Abusability Testing Framework

August 2022 • Public Draft Proposal V1.0.0 by The Mobius Project

1 Objective

We provide a systematic and proactive way to approach harm on technology platforms. Many marginalised communities have benefited from technology to expand their advocacy for a broader impact and provide more support for their members (e.g. crisis helplines). Yet, technology has also amplified the harms and created new avenues for exclusion (e.g. identity-based harassment).

Currently, platform abuse strategies consist of idiosyncratic ad hoc approaches that attempt to report and curb abuse. Product teams do not often incorporate perspectives of marginalized communities even when these groups are represented in some tech teams. This framework centers the perspectives of affected groups who have repeatedly called for more proactive strategies for harm reduction.

Our goal is to help guide a starting point for technologists, product managers, and engineers to foster further research, cross-platform collaboration, and guidelines for abusability testing as a standard, expected, and well-executed practice in the evaluation of products. Architecting the world that we want to see with values we want to uphold (fostering belonging, safe communities, equitable outcomes, democracy) requires an intentional industry-wide effort.

2 Overview

The key components of the Abusability Testing Framework are briefly introduced to establish a shared understanding of the terminology that we will be using throughout. While various disciplines and scholars have defined terms such as “harm”, “platform”, and “abuse”, this Abusability Testing Framework does not use these taxonomies. Instead, the framework defines specific terms to further our understanding of social issues and technology and provide a comprehensive breakdown of how we can address platform abuse.

Abusability Testing is the process of taking a systematic approach to evaluate technology products, features, or services that may facilitate harm. This process holistically analyzes already occurring harms, current threats to users[1], and future harms.

Abusability Testing Framework is a structured guideline to conduct abusability testing that we are introducing in this document. It includes terminology to streamline communication and an operational strategy to design products, policies, and procedures that concern platform abuse. This framework prioritizes the identification of five components: definitions of *harm*, technology *features*, *affected audiences*, *builders*, and *sources of harm*. The components make up the analyses points that are presented comprehensively in 3 Steps.

[1] See Appendix C for a brief note of why we choose the term “users”

Platform Abuse is the intentional or unintentional[2] use of technology platforms (including misuse, weaponization, beneficial use) that result in *harm* to self or others.

Technology platform is any online or digital environment, software or hardware tools and infrastructure, that affects any person (regardless of whether they use the platform). Examples: AWS (cloud services), Github (online repository), Facebook (social media).

Harm has been outlined to encompass a wide range of issues. In this framework, we maintain a flexible definition that includes consideration of physical, psychological, social, financial, and legal injuries and damage. Negative consequences can result from engaging with technology individually, engaging with others on technology platforms, and having technology used on an individual or group . We adopt an adaptable definition because harms are intersecting and changing in nature. Our framework remains applicable to these developments when we design for ambiguity and unpredictability.

Features are the aesthetic or functional components (i.e. affordances) of a technology. Features include policies and processes that influence users' behavior, placing constraints and avenues for interacting with the platform or others (e.g. content moderation policies).

Affected Audiences are individuals or group(s) that are negatively impacted by products, features, and harms. These parties can be users or non-users of a technology platform.

Builders are individual technologists, teams, and companies constructing the technology *features* that dictate conduct and activity on a platform.

Sources of Harm are actors with agencies who deploy attacks and take advantage of *features* to misuse technology. *Sources of Harm* also include more passive structural and integrated design *features* (affordances and policies) that can facilitate harm (e.g. doom scrolling).

Mitigations are calculated modifications, such as redesigns or policy adjustments, aimed to curb or reduce platform abuse.

3 How To Use This Framework

Platform abuse is proliferating at an exponential rate. As technologists, designers, and engineers we deal with barriers when we try to address platform abuse. We are constantly limited by factors such as:

[2] In Appendix C, we describe how intentions play a role in both defining and contextualizing instances of platform abuse. We will explore the implications of intent on abusability testing more thoroughly in future versions of this framework.

- **Lack preventative measures:** there are little consolidated preventative designs to combat abuse
- **Financial resources:** few tech companies are devoting adequate resources to mitigate it
- **Information resources:** it is not easy to find information about platform abuse or industry-wise efforts to fix specific issues
- **Cultural or personal friction:** even if out teams know about certain harms and have the resources for it, they may not think they can solve or address large scale problems
- **Institutional political willpower:** many companies do not enough have the internal political willpower to address it at the lowest and highest level
- **Limited experience/view**: We design for ourselves when we think of the "average" user, making it difficult to address issues at the margins. Design exercises that push us to think about personas don't adequately consider diversity of users
- **Business goals conflict:** Ambitious user acquisition and expansion goals make it difficult for teams to advocate for scaling thoughtfully because engagement is more aligned with business goal

These hurdles may remain, but this framework can be used to establish a shared terminology of how platform abuse can happen on an online platform. It can provide a structured approach to foreground multiple platform abuse issues. For example, it can be employed:

- When you or similar companies experience abuse situations or incidents that you might expect on your own platform given features that you have in common
- Before a new feature is developed or launched, abusability testing ensures product teams give careful consideration to potential harms, and not just benefits. If you have a product but don't think that abuse is a problem, your team should think about why abuse either has not been reported or has not been properly addressed.
- At the start of every quarter, to ensure vigilance and continuous improvement and shoring up defenses. If you are planning to launch a new product or feature, your team should go through our exercises together at least once during the product feature roadmap review. This way you can inspect whether the features you are implementing achieve the intended effect without incurring harms to specific users

Platform abuse evolves constantly. As you learn through the process of abusability testing your products, sharing this feedback with other practitioners, including platformabuse.org and Trust & Safety communities, will make this framework more robust, and its resources more useful.

