



2. Notes App

- Users
 - Notes (title, content, pinned)
- Practice: Basic data modeling, search filters

Models

1. users

Field Name	Type	Required	Description
<code>_id</code>	Objectid	Yes	Auto-generated unique identifier
<code>name</code>	String	Yes	Full name of the user
<code>email</code>	String	Yes	Unique email address of the user
<code>password</code>	String	Yes	Hashed password
<code>createdAt</code>	Date	Yes	Timestamp of user creation (auto)
<code>updatedAt</code>	Date	Yes	Timestamp of last update (auto)

User API Route List

Method	Endpoint	Auth Required	Description
POST	/api/users/register	✗ No	Register a new user
POST	/api/users/login	✗ No	Log in a user and return JWT token
DELETE	/api/users/delete	✓ Yes	Delete the authenticated user account
GET	/api/users/logout	✗ (Client-side)	Logout handled on frontend (clear token)
PUT	/api/users/change-password	✓ Yes	Change the password of the authenticated user
PUT	/api/users/change-username	✓ Yes	Change the username of the authenticated user

✓ 2. Profile Model (extended info)



Profile APIs

Method	Endpoint	Auth Required	Description
POST	/api/profiles	✓ Yes	Create a profile for the authenticated user (if not already exists)
GET	/api/profiles/me	✓ Yes	Get the profile of the authenticated user
GET	/api/profiles/:id	✗ No	Get a profile by Profile ID (public view)
PUT	/api/profiles/update	✓ Yes	Update authenticated user's profile details (bio, profilePic, dob, location)
PATCH	/api/profiles/profile-pic	✓ Yes	Update only the profile picture
DELETE	/api/profiles/delete	✓ Yes	Delete the authenticated user's profile

Field	Type	Required	Description
<code>_id</code>	ObjectId	Yes	Profile ID
<code>user</code>	ObjectId	Yes	Ref to <code>User</code>
<code>profilePic</code>	String	No	URL/path to profile picture
<code>bio</code>	String	No	Short biography
<code>dob</code>	Date	No	Date of birth
<code>location</code>	String	No	(optional) Location info
<code>createdAt</code>	Date	Yes	Auto-generated
<code>updatedAt</code>	Date	Yes	Auto-generated

3. Notes

✓ Minimal Note Model – Fields (Tabular Format)

Notes → user (relations)



Note APIs

Method	Endpoint	Auth Required	Description
<code>POST</code>	<code>/api/notes</code>	✓ Yes	Create a new note
<code>GET</code>	<code>/api/notes</code>	✓ Yes	Get all notes for the authenticated user
<code>GET</code>	<code>/api/notes/:id</code>	✓ Yes	Get a specific note by its ID (if owned by user)
<code>PUT</code>	<code>/api/notes/:id</code>	✓ Yes	Update a note (title, content, tags, pinned, archived)
<code>DELETE</code>	<code>/api/notes/:id</code>	✓ Yes	Delete a note by its ID
<code>PATCH</code>	<code>/api/notes/:id/pin</code>	✓ Yes	Toggle pinned state for a note
<code>PATCH</code>	<code>/api/notes/:id/archive</code>	✓ Yes	Toggle archived state for a note

Field Name	Type	Required	Description
<code>_id</code>	ObjectId	Yes	Auto-generated unique ID for each note
<code>title</code>	String	Yes	Title of the note
<code>content</code>	String	Yes	Main text content of the note
<code>user</code>	ObjectId	Yes	Reference to the <code>User</code> model (owner of the note)
<code>tags</code>	String[]	No	Array of tags/labels for organization
<code>isPinned</code>	Boolean	No	Whether the note is pinned
<code>isArchived</code>	Boolean	No	Whether the note is archived
<code>createdAt</code>	Date	Yes	Auto-generated timestamp on creation
<code>updatedAt</code>	Date	Yes	Auto-updated timestamp on modification