HSNC University Mumbai

(2025-2026)
Ordinances and Regulations
With Respect to
For the Programme Under

Bachelor of Science in
Web and Emerging Technologies

Curriculum – First Year B.Sc. Programme
Semester-I and Semester -II

2025-2026

Preamble

Bachelor of Science in Web and Emerging Technologies
HSNC University, Mumbai

The *Bachelor of Science in Web and Emerging Technologies* is an innovative and industry-aligned undergraduate degree program developed by the **Board of Studies in B.Sc. Web and Emerging Technologies**, under the aegis of **HSNC University, Mumbai**. This programme is designed in accordance with the guidelines of the **National Education Policy (NEP) 2020**, which emphasizes flexibility, interdisciplinary learning, vocational training, skill enhancement, and academic excellence.

In the rapidly evolving digital era, there is a growing demand for professionals who possess strong foundational knowledge in web technologies, programming, UI/UX design, and modern software tools. This programme aims to equip students with both theoretical understanding and hands-on practical skills in areas such as web designing, object-oriented programming, UI/UX, game development using Unity, Python programming, and industry-relevant tools like Canva and Office Suites.

The curriculum also integrates essential elements of holistic education through courses on **Indian Knowledge Systems**, **Value-Added Courses**, **Ability Enhancement Courses**, and **Vocational Education**. The inclusion of **Skill-Based Learning**, **Mini Projects**, and **Self-Learning Evaluation** ensures that students develop critical thinking, creativity, and professional competencies needed for the 21st-century digital economy.

The programme not only prepares students to become *frontend-ready developers and designers* but also fosters an entrepreneurial mindset, adaptability, and ethical responsibility in the digital landscape. It offers students multiple pathways for higher studies, employment, and startup ventures in fields such as web development, UI/UX design, gaming, and software engineering.

The design, review, and implementation of this curriculum have been carried out by a panel of experienced academicians and industry experts to ensure that it remains current, relevant, and comprehensive in addressing global technological advancements and local industry requirements.

This document outlines the **structure, scheme of teaching and examination, syllabus, learning outcomes**, and **evaluation pattern** for **Semester I and Semester II** of the First Year of the B.Sc. Web and Emerging Technologies programme for the academic year **2025–2026**.

HSNC UNIVERSITY, MUMBAI
Board of Studies in B.Sc. Web and Emerging Technologies
K.C College.

Name of Chairperson: -

a. **Dr. Rakhi Gupta,** HOD, Chairperson , Dept. of Information Technology, K.C College, HSNC University, rakhi.gupta@kccollege.edu.in, 9619914191

2. Name of Co-chairperson: -

a. Ms. Sandhya Bhavsar, Assistant Professor, Dept. of Information Technology., K.C College, HSNC University, sandhya.bhavsar@kccollege.edu.in, 8446677463

3. Two to five teachers each having minimum five years teaching experience amongst the full time teachers of the Departments, in the relevant subject.

a. Dr. Pragati V. Thawani, Co-Chairperson**,** Assistant Professor, Dept.of Information Technology, K.C College, HSNC University pragati.thawani@kccollege.edu.in 9960782000

b.Ms. Neha Patel, Assistant Professor, Dept. of Information Technology, K.C College, HSNC University, neha.patel@kccollege.edu.in, 9820609142

c.Ms. Nashrah Gowalker, Assistant Professor, Dept. of Information Technology, K.C College, HSNC University, nashrah.gowalker@kccollege.edu.in, 9664774108

d.Ms. Aisha Shaikh, Assistant Professor, Dept. of B.Sc. Web and Emerging Technologies, K.C College, HSNC University, aisha.shaikh@kccollege.edu.in, 9224261360

4. One Professor / Associate Professor from other Universities or professor / Associate Professor from colleges managed by Parent Body; nominated by Parent Body;-

a Dr.Rasika S. Mallya Associate Professor, Navinchandra Mehta Institute of Technology& Development, Mumbai. rasikamallya@gmail.com 9819682436.

