**Bayero University, Kano (BUK)**

**Faculty of Computing**

**Department of Computer Science**

**B.Sc. Computer Science**

**Proposed 30% addition to the CCMAS Course Structure**

**Level 100**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Course Code** | **Course Title** | **Unit** | **Status** | **LH** | **PH** |
| BUK-ICT 102 | Introduction to Information and Communication Technology | 2 | Core | 30 | 0 |
| BUK-CSC 101 | Computer Application Packages | 2 | Core | 30 | 45 |
| BUK-COS 103 | Introduction to Computer Programming | 3 | Core | 30 | 45 |
|  | **Total** | **7** |  |  |  |

**Level 200**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Course Code** | **Course Title** | **Unit** | **Status** | **LH** | **PH** |
| BUK-IFT 201 | Introduction to Web Technologies | 2 | Core | 15 | 45 |
| BUK-INS 202 | Human-Computer Interactions | 2 | Core | 15 | 45 |
| BUK-INS 204 | System Analysis and Design | 3 | Core | 30 | 45 |
|  | **Total** | **7** |  |  |  |

**Level 300**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Course Code** | **Course Title** | **Unit** | **Status** | **LH** | **PH** |
| BUK-IFT 302 | Web Application Development | 2 | Core | 15 | 45 |
| BUK-SEN 301 | Object-Oriented Programming | 2 | Core | 15 | 45 |
| BUK-IFT 303 | Mobile Application Development | 2 | Core | 15 | 45 |
| BUK-DTS 302 | Big Data Computing | 2 | Core | 15 | 45 |
|  | **Total** | **8** |  |  |  |

**Level 400**

| **Course Code** | **Course Title** | **Unit** | **Status** | **LH** | **PH** |
| --- | --- | --- | --- | --- | --- |
| BUK-CSC 403 | Introduction to Parallel Programming | 3 | Elective | 30 | 45 |
| BUK-CSC 404 | Operations Research | 3 | Elective | 45 | 0 |
| BUK-COS 401 | Soft Skills in Computing | 2 | Core | 15 | 0 |
| BUK-CYB 404 | Cloud Computing | 2 | Core | 30 | 45 |
| BUK-DTS 404 | Data Mining | 3 | Core | 30 | 45 |
| BUK-DTS 403 | Data Visualisation for Data-driven Decision Making | 2 | Core | 15 | 45 |
| BUK-DTS 405 | Statistical Computing Inference and Modelling | 2 | Core | 30 | 45 |
|  | **Total** | **17** |  |  |  |
| **Grand Total** | | **36** |  |  |  |

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK- ICT 102 Introduction to Information and Communication Technology, 2 Units, Elective, 30 LH, 0 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

The Fundamentals of Information Technology course is designed to provide students with an understanding of the basic principles of computing and information technology. The course covers a wide range of topics, including computer hardware and software, computer networks and the internet, database management systems, programming languages, and cyber security.

The course is typically divided into several modules, each of which covers a specific topic. In the first module, students will learn about computer hardware, including the various components of a computer system and how they work together. The next module will focus on computer networks and the internet, including the basics of networking, network protocols, and the World Wide Web. Students will also learn about database management systems and how they are used to organize and store information. The course will also cover the basics of cyber security, including threats to information security, cyber security principles, and best practices for securing computer systems and networks.

Throughout the course, students will have the opportunity to apply what they have learned through hands-on exercises and projects. They will also be introduced to tools and technologies commonly used in the field of information technology, such as software development environments, network monitoring tools, and database management systems.

Overall, the Fundamentals of Information Technology course provides students with a solid foundation in the principles of computing and information technology, preparing them for further study in the field or for entry-level positions in the IT industry.

**Objectives**

1. Develop a basic understanding of the history and evolution of information technology and its impact on society.
2. Familiarise students with the role of computers in their everyday lives

3. Develop the ability to identify, install, and configure computer hardware and software components.

4. Familiarise students with emerging trends in information technology, such as cloud computing, artificial intelligence, and the internet of things.

5. Understand the ethical and social implications of information technology and its impact on society.

**Learning Outcomes**

1. Identify components of the computer and know-how the components communicate;
2. Know the concept of data transfer and memory types and management;
3. Understand graphics processing;
4. Identify and be able to use different communication ports;
5. Know the software types;
6. State different computer network topologies and their sizes; and
7. Use office applications and the internet.

**Course contentss**

Basic principle of computers. Computer "backbone". Data transmission. Random Access Memory. Permanent Memory. Graphic processing. Communication Ports. Input and Output Devices. Software types. Accessibility options. Computer types. Portable digital devices. Network Types. Internet. Instant messaging. Voice over Internet Protocol. Really Simple Syndication. Network communication. Internet data transfer. Data rate units. Internet access. Virtual (online) communities. Computer in the workplace. Telecommuting (telework).

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK-COS-101 Computer Application Packages. 2 Units, Elective, 15 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

This course is aimed at beginning to intermediate computer users. It teaches a range of computer skills from the basics of using Windows, to basic internet literacy, to creating projects using Microsoft Office. Assignments show step-by-step visuals to help students complete projects, and include integration across Microsoft Word and Excel.

This course will apply learning sciences to engage students and better support the learning process. The material is aimed to deliver an active learning experience. They include text, images, videos, assessments, directed feedback, and practice questions that invite students to apply their knowledge, improve their understanding, and perform better.

**Objectives**

1. Navigate and perform common tasks in Word, such as opening, viewing, editing, saving, and printing documents, and configuring the application.
2. Format text and paragraphs, Perform repetitive operations efficiently using tools such as Find and Replace, Format Painter, and Styles.
3. Enhance lists by sorting, renumbering, and customising list styles, create and format tables.
4. Insert graphic objects into a document, including symbols, special characters, illustrations, pictures, and clip art, format the overall appearance of a page through page borders and colours, watermarks, headers and footers, and page layout.
5. Examine spreadsheet concepts and explore the Microsoft Office Excel environment, Learn all about formatting text in Excel, understanding Number Formats and formatting, work with Multiple Worksheets, learn how to perform computations in Worksheets.

