**Terms and Acronyms**

This list contains the most common terms used in general and in most projects.

**Table of Contents:**

* [uTest Terms](https://www.utest.com/#uTest)
* [Testing Terms](https://www.utest.com/#Testing)
* [Website/App Elements Terms](https://www.utest.com/#Elements)
* [Software Development Life Cycle Terms](https://www.utest.com/#SDLC)
* [Streaming Terms](https://www.utest.com/#Streaming)
* [Networking Terms](https://www.utest.com/#Networking)
* [HTTP Error Codes](https://www.utest.com/#HTTP)
* [Gambling Terms](https://www.utest.com/#Gambling)
* [Cloud Computing Terms](https://www.utest.com/#Cloud)
* [Authentication Terms](https://www.utest.com/#Authenication)
* [Voice Testing Terms](https://www.utest.com/#Voice)
* [Other Terms](https://www.utest.com/#Other)

**uTest Terms**

**Accessibility Auditor (ACA)** - A tester that participates in audits by reporting accessibility bugs and completing test cases. They may still require some direction and assistance from other members of the team when testing and writing bug reports.  
**Accessibility Expert (ACE)** - A team member that leads numerous audits simultaneously, mentors other accessibility members of the team and presents final audit results to clients.  
**Bug Fix Verification (BFV)** - A process of verifying if a reported bug has been fixed when a fix or a new build for the product is released. Applause allows customers to run a re-test once a new build with fixes for those bugs is available, effectively verifying that the bug has been fixed.  
**Client or Customer** - A company that has engaged Applause and the uTest Community to test their product.  
**Community Engineer (CE)** - Members who work closely with the CM team and help with their tasks and responsibility.  
**Community Manager (CM)** - Members of the Community Management Team. The Community Management Team's goal is to help our global community to learn how to be excellent testers, to get opportunities for paid projects, and to connect with fellow testers across the community and the globe.  
**Environment** - Refers to device, OS, OS version, browser, or any specific setup that is used for testing.  
**Did Not Follow Instructions (DNFI)** - The reported bug did not follow the instructions given in the overview of a test cycle on uTest.  
**Dedicated Tester (DT)** - A tester whose function is to provide the team with speed, specialized expertise, and quality control. This role differs from the traditional role of our community testers and they are supposed to have a singular focus and dive deep. These testers should be seen as experts by their community peers and help support the TTL in chat management as an expert. Testers who have a strong history of quality work are more likely to be approached by a TSM/TE and asked about becoming a DT.  
**Duplicate (DUP)** - The issue is the same as one already reported in the current testing cycle or is a known issue.  
**Knowledge Base (KB)** - A self-serve online library of information about a product, service, department, or topic.  
**Known Issue (KI)** - Issues that are already been found in previous Test Cycles or the issues that the customer already knows about. Known issues are helpful to prevent duplicate submissions, and in order to avoid rejections, testers should always check them before starting testing. The known issues can be added in the Cycle in sort of a spreadsheet or you can recognize them by seeing a blue “bookmark” tag next to the issue to indicate that this is a Known Issue in the title column of the issues page.  
**Moderator** - In software testing, a moderator is a live instructor. Moderators are tasked with guiding individuals to use the product according to the criteria, answering questions, resolving issues, and gathering feedback.  
**Need more info (INF)** - A rejection type on uTest where a tester did not provide the requested information on an issue report when an info request was sent.  
**Non Disclosure Agreement (NDA)** - A legal binding agreement or contract by which two or more parties agree not to disclose confidential information that they have shared with each other.  
**Out of Scope (OOS)** - The reported bug falls into what is explicitly stated in the Out Of Scope section in the overview of the cycle.  
**Info Request (IR)** - More information is requested on a bug report or test case, or a tester is required to fix the bug report or test case.  
**Placeholder** - A submitted issue report without complete information or required attachments for the purpose of reserving the position of the report and later on edit and completing the report or changing it to a different bug.  
**Reproduction** - The process of recreating a bug by following the action performed steps in a bug report.  
**Solution Delivery Manager (SDM)** - The customer’s primary point of contact, and works with the customer directly.  
**Special Requirement Survey (SRS)** - A tool that is widely used by Testing Services and Community Management to recruit testers for test cycles where data points are needed that the platform doesn’t capture yet.  
**Test Case Writer (TCW)** - A person responsible for writing test cases and ensuring each test case has detailed, clear and easy to follow steps and expected results as well as verifying that the documentation and test cases provided by the customer are valid and can be used and making any necessary adjustments.  
**Test Engineer (TE)** - Builds the test cycle, assembles the testing team and is responsible for the overall execution of the test cycle. Also, the TE is responsible for marshaling the test team through completion of the test cycle achieving the goals of the request.  
**Test Team Lead (TTL)** - The primary point of contact for testers. The TTL helps testers within test cycles and reviews all submitted bug reports and test cases, etc.  
**Testing Service Manager (TSM)** - Works directly with the client. They manage a team of TTLs and TEs to identify and provide appropriate solutions for the client’s testing, feedback or research needs.  
**Triage/Triaging** - A process of reviewing a bug report or test case and then sending an info request or recommending the bug report or the test case for approval or rejection.  
**Turnaround Time** - the time limit when testing work should be submitted. For example, a slot or test case with a turnaround time of 6 hours should be submitted within 6 hours after claiming the slot or test case.  
**Won't Fix (WNF)** - Indicates that the issue is valid and approved but the customer is not interested in it or not planning to fix it.  
**Work as Designed (WAD)** - Expected behavior in the functionality or design of the product.

**Testing Terms**

**A/B Testing** - A testing method where two or more versions of a webpage element are compared by showing it to different segments of users to determine which version performs better based on specific metrics.  
**Accessibility (A11Y)** - The practice of designing and developing websites, applications, or other digital products and services to be usable by everyone, including those with disabilities. Accessibility focuses on providing an inclusive user experience, ensuring all users can access and interact with content easily.  
**Alpha Testing** - The initial phase of software testing which is typically conducted by the internal team before the release. The main goal is to identify and fix bugs early, evaluate the functionality, performance, and usability to meet the business requirements and be ready for further testing.  
**API Testing** - Validates Application Programming Interfaces (APIs). The purpose of API Testing is to check the functionality, reliability, performance, and security of the programming interfaces. It is one of the most challenging testing types at uTest  
**Beta Testing** - Testing conducted on real products in real environments prior to public release. Aimed at finding usability or functional problems in a controlled reporting environment.  
**Black Box Testing** - A software testing methodology where the tester performs testing with no prior knowledge of its internal workings. The tester can focus on testing from the user's perspective, by using the inputs and observing the outputs.  
**Bug Hunt Testing** - This is a robust exploratory test. The goal of this kind of testing is to only find out a specific bug or a specific bug on a specific device or a specific type of bug occurs within the testing scope. Testers invited to this cycle should carefully read and understand the overview and the requirements and they must avoid reporting issues that are not in scope.  
**Customer Experience (CE)** - This type of testing goes beyond the usability and user experience testing of websites, applications, and software. It aims to test a range of digital and physical products as well as services that span multiple online and offline points throughout the customer's journey.  
**Customer Journey Testing (CJT)** - A test aiming to evaluate the customer journey in a product, from accessing the product up until completing a certain task.  
**Design Verification Testing (DVT)** - A testing phase focuses on verifying that the design of the product meets the specified requirements and functions as intended.  
**Exploratory Testing (ET)** - A testing type that allows testers to use their knowledge and experience to identify areas of the product that may be vulnerable to potential failure. This type of testing also allows testers to think outside the box and try different approaches when testing the product.  
**Functional (FN)** - A type of testing on the features/functionality of a product with the intent of locating issues.  
**Live Testing** - Testing at a specific time, testers must perform testing at that time, they cannot be late or test earlier.  
**Load Testing** - A type of performance testing to check the system behavior and performance under a specific workload. This helps to determine how well the system can perform in realistic user loads before deployment.  
**Localization (LN)** - A type of testing that aims to verify the quality of a product in terms of a particular target culture/locale.  
**On-site Testing** - Visiting a physical location to evaluate the quality of the service and collect feedback.  
**Penetration Testing (PEN Testing)** - An authorized simulated attack performed on a product to evaluate its security.  
**Regression Testing** - A testing process designed to ensure that the previously fixed bugs do not resurface or introduce new issues after deploying the new changes. The process includes re-running the test cases on the modified software version and addressing any regression or new issues identified from the testing.  
**Sanity Testing** - A testing performed after a small change or bug fix to verify if the changes or fixes have been implemented correctly and do not introduce new issues before proceeding with further testing.  
**Slot** - A reserved position for a tester with specific requirements such as location, language, device, OS, payment method, specific setup to use, etc. It is very important to meet all of the requirements that are displayed in each slot before claiming the slot, otherwise, please don’t claim it.  
**Smoke Testing** - A preliminary testing process to check if the core functions of the deployed software build are stable or not. This process is useful to detect critical issues early and to improve the build quality before performing further testing.  
**Test Case** - A set of actions performed on a product to determine if it satisfies software requirements and functions correctly  
**User Experience (UX)** - Refers to the overall experience the user has throughout their interaction with the product. It may include user expectations, satisfaction, usability, accessibility, and other factors to make the product more easy to use and intuitive so users will have a more enjoyable experience when using the product.  
**Voice Testing** - Testing voice-enabled products with native speakers. It combines functional testing, dialogue verification, usability testing, and payment testing to help companies deliver voice experiences that foster ongoing customer engagement and satisfaction.

