

Roblox Spaces People Calendars Create ...

Design

Calendars

PAGE TREE

- > Product Designs
- > Style Guides & Resources
- > User Experience Research
 - > Reports & Insights
 - > Internal Resources
 - > Usability Testing Program
 - Studio User Research
 - External Resources
- > Team Documentation
- > Page Archive

Pages / Design / User Experience Research

Studio User Research

Created by Grace Lin, last modified on Aug 07, 2018

First Time User Experience

- Method: Usertesting.com
- Participants: 5 new users
- Date: Jan, 2017
- Result (Overview):
 - Studio download install process is frustrating for users. Hard to find where/how to download Studio, and confusing results after installing Studio: Roblox player and Studio icon show up at the same time, users don't know what's going on.
 - In general, current Studio UI is less-organized and inefficient. The richness of contents could be overwhelming to first-time users. Structure and interaction are not fully predictable, and the visibility of the system status is unclear. Specifically, We categorized our finding into six groups: structure, behavior, style, 3D view, start, specific functions, and the miscellaneous.
- Report:
 - [Usertesting Videos](#) (username password required)
 - [Raw Data - Video Notes](#)
 - [Findings & Analysis Report](#)
- Action Item:
 - [Redesigned Create Landing Page](#)
 - [Fixed Studio Download Install Flow](#)

User Interview - Incubators

- Method: Interview + Observation (60 min)
- Participants: 8 Incubators
- Date: September - October, 2017
- Result (Overview):
 - To our top developers, Studio is a powerful, intuitive, and configurable tool, but it's also buggy and inefficient. However, we should keep in mind the fact that most of them have used Studio for more than 7 years generates bias in the evaluation. Another key finding is that even though they are experienced developers now, they started using Studio as kids with barely prior coding experience. That's why Studio, as a simplified game engine plus modeling tool at that time, could beat other more professional softwares and keep its first batch users till now. Therefore, the next step of Studio user research should focus on the attitude and behavior of nowadays kids with no coding experience, our real potential target users, instead of experienced developers who are already familiar with other game engines or 3D modeling software.
- Report:
 - [User Interview Report](#)
- Appendix:
 - [Raw Data - Interview Questions & Notes](#)
 - [Affinity Wall Bullet Points](#)
 - [Protocols](#)

User Survey

- Method: Survey Gizmo

Space tools

Define

- Problem Statement
- **User Research**
 - Interview
 - Survey
 - Contextual Inquiry
 - Persona

Analyze

- Heuristic Evaluation
- Comparative Analysis

Ideate

- Approaches
- User Flows

Prototype

- Wireframes
- Hi-fi Prototypes
- Interactive Prototypes

User Test

Design QA

<https://confluence.roblox.com/display/DESIGN/Studio+User+Research>

	ROBLOX Studio	Unity	Sketch	Figma	Balsamic
		Game Engine	Interface Design	Interface Design	Wireframing
Feature Name	Packages	Prefabs	Symbols (relevant: Libraries)	Components	Symbols
Have a _Feature_ Master	No	No	Yes	Yes	Yes
Master Name	-	-	-	Component Master	Symbol Source
Master Editable	-	-	Yes	Yes	Yes
Auto Update (Instances)	No	No	Yes	Yes	Yes
Separate Place/Mode for Master	No (for V1) Yes (for V-2019)	Yes	Yes	No	Yes
Status	1) Non-Package 2) Package default 3) Package w/ change 4) Pacakge w/ available update 5) Package w/ change + update	1) Non-Prefab 2) Prefab default 3) Prefab w/ Change	1) Non-Symbol 2) In-file Symbol 3) External Symbol 4) Symbol w/ update (only for external ones)	1) Non-Component 2) Component Master 3) Instance default 4) Instance w/ change	1) Non-Symbol 2) Symbol default 3) Symbol w/ change
Status Indication [where] how -> for what		[Hierarchy Tree] Item Name Color -> Non-Prefab/ Prefab [Properties Panel] Propertie Name Font-weight -> Changed or not	[Hierarchy Tree] Item Icon -> non-Symbol/ in-file Symbol/ external Symbol [In General] Notification (only for external symbols) -> Updated or not	[Hierarchy Tree] Item Icon + Hierarchy -> non-Component/ Master/ Instance [Properties Panel] Button Status [Toolbar] Button Status -> Updated or not	[Workspace] Selection Box Color -> Non-Symbol/ Symbol [Properties Panel] Section Name Color + Button Status -> Changed or not

Define

- Problem Statement
- User Research
 - Interview
 - Survey
 - Contextual Inquiry
 - Persona

Analyze

- Heuristic Evaluation
- Comparative Analysis

Ideate

- Approaches
- User Flows

Prototype

- Wireframes
- Hi-fi Prototypes
- Interactive Prototypes

User Test

Design QA

Research - Package* Status Indication

* Different products name Package-equivalent feature differently. * "_Feature_" below refers to the general concept.

Key Takeaways

- Multiple indicators are used to indicate different status. (not only icons, and not all products are using icons as indication actually)
- Icons are generally used to indicate the binary status: whether an item IS or IS NOT _Feature_, (or to indicate _Feature_ type at the same time at most)
- Icons are barely used to indicate the properties status (whether it's been changed or has updates available)
- As indicators, completely different icons are used. (not adding auxiliary part)

1. Unity

Feature name:

Prefab

Have a _Feature_ Master:

No

Auto Update:

No

(Other instances):

Status:

1) Non-Prefab

2) Prefab default

3) Prefab w/ change

Status Indication: Name Color + Properties Highlight

[Hierarchy Tree] Item Name Color -> Non-Prefab/ Prefab

Non-Prefab

Prefab

Y Blender (all)

Camera

Cube

Sphere

Lamp

Y Blender (all)

Camera

Cube

Sphere

Lamp