UNIVERSITY OF CALICUT

Abstract

General and Academic - Faculty of Science - Scheme and syllabus of Master Of Vocation (M.Voc) In Software Development programme as per the Regulations for the Post Graduate Programmes under Vocational Studies (Master of Vocation-M.Voc) 2019 - Approved - Subject to ratification by Academic Council - Orders issued.

G & A - IV - J

U.O.No. 16362/2021/Admn Dated, Calicut University.P.O, 12.11.2021

Read:-1. U.O.No.10272/2019/Admn, dated 01.08.2019

2. Item No.2 in the minutes of the Board of Studies in Computer Science &Application PG, held on 11.10.2021
3. Remarks from the Dean, Faculty of Science, dated 28.10.2021
4. Orders of the Vice Chancellor in the file of even no., dated on 30.10.2021

ORDER

1. Regulations for the Post Graduate Programmes under Vocational Studies (Master of Vocation-M.Voc) 2019, with effect from 2019 Admissions, has been implemented in this University, vide paper read (1) above.
2. The meeting of Board of Studies in Computer Science and Application (PG) held on 11.10.2021, approved the scheme and syllabus of Master Of Vocation (M.Voc) in Software Development programme, vide paper read (2) above.
3. The Dean, Faculty of Science recommended to approve the scheme and syllabus of Master Of Vocation (M. Voc) In Software Development programme w.e.f 2020 admissions, vide paper read (3) above.
4. Considering the urgency, the Vice Chancellor has accorded sanction to implement the Scheme and Syllabus of Master Of Vocation (M. Voc) In Software Development programme in accordance with the Regulations for the Post Graduate Programmes under Vocational Studies (Master of Vocation-M.Voc) 2019, in the University with effect from 2020 Admissions, subject to ratification by the Academic Council.
5. Sanction is therefore accorded for implementing the Scheme and Syllabus of Master Of Vocation (M. Voc) In Software Development programme in accordance with the Regulations for the Post Graduate Programmes under Vocational Studies (Master of Vocation-M.Voc) 2019, in the University with effect from 2020 Admissions, subject to ratification by the Academic Council.
6. Orders are issued accordingly. (Syllabus appended).

Arsad M

Assistant Registrar

To

Principals of affiliated colleges offering M.Voc Software Development.

Copy to: PS to VC/PA to PVC/ PA to R/PA to CE/JCE I/JCE V/JCE VII/SF/DF/FC

Forwarded / By Order

Section Officer

UNIVERSITY OF CALICUT

THENHIPALAM, CALICUT UNIVERSITY P.O -673635

DEGREE OF

MASTER OF VOCATION (M. Voc)

IN

SOFTWARE DEVELOPMENT

(AS PER THE GUIDELINES OF UGC)

UNDER THE

FACULTY OF SCIENCE

SYLLABUS

(CHOICE BASED CREDIT & SEMESTER SYSTEM- CBCSS)

(FOR THE STUDENTS ADMITTED FROM THE ACADEMIC YEAR 2020 – 21 ONWARDS)

BOARD OF STUDIES IN COMPUTER SCIENCE (PG)

THENHIPALAM, CALICUT UNIVERSITY P.O

KERALA, 673 635, INDIA

AUGUST, 2020

CONTENTS

REGULATIONS

FOR THE DEGREE OF M.Voc SOFTWARE DEVELOPMENT

Effective From the Academic Year 2020 – 21

1. TITLE OF THE PROGRAMME :

This programme shall be called **Master of Vocation in Software Development** as per the guidelines of UGC.

2. PROGRAMME OBJECTIVES

M.Voc Software Development is a vocational Master's Degree Programme designed for highly enthusiastic graduates with a good honours degree in any B.Voc or other degree having the same discipline.

This programme is targeted to those graduates who are interested in advancing their existing degree to gain expert knowledge of software development in the latest programming languages including data science and those who wish to enter the software industry. It also aims to mould expert teaching professionals in the vocational software discipline and thereby help them to pursue their research studies.

This vocational training program deals with software development and its allied areas especially in data analytics which is the emerging trend of current scenario. It is an advanced level software development training program.

The Software industry is becoming increasingly complex, requiring higher calibre skills than ever before. However, the future prospects of the industry are not yet armed with those analytical, technical and digital skills required to propel the industry forward. Post Graduation in Vocational degree is in a unique position to help bridge the skills gap for new joiners and help the industry up skill their existing workforce.

Sufficient industrial partnerships and the various levels of internships and also the live projects provided by the industries make the students work ready at each exit point.

3. ELIGIBILITY CRITERIA FOR ADMISSION.

Selection of the students shall be done based on overall CGPA / Total marks. The indexing rules and Index Calculation are given below:

For CCSS (0-4 grade point system)/ CUCBCSS (0-6 grade point system) candidates

Achievements used for SIS	Rule for SIS	Remarks
CG1 = % of Overall CGPA	$\frac{CG1 \times 1000}{k}$	k=100

For Candidates under Mark System

Achievements used for SIS	Rule for SIS	Remarks
M1(Total Marks scored)	$\frac{M1 \times 1000}{Max1}$	Max1= Maximum of Total Marks

- 20% weightage should be given to students who completed B.Voc Software Development.
- Basic eligibility for M.Voc Software Development is with minimum 50% overall CGPA or 50% marks in the following:

1. BVoc Programmes as listed below:

• B.Voc Software Development • BVoc Software Technology • B.Voc Data Science • B.Voc Information Technology • BVoc Web Technology • BVoc Software Development and System Administration • BVoc Mathematics and Artificial Intelligence • BVoc Artificial Intelligence and Data Analytics	• BVoc Artificial Intelligence and Robotics • BVoc Website Designing & Mobile App Development • BVoc Mobile Application Development • BVoc Robotics and Artificial Intelligence • BVoc Data Analytics and Machine Learning • BVoc Data Science and Analytics • B.Voc Data Analytics and Software Development • BVoc Cyber Security

2. B.Sc Computer Science / B.Sc IT / BSc Cyber Security / BCA / BSc Data Science
3. B.Tech Computer Science / B.Tech IT/ B.Tech Artificial Intelligence and Data Science
4. Any B.Sc Degree with Maths as subsidiary

- OBC/OEC candidates are eligible to relaxation upto 5%. SC/ST candidates need only to get a pass.

- Criteria for selection and method of admission to seats for M.Voc Software Development degree programme conducted by university Centres / Colleges affiliated to University of Calicut shall be governed by the rules/regulations framed by the University from time to time. In all matters related to selection and admission, the decisions of the University shall be final.

4. DURATION OF THE PROGRAMME

1. The duration of M.Voc Software Development Programme is Two years with 4 semesters. To those students who have completed their programme without multiple entry and exit, the minimum period for completion of the programme is ***two years*** and the maximum period permissible for completing the programme is fixed as **4 years.**

2. In the case of multiple entry and exit, students who successfully complete first two semesters can discontinue the programme, if they wish and can get a Post-graduate Diploma in Software Development (at NSQF Level 8). Those who continue the course further and finish four semesters successfully will be eligible for M.Voc. Software Development Degree ie (at NSQF Level 9). Students who discontinue after second semester with P.G. Diploma in Software Development can come back and opt for a lateral entry to third semester, later if they wish to do so, and can finish their M.Voc Degree in Software Development.

3. In multiple entry, the rejoining to the third semester shall be done by the students within **three years** from the date of their exit from the second semester. In such a case the maximum period for completion of the programme will be **5 years.**

4. The duration of each semester shall be 90 working days, inclusive of examinations, spread over five months. Odd semesters shall be held from June to October and even semesters from November to March subject the academic calendar of the University.

5. LANGUAGE OF EXAMINATIONS

- The language of writing the examination shall be English only.

6. Intake of Students

- As per UGC guidelines the intake of M.Voc programme is 50. For M.Voc Software Development is it recommended upto 30.

7. Fee Structure

- The course fee and examination fee for the first two years will be decided by the University.
- If the programme is conducted under Aided Stream, the fee structure is equivalent to M.Sc Computer Science (Aided Stream) and if it is conducted under self financing mode, the fee structure is equivalent to MSc Computer Science Self Financing Stream
- The college can deposit, PTA fund, special fees, university fees, sports fee etc according to the norms provided by the university and collect Caution at the time of admission.

8. Bridge Course

The students with any B.Sc Degree with maths as subsidiary background may require a bridge course named 'Programming Methodology' that prepares them for programming and algorithms. The purpose of this bridge course is only to fill the gaps in knowledge that a student may have while pursuing this master level vocational degree programme, M.Voc Software Development. The maximum duration of the bridge course is 20 hours.

The bridge course has zero credit and therefore, it will not be counted for evaluating the overall SGPA & CGPA. The College / Department shall conduct examination. Students have to obtain only minimum pass requirements in this Bridge Course.

Contents :

Programme Methodology: Introduction - Types or Concepts of programme methodologies (Procedural, OOP, Functional, Logical, Top-down, bottom-up) – Understanding the problem (Software development process /Life Cycle) – Identifying the problem (flow chart, DFD etc) - Applying modulartechniques – Writing the Algorithm – Flow Chart Elements – Using Clear Instructions – Correct Programming Techniques (Analyzing a Source Programme) – Debugging- Program Documentation – Program Maintenance –Operating System Concepts

M.Voc SOFTWARE DEVELOPMENT

PROGRAMME STRUCTURE

LEGEND	
Item	**Description**
C	Credits
CA	Continuous Assessment
ESE	End Semester Evaluation
L+T	Lecture + Tutorial Hours
P	Practical Hours
T / Tot	Total

SEMESTER I										
Course No	Course Code	Course Name	Credit	Weightage			Hrs /week			Hrs/ Sem
				CA	ESE	To	L+T	P	T	Tot
1.1	GEC1SD01	Communication Skills Development	4	1	4	5	3+1	0	4	60
1.2	SDC1SD01	Object Oriented Programming with Java and SQL	5	1	4	5	0	5	5	75
1. 3	SDC1SD02	PHP Programming	4	1	4	5	3+1	0	4	60
1. 4	SDC1SD03	Introduction to Mobile Application Development and Web Technologies	4	1	4	5	3+1	0	4	60
1. 5	SDC1SD04	Android App Development for Beginners	4	1	4	5	3+1	0	4	60
1.6	SDC1SDL1	LAB I : PHP Programming - Lab	5	1	4	5	0	5	5	75
1.7	SDC1SDL2	LAB 2:Software Lab I (Android I,Java &SQL)	4	1	4	5	3+1	0	4	60
1.8	SDC1SDA1	Ability Enhancement Course (AEC) Audit Course – 1	4	5	0	5	0	0	0	0
		Total	**30**				**20**	**10**	**30**	**450**

SEMESTER II **NSQF Level 8**										
Course No	Course Code	Course Name	Credit	Weightage %			Hrs/wk			Hrs/ Sem
				CA	ESE	To	L+T	P	Tot	Tot
2.1	GEC2SD02	Professional Skills Development (Training Programme)	4	1	4	5	3+1	0	4	60
2.2	SDC2SD05	Database and Backend Technologies	4	1	4	5	3+1	0	4	60

2.3	SDC2SD06	Advanced Java Programming	4	1	4	5	3+1	0	4	60
2.4	SDC2SD07	Android App Development - Advanced	4	1	4	5	3+1	0	4	60
2.5	SDC2SDL3	LAB 3: Advanced Java – Lab	5	1	4	5	0	5	5	75
2.6	SDC2SDL4	LAB 4 : Software Lab II (Android II and Database) (LAB)	4	1	4	4	0	4	4	60
2.7	SDC2SDL5	LAB 5 : Mini Project/Internship [Android App Development]	5	1	4	5	0	5	5	75
2.8	SDC2SDA2	Professional Competency Course (PCC)- Audit Course II	4	5	0	5	0	0	0	0
		Total	**30**				**20**	**10**	**30**	**450**

SEMESTER III

Course No	Course Code	Course Name	Credit	Weightage %			Hrs/wk			Hrs/ Sem
				CA	ESE	Tot	L+T	P	Tot	Tot
3.1	GEC3RM04	Research Methodology	3	1	4	5	2+1	0	3	45
3.2	SDC3SD08	Programming with Swift	4	1	4	5	3+1	0	4	60
3. 3	GEC3SD03	Elective – I	4	1	4	5	2+1	1	4	60
3. 4	SDC3SD10	Machine Learning	4	1	4	5	3+1	0	4	60
3. 5	SDC3SD11	iOS App Development –Fundamentals	4	1	4	5	3+1	0	4	60
3.6	GEC3SD04	Elective – II	4	1	4	5	0	4	4	60
3.7	SDC3SDL6	LAB 6:Android App Development-Lab	4	1	4	5	0	4	4	60
3.8	SDC3SDL7	LAB 7: Software Lab III (iOS and Swift)	3	1	4	5	0	3	3	45
		Total	**30**				**20**	**10**	**30**	**450**

SEMESTER IV

NSQF Level 9

Course No	Course Code	Course Name	Credit	Weightage			Hours/ wk			Hrs/ Sem
				CA	ESE	Tot	L+T	P	Tot	Tot
4.1	SDC4SDTP	Term Paper	30	5	0	5	0	30	30	900
4.2	SDC4SDL8	LAB 8: Internship & Project		1	4	5				
		Total	**30**				**0**	**30**	**30**	**900**

Grant Total	**120**						**120**	

The following Transaction modes are used for each subjects:

Modes of Classroom Transaction:

1. Lecture
2. Demonstration
3. Project Method
4. Inquiry training
5. Seminar
6. Group discussion
7. Flipped learning
8. Tutorial
9. Self-learning
10. Case study

The following tools can be used in different transactional modes:

1. PPT
2. Video
3. e-content
4. Google drive

SEMESTER 1

GEC1SD01 COMMUNICATION SKILLS DEVELOPMENT

Course Number: 1.1
Contact Hours per Week : 4 (L+T+P = 3 +1+0)
Number of Credits: 4
Number of Contact Hours : 60 Hrs

Course Objectives

- To provide a platform for future mobile phone application development professionals to comprehend the importance of business communication and business etiquette, by understanding the nature and process of communication and make it effective in use.

Course Outcomes:

- Apply business communication theory to solve workplace communication issues.
- Display competence in oral, written, and visual communication.
- Communicate effectively with colleagues in meetings,prepare agenda, minutes, and memos, and write different types of business letters, tenders, and quotations
- Prepare resumes, job application cover letters, and effective PowerPoint presentations

Course Outline

Module 1

Nature and Definition of Communication, Process of Communication, Types of Communication (Verbal & Non Verbal), Importance of Communication, Different forms of Communication, Managing Language, Use of Online Tools for vocabulary building, Use of software for editing

Module II

Barriers to Communication Causes, Linguistic Barriers, Psychological Barriers, Interpersonal Barriers, Cultural Barriers, Physical Barriers, Organizational Barriers, Effective Speaking, Oral Presentation Importance, Characteristics, Presentation Plan, Power point presentation, Visual aids, Use of presentation software and tools, Presenting data and charts.

Module III

Principles of Letter Writing, Nature & Function of letters, Principles, Elements of structure, Forms of Layout, Styles of presentation. Business Correspondence: Inviting quotations, Sending quotations, Placing orders, Sales letters, Claim & adjustment letters and social correspondence.

Module IV

Other Business Communication, Memorandum, Essentials of a memorandum, Drafting Inter - office Memo, Notices, Agenda, Minutes, Job application letters, preparing the Resume. Report Writing: Business reports- Types, Characteristics, Importance, Elements of structure, Process of

writing reports

Module V

Modern communication Blogs, Online forums, Etiquette in using modern communication tools.

Interviews: Objectives, Types, Group Communication: Forms, Body language in group communication, Group Discussions, Meetings, Conferences, Negotiations, Business etiquette.

