Idea 6: Detecting Dark Patterns

Recently a team of researchers (including UChicago's own Marshini Chetty) tried to detect dark patterns at scale. Take some of their ideas and implement your own detector of some sort of dark patterns from scratch; you are not permitted to reuse any code they or others have published for detecting dark patterns, though you of course may reuse code per the course policies for auxiliary aspects of this task (e.g., parsing webpages). Validate your detector on real webpages. Your tool may run locally (e.g., Python code running Selenium takes as input a URL and tries to detect dark patterns on that URL), a browser extension (trying to detect dark patterns on the current page), or whatever other architecture you prefer.

Part 1: Research to begin the project

For the first part of the project, I began researching ways to categorize dark patterns as well as establishing what I should look out for as I begin my investigation into the various types of generally accepted Dark Pattern categorizations that exist in the realm of User Interface and User Experience (UI/UX) design.

A phrase I noticed early on in my readings was “deceptive design” and how this phenomenon plays into the Dark Patterns a user sees and interacts with. There are many aspects of Dark Patterns and deceptive design that are starting to gain large amount of political attraction such that law makers have begun to consider Dark Patterns in their decision making in attempts to protect the general computer using public. Since Dark Patterns are considered to be aspects and designs on a webpage that trick the user into doing things (signing up for lists, paying for products) that they do not intend to, the Dark Patterns are deemed manipulative and detrimental design flaws as some tend to be unintentional.

A clear example can be seen in Trump’s fundraising campaign as discussed in class. This instance included Stacy Blatt, a 63-year-old battling cancer, who donated $500 which quickly multiplied into $3,000 over the course of the next 30 days. This is just one of many examples of the detriment that Dark Patterns can cause as “contributors had to wade through a fine-print disclaimer and manually uncheck a box to opt out” (Goldmacher). This is a prime example of both a Dark Pattern and how detrimental they can be to any given user. In particular, this example goes to show how default versus opt in availability impacts the way that users interact with a platform and the repercussions that arise from making a decision to opt the user in automatically versus having them opt in themselves. Additionally, designs that require the user to jump through unnecessarily difficult hoops also play into the idea of deceptive design, a fundamental phenomenon in the creation and implementation of Dark Patterns.

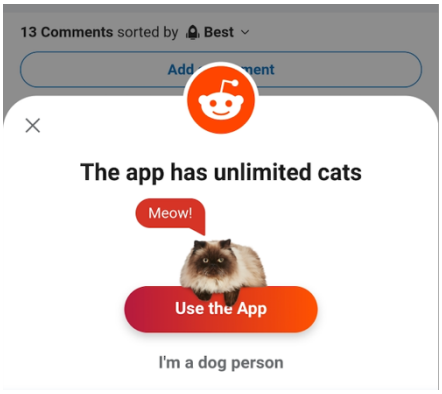

Figure 1

Many of the top companies that implement deceptive design include but are not limited to Google, Facebook, Reddit, Amazon, New York Times, and Venmo. In each of these cases we can see how Dark Patterns can range from merely annoying the users to being financially devastating.

Reddit uses light-hearted, fun Dark Patterns as we can see how, in declining to use the app, the user has to accept that, in the eyes of Reddit, they are a “dog person”. Additionally, it is clear how this pop-up nudges

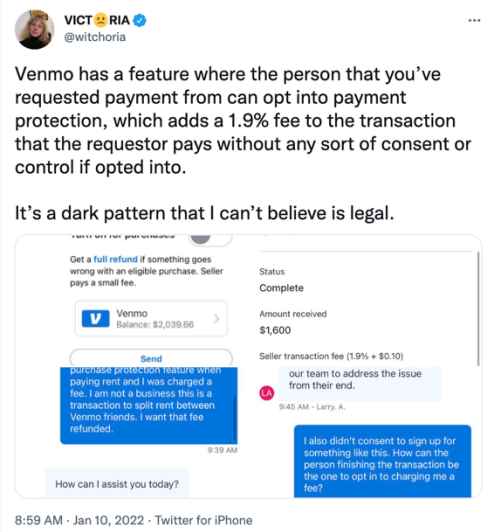

Figure 2

the user to be inclined to pick the option that is in bright red with a fluffy cat laying on it. This example is clearly low potential for lasting consequences when compared to the example of the Trump campaign donation subscription.