**Website/App Elements Terms**

**About Us** - The About Us page is commonly used by all types of businesses to give customers more insight into who is involved with a given business and exactly what it does.  
**Address Bar** (also known as URL Bar or Location Bar) - A text field typically at the top of a web browser that displays the current web address (URL) and is used to navigate to a web address by typing the address and pressing Enter.  
**Backend** - Refers to the server-side engine that handles data storage, processing, and management and sends the processed data back to the front end.  
**Bandwidth** - A measurement of the amount of data that can be transmitted over a network at any given time.  
**Banner** - A digital advertisement displayed in the header, footer, or sidebars of websites.  
**Binge** - A pop-up at the end of a TV show or movie leading to the next episode or trailer with similar content.  
**Because You Watched (BYW)** - Personalized suggestions based on the content a user already watched.  
**Breadcrumb** - A website navigation feature that shows the path from the previous pages to the current page using links.  
**Call to Action (CTA)** - Refers to a button or link that encourages the user to take a specific action. Examples of CTA in streaming services include:  
• Subscribe button, provides the user with subscription plans.  
• The Watch Now button directs the user to view the content.  
• Start Your Free Trial button, allows the user to try the service for free for a limited period of time.  
**Carousel** - A slideshow for cycling through a series of images or any visual content.  
**Cart/Bag/Basket** - The page where users can pile up what they want to buy from the app/website.  
**Captcha** - A challenge-response test to determine whether the user is human or an automated bot. It comes in various forms, such as distorted text, image recognition or audio verification.  
**Casting/AirPlay** - Casting allows viewers to stream video and audio content to a supported TV or sound system using Google Cast or Apple AirPlay. Viewers can use the cast feature of a player to stream content to a device on the same network if the casting feature is enabled for the player.  
**Checkbox** - A small interactive box that can be toggled by the user to indicate a positive or negative choice.  
**Closed Captions (CC)** - Time-synchronized text that represents the audio part of the video content. Closed captions are primarily used to help viewers who cannot hear the audio in a video. They can also be used when audio can't be heard clearly and for accessibility purposes. Closed captions are usually in the same language as the audio in the video.  
**Continue Watching (CW)** - A video playback feature that allows a user to continue watching a video from where they have stopped previously.  
**Cookie** - A small piece of data created by a website and stored on the user's computer or device when they visit the website. You may be asked to accept using cookies when opening a website or connecting to certain servers via a web browser for optimizing the website and its service.  
**Docked View** - Refers to a video window that stays fixed in a specific location on the screen, usually in a corner or at the bottom, while the rest of the screen can be utilized for other apps or use other features of the streaming service while watching the content. The difference between docked view and PiP is that the docked view display mode will remain in a fixed position, while PiP can be repositioned on the screen.  
**Drop-down menu/list** - A hidden list of options revealed by clicking a button or icon. The user can choose one value from the list.  
**Expanded View** - Refers to a video playback mode that allows you to see the video in a larger or more detailed view with more options, controls, and features displayed on the screen. This can include a list of related contents, a program guide, or more information and context about the content. In this mode, the playback controls such as pause, fast forward, volume controls, and other playback settings are usually hidden unless you tap on the player. This differs from the full-screen view, where the video playback takes up the entire screen.  
**Favorite Icon (Favicon)** - A small icon appears in the browser tab, search results, or bookmark to represent the website visually.  
**Field Validation** - A process of ascertaining that each field contains the correct value before the form is accepted.  
**Footer** - An area located at the bottom of every page on a website, below the main body content.  
**Frequently Asked Questions (FAQ)** - A section that lists commonly asked questions along with their answers compiled for a specific topic. The user can refer to this section to find answers to their questions before proceeding to ask for additional support if they can't find the answer they need.  
**Frontend** - Refers to the part of the website or application that the user sees and interacts with. It is responsible for presenting data to the user, handling the user input and interactions, and making the user interface visually appealing.  
**Full Event Replay (FER)** - Refers to the complete recording or playback of an entire event, typically in the context of sports or entertainment, allowing viewers to watch the event from start to finish without missing any moments.  
**Hamburger Menu** - Used to save space on the screen and denote a hidden menu.  
**Hero** - An oversized banner image at the top of a website. The hero image is often the first visual element a visitor encounters on the site; it presents an overview of its most important content.  
**Homepage** - The main page of a website; it is the first page that the users see when they load the website URL.  
**Immersive Highlights (IH)** - Content highlights such as sports that give control to the user over what they want to see from key positions, providing more freedom, choice, and personalization.  
**In-app notification** - Messages that appear within an app while the app is opened.  
**Landing Page** - A standalone webpage that opens after clicking on an ad, email link, or other advertising campaign. The purpose is to guide the user to take specific action, such as to purchase a product, sign up for a newsletter, or download a resource.  
**Left-Hand Navigation (LHN)** - Refers to the vertical menu positioned on the left side of the screen which provides the links to navigate through the various sections or pages of the website.  
**Live Window** - A content window in a live stream that allows the user to use the player controls during the Live Event stream.  
**More Like This (MLT)** - A feature that suggests items (e.g. products, movies) that are similar or related to the ones a user is currently viewing.  
**Navigation bar** - A horizontal or vertical menu bar that provides users with quick access to the main sections (sometimes to subsections) of a website.  
**Open Captions** - Subtitles that are permanently embedded into the video file. Unlike closed captions, open captions can't be turned off by the viewer and are always visible on the screen as they are part of the video.  
**Parental Control** - This is a group of settings that enable a parent to control what content a child or other user has access to and can watch on a device.  
**Permanent Link (Permalink)** - A URL that links to a specific webpage or blog post and is designed to remain the same even if the content of the page or post is updated.  
**Personal (or Personalized) URL (PURL)** - A unique web address created for an individual or specific target of a marketing campaign. PURL is commonly used in social media to create a URL for your profile or in marketing campaigns to create a customized landing page for the recipient.  
**Picture in Picture (PiP)** - A feature that allows the user to continue watching video content in a floating (small) window while the user does other things on the device. This feature is useful for multitasking, as the user can watch something while they browse the web, check emails, or do other tasks.  
**Playhead** - The line or marker to indicate the current position within a timeline or playback progress bar. Users typically can move the playhead by dragging it along the timeline or progress bar to navigate to different points within the media.  
**Pop-up** - A small window that pops up over the top of web pages.  
**Product Detail Page (PDP)** - The page with detailed information about a specific product.  
**Product Listing Page (PLP)** - The page where the products offered by the app/website are listed; this page often contains features like sort and filter.  
**Push Notification** - A short message that appears as a pop-up in a desktop browser, on the mobile home screen, or in the device notification center.  
**Quick Response (QR) Code** - A square-shaped barcode that stores information such as URLs, text, or location data. It can be scanned by digital devices (e.g. smartphones and tablets) to get access to the encoded information.  
**Radio Button** - An icon representing one of a set of options, only one of which can be selected at any time.  
**Rail** - A horizontally scrolling collection of tiles often organized by theme.  
**Really Simple Syndication (RSS)** - A web feed format that allows users to stay updated with new content or changes on the websites. The users need an RSS reader to easily read the feed as the feed is provided in XML format, which is uncommon for the user. RSS feeds are commonly used in news, blogs, social media, or podcast websites.  
**Scrub** - The action of jumping backward or forward by dragging the marker (playhead) on a timeline. Timelines are used in various applications to represent a sequence of events, such as a video, an audio recording, a project schedule, or an animation.  
**Single Live Event (SLE)** - A content event that streams one live event, for example, a live sports game, or news event.  
**Sitemap** - A page or file designed to list all the important pages and contents available on the website. It makes it easier for the user to navigate and find the content they are looking for.  
**Subtitles** - These are lines of dialogue or other text displayed in films, television programs, video games, or other visual media. Subtitles are primarily used to help viewers who do not understand the language being spoken in a video and they are usually available in a variety of languages for the same video.  
**Tabs** - A navigation element that allows multiple views to be contained within a single window.  
**Textbox/Text Field** - A section or object on a page that allows a user to enter text.  
**Thumbnails** - This refers to a small image that represents a larger image or video. In streaming apps, a thumbnail represents a video or content in a library or playlist. It's used to help the user identify the content quickly without having to open it and to put more interest in the user with visuals. Additionally, thumbnails refer to the image that appears above the timeline when scrubbing.  
**Toggle Button** - Allows the user to change a setting between two states.  
**Tooltip** - A block of information to specify or explain something when the user moves the mouse pointer over an element or clicks on the tooltip icon.  
**Uniform Resource Locator (URL)** - An address on the internet that works as a reference to a specific resource such as a webpage, document, image, video, or other resource accessible on the internet. It usually consists of Protocol, Domain, Path, Parameters, and Fragment.  
**User Generated Content (UGC)** - Content created by the user and shared online in the form of text, images, videos, reviews, or a podcast. This content can increase trust and provide credibility to the related product or service as it's provided directly by the consumers.  
**User Interface (UI)** - Refers to the visual and interactive elements the users interact with when using digital products. It includes buttons, icons, text fields, menus, and other interactive elements and all other visual elements that the user can see.  
**Video Start Time (VST)** - The time it takes from clicking on the Play button to the video starting to play. VST is commonly used in video streaming testing.  
**Website/App Modal** - A website/app page element that displays on top of a page and deactivates all other page content.  
**Website/App Overlay** - A content box that appears on top of a page on a website/app and obscures the background content.