REFERENCES

1. *Krishmohan and Meera Banerjee,* Developing Communication Skills*, Macmillan India Ltd,2015, ISBN-13: 978-9384872892.*
2. *Meenakshi Raman and Sangeeta Sharma,* Technical Communication: Principles and Practice, *2nd Edition, Oxford University Press, 2011, ISBN-13: 978-0198065296.*
3. *Chaturvedi P.D and MukeshChaturvedi,* The Art and Science of Business Communication*, 4e, Pearson, 2017, ISBN-13: 978-9332587281.*
4. *Sanjay Kumar and Pushp Lata,* Communication Skills*, Oxford University Press, 2011, ISBN-13: 978-0198069324.*
5. *Beebe T. and Mottet X.,* Business and professional communication: Principles and skills for leadership*, NY: Pearson, 2015, ISBN-13: 978-1323151471.*
6. *Thomas Elliott Berry,* The Most Common Mistakes in English Usage *,TMH Publication, 1971, ISBN-13: 978-0070050532.*
7. *AshaKaul,* Effective Business Communication*, Pearson /Prentice Hall of India Pvt Ltd, 2000, ISBN-13: 9788120317093.*

SDC1SD01 – OBJECT ORIENTED PROGRAMMING WITH JAVA AND SQL

Course Number: 1.2
Contact Hours per Week: 4
Number of Credits: 5
Number of Contact Hours : 75 Hrs

Course Objectives

Object Oriented Programming serves as the foundation to modern day programming. The subject provides an in-depth knowledge with OOPS concepts and provides hands on experience to work with Java and Database applications.

Course Outcomes

- Understand the basics of programming to write simple programs in java and understand the syntax
- and semantics of database programming using SQL.
- Apply the programming structures to write simple/intermediate programs and debug it using
- exception handling.
- Design and create intermediate/complex solutions using advanced java concepts
- Analyze and create database programming aspects to design and manage robust databases and synthesize efficient queries.

Course Outline

Module I

Introduction to object oriented programming, Java Virtual Machine, Java overview, Data Types, Variables, Arrays, Operators, Expressions, Decision Making through branching and looping.

Constructor, overloading constructor, Argument passing, returning objects, Recursive methods, Static variables block methods, Controlling access to class members, Nested classes, inner classes, command line Arguments, this keyword, variable length Arguments.

Module II

Introduction, Creating Subclass, calling super class constructor, Referring super class members, Referring subclass objects with super class variables, Multilevel inheritance and its implementation, Overriding Methods and dynamic method dispatch, Abstract methods, Abstract classes, Interfaces, Multiple inheritance, Interface, final keyword, using variables in interfaces, Extending interfaces, finalize() method. Packages: introduction to Packages, importing from other packages, access protection in packages, static import, introduction to wrapper classes, Auto Boxing and Auto unboxing, Character class, Boolean class. Exception handling: Introduction, try and catch blocks, single try and multiple catch blocks, Nested try, throw statement, throws statement finally block, creating our Exceptions, Chained Exceptions.

Module III

Introduction, Thread Creation, Thread priorities, Threads Synchronization, Producer-Consumer problem, Wait and notify Methods, Deadlocks, Suspending and Resuming a Thread. I/O Basics:- Introduction, Concepts of Streams, Stream classes, byte stream classes, characterstream classes, using streams, other useful I/O classes, using File classes, Input Output Exceptions, Creation Of files, reading and writing characters, Reading/ Writing Bytes, Handling primitive Data types, concatenating and buffering Files, random access Files, interactive Input Output. GUI Programming using Java FX: Introduction to JavaFX, Hello JavaFX, Event Handling, Scene Layout, Layout Panes, User Input, JavaFX Node Hierarchy, Lists, Tables, Menus.

Module IV

Getting Started with SQL: Relational Database Fundamentals, Database Models, Database Design Considerations. SQL Fundamentals: SQL Statements, Keywords, Datatypes, Nulls, Constraints. The Components of SQL: DDL, DML, DCL. Building and Maintaining a Simple Database Structure: Simple Database, Create, Alter, Deleting. Multitable Relational Database: Design, Indexes, Integrity, Normalization.

Module V

Manipulating Database Data: Add, Select, Update, and Delete data, Zeroing In on the Data You Want: Select, Where, Logical Operators, Group By, Having, Order By. Using Relational Operators: Union, Intersect, Except, Join, On vs. Where, Nested Queries, Recursive Queries. Database Security, Protecting Data: Transaction, Commit & Rollback, Locking Database Object, Backup, Savepoint & Subtransactions, Accessing Data with ODBC and JDBC, Advanced Topics: Cursors, Procedures, Triggers, Exception Handling.

References:

1. *Herbert Schildt,* Java: The Complete Reference, *10th Edition, McGraw-Hill Education, 2017, ISBN-13: 978-1259589331.*
2. *Balagurusamy,* Programming with Java, *5th edition, Mc Graw Hill India, 2014, ISBN-13: 978-93513 43202.*
3. *Herbert Schildt,* Java: A Beginner's Guide, *7th edition, Mc Graw Hill Education, 2017, ISBN-13: 978-1259589317.*
4. *Debasish Jana,* Java and object-oriented programming paradigm, *PHI Learning, 2009, ISBN- 13: 978-8120327757.*
5. *Gregory David Speegl,* JDBC: Practical Guide for Java Programmers, *Morgan Kaufmann Publishers, 2001, ISBN-13: 9781558607361.*
6. *Allen G. Taylor,* SQL For Dummies, *8th edition, John Wiley & Sons Inc, 2013, ISBN-13: 978- 1118607961.*
7. *Ivan Bayross,* SQL, PL/SQL the Programming Language of Oracle, *BPB Publications, 2010,*

ISBN-13: 978-8176569644.

8. *Doug Lowe*, JavaFX for Dummies, *John Wiley & Sons, 2015, ISBN-13: 978-1118385340.*

SDC1SD02 - PHP PROGRAMMING

Course Number: 1.3
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hrs

Course Objectives

- To understand the concepts, methodologies and architectures required for a web developer
- Understand how server-side programming works on the web.
- To provide hands on experience in web development.
- To equip the students to work with a professional web development environment.

Course Outcomes:

After successful completion of this course, students will be able to:

- Learn how to make dynamic web applications using PHP
- Write PHP scripts to handle HTML forms and regular expressions including modifiers, operators, and meta characters.
- Create PHP programs that use various PHP library functions, and that manipulate files and directories.
- Analyze and solve various database tasks using the PHP language.
- Learn how to Test and debug a PHP application

Course Outline

Module I :

Introduction to PHP as a programming Language: - Advantages of PHP, the server side architecture Decomposed, overview of PHP, history, object oriented support, benefits in running PHP as a server side script.

The basics of PHP: - data types, variables, constants, operators, Arrays, Conditional statements (if statement, Executing Multiple Statements, else if clause and switch statement), Iterations (for loop, while loop, controlling an array using a while loop, do while statement, for each loop and special loop key words)

Module II :

Functions, user defined functions, functions with arguments, PHP server Variables, working with date and time , System Variable (GET, POST, Cookies & Session, Forums)

Module III :

Working with forms, form elements (Text Box, Text Area, Password, Radio Button, Checkbox, The Combo Box, Hidden Field and image), adding elements to a form, uploading files to the Web Server using PHP, Error handling in PHP: - Displaying errors, warnings, types of errors, error levels in PHP, logging Errors and Ignoring errors.

Module IV :

Data base connectivity using PHP (MySQL, ODBC, ORACLE, SQL) Performing, executing Commands, different types of Data Base Operations like Insertion, deletion, update and query on data

Module V :

Introduction of Laravel, Installation , Directory Structure , URL Routing , Controller , View , Passing Data in View , Print Variable , Control Statement , Build Master Layout , Include a File , Form Handling , Connecting Database , MODEL , Error Handling , Validation Preserving the Data , File Uploading , Session , Removing Public From URL

References:

1. The Joy of PHP Programming. A beginner's guide by Alan Forbes
2. PHP: The Complete reference by Steven Holzner
3. PHP and MySql by Murach and Ray harris
4. PHP: A beginners Guide by Vikram Vaswani

SDC1SD03 –INTRODUCTION TO MOBILE APPLICATION DEVELOPMENT AND WEB TECHNOLOGIES

Course Number: 1.4
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hrs

Course Objectives

To introduce the mobile phone application development platform, the common architectures, concepts and technologies used in the background of Mobile phone Application Development. To give a basic level knowledge and awareness to different modern web architectures, protocols and networking.

Course Outcomes:

- Student knows mobile devices and mobile platforms
- Understand the basic concepts for mobile platforms and their supporting technology, and classify the different architectures used in server/client/cloud systems.
- Evaluate the architecture used for web-based development and deployment

Course Outline

Module I

Introduction to mobile phone generations – 1G to 5G, Smart phone architecture- ARM and Intel architectures, Power Management, Screen resolution, Touch interfaces, Memory-Sensors, I/O interfaces, GPS, Application deployment. Mobile OS Architectures-Kernel structure- Comparing and Contrasting architectures of Android, iOS and Windows, Darwin vs. Linux vs. Windows, Runtime (Objective-C vs. Dalvik vs. WinRT), Approaches to power management and Security

Module II

Mobile Application Architectures: Client-Server-Connection Types-Synchronization-Architectural Patterns-Architectural Design Tenets. Mobile Infrastructure: Mobile Device Types-Mobile Device Components-Connection Methods. Mobile Client Applications: Thin Client-Fat Client-Web Page Hosting- Best Practices, Issues- Existing Web Architectures and Back-End Systems Security Issues.

Module III

Introduction to Web: In Perspective, Origin. Before the web: TCP/IP. Birth of WWW: HTTP. Web Servers, Web Browsers. HTML & its Roots, XML & Applications, Dynamic Web

Applications, Approaches to web application development.

Module IV

Internet Programming: IP: Packet Format, Addressing, Addressing Class, Routing, Protocols -- Network: ARP, ICMP, DHCP, Transport: TCP, UDP. IPv6, Wireless IP, FTP, SNMP, SMTP. Domain: DNS, DDNS, NIS, LDAP. File: FTP, SFTP, TFTP. Mail: SMTP, MIME, POP, IMAP.
WAP, VoIP, IPTV. JSON: Introduction, Datatypes, Objects, Schema, Encode and Decode. JSON with PhP, JSON with Java, JSON with Python.

Module V

GraphQL : Introduction- GraphQL & Relay, GraphQL vs REST. GraphQL Server:Setting Up, Node.js, Schema, MongoDB, HTTP Interface, Editor. The Query Language: Documents & Operations, Fields, Variables, Directives, Aliases, Fragments & Mutations. The GraphQL Schema: Object, Introspection, type System, resolve function, validation & versioning. Web Socket Programming: Introduction: Handshake, Life before WebSocket, HTML5, WebScoket Protocol. WebSocket API: Overview, Events, Actions. Server Configuration: Setting up, Connection, Other Methods. Data Transfer: Sending, Receiving, Decoding. Security: Common Attacks & Defenses, Security Tools, Error Handling.

References:

1. *Martin Sauter*, 3G, 4G and Beyond–Bringing Networks, Devices and the Web Together, Second Edition, *Wiley, January 2013, ISBN-13: 9781118341483.*
2. *Sajal Kumar Das*, Mobile Handset Design, *John Wiley & Sons, 2013, ISBN- 13: 9780470824672.*
3. *Godbole and Khate, Web Technologies, 3rd Edn. McGraw Hill Education, 2017, ISBN- 13: 978-1259062681.*
4. *Leonard Richardson and Sam Ruby*, RESTful Web Services: Web services for thereal world, *O'Reilly Media, 2007, ISBN-13: 978-0596529260.*
5. *B V Kumar and S.V Subrahmanya;* Web Services: An Introduction, *McGraw Hill Education (India) Private Limited, 2012, ISBN-13: 978-1259002762.*
6. *Michael Rosen, Boris Lublinsky, Kevin T. Smith, and Marc J. Balcer*, AppliedSOA: Service-Oriented Architecture and Design Strategies, *Wiley, 2008, ISBN-13: 978047022365-9.*
7. *Sam Key*, XML Programming Success in a Day: Beginner's Guide to Fast, Easy, and Efficient Learning of XML Programming, *CreateSpace Independent Publishing Platform, 2015, ISBN-13: 978-1515212119.*

SDC1SD04- ANDROID APP DEVELOPMENT FOR BEGINNERS

Course Number: 1.5
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hrs

Course Objectives

To provide in depth knowledge and hands on experience in android application development, the latest trends and features. Successful completion will equip the aspirants to do android based development right from scratch.

Course Outcome

Students will able to:

• Understand Java and Android development framework components and Java/Android Development Tools

• Evaluate the core building blocks of android and android lifecycle architecture

• Use Intent , Broadcast receivers and Internet services in Android App.

• Design and implement Database Application and Content providers.

• Evaluate different messaging constructs and themes in android.

• Discuss various security issues in Android platform

Course Outline

Module I

Introduction of Mobile Technology: History of mobile phones, Evolution of Mobile Technology. Introduction to Java Enabled Mobiles: Micro Edition, J2ME.

Introduction to Android: History, Danger, Android Inc, HTC Sooner, T-Mobile G1. Introduction to Software's and Tools: Android Studio. Introduction to Building Tools: Java, JDK, JRE, Android SDK, Android Developer Tools. Setting up Android Environment, Introduction to Android OS: Android Architecture.: Overview of the Stack , Linux Kernel , Native Libraries , Dalvik Virtual Machine, Android Virtual Machine (ADT), Dalvik Debug Monitor Server (DDMS), LogCat, Application Framework, Application Licensing, Gradle.

Module II

Introduction to Android Development: Android Core Building Blocks, Android Emulator, AndroidManifest.xml, R.java file, uses-permission, Project Structure, Layout resource. Hello World User Interface: Working with Button, Toast, Button, Toggle Button, Switch Button, Image Button, Check Box, Alert Dialog, Spinner, Auto Complete Text View, Rating Bar, Date Picker, Time Picker, Progress Bar. Android Life Cycle: Activity, Intent, Android Menus, Layout Manger.

Module III

Intermediate Development I: Notifications, Toast, Custom Toast, Dialogs, Status bar Notifications. Multithreading: Using Java Muti threading classes, AsyncTask, Handler, Post. Styles and Themes: Creating and applying simple Style, Inheriting built-in Style and User defined style, Using Styles as themes. Resources and Assets: Android Resource, Using resources in XML and code, Localization, Handling Runtime configuration changes.

Module IV

Intermediate Development II: Fragment Lifecycle, Fragment Example, Dynamic Fragment. Adaptor: Array Adaptor, ArrayList Adaptor, Base Adaptor. View: Grid View, Web View, Scroll View, Search View, Tab Host, DynamicListView, ExpandedListView. Creating UI through code and the XML, Communicating data among Activities, Overview of services in Android, Implementing a Service, Service lifecycle, Recycler View.

Module V

Advanced Intent and Broadcast Receivers: Role of filters, Intent-matching rules, Filters in your manifest, Filters in dynamic Broadcast Receivers, Creating Broadcast receiver. Receiving System Broadcast: Understanding Broadcast action, category and data, Registering Broadcast receiver through code and through XML, Sending Broadcast.

