UNIVERSITY OF TURKU

Do modern social media applications suffer from feature creep?

Case YouTube

Department of Computing

Master of Science Thesis

Computer Science

September 2024

Kalle Sundell

Department of Computing
University of Turku

Subject: Interaction design
Programme: Master's Degree Programme in Computer Science
Author: Kalle Sundell
Title: Do modern social media applications suffer from feature creep? Case YouTube
Number of pages: 48 pages, 6 appendix pages
Date: September 2024

This study was conducted to examine the prevalence of excess features in modern social media platforms. Social media companies are constantly trying to find ways to improve their product, impress their audience and keep them satisfied. A common approach for achieving this appears to be the introduction of more functionality by expanding the software's existing feature set. If these newly introduced features are poorly adopted but remain in the software's feature set, the software may suffer from "creeping featurism", more commonly known as "feature creep". The presence of feature creep may have many unwanted consequences, such as decreased user satisfaction, increased difficulty in software maintenance and performance issues. This study aims to determine whether modern social media platforms suffer from this phenomenon by using the video-sharing platform YouTube as a case example.

Does YouTube suffer from feature creep? One way to find out is to ask directly from its end users. For this purpose, a survey was created. In the survey, the respondents were first asked to express their personal feelings about the platform's current state. Then, the respondents were presented with a set of features and asked to assess how useful they found a given feature to be. After this, another set of features was presented asking to assess how often they engage with a given feature. The survey was distributed across several social media platforms and a total of 44 responses were amassed.

Based on the obtained results, YouTube does not seem to suffer from feature creep in its current state despite offering quite an extensive set of features. Users were mainly happy with the user interface and the vast majority of features were at least seen as "good to have". However, an interesting observation is that a large portion of YouTube's features appear to remain unused by the general population. This portion is even more pronounced among those who are infrequent (non-daily) users of the platform. In addition, an interesting recurring pattern was detected. Out of the features in the presented feature sets, approximately 20% of features were seen as necessary, 50% as useful and 30% as unnecessary. It remains unclear whether this pattern applies for other social media platforms as well.

Keywords: Feature creep, featuritis, feature fatigue, software bloat, social media, user experience, user interface, web applications.

Table of contents

1 Introduction 1

1.1 Motivation 2

1.2 Structure 2

2 Literature 4

3 Features 7

3.1 Features in the context of web applications 8

3.1.1 Web 1.0 8

3.1.2 Web 2.0 9

3.1.3 Conclusion 11

3.2 Feature creep 11

3.2.1 Related terminology 12

3.2.2 Causes 13

3.2.3 Impact 14

3.2.4 Treatment and avoidance 16

4 Research survey 19

4.1 YouTube 19

4.2 Hypotheses & research questions 20

4.3 Survey structure 20

4.3.1 Information about the respondent 22

4.3.2 Assessing users' feelings 23

4.3.3 Assessing the usefulness of certain features 24

4.3.4 Assessing the adoption rate of features 27

4.3.5 Assessing features' effect on user engagement 29

5 Results 30

5.1 Preliminary statistics about the respondents 30

5.2 Subjective feelings 31

5.3 Perceived usefulness of features 34

5.4 Feature adoption 36

5.5 Features' effect on user engagement 37

6 Analysis 40

6.1 RQ1: Does YouTube suffer from feature creep? 40
6.2 RQ2: How well have users adopted YouTube's current feature set? 42
6.3 RQ3: How do certain features affect user engagement with the site? 45
7 Conclusion 46
References 49
Appendices 52
Appendix 1: Survey (preliminary information) 52
Appendix 2: Survey (part 1) 53
Appendix 3: Survey (part 2) 54
Appendix 4: Survey (part 3, individual features) 55
Appendix 5: Survey (part 3, platforms) 56
Appendix 6: Survey (part 4) 57

1 Introduction

Designing, developing, and delivering software has never been as seamless and straightforward as it is today. Physical media is a thing of the past; now everything is in the cloud. If you happen to notice a bug in production code, no worries – modern software delivery pipelines ensure that once a bug is patched, the latest production code swiftly reaches end users.

This is just one example of how the landscape has evolved over the past few decades. While this may all seem positive, there is a flip side. Many conveniences associated with modern software development have led to a host of new problems. With the ease of deploying functional software, the market has become more competitive. Consequently, businesses are feverishly seeking ways to distinguish themselves from competitors. After all, designers and developers must have something noteworthy to showcase to their superiors.

Modern software development is rooted in continuous delivery. In practice, this translates to the belief that the more products are released, the better. However, there is a significant flaw in this mindset. A piece of software can only accommodate a limited number of functionalities before it inevitably begins to veer away from its initial scope. The continual addition of new features actually results in a decrease in firms' profitability, as it somewhat ironically leads to diminished quality and user satisfaction, contrary to intuitive expectations. [1]

Indeed, this phenomenon has become so widespread that various terms have been coined to describe it: *feature creep* (sometimes *creeping featurism)*, *featuritis* [2], and *software bloat* [3] are among the most used to characterize this vice that exists in a lot of software these days. This thesis aims to scrutinize this topic, focusing specifically on how feature creep manifests in contemporary social media applications, with a primary emphasis on the video-sharing platform YouTube, which happens to be the second most visited website on the planet in March 2024. [4]

1.1 Motivation

The idea for the topic of this thesis arose in early 2024, as I was brainstorming different ideas. Due to lack of resources, I figured that the best research method for this thesis would be a survey. After that, in order to ensure an acceptable number of responses, I figured it would make sense to pick a subject that many people know about. In the end, I decided to go with something that would combine social media with computer science, or more precisely, (human-computer) interaction design.

Having read Alan Cooper's book *About Face: The Essentials of Interaction Design* and Don Norman's *The Design of Everyday Things,* I chose to go with something related to software application features. In the latter, Don Norman talks about a concept called *featuritis*: the ever-increasing amount of functionality in a piece of software. It immediately sparked my interest. Having been active on some kind of social media platform since 2005, I have personally seen and experienced the rise of this phenomenon. At this point, it came down to narrowing down the subject; which platform(s) to pick and what areas to analyze, and how? Having consulted my supervisor, I ended up going with YouTube due to its popularity and prevalence in the social media space. Moreover, my personal experience with the platform spans almost two decades.

Another aspect that motivated me to research this topic was the lack of scientific research on it. While it certainly made looking for credible sources more difficult, it brings a sense of meaning to this work, because I feel like I am making a concrete contribution with this work. All these attributes combined gave me a strong sense of confidence in pursuing this very topic.

1.2 Structure

This thesis is divided into two main parts: theory and research. Chapters 2 and 3 will cover the relevant background needed to understand the topic. In Chapter 2, the relevant literature on the topic is reviewed, followed by an explanation of computer software and what it embodies specifically in the context of this thesis to strengthen the reader's understanding of the main beef of this work: the feature creep. With that said, Chapter 3 covers the concept of feature and in particular, how features have manifested themselves within computer software

over time. Finally, the previously mentioned feature creep is presented by explaining what it is, how it occurs, its effects, and how to avoid it.

Chapters 4, 5 and 6 will present the research that was conducted for this thesis. The main purpose of the research part is to find answers for the three research questions that were set for the thesis. Chapter 4 covers the background of the research; why it was conducted the way it was, the research questions and the pre-existing hypotheses that inspired them. Chapter 5 is reserved for going through the results and the respondent demographics. As the last chapter of the research part, Chapter 6 is reserved for the analysis of the results of Chapter 5. We will see how and if they answered the research questions and how well they weighed against the pre-existing hypotheses. Finally, in Chapter 7, the findings of Chapters 5 and 6 are briefly summed together to conclude the work.

2 Literature

Scientific literature focusing specifically on feature creep is scarce. Using the search keyword *allintitle: "feature creep"* in Google Scholar yields 22 results. Likewise, the keyword *allintitle: "featuritis"* yields only 8 results. These are works that have the words in the title – it is obvious that many works mention the term within the text. Regardless, this alone indicates that little research has been done on the subject.

With 126 citations, the most cited paper that has the term "feature creep" in its title is *The Social Utility of Feature Creep* by Thompson & Norton (2011). In the paper, Thompson & Norton claim that "the consumer prefers feature-rich products for public display, but they may prefer feature-poor products for public performance". A good general overview for understanding feature creep as a phenomenon would be Bill Elliott's 2007 conference paper *Anything is Possible: Managing Feature Creep in an Innovation Rich Environment*, which was originally presented at the 2007 IEEE International Engineering Management Conference. In the paper, Elliott examines the "inevitable product feature creep" from the perspective of a project manager. In addition, Elliott offers several solutions for managing feature creep. The paper is referenced in this work also, namely in Chapter 3.2.4.

Based on literature analysis on the topic, it appears that – especially in the past – several other terms have been used to refer to phenomena similar to feature creep. There exists quite an extensive catalogue of papers and books focusing on software usability and user experience from a broader perspective. Many of these publications touch on concepts closely related to feature creep. The most commonly recurring terms to describe similar phenomena are:

- *Software bloat*
- *Fatware*
- *Scope creep*
- *Feature fatigue*

My arguments for why these are *not* exactly synonymous with feature creep are presented in Chapter 3.2.1. However, many of the proposed solutions to the aforementioned phenomena may also be viable for curing feature creep. Thus, some of the papers referenced in this thesis do not directly focus on feature creep, but phenomena closely related to it. An example would be *Feature Fatigue: When Product Capabilities Become Too Much of a Good Thing* (2005)

by Thompson, Hamilton and Rust. The paper offers great insight into feature fatigue as a psychological response to experiencing feature creep and how it can be mitigated by having multiple applications with a limited feature set, compared to a single monolith that takes care of everything.

Out of the books that have been heavily referenced in this work, it would be difficult to go about without mentioning the names Don Norman and Jakob Nielsen. Perhaps the most notable and respected scholar to talk about the subject is Norman, namely in his 2013 edition of *The Design of Everyday Things*. In the book, Norman owns an entire chapter for a concept called featuritis, which is synonymous with feature creep. Likewise, Nielsen in his 1993 book *Usability Engineering* uses the earlier mentioned terms *software bloat* and *fatware* to describe the phenomenon. Despite never mentioning the term "feature creep" in the book, Nielsen not only offers a variety of solutions for preventing feature creep from happening, but also gives a fair share of criticism for many commonly applied solutions to the problem. Despite being over 30 years old, the book's teachings still hold true today.

While unideal from a scientific perspective, this thesis references many online publications that may be considered unscientific, such as blog posts from individuals and/or organizations. An example of such is the wikibook *Lentis: The Social Interface of Technology*, which is an online book originally authored by students at the University of Virginia's School of Engineering and Applied Science. It claims to be a general guidebook for understanding how the world's societal and technological issues overlap. [5] One of the chapters in the book is about feature creep, which ended up being of much use. Coincidentally, among the references used in the book is the previously mentioned 2005 paper by Thompson et. al. While sources of this kind can be a good source of quick bits of information, they may be more subject to inaccuracies compared to peer-reviewed papers.

Other works referenced in this work that were not directly related to feature creep were Roger Pressman's book *Software engineering: a practitioner's approach* (2010) and Paul Ceruzzi's *The Early History of Software* (2003). To understand the topic at hand, having some elementary knowledge on computer software is necessary.

Ceruzzi [6] defines software as "the set of instructions that direct a computer to do a specific task*"*. Ceruzzi argues that simply defining software as a series of formalized and written-down steps is insufficient, as this definition can also be applied to physical, real-world tasks. Indeed, the definition of software is somewhat ambiguous. Software is not a single entity that

simply works in collaboration with hardware. Instead, a computer system consists of multiple layers of software ranging from low-level microcode all the way to high-level operative tasks, such as pressing the buttons of an ATM.

According to Pressman [7], computer software has cemented its status as the leading technological invention of our time. The rise of computer software can be considered unprecedented, as nowadays software is present in virtually all systems across all fields, ranging from transportation to telecommunication to medical and entertainment systems. The role of computer software in our society has dramatically changed over the past 50 years. Processor and memory speeds and storage capacity keeps increasing by the year, offering not only a host of possibilities, but also new problems at hand. The topic of this thesis – feature creep – can be considered one of the many indirect consequences of this development.

Pressman [7] states that software can be divided into seven different categories by their application domain. This makes sense as the definition of software is extremely broad. One of the categories is *web applications*. Essentially this term encompasses most resources that are accessed via Internet or Intranet. It can refer to a single website, a specific functionality within a website or other processing tools that are typically accessed using a web browser. More practical examples of web applications include social media platforms (e.g. Instagram, YouTube), a travel reservation system or a streaming service (e.g. Netflix, Amazon Prime Video). From this point forward, the terms "software" and "application" will be used to refer to specifically web applications.

3 Features

Any software is defined by its *feature set*; the collection of individual features that ultimately comprise its purpose. Hence, the concept of *feature* plays a major role in this thesis. The English word feature originates from its French counterpart *feture,* which in turn is derived from the Latin word *factura.* As of writing this thesis, its earliest known use in the English language is from 1325. [8] The exact meaning of the word depends on the context. In fact, while it is most commonly used as a noun, it can also be used as a verb.

Oxford English Dictionary and Cambridge Dictionary list 16 and 12 different definitions for the word respectively. In the context of business, Cambridge Dictionary defines feature as "something that makes a product, machine, or system different, and usually better, than other of a similar type." [9] While one can attribute such a definition to software, it is still somewhat business-centric and does not consider the term through the lens of software development. Due to the lack of a clear definition, the term *software feature* is occasionally used. Apel and Kästner [10] conclude that due to the diversity of research, several definitions for the word "feature" exist even in the context of software, ranging from very abstract to more technical definitions. In the context of this thesis, perhaps the most accurate definition comes from Kang et al. [11], who define a feature as "a distinctively identifiable functional abstraction that must be implemented, tested, delivered, and maintained." With this definition in mind, some practical examples of software features could be:

- Sorting items based on a condition (e.g. popularity, price)
- The ability to search for items
- Adding an item to a list (e.g. shopping cart, favorites)
- The ability to switch themes (light/dark mode)

The examples mentioned above are very general features that tend to exist in the feature set of applications across many domains. Oftentimes a feature set contains features that are more specifically tailored to the purpose of the application, such as the one shown in Figure 1. The "Notify me" button triggers a countdown timer leading up to the livestream's scheduled start time, automatically alerting the user when the event begins.

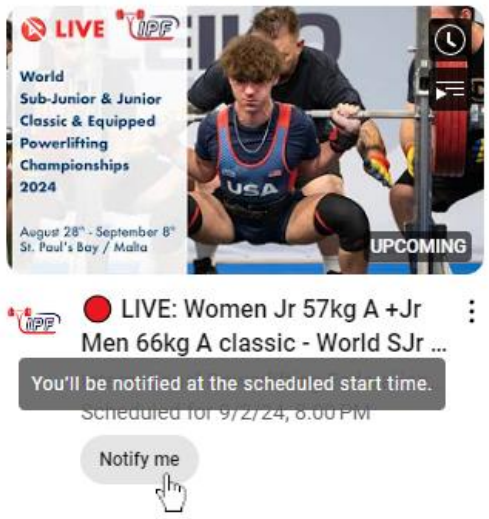


Figure 1: An example of a software feature.

3.1 Features in the context of web applications

In the research part of this thesis, we will be specifically looking at features in the context of web applications. Hence, it is good to understand the historical context of features in the web, and how we have reached the current state of things.

The World Wide Web (WWW) was officially invented in 1989. Towards the end of 1990, Tim Berners-Lee made the first successful HTTP connection with the first running web server, a NeXT computer at CERN. [12] Unsurprisingly, much of early web consisted of simple HTML documents with little to no features due to the constraints set by the technology at the time. However, since then, web applications have become increasingly more complex. The existence of World Wide Web can be divided into two separate generations: *Web 1.0* and *Web 2.0*. Both generations are marked by certain characteristics that directly influenced feature sophistication and user interaction.

3.1.1 Web 1.0

The first generation of the web – Web 1.0 – is sometimes referred to as the *read-only web*. In practice, this meant that many of the early web pages were static, meaning they had no dynamic content, which in turn greatly limited the possibilities for users to interact with the site. One-way communication was the primary focus in this era of the web; it was used by businesses to establish a form of online presence in the form of websites and newsletters. On websites, end users were mainly limited to basic forms of interactions such as simple buttons, guestbooks [13] and hypertext.

