How to send mail to tradesperson when they got a new booking :

1. **Backend Requirements (Node.js & Express)**

* **Nodemailer (for email)** or **Twilio (for SMS)**
* **New API route (/api/auth/notify-tradesperson)** to send the notification
* **Fetch tradesperson email/phone** from the database
* **Format a friendly notification message**

**2️) What You Need in Backend**

✔️ **Install Nodemailer (for emails):**

npm install nodemailer

✔️ **Install Twilio (for SMS - optional):**

npm install twilio

✔️ **Create a New API Route (notifyTradesperson)**

* Fetch **tradesperson’s email/phone** from DB
* Send **a friendly message** like:  
  **"Hey {name}, you got a new booking! 🎉 Here are the details..."**
* Return success/error response

**3️)Frontend Changes in TradespersonProfile.js**

* **After booking submission, call notifyTradesperson API**
* Send **tradesperson ID, email/phone, and booking details**

**4️)Backend API Logic (authController.js):**

const nodemailer = require("nodemailer");

exports.notifyTradesperson = async (req, res) => {

try {

const { tradespersonId, date, time, service, comments } = req.body;

// Fetch tradesperson details from DB

const tradesperson = await Tradesperson.findById(tradespersonId);

if (!tradesperson) {

return res.status(404).json({ message: "Tradesperson not found" });

}

// Email setup

let transporter = nodemailer.createTransport({

service: "Gmail",

auth: {

user: process.env.EMAIL\_USER, // Your email

pass: process.env.EMAIL\_PASS, // Your email password

},

});

const mailOptions = {

from: process.env.EMAIL\_USER,

to: tradesperson.email, // Tradesperson's email

subject: "🎉 New Booking Alert!",

text: `Hey ${tradesperson.name}, you got a new booking! 🚀

📅 Date: ${date}

⏰ Time: ${time}

🛠 Service: ${service}

📝 Comments: ${comments}

Log in to your TradeTrek account to manage it.`,

};

await transporter.sendMail(mailOptions);

res.status(200).json({ message: "Notification sent to tradesperson!" });

} catch (error) {

console.error("Error sending notification:", error);

res.status(500).json({ message: "Failed to send notification" });

}

};

**Call This API from Frontend (TradespersonProfile.js):**

const handleBooking = async (e) => {

e.preventDefault();

const bookingData = {

userId: userid,

tradespersonId: t.id,

date: formData.date,

time: formData.time,

service: formData.service,

comments: formData.comments,

};

try {

const response = await fetch(`${process.env.REACT\_APP\_API\_URL}/api/auth/bookings`, {

method: "POST",

headers: {

"Content-Type": "application/json",

"Authorization": `Bearer ${localStorage.getItem("token")}`,

},

body: JSON.stringify(bookingData),

});

if (response.ok) {

alert("Booking successful!");

// Notify tradesperson

await fetch(`${process.env.REACT\_APP\_API\_URL}/api/auth/notify-tradesperson`, {

method: "POST",

headers: {

"Content-Type": "application/json",

},

body: JSON.stringify(bookingData),

});

Navigate("/my-bookings");

} else {

const data = await response.json();

alert(data.message);

}

} catch (error) {

console.error("Error booking:", error);

alert("Something went wrong. Please try again.");

}

};