



SMS Sending System with Web Interface

Luigi Matteo Girke

Agenda

01

User Journey

Everything a user can do

02

Tech Deep-Dive

How it works under the hood

03

Trials & Successes

Turning problems into solutions



01

User Journey

Everything a user can do

Login

- AD integration for secure login
- No need for separate accounts
- Permission based on AD groups



The login form is a white card with rounded corners. At the top left is the etpzp logo, which consists of a blue stylized 'e' and 'p' with 'zp' in a darker blue, and a speech bubble above it containing the text 'SMS'. Below the logo, the title 'Iniciar sessão' is displayed in bold black text, followed by the subtitle 'para começar a enviar mensagens SMS' in a smaller, regular black font. The form contains two input fields: 'E-mail' with the value '2812@etpzp.pt' and 'Palavra-passe' with masked characters '.....'. A black button with the text 'Iniciar sessão' is positioned at the bottom of the form.



Iniciar sessão

para começar a enviar mensagens SMS

E-mail

2812@etpzp.pt

Palavra-passe

..... 

Iniciar sessão

Compose & Send

- Dedicated new message page
- Fields: Sender (fixed), Recipients, Subject, Content
- Send immediately or schedule
- Manual + automatic draft saving

The screenshot displays the 'ETPZP-SMS' web interface. At the top, the header includes the 'ETPZP-SMS' logo, the user name 'Luigi Girke' with the role 'Administrador', and a profile icon. The main content area is titled 'Nova mensagem' (New message) in a red button. Below this, there is a sidebar menu with options: 'Enviado', 'Agendado', 'Falhado', 'Rascunhos' (1), 'Lixo', 'Configurações', 'Contactos' (3), and 'Painel de controlo do administrador'. The main message composition area shows the subject 'Olá Mundo!' and a recipient field with 'Pessoa' selected and 'Número de telefone' entered. The message body contains the text 'Olá Mundo!' and a placeholder 'Comece a escrever a sua mensagem'. At the bottom, there are buttons for 'Terminar sessão', 'Descartar', and 'Enviar'.

ETPZP-SMS

Luigi Girke
Administrador

Nova mensagem

Olá Mundo!

ETPZP

Para Pessoa X Número de telefone

Olá Mundo!

Comece a escrever a sua mensagem

Terminar sessão Descartar Enviar

Smart Recipient Input

- Add multiple recipients with keyboard or mouse
- Real-time suggestions from contacts & past recipients
- Error handling for invalid or duplicate numbers

The screenshot displays the ETPZP-SMS web interface. At the top, the header includes the ETPZP logo, the text "ETPZP-SMS", and the user profile "Luigi Girke Administrador" with a blue circular avatar containing "LG". Below the header, a red button labeled "Nova mensagem" is on the left. The main area is titled "Olá Mundo!" and shows a message composition form. The form includes a "Para" field with a dropdown menu currently showing "Pessoa". To the right of the dropdown is a red input field containing the phone number "+351111111111". Below the "Para" field, the text "Olá Mundo!" is visible. To the right of the main form, a dropdown menu titled "Frequentemente utilizado" is open, showing two suggestions: a contact with the name "Luigi" and phone number "+4917633446689", and another contact with the phone number "+351911521967". At the bottom of the interface, there is a "Terminar sessão" button on the left and "Descartar" and "Enviar" buttons on the right.

Scheduling Messages

- Choose future date & time
- Preview message status before send
- Cancel anytime before scheduled time

The screenshot shows a mobile application interface for scheduling an SMS. At the top, there's a header "P-SMS" and a greeting "Olá Mundo!". Below this, a modal dialog titled "Selecionar data e hora" (Select date and time) is displayed. The dialog contains a calendar for April 2025, with the 29th selected. To the right of the calendar, there are two dropdown menus for "Hora" (Hour) and "Minuto" (Minute), currently set to 18 and 36 respectively. At the bottom of the dialog, there are two buttons: "Reiniciar" (Reset) and "Selecionar data" (Select date). The background of the app shows a message preview with the text "mensagem" and "Olá Mundo!".

Selecionar data e hora
Escolha quando enviar a sua mensagem.

