



Frontend Technical Workshop



Contact Presenter

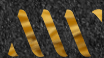
Manu Mohanan Nair

(+91) 8921 41 8921

future@mmnair.com

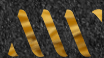
What's Angular ?

- A TypeScript-based open-source web application framework
 - Component-based architecture
 - Two-way data binding
 - Dependency injection
 - CLI for rapid development
 - Rich ecosystem of libraries and tools



Why Angular ?

- Improved code maintainability and scalability
- Faster development cycles
- Enhanced user experience
- Strong community support
- Integration with other popular tools and frameworks



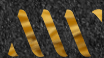
Angular Architecture

- Components: Building blocks of Angular applications
- Templates: HTML structures with Angular-specific syntax
- Modules: Groups of related components, directives, and pipes
- Services: Reusable logic and data management



Common Angular Commands

- `ng generate component my-component`
- `ng serve`
- `ng build`
- `ng test`
- `ng add [package]`



Module 01: Angular Key Areas

- Components and Templates
 - Data Binding
 - One-way Binding
 - Two Way Binding
 - Directives
 - Services
 - Dependency Injection
- Routing
 - Forms
 - HTTP Requests
 - Observables

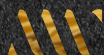


Workshop 01:

Create a simple Angular app

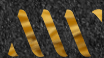
- Create a new Angular app using the Angular CLI.
- Add a component to the app.
- Bind data to the component's template.
- Create a service to fetch data from an API.
- Display the data in the component's template.

NB: For all angular apps that you are creating need to have a backend API to integrate it to.



Module 02: Angular Key Areas

- Forms
 - Pipes
 - Animations
- State management
 - Testing
 - Deployment



Workshop 02:

Create a Real-world Angular App

Option 1

To-Do List App:

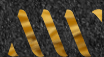
- **Core Features:**
 - Add, edit, and delete tasks
 - Mark tasks as completed/incomplete
 - Filter tasks by status (completed, incomplete, all)
- **Additional Features (if time permits):**
 - Set due dates and reminders
 - Organize tasks into categories or projects
 - Integrate with a task management API

Option 2

Simple Blog App:

- **Core Features:**
 - Create, edit, and delete blog posts
 - Display blog posts in a list or grid view
 - Allow users to comment on posts
- **Additional Features (if time permits):**
 - Add categories or tags for better organization
 - Implement a search functionality
 - Integrate with a content management system (CMS)

NB: For all angular apps that you are creating need to have a backend API to integrate it to.



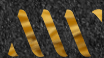
Angular Auth2

- Authentication
- Authorization



Angular Material

- Pre-built UI Components: Buttons, cards, forms, grids, and more
- Customization: Theming and customization options



Angular i18n

- Internationalization (i18n)
 - Separating text content from code.
 - Using placeholders for dynamic content.
 - Considering layout and design for different languages and cultures.
- Localization (l10n)
 - Translating text content
 - Formatting numbers, dates, and currencies
 - Handling time zones and calendars
 - Adapting layout and design



Workshop 03:

Deploy the App

- Identify the potential of App
- Choose a platform for deployment
- What all needs to deploy?
- Deploy the Backend App
- Deploy the Angular App
- Share the weblink to access the App.



Angular 16 v/s 17

- **Stable Signals and Effects:** This feature has been stabilized, making it ready for production use. It provides a more intuitive and efficient way to build reactive Angular applications.
- **Improved Router:** Enhanced lazy loading and error handling capabilities for a more robust routing experience.
- **New Control Flow Syntaxes (Developer Preview):** Introduced experimental syntax for more concise and expressive template logic.



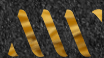
Angular 17 v/s 18

- **Zoneless Architecture (Experimental):** This feature aims to improve application performance by reducing the overhead of change detection. While experimental, it has the potential to significantly boost app speed.
- **Material 3:** The latest Material Design system is now stable, offering modern UI components and theming options.
- **Enhanced Developer Experience:**
 - Improved debugging tools and error messages.
 - More streamlined development workflows.



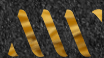
Angular Universal

- **Server-Side Rendering (SSR):** Improving SEO and initial page load times



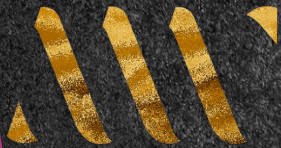
Angular Best Practices

- **Component Reusability:** Creating reusable components
- **Code Organization:** Following Angular style guidelines
- **Testing:** Writing unit and integration tests
- **Performance Optimization:** Optimizing application performance





Thank You!!!



Contact Presenter

Manu Mohanan Nair

(+91) 8921 41 8921

future@mmnair.com