On another hand, we see more harmful monetary dark patterns on platforms that are still considered to be social media. In the case seen in Figure 2, Venmo includes an option for the user to opt in (hence the default is to be opted out) of a 1.9% fee for payment protection. This is particularly devastating for those who transfer large amounts of money through the application, those who use Venmo for small businesses, etc. and do not anticipate this charge to be deducted.

Lastly, a commonly seen dark pattern (and perhaps the most prevalent in a user's website usage is seen in the ability to opt in and out of Cookies and other types of cross-platform tracking, as seen in Figure 3. Companies make opting out of Cookies hard or even impossible, while also setting the default to opting in. This provides issues as a user may not know the implications of what they are opting in to, they might not want Cookies to be used in the first place, or they might not have the patience, time, or knowledge to turn off such tracking once they are on. This is a prime example of multiple Dark Patterns being combine to dupe the user.

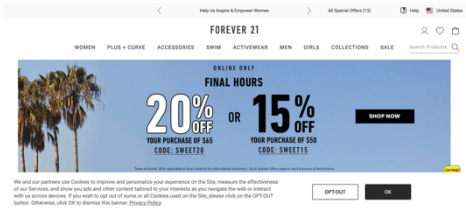

Figure 3

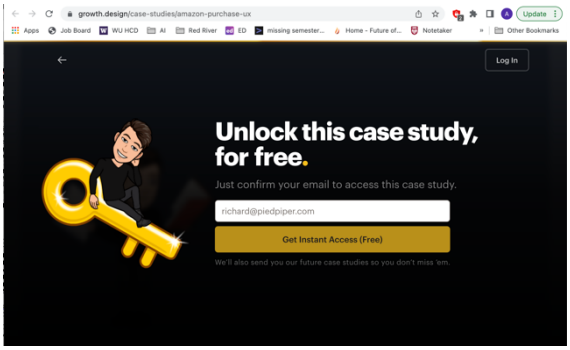

Figure 4

In my own research to learn more about the prevalence and detriment that Dark Patterns pose to website usability and company short term gain, I encountered one on a website

The WireWheel article specified in the additional resources section at the end of the Write-Up discusses 12 types of Dark Patterns commonly seen on the front end of websites. These 12 types of deceptive design categories provide the jumping board for my project, as I will work to use HTML and web-scrapin g to detect and categorize certain key words and HTML attributes that may signal a dark pattern. The 12 types of Dark Patterns, as put by Harry Brignull, a UX designer, are the following:

1. **Trick Questions:** The user, when filling out a form, is mislead or tricked in to giving a response that might not accurately reflect their true answer (asking one thing, but meaning another)
2. **Sneak into Basket:** Seen on shopping platforms, the website will sneak something into the basket of the user without their knowledge
3. **Roach Motel:** A website makes it easy for the user to enter a situation but hard for them to leave

4. **Privacy Zuckering:** Similar to trick questions: the user is tricked into sharing more information about themselves than is necessary or intended
5. **Price Comparison Prevention:** A given retail platform makes it hard for the user to compare prices of items
6. **Misdirection:** The webpage's design is meant to divert the attention from one thing to another
7. **Hidden Costs:** Last stages of a check out process contain unexpected charges
8. **Bait and Switch:** User takes action to do one thing, but another happens as a result
9. **Confirmshaming:** User is guilted into certain options via wording and shaming
10. **Disguised Ads:** Advertisements that are disguised to get a given user to interact with it
11. **Forced Continuity:** When something like a free trial comes to an end, charges begin to appear without notice or user opt in abilities
12. **Friend Spam:** An application or website uses contacts and social media to send unnecessary material to a given user's connections

For the sake of scale readability, I condensed by "binning" for each page to one of the 5 categories: 1. Confirmshaming, 2. Friend Spam, 3. Privacy Zuckering, 4. Opt-In/Opt-Out Checkboxes (defaults and text associated with text options), and 5. Urgency (to include potentially illegal indicators of a time restriction). From the HTML scraped, alone, I believe these five categories can be detected and adequately binned by analyzing both the available content as well as the text found within the HTML text files.