**Learning Outcomes**

The students will be able to

1. Select the most appropriate software to use to complete a task
2. Identify the key features of a word processor and spreadsheet application
3. Apply the key features of a word processor to format a text, paragraph and document
4. Evaluate formatting techniques to understand why we format documents
5. Develop and apply fundamental spreadsheet skills.
6. Demonstrate proficiency in using moderately complex spreadsheet tools such as tables and chart

**Course contentss**

Unit 1-Introduction to Computer Application: What are computer application packages? Types of computer application packages Overview of popular computer application packages. Unit 2-Microsoft Office Suites - Microsoft Word: Creating, editing, and formatting documents; Working with tables, graphics, and charts; Collaborating on documents. Microsoft Excel: Creating, editing, and formatting spreadsheets; Working with formulas and functions; Creating charts and graphs. Microsoft PowerPoint: Creating, editing, and formatting presentations; Adding multimedia elements; Delivering presentations. Unit 3- Google Slides: Creating, editing, and formatting presentations; Adding multimedia elements; Delivering presentations

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK-COS-103 Introduction to computer programming. 2 Units, Elective, 15 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines**.**

**Overview**

In this course, students will learn how Python works and its place in the world of programming languages; to work with and manipulate strings; to perform maths operations; to work with Python sequences; to collect user input and output results; flow control processing; to write to, and read from, files; to write functions; to handle exception; and work with dates and times.

**Objectives**

1. Understand why Python is a useful scripting language for developers

2. Design and program Python applications.

3. Ue lists, tuples, and dictionaries in Python programs and identify Python object types.

4. Use indexing and slicing to access data in Python programs, write loops and decision statements in Python.

5. Write functions and pass arguments in Python, build and package Python modules for reusability.

6. Read and write files and design object‐oriented programs with Python classes

**Learning outcome**

The students will be able to:

1. Build basic programs using fundamental programming constructs like variables, conditional logic, looping, and functions

2. Use indexing and slicing to access data in Python programs

3. Work with user input to create fun and interactive programs

4. Create simple games with images, animations, and audio using our custom beginner-friendly programming library, Wizardlib

5. Create and manipulate files

**Course contentss**

Vital Python – Math, Strings, Conditionals, and Loops. Vital Python. Numbers: Operations, Types, and Variables. To Open a Jupyter Notebook. Python as a Calculator. Standard Math Operations. Basic Math Operations. Order of Operations. Spacing in Python. Number Types: Integers and Floats. Complex Number Types. Errors in Python. Variables. Variable Assignment. Changing Types. Reassigning Variables in Terms of Themselves. Variable Names. Multiple Variables. Comments. Docstrings. Theorem in Python. Strings: Concatenation, Methods, and input(). String Syntax. Escape Sequences with Quotes. Multi-Line Strings. The print() Function. String Operations and Concatenation. String Interpolation. Comma Separators. Format. The len() Function. String Methods. Casting. The input() Function. String Indexing and Slicing. Indexing. Slicing Strings and Their Methods. Booleans and Conditionals. Booleans. Logical Operators. Comparison Operators. Comparing Strings. Conditionals. The if Syntax. Indentation. if else. The elif Statement. Loops. The while Loops. An Infinite Loop. break. Programs. The for Loop. The continue Keyword. Python Structures. The Power of Lists. List Methods. Accessing an Item from a List. Adding an Item to a List. Dictionary Keys and Values. a List and a Dictionary. Zipping and Unzipping Dictionaries Using zip(). Dictionary Methods. Tuples. A Survey of Sets. Set Operations. Choosing Types. Executing Python – Programs. Algorithms, and Functions Introduction. Python Scripts and Modules.

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK-IFT-201 - Introduction to Web Technologies. 2 Units, Elective, 15 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

Introduction to Web Computing is a course that focuses on the basic concepts and technologies used in building websites and web applications. The course is designed to provide students with a comprehensive understanding of the components of web computing and how they work together to create interactive and engaging web content.

The course begins by introducing students to the basics of HTML, CSS, and JavaScript, which are the three core technologies used in building websites. Students will learn how to use HTML to structure web content, CSS to style and layout web pages, and JavaScript to add interactivity and dynamic behaviour to web pages.

The course then progresses to more advanced topics, such as server-side programming, databases, and web frameworks. Students will learn how to build dynamic web applications using technologies such as PHP, Node.js, and Python.

Throughout the course, students will explore various aspects of web design, including user experience, accessibility, and responsive design. They will also learn about web security, including best practices for preventing and mitigating common web attacks.

By the end of the course, students will have a solid understanding of the basic components of web computing and how they work together to create engaging and interactive web content. They will be able to build their own websites and web applications, and they will have the skills to continue their study of advanced web technologies. This course is essential for anyone looking to pursue a career in web development or anyone who wants to gain a better understanding of how the web works.

