

BHAKTA KAVI NARSINH MEHTA UNIVERSITY - JUNAGADH

(Gujarat) INDIA



**CURRICULUM FOR
M. Sc. (IT & CA)**

(Master of Science in Information Technology & Computer Application)
(Semester - 1)

Effective From June – 2018

BHAKTA KAVI NARSINH MEHTA UNIVERSITY**Master of Science (Information Technology & Computer Application) – Sem – 1**

[2 Years– Four Semester Full Time Program]

Ordinance, Regulations and Examination Scheme:**Ordinance:**

- O. M.Sc.(IT & CA) – 1** Candidate seeking admission to the Master of Science (Information Technology & Computer Application) must have a Bachelor's degree of minimum three years duration with 48% or more in the discipline.
- O. M.Sc.(IT & CA) – 2** BCA / B. Sc. (Computer / IT) / B. Com. (With Computer Science) / B. E. (Computer / IT) / B. Pharm. / B. Arch. / PGDCA / PGDACA
- O. M.Sc.(IT & CA) – 3** The duration of the course is full time two academic years. The Examination for the Master of Science (Information Technology & Computer Application) course will be conducted under the semester system. For this purpose the academic year is divided into two semesters. No candidate will be allowed to join any other course simultaneously.
- O. M.Sc.(IT & CA) – 4** Candidate who have passed an equivalent examination from any other university or examining body and is seeking admission to the M.Sc. (IT & CA) programme shall not be admitted without producing the eligibility certificate from the Bhakta Kavi Narshinh Mehta University – Junagadh.
- O. M.Sc.(IT & CA) – 5** No candidate will be admitted to any semester examination for the Master of Science (Information Technology & Computer Application) unless it is certified by the Head of the Department/ Director of institute.
“That candidate has attended the course of study to the satisfaction of the Head of Department/Director of institute)
- O. M.Sc.(IT & CA) – 6** Candidate desirous of appearing at any semester examination of the M. Sc. (IT & CA) programme must forward their application in the prescribed form to the Controller of Examination through Head of Department / Director of Institute on or before the date prescribed.
- O. M.Sc.(IT & CA) – 7** No candidate will be permitted to reappear at any semester examination, which he/she has already passed.
- O. M.Sc.(IT & CA) – 8** To pass the whole M.Sc. (IT & CA) examination, candidate must clear all the four semester examinations within a period of Four years from the date of his/her registration, otherwise candidate has to register him/her self again as a fresh candidate and keep attendance and appear and pass all the four semester examinations.
- O. M.Sc.(IT & CA) – 9** There shall be an examination at the end of each four semesters to be known as First semester examination, Second semester examination respectively, at which a student shall appear in the portion of papers practical and Project viva-voce if any, for which he has kept the semester in accordance with the regulations in this behalf.

A candidate whose term is not granted for whatsoever reason shall be required to keep attendance for that semester of terms when the relevant papers are actually taught at the institute.

O. M.Sc.(IT & CA) – 10 A candidate will be permitted to go to the next semester, irrespective he/she is failing in any number of subjects.

O. M.Sc.(IT & CA) – 11 No candidate will be allowed to reappear in examination of any subject which he/she has already passed.

Regulations & Examination Schemes :

R. M.Sc.(IT & CA) – 1 **The standard of passing the M.Sc. (IT & CA) degree examination will be as under:**

(1) To pass any semester examination of the M.Sc. (IT & CA) degree, a candidate must obtain at least 40% marks in the university examination separately in each course of theory and practical.

(2) Class will be awarded based on Earned Grade Point, SGPA and CGPA as per rules of University.

R. M.Sc.(IT & CA) – 2 **Marks and credit hours of each course**

Marks of Internal examination, university examination and credit hours will be as under:

(1) Total marks of each theory course are 100 (university examination of 70 marks + internal examination of 30 marks).

(2) Marks of each unit in the course are equal (i.e. 14 Marks). Total marks of each course are $14 \times 5 = 70$ for university examination.

(3) Credit hours (lectures) for each unit in the course are equal (i.e. 14 hours). Total credit hours (lectures) of each course are $14 \times 5 = 70$.