While there are various other accounts of Dark Patterns occurring on a given webpage, these are the generally accepted classifications many designers keep in mind when working to develop websites and applications. In many cases, Dark Patterns run legality risks as 15 U.S. Code 54 discusses the penalties for false advertisements that may pressure a user into making decisions they might not have otherwise made in the absence of said Dark Patterns. Additionally, in the European Union, "refusing service when the user does not consent to non-essential data sharing is illegal" (bbarnett). This comment was made in reference to Google Maps requiring that WiFi scanning be enabled to use navigation which has been shown on multiple occasions to not only be necessary for the functionality of the application but breaks Google's developer agreement.

Lastly, a field wide ethical debate runs rampant as many designers have argued that *all* design is manipulation explaining that manipulation is just a natural feature of design. The moral and ethical implications arise with this discussion as many designers believe that, with training founded in ethics and morality, design can be unbiased and avoid user deception.

Part 2: Data collection through web scraping

Regarding the data collection process, I wanted to focus on the 100 most visited websites currently online. This, unfortunately, did not pan out like I wanted because my computer kept over heating and crashing, so I just took what I could get and did the 50 most visited websites online which *also* caused my computer to overheat. From there, I went with 25 websites, which seemed to be alright for my

```
25
['google.com', 'youtube.com', 'facebook.com', 'baidu.com', 'yahoo.com', 'instagram.com', 'wik
ipedia.org', 'twitter.com', 'whatsapp.com', 'qq.com', 'bing.com', 'linkedin.com', 'reddit.co
m', 'yandex.ru', 'live.com', 'zhihu.com', 'zoom.us', 'microsoft.com', 'vk.com', 'github.com',
'office.com', 'xvideos.com', 'csdn.net', 'tiktok.com', 'taobao.com']
```

Figure 5

computer's running capabilities! To know which websites to use, I checked that the website was both up online and useable. From the most frequently visited websites, the top 25 most used websites that are still online and accessible are seen in Figure 4. From this list, I wrote said list to urls.txt to ensure that I wouldn't have to rerun the code that checks to see whether or not the website meets the criteria for analysis since it was intensive for my computer. I then used this list of "testers" to pull the HTML from the webpages, where I also store them in text files, all of which can be viewed in HTML_Text.

Part 3: Dark pattern weight assignments

My first step was analyzing the HTML text for text that is associated with Anthropologie.com. I proceeded to use Anthropologie.com to begin my analysis since I knew this website would most likely have Dark Patterns that I, as someone who is not well versed in the UX design space, would be able to notice both visually on the webpage and in the HTML text that would be beautiful soupified. I began by looking at the actual HTML from the website to see what HTML elements I would need to pull from the documentation for analysis. From scanning the HTML printed, I concluded that looking for <a>, <p>, <img>, <button>, <form>, <input> , and <label> elements would be the most helpful in my analysis for a foundation to start analyzing the text for all five categories as seen in Figures 6 - 9.

```
<img alt="Shop an extra 40% off sale items" class="o-pwa-image__img c-pwa-image-component__img" he
ight="350" src="//images.ctfassets.net/5de70he6op10/6kduFzuWQqoIG3jJei9K6S/a3fd85debe631e8c54e4098ec5ee21f2/052722_
40offSOS_HPG_Secondary_SS_2.jpg?w=1125&amp;q=80&amp;fm=jpg&amp;fl=progressive" width="1125"/>
</picture>
```