4 Abusability Testing Components

We have covered a few reasons why abuse is important to address and what terminology we will be using throughout. In this section, we provide the five components that are part of the Abusability Testing Framework along with some examples. The Abusability Testing Framework relies on these component definitions to move into analysis, which will be structured into 3 Steps.

Harms: physical, psychological, social, financial, and legal injuries and damage

- *Psychological harms* include activities that induces stress, trauma, and undue discomfort, such as: trollying, bullying, verbal harassment, online harassment, and domestic violence incidents
- *Physical harms* include scenarios where bodily injury occurs or is threatened
- *Sexual harms* include child sexual abuse material, domestic violence incidents, and sexual harassment
- *Financial/economic harms* include partial or total loss of financial assets (e.g. property, goods) and/or income, such as: scamming, fraud, property damage, and loss of opportunity
- *Social harms* include harms to one's reputation, self-image, loss of access to interpersonal relationships and social networks which provide value, activities that hinder or fragment community organizing. This includes users getting kicked off a site or not being able to participate in civil, public discourse
- *Political harms* include threats to democratic or governance processes, such as foreign bot accounts that target public opinions and undermine public trust

Affected Audience: An individual, group(s), or society(s) that is impacted by products, features, and harms

- Users: Anyone who has access to the *technology platform* and employs it. A given user is able to interact within a technology platform via any number of features with other user(s) (Figure 1), where either or both users may be a member of an Affected Audience, a Source of Harm, etc. The interaction from one user to another via a feature on the technology platform may be intentionally or unintentionally positive or cause harm (see Appendix A: Intentions).

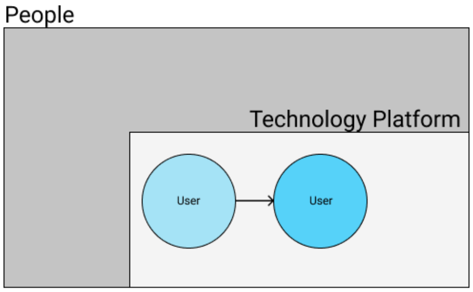


Figure 1

- Non-Users: people who don't use the technology platform but are still affected by technology use by others
- Marginalised Group: People who have been historically and disproportionately socially excluded, targeted, harmed, surveilled, left behind. These groups can be users, builders, neither, or both. We focus on these specific groups:
    - Black, Indigenous, People of Colour (BIPOC), LGBTQ groups, current and former sex workers, domestic and sexual violence victims, homeless and socioeconomically insecure individuals, activists to name a few. Each product, group, organization has data from reported abuses, user research, and analytics to add to this list and to prioritize which groups have been previously harmed, most susceptible to harm

Mitigations: we consider 3 different ways: policy, process, product, to address abuse

- Policy: decisions about enforcement, such as new guidelines to ban white supremacists or hate speech groups
- Process: trust and safety operations, such as how content moderators are trained and whose reports they prioritize or escalate
- Product: implementation through technology, such as community moderation tools and machine learning hate speech detection

Builders: People, institutions, and organizations who are creating, designing, or maintaining technology platforms. Among other factors, their work impacts the wider body of people and *affected audiences*. Additionally, *builders* may be *users* of the *technology platform* itself (Figure 2)

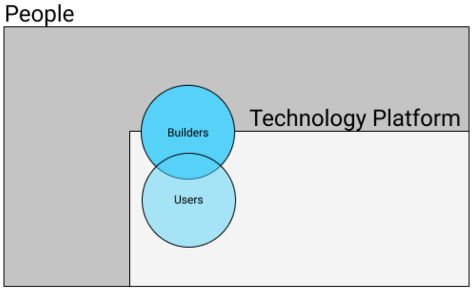


Figure 2

Feature: the aesthetic or functional components (i.e. affordances) of a technology. This could include default settings, ease of resharing content or creating multiple accounts, being able to see someone's friends list.

These components make up the starting point of the Abusability Testing Process.

5 Abusability Testing in 3 Steps

Our framework for mitigating platform abuse is outlined in the series of questions and guidelines that are displayed in the graphic Figure 3 and the structured questions in Table 1.

There are three steps to the abusabilty testing framework: (1) Understand the Situation, (2) Anticipate Future Impact, (3) Develop Proactive Mitigations. Within each step, the Abusability Testing Components are used to break down the analysis of each step. For example, in order to (1) Understand the Situation, our framework delineates that teams address the features, affected audiences, harms, source of harm, and builders involved in the situation that has occurred. In (2) Anticipate Future Impact, the same components (features, affected audiences, etc) are critically analysed to list related features or similar affected audiences, that should be monitored given the situation that was addressed in Step (1) Understand the Situation. The entire framework is outlined in Table 1 showing each step of the abusability testing process.

The consolidated graphic in Figure 3 shows the holistic aspects of the abusability testing framework. Each ring layer of the diagram (different shades of blue) corresponds to the three different columns in Table 1. Table 1's first column called (1) Understand the Situation corresponds to the dark blue inner circle in Figure 3. The second Table 1 column, (2) Anticipate Future Impact, corresponds to the outer light blue circle in Figure 3.

Finally, we include an example case study (Scenario 1) for how to follow the abusability testing process in Table 2. In Appendix B, we provide 2 additional scenario case studies (Scenarios 2 and Scenario 3) that you or your team can use to follow the abusability testing framework we propose.