After Berners-Lee released the World Wide Web to public use in 1991 [12], enthusiastic developers started developing web browsers. In fact, it became a market of its own, eventually starting the infamous browser wars of the 1990s. ViolaWWW was among the first web browsers with more additional features, such as forward, back and home buttons, a custom scripting language and stylesheets. [14] Despite its initial success, it was shortly trumped by Mosaic, a web browser developed by a friend of Berners-Lee, Marc Andreessen. Mosaic was easier to install, and it was the first browser with the ability to display inline images, instead of opening them in a new window. In terms of available features, ViolaWWW was initially ahead of Mosaic, but the ease of installation and customer support made Mosaic a more attractive option for early web adopters. [15]

Moving from browsers to websites, a major turning point from an interaction and feature design perspective occurred in 1995 when Brendan Eich invented JavaScript. Netscape – the company behind the widely successful Navigator web browser – were looking for ways to add more interactivity to regular HTML-based websites. Previously in the same year, Sun Microsystems released Java. However, as Java was a general-purpose programming language, it was not the optimal solution for the web. Eich wanted a solution that was malleable; something that could be used or various purposes in the web. In hindsight, this was the right approach, as these days JavaScript is widely used in both server and client-side programming. [16]

When describing JavaScript in their documentation, Netscape stated that JavaScript "allows the page designer to access events such as startups, exits and user mouse clicks" [17] and "verify that users enter valid information into a form.” [18] Advancements like these naturally opened up doors for many new features for the web, many of which were considered undesirable, such as scrolling messages or flashing images. [16] Regardless, JavaScript was such a monumental advancement that it was there to stay.

3.1.2 Web 2.0

The generation of Web 2.0 saw the rise of user-generated content. While Web 1.0 is called the read-only web, Web 2.0 is known as the *participatory web*. Services like social media, personal blogs and RSS-feeds became commonplace. In other words, two-way communication is at the core of Web 2.0. It is difficult to pinpoint the exact moment of transition from Web 1.0 to Web 2.0, but the first mention of Web 2.0 is from 1999. It is reasonable to say that by year 2004, Web 2.0 was already in full swing. [19]

The advancements in web development tools towards the end of Web 1.0 sparked a new era of interaction in the web. *Rich Internet Applications* (RIA) were some of the first actual web applications. RIAs are essentially small applications embedded in a web page that resemble the look and feel of desktop applications. They offered new possibilities such as dragging and dropping, keyboard shortcuts and the utilization of the user's local computing power [20], which, in the early days of Web 2.0, was considered new.

Another feature that was considered state-of-the-art was the *web widget*. They are lightweight portable web components that can be installed and executed within any web page that is based on HTML. They do not require any installation and are cross-browser and cross-platform compatible. They are small, interactive tools that usually display live content such as news or weather forecast. [21]

JavaScript saw its conception towards the end of Web 1.0. However, it was not until Web 2.0 that the language really started to blossom in the commercial realm. Along with the advancements in server-side web APIs and web development frameworks, the three combined allowed the development of lightweight, dynamic and asynchronous web applications. In Web 1.0, websites were static pages that had links to other pages. In Web 2.0 and emergence of JavaScript, so-called *single-page applications* (SPA) became commonplace. In regular multi-page applications, the approach is to have multiple HTML files that are then dynamically generated by the backend as they are requested. In SPA, the data and content of a web page is sent and received in the background without affecting the current page view from the user’s perspective. An SPA-type approach was first suggested in 2002 as a more cost-efficient solution, offering a more seamless browsing experience. [22]

Moreover, the proliferation of social media platforms such as Facebook, YouTube, Instagram and Pinterest embody the nature of Web 2.0 and nothing short of revolutionized the way we consume media on the internet. Features such as liking, commenting, following and subscribing became mainstream concepts. Furthermore, many of these sites started expanding well beyond their original scope.

As a concrete example, Facebook quite literally began as a face book initially for the students of Harvard University to network with other students. [23] It then spread across most US universities, eventually opening its doors to the public and reaching mainstream success. It has since then went on to become a larger monolith that offers a variety of services, such as a dedicated marketplace [24] and even a dating platform. [25]

3.1.3 Conclusion

Given the background on Web 1.0 and Web 2.0, it is easy to deduce that as toolkits for development became more advanced, so did the platforms themselves. In terms of user experience, “more advanced” does not necessarily translate into “better”. Especially the expansion of large social media platform such as Facebook, Instagram and YouTube raise interesting questions: does everything have to be centralized? Is there really a need for many of the services that these platforms offer? Could some of the sub-platforms be separate platforms altogether? Would it make sense to ask the users about their opinion?

Rich user interfaces can offer fascinating visual experiences, but can businesses go too far with their interface sophistication? Are they inherently subject to feature creep? Is there a point where application features become counterproductive for the user and for the business? If so, what would such point be? In the research part of this thesis, we aim to find answers to questions like these by asking the end users directly.

3.2 Feature creep

As mentioned in the introduction to this chapter, every piece of software has its own feature set, in other words, a collection of features that determine what the software does. A feature set essentially justifies the software’s reason to exist. For instance, a simple text editor could be used to write text, edit the font and save the text in a given file format. These features would make up the feature set of the software.

When a feature set in a piece of software grows in excess, such as in Figure 2, the software is said to suffer from *feature creep*, sometimes referred to as *featuritis.* [2] Especially among commercial products, it is not uncommon to see the product slowly veer away from its original purpose by incorporating additional features into the product, usually as a reaction to industry trends and increasing competition in the market. [1]

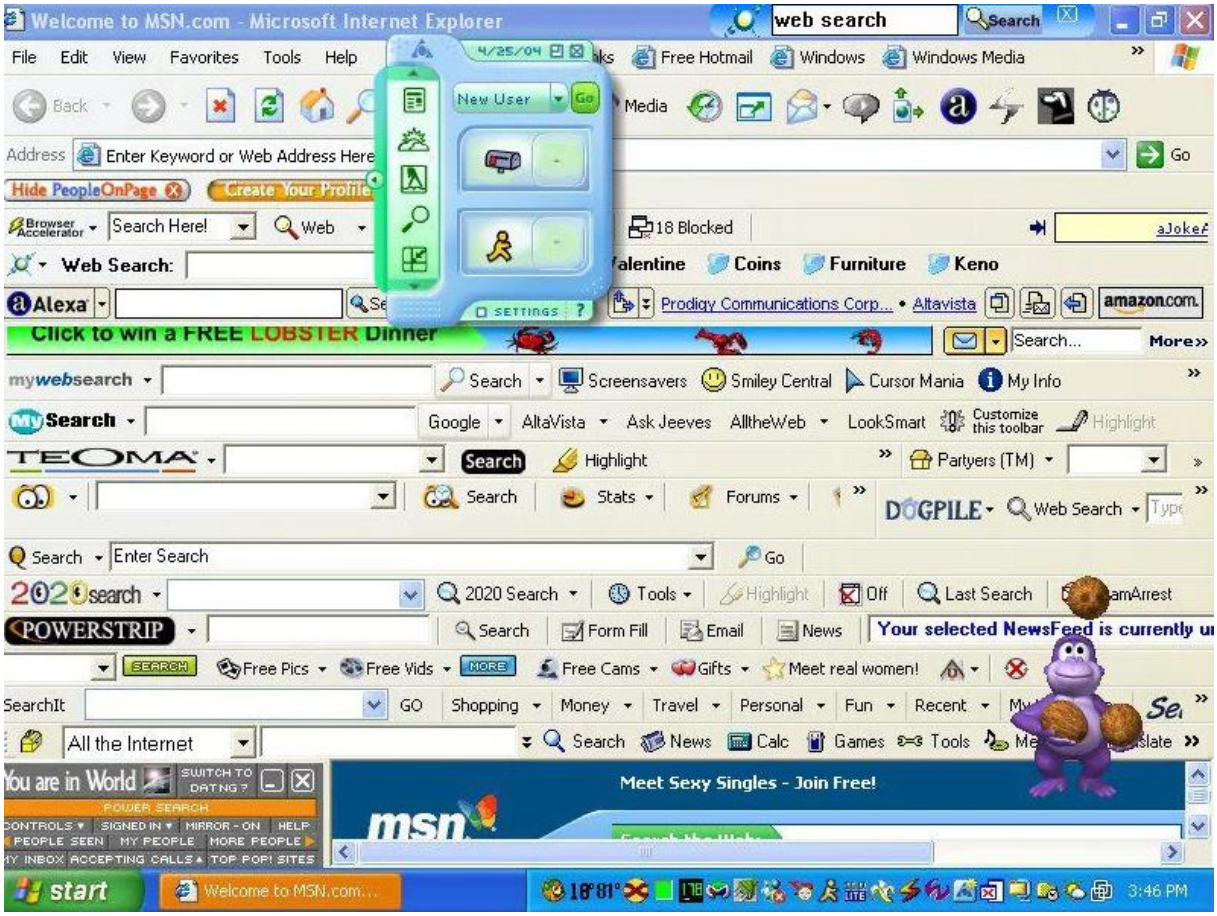


Figure 2: Third-party browser toolbars that were popular in the mid-2000s are a good example of feature creep. Picture: Unknown reddit user on r/Nostalgia

3.2.1 Related terminology

As mentioned in Chapter 2, in both scientific and non-scientific literature the terms *software bloat, scope creep, fatware* and *feature fatigue* regularly appear to refer to similar or identical phenomena. All of the terms are closely related to one another, but they touch on the topic from slightly different perspectives.

Software bloat is mostly concerned about the issue from a hardware resource consumption perspective. As programs grow in size, they consume more memory and computing power.

Swiss computer scientist Niklaus Wirth stated in his 1995 article that software is getting slower more rapidly than hardware is becoming faster [26]. This is sometimes referred to as *Wirth's law*, and it encompasses why software bloat can be a serious issue.

Scope creep is sometimes used interchangeably with feature creep. This, however, is erroneous, because scope creep looks at the issue from a human resource perspective; when a project scope is not properly defined and it is not adhered to, it can result in side effects such as decreased quality, delayed schedules, increased costs and decreased customer satisfaction. [27] Feature creep looks at the issue through a user experience lens. As software grows in complexity, it may hinder its usability and result in user drop off[1] and even user exit[2]. [28]

Fatware is practically synonymous with software bloat. The term appears to have reached its peak usage in early-to-mid 90s with little use thereafter. When making a search in Google Scholar with the query *allintitle: "fatware"*, only ten results appear, with eight of them being from the 90s and the remaining two from early 2000s.

Feature fatigue refers to the mental state of dissatisfaction and frustration that users tend to experience after using an overly feature-rich product that do not exceed their expectations. [1] It differs from feature creep in the sense that it examines the psychological reaction *after* feature creep has occurred, not the underlying reason *why* products get overly complex in the first place. In simpler terms, feature fatigue can be thought of as a psychological response to feature creep.

3.2.2 Causes

The underlying cause behind feature creep differs on a case-to-case basis. In the book mentioned in the previous chapter, Don Norman [2] identifies three main reasons feature creep occurs:

1. *Existing users.* They may already like the software but demand more capabilities and sophistication.
2. *Market competition.* Keeping track on what competitors are doing increases the temptation to incorporate features to get ahead in the market and match the competitor in terms of technical progress.

[1] A scenario in which a user's perceived user experience deviates from the planned user experience. [28]
[2] A scenario in which bad user experience leads to the user abandoning the product altogether. [28]

3. *Declining sales.* Reaching a plateau in sales figures might increase the temptation to offer enhancements, such as upgrades or DLCs.

Likewise, Thompson, Hamilton & Rust double down on the point that product differentiation seems to drive feature creep. There is evidence suggesting that an increased number of features will initially make a product more appealing to its target audience. This creates an effect of positive differentiation in the sense that it will give the product some perceived advantages over competing products. [1]

Ultimately, each case of feature creep is different, and the aforementioned points are certainly not the only reasons for feature creep to occur. In fact, the root cause may often stem from within the organization, with no external pressure being present. An individual team within an organization may put their personal needs above the big picture, whereas sometimes a non-technical, high-ranking person within the organization may demand a feature to be added without fully understanding the consequences of such additions. [29]

3.2.3 Impact

Feature creep manifests itself in various ways. Nielsen [3] states that every time a feature is added to a system, it grows in size, becomes more intimidating and harder to search, as there is yet another aspect for the users to learn.

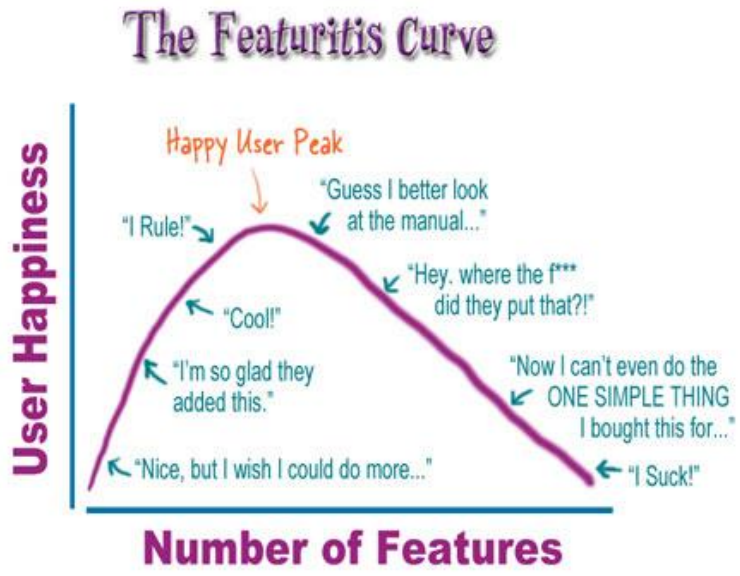


Figure 3: The Featuritis Curve. Picture: Kathy Sierra

Feature creep often results in unsatisfactory user experience, as Sierra illustrates in her idea of the “Featuritis Curve” [30], presented in Figure 3. Incorporating new features can initially seem like a way to enhance the user experience, but often the effect is opposite. Users might have become familiar with a specific interface and have an existing mental map of its capabilities in their head, and unnecessary enhancements might have a negative impact on its usability, resulting in suboptimal user experience. In addition, a bloated interface is almost certain to become slower due to the increased hardware consumption caused by the increased demands in processing power. [5] This phenomenon, as previously discussed, is known as software bloat.

To provide a practical, worst-case example, the fate of the Netscape browser series is a good pick. In early 1998, the developers realized that the codebase had become too complicated, which eventually lead to its complete rewriting, and Netscape 5 was completely skipped. [31] When Netscape 6, eventually came out (see Figure 4), it was criticized for its inconvenient user experience and questionable set of new features. It was generally deemed to be too slow to use for most computers of that era. [32] Eventually, in 2001, the release of Netscape 6.1 and 6.2 fixed most of the issues, but by this time Internet Explorer had already become the industry standard for web browsers, ending the reign of Netscape. [33]

Figure 4: Netscape 6.0 was criticized for its creeping featurism and instability. Picture: CVC Internet LLC

3.2.4 Treatment and avoidance

The root cause of feature creep differs on a case-to-case basis. Therefore, the proper treatment option must be selected accordingly. To understand how to approach treating the problem, one must first have firm understanding of its roots. In this chapter, I offer four different approaches for treating feature creep.

1. *Improving organizational issues*

On many occasions, feature creep is caused by an organizational issue. Lack of coordination among teams within an organization is a good example. Ideas for new features may come from different parts of the organization, mainly from customer-facing teams such as sales, marketing and those in the upper echelon of the organization. They may come up with ideas with the best of intentions, though unknowingly contributing to the feature creep problem. [34] It is advisable for teams involved in the project to collaborate early to prevent constant modifications to the product caused by individual needs. [29]

Remaining faithful to the project scope is another effective way to combat feature creep at an organizational level. A well-defined project scope that is adhered to serves as a good starting point for keeping all stakeholders grounded. [29] The project manager in particular is the person with the greatest stake in keeping feature creep at bay. In addition, engineers have the tendency to identify, self-approve and implement new features into a product. It can be a beneficial trait in tasks such as bug fixing or building robust software, but less optimal when trying to stop features from creeping in. [34]

2. *User analytics*

Having a firm understanding of how users use the application is an effective way to differentiate the deadweight features from those that receive active usage. This can be achieved through observing user behavior through analytics, but also by integrating an easy feedback loop within the application, or even physically observing the user using the system. [3]

Combining practical user analytics with a user-centered design philosophy is a good combination. In his 2013 book, Norman uses Amazon as an example. He claims that the approach of its CEO Jeff Bezos is "customer obsessed". At Amazon, competition and traditional marketing requirements are ignored, and the development is purely customer-

driven – the idea is to figure out what the customers want, how their needs can be satisfied and what can be done better to enhance customer experience. [2] Same philosophy can be applied to interface design, among other things.

3. Pruning

A rather blunt yet effective approach is pruning. It is the act of getting rid of excess features in an application, i.e. removing them altogether. The idea is that after a successful pruning, the focus should naturally gravitate back towards the main features. [5] Nielsen suggests that user analytics may be used as a guiding tool in order to figure out which features to remove, or at the very least improve them. [3] Cotterell [28] offers three strategies for determining which features to remove (see Table 1).

Table 1: The three different strategies for determining which features to remove. [28]

Strategy	**Explanation**
Usage	Remove features that have few to no users.
Feedback	Remove features based on user feedback.
Legacy exit	Remove features related to legacy products.

