



**SWAMINARAYAN
UNIVERSITY**
THE RESERVOIR OF ALL KNOWLEDGE

FACULTY OF IT AND COMPUTER SCIENCE

COURSE CURRICULUM

Course Title: MOBILE APPLICATION USING FLUTTER
(Code: MCA1301)

IT and CS Programmes in which this course is offered	Semester in which offered
Master of Computer Applications -(M.C.A)	Third Semester

Course Objectives – This Course will enable students to create mobile applications across the mobile operating systems - Android and iOS. Students understand how to implement OOPS concept to create efficient and Implementable applications.

Smartphones and mobile applications have grown in popularity in recent years. This trend is expected to continue, resulting in an increased demand for professionals who can create mobile applications. This course will help students to build core competencies in mobile application development with relevant skills and knowledge, practical experience, and career opportunities in a growing and dynamic industry. At the end of this course, students will be able to: describe Flutter architecture, widgets and their life cycle. practice the designing a well interface using Widgets and Components. perform CRUD Operations Using Local Database and Rest API's. use navigation between different pages and refresh screens with data.

Teaching and Examination Scheme –

Content Hours per Week			Course Credits		Examination Marks				
L	T	P	T	P	Theory Marks		Practical Marks		150
			3	2	E	M	I	EV	
3	0	4	5		60	40	30	20	

*E - External Examination, M- Mid Semester Examination, I- Internal Submission & Viva,
EV - External Viva

DETAILED COURSE CONTENTS -

Sr.	Topic	Teaching Hrs.	Weightage %
1	Introduction to Flutter <ul style="list-style-type: none">• Features of Flutter• Advantages of Flutter• Disadvantages of Flutter.• Flutter Installation in Windows• Flutter Installation in MacOS• Architecture of FlutterApplications• Creating Simple Application in Android Studio Introduction To OOPS Concept Using Dart <ul style="list-style-type: none">• Variables and Data types,• Decision Making and Loops,• Functions(Named parameters, Optional positional parameters, Optional parameter with default values, Anonymous functions),• Class, Constructor, Object,• Abstraction, Encapsulation, Inheritance,• Polymorphism(Compile-time polymorphism, Runtime polymorphism).	15	20
2	Flutter Architecture & Inbuilt Widgets <ul style="list-style-type: none">• Flutter Architecture,• Project Structure (Pubspec File, Assets, Lib, Android, iOS),• Run Window, Inbuilt Widgets,• Stateless Widget, Statefull Widget,• Single Child Widget (Scaffold, Safearea, Text, TextField, TextFormField, Button, Container, Card, Padding),• Multi Child Widget (Row, Column, Stack),• SnackBar, App Bar, etc.	10	30
3	Scrollable Widgets, Dialogs & State Management <ul style="list-style-type: none">• List View, Grid View,• Difference between Listview & Listview.builder,• Difference between Gridview & Gridview.builder,• PageView, Single Child Scroll View, Nested Scroll View,• Bottom Sheet, Drawer, BottomNavigationBar, Date Picker Dialog,• Alert Dialog, Mixin,• What is state management,• Categories of State Management (Ephemeral State, App State),• State management approaches, Provider, setState,• CheckboxGroup and RadioButtonGroup	10	20
4	Database Connectivity & Navigation <ul style="list-style-type: none">• MaterialPageRoute,• Navigator.push(), Navigator.pop(),• Return Data to Previous Page using pop() & push().then(),• Routing, Future, FutureBuilder,• Async Await, Shared Preferences,• Database in Flutter, Sqlite Database,• CRUD Operations.	15	30

***Continuous Evaluation:**

It consists of Assignments/Seminars/Presentations/Quizzes/Surprise Tests (Summative/MCQ) etc.

Revised Bloom's Taxonomy

Distribution of Theory Marks					
Remember	Understand	Apply	Analyze	Evaluate	Create
10	15	15	10	05	05

*Legends: R = Remembrance; U = Understanding; A = Application N :Analyze and E :Evaluate C : Create and Above Levels (Revised Bloom's Taxonomy)

SUGGESTED LEARNING RESOURCES**A. List of Books**

SR NO.	TITLE OF BOOKS	AUTHOR NAME	PUBLICATION
1	Beginning App Development with Flutter (TextBook) By	Rap Payne	Apress
2	Beginning Flutter: A Hands On Guide to App Development	Marco L. Napoli	Wrox Publication
3	Pragmatic Flutter (TextBook)	Priyanka Tyagi	CRC Press
4	Flutter Apprentice	Moore K.	Wrox Publication
5	Flutter Complete Reference	Alberto Miola	-

B. List of Software/Learning Websites

https://onlinecourses.swayam2.ac.in/nou21_ge41/preview

<https://www.coursera.org/specializations/android-app-development>

<https://www.geeksforgeeks.org/introduction-to-android-development>

<https://www.tutorialspoint.com/android/index.htm>

<https://onlinecourses.swayam2.ac.in/>

SUGGESTED LIST OF PRACTICAL/EXERCISES

Sr. No	Practical/ Exercises	Approx Hrs. Required
1.	Flutter Introduction and getting started with Flutter.	04
2.	Create a Hello World Application in Flutter.	04
3.	Write an application as ask the user for a number and determine whether the number is prime or not in Dart.	04
4.	Write a program in Dart that find the area of a circle using function.	04
5.	Write a Dart program to create a class Laptop with properties [id, name, ram] and create 3 objects of it and print all details.	04
6.	Create a simple Calculator application in Flutter.	04
7.	Create a simple Navigation App.	04
8.	Create forms in Flutter, Implementing Text input, drop-down lists, and more form elements in forms, Applying validation in form.	04
9.	Build an application for adding firebase to Flutter application.	04
10.	Code for Applying Flutter with JSON	04