**Objectives**

1. Develop a basic understanding of the components of web computing, including HTML, CSS, and JavaScript.

2. Gain knowledge of the principles of web design, including user experience, accessibility, and responsive design.

3. Develop skills in building static web pages and basic web applications using HTML, CSS, and JavaScript.

4. Use server-side programming languages, such as PHP, Node.js, and Ruby on Rails, to build dynamic web applications.

5. Understand the basics of database design and management, including how to use SQL to interact with a database.

**Learning Outcomes**

1. State the origin of the internet and the World Wide Web;

2. Create simple web content using HTML, CSS, and JavaScript;

3. Use simple application frameworks to develop web content; and

4. Appraise the impact of the World Wide Web on people’s lives over time.

5. Understand client and server-side scripting and their applicability

**Course contents**

Introduction to the internet, the World Wide Web (WWW), and web development. WWW as a platform for interactive applications, content publishing, and social services. The role of HTTP and HTTPS in the context of web applications. Roles and operations of web browsers and the webserver. Interacting with web applications through forms.. Using style sheets to separate document structure and document formatting. Web development tools and frameworks. Build a simple website that: organises information effectively, uses valid HTML and CSS, and applies appropriate web standards from standards bodies such as W3C. HTTP communication protocol. The mark-up languages HTML, XHTML, and XML, the CSS and XSLT standards for formatting and transforming web content.. Interactive graphics and multimedia content on the web.. Client-side programming using JavaScript. Impact of the World Wide Web on people’s lives over time.

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK- INS 202 Human-Computer Interactions 2 Unit, Elective, 15 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

Human-Computer Interaction (HCI) is a field that explores the design, development, and evaluation of interactive computer systems for human use. The course provides an overview of HCI concepts, principles, and techniques that enable the design and development of effective interactive systems.

The course covers the basics of human perception, cognition, and behaviour, as well as the ways in which people interact with computer systems. Students will learn about the principles of usability, user-centred design, and user experience design, and the role of feedback, interactivity, and feedback in designing effective interactive systems.

The course also covers a range of design and evaluation methods, such as user research, prototyping, and usability testing. Students will learn how to identify user needs, design interfaces that meet those needs, and evaluate the usability and effectiveness of interactive systems.

Finally, the course explores emerging trends in HCI, such as mobile computing, social computing, and ubiquitous computing. Students will learn how to design and develop interactive systems that are adapted to different contexts and platforms, and how to evaluate the effectiveness of such systems.

By the end of the course, students should have a solid understanding of the fundamental concepts of HCI, and be able to apply these concepts to the design and development of effective interactive systems. Students will also gain hands-on experience in designing, prototyping, and evaluating interactive systems, and be prepared for further study or careers in HCI, design, or user experience.

**Objectives**

1. Understand the fundamentals of human-computer interaction (HCI) and its importance in design.

2. Develop skills in analysing, designing, and evaluating user interfaces for digital systems and gain knowledge of various design principles, guidelines, and best practices for creating effective user interfaces.

3. Learn different approaches to user research and testing to gather user feedback and improve design, emerging technologies and their impact on HCI, such as virtual and augmented reality, conversational interfaces, and machine learning.

4. Develop critical thinking skills to evaluate the ethical and social implications of technology design and use.

5. Explore and analyse case studies and real-world examples of successful and unsuccessful HCI design.

**Learning Outcomes**

1. Discuss the foundations and concept of the human-computer interface;

2. Explain Understanding of principles of human-computer interface;

3. Explain the design and development of the human-computer interface; and

4. Explain the importance of user feedback.

5. Outline and discuss usability goals and user experience goals for designing an interactive product

6. Discuss the conceptual, practical, and ethical issues involved in evaluation

7. Produce simple prototypes of interactive products

**Course contents**

Foundations of HCI. The concept underlying the design of HCI. Principles of GUI. GUI toolkits. System design methods. User conceptual models and interface metaphors. Human cognitive and physical ergonomics. Human-centred software evaluation and development. GUI design and programming.

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK- INS 204 System Analysis and Design 2 Unit, Elective, 15 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

The System Analysis and Design course teaches students how to analyse, design, and develop information systems. The course covers a range of concepts, methodologies, and tools used in the system development life cycle, including gathering and analysing requirements, designing the system architecture and user interface, implementing the system, and testing and maintaining it.

Throughout the course, students will learn about different frameworks and methodologies used in system analysis and design, as well as programming languages, software development methodologies, and tools used to implement and test the system.

The course is designed to provide students with a strong foundation in the process of designing and developing information systems, as well as the skills and knowledge necessary to apply this process to real-world projects. Upon completion of the course, students should be able to understand and apply the system development life cycle, develop comprehensive system requirements, create an appropriate system design based on those requirements, implement and test the system, and maintain and improve it over time.

Overall, the System Analysis and Design course is an essential course for students interested in pursuing careers in software development, information technology, or systems analysis, as it provides the skills and knowledge necessary to design and develop effective information systems that meet business goals.

**Objectives**

1. Understand the basic concepts and principles of systems thinking and system analysis, and how they can be applied to the development of information systems.

2. Gather, document, and analyse requirements for a system, and how to translate those requirements into specific system functionality.

3. Implement and test a system, and how to maintain and improve it over time.

4. Familiarise with different methodologies and frameworks used in system analysis and design, including the Waterfall Model, Agile Methodology, and Rational Unified Process (RUP).

5. Develop critical thinking and problem-solving skills necessary to identify and resolve issues that may arise during the system development process.

**Learning Outcomes**

1. Describe system requirements gathering techniques;

2. Explain data modelling technique (entity relationship modelling);

3. Explain process modelling technique (data flow diagram);

4. Describe system architectural design;

5. Describe process and database design; and

6. Explain user interface design.

**Course contents**

Structured approach to analysis and design of information systems for businesses. Software development life cycle. Structured top-down and bottom-up design. Dataflow diagramming. Entity relationship modelling. Computer aided software engineering. Input and output, prototyping design and validation. File and database design. Design of user interfaces. Comparison of structured and object-oriented design.

**Lab work:** Practical exercises on software development life cycle (SDLC) activities with different case studies. Use of different information systems case studies to apply the knowledge of structured top-down and bottom –up design, data flow diagram and entity relationship models.

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK- IFT 302 - Web Application Development. 2 Units, Elective, 15 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

The Web Application Development course is designed to teach students how to develop web-based applications using modern web technologies. Throughout the course, students will learn how to design and develop web applications that are both functional and user-friendly.