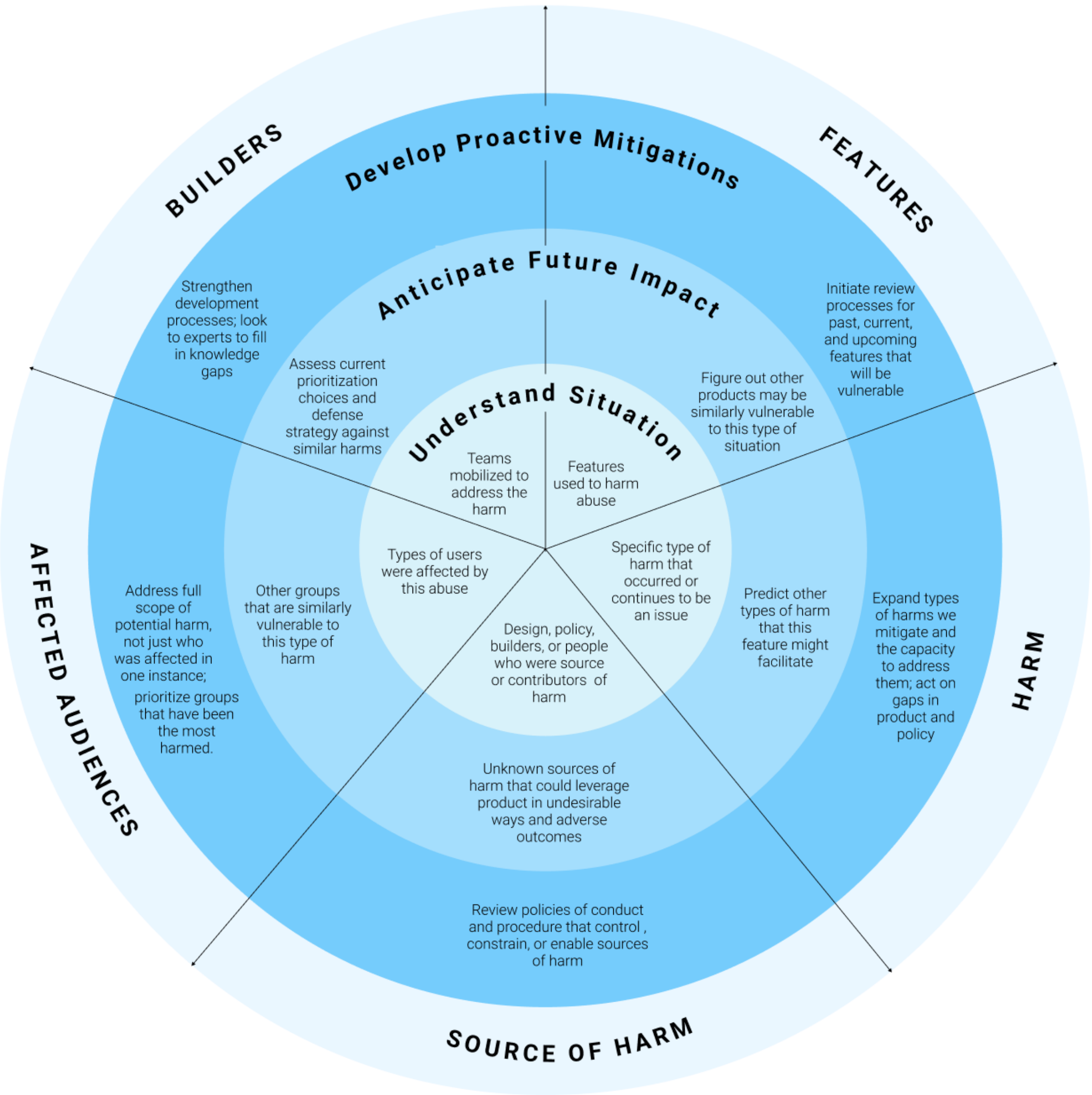

BUILDERS
FEATURES
HARM
SOURCE OF HARM
AFFECTED AUDIENCES
Develop Proactive Mitigations
Anticipate Future Impact
Understand Situation
Strengthen development processes; look to experts to fill in knowledge gaps
Initiate review processes for past, current, and upcoming features that will be vulnerable
Expand types of harms we mitigate and the capacity to address them; act on gaps in product and policy
Review policies of conduct and procedure that control , constrain, or enable sources of harm
Address full scope of potential harm, not just who was affected in one instance; prioritize groups that have been the most harmed.
Assess current prioritization choices and defense strategy against similar harms
Figure out other products may be similarly vulnerable to this type of situation
Predict other types of harm that this feature might facilitate
Unknown sources of harm that could leverage product in undesirable ways and adverse outcomes
Other groups that are similarly vulnerable to this type of harm
Teams mobilized to address the harm
Features used to harm abuse
Specific type of harm that occurred or continues to be an issue
Design, policy, builders, or people who were source or contributors of harm
Types of users were affected by this abuse

Figure 3. The abusability testing process we propose in this document consists of 3 Steps, which are represented by the 3 concentric rings in 3 different shades of blue. Each of the Abusability Testing Components can be analyzed at each point of the 3 Steps. For example, we assess which “Affected Audiences” were harmed in “Understand the Situation” (Step 1), then we also Anticipate Future Impact (Step 2) based on the Affected Audiences we identified in Step 1. This process can be customized depending on the scenario; for example, if a feature is pre-launch, “Understand the Situation” (Step 1) could be looking at an incident from a similar feature launched by a competitor.

