

Tallinn University
School of Digital Technologies

Dark Patterns of Web Design

Master Thesis

Risto Kitsing

Author: "....."

Supervisor: "....."

Supervisor: "....."

Head of the School: "....."

Spring 2018



TALLINN UNIVERSITY

Non-exclusive license for reproducing the thesis and making it publicly available

I, Risto Kitsing

1. grant Tallinn University a permit (a non-exclusive license) to freely reproduce and make publicly available in the repository of Tallinn University Academic Library a piece of work created by me

Dark Patterns of Web design

supervised by

Ilja Šmorgun, PhD

David Lamas, PhD

2. I am aware of the fact that the author also retains the rights mentioned in Clause 1.
3. I certify that granting the non-exclusive license does not infringe on the intellectual property rights of other persons or the rights arising from the Personal Data Protection Act.

Signed in Tallinn,

(signature and date)

Declaration

I declare that apart from the work whose authors are clearly acknowledged, this thesis is the result of my own and original work.

This work has not and is not being submitted for any other comparable academic degree.

The thesis was supervised by Ilja Šmorgun, PhD and David Lamas, PhD.

Risto Kitsing

..... (date)

..... (signature)

Acknowledgements

I would like to express my sincere gratitude to my supervisor Ilja Šmorgun. He was utterly patient with me throughout the journey and was always willing to help. I am also thankful for his wonderful lectures where he demonstrated that not all academic professors are boring, demonstrating his passion for innovation and interest in cutting-edge technology.

I would also like to thank David Lamas for sharing his expertise in multiple fields of Human-Computer Interaction and introducing me to new approaches in researching.

Abstract

When interaction designers have been given a task to compose a new product or service on the Web, they tend to refer to already existing and proven tools, such as usability patterns. Where regular patterns describe good practices and anti-practices serve as a warning of non-working solutions, a third and a more sinister type has been identified within the Human-Computer Interaction (HCI) community - dark patterns - that have been crafted to trick users to serve the interests of a business. These organizations that use dark patterns operate within gray areas, where they are not always legally liable, however guilty by violating ethical practices. As there is no clear definition of what is a dark pattern, it becomes difficult to document and observe its behavior and effects in an ethical way. In this paper, we provide such insight to what constitutes a dark pattern, how does it differ from other usability design patterns and why are they so commonly found on the Web. Describing suitable approaches to raising awareness about dangers of dark patterns enables knowledge transfer within HCI community.

Keywords: *patterns, design patterns, anti-patterns, dark patterns*

Contents

1	Introduction	1
1.1	Dark patterns	1
1.2	Research problem	2
1.3	Research goal	3
1.4	Research questions	3
1.5	Thesis structure	4
1.6	Research methodology	4
2	Historical overview of patterns	5
2.1	Patterns in Urban Architecture	5
2.2	Patterns in Software Engineering	8
2.3	Patterns in HCI	10
3	Usability design patterns	12
3.1	Alexandrian pattern	12
3.2	Regular pattern	14
3.3	Anti-pattern	15
3.4	Dark patterns	17

3.5	Closing remarks	18
4	Dark patterns in Web design	21
4.1	Motivation to use dark patterns	21
4.2	Effects from using dark patterns	23
4.3	How to stay safe from dark patterns	24
4.4	Raising awareness	25
4.5	Closing remarks	27
5	Discussion	29
6	Conclusion	31
7	Kokkuvõte	33

List of Tables

- 3.1 Alexandrian form pattern structure 13
- 4.1 Usability heuristics inverted to a dark pattern (Brignull, 2013a) 23

Chapter 1

Introduction

1.1 Dark patterns

Usually, when user experience specialists help to design a product or a service, they tend to use general guidelines and methods that all have been widely accepted in the community as good practice. When we encounter poorly designed interfaces, we might think of it as bad design, going against the general consensus of good design. These types of interfaces could be a result of various factors, such as laziness or inexperience, where designers were not passionate enough about the product or design was executed by inexperienced specialists. This type of bad design is usually referred to as an anti-pattern, the opposite of a regular design pattern, which describes a generally repeatable solution to a commonly observed usability problem (Folmer, n.d.).

There exists, however, a more sinister approach to patterns that can be used when designing interfaces. These patterns are called dark patterns, a phenomena which entails great attention to detail with a solid understanding of human psychology to trick users into doing actions that they otherwise would not do (Brignull, 2010a). According to Brignull (2010b), when people are using the web, they skim the contents and make their personal assumptions. Whenever a company wants to take advantage of the user, they will do it in a manner by making a part of a website look like it is saying one thing, when in reality it

does something absolutely different.

In order to defend users and help those who had fallen prey to deceiving design, Brignull started to catalog dark patterns, giving them a provocative name. For example, a service that demands little effort to sign up, yet makes it near impossible to cancel the subscription, is called a *Roach Motel* (Brownlee, 2017). If more can be identified about what constitutes a dark pattern, perhaps it can be more easily observed and documented, in order to safeguard users and serve as warnings to experience designers.

So far, there have been few advances in this area, although dark patterns still remain an understudied topic in HCI (Human-Computer Interaction) literature. An initial analysis and classification of regular, anti- and dark patterns has been conducted by Mirnig and Tscheligi (2017), in which they give an overview of the minimal requirements per pattern type, which will also serve as the groundwork for this research. Another, more recent research has been conducted by Gray, Kou, Battles, Hoggatt, and Toombs (2018), where the researchers analyzed instances of dark patterns and set an agenda for future studies about the ethical dimensions of using dark patterns in UX (user experience).

1.2 Research problem

Companies are implementing deceptive design practices on the Web in such a way, that products or services are more likely to be presented as a benefit or have been obscured in a such manner, where in reality they are not. These companies act in a gray area, where they are not in direct danger of being prosecuted or held legally liable. In some cases, this might be by accident, however, given the reach and interactive nature of the Web, dark patterns have been carefully crafted to be effective and not have the user interests in mind.

This presents a threat towards the users, where they are in danger of being exploited of their resource to benefit a company or organization. According to Brownlee (2017), companies are driven by metrics and to fulfill business quotas, designers are being forced into implementing dark patterns to boost key performance indexes. As there is no clear

definition of what is a dark pattern, it becomes open to anyone to interpret in their own fashion.

1.3 Research goal

The goal of this thesis is to obtain knowledge and insight concerning what constitutes a dark pattern, can it be observed in a similar fashion to a regular usability design pattern and what are the reasons dark patterns are so widely used and how to raise awareness about the topic.

This paper is not a *how-to* on using nefarious approaches on the Web to obtain monetary or any other kind of advantage over users. The author of the research believes that designers have a major responsibility in the work they put out to the real world, and doing so must follow a code of ethics, for example, as described by Monteiro (2017):

Before you are a designer, you are a human being. Like every other human being on the planet, you are part of the social contract. We share a planet. By choosing to be a designer you are choosing to impact the people who come in contact with your work, you can either help or hurt them with your actions. The effect of what you put into the fabric of society should always be a key consideration in your work. Every human being on this planet is obligated to do our best to leave this planet in better shape than we found it. Designers don't get to opt out.