Students will be introduced to fundamental web development concepts, including HTML, CSS, and JavaScript. They will learn how to use these technologies to structure web pages, style them, and add interactivity and functionality to web applications.

The course will also cover popular web development frameworks such as React, Angular, and Vue.js, which students will use to build dynamic, scalable web applications. In addition, students will learn about server-side programming languages like Node.js and Ruby on Rails and how to use them to build robust backend systems that power web applications.

Databases and data management will also be covered in the course, with students learning about database technologies such as SQL and MongoDB and how to use them to store and manage data within web applications.

Students will also learn about web application security and performance, with a focus on best practices for securing web applications against common vulnerabilities like cross-site scripting (XSS) and SQL injection, as well as techniques for optimising web application performance.

Throughout the course, students will learn about software development methodologies like Agile and Scrum, and how to apply them to web development projects. This will help students to work effectively within a team to develop and deploy web applications.

The Web Application Development course is designed to provide students with a comprehensive understanding of the web development process, as well as the skills and knowledge necessary to design and develop functional, scalable, and secure web applications. By the end of the course, students should be able to develop responsive web applications, design and develop secure backend systems, manage databases, and optimise web application performance to ensure efficient and speedy user experiences.

**Objectives**

1. Provide students with a solid understanding of web development concepts and technologies, including HTML, CSS, JavaScript, and web frameworks.

2. Teach students how to design and develop responsive, user-friendly web applications using modern web technologies.

3. Introduce students to server-side programming languages like Node.js and Ruby on Rails, and how to use them to build robust backend systems that power web applications.

4. Teach students how to manage databases and utilise database technologies to store and manage data within web applications.

5. Introduce students to web application security and performance best practices, including techniques for securing web applications against common vulnerabilities and optimising web application performance.

6. Prepare students for careers in web development or software engineering by providing them with the skills and knowledge necessary to design and develop functional, scalable, and secure web applications.

**Learning Outcomes**

1. Understand the development of a client-side browser based web application including its capabilities and limitations.

2. Design and implement simple client-side and server-side web applications;

3. Demonstrate hands-on skills in PHP and Python programming uses open-source software;

4. Compare and contrast web programming with general-purpose programming; and

5. Develop a fully functioning website and deploy it on a web server.

**Course contents**

Introduction to framework-based web development using a contemporary language like PHP and ASP.net. Principles of web pages (dynamic and static) and website design. The tool used in web development. Client-side and server-side languages. Creation of interactive, dynamic websites using a common web architecture and object-based database access. Design, implementation, and testing of web-based applications including related software, databases, interfaces, and digital media. Standard object models, and the use of server-side programmes for database and file access; testing, software quality assurance; and the process of publishing Web sites. Hands-on PHP and Python programme using open-source software (Apache, PHP, Python, JavaScript, and MySQL). Programming for web development includes control structures, objects, functions, and the use of composite data types. Deploying dynamic content using JavaScript. Designing and developing dynamic web pages and creating, validating, transforming, and formatting data using PHP. **Lab Work:** Simple PHP programming. Design of simple web pages. Creation of dynamic websites. Design of client-side and server-side programmes. Demonstration of web-based applications with database access. Use of JavaScript to develop dynamic content. Use of Python to develop dynamic web pages.

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

**Bayero University, Kano.**

**Faculty of Computing**

**Department of Computer Science**

**BSc. Computer Science**

**BUK- SEN 301 Object-Oriented Programming 2 Unit, Elective, 15 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

The Object-Oriented Programming course is designed to introduce students to the fundamental principles of object-oriented programming (OOP) and teach them how to apply these principles to build software systems. Throughout the course, students will learn about key OOP concepts such as classes, objects, inheritance, encapsulation, and polymorphism.

The course will begin by introducing students to the basic syntax and structure of a programming language such as Java or Python, and then gradually move on to more advanced topics such as designing and implementing classes, working with inheritance, and managing program flow with control structures like loops and conditionals.

Students will learn how to use OOP concepts to design and implement software systems that are modular, reusable, and easy to maintain. They will also learn about key software development concepts such as debugging, testing, and version control.

The course will also cover design patterns, which are reusable solutions to common programming problems. Students will learn how to identify and apply design patterns to software systems to make them more efficient, scalable, and maintainable.

In addition, students will learn about graphical user interface (GUI) programming, which involves designing and building user interfaces for software applications. They will learn how to use OOP concepts to create interactive and user-friendly GUIs that are both functional and aesthetically pleasing.

Throughout the course, students will be expected to work on programming assignments and projects, which will give them hands-on experience in applying OOP concepts to real-world programming problems. By the end of the course, students should be able to design and implement software systems using OOP principles, apply design patterns to software systems, and create GUIs that are both functional and user-friendly.

Overall, the Object-Oriented Programming course is designed to provide students with a solid foundation in OOP concepts and principles, as well as the skills and knowledge necessary to design and build software systems using OOP principles.

**Objectives**

1. Introduce students to the fundamental principles of object-oriented programming (OOP), including classes, objects, inheritance, encapsulation, and polymorphism.

2. Teach students how to use OOP concepts to design and implement software systems that are modular, reusable, and easy to maintain.

3. Provide students with a solid understanding of key programming concepts such as control structures, functions, and data types.

4. Teach students how to apply debugging, testing, and version control techniques to software development projects.

5. Prepare students for careers in software development or related fields by providing them with the skills and knowledge necessary to design and build high-quality software systems using OOP principles.