4. Making better design choices

Feature creep may also be combatted – or outright prevented – by making proper design choices. In particular, two approaches may be effective in this: *modularization* and *configurable user modes*. [5]

Modularization is the idea of splitting an application into core components, and then allowing its users to complement the core components with additional add-on components. [5] This approach has been adopted by many popular applications, such as Visual Studio Code and Unity.

Another approach is to have configurable user modes to hide away some of the complexity of the application. User modes are different from a modularized approach in the sense that while both are used to obscure the complexity of the application, configurable user modes are better suited for inherently complex applications that are designed for a variety of different use cases. [5] The user mode approach, just like modularization, is an effective way of

minimizing the user's initial feature fatigue. Yet, it is crucial to point out that the idea of configurable user modes as a solution to feature creep has been subject to criticism. One can argue that the limited number of actions that can be taken in a specific mode may frustrate certain users. [3]

In terms of interface design, Bishop [35] states three needs that must be fulfilled to in order to create user-friendly interfaces:

1. Being disciplined enough to eliminate unnecessary features.
2. Organizing remaining ones for easy accessibility and use.
3. Transferring complexity away from users to more suitable areas.

4 Research survey

For the research part of this thesis, the aim was to pick a popular social media platform, analyze whether it is subject to feature creep in its current state and if so, offer possible solutions to the problem. To determine this, a survey was made and distributed across several social media platforms to get viewpoints from different user demographics.

There are a few reasons why this approach was chosen:

1. *Abundance of answers:* by selecting a relatively popular platform, more people are likely to have used the platform, leading to an increased number of responses, making the results more meaningful.
2. *Lack of resources:* due to lack of personal resources and experience, a more sophisticated form of research would have been unfeasible and risky to conduct.
3. *Simplicity:* relating to the previous point, an information gathering method – such as a survey – is a relatively straightforward way to gather meaningful information for analysis.

The results of this thesis will not be sufficient to determine the overall prevalence of feature creep in the social media space or provide any groundbreaking scientific findings, but it is a chance to put one of the most used social media platforms to a test and hopefully inspire someone along the way.

4.1 YouTube

The social media platform YouTube was chosen for analysis. YouTube is a video-sharing platform established in 2005. The initial idea behind YouTube was to provide a platform where people could share their "home videos" but the skyrocketing spike in popularity caught the founders off guard. Eventually, YouTube was sold to Google in 2008 [36] and has cemented its place as the leading video-sharing platform. As of April 2024, YouTube was the 2nd most visited website in the world with over 31 billion visits that month. [37]

Over the years, YouTube – like any other platform – has gone through a series of changes in terms of interface design and the set of features it offers. Around the time of YouTube's conception in early 2005, the front page was very stripped-down, consisting of a search bar, popular tags and five "featured" videos. [38] The site was essentially a large database of

videos little extra functionality. Since then, YouTube has introduced a plethora of features, including its own video editor, pay-to-use services, livestreaming, analytics tools, sophisticated recommendation algorithms and more.

4.2 Hypotheses & research questions

Having used YouTube extensively over the past 18 years, I went into this thesis with several pre-existing hypotheses in mind which I believe to be true based on personal experience and public discussion.

1. Desktop (computer) users are more likely to experience feature creep.
2. Age affects the perceived level of feature creep.
3. There are a lot of features whose existence people are not even aware of.
4. The recommendation algorithm has a positive effect on user retention.
5. A considerable portion of YouTube's features remain largely unused by the general population.
6. A considerable subset of YouTube users feel like the platform suffers from feature creep.
7. YouTube's additional platforms and services remain largely unused by the general population.

The following research questions were selected:

1. Does YouTube suffer from feature creep?
2. How well have users adopted YouTube’s current feature set?
3. How do certain features affect user engagement with the site?

The hypotheses and research questions work hand in hand in the sense that the survey is inspired by the hypotheses, and the survey results will be used to answer to the research questions.

4.3 Survey structure

The survey was made with Google Forms. The idea behind the survey was the following:

1. Acquire relevant (non-identifying) information about the respondent.
2. Gather the respondent's subjective feelings about YouTube's current state.
3. Present the respondent with a set of YouTube features and ask them to evaluate…
    a. …how useful they deem them to be.
    b. …how well they have adopted them (feature adoption rate).
    c. …how certain features affect user engagement levels with the site.

The features in item 3 were selected from two different places on the YouTube platform: the front page and the video playback page, both of which are presented in Figure 5 respectively. There were no real criteria for what features were chosen for the survey. Rather, the feature selection was done so that features with varying levels of pre-assumed relevance were selected for evaluation. In other words, the idea was to have a fair of distribution of features; some that were assumed to be obsolete, others that were assumed to be widely used, with most of the selected features being somewhere in between, as the results later will prove.

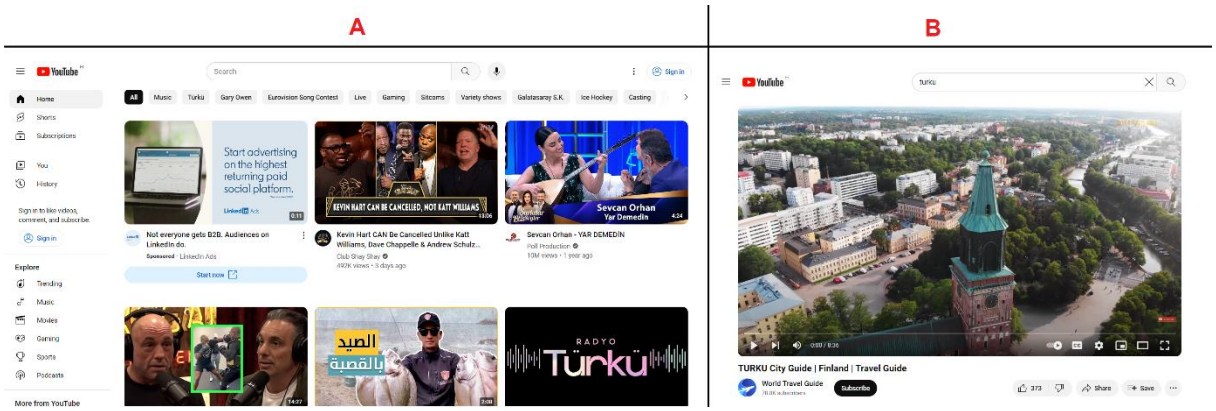


Figure 5: The places on the site where the features were selected: the front page (A) and the video playback page (B).

Different feature sets were selected for points 3a, 3b and 3c. In total, 36 different features were picked for evaluation for the final version of the survey. Each individual part of the survey will be presented in detail from Chapter 4.3.1 to Chapter 4.3.5.

The survey was distributed across several social media platforms with the intention of getting as many replies as possible from people of different backgrounds. Table 2 provides a summary of social media channels to which the survey was distributed. Unfortunately, there was no way to determine the number of responses from each social media platform.

Table 2: The social media platforms the survey was shared on.

Platform	**Additional information**
LinkedIn	The post received 570 impressions as of the survey's closure.
Facebook	Initially shared to ~250 friends with some distributing it further.
Discord	Posted on Asteriski's (UTU CS student union) Discord channel.

4.3.1 Information about the respondent

In the first part of the survey, general questions about the respondent were asked to better understand the demographics when analyzing the results. The questions are presented in Table 3.

Table 3: Questions of part 1 of the survey.

Question	**Type**	**Possible options**
Age	Single selection	18 or under, 19–24, 25–29, 30–39, 40–49, 50–59, 60–69, 70 or above
I have been using YouTube for…	Single selection	15 years or more, 10-14 years, 6-9 years, 3-5 years, Under 3 years
Select the option that best describes your YouTube consumption	Single selection	2+ hours/day, 1-2 hours/day, 0-1 hours/day, Few times/week, Less than once/week
When I use YouTube, I use it on a…	Multi selection	Computer, Smartphone, Tablet, Smart TV, Other[3]

[3] The user could optionally fill in one or more devices in case their device of choice was not mentioned.

These questions were deemed to be sufficient for gathering relevant details about the respondents. No personal details were asked – even the respondent's gender was omitted.

4.3.2 Assessing users' feelings

In the second part of the survey, the respondent was presented with a set of statements that they were asked to evaluate. After evaluating the statements, the respondent was then given the option to list features that they found useful or useless.

The selected statements (see Table 4) were based on the pre-existing hypotheses presented in Chapter 4.2. A Likert scale was chosen as the approach – in other words, the users could choose between s*trongly disagree*, *disagree*, *neutral*, *agree* and *strongly agree*.

Table 4: Questions of part 2 of the survey.

Statement	**Required**	**Type**
From a user experience perspective, YouTube has gone in the right direction over the years.	Yes	Likert (1-5)
Whenever I discover a new feature on YouTube, I am mostly pleased by it.	Yes	Likert (1-5)
Sometimes the abundance of features makes it hard for me to find what I'm looking for.	Yes	Likert (1-5)
I would prefer to use a YouTube Lite with less features over the current version of YouTube.	Yes	Likert (1-5)
In its current state, YouTube has too many features.	Yes	Likert (1-5)
What features do you find useful?	No	Long-answer text
What features do you find useless?	No	Long-answer text

In the fourth question, the concept of YouTube Lite was mentioned. It is worth noting that it is a purely hypothetical concept that does not exist in the real world. With that being said, the concept was briefly explained to the respondent in the survey so that they would have an understanding of what is being talked about. It was explained as "a hypothetical concept of a

lightweight YouTube client with only some of the absolute core features (e.g. video feed, search function, video playback).”

4.3.3 Assessing the usefulness of certain features

The third part of the survey was the first out of three parts where a set of features was presented to the respondent. The respondent was then asked to evaluate how useful they deemed the given feature to be by choosing the option that best described their feelings. The options were:

1. Essential
2. Nice to have
3. Unnecessary
4. I haven’t even heard of this

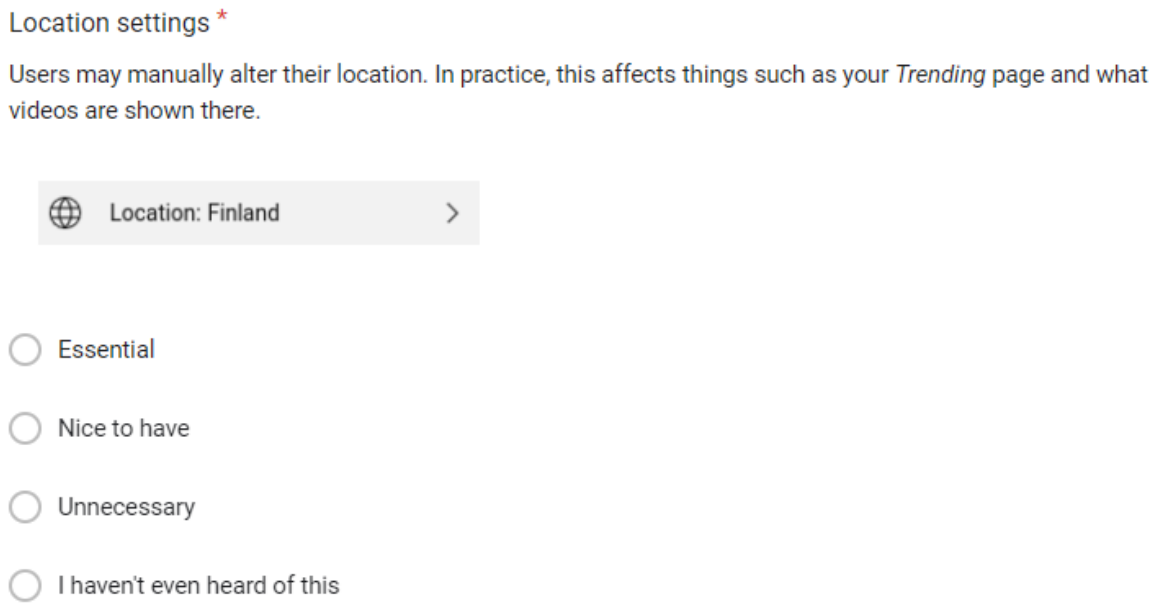


Figure 6: An example of how the feature was presented to the respondent.

As can be seen in Figure 6, there was a brief explanation of what the feature does in cases where it was not obvious. Each question was also associated with a screenshot of the said feature.

Table 5 and Table 6 will present the features were selected for evaluation. All the screenshots will be included in the appendix section of the thesis.

Table 5: Front page view features the respondent was asked to evaluate (n=10).

Feature name	**Explanation (if needed)**
Dark/light theme	
Location settings	Users may manually alter their location. In practice, this affects things such as your *Trending* page and what videos are shown there.
Search with voice	
Search predictions	
Home feed with algorithm-based recommendations	When a user is logged in, the feed tends to consist of videos based on previous watch history.
Home feed filter	Seems to present some AI-generated suggestions for topics that can be used to further filter the home feed.
Infinite scrolling	As the user scrolls down the video feed, new videos are being generated in the feed without explicitly having to click a *Show more* button.
Custom thumbnails	Thumbnail (marked in red) is the mini image whose purpose is to communicate the content of the video in a single image. Content creators may add *custom thumbnails* to their videos in order to attract more viewers.
Video preview	A user may hover over the thumbnail with their cursor in order to preview the video if they wish. The thumbnail turns into a mini player.
Trending videos	Show videos that have gained the most traction recently among users in your country.

Table 6: Video playback view features the respondent was asked to evaluate (n=13).

Feature name	**Explanation (if needed)**
Autoplay	Upon finishing watching a video, the autoplay feature automatically recommends & plays a new video after 10 seconds.
Ambient mode	A lighting effect to make watching videos more immersive by casting gentle colors from the video into your screen's background.
Automatically generated subtitles	AI-generated subtitles that are more or less accurate depending on a) the language b) the clarity of the audio track and c) pronunciation.
Uploader submitted subtitles	Instead of relying on AI-generated subtitles, content creators may upload the actual script of the video which is therefore guaranteed to be correct.
Subtitle customization	
Multiple audio tracks	Content creators with bigger resources may hire voice actors to narrate the original script in various languages, similar to how it works in traditional movies.
Playback speed adjustment (with preset values)	
Playback speed adjustment (by providing a custom value, e.g. 1.45 instead of 1.5)	
"Most replayed"	The grey bumps on the timeline indicate the most replayed parts of a video based on the behavior of other viewers.
Video chapters	A video can be divided into chapters that are either user or computer-generated.
Clip	The "clip" feature allows you to take small (e.g. 10 second) snippets of the video that can then be shared with others.
Keyboard shortcuts	A collection of keyboard combinations that can be used to perform different operations during video playback.
Livestream: separation between "Live chat" and "Top chat"	

4.3.4 Assessing the adoption rate of features

In the fourth part of the survey, the respondent was presented with a different set of features. This time the respondent had to assess how well they have adopted a feature by choosing the option that best described their feelings. The available options were the following:

1. I actively use this feature
2. I sometimes use this feature
3. I never use this feature

First, two features that do not require the user to be logged in were presented (see Table 7).

Table 7: The first two features to evaluate.

Feature name	**Explanation (if needed)**
Watch queue	May be considered to be a "disposable playlist" in the sense that the "watch queue" is not made to be a long-term solution for video storage.
Automatically generated mixes	A non-stop playlist tailored by your personal watch history.

After this, the user was asked if they are logged in to YouTube while using the platform. If the respondent answered *Yes*, they were presented with more features to evaluate. In the case of a *No* answer, the user was directly taken to part 5 of the survey. The rest of the features (see Table 8) require the user to be logged in to use.

Table 8: The remaining features to evaluate for those who used YouTube while logged in to the platform.

Feature name	**Explanation (if needed)**
Subscription feed	As the name implies, the subscription feed works similarly to the regular front page feed but shows the latest videos from the channels the user has subscribed to.
Subscription feed view customization	The option to display the subscription feed in list/grid mode.
Watch history	Shows your watch history from the most recently viewed video to the oldest.
Liked videos	A collection of videos the user has "liked".
Playlists	A user may create custom playlists that can house an arbitrary number of videos. Playlists can be used for any purpose, but typically they revolve around a specific topic, e.g. *programming 101 lecture* or *funny cat videos.*
Collaborative playlists	Regular playlists can be turned into *collaborative playlists*, which essentially turns the playlist into a shared playlist among two or more YouTube users.
“Watch later”	Videos can be added to this list for later viewing.
YouTube Shorts	YouTube's own short-form video service where users can upload short videos up to 60 seconds in length.
YouTube Premium	A subscription service offered by YouTube that offers benefits such as ad-free viewing, ability to download videos and access to other services such as YouTube Music. The list of benefits can be seen here.
YouTube Music	Offers a Spotify-esque client that gives access over 100 million songs, videos and live performances. The regular plan is 9,99€/month, though a free version is available. Read more about the differences here.
YouTube Movies	A service that allows users to buy or rent movies for viewing.