< April 2025 >

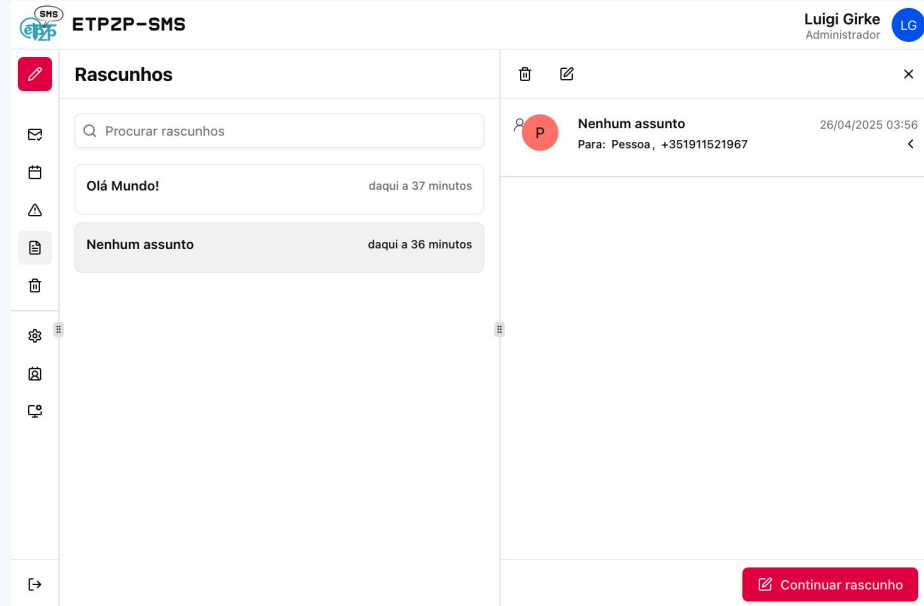
Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

Hora: 18
Minuto: 36

Reiniciar
Selecionar data

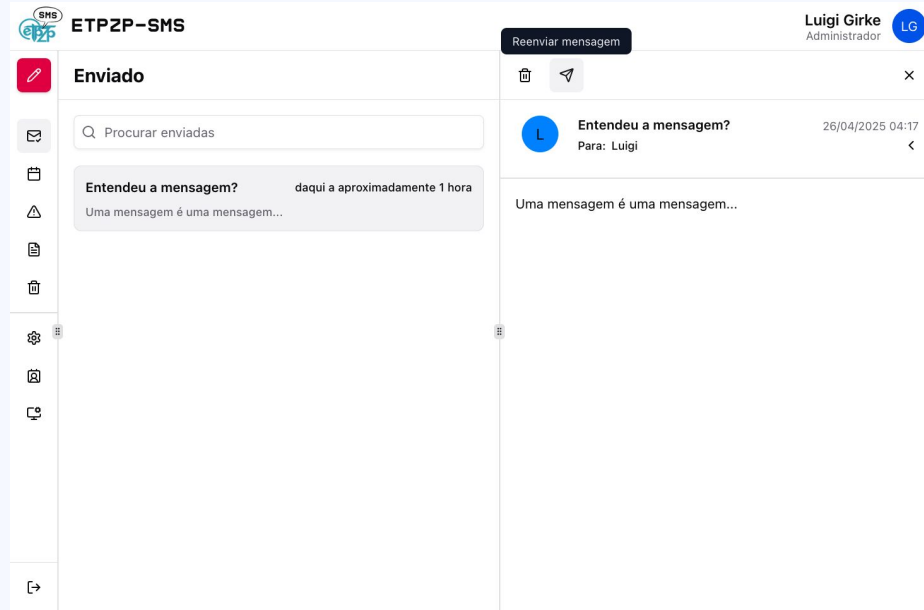
Message Management

- View: Sent, Drafts, Scheduled, Failed, Trash
- Each message shows delivery status, and recipients
- Ability to filter by name and content