1.4 Research questions

To achieve the goal of the research, the author has determined five research questions (RQ) with RQ0 being the main question:

- RQ0: What defines a dark pattern in Web design?

- RQ1: What structure can be used to define a usability design pattern?
- RQ2: What are the differences of dark patterns to regular and anti-patterns?
- RQ3: What are the reasons why designers use dark patterns?
- RQ4: What are the suitable approaches to raising awareness about dangers of dark patterns?

1.5 Thesis structure

Chapter 1 presents an introduction to the problem and **Chapter 2** presents the historical overview of patterns and their applications in different fields. **Chapter 3** is intended to answer RQ1, with providing a general structure by which usability design patterns can be defined and finds differences between different types of patterns, addressing RQ2. **Chapter 4** provides some insight to why designers use dark patterns to answer RQ3 and suitable strategies to raise awareness about the dangers of dark patterns, to answer RQ4. Finally, **Chapter 5** and onward outlines the main conclusions and discussion about RQ0 and identifies limitations to the study and recommendations for further research.

1.6 Research methodology

Due to the complex nature of the topic and its brevity among HCI discussion groups, the author of the research relies mostly on meta-analysis to further distill findings from another authors and systematic literature review by aforementioned sources. A great deal of literature has also been published based on interviews of the original author, Harry Brignull, who coined the term dark patterns.

Chapter 2

Historical overview of patterns

2.1 Patterns in Urban Architecture

In order to understand what is a pattern and how they are used, what are its relations to one another and how patterns can be used in different fields, we have to investigate its humble beginnings in architecture design. The original concept of pattern languages, a tool to be used by architects in urban planning and construction was conceived by Alexander (1979) in his book, *"The Timeless Way of Building"*. In it, he proposes an approach for architects to make use of hierarchically collected design patterns to make urban environments more usable and pleasing for its inhabitants.

Based on Appleton (2000), Alexander believed that architects should strive towards high quality of work that caters to the needs of inhabitants and users in their respective communities. In *"The Timeless Way of Building"*, Alexander (1979) describes a paradigm that is based on three concepts:

- The Quality, also known as *"The Quality Without a Name"*
- The Gate
- The Way, also known as *"The Timeless Way"*

According to Appleton (2000), *The Quality* is the philosophical concept of describing the essence of all living things that have qualities, such as: freedom, harmony, resilience etc - the factor that invokes feelings like "alive" and "sated". Appleton (2000) describes *The Gate* as a manifestation of the pattern language, which describes the relationships between patterns and their domains, used as a conduit to *The Quality*. Finally, Appleton (2000) explains that using *The Timeless Way* is a technique to apply the patterns in an ordered, sequential *piecemeal growth* that as a result progressively evolves into a more complex architecture, inhibiting all characteristics of *The Quality*.

This can be described in a way that Alexander wanted to make sure that architects and urban planning would follow a more organic, life-like approach to populating homes of people. Without the aforementioned concepts, his pattern language would have been mechanical, analytical, void of feelings. With his idea of a *piecemeal growth*, Alexander envisioned an incremental change occurring in the environment, affecting the architecture and its inhabitants within (Appleton, 2000).

In a more earlier book, "*A Pattern Language*", Alexander et al. (1977) presents over 250 different patterns as solutions to recurring urban architecture problems. This book is devised in such a manner, where an architect designer can use smaller-scale patterns on a single level building that scales in tandem with other patterns to, for example, a "*COMMUNITY OF 7000*" (Alexander et al., 1977). According to Appleton (2000), Alexander defined pattern languages as a collection of patterns that form a vocabulary for understanding and communicating ideas, which may be skillfully woven together into a unified unit that in turn reveal inherent structures and relationships of its parts to fulfill a shared objective.

Appleton (2000) explains that unlike a pattern compilation or a catalog, pattern languages consist of rules and guidelines that explain when and how, when faced with a problem, to apply a pattern or a set of patterns. Patterns can be classified into different collections that possess different structures than pattern languages: pattern catalogs and systems. By Appleton (2000), a pattern catalog is a collection of related patterns, that may loosely

be related, and that divides patterns into a smaller number of broad categories, enabling minor cross-referencing among patterns. However, a pattern system is a more cohesive set of related patterns that possesses multiple levels of granularity and describes interrelationships between patterns, their groupings and how to combine multiple patterns in order to solve more complex problems. One of the requirements is that all patterns must have a consistent and uniform style that enable compilation of a complete architecture to be built (Appleton, 2000).

In Alexander et al. (1977), patterns were described as individual entities that:

- describe a recurring problem within an environment;
- describe a solution for said problem in a repeatable manner;
- maintaining uniqueness over repetitions through ability of combination;
- follow the same layout

One of the key components to a successful pattern system was the notion of patterns following a similar layout that enabled Alexander to define the classical pattern. Based on Alexander et al. (1977), the first component necessary to document a pattern was an illustration or a photo, followed by an introductory paragraph that set the context of the issue and introduced methods to complete larger patterns. Next, a short headline described main problem the pattern was supposed to solve, in a few short sentences. The lengthiest component of a pattern was the body of the problem, which described multiple facets the issue and how they could manifest under certain conditions. After the problem statement, the *heart of the pattern* described how to solve the issue using text and labeled diagrams, when possible. Each pattern ended with a short paragraph that ties connections to other patterns.

Being first of its kind to create such a cohesive pattern system, his approach to describing patterns later inspired different fields to follow suit.

2.2 Patterns in Software Engineering

Software engineering saw an opportunity to pick up on the pattern idea, described by Alexander, and apply it to object-oriented development and constructing early graphical user interface designs. The idea of using patterns came to Cunningham (2011) in 1987, when he and Kent Beck were working with Tektronix development team, who were struggling with finishing up a graphical design interface, written in Smalltalk. Cunningham came up with a five pattern pseudo-language that helped the team to take advantage of Smalltalk's strengths and avoid its weaknesses. As a result, the team had developed an interface which, according to Cunningham (2011) looked spartan, yet elegant.

Cunningham (2011) saw an opportunity to share their newfound knowledge among their peers at OOPSLA (*Object-Oriented Programming, Systems, Languages and Application*) conference in 1987. In preparation, they wrote a panel position and took part of a workshop named "*Where do objects come from?*", where they talked about patterns and pattern languages, but the community did not pick up on the idea just yet. In later years, the results from the workshop started vivid discussions among software architects about various design patterns and their potential in software engineering (J. Borchers, 2000).

According to (J. Borchers, 2000), an influential collection of object-oriented software patterns was published by Erich Gamma during the newly established, annual Pattern Languages of Programming (PLoP) conferences that enabled developers and engineers to hold forums and exchange proven, generalized solutions to software design and domain problems. In 1994, Erich Gamma and his colleagues (Richard Helm, Ralph Johnson and John Vlissides) published "*Design Patterns: Elements of Reusable Object-Oriented Software*" and became known as the "*Gang of Four*" (Cunningham, 2011). The book documents 23 object-oriented design related patterns for recurring problems.

