

User Guide: Lead Dashboard UI (Angular application)

The Lead Dashboard is a web-based interface that allows users to view and manage leads fetched from the API. It provides a list of all leads and user can select a lead for more information of each lead.

Accessing the Dashboard

1. Run the Angular Application:

- Navigate to the project directory and run *npm install* to build in bash the application
- Navigate to the project directory and run *npm start* or *ng serve* in bash
- The dashboard will be available at <http://localhost:4200>

2. Open the Dashboard:

- Open a browser and navigate to <http://localhost:4200> to see below page:



Leads Dashboard

Lead Name Phone Number Zip Code

[Azim Karim](#) 916.693.4592 95742

[Bill Smith](#) 123-100-1888 96023

[John Doe](#) 402-122-9900 94303

[Chris Doe](#) 944.819.3701 19303

[Smith Will](#) 233-244-9901 96573

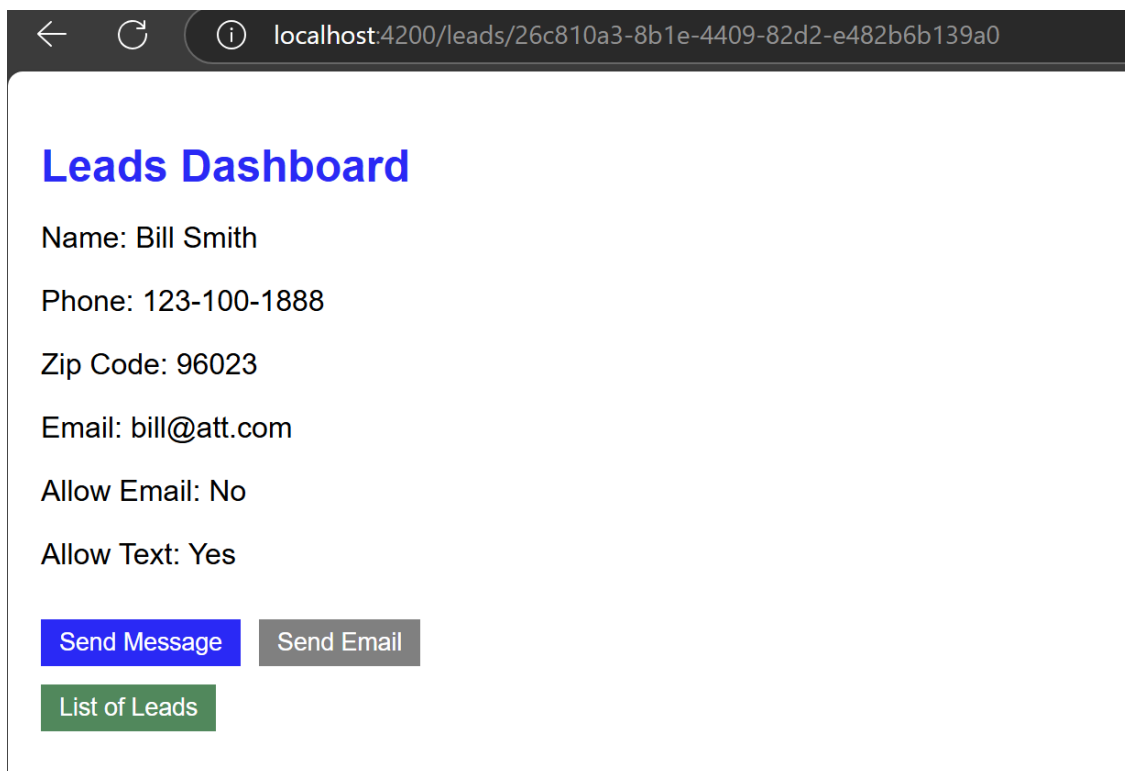
Using the Dashboard Workflow

1. Viewing All Leads

- The homepage displays a list of all leads.
- Each lead is shown with its name, phone number, and zip code. Lead's name is a link to view the Lead's detail.