Figure 6

```
<li class="c-pwa-header-navigation__item" data-nav-slug="sale-all">
 <div class="c-pwa-header-navigation__toggle">
  <!-- -->
  <a class="c-pwa-header-navigation__link js-nav-link c-pwa-header-navigation_
_link--sale c-pwa-link c-pwa-link--client" data-qa-header-l1-children="true" href="/sale-al
l" id="l1-sale-all" tabindex="0">
   Extra 40% Off Sale
  </a>
 </div>
 <!-- -->
 <!-- -->
</li>
```

Figure 7

My first step was to consider the phrases associated with Confirmshaming. Based on an article from BuiltIn, I sought out to test strings of text found in the HTML files for phrases such as "No thanks, I like full price," "No thanks, I'm not into savings," "Don't like, "Don't want", etc. I also included words that would indicate persuading or drawing the user to make a conclusive choice such as "don't", "want", "yes", "no", and "thank". These words are traditionally associated with guilting the user to pick a specific answer that the designer or website wants you to.

```
<a class="c-pwa-header-navigation__link js-nav-link c-pwa-header-navigation__link--sale c-pwa-link c
-pwa-link--client" data-qa-header-l1-children="true" href="/sale-all" id="l1-sale-all" tabindex="0">
 Extra 40% Off Sale
</a>
```

Figure 8

My second set of phrases to analyze are correlated to friend spamming. I sought out to look for <form>, <label> , <button>, <a> , and (most importantly), <input> elements that would indicate a space for the user to input various types of personal information or confirmation that would allow the website to obtain access to a user's

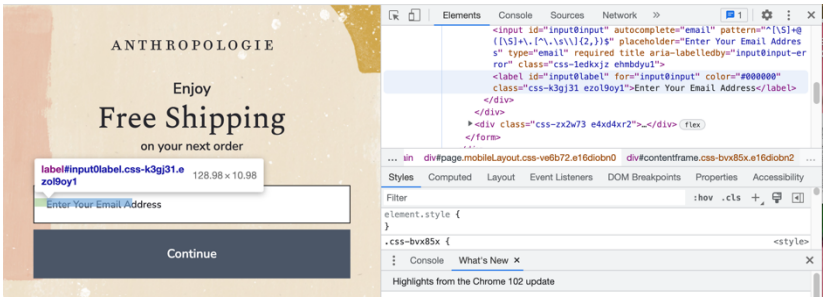

Figure 9

network. An example with Anthropologie.com would be Figure 9 as this information regarding email input from the user would be unnecessary or have buzz words that would place the inputs required in the "unnecessary" information collection category. From here, I decided to implement the categorization on both the main page for Anthropologie.com as well as this email collection form page. I was unable to use previously written code for this, however, as the email form seems specific for each user. Instead, I inspected either page and copied the code into .txt documents which can be viewed in the Anthropology_Example code folder and read in either document name to use the HTML information in the various functions which take a text input (all of the HTML text from the files, in this case which is extracted from each respective file using a helper function).

My third function incorporates searching all accounts of HTML tags <a>, <p>, <label>, <form>, and <input> as these are the tags, I noticed were typically associated with asking a user for personal information such as their name, address, email, etc. These are also adjacent to the quotes/phrases/strings I search for within each element that is pulled using the .search_all functionality that comes with the Beautiful Soup API.

Next, I look for indicators of checkboxes that could be misleading or defaulted without the user's consent. I do so by searching the HTML text for each URL for <input>, <form>, and <label> since these are the places that a user will be filling out a form or interacting with a website beyond the traditional clicking and searching functionality. I search for both "checkbox" and "radio" as these are words in HTML that can signify a form value where the user has a binary "checked" or "unchecked" value.