Table 1. Abusability Testing in 3 Steps

	1. Understand the Situation *Create a comprehensive summary of how and why the situation or incident occurred.*	**2. Anticipate Future Impact** *After immediate harm reduction response, understand the situation's broader potential for future harm.*	**3. Develop Proactive Mitigations** *Use insights from broader context to create a stronger and more cohesive mitigation plan*
Feature	Evaluate what features were involved in the situation (how did the attacker use different features/what features enabled the harm to get amplified?)	The next step is to use this list of features to figure out 1) related features/products we need to be aware of, 2) how feature matches up with our current safety guidelines	From related features, we will 1) learn from past situations or incidents; 2) try to minimize spillover effects into other platforms. From evaluating feature's relationship with current safety guidelines, we will discuss what needs to be changed from product/policy perspective
Affected Audience	Evaluate what demographics and types of users were affected by this specific situation	Figure out the true potential scope of the harm: aside from who was affected by this situation, who else could be affected? We know it might not just be one group	Make sure our mitigations actually address the broader scope of potential harm, not just who was affected by the specific situation. In our mitigation process, we want to prioritize folks who have been the most harmed.
Harm	Understand the specific type of harm that occurred in the situation	Understand other categories of harm (including ones that may not be an issue yet) that this feature might create	Understand gaps between types of harms that were already part of the organization's safety process, and the harms that could happen as a result of this feature. If needed, expand types of harms that we anticipate and capacity to address them.
Source of harm	Understand who (bad actor) or what (platform feature) was the source of harm in this situation	What other sources of harm could leverage this feature in an undesirable way?	Understand what sources of harm were imagined and already a part of the organization's safety process. If needed, expand the sources of harm that we anticipate and capacity to address them.

Builders	Understand what structural business issues led to this harm, and what people and processes were mobilized to address it.	Who or what do our current teams proactively protect, and who or what do they fail to protect? What was overlooked or missing in terms of people or process? What are the consequences of those prioritizations?	From a team and capacity perspective, understand the weaknesses of our organization with regards to platform safety and look for experts from the affected audience groups to guide us to more robust teams

6 Example Case of Abusability Testing

Scenario 1 Instagram business accounts misused by minors

According to estimates from a data scientist, for over a year at least 60 million users on Instagram under the age of 18 were given the option to easily change their profiles to business accounts. This existing feature required their contact information and made this information accessible to others on their profile page in the Instagram app. While not intended for minors, minors said that they used the business account feature because it would allow them to see metrics, such as how many people had visited their profile and seen individual posts.

By clicking on the “Email” button on a minor’s business account, the user’s default email program would launch and the actual email address of the recipient (not an anonymized version) would be clearly displayed in the “To:” section of the email.

In an article published in November of 2019, NBC News reported that they were able to find numerous examples of children, some under 13, who exposed their personal details to the world through this feature. This information has also been leaked through the website’s source code, which allowed hackers to scrape the data from Instagram to obtain contact information for millions of Instagram accounts, some of whom were minors.

Although Instagram says that worldwide there are around 25 million business accounts, they do not have data on how many are run by users under 18. This is in part because for the first nine years of existence, Instagram did not ask users their age upon signup, which allows them to feign ignorance about how old they are and means that they cannot be held liable for $40,000 per violation of the Child Online Privacy Protection Act (COPPA). To clarify, Instagram is liable for COPPA violations but were trying to avoid assuming responsibility.

In December of 2019, TechCrunch reported that Instagram finally launched 13+ age check-ups, which means that it now asks new users to input their birth date and then bars users younger than 13 from joining. However, as it will not ask existing users their age, underage kids already amongst its 1 billion members will not be subject to this new process. Instagram also heavily relies on users reporting other users using the "flag/report abuse" feature to limit another user's access to the platform; if a suspected under-13 user is reported, Instagram will freeze the account temporarily and delete the account if that user cannot show verification that they are 13 or older.

Related features:

- Teenage girls say that they have been subject to unwanted and inappropriate messages through Instagram's direct messaging tool -- phone numbers and email addresses only creates more channels for contact with a minor
- Instagram's "search" feature: NBC reported that they were able to easily find minors through searching keywords like "cheerleading"

Instagram has since released features that aim to better protect teens.

Instructions for using this scenario: Place yourself in the shoes of employees at Instagram who are working on the business accounts team, and have not yet been informed or had a major incident occur over minors using business accounts. (Assume that the TechCrunch and other articles have not yet been published.)

Table 2: Abusability Testing Worksheet Using Scenario 1

An example worksheet using the Abusability Testing guidelines, with one sample question per section of the 3 Step framework. The more extensive chart of the Abusability Testing Worksheet can be found in Appendix 1. This table is loosely based on Scenario 1, which is an instance where an adverse event is reported to the company or to those deemed responsible for the adverse event. It can be tweaked for Scenarios 2 and 3 in Appendix B as well.

	Understanding the situation (Q)	Understanding Context	Developing a mitigation plan

	Example question	Example answer	Example takeaway	Example question	Example Answer	Example question	Example Answer
Feature	What were the products / features involved? Was the product/feature working as it was designed to work or intended to be used?	Business account sign up process, business accounts analytics, business contact information display and access	All 3 involved features were working as intended. We didn't foresee how minors have been using them or how they would use them due to the information that business accounts provide that would appeal to this user base given their highly socially networked lives	What are other products with similar features that may have been exploited in ways we have not anticipated?	Slack, Zoom, Skype, LinkedIn, Facebook with features that have promotional social media contact us options or usage with multiple types of users.	What was the justification for building the feature that enabled harm? Are there known solutions for making it safer?	Craigslist creates an anonymous email per listing. Delivery services (e.g. Doordash) use temporary (?)
Affected Audience	Who was harmed?	Minors who are 13-17 years old, and other minors who report false dates of birth	Teens between the ages of 13-17 already have a substantial user group for Instagram. However, these metrics are based on self-reported ages	Who else could be similarly harmed?	Activists, women-owned businesses, influencers, groups targeted by white supremacist or other hate groups	Who is already protected by law? Who is not, and therefore should be protected by design, policy, and/or precedent?	Children are already protected by law.
Harm	How were they harmed?	No direct harm, but personal info exposure	We know about broad categories of harm (ex. privacy) that this feature might create, and are trying to better understand the incident's potential severity and reach.	What were the harms you had in mind that you were trying to mitigate from the onset when building this product?	Because it is a business feature, in this case we were thinking more about fraud and scammers	How do we prioritize which harms to address and when?	We want to create a mitigation plan that reflects the severity and frequency at which particular groups are affected. We should respond first to the incidents of the most affected or marginalized groups.
Sources of Harm	What do we know about this source of harm? (What is their intent)	Harassers and abusers have used our DM feature before to groom / contact / coerce minors	The platform is making it easier for harassment or manipulation to occur, because harassers can contact their victims off-platform (email in addition to DM)	Are we actively giving a known source of societal harm another platform to perform harm?	Yes -- they can contact victims over email without risk of losing their Instagram account, and victims will not know which accounts to block on Instagram	How can we anticipate and address this source of harm?	Look at the possible ways that data and personal information can leave our platform to conduct off-platform harm. We should

