**Prerequisites**

* 1. Moderate knowledge of HTML, CSS, and JavaScript
  2. Basic Model-View-Controller (MVC) concepts
  3. The Document Object Model (DOM)
  4. JavaScript functions, [events](https://developer.mozilla.org/en-US/docs/Web/API/Event), and [error handling](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/try...catch)

1. **Angularjs**
   1. Overview of Angular JS Framework (Available Features Which makes it robust)
   2. Single Page Application (SPA Benefits)
   3. Data binding
   4. Expressions
   5. Scope and data model
   6. Controllers
   7. Directive (ng-repeat, switch, ng-if etc.)
   8. Module
2. **Life cycle of Angularjs**
3. Bootstrap phase
4. Compile phase
5. Runtime phase

**By 1st day**

1. **Scope**
2. Relationship between scope and controller
3. Relationship between scope and template
4. Scope life cycle
5. **Dependency injection**
6. Value
7. Factory
8. Service
9. Provider
10. Constant

**By 2nd day**

1. **Directives**
2. Predefined directives (ng-repeat, ng-veiw)
3. Custom directive ( Advantage of creating Custom Directives)
4. Directive Phases
5. Compile and link Phase
6. Restrict
7. Template
8. Transclude
9. Scope Vs isolated scope
10. Parameter passing to directive
11. Link and controller functions

**By 3rd day**

1. **Events**
2. Event life cycle
3. Predefined events
4. Custom events
5. $watch, $emit and $broadcast
6. **Angular templates**
7. **Create Custom Templates**
8. How to reuse & Include Templates with in a View
9. **Routing and location**
10. Basic routing
11. Data passing to route

**By 4thday**

1. **Services(REST API Implementation)**
2. q service and promises
3. Exception handling
4. **Animation**
5. **Debugging Angular app**
6. Basic debugging
7. Debugging with browser extensions
8. Debugging the scope values
9. Debugging in devices
10. **Performance**
11. Difference prospective to consider performance of the applications

**\*\*\*\*\*\*\*\*\*\*Doubts clarifications\*\*\*\*\*\*\*\*\***

**By 5th day**