**Learning Outcomes**

1. Explain the concept of the object-oriented approach to modelling;

2. Describe the conceptual model of the UML-based software development life cycle;

3. Demonstrate how to use the major UML diagrams for object-oriented analysis and design;

4. Demonstrate the use of UML-based CASE tools.

5. Use object oriented programming concepts to solve real world problems.

**Course contents**

Object-oriented approach to information system development, particularly in. reference to the earlier stages of analysis and design. Importance of modelling.. Principles of modelling.. Object-oriented modelling.. Conceptual model of the Unified Modelling Language (UML).. Architecture.. Software development life cycle. The principles and basic concepts of object orientation and the different aspects of object-oriented modelling as represented by the UML technique. Case study of a typical UML-based CASE tool.

**Lab Work:** Practical exercises on different requirements specification and design activities; developing problem statements, SRS documents and Use Case Diagrams; designing UML Activity diagrams, UML Class diagrams and State Chart diagrams; drawing partial layered, logical architecture diagram with UML package diagram notation; Designing Component and Deployment diagrams.

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK-IFT-303 - Mobile Application Development. 2 Units, Elective, 15 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

The Mobile Application Development course is designed to teach students how to design and build mobile applications for different platforms, such as Android and iOS. In this course, students will learn about the different mobile operating systems, software development kits (SDKs), and mobile app development frameworks that are used in mobile application development. They will also learn how to design and implement mobile applications that are user-friendly, efficient, and responsive.

Throughout the course, students will gain a solid understanding of key mobile application development concepts such as mobile UI design, navigation, data storage, and integration with other mobile devices. They will also learn how to apply debugging, testing, and version control techniques to mobile application development projects. Additionally, they will be introduced to different mobile application development platforms, such as Android and iOS, and will learn how to develop applications for these platforms using appropriate tools and technologies.

To provide students with hands-on programming experience, the course includes assignments and projects that involve designing and building mobile applications using appropriate tools and technologies. By completing these assignments and projects, students will gain practical experience and build a portfolio of mobile applications that they can showcase to potential employers.

The Mobile Application Development course is designed to prepare students for careers in mobile application development or related fields by providing them with the skills and knowledge necessary to design and build high-quality mobile applications that meet the needs of users and organisations.

**Objectives**

1. Introduce students to the concepts and technologies used in mobile application development, including mobile operating systems, software development kits (SDKs), and mobile app development frameworks.

2. Teach students how to apply debugging, testing, and version control techniques to mobile application development projects.

3. Introduce students to different mobile application development platforms, such as Android and iOS, and teach them how to develop applications for these platforms using appropriate tools and technologies.

4. Teach students how to optimise mobile applications for different mobile devices, such as smartphones and tablets, and for different mobile networks and connectivity conditions.

5. Prepare students for careers in mobile application development or related fields by providing them with the skills and knowledge necessary to design and build high-quality mobile applications that meet the needs of users and organisations.

**Learning Outcomes**

1. Identify the basic knowledge on mobile application environment and technology;

2. Explain the concepts and processes of mobile application development;

3. Discuss design and development issues specific to mobile applications;

4. Design and develop mobile applications, using development tools and environments;

5. Evaluate the performance of a mobile application and give its result; and

6. Appreciate perspectives of mobile applications and their impact

**Course contents**

Introduction to developing mobile applications. Mobile operating systems capabilities, application architecture, and major components, such as activities, services, broadcast receivers, etc. Development of interactive applications using widget libraries, web-based services. Basic concepts of 2D graphics and animation. An SQL database engine, and multithreading. Multiplatform mobile application development. Mobile application basics and features; Android application basics, UI design. Data storage; networking application design. Advanced application design (sensors, camera, GPS, Audio, etc.), graphics and games, web-based hybrid application design. Design and implement a simple mobile application for a given mobile platform. Metrics and methods to evaluate the performance of mobile applications. Mobile application perspectives and impact. **Lab Work:** Demonstration of a Simple Mobile Application. Design and Development of interactive mobile applications. Demonstration of multiplatform mobile application development. Development of Android applications including UI design and data storage design. Demonstration of advanced mobile application design. Illustration of metrics for measuring the performance of mobile applications.

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK-DTS 302: Big Data Computing. 2 Units, Elective, LH 15; PH 45**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

The Big Data Computing course is designed to teach students about the challenges and opportunities presented by the processing and analysis of large-scale data sets, also known as big data. The course covers a range of topics related to big data, including data storage, distributed computing, data management, and analysis.

Throughout the course, students will gain an understanding of the technologies and tools used in big data computing, such as Hadoop, MapReduce, and Spark. They will also learn about data management techniques for big data, including data warehousing and data lakes, and the importance of data quality and integrity in big data systems.

The course will also cover data analysis techniques for big data, such as machine learning, data mining, and predictive analytics. Students will learn how to use these techniques to extract insights and knowledge from large-scale data sets.

To provide students with hands-on experience, the course includes assignments and projects that involve working with real-world big data sets and using big data technologies and tools to process and analyse the data. By completing these assignments and projects, students will gain practical experience and build a portfolio of big data projects that they can showcase to potential employers.

The course is designed to prepare students for careers in big data computing or related fields by providing them with the skills and knowledge necessary to manage, process, and analyse large-scale data sets.

**Objectives**

1. Introduce students to the concepts and challenges of big data computing, including data storage, distributed computing, and data management.

2. Provide students with an understanding of data management techniques for big data, including data warehousing and data lakes, and the importance of data quality and integrity in big data systems.

3. Teach students about data analysis techniques for big data, such as machine learning, data mining, and predictive analytics.

4. Provide students with hands-on experience in processing and analysing large-scale data sets using big data technologies and tools.

5. Prepare students for careers in big data computing or related fields by providing them with the skills and knowledge necessary to manage, process, and analyse large-scale data sets.

**Learning Outcomes**

* + - 1. Identify Big Data;
      2. Conceptualise and design different types of data analysis tasks (e.g. supervised, semi-supervised and unsupervised learning tasks).