**SWAMINARAYAN
UNIVERSITY**
THE RESERVOIR OF ALL KNOWLEDGE

FACULTY OF IT AND COMPUTER SCIENCE

COURSE CURRICULUM

Course Title: ARTIFICIAL INTELLIGENCE
(Code: MCA1302)

IT and CS Programmes in which this course is offered	Semester in which offered
Master of Computer Applications -(M.C.A)	Third Semester

Course Objectives – The objective of the course is to Introduce the necessary understanding of human intelligence and to explore the mechanisms that enables the intelligent thought and action. Understand and learn effective ways for representing knowledge, applying intelligent problem solving techniques & searching techniques.

Teaching and Examination Scheme –

Content Hours per Week			Course Credits		Examination Marks				150
L	T	P	T	P	Theory Marks		Practical Marks		
			2	2	E	M	I	EV	
2	0	4	2		60	40	30	20	

*E - External Examination, M- Mid Semester Examination, I- Internal Submission & Viva, EV - External Viva

DETAILED COURSE CONTENTS -

Sr.	Topic	Teaching Hrs.	Weightage %
1	Introduction <ul style="list-style-type: none"> The AI Problems, The Underlying Assumption, AI techniques, The Level of The Model, Criteria For Success Problems, State Space Search & Heuristic Search Techniques <ul style="list-style-type: none"> Defining The Problems As A State Space Search, Production Systems, Production Characteristics, Production System Characteristics and Issues in the Design of Search Programs, Generate-And-Test, Hill Climbing, Best-First Search, Problem Reduction, 		

	Constraint Satisfaction, Means-Ends Analysis.		
2	Knowledge Representation: <ul style="list-style-type: none"> Representations And Mappings, Approaches To Knowledge Representation, Representation Simple Facts In Logic, Representing Instance And Isa Relationships, Computable Functions and Predicates, Resolution, Procedural versus Declarative Knowledge, Logic Programming, Forward versus Backward Reasoning. Symbolic Reasoning Under Uncertainty: <ul style="list-style-type: none"> Introduction To Nonmonotonic Reasoning, Logics For Non-monotonic Reasoning. Probabilistic Reasoning: <ul style="list-style-type: none"> Probability And Bays' Theorem, Certainty Factors And Rule-Base Systems, Bayesian Networks, Dempster-Shafer Theory, Fuzzy Logic 	10	30
3	Game Playing: Overview, MiniMax Search Procedure, Alpha-Beta Cut-offs, Refinements, Iterative deepening. Planning: The Blocks World, Components Of a Planning System, Goal Stack Planning, Nonlinear Planning Using Constraint Posting, Hierarchical Planning, Reactive Systems Natural Language Processing: Introduction, Syntactic Processing, Semantic Analysis, Discourse And Pragmatic Processing, Spell Checking	10	20
4	Connectionist Models: <ul style="list-style-type: none"> Introduction: Hopfield Network, Learning In Neural Network, Application Of Neural Networks, Recurrent Networks, Distributed Representations, Connectionist AI And Symbolic AI. Expert Systems: <ul style="list-style-type: none"> Representing and Using Domain Knowledge, Expert System Shells, Explanation, Knowledge Acquisition. Genetic Algorithms: <ul style="list-style-type: none"> A Peek into the Biological World, Genetic Algorithms (GAs), Significance of the Genetic Operators, Termination Parameters. Introduction to Prolog: <ul style="list-style-type: none"> Introduction, Converting English to Prolog Facts and Rules, Goals, Prolog Terminology, Variables, Control Structures, Arithmetic Operators, Matching in Prolog, Backtracking, Cuts, Recursion, Lists. 	15	30

***Continuous Evaluation:**

It consists of Assignments/Seminars/Presentations/Quizzes/Surprise Tests (Summative/MCQ) etc.

Revised Bloom's Taxonomy

Distribution of Theory Marks					
Remember	Understand	Apply	Analyze	Evaluate	Create
10	15	15	10	05	05

*Legends: R = Remembrance; U = Understanding; A = Application N :Analyze and E :Evaluate C : Create and Above Levels (Revised Bloom's Taxonomy)

SUGGESTED LEARNING RESOURCES

A. List of Books

SR NO.	TITLE OF BOOKS	AUTHOR NAME	PUBLICATION
1	Artificial Intelligence	By Elaine Rich And Kevin Knight (2nd Edition)	Tata Mcgraw-Hill
2	Artificial Intelligence A Modern Approach	By Stuart Russel, Peter Norvig	PHI
3	Introduction to Prolog Programming	By Carl Townsend	-

