



Jio Finance

Bank anytime anywhere

2025

Pushkar Pandey

Explore now

Design Process

●

Beta Version

Beta Version is available for internal users

●

Foundation Research

A comprehensive set of best practices and actionable guidelines in DS for designing user-friendly and risk-aware destructive action (DA) interfaces.

●

Feature Evaluation

Evaluating what features needed and the use.

●

User Experience Audit

Evaluating user experience while using the app.

●

In - App Journey Audit

Find the In-App broken journeys

●

Experience Principle Review

Evaluating JFS on Jio's 10 experience principles.

●

UI Review

UI Audit according to Jio Design System STD

●

Data Viz Review

Data Viz Audit according to Jio Design System STD

●

Version 1

Version 1 launched in app store

Foundation Research

The study aims to achieve 5 primary objectives:

- Identify fear points in user decision-making during important actions**
Understand where users hesitate, abandon, or feel anxious in the flow.
- Analyze the psychological triggers of fear**
Determine what factors (e.g., irreversibility, financial risk, privacy concerns) amplify user hesitation while they perform destructive actions.

- 3 **Evaluate existing mitigation strategies**
Assess how users perceive current confirmation dialogs, error messages, and security indicators.

- 4 **Measure user trust and confidence levels**
Identify how design choices (e.g., wording, color, visual cues) impact perceived risk and trust.

- 5 **Develop UX principles to reduce fear**
Explore best practices to enhance user confidence in high-stakes decisions.

Outcome

A comprehensive set of best practices and actionable guidelines in DS for designing user-friendly and risk-aware destructive action (DA) interfaces.

Secondary research and Design	1:1 In-Depth user Interview	Analysis
--------------------------------------	------------------------------------	-----------------

- 14 user interviews
- Segments : 10 external and internal participants

Types of decisions

There are 2 layers of decisions which users can encounter :

- Severity of decision (High, Medium, Low)

- Affirmative destructive actions: Destructive actions initiated by the user that takes the users further in their journey (for example, Save or Delete)

- Accidental destructive actions: Destructive action aren't always affirmative. In certain cases, destructive actions might be initiated accidentally by the user or by the system.
- ### Fears which exist

Triggered while users accidentally navigate away from a page without saving their data.

Users are uncertain about the impact, outcome, or reversibility of their decisions.

Irreversible outcomes

- Actions which lead to permanent and irreversible consequences that cannot be undone which leads to panic

Secondary research

[illegible]

High severity

- A destructive action is critical to users or data and/or irreversibly, strongly encourages implementing a double check confirmation pattern and/or an undo button.
- A double check confirming a destructive action should not be the single trigger to perform the action.
- Always keep confirmation of the destructive action away from the destructive button, separate, either by space, user reading or color.
- A message request confirmation should be placed before the button, unless the danger of clicking is not the correct for the user.
- A key control on an interactive element is the consequence of the destructive action.
- A key trigger on a button is the correct choice to make.

1. Never ask for a warning of high severity consequence. [Click here](#)

Medium severity

Destructive actions may be harmful to a user and/or potentially irreversible, encouraging them these actions is difficult and may have potential harm. Consider encouraging additional key controls and/or confirmation patterns to ensure destructive actions are taken for example, put the entire action confirmation button a distance of mouse clicks away.

Low severity

The destructive action will not be critical to users and/or not irreversible. Consider placing the button on the side to prevent the button, especially if there are no action back to be performed back. These level of destructive actions are the result of using the design pattern of primary and secondary button combination.