References:

1. *Dawn Griffiths,* Head First Android Development, *O'Reilly Media, Inc, 2015. ISBN- 13:978- 1449362188.*
2. *Mark L. Murphy,* The Busy Coder's Guide to Android Development, *Commons Ware, LLC, 2015, ISBN-13:978-0981678009.*
3. *Wallace Jackson,* Android Apps for Absolute Beginners, *Apress, 2012. ISBN-13:978- 1430247883.*
4. *Dawn Griffiths and David Griffiths,* Head First Android Development: A Brain-Friendly Guide, *O'Reilly Media, Aug 19, 2017, ISBN-13: 978-1491974056*
5. *Bill Phillips, Chris Stewart,* Android Programming: The Big Nerd Ranch Guide, *Addison- Wesley Professional, 2015, ISBN-13: 978-0134171456.*

SDC1SDL1- LAB 1: PHP PROGRAMMING - LAB

Course Number: 1.6
Contact Hours per Week: 5
Number of Credits: 5
Number of Contact Hours : 75 Hrs

Course Objectives:

- To acquire knowledge and skills for creation of web site considering both client and server-side programming
- To be well verses with XML and web services technologies
- To be familiarized with open source frameworks for web development

Course Outcome:

• Develop simple application using server side PHP programming and database connectivity
• Learn to do validation using JavaScript objects by applying different event handling mechanism.
• Use AJAX programming technique to develop RI

Course Outline

Lab Exercises:

Section I

1. Simple programs using PHP
2. Factorial of a number
3. Sum of Digits
4. Fibonacci series
5. Armstrong number or not
6. Sort an array of numbers, Strings etc

Section II

1. Create a registration form and insert the data into database using the form through PHP Post method.
2. Create a login form in PHP to sign into your registered account (using PHP session). And also add logout button to exit from the account.
3. Create a form to upload images, pdf and videos into database. And also create a PHP page to view all resources uploaded.
4. Create a PHP page to upload multiple images into a database and view it.
5. Create a form to dynamically fetch all the states and districts from the database in the "onchange" of dropdown list. (select tag)
6. Create a PHP page to edit your profile and update your details in the database.
7. Create a PHP page to change your account password by specifying your correct password, new password and confirm password.
8. Create a PHP page to show all the images you uploaded and delete it from the database.
9. Create a chat box in your webpage using PHP

DATA BASE

1. Create a webpage with laravel framework to learn controller, view and model concept.
2. Create a webpage with sufficient forms to insert data into and select data from the data base.
3. Develop a webpage to edit and delete the details you are added into your database.
4. Develop a webpage to upload and view image into and from the database.
5. Create a webpage to learn account login and logout.
6. Create a small website with the above all applications

SDC1SDL2- LAB 2: SOFTWARE LAB I (ANDROID I, JAVA&SQL)

Course Number: 1.7
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hrs

Course Objectives:

- The Software Lab I provide hands on experience in basic Android, Java and SQL concepts.
- The Lab is centred on the concepts covered on the subject Android App Development for Beginners and Object-Oriented Programming with Java & SQL in Semester I.

Course Outcome:

1. Create advanced applications based on Java
2. Create advanced databases based on SQL and SQLite
3. Design and Create advanced projects based on Android

Course Outline:

Java Basics: Programs to familiarize with the following concepts.

a. Data Types, Variables, Arrays, Data Structures.
b. Operators, Control Statements and Loops.
c. Class, Inheritance Polymorphism.
d. Packages and Interface.
e. Exception Handling.
f. Threading: Simple andMultithreading.
g. Java Collections: Basic andAdvanced.
h. Java IO: File Handling.
i. Java Networking: Sockets.

SQL Basics: Programs to familiarize with the following concepts.

a. DDL and DML Queries.
b. Grouping and Aggregating.
c. SQL Joins.
d. Inner Queries and Multilevel Queries.

SQLite Basics: Programs to familiarize with the following concepts.

e. SQLite Queries: Create, Drop, Alter, Insert, Select, Delete.
f. Query Filters: Group By, Order By.
g. Conditions: Where and Having.
h. Aggregate Functions.

Android Basics: Programs to familiarize with the following concepts.

a. Sample Program: Hello World.
b. Layouts and Positioning.
c. Controls and Widgets.
d. Activity and Intents.
e. Activity Permissions.
f. Notifications.
g. Threading and AsyncTask.
h. Resources.
i. Views.

References:

1. *Schildt* , Java 2:The Complete Reference, *Tata McGraw-Hill Education, 2002, ISBN-13 : 9780070495432*
2. *Jay Kreibich,* Using SQLite, *O'Reilly Media Inc., 2010, ISBN-13 9780596521189.*
3. *Wilton,* Beginning SQL, *John Wiley & Sons, 2005, ISBN-13: 9788126505852*
4. *Mark L. Murphy,* The Busy Coder's Guide to Android Development, *Commons Ware, LLC,2015, ISBN-13:978-0981678009*
5. *Reto Meier,* Professional Android 4 Application Development, *Wrox, 2012, ISBN-13:978-1118102275.*

SDC1SDA1 ABILITY ENHANCEMENT COURSE (AEC) - AUDIT COURSE– 1

Course Number: 1.8
Number of Credits: 4

Course Evaluation: 100 % (Internal)

Examination : 2 hrs
Weightage : 20

Course Evaluation & Course Credit

The Ability Enhancement Audit Course has 4 credits which will not be counted for evaluating the overal SGPA & CGPA. The College/Department shall conduct examination of 2 Hrs duration with a minimum o 20 weightage before the conclusion of first semester classes and have to intimate /upload the results of th same to the University on the stipulated date during the III Semester. Students have to obtain only minimun pass requirements in this Audit Course.

Course Delivery Mode

This course is an Ability Enhancement Audit Course. The course content is not delivered in th classrooms. Instead, the students have enroll themselves for the online course offered at NPTEL, MOOC etc The online course is available at https://nptel.ac.in/courses/121106007/. Students can either view the vide module online or can download the video lessons and transcripts to view or read them offline.

The student can attempt any one of the following for securing 4 credits.

1. Communicative Skill Development Course
2. Community Linkage Programme in a nearby Local Government.
3. One case study analysis approved by the Department Council.
4. One seminar presentation of 15 minutes duration, on a relevant topic.
5. An internship in an Industry or academic /research institution or in any related organization suitable to the topic under study, under a supervisor / teacher official.
6. Review of one recently published book related to computer Science / Information technology.

7. Practical Training / Internship in IoT

If the student is opting for a practical training or an internship (maximum of two weeks) for IoT, they shall follow the following syllabus:

INTERNET OF THINGS (IOT) – LAB

Course Objectives

1. Able to learn the application of physical components with cyber space
2. Learn to develop small hardware embedded devices

Course Outline

Lab Exercises

1. LED on and off manually.
2. LED on and off using LDR (automatic)
3. Distance Calculation using Ultrasonic Sensor
4. Rotating Servo Moy
5. Create LED Chaser using 4 LEDs
6. Control an LED using switch.
7. Object Detection
8. Serial print temperature in-room
9. LED blink node mcu
10. Control brightness of an LED

References:

1. Video Lessons and Transcripts available (including in the regional language) at https://nptel.ac.in/courses/nptel_download.php?subjectid=121106007

SEMESTER 2

GEC2SD02- PROFESSIONAL SKILLS DEVELOPMENT (TRAINING PROGRAMME)

Course Number: 2.1
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hrs

Course Objectives

- To acquire various soft skills and presentation skills
- To equip students with basic skills to become effective team players, problem solvers at personal interpersonal, group and organizational levels**.**

Course outcome:

- Understand and apply skills in interpersonal relationships in the workplace
- Apply productivity improvement techniques at work.
- Understand and demonstrate knowledge of problem- solving and creativity techniques.
- Understand demonstrate skills in public speaking, oral presentations, and teamwork.

Course Outline

Module I: Personal skills

Developing Self-Awareness, Managing Personal Stress, Goal setting, Personal SWOT analysis, Personal productivity techniques, Solving Problems Analytically and Creatively, Positive thinking.

Module II: Interpersonal skills

Training on Johari Window - Building Relationships by Communicating Supportively, Gaining Power And Influence, Motivating Others- Building Emotional intelligence Managing Conflict.

Module III: Group skills

Empowering and Delegating, Building Effective Teams And Teamwork, Negotiating effectively, Leading Positive Change.

Module IV: Specific communication skills

Extempore and public speaking, Group Discussion, Building Skills for cracking

Interviews, Conducting Meetings

Module V: Leadership skills

Games, Understanding power, Task Assignments, Role perception, organizing events, feedback.

SDC2SD05- DATABASE AND BACKEND TECHNOLOGIES

Course Number: 2.2
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hrs

Course Objectives

- To give awareness to different types of databases and its usage in mobile phone application development.
- To work with different database servers and backend technologies to communicate with Mobile user interface.
- To give hands on experience in database design and implementation for real world problems.

Course Outcomes:

- Analyze different types of DBMS and Employ it in real- life problems
- Evaluate different means of advanced DBMS functions and implement them in the production environment.
- Design and create databases based on MongoDB tool.
- Understand the concepts of BigData and its application.
- Design and create queries based on triggers, aggregate functions, stored procedures, SQL joins, DDL, DML, and views (Create).

Course Outline

Module I

Relational vs. Non-relational databases, SQL, Popular SQL databases and RDBMS's, document oriented databases, Categories of Non-relational databases, popular non-Relational databases, Comparing document-oriented and relational

databases using examples, How to choose SQL or No-SQL database to work with a real- life problem.

Module II

SQL Server - Understanding SQL, Introducing SQL Server, Retrieve, sort, and format database contents, filtering techniques, using aggregate functions, join two or more related tables, Insert, update, and delete data; Create and alter database tables; Work with views, stored procedures, Manage Transaction Processing; Work with XML and JSON; Advanced SQL Features.

Module III

Mongo DB server- Installation, MongoDB Basics, MongoDB CRUD Operations, SQL to MongoDB, MongoDB Indexing, Monitoring & Backup, Create User in MongoDB & assign Roles, Indexing, Aggregation, MongoDB Shared Cluster, Deployment

Module IV

Work with five real world examples, Design SQL or Non-SQL database for each, Implement and deploy it using appropriate server.

Module V

Introduction to Big data systems, Real-time processing of web-scale data, Hadoop- Introduction, HDFS and MapReduce concepts with examples, Streaming-Working with an example.

References:

1. *Guy Harrison,* Next Generation Databases: NoSQL and Big Data, *1st ed. Edition, Apress, 2015, ISBN-13: 978-1484213308*

2. *Luc Perkins , Eric Redmond, Jim Wilson,* Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement, *2nd Edition, Pragmatic Bookshelf, 2018, ISBN-13: 978-1680502534*

3. *Ben Forta,* Microsoft SQL Server T-SQL in 10 Minutes, *Sam's Teach Yourself, Sams publication, 2016, ISBN-13: 978-0672337925.*

4. *Dejan Sarka,* SQL Server 2016 Developer's Guide, *Packt Publishing Limited , 2017, ISBN-13: 978-1786465344*

5. *Cyrus Dasadia , Amol Nayak ,* MongoDB Cookbook , *Packt Publishing , 2016, ISBN- 13: 978-1785289989*

6. *Nathan Marz, James Warren,* Big Data: Principles and best practices of scalable realtime data systems, *Manning Publications, 2015, ISBN-13: 978-1617290343*
7. *Benjamin Bengfort* , Data Analytics with Hadoop: An Introduction for Data Scientists, *O'Reilly Media, 2016, Shroff Publishers, 2016, ISBN-13: 978-9352133741*

SDC2SD06 - ADVANCED JAVA PROGRAMMING

Course Number: 2.3

Contact Hours per Week: 4

Number of Credits: 4

Number of Contact Hours : 60 Hrs

Course Objectives

• To learn the advanced features of java programming language that equip the students to develop web based applications to develop web based applications

Course Outcome

The students will :

• Get knowledge about JVM architecture
• Be able to write advanced Java Programs using Hibernate, Spring technologies
• Be able to develop Spring based applications
• Get knowledge about J2ME applications

Course Outline

Module I

JVM Architecture - Class Loader Subsystem - Runtime data Area - Method Area - Heap Area - Stack Area - Native Method Stack - Execution Engine - Garbage Collection. Collections **:** Collection Interfaces, Collection Classes, Collection Algorithms Java Beans: Introduction, Properties, Bean Builder, Advantages, BDK Introspection, Beaninfo interface, Persistence, Customizer,Javabeans API

Module II

JSP: Overview of JSP, JSP Architecture & life cycle, Components of JSP, Implicit Objects & standard JSP Tags, Scope of JSP objects. JDBC: JDBC Overview & Architecture, Step By StepUsage of JDBC API, Connecting to Database in Java, Prepared Statement & JDBC Transactions

Developing Web applications with MySQL Database by implementing Java Beans, DAO's & MVC Architecture.

Module III

Hibernate: Introduction to Hibernate, ORM Overview, Hibernate Environment, Hibernate Architecture

& API, Hibernate Configuration, Hibernate Sessions, Persistent Class & Mapping Files, Building Hibernate application, Hibernate Query Language (HQL), Hibernate O/R Mapping s – Collection & Association Mappings (ManytoOne, OnetoOne, OnetoMany, ManytoMany), Implementing Hibernate in Java Web Applications using Netbeans with MySQL.

Module IV

Spring:Introduction to Spring Framework Architecture, Bean Definition, Bean Scopes & Bean Definition Inheritance, Spring IoC Containers, Understanding inversion of control (IoC) - Dependency Injection (DI), Spring Setter Injection, Spring Constructor Injection, IoC in Action Architecture of Spring Web MVC Framework, Spring MVC Getting Started – constructing web MVC application using Spring Framework, Abstract Controller in Spring MVC, Spring MVC Controllers hierarchy, SimpleFormController, Spring DAO design pattern, Building Spring MVC Framework Applications by using Netbeans.

Module V

Java J2ME: J2ME Overview: Inside J2ME ,J2ME and Wireless Devices, Small Computing Technology:

Wireless Technology-Mobile Radio Networks, Messaging, PDAs, Mobile Power, set Top Boxes, smart cards. J2ME Architecture and Development Environments: J2ME Architecture, Small computing Device Requirements, MIDlet programming, J2ME Software Development Kits, Helloworld J2ME Style, J2ME Wireless Toolkit.

References

1. Cay S. Horstmann, Gary Cornell, “Core Java, Vol II” - Advanced Features, Pearson,9th edition
2. Craig Walls , "Spring in Action" - Manning Publications, 4th Edition (2014).
3. H. M.Deitel, P. J. Deitel, S. E. Santry, “Advanced Java 2 Platform HOW TOPROGRAM”
4. Prentice Hall.Jim Smith, Ravi Nair, “Virtual Machines”, Morgan Kaufmann, Chapter 6
5. Raoul-Gabriel Urma, Mario Fusco, and Alan Mycroft , “Java 8 in Action: Lambdas, Streams, and functional-style programming”, Manning Publications, 1st Edition (2014).
6. Uttam K.Roy, “Advanced Java Programming”, Oxford University Press (2015)
7. Jim Keogh, “The Complete Reference J2EE”, Tata McGraw-Hill
8. J2ME- The Complete Reference- James Keogh- TATA McGRAW-HILL
9. Cay S.Hortsmann,Gary Cornell,“Core Java Volume II-Advanced features, Pearson, 9th Edition
10. Subrahmanyam Alamaraju and RIC Buest- “Professional Java Server Programming-J2EE”, Apress Publication, 1.3 Edition

SDC2SD07-ANDROID APP DEVELOPMENT--ADVANCED

Course Number: 2.4
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hrs

Course Objectives

- To understand the advanced concepts, methodologies and architectures required to be an avid android programmer.
- To provide hands on experience in advance android concepts.
- To equip the students to work with a professional android app development environment.

Course Outcomes:

Students will able to:

- Describe Android platform, Architecture and features
- Design MVC architecture .
- Solve problems using SQLite and Content Providers
- Use Intent , Broadcast receivers and Internet services in Android App.
- Design and implement Database Application and Content providers.
- Use multimedia, camera and Location based services in Android App.
- Discuss various security issues in Android platform
- Create solutions based on the REALM framework.

Course Outline

Module I

Relational vs. Non-relational databases, SQL, Popular SQL databases and RDBMS's, document oriented databases, Categories of Non-relational databases, popular non-relational databases, Comparing document-oriented and relational databases using examples, How to choose SQL or No-SQL database to work with a real-life problem.