Lastly, my *urgency* helper function incorporates searching the HTML for words that indicate a pressure being put on the user to decide or creating a (false) sense of restriction to either time or quantity in some way. I particularly look for string elements correlating to time such as days left for a promotion, words that indicate urgency, and limited availability of a product (when considering shopping platforms).

```
anthropologie main page stats:
confirmshaming:
475

friendspam:
186

privacy zuckering:
324

checkbox opting:
711

urgency signalling:
306


anthropologie email form stats:
confirmshaming:
2

friendspam:
6
privacy zuckering:
9

checkbox opting:
1

urgency signalling:
2
```

Figure 10

For each URL that has been scraped according to the values found in urls.txt, I store the number of Dark Patterns in the form of a CSV file, all of which can be viewed in the Number_DP file that contain the outputs from each helperfunction previously referenced to count the number of Dark Pattern occurances for each category. The values that are saved in each CSV file are those referenced Figure 10.

From each of these values, I conferred with friends to figure out a general guideline for establishing a scaling value to associate with each type of Dark Pattern found in the HTML text files of each website. From this point, I decided on a standardize scale for each one of the Dark Patterns based on ranking of the "badness" or negative consequence on the user. The ranking system is as follows (and can be seen in the "scale" function in Dark_Pattern_Weighing.ipynb) with 5 being the "worst" and 1 being the "least worst": 1. Urgency, 2. Confirm Shaming, 3. Checkbox defaults and Opting In Versus Opting Out, 4. Privacy Zuckering, and 5. Friend Spam.

From this scaling value, I multiply each one of these ranking place values to the number of times that said Dark Pattern Category happens. These values being read in are from a daataframe that contains the counts from each Dark Pattern category according to the

corresponding CSV file that can be referenced in the Number_DP folder. Then, I use said scale value and find the average of the Dark Pattern Scaled values to find the overall "badness" of each individual URL. These "badness" values can be seen in the scaled.txt files.

From these "badness" values, I wanted a way to compare them among the given urls, so, at the end of the scaled.txt document, there is a "total" value that can be seen, which is the total summation of all of the "badness" values of the scaled Dark Pattern values for each URL. This summation will allow us the find the percentage that each website contributes to the whole and, if one website is worse than another, we should be able to see that reflected in the outputted percent values. In scaled_proportion in the Dark_Pattern_Weighting.ipynb Python code file, the function helps to determine the percentage of "badness" that a given website contributes to the whole. To better understand the outputted data, the mean provides a meaningful starting point to best understanding the data. These percentages allow us to best understand the data and each percent and corresponding website are saved to final_scores.txt. This file includes the "scores" of the Dark Patterned Badness of each of the specified websites.

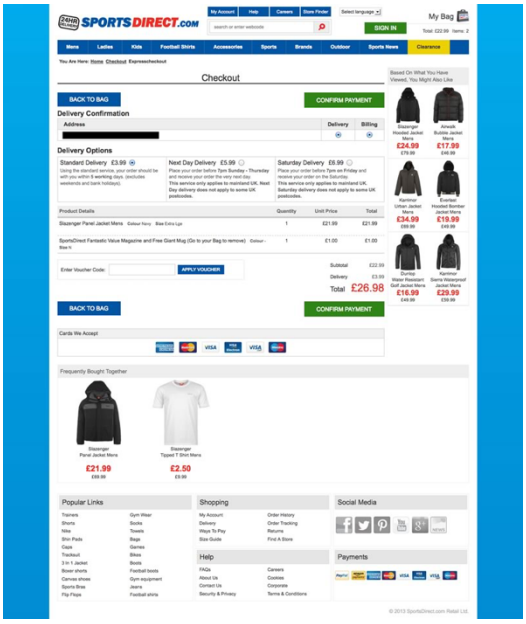

Figure 11

Since I was a bit worried about the ability to actually analyze the HTML of these websites in a way that is meaningful in the context of Dark Patterns, I would hope to utilize the Dark Patterns found in "Dark Patterns at Scale: Findings from a Crawl of 11K Shopping Websites" to use in my own classification system in future experimentation and code development. When analyzing the information for the websites pulled, I noticed that some were not in English (such as Yandex.ru) and some had different layouts that made searching for Dark Patterns harder than others. Many of the websites not being in English made searching for text harder (which is a large portion of the Dark Patterns I chose to look for — deceptive design in language). This means that many of the buzzwords I chose to search for to designate Dark Patterns would not be applicable. Turning attention toward the layout issue of the HTML, I also found that various websites have different ways of concealing dark patterns. Many have popups whereas some don't reveal its Dark Patterns until the *very end* of the check-out process, as seen in Figure 11.