When comparing the work of Alexander to Gamma's, J. Borchers (2000) claims that the definition of a pattern has not changed much, having *name*, *context*, *problem*, *solution*, *examples*, *diagrams* and *cross-references* as the essential requirements. Appleton (2000),

however, describes two different schools of thoughts in software engineering that took a slightly more context-based approach to structuring patterns. Both versions display three conceptual levels that can be composed to create a pattern language. First of them, created by authors of *"Patterns of Software Architecture"*, define them as such (Buschmann, 1996):

- **architectural patterns**, which express a schema for software systems;
- **design patterns** that solve a general design problem within a software system;
- **idioms**, which express low-level patterns, specific to a programming language

As a comparison, authors of *"Understanding and Using Patterns in Software Development"* take a similar approach, yet more specific to software engineering terms (Riehle & Züllighoven, 1996):

- **conceptual patterns** describe concepts from an application domain point of view;
- **design patterns**, are used to describe software design constructs among software engineering entities, such as objects, classes, aggregations etc.
- **programming patterns** that describe only on a language construct

This has caused a problem, as the original approach by *"Gang of Four"* had a similar approach to Alexander's, where patterns could be defined in a similar fashion, even extended to other fields. The author of this research believes that due to experts coming from different programming language backgrounds and having unique problem spaces that were tied to hardware, organization or some other factors, software architects took the liberty to adapt the definitions of a pattern or a pattern language to their own need, disregarding already proposed structures. This most probably caused the fragmentation we see today, when it comes to defining the characteristics of a pattern.

2.3 Patterns in HCI

For user experience designers to create highly successful interactive systems, they have to cooperate within an interdisciplinary team that has expertise from stakeholders to developers. Different groups that collaborate on a project often lack a common terminology to exchange thoughts, opinions and ideas. A novel idea, proposed by Tidwell (1999), is that even novice designers have a hard time remembering various principles and might even lack the experience apply them in work, which is why usability design patterns can be used to share expertise.

According to J. Borchers (2000), about 20 years ago there had been a workshop dedicated to pattern languages in interaction design, that demonstrated an interest, but proved to be mixed among the community. The results that were reported by the workshop yielded activity patterns that described an observed behavior, rather than design patterns (J. Borchers, 2000). Borchers emphasizes on the stressed note expressed during the workshop that patterns tend to represent the values of their author due to the qualities a pattern author might consider important in a design artifact. Similar behavior was observed in the previous chapter, reinforcing the idea yet again that pattern definitions are tailored to fit the problem.

Such polarization is noticeable throughout HCI journals that have been published, with Fincher et al. (2003) bringing out three major points of conflict among UX designers, as to why to why patterns have not found wide usage. The first problem is described through the lens of a newcomer that has to grasp the scope of a pattern language and then understand multiple low-level patterns and their relations within a project, which may be described in a chaotic body of work. Secondly, she explains how multiple perspectives to defining pattern language has afforded an influx of breadth and richness of options that in the end become confusing or obscured, with some examples describing focus on egoism, defining patterns "their way". Finally, Fincher et al. (2003) describes how UX designers have lost the way of what "makes a pattern a pattern - or language a language", having imbalance in conceptual approaches and considerations to levels of detail. In addition,

Fincher et al. (2003) emphasizes on troubles that are related to pattern mining and the tools related to it.

In order to ensure quality of usability patterns and usability pattern languages, pattern mining is an important process in which identified patterns are evaluated and improved upon (Seidel, 2017). The whole process emphasizes on an incremental nature of evaluation activities, in which patterns act as theory driven prototypes that have to go through cycles of design, testing, analysis and redesign (Seidel, 2017). As such design-science research is highly resource demanding, it would be much more beneficial for the HCI community to maintain a more unified pattern language about good usability design patterns. According to J. O. Borchers and Thomas (2001), HCI has much more in common with the art of architecture than software engineering, which is why usability experts should adopt Alexander's notion of patterns and "change it consciously for HCI". Instead of using patterns as a conduit for expert communication, J. O. Borchers and Thomas (2001) believes that interactive design systems should be understandable by professionals and non-professionals alike, which is better suited for the interdisciplinary design team to exchange ideas and opinions.

The author of the research agrees with this notion, although in the context of (dark) patterns is not satisfied by the notion of existing solutions being either too shallow (*invaluable for designers*) or too thorough (*incomprehensible for regular users*) with no good solution in between. This proposes an area for future research among the HCI community to learn from previous mistakes and rather put attention towards reducing fragmentation between different fields. While architecture design might not be closely related with usability experts of today, software engineers are and finding a balance point between these two sides, with the inclusion of stakeholders would greatly enhance understandability. In case of dark and anti-patterns, they could also be documented and when situation arises, a usability expert would be able to describe the dangers of using such approaches. When put to a learning context, unified usability design pattern language would vastly decrease cycle time for learning new practices and approved methods.

Chapter 3

Usability design patterns

3.1 Alexandrian pattern

Appleton (2000) describes a pattern description format called "*Alexandrian form*", which provides essential elements to a standardized pattern structure (Table 3.1). Based on literature by Lea (2000) and Appleton (2000), if a pattern can not be defined by an Alexandrian form, it is a proto-pattern and can not be a part of a pattern catalog nor system. Appleton (2000) states, that a good pattern might not have all the traits on initial review as well as some patterns that do exhibit necessary elements might instead be bad patterns. As there is no good definition in the industry of what constitutes a pattern in general, the following model provides the bare minimum to describe a pattern. By Lea (2000) and Hillside Group (*About patterns*, n.d.), pattern requires a *context*, a *problem* and a *solution* to exist. In order to better understand each of the terms, Lea (2000) offers a short definition:

- "*Context* refers to a recurring set of situations in which the pattern applies."
- "*Problem* refers to a set of *forces* – goals and constraints – that occur in this context."
- "*Solution* refers to a canonical design form or design rule that someone can apply to *resolve* these forces."

Structure	Content
Name	Refers to the pattern; ideally would form a vocabulary name for discussing conceptual abstractions
Problem	Statement problem, describing its intent: goals and objectives that need to be reached within the context and forces
Context	Initial state of the environment before a pattern is applied to it
Forces	Motivation for the pattern to be frequently used, reveals potential trade-offs and encapsulates all areas of impact
Solution	Description of static relationships and dynamic behavior used to solve the problem, may offer variants
Examples	Aid to understand pattern's use and applicability through specific context
Resulting context	State of the environment after a pattern has been applied to it
Rationale	Justification of its steps or rules in resolving the forces
Related patterns	Similar patterns that tend to share common forces or context
Known uses	Describes known occurrences of a pattern, validating it is a proven solution to a recurring problem

Table 3.1: Alexandrian form pattern structure

From Lea's definitions, we can conclude that the existence of a context is necessary for a pattern to exist, and doing so, must occur or be needed multiple times, otherwise it would be a one-time solution. Having a problem gives meaning to a pattern within the set of *forces*. Alexander (1979) described this as "*heart of a pattern*" - a cause to resolve conflict within a context. The author of the research sees this as a key to determine and identify all other types of patterns because of its versatility to be adaptive in different contexts, either solving morally good or bad problems. Finally, a solution that brings equilibrium to the surrounding environment or a system. According to J. O. Borchers and Thomas (2001), usability designers should rather follow the pure Alexandrian form, as the structure of a pattern is much more expandable and can be written in a way that non-UX practitioners could also understand.