**Software Development Life Cycle Terms**

**Android Debug Bridge (ADB)** - A programming tool used for the debugging of Android-based devices.  
**Alpha Build** - The first working version of a product with core functionalities implemented that is ready to test. This will often have a lot of bugs to find and some features might not be implemented in this build.  
**Application Programming Interface (API)** - A way for two or more computer programs or components to communicate with each other.  
**Azure DevOps** - A Microsoft product that provides version control, reporting, requirements management, project management, automated builds, testing and release management capabilities.  
**Bug Tracking System (BTS)** - An application that lets you keep track of bugs (and often suggestions) for your software project in a database.  
**Command Line Interface (CLI)** - A text-based user interface used to run programs, manage computer files, and interact with the computer. For example, CMD, Powershell, and Terminal are all CLIs.  
**DevOps** - The combination of cultural philosophies, practices, and tools that increases an organization's ability to deliver applications and services at high velocity.  
**Dogfooding** - In software development, using the software you developed. Dogfooding or “eating your own dog food” is testing your own software in beta form to assist in development.  
**Firebase** - A backend cloud computing service provided by Google that includes realtime database, cloud filestore. At times, testers are invited to Firebase in order to distribute latest builds of the product being tested.  
**Information Security (InfoSec)** - Refers to the processes and tools designed to protect digital data and information from unauthorized access, use, modification, disclosure, or destruction. InfoSec measures, such as being cautious before clicking on suspicious links or attachments in emails or the use of strong passwords, can reduce the risk of data breaches.  
**Jailbreaking** - The process of exploiting a device's flaws to install software that isn't available from the manufacturer. It is also known as cracking and rooting and allows the user to access the operating system’s root files to do modifications. It is not suggested that you buy a rooted or jailbroken device or root the device yourself as it will void the device warranty and even break the device OS.  
**Java** - A general purpose programming language commonly used in conjunction with web pages that feature animation. Small Java applications are called Java applets.  
**JavaScript** - A publicly available scripting language that shares many of the features of Java; it is used to add dynamic content (various types of interactivity) to web pages.  
**JIRA** - A proprietary product developed by Atlassian that allows bug tracking, issue tracking and agile project management.  
**Production Environment (PROD)** - This refers to the environment that is publicly available and ready for consumers, such as an app or website that is already tested and available for all users.  
**Quality Assurance (QA)** - A process involving efforts to assure that the product delivered to the customer meets the contractual and other agreed upon performance, design, reliability, and maintainability.  
**QA/Stage Environment** - This refers to a specific app build or environment for testing new features or fixes before being released to the public (Production environment).  
TestFlight - A tool that allows developers to distribute beta versions of their iOS apps to testers.  
**User Acceptance Testing (UAT)** - A phase of software development in which the software is tested in the real world by the intended audience or business representative.

**Streaming Terms**

**Buffering** - Refers to an issue that happens because the device does not receive data fast enough to keep up with the playback speed, resulting in the playback pausing frequently. A buffering video may also refer to the delay before streaming begins. Every video on the internet buffers when streaming, which is not necessarily a problem.  
**Connected TV (CTV)** - Refers to televisions that are connected to the internet, enabling the user to access streaming content from OTT providers such as Netflix, Amazon Prime Video, etc. This includes Smart TVs, which have built-in internet capabilities, and streaming devices, such as Amazon Fire TV, Apple TV, and Roku, which can be connected to conventional TVs to enable internet access. Also, it may be referred to as CD (Connected Devices) or living room devices.  
**Digital Rights Management (DRM)** - The use of technology to protect copyrights for digital media by not allowing it to be copied or recorded. Most apps for streaming services are DRM protected.  
**Freezing** - Refers to an issue where the video playback stops playing completely and does not resume even after waiting. In this condition, typically the viewer needs to restart the app to be able to resume playback.  
**HTTP Live Streaming (HLS)** - A widely used protocol for streaming video that runs on almost any server. HLS allows users' devices to adapt smoothly to changing network conditions by increasing or reducing the stream quality.  
**Linear Channel** - A scheduled or live broadcasting of an event.  
**Multichannel Video Programming Distributor (MVPD)** - A streaming service provider that provides multiple TV channels and VOD through cable, fiber, or satellite on a subscription basis. In contrast to OTT content, MVPD services are primarily viewed on TVs. A few examples of MVPDs are Xfinity, DISH Network, DirecTV, Spectrum, and Time Warner Cable.  
**Network Personal Video Recorder (NPVR)** - A video recorder feature that allows the user to save digital video recordings to the service provider's server instead of the user’s device.  
**Over The Top (OTT)** - Refers to the delivery method of film and TV content over the internet without the need for a traditional cable or satellite TV subscription. Consumers can access OTT content through streaming services such as Netflix, Disney+ and Hulu.  
**Personal Video Recorder (PVR)** - An electronic device that records video in a digital format to a disk drive, USB flash drive, memory card, or a network attached storage.  
**Stalls** - Refers to an issue that happens because there is not enough available bandwidth to continue playing the video. This can be due to high traffic, resulting in loading until the underlying issue is resolved or until more bandwidth becomes available.  
**Set Top Box (STB)** - A device that allows a television set to receive and decode digital television signals. The device is capable of connecting TVs to the home network for internet access and converting video from over the top providers into TV signals.  
**Upsell** - Upselling is a sales technique to entice users to buy a higher-priced or upgraded version of the product or account. There might be options for streaming services to offer non-subscribed users access to certain content by allowing them to subscribe, or they might offer annual subscriptions instead of monthly subscriptions.  
**Video Metadata** - Refers to the information associated with the video content, such as description, title, creation date, duration, and tags.  
**Video On Demand (VOD)** - Refers to a type of service that allows the user to watch video content on an internet-connected device whenever they want.

**Networking Terms**

**Domain Name System (DNS)** - The system that translates domain names (e.g. [www.academybugs.com](http://www.academybugs.com/)) into IP addresses (e.g. 192.168.0.1) so that the user can access the websites without having to remember their numerical IP addresses.  
**File Transfer Protocol (FTP)** - A standard communication protocol used for the transfer of computer files from a server to a client on a computer network.  
**Firewall** - A network security system that monitors and controls incoming and outgoing network traffic based on predetermined security rules.  
**Hypertext Transfer Protocol (HTTP)** - A set of rules for transferring files such as text, images, sound, video, and other multimedia files over the web.  
**Hypertext Transfer Protocol Secure (HTTPS)** - The secure version of HTTP that encrypts the data transmission process over the web. The use of HTTPS is important when the user transmits sensitive information such as login credentials, personal data, and payment details as it will be encrypted to reduce the risk of being intercepted.  
**Internet of Things (IoT)** - Refers to the network of physical objects (e.g. devices, vehicles, appliances) that are embedded with sensors, software, and other technologies that allow them to connect and exchange data with other devices and systems over the internet. IoT technology is being applied across various sectors, such as smart homes, healthcare, transportation, etc. With this technology, we can manage devices from afar, automate tasks, and improve our lives in general.  
**Internet Protocol address (IP address)** - A unique identifying number for every computer connected to the internet.  
**Internet Service Provider (ISP)** - An organization that provides services for accessing, using, managing, or participating in the internet.  
**Latency** - Usually measured in milliseconds, latency is the measure of delay or time it takes for data to reach a destination across a network.  
**Peer to Peer (P2P)** - A decentralized platform whereby two individuals interact directly with each other, without intermediation by a third party.  
**Proxy** - A service that acts as an intermediary between a client (device) requesting a resource and the server providing that resource in an internet network.  
**Server Message Block (SMB)** - A network protocol that allows users and applications on the same network to share files, access resources, and communicate.  
**Service Set Identifier (SSID)** - A name that identifies a wireless network.  
**Virtual Private Network (VPN)** - A tool that establishes a digital connection between a device and a remote server, creating a point-to-point tunnel that encrypts personal data, and masks the device IP address. This ensures that your online experiences are private, protected, and more secure.

**HTTP Error Codes**

**400 Bad Request** - Indicates that the server cannot or will not process the request due to an apparent client error (e.g., malformed request syntax, size too large, invalid request message framing, or deceptive request routing)  
**401 Unauthorized** - Indicates that the server cannot or will not process the request due to an apparent client error (e.g. malformed request syntax, size too large, invalid request message framing, or deceptive request routing).  
**403 Forbidden** - Indicates the server understands the request but refuses to authorize it. This error can be caused by IP restriction, incorrect permission, or other security measures. You can try to recheck the URL and clear the browser cache and cookies to resolve this issue.  
**404 Not Found** - Indicates that the browser was able to communicate with a given server, but the server could not find what was requested. The error may also be used when a server does not wish to disclose whether it has the requested information.  
**406 Not Acceptable** - Indicates that the requested resource is capable of generating only content not acceptable according to the Accept headers sent in the request  
**500 Internal Server Error** - Indicates that an unexpected condition was encountered and no more specific message is suitable.  
**503 Service Unavailable** - Indicates that the server cannot handle the request (because it is overloaded or down for maintenance). Generally, this is a temporary state.  
**504 Gateway Timeout** - Indicates that the server was acting as a gateway or proxy and did not receive a timely response from the upstream server.