In this example, Sports Direct ads items to the user's bag without their knowledge and can only be seen when the user navigates to the basket. This is an issue as, if the user goes straight to checkout without paying attention to their bag/what has been added to it, they may miss items that have been added without their knowing. This is a prime example of designers "tak[ing] actions on behalf of the users" that are costly to the user if said actions go unnoticed, according to the Interaction Design Foundation (IDF). The same article referenced by IDF also incorporates examples from previously mentioned Brignull, a Dark Pattern expert, who claims "that this maneuver is the equivalent of a supermarket worker putting things into your shopping cart without your knowledge, items which only come to your attention when you reach the checkout, if they even do so at all" (Interactive Design Foundation).

Part 5: Understanding the impacts of dark patterns and how certain groups of people can be at a disadvantage – what is the solution?

Thinking of customer experience (CX) as a UX designer is widely agreed on to be key to protecting the user of a platform as well as creating an experience that is improved and less frustrating in the long run. Transparency and obvious choices will ultimately lead users to make more informed and more accurate choices when navigating many websites that contain Dark Patterns and manipulative design choices. Large amounts of trust fall on UX designers in this school of thought, however, and tend to be vague. This lack of structure in "keeping things transparent" tends to leave the door open for interpretation that may lead to accidental deceptive design choices on the part of the UX designer. Keeping this in mind, it may also be beneficial for both UX and CX to be protected under the law.

Political regulations and policies could and have been used to protect the users and customers across various types of platforms, as seen in the California Privacy Rights Act (CPRA) and the Colorado Privacy Act (CPA). Both acts "prohibit the use of dark patterns to obtain consumer consent" (Strauss and Weber) and protect the users' personal information when navigating online platforms.

Dark Patterns are also recognizable and combatable by the user they are meant to dupe in many instances. A large defense against the discussed deceptive designs is awareness, educating oneself, and intentionality on the part of a user when interacting with a website. One website article claims that the best way to ensure user security is to change account settings (as there are many deceptive default values) and knowing which account settings to change. Other articles recommend that, even in the event of legislative changes and laws to protect the user, it is vital that said user works to continue to inform themselves so that they are not only able to recognize Dark Patterns but be prepared with action to take when facing them. Lastly, some articles propose what *not* to do considering a rise in Dark Patterns when using the internet. While such legislation like Europe's General Data Protection Regulation (GDPR), CCPA, and CPRA, Dark Patterns are still relevant and while we must know what to do when interacting with Dark Pattern heavy websites, we must also be intention with our actions when considering what *not* to do. Such examples of what not to do include trusting websites wording when it comes to the ease of choice, declining for all cookies to be blindly and default enabled, and compliantly taking rewards from a website as seen in Carrot and Stick deceptive design patterns. The same article also calls on businesses to be intention when designing websites and when considering word choice as businesses have just as much of a responsibility to consider how their platform designs are manipulative as much as the user has a responsibility to stay informed and alert when using the internet.

Works Cited (Direct Quotes in Write Up

bbarnett. “Google Maps Now Requires WiFi Scanning to Use Navigation.” *Hacker News*, 2022, https://news.ycombinator.com/item?id=30167865%27.

Goldmacher, Shane. “How Trump Steered Supporters into Unwitting Donations.” *The New York Times*, The New York Times, 3 Apr. 2021, https://www.nytimes.com/2021/04/03/us/politics/trump-donations.html.