4.3.5 Assessing features' effect on user engagement

Finally, in the last part of the survey, the respondent was again asked to evaluate a set of statements (see Table 9) based on their personal feelings. The aim of this part of the survey was to figure out how certain features affect user engagement levels, and in turn, how they might affect factors such as user retention. The statements were based on the pre-existing hypotheses presented earlier.

Table 9: The statements the respondent was asked to evaluate.

Statement	**Possible options**
I spend more time on YouTube than I should because of the video recommendation algorithm.	Likert (1-5)
I often find myself endlessly scrolling the front page feed with no real purpose.	Likert (1-5)
The video thumbnail affects my decision to watch a video.	Likert (1-5)
The video preview feature affects my decisions to watch a video.	Likert (1-5)
I find myself skipping videos that have ads at the start or in the middle.	Never/Sometimes/Often
I engage in public discussion on YouTube in the form of video comments, livestream chat or such.	Never/Sometimes/Often
I observe the public discussion on YouTube in the form of video comments, livestream chat or such.	Never/Sometimes/Often

5 Results

The survey was released to the public on the 8th of March 2024 and was closed the 17th of April 2024, amassing a total of 44 responses. This was deemed as a sufficient sample size for the study. For the inclusion criteria, responses were wanted from active users of YouTube, preferably from those who use the desktop version on PC. However, there was no way to truly confirm the respondents' background as the survey was public and anonymous.

5.1 Preliminary statistics about the respondents

While the survey was anonymous, some information was gathered from the respondents, such as age and consumption habits. In order to conduct a legitimate analysis of the results, one must take a closer look at the respondent demographics. Having a firm grasp of what kind of people have answered the survey will help identify possible biases in the results, which in turn yields a more accurate conclusion.

As was stated before, a total of 44 responses were received, and every response was considered acceptable. With public surveys, there is always a risk of receiving responses from people with malicious intentions. A positive attribute about this research topic is that it is very apolitical by its nature, meaning that people have a lower motivation to intentionally skew the results.

The age distribution of the respondents is presented in Table 10. Ages 19-29 accounted for 68% of the responses. The most popular age group among respondents was ages 25-29 (n=17) with age group 19-24 coming in second (n=13). Only one respondent was 18 or under. The oldest respondent belonged to the age group 50-59.

Table 10: The respondents' age distribution.

Age	**18 or under**	**19-24**	**25-29**	**30-39**	**40-49**	**50-59**	**60-69**	**70 or above**
N	1	13	17	8	4	1	0	0

The respondents were asked to assess their YouTube consumption habits. Over half (n=24) claimed to have used the platform for 15 years or more (see Table 11). 18 respondents claimed to have 10-14 years of experience with the platform, with only 2 people claiming 6-9 years of experience. None of the respondents under 6 years of experience with the platform.

Table 11: Response distribution to the question: "I have been using YouTube for...".

Selection	15 years or more	10-14 years	6-9 years	Under 6 years
N	24	18	2	0

Roughly 4 out 5 (80%) of the respondents were daily users of the platform (see Table 12). Even the remaining respondents – apart from one – could still be considered as active users. Indeed, only one of the respondents claimed to use the platform less than once a week.

Table 12: Response distribution to the question: "Select the option that best describes your YouTube consumption."

Selection	2+ h/day	1-2 h/day	0-1 h/day	Few times/week	Less than once/week
N	12	13	10	8	1

Lastly, the respondents were asked to express which platform(s) they use YouTube on, the results of which can be seen in Table 13. Out of 44 respondents, 37 (~84%) used the platform on a computer, which was the desired platform from an analysis standpoint. The question was a multiple-choice question, although it was not explicitly stated. The vast majority of the respondents seemed to get it – the options were squares instead of circles, which typically indicates a multiple-choice question.

Table 13: The respondents' choice of platform(s) for using YouTube.

Platform	Computer	Smartphone	Smart TV	Tablet	Gaming console[4]
N	37	39	16	4	2

5.2 Subjective feelings

In the first part of the four-part survey, the respondents were presented with a set of five statements whose legitimacy they had to assess on a *Likert scale*. After answering to the

[4] Two of the respondents used the *Other* option to provide a free-form answer, as their platform of choice was not an available option. Respondent 1 answered “Playstation” while respondent 2 answered “Gaming console. ps5, ps4 etc.”. These two responses were ultimately the same, so they were consolidated into a single *Gaming console* for the sake of readability.

statements, the respondents then had the option to list YouTube features that they found particularly useful and/or useless.

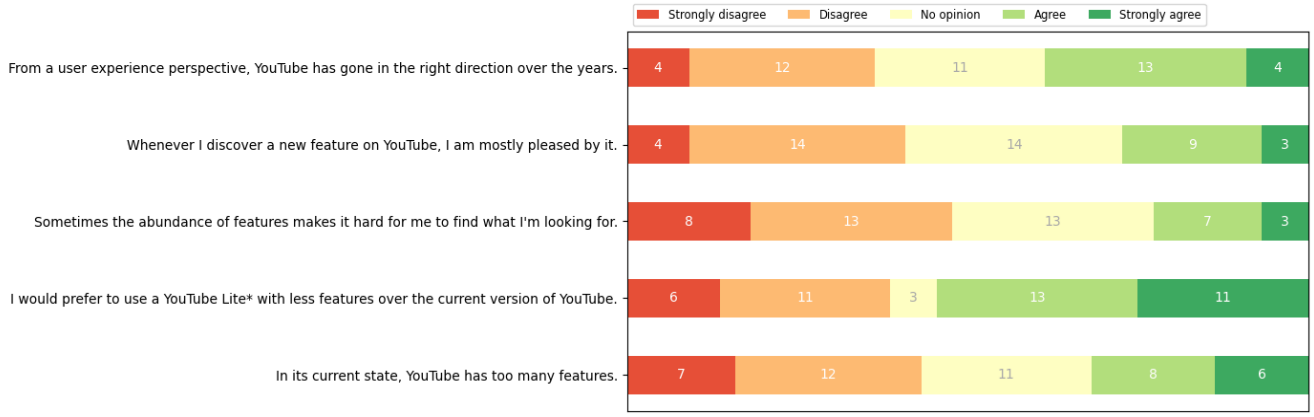


Figure 7: Response distribution for the statements.

A striking aspect in the results in Figure 6 is that a relatively large portion of respondents chose *No opinion* as their answer (apart from question #4), which seems to indicate that the statements do not evoke many feelings in some users.

Only ca. 30% of the respondents think YouTube has too many features, yet over half of the respondents would prefer to use a lightweight YouTube client over the current one. This seems to be in line with the current trend in minimalist approach that is prevalent in visual and UX design. Users may tolerate or even prefer the regular client, but if they had the option to use a lightweight version, majority would opt for that instead. This is not to say that Google ought to release a lightweight client this instant; users might initially think that a “lite” version would suit their needs better, only to later find out that they do indeed need many of the features present in the regular client. Nevertheless, it would be interesting to see how widely adopted a possible YouTube Lite would be.

Just 1 in 4 respondents express some form of satisfaction when they discover a new feature on YouTube, with 40% of the respondents expressing dissatisfaction in a similar situation (question #2). Despite this, users do not seem to suffer from feature fatigue at least for now (question #3). Only ~23% of the respondents experience difficulties finding what they are looking for due to an excess number of features, with half of the respondents stating that they do not have such problems. For now, it seems that YouTube’s current feature set is laid out so that it does not pollute the user interface.

When asked to provide concrete examples of useful and/or useless features on YouTube, some recurring themes were present.

Table 14: Recurring themes in the optionally provided features that users deemed useful and/or useless.

Useful	**Useless**

• Recommended videos
• Subtitles
• Playback speed adjustment
• Different types of playlists
  o Watch history
  o "Watch later"
  o Custom playlists
	• YouTube's other platforms
	  o Shorts, Music, Premium
	• Trending videos
	• Ads
	• Autoplay

As seen in Table 14, users found different types of video categorization tools valuable. The ability to create own playlists and to save videos for later viewing was viewed in high regard. The video recommendation algorithm also gained praise among the respondents. When watching a video, users seem to value the ability to manually adjust the playback speed to their liking particularly in longer videos. Subtitles also received its fair share of appreciation, as occasionally users might not have access to audio. Also, due to YouTube's international nature, many videos are simply narrated in foreign languages.

Users did not seem to approve YouTube's other platforms. This is most likely due to the fact that they are not seen as noteworthy competitors in their respective fields; the average user likely prefers TikTok over YouTube Shorts, Spotify over YouTube Music and Netflix over YouTube Movies. Many major players in the realm of social media seem to have this massively ambitious idea of providing a single do-it-all application that serves all user needs. A lot of time and effort is spent on realizing these grand visions, only to find out that it mostly results in feature creep and lacklustre adoption.

5.3 Perceived usefulness of features

In the second part of the survey, the respondents were presented two sets of features. The features of the first set (n=8) were picked from the YouTube front page, while the features of the second set (n=13) were picked from the video playback view.

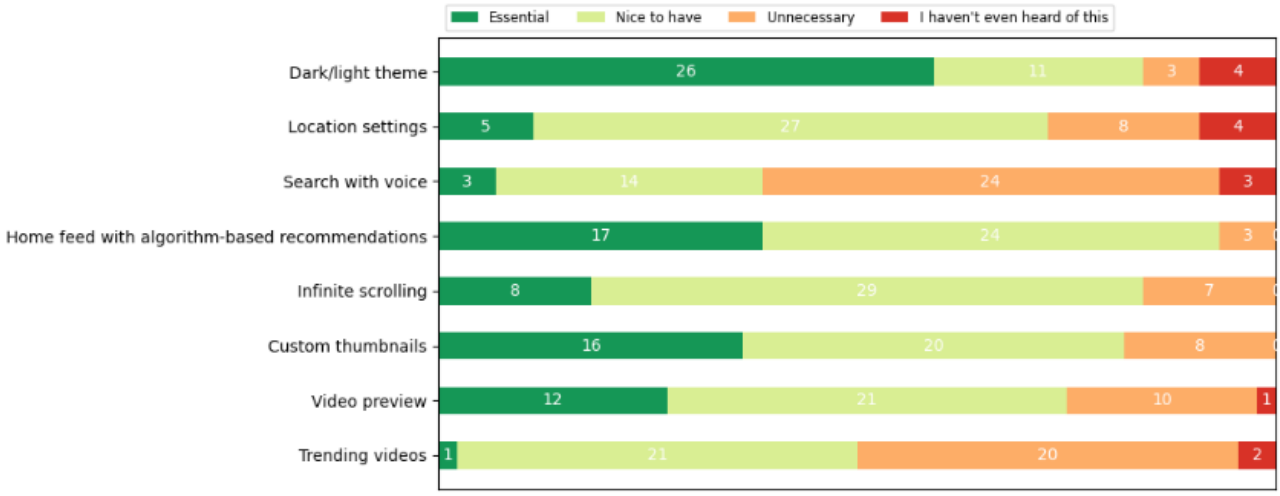


Figure 8: How useful the respondents found YouTube's front-page features to be.

In Figure 8, one can see how the respondents evaluated the usefulness of YouTube's front-page features. First of all, given the infrequence of the red bar in the chart, one can conclude that the users seem to be, above everything, *aware* of the existence of the chosen features, which in itself is already a positive note.

What's also striking is the prevalence of the light and dark green bars – users seem to mostly approve of the features on the front page, or at least consider them "nice to have", meaning that majority of the features are not seen as futile. In fact, there are only two real opinion dividers: *trending videos* and *search with voice*.

An indispensable feature seems to be the ability to switch between dark & light modes. What's interesting is that this feature in general is relatively new across all systems, having reached mainstream adoption in the late 2010's. Regardless, users have already adopted it as one of the *must-have* features.

When it comes to assessing the usefulness of features such as the ability to search with voice, it is good to note that while the average individual might not find value in such features, they may exist simply for the sake of accessibility, and not just so show off. As a popular platform, YouTube must cater to people with special needs.

Continuing with the second feature group, which included features from the video playback page, one can detect some variation in Figure 9.

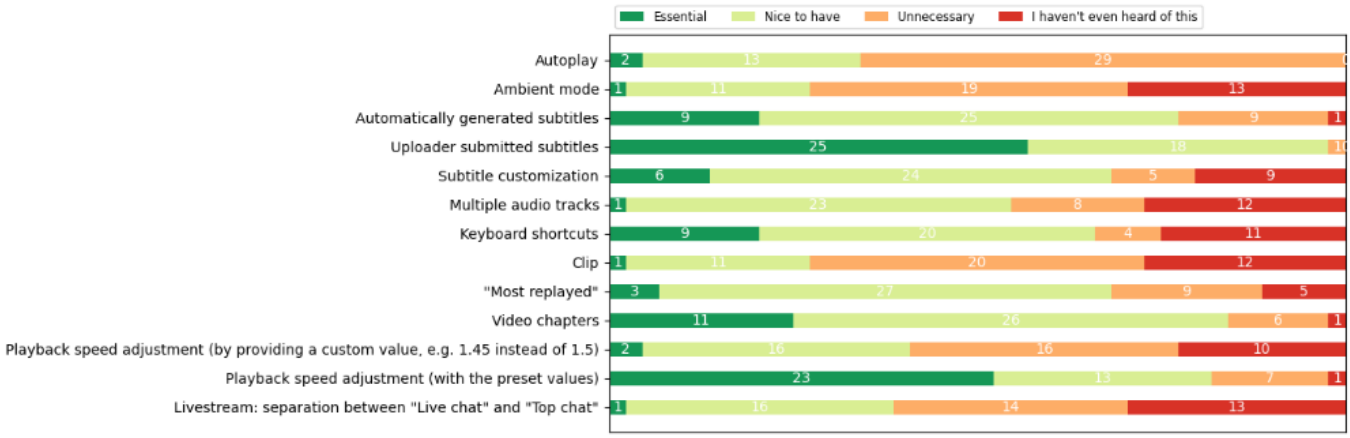


Figure 9: How useful the respondents found the video playback view's features to be.

Just like in Figure 8, there are a couple of features that stand out due to their usefulness. Correlating directly with the written responses presented in Table 14, one can see that subtitles and playback speed adjustment are found valuable. Still, only these features received an overwhelmingly positive response. Once again, most of the features were considered to be *nice to have*. In comparison to Figure 8, however, the prevalence of the red bar is a lot more pronounced. In practice this means that a large portion of users are not even aware of the available features. In fact, when looking at the average response presented in Figure 10, one can see that on average, the existence of a given feature was not even acknowledged by roughly 15% of the users.

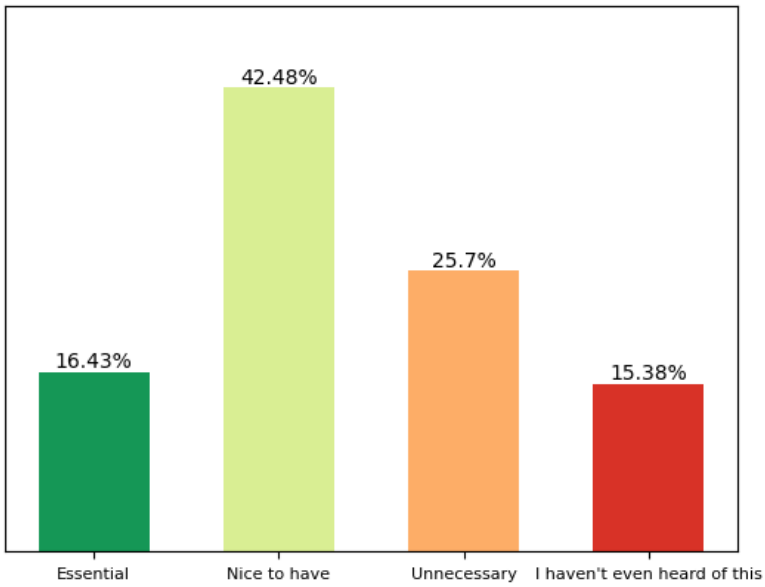


Figure 10: The average response distribution for features in the video playback page.

Moreover, the sorted table in Figure 11 shows that 5 out of the 13 features were considered more unnecessary than necessary by the respondents. Still, it is worth noting that each feature has a significant portion of users that consider the feature in some way necessary.

	Necessary	Unnecessary
Uploader submitted subtitles	43	1
Video chapters	37	7
Playback speed adjustment (with the preset values)	36	8
Automatically generated subtitles	34	10
Subtitle customization	30	14
"Most replayed"	30	14
Keyboard shortcuts	29	15
Multiple audio tracks	24	20
Playback speed adjustment (by providing a custo...	18	26
Livestream: separation between "Live chat" and ...	17	27
Autoplay	15	29
Ambient mode	12	32
Clip	12	32

Figure 11: A simplified table representing the level of necessity of each feature as rated by respondents.

5.4 Feature adoption

In the third part of the survey, respondents were presented a set of features and were asked to assess their interaction frequency with the feature. In total, 13 features (see Table 7 & Table 8) were chosen for evaluation. The first set of features (see Figure 12) contains some general YouTube features. The second set (see Figure 13) assesses the adoption of YouTube's other platforms.

