**Angular js**

**Duration: 2.5 days**

**Level – Intermediate to Advanced**

**Prerequisite : Javasrcript OOPS,DOM,CSS,Node & Angular Anatomy**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Day – 1:**

**Angular Vs Backbone/EXT JS/Knockout & Ember JS**

**REST application with Angular JS:**

***Getting Started:part 1***

***Preparing the Data***

***Preparing the Application***

***Displaying the Product Data***

***Creating the Controller***

***Displaying the Product Details***

***Displaying the Category List***

***Creating a List of Categories***

***Generating the Category Navigation Links***

***Selecting the Category***

***Highlighting the Selected Category***

***Adding Pagination***

***Navigation and Checkout:part 2***

***Using the Real Product Data***

***Handling Ajax Errors***

***Creating Partial Views***

***Creating the Cart***

***Defining the Cart Module and Service***

***Creating a Cart Widget***

***Adding Product Selection Buttons***

***Adding URL Navigation***

***Defining URL Routes***

***Using URL Routing to Navigate***

***Starting the Checkout Process***

***Applying the Checkout***

***Orders and Administration :part 3***

***Getting Shipping Details***

***Adding Form Validation***

***Adding the Remaining Form Fields***

***Placing Orders***

***Extending the Deployd Server***

***Defining the Controller Behavior***

***Calling the Controller Behavior***

***Defining the View***

***Making Improvements***

***Administering the Product Catalog***

***Creating the Admin Application***

***Implementing Authentication***

***Defining the Main View and Controller***

***Implementing the Orders Feature***

***Implementing the Products Feature***

***Working with Forms***

***Preparing the Example Project***

***Using Form Elements with Two-Way Data Bindings***

***Implicitly Creating Model Properties***

***Checking That the Data Model Object Has Been Created***

***Validating Forms***

***Performing Basic Form Validation***

***Providing Form Validation Feedback***

***Using CSS to Provide Feedback***

***Using the Special Variables to Provide Feedback***

***Deferring Validation Feedback***

***Using the Form Directive Attributes***

***Using Input Elements***

***Using Text Areas***

***Using Select Elements***

***Using Controllers and Scopes***

***Why and When to Use Controllers and Scopes***

***Preparing the Example Project***

***Understanding the Basics***

***Creating and Applying Controllers***

***Setting Up the Scope***

***Modifying the Scope***

***Organizing Controllers***

***Using a Monolithic Controller***

***Reusing a Controller***

***Using Controller Inheritance***

***Using Multiple Controllers***

***Using Scope-less Controllers***

***Explicitly Updating the Scope***

***Working with Filters:***

***Creating Custom Filters***

***Creating a Filter That Formats a Data Value***

***Creating a Collection Filter***

***Building on Existing Filters***

***Creating Complex Directives***

***Preparing the Example Project***

***Defining Complex Directives***

***Defining How the Directive Can Be Applied***

***Using Directive Templates***

***Using a Function as a Template***

***Using an External Template***

***Selecting an External Template with a Function***

***Replacing the Element***

***Managing Directive Scopes***

***Creating Multiple Controllers***

***Giving Each Directive Instance Its Own Scope***

***Creating Isolated Scopes***

**Day – 2:**

***Advanced Directive Features***

***Preparing the Example Project***

***Using Transclusion***

***Using Compile Functions***

***Using Controllers in Directives***

***Adding Another Directive***

***Creating Custom Form Elements***

***Handling External Changes***

***Handling Internal Changes***

***Formatting Data Values***

***Validating Custom Form Elements***

***Services for Global Objects, Errors, and Expressions***

***Preparing the Example Project***

***Accessing the DOM API Global Objects***

***Why and When to Use the Global Object Services***

***Dealing with Exceptions***

***Why and When to Use the Exception Service***

***Working with Exceptions***

***Working Directly with the Exception Service***

***Implementing a Custom Exception Handler***

***Working with AngularJS Expressions and Directives***

***Why and When to Use the Expression and Directive Services***

***Converting Expressions into Functions***

***Interpolating Strings***

***Compiling Content***

***Services for Ajax and Promises***

***Configuring Ajax Requests***

***Setting Ajax Defaults***

***Using Ajax Interceptors***

***Working with Promises***

***Getting and Using the Deferred Object***

***Consuming the Promise***

***Understanding Why Promises Are Not Regular Events***

***Chaining Outcomes Together***

***Grouping Promises***

***Services for REST***

***Why and When to Use the REST Services***

***Preparing the Example Project***

***Creating the RESTful Service***

***Creating the AngularJS Application***

***Using the $http Service***

***Listing the Product Data***

***Deleting Products***

***Creating Products***

***Updating Products***

***Testing the Ajax Implementation***

***Hiding the Ajax Requests***

***Installing the ngResource Module***

***Using the $resource Service***

***Configuring the $resource Service Actions***

***Creating $resource-Ready Components***

***Services for Views***

***Why and When to Use the View Services***

***Preparing the Example Project***

***Understanding the Problem***

***Using URL Routing***

***Installing the ngRoute Module***

***Defining the URL Routes***

***Displaying the Selected View***

***Wiring Up the Code and Markup***

***Using Route Parameters***

***Accessing Routes and Routes Parameters***

***Configuring Routes***

***Using Controllers with Routes***

***Adding Dependencies to Routes***

***AngularUI Router***

***Services for Provision and Injection***

***Why and When to Use the Provision and Injection Services***

***Preparing the Example Project***

***Registering AngularJS Components***

***Managing Injection***

***Determining Function Dependencies***

***Obtaining Service Instances***

***Simplifying the Invocation Process***

***Getting the $injector Service from the Root Element***

***Localization***

***angular-translate***

***Installation***

***Teaching Your App a New Language***

***Multi-language Support***

***Switching the Language at Run Time***

***Loading Languages***

***angular-gettext***

***Installation***

***Usage***

***String Extraction***

***Translating Our Strings***

***Compiling Our New Language***

***Changing Languages***

***Caching***

***What Is a Cache?