B. List of Software/Learning Websites

<https://nptel.ac.in/courses/106/105/106105077/>

<http://www.journals.elsevier.com/artificial-intelligence/>

SUGGESTED LIST OF PRACTICAL/EXERCISES

Sr. No	Practical/ Exercises	Approx Hrs. Required
1.	Write a program to implement BFS	02
2.	Write a program to implement DFS	02
3.	Write a program to implement Single Player Game	02
4.	Write a program to implement 8 puzzle problem	02
5.	Write a program to solve N-Queens problem using Prolog.	06
6.	Write a Program to find the solution for travelling salesman Problem	04
7.	Write a program to implement Simulated Annealing Algorithm	02
8.	Write a program to find the solution for wampus world problem	04
9.	Write a program to implement Towers of Hanoi problem	04
10.	Write a program to implement A* Algorithm	04
11.	Write a program to implement Hill Climbing Algorithm	02
12.	Write a higher-order function count that counts the number of	02

	elements in a list that satisfy a given test. For instance: <code>count(lambda x: x>2, [1,2,3,4,5])</code> should return 3, as there are three elements in the list larger than 2. Solve this task without using any existing higher-order function.	
13.	Write a function that allows you to generate random problem instances for the knapsack program. This function should generate a list of items containing N items that each have a unique name, a random size in the range 1 5 and a random value in the range 1 10.	02
14.	Assume that you are organising a party for N people and have been given a list L of people who, for social reasons, should not sit at the same table. Furthermore, assume that you have C tables (that are infinitely large). Write a function <code>layout(N,C,L)</code> that can give a table placement (ie. a number from 0 : : C -1) for each guest such that there will be no social mishaps. For simplicity we assume that you have a unique number 0N-1 for each guest and that the list of restrictions is of the form [(X,Y), ...] denoting guests X, Y that are not allowed to sit together. Answer with a dictionary mapping each guest into a table assignment, if there are no possible layouts of the guests you should answer False.	02



**SWAMINARAYAN
UNIVERSITY**
THE RESERVOIR OF ALL KNOWLEDGE

FACULTY OF IT AND COMPUTER SCIENCE

COURSE CURRICULUM

Course Title: DATA SCIENCE WITH PYTHON
(Code: MCA1303)

IT and CS Programmes in which this course is offered	Semester in which offered
Master of Computer Applications -(M.C.A)	Third Semester

Course Objectives – This course typically covers fundamental to advanced concepts of Python programming, along with its application in data science

Teaching and Examination Scheme –

Content Hours per Week			Course Credits		Examination Marks			
L	T	P	T	P	Theory Marks		Practical Marks	
			2	2	E	M	I	EV
2	0	4	4		60	40	30	20
150								

*E - External Examination, M- Mid Semester Examination, I- Internal Submission & Viva, EV - External Viva

DETAILED COURSE CONTENTS -

Sr.	Topic	Teaching Hrs.	Weightage %
1	Overview of Python and Data Structures <ul style="list-style-type: none"> Basics of Python including data types, variables, expressions, objects and functions. Python data structures including String, Array, List, Tuple, Set, Dictionary and operations them. 	10	20
2	Data Science and Python <ul style="list-style-type: none"> Discovering the match between data science and python: Considering the emergence of data science, Outlining the core competencies of a data scientist, Linking data science, big data, and AI , Understanding the role of programming, Creating the Data Science Pipeline, Preparing the data, Performing exploratory data analysis, 	15	30

	Learning from data, Visualizing, <ul style="list-style-type: none"> Obtaining insights and data products Understanding Python's Role in Data Science: Introducing Python's Capabilities and Wonders: Why Python?, Grasping Python's Core Philosophy, Contributing to data science, Discovering present and future development goals, Working with Python, Getting a taste of the language, Understanding the need for indentation, Working at the command line or in the IDE 		
3	Getting Your Hands Dirty With Data <ul style="list-style-type: none"> Using the Jupyter Console, Interacting with screen text, Changing the window appearance, Getting Python help, Getting IPython help, Using magic functions, Discovering objects, Using Jupyter Notebook, Working with styles, Restarting the kernel, Restoring a checkpoint, Performing Multimedia and Graphic Integration, Embedding plots and other images, Loading examples from online sites, Obtaining online graphics and multimedia. 	15	20
4	Data Visualization <ul style="list-style-type: none"> Visualizing Information: Starting with a Graph, Defining the plot, Drawing multiple lines and plots, Saving your work to disk, Setting the Axis, Ticks, Grids, Getting the axes, Formatting the axes, Adding grids, Defining the Line Appearance, Working with line style, Using colors, Adding markers, Using Labels, Annotations, and Legends, Adding labels, Annotating the chart, Creating a legend. Visualizing the Data: Choosing the Right Graph, Showing parts of a whole with pie charts, Creating comparisons with bar charts, Showing distributions using histograms, Depicting groups using boxplots, Seeing data patterns using scatterplots, Creating Advanced Scatterplots, Depicting groups, Showing correlations, Plotting Time Series, Representing time on axes, Plotting trends over time, Plotting Geographical Data, Using an environment in Notebook, Getting the Basemap toolkit, Dealing with deprecated library issues, Using Basemap to plot geographic data, Visualizing Graphs, Developing undirected graphs, Developing directed graphs. 	10	30

***Continuous Evaluation:**

It consists of Assignments/Seminars/Presentations/Quizzes/Surprise Tests (Summative/MCQ) etc.