Builders	Who should have had (including ourselves) the incentive to fix this incident in our organization chart?	Business accounts team, trust & safety, privacy engineering, legal counsel influencer team	The teams listed in the cell to the left are the people who should go through these qs together. We should also create/align incentives to address these types of harms. How do we make sure the incentive structure/accountability don't fail again?	Who should not be involved given competing and/or conflicting interests to protect users?	There may be teams who should not have decision making power because shipping safety features would hurt the metric by which they are evaluated (e.g. engagement)]	Who are the stakeholders that should be involved and/or at the table?	[This could include internal teams as well as external organizations and users of the platform with lived experiences]
Mitigations	What systems / teams / people at the company did or did not respond to the incident?	Known legal problem - no age verification to not make ourselves liable for minors		Who do our existing mitigations actually protect?	Because our Trust & Safety team is mostly focused on fraud with regards to business accounts, our mitigations currently prioritize consumers who may be scammed by false businesses.	What mitigations will we implement long term?	

7 Abusability Testing and SDLC

Where does abusability testing fit into the Software Development Lifecycle SDLC? In software and hardware development, best modern industry practices standardise the development lifecycle using various methodologies. The typical Software Development Lifecycle (SDLC) has looked like the following[3] (Figure 4):

Typical Software Development Lifecycle (SDLC):

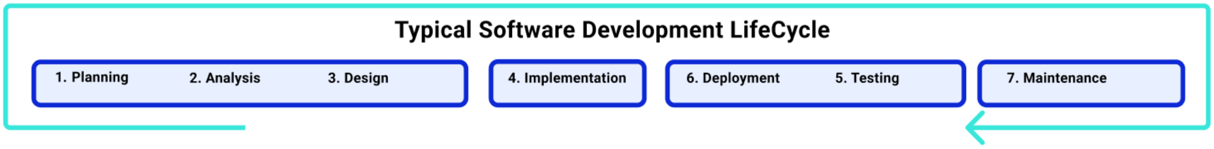


Figure 4. The typical Software Development Lifecycle (SDLC) incorporates ordered stages above such as the one above.

In recent decades, the Software Development Lifecycle has been expanded, modified, altered, to accommodate risk mitigation in privacy or security engineering. For example, an expanded version that incorporates security engineering into SDLC can look like the following (Figure 5):

Modified SDLC to incorporate Security Engineering:

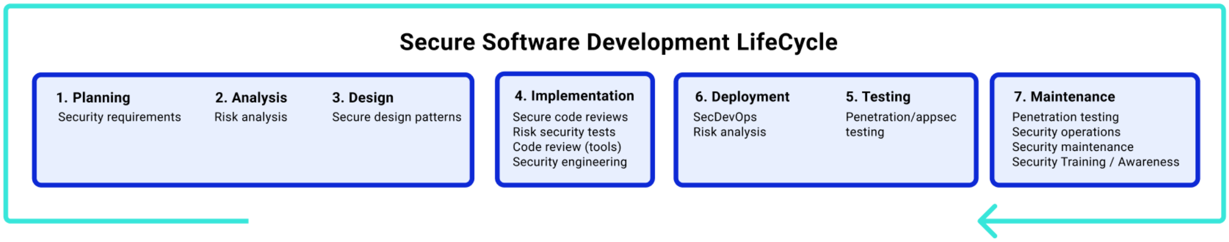


Figure 5. The light blue highlighted portions indicate the specific stages in which security engineering was incorporated into previous SDLC models.

Existing models of development lifecycles and programs, even those that expand and/or interface with security and privacy engineering, do not address combating *platform abuse*. The 3 Step Mitigation Process (detailed in the next section *Abusability Testing Process 3 Step Mitigation Process*)

Step 1: Understanding the Situation / Incident
Step 2: Anticipate Future Impact
Step 3: Develop proactive mitigations

can occur at any stage of the SDLC. We propose one version of the incorporation could look like (Figure 6):

Modified SDLC to incorporate Abusability Testing:

[3] Royce, Winston W. *Managing the Development of Large Software Systems*. IEEE WESCON, 1970.

