

useUpdateEntity

Overview

The `useUpdateEntity` hook is used to update the details of an existing entity. It allows for partial updates, meaning only the provided fields will be modified while other fields remain unchanged.

Usage Example

```
import { useUpdateEntity } from "@replyke/react-js";

function UpdateEntityForm({ entityId }: { entityId: string }) {
  const updateEntity = useUpdateEntity();

  const handleUpdateEntity = async () => {
    try {
      const updatedEntity = await updateEntity({
        entityId,
        update: {
          title: "Updated Title",
          content: "Updated content of the entity.",
          keywords: ["updated", "entity"],
          location: { latitude: 40.7128, longitude: -74.0060 },
          metadata: { category: "updated category" },
        },
      });
      console.log("Entity updated successfully:", updatedEntity);
    } catch (error) {
      console.error("Failed to update entity:", error.message);
    }
  };

  return <button onClick={handleUpdateEntity}>Update Entity</button>;
}
```

Parameters & Returns

Parameters

The hook returns a function that accepts an object with the following fields:

Parameter	Type	Required	Description
entityId	string	Yes	The ID of the entity to be updated.
update	object	Yes	An object containing the fields to update.
update.title	string null	No	The new title for the entity.
update.content	string null	No	The new content for the entity.
update.attachments	Record<string, any>[]	No	The updated attachments items for the entity.
update.keywords	string[]	No	The updated keywords for the entity.
update.location	object	No	The new geographic location for the entity.
update.metadata	Record<string, any>	No	Additional metadata to update for the entity.
update.mentions	Mention[]>	No	Array of mentioned users

Returns

The function resolves with an object representing the updated entity:

Return Value	Type	Description
Entity	object	The details of the updated entity.

© Replyke 2025