(4) Total marks of each practical and project-viva course are 100. No internal examination of marks in practical and project-viva courses

R. M.Sc.(IT & CA) – 3 **Structure of Question Paper**

Question Paper contains 5 questions (each of 14 marks). Every question will be asked from corresponding unit as specified in the syllabus of each course. (i.e. Question-1 from Unit No. 1 and remaining questions from their corresponding units)

Every question is divided in Three parts like (a), (b), (c).

Part (a) contains Three objective type questions (not MCQ) like definition, reason, answer in one line, answer in one word etc., each of one marks and no internal option.

Part (b) contains Four questions each of three marks and student will attempt any Two out of four.

Part (c) contains two questions each of five marks and student will attempt any one out of two.

BHAKTA KAVI NARSINH MEHTA UNIVERSITY**Master of Science (Information Technology & Computer Application) – Sem – 1**

[2 Years– Four Semester Full Time Program]

Code	Course	No. Of Lectures/ Lab (Per Week)	Credit
CS - 01	APPLICATION DEVELOPMENT USING ADVANCE JAVA	5	5
CS - 02	ADVANCE WEB DEVELOPMENT IN Laravel	5	5
CS - 03	NoSQL DATABASE: MongoDB	5	5
CS - 04	PRACTICAL - 1 (BASED ON CS-01)	5	5
CS - 05	PRACTICAL - 2 (BASED ON CS-02 and CS-03)	5	5
CS - 06	PROJECT DEVELOPMENT (In House)	5	5
Total Credits of Semester - 1			30

CS - 01: APPLICATION DEVELOPMENT USING ADVANCE JAVA

Objective:

- Learn how to download, setup and configure the Spring Framework Explore the Spring Container and Modules Understand dependency injection Learn aspect oriented programming and how it is used to provide cross cutting concerns
- Understand how spring deals with transaction management and ORM Hibernate: Inheritance mapping collection mapping. Understand the HQL.

Pre-Requisites:

- Students must have strong background of Java programming knowledge and exposure to J2EE technology.

Unit: 1

Basics of JDBC, JSP And MVC

- Introduction to JDBC
- JDBC Architecture
- Connecting with Database
- Introduction to JSP
- JSP with JDBC
- Introduction to MVC

Basics of Spring, Spring with IDE And IOC container

- What is Spring
- Spring Modules
- Spring Application
- Spring in Myeclipse
- Spring in Eclipse

Dependency Injection

- Constructor Injection
- CI Dependent Object
- CI with collection
- CI with Map
- CI Inheriting Bean
- Setter Injection
- SI Dependent Object
- SI with Collection
- SI with Map
- CI vs SI
- Autowiring
- Factory Method

Unit: 2

Spring JDBC

- JdbcTemplate Example
- PreparedStatement
- ResultSetExtractor
- RowMapper
- NamedParameter
- SimpleJdbcTemplate

Spring with Hibernate And SpEL

- Spring with JPA
- SpEL Examples
- Operators in SpEL
- variable in SpEL

Spring 3 MVC and Remoting with Spring

- Spring with RMI
- Http Invoker
- Hessian
- Burlap

Unit: 3**OXM Frameworks, Spring Java Mail And Web Integration**

- Spring with JAXB
- Spring with Xstream
- Spring with Castor
- Spring with Struts2
- Login and Logout Application

Basics of Hibernate And Hibernate with IDE

- Hibernate Introduction
- Hibernate Architecture
- Understanding First Hibernate application
- Hibernate in Eclipse
- Hibernate in MyEclipse

Hibernate Application And Hibernate Logging

- Hibernate with annotation
- Hibernate Web application
- Hibernate Generator classes
- Hibernate Dialects

Unit: 4**Inheritance Mapping**

- Table Per Hierarchy
- Table Per Hierarchy using Annotation
- Table Per Concrete
- Table Per Concreteusing Annotation
- Table Per Subclass
- Table Per Subclass using Annotation

Collection Mapping

- Mapping List
- One-to-many by List using XML
- One To Many by List using Annotation
- Mapping Bag
- One-to-many by Bag
- Mapping Set
- One-to-many by Set
- Mapping Map
- Many-to-many by Map