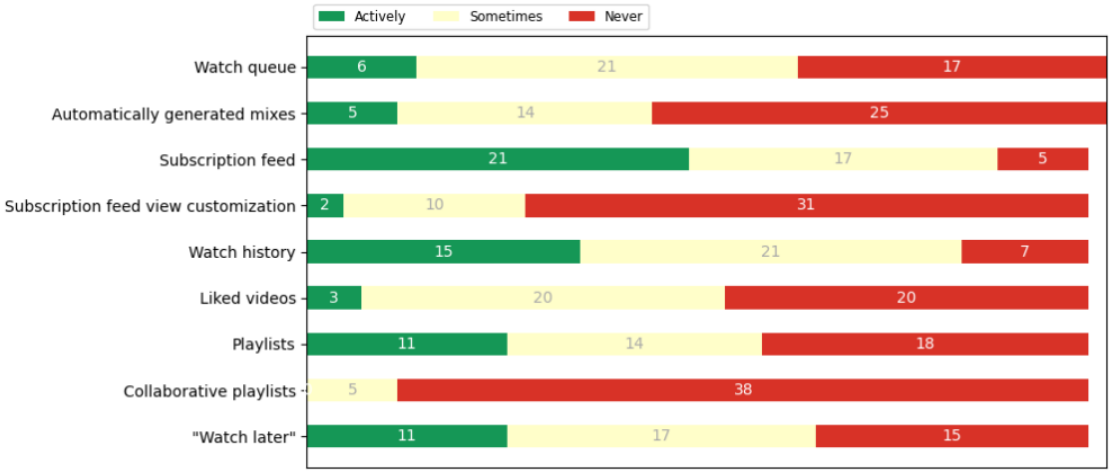


Figure 12: Adoption of general features among respondents.

When the respondents were asked what features they found useful (see Table 14), many found different types of playlist tools useful. Indeed, Figure 12 seems to support these claims,

considering the adoption rate of features such as the subscription feed or watch history, which positively stand out from the rest of the features in terms of their adoption rate.

Continuing to the evaluation of YouTube's other platforms, hardly any surprises are present in the responses seen in Figure 13.

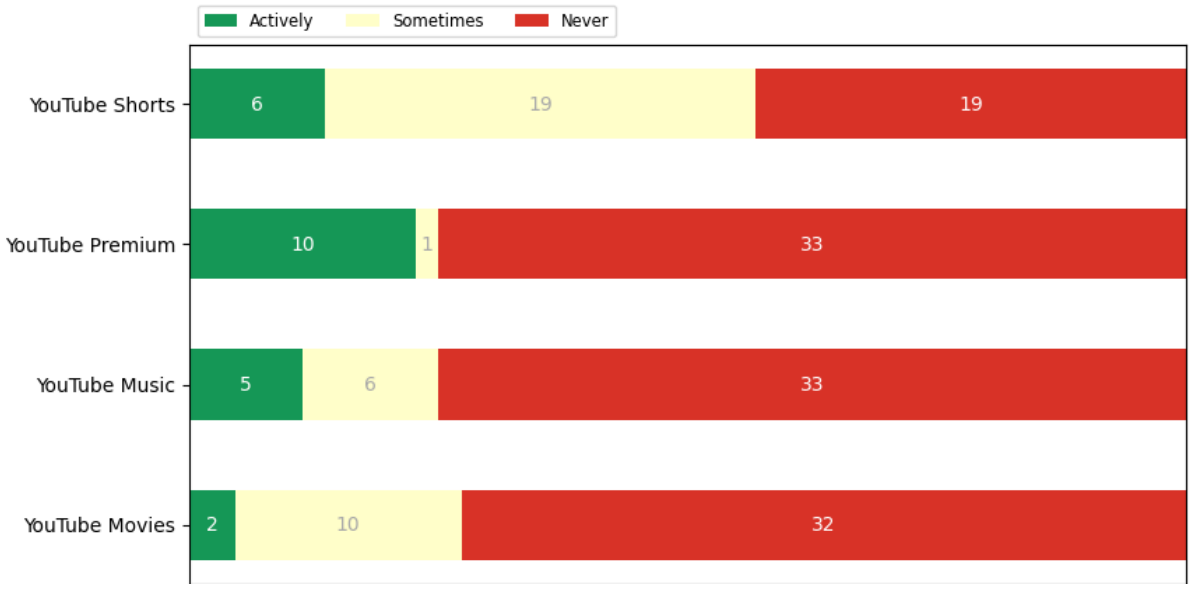


Figure 13: Adoption rates of YouTube's other platforms.

The platforms presented in Figure 13 – apart from YouTube Shorts – remain largely unused. Additionally, approximately 23% of respondents claim to use YouTube Premium, which could be explained by the bias among respondents (young, tech-savvy, active users of the platform). With that said, it is not surprising that such users would opt to use some of the more advanced features the platform has to offer.

5.5 Features' effect on user engagement

In the last part of the survey, the aim was to find out how effective certain features are at influencing user decision making on the site and how valuable do users find those features to be. One of the hypotheses going into this survey was that certain features exist to keep the user hooked on the site. Asking the end users directly is a good way to query their effectiveness.

The respondents were presented with seven statements whose legitimacy they had to assess. For the first four statements (see Figure 14), the Likert scale was used for evaluation. First, three front-page features that were thought to heavily affect user engagement were put under scrutiny: the recommendation algorithm, endless scrolling and video thumbnail.

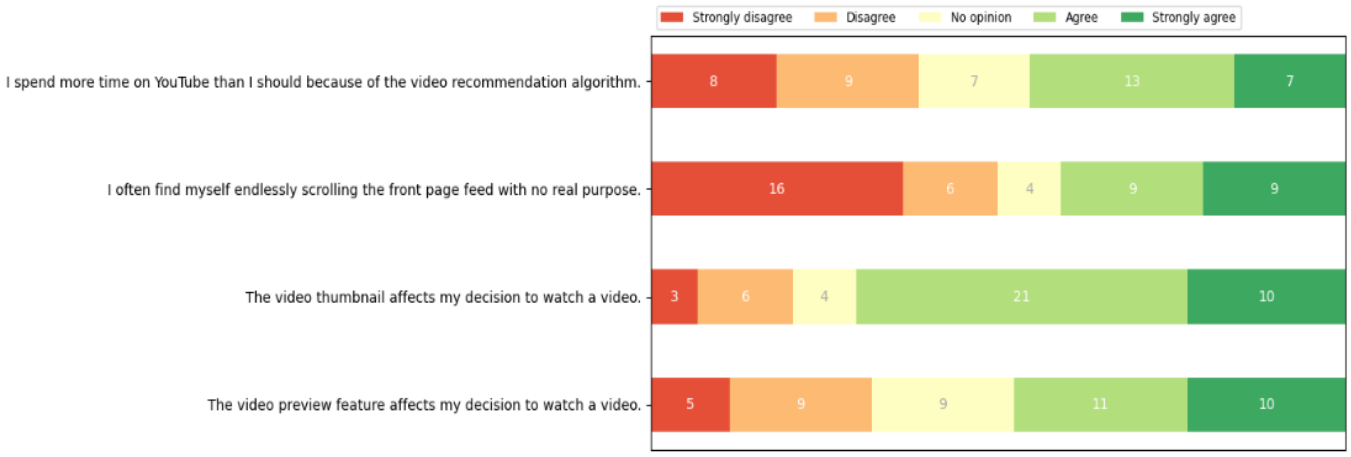


Figure 14: Assessment of the effectiveness of the recommendation algorithm, endless scrolling and the video thumbnail.

The effectiveness of the recommendation algorithm seems to divide respondents, with a slight majority admitting that the recommendation algorithm makes them spend more time on the platform. For further research, it would be interesting to study what kind of individuals are more prone to being "hooked" by the recommendation algorithms on social media platforms. Somewhat surprisingly, a large portion (around 1 in 3) of respondents "strongly disagree" with the idea of regularly engaging in pointless auto scrolling (sometimes referred to as *doomscrolling*), although the results still imply that a considerable subset of users seem to fall prey to the feature.

The remaining statements (see Figure 15) were somewhat auxiliary, and they were mainly curated with the idea of figuring out how users feel about the social aspect of YouTube, though a question about ads was also included. Initial analysis of the results indicate that the vast majority of people do not participate in the social aspect of YouTube. Moreover, more than half of users skip videos with ads at least occasionally.

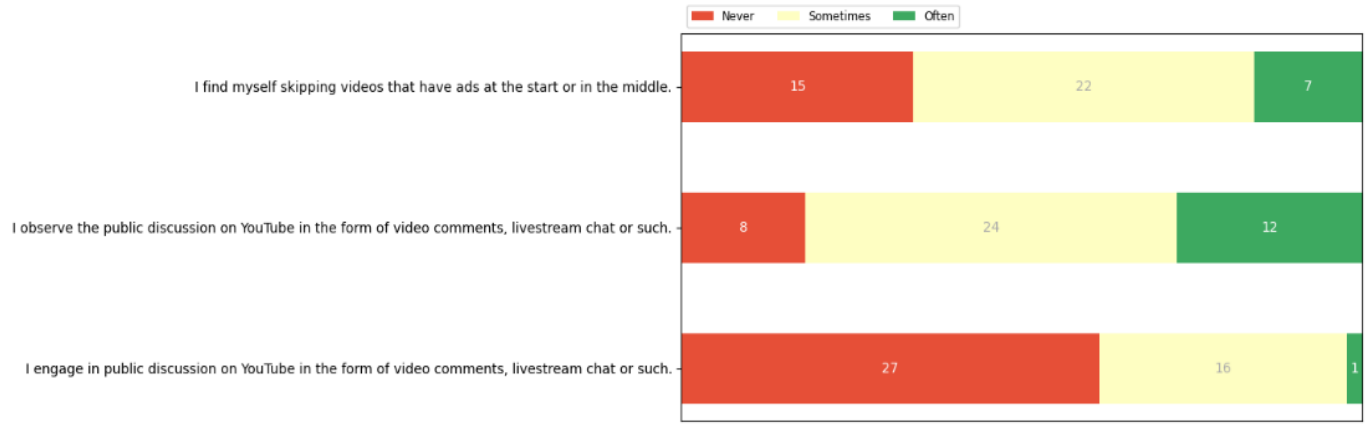


Figure 15: The respondents' thoughts on ads and the social aspect of YouTube.

6 Analysis

Having presented the survey results via visual and literary means, a logical next step would be to delve deeper into the findings and possibly detect new patterns. In this chapter, we will recall the previously presented research questions and see how well they were answered and how well the pre-existing hypotheses hold up against the results.

6.1 RQ1: Does YouTube suffer from feature creep?

Based on the obtained results as a whole, there seems to be no indication that YouTube suffers from feature creep despite its extensive set of features. This conclusion was reached by evaluating the average opinion across all questions. Regardless of this outcome, there are some interesting details that ought to be brought to light.

Going back to the answers presented in Figure 7, one can see that the results are relatively evenly scattered – there is no clear consensus to be detected. Respondents did not seem to experience feature fatigue (question #3), although the majority would still prefer to use a lite version of YouTube with fewer features. Moreover, when the respondents were indirectly asked whether they feel like YouTube suffers from feature creep in its current state (question #6), only 1 in 3 agreed with the sentiment. Despite the fact that the majority of the respondents did not feel YouTube suffers from feature creep, only 27% of respondents were pleased when they discovered a new feature on YouTube. Nonetheless, based on the responses obtained and presented in Figure 7, one could not, with good conscience, claim that YouTube suffers from feature creep in its current state.

Going forward in the survey, when the respondents were asked to evaluate the usefulness of the two different sets of features, the most common answer – by a quite large margin – was “nice to have”, with the mean frequency for the selection being 47.5% and 42.5% for the front page and video playback page features, respectively. Indeed, even across the major age groups (18-24, 25-29, 30-39), there is little to no variation in the answers (see Figure 16 and Figure 17).

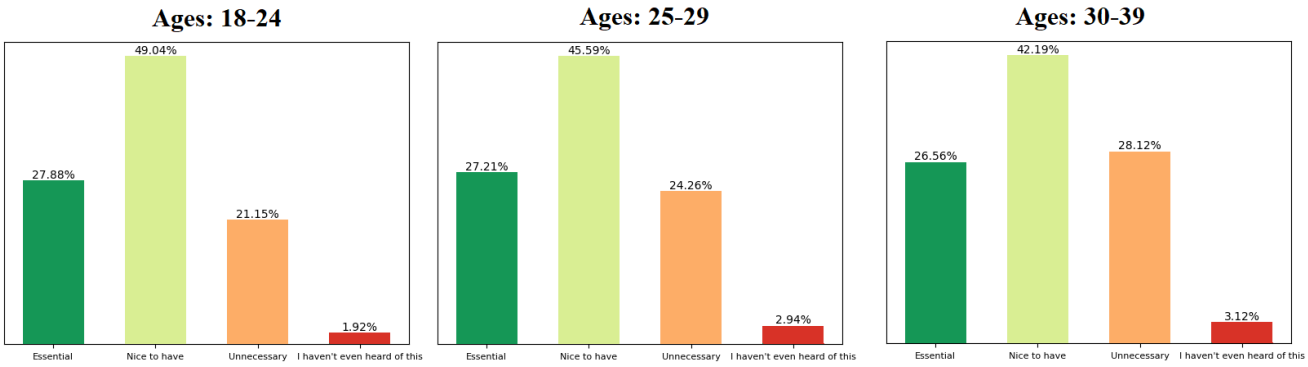


Figure 16: Distribution of responses across the major age groups for feature set #1 (front page).

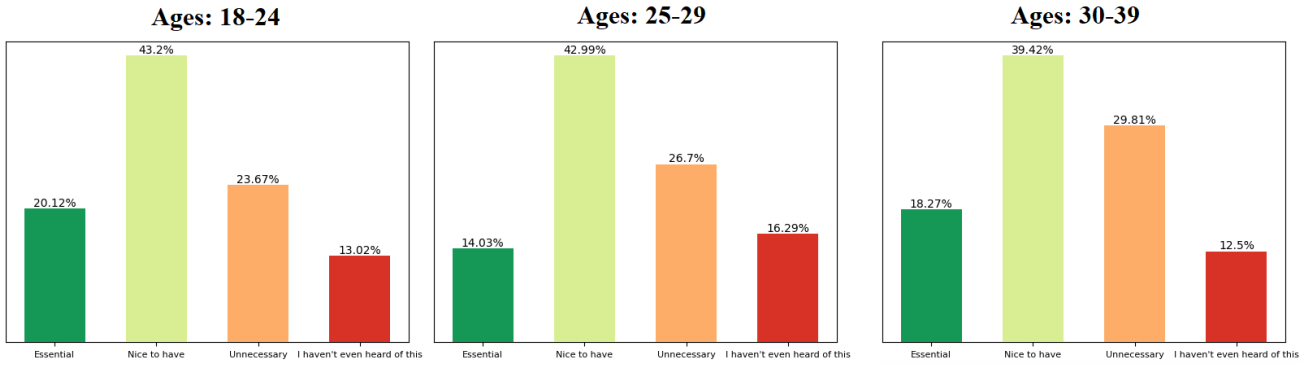


Figure 17: Distribution of responses across the major age groups for feature group #2 (video playback page).

Despite the subtle difference in the distribution for the second feature group in the oldest age group, age does not seem to affect the level of perceived feature creep. The mean response across different age groups is hauntingly similar. The front page features are viewed in positive light by users in all age groups, with more variation being present for the evaluation of the video playback page features. Regardless “nice to have” was the dominant response for both feature sets in all major age groups.

Going into this research, one pre-existing hypothesis was that desktop users were more likely to experience feature creep. This was based on the notion that user interfaces in other platforms (mobile phone, smart TV, etc.) appear to be more unembellished on the outside. However, the findings of this survey do not seem to support that claim. When comparing the responses between non-computer (n=7) and computer groups (n=37), no noteworthy differences in the responses were found. At best, only certain individual features were rated differently by the two groups, but no notable differences in response patterns were to be seen

between the two groups. For further analysis, a larger sample size of non-computer users would be ideal.

Based on the results presented in Figure 16 and Figure 17, the following statistical findings emerge across all features in both feature sets and all major age groups:

Table 15: Statistical analysis of the perceived importance of YouTube's features.