Revised Bloom's Taxonomy

Distribution of Theory Marks					
Remember	Understand	Apply	Analyze	Evaluate	Create
10	15	15	10	05	05

*Legends: R = Remembrance; U = Understanding; A = Application N :Analyze and E :Evaluate C : Create and Above Levels (Revised Bloom's Taxonomy)

SUGGESTED LEARNING RESOURCES

A. List of Books

SR NO.	TITLE OF BOOKS	AUTHOR NAME	PUBLICATION
1	Python for data science for dummies	John Paul Mueller, Luca Massaron	Wiley
2	Programming through Python	M. T. Savaliya, R. K. Maurya, G. M. Magar	STAREDU Solutions
3	Pandas for everyone :Python Data Analysis	Daniel Y. Chen	Pearson
4	Introducing Data Science: Big Data, Machine Learning, and More, Using Python Tools	Davy Cielen, Arno D.B. Meysman, Mohamed Ali	-
5	Applied Data Science with Python and Jupyter	Alex Galea	Packt

B. List of Software/Learning Websites

https://onlinecourses.swayam2.ac.in/nou21_ge41/preview

<https://www.coursera.org/specializations/android-app-development>

<https://www.geeksforgeeks.org/introduction-to-android-development>

<https://www.tutorialspoint.com/android/index.htm>

<https://onlinecourses.swayam2.ac.in/>

SUGGESTED LIST OF PRACTICAL/EXERCISES

Sr. No	Practical/ Exercises	Approx Hrs. Required
1.	Write a program to display sum of two complex numbers.	02
2.	Write a program to swap two numbers without taking a temporary variable.	02
3.	Write a program to create a byte type array, read, modify, and display the elements of the array	02
4.	Create a sequence of numbers using range datatype to display 1 to 30, with an increment of 2.	02
5.	Write a program to find out and display the common and the non common elements in the list using membership operators	02
6.	Create a program to display memory locations of two variables using id() function, and then use identity operators to compare whether two objects are same or not.	02

7.	Write a program to search an element in the list using for loop and also demonstrate the use of “else” with for loop.	02
8.	Write a python program to find the sum of even numbers using command line arguments.	02
9.	Write a python program that asks the user to enter a length in centimeters. If the user enters a negative length, the program should tell the user that the entry is invalid. Otherwise, the program should convert the length to inches and print out the result. (2.54 = 1 inch).	02
10.	Write a program to assert the user enters a number greater than zero.	02
11.	Write a program to create a Student class with name, age and marks as data members. Also create a method named display() to view the student details. Create an object to Student class and call the method using the object.	02
12.	Write a program to create Student class with a constructor having more than one parameter.	02
13.	Write a program to demonstrate the use of instance and class/static variables.	02
14.	Write a program to store data into instances using mutator methods and to retrieve data from the instances using accessor methods.	02
15.	Write a program to use class method to handle the common features of all the instance of Student class.	02
16.	Write a program to create one array from another array.	02
17.	Create a program to retrieve, display and update only a range of elements from an array using indexing and slicing in arrays.	02
18.	Write a program to understand various methods of array class mentioned: append, insert, remove, pop, index, tolist and count.	02
19.	Write a program to create a static method that counts the number of instances created for a class.	02
20.	Create a Bank class with two variables name and balance. Implement a constructor to initialize the variables. Also implement deposit and withdrawals using instance methods.	02



**SWAMINARAYAN
UNIVERSITY**
THE RESERVOIR OF ALL KNOWLEDGE

सर्व विद्या अधिपतये

FACULTY OF IT AND COMPUTER SCIENCE

COURSE CURRICULUM

Course Title: MACHINE LEARNING
(Code: MCA1304)

IT and CS Programmes in which this course is offered	Semester in which offered
Master of Computer Applications -(M.C.A)	Third Semester

Course Objectives – Basics of computer science including algorithms, data structure, Probability theory and Bayesian Concept Learning *

Teaching and Examination Scheme –

Content Hours per Week			Course Credits		Examination Marks				150
L	T	P	T	P	Theory Marks		Practical Marks		
			3	1	E	M	I	EV	
3	0	2	4		60	40	30	20	

*E - External Examination, M- Mid Semester Examination, I- Internal Submission & Viva, EV - External Viva

DETAILED COURSE CONTENTS -

Sr.	Topic	Teaching Hrs.	Weightage %
1	Introduction to Machine Learning: <ul style="list-style-type: none"> What is human learning? What is Machine Learning? Human learning versus machine learning, Types of machine learning, Applications of machine learning, Tools for machine learning. 	8	10
2	Preparing to Model , Feature Engineering: <ul style="list-style-type: none"> Machine Learning activities, Basic Types of data in Machine Learning, Structures of data, Data Quality and Remediation, Data Pre-Processing, Introduction to Feature Engineering, Feature Transformation, Feature Subset Selection 	10	20
3	Modelling and Evaluation: <ul style="list-style-type: none"> Selecting a Model, Training a Model, Model 	10	25

	Representation and Interpretability, Evaluating Performance of a Model, Improving Performance of a Model		
4	Supervised Learning: Classification and Regression: <ul style="list-style-type: none"> Example of Supervised Learning, Classification Model, Classification Learning Steps, Common Classification Algorithms Example of Regression, Common regression Algorithms 	10	25
5	Unsupervised Learning: <ul style="list-style-type: none"> Unsupervised Learning versus Supervised Learning, Applications of Unsupervised Learning, Clustering, Finding Pattern using Association Rules 	10	20

***Continuous Evaluation:** It consists of Assignments/Seminars/Presentations/Quizzes/Surprise Tests (Summative/MCQ) etc.

