Course Code	*PCA20D01J*	Course Name	ADVANCED WEB APPLICATION DEVELOPMENT	Course Category	D	*Discipline Elective* Course	L	T	P	C
							3	0	2	4

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computer Applications		Data Book / Codes/Standards	Nil	

Course Learning Rationale (CLR):	The purpose of learning this course is to,
CLR-1 :	*Learn fundamental Javascript concepts that power AngularJS.*
CLR-2 :	*Write quicker, better AngularJS code by discovering how AngularJS itself is built.*
CLR-3 :	*Become fluent in AngularJS terminology, such as dependency injection, services, directives, transclusion, and more.*
CLR-4 :	*Realize the power of dependency injection, and how AngularJS accomplishes it*
CLR-5 :	*Design custom directives and save time and energy with easily reusable components. Understand what a Single Page Application (SPA) is, and how they work.*
CLR-6 :	*Build a Single Page Application (SPA) in AngularJS. Be the coder that explains AngularJS to everyone else, because you understand it better than anyone else.*

Course Learning Outcomes (CLO):	At the end of this course, learners will be able to:	Learning Level of Thinking (Bloom)	Expected Proficiency (%)	Expected Attainment (%)
CLO-1 :	Understand the design of single-page applications and how AngularJS facilitates their development	2	85	80
CLO-2 :	Properly separate the model, view, and controller layers of your application and implement them using AngularJS	3	85	80
CLO-3 :	Master AngularJS expressions, filters, and scopes	3	85	80
CLO-4 :	Build Angular forms	3	85	80
CLO-5 :	Elegantly implement Ajax in your AngularJS applications	3	85	80
CLO-6 :	Write AngularJS directives, Unit test and end-to-end test your AngularJS application	3	85	80

Program Learning Outcomes (PLO)	1 Disciplinary Knowledge	2 Critical Thinking	3 Problem Solving	4 Analytical Reasoning	5 Research Skills	6 Team Work	7 Scientific Reasoning	8 Reflective Thinking	9 Self-Directed Learning	10 Multicultural Competence	11 Ethical Reasoning	12 Community Engagement	13 ICT Skills	14 Leadership Skills	15 Life Long Learning
CLO-1 :	*L*	*H*	-	*H*	*L*	-	-	-	*L*	*L*	-	*H*	-	-	-
CLO-2 :	*M*	*H*	*L*	*M*	*L*	-	-	-	*M*	*L*	-	*H*	-	-	-
CLO-3 :	*M*	*H*	*M*	*H*	*L*	-	-	-	*M*	*L*	-	*H*	-	-	-
CLO-4 :	*M*	*H*	*M*	*H*	*L*	-	-	-	*M*	*L*	-	*H*	-	-	-
CLO-5 :	*H*	*H*	*M*	*H*	*L*	-	-	-	*M*	*L*	-	*H*	-	-	-
CLO-6 :	*L*	*H*	-	*H*	*L*	-	-	-	*L*	*L*	-	*H*	-	-	-

Duration (hour)		15	15	15	15	15
S-1	SLO-1	Introducing Full-Stack Development	Creating and setting up a MEAN Project	Building a data model with MongoDB and Mongoose	REST API- POST Methods: Adding data to MongoDB	Building a single-page application with Angular: Foundations
	SLO-2	Why learn the full stack?	A brief look at Express, Node, and npm	Connecting the Express application to MongoDB	PUT Methods: Updating data in MongoDB	Setting the groundwork for an Angular SPA
S-2	SLO-1	Introducing Node.js: The web server/platform	Creating an Express project	Why model the data?	Updating an existing subdocument in MongoDB	Switching from Express routing to Angular routing
	SLO-2	Introducing Express: The framework	Modifying Express project	Defining Simple Mongoose schemas	Deleting method: Deleting data from MongoDB	Adding the first views, Controllers, and services

