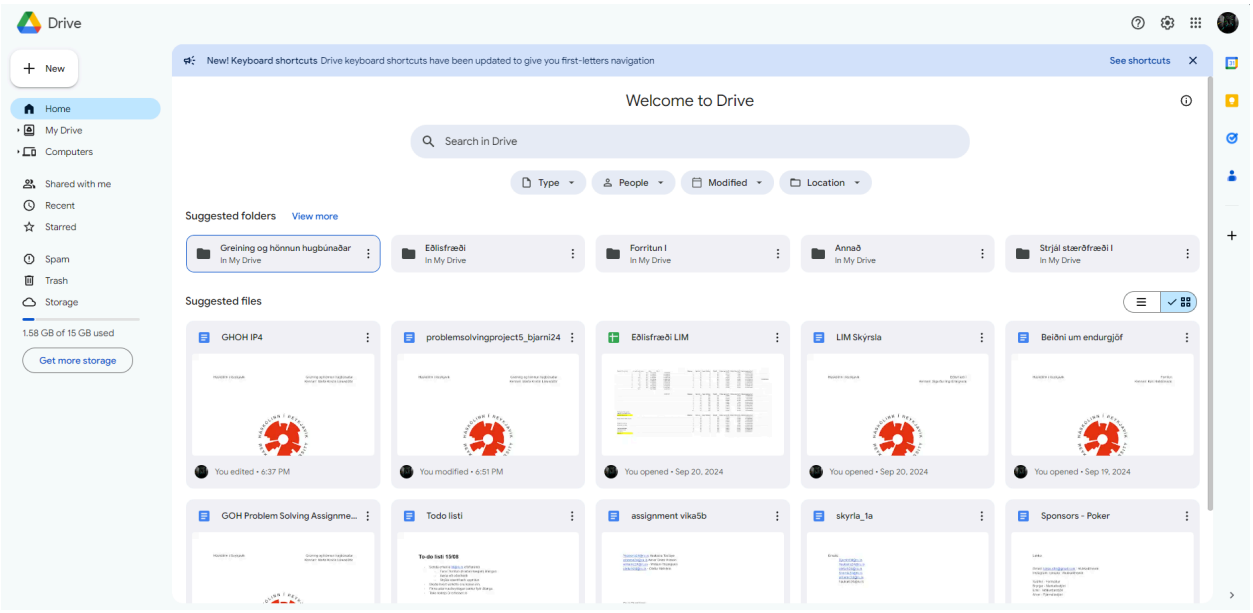


Bypassing the constraints filter on this chat bot has proven to be far too simple and needs some time to improve. Not having this filter can lead to people not trusting their children or kids to use the chatbots.