2. Viewing Lead Details

- Click on a lead's name to view its details.
- The details page shows:
 - Name
 - Phone Number
 - Zip Code
 - Email (optional)
 - Communication Permission – Allow Email (Yes/No)
 - Communication Permission - Allow Text (Yes/No)



← ↻ ⓘ localhost:4200/leads/26c810a3-8b1e-4409-82d2-e482b6b139a0

Leads Dashboard

Name: Bill Smith

Phone: 123-100-1888

Zip Code: 96023

Email: bill@att.com

Allow Email: No

Allow Text: Yes

[Send Message](#) [Send Email](#)

[List of Leads](#)

3. Send email or sms/text if the Lead has the Allow permission

- The button Send Message and Send Email are enabled if the lead has communication permission 'Yes' respectively. So, user can click the button to send the message/email to the lead.
- Successful message/email message will be show after clicking the Send buttons.
- If Allow Email/Allow Text is 'No' the button is disabled and user cannot send message or email to the lead.

4. List of Leads button

- Click List of Leads button to go back to the Leads/List of Leads main page
-

Troubleshooting

- **No Leads Displayed:**
 - Ensure the API is running and accessible.
 - Check the browser console for errors.
 - **Details Not Loading:**
 - Verify the **Lead Id** in the URL is valid.
-

Appendix

API Base URL

- Development: <https://localhost:7169/api> in environment.ts file
- QA/Production: <https://api.yourdomain.com> (Update as needed the environment.prod and environment.qa files respectively)

Dashboard URL

- Development: <http://localhost:4200>
- Production: <https://yourdomain.com> (Update as needed)

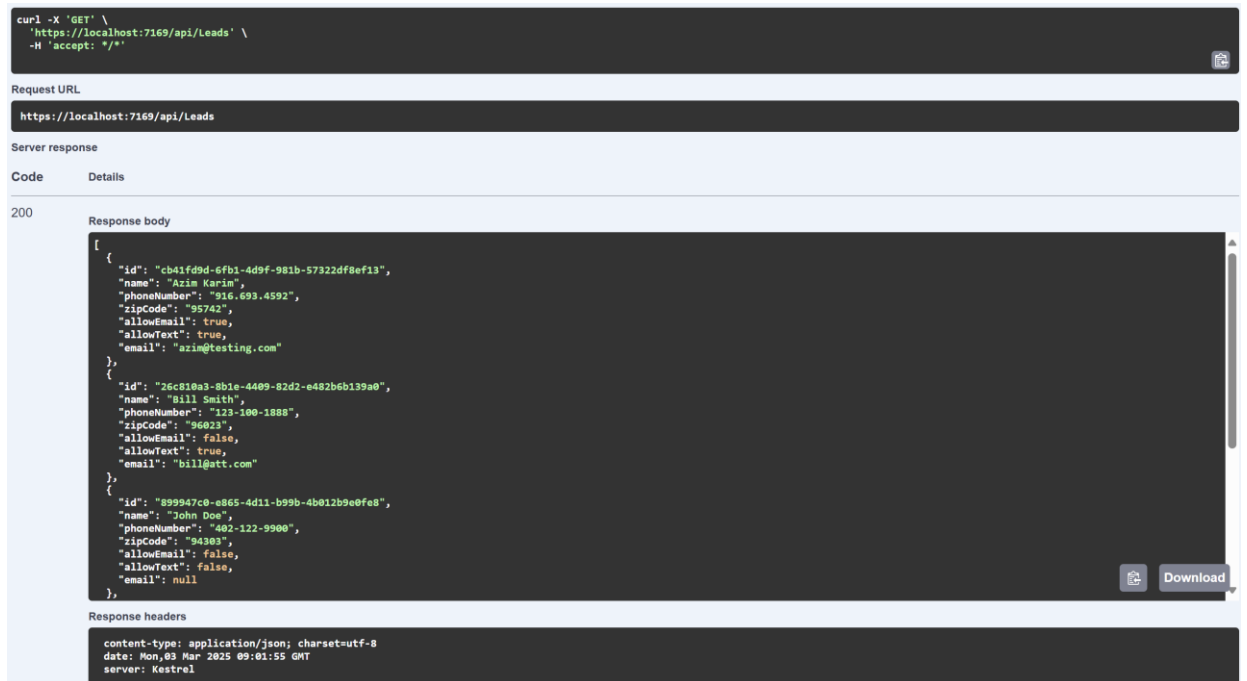
Functional Specification: Leads RESTful API (C# Web API)

The Leads API is a RESTful service to receive leads in real-time from third-party providers. This uses in-memory collection of Leads and provides endpoints to retrieve the leads for the frontend Leads Dashboard purpose.