**Gambling Terms**

**50** - 50 Split pot - A gambling term referring to the action when two or more players have identical hands, resulting in the pot being equally divided among them.  
**All-in** - This is when a player bets all their remaining chips on a single bet. If another player wants to call the bet, they must have enough chips to cover the full amount of the all-in bet.  
**Bet slip** - A gambling term referring to the ticket that includes and records all your bet information like the bets you place, the bet amounts and the odds for the bets.  
**Bet/Wager/Stake** - A gambling term referring to the amount of money that you risk on the result of an event or a competition.  
**Call** - A gambling term referring to the action when a player matches the current bet placed by another player in the same betting round. For example, if a player bets $10 and another player wants to continue playing, they would need to "call" and match the $10 bet.  
**Cash-out** - A gambling term referring to the action when a player exchanges their chips for real money when leaving a cash game or poker tournament. This is done at the end of a session or when a player is ready to leave the table.  
**Check** - A gambling term referring to the action when a player decides not to bet or raise but still wants to continue playing in the current round. The player can do this by indicating that they want to "check" and the betting will move to the next player.  
**Flop cards** - In Texas Hold'em and Omaha gambling, the "flop" is the first round of community cards dealt face up on the table. These are the first three community cards that all players can use to make their best hand.  
**Hand** - A gambling term referring to the set of cards a player holds in a poker game.  
**Lobby** - A gambling term referring to the area of an online casino where you can choose which games you would like to play.  
**Odds** - The probability of an outcome. Most often represented in a fractional (3/1) or decimal (1.35) form, or as a whole number (+200).  
**Raise** - A gambling term referring to the action when a player increases the current bet placed by another player in the same round of betting. For example, if a player bets $10 and another player wants to increase the bet to $20, they would need to "raise" the bet by $10.  
**River card** - A gambling term referring to the fifth and final community card dealt face up on the table in Texas Hold'em and Omaha. This card is dealt after the turn.  
**Run it twice** - A gambling term referring to the action when the remaining community cards are dealt twice in separate sequences after two or more players are all-in, with the pot split accordingly between the two sequences.  
**Side Bet** - A gambling term referring to a type of bet that can be placed in some casino games on an outcome that is not directly related to the result of the game. For example, in some blackjack variants, you can place a side bet on whether the total value of your first two cards will be higher or lower than 13.  
**Sit out** - A gambling term referring to the action when a player temporarily leaves the game without giving up their seat, allowing them to rejoin later. The player's chips will remain on the table, but they will not be dealt cards or participate in betting until they return to the game.  
**Turn card** - A gambling term referring to the fourth community card dealt face up on the table in Texas Hold'em and Omaha. This card is dealt after the flop and before the river.

**Cloud Computing Terms**

**Amazon Web Services (AWS)** - A subsidiary of Amazon that provides on-demand cloud computing platforms and APIs to individuals, companies, and governments.  
**Artificial Intelligence (AI)** - Synonymous with systems endowed with the intellectual processes characteristic of humans, such as the ability to reason, discover meaning, generalize, or learn from past experience. AI is the ability of a digital computer or computer-controlled robot to perform tasks commonly associated with intelligent beings.  
**Machine Learning (ML)** - A branch within AI and computer science focused on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.  
**Software as a Service (SaaS)** - Software based on the cloud that delivers the app to the user over the internet. Users can access the software through a web browser without having to install and usually, the software is provided on subscription-based pricing models. (e.g. Slack, Google Workspace, DocuSign)

**Authentication Terms**

\*\*Biometric Authentication\*\* - An authentication method using unique physical or behavioral traits such as fingerprints, facial features, iris, or voice. This method is more secure, convenient, personalized, and efficient than the traditional method. \*\*Multi-Factor Authentication (MFA)\*\* - A security process that requires two or more verification factors when logging in. \*\*One-time password (OTP)\*\* - A temporary numeric or alphanumeric code sent to a user via SMS or email that is valid only for one session. An OTP can be used only once. \*\*Single Sign On (SSO)\*\* - An authentication service that allows the user to access and use all their digital services and applications with just one login. The use of SSO can eliminate the need for users to manage multiple passwords as the users only need to remember one set of credentials and improve security by centralizing the authentication. \*\*Two-factor Authentication (2FA)\*\* - A security process in which a user provides two different authentications when logging in, for example, a password and a verification code sent to a registered phone number or email.

**Voice Testing Terms**

**Automatic Speech Recognition (ASR)** - Technology that enables users of information systems to speak input rather than typing it in. Aka computer speech recognition or speech-to-text.  
**Dictation** - Speech dictation is saying words with the intent those spoken words will be written down. In development, dictation refers to text-to-speech, wherein spoken words will be transcribed by machine learning.  
**Intent** - A task or an action that a user requests to be fulfilled by a voice application.  
**Natural Language Processing (NLP)** - A branch of AI intended to program computers with the same comprehension of spoken words as humans. NLP combines computational linguistics with statistical, machine learning and deep learning models. Together, these technologies enable computers to process human language in the form of text or voice data and to “understand” its full meaning, including the speaker or writer’s intent and sentiment.  
**Utterance** - A small unit of speech. The stretch of speech or utterance is measured by clean breaks before and after.  
**Voice Assistant** - A digital assistant that uses voice recognition, speech synthesis and natural language processing (NLP) to provide a service through a particular application.  
**Wake word** - The phrase used to trigger a voice assistant. When a Wake word is detected, the system will engage in capturing, decoding and servicing the request.

**Other Terms**

**Adobe Experience Manager (AEM)** - An enterprise content management system that optimizes the authoring, management, and delivery of content and digital media.  
**Always On Display (AOD)** - A feature available on certain smartphones that allows the user to view the time, date, notifications, missed calls, and more when the screen is turned off.  
**Augmented Reality (AR)** - The integration of digital information with the user's environment in real time.  
**Compress** - The process of making a file smaller so that it will save disk space and transfer faster over a network.  
**Direct to Customer (D2C)** - A business model of selling products or services directly to customers without the use of third-party retailers, wholesalers, or middlemen.  
**Dots Per Inch (DPI)** - A measure of a device's resolution.  
**Emulation** - The ability of a program or device to imitate another program or device.  
**End User** - Refers to the individual or groups of individuals who will use the product or service. They are the final consumers who interact with the product and become the consideration for the development team to design and improve the product.  
**First Time User (FTU)** - An individual who for the very first time uses a product. FTU are often shown a tutorial or tooltips on how to use the product.  
**Heads-up Display (HUD)** - A transparent display that presents data without requiring users to look away from their usual viewpoints, e.g. playback progress bar, sports score card.  
**Hosting** - Refers to the service that provides storage and resources required for a website to be accessible to others via the internet.  
**Incognito** - A browsing feature that allows the user to privately browse the internet as the browser doesn't save your browsing history, cookies, or other temporary data during the browsing session.  
**Know Your Customer (KYC)** - The process of identifying and verifying user identity, KYC is common in financial services.  
**Open Source Software (OSS)** - Refers to the software whose source code is designed to be accessible for anyone to inspect, modify, enhance, and distribute under an open source license.  
**Original Equipment Manufacturer (OEM)** - An organization that makes devices from component parts bought from other organizations.  
**Payment Instrument (PI)** - A physical or virtual way to transfer or pay money or monetary value.  
**Smart Assistant** - Software typically installed on a smart device that provides various services when triggered (often through built-in active listening triggered by a Wake word), activating and performing certain tasks.  
**Unique Device Identifier (UDID)** - A unique identifier for Apple devices such as iPhones, iPads, Apple TV, MacBooks, etc.  
**User Experience Researcher (UXR)** - A researcher that systematically studies users to collect and analyze feedback that will help improve the product design and user experience.  
**Virtual Reality (VR)** - The use of computer modeling and simulation that enables a person to interact with an artificial three dimensional visual or other sensory environment.  
**Whitelist (or Allowlist)** - A security measure used to restrict access to resources to only those who have been pre-approved. In testing, one common use of whitelisting is for email addresses. Usually, testers will be asked to send their email and the company will whitelist them for testers to be able to access the testing website or app or specific features.  
**Work in Progress (WIP)** - An indication that the work has not yet completed and is still being worked on.

**The Importance of Reading**

Reading is important because it develops the mind, imagination, creativity and it enhance your life in several ways. Developing critical thinking and improving testing skills can also be achieved through reading.

**Why Reading Instructions Is Important At uTest?**

At uTest, reading and understanding the requirements and instructions is crucial and would help you to stand out among others as well as being a professional, hard worker, reliable, motivated, polite, and going above and beyond expectations.

Reading and understanding the instructions would help you learn the project goals and what is expected of you as well as help you to find very valuable bugs, write well-written reports and execute the test case correctly.

**When and What Instructions Should be Carefully Reviewed?**

At uTest, testers can apply for projects on the Projects Board as well as respond to Special Requirement Surveys (SRS) when an SRS has been sent to the tester, and if meet the requirement a tester might be invited to a test cycle. Please ensure to do the following when responding to SRS or applying for a project on the Projects Board.

**Special Requirements Surveys (SRS)**

Before answering the SRS questions be sure you have read and understood the question, then honestly and correctly answer the SRS questions. Do not falsify or randomly select answers to the questions. When asked to update your tester profile in an SRS make sure to correctly update your profile. You may proceed to the next question if the profile question you were asked is already up-to-date on your profile.

**Applying For Projects On Projects Board**

Before applying for Projects on the Projects Board be sure to carefully review the requirements and details in the project description and details. After determining that you meet the requirements, proceed with applying for the project.

Fill out the project form correctly with accurate and complete information and double check all fields before submitting it.

Once invited to a test cycle and upon carefully reading and understanding the cycle requirements and accepting the cycle invitation, it is recommended to review the instructions once more before completing specific tasks in the cycle. The most important sections to review and when are listed below.

**Slot Instructions**

You should read and understand the slotting instructions before claiming a slot, as well as the description of the slot you intend to claim, to ensure that you fully understand the requirements and rules regarding claiming slots for the cycle and the specific requirements for the slot you wish to claim. Be sure to expand the slot description by clicking on it prior to claiming it.

**Setup Instructions**

Before starting to test, submit bugs or execute a test case, make sure to review the Setup Instructions in the cycle overview. This section will contain instructions, if there are any, about specific requirements a tester needs to follow for setting up the environment for testing, how to access the test environment, or how to install the test app.

**In Scope**

Prior to reporting a bug, make sure to review the In scope section very carefully to understand the cycle's goals and ensure that the issue you have found falls within the cycle's scope.

**Out Of Scope**

Make sure to review the Out of Scope section very carefully before reporting a bug to ensure that the issue you have found does not fall under the Out of Scope areas and should not be reported. Also, review the Known Issues, if attached to the cycle, to ensure that the issue found is not a Known Issue.

**Issue Reporting Instructions**

Review the Issue Reporting Instructions before reporting a bug and make sure your bug report meets the requirements in this section such as:

* Issue Title format
* Required Attachments
* Additional environment info, if required

Also, be sure to review the bug report form and correctly fill the fields with correct information, especially when there are custom fields in the cycle or any special requirements.