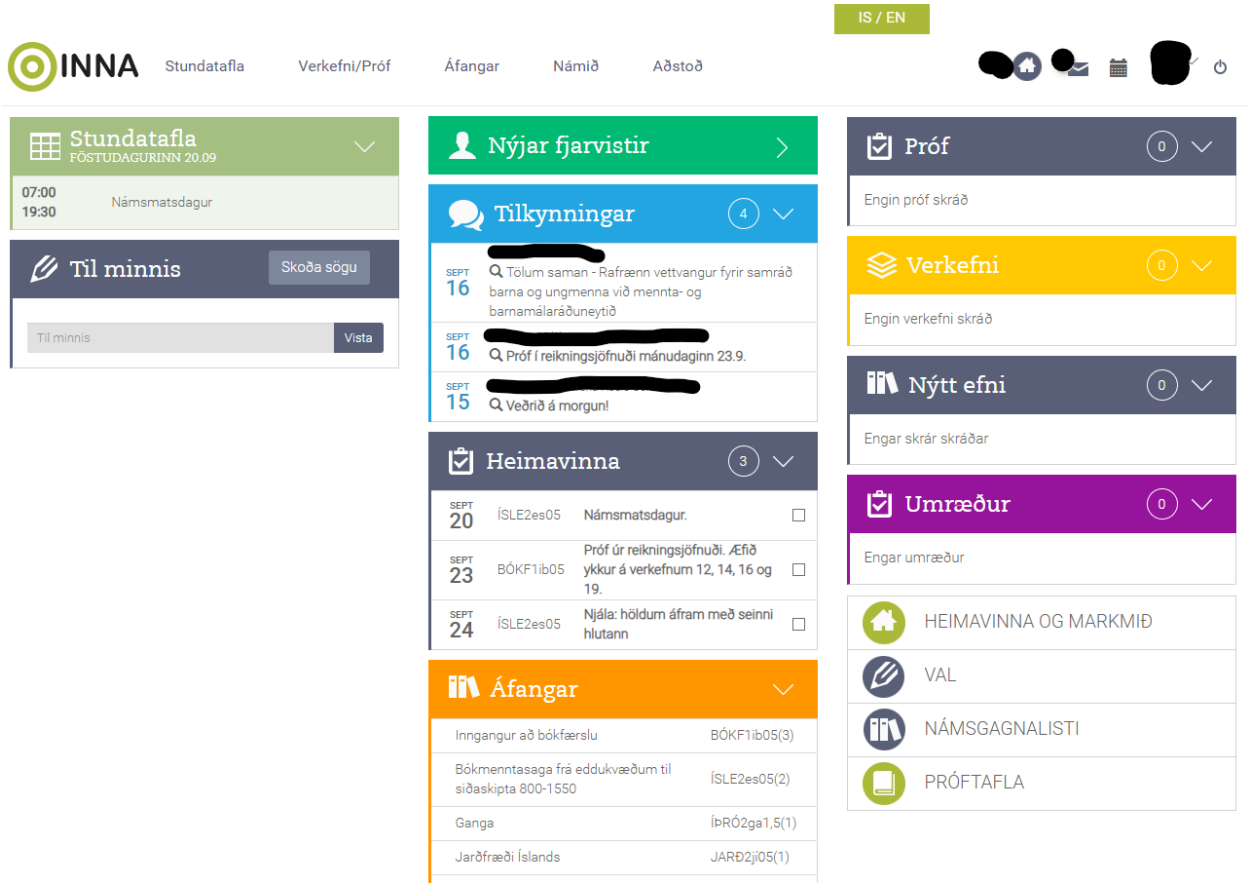
Consistency

Having a seamless experience within the software, where the user feels familiar with the app from the start. Consistency is a crucial goal to accomplish. Without it, the user needs to learn how to do everything. Consistency is making it easy to navigate and communicate with the user how to do tasks without showing them and changing the execution as little as possible.

Google Drive accomplishes this very well. With no prior experience, a user is able to navigate through the software with ease. All tasks are straightforward to use and the user never has to figure out how to accomplish certain tasks.



Inna is a website that does not follow this. The colors are not pleasing to look at, all information is kind of scattered around the website. Additionally, finding information requires the user to explore the app and find the information themselves. Even worse, the titles are very misleading. Quite often the information in a tab does not withhold the information users are expecting.



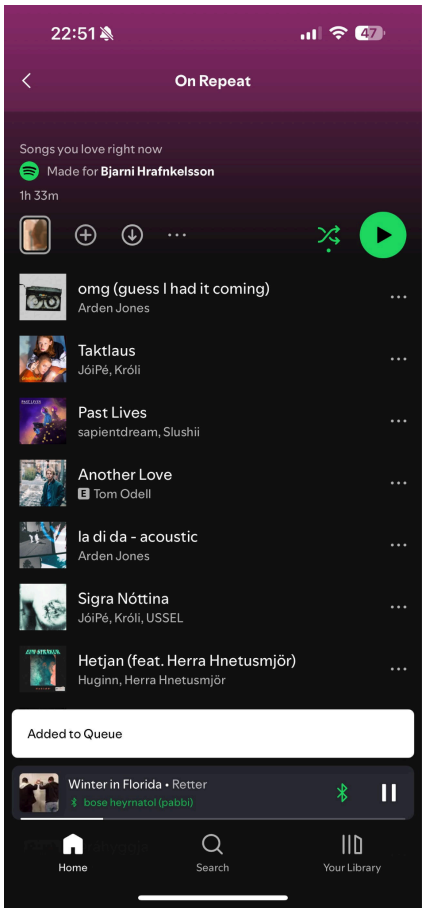
A substantial amount of options are not used, f.x. Heimavinna og Markmið and Próftafla. Those tabs are empty and not used. Furthermore, looking for your grades is not in the Verkefni/Próf(Assignments/Tests) where you would naturally think it is, but it is in Áfangar(Classes).

Affordance

Affordance is displaying to the user what each and every action and button does. Having clear icons that makes it clear to the user what their action does. Doing so can be accomplished through icons, or somehow displaying to the user that a certain action is available.

Spotify

Spotify is an example of software that does everything well. Spotify has a good implementation of affordance. All symbols have a clear purpose that is easy to recognize. The options at the bottom have a clear purpose, f.x. The home icon to go home, search symbol to search, etc.



Lings Cars

Lings Cars is a website I found by searching for poorly designed websites. Lings Cars's affordance is very poorly designed. In the bars on the left and right side, there is no distinction between clickable sections to go to and pictures he puts for dramatic use.

Users navigating the website might have some trouble knowing what they can do or knowing how to accomplish certain tasks