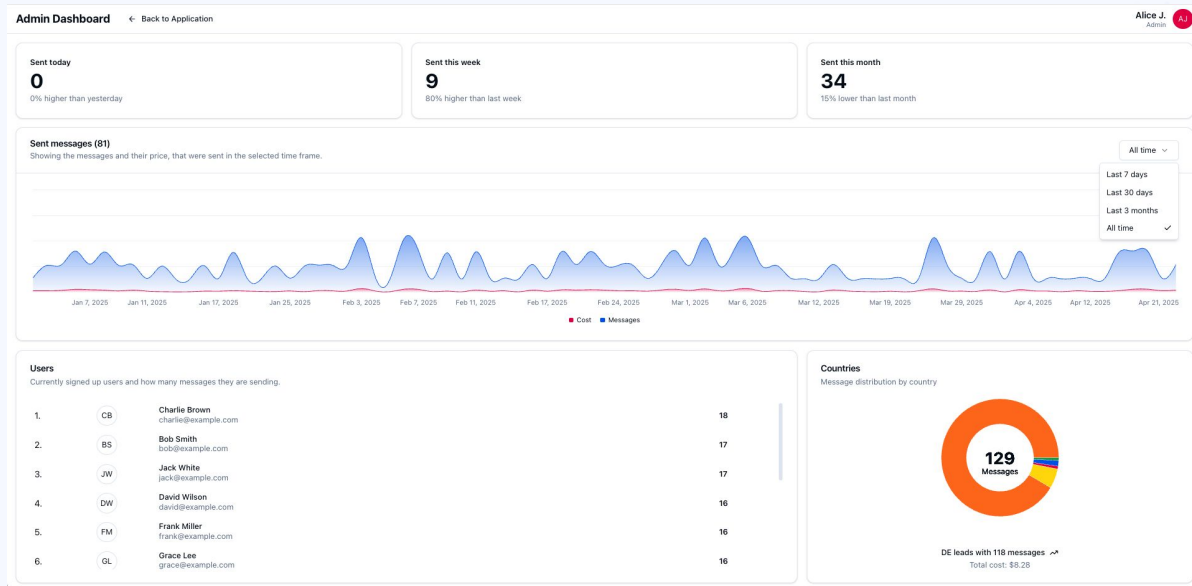
Full Message Actions

- Resend or edit messages
- Move to trash or recover
- Cancel scheduled messages
- Detailed recipient chips with profile tooltips



Admin Dashboard

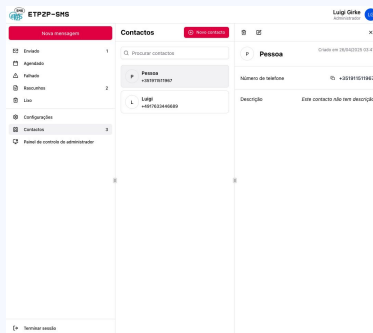
- Stats by user/date/country
- Area and pie charts
- Toggle date range



Other Pages

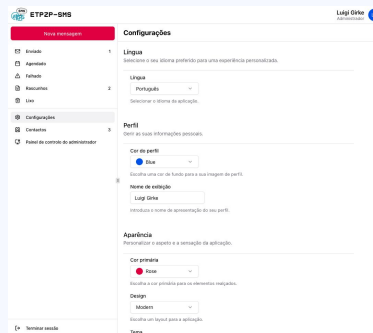
Contacts

- Add, edit, delete contacts
- View contact details



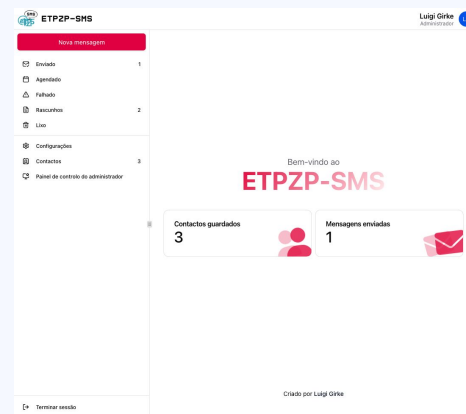
Settings

- Change themes and language
- Update user preferences



Home

- Quick links to pages





02

Tech Deep-Dive

How it works under the hood

Client-Server Model

Client (Browser):