Singer, Paul, and Jessica Rich. “Dark Patterns: A New Legal Standard or Just a Catchy Name? (Part One).” *Ad Law Access*, 4 Feb. 2022, https://www.adlawaccess.com/2022/02/articles/dark-patterns-a-new-legal-standard-or-just-a-catchy-name-part-one/.

Stauss, David, and Stacey Weber. “How Do the CPRA, CPA & VCDPA Treat Dark Patterns?” *Byte Back*, 3 Apr. 2022, https://www.bytebacklaw.com/2022/03/how-do-the-cpra-cpa-and-vcdpa-treat-dark-patterns/#:~:text=The%20CPRA%20and%20CPA%20both,that%20have%20already%20been%20exercised.

VICTORIA [witchoria]. “Venmo has a feature where the person that you’ve requested payment from can opt into payment protection, which adds a 1.9% fee to the transaction that the requestor pays without any sort of consent or control if opted into. It’s a dark pattern that I can’t believe is legal.” *Twitter,* January 10, 2022, https://twitter.com/witchoria/status/1480554922429014023.

“What Is Sneaking into Basket?” *The Interaction Design Foundation*, https://www.interaction-design.org/literature/topics/sneaking-into-basket#:~:text=Sometimes%2C%20dark%20patterns%20can%20be,adjust%20the%20amount%20to%20pay.

Additional Resources Referenced

Github Repository (Data):

- https://github.com/aruneshmathur/dark-patterns
- CSV from Assignment 8 also used in regards to top one million websites visited

Examples of Manipulative Design:

Trump in class example:

- https://www.nytimes.com/2021/04/03/us/politics/trump-donations.html

Research paper categorizing dark patterns:

- https://arxiv.org/pdf/1907.07032.pdf

Types of Dark Patterns:

- https://wirewheel.io/blog/dark-patterns-and-privacy/
- https://www.deceptive.design/types
- https://tangibleai.com/training-a-python-to-explore-holes-in-dark-patterns/

Other examples of dark patterns:

- https://www.vox.com/recode/22351108/dark-patterns-ui-web-design-privacy
- https://www.deceptive.design/hall-of-shame/all

Legality and Penalties:

- https://www.law.cornell.edu/uscode/text/15/54

Confirmshaming:

- https://builtin.com/design-ux/confirmshaming

Friendspam:

- https://www.deceptive.design/types/friend-spam

Disguised Ads:

- https://blog.mobiversal.com/dark-patterns-or-how-ux-exploits-the-user-disguised-ads-forced-continuity.html#:~:text=One%20of%20the%20most%20ubiquitous,of%20the%20ad%20intended%20it
- https://www.makeuseof.com/tag/spot-avoid-ads-disguised-download-buttons/

Checkbox:

- https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input/checkbox
- https://econsultancy.com/13-examples-of-dark-patterns-in-ecommerce-checkouts/

Urgency:

- https://www.w3schools.com/howto/howto_js_countdown.asp

Sneak into Basket:

- https://90percentofeverything.com/2013/10/21/dark-patterns-and-sports-direct-on-bbc-watchdog/
- https://darkpatterns.uxp2.com/pattern/sports-direct-sneak-into-basket/
- https://www.interaction-design.org/literature/topics/sneaking-into-basket#:~:text=Sometimes%2C%20dark%20patterns%20can%20be,adjust%20the%20amount%20to%20pay
- https://www.interaction-design.org/literature/topics/sneaking-into-basket#:~:text=Sometimes%2C%20dark%20patterns%20can%20be,adjust%20the%20amount%20to%20pay

- https://www.deceptive.design/types/sneak-into-basket

Checkbox defaults:

- https://www.w3schools.com/tags/att_input_checked.asp
- https://www.w3schools.com/tags/tryit.asp?filename=tryhtml5_input_type_checkbox

How to protect the user:

- https://careerfoundry.com/en/blog/ux-design/dark-patterns-ux/#what-to-do-instead-of-using-dark-patterns
- https://www.bytebacklaw.com/2022/03/how-do-the-cpra-cpa-and-vcdpa-treat-dark-patterns/#:~:text=The%20CPRA%20and%20CPA%20both,that%20have%20already%20been%20exercised
- https://www.squirepattonboggs.com/-/media/files/insights/publications/2021/09/develop-a-preparedness-plan-now/42123-cpracdpacpa-unpacked-brochure.pdf
- https://cennydd.com/writing/if-you-think-all-design-is-manipulation-please-stop-designing
- https://news.ycombinator.com/item?id=30167865
- https://www.termsfeed.com/blog/dark-patterns/#Summary
- https://hellofuture.orange.com/en/what-are-the-measures-against-dark-patterns/
- https://clario.co/blog/avoid-dark-patterns/
- https://uspirg.org/blogs/blog/usp/dark-patterns-step-step-guide-protect-your-privacy-your-phone

Links for code development:

Web-Scraping:

- https://realpython.com/python-web-scraping-practical-introduction/
- https://stackoverflow.com/questions/41982475/scraper-in-python-gives-access-denied
- https://support.payjunction.com/hc/en-us/articles/214132488-How-do-I-reset-or-enable-my-JavaScript-and-Cookie-settings-
- https://stackoverflow.com/questions/55749558/webscraping-crunchbase-access-denied-while-using-user-agent-header

Writing data frame to CSV:

- https://pythonguides.com/python-dictionary-to-csv/
- https://www.adamsmith.haus/python/answers/how-to-save-a-pandas-dataframe-in-python

Beautiful Soup and HTML related links:

- https://www.crummy.com/software/BeautifulSoup/bs4/doc/
- https://riptutorial.com/beautifulsoup/example/6339/locate-a-text-after-an-element-in-beautifulsoup
- https://linuxpip.org/beautifulsoup-get-text/
- https://www.geeksforgeeks.org/find-the-text-of-the-given-tag-using-beautifulsoup/

Data Analysis

- https://www.stechies.com/find-mean-mode-median-python-data-science/

Other

- https://realpython.com/python-web-scraping-practical-introduction/

- https://www.diva-portal.org/smash/get/diva2:1570073/FULLTEXT02
- https://wirewheel.io/blog/dark-patterns-and-privacy/
- https://www.vox.com/recode/22351108/dark-patterns-ui-web-design-privacy
- https://www.wired.com/story/how-to-spot-avoid-dark-patterns/
- https://www.netsolutions.com/insights/dark-patterns-in-ux-disadvantages/
- https://designmodo.com/dark-patterns/
- https://uxdesign.cc/dark-patterns-in-ux-design-7009a83b233c
- https://www.iccl.ie/news/gdpr-enforcer-rules-that-iab-europes-consent-popups-are-unlawful/
- https://www.adlawaccess.com/2022/02/articles/dark-patterns-a-new-legal-standard-or-just-a-catchy-name-part-two/
- https://github.com/SeleniumHQ/selenium
- https://selenium-python.readthedocs.io/installation.html
- https://www.selenium.dev/
- https://www.geeksforgeeks.org/selenium-python-tutorial/
- https://www.google.com/search?q=how+to+see+popups+in+html&rlz=1C5CHFA_enUS864US867&sxsrf=ALiCzsZiConHbaUefGmRbjUUgocZVPWDvw%3A1653767401029&ei=6XySYquuAaWPwbkP6MucwAg&ved=0ahUKEwjrxurT-4L4AhWlRzABHeglB4gQ4dUDCA4&uact=5&oq=how+to+see+popups+in+html&gs_lcp=Cgdnd3Mtd2l6EAM6BwgAEEcQsANKBAhBGABKBAhGGABQggxYpQxgqA5oA3ABeAGAAeMBiAGeA5IBBTAuMS4xmAEAoAEByAEIwAEB&sclient=gws-wiz