3. Identify some of the foundational tools, systems, and platforms that feature in working with Big Data across several domains;

4. Install Big Data working tools on a computer; and

5. Analyse Big Data contents.

**Course contents**

Installation: Cloudera VM, Jupyter server. Big data retrieval and relational querying: Postgres databases, NoSQL data, MongoDB, Aerospike, and Pandas for data aggregation and working with data frames. Big Data Integration: Splunk and Datameer. Big Data Processing: Apache Spark, Hadoop, Spark Core (Spark MLlib and GraphX). Big Data Applications (Graph Processing). Big Data Streaming Platforms for Fast Data. **Lab Work:** Analysing Twitter Data using Spark and MongoDB. Learn Big Data analytics skills. Practical procedure for the crafting of an enterprise-scale cost-efficient Big. Data and machine learning solution to uncover insights and value from data. Use the practical exercises to bridge the gap between the theoretical world of technology with the practical ground reality of building corporate Big Data and data science platforms. Hands-on exposure to Hadoop and Spark (or any of the BD tools), build machine learning dashboards using R and R Shiny, create web-based apps using NoSQL databases. Practical assignment of BD security.

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK-CSC 403 Introduction to Parallel Programming. 2 Unit, Elective, 15 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

This course provides a basic, in-depth look at techniques for the design and analysis of parallel algorithms and for programming them on commercially available parallel platforms. Principles of parallel algorithms design and different parallel programming models are both discussed. MPI, POSIX threads, and Open MP all are discussed. This course is for anyone wanting to gain proficiency in all aspects of parallel and distributed programming.

**Objectives**

1. Expose students to basic techniques of parallel algorithm development and programming on different parallel platform
2. Discuss the content and use of the terminology for how one measures the performance of parallel algorithms and parallel computers,
3. Understand the key parallel computational models
4. Write parallel programs using message passing and shared memory paradigms
5. Implement key algorithms using data parallel programming model
6. Understand basic principles of performance modelling and optimization

**Learning Outcomes**

1. Develop and apply knowledge of parallel and distributed computing techniques and methodologies.
2. Analyse the performance of parallel and distributed applications.
3. List architectural elements of modern processors and explain their impact on performance
4. Explain the functionality of MPI primitives (including send/recv variants, collectives, sub-communicators, etc.)
5. Describe and characterise the behaviour of MPI programs
6. Create and debug MPI programs to accomplish a computational task
7. Explain workings of covered parallel algorithms and reason about their efficacy and of variants
8. Read and write programs in OpenMP and MPI

**Course contents**

Why Parallel Computing? Modifications to the von Neumann Model. Parallel Hardware. Parallel Software. Input, Output and Performance. Distributed-Memory Programming with MPI. The Trapezoidal Rule in MPI. Collective Communication. Performance Evaluation of MPI Programs. A Parallel Sorting Algorithm. Processes, Threads, and Pthreads. Matrix-Vector Multiplication. Critical Sections, Busy-Waiting & Mutexes. Producer-Consumer Synchronisation and Semaphores. Barriers and Condition Variables. Read-Write Locks. Caches, Cache Coherence, and False Sharing.

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK-CSC 404 Operation Research. 2 Unit, Elective, 15 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

This course provides an introduction to Operations Research, a field of study that uses mathematical and analytical methods to help organisations make better decisions. The course covers a range of topics, including linear programming, network analysis, queuing theory, decision analysis, and simulation modelling. Emphasis is placed on the use of modelling techniques and optimization software to solve practical problems.

**Objectives**

1. Develop a solid understanding of the fundamental principles and concepts of Operations Research.
2. Develop proficiency in formulating and solving linear programming problems, including optimization of resource allocation, production planning, and transportation problems.
3. Learn network analysis techniques for solving problems such as shortest path, maximum flow, and project scheduling.
4. Gain an appreciation for the role of Operations Research in a wide range of industries, including manufacturing, transportation, healthcare, and finance.
5. Understand the ethical considerations involved in Operations Research and its impact on society.

**Learning Outcomes**

1. Identify and develop operational research models from the verbal description of the real system.
2. Understand the mathematical tools that are needed to solve optimisation problems.
3. Use mathematical software to solve the proposed models.
4. Expand existing formulations;
5. Critically evaluate the impact of model assumptions and choose an appropriate solution technique for a given formulation.

**Course contents**

Phases of operation research study. Classification of operation research models’ linear dynamic and integer programming. Decision Theory. Inventory Models, Critical Path Analysis and Project Controls.

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK-COS 401: Soft Skills and Professionalism in Computing. 2 Units, Elective, LH 15; PH 0**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

This course would prepare graduates from this program to develop their interpersonal, entrepreneurial and professional skills. The course participants would be exposed to principles of communication skills, Problem solving, critical thinking, and time management skills. The course participants would be furthermore exposed to ingredients of successful team management and Leadership skills.

**Objectives**

1. Develop Effective communication, Teamwork and Collaboration Skills
2. Learn how to manage time and enhance productivity
3. Identify the use of creativity and Critical Thinking to solve societal problems
4. Enable Students to develop professional entrepreneurial management skills for IT consultancy
5. Describe the general principles of ethics in Computing and IT

**Learning Outcomes**

1. Demonstrate effective communication skills in dealing with clients, and the general public.
2. Describe the process of effectively solving problems.
3. Apply strategies and skills for enhancing adaptability and resilience to a variety of situations when managing yourself and others
4. Develop a Career plan
5. Use Creative and Critical Thinking to solve societal problems
6. Create & deliver impactful presentations
7. Apply Persuasion and Negotiation Skills to Business and IT consultancy
8. Mention Strategies for team management and effective Leadership

**Course contents**

Effective communication skills for IT marketers. Teamwork and Collaboration for Professional Success. Creative and Critical Thinking. Effective Approaches to Solving Problems, & Evaluating Solutions. Time management for Personal & Professional Productivity. Effective approaches to Decision making. Developing Trust and Increasing Your Influence in Society. Creating & delivering impactful presentations. Stress management and mental wellbeing . Adaptability and Building personal resilience. Career Planning. Pitches and Persuasion. Negotiation Skills for Business and IT consultancy. Openness to criticism and conflict response Style. Leadership: Theories, principles and styles of leadership. Team management Strategies. Introduction to professionalism in Computing. Code of ethics and principles for various professions in Computing and IT.