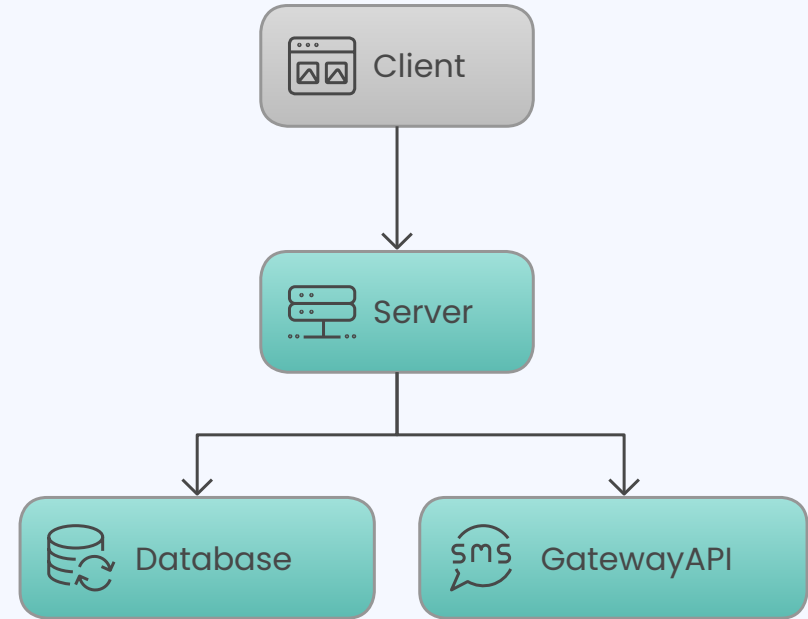
- Renders the UI
- Sends requests to server

Server:

- Handles logic and routing
- Connects to database and GatewayAPI

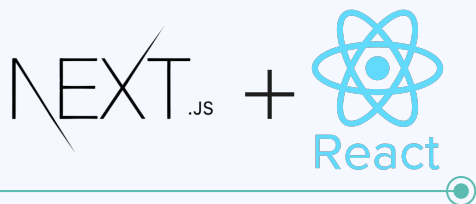
External Services:

- Database: Stores users, messages, contacts
- GatewayAPI: Sends and cancels SMS



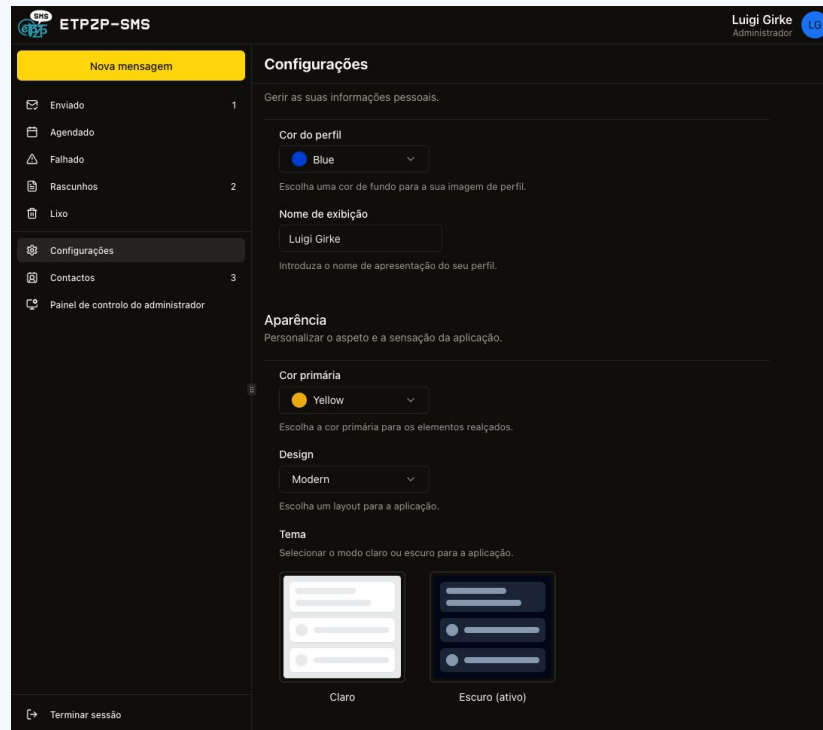
Framework Basics

- Built with React + Next.js
- File-based routing
- Layout and server components



UI & Theming

- ShadCN + TailwindCSS
- Resizable panels
- Dynamic light/dark + color themes



Database Schema

- PostgreSQL
- Users, contacts, messages, recipients
- One message table – one status field

