Team Project Assignment Submission 2 – Open Table App

Shoaibuddin Ahmed Mohammed Abdul Mannan Mohammed Sanjana Balannagari Amulya Appani

CPSC – Internet Programming – 8720 – 03 – 25SP

Research Assignment

1. PRG (Post/Redirect/Get): Avoid Missing Submissions

As an illustration, picturing us completing an online flight reservation form and selecting the "Book Now" option. Duplicate reservations might result from the browser attempting to submit the same request again if we refresh the page after submitting the form.

How PRG Operates:

The browser is sent to a different page (such as a confirmation page) by the server once the form has been submitted.

This redirection stops multiple submissions by preventing the original form data from being sent again when the page is refreshed.

Advantages: Easy to use and popular Prevents unintentional multiple submissions by using the back or refresh buttons.

Cons: Doesn't stop deliberate duplicate submissions slows down the operation by adding an extra step.

2. RRG: Avoiding Intentional and Accidental Submissions

As an illustration, suppose we are making an online purchase. The system shouldn't process multiple payments if we inadvertently click "Pay" more than once or reload the page.

How RRG Operates:

The system creates a special request token and links it to the transaction before processing the payment.

The system determines if the token has already been used if the same request is submitted again (for example, by refreshing or clicking the button again). The request is disregarded if it was.

Advantages: It prevents both inadvertent and deliberate duplicate submissions, making it more secure than PRG. It is dependable for significant transactions and performs well with sessions.

Cons: More difficult to deploy than PRG. Tracking tokens are needed for every request.

3. Keeping Duplicate Transactions Out of APIs with Idempotency Key

As an illustration, consider transferring money using a banking app. We don't want the bank to transfer twice as much money if we inadvertently press the "Transfer" button twice.

The Operation of Idempotency Key:

For every transaction, a distinct key is generated by the client, which is the user's browser or application.

This key is stored by the server, which also verifies if it has been used previously.

To guarantee that the transaction only occurs once, the server rejects requests containing the same key.

Advantages: Ensures that only one request is handled, even if submitted several times Works well for online payments, APIs, and other crucial operations.

Cons: Key management adds additional processing cost and necessitates a database or cache for tracking and storing keys.

Name	PRG (Post/Redirect/Get)	RRG (Redirect with Request Guard)	Idempotency Key
Description	Redirects after a POST request to prevent resubmission.	Uses a unique request token to ensure a request is processed only once.	Generates a unique key for a transaction and ignores repeated requests with the same key.
Example	Prevents double submission when booking a flight.	Prevents duplicate payments when clicking "Pay" multiple times.	Ensures a money transfer is processed only once, even if requested multiple times.

Pros	Simple and widely used; prevents	More secure than PRG as	Ideal for APIs and
	accidental duplicate submissions	it prevents intentional	financial transactions;
	due to refresh or back button.	resubmissions; works well	ensures only one
		with sessions.	execution of a request.
Cons	Does not prevent intentional	Adds implementation	Requires database/cache
	resubmissions; requires extra	complexity; requires	storage; additional
	redirects.	tracking tokens.	overhead in managing
			keys.

Screenshots

