Module II

SQL Server - Understanding SQL, Introducing SQL Server, Retrieve, sort, and format database contents, filtering techniques, using aggregate functions, join two or more related tables, Insert, update, and delete data; Create and alter database tables; Work with views, stored procedures, Manage Transaction Processing; Work with XML and JSON; Advanced SQL Features

Module III

Mongo DB server- Installation, MongoDB Basics, MongoDB CRUD Operations, SQL to MongoDB, MongoDB Indexing, Monitoring & Backup, Create User in MongoDB & assign Roles, Indexing, Aggregation, MongoDB Shared Cluster, Deployment .

Module IV

Work with five real world examples, Design SQL or Non-SQL database for each, Implement and deploy it using appropriate server.

Module V

Introduction to Big data systems, Real-time processing of web-scale data, Hadoop-Introduction, HDFS and MapReduce concepts with examples, Streaming-Working with an example.

References:

1. *Micheal Burton, Android* App Development for Dummies, *3/e, John Wiley & Sons, 2015, ISBN-13:978-1118387108.*
2. *Lauren Dracy, Shane Conder,*Android Wireless Application Development-Advanced Topics, Volume II, *Addison-Wesley Professional, 2012, ISBN-13:978-0321813848 3.Joseph Annuzzi Jr,*Advanced Android Application Development, *Addison-Wesley Professional, 2014, ISBN-13:978-9332552012*
4. *Erik Hellman,* Android Programming: Pushing the Limits, *John Wiley & Sons, 2014, ISBN-13:978-8126547197.*
5. *Mark Wickham.*, Practical Android: 14 Complete Projects on Advanced Techniques and Approaches,*1st edn, Apress, Jan 2018, ISBN-13: 978-1-4842-3333-7*
6. *Reto Meier,* Professional Android, *4thedition, Wrox;, April 2018, ISBN-13: 978- 1118949528*
7. *Barry Burd,* Android Application Development All-in-One For Dummies,*Second edition, Wiley, 2015, ISBN-13: 978-8126557943*

SDC2SDL3 LAB3: ADVANCED JAVA - LAB

Course Number: 2.5
Contact Hours per Week:
5Number of Credits: 5
Number of Contact Hours : 75 Hrs

Course Objectives

- To get knowledge about the advanced techniques of java programming
- To get an overall idea about the java architectures, frameworks, interfaces etc
- To know more about server side programming.

Course Outcome

• Able to do advanced level programming in Java
• Able to do a small website using Java

Course Outline

Lab Exercises :

1. Recap: Simple Programs in Java like array, Strings etc
2. Develop Simple programs using JSP
3. Create a JSP application to perform database table operations such as insert, delete, update and select
4. Create a Login form to login into your account using username and password (JSP Session)
5. Create a JSP application to upload image into database and retrieve it from database
6. Create a servlet program to register the details with name, password, email and country
7. Create a servlet program to login with username and password
8. Programs with hibernate framework
9. Create a spring project to print a message in the console windows
10. Create a spring project to implement inheritance
11. Create a spring project to print list of strings
12. Spring applications of database
13. Simple programs using J2ME
14. Develop a simple website with a database using the above all technologies

References

1. Core Java 2, Vol 11 Advanced Features, 7th edition by Cay Horetmann, Gray Cornell Person Publisher
2. Professional Java Programming by Brett Spell, Wrox publication
3. Advanced java 2 Platform, How to program, 2nd edition, Harvey.M.Dietel, Prentice Hall

SDC2SDL4 LAB 4: SOFTWARE LAB II (ANDROID II AND DATABASE)

Course Number:
2.5 Contact
Hours per Week:
4Number of Credits: 4
Number of Contact Hours : 60 Hrs

Course Objectives

- To provides hands on experience in Advanced Android concepts. The Lab is centred on the concepts covered on the subject Android App Development – Advanced and Database & Backend Technologies in Semester II.

Course Outcomes:

- To have a review on concept of Android programming.
- To learn Android Programming Environments.
- To practice Design Solution based on advanced android concepts.
- To learn GUI Application development in Android platform with XML
- To Apply fundamentals of database concept and entity relationship model in database applications.
- Design solution based on NoSQL databases

Course Outline

Android Advanced: Programs to familiarize these concepts

- Advanced Intent and Broadcast Receivers
- Content Providers and MIME Types
- Services
- Multimedia : Sound and Video
- Location : GPS and Internet, Proximity Alerts
- Google Map: Overlays , Geocoder.
- Web Views
- Sensors : Accelerometer, Motion , Compass, Orientation
- Telephone: Managing Calls.

- Contacts: Accessing Updating and Deleting.
- Email: Sending and Receiving.
- Camera: Surfaces Manipulation.
- Animations : 2D and 3D,OpenGL

Database & Backend

- NoSQL
- Mongo DB
- CURD

References:

1. *Erik Hellman, Android Programming: Pushing the Limits, John Wiley & Sons, 2014, ISBN-13:978-8126547197.*
2. *Micheal Burton, Android App Development for Dummies, 3/e, John Wiley & Sons, 2015, ISBN-13:978-1118387108.*
3. *Zigurd Mednieks, Laird Dornin, G. Blake Meike, Masumi Nakamura,* Programming Android: Java Programming for the New Generation of Mobile Devices, *O'Reilly Media 2nd edition, 2012, ISBN-13: 978-1449316648*
4. *Lauren Dracy, Shane Conder,* Android Wireless Application Development Volume II: Advanced Topics: 2 (Developer's Library), *2012, ISBN-13: 978-0321813848*
5. *Joseph Annuzzi Jr, Lauren Darcey , Shane Conder ,* Advanced Android Application Development (Developer's Library), *ISBN-13: 978-9332552012*
6. *Guy Harrison,* Next Generation Databases: NoSQL and Big Data, *1st ed. Edition, Apress, 2015, ISBN-13: 978-1484213308*
7. *Cyrus Dasadia , Amol Nayak,* MongoDB Cookbook, *Packt Publishing , 2016, ISBN-13: 978- 1785289989*

SDC2SDL5- MINI PROJECT / INTERNSHIP [Android App Development]

Course Number: 2.7

Contact Hours per Week: 5 (L+T+P = 0 +0+5)

Number of Credits: 5Number of Contact Hours : 75 Hrs

Course Objectives

- Address the real world problems and find the required solution.
- Design the problem solution as per the requirement analysis done.
- Fabricate and implement the mini project intended solution for project based learning.
- Build and test the mini project successfully.
- Improve the team building, communication and management skills of the students.

Course Outcome

Student will able to:

- Identify the requirements for the real world problems.
- Study and enhance software/ hardware skills.
- Demonstrate and build the project successfully by hardware requirements, coding,

emulating and
- testing.
- To report and present the findings of the study conducted in the preferred domain

- Demonstrate an ability to work in teams and manage the conduct of the research study
- Evaluate client requirements efficiently
- Design software requirement specifications accurately
- Design solutions based on SRS, and design principles

Course Outline

To provide students with industrial exposure to real world of Android application development whereby they get an opportunity to apply the knowledge and skill acquired through the course.

The internship will be of 40 working days duration. The students are required to undertake the internship training with a reputed IT / Software Development Company on **Android Application** development or related technologies.

Student should submit the company details and work details to their concerned Faculty Guide for approval and feedback before the starting date stipulated by the centre. The internship work will be duly evaluated continuously by the Faculty Guide. Students have to produce a certificate of internship completion from the company at the time of rejoining the department, specifying the period of internship.

One copy of the final report, duly signed by the Faculty in Charge and the Director is to be submitted as part of completion of the Semester II. Individual presentation/demo of the work done and Viva voce is also included as a part of project evaluation.

Students are expected to adhere to highest standards of academic integrity. Specific instructions/guidelines for carrying out internship shall be prepared and given to the students by the centre every year.

Distribution	Marks
Content and relevance or Dissertation	60
Viva	20
Presentation	20

SDC2SDA2 PROFESSIONAL COMPETENCY COURSE (PCC) – AUDIT COURSE II

Course Number: 2.8

Number of Credits: 4

Course Evaluation: 100 % (Internal Component)

Examination : 2 Hrs

Weightage : 20

Course Evaluation & Course Credit

The Professional Competency Audit Course has 4 credits which will not be counted for evaluating the overall SGPA & CGPA. The College/Department shall conduct examination of 2 Hrs duration with a minimum of 20 weightage before the conclusion of second semester classes and have to intimate / upload the results of the same to the University on the stipulated date during the III Semester. Students have to obtain only minimum pass requirements in this Audit Course.

Course Delivery Mode

This course is an Professional Competency Audit Course. The course content is not delivered in the classrooms. Instead, the students have enroll themselves for the online course offered at NPTEL, MOOC etc. The online course is available at https://nptel.ac.in/courses/121106007/. Students can either view the video module online or can download the video lessons and transcripts to view or read them offline.

SEMESTER 3

GEC3RM04 RESEARCH METHODOLOGY

Course Number: 3.1
Contact Hours per Week: 3
Number of Credits: 3
Number of Contact Hours : 45 Hr

Course Objectives

- Be aware of the ethical principles of research
- To experience the various research level tools and techniques
- Identify the components of a literature review process
- Critically analyse published research

Course Outcome

• To enable the students to roll in to research level areas
• Develop to make use of online software tools

Course Outline

Module I

Introduction to Research Methodology: Concepts of Research, Meaning and Objectives of Research, Research Process. Type of research: Descriptive vs. Analytical, Applied vs. Fundamental, Quantitative vs. Qualitative, and Conceptual vs Empirical. Criteria of Good Research, Research Problem, Selection of a problem, Techniques involved in definition of a problem. Research Proposals : Types, contents, Ethical aspects, IPR issues like patenting, copyrights.

Module II

Research Design : Meaning, Need and Types of research design, Literature Survey and Review, Identifying gap areas from literature review, Research Design Process, Sampling fundamentals, Measurement and scaling techniques. Data Collection : concept, types and methods, Design of Experiments.

Module III

Quantitative Techniques: Probability distributions, Fundamentals of Statistical analysis, Data Analysis with Statistical Packages, Multivariate methods. Concepts of correlation and regression : Fundamentals

of time series analysis and spectral analysis.

Module IV

Report Writing: Principles of Thesis Writing, Guidelines for writing reports & papers, Methods of giving references and appendices, Reproduction of published material, Plagiarism, Citation and acknowledgement.

Module V

Documentation and presentation tools : LaTeX, Office with basic presentations skills, Use of Internet and advanced search techniques.

References:

1. Research Methodology: An Introduction for Science & Engineering Students', by Stuart Melville and Wayne Goddard, Juta and Company Ltd, 2004
2. Research Methodology: An Introduction' by Wayne Goddard and Stuart Melville, Juta and Company Ltd, 2004
3. Research Methodology: Methods and Techniques', by Dr. C. R. Kothari, New Age International Publisher, 2004
4. Research Methodology: A Step by Step Guide for Beginners' by Ranjit Kumar, SAGE Publications Ltd; Third Edition

SDC3SD08- PROGRAMMING WITH SWIFT

Course Number: 3.2
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hrs

Course Objectives

- To understand the basics of iOS programming using Swift.
- To provide theoretical knowledge and hands on experience in swift programming.
- The students will learn to use the concepts of Swift in iOS app development using both playgrounds and XCode projects.

Course Outcomes

Upon successful completion of this course, students should be able to:

- Define key programming terms relevant to Swift and iOS programming.
- Understand the operators, data structures,inheritance, and error handling in Swift
- State the purpose of the Apple developer tools, such as Xcode, Instruments, debugger, analyzer, and iOS Simulator.
- Create programs based using class,methods, protocols, generics, flow control, operators, and functions.

- Analyze access control and enumeration..
- Demonstrate programming best practices in Swift.
- Examine and subdivide app functionality into properly designed components.

Course Outline

Module I

Introduction: History of Swift, Features, Benefits, Objective-C and Swift, Introduction to Xcode and the iOS Simulator. Data Types: Basic Data Types, Tuples, Optional Types, Enumerations. Basic Operators: Assignment Operators, Arithmetic Operators, Comparison Operators, Range Operators, Logical Operators, Strings and Characters: Strings, Common String Functions, Interoperability with NSString, Collections: Arrays, Dictionaries, Copying the behavior of Arrays and Dictionaries.

Module II

Flow Control: Selection, Conditions, Boolean logic and IF Statements, Optionals, IF Let Statements, Testing for nil and Optional Bindings, Switch Statements, Range Operators; Looping: For-Loops, Nested Loops, For-in Loops, Half-Open Range Operators, While Loops and Repeat- While Loops, Functions: Defining and Calling a Function, Function Types, Nested Functions, Closures: Understanding Closures, Closure Functions of Arrays, Using Closures in our Functions.

Module III

Structures and Classes: Structures, Classes. Properties: Stored Properties, Computed Properties, type Properties, Property Observers, Methods: Instance Methods, Type methods, Subscripts. Inheritance: Understanding Inheritance, Types of Initializers, Deinitialization, Optional Chaining, Error Handling.

Module IV

Protocols and Delegates: Understanding Protocols, Protocols as Types, Protocol Inheritance, Class-Only Protocols, Protocol Composition, Understanding Delegates. Generics: Understanding Generics, Implement Generic Functions, Implement

Generic Functions using Multiple Parameters, Generic Types, Associated Types, Generic Subscripts.

Module V

Extension: Extension Syntax, Working with computed properties, methods, initializers and subscripts, Adding Protocol Conformance with an Extension, Protocol Extensions, Extensions with Generic Types, Access Control: Access levels, Access Control Syntax, Access control to classes, structures and enumerations, Assigning access levels to protocols and extensions.

Memory management: Automatic Reference Counting (ARC), Strong Reference Cycles between Class Instances, weak references, unowned references, Memory Safety, Understanding Conflicting Access to Memory.

References:

1. *Jon Hoffman*, Mastering Swift 4 - Fourth Edition: An in-depth and comprehensive guide to modern programming techniques with Swift, *Packt publishing, 2017, ISBN-13: 978- 1788477802*
2. *Donny Wals*, Mastering iOS 11 Programming - Second Edition: Build professional-grade iOS applications with Swift 4 and Xcode, *Packt publishing, 2017, ISBN-13: 978- 1788398237*
3. *Keith Moon*, Swift 4 Programming Cookbook, Packt Publishing, 1 edition (September 28, 2017), ISBN-13: 978-1786460899
4. *Matt Neuburg*, iOS 11 Programming Fundamentals with Swift: Swift, Xcode, and Cocoa Basics *1st Edition,* O'Reilly Media; 1 edition (October 19, 2017), *ISBN-13: 978- 1491999318.*
5. *Chris Eidhof, Ole Begemann, Airspeed Velocity,* Advanced Swift: Updated for Swift 4, CreateSpace Independent Publishing Platform, *2016, ISBN-13: 978-1539154716 6.Matthew Mathias, John Gallagher*, Swift Programming: The Big Nerd Ranch Guide (2nd Edition), Big Nerd Ranch Guides*; 2 edition (December 8, 2016), ISBN-13: 978- 0134610610*

GEC3SD03 - ELECTIVES I

Course Number: 3.3
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hr

LIST OF ELECTIVES

GEC3SD03E1- Mobile and wireless security

GEC3SD03E2-Cross Platform App Development Using React Native

GEC3SD03E3 - iOS App Development – Advanced Technologies

GEC3SD03E4- Watch OS Programming

GEC3SD03E5 - Healthkit and Homekit programming

GEC3SD03E6- Retail App Development Frameworks

.

GEC3SD03E1- MOBILE AND WIRELESS SECURITY

Course Number: 3.3
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hr

Course Objective:

- Provide knowledge of information security technology and methods for communication systems that provide services for mobile users by wireless access networks.
- Knowledge and understanding of security mechanisms and protocols in wireless communication systems.