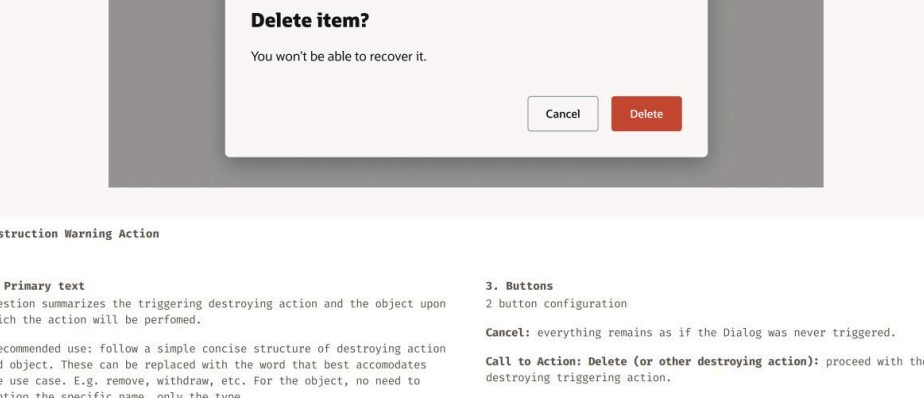
References

- [5 Usability Principles for User Interface Design](#), Nielsen Norman Group

Last updated on Wednesday, October 16, 2024 12:17 AM

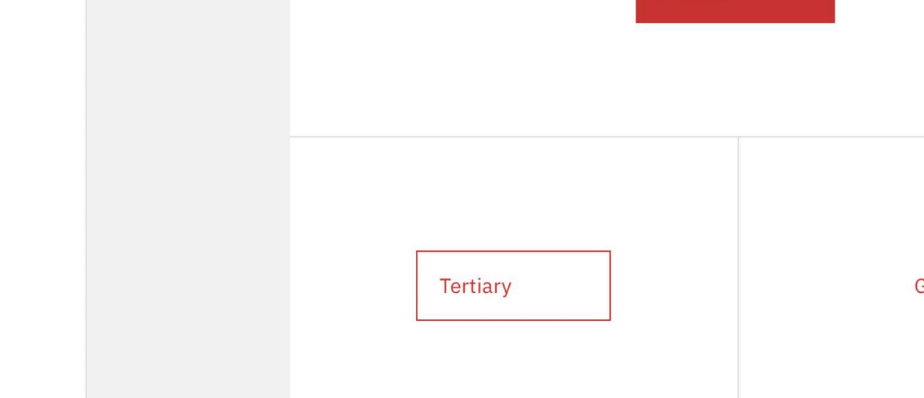
Oracle DS

The Oracle Redwood Design System outlines best practices for destructive action dialog, emphasizing clear messaging hierarchy (primary text, secondary text, and action buttons) and intent-based button labeling to reduce user hesitation and errors. It recommends using strong visual cues (e.g., color, spacing) and requiring explicit confirmation for irreversible actions.



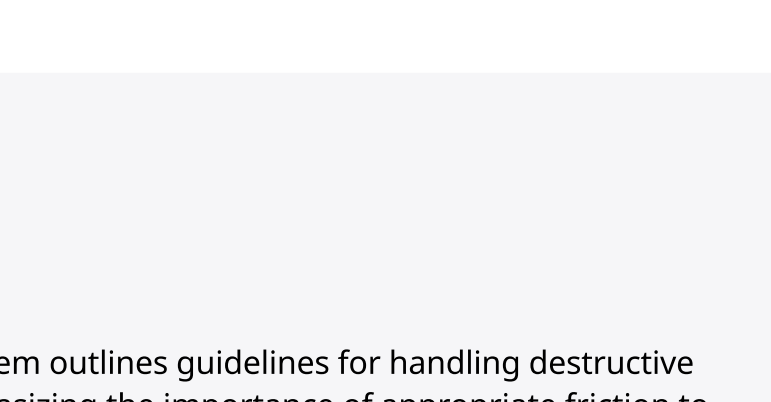
Carbon DS

The Carbon Design System advises that destructive actions should be clearly labeled, visually distinct (e.g., red buttons), and require confirmation for irreversible changes. It recommends using modals or alerts for critical actions, ensuring users understand the consequences before proceeding.