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK-CYB-404 Cloud Computing and Security. 2 Units, Elective, 15 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

Cloud Computing is a technology that allows you to use the resources of a large number of computers connected through a real-time communication network. By using cloud computing, you can gain access at any time through any device, via the Internet, to data and files which you have uploaded, or to software applications which you need to use for personal or professional use. Cloud computing being used more and more in business today and it is very important for any professional to understand what it is all about.

This course defines Cloud Computing and establishes a strong working knowledge of the concepts and technologies needed to work effectively with the cloud. The course allows you to understand what cloud computing is and how it works. It describes the benefits of cloud computing along with its potential drawbacks. The course enables you to determine which cloud is appropriate from a business and technical perspective, to select appropriate cloud providers and to plan and implement a cloud adoption strategy. Formation of theoretical knowledge and practical skills in practical realisation of the benefits of cloud computing in today's business, learning the tools of the technology. The course covers technologies required to build classic (traditional), virtualized, and cloud data centre environments. These technologies include compute, storage, networking, desktop and application virtualization.

**Objectives**

1. Introduction to the basic concepts and terminology of cloud computing;
2. Familiarisation with areas of cloud technologies;
3. Acquaintance with the concept of cloud computing in relation to business activities;
4. Security Studies, scaling, deployment, backup, in the context of cloud infrastructure; learning techniques cloud programming;
5. Development of system administration skills for the develop

**Learning Outcomes**

1. Articulate the main concepts, key technologies, strengths, and limitations of cloud computing and the possible applications for state-of-the-art cloud computing
2. Identify the architecture and infrastructure of cloud computing, including SaaS, PaaS, IaaS, public cloud, private cloud, hybrid cloud, etc.
3. Explain the core issues of cloud computing such as security, privacy, and interoperability.
4. Choose the appropriate technologies, algorithms, and approaches for the related issues.
5. Identify problems, and explain, analyse, and evaluate various cloud computing solutions.

**Course contents**

Introduction to cloud computing. Objectives, challenges, application domains, advantages. Computational and storage cloud architectures; Service level agreements, service lifecycle management. Elasticity and scalability techniques. Information, account and billing management. Cloud service model, service provisioning and access models. Cloud Service Models: Software as a Service layer; Platform as a Service layer; Infrastructure as a Service layer. Virtualization and resource management. Distributed object storage clouds. Data storage and retrieval based on content. Computational tasks execution in storage clouds. Quality of service approaches. Requirements and parameters classification. Monitoring and control mechanisms. Quality of service guarantees. Security in the Cloud: Cloud threats; Threat Mitigation, Cloud and Security Risks. Google AppEngine. OpenStack. Apache Hadoop / MapReduce

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK-DTS 404 Data Mining. 2 Units, Elective, 15 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

Data Mining is a course that teaches students how to extract valuable information from large datasets using various statistical and machine learning techniques. In this course, students will learn how to use tools like Python, R, and SQL to explore, pre-process, and visualise data.

The course will cover topics such as data cleaning, feature selection, dimensionality reduction, classification, clustering, association analysis, and outlier detection. Students will also be introduced to advanced topics such as deep learning, natural language processing, and big data analytics.

Throughout the course, students will work on real-world projects and use popular data mining tools. They will also gain hands-on experience in data analysis, modelling, and evaluation using various metrics.

By the end of the course, students will be able to develop data mining solutions for a variety of real-world problems such as customer segmentation, fraud detection, recommendation systems, and sentiment analysis. They will also be equipped with the knowledge and skills to pursue advanced studies in data science or related fields.

**Objectives**

1. Prepare students for a career in data science, data analytics, or related fields that require expertise in data mining.
2. Understand the fundamental concepts and techniques of data mining, including data pre-processing, classification, clustering, association analysis, and anomaly detection.
3. Apply data mining algorithms to solve real-world problems in various domains, such as finance, healthcare, marketing, and social media.
4. Use popular data mining tools and platforms, to implement and evaluate data mining models.
5. Develop a strong foundation in statistics, machine learning, and database systems that are essential for data mining.

**Learning Outcomes**

1. Understand what Is Data Mining, what kinds of data can be mined, what kinds of patterns can be mined, and what kinds of applications are targeted.
2. Explain the principles and best practices of managing data with efficiency and effectiveness;
3. Explain major Issues in data mining.
4. Apply machine learning, pattern recognition, statistics, visualization, algorithm, database technology and high-performance computing in data mining applications.
5. Demonstrate knowledge of SQL and NoSQL;
6. Explain data warehouse concepts, methodologies and tools; and
7. Explain data mining architecture and applications.

**Course contents**

Relational Databases: Mapping conceptual schema to relational schema; Database Query Languages (SQL) and NoSQL, Concept of functional dependencies & multi-valued dependencies. Transaction processing; distributed databases, XML and semantic Web. Data warehousing. Introduction to data science. Introduction to Data Warehouse, OLTP Systems; Differences between OLTP Systems and Data Warehouse: Characteristics of Data Warehouse; Functionality of Data Warehouse: Advantages and Applications of Data Warehouse. Advantages, Applications: Top- Down and Bottom-Up Development Methodology: Tools for Data warehouse development: Data Warehouse Types. Introduction: Scope of Data Mining: What is Data Mining. How Data Mining Works, Predictive Modelling: Data Mining and Data Warehousing: Architecture for Data Mining: Profitable Applications: Data Mining Tools. **Lab work**: Practical exercises on basic R commands and data structures for manipulating data; how to read data from multiple formats in and out of R, using loops, conditional statements, and functions to automate common data management tasks. Exercises on how to clean and manage multiple complex datasets, manipulate textual data, basic web scraping techniques, for both standard web pages and the Twitter API. Work on techniques and hardware necessary to manage large datasets efficiently. Practical exercise on managing multiple data sets by example; working with text data; converting long- and wide-format data; and dealing with messy data. R Programming Fundamentals for data I/O and packages, looping and conditional statements, and functions.