5. Four external experts from Industry / Research / eminent scholar in the field relevant to the subject nominated by the Parent Body;

a. Mr. B. Wilson Rao, Co-ordinator, Dept. of Information Technology & BVOC Software Development, Jai Hind College, Autonomous, Wilsonrao@gmail.com, 9821354297

b. Mr. Shahid Ansari, Assistant Professor, Dept. of Information Technology, Maharashtra College, Mumbai University,shahid9585@gmail.com, 9821771057

c. Mr. Kaushal Shah ,Senior Manager Reliance Power Ltd. kaushalshah78@gmail.com 9869069203.

d. Mr. Sagar Mehta, Content & Quality SME-AI, Edunet Foundation Technology, sagarmehta4268@gmail.com,7977082008

B.Sc. Web and Emerging Technologies

SEMESTER I

Subject Type	Subject	Lecture Hours	Practical Hours	No. of Credits
Major 1	Introduction to Web Designing & Application -I	45H	30H	**4(3+1)**
Major 2	Object Oriented Programming with C++	45H	30H	**4(3+1)**
Minor	Introduction to UI/UX Design	45H	30H	**4(3+1)**
VOC	Digital Typing & Office Suites		30H	**1**
IKS	Introduction to Indian Knowledge Systems - I	15H		**1**
VAC	Contemporary India: Values and Issues -I	30H		**2**
AEC	Communication Skills in English - I	30H		**2**
GE		30H		**2**
Co-curricular				**2**
	Total			**22**

SEMESTER II

Subject Type	Subject	Lecture Hours	Practical Hours	No. of Credits
Major 1	Introduction to Web Designing & Application -II	45H	30H	**4(3+1)**
Major 2	Logic, Structures & Procedures with Python	45H	30H	**4(3+1)**
Minor	Prototyping and Development in Unity	45H	30H	**4(3+1)**
VOC	Canva for Beginners		30H	**1**
IKS	Introduction to Indian Knowledge Systems - II	15H		**1**
VAC	Contemporary India: Values and Issues -II	30H		**2**
AEC	Communication Skills in English - II	30H		**2**
GE		30H		**2**
Co-curricular				**2**
	Total			**22**

The Scheme of Teaching and Examination is as under NEP
Summary

Major/Minor

Year	Sem.	Papers	Paper Code	Course Title	No of Credits	No of Lectures Hours	Total Credits
1	I	M1		Introduction to Web Designing & Application -I	4(3+1)	45+30	Major =8 Minor=4
		M2		Object Oriented Programming with C++	4(3+1)	45+30	
		Mi		Introduction to UI/UX Design	4(3+1)	45+30	
	II	M1		Introduction to Web Designing & Application -II	4(3+1)	45+30	Major =8 Minor=4
		M2		Logic, Structures & Procedures with Python	4(3+1)	45+30	
		Mi		Prototyping and Development in Unity	4(3+1)	45+30	

Vocational Courses Linked to Major/Minor

Year	Sem.	Papers	Paper Code	Course Title	No of Credits	No of Lectures Hours	Total Credits
1	I	V1		Digital Typing & Office Suites	1	30	1
	II	V2		Canva for Beginners	1	30	1

Indian Knowledge Systems

Year	Sem.	Papers	Paper Code	Course Title	No of Credits	No of Lectures Hours	Total Credits
1	I	IKS1		Introduction To Indian Knowledge Systems - I	1	15	1
	II	IKS2		Introduction To Indian Knowledge Systems - II	1	15	1

Value Added Course

Year	Sem.	Papers	Paper Code	Course Title	No of Credits	No of Lectures Hours	Total Credits
1	I	VAC1		Contemporary India: Values and Issues -I	2	30	2
	II	VAC2		Contemporary India: Values and Issues -II	2	30	2

Ability Enhancement Course

Year	Sem.	Papers	Paper Code	Course Title	No of Credits	No of Lectures Hours	Total Credits
1	I	AEC1		Communication Skills in English - I	2	30	2
	II	AEC2		Communication Skills in English - II	2	30	2

Detailed Scheme Theory Semester I

MAJOR SUBJECT 1

Introduction to Web Designing & Application-I

Course Objectives

- Understand the fundamentals of the Internet, including key applications and protocols.
- Master HTML5 for structuring web content and creating effective hyperlinks and anchors.
- Apply CSS3 for styling web pages and implementing responsive design principles.
- Create navigational aids, tables, forms, and integrate multimedia using HTML5.
- Gain familiarity with web browsers, servers, and their roles in web development.