Importance	**Essential**		**Nice to have**		**Unnecessary**		**Unheard**	
Low/High	*14,03%*	*27,88%*	*39,42%*	*49,04%*	*21,15%*	*29,81%*	*1,92%*	*16,29%*
Median	*23,34%*		*43,10%*		*25,48%*		*7,81%*	
Mean	22,34%		43,74%		25,62%		8,3%	

As the previous findings combined with the analysis provided in Table 15 suggest, on average, approximately 1 in 3 features are considered unnecessary by users. While this is not a direct indication of feature creep in itself, it indicates that there is a high likelihood that the current feature set of YouTube contains many features whose importance could be reassessed. In particular, the video playback view seems to contain features of whose existence users are simply not even aware of, which is unlikely to have been the original vision the developers had in mind. Nonetheless, there seems to be a recurring pattern which, with rough approximation and for the sake of simplicity, could be referred to as the *20/50/30 pattern*: 20% of features are seen as essential, 50% are seen as useful and 30% are seen as unnecessary. It is unclear how well this pattern holds for other social media applications, as research on the topic is limited.

6.2 RQ2: How well have users adopted YouTube's current feature set?

The features of YouTube appear to have a varying adoption rate. There are a few outliers among the selected features when it comes to their adoption. *Subscription feed* and *watch history* are used to some extent by a large portion of users; 88% and 84% respectively. On the contrary, *collaborative playlists* and *subscription feed view customization* appear to receive very little love from the users, with 88% and 72% of users respectively saying they never use the features. Figure 18 displays the mean response across all features, with the left one showing all 9 features and the right one with the outliers excluded from the calculation.

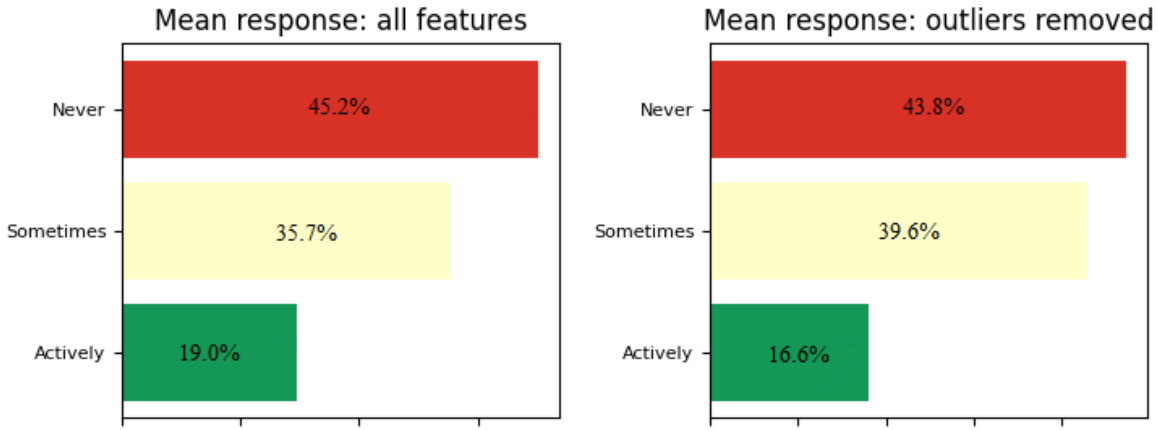


Figure 18: Mean responses for all features (n=9) and the outliers removed (n=5).

As can be seen in Figure 18, the average responses hardly differ. However, a point worth raising is the prevalence of the "Never" response. Approximately 45% of users never use a given feature, with only one feature having "Actively" as the most dominant response: the subscription feed. Moreover, based on the graphs in Figure 19, it appears that people who consume YouTube frequently (daily) use the platform's features noticeably more extensively than those who use the platform less frequently (less than daily).

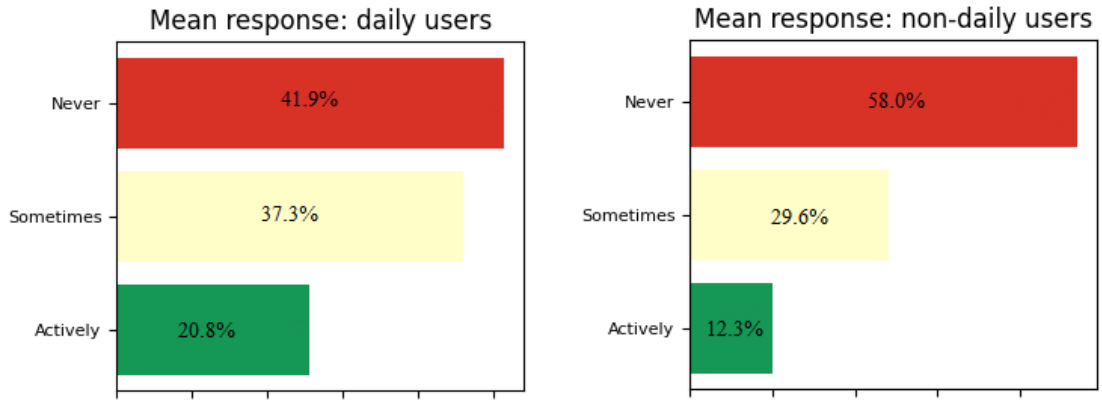


Figure 19: Differences in feature adoption between frequent (daily) and non-frequent (<daily) users.

In all fairness, despite the share of users never using a given feature, most features still have their users. In fact, Figure 20 shows that if "Actively" and "Sometimes" were to be combined into one uniform selection, only 3 out of 9 features would have "Never" as the most dominant answer, yet again contributing to the notion that despite many people not being aware or simply not using many of YouTube's features, most still have their fair share of users.

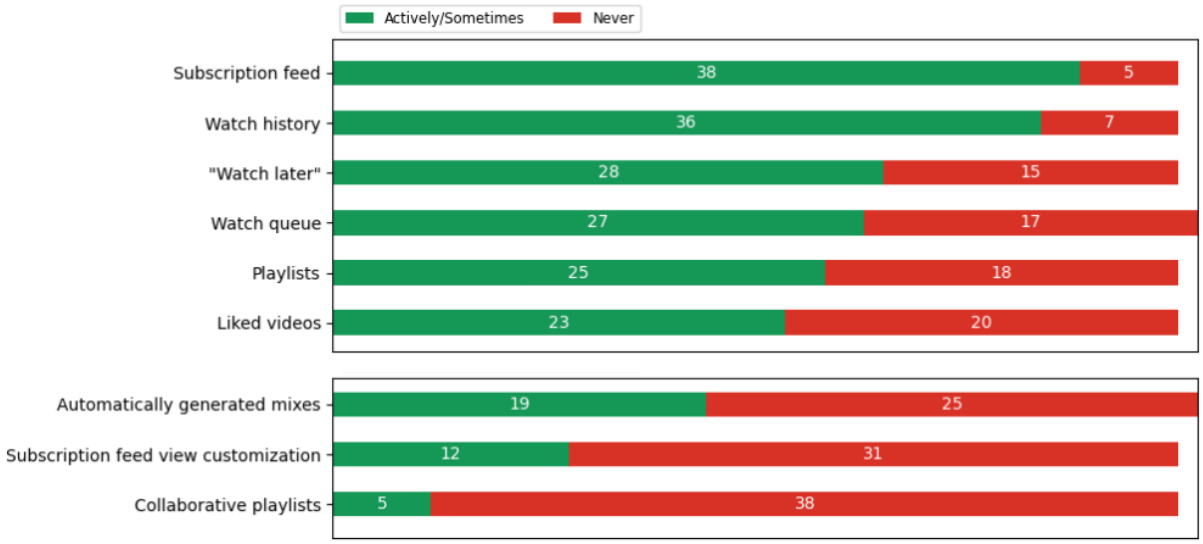


Figure 20: When "Actively" and "Sometimes" are combined into a single answer, one can see that only three features have "Never" as the dominant answer. [5]

Lastly, as was seen in Figure 13, it appears that YouTube's own services have not been well adopted by users. This is likely due to the fact that seemingly much better alternatives already exist on the already competitive market but could also be the cause of unsuccessful marketing.

Table 16: The respondents' assessment of their usage of YouTube's auxiliary services.

	Actively		**Sometimes**		**Never**	
YouTube Shorts	*N*	*%*	*N*	*%*	*N*	*%*
	6	14%	19	43%	19	43%
YouTube Premium	*N*	*%*	*N*	*%*	*N*	*%*
	10	23%	1	2%	33	75%
YouTube Music	*N*	*%*	*N*	*%*	*N*	*%*
	5	11%	6	14%	33	75%
YouTube Movies	*N*	*%*	*N*	*%*	*N*	*%*
	2	5%	10	23%	32	73%

The data presented in Table 16 raise a few points that ought to be brought to light. Over half of the respondents use YouTube Shorts to some extent. In addition to being well promoted on the platform, media consumption across all social media has been shifting towards short

[5] Two of the features have one extra response for reasons explained in Chapter 4.3.4.

videos over the years, which likely contributes to its relative popularity. Moreover, approximately 1 in 4 use YouTube Premium. This could reflect a bias among respondents; in Chapter 5.1, it was stated that most respondents were active and experienced users. With that said, it is not surprising that such users would opt to use some of the more advanced features the platform has to offer.

6.3 RQ3: How do certain features affect user engagement with the site?

It appears that the effect of two common social media features – the recommendation algorithm and "doomscrolling" – do not appear to be as strong as expected. This finding is somewhat conflicting with the fourth hypothesis of this survey, which stated that the recommendation algorithm in particular is expected to have a positive effect on user retention. Indeed, a specific subset of users do find that the algorithm plays a role in their behavior on the site, but this subset represents less than half of the users. However, when talking about user retention rate, something that must be mentioned is the video thumbnail feature, which appears to play a major role in whether a user decides to watch a video or not. In particular, having a human face on the thumbnail has a great effect on the click-through rate (CTR)[6]. A 2020 study showed that in their sample group, 72% of thumbnails contained a human face in them. The ones with a human face averaged almost one million more views that those that did not have a human face in them. [42] An interesting observation is that the recently added video preview feature does not seem to affect viewing decision as strongly, but the results still imply that it plays a significant role.

It was observed that most users do not participate in public discussion. A majority of respondents do observe the discussion, but do not actively participate. Indeed, only 1 respondent out of 44 claimed to "often" engage in public discussion on YouTube, with over half of the respondents saying they never do so. This ratio seems to be in line with other studies as well – a 2021 study found that 25% of Twitter (present-day X) users produce a staggering 97% of all tweets. [43] Finally, as was previously expressed in the written answers, adverts that are played during video playback were mainly seen as a nuisance, with over half of the respondents stating that they sometimes skip videos with ads in them.

[6] Click-through rate (CTR): number of clicks divided by the number of impressions. [40] It is a measure of how effective the thumbnail is at converting impressions into clicks.

7 Conclusion

While no definite proof was found that would suggest YouTube suffers from feature creep in its current state, many other noteworthy observations were made. It is fair to state that while YouTube offers an extensive set of features, they have done a great job at designing the system such that it does not exhaust its users. In its current state, YouTube seems to have only a handful of features that might be considered obsolete. Other than these outliers, most features had their loyal users. Relating to this, the 20/50/30 pattern was discovered, which indicates that as a rough estimate, 20% of features are seen as necessary, 50% as useful and 30% as unnecessary. Further analysis is needed to understand whether this same pattern can be applied to other systems.

In retrospect, more effort should have been put in the survey design. This topic has a lot of potential but, alas, as this was my first attempt at research, some mistakes were made. There are several things that could have been done better in this research:

1. **Wider range of different demographics**

When considering the results of this research, an important point worth reiterating is that the respondents were mostly tech-savvy young individuals with extensive experience with the platform, meaning that they likely had a strong pre-existing mental map of the platform. For more accurate results, a wider range of different demographics should have been present, namely users from the far ends of the age spectrum and those who have less experience with the platform. This would have given a better sample size for analyzing whether attributes like age or previous experience with the platform affects the perceived level of feature creep.

2. **Compare only two devices (e.g. desktop vs. TV)**

Only after releasing the survey did I realize how big of a mistake it was to allow respondents to select more than one device that they use YouTube on. From a scientific standpoint it would have made more sense survey e.g. two groups that use the application *exclusively* on a given platform. This would have made it easier to compare the perceived level of feature creep across different devices, which was one of the questions that intrigued me personally.

3. **Go easy on the scope**

As ironic as it sounds, I felt like the survey (and coincidentally, this thesis) suffered from scope creep in a sense. When considering the scope, I felt like the last portion of the survey

(features and user interaction) could have been omitted altogether. The point was to figure out how certain features affect user engagement with the site. While the question is interesting and certainly worth studying from a business perspective, I feel like it wasn't that relevant in the context of feature creep.

What about the future? For one, it would be interesting to see how well the proposed 20/50/30 pattern applies to other platforms and systems as well, if at all. Additionally, it could be worth studying how companies like Google manage to develop and design elegant systems such as YouTube *without* introducing feature creep along the way; what exact design patterns, tricks and philosophies contribute to this effort? Lastly, one major area of the platform that was completely omitted in this thesis was YouTube Studio, which is a tool specifically for content creators for managing, editing and monitoring the performance of their content. One possible future endeavor would be to survey content creators exclusively to see how they rate the user interface from a usability perspective and determine whether feature creep is present there.

Since the results have now been presented and analyzed Chapter 5 and Chapter 6 respectively, it would be a good time to return back to the hypotheses that served as the basis for the entire survey and see how well they held true. Table 17 concludes this work by putting each hypothesis is scrutiny.

Table 17: Assessment of the hypotheses.

Hypothesis	**Verdict**	**Comment**
Desktop (computer) users are more likely to experience feature creep.	False[7]	Although only 7 out of 44 respondents were non-desktop users (a somewhat limited sample size), there was no noteworthy difference in the perceived level of feature creep.
Age affects the perceived level of feature creep.	False[8]	The findings presented in Figure 16 and Figure 17 suggest that there seems to be no noteworthy correlation between age and perceived level of feature creep.
There are a lot of features whose existence people are not even aware of.	True	This became apparent when users were asked to assess the features of the video playback view (see Figure 9).
The recommendation algorithm has a positive effect on user retention.	Unclear	There appears to be a subset of users (<50% users) who feel like the recommendation algorithm dictates their behavior on the site. Further analysis is needed.
A considerable portion of YouTube's features remain largely unused by the general population.	True	The findings in Chapter 6.2. suggest that a considerable portion of users never use most of the features the platform has to offer.
A considerable subset of YouTube users feel like the platform suffers from feature creep.	Unclear	Depends on what is seen as “considerable”. Based on the chart presented in Figure 7, approximately 1 in 3 users feel this way.
YouTube's additional platforms and services remain largely unused by the general population.	True	The chart presented in Figure 13 proves that these platforms and services have generally low adoption rates.

[7] The vast majority of users were desktop users. For more accurate results, more responses should have been acquired from users who exclusively use the platform on another device than desktop.
[8] This conclusion could be challenged by having a more evenly distributed sample group (by age).

References

[1] D. Thompson, R. Hamilton, and R. Rust, “Feature Fatigue: When Product Capabilities Become Too Much of a Good Thing,” *J. Mark. Res. Am. Mark. Assoc. ISSN*, vol. XLII, pp. 431–442, Dec. 2005, doi: 10.1509/jmkr.2005.42.4.431.

[2] D. Norman, *The Design of Everyday Things*. Basic Books, 250 West 57th Street, 15th Floor, New York, New York 10107: Basic Books, 2013.

[3] J. Nielsen, *Usability engineering*. Boston: Academic Press, 1993.

[4] Similarweb, “Top Websites Ranking,” Top Websites Ranking - Most Visited Websites in April 2024. Accessed: May 21, 2024. [Online]. Available: https://www.similarweb.com/top-websites/

[5] Wikibooks contributors, “Lentis,” Lentis - Wikibooks, open books for an open world. Accessed: May 21, 2024. [Online]. Available: https://en.wikibooks.org/w/index.php?title=Lentis&oldid=4222747

[6] P. E. Ceruzzi, “The Early History of Software, 1952-1968,” in *A History of Modern Computing, second edition*, MIT Press, 2003, pp. 79–80.

[7] R. S. Pressman, *Software engineering: a practitioner’s approach*, 7th ed. Dubuque, IA: McGraw-Hill, 2010.

[8] Oxford English Dictionary, “Feature, N.” Oxford University Press, Mar. 2024. Accessed: May 21, 2024. [Online]. Available: https://doi.org/10.1093/OED/1067298650

[9] Cambridge Dictionary, “Feature.” Cambridge Dictionary, May 15, 2024. Accessed: May 21, 2024. [Online]. Available: https://dictionary.cambridge.org/dictionary/english/feature

[10] S. Apel and C. Kästner, “An Overview of Feature-Oriented Software Development.,” *J. Object Technol.*, vol. 8, no. 5, p. 49, 2009, doi: 10.5381/jot.2009.8.5.c5.

[11] K. C. Kang, S. Kim, J. Lee, K. Kim, E. Shin, and M. Huh, “FORM: A feature-oriented reuse method with domain-specific reference architectures,” *Ann. Softw. Eng.*, vol. 5, no. 1, pp. 143–168, Jan. 1998, doi: 10.1023/A:1018980625587.