- Bidirectional
- Lazy Collection

Unit: 5

Component Mapping, Association Mapping, Transaction Management, HQL & HCQL

- One-to-one using Primary Key
- One-to-one using Foreign Key

Named Query, Hibernate Caching and Integration

- First Level Cache
- Second Level Cache
- Hibernate and Struts
- Hibernate and Spring

References Books

No.	Name	Author / Publication
1	Spring and Hibernate	Santosh Kumar K. - Tata McGraw-Hill Publishing
2	Spring persistence with Hibernate	Paul Tepper Fisher and Brian D. Murphy - Apress
3	Spring 4 and Hibernate 4: Agile Java Design and Development	McGraw-Hill Education, 2015
4	Pro Spring	Chris Schaefer, Clarence Ho, and Rob Harrop Apress

Web site References :

- <https://www.javatpoint.com/java-tutorial>

CS - 02: Advance Web Development in Laravel

Objective:

- Student should know OOP in PHP
- Student should be able to implement Laravel framework
- Student should be able to design and code responsive website
- Student should be able to meet current modern market requirement and create fruitful products

Pre-Requisites:

- Strong background and Knowledge of HTML, CSS, JavaScript and PHP is mandatory.

Unit: 1

Object Oriented Programming in PHP

The Basics of Php, Properties, Class Constants, Autoloading Classes, Constructors and Destructors, Visibility, Object Inheritance, Scope Resolution Operator (::), Static Keyword, Object Interfaces, Anonymous classes, Overloading, Object Iteration, Magic Methods, Final Keyword, Object Cloning, Comparing Objects, Type Hinting, Late Static Bindings, Objects and references

Bootstrap Basics

Introduction: File Structure, Basic HTML Template, Global Styles, Default Grid System, Basic Grid HTML, Offsetting Columns, Nesting Columns, Fluid Grid System, Container Layouts, Responsive Design, What Is Responsive Design?

Implementation: Typography, Code, Tables, Forms, Buttons, Images, Icons, Glyphicons, Dropdown Menus, Button Groups, Button with Dropdowns, Navigations, Navbar, Breadcrumb, Pagination, label, badges, Typographic elements, thumbnails, alerts, progress bar, wells

Unit: 2

Introduction to Laravel

What is Laravel, features, MVC architecture, structure of Laravel application (Laravel directory structure),

Installation

Basic requirements for Laravel, Using Laravel Installer, Using Composer, To create project using composer , how does Composer work? Installation, Linux & Windows, Finding and installing new packages

Configuration

Introduction, Environment configuration, Protecting sensitive configuration, Maintenance mode, database configuration (setting database connection parameter for laravel and artisan)

Unit: 3

Basic to laravel

create namespace , create controller , how to use Request Response , Cookies ,session, Define view , redirection

Artisan

create only model, create model with migration, create model with migration & controller, create controller , Artisan Command Line Tool, database creation, artisan migration, migration structure, creation migration, Database seeding

Routing in Laravel

Basic Routing, Route Parameters, Route Filters, Named Routes, Route Groups, Sub-Domain Routing, Route Prefixing, Route Model Binding, check used routes in whole project ,Throwing 404 Errors, Routing to Controllers

Better performance & Maintenance Mode

remove cache, remove compiled blade views , send project in Maintenance model, remove Maintenance model

Unit: 4

Basic to forms

What is the use of forms , installation of forms , use the following controls (Opening a Form , Generating a Label Element,Generating a Text Input,Specifying a Default Value,Generating a Password Input,Generating a File Input,Generating a Checkbox Or Radio Input,Generating a Checkbox Or Radio Input That Is Checked,Generating a Drop-Down List,Generating A Submit Button) , Localization feature

Blade Template

Template inheritance, Master layout, Extending the master layout, display variables, Blade conditional statements, Blade Loops, Executing PHP functions in blade

SQL Interaction

Introduction, Running Raw SQL Queries, Database Transactions

Unit: 5

Eloquent ORM

Eloquent ORM Models: Naming conventions, table name & primary keys, timestamps

Basic Operations: Create, Retrieve, Update, Delete Using Models, displaying data from models in views.