S-3	SLO-1	Introducing MongoDB: The database	Import Bootstrap for quick, responsive layouts	Creating more complex schemas with subdocuments	Deleting a subdocument from MongoDB	Improving browser performance
	SLO-2	Introducing AngularJS: The front-end framework	Make it Live on Heroku	Final schema	How to call an API from Express	Manually injecting dependencies to protect against minification
S-4 S-5	SLO-1 SLO-2	**Lab 1: Sample application**	**Lab 4: How to move data from view to the controller**	**Lab 7: Pushing up the data**	**Lab 10: Building the API request**	**Lab 13: Passing Data into Modal**
S-6	SLO-1	Supporting cast	Getting Heroku set up	Compiling Mongoose schemas into models	Using the request module	Using Uglify JS to minify and concatenate scripts
	SLO-2	Hosting with Heroku	Pushing the site live using Git	Using the MongoDB shell to create a MongoDB database and add data	Using list of data from an API: The Loc8r homepage	Building an SPA with Angular: The next level
S-7	SLO-1	Designing a MEAN stack architecture	Defining the routes in Express	MongoDB shell basics	Separating concerns: Moving the rendering into a named function	A full SPA: Removing reliance on the server-side application
	SLO-2	A Common MEAN stack architecture	Building basic controllers	Creating a MongoDB database	Catching errors returned by the API	Adding additional pages and dynamically injecting HTML
S-8	SLO-1	Looking beyond SPAs	Creating some views	Getting our database live	Adding Angular components to an Express application	Creating a filter to transform the line breaks
	SLO-2	Designing a flexible MEAN architecture	A look at Bootstrap	Setting up Mongo Lab and getting the database URI	Uncovering two-way data binding	Sending HTML through an Angular binding
S-9 S-10	SLO-1	**Lab 2: Planning a real application**	**Lab 5: Setting up the HTML framework with Jade templates and Bootstrap**	**Lab 8: Making the application use the right database**	**Lab 11: Displaying and filtering the homepage list**	**Lab 14: More complex views and routing parameters**
	SLO-2	**Breaking the development into stages**	**Jade templates and Bootstrap**	**Pushing up the data**	**Using Angular filters to format data**	**Using URL parameters in controllers and services**
S-11	SLO-1	The Steps to built Loc8r	Building a Template	Writing a REST API" Exposing the MongoDB database to the application	Adding HTML geolocation to find places near you	Building the Details page view
	SLO-2	Hardware architecture	Adding the rest of the views	The rules of a REST API	Using services for data	Using Angular UI components to create a modal popup
S 12-13	SLO-1 SLO-2	How the MEAN stack components work together	How to more data from the view to the controller	Deleting document in MongoDB	Modifying data before displaying it: Fixing the distances	Creating Modal using Angular UI Components
S-14	SLO-1	**Lab 3: Development hardware**	**Lab 6: Take the data out of the views and make them smarter**	**Lab 9: Setting up the API in Express**	**Lab 12: Making HTTP requests from Angular to an API**	**Lab 15: Adding and using a click handler**
S-15	SLO-2	**Production hardware.**	**Updating the rest of the views and controllers**	**GET methods : Reading data from MongoDB**	**Ensuring forms work as expected**	**Using the form to submit a review.**

Learning Resources	*Text Book: Getting MEAN with Mongo, Express Angular and Node, Simon Holmes* *Practical Node JS: Building a Real World Scale Web Apps, Basarat Syed, A Press, 2014.* *Learning Angular JS: A Guide to Angular JS Development, Ken Williamson, O' Reilly, 2015*	*Reference Book:* *1. MEAN Web Development, AMOS Q. HAVIV* 2. AngularJS: https://angular.io/docs 3. MongoDB: https://docs.mongodb.com/manual/tutorial/getting-started/

Learning Assessment

Level	Bloom's Level of Thinking	Continuous Learning Assessment (50% weightage)								Final Examination (50% weightage)	
		CLA – 1 (10%)		CLA – 2 (10%)		CLA – 3 (20%)		CLA – 4 (10%)#			
		Theory	Practice	Theory	Practice	Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	20%	20%	15%	15%	15%	15%	15%	15%	20%	20%
	Understand										
Level 2	Apply	20%	20%	20%	20%	20%	20%	20%	20%	20%	20%
	Analyze										
Level 3	Evaluate	10%	10%	15%	15%	15%	15%	15%	15%	10%	10%
	Create										
	Total	100 %		100 %		100 %		100 %		100 %	

Course Designers

Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
Mr.G.Muruganandam, Group Project Manager, HCL Technologies, Chennai	*Dr.S.Gopinathan, Professor, University of Madras, Chennai*	*Mr.N.KRISHNAMOORTHY*
Mr.M. Hemachandar, Tech Lead, Wipro Limited, Chennai		