Perfect! Since your 10 hours are just for learning Angular concepts (not building the project yet), here’s what you should focus on:

**Core Angular Concepts (Must-Have)**

**1. Environment Setup and Project Structure (30 min)**

* Install Node.js and Angular CLI.
* Create a new Angular project: ng new project-name.
* Understand the structure of an Angular project (src/app, app.module.ts, etc.).

**Key Files:**

* app.component.ts: Root component.
* app.module.ts: Defines your app and its modules.

**2. Components and Data Binding (1.5 hours)**

**Components:** Building blocks of Angular. (ng generate component component-name)

**Learn:**

* Property Binding ([ ]): Bind component properties to the DOM.
* Event Binding (( )): Handle user events like clicks.
* Two-Way Binding ([(ngModel)]): Sync input values with component properties.

**Example:** Build a counter component with + and - buttons.

**3. Directives (1 hour)**

Directives are used to manipulate DOM elements.

**Structural Directives:**

* \*ngIf: Conditional rendering.
* \*ngFor: Loop through a list.

**Attribute Directives:**

* ngClass and ngStyle for styling.

**Example:** Build a list of tasks and toggle their completion status with \*ngIf.

**4. Services and Dependency Injection (1.5 hours)**

Services handle business logic and data sharing between components. (ng generate service service-name)

**Learn:**

* Create and inject a service using @Injectable.
* Share data between components using a service.
* Use HttpClient for making HTTP requests.

**5. Routing and Navigation (1.5 hours)**

Routing helps you build a single-page application with multiple views.

**Learn:**

* Configure routes in app-routing.module.ts.
* Use router-outlet and routerLink for navigation.
* Pass route parameters and fetch them using ActivatedRoute.

**Example:** Create a multi-page app with Home, About, and Contact pages.

**6. Forms (2 hours)**

**Template-Driven Forms** (simple) vs **Reactive Forms** (powerful and scalable).

**Focus on Reactive Forms:**

* FormGroup, FormControl, Validators.
* Handle form submission and validation.

**Example:** Build a registration form with validation for fields like name, email, and password.

**7. HTTP Client (1.5 hours)**

Learn to interact with a backend API using HttpClient.

**Common Operations:**

* GET: Fetch data.
* POST: Add new data.
* PUT: Update existing data.
* DELETE: Remove data.

**Example:** Fetch and display a list of products from a REST API.

**8. Error Handling and Observables (30 min)**

* Handle errors in HTTP requests with catchError.
* Understand basic Observables and how to use .subscribe().

**Summary (What You’ll Master in 10 Hours)**

* Components and Data Binding
* Directives
* Services and Dependency Injection
* Routing and Navigation
* Reactive Forms
* HTTP Client and Backend Integration

**Optional (if time allows):**

* Angular Material for UI components (ng add @angular/material).
* State Management with RxJS or @ngrx/store.

Want me to create a detailed schedule (hour-by-hour) with learning resources?