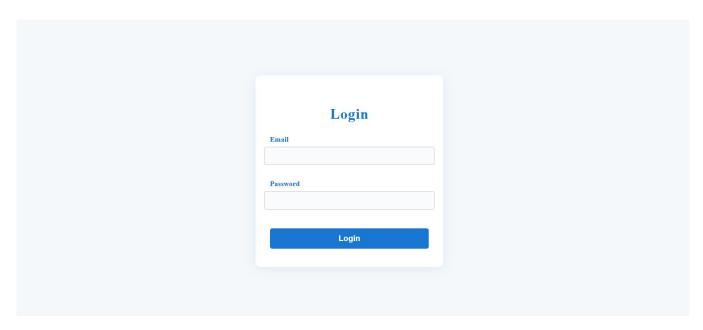# User Application Demonstrating
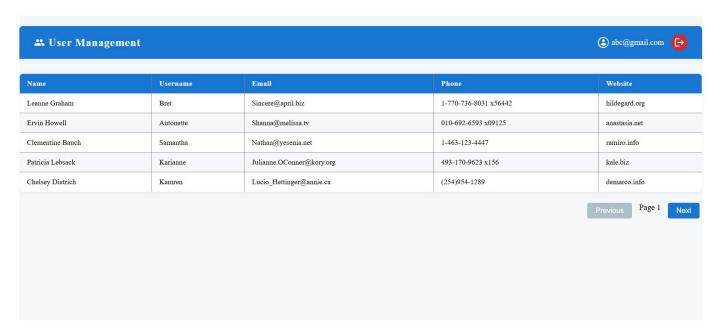
Login Page

What we see:
A centered login form with fields for email and password, styled with a modern card look.

Features:
Email and password fields with validation (required, email format, min length for password).
Error messages shown for invalid input.
On successful login, the user's email is stored in local Storage and you are redirected to the user list.

User List Page:

What you see:

A header bar with the title **"User Management"**, the logged-in user's email, and a logout icon button.
A styled table grid listing users with columns: Name, Username, Email, Phone, Website.
Pagination controls (**"Previous"**, **"Next"**) aligned to the right.

If there are no users on the current page, a "No Result Found" message is shown.

Features:
Pagination: Navigate through user pages using Previous/Next buttons.
Logout: Clicking the logout icon clears local Storage and redirects to the login page.
Auth Guard: Direct access to users without login redirects to /login.

Reusable Components
Button Component:

Used for pagination and other actions.
Accepts a title (btnTitel), disabled state, and emits an event on click.
Header Component:

Displays the app title, logged-in user, and logout icon.
Used at the top of the user list page.

Routing
Lazy-loaded routes for both login and user list.
Auth Guard protects the user list route.

Styling
Modern, responsive UI with Material Icons.
Consistent color scheme and spacing for a professional look.

Demonstrate:
- Start the app (ng serve).
- Go to /login:
- Try submitting empty or invalid forms to see validation.
- Enter valid credentials and submit.
- You are redirected to users:
- See the user grid, header, and pagination.
- Try pagination and logout.
- After logout, you are redirected to /login and cannot access users without logging in again.