

CHALLENGES — API

- Restrict users so they can only review a tour they have actually booked
 - Implement nested booking routes:
 - /tours/:id/bookings
 - /users/:id/bookings
 - Improve tour dates:
 - Add participants and soldOut fields to each date
 - A date = instance of the tour
 - When a user books → they must select a date
 - New booking → increments participants
 - If participants \geq maxGroupSize → mark as soldOut
 - On booking → check if selected date is still available
 - Implement advanced authentication features:
 - Confirm user email
 - Keep users logged in with refresh tokens
 - Two-factor authentication (2FA)
-

CHALLENGES — Website (Frontend)

- Implement a **sign-up form** (similar to login form)
- On **tour detail page**:
 - If a user has taken a tour → allow them to add a review (form)
- Hide booking section if user already booked the tour
 - Also prevent duplicate bookings on the model
- Implement “**like tour**” functionality → with Favorites page
- On **user account page**:
 - Add a “**My Reviews**” page (list all reviews, allow editing)
 - *(If you know React, this would be a great way to consume the Natours API)*
- For **administrators**:
 - Implement “Manage” pages to CRUD:
 - Tours
 - Users
 - Reviews
 - Bookings