[12] CERN, “A short history of the Web,” CERN. Accessed: May 21, 2024. [Online]. Available: https://home.cern/science/computing/birth-web/short-history-web

[13] K. P. Singh, Bebi, and D. Gulati, “Technological March from Web 1.0 to Web 3.0: A Comparative Study,” *Libr. Her.*, vol. 49, no. 2, pp. 146–157, Jun. 2011.

[14] B. Edwards, “Web Browsers Through the Ages,” *PCWorld*, Dec. 23, 2010. Accessed: May 21, 2024. [Online]. Available: https://www.pcworld.com/article/499571/web_browsers_through_the_ages.html

[15] T. Berners-Lee, “Frequently asked questions,” Frequently asked questions by the Press - Tim BL. Accessed: May 21, 2024. [Online]. Available: https://www.w3.org/People/Berners-Lee/FAQ.html#browser

[16] C. Severance, “JavaScript: Designing a Language in 10 Days,” *Computer*, vol. 45, no. 2, pp. 7–8, Feb. 2012.

[17] Netscape Communications Corporation, “NETSCAPE NAVIGATOR 2.0b3 FOR MACINTOSH,” Welcome to Netscape Navigator Version 2.0b3. Accessed: May 23, 2024. [Online]. Available: https://web.archive.org/web/19970614001401/http://home.netscape.com/eng/mozilla/2.0/relnotes/mac-2.0b3.html

[18] Netscape Communications Corporation, “Javascript Basics,” Javascript Basics. Accessed: May 23, 2024. [Online]. Available: https://web.archive.org/web/19970613234917/http://home.netscape.com/eng/mozilla/2.0/handbook/javascript/index.html

[19] C. Aced, “Web 2.0: the origin of the word that has changed the way we understand public relations,” presented at the International PR 2013 Conference, Barcelona, Jul.

2013. [Online]. Available: https://www.researchgate.net/publication/266672416_Web_20_the_origin_of_the_word_that_has_changed_the_way_we_understand_public_relations

[20] G. Toffetti, S. Comai, J. Preciado, and M. Linaje, "State-of-the Art and trends in the Systematic Development of Rich Internet Applications.," *J Web Eng*, vol. 10, pp. 70–86, Mar. 2011, doi: 10.13052/jwe1540-9589.1014.

[21] M. Han and P. Park, "A study of interface design for widgets in web services through usability evaluation," in *Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human*, in ICIS '09. New York, NY, USA: Association for Computing Machinery, Nov. 2009, pp. 1013–1018. doi: 10.1145/1655925.1656109.

[22] K. Kowalczyk and T. Szandala, "Enhancing SEO in Single-Page Web Applications in Contrast With Multi-Page Applications," *IEEE Access*, vol. 12, pp. 11597–11614, 2024, doi: 10.1109/ACCESS.2024.3355740.

[23] E. McGirt, "Facebook's Mark Zuckerberg: Hacker. Dropout. CEO.," *Fast Company*, May 01, 2007. Accessed: May 21, 2024. [Online]. Available: https://www.fastcompany.com/59441/facebooks-mark-zuckerberg-hacker-dropout-ceo/

[24] J. Constine, "Facebook launches Marketplace, a friendlier Craigslist," *TechCrunch*, Oct. 03, 2016. Accessed: May 21, 2024. [Online]. Available: https://techcrunch.com/2016/10/03/facebook-marketplace-2/

[25] N. Sharp, "It's Facebook Official, Dating Is Here," Meta. Accessed: May 21, 2024. [Online]. Available: https://about.fb.com/news/2019/09/facebook-dating/

[26] N. Wirth, "A Plea for Lean Software," *Computer*, vol. 28, no. 02, pp. 64–68, Feb. 1995, doi: 10.1109/2.348001.

[27] B. Komal *et al.*, "The Impact of Scope Creep on Project Success: An Empirical Investigation," *IEEE Access*, vol. 8, pp. 125755–125775, 2020, doi: 10.1109/ACCESS.2020.3007098.

[28] M. Cotterell, "27. Featuritis - Human-Computer Interaction," CSCI 4800/6800 Spring 2023. Accessed: May 21, 2024. [Online]. Available: https://michaelcotterell.com/hci/2023sp/html/ddq/2023-03-21-TUE.html

[29] I. Batterbee, "Feature Creep: what is it, and how does it affect the user experience," How feature creep affects UX | UX Collective. Accessed: May 21, 2024. [Online]. Available: https://uxdesign.cc/features-creepers-the-customer-experience-horror-story-124c8fa73edf

[30] K. Sierra, "Creating Passionate Users: My Favourite Graphs... and the future," Creating Passionate Users. Accessed: May 21, 2024. [Online]. Available: https://headrush.typepad.com/creating_passionate_users/2007/04/my_favorite_gra.html

[31] J. Spolsky, "Things You Should Never Do, Part I," Joel on Software. Accessed: May 21, 2024. [Online]. Available: https://www.joelonsoftware.com/2000/04/06/things-you-should-never-do-part-i/

[32] J. Heid, "Netscape 6 Laden With Design Flaws," *Los Angeles Times*, Dec. 14, 2000. Accessed: May 21, 2024. [Online]. Available: https://www.latimes.com/archives/la-xpm-2000-dec-14-tt-65280-story.html

[33] S. Pruitt, "Study: IE 6 usage zooms past Netscape 6," *CNN*, Sep. 10, 2001. Accessed: May 21, 2024. [Online]. Available: https://edition.cnn.com/2001/TECH/internet/09/10/ie.netscape.idg/

[34] B. Elliott, "Anything is possible: Managing feature creep in an innovation rich environment," in *2007 IEEE International Engineering Management Conference*, Jul. 2007, pp. 304–307. doi: 10.1109/IEMC.2007.5235049.

[35] D. Bishop, "The theory of conservation of complexity," *Interactions*, vol. 15, no. 4, pp. 61–63, Jul. 2008, doi: 10.1145/1374489.1374504.

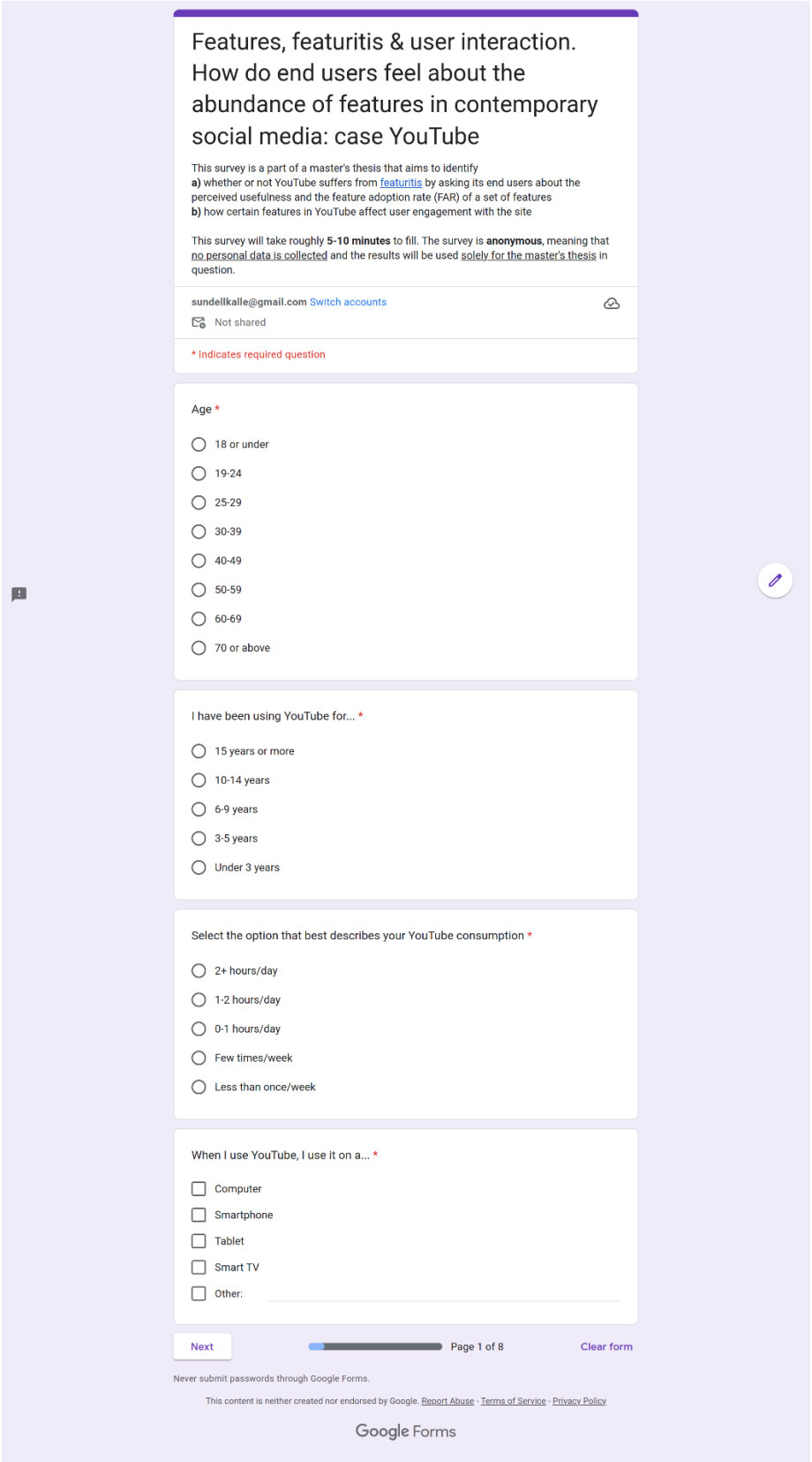
[36] W. L. Hosch, “YouTube,” YouTube | History, Founders, & Facts | Britannica. Accessed: May 21, 2024. [Online]. Available: https://www.britannica.com/topic/YouTube

[37] Similarweb, “youtube.com,” youtube.com Traffic Analytics, Ranking & Audience [April 2024] | Similarweb. Accessed: May 21, 2024. [Online]. Available: https://www.similarweb.com/website/youtube.com/#overview

[38] Version Museum, “Design History of youtube.com,” Version Museum. Accessed: May 21, 2024. [Online]. Available: https://www.versionmuseum.com/history-of/youtube-website

Appendices

Appendix 1: Survey (preliminary information)



Features, featuritis & user interaction. How do end users feel about the abundance of features in contemporary social media: case YouTube

This survey is a part of a master's thesis that aims to identify
a) whether or not YouTube suffers from featuritis by asking its end users about the perceived usefulness and the feature adoption rate (FAR) of a set of features
b) how certain features in YouTube affect user engagement with the site

This survey will take roughly **5-10 minutes** to fill. The survey is **anonymous**, meaning that no personal data is collected and the results will be used solely for the master's thesis in question.

sundellkalle@gmail.com Switch accounts

Not shared

* Indicates required question

Age *

- 18 or under
- 19-24
- 25-29
- 30-39
- 40-49
- 50-59
- 60-69
- 70 or above

I have been using YouTube for... *

- 15 years or more
- 10-14 years
- 6-9 years
- 3-5 years
- Under 3 years

Select the option that best describes your YouTube consumption *

- 2+ hours/day
- 1-2 hours/day
- 0-1 hours/day
- Few times/week
- Less than once/week

When I use YouTube, I use it on a... *

- [ ] Computer
- [ ] Smartphone
- [ ] Tablet
- [ ] Smart TV
- [ ] Other:

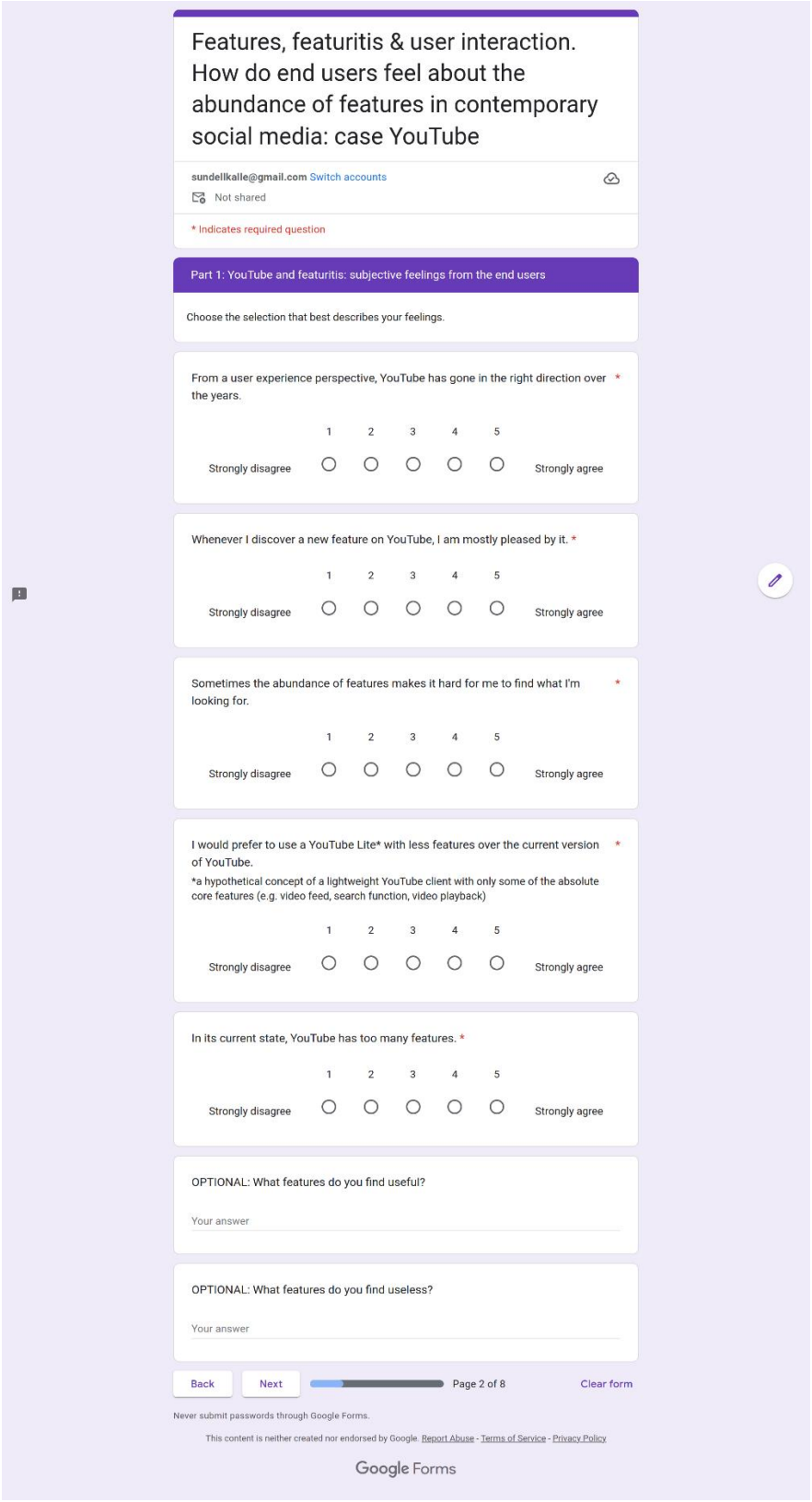
Next Page 1 of 8 Clear form

Never submit passwords through Google Forms.

This content is neither created nor endorsed by Google. Report Abuse · Terms of Service · Privacy Policy

Google Forms

Appendix 2: Survey (part 1)



Features, featuritis & user interaction. How do end users feel about the abundance of features in contemporary social media: case YouTube

sundellkalle@gmail.com Switch accounts

Not shared

* Indicates required question

Part 1: YouTube and featuritis: subjective feelings from the end users

Choose the selection that best describes your feelings.

From a user experience perspective, YouTube has gone in the right direction over the years. *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

Whenever I discover a new feature on YouTube, I am mostly pleased by it. *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

Sometimes the abundance of features makes it hard for me to find what I'm looking for. *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I would prefer to use a YouTube Lite* with less features over the current version of YouTube. *

*a hypothetical concept of a lightweight YouTube client with only some of the absolute core features (e.g. video feed, search function, video playback)

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

In its current state, YouTube has too many features. *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

OPTIONAL: What features do you find useful?

Your answer

OPTIONAL: What features do you find useless?