Course Outcomes:

In this course you will:
• Acquire solid knowledge on mobile networks and mobile security
• Acquire experience and capability to team work
• Become familiar with wireless systems and standards
• Able to get an idea about the framework of mobile handset hardware design

Course Outline

Module I

Introduction & The Cellular Concept-System Design Fundamentals: Evolution Of Mobile Radio Communications- Introduction- Frequency Reuse- Channel Assignment Strategies – Handoff Strategies- Interference And System Capacity- Trunking And Grade Of Service- Improving Capacity In Cellular Systems.

Module II

Multiple Access Techniques And Wireless Networking : Introduction- FDMA- TDMA- Spread Spectrum- Multiple Access: Space Division Multiple Access- Packet Radio. Introduction To Wireless Networks- Differences Between Wireless And Fixed Telephone Networks- Development Of Wireless Networks- Traffic Routing In Wireless Networks- Integrated Services Digital Network (ISDN)- Protocols For Network Access

Module III

Wireless Systems And Standards : Global System for Mobile – CDMA Digital Cellular Standard (IS95) – CT2 Standard for Cordless Telephones Digital European Cordless Telephones (DECT).

Module IV

Mobile And Wireless Security : Security Primer- Creating A Secure Environment- ThreadsTechnologies-

Other Security Measures- WAP Security Measures- Smart Client Security- Overview of Smart Client Architecture- Mobile Operating Systems.

Module V

3G Handset Hardware Design : Spectral Allocations- Impact On Handset Hardware DesignGPRS/EDGE Handset Hardware- Design Issues For Multislot, Multiband, Multimode PhonesTransmitter Architectures- 3G Handset Hardware Code Properties- Code Generation- Radio Bandwidth Quality/Time And Frequency Domain Issues- 3G Handset Hardware Form Factor And Functionality.

References:

1. Theodore.S.Rappaport, Wireless Communications-Principles and practice, Prentice Hall Communications Engineering and Emerging Technologies Series, Upper Saddle River, New Jersey 07458, 1996
2. Martyn Mallick, Mobile&Wireless Design Essentials,Wiley Dreamtech India pvt ltd., 2003
3. Geoff Varall, Roger Belcher,3G Handset & Network Design, Wiley Dreamtech India pvt ltd., 2003

GEC3SD03E2-CROSS PLATFORM APP DEVELOPMENT USING REACT NATIVE

Course Number: 3.3
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hr

Course Objective:

- To provide in-depth understanding and knowledge of different architectures and technologies required to develop a cross platform mobile application successfully.
- To learn and understand the components and APIs required to work with the React Native.
- To provide hands-on experience in developing mobile apps for different platforms like iOS, Android etc. using React Native.

Course Outcome :

- The developers can make complex tasks simple by using simple codes
- Assess the difference between native, hybrid and cross-platform application development
- The developers can run new features and can maintain the application state at the same time.
- The app developers will have a direct preview of the code and its impact.
- Understand the architecture used in React Native
- Develop application using a third party, geolocation, states, storage and navigation components

Course Outline :

Module I

Introduction: Different types of Mobile app development platforms: Native, Hybrid, Web applications, “Native” cross-platform apps, Hybrid HTML5 cross-platform apps, Advantages of cross-platform apps. Introduction to React Native, History of React Native, Motivation behind creating React Native App. React Native vs hybrid applications. React Native: Information flow, Architecture, Threading model. React Native benefits.

Module II

Setting up an environment for developing iOS and Android apps, Introduction to JSX, Creating your First Application with create-react-app and Expo, Stateful versus presentational components, React lifecycle methods, The folder structure

React Native Components: Basic components: View, Layouting, Touch events, Accessibility, Text, StatusBar, Images, and media.Basic user interaction: Button, TouchableOpacity, TouchableHighlight, TouchableWithoutFeedback. ActivityIndicator, Modal, ListView, ScrollView, RefreshControl, FlatList SectionList, VirtualizedList, Embedding web content, Handling user input,TextInput,Restricted choice inputs, Platform-dependent components, Detecting specific platform, Extensions, DatePickerIOS, Progress bars, Additional controls

Module III

Debugging and Testing React Native: Debugging your React Native apps, Remote debugging, Logging, Inspecting React Native components. Testing: Introduction to the Jest testing framework, Snapshot testing your React Native components, working with functions, mocking modules

Module IV

Platform APIs: Using Geolocation, Accessing the User’s Images and Camera, Storing Persistent Data with AsyncStorage, Modules and Native Code, Installing Third-Party Components with Native Code, Writing an Objective-C Native Module for iOS, writing a Java Native Module for Android, Cross-Platform Native Modules

Module V

Navigation and Structure in Larger Applications: The Flashcard Application: Project Structure, Application Screens, Reusable Components, Styles, Data Models, Using React-Navigation, Creating a StackNavigator, Using navigation.navigate to Transition between Screens, Configuring the Header with navigation options, implementing the RestState Management in Larger Applications, Using Redux to Manage State, Actions, Reducers, Connecting Redux, Persisting Data with AsyncStorage, Flexbox styling concepts and

techniques, Best practices and techniques for styling your React Native applications

References:

1. *Vladimir Novick,* React Native - Building Mobile Apps with JavaScript,*PACKT Publishing Limited, 2017, ISBN-13: 978-1787282537*
2. *Bonnie Eisenman*, Learning React Native, 2nd Edition, Building Native Mobile Apps with JavaScript, *O'Reilly Media Inc., 2017, ISBN-13: 978-9352136568*
3. *Eric Masiello, Jacob Friedmann,* Mastering React Native, *Packt Publishing, 2017, ISBN-13: 978-1785885785*
4. *Stan Bershadskiy, Crysfel Villa,* React Native Cookbook, *Packt Publishing, 2017, ISBN- 13: 978-1786462558*
5. *Emilio Rodriguez Martinez,* React Native Blueprints: Create eight exciting native cross-platform mobile applications with JavaScript, *Packt Publishing, 2017, ISBN-13: 978-178728809*

GEC3SD03E3 - IOS APP DEVELOPMENT – ADVANCED TECHNOLOGIES

Course Number: 3.3
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hr

Course Objective:

- To provide in-depth knowledge in all frameworks and classes to equip the students with advanced level expertise in iOS App development.
- To give hands on experience in working with advanced level core data handling, Map and geofencing, and notifications.
- It equips the students to develop apps using cloud and database servers; enables them to develop full-fledged iOS apps using latest techniques and methodologies.

Course Outcomes:

- This advanced programming course teaches students the skills necessary to develop applications for Apple mobile computing devices running the iOS operating system
- Gives students hands-on experience with the technologies, tools, and techniques used to develop mobile software solutions for business and entertainment.
- will build data-driven and location-aware applications and be introduced to a variety of object-oriented software design patterns common to mobile application development.
- Develop applications using Managed Objects and Serialization

- Deploy the application using Map and Geolocation components
- Differentiate Notifications in iOS and its implementation

Course Outline:

Module I

Introduction to Core Data, Building Managed Object Model, Setting Up Default Data, Displaying Managed Objects, Introducing the Fetched Results Controller, Adding, Editing, and Removing Managed Objects, Encoding, Decoding and Serialization, JSON Encoder.

Module II

Introduction, Creating and Interacting with Map View, Overlay Views, Annotations, Polyline, Polygon and Circles.
Location Manager and Permissions. Registering Geofences. Reacting to Geofence Events, Notifying User of Geofence Event.

Module III

Differences Between Local and Push Notifications, App Setup, Creating Development Push SSL, Certificate, Development Provisioning Profile, Custom Sound Preparation, Registering for Notifications, Scheduling Local Notifications, Receiving Notifications, Push Notification Server, Sending the Push Notifications

Module IV

CloudKit Basics, Setting Up a CloudKit Project, CloudKit Concepts, CloudKit Basic Operations, Subscriptions and Push, User Discovery and Management, Managing Data in the Dashboard.

Module V

Types of Extensions, Understanding Extensions, API Limitations, Creating Extensions, Today Extension, Sharing Code and Information between Host App and Extension, Apple Watch Extension, 3D Touch, Search, Siri, Touch ID, Certificate Creation, Apple Pay

References:

1. *Matt Neuburg* , iOS 11 Programming Fundamentals with Swift: Swift, Xcode, and Cocoa Basics, *O'Reilly Media; 1 edition, 2017, ISBN-13: 978-1491999318 2.Christian Keur, Aaron Hillegass,* iOS Programming: The Big Nerd Ranch Guide (6th Edition), *Big Nerd Ranch Guides, 2017, ISBN- 13: 978-0134682334.*
3. *Donny Wals* , Mastering iOS 11 Programming , Packt Publishing Limited; 2nd Revised edition, *2017, ISBN-13: 978-1788398237*
4. *Matt Neuburg,* Programming iOS 11: Dive Deep into Views, View Controllers, and Frameworks, *1st Edition, O'Reilly Media, January 2018, ISBN-*

13: 978-1491999226

5. *raywenderlich.com Team, Janie Clayton, Alexis Gallagher, Matt Galloway, Ben Morrow, Cosmin Pupaza, Steven van Impe,* Swift Apprentice: Beginning Programming with Swift 4, *Third Edition, Razeware LLC, 2017, ISBN-13: 978-1942878438*
6. *Kyle Richter, Joe Keeley,* Mastering iOS Frameworks: Beyond the Basics, *Addison- Wesley Professional, 2015, ISBN-13 : 9780134052526*

GEC3SD03E4- WATCH OS PROGRAMMING

Course Number: 3.3
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hr

Course Objective:

- To identify apple smart watch architecture and familiarize its programming.
- To give hands-on experience in working with programming environment for wearable devices in iOS.
- To introduce the latest trends and technologies in Apple watch programming.

Course Outcome:

- Experience focuses on quick actions that achieve useful tasks through brief, punctuated interactions.
- Understand Apple Watch and its lifecycle
- learn how to develop and deploy a native application for this hugely popular smart watch using watchOS 7 and SwiftUI.
- Understand the user controls and its implementation

Course Outline:

Module I

WatchKit App Architecture, Interaction between Apple Watch and iPhone, Types of Apple Watch Applications, WatchKit App Lifecycle.

Module II

Interface Controllers and Storyboard, Life Cycle of an Interface Controller, Navigation between Interface Controllers, Hierarchical Navigation, Page-Based Navigation, Passing Data between Interface Controllers.

Module III

Button, Switch, Slider, Labels, Images, Tables, Text, Emojis, Laying Out Controls, Force Touch, Context Menu

Module IV

Localization-User Interface, Strings, Date Control.

Communicating between the WatchKit App and the Extension-Location Data, Displaying Maps, Accessing Web Services, Sharing Data.

Module V

Notifications-Types of Notifications on Apple Watch and its Implementation.

Glances-Implementation, Customization and Testing of Glance and Glance

Updation. Advanced Technology: TV OS.

References:

1. *raywenderlich.com Team, Ehab Amer, Scott Atkinson, Soheil Azarpour, Matthew Morey, Ben Morrow, Audrey Tam, Jack Wu,* watchOS by Tutorials: Making Apple Watch Apps with watchOS 4 and Swift 4, *Razeware LLC, 3rd edition, 2017, ISBN-13: 978-1942878452*
2. *Scott La Counte* , A Beginners Guide to Apple Watch Series 2 and Watchos *Createspace Independent Publications, 2016, ISBN-13: 978-1537740546.*
3. *Steven F. Daniel,* Apple Watch App Development, *Packt Publishing, 2016, ISBN-13: 978-1785886362*
4. *Jeff Kelley,* Developing for Apple Watch: Create Native watchOS Apps with the WatchKit SDK, *Pragmatic Bookshelf; 2 edition, 2016 , ISBN-13: 978-1680501339*
5. *Wei-Meng Lee,* Learning WatchKit Programming: A Hands-On Guide to Creating watchOS 2 Applications, *Addison-Wesley Professional, 2 edition, 2015, ISBN-13: 978-0134398983*
6. *Christian Keur, Aaron Hillegass,* iOS Programming: The Big Nerd Ranch Guide (6th Edition), *Big Nerd Ranch Guides, 2017, ISBN-13: 978-0134682334.*

GEC3SD03E5 - HEALTHKIT AND HOMEKIT PROGRAMMING

Course Number: 3.3
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hr

Course Objective:

- To understand and learn the Apple Healthkit and Homekit Tools.
- To learn how to integrate iOS HealthKit into health and fitness apps for iOS and watchOS.

- To provide basic knowledge required to connect iPhone/iPad and control stuff like garage-door openers, lights, security camerasetc. using Homekit.

Course Outcomes:

- Apply reading and writing health kit data in projects
- Understand advanced health kit functions like threading
- Understand HomeKit and its application
- Apply Observable and Delegates in the HomeKit framework

Course Outline

Module I

Introduction to HealthKit App, Adding HealthKit to a Project, Requesting Permission for Health Data Reading, Characteristics of HealthKit Data, Reading and Writing Basic HealthKit Data, Reading and Writing Complex HealthKit Data.

Module II

HealthKit and Privacy, Benefits from Adopting HealthKit, HealthKit's Design Philosophy, Setting Up HealthKit, Adding Samples to the HealthKit Store, Managing Sample Sizes, Working with Units, Threading, Adding Digital Signatures, HeathKit Classes, Development of HealthKit Apps.

Module III

Introduction to HomeKit, Enabling HomeKit, Getting the Home Layout, Getting the Home Manager Object, Getting the Primary Home and Collection of Homes, Getting the Rooms in a Home, Getting the Accessories in a Room, Getting Accessories in a Home, Rules for Naming Objects, Creating Homes, Adding a Room to a Home, Discovering Accessories, Adding Accessories to Homes and Rooms, Changing Names of Accessories, Adding Bridges to Homes and Rooms, Creating Zones.

Module IV

About HomeKit Delegation Methods, Observing Changes to the Collection of Homes, Observing Changes to Individual Homes, Observing Changes to Accessories, Getting Services and Their Properties, Changing Names of Services, Accessing Values of Characteristics, Creating Service Groups.

Module V

Adding Accessories, Adding Services to Accessories, Adding Characteristics to Services, Adding Accessories to a Home, Controlling Accessories, Adding Bridges, Adding Bridges to a Home, Controlling Accessories Behind a Bridge, Testing Multiple iOS Devices and Users**,** Creating Write Actions, Creating and Executing Action Sets, Creating and Enabling Timer Triggers, Adding and Removing Users, Getting User Names.

References:

1. *Jesse Feiler,* Learn Apple HomeKit on iOS: A Home Automation Guide for Developers, Designers, and Homeowners, *Apress, 2016, ISBN-13: 978-1484215289.*

2. *Michael Galeso ,* Apple Homekit: An Easy Guide to the Best Features, *CreateSpace Independent Publishing Platform, October 2016, ISBN-13: 978-1539660156* 3.*Christian Keur, Aaron Hillegass,* iOS Programming: The Big Nerd Ranch Guide (6th Edition), *Big Nerd Ranch Guides, 2017, ISBN-13: 978-0134682334.*
4.Vandad Nahavandipoor, iOS 11 Swift Programming Cookbook: Solutions and Examples for iOS Apps, *O'Reilly Media; 1 edition, 2017, ISBN-13: 978-1491992470.*
5.Vandad Nahavandipoor, iOS 8 Swift Programming Cookbook, *O'Reilly Media, November 2014, ISBN-13: 978-1491908693*

6.Bakir, Ahmed, de la Torriente, Manny, Chesler, Gheorghe, Program the Internet of Things with Swift for iOS, *Apress, 2016, ISBN-13: 978-1-4842-1194-6,*

GEC3SD03E6- RETAIL APP DEVELOPMENT FRAMEWORKS

Course Number: 3.3
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hr

Course Objective:

To understand Apple wallet and payment gateways in application development. Retail framework serves as the backbone of banking and financial applications. The subject introduces the retail application framework and also provides in-depth knowledge in retail application development.