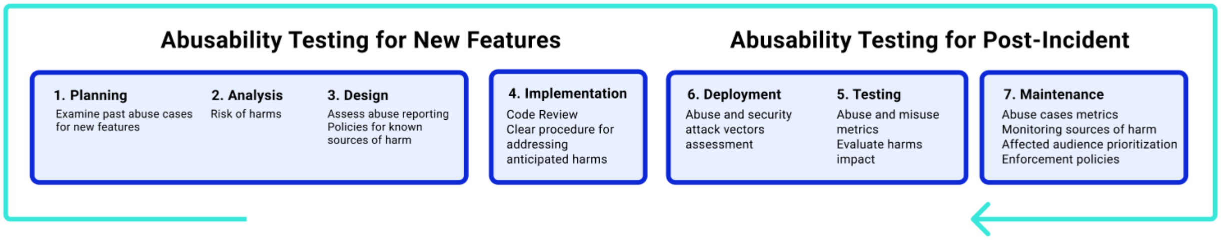

Abusability Testing for New Features
Abusability Testing for Post-Incident
1. Planning
Examine past abuse cases for new features
2. Analysis
Risk of harms
3. Design
Assess abuse reporting
Policies for known sources of harm
4. Implementation
Code Review
Clear procedure for addressing anticipated harms
6. Deployment
Abuse and security attack vectors assessment
5. Testing
Abuse and misuse metrics
Evaluate harms impact
7. Maintenance
Abuse cases metrics
Monitoring sources of harm
Affected audience prioritization
Enforcement policies

Figure 6. The blue highlighted portions indicate the specific stages in which abusability testing could be incorporated into previous SDLC models

We will provide alternative versions of an amended SDLC that incorporates abusability testing in future releases.

8 Existing Frameworks

We list some existing frameworks that have shaped the process of software development as it is today, but do not adequately address the harms that technology platforms have created, particularly for the most vulnerable users. The Abusability Testing Framework draws inspiration from these frameworks to propose a new process to support the mitigations of harms in a systematic approach. Our goal is to develop a framework that is as commonly used in software development as these frameworks below, but with a main focus of prioritizing safety and the concerns of people who are most harmed by technological platforms today.

Human-centered design / design thinking

- Human-centered design is an approach to innovation that involves building a deep empathy with the people you're designing for, building prototypes, sharing what you've made with the people you're designing for, and eventually putting your innovative new solution out into the world (IDEO)
- One common design framework is the double diamond: Discover, Define, Develop, Deliver

Lean startup /agile process

- The lean startup method provides an approach to creating and managing startups and get a desired product to customers' hands *faster*
- Through lean startup, if an idea is likely to fail, it will fail quickly and cheaply instead of slowly and expensively, hence the term "fail-fast."

Privacy engineering

- NIST Privacy Framework 1.0: (a) Identify, (b) Govern (c) Control (d) Protect (e) Communicate." This framework is concerned with a specialty discipline of systems engineering focused on achieving freedom from conditions that can create problems for individuals with unacceptable consequences that arise from the system as it processes PII."

Security engineering

- "An interdisciplinary approach and means to enable the realization of secure systems. It focuses on defining customer needs, security protection requirements, and required functionality early in the systems development life cycle, documenting requirements, and then proceeding with design, synthesis, and system validation while considering the complete problem." - NIST SP 800-12 Rev. 1 under Security Engineering CNSSI 4009

Appendices

Appendix A. Scenarios of Platform Abuse
We provide 2 additional scenario case studies (Scenarios 2 and Scenario 3) that you or your team can use to follow the abusability testing process.

Appendix B. Abusability Testing Chart (Extended)
In Section 6 Table 2, we provided a set of example questions, example answers, and example takeaways for each stage of the 3-Step Abusability Testing process with respect to the Abusability Testing Components (Sources of Harm, Affected Audiences, Features, Mitigations). Below, we expand that worksheet to include supplementary questions that are part of the framework.

Appendix C. "Users" as a term
We include a brief discussion of why we chose the term "users" despite ongoing debate regarding the term in the human-centered design community.

Appendix D. Intentions
We include a discussion section on how intentions play a role in both defining and contextualizing instances of platform abuse. As a complex topic, we plan to incorporate these considerations in the next iteration of the framework.

Appendix A: Scenarios of Platform Abuse

The following scenarios provide examples of platform abuse at different stages of the product or research development cycle.

Scenario 1 shows an instance where a product team is trying to develop a new feature. We showcase how abusability testing should always happen before a new feature is developed and/or launched. We apply the abusability framework **prior** to any situation or incident and draw on previously reported incidents and similar products that share the same characteristics as a way to actively prevent harm as opposed to waiting for adverse events to occur. (See When (or at what point in the product or research lifecycle) should this be used?

Scenario 2 demonstrates the benefit of abusability testing at the start of every quarter, to ensure vigilance and continuous improvement and shoring up defenses. We apply the abusability framework **continuously, even if no adverse event is detected,** to make sure any predictions on potential changes and projections such as increased user demand or new user market acquisition rollouts, can be properly and safely executed.

Scenario 3 outlines an instance where an adverse event is reported to the company or those deemed responsible for the adverse event. We apply the abusability framework **after** an incident is reported through media, internal reporting channels, whistleblowers, users, etc.

A case study for scenario 1 is on page 11-12 of the framework, with an example of how we would do abusability testing for this scenario in Table 2. Below are cases for scenario 2 and 3; you are welcome to try the 3-step mitigation process on these cases yourself.

Scenario 1 Instagram Business Accounts

(See Scenario 1 Instagram business accounts misused by minors)

Scenario 2 Twitter’s Fleet feature using DMs

On 4 March 2020, Twitter’s product lead announced that Brazil (and later on announced other select markets: Italy, India, South Korea) users with either the iOS or Android Twitter applications would be able to test a new feature called Fleets: a feature that allows users to post text, images, or video up to 24 hours. The Fleets feature as a concept is not new, as it comes after a history of other platforms (e.g. Instagram, Facebook) implementing the same features with entire companies i.e. Snapchat revolving around the concept of temporary social media posts, most with the ability to respond to a user’s story.

This feature rolled out with the ability to Reply by DM. Direct Messages (i.e. DMs), are a long existing feature of Twitter where users may directly message another Twitter user. Over the years, the DM feature has built various user safety, privacy, and security features such as the ability to restrict the DM inbox to only those that a user follows to avoid safety issues such as [targeted] harassment from strangers.