***

***Angular Caching***

***Caching through $http***

***Setting Default Cache for $http***

***Security***

***Strict Contextual Escaping: the $sce Service***

***Whitelisting URLs***

***Blacklisting URLs***

***$sce API***

***Configuring $sce***

***Trusted Context Types***

***Scalable Architecture for Deployments on AWS:***

***Understanding the various services in Amazon AWS***

***Delving into AWS deployment architectures***

***The EC server-based architecture***

***The Server-less Architecture***

***Deploying our app in a Server-less Architecture on AWS***

***Mapping a domain to S3***

***Mapping the S3 bucket to a CloudFront distribution***

***Getting your app ready for production deployment***

***Improving the page-load time of your app***

***Setting Expires headers***

***Performance***

***Building an E-Commerce Store:***

***Backend as a Service***

***Building a BaaS platform on AWS***

***Setting up an S3 Bucket with public read access***

***Setting up the CORS policy on your S bucket***

***Creating our DynamoDB tables***

***Creating the Identity and Access Management (IAM) role***

***Creating our e-commerce app***

***Building nested views using UI-Router***

***Mapping states to URL, views, and controllers***

***Prototyping our application***

***Setting up our index.html file***

***Creating the controllers***

***Creating the product partials***

***Adding animations to the view transitions***

***Adding in the CSS transition effects***

***Creating our application-level controller***

***Adding a Facebook login***

***Integrating AWS JS SDK with our application***

***Creating the AWS service provider***

***Building our Add Products page***

***Saving data in DynamoDB tables***

***Creating the view for the add product form***

***Building the controller for the add products view***

***Uploading images to S3***

***Fetching the products lists for a category***

***Using resolves to preload data***

***Creating our product details page***

***Adding products to cart***

***The checkout page***

***Saving the orders***

***Completing the order***

***Best Practices :***

***Organizing your Code***

***Writing Controllers & Services***

***Writing Directives and Views***

***Dealing with Scope***

***Communicating Between Components***

***Breaking Down A Page into Components***

***Communicating with The Server***

***Dealing with Models***

***Separation of Concerns***

***Organizing Your App***

***Modules***

***Readable Code and AngularJS***

***Controller Patterns***

***Exception Handling***

***Using a Team Style Guide***

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Day – 4 :**

***Angular Testing with Jasmine & Protactor:***

***Setup and Configuration:***

***Introduction***

***Creating an AngularJS application***

***Running a simple test using Jasmine***

***Installing Protractor***

***Running a simple test using Protractor***

***Installing Karma***

***Running tests using Karma***

***Installing Testem***

***Running tests using Testem***

***Automating test runners using Grunt***

***Automating test runners using Gulp***

***Getting Started with Testing and AngularJS:***

***Introduction***

***Loading a module***

***Writing a test spec***

***Debugging AngularJS code***

***Mocking injected instances using an object***

***Mocking injected instances using spies***

***Test Navigation and Routing:***

***Introduction***

***Getting started with testing using ngRoute***

***Testing route parameters with ngRoute***

***Getting started with testing using ui-router***

***Testing the transitioning state with ui-router***

***Testing URL parameters with ui-router***

***Testing page loading using Protractor***

***Testing navigation using Protractor***

***Testing redirection using Protractor***

***Testing Controllers***

***Introduction***

***Setting up for testing a controller***

***Testing the initial state of a scope object***

***Testing the initial state of a scope object with Protractor***

***Testing interactive scope changes with Protractor***

***Testing navigation scope changes with Protractor***

***Testing User Interaction and Directives:***

***Introduction***

***Starting with testing directives***

***Setting up templateUrl***

***Searching elements using selectors***

***Accessing basic HTML content***

***Accessing repeater content***

***Scope changes based on user input***

***Scope changes based on DOM events***

***Class changes based on window properties***

***Directive changes on interaction using Protractor***

***Using Spies to Test Events***

***Introduction***

***Testing event dispatches***

***Testing the handling of dispatched events***

***Testing the handling of external events***

***Testing the handling of callbacks***

***Testing events using Protractor***

***Testing Filters***

***Introduction***

***Testing a filter that formats a number as text***

***Testing a filter that formats seconds to a time string***

***Using Protractor to test filter changes based on input***

***Using Protractor to test filter changes based on events***

***Service and Factory Testing with Mocks and Spies***

***Introduction***

***Getting started with testing a service***

***Testing HTTP GET requests using $httpBackend***

***Testing HTTP POST requests using $httpBackend***

***Using spies to test HTTP GET requests***

***sing spies to test HTTP POST requests***

***Testing service data using mock helpers***

***Testing rejected $http promises***

***Testing constants***

***Using Protractor to test HTTP requests***

***Testing Animations:***

***Introduction***

***Synchronous testing of animations***

***Testing animations with ngAnimateMock***

***Asynchronous testing of animations***