user	
id	serial
name	varchar(255) NN
email	varchar(255) NN
role	varchar(20) NN
created_at	timestamp NN
updated_at	timestamp NN
first_name	varchar(50) NN
last_name	varchar(50) NN
lang	varchar(2) NN
profile_color_id	smallint NN
display_name	varchar(50) NN
primary_color_id	smallint NN
appearance_layout	varchar(20) NN
dark_mode	boolean NN

contact	
id	serial
user_id	integer NN
name	varchar(255) NN
phone	varchar(50) NN
description	varchar(255)
created_at	timestamp NN
updated_at	timestamp NN

message	
id	serial
user_id	integer NN
sender	varchar(255)
subject	varchar(255)
body	text NN
created_at	timestamp NN
send_time	timestamp NN
sms_reference_id	bigint
status	varchar(20) NN
in_trash	boolean NN
api_error_code	smallint
api_error_details_json	text
cost	numeric(6,4)
cost_currency	varchar(10)

recipient	
id	serial
message_id	integer
phone	varchar(50) NN
index	smallint NN

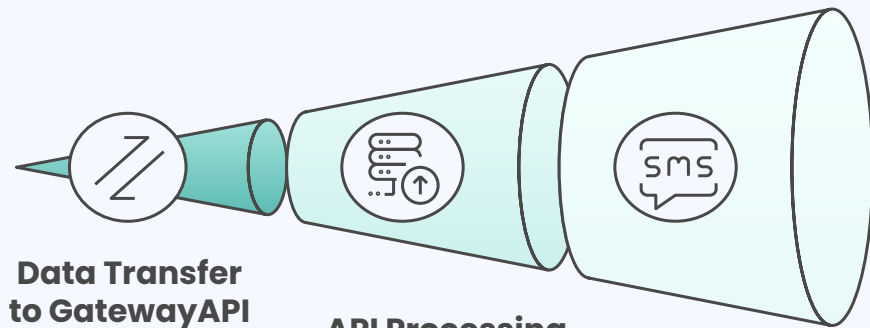
Database Querying

- PostgreSQL + `pg`
- Helper for safe pool-based queries
- Query separation in `lib/db`

```
lib > db > ts index.ts > ...
1  import { Pool, QueryResult } from "pg";
2
3  const pool = new Pool({
4    host: process.env.POSTGRES_HOST,
5    port: Number(process.env.POSTGRES_PORT),
6    user: process.env.POSTGRES_USER,
7    password: process.env.POSTGRES_PASSWORD,
8    database: process.env.POSTGRES_DB,
9  });
10
11 async function db(query: string, params?: any[]): Promise<QueryResult> {
12   const client = await pool.connect();
13   try {
14     const res = await client.query(query, params);
15     return res;
16   } catch (err) {
17     console.error("Database query error", err);
18     throw err; // Rethrow the error for handling in the calling function
19   } finally {
20     client.release(); // Always release the client back to the pool
21   }
22 }
23
24 export default db;
```

SMS Gateway Integration

- GatewayAPI
- REST calls with error feedback
- Cancel via API for refund
- Used SMSAPI first



Data Transfer to GatewayAPI

Data is sent from the app to the API

API Processing

The API processes and prepares the data

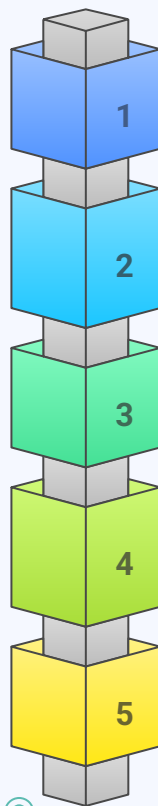
SMS Delivery to Phone

The SMS is delivered to the recipient's phone



Authentication

- Active Directory
- Iron-session for cookies
- Admin detection from AD groups



User Submits Form

User fills out and sends the login form.

Server Verifies Credentials

Server checks user credentials against the AD server.

Cookie Creation

If credentials are valid, a secure cookie is created.

Cookie Check in Middleware

On subsequent requests, the cookie is validated.