Revised Bloom's Taxonomy

Distribution of Theory Marks					
Remember	Understand	Apply	Analyze	Evaluate	Create
10	15	15	10	05	05

*Legends: R = Remembrance; U = Understanding; A = Application N :Analyze and E :Evaluate C : Create and Above Levels (Revised Bloom's Taxonomy)

SUGGESTED LEARNING RESOURCES

A. List of Books

SR NO.	TITLE OF BOOKS	AUTHOR NAME	PUBLICATION
1	Machine Learning	Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das	Pearson Education
2	Machine Learning	Tom M Mitchell	McGraw Hill
3	Machine Learning	Anuradha Srinivasaraghavan, Vincy Joseph	Wiley India
4	Machine Learning in Action	Peter Harrington	DreamTech
5	Real-World Machine Learning	Henrik Brink, Joseph Richards, Mark Fetherolf	DreamTech
6	Pattern Recognition and Machine Learning	Christopher Bishop	Springer
7	Data Mining: Concepts and Techniques	Jiawei Han and Michelline Kamber	Morgan Kaufmann

B. List of Software/Learning Websites

<https://www.simplilearn.com/>

<https://www.coursera.org/learn/practical-machine-learning>

SUGGESTED LIST OF PRACTICAL/EXERCISES

Sr. No	Practical/ Exercises	Approx Hrs. Required																																																																																																																								
1.	Write a python code to implement decision tree for below given dataset. Identify the root node and all subpart or children of node and draw the tree.	02																																																																																																																								
	<table><tr><th>Item no</th><th>Age</th><th>Income</th><th>Student</th><th>Credit</th><th>Rating</th><th>Buy</th><th>Computer</th></tr><tr><td>1</td><td>Youth</td><td>High</td><td>No</td><td>Fair</td><td>No</td><td>Youth</td><td>High</td></tr><tr><td>2</td><td>Youth</td><td>High</td><td>No</td><td>Excellent</td><td>No</td><td>Youth</td><td>High</td></tr><tr><td>3</td><td>Middle</td><td>High</td><td>No</td><td>Fair</td><td>Yes</td><td>Middle</td><td>High</td></tr><tr><td>4</td><td>Senior</td><td>Medium</td><td>No</td><td>Fair</td><td>Yes</td><td>Senior</td><td>Medium</td></tr><tr><td>5</td><td>Senior</td><td>Low</td><td>Yes</td><td>Fair</td><td>Yes</td><td>Senior</td><td>Low</td></tr><tr><td>6</td><td>Middle</td><td>Low</td><td>Yes</td><td>Excellent</td><td>No</td><td>Middle</td><td>Low</td></tr><tr><td>7</td><td>Senior</td><td>Low</td><td>Yes</td><td>Excellent</td><td>Yes</td><td>Senior</td><td>Low</td></tr><tr><td>8</td><td>Youth</td><td>Medium</td><td>No</td><td>Fair</td><td>No</td><td>Youth</td><td>Medium</td></tr><tr><td>9</td><td>Youth</td><td>Low</td><td>Yes</td><td>Fair</td><td>Yes</td><td>Youth</td><td>Low</td></tr><tr><td>10</td><td>Senior</td><td>Medium</td><td>Yes</td><td>Fair</td><td>Yes</td><td>Senior</td><td>Medium</td></tr><tr><td>11</td><td>Youth</td><td>Youth</td><td>Youth</td><td>Youth</td><td>Youth</td><td>Youth</td><td>Youth</td></tr><tr><td>12</td><td>Middle</td><td>Medium</td><td>No</td><td>Excellent</td><td>Yes</td><td>Middle</td><td>Medium</td></tr><tr><td>13</td><td>Middle</td><td>High</td><td>Yes</td><td>Fair</td><td>Yes</td><td>Middle</td><td>High</td></tr><tr><td>14</td><td>Senior</td><td>Medium</td><td>No</td><td>Excellent</td><td>No</td><td>Senior</td><td>Medium</td></tr></table>		Item no	Age	Income	Student	Credit	Rating	Buy	Computer	1	Youth	High	No	Fair	No	Youth	High	2	Youth	High	No	Excellent	No	Youth	High	3	Middle	High	No	Fair	Yes	Middle	High	4	Senior	Medium	No	Fair	Yes	Senior	Medium	5	Senior	Low	Yes	Fair	Yes	Senior	Low	6	Middle	Low	Yes	Excellent	No	Middle	Low	7	Senior	Low	Yes	Excellent	Yes	Senior	Low	8	Youth	Medium	No	Fair	No	Youth	Medium	9	Youth	Low	Yes	Fair	Yes	Youth	Low	10	Senior	Medium	Yes	Fair	Yes	Senior	Medium	11	Youth	Youth	Youth	Youth	Youth	Youth	Youth	12	Middle	Medium	No	Excellent	Yes	Middle	Medium	13	Middle	High	Yes	Fair	Yes	Middle	High	14	Senior	Medium	No	Excellent	No	Senior	Medium
	Item no		Age	Income	Student	Credit	Rating	Buy	Computer																																																																																																																	
	1		Youth	High	No	Fair	No	Youth	High																																																																																																																	
	2		Youth	High	No	Excellent	No	Youth	High																																																																																																																	
	3		Middle	High	No	Fair	Yes	Middle	High																																																																																																																	
	4		Senior	Medium	No	Fair	Yes	Senior	Medium																																																																																																																	
	5		Senior	Low	Yes	Fair	Yes	Senior	Low																																																																																																																	
	6		Middle	Low	Yes	Excellent	No	Middle	Low																																																																																																																	
	7		Senior	Low	Yes	Excellent	Yes	Senior	Low																																																																																																																	
	8		Youth	Medium	No	Fair	No	Youth	Medium																																																																																																																	
	9		Youth	Low	Yes	Fair	Yes	Youth	Low																																																																																																																	
	10		Senior	Medium	Yes	Fair	Yes	Senior	Medium																																																																																																																	
	11		Youth	Youth	Youth	Youth	Youth	Youth	Youth																																																																																																																	
	12		Middle	Medium	No	Excellent	Yes	Middle	Medium																																																																																																																	
13	Middle	High	Yes	Fair	Yes	Middle	High																																																																																																																			
14	Senior	Medium	No	Excellent	No	Senior	Medium																																																																																																																			
2.	Write a python code to implement K-nearest neighbourhood program for the given dataset.	02																																																																																																																								
	<table><tr><th>Item no</th><th>Temp</th><th>Humidity</th><th>Wind Speed</th><th>Play</th></tr><tr><td>1</td><td>85</td><td>85</td><td>12</td><td>No</td></tr><tr><td>2</td><td>80</td><td>90</td><td>9</td><td>No</td></tr><tr><td>3</td><td>83</td><td>86</td><td>4</td><td>Yes</td></tr><tr><td>4</td><td>70</td><td>96</td><td>3</td><td>Yes</td></tr><tr><td>5</td><td>68</td><td>80</td><td>5</td><td>Yes</td></tr><tr><td>6</td><td>65</td><td>70</td><td>20</td><td>No</td></tr><tr><td>7</td><td>64</td><td>65</td><td>2</td><td>Yes</td></tr></table>		Item no	Temp	Humidity	Wind Speed	Play	1	85	85	12	No	2	80	90	9	No	3	83	86	4	Yes	4	70	96	3	Yes	5	68	80	5	Yes	6	65	70	20	No	7	64	65	2	Yes																																																																																
	Item no		Temp	Humidity	Wind Speed	Play																																																																																																																				
	1		85	85	12	No																																																																																																																				
	2		80	90	9	No																																																																																																																				
	3		83	86	4	Yes																																																																																																																				
	4		70	96	3	Yes																																																																																																																				
	5		68	80	5	Yes																																																																																																																				
6	65	70	20	No																																																																																																																						
7	64	65	2	Yes																																																																																																																						