3.2 Regular pattern

As a pattern can describe a proven solution to reoccurring problems, it means that every pattern starts from a problem that requires a certain solution. In general, Gamma's proposed *rule of three* can be used to verify a working solution proposed by the pattern to be valid. This means the first occurrence of a solution displays the resolution of forces, second occurrence instills interest, whereas the third occurrence already suggests the working design might be worthy of a pattern (Gamma, 1994).

Mirnig and Tscheligi (2017) have conducted a meta-analysis on what requirements a successful pattern should usually satisfy, in order to be considered of sufficient quality. This criteria is important to ensure that a pattern is in accord to predefined structures or templates, which contain essential fields of information in a certain domain. For example, Gamma (1994) has proposed a detailed structure consisting of 13 fields, however, it is heavily geared towards software engineering and thus does not follow the Alexandrian form.

Mirnig and Tscheligi (2017) have also emphasized on works by Tidwell (1999), who proposes two different kinds of structure. In her book, Tidwell does mention the Alexandrian form, consisting of *name, examples, context, problem, forces, solution, resulting context* and additional *notes*. However, in later editions, Tidwell (1999) proposes a minimalistic Alexandrian form, albeit expanded, which contains *name, what, use when, why, how* and *examples*. Mirnig and Tscheligi (2017) have also identified an a more earlier research a very Alexandrian form, attuned to the automotive industry, only having characteristics like: *name, intent, topics, problem, scenario, solution, examples, keywords* and *sources*.

On observations made by Mirnig and Tscheligi (2017), they concluded that a successful pattern should have:

- *means of reference* - name, type, keywords and other similar elements that help to distinguish a certain pattern or a solution from one another;

- *problem description* - as patterns are not guidelines, they are always targeted at a specific problem;
- *context description* - determines the feasibility of a pattern depending on the context;
- *solution description* - it is best to not have a brief overview; ideally, it should be described in a step-by-step manner
- *examples* - in order to prove worthy as a pattern, there should exist at least one case of a successful solution implementation

Findings by Mirnig and Tscheligi (2017) seem to be in accord to what other literature has stated about the minimum viable structure of a pattern, especially the Alexandrian form. It is also interesting to note, how Tidwell (1999) approached documenting their patterns as essays, emphasizing on human behaviors, rather than structuring it like a interface or usability design pattern.

As there is no globally accepted definition, literature review suggests that in general, it is best to follow the Alexandrian form, as it can be used as a tool to verify a pattern and during the process of pattern mining, improve a proto-pattern. What is interesting to note, is that how well Mirnig et al. (2015) were able to apply the Alexandrian form to the automotive industry - this raises a question that perhaps a widely acknowledged pattern structure is needed, which can be enhanced upon, instead of reinventing already proven solutions, as was left an impression by software engineering community.

3.3 Anti-pattern

When a regular usability pattern is considered to be good practice, then anti-patterns are the opposite. They have been either mistakes in pattern creation or specifically documented non-working solutions to a set of forces within a certain context. Appleton (2000) distinguishes two types of anti-patterns:

- "Those that describe a bad solution to a problem which resulted in a bad situation."
- "Those that describe how to get out of a bad situation and how to proceed from there to a good solution."

The second situation is referred to as an *amelioration* pattern, a phenomena of an anti-pattern. Cunningham (2012) describes it as a prescription to move from an existing situation, which may or may not be desirable, to a more desired situation. Only requirement of an amelioration pattern is its transformational nature - to improve the relationship between old and new state. Due to its transformational characteristics, it is difficult to categorize them, however, Mirnig and Tscheligi (2017) propose that they are considered a fusion of anti- and regular patterns, where first describes a bad solution and latter a standardized, desirable behavior. Further research may be conducted in this area, however their appearance is highly unlikely. The author of the research considers amelioration patterns as a potential candidate to mitigate the effects of dark patterns, however, given the limited availability of literature, it remains as a hypothesis.

Mirnig and Tscheligi (2017) also reference anti-patterns in software design, albeit briefly. As discussed in Chapter 2, software engineering community has come up with a great deal of variants of patterns but it is not feasible to focus on that area. Rather, Cunningham (2004) describes an Alexandrian structure with few minor additions: *related anti-patterns* (or, in case of amelioration pattern, *applicable positive patterns*) and *anti-pattern category*.

Following regular pattern requirements and its quality criteria, Mirnig and Tscheligi (2017) believe it is important for a genuine anti-pattern to ensure non-reproducibility of the solution and describe in-depth, why the solution did not work as intended, providing reasons from desired and actual outcome. They conclude that a true anti-pattern should consist of same elements as regular patterns, with the addition of *result description*. It is also important to note that even though a solution might have failed, it must be documented for either teaching purposes or in project's interest to save resources.

3.4 Dark patterns

The final, least observed and documented usability design pattern - a dark pattern. According to Brownlee (2017), a dark pattern can be defined as a misleading or a deceptive technique that exploits aspects of human psychology to get users into carrying out acts they otherwise would not do. There has not been a great deal of research in journalism of what constitutes a dark pattern, rather, they are generalized and described with examples from the real world (Brignull, 2010b). This approach is similar to Tidwell (1999), where for behavioral patterns, they were presented as miniature essays or brief descriptions of bad practices. This form tends to suit end users, where a practice is put into context through an example, reducing its versatility. One characteristic of a pattern is the ability to solve a recurring problem, where binding a dark pattern to a real world example inhibits discovery of forces and problems. By documenting dark patterns in a reactive manner, describing a new pattern only when it has been discovered, could take time. Rather, if there was a way to be proactive, identifying certain characteristics and traits of a dark pattern, perhaps more users and usability designers could be informed of its malicious intent sooner.

In a turn to academic papers, Mirnig and Tscheligi (2017) describe the conflict of the nature of a dark pattern: "where a pattern describes a well-working solution and an anti-pattern describes one, which does not work well ..., a dark pattern describes a solution, where the *intention* behind it is a negative one". Both Greenberg, Boring, Vermeulen, and Dostal (2014) and Mirnig and Tscheligi (2017) agree that the one of the main distinguishing characteristics of a dark pattern from an anti-pattern is its sinister intention. Brignull (2013b) warns that dark patterns can be crafted from rulebooks that interaction designers refer for enhancing usability by inverting a proven practice. In a way, anyone could take a proven, working pattern and turn it into a dark pattern, just by having malicious intent.

Distinguishing characteristics of a dark pattern, Mirnig and Tscheligi (2017) believe it is reasonable to consider dark patterns to be closer by definition to regular patterns because of uncertainty of an applied solution. In case of an effective solution, a dark pattern should

not make the state of the environment any worse as a result. As such, if a dark pattern provides a solution that yields a worse state than before, it is just an anti-pattern candidate. Mirnig and Tscheligi (2017) consider that by observing dark patterns, it is not about reproducing a nefarious solution, but rather act a statement of how and why a certain solution should be avoided. Because of that, the researchers attribute *undesirability statement* and *undesirability justification* to dark patterns, in addition to the regular usability pattern characteristics.