Course Outcomes

- Understand Internet fundamentals and its applications
- Understand the concepts of World Wide Web and HTTP.
- Develop structured web pages with proper navigation and multimedia integration in HTML5
- Construct style sheets elements according to CSS3 standards.
- Construct tables, forms, and multimedia elements, using HTML5.
- Apply best practices for user-friendly and accessible web development.
- Implementation of tags, web pages, hyperlinks, style sheets.
- Implementation of multimedia elements, tables and forms.

Unit	Content	No. of Lectures
1	**1.1 Introduction to the Web:** **1.2 Introduction to Word Wide Web**	**15**
2	**2.1 Introduction, Why HTML5?** **2.2 Creating navigational aids**	**15**
3	**3.1 HTML5 Tables, Forms and Media:** **3.2 Creating user forms:** **3.3 Introduction to CSS3 :**	**15**

SLE Topics

- Creating a personal portfolio website with HTML, CSS, and optional JavaScript.
- Building a responsive landing page using media queries and flexible grids.
- Designing a blog template with HTML for structure and CSS for styling.
- Implementing an image gallery with a lightbox effect using HTML, CSS, and JavaScript.
- Developing a to-do list web app with HTML forms, CSS styling, and JavaScript functionality.
- Building a countdown timer using JavaScript for countdown logic and CSS for styling.

- Designing a contact form with HTML structure, CSS styling, and JavaScript validation.
- Developing a simple quiz game with HTML questions, CSS styling, and JavaScript logic.
- Building a responsive image slider with HTML, CSS for design, and JavaScript for transitions.

Online Resources

https://www.w3schools.com/html/html_css.asp
https://www.javatpoint.com/html-with-css
https://developer.mozilla.org/en-US/docs/Web/HTML

Reference Books

1. Web Design The Complete Reference by Thomas Powell, Tata McGraw
2. HTML5 Step by Step by Faithe Wempen, Microsoft Press, 2011.
3. HTML & CSS: Design and Build Websites Book by Jon Duckett

Practicals: Introduction to Web Designing & Application-I

Practical No.	Practical
1.	Demonstrate the use of List Tags.
2.	Demonstrate the use of block formatting Tags.
3.	Design web pages for your college containing a description of the courses, departments, teaching staff members. Use hyperlinks, list tags and image tags wherever necessary.
4.	Program to create text based and graphic based navigation bars.
5.	Create hyperlinks and anchors in web documents.
6.	Insert an image on the webpage and link another page to it.
7.	Create image maps with at least 2 hotspots of different shapes.
8.	Demonstrate checkboxes, radio buttons, and various types of buttons in HTML5.
9.	Insert an audio file to a web document.
10.	Insert a video file to a web document.
11.	Program on Stylesheets.

MAJOR SUBJECT 2

Object Oriented Programming with C++

Course Objectives

- To explain the fundamental OOP paradigm and its components: Objects, Classes, Data Abstraction, Data Encapsulation, Inheritance, Polymorphism, Dynamic Binding, and Message Passing.
- To explore advanced concepts such as friend classes, pointers to objects, and arrays of pointers to objects.
- To explain derived class declaration, constructors in derived classes, class hierarchies, multiple inheritance, and hybrid inheritance.

- To demonstrate the overloading of unary and binary operators, comparison operators, and arithmetic assignment operators.

Course Outcomes

- Understand the difference between procedure oriented and object- oriented approach.
- Describe the object-oriented programming using classes and objects.
- Implement various types of constructors. Apply the concepts of Inheritance and its types.
- Understanding the concepts of Polymorphism. Apply virtual and pure virtual function and implement the concept of late and early binding.
- Apply the concepts of Encapsulation, Data Abstraction and data hiding
- Understand the concepts of operator overloading.
- Illustration of Classes and objects in C++, friend functions, inheritance, abstract classes
- Illustration of arrays, constructor, operator and function overloading using C++.

Unit	Content	No. of Lectures
1	**1.1 Object Oriented Methodology** **1.2 Principles of OOPS:** **1.3 Classes and Objects**	**15**
2	**2.1 Use of arrays** **2.2 Constructors and Destructors:** **2.3 Program development using Inheritance:**	**15**
3	**3.1 Virtual Functions:** **3.2 Polymorphism:** **3.3 Operator Overloading In C++:**	**15**

SLE Topics

Use of arrays to represent textual data. Multidimensional arrays. Design of medium size programs. A miniature program for marks and ranks display. Command line arguments.
Multilevel inheritance, containership, hybrid inheritance.
Overloading comparison operator, overloading arithmetic assignment operator, data conversion between objects and basic types.
Dynamic memory allocation. Basic mechanisms and pitfalls. Design of a "String" class that has automated memory management. Copy constructors and destructors. Introduction to the standard library. Use of the standard library in designing programs