Course Outcomes :

- Analyze the wallet ecosystem
- Understand about Pass system and its features Understand
- Analyze different features of Pass Analyze
- Apply Touch Id and Payment Token (Apply). Apply
- Understand the Bluetooth features of the kit

Course Outline

Module I

Introducing Wallet, Wallet Ecosystem Design, Passes and Stages of Pass Lifecycle, PassKit Framework.

Module II

Creating and Populating the Pass Package, Setting the Pass Type Identifier and Team ID,

Signing and Compressing the Pass, Changing the Offer, Viewing the Pass, Pass Design and Creation, Pass Styles, Designing Passes for Apple Watch, Fields Contain Text Displayed onthe Pass, Fields Support Formatting, Barcodes Link Passes, Customization of Pass Colors, Handling Images, Relevance Information Displays Passes on the Lock Screen, Passes Support Localization, Passes Are Cryptographically Signed and Compressed, Debugging and distribution of Passes.

Module III

Updating passes, Overview of the Communication, Interacting with Passes, Enabling Passbook Capabilities, Accessing, Getting, Reading, Adding, Changing and Removing a Pass.

Module IV

Using Touch ID with Keychain and Local Authentication, Security Considerations, LAContext. Apple Pay or In-App Purchase, Prerequisites, App Review Guidelines, Presenting the Apple Pay button, Presenting the Payment Sheet, The Payment Token, Supported Transaction Types, Best Practices.

Module V

iBeacon Software: Core Location APIs, Broadcasting an iBeacon, Physical Limitations,Best practices.

References:

1. *Ernest Bruce,* Apple Pay Essentials, *Packt Publishing, February 2016, ISBN-13: 9781785886386.*
2. *Allister Banks, Charles S. Edge,* Learning iOS Security, *Packt Publishing Limited, 2015, ISBN-13: 978-1783551743*
3. *Christian Keur, Aaron Hillegass,* iOS Programming: The Big Nerd Ranch Guide (6th Edition), *Big Nerd Ranch Guides, 2017, ISBN-13: 978-0134682334.*
4. *Matthew S. Gast,* Building Applications with iBeacon: Proximity and Location Services with Bluetooth Low Energy 1st Edition, *O'Reilly Media; 1 edition (October 12, 2014), ISBN- 13: 978-1491904572,*
5. *Craig Gilchrist,* Learning iBeacon, *Packt Publishing, November, 2014, ISBN-13: 978-1784397128*
6. *Kyle Richter, Joe Keeley,* Mastering iOS Frameworks: Beyond the Basics, *Addison-Wesley Professional, 2015, ISBN-13: 9780134052526*

SDC3SD10 - MACHINE LEARNING

Course Number: 3.4
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hrs

Course Objectives

- To introduce concepts of Artificial Intelligence and Machine Learning.
- To get an idea about neural networks, genetic programming etc
- To kno about game palying

Course Outcomes:

- Develop an appreciation for what is involved in Learning models from data
- Understand a wide variety of learning algorithms
- Understand how to evaluate models generated from data
- Apply the algorithms to a real problem, optimize the models learned and report on the expected accuracy that can be achieved by applying the models

Course Outline:

Module I

Introduction - Artificial Intelligence - problems, scope and applications, problem space and search - production system- characteristics - the predicate calculus, inference rules, structures and strategies for state space search, strategies for space search, using state space to represent reasoning with the predicate calculus.

Module II

Heuristics Search: control and implementation of state space search, generate and test, hill climbing, Best– first search, problem reduction, constraint satisfaction, means-ends analysis, heuristic in games, complexity issues

Module III

Knowledge representation issues, representation and mappings, representing simple facts in logic, representing instances and ISA relationships, computable functions and predicates, resolution, natural deduction, knowledge representation using rules, logic programming, forward versus backward reasoning, symbolic reasoning under uncertainty- non monotonic reasoning, depth first search, breadth first search.

Module IV

Game playing – the Minimax search procedure, adding Alpha-beta cutoffs, additional refinement, iterative deepening, planning system and its components, understanding, understanding as constrained satisfaction

Slot and filler structures: semantic nets, frames, conceptual dependency, scripts. Definition and characteristics of expert system, representing and using domain knowledge, expert system shells. Knowledge engineering, knowledge acquisition, expert system life cycle & expert system tools, MYCIN & DENDRAL examples of expert system.

Module V

Machine learning – rote learning, learning by taking advice, learning in problem solving, learning from examples, explanation based learning, analogy, formal learning theory, connectionist models - hopfield networks, learning inneural networks, back propagation, the genetic algorithm, classifier systems and genetic programming, artificial life and society based learning.

References:

2. Elaine Rich, Kevin Knight and Shivshankar B. Nair, Artificial Intelligence , 3 rd Edition, Tata – McGraw Hill, New Delhi, ISBN: 0070087709.
3. V S Janakiraman, K Sarukesi and P Gopalakrishnan, Foundations of ArtificialIntelligence and Expert System , Macmillan India Limited, ISBN: 0333926250.
4. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach , 3 rdEdition, Prentice Hall, ISBN: 0136042597.
5. G. F. Luger and W.A Stubblefield, Artificial Intelligence – Structures and Strategies for Complex Problem Solving , Addison-Wesley, 6 th Edition, ISBN: 9780321545893.
6. P. H. Winston, Artificial Intelligence , Addison-Wesley, 3 rd Edition, ISBN: 0201533774.
7. Nils J. Nilsson, Artificial Intelligence, A New Synthesis , 1 st Edition, MorganKaufmann Publishers, Inc,

SDC3SD11- IOS APP DEVELOPMENT- FUNDAMENTALS

Course Number: 3.5
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hrs

Course Objectives

- To introduce the iOS application development environment, fundamental frameworks, and foundation classes.
- To provide a basic level knowledge and hands on experience in User Interface Views, controllers and design patterns. The course enables the students to acquire a beginner level experience in iOS App development.

Course Outcome

Students will able to:

- Describe Android platform, Architecture and features
- Apply the Cocoa framework for iOS development
- Understand the fundamentals of iOS.
- Use Intent , Broadcast receivers and Internet services in Android App.
- Design and implement Database Application and Content providers.
- Evaluate and create Story Board, MVC, Protocols and Delegates, View System, Controllers, and devise solution based on it
- Design and create projects based on multi-scene storyboards, toolbars, and pickers

Course Outline

Module I
Introduction to iOS Platform, iOS Devices and the Apple Developer Tools, UI Guidelines to IOS, Introduction to Xcode and the iOS Simulator, Exploring the iOS Technology Layers, iOS Application Life Cycle, Cocoa Fundamentals-Application Classes, Data Type Classes and Interface Classes, Foundation Framework, iOS Coding Standards.

Module II
Introduction to Interface Builder and Storyboard, Creating User Interfaces, Autolayout, Customizing the Interface Appearance, Connecting to Code, Outlets and Actions, Introduction to MVC Design Pattern, Implementing MVC with Xcode, Single View Application Template, Building Applications, Developer Guidelines.

Module III
Protocols and Delegates, Working with labels, Basic User Input and Output Using Text Fields, Text Views, Buttons, Image Views, Animation, Sliders, Steppers, Search Box, Switches and Segmented Controls.

Module IV
Web Views, Scrolling Views, Alert Controllers, System Sound Services, Vibrations, Tables and Split View Controllers and Collection View.

Module V
Multi-scene Storyboard, Passing Data between Scenes, Segues, PopOvers, Understanding the Role of Toolbars, Exploring Pickers.

REFERENCES
1. Matt Neuberg, iOS 11 Programming Fundamentals with Swift, O′Reilly, 2017, ISBN- 13: 978-1491999318
2. Serhan Yamacli , Beginner's Guide to IOS 11 App Development Using Swift 4: Xcode, Swift and App Design Fundamentals, Createspace Independent Publishing Platform; 1 edition, 2017, ISBN-13: 978-1977891754

3. Donny Wals, Mastering iOS 11 Programming, Packt Publishing Limited, 2017, ISBN-13: 978-1788398237
4. Molly K. Maskrey, Beginning iPhone Development with Swift 4: Exploring the iOS SDK , Apress; 4th ed. edition (27 November 2017), ISBN-13: 978-1484230718
5. Christian Keur and Aaron Hillegass, iOS Programming: The Big Nerd Ranch Guide (6th Edition), Big Nerd Ranch Guides; 6th Edition, 2017, ISBN-13: 978-0134682334.
6. Web Reference: https://developer.apple.com/

GEC3SD04EX - ELECTIVES II

Course Number: 3.6
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hr

LIST OF ELECTIVES

GEC3SD04E7 -Programming with Objective-C
GEC3SD04E8 - Programming with Python
GEC3SD04E9 - Data Analytics
GEC3SD04E10 -Kotlin Programming
GEC3SD04E11- Internet of Things (IoT)
GEC3SD04E12- Low Code Platform

GEC3SD04E7 -PROGRAMMING WITH OBJECTIVE-C

Course Number: 3.6
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hr

Course Objectives

- To provide a complete theoretical knowledge and practical exposure to concepts behind Objective-C.
- To give hands on experience in iOS app development using Objective-C
- To give awareness in application porting from Objective-C to Swift in iOS app development

Course Outcomes

- Understand the variables and data-structures in Objective C
- Apply Object-Oriented features
- Evaluate advanced-structures and Object manipulation
- LO4 Deploy applications using Error Handling and MVC
- LO5 Analyze Object features and Data Management

Module I

The Objective-C Language: Introduction, Programming Environment, Variables, Data Types and Expressions, Numbers, Pre-processor, Enumerated data types, Arrays, Blocks, Structures, Pointers and Addresses, Looping, Decision Making, Functions, Type conversions, Bitwise operators, Scope.

Module II

Object-oriented concepts: Classes, Objects, Methods and Messages, Properties, Object initialization, Inheritance, Polymorphism, Dynamic Typing, Dynamic Binding, Categories and Protocols, Class Extensions

Module III

The Foundation Framework: Introduction to the Foundation Framework, Numbers, Strings, and Collection, Working with Files, Memory Management and Automatic Reference Counting, Copying Objects, Archiving, Collection Classes, Callbacks,

Module IV

Event-Driven Applications: GUI-based applications, Model-View-Controller, application delegate. Errors: Error Objects, Domains, and Codes, Using and Creating Error Objects, and Exceptions: Exceptions and the Cocoa Frameworks, Handling Exceptions, Throwing Exceptions, Nested Exception Handlers, Predefined Exceptions, Controlling a Program's Response to Exceptions.

Module V

The Objective-C Runtime: Basics, Dynamic method lookup and execution, Management of classes and inheritance hierarchies, Auto-release pool mechanism, Object Creation And Storage, Key-Value coding, Key-Value Observing and its working, Preventing Memory Leaks

References:

1. *Gibson Tang, Maxim Vasilkov,* Objective-C Memory Management Essentials, *Packt Publishing, March 2015, ISBN-13: 978-1849697125*
2. *Stephen G. Kochan,* Programming in Objective-C, Developer's Library, *Addison-Wesley Professional; 6 edition (December 13, 2013), ISBN-13: 978-0321967602*
3. *Wiley,* Objective-C Programming: Introduction to Programming for iOS, *Wiley February 2014, ISBN-13: 978-0672334498*

4. *Jesse Feiler,* Sams Teach Yourself Objective-C in 24 Hours, *Second Edition,Sams, March 2014, ISBN-13: 978-0672334498*

5. *Carlos Oliveira,* Objective-C Programmer's Reference, *Apress, November 2013,ISBN-13: 978-1430259053*

6. *Mikey Ward , Aaron Hillegass,* Objective-C Programming: The Big Nerd Ranch Guide,*Big Nerd Ranch Guides, November 2013, ISBN-13: 978-0672334498*

GEC3SD04E8 - PROGRAMMING WITH PYTHON

Course Number: 3.6

Contact Hours per Week: 4

Number of Credits: 4

Number of Contact Hours : 60 Hr

Course Objectives

To provide a complete programming knowledge with hands on experience to work with Python and related technologies.

The subject enables the students to understand and learn object-oriented programming using python. Also, it provides hands-on experience in data analytics using python.

Course Outcomes:

- Understand the fundamentals of Python and its environment
- Understand syntax and semantics and advanced python integration
- Understand advanced Class and Object-Oriented features and its implementation
- Develop application using statistical and analytical features
- Design solutions based on visualization

Course Outline:

Module I

Introduction to Python: Scripting language, Downside, Uses, applications and Python Trade-off, Technical Strength. Running Python Programs, Installation & Configurations, Windows Launcher, Introducing Python Object Types: Numeric Type, The Dynamic Typing Interlude, String Fundamentals, List and Dictionaries, Tuples, Files and Others.

Module II

Introducing Python Statements: Assignments, Expressions, and Prints, if and Syntax Rules, while and for loops, Iterations and Comprehensions, Documentation Interlude. Functions and Generators: Function Basic, Scope, Arguments. Advance Functions, Comprehensions and Generations, the Benchmarking Interlude.

Modules: The Big Picture. Module Coding Basics: Creation, Usage, Namespaces, Reload. Module Package: Setting, File, Import, Namespaces. Advanced Module Topics, OOP: The Big Picture. Class Coding Basics: Class, Object, Instance, Inheritance, Constructor, Testing, Behaviour methods, Operator Overloading, Subclassing, Customized constructor, Introspection Tools, Store Objects in database, Class coding details.

Module III

Operator Overloading, Designing with classes, Advanced Class Topics, Exception Basics, Exception Details: try, except, else, finally, raise, assert. Exception Objects: Exception Class, Hierarchy, Exception Details, and methods. Designing With Exceptions: Nesting, Idioms, Design Tips and Gotchas, Advanced Topics: Unicode and Byte Strings, Managed Attributes, Decorators, Metaclasses.

Module IV

Introduction to Data Analysis, Installation & Setup, Libraries. Interactive Computing and Development Environment: IPython Basics, Command History, Interacting with OS, Software Development Tools. NumPy Basics: Arrays and Vectorized Computation. Getting Started with pandas: Introduction, Functionality, Descriptive Statistics,

Handling missing data, Hierarchical indexing. Data Loading, Storage, and File Formats: Read and Write text, Binary Data Formats, Interacting with Web API and Databases.

Module V

Data Wrangling Clean, Transform, Merge, Reshape: Combining and Merging Data Sets, Reshaping and Pivoting, Data Transformation, and String Manipulation. Plotting and Visualization: A Brief matplotlib API Primer, Plotting Functions in pandas, Plotting Maps: Visualizing Haiti Earthquake Crisis Data, Python Visualization Tool Ecosystem. Data Aggregation and Group Operations: GroupBy Mechanics, Data Aggregation, Group-wise Operations and Transformations, Pivot Tables and Cross-Tabulation. Time Series: Basics, Date Ranges, Frequencies, and Shifting, Time Zone Handling, Periods and Period Arithmetic, Resampling and Frequency Conversion, Time Series Plotting. Financial and Economic Data Applications: Data Munging Topics , Group Transforms and Analysis.

References:

1. *Wes McKinney*, Python For Data Analysis, *O'Reilly Publishers, 2017, ISBN-13: 978- 1491957660*
2. *Mark Lutz* , Learning Python, *O'Reilly Publishers, 2013, ISBN-13: 978-1449355739*
3. *Jake VanderPlas,* Python Data Science Handbook: Essential Tools for Working with Data,*Shroff/O'Reilly Publishers, 2016, ISBN-13: 978-9352134915*
4. *Rick van Hattem,* Mastering Python, *Packt Publishing Limited, 2016,ISBN-13: 978- 1785289729*
5. *Armando Fandango,* Python Data Analysis, *Packt Publishing Limited, 2017,ISBN-13: 978-1787127487*

GEC3SD04E9 - DATA ANALYTICS

Course Number: 3.6

Contact Hours per Week: 4

Number of Credits: 4

Number of Contact Hours : 60 Hr

Course Objectives

- To introduce the data science platform and also to provide in-depth knowledge in using different tools for data analysis.
- To provide a complete theoretical knowledge of data pre-processing and analysis.
- To give practical awareness to data analytics using a Tool

Course Outcomes:

- Understand the fundamentals of data management in data mining & analysis
- Apply Descriptive statistics and sampling techniques
- Evaluate regression analysis and the tools used for it
- Understand the commonly used data analytics models
- Develop solutions using the R platform

Course Outline:

Module I

Data & Data Science Process: - overview of a data science project. Data Storage Concepts: Files, Database, Internet Repos, Steams, Blobs. Exploring Data: understand types of data (time series, numerical, text, etc), Cleaning input data: Stemming,

Tokenization, Stop word removal, Length Based selection, replacing nulls & missing data correction.