**Test Case Instructions**

You can see this section in Exploratory with Test Cases cycles. In this section, you can find specific requirements for test cases and other rules that you must follow to ensure that you execute the test case correctly if there are any.

**UX Survey Responses**

It is extremely important to read the instruction text in the survey thoroughly, even (and especially) if they’re long. They might have instructions on when to start a video recording or what questions to answer before doing a certain task. If you don’t follow those instructions, you risk getting the whole survey rejected because the answers are no longer valid.

**Cycle Announcements**

During the cycle, announcements might be sent with further instructions, requirements, advice, warnings or requests. Be sure to immediately read and understand all the announcements that are sent through the cycle as well as follow the instructions in the announcement and acknowledge the announcement if applicable.

**Info Requests & Tester Messenger Messages**

Once you have submitted an issue report or a test case a message or more info could be requested, so be sure to read the info request or message carefully and respond to the message and provide the required information. Upon providing the necessary information or fixing your work, click the"Confirm all requested info is added" link to let the TTL, TE or customer know your work is ready for further review.

Ensure you read and understand the other sections of the cycle, including:

* Team Contact Information
* Special Instructions
* Tester Special Requirement
* Bonus Instructions
* Review questions, if there is a review in the cycle
* As well as any email from the cycle TE or from uTest

**What Is Recommended To Do?**

* Make sure you read extensively, intensively and thoroughly
* Pay very close attention to instructions and details
* It is important that you read and understand the requirements before accepting a cycle invitation and before you begin testing
* Keep it a habit to read the instructions more than once to fully understand the cycle requirements and what is expected from you
* The test case must be executed according to the instructions
* Double check your work to ensure it meets the requirements of the cycle
* Read and understand the cycle announcements and take action as required
* Whenever the instructions in the cycle, test case or announcements are unclear, ask the TTL in the cycle chat room
* Immediately respond to info requests and messages in your bug reports and test cases
* Ensure your issue reports and test cases meet the cycle requirements and you have followed all the instructions on your work
* Frequently check the cycle chat room and check previously asked questions before asking a new question
* Review issues already reported and Known Issues in the cycle before reporting a new issue
* Correctly collect and upload all required attachments to the bug report and test case
* Keep your tester profile updated and provide accurate information every time
* Again, read extensively, intensively and pay close attention to requirements and instructions

**What Should You Avoid?**

* Do Not skim, scan or quick/speed reading
* Do Not skip any small details when reading the instructions or requirements
* Do Not rush through reading the cycle overview, scope and expectations
* Do Not claim a slot without reading and understanding the requirements
* Do Not claim a slot and sit on it without making any progress
* Do Not execute the test case with false information and with the wrong environment
* Do Not ignore cycle announcements
* Do Not ignore info requests and messages in your issue reports and test cases. Issue reports and test cases with mistakes are at risk of rejection
* Do Not apply to projects that you do not meet the requirements
* Do Not respond to SRS or apply for projects with false and/or inaccurate details
* Do Not repeatedly ask the same question that was already asked and answered in the cycle chat room

**The Team Behind Test Cycles**

Team players behind each cycle play an essential role in ensuring the cycle produces the best outcome and goes beyond. Here are their roles in each cycle:

**Testers**

Testers are the most important role in the test cycle. They find bugs, execute test cases, provide feedback and report any issues related to the tested product.

**Responsibility/Duties/Tasks**

* Understanding the product under testing
* Reporting bugs, executing test cases, submitting reviews and other testing tasks available in the test cycle
* Responding to any info request asked as soon as possible
* Reporting high-quality issue reports
* Finalizing test cases within the deadline
* Reviewing the overview and reporting blocking issues to the TTL
* Helping fellow testers in the chat room if you know the answer

**What Testers Should Do**

* Follow the[uTest Guidelines](https://www.utest.com/utest-guidelines)
* Be professional and polite when communicating with others
* Follow test cycle instructions
* Utilize cycle chat properly
* Complete assigned tasks thoroughly and honestly
* Accept the test cycle when having a commitment to the cycle
* Report issues within the scope
* Claim slots that match your environment
* Submit test case after being done executing it
* Provide feedback thoroughly in the review
* You can read the [uTest Guidelines](https://www.utest.com/utest-guidelines" \t "_blank) for other points

**What Testers Should Not Do**

* Violate the [uTest Guidelines](https://www.utest.com/utest-guidelines)
* Dishonesty
* Publicize customer information
* Discuss payout in front of customers
* Create more than one uTest account
* Falsify any information
* Using a VPN
* Ask to approve the work submitted
* Report invalid issues
  + Always check the already reported issues, the known issues list attached to the cycle, and the OOS section
* Claim slots until the maximum limit that is allowed to prevent other testers from claiming them
* Claiming slots and sit on them without making any progress
* Execute the test cases incorrectly
* Unclaim the test case suddenly and then idle in the cycle
* Provide invalid attachments in the report or test case
* Use add-ons, scripts, etc. intentionally to find issues
* Dispute any rejected bugs without reasons
* Being rude and unprofessional
* Do not use All caps to draw attention
* You can read the [uTest Guidelines](https://www.utest.com/utest-guidelines" \t "_blank) for other points

**Test Team Lead (TTL)**

TTLs are hand-selected members of the uTest community who work side-by-side with TSMs and TEs to facilitate the flow and outcome of test cycles. Although the tasks required per test cycle may differ, the goal is the same: to increase the value that uTest brings to the customer by maximizing the output and minimizing the noise level.

TTLs are the primary contact for testers that can be reached through the cycle chat. They are responsible for helping testers within the test cycle, reviewing all submitted bug reports, test cases, etc., and escalating problems to the Test Engineer if needed.

**Responsibility/Duties/Tasks**

* Communicating with the testers during a test cycle
* Triaging the issue reports and test cases submitted by the testers in a timely manner
* Supporting the Test Engineer with any duties assigned
* Reviewing the test cycle overview and escalating any issues to the TE
* Shortly testing the product being tested to ensure there are no blockers
* Monitoring general cycle progress and the cycle chat
* Assisting testers with resolving questions and problems
* Reviews bug disputes and provide recommendations

**What Testers Expect from TTL**

* Being helpful, polite, patient and responsive when communicating with testers
* Bugs and test cases submitted are reviewed in a timely manner
* Should answer any questions regarding the cycle that adheres to the [uTest Guidelines](https://www.utest.com/utest-guidelines" \t "_blank)
* Private chats can be used only for personal topics that can not be discussed in the general chat room

**What Testers Should Not Expect from TTL**

* Questions and/or issues related to the payouts should not be asked to TTL, but TE or TSM instead
* Bugs and test cases will not be reviewed or approved based on testers' request
* Questions directed to TTL should only be asked in the cycle chat. Other contact methods can be used only if allowed by TE or TSM of the project

**Test Engineer (TE)**

Test Engineers build and manage the test cycles, assemble the testing team and are responsible for the overall execution of the test cycle. Also, they ensure that the team achieves the best outcome possible for the customer.

**Responsibility/Duties/Tasks**

* Setting up test cycles
* Monitoring cycle progress
* Recruiting community test teams
* Ensure the testing and triage are progressing as scheduled
* Helping resolve testers' issues and escalating to the TSM if needed
* Resolve disputed issues
* Write and adjust test cases
* Writing Special Requirements Surveys (SRS)
* Taking pre-defined actions to bring a cycle back on track
* Working directly with TSM to achieve strategic goals
* Operate as a TTL for some projects

**Testing Services Manager (TSM)**

They work directly with the client and manage a team of TTLs and TEs to provide the appropriate solution for the client's testing, feedback, or research needs. Also, they are responsible for making financial decisions in the test cycle.

**Responsibility/Duties/Tasks**

* Working with the TEs to develop test strategies
* Managing teams of TEs, TTLs, DTs and TCWs
* Monitoring account finances and handling payouts
* Ensuring testing meets the expectations of the customer
* Providing directions to the TEs
* Monitoring account spending
* Processing and approving payments

**Scenarios & Solutions For Potential Issues**

**Tester did not get an answer from TTL at a specific time**

If you did not get an answer in a timely manner or within the time frame stated in the team contact information in the cycle overview, you can email the TE of the cycle. The TE email is also in the team contact information section in the overview. After that, please wait until the TE responds to your query.

**TE did not respond to the tester**

If you did not receive a reply from TE during the timeframe as specified in the team contact information section in the overview, you can proceed to contact TSM. You can also find the TSM email under the team contact information section.

Make sure to provide a clear explanation and provide all the necessary info such as Cycle ID, Cycle name, Tester ID, Bug report ID, Test Case ID, etc when sending an email to the TE or TSM.

**TSM did not respond to the tester**

This is a rare situation where you do not receive any reply from the TTLs, TEs or TSMs. You can recheck your query to see if it adheres to the [uTest Guidelines](https://www.utest.com/utest-guidelines). If the query is already according to it and no response is received, you can proceed to Tester Support to ask for help (<https://support.utest.com/csp> > Request Support > Test Cycle Issues for the Request Category) and they will help you with your query.

Make sure to provide a clear explanation and provide all the necessary info such as Cycle ID, Cycle name, Tester ID, Bug report ID, Test Case ID, etc when you submit a support ticket.

**Customer rejected bugs fairly or unfairly**

Invalid bugs should be rejected reasonably by the customer. You can check the rejection reason and you have the right to politely dispute if you disagree with the rejection. Check this [Course](https://www.utest.com/academy/tracks/17/courses/best-practices-bug-reports) for more information

In case bugs are unfairly rejected, you can check the rejection reasons first before escalating it. After you confirm the situation, you can collect the evidence in the form of screenshots or videos and you can then email the TE to escalate the issue regarding unfair rejections. As mentioned before, when you do not get a reply, you can proceed to escalate to TSM. They will discuss and provide a solution regarding this matter.