Online Resources

https://www.w3schools.com/cpp/ **https://www.geeksforgeeks.org/cpp-tutorial/** **https://www.youtube.com/watch?v=e7sAf4SbS_g**

Reference Books

1. Object Oriented Programming with C++ E. Balagurusamy, Tata McGraw Hill 4th, 2012.
2. Object Oriented Analysis and Design Timothy Budd TMH 3rd Edition, 2012
3. The C++ Programming Language by Bjarne Stroustrup
4. Programming in C++, Ashok N Kamthane, and Pearson 2nd Edition.

PRACTICALS: Object-Oriented Programming C++

Practical No.	Practical
1.	Programs based on Classes and methods.
2.	Programs based on friend functions.
3	Programs on arrays
4	Programs based on Constructors, Destructors.
5	Programs based on Inheritance.
6	Programs based on method overloading.
7	Programs based on Virtual functions and abstract classes.
8	Programs on Operator Overloading.

MINOR SUBJECT

Introduction to UI/UX Design

Course Objectives

- To understand the basic principles of UI and UX design.
- To analyze user behavior and design for user needs.
- To create low-fidelity wireframes and high-fidelity mockups.
- To conduct usability testing and apply feedback for improvement.
- To introduce industry-standard tools and best practices.

Course Outcomes

- Differentiate between UI and UX in digital design.
- Design intuitive web interfaces using layout, color, and typography.
- Build wireframes and clickable prototypes for websites and apps.
- Apply usability and accessibility principles in design.
- Present design ideas with confidence and justification.
- Design and prototype user interfaces including wireframes, user flows, and clickable mockups for web and mobile applications.

- Evaluate and improve UI/UX designs through analysis, usability testing, and user feedback.

Unit	Content	No. of Lectures
1	**1.1 Introduction to UI/UX** **1.2 UI Design Basics**	**15**
2	**2.1UX Design Fundamentals** **2.2Wireframing & Prototyping**	**15**
3	**3.1Usability & Testing** **3.2Mini Project**	**15**

SLE Topics

Difference Between UI and UX with Real-Life Examples
Top 5 UI Elements Used in Websites
Good vs Bad UI Design (With Example Screenshots)
Importance of Color in Web Design
Why Fonts Matter in UI Design
What is a Wireframe? (Create a Paper Sketch)
Basics of Figma / Canva for UI Design
Introduction to Responsive Design (Mobile vs Desktop)
Create a Simple Persona for a Library App
How Feedback Improves UI/UX Design

Online Resources

https://onlinecourses.nptel.ac.in/noc21_ar05/preview
https://onlinecourses.nptel.ac.in/noc24_ar23/preview

Reference Books

1. The UX Book by Rex Hartson and Pardha Pyla
2. Smashing UX Design by Jesmond Allen and James Chudley
3. Lean UX: Applying Lean Principles to Improve User Experience by Jeff Gothelf and Josh Seiden
4. Don’t Make Me Think, Revisited by Steve Krug
5. The User Experience Team of One by Leah Buley
6. The Elements of User Experience by Jesse James Garrett
7. Sketching User Experiences: The Workbook by Saul Greenberg Sheelagh Carpendale, Nicolai Marquardt and Bill Buxton

Practicals: Introduction to UI/UX Design

Practical No.	Practical
1.	Analyze 3 websites and identify good and bad UI elements
2.	Sketch a simple login screen UI on paper
3	Design a homepage wireframe for a college website
4	Create a user persona for a library app
5	Draw a user flow diagram for a food ordering app
6	Redesign a form for better usability
7	Create a clickable prototype of a 3-page mobile app
8	Conduct a peer usability test and collect feedback
9	Improve your wireframe based on feedback and present the changes
10	Mini Project: Design a simple 3-screen UI for any application (library/shop etc.)

Detailed Scheme Theory Semester II

MAJOR SUBJECT 1

Introduction to Web Designing & Application-II

Course Objectives

- Understand and apply the fundamentals of JavaScript, including client-side and server-side scripting, data types, operators, control structures, and core objects.
- Implement and manipulate web elements using the Document Object Model (DOM), including handling forms, validations, and events.
- Design dynamic and interactive web pages by integrating JavaScript with HTML and CSS (DHTML), and by handling user interactions through various event handlers.
- Utilize jQuery for efficient DOM traversal, manipulation, and effects, including selectors, animations, and event handling.
- Demonstrate practical skills in debugging, validating, and securing JavaScript code within modern web browser environments.