Undesirability becomes an interesting addition to traits of a dark pattern, however, following the Alexandrian pattern, it can be also described as a *force* of a pattern. Mirnig and Tscheligi (2017) believe that having an undesirability attribute helps to demarcate a dark pattern from other usability patterns. While this can be true, it is not crucial in determining of what constitutes a dark pattern. From an ethical standpoint, it is worth noting that by documenting a dark pattern within a library, its *undesirability statement* must be clear to differentiate it from other usability design patterns. Furthermore, *undesirability justification* must be described as a three-part construct (Mirnig & Tscheligi, 2017), depending on the context:

- a dark pattern must state which moral and/or legal boundaries it is crossing;
- a dark pattern must be judged within the context of a society, based on local moral codes;
- a dark pattern must justify its usage within the context of legal constraints

3.5 Closing remarks

Based on literature and information introduced in previous sections, it has become apparent that there are multiple methods to identifying what constitutes a usability design pattern. While the more minimalistic approaches, such as Tidwell (1999), are suitable

to describe a pattern and its effects, the result resembles more a proto-pattern, or a potential candidate to a pattern. It can be argued that brevity is more important due to the interdisciplinary nature of HCI and it would be nigh impossible to accommodate a unified structure in all different areas. To contradict this notation, Mirnig et al. (2015) displayed that Alexandrian form is flexible enough to find usage in a narrow field of automotive industry and proved to yield positive results in pattern creation. This idea is further reinforced by J. O. Borchers and Thomas (2001), stating that HCI should avoid the mistakes done in software engineering, where fragmentation was quick to spread, based on challenges set by programming paradigms and languages. As such, the Alexandrian form seems to be an effective tool for both validating usability design patterns and creating pattern languages.

When comparing three usability patterns to each other, their characteristics depend of the context in which they are used. Regular patterns, for example, have no special traits, mostly because this is the expected, default behavior. Anti- and dark patterns, however, differ in such a way where extra attributes help to distinguish them from one another. In case of anti-patterns, it is the relationship between other failed patterns that is important to highlight within a pattern system because it acts as a warning to not use these types of patterns within a design system because they already taken resources to be explored, tested and documented but to no avail. Such habit of documenting and researching anti-patterns is also useful in teaching and learning HCI, as it improves the quality of future work.

Dark patterns, on the other hand, from an unethical standpoint would not need such notation. As deceiving design is purposely focused on obscuring information, masquerading a dark pattern as a regular pattern could have more usage within a design system, however its consequences towards the users, organizations and designers is questionable. For these reasons, the author agrees with Mirnig and Tscheligi (2017) that dark patterns should exhibit a special trait for them to be immediately distinguishable, such as the proposed *undesirability* aspect. The malicious intents of a dark pattern should still be prevalent in an Alexandrian form, so further studies can be conducted to find necessary counter-

measures. One such way is to further develop the concept of amelioration patterns that define an undesirable context from the user-centered standpoint, in order to improve the status of the system to a much more user-friendlier. However, this might prove to be difficult, as there have to be reasons, why designers and organizations employ dark patterns and why there exists so few, if any academic approaches to finding alleviations to these patterns.

Chapter 4

Dark patterns in Web design

4.1 Motivation to use dark patterns

According to Brignull (2016), when a con artist holds some secret knowledge that they intend to use towards a victim, it becomes dangerous. This gives companies and organizations the competitive edge against their users. Brignull (2016) proposed a recipe, consisting of four points, in which he finds what are the reasons why designers are being pushed towards using and employing dark patterns. He also proposes a few additional reasons as to why this could be happening.

The first two reasons as of why dark patterns are finding use, according to Brignull (2016), is due to the reason that designers work in an aggressive environment where they are constantly pressured in various ways by superiors that in turn have a fixated idea on pursuing metrical data. In an example, he describes how a designer on their first day of work is being pressured by an executive to boost income revenue by introducing some design changes to the checkout process. As such, the designer can opt to use a dark pattern, called "hidden costs", which is described by Brignull (2010b): "You get to the last step of the checkout process, only to discover some unexpected charges have appeared, e.g. delivery charges, tax, etc".

Following the example, Brignull (2016) identifies that even, if a designer is not aware of dark patterns, they might be influenced by social proof. Usually, bigger brands and organizations have a lot more resources and thus can seemingly justify their usage of dark patterns. A key component noted by Brignull (2016) is that the critical mass of users required to make a change is often too small, even for a large company. People who are irked by the fact that they fell into a trap of a dark pattern either do not seem to notice abuse and consider it as a minor inconvenience or those who do notice do not see the return on investment on the time they have to put into a resolution. Similar tactic is employed as "sneak into basket", where with a purchase something extra, undesired by the user, is added to the shopping cart by the vendor.

Final component addresses a more complex issue, which can be perceived as of now as a wicked problem. In the art of search engine optimization (SEO), two different schools of practice are identified - white hat and black hat (Brignull, 2016). Both employ methods that boost the relevancy of a website in search results, however white hat practitioners are very similar to usability experts using regular design patterns and black hat SEO can be associated with dark patterns. According to Brignull (2016), SEO practices can be identified by a search engine provider by executing and validating code, which can then be verified to either comply with white hat practices or be penalized for deceiving design practices.

As of now, usability approaches utilizing dark patterns are skirting on the shades of gray. There is no affirmative way to determine if a website is using dark patterns, unless it is specifically reported by its users or usability researchers. Brignull (2013a) also emphasizes on how easy it is to turn a simple usability heuristic, considered otherwise as a good practice to a deceptive pattern that benefits businesses (Table 4.1).

When in the previous chapter Mirnig and Tscheligi (2017) brought out that when defining a structure of a dark pattern, it is not necessary that the solution performs well, then according to Brignull (2013a) dark patterns do show a higher effectiveness in multivariate testing. This is simply because a "good" dark pattern is devised in such manner which

Psychological insight	Honest heuristic	Deceptive application
“We don’t read pages. We scan them” — Steve Krug	Aid rapid comprehension: ensure key content is shown in headings, subheadings (etc), using a strong visual hierarchy.	Hide key information: Bury facts within paragraphs of text, so some users will proceed without fully understanding the transaction.
“People tend to stick to the defaults” — Jakob Nielsen	Prevent mistakes: Default to the option that’s safest for the user. In important contexts, don’t use defaults and require the user to make an explicit choice.	Benefit from mistakes: Ensure default options benefit the business, even if this means some users convert without meaning to.
“People will do things that they see other people are doing” — Robert Cialdini	Show unedited feedback: Allow real customers to share their experiences, so they can make accurate pre-purchase evaluations.	Bury negative feedback: Hand-pick positive feedback and display it prominently. Bury negative feedback so it is hard to find.

Table 4.1: Usability heuristics inverted to a dark pattern (Brignull, 2013a)

is likely to achieve more conversions compared to an informed decision made by the user. As such, Brignull (2013a) believes that multivariate testing has led the Web to evolve naturally towards dark patterns, as the metrics imply their effectiveness and thus become desired by organizations and managements of a team, which is why designers are effectively forced into devising such harmful patterns.