		8	72	95	12	No	
		9	69	70	5	Yes	
		10	75	80	2	Yes	
		11	75	70	3	Yes	
		12	72	90	4	Yes	
		13	81	75	5	Yes	
		14	71	91	15	No	
3.	Write a python code to implement Apriori algorithm, apply join and prune method and find frequent item set						02
		Sr No.	Item no	Name			
		1	T1	Bread, butter, milk, soda			
		2	T2	Coke, egg, milk			
		3	T3	Bread, butter, egg			
		4	T4	Break, coke, jam			
		5	T5	Bread, butter			
		6	T6	Potato chips, soda			
		7	T7	Coke, fruit, juice			
		8	T8	Bread, coke, milk			
		9	T9	Coke, soda, jam, milk			
		10	T10	Bread, butter, egg, milk, soda			
		11	T11	Bread, milk			
		12	T12	Bread, jam			
4.	Write a python code to apply Naive Bayesian and Logistic Regression algorithm to classify that whether a person can buy computer or not based on given test data:						02
	Item no	Age	Income	Student	Credit Rating	Buys-Computer	
	1	Youth	High	No	Fair	No	
	2	Youth	High	No	Excellent	No	
	3	Middle	High	No	Fair	Yes	
	4	Senior	Medium	No	Fair	Yes	
	5	Senior	Low	Yes	Fair	Yes	
	6	Middle	Low	Yes	Excellent	No	
	7	Senior	Low	Yes	Excellent	Yes	
	8	Youth	Medium	No	Fair	No	
	9	Youth	Low	Yes	Fair	Yes	
	10	Senior	Medium	Yes	Fair	Yes	
	11	Youth	Medium	Yes	Excellent	Yes	
	12	Middle	Medium	No	Excellent	Yes	
	13	Middle	High	Yes	Fair	Yes	
	14	Senior	Medium	No	Excellent	No	
	Test Data Age: Youth Income: LOW Student: No Credit Rating: Fair Buy Computer - ??						
5.	Build a prediction model using regression technique for (1) Boston house-prices (from sklearn.dataset import load_boston) (2) Diabetes (from sklearn.dataset import load_diabetes) datasets. Also, evaluate the model.						04
6.	Implement support vector machine approach to predictive modelling for (1) Boston house-prices (from sklearn.dataset import load_boston) (2) Diabetes						04

	(from sklearn.dataset import load_diabetes) datasets. Also, evaluate the model.	
7.	Implement unsupervised machine learning algorithm (Clustering – K Means) in python on Titanic dataset (https://www.openml.org/d/42438) to cluster data by removing the classlabel. Use elbow method to find the optimal number of clusters. Also, measure the quality of clustering via silhouette coefficient.	02
8.	Implement unsupervised machine learning algorithm (Clustering – K Means) in python on Breast Tumour dataset to cluster data (use Breast Tumour dataset) by removing the classlabel.	04
9.	Implement unsupervised machine learning algorithm (Clustering – Hierarchical) in python on Titanic dataset to cluster data (use Titanic dataset).	04
10.	Implement A priori algorithm in python to find rules which explain association between different products for given transactions at a retail store. (The data is available at https://drive.google.com/file/d/1NUXoptUIHY8z4KcFKpFA6sQN5KnWzk3p/view?usp=sharing)	04