Module II

Descriptive Statistics: Uncertainty, probability, variance, sampling, randomness, elements of data analysis. Describing and displaying data, correlation. Joint probability distribution, Conditional probability distribution, prior and posterior probability distribution, Time series data. Sampling techniques & Measurement of Centre Values

Module III

Regression Analysis, Multicollinearity, Correlation analysis, Rank correlation coefficient, multiple correlation, Least square, Curve fitting and goodness of fit, Time Series:
ANOVA, Latin square, Factorial Design.

Module IV

Bayesian, Regression Model, Random Forest, K nearest Neighbor, Association Rule Mining, Decision Trees, Pruning, Text Mining: Clustering, Market Basket, Support Vector

Machine, Fuzzy Logic, Neural Network, Back Propagation Deep Learning.

Module V

Introduction to R: Installation , RStudio, Objects, Scripts, Vectors , matrices, Generating Data, Objects , Files, Packages, Read and process data, Special data types, Visualizing data, Basic statistics,Linear modelling, functions ,lists, loops, project management and workflows.

References:

1. *Anil Maheshwari*, Data Analytics, *McGraw HillEducation, 2017, ISBN-13: 978-9352604180*
2. *Herbert Jones*, Data Analytics: An Essential Beginner's Guide to Data Mining, Data Collection, Big Data Analytics for Business, and Business Intelligence Concepts, *Createspace Independent Publishing, 2018, ISBN-13: 978-1985097971*
3. *Paul Kinley*, Data Analytics for Beginners: Basic Guide to Master Data Analytics,*Createspace Independent Publishing, 2016, ISBN-13: 978-1539896739*

Remko Duursma, A Learning Guide to R, *Jeff Powell & Glenn Stone*

4. *Torsten Hothorn, Brian S. Everitt*, A Handbook of Statistical Analyses, *Chapman and Hall/CRC, 2009, ISBN-13: 978-1420079333*
5. *John Maindonald, W. John Braun*, Data Analysis and Graphics Using R: An Example-Based Approach, *Cambridge University Press; 3 edition, 2010, ISBN-13: 978- 0521762939*

GEC3SD04E10 -KOTLIN PROGRAMMING

Course Number: 3.6

Contact Hours per Week: 4

Number of Credits: 4

Number of Contact Hours : 60 Hr

Course Objectives

This course introduces Kotlin programming. The students assess Kotlin's development environment, its data structures, operators, control statements, Objectoriented concepts, layouts, events, menus, navigation, intent, broadcasts, threads, SQLite, media frameworks, maps and fingerprint authentications to get an in-depth understanding of Kotlin programming.

Course Outcomes

- Understand the fundamentals of Kotlin
- Understand the variables, operators, control flow, and lambda
- Analyze Object Oriented capabilities of Kotlin
- Understand and analyze the UI components in Kotlin
- Design solutions based on advanced features in Kotlin

Module I

Kotlin: - Introduction, Advantages & Disadvantages, Kotlin for Android, Kotlin for Server Side Development. Setting up Kotlin Environment (Android). Environment Familiarization: - User Interfaces, AVD, Emulator, Testing, Android Studio Code Editor. Android Architecture: Stack, Kernel, Runtime, Libraries.

Module II

Activities, Intents, Broadcast (Intents & Receivers), Service, Content Providers, Manifest, Resource, Gradle, Resource & Context. Kotlin: - Kotlin & Java, Java to Kotlin, Kotlin & Android Studio, Semi-Colon in Kotlin. Kotlin Datatypes , Kotlin Variables (Mutable & Immutable), Variable Declaration, Type Annotation & Inference, Nullable, Safe Call Operator, Not Null Assertion , Nullable Type and let Function, Elvis Operator, Type Casting & Checking. Operators & Expression: - Expression, Assignment, Arithmetic, Augmented Assignment, Increment & Decrement, Equality, Boolean Logical, Range & Bitwise. Flow Control: - for – in, while, do – while, Break & Continue, Break & Continue Labels, if, if – else, if – else – if, when. Functions & Lambda: - Type, Return Values, Parameters, Variable Number, Lambda, Higher Order.

Module III

Basics Of OOP‟s, Inheritance & Sub classing, Activity Life Cycles, Activity State Changes, Save & Restore State, Views, View Groups & Layouts, Layout Editor, Constraint Layout, Constraint Layout Chain, Ratios, Constraint Set, Android Event Handling, Multi Touch Event Handling, Gesture Detector, Custom Gesture, Fragments, Menus & Overflow Menus, Animation & Transitions, Floating Action Button & Snackbar, Tab Layout, Recycler View & Card View, App Bar, Collapsing Toolbar, Navigation Drawer.

Module IV

Master – Details Flow, Android Intents , Broadcast Intent & Receivers, Thread & Async Task, Started & Bound Service, Remote Bound Service, Notification , Multi Windows Support, Split Screen & Freeform, SQLite, TableLayout, TableRow, Content Providers, Cloud Storage & Access Framework, Video View, MediaController, Picture in Picture, Video Recording & Image Capture, Runtime Permission Requests.

Module V

Google Maps API, Android Printing Framework, Custom Document Printing, Android App Links, Instant Apps , Android Studio Profiler , Android Fingerprint Authentication, Handling Different Devices & Displays, Signing Android App for Release, Gradle in android Studio.

REFERENCES

1. Antonio Leiva , Kotlin for Android Developers: Learn Kotlin while developing an Android App, CreateSpace Independent Publishing, 2016, ISBN-13: 978- 1530075614
2. Marcin Moskala, Igor Wojda , Android Development with Kotlin, Packt Publishing, 2017, ISBN-13: 978-1787123687
3. Dmitry Jemerov, Svetlana Isakova, Kotlin in Action, Manning Publications, 2017, ISBN- 13: 978-1617293290
4. Stephen Samuel, Stefan Bocutiu , Programming Kotlin, Packt Publishing Limited, 2017, ISBN-13: 978-1787126367
5. Milos Vasic, Mastering Android Development with Kotlin, Packt Publishing Limited, 2017, ISBN-13: 978-1788473699

GEC3SD04E11- INTERNET OF THINGS (IOT)

Course Number: 3.6
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hr

Course Objectives:

- To familiarize with IoT and its applications in Mobile app development.
- To provide in-depth knowledge in IoT and its related architectures, methodologies.
- To enable the students to learn how to apply this knowledge in real world scenarios.

Course Outcomes:

- Understand the fundamentals of IoT, its application and commonly used tools
- Analyze the Raspberry Pi tool and its features
- Understand the Constraint Application Protocol (CoAP) and its use
- Understand the XMPP protocol and its features
- Analyze and evaluate the protocols and architectures used

Course Outline:

Module I

IoT Overview, IoT Hardware, IoT Software, Technologies & Protocols, Common uses, IoT – Media, Marketing, Advertising. IoT – Environmental Monitoring, IoT– Manufacturing, IoT – Energy, Healthcare, Building, Transportation, Education, Government, Law & Consumer Applications. IoT Thingworx, IoT Cisco virtualized packet core, IoT Salesforce, IoT GE PREDIX, Eclipse IoT, IoT CONTIKI, IoT Identity Protection, IoT Liability.

Module II

Raspberry Pi: - Creating Sensor Project, actuator project, controller, camera. HTTP: - Http Support to sensor, actuator & controller. UPnP: - Introduction, Device description document, service description document, web interface, UPnP interface, Still image service, using camera.

Module III

CoAP:- HTTP Binary, Add CoAP to sensor, Add CoAP to actuator, Using CoAP in controller. MQTT: - Support for Sensor, support for actuator, support for controller.

Module IV

XMPP: - Basics, Support to a thing, additional layer of security, support to actuator, support to camera, support to controller, Connecting it all together.

Module V

IoT Service Platform: - Selection of Platform, clayster platform, interfacing using xmpp, creating control application. Protocol Gateways: - protocol bridging, abstraction model, clayster abstraction model, CoAP gateway architecture. Security& Interoperability: - Understanding risk, modes of attack, tools for security, need for interoperability.

References:

1. *Peter Waher*, Learning Internet of Things, *Packt Publishing, 2015, ISBN-13: 978- 1783553532*
2. *Adrian Mcewen,Hakin C assimally*, Designing The Internet of Things, *John Wiley and Sons, 2015, ISBN-13: 978-8126556861*
3. *Arsheep Bahga, Vijay Madisetti,* internet of Things: A Hands-On Approach, *Orient Blackswan Private Limited, 2015, ISBN-13: 978-81737195*
4. *Stephen Chin, James Weaver,* Raspberry Piwith Java: Programming the Internet of Things (IoT), *McGraw-Hill Education, 2015, ISBN-13: 978-0071842013*
5. *Madhur Bhargava*, IoT Projects with Bluetooth Low Energy: Harness the powerof connected things, *Packt Publishing, 2017, ISBN-13: 978-1788399449*

GEC3SD04E12- LOW CODE PLATFORM

Course Number: 3.6

Contact Hours per Week: 4

Number of Credits: 4

Number of Contact Hours : 60 Hr

Course Objectives:

- To introduce Low code platform and its importance in industry as a platform for developers with little technical knowledge.
- To familiarize with Salesforce as a low code tool for app development.
- To give hands-on experience in using salesforce with the advanced tools to create different mobile apps

Course Outcomes:

- Understand the fundamentals of the Low Code Platform and its features
- Understand different data models and their functions
- Apply workflow, verification and error handling techniques
- Apply visual workflow into projects
- Deploy the application using security and permissions

Course Outline

Module I

Introduction- Low-code development platform, History of Low-code Development Platforms, Low-Code App Development, Background of Low-code. Benefits of a low-code development platform. Selection of Low-code Platform (speed/stability). Building an Application Without Coding, Breaking Down the Low-Code Landscape, Low-code Myths, Featured Low-Code Development Platforms Reviews: (Appian, PowerApps, Mendix, outsystems, Google AppMaker, QuickBase, TrackvIA, Zoho Creator, Salesforce App CLoud)

Module II

Data model: Objects, Fields, and Relationships, Formula Functions, Salesforce.com Development Using Functions, All About Formula Fields, Database Essentials, Database Security.

Module III

Workflow and Rules, Building a Workflow Rule and Actions: Step by Step, Workflow Rules: Real-World Examples, Enforcing Business Rules with Salesforce.com Validation Rules, Building an Effective Validation Rule, Security Model Impact, Error Condition Grouping, Existing Error Conditions, Unavailable Functions, Building Effective Approval Processes for the business, Entitlements, and Milestones to Drive Case Automation, Creating an Entitlement Record, Case Page Layout Considerations

Module IV

Visual Workflow, User Input and Decision Points, The Cloud Flow Designer, Developing a Visual Workflow, Creating a Custom Button to Start the Flow, Develop Friendlier Solutions with Custom Settings, Streamline Process with Publisher Actions, Using Web-to-Lead, Customizing the Look and Feel of Salesforce.com, report Filters

Module V

Applying the Proper Security Model to Support Your Solutions, Understanding the

Salesforce.com Security Model, Object and Field Permissions, Record Access via Sharing, Practical Application of Security Elements, Field-Level Security, Object-Level Security, Using

Permission Sets and Validation Rules, Record Access Through Sharing, Object Permission and Record-Sharing Overlap

Managing Salesforce.com Data with Data Loader, environment management and Solution deployment, Sandbox Model

References:

1. *Philip Weinmeister* , Practical Salesforce.com Development Without Code, *Springer Nature 2015, ISBN-13: 978-1484200988*
2. *Jason Ouellette,* Development with the Force.com Platform: Building Business Applications in the Cloud, *Addison-Wesley Professional, 2011, ISBN-13: 978-0321767356*
3. *Richard Wentk, Justin Davis,* Teach Yourself VISUALLY Salesforce.com, *John Wiley & Sons, 2013, ISBN-13: 978-1118551592*
4. *Rakesh Gupta,* Learning Salesforce Visual Workflow, *Packt Publishing Limited, 2015, , ISBN-13: 978-1785289835*
5. *David Taber,* Salesforce.com Secrets of Success: Best Practices for Growth and Profitability, *Prentice H*

SDC3SDL6 : LAB 6: ANDROID APP DEVELOPMENT - LAB

Course Number: 3.7
Contact Hours per Week: 4
Number of Credits: 4
Number of Contact Hours : 60 Hr

Course Objectives:

- To provides hands on experience in Advanced Android concepts. The Lab is centered on the concepts covered on the subject Android App Development

Course Outcome

- Experiment on Integrated Development Environment for advanced Android Application Development.
- Design and Implement User Interfaces and Layouts of Android App in advanced level.
- Use Intents for activity and broadcasting data in Android App.
- Design and Implement Database Application and Content Providers.
- Experiment with email, Camera and Location Based service and animations.

- Develop Android App with Security features

Course Outline

Lab Exercises:

1. Advanced Intent and Broadcast Receivers
2. Content Providers and MIME Types
3. Services
4. Multimedia : Sound and Video
5. Location : GPS and Internet, Proximity Alerts
6. Google Map : Overlays , Geocoder.
7. Web Views
8. Sensors : Accelerometer, Motion , Compass, Orientation
9. Telephone : Managing Calls.
10. Contacts : Accessing Updating and Deleting.
11. Email : Sending and Receiving.
12. Camera : Surfaces Manipulation.
13. Animations : 2D and 3D, OpenGL.