The screenshots and videos from test rollouts showed that any user on Twitter had the ability to Reply by DM any other user as a default for all published Fleets – even if user A had blocked user B or had placed privacy restrictions on whether user B could respond to user A’s Tweets. For the DM features, Twitter usually allows users to restrict potential respondents via the Reply by DM feature. However, when launching Fleets, Twitter essentially allowed users to override the various privacy controls already built into the DM feature. On 17 November 2020, Twitter officially announced in a blog post the Fleets rollout to all users, confirming the overriding of the existing features built into the DM feature through the usage of Fleets, alongside the ability for users to tag other users that have blocked them.

Twitter users demonstrated the abusability of the Reply by DM feature and brought up concerns on the platform. Shortly after, Twitter mitigated the abusable Fleets features to realign with existing DM features (e.g. user safety, privacy, security) on the Fleets feature shortly.

Instructions for using this scenario: Place yourself in the shoes of employees at Instagram who are working on the business accounts team, and have not yet been informed or had a major incident occur over minors using business accounts. (Assume that the TechCrunch and other articles have not yet been published.

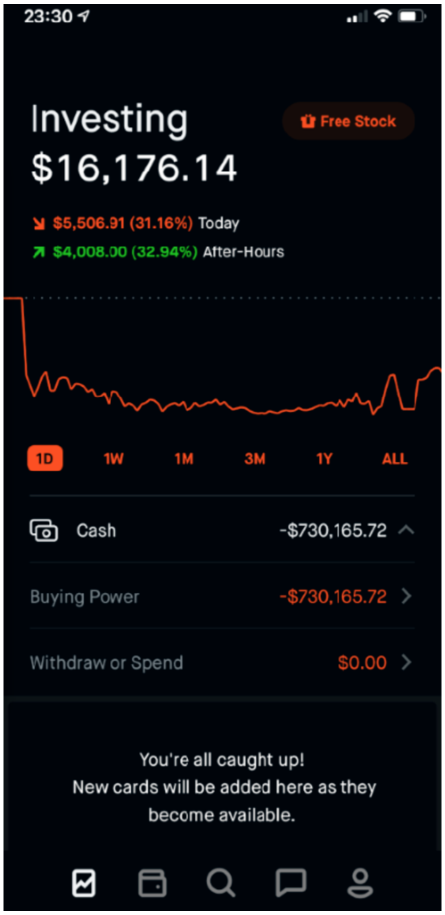
Scenario 3 Robinhood Trading Platform

On June 19 2020, a 20-year old college student Alex Kearns died by suicide after using Robinhood, the stock trading app, and believing that he had lost over $700,000 in a trade. On June 11, Robinhood restricted Kearn's account and later sent an automated email requesting a payment of more than $170,000 due in a few days' time.

The design of Robinhood's trading platform led him to believe that he owed more money than he could afford to pay. He emailed customer service three times to get clarification, writing "I was incorrectly assigned more money than I should have, my bought puts should have covered the puts I sold. Could someone please look into this?"

Robinhood had no customer service phone number, and Alex instead received a "Thanks for reaching out to our support team! We wanted to let you know that we're working to get back to you as soon as possible, but that our response time to you may be delayed."

The family of Alex Kearns recently sued Robinhood for wrongful death in 2021, citing negligence and unfair business practices.



Investing
$16,176.14

$5,506.91 (31.16%) Today
$4,008.00 (32.94%) After-Hours

Cash -$730,165.72

Buying Power -$730,165.72

Withdraw or Spend $0.00

You're all caught up!
New cards will be added here as they become available.

Figure 7. Screenshot of the Robinhood app from Alex Kearn's iPhone that shows a negative cash balance. https://www.cbsnews.com/news/alex-kearns-robinhood-trader-suicide-wrongful-death-suit/

Beyond the ethics of the case, whether it is the fault or responsibility of Robinhood is beyond the scope of this framework. But the lesson here, that hopefully everyone can agree, is that we do not want anything like this to repeat with any platform ever.

Instructions for using this scenario: Break down what happened, what are the ways that the platform caused, or led to, this belief of massive debt.

What does it mean when you are dealing with financial assets and debt, which is consequential than mock money (e.g. game currency that is non-transferrable and convertible to financial assets?

What does that mean for platforms like Pinterest and Instagram where self harm is glorified by certain demographics of users death by suicide has been dealt with, addressed, or still a prevalent on-going challenge?

It is very common for new tech companies to not put in customer service lines - this is one of the main areas that are being automated because the technology industry believes that economically, humans are the biggest operation costs to customer service and they'd rather automate to slow costs. What is the lesson for your product?

Appendix B: Abusability Testing Worksheet (Extended)