**High-severity issues approved with low value**

Please note that it is up to the customer to determine how much the bug is valuable to them and that is something we do not have control over. However, the same as the previous point can apply here and you can escalate it to TE. But, first, make sure to recheck your report and ensure that it is categorized as high severity and in focus on the cycle overview (crash issue, for example). Collect any supporting evidence. After that, you can email TE and include the attachments as well. They will take the necessary actions and will provide appropriate feedback on your issue.

**Issues approved as won't fix (WNF)**

Won't Fix, means that the issue is valid, however, the customer would not allocate the time and effort into fixing the issue and focus on other kinds of issues. In case many issues are accepted as WNF even though they are within the testing focus stated in the overview or it is related to the test case steps or blocking issues, you can ask for confirmation from the TE whether the assessment of the issues has been done correctly or not.

**Issues not approved or rejected after a long time**

If your issues have not been approved or rejected after the cycle locks, you can wait up to 15 days as the cycle will automatically be closed and your issues will automatically be approved or rejected based on the TTL recommendation. This also applies to your test case. For reviews, all will be accepted if they are not checked until the cycle is closed.

In case the issues are not yet checked or are still in new status even after the cycle locks, you can let the TE know if there might be an unforeseen issue.

**Test Case unclaimed/rejected for no reason**

If your test case has been unclaimed/rejected for no reason, even though you have made progress or completed the test case correctly according to the provided instructions, you can email the TE. There are various reasons that can lead this to happen:

* The slot is automatically unclaimed because it was not submitted within the allotted time
* Mistakenly unclaimed or rejected
* Test case is poorly executed, but the team forgot to put the reason when unclaiming or rejecting it
* There are updates in the test case and TE/TSM decided to unclaim all the test cases but did not put a clear reason. You can expect an announcement regarding this issue as you might be asked to claim a new one

**Getting blocked from testing due to a problem with the product being tested**

You can inform TTL when you face a blocking issue and they will escalate it to the TE/TSM. Also, you can try to find a workaround for the issue and let the team know as well.

**Regarding one-time issues**

For this kind of issue, if you happen to be able to collect the proof, a video and logs, you can report the issue even though it is hard for you to reproduce it again. For example, when you try to reproduce an issue while recording it, the app suddenly crashes. Since you have video and logs generated from it, you can report the issue. Before that, ensure that one-time or non-reproducible issues are not listed under the OOS section, and you can check with the TTL for confirmation.

**Issues that get fixed by themselves or temporary issues**

If the issue is fixed when rechecking it after some time or is a temporary issue, as a tester, if you notice this while the issue is not yet reviewed you can discard it. If you notice after being asked for reproduction, then you can let the TTL know that it is no longer reproducible through the Messages tab of the issue.

**Being asked to discard the report even though it's reproducible**

If the issue was reproducible for you but no longer when asked to reproduce it, and then you were asked to discard the bug for that reason, you can inform the TE to reconsider it since the issue was valid and reproducible by the time you reported the issue. However, the bug report might get rejected because it might be a temporary issue on the website/app, or related to your device, browser, etc. So, if you are confident that it is not the case, then you might collect the required evidence and report to the TE to investigate and the TE will look into the issue and provide feedback to you.

You can also inform the TE in a situation where the customer provided the wrong build or asked to test the new build while the previously reported bugs are no longer reproducible in the new build. Since this is not your mistake, you can dispute if your reported issues turn out rejected, as it's no longer reproducible.

**Verifying an email is legit (for example, the email mentioned a project and offered to join)**

To identify that an email is from uTest, you can check the sender's email address. You can do that by opening the email and checking the sender's email beside the name or you can click the down arrow icon to show more detailed info and check the 'from' line. Note that uTest will only send emails from these domains:

* uTest.com
* m.uTest.com
* Applause.com
* Applausemail.com

For example, you received an email from [us@utest.com](mailto:us@utest.com), the domain for this email is utest.com, so it is legit. And if you notice that the email does not use the mentioned domains, you can be sure that they are not legit. You can also report it by submitting a ticket from <https://support.utest.com/csp> > Request Support > Request Category > Potential Cheating/Fraud/Misbehaviour > Fill out the form accordingly. You can check the [Tester Security Course](https://www.utest.com/academy/tracks/30/courses/tester-security) for more info

**Useful Courses to Review**

* [uTest Basics](https://www.utest.com/academy/tracks/30)
* [Testing Basics](https://www.utest.com/academy/tracks/34)
* [uTest Cycle Process](https://www.utest.com/academy/tracks/31)
* [Slots, Test Cases and Reviews](https://www.utest.com/academy/tracks/33)
* [Bug Reports](https://www.utest.com/academy/tracks/17)

**Building a Positive Reputation at uTest**

**Your Reputation at uTest**

Building a positive reputation at uTest is an important part of achieving success as a tester.

The Test Engineers that setup and manage uTest projects rely on testers who contribute by reporting bugs, completing test cases and participating in the project chat room.

Testers that consistently contribute in uTest projects set themselves apart as a reliable tester and are invited again and again to uTest projects.

**You can start to build a positive reputation by:**

1. Following the instructions in the project overview
2. Writing quality bug reports
3. Completing test cases correctly and on time
4. And responding quickly to messages from the TTL or TE

The next time you receive a uTest project invitation keep these points in mind as you work to build your reputation in the uTest community.

**How to Build a Positive Reputation**

There are many different ways to positively enhance your reputation. Let’s look at 3 specific things you can do to build your reputation as a quality tester in the uTest community:

1. Pay attention to details
2. Be reliable
3. Submit quality work

**Paying attention to details means**

* Reading and following all of the instructions in the project overview
* Being aware of any updates or changes to the project by reading the project announcements and chat room messages

**Being reliable means**

* Actively participating in the projects that you accept
* Completing your work correctly and on time

**Submitting quality work means**

* Providing high value bug reports that follow the project requirements
* Completing test cases and reviews with useful information within the time limit of the project

Every uTest project invitation is an opportunity for you to contribute and build your reputation as a quality tester in the community. Remember to pay attention to details, be reliable and submit quality work.

**Anger Management, Issue Comprehension and Resolution**

The key to being a good tester is to remain calm and collected. However, this doesn’t mean that you are not allowed to be upset when mistreated.

The goal of this course is to give guidance on how to handle situations that trigger your anger to avoid doing unexpected behaviors that could violate uTest Guidelines, leading to account suspension or termination. When we can express our emotions without losing control, our needs will likely be met and we will likely improve future outcomes.

Once you participate in test cycles, as a tester you might find yourself in various situations because of different requirements, meeting different team members or new customers, etc. There may be a time when you have put in your best efforts and yet the results are not as expected. You might feel that an issue should have been rejected as **WAD** but it got rejected as **Other** instead which affects your rating.

There may also be a time when you spent a lot of time completing the test case but it got rejected or unclaimed without a clear reason. In these times, you might get irritated and want everything to be resolved as soon as possible according to your expectations. We can understand your feelings as we ever faced the same issues. In these situations, please calm down and follow the below tips.

**Avoid Doing Anything When Angry**

If anger controls you, you will likely want to act something to express it immediately whether you are aware or not, and the act likely will leave regret for you and hurt others. If you are aware of this situation, then avoid doing anything you want to do. You can move your hands from the keyboard and stand up if that can stop you and make you feel better.

**Take a Timeout**

Time may help you calm down, so you can take a break until you feel better. You can also try other techniques that can help you cool down, such as doing physical activities, taking deep breaths, or washing your face to help you feel refreshed and improve your mood.

**Identify and Understand the Issue**

Once you feel better and have a clearer head, check again to see what is the cause of the problem.

* Were you angry after seeing a TTL or TE response?
* Were you not in agreement with the TE decision?
* Did you misunderstand an instruction but get irritated after rechecking?

You have to eliminate any negative thoughts that fuel your anger, only then will you be able to see the issue in a broader light and gain a better perspective on it. If you keep thinking “It should or must be like this” then the conversation will not go anywhere because you will likely get angry when the fact doesn’t line up with your expectations. Learn to reframe the way you think and try to ask yourself to determine if what you are about to say is true, valid, relevant, and fundamental.

If you are ready to move to the next step, be ready to accept any outcomes despite the efforts and **believe the efforts will not be wasted**. The next step is to try to understand the message or the decision. The first thing worth doing is to double-check the cycle overview thoroughly and carefully, as well as the test case and slot instructions, or any other information that has been provided, and then reevaluate them. By rechecking the instructions, you will gain a better understanding of the goal of the cycle and can view the issue from another's point of view.

After you have understood the message or made a decision, it might be easier for you to come to a conclusion and focus on your work again. You will also have a better understanding of the company’s preferences and be better prepared for future projects. You might also find it useful to understand the following:

* Each test cycle or project might have its own requirements and conditions.
* It’s necessary to redo the test if you mistakenly execute the instructions as it affects the results.
* It’s necessary to unclaim the slot if you mistakenly claim a wrong slot as it will provide invalid results.
* Following uTest Guidelines is a must and violating it may end up in a termination of the account.
* You need to treat everyone with respect and always be professional and polite when communicating.

Avoid discussing your concerns or disagreements toward the message or decision in the cycle chat room, or messages tab in the issue report, test cases, or reviews. Also, refrain from seeking others' agreement in the community or escalating any issues directly to TSM or higher without further understanding of the issue. These actions can unnecessarily complicate the situations without leading to a concrete resolution and you might even violate the guidelines without realizing it in the heated discussion.

The key to resolving the issue is good communication and good communication can only be achieved if you have your anger under control and can accept other people’s perspectives and discuss the issue in a professional and good manner.