4.2 Effects from using dark patterns

Every occurrence of a dark pattern and its effects on the system should be measured on a case-by-case basis, however there are some general guidelines that serve as a warning. When a business implements a dark pattern by mistake or on purpose, metrics will most probably indicate a positive result. However, this result is short-lived as users will learn about the hostile behavior (Brownlee, 2017). In the previous section, Brignull (2016) described how a con artist relies on secluded information that gives them an unfair advantage. As this knowledge is transferred through personal experiences, media or literature,

the con becomes widely known, rendering it useless. As a direct result, companies will turn to usability experts to "fix" the decrease they experienced after introducing a dark pattern to their website (Brownlee, 2017).

When users find out about the malicious intent, they tend to share it among their peers who might also use the same product or service and social media. New users will be hesitant of becoming a customer and existing customers will turn to a competitor who offers a better experience (Brownlee, 2017). As such, it is important to acknowledge that when a dark pattern is used, it is thoroughly tested within the context of a product or service. In some situations, like online games or applications, dark patterns can be used in modicum to promote a healthy engagement (Lewis, 2014).

Removing dark patterns from a website requires dedication and allocation of resources by a company, however, it does not guarantee that brand reputation will be restored. Brignull (2013a) brings an example where a hotel booking website was using dark patterns to earn more money from their users, which brought an uproar among the media. As a result, the company decided to use an honest interface, which clearly stated full prices, indicated an scarcity instead of artificially creating it etc. Doing so, based on Brignull (2013a), the company won numerous awards afterwards. The author agrees with Brignull (2013a) that a short-term quantitative metric mindset leads to problems fast, however, building user trust and using organic expansion leads to credibility and eventually more loyal customers. Hopefully this culture shift will happen soon and HCI community will learn from previous mistakes.

4.3 How to stay safe from dark patterns

It is difficult to produce a formula on how to avoid dark patterns, rather there are some helpful tips and best knowledge comes from experience. Even, if dark patterns were to be regulated overnight by larger associations where industry leading experts have the power to influence designers in their every day work, there are so many possible workarounds

for dark patterns (Gray et al., 2018). As such, dark patterns have the power over the user to take advantage of cognitive bias (Singer, 2017). When there is a registration form, where the user can simply click "Next" as if on autopilot, there probably are some hidden costs or obscured information that is not displayed (Singer, 2017). A very popular dark pattern is the use of double negatives - we do not want to not fall into a trap, do we? It is important to read and understand what is being said, as often enough, according to Brignull (2013a) companies state everything required from a legal standpoint, however users do not acknowledge fine print and later are defenseless against the company.

4.4 Raising awareness

As patterns expose knowledge about a software construction or a usability design pattern, these most probably have been documented in some form by numerous experts over multiple years. All work that has been done on patterns should therefore be widely available as a precious resource. Designers and engineers can use patterns effectively, provided there exists a pattern or a pattern language for their context and problem. According to Hillside *About patterns* (n.d.), generative pattern language development process has been practiced for about 20 years. While there are academic journals and papers available for reading, these cater mostly UX designers.

When a student, a victim or an inquiring user would like to know about dark patterns, there exist a few articles in media that try describing some of the more common patterns (Grauer, 2016). One must compile an exhaustive list that would cover most dark patterns by different media publishers. To counter that problem, Brignull (2010b) created a dedicated website for documenting dark patterns from 2010. It offers an extensive list of different dark patterns, most with a detailed example. Brignull (2010b) relies mostly on the popular Twitter platform to display real life examples as he discovers a pattern. Due to the nature of Twitter, anyone can tweet at Brignull's official account, *@darkpatterns*, contributing their own findings. In addition, Brignull (2014) has a thorough conference

video recording of dark patterns.

This proposes a problem, albeit, not severe, that someone willing enough has to go through a barrage of various mediums: academic papers, media articles, Twitter platform, personal website, conference videos. The breadth of information is not inherently bad, but when the topic is fragmented among mediums, it becomes difficult to fetch required information. Gray et al. (2018) however and their team of researchers were able to conduct a similar study on ethics of dark patterns, but as the amount of information that is created, HCI requires more and more a unified pattern library.

The author of the research proposes for future work a versatile pattern library, which combines Brignull (2010b) and Tidwell (1999) approach to patterns by being highly accessible on a generic level, and Šmorgun (2018) and Appleton (2000) offering a low-level, highly detailed Alexandrian form view of a pattern. This would require great effort, however, open-sourcing enables this sort of library to be maintainable and accessible by everyone, contributing to the body of knowledge inside HCI. In order to contribute to such a library, one would have to know what would be a good guideline to follow. There exist some general recommendations on writing patterns, albeit from software engineering aspect (Lea, 2000):

- it is more important to write patterns, rather than writing about them
- it is possible to mine existing resources for patterns, or they could be written based on personal experience
- finding similar patterns helps to validate the pattern writing process
- when writing a pattern, it is more important to focus on the quality, not quantity
- when starting to write about a pattern, it is important to figure out why and how it manifests
- a strict format should be followed when writing a pattern (the author of the thesis proposes Alexandrian form)

- peer review is important - sharing the process of writing with others improves the quality of a pattern

Of course it is important to note that by Seidel (2017), this has to be an iterative process, and as proposed, this can be an opportunity for contributors of the open-source platform to do in order to maintain quality.

4.5 Closing remarks

According to multiple sources by Brignull (2013a, 2016), it is the cultural mindset of organizations that are metric driven. In order to stay on an exponential business growth, executives are forcing designers to introduce dark patterns into a product or design to boost those numbers. Initially, it might seem that these "quick fixes" perform well either in the real world or multivariate testing, however, by Grauer (2016) it is not so. In contrast, well known brands that use dark patterns act as sort of a role model for the more inexperienced designers and the latter are influenced by their actions. As already mentioned in the first section, the lack of a tangible artifact makes it difficult to judge, if a design pattern has been used with a malicious intent, unless extracted from the system and observed.

At least as the awareness of using dark patterns increases and its effects to brands, organizations, designers and users become more apparent, it becomes less justifiable and will more fall into the category of bad design. The author believes that there should be a more dynamic pattern library, perhaps enforced by a consortium of major HCI organizations. As these patterns can be used in multiple dimensions and fields, the opportunity to collaborate and contribute is important not only from the academic standpoint, but also by the enthusiasts, who, for example, are tweeting dark patterns on Brignull (2010b) website.

Being open and accessible makes it also easier for journalists to publish new material that has a wider reach than traditional mediums. The author believes that efforts by Brignull

(2010b) has already planted a seed among the HCI community and that this topic on how to fight for the users, from a designer standpoint, becomes more important. The fact that recent publications also observe the phenomena of dark patterns will most probably lead to new discoveries and their publication and further sharing of knowledge (Gray et al., 2018; Mirnig & Tscheligi, 2017).

Chapter 5

Discussion

Based on the extensive literature review, we can derive that dark patterns do not differ much from regular design patterns that are used in Web design. As there is no clear definition to what is a pattern, every author can interpret it in their own rite. However, the state of affairs is that based on the structure proposed by Christopher Alexander in architecture design, minimum of three essential elements must be present: context, problem and a solution. The expanded structure, called Alexandrian form, describes a more complete pattern structure that can be used either for regular, anti- or dark patterns. This means all patterns can be reproduced in a similar manner, but from an ethical design point, they have to be specifically marked, especially in a pattern system where multiple patterns can be combined and used in a service or product.