User Access Granted

If the cookie is valid, the user gains access.

Self-Hosting



Docker

- Runs node.js server and Postgres in isolated containers
- Ensures consistent environment and easy restarts



Nginx

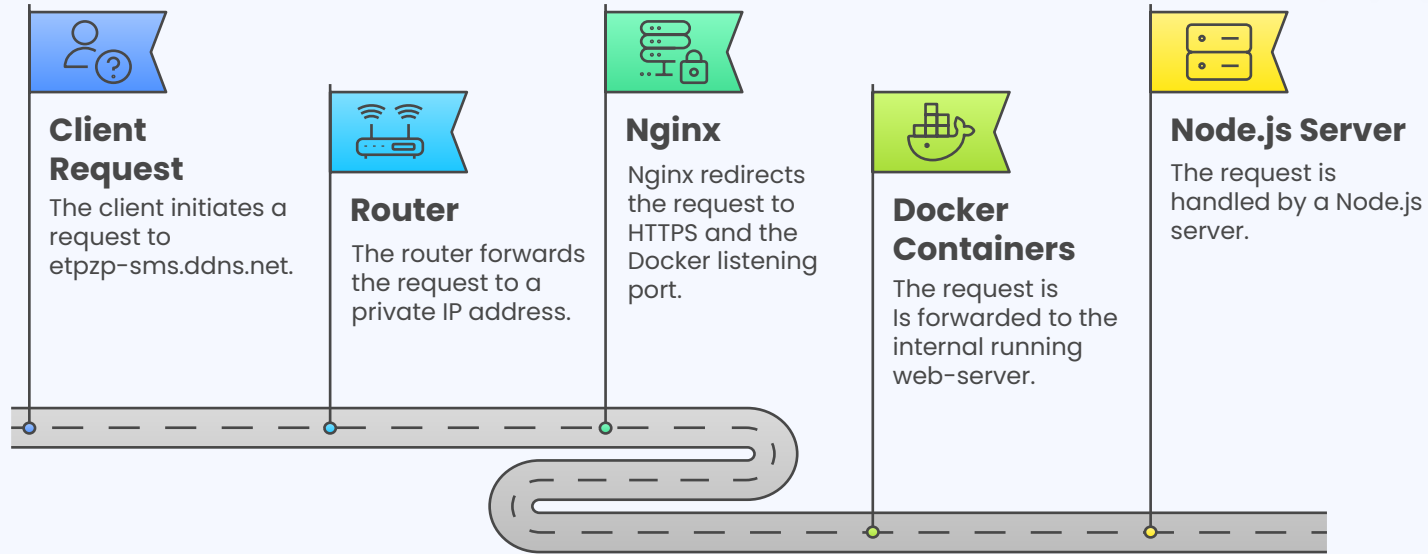
- Reverse proxy for routing traffic to Docker containers
- Sets up SSL and redirects HTTP to HTTPS



No-IP

- Maps public IP to local network
- Enables remote access without a static IP

SMS Application Request Flow





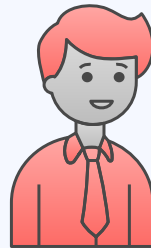
03

Trials & Successes



Turning Problems Into Solutions

Draft Saving Challenge



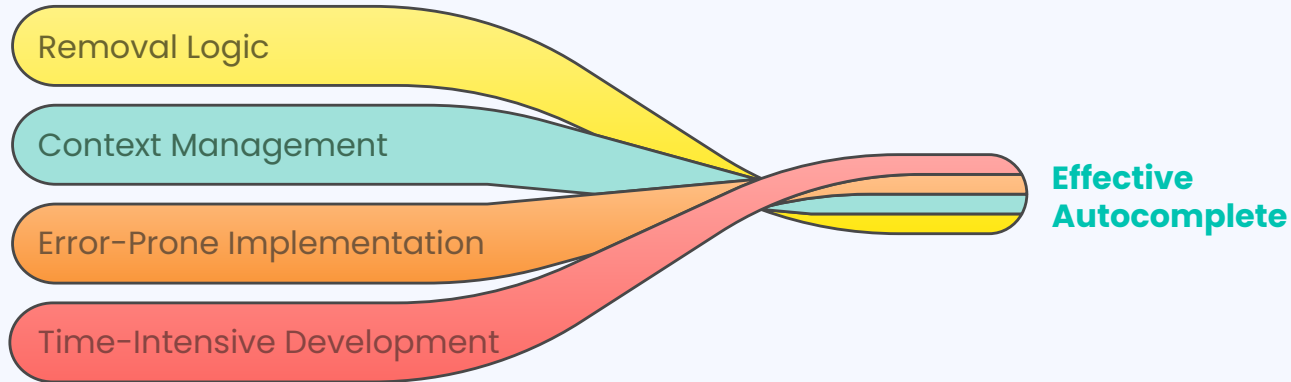
URL updates for saving drafts triggered full re-renders, clearing form values and closing popups.