Course Outcomes

- Understand the syntax, data types, and control structures of JavaScript for client-side and server-side development.
- Use JavaScript operators, built-in objects, and functions to perform logical, arithmetic, and string operations.

- Apply Document Object Model (DOM) to access and manipulate HTML elements and handle browser events effectively.
- Implement form validation, event handling, and dynamic content updates using JavaScript and DHTML techniques.
- Develop interactive web applications using jQuery for DOM manipulation, animations, and event management.
- Demonstrate the ability to write clean, secure, and unobtrusive JavaScript code following best practices and modern standards.
- Apply JavaScript programming concepts such as operators, control statements, functions, and event handlers to create dynamic and interactive web applications.
- Utilize DOM manipulation and jQuery techniques, including selectors and animation effects, to enhance user experience and perform form validations.

Unit	Content	No. of Lectures
1	**1.1 Java Script:** **1.2 Core JavaScript (Properties and Methods of Each)**	**15**
2	**2.1 Events and Event Handlers :** **2.2 What is jQuery?**	**15**
3	**3.1Basic Selectors** : **3.2 HTML Manipulation**:	**15**

SLE Topics

Difference Between Client-Side and Server-Side JavaScript
Understanding JavaScript Variables and Data Types
Basic JavaScript Operators with Examples
Write a JavaScript Program for Even or Odd Number
Difference Between == and === in JavaScript
Create a Simple Form and Validate Input Using JavaScript
Change Text or Image on Button Click Using DOM
Show/Hide Content Using a Button (DOM Manipulation)
Basic Event Handling in JavaScript (onClick, onMouseOver)
Create a Digital Clock Using JavaScript

Online Resources

https://www.coursera.org/learn/javascript-jquery-json
https://www.coursera.org/learn/programming-with-javascript

References

1. Javascript 2.0 the complete reference by Thomas Powell and Fritz Schneider Tata Mcgraw Hill
2. JavaScript Pocket Reference by David Flanagan
3. JQuery Pocket Reference by David Flanagan

Practicals: Introduction to Web Designing & Application-II

Practical No.	Practical
1.	Write a JavaScript program to demonstrate operators.
2.	Write a JavaScript program to illustrate statements in JavaScript.
3.	Write a JavaScript program to display tomorrow's date.
4.	Write a JavaScript program to accept a value from the user, display whether the number is odd or even.
5.	Design a basic calculator in JavaScript.
6.	Demonstrate any 5 string functions in JavaScript.
7.	Demonstrate the onBlur, onFocus, onKeyPress and onMouseDown event handlers.
8.	Demonstrate the DOM and perform validations.
9.	Demonstrate basic selectors in JQuery.
10.	Demonstration animation effects in JQuery.

MAJOR SUBJECT 2

Logic, Structures and Procedures with Python

Course Objectives

- Develop algorithmic thinking and problem-solving skills using flowcharts, pseudocode, tracing, and dry runs.
- Understand and apply fundamental programming constructs in Python including variables, data types, control structures, and operators.
- Implement and manipulate core Python data structures such as strings, lists, tuples, sets, and dictionaries to solve real-life logic-based problems.
- Apply object-oriented programming concepts by creating classes, objects, and using encapsulation for modeling real-world entities.
- Build modular and reusable Python programs using functions, recursion, exception handling, and file operations for application-oriented problem solving.

Course Outcomes

- Apply algorithmic thinking and represent solutions using flowcharts and pseudocode.
- Write Python programs using variables, data types, input/output operations, and control structures.
- Implement logic-based solutions using Python's built-in data structures such as strings, lists, tuples, sets, and dictionaries.
- Apply object-oriented programming principles to model and solve problems using classes and objects.
- Develop modular and reusable code using functions, recursion, and exception handling.
- Perform file handling operations and build basic application-oriented programs involving reading, writing, and processing text files.
- Develop and implement Python programs using conditional logic, loops, strings, data structures, functions, recursion, and file handling to solve real-world problems.

- Design modular, reusable, and robust applications by applying exception handling and object-oriented principles, culminating