SDC3SDL7: LAB 7: SOFTWARE LAB III (IOS AND SWIFT)

Course Number: 3.8
Contact Hours per Week:3
Number of Credits: 3
Number of Contact Hours : 45 Hr

Course Objectives:

The Software Lab III provides hand on experience in full stack development including front end, backend and third-party library integration, using XCode and Swift. It equips the students to migrate from a beginner to intermediate and then to a advanced level iOS programmer.The Lab is based on the practical oriented papers Programming with Swift, iOS App development,Fundamentals & iOS App development – Applied Methodologies

Course Outcomes:

- Develop projects using the iOS framework

- Develop solutions based on advanced iOS frameworks
- Deploy Swift based projects

Course Outline

1. iOS App development - Fundamentals: Implement iOS apps using followingbasic concepts:-

- Basic Concepts: Storyboard-Single, Multiscene, Code, Outlets, Segues and Patterns.
- MVC: Single View Application Template.
- Table View: Create, Edit, Delete, Cocoa Fundamentals
- Controls: Labels, Text Views, Buttons, ImageViews, Animations, Sliders, Steppers, SearchBox, Switch, Segmented Controls.
- Views: Web, Scrolling, Table, Split.
- Navigation and Tab Bar Controllers, Sharing Data between Tab Bar Scenes
- Table Views and Split View Controllers
- AutoLayout and Responsive User Interfaces
- Universal and Background-Ready Applications

2. iOS App development – Advanced & Applied Methodologies:

Advanced level expertise in iOS App development using following concepts and frameworks

- Data Storage: Files, Core Data, Key Chain Storage.
- Search Result Controller
- Touches and Gestures
- Orientation and Motion
- Image Picker and Media Player
- Address Book, E-mail and Browser
- Core Location and MapKit : Current location and Annotation
- Geofencing events handling
- Local and Push Notifications
- Working with CloudKit
- Familiarity with Extensions and 3D Touch

3. Swift: Familiarize the following swift programming concepts through iOS App development.

- Basics: Scope & Lifetime, Namespace, Module, Instances.
- Collections and Flow Control
- Functions and Closures.
- Optionals.
- Structure and classes

- Methods
- Protocols and Delegates
- Extensions
- Generics

REFERENCES

1. *Matt Neuberg,* iOS 11 Programming Fundamentals with Swift, O′Reilly, 2017, *ISBN- 13: 978-1491999318*
2. *Serhan Yamacli* , Beginner's Guide to IOS 11 App Development Using Swift 4: Xcode, Swift and App Design Fundamentals, *Createspace Independent Publishing Platform; 1 edition, 2017, ISBN-13: 978-1977891754*
3. *Jon Hoffman,* Mastering Swift 4 - Fourth Edition: An in-depth and comprehensive guide to modern programming techniques with Swift, *Packt publishing, 2017, ISBN-13: 978- 1788477802*
4. *Matt Neuburg,* Programming iOS 11: Dive Deep into Views, View Controllers, and Frameworks, *1st Edition, O'Reilly Media, January 2018, ISBN-13: 978-14919992265.Kyle Richter, Joe Keeley,* Mastering iOS Frameworks: Beyond the Basics, *Addison- Wesley Professional, 2015, ISBN-13 : 9780134052526*
6. *Donny Wals,* Mastering iOS 11 Programming - Second Edition: Build professional-grade iOS applications with Swift 4 and Xcode, *Packt publishing, 2017, ISBN-13: 978-178839823*

SEMESTER 4

SDC4SDTP TERM PAPER

Course Number: 4.1
Course Evaluation:100 % (Internal Component)
Credits and No. of Hours shared with Internship and Project

Course Objectives

- To enable the students to gain knowledge in any of the technically relevant current topics on CS/ IT/Research and acquire the confidence in presenting the topic and preparing a report.

Expected Outcome

- To enable the student to the techniques of literature survey
- To acquire the skill of presentation

Course Guidelines

The student is expected to do an extensive literature survey and analysis in an area related to computer science, chosen by him/her, under the supervision of a faculty member from the department. Evaluation of term paper should be done internally. A faculty member can be appointed as a guide/ supervisor.

The student has to choose an area for his/her work after due consultation and approval from the guide. The study should preferably result in a critical review of the present works /design ideas/ designs / algorithms/ theoretical contributions in the form of theorems and proofs/ new methods of proof/new techniques or heuristics with analytical studies/implementations and analysis of results.

Articles from ACM / IEEE / INFLIBNET Journals / Conference Proceedings and / or equivalent documents, standard textbooks and web based material, approved by the supervisor. Each student has to submit a seminar report, based on these papers; the report must not be reproduction of any original paper. The topic shall be presented in the class taking a duration of 15-20 minutes. The report and slides for presentation shall be prepared using free typesetting software such as LATEX. A committee consisting of three/four faculty members shall evaluate the seminar presentation.
Following guidelines shall be used for the assessment of Seminar.

Assessment Criteria	% of Marks
Scope and relevance of topic	20%
Quality of presentation slides	10%
Presentation skills	30%
Knowledge in the topic	20%
Report	20%

SDC4SDL8 INTERNSHIP & PROJECT

Course Number: 4.2
Contact Hours per Week: 30 (L+T+P = 0 +0+30)
Number of Credits: 30
Number of Contact Hours : 900 Hrs
Course Evaluation: 20 % (Internal Component) + 80% (External Component)

Course Objective

- To provide students with advanced exposure in the IT live projects of industrial sector.
- To develop quality software solutions applying software engineering principles and practices.
- Students are also encouraged to take up a research oriented work to formulate a research problem and produce results based on its implementation / simulation /experimental analysis.

Expected Outcome

- An industry ready software professional at the exit point
- Able to become a part of the industry through the whole semester internship in the industries
- Experience in handling Live projects

Course Guidelines

- The student shall undergo Industrial training and project of five month duration (From November to March). Each student is required to undergo an internship of 900 Hours in a IT Industry /Government IT sectors/ any other IT production company etc approved by the institution under thesupervision of a guide.

- Along with the internship each student shall do an individual Project Work. (The work of Internship and Project should be the same. Two separate works are not preferred as it reduces the depth of the project)

- Students can either take up a real-life application oriented project work or research and development project. The student can formulate a project problem with the help of her/his guide and submit the project proposal of the same. Approval of the project proposal is mandatory. If approved, the student can commence working on it, and complete it.

- At the end of the Internship he/she is required to prepare and submit a detailed report in the prescribed format with a diary of daily work done during the internship period. Monthly Reviews should be done by the department guides to check and rate the student's progress in the internship and project.
- The evaluation of the Internship and the Project Production work (along with a Viva-Voce pertaining to Internship work and Project) will be done by an External Examiner appointed by the University.

- Publishing the work in Conference Proceedings/ Journals with National/ International status with the consent of the guide will carry an additional weightage in the review process.

- An Internship Completion Certificate from the organization in which the student is undergoing internship shall also be submitted at the Department. If the student is doing a live project, then he/she shall be submit the Certificate of Appraisal from that organization/ institute / company etc to whom the project is to be done.

- All students shall submit 2 copies of the Internship and project report to the department before the commencement of the 4th semester university exam.

- If any student fails to do the Internship or Project or Both, his/her result will be withheld until the internship requirement is met within 12 months from the completion of the course.

- Students are also encouraged to present their work in IT fest /conference /workshop / journal with the assistance and guidance of the supervisor. This should pave as a good start for the student in the art of publishing /presenting his/her work to the outside world. Due weightage is accommodated for publications out of the project work in the final evaluation.

- The split up of the weightage is given below:

Distribution	Weightage
Content and relevance	30%
Live Project	10%
Dissertation / Report	10%
Presentation	20%
Viva	10%

Guidelines For Submission Of Report (Dissertation)

The distinguishing mark of a dissertation is an original contribution to knowledge. The dissertation is a formal document whose sole purpose is to prove that you have made an original contribution to knowledge. Failure to prove that you have made such a contribution generally leads to failure.

It is a test of the student's ability to undertake and complete a sustained piece of independent research and analysis / application development, and to write up the work in a coherent form according to the rules and conventions of the academic community. It should, moreover, have a logical and visible structure and development that should at all times assist the reader understands the arguments being presented. The layout and physical appearance of the dissertation should also conform to university standards. The dissertation is to be prepared in tex format (either Latex or using a suitable Windows Tex variant). The format of the report is included in Appendix A

APPENDIX A – Guidelines for Project Report & Layout

Cover Page and First Page

<TITLE>>

A PROJECT REPORT

SUBMITTED BY

<< NAME OF THE STUDENT >>

FOR THE AWARD OF THE

DEGREE OF MASTER OF VOCATION (M.VOC.)

IN

SOFTWARE DEVELOPMENT

(UNIVERSITY OF CALICUT)

<<COLLEGE EMBLEM>>

<<NAME OF THE DEPARTMENT>>

<<NAME OF THE INSTITUTION>>

(AFFILIATED TO THE UNIVERSITY OF CALICUT)

<<ADDRESS>>

MONTH YEAR

ACKNOWLEDGEMENT

Acknowledgement

I would like to thank ……..

Date: Name of the Student
Name of the student

Declaration by the Student

DECLARATION

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person or material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Date:

Signature:
Name:
Reg. No.:
Semester

Certificate from Guide & HoD

CERTIFICATE

This is to certify that the project report entitled <<TITLE HERE>> submitted by <<Name of the Student>> (Register Number: << Reg, No>>) to University of Calicut for the award of the degree of **Master of Vocation** (M.Voc.) in **Software Development** is a bonafide record of the project work and internship carried out by him/her under my supervision and guidance. The content of the report, in full or parts have not been submitted to any other Institute or University for the award of any other degree or diploma.

Signature

<<Name Project Guide>>
<<Designation>>

Signature

<<Name of the HOD>>
<<Designation>>

PROJECT EVALUATION REPORT OF THE EXAMINERS

Certified that the candidate was examined by us in the Project Viva Voce Examination held on .... and his/her Register Number is .....

Examiners:

1.

2.

Certificate of Internship from Industry

<< Company Details >>

CERTIFICATE

Page

Certificate of Appraisal from Organisation (if it is Live Project)

<< Organisation Details >>

CERTIFICATE

Contents

INDEX

Abstract

ABSTRACT

The abstract is a very brief summary of the report's contents. It should be about half a page long. Somebody unfamiliar with your project should have a good idea of what it's about having read the abstract alone and will know whether it will be of interest to them.

An abstract is a section at the beginning of a report, dissertation, thesis or paper summarizing the contents, significant results and conclusions of said document. It allows people to rapidly ascertain the documents purpose and if the document will be useful for them to read.

The abstract is not the same as a summary in the sense you are think of. It is a standalone account of the document giving purpose of the work (objectives), method used, scope of the work, results, conclusions and recommendations.

The abstract, although it comes first logistically, always should be written at the completion of the other chapters of the project report. It needs to be written last because it is the essence of your report, drawing information from all of the other sections of the report. It explains why the experiment was performed and what conclusions were drawn from the results obtained.

A general guideline for an abstract has five sections or areas of focus: why the experiment was onducted; the problem being addressed; what methods were used to solve the problem; the major results obtained; and the overall conclusions from the experiment as a whole.

Do not be misled, however, from this list into thinking that the abstract is a long section. In fact, it should be significantly shorter than all of the others. All of this information should be summarized in a clear but succinct manner if the abstract is going to be successful. An estimated average length for all of this information is only a single paragraph. Although this may seem as though it is a short length to contain all of the required information, it is necessary because it forces you to be accurate and yet compact, two essential qualities.

There are many useful web pages such as http:// writing2.richmond.edu /training/proiect / biologv/abslit.html to get few sample abstracts and the common mistakes we make when we write an abstract.

List of Tables

LIST OF TABLES

List of Figures

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

This is a general introduction about the project. Briefly summarize the relevance and background information about the proposed work. It should have the following sections.

1. About the proposed work, underlying technologies and techniques – outline briefly the technological /engineering /scientific / socioeconomic/relevance or significance of theproject work being reported.
2. Project Profile – Tile, Area and Category and other relevant information.
3. About the Organization – to whom the internship and Project Work is carried out.
4. Major Contributions of the Project Work.
5. Whether it is a live project or not and if yes, give details about them .
6. Are you decided to implement or host it ? if yes, give details.

CHAPTER 2

PROBLEM DEFINITION AND METHODOLOGY

This chapter is meant for giving a detailed description about the problem. This chapter includes the following subsections.

1. Problem Definition
2. Objectives
3. Motivation
4. Methodology
5. Scope

CHAPTER 3

REQUIREMENT ANALYSIS AND SPECIFICATION

This chapter includes the following subsections.

1. Requirement Analysis/Literature Review
2. Existing System
3. Proposed System
4. Requirement Specification
    a. Functional Requirements
    b. Non-functional Requirements
    c. Environmental Details (Hardware & Software Requirements)
5. Feasibility Study
    a. Technical Feasibility
    b. Economical Feasibility
    c. Operational Feasibility
6. Project Planning and Scheduling
    a. PERT Chart
    b. GANTT Chart
7. Software Requirement Specifications (IEEE format preferred)

CHAPTER 4

REQUIREMENT ANALYSIS AND SPECIFICATION

This chapter includes the following subsections.

1. Users of the System
2. Modularity Criteria
3. Architecture Diagrams (whichever of the following if applicable)
    a. DFD
    b. UML Diagrams
    c. Flowchart
4. User Interface Layout
5. Structure of Reports Being Created
6. Database Design
    a. List of Entities and Attributes
    b. E R Diagram
    c. Structure of Tables

CHAPTER 5

IMPLEMENTATION

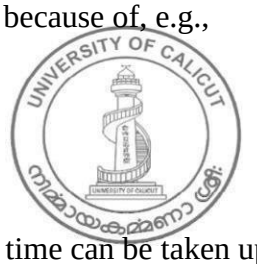
This chapter is about the realisation of the concepts and ideas developed earlier. It can also describe any problems that may have arisen during implementation and how you dealt with them.

Do not attempt to describe all the code in the system, and do not include large pieces of code in this section. Instead pick out and describe just the pieces of code which, for example:

- Are especially critical to the operation of the system;
- You feel might be of particular interest to the reader for some reason;
- Illustrate a non-standard or innovative way of implementing an algorithm, data structure, etc.

You should also mention any unforeseen problems you encountered when implementing the system and how and to what extent you overcame them. Common problems are:

- Difficulties involving existing software, because of, e.g.,
  - its complexity,
  - lack of documentation;
  - lack of suitable supporting software;
  - over-ambitious project aims.



A seemingly disproportionate amount of project time can be taken up in dealing with such problems. The Implementation section gives you the opportunity to show where that time has gone.

Complete source code should be provided separately as an appendix.

This chapter includes the following subsections.

1. Brief description about the Tools/Scripts for Implementation
2. Module Hierarchy
3. Coding
4. Problems Encountered

CHAPTER 6

TESTING

This chapter includes the following subsections.

1. Test Plans
2. Unit Testing
    a. Test Items (Test Cases)
3. Integration Testing
4. System Testing
    a. Test Items (Test Cases)
5. Implementation - Changeover Plans

CHAPTER 7

CONCLUSION

The purpose of this section is to provide a summary of the whole thesis or report. In this context, it is similar to the Abstract, except that the Abstract puts roughly equal weight on all report chapters, whereas the Conclusion chapter focuses primarily on the findings, conclusions and / or recommendations of the project.

There are a couple of rules for this chapter:

- All material presented in this chapter must have appeared already in the report; no new material can be introduced in this chapter (rigid rule of technical writing).
- Usually, you would not present any figures or tables in this chapter (rule of thumb).

Conclusions section can have the following (typical) content. These contents must **not** be given in bulleted format.

- Re-introduce the project and the need for the work though more briefly than in the introduction.
- Reiterate the purpose and specific objectives of your project.
- Recap the approach taken similar to the road map in the introduction.
- However, in this case, you are re-capping the data, methodology and results as you go.
- Summarize the major findings and recommendations of your work.

Future Enhancements

Identify further works that can be added to make your system to meet the challenges of tomorrow. You can also include whatever requirements you could not fully due to the scarcity of time/resources.

BIBLIOGRAPHY

Ideas or contents taken from other sources should be properly cited. It is important that you give proper credit to all work that is not strictly your own, and that you do not violate copyright restrictions.

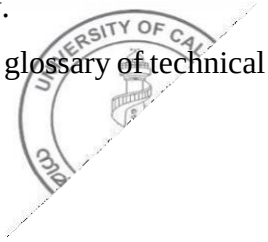
References should be listed in alphabetical order of authors' surname, and should give sufficient and accurate publication details. IEEE format is to be followed while preparing citations.

PUBLICATIONS OUT OF THE PROJECT WORK

A list of publications made or communicated out of the work done in the project is to be included here.

GENERAL INSTRUCTIONS

1. All chapters should contain an introduction and summary (summarizes the entire chapter content in one or two lines) sections.
2. Students have to take care that only chapters/sections relevant to their work are to be included in their report.
3. Instead of merely replicating the definitions for these sections from standard text books of software Engineering, the student has to describe the information related to his/her work (For eg, Feasibility study should be about how the proposed work is echnically /economically / operationally feasible).
4. Figures and tables are to be clear and legible.
5. Citations are to be provided wherever necessary.
6. Important code, screenshots, report formats and glossary of technical terms are to be attached as Appendices A, B, C and D respectively.



General Rules: STYLE OF PRESENTATION

1. Report Length: 50 to 70 pages excluding Appendix and Certificates
2. Alignment: Justify
3. Font: Times New Roman
4. Font size: 12
5. Line spacing: 1.