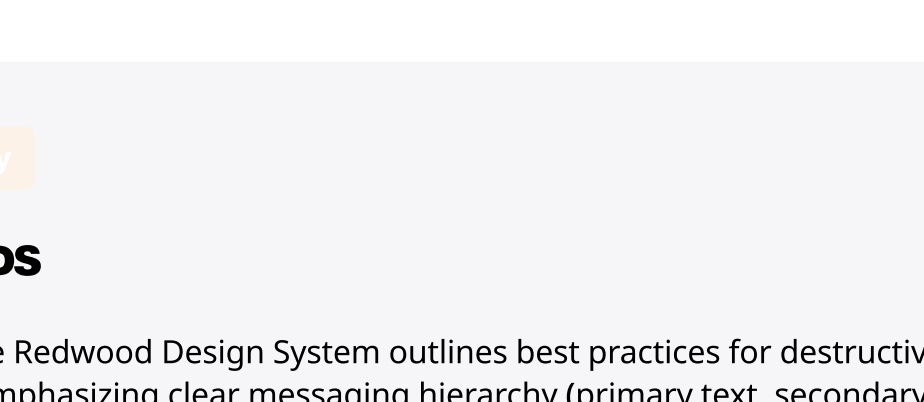
GitLab DS

The GitLab Pajamas Design System outlines guidelines for handling destructive actions in user interfaces, emphasizing the importance of appropriate friction to prevent accidental actions and help users understand consequences.



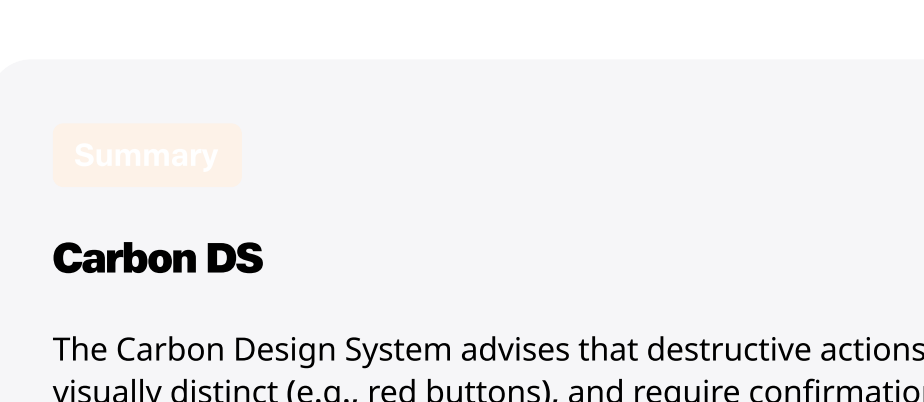
Oracle DS

The Oracle Redwood Design System outlines best practices for destructive action dialog, emphasizing clear messaging hierarchy (primary text, secondary text, and action buttons) and intent-based button labeling to reduce user hesitation and errors. It recommends using strong visual cues (e.g., color, spacing) and requiring explicit confirmation for irreversible actions.



Carbon DS

The Carbon Design System advises that destructive actions should be clearly labeled, visually distinct (e.g., red buttons), and require confirmation for irreversible changes. It recommends using modals or alerts for critical actions, ensuring users understand the consequences before proceeding.



Linking Secondary Research to Primary

Key Takeaways from Secondary research

- Types of Destructive Actions:
- Defined affirmative vs. accidental actions

Insights into severity levels

Insights from GitLab, Oracle, and Carbon DS emphasise severity-based handling—this framework helps apply those

Insights into severity levels

Insights from GitLab, Oracle, and Carbon DS emphasise severity-based handling—this framework helps apply those

GitLab, Oracle, and Carbon DS emphasize clear hierarchy, warnings, and confirmations, leading to a structured low, medium, high severity approach.	Severity helps differentiate these cases. Clear severity levels prevent overuse of warnings, ensuring users only pause when truly necessary.
---	--

- High severity : JioFinance, Low severity : Deleting address and Medium severity : E-commerce order cancellation

- | | |
|--|---|
| <p>Key Takeaways from Secondary research</p> <ul style="list-style-type: none"> Types of Destructive Actions:
Defined affirmative vs. accidental actions. User Fears & Pain Points:
Identified major concerns: accidental triggers, ambiguous actions, irreversible outcomes. | <p>Division into severity levels</p> <p>Insights from GitLab, Oracle, and Carbon DS emphasise severity-based handling—this framework helps apply those best practices systematically. So the screens were tested on basis of hierarchy</p> <ul style="list-style-type: none"> Assess User Impact: |
|--|---|

Clear severity levels prevent overuse of warnings, ensuring users only pause when truly necessary.

- High severity : Jiofinance, Low severity : Deleting address and Medium severity : E-commerce order cancellation