API Endpoints

1. Retrieve All Leads

- **Endpoint:** GET /api/leads
- **Description:** Returns a list of all leads (stored in memory)
- **Response:**



```
curl -X 'GET' \
  'https://localhost:7169/api/Leads' \
  -H 'accept: */*'

Request URL
https://localhost:7169/api/Leads

Server response
Code    Details
200

Response body
[
  {
    "id": "cb41fd9d-6fb1-4d9f-981b-57322df8ef13",
    "name": "Azim Karim",
    "phoneNumber": "916.693.4592",
    "zipCode": "95742",
    "allowEmail": true,
    "allowText": true,
    "email": "azim@testing.com"
  },
  {
    "id": "26c810a3-8b1e-4409-82d2-e482b6b139a0",
    "name": "Bill Smith",
    "phoneNumber": "123-100-1888",
    "zipCode": "96023",
    "allowEmail": false,
    "allowText": true,
    "email": "bill@att.com"
  },
  {
    "id": "899847c8-e865-4d11-b99b-4b012b9e0fe8",
    "name": "John Doe",
    "phoneNumber": "402-122-9900",
    "zipCode": "94303",
    "allowEmail": false,
    "allowText": false,
    "email": null
  }
]
```

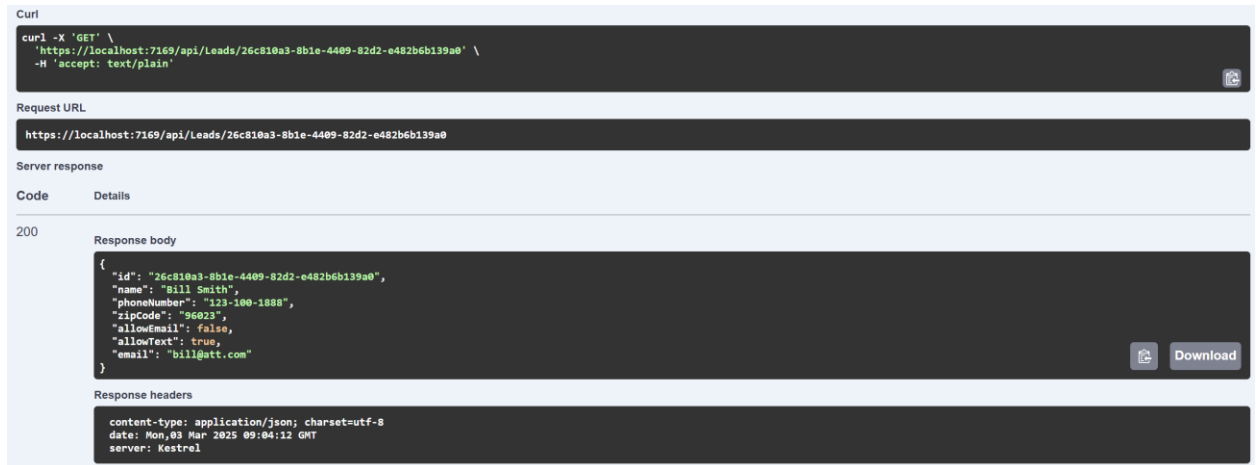
Response headers

```
content-type: application/json; charset=utf-8
date: Mon, 03 Mar 2025 09:01:55 GMT
server: Kestrel
```

2. Get Lead by Lead Id

- **Endpoint:** GET /api/leads/{id}
- **Description:** Returns a lead that matches the lead id (stored in memory)

- **Response:**



How to Use the API

1. Testing the API Locally

1. Run the API:

- Navigate to the project directory and run: `dotnet run`
- The API will start at <https://localhost:7169/swagger/index.html>

2. Retrieve Leads:

- Send a GET request to `https://localhost:7169/api/leads` to view all leads

3. Retrieve Lead by Lead Id:

- Send a GET request to `https://localhost:7169/api/leads/{leadid}` to fetch the Lead. For example: `https://localhost:7169/api/leads/26c810a3-8b1e-4409-82d2-e482b6b139a0`

2. Integrating with Lead Providers

- Provide the api endpoints to the lead providers
- Ensure they respond with the leads data in the specified JSON format

Error Handling

- **400 Bad Request:** Invalid or missing lead data.
- **500 Internal Server Error:** Unexpected server errors.