**Possible Issues and Solutions**

After understanding the message or the decision, if you think they are not aligned with the instructions or any information provided, you may proceed to the next step, which is disputing your issue report if it’s related to the issue report, or escalate to the TE if it’s related to other problems, such as test cases or reviews being rejected or unclaimed without a clear reason or TTL misconduct, etc.

If the issue is related to your report, which means you disagree with the rejection reason, then you may dispute the report. Remember that you should check the rejection reason in your issue report, not only the rejection type. Check here to learn more about [Disputes](https://www.utest.com/academy/tracks/17/courses/bug-rejections).

If the issue is related to other problems, you may need to escalate that first to the TE of the cycle according to the escalation rules mentioned under the Team Contact Information section. Note that escalation would be better avoided unless it is something urgent or if it needs attention and help from a higher-up to resolve the issue.

Next, prepare what you would like to send. Gather any useful information that can support your claim and provide screenshots if needed. Check the testing website or app, any provided documentation, and other relevant information.

Once you gather your thoughts and evidence, start drafting the email. Note that you can find the TE email provided in the cycle overview. Don’t use other TE emails on other cycles as they are not responsible for all cycles.  
When writing the email, keep in mind the following:

* Mention the cycle ID in the email subject.
* Make sure to be polite and professional when writing the content to state your concerns.
* Explain your concern clearly and focus on the issue.
* Avoid prolonging the message without a clear point.
* Recheck your writing and ensure that it’s clear and easy to understand.
* Don’t CC or BCC the email to other TEs or TSMs.

Note that TSM should only be contacted if there is an issue with the payout, it's an emergency, or if you haven’t received a reply from TE after the expected reply time as mentioned in the escalation rules.

Be assured that the project team is committed to resolving the issue promptly. If you have any feedback on the situation, feel free to share your thoughts in the email. Your feedback on the situation is valuable to the team as it may help improve the processes.

**Possible Outcomes**

Outcome may come in three ways, accepted, rejected, or future consideration and improvement. Understand that not all points may be accepted and some feedback may need time to be applied. At this point, you are expected to be calm and rational, it’s no more “I am always right” or to interpret other perspectives as incorrect. If you can’t overcome it, then you should go back to the first and second points or better stop here as it may affect your mental health. If you believe you can accept any outcomes, other perspectives, or disagreement, you can continue reading this point.

First, about your dispute. If your dispute got rejected, make sure to read the rejection reason carefully. You are expected to get a more detailed reason for the rejection reason after a dispute. If the rejection reason is reasonable then you need to accept it and move on. We believe that accepting it will also make us a better tester and we can avoid reporting a similar issue for the customer if it should be avoided.

Next, about the email. If you receive a response from the TE, read the message carefully. If there is a need to do something on your part and you are willing to assist, help the TE so that it can be resolved. If they will resolve it on their end, you can wait if it will be resolved during the cycle duration or in the future if it’s considered.

In any case, ensure to keep your communication professional. If your temper surfaces, try to cool down again before replying to the message. Also, keep the communication within a thread and avoid making a new email thread to respond to the TE message.

**Note**: If your issue is an emergency and you haven’t received a reply after the expected reply time from TE or TSM, you may seek help from Tester Support. You can open [Tester Support](https://support.utest.com/csp) Portal > Select Request Support > Select Test Cycle Issues for the request category and then complete the form accordingly.

**Be Proud of Yourself**

If you can handle your anger appropriately while facing challenging issues, let us congratulate you. Anger is a difficult emotion to overcome, yet it is an important accomplishment in life. You are not only being professional but also shaping the community for the better.

We hope these tips help you to maintain a healthy habit so that you can naturally respond calmly and professionally in the future. Also, you can handle challenging situations effectively, achieve the most effective outcomes, and contribute to the improvement of future testing.

**Anger Management, Issue Comprehension and Resolution**

The key to being a good tester is to remain calm and collected. However, this doesn’t mean that you are not allowed to be upset when mistreated.

The goal of this course is to give guidance on how to handle situations that trigger your anger to avoid doing unexpected behaviors that could violate uTest Guidelines, leading to account suspension or termination. When we can express our emotions without losing control, our needs will likely be met and we will likely improve future outcomes.

Once you participate in test cycles, as a tester you might find yourself in various situations because of different requirements, meeting different team members or new customers, etc. There may be a time when you have put in your best efforts and yet the results are not as expected. You might feel that an issue should have been rejected as **WAD** but it got rejected as **Other** instead which affects your rating.

There may also be a time when you spent a lot of time completing the test case but it got rejected or unclaimed without a clear reason. In these times, you might get irritated and want everything to be resolved as soon as possible according to your expectations. We can understand your feelings as we ever faced the same issues. In these situations, please calm down and follow the below tips.

**Avoid Doing Anything When Angry**

If anger controls you, you will likely want to act something to express it immediately whether you are aware or not, and the act likely will leave regret for you and hurt others. If you are aware of this situation, then avoid doing anything you want to do. You can move your hands from the keyboard and stand up if that can stop you and make you feel better.

**Take a Timeout**

Time may help you calm down, so you can take a break until you feel better. You can also try other techniques that can help you cool down, such as doing physical activities, taking deep breaths, or washing your face to help you feel refreshed and improve your mood.

**Identify and Understand the Issue**

Once you feel better and have a clearer head, check again to see what is the cause of the problem.

* Were you angry after seeing a TTL or TE response?
* Were you not in agreement with the TE decision?
* Did you misunderstand an instruction but get irritated after rechecking?

You have to eliminate any negative thoughts that fuel your anger, only then will you be able to see the issue in a broader light and gain a better perspective on it. If you keep thinking “It should or must be like this” then the conversation will not go anywhere because you will likely get angry when the fact doesn’t line up with your expectations. Learn to reframe the way you think and try to ask yourself to determine if what you are about to say is true, valid, relevant, and fundamental.

If you are ready to move to the next step, be ready to accept any outcomes despite the efforts and **believe the efforts will not be wasted**. The next step is to try to understand the message or the decision. The first thing worth doing is to double-check the cycle overview thoroughly and carefully, as well as the test case and slot instructions, or any other information that has been provided, and then reevaluate them. By rechecking the instructions, you will gain a better understanding of the goal of the cycle and can view the issue from another's point of view.

After you have understood the message or made a decision, it might be easier for you to come to a conclusion and focus on your work again. You will also have a better understanding of the company’s preferences and be better prepared for future projects. You might also find it useful to understand the following:

* Each test cycle or project might have its own requirements and conditions.
* It’s necessary to redo the test if you mistakenly execute the instructions as it affects the results.
* It’s necessary to unclaim the slot if you mistakenly claim a wrong slot as it will provide invalid results.
* Following uTest Guidelines is a must and violating it may end up in a termination of the account.
* You need to treat everyone with respect and always be professional and polite when communicating.

Avoid discussing your concerns or disagreements toward the message or decision in the cycle chat room, or messages tab in the issue report, test cases, or reviews. Also, refrain from seeking others' agreement in the community or escalating any issues directly to TSM or higher without further understanding of the issue. These actions can unnecessarily complicate the situations without leading to a concrete resolution and you might even violate the guidelines without realizing it in the heated discussion.

The key to resolving the issue is good communication and good communication can only be achieved if you have your anger under control and can accept other people’s perspectives and discuss the issue in a professional and good manner.

**Possible Issues and Solutions**

After understanding the message or the decision, if you think they are not aligned with the instructions or any information provided, you may proceed to the next step, which is disputing your issue report if it’s related to the issue report, or escalate to the TE if it’s related to other problems, such as test cases or reviews being rejected or unclaimed without a clear reason or TTL misconduct, etc.

If the issue is related to your report, which means you disagree with the rejection reason, then you may dispute the report. Remember that you should check the rejection reason in your issue report, not only the rejection type. Check here to learn more about [Disputes](https://www.utest.com/academy/tracks/17/courses/bug-rejections).

If the issue is related to other problems, you may need to escalate that first to the TE of the cycle according to the escalation rules mentioned under the Team Contact Information section. Note that escalation would be better avoided unless it is something urgent or if it needs attention and help from a higher-up to resolve the issue.

Next, prepare what you would like to send. Gather any useful information that can support your claim and provide screenshots if needed. Check the testing website or app, any provided documentation, and other relevant information.

Once you gather your thoughts and evidence, start drafting the email. Note that you can find the TE email provided in the cycle overview. Don’t use other TE emails on other cycles as they are not responsible for all cycles.  
When writing the email, keep in mind the following:

* Mention the cycle ID in the email subject.
* Make sure to be polite and professional when writing the content to state your concerns.
* Explain your concern clearly and focus on the issue.
* Avoid prolonging the message without a clear point.
* Recheck your writing and ensure that it’s clear and easy to understand.
* Don’t CC or BCC the email to other TEs or TSMs.

Note that TSM should only be contacted if there is an issue with the payout, it's an emergency, or if you haven’t received a reply from TE after the expected reply time as mentioned in the escalation rules.

Be assured that the project team is committed to resolving the issue promptly. If you have any feedback on the situation, feel free to share your thoughts in the email. Your feedback on the situation is valuable to the team as it may help improve the processes.

**Possible Outcomes**

Outcome may come in three ways, accepted, rejected, or future consideration and improvement. Understand that not all points may be accepted and some feedback may need time to be applied. At this point, you are expected to be calm and rational, it’s no more “I am always right” or to interpret other perspectives as incorrect. If you can’t overcome it, then you should go back to the first and second points or better stop here as it may affect your mental health. If you believe you can accept any outcomes, other perspectives, or disagreement, you can continue reading this point.

First, about your dispute. If your dispute got rejected, make sure to read the rejection reason carefully. You are expected to get a more detailed reason for the rejection reason after a dispute. If the rejection reason is reasonable then you need to accept it and move on. We believe that accepting it will also make us a better tester and we can avoid reporting a similar issue for the customer if it should be avoided.