While there are not many differences between patterns, what distinguishes a dark pattern from others is its intentional hostility towards users, taking advantage of them. Unlike an anti-pattern, where there was an attempt to propose a working solution to an occurring problem, that did not work for whichever reasons, anti-patterns exhibit a sub-type called amelioration pattern. These can be technically used to reach a new desired behavior within a design system. As dark patterns have emphasis on behavior, amelioration patterns could mitigate their dangers.

As of now, dark patterns are widely used in Web design, mostly because they cannot be

governed by an external force and businesses find ways to obscure fine legal print to avoid potential lawsuits. It is interesting to note here, that based on Brignull (2013a), it is the executives and marketing that are goal-oriented, using quantitative data as a driving force. Because of this, usability designers are being pressured into implementing dark patterns to see a boost in numbers, however, it is often short lived. While more experienced usability specialists can potentially find alternatives, more gullible are pressured by social proof - if big organizations can use dark patterns without repercussions, why it would not work elsewhere.

For users to avoid falling into traps, set by dark patterns, there firstly has to be more attention towards the topic. As usability experts agree on an official definition, dark patterns can be researched further, finding alternative ways to achieve a similar result in a more ethical manner. When dark patterns exist as an official entity, its dangers can be taught to the wider audience in HCI as well as the usual web user. Ideally, we would reach a point, where we acknowledge dark patterns as anti-patterns - solutions that do not work. Arguably, there probably will be those who try to use them for nefarious purposes, however, as a user who has read articles, seen conference talks, and consumed other forms of media about the topic, will know to avoid the dangers and ideally contribute their finding to a similar "Hall of Shame" that is being led by Brignull (2010b).

The limitations that the author faced was a restricted library of resources. As there has not been many research so far to what constitutes a dark pattern, the author had to retrace their steps to architecture design, where patterns in HCI and software engineering, stem from. It proved to be difficult, as a lot of material had been lost to the void of the Web. However, the author is thankful for the work of Mirnig and Tscheligi (2017) for providing a good groundwork.

Chapter 6

Conclusion

Dark pattern is a tool used in Web design to employ business-oriented tricks in order to gain something from the user. Its structure is in conformity with regular usability patterns that was initially described by Alexander (1979) and is similar to regular and anti-patterns. As every type of pattern has an intention in which context it is used, a dark pattern polarizes with a regular pattern by not describing the best practices to building user experiences, rather abusing already existing usability heuristics by inverting them for deceiving practices. Because of this, it is difficult to judge a dark pattern practitioner and instead, the brand or organization is to blame. They are the root cause that force designers to look for shortcuts in boosting metrics. Because dark patterns have a trait to perform better in multivariate testing, it is often given an oversight and taken into use without second thought. To stop this behavior and start acknowledging the problem among web users and usability experts in human-computer interaction, Brignull (2010b) started a website where he collects different dark patterns and shames a business guilty of using dark patterns.

We reached this conclusion based on systematic literature review, mostly opposing to what Mirnig and Tscheligi (2017) have found in their analysis. We believe that defining a phenomena that has yet been thoroughly investigated in HCI community, needs a clear definition in order to be researched. Due to the threats that dark patterns oppose, we

believe we are moving towards a solution in which we can find possible solutions to mitigate its use and work on proactive methods to reduce the harm that is being done.

As such, future work will focus towards creating an open-source pattern library in which all usability design patterns can be identified and categorized and improved by the community. The goal of this library is to be open and understandable for everyone, raising awareness in bad practices and offering possible alternatives or solutions to mitigate non-working solutions.

Chapter 7

Kokkuvõte

Käesolev magistritöö "Tumedad mustrid veebidisainis" uurib seni vähekirjeldatud fenomeni, milleks on tumedad mustrid kasutajakogemuses. Need on kasutatavuse käitumismustrid, millest eksisteerib kolm erinevat varianti: regulaarsed mustrid, vastand-mustrid ja tumedad mustrid. Regulaarsed mustrid on eetiliste disainerite töövahendid, mis aitavad hõlpsasti rakendada olemasolevatele probleemidele töötavaid lahendusi. Kokku kogutud mustritest saab luua kogumiku või keele, kus erinevad mustrid on omavahel kas grupeeritud või seotud. Vastand-mustrid kirjeldavad, kuidas konkreetse ülesande lahendamiseks on leitud mittetoimiv viis. Viimaks, tumedad mustrid esindavad mitte-eealist poolt ning keskenduvad sellele, et mõjutada enda olemasoluga veebidisainis kasutajaid nii, et see teeniks firma huve.

Eelmainitud mustrid sarnanevad oma struktuurilt teineteisega, lähtudes linnaplaneerimise praktikast pärit malliga (Alexander, 1979). Mirnig and Tscheligi (2017) esitavad oma seisukoha, et tumedaid mustreid ei tohiks võrrelda teiste mustritega ning nende uurimist tuleks piirata. Käesoleva töö autor usub, et see on väär lähenemine ning selleks, et piirata tumedate mustrite levikut ja praktiseerimist veebidisainis, tuleb see esmalt defineerida kindla struktuuriga ja piiritleda selle toime ja mõju. Käesolev töö ka pakub eelnimetatud struktuuri. Alles seejärel saab leida viise, kuidas rakendada erinevaid kasutusmustreid või leida alternatiive selleks, et neutraliseerida tumedate mustrite halb mõju kasutajate suhtes.

Esimene osa tööst annab üldise ülevaate tumedatest mustritest ning motivatsioonist uurida antud teemat. Järgmises peatükis kirjeldatakse mustrite ajalugu üldiselt ning antakse ülevaade nende kasutamisest arhitektuuris, tarkvaratehniks ja kasutaja kogemuse disainis. Kolmas peatükk toetub kirjanduslikule analüüsile, et defineerida konkreetne mall, mille järgi defineerida erinevaid mustreid. Samas tuuakse välja ka mustrite unikaalsed omadused, mida on vajalik märkida selleks, et uurida ja/või kasutada mustreid eetilisel moel. Töö neljas peatükk hõlmab endas kirjanduslikku uurimust selleks, et mõista miks kasutatakse tumedaid mustreid, mis on nende mõjud ja ohud ning kuidas saab läbi teaduse suurendamise hoiduda langemaks tumedate mustrite lõksu.

Töö tulemusena valmis põhjalik analüüs välja selgitamaks, mis on tumedad mustrid kasutajakogemuses ja kuidas neid defineerida. Samas selgus ka, mis on erinevate kasutusmustrite erinevused ja mistõttu rakendatakse veebidisainis tumedaid mustreid, mis eksitavad kasutajaid ning kuidas nendest mustritest hoiduda.