	Qs for Understanding the situation	Qs for Understanding Context	Qs for Developing a mitigation plan
Feature	What were the products/features involved in the incident?	What other products have similar features that may have been exploited in ways we have not anticipated?	What are known learnings we can derive from other products with similar features? How can we apply these insights to make the features that were abused safer?
	Was the product/feature working as it was designed to work or intended to be used?	Does this feature override our platform safety/privacy guidelines or the spirit of the policy? Did it skip over any internal processes for safety?	Do we need to adjust product, policy, or process?
	Does this feature make it easier or harder for this harm (or others) to happen?	What other features do affected audiences use on our platform that can also be exploited/used against the affected audiences?	How do we make sure our overall product is safer for the affected audience of this harm, rather than only fixing one feature?
	What was the original intention or reason for implementing the feature?	What are other features that we built under similar intentions and how might they fall short of anticipating potential harms?	Do you intend for this feature to be a permanent addition to your product or is this a temporary solution for a later version?
Affected Audience	Who was harmed? What kinds of users were targeted or at risk as a result of the incident or design?	Who else could be similarly harmed?	How do we make sure our mitigations address not just harms to the affected audience of the incident, but also support other groups that might be similarly harmed?
	Are they users who are over- or under-represented in our ideal user profile, i.e. intended users?	What support does this affected audience currently get from us, other organizations, or society?	Are we over-accommodating a group that we already actively consider in all of our design decisions (are they represented in our company, or we have a partnership with another org, etc)?
	Are they the type of users who are not intended users (e.g. new population, subgroup that repurposed)?	Who are users from similar products that have been excluded from use? Why should we also exclude them?	Who is already protected by law? Who is not, and therefore should be protected by design, policy, and/or precedent?
	Are they an underserved or marginalized group?	Are there some types of users who will have concerns about the feature? Are those marginalized users?	How can we make sure we are not perpetually under-serving a marginalized group?
	How does the incident affect our users' objectives with our product, and therefore affect us (the company)?	What are ways in which incidents like these impact the affected audience by hindering their objectives with our product? (i.e. loss of opportunity)	How do we ensure that our mitigations are not just about stopping bad behavior, but also actively support the affected audience in successfully accomplishing their goals with our product?

Harm	What type(s) of harm occurred?	Is this type of harm already being experienced in other technologies and societies?	Do we need to expand the types of harm we care about and actively address today?
	What is the impact, severity, and reach of the harm?	What are related harms that have been enabled by incidents like these?	Which dimensions of harm can we prioritize?
	Were there known issues/harms related to these features before shipping?	Are there known issues that are not being prioritized?	How often do you, should you, and will you evaluate potential harms?
	What were the harms you had in mind that you were trying to mitigate from the onset when building this product?		
Sources of Harm	What was the source of harm?	Why does this source of harm happen online or offline? Are we perpetuating or scaling an existing societal problem of inequality or lack of justice? At what point do we determine a harm is systemic?	
	What do we know about this source of harm?	Has this source caused/been involved in previous incidents (at our organization or others)?	What is our primary way of discovering sources of harm that specifically impact our users?
	Do we already have active efforts to curtail their harm?	Are we actively giving a known source of societal harm another platform to perform harm?	How can we anticipate and address this source of harm?
Builders	What was the process that this feature /policy went through to be vetted for safety / harm reduction?	What areas or groups does our company currently have robust processes and dedicated, full-time staff focused on protecting? What areas does it not?	What team should be formed or what roles or systems instated to fill the gaps so we can better anticipate future incidents like this?
	Which teams and roles were involved from beginning-to-end of handling this incident?	Who are the stakeholders that should be involved and/or at the table? Who should not be involved given competing and/or conflicting interests to protect users?	
	Who should have had the incentive to fix this incident in our organization chart? What and/or who was missing in the process of creating the system of incentives/accountability?	Were there people internally/externally who raised concerns pre-launch? If so, what happened with those concerns?	How do we promote and resource people who raise valid concerns about abuse to drive product safety decisions?
Mitigations	What feature or process was used to report the harm?	Who do our existing features and systems architecture actually protect?	What mitigations will we implement long term?
	What mitigations did we implement immediately?		
	What systems/teams/people at the company did or did not respond to the incident?		

Appendix C: “Users” as a term

We want to highlight ongoing discussions revolving around usage of the language “user” in areas including Human-Centered Design (HCD), User Experience (UX), Customer Experience (CX), Value Sensitive Design (VSD). Alternative language to “users” may include:

- “Customers”: individuals, groups, or companies who need to, are able to, can purchase, or resell a product or service;
- “Direct stakeholders”: people who directly interact with and use a platform;
- “Clients”: individuals, groups, or companies that obtain a product or service in an on-going relationship;
- “Consumer”: an individual that utilizes a product or service for personal use.

Due to varying demographics that have access to and employ various technology platforms, we have chosen “users” as a generic term and umbrella in this framework to encompass all demographics of users, and specific types of users as necessary (e.g. Affected Audience, see Abusability Testing Components Overview).

Appendix D: Intentions

“Intention” is defined here as the planned aim or purpose of an act and how it was done, such as a feature or mitigation. Intentions are foundational in helping align values and goals, understand the short-term and long-term picture, and can help provide accountability for teams.

Intentions as a concept under abusability testing may be applied as a major category and component of its own, such as next to builders, features, harms, or applied to a major component. As an example, when examining the builders component there are several types of technologists, and they may be differentiated by intent who: 1.a) are unaware or unintentional, lack of diverse experience or background; 1.b) are bad actors, insider threat.

Example: intentions raised to the same level as builders, users, harms

Builders → Intentions → Users → Harms → (repeat)

When examining intentions as a major component, we break it down into three major types:

Intentions			
Unintentional harms	Intent to build or not build for	Intentional harms	
An intentional unknown action to be “good” or “bad” that leads to unintentional and/or unexpected harms (due to lack of understanding)	Negligence that leads to harms	An intentional action known to be “bad” that leads to expected harms	An intentional action known to be “bad” that leads to unintentional or expected harms
No conscious decision to act; Did not expect bad or severe outcomes; No motivations to harm	No conscious decision to act; No motivations to harm or prevent harm from occurring	Conscious decision to act; Motivation to harm	Conscious decision to act but didn’t expect bad or severe outcomes; No motivations to harm

We believe that intentions are important to explore, and will be expanding intentions further in upcoming releases.

Note on answering the questions: we understand that intent plays an integral role in the ways that someone might answer these questions. For example, how a user (benevolent or malicious) intended to use a product vs how a builder intended a user (benevolent or malicious) to use the product. We invite you to explore this additional aspect in the answers to these questions