Next, about the email. If you receive a response from the TE, read the message carefully. If there is a need to do something on your part and you are willing to assist, help the TE so that it can be resolved. If they will resolve it on their end, you can wait if it will be resolved during the cycle duration or in the future if it’s considered.

In any case, ensure to keep your communication professional. If your temper surfaces, try to cool down again before replying to the message. Also, keep the communication within a thread and avoid making a new email thread to respond to the TE message.

**Note**: If your issue is an emergency and you haven’t received a reply after the expected reply time from TE or TSM, you may seek help from Tester Support. You can open [Tester Support](https://support.utest.com/csp) Portal > Select Request Support > Select Test Cycle Issues for the request category and then complete the form accordingly.

**Be Proud of Yourself**

If you can handle your anger appropriately while facing challenging issues, let us congratulate you. Anger is a difficult emotion to overcome, yet it is an important accomplishment in life. You are not only being professional but also shaping the community for the better.

We hope these tips help you to maintain a healthy habit so that you can naturally respond calmly and professionally in the future. Also, you can handle challenging situations effectively, achieve the most effective outcomes, and contribute to the improvement of future testing.

**Effective Written Communication**

As a tester, it’s important to communicate clearly and professionally with the project team and community. Effective communication ensures that your writing, whether it’s general messages, issue reports, or reviews, is easily understood by the reader. Mastering this skill can help you improve in clarity and avoid misunderstandings.

This course focuses on practical tips to enhance your written communication skills. Each tip includes an example to show how it can be applied to the issue reports, reviews, or messages in general. You can follow the below tips to improve your written communication:

**Concise and Clear**

Your work must be easily understandable by both the customer and the project team. Keeping your writing concise and clear would be best. This will help you to get your point across quickly and avoid confusion. Additionally, it will help to keep the reader's attention and make it easier to read and understand. Avoid using abbreviations that are not commonly used, avoid using vague words such as “Can’t, unable, or not working” as they don’t clearly explain the problem, and avoid adding irrelevant information only to prolong the sentence.

**Example 1:**

 Instead of writing: “After filtering the results, some products are not displayed even though the filter shows that there are 25 results but there are only 23 products on the page”. This can leave the impression that there are more results but they are not displayed on the page and the user will be confused about it.”

 You can say: “There are only 23 products after filtering the search results using the “Black” color filter option even though it shows 25 beside the filter option.”

**Example 2:**

 Instead of writing: “Can’t register an account” ; “Unable to checkout” ; “Search is not working”

 You can explain what is happening: “An error page opens after tapping on the sign-up button” ; “An infinite loading occurs after tapping on the checkout button” ; “The homepage is reloaded after clicking on the search suggestions”.

Note that this doesn’t always mean to shorten your message as much as possible. Instead, focus on explaining the issue clearly without adding unnecessary information and avoid redundancy. Additionally, avoid trying to convince the customer to accept your work or push them to fix the issue as soon as possible in your message.

**Structure and Organization**

When drafting your message, arrange it in a logical sequence so the reader can easily follow and understand it in the way you intend. A well-organized message ensures that your point is easy to follow and clearly understood. To write a structured message, start with an introduction, then provide supporting details, and finish with a conclusion or actionable suggestions.

**Example:**

 Instead of writing: “I noticed that the performance was not good at all as it took a lot of time to make an account. Moreover, the app crashed during the process which can frustrate the user.”

 You can say: “I experienced a slowdown during the registration process. For example, it took around 1 minute for the sign-up page to fully load. After filling in all the fields and then tapping on the sign-up button, an infinite loading was displayed. When I tapped the screen while the loading was still displayed, the app suddenly crashed. Fortunately, I was able to log in after reopening the app.”

By explaining what happened step by step sequentially, the reader can better understand your experience during the process especially if it was not good. A well-structured message helps the reader understand it the first time they read it, improving communication effectively.

**Professional Tone**

Using and maintaining the correct tone according to the topic, audience, and situation ensures your message is received well and respectfully. Once you know the topic and audience, draft the message in a way that effectively reaches the reader.

The professional tone is respectful, polite, and objective. This ensures the conversation remains constructive and within the topic without sounding overly emotional or aggressive even in tense situations. Keep in mind that using an incorrect tone when writing might cause misinterpretation and even be considered impolite without realizing it. Also, it would be better not to use offensive language under any circumstance to avoid unnecessary issues or violations.

Additionally, using a professional tone helps to create a positive atmosphere and encourages collaboration and understanding. It is important to remain professional in order to maintain trust and respect between all parties involved.

**Example 1:**

 Instead of writing: “I can’t open the test app at all. What do you expect me to test this way?”

 You can say: “I would like to inform you that the test app can’t be opened. I already tried opening it on other devices but nothing worked. It would be helpful if you could take a look at it and provide another app file. Thanks”

**Example 2:**

 Instead of writing: “The app performance seems like a joke. It didn’t let me open a section unless I waited for a minute”

 You can say: “During the testing, I noticed that the app’s performance was slower than expected as I had to wait around a minute to open a section. It would be helpful if the app’s performance becomes the main focus for improvement to ensure a smoother user experience”

You can see from the examples how the message is more positive and respectful when it is written in a professional tone, while also preventing negative impressions.

**Proofreading**

The final step is proofreading your writing to ensure that your message is clear and free of mistakes. Check if there are spelling, grammar, punctuation, and formatting mistakes. Also, check if your message is according to what you intend to write, you can read it and ask yourself if the reader will understand it. Proofreading might be time-consuming but it ensures that your message is clear and professional, showing that you appreciate the reader’s time. It can also help avoid miscommunication and improve credibility.

**Example 1:**

 Instead of writing: “I couldnt find the feature mention in the test case step1 and 3. I have restrasted and reinstall the app but still the same”

 You can adjust it to: “I couldn’t find the feature mentioned in the test case steps 1 and 3. I have restarted and reinstalled the app but I still couldn’t find it. Could you please confirm if the feature is available?”

**Example 2:**

 Instead of writing: “The app version attach in the cycle overview is wrong. I can’t perform test case steps this way.”

 You can adjust it to: “I would like to inform you that the app version attached in the cycle overview does not match the version mentioned in the test case step. Could you please check if the attached app is correct?”

By reviewing your writing, you can understand it better from the reader's perspective and adjust the message accordingly to match your intent. However, note that even a well-written message can sometimes be misinterpreted but at least we can try our best to appreciate the reader's time by writing the message as clearly as possible.

We hope the above tips will enhance your writing skills and inspire the community to strive for excellence. Additionally, it can help build trust with customers and the Applause team, increasing opportunities for future projects.

**The impact of AI on uTesters**

AI, freely available to everyone, tempts us to use it to make our work easier, especially when it comes to writing reviews, completing usability studies, giving feedback, or answering survey questions. **But is it actually making our work easier?** Or is it creating problems we don’t yet see? Before diving into the impacts AI can have on you as a uTester, let’s first understand the foundation of what makes a review true, meaningful, and valuable.

**True Review / Opinion / Feedback / Usability Study**

A true review is **honest**, **personal**, and **rooted in your genuine experience** with a product. It reflects what you, as the tester, saw, felt, and thought while navigating an app, website, or platform or even testing a hardware product. For the customer, these reviews are one of the reasons for using our services. The customers rely on us to provide **real, human insights** that go beyond surface-level observations to truly understand how their product feels and functions in the hands of an actual user.

**What impact will AI have on you as a Tester?**

When feedback is authentic, it drives real improvements. When it’s not, for example when it’s written by AI or lacks personal engagement, it fails its purpose entirely.

This failure can have serious consequences for you as a tester:

**Damaged Credibility**

When a TE, TTL, or customers notice that your feedback feels vague, AI-written, generic, robotic, or disconnected from the test environment, they lose trust in you. A negative reputation on uTest can lead to fewer invitations to projects.

**Reduced Opportunities**

High-quality, insightful feedback sets top testers apart. If AI-generated responses become part of your process, your value as a tester diminishes. *If the customer wanted AI’s feedback, why would they rely on you when they could use AI themselves?*

**Terms and Conditions**

All uTesters must comply with the [uTest Guidelines](https://www.utest.com/utest-guidelines" \t "_blank) and [Terms and Conditions](https://www.utest.com/terms-and-conditions). A violation of the uTest Guidelines and Terms and Conditions can result in the termination of your account. Make sure to review and understand the acceptable use policy for generative AI at uTest.

Now let’s explore why relying on AI to provide reviews or feedback isn’t just a bad practice but potentially career-damaging.

On uTest, we pride ourselves on delivering authentic, insightful feedback that helps customers create better apps, websites, and products. But lately, a concerning trend has emerged: users turning to AI to write eveything for them, such as reviews and answer open-ended survey questions.

**AI tools like ChatGPT can be useful in daily life but using them to replace your insights as a tester is a dangerous shortcut.**

**Why so?**

**It’s Not Personal:**

AI generates overly polished text and it can’t replicate your actual experience with a product. Customers want to know what you found confusing, delightful, or frustrating—not a generic response from a machine.

**It Can Mislead Customers:**

AI-generated feedback might oversee real issues or falsely praise functionality, it leads the customers in the wrong direction. This wastes their time and resources and reflects poorly on uTest as a platform.

**Why Honest Feedback Matters to Customers**

The end goal of every usability survey or review is to help customers create products that users love. When your feedback comes from your actual experience, it adds value. You are helping shape the user experience in a way that only a real human can.

AI can’t feel frustration when a button doesn’t work or a page keeps on loading. It can’t experience joy when a feature works smoother than we expect. And it certainly can’t explain the subtle ways an interface makes you feel overwhelmed or empowered. That’s why your honest feedback is irreplaceable.