References

- About patterns.* (n.d.). Hillside Group. Retrieved from <http://hillside.net/patterns/about-patterns>
- Alexander, C. (1979). *The timeless way of building*. Oxford Univ. Press.
- Alexander, C., Ishikawa, S., Silverstein, M., Jacobson, M., Fiksdahl-King, I., & Shlomo, A. (1977). *A pattern language: towns, buildings, construction*. Oxford University Press.
- Appleton, B. (2000, Feb). *Patterns and software: Essential concepts and terminology*. Retrieved from <http://www.bradapp.com/docs/patterns-intro.html>
- Borchers, J. (2000). *A pattern approach to interaction design*. Wiley.
- Borchers, J. O., & Thomas, J. C. (2001). Patterns: What's in it for hci? In *Chi '01 extended abstracts on human factors in computing systems* (pp. 225–226). New York, NY, USA: ACM. Retrieved from <http://doi.acm.org.ezproxy.tlu.ee/10.1145/634067.634201> doi: 10.1145/634067.634201
- Brignull, H. (2010a, Jul). *Dark patterns: dirty tricks designers use to make people do stuff*. Retrieved from <https://www.90percentofeverything.com/2010/07/08/dark-patterns-dirty-tricks-designers-use-to-make-people-do-stuff/>
- Brignull, H. (2010b, Aug). *What are dark patterns?* Spiral Scout. Retrieved from <https://darkpatterns.org/>
- Brignull, H. (2013a, Jan). *Dark patterns: Deception vs. honesty in ui design*. A List Apart. Retrieved from <https://alistapart.com/article/dark>

-patterns-deception-vs.-honesty-in-ui-design

Brignull, H. (2013b, Aug). *Dark patterns: inside the interfaces designed to trick you.*

The Verge. Retrieved from <https://www.theverge.com/2013/8/29/4640308/dark-patterns-inside-the-interfaces-designed-to-trick-you>

Brignull, H. (2014, Mar). *Dark patterns: User interfaces designed to trick people.* Re-

trieved from <https://youtu.be/1KVyFio8gw4>

Brignull, H. (2016, Dec). *Dark patterns: User interfaces designed to trick people*

(louder version). YouTube. Retrieved from <https://www.youtube.com/watch?v=zaubGV2OG5U>

Brownlee, J. (2017, Sep). *Why dark patterns won't go away.* Co.Design. Re-

trieved from <https://www.fastcodesign.com/3060553/why-dark-patterns-wont-go-away>

Buschmann, F. (1996). *Pattern-oriented software architecture.* Wiley.

Cunningham, H. G. (2004, Dec). *Anti pattern template.* Retrieved from <http://wiki.c2.com/?AntiPatternTemplate>

Cunningham, H. G. (2011, Mar). *History of patterns.* Retrieved from <http://wiki.c2.com/?HistoryOfPatterns>

Cunningham, H. G. (2012, Dec). *Amelioration pattern.* Retrieved from <http://wiki.c2.com/?AmeliorationPattern>

Fincher, S., Finlay, J., Greene, S., Jones, L., Matchen, P., Thomas, J., & Molina, P. J.

(2003). Perspectives on hci patterns: Concepts and tools. In *Chi '03 extended abstracts on human factors in computing systems* (pp. 1044–1045). New York, NY, USA: ACM. Retrieved from <http://doi.acm.org/10.1145/765891.766140> doi: 10.1145/765891.766140

Folmer, E. (n.d.). *Interaction design patterns.* Retrieved from <https://www.interaction-design.org/literature/book/the-glossary-of-human-computer-interaction/interaction-design-patterns>

- Gamma, E. (1994). *Design patterns: elements of reusable object-oriented software*. Addison-Wesley.
- Grauer, Y. (2016, Jul). *Dark patterns are designed to trick you (and they're all over the web)*. Retrieved from <https://arstechnica.com/information-technology/2016/07/dark-patterns-are-designed-to-trick-you-and-theyre-all-over-the-web/>
- Gray, C. M., Kou, Y., Battles, B., Hoggatt, J., & Toombs, A. L. (2018). The dark (patterns) side of ux design. In *Proceedings of the 2018 chi conference on human factors in computing systems* (pp. 534:1–534:14). New York, NY, USA: ACM. Retrieved from <http://doi.acm.org.ezproxy.tlu.ee/10.1145/3173574.3174108> doi: 10.1145/3173574.3174108
- Greenberg, S., Boring, S., Vermeulen, J., & Dostal, J. (2014). Dark patterns in proxemic interactions: A critical perspective. In *Proceedings of the 2014 conference on designing interactive systems* (pp. 523–532). New York, NY, USA: ACM. Retrieved from <http://doi.acm.org.ezproxy.tlu.ee/10.1145/2598510.2598541> doi: 10.1145/2598510.2598541
- Lea, D. (2000, Nov). *Patterns-discussion faq*. Retrieved from <http://g.oswego.edu/dl/pd-FAQ/pd-FAQ.html>
- Lewis, C. (2014). *Irresistible apps motivational design patterns for apps, games, and web-based communities*. Springer.
- Mirrig, A. G., Meschtscherjakov, A., Perterer, N., Krischkowsky, A., Wurhofer, D., Beck, E., ... Tscheligi, M. (2015, Mar). User experience patterns from scientific and industry knowledge: An inclusive pattern approach. *PATTERNS 2015, The Seventh International Conferences on Pervasive Patterns and Applications*. Retrieved from http://www.thinkmind.org/index.php?view=article&articleid=patterns_2015_2_30_70011
- Mirrig, A. G., & Tscheligi, M. (2017, Feb). (don't) join the dark side: An initial analysis and classification of regular, anti-, and dark patterns. *PATTERNS 2017, The Ninth International Conferences on Pervasive Patterns and Applica-*

- tions*. Retrieved from http://www.thinkmind.org/index.php?view=article&articleid=patterns_2017_5_20_70023
- Monteiro, M. (2017, Jul). *A designer's code of ethics*. Mule Design. Retrieved from <https://muledesign.com/2017/07/a-designers-code-of-ethics>
- Riehle, D., & Züllighoven, H. (1996, November). Understanding and using patterns in software development. *Theor. Pract. Object Syst.*, 2(1), 3–13. Retrieved from [http://dx.doi.org/10.1002/\(SICI\)1096-9942\(1996\)2:1<3::AID-TAPO1>3.0.CO;2-\#](http://dx.doi.org/10.1002/(SICI)1096-9942(1996)2:1<3::AID-TAPO1>3.0.CO;2-\#) doi: 10.1002/(SICI)1096-9942(1996)2:1(3::AID-TAPO1)3.0.CO;2-\#
- Seidel, N. (2017). Empirical evaluation methods for pattern languages: Sketches, classification, and network analysis. In *Proceedings of the 22nd european conference on pattern languages of programs* (pp. 13:1–13:24). New York, NY, USA: ACM. Retrieved from <http://doi.acm.org.ezproxy.tlu.ee/10.1145/3147704.3147719> doi: 10.1145/3147704.3147719
- Singer, N. (2017, Dec). *When websites won't take no for an answer*. The New York Times. Retrieved from https://www.nytimes.com/2016/05/15/technology/personaltech/when-websites-wont-take-no-for-an-answer.html?_r=0
- Tidwell, J. (1999). *Common ground: A pattern language for human-computer interface design*. Retrieved from http://www.mit.edu/~jtidwell/interaction_patterns.html
- Šmorgun, I. (2018, Mar). *Pattern library for supporting the design of distributed user interfaces*. Tallinn University. Retrieved from <https://www.digar.ee/arhiiv/en/books/90740>