**SWAMINARAYAN
UNIVERSITY**
THE RESERVOIR OF ALL KNOWLEDGE

FACULTY OF IT AND COMPUTER SCIENCE

COURSE CURRICULUM

Course Title: BIG DATA ANALYTICS
(Code: MCA1306)

IT and CS Programmes in which this course is offered	Semester in which offered
Master of Computer Applications -(M.C.A)	Third Semester

Course Objectives – Today's world is a data-driven world. Increasingly, the efficient operation of organizations across sectors relies on the effective use of vast amounts of data. Big data analytics helps us to examine these data to uncover hidden patterns, correlations, and other insights. It is a fast-growing field and skills in the area are some of the most in-demand today.

Teaching and Examination Scheme –

Content Hours per Week			Course Credits		Examination Marks				150
L	T	P	T	P	Theory Marks		Practical Marks		
			2	1	E	M	I	EV	
2	0	2	3		60	40	30	20	

*E - External Examination, M- Mid Semester Examination, I- Internal Submission & Viva, EV - External Viva

DETAILED COURSE CONTENTS -

Sr.	Topic	Teaching Hrs.	Weightage %
1	Introduction to Big Data: Introduction to Big Data, Big Data characteristics, Challenges of Conventional System, Types of Big Data, Intelligent data analysis, Traditional vs. Big Data business approach, Case Study of Big Data Solutions.	8	20
2	Hadoop: History of Hadoop, Hadoop Distributed File System: Physical organization of Compute Nodes, Components of Hadoop Analyzing the Data with Hadoop, Scaling Out, Hadoop	15	

	Streaming, Design of HDFS, Java interfaces to HDFS Basics, Developing a Map Reduce Application, How Map Reduce Works, Anatomy of a Map Reduce Job run, Failures, Job Scheduling, Shuffle and Sort, Task execution, Map Reduce Types and Formats, Map Reduce Features, Hadoop environment. Setting up a Hadoop Cluster, Cluster specification, Cluster Setup and Installation, Hadoop Configuration, security in Hadoop, Administering Hadoop, Monitoring-Maintenance, Hadoop benchmarks, Hadoop in the cloud		30
3	NoSQL: What is NoSQL? NoSQL business drivers; NoSQL case studies; NoSQL data architecture patterns: Key-value stores, Graph stores, Column family (Bigtable) stores, Document stores, Variations of NoSQL architectural patterns; Using NoSQL to manage big data: What is a big data NoSQL solution? Understanding the types of big data problems; Analyzing big data with a shared-nothing architecture; Choosing distribution models: master-slave versus peer-to-peer; Four ways that NoSQL systems handle big data problems	10	20
4	Mining Data Stream: Introduction to Streams Concepts, Stream Data Model and Architecture, Stream Computing, Sampling Data in a Stream, Filtering Streams, Counting Distinct Elements in a Stream, Estimating moments, Counting oneness in a Window, Decaying Window, Real time Analytics Platform (RTAP) applications, Case Studies, Real Time Sentiment Analysis, Stock Market Predictions. Using Graph Analytics for Big Data: Graph Analytics	10	20
5	Frameworks: Applications on Big Data Using Pig and Hive, Data processing operators in Pig, Hive services, HiveQL, Querying Data in Hive, fundamentals of HBase and ZooKeeper, IBM InfoSphere BigInsights and Streams. Spark: Introduction to Data Analysis with Spark, In-Memory Computing with Spark, Spark Basics, Interactive Spark with PySpark, Writing Spark Applications	5	10

***Continuous Evaluation:**

It consists of Assignments/Seminars/Presentations/Quizzes/Surprise Tests (Summative/MCQ) etc.

Revised Bloom's Taxonomy

Distribution of Theory Marks					
Remember	Understand	Apply	Analyze	Evaluate	Create
10	15	15	10	05	05

*Legends: R = Remembrance; U = Understanding; A = Application N :Analyze and E :Evaluate C : Create and Above Levels (Revised Bloom's Taxonomy)

SUGGESTED LEARNING RESOURCES

A. List of Books

SR NO.	TITLE OF BOOKS	AUTHOR NAME	PUBLICATION
1	Intelligent Data Analysis	Michael Berthold, David J. Hand	Springer

2	Taming The Big Data Tidal Wave: Finding Opportunities In Huge Data Streams With Advanced Analytics	Bill Franks	Wiley
3	Professional Hadoop Solutions	Boris lublinsky, Kevin t. Smith, Alexey Yakubovich	Wiley.
4	Understanding Big data	Chris Eaton and Paul C. Zikopoulos	TMH
5	Big Data and Analytics	Seema Acharya, Subhashini Chhellappan	Wiley,
6	MongoDB in Action	Kyle Banker, Piter Bakkum , Shaun Verch	DreamTech,