Introducing a dedicated context to persist draft state across re-renders fixed the problem.



Recipient Input Complexity

- Building the autocomplete/suggestions dropdown was time-intensive and error-prone
- Ensured up-to-date values by carefully managing context and removal logic



Settings Architecture Challenge

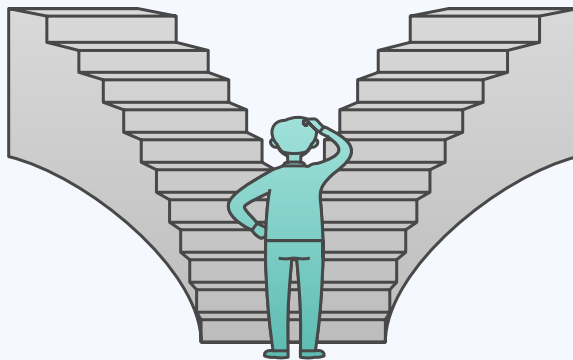
- Local storage writes don't trigger React re-renders
- Settings context enables per-item updates

One Form for All Settings (rejected)

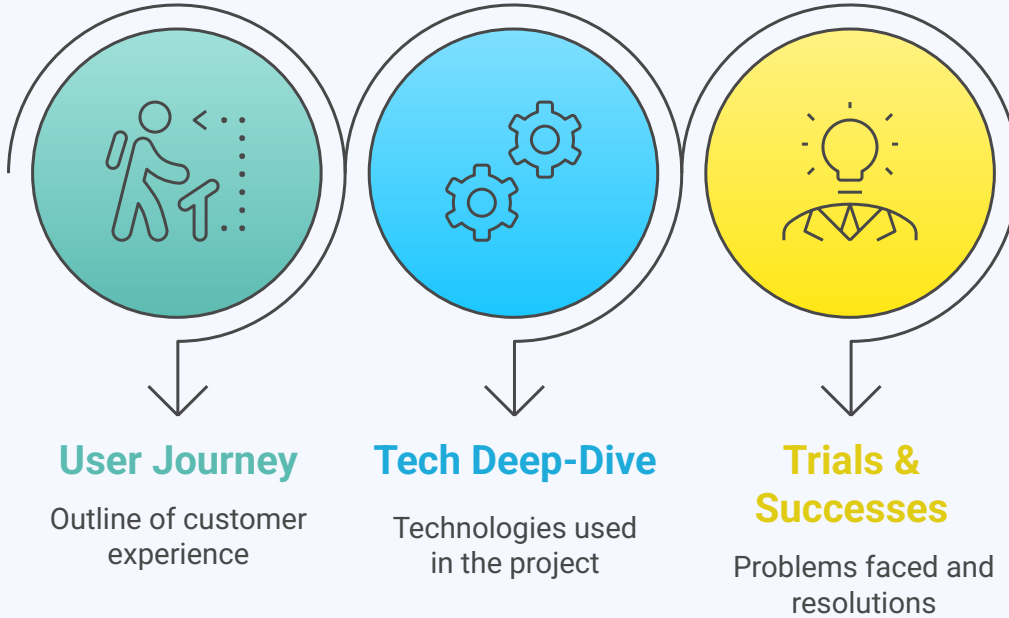
whole-payload saves made
validation/error handling
cumbersome

Multiple Per-Item Forms (chosen)

per-item saves provide instant
persistence, clear errors, and kept UI
reactive



Summary





Thank you

I'm happy to take your questions now