Validation

Defining The Routes, Creating The Controller, Writing The Validation Logic, Displaying The Validation Errors, Array validations, creating new validators, Error messages & custom errors

Available Validators: Accepted, After (Date), Alpha, Alpha Dash, Alpha Numeric, Array, Before (Date), Between, Boolean, Date, Date Format, Different, Digits, Digits Between, E-Mail, Exists (Database), Image (File), In, Integer, Max, Min, Not In, Numeric, Regular Expression, Required, String Custom validation Rules.

References

No.	Name	Author / Publication / Links
1	Online Laravel 5.2 Documentation	https://laravel.com/docs/5.2
2	Laravel 5 Essentials	Martin Bean, Packet Publishing, ISBN 978-1-78528-301-7
3	Bootstrap	Jake Spurlock, O'reilly, ISBN: 978-1-449-34391-0

Web site References :

- <https://laravel.com/docs/4.2/introduction>
- <https://www.tutorialspoint.com/laravel/index.htm>

CS - 03: NoSQL DATABASE: MongoDB

Objective:

- To develop proficiency in the specification, representation and various other types in MongoDB using PHP.
- To be able to perform various Analytical as well as to increase the programming skills in PHP using MongoDB.
- To get a good understanding regarding various styles in Programming.
- To develop a good base for No-SQL queries.

Pre-Requisites:

- Knowledge of PHP is mandatory.

Unit: 1

Introduction to NoSQL Database

- Define NoSQL, its characteristics and history, and the primary benefits for using NoSQL databases.
- Define the major types of NoSQL databases including a primary use case and advantages / disadvantages of each type.

Introduction to MongoDB

- MongoDB concepts - Databases, collections, and documents
- Downloading Installing and running MongoDB, Installing PHP
- Driver for MongoDB on various OS Platforms
- The Data Model and Working with Data

Unit: 2

Learning MongoDB by implementing web Application

- Inserting documents in MongoDB, Querying documents in collection.
- Doing advance queries in MongoDB, Updating documents MongoDB,
- Deleting documents in MongoDB, Managing relationships between documents

Using MongoDB with relational Databases

- MongoDB and RDBMS together
- Defining the relational model

Unit: 3

Session Management

- Understanding HTTP sessions.
- Understanding PHP native session handling,
- Implementing session handling with MongoDB.
- Putting Session Manager.
- Building user authentication module, creating login, logout and user profile.

Unit: 4

Aggregation Queries

- Generating Sample Data.
- Understanding MapReduce,
- Performing MapReduce in MongoDB and PHP, Aggregation using group()
- Listing distinct values for field
- counting documents with count()

Web Analytics using MongoDB

- Logging with MongoDB,
- Extracting analytics data with MapReduce
- Real-time analytics using MongoDB

Unit: 5

Handling Files with GridFS

- What is Grid?
- Storing files in GridFS
- Serving files from GridFS
- Reading files in chunks

Database Management

- Database Administration
 - Optimization
 - Replication
 - Sharding

References book

No.	Name	Author / Publication / Links
1	MongoDB the definitive guide	O'Reilly Kristina Chodorow & Michal Dirolf
2	MongoDB in Action	Kyle Banker Manning Sheltar Island
3	The definitive guide to MongoDB	NoSQL Database for cloud and desktop computing- Apress- Eelco Plugge, Peter membrey and Tim Hawkins
4	PHP and MongoDB Web Development Beginers guide	Rubayeet Islam - Open Source

CS – 04	PRACTICAL – 1 (Based on CS - 01) Application Development Using Advance JAVA	100 Marks
CS – 05	PRACTICAL – 2 (Based on CS – 02 & CS – 03) Advance Web Development in Laravel NoSQL Database : MongoDB	100 Marks
Note : ▪ Each session is of 3 hours for the purpose of practical Examination. ▪ Practical examination may be arranged before or after theory exam		
CS – 06	PROJECT	100 Marks
	Project must be developed in the computer laboratory of concern institute under the supervision of faculties of concern institute on any subject of previous semester or current semester. (At the time of Project-Viva examination student must show Project Report (in hard copy) along with all the Workouts in workbook, implementation of project in SDLC, Documentation, Program codes and project in running mode) _____ Note : ▪ Project must be submitted before two week of commencement of theory exam. ▪ Project viva examination may be arranged before or after theory exam. ▪ During the project viva examination project must be run.	