B. List of Software/Learning Websites

<http://in.reuters.com/tools/rss>

<http://www.altova.com/xmlspy.html>

<https://www.w3.org/RDF/>

SUGGESTED LIST OF PRACTICAL/EXERCISES

Sr. No	Practical/ Exercises	Approx Hrs. Required
1.	<p>Case Study:</p> <p>Stage 1: Selection of case study topics and formation of small working groups of 2/3 students per group. Students engage with the cases, read through background material provided in the session and work through an initial set of questions to deepen the understanding of the case. Sample applications and data will be provided to help students familiarize themselves with the cases and available (big) data.</p> <p>Stage 2: The groups are given a specific task relevant to the case in question and are expected to develop a corresponding big data concept using the knowledge gained in the course and the parameters set by the case study scenario. A set of questions that help guide through the scenarios will be provided.</p> <p>Stage 3: Each group prepares a short 2 – 5 page report on their results and a 10 min oral presentation of their big data concept.</p>	06
2.	To draw and explain Hadoop Architecture and Ecosystem with the help of a case study using WorkCount example. To define and install Hadoop.	02
3.	To understand the overall programming architecture using Map	02

	Reduce API.	
4.	To implement the following file management tasks in Hadoop System (HDFS): Adding files and directories, Retrieving files, Deleting files.	02
5.	Store the basic information about students such as roll no, name, date of birth ,and address of student using various collection types such as List, Set and Map.	02
6.	Basic CRUD operations in MongoDB.	02
7.	To perform NoSQL database using MongoDB to create, update and insert.	02
8.	Retrieve various types of documents from students collection.	02
9.	To find documents from Students collection.	02
10.	To run a basic Word Count MapReduce program to understand MapReduce	02
11.	Paradigm: To count words in a given file, To view the output file, and To calculate	02



**SWAMINARAYAN
UNIVERSITY**
THE RESERVOIR OF ALL KNOWLEDGE

FACULTY OF IT AND COMPUTER SCIENCE

COURSE CURRICULUM

Course Title: MINOR PROJECT
(Code: MCA1307)

IT and CS Programmes in which this course is offered	Semester in which offered
Master of Computer Applications -(M.C.A)	Third Semester

Course Objectives - New technologies are replacing older one very quickly whose effects can be seen in our society. Internship Project is a opportunity for the students to get flavor of such emerging technology and familiar with industry environment to identify scope as well as focus of their career development path. The main objective of the internship is to get hands-on practice. In turn it will help students to lay down a path for their further professional career by observing, understanding, adopting mechanism of working in the industry. Students will also develop various types of skills by pursuing the internship project programme.

The duration of internship will be six weeks. It will be after completion of 3rd Semester and in the Semester 4th. Any options from following can be chosen by the students:

1. Internship in the industry - The student is suggested to select branch-specific training in the IT or relevant Industry where they learn the skills. The student shall produce a joining letter in the beginning and relieving letter after the completion of the internship. It is desirable to do the internship offline. In the case of the online conduction mode, the student has to submit supporting evidence in the form of a screenshot and activity log report of the conducted session along with other necessary documents.

2. Project – If student is not able to carry internship in the industry, student may develop a project on topic related to respective computer engineering branch under faculty guidance. Additionally, Institute/ department may arrange technical sessions/ workshops from industry experts to give exposure to students about recent technologies and tools. It can be fabrication / experimental results/ simulations/ Application development / Design and / or Analysis of System(s) etc. related to the emerging technologies of the Computer/IT.

Students needs to report at institute every 15 days about progress to internal guide in suggested reporting format which is given in syllabus and represent his/her work carried out for monitoring and evaluation purpose.

COMPETENCY

The purpose of this course is to help the student to attain the following industry identified competency through various teaching-learning experiences:

Develop effective programming skills, problem-solving abilities, and technical knowledge to solve challenge innovatively while gaining hands-on experience for professional development opportunities.

COURSE OUTCOMES (COs)

The practical exercises, the underpinning knowledge, and the relevant soft skills associated with this competency are to be developed in the student to display the following COs:

The student will develop underpinning knowledge, adequate programming skills of competency for implementing various applications using tools and technologies to attain the following course outcomes.

- a) Apply acquired knowledge to solve domain related societal problems.
- b) Develop effective communication within academic and industrial environments.
- c) Utilize industry-standard tools as well as technologies to design, develop, and test the systems.
- d) Apply problem-solving skills to resolve technical issues encountered in IT Industry.
- e) Develop life-long learning skills for a successful professional career.

Teaching and Examination Scheme –

Content Hours per Week			Course Credits		Examination Marks			
L	T	P	T	P	Theory Marks		Practical Marks	
0	0	4	0	4	E	M	I	EV
0	0	4	2		0	0	60	40

*E - External Examination, M- Mid Semester Examination, I- Internal Submission & Viva, EV - External Viva

List of Documents to be prepared for Submission:

1. Student has to submit Work Report Sheet duly signed by industry resource person and internal guide fortnightly (suggested format given in syllabus).
2. Student Attendance Sheet Report (suggested format given in syllabus) duly signed by industry resource person. In the case of the online internship, student has to take screenshot of daily attended session in softcopy and submit to institute resource person by email for every 15 days.
3. Detailed report of said work approved and duly signed by the institute and/or industry resource person.
4. A softcopy of Presentation approved by the institute and/or industry resource person.
5. A Poster showing overall work carries out during the Internship project approved by institute and/or industry resource person

Note: Faculty should counsel as well as inform students in advance about the internship or project. Student needs to finalize training from industry or project at institute during 4th semester and report at institute.