Database Schema Definitions

The application uses MongoDB (NoSQL) with the Mongoose ODM. The database consists of two primary collections: **Users** (for authentication) and **Links** (for storing the encrypted payloads and file metadata).

1. users Collection

Stores registered user accounts.

Field	Type	Constraints	Description
_id	ObjectId	Primary Key, Auto-generated	Unique identifier for the user document.
username	String	Required, Unique	The user's chosen login name.
password	String	Required	The user's password, securely hashed using bcrypt before saving.
createdAt	Date	Default: Date.now	Timestamp of when the account was created.

2. links Collection

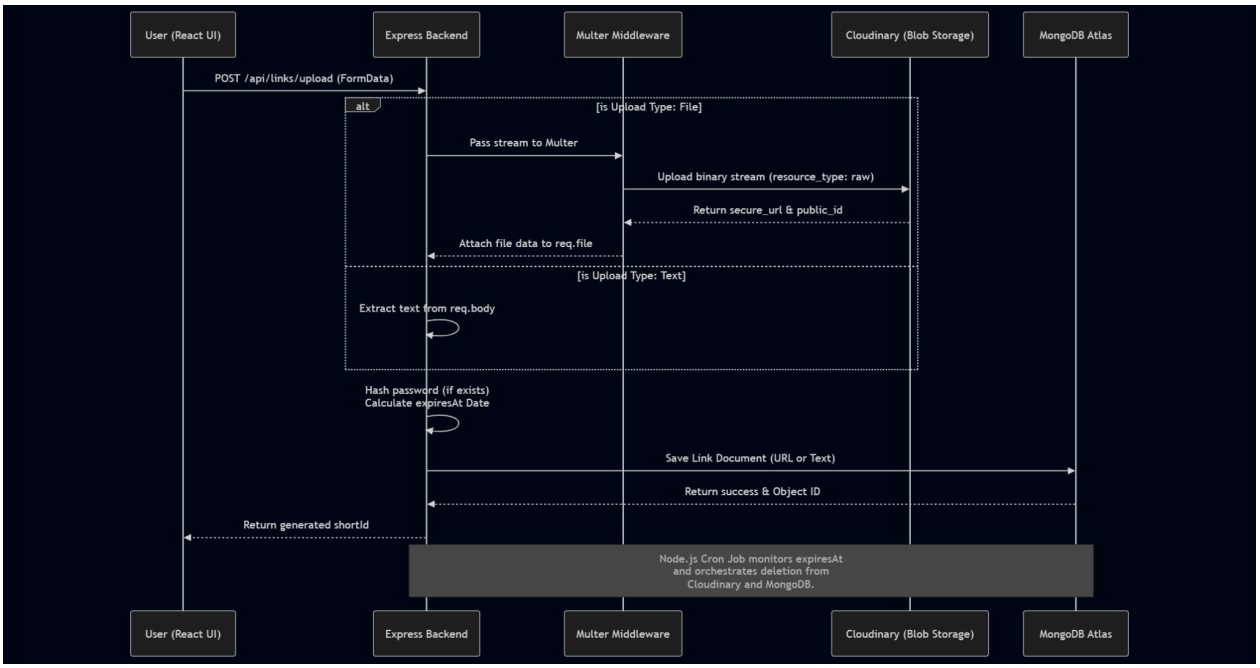
The core collection storing both text snippets and file metadata

Field	Type	Constraints	Description
_id	ObjectId	Primary Key, Auto-generated	Unique identifier for the link document.
shortId	String	Required, Unique	The generated unique short string used in the shareable URL.
type	String	Required, Enum: ['text', 'file']	Identifies if the payload is raw text or a binary file.
content	String	Required	Stores the actual text snippet OR the secure Cloudinary file URL.
originalName	String	Optional	Preserves the original file name (e.g., document.pdf) for the download UI.
createdAt	Date	Default: Date.now	Timestamp of when the link was generated.
expiresAt	Date	Required, Indexed	The exact future time the link expires. **Has a TTL index** (expireAfterSeconds: 0) for automatic background deletion by MongoDB.
password	String	Optional	Bcrypt-hashed password if the user chose to lock the link.

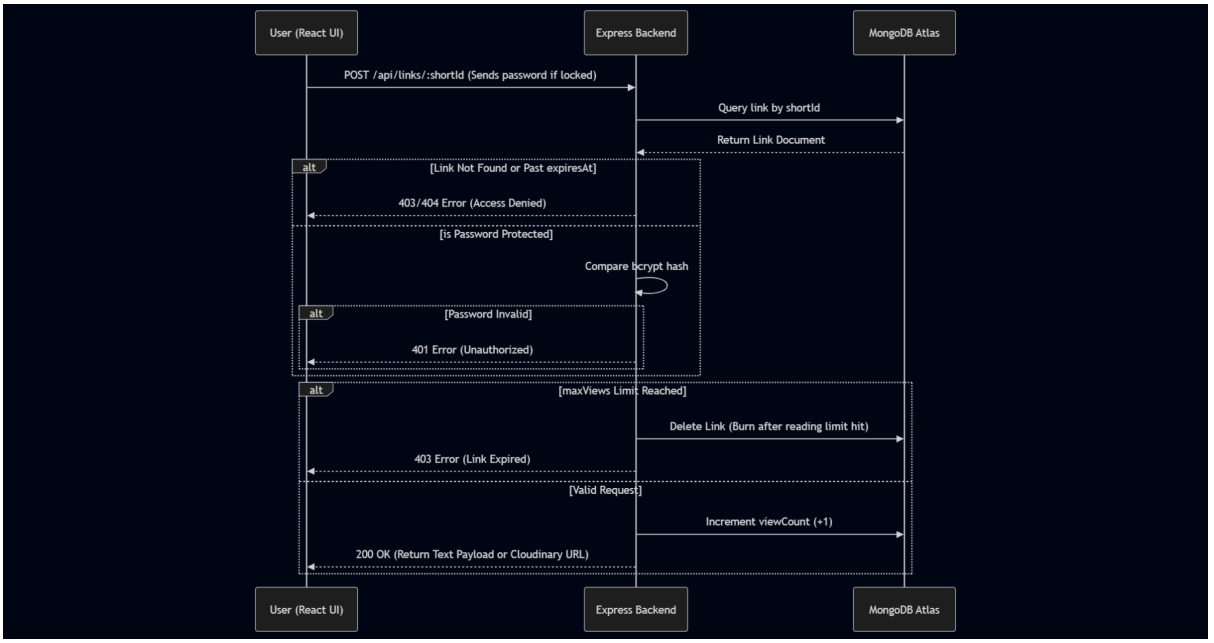
viewCount	Number	Default: 0	Tracks how many times the payload was successfully viewed/downloaded.
maxViews	Number	Optional	The maximum allowed views before the link is considered dead.
owner	ObjectId	Optional, Ref: User	References the users collection if an authenticated user created it. Null for anonymous guests.

Data flow diagram :

1. Pipeline



2. Data Retrieval & Security Flow



3. Background Cleanup Flow (Cron Job)