Your answer

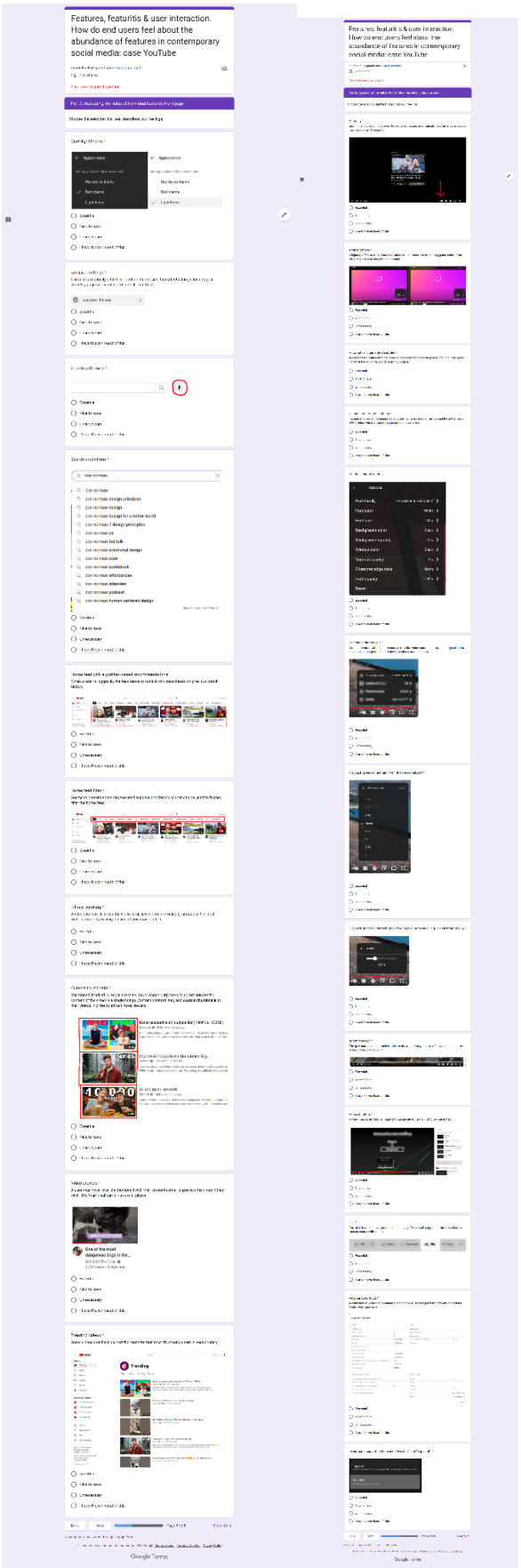
Back Next Page 2 of 8 Clear form

Never submit passwords through Google Forms.

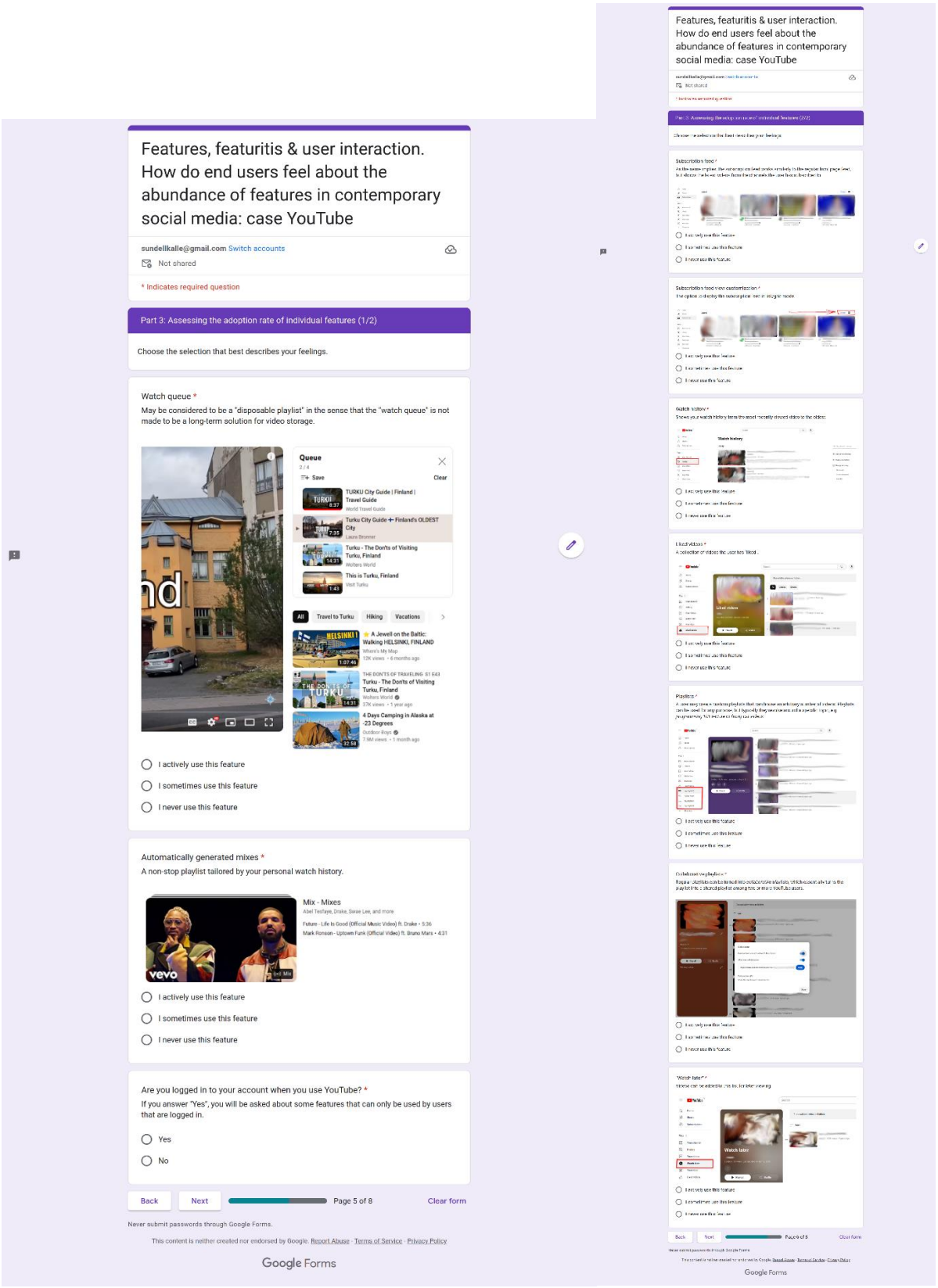
This content is neither created nor endorsed by Google. Report Abuse - Terms of Service - Privacy Policy

Google Forms

Appendix 3: Survey (part 2)



Appendix 4: Survey (part 3, individual features)



Features, featuritis & user interaction. How do end users feel about the abundance of features in contemporary social media: case YouTube

sundellkalle@gmail.com Switch accounts

Not shared

* Indicates required question

Part 3: Assessing the adoption rate of individual features (1/2)

Choose the selection that best describes your feelings.

Watch queue *

May be considered to be a "disposable playlist" in the sense that the "watch queue" is not made to be a long-term solution for video storage.

- I actively use this feature
- I sometimes use this feature
- I never use this feature

Automatically generated mixes *

A non-stop playlist tailored by your personal watch history.

- I actively use this feature
- I sometimes use this feature
- I never use this feature

Are you logged in to your account when you use YouTube? *

If you answer "Yes", you will be asked about some features that can only be used by users that are logged in.

- Yes
- No

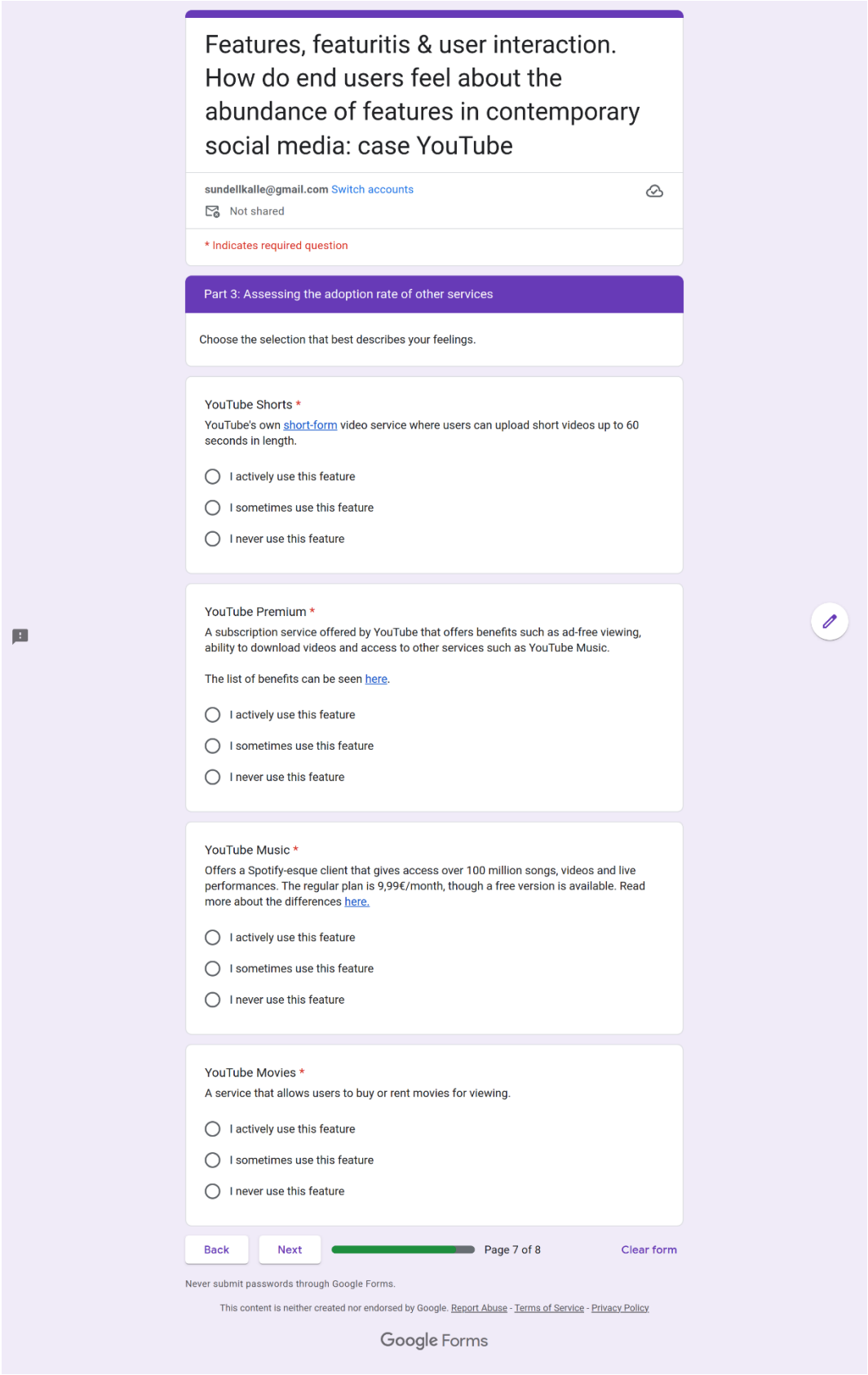
Back Next Page 5 of 8 Clear form

Never submit passwords through Google Forms.

This content is neither created nor endorsed by Google. Report Abuse · Terms of Service · Privacy Policy

Google Forms

Appendix 5: Survey (part 3, platforms)



Features, featuritis & user interaction. How do end users feel about the abundance of features in contemporary social media: case YouTube

sundellkalle@gmail.com Switch accounts

Not shared

* Indicates required question

Part 3: Assessing the adoption rate of other services

Choose the selection that best describes your feelings.

YouTube Shorts *

YouTube's own short-form video service where users can upload short videos up to 60 seconds in length.

- I actively use this feature
- I sometimes use this feature
- I never use this feature

YouTube Premium *

A subscription service offered by YouTube that offers benefits such as ad-free viewing, ability to download videos and access to other services such as YouTube Music.

The list of benefits can be seen here.

- I actively use this feature
- I sometimes use this feature
- I never use this feature

YouTube Music *

Offers a Spotify-esque client that gives access over 100 million songs, videos and live performances. The regular plan is 9,99€/month, though a free version is available. Read more about the differences here.

- I actively use this feature
- I sometimes use this feature
- I never use this feature

YouTube Movies *

A service that allows users to buy or rent movies for viewing.

- I actively use this feature
- I sometimes use this feature
- I never use this feature

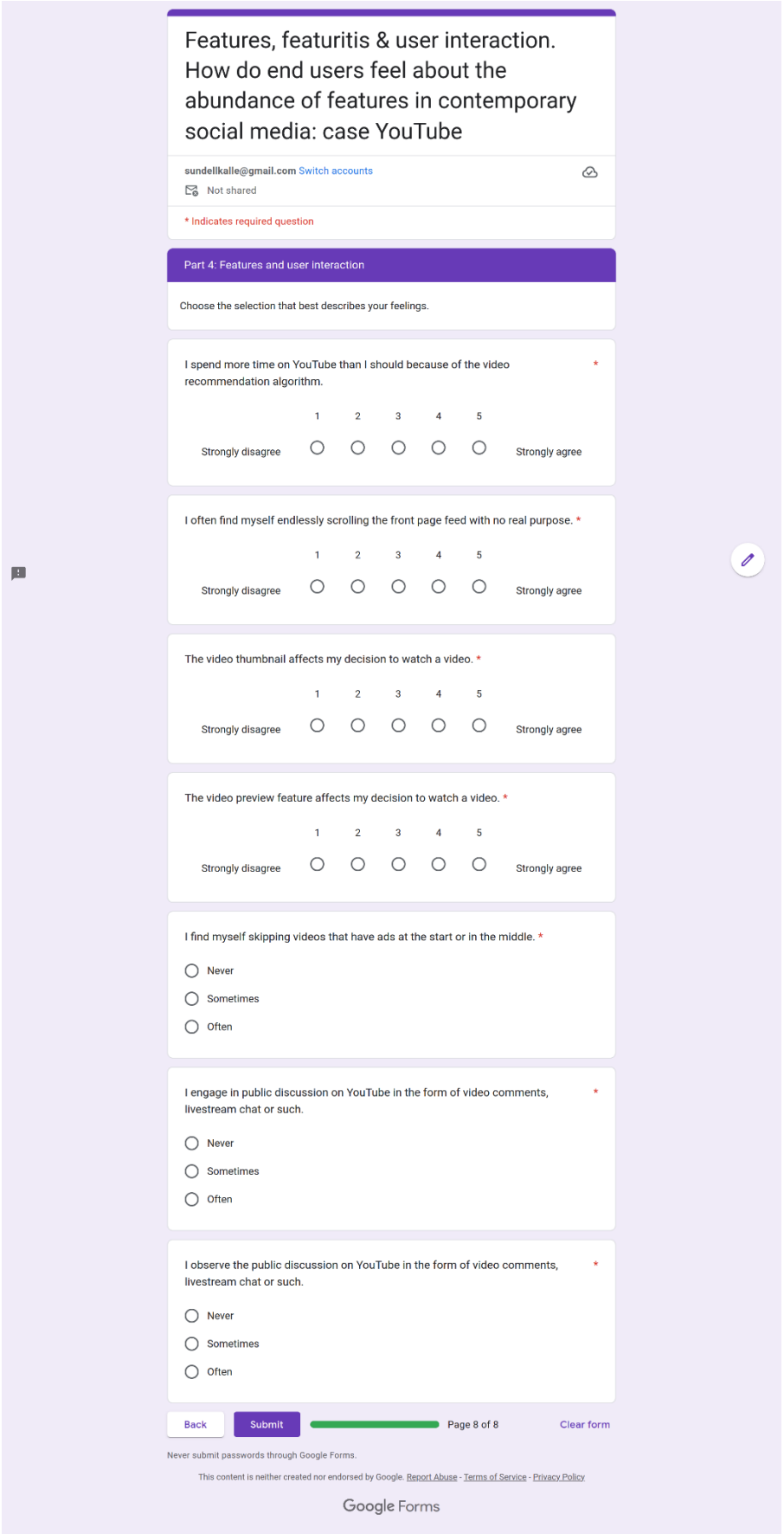
Back Next Page 7 of 8 Clear form

Never submit passwords through Google Forms.

This content is neither created nor endorsed by Google. Report Abuse - Terms of Service - Privacy Policy

Google Forms

Appendix 6: Survey (part 4)



Features, featuritis & user interaction. How do end users feel about the abundance of features in contemporary social media: case YouTube

sundellkalle@gmail.com Switch accounts

Not shared

* Indicates required question

Part 4: Features and user interaction

Choose the selection that best describes your feelings.

I spend more time on YouTube than I should because of the video recommendation algorithm. *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I often find myself endlessly scrolling the front page feed with no real purpose. *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

The video thumbnail affects my decision to watch a video. *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

The video preview feature affects my decision to watch a video. *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I find myself skipping videos that have ads at the start or in the middle. *

- ○ Never
- ○ Sometimes
- ○ Often

I engage in public discussion on YouTube in the form of video comments, livestream chat or such. *

- ○ Never
- ○ Sometimes
- ○ Often

I observe the public discussion on YouTube in the form of video comments, livestream chat or such. *

- ○ Never
- ○ Sometimes
- ○ Often

Back Submit Page 8 of 8 Clear form

Never submit passwords through Google Forms.

This content is neither created nor endorsed by Google. Report Abuse - Terms of Service - Privacy Policy

Google Forms