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK-DTS 403 Data Visualisation for Data-driven Decision Making. 2 Units, Elective, 15 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

The "Data Visualisation for Data-driven Decision Making" course is designed to help students develop the skills and knowledge needed to effectively use data visualisation to support decision-making processes. In this course, students will learn how to design and create effective visualisations that communicate insights from data in a clear and compelling way.

Throughout the course, students will gain familiarity with various data visualisation tools and software, such as Tableau and Excel. They will also learn different techniques for visualising different types of data, such as time series, categorical, and quantitative data.

In addition to learning how to create visualisations, students will also develop critical thinking skills to evaluate the quality and effectiveness of data visualisations. They will learn how to interpret and communicate the insights derived from data visualisation effectively, and understand best practices for designing and presenting data visualisations.

The course will also cover principles of data-driven decision-making and how data visualisation can help in this process. Students will learn how to use data visualisation to tell compelling stories and make persuasive arguments, and will gain an understanding of ethical and privacy considerations when working with data visualisations.

The course will provide students with a comprehensive understanding of the role that data visualisation plays in decision-making processes and equip them with the skills needed to effectively create and communicate insights from data visualisations.

**Objectives**

1. Understand the importance of data visualisation in decision-making processes
2. Develop skills in designing and creating effective visualisations to communicate insights from data
3. Learn different techniques for visualising different types of data, such as time series, categorical and quantitative data
4. Learn how to use data visualisation to tell compelling stories and make persuasive arguments
5. Develop an understanding of ethical and privacy considerations when working with data visualisations.

**Learning Outcomes**

1. Identify the fundamental problems, concepts, and approaches in the design and analysis of data visualization systems.
2. Prepare data for visualization
3. Design visualizations
4. Use web technology to create visualizations
5. Create static charts, interactive Dashboards and data stories using Tableau Desktop.

**Course contents**

Various methods for presenting data for visualisation as well as how to choose between them. Fundamentals of data presentation using tables, graphs, images and video animations. Create engaging visualisations using graphs, images and video animations. Data summaries, working with tables, presenting data through graphs and plots, presenting data through video animation, creating interactive/augmented visualisation of data (ability to zoom into sections). Lab work: Practical experiments on different methods of presenting data for visualisation. Practice on how to use graphs, tables, images, and video on animation for data presentation

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities

Bayero University, Kano.

Faculty of Computing

Department of Computer Science

BSc. Computer Science

**BUK-DTS 405 Statistical Computing Inference and Modelling. 2 Units, Core, 30 LH, 45 PH**

**Senate approved relevance to mission and strategic goals of the university**

This course is designed in line with the vision and mission of Bayero University Kano to provide high quality instruction to our students, and equip them with the state-of-the-art knowledge and skills in computer science that they need to take up real-world challenges. To conduct cutting-edge research in areas of national need, frequently in collaboration with other disciplines.

**Overview**

Statistical inference is the procedure through which inferences about a population are made based on certain characteristics calculated from a sample of data drawn from that population. In statistical inference, we wish to make statements not merely about the particular subjects observed in a study but also, more importantly, about the larger population of subjects from which the study participants were drawn.

**Objectives**

1. Explore the statistical theory of modelling and analysis.
2. Derive the key results needed for statistical modelling and inference.
3. Identify statistical techniques for parameter estimation.
4. Analyse data using the theory of statistical modelling and inference to solve real-world problems.
5. Discuss the principles and results of statistical modelling and analysis using clear language and appropriate terminology.

**Learning Outcomes**

1. Demonstrate their understanding of the mathematics of statistical inference;
2. Derive the distributional results needed for statistical inference;
3. Conduct appropriate hypothesis tests for comparing two means and for regression;
4. Recognise that hypothesis tests, regression and analysis of variance belong to the same theory of linear models;
5. Demonstrate their understanding of the theory of maximum likelihood estimation for a scalar parameter;
6. Analyse a variety of datasets and fit linear regression models using R; and
7. Interpret and communicate the results of statistical analyses, orally and in writing.

**Course contents**

Fundamentals of Data Science. Methodology of extracting knowledge from big datasets as well as various tools and platforms for Data Science. What is Data and why is it important? Basic classification of Data (Structured, semi-structured and unstructured data), Scope of Data Science, Steps of Data Science Process: Data collection, Pre-processing, training, and testing. Rudiments of data visualisations; Distributions, Probability, and Simulations; Predictions and Models. Use cases in various domains such Image, Natural Language, Audio and Video. Basic introduction to knowledge extraction: Data mining, Business Intelligence & Knowledge management, Introduction to Big Data integration and intelligence, Introduction to Data Analytics, Introduction to programming. Lab work: Practical experiments on data science process steps in simulated models. Practical application of the methods and tools used in data science for prediction models with some simulated exercises. Practical experiments on how to extract knowledge; how to mine valuable data from large sets of data sets using data mining processes and methods. Learn how to integrate business intelligence in big data along with some data analytics practical exercises. Simple exercises on R programming to enhance the coding knowledge acquired during theory class.

**Minimum Academic standards requirements**

Computer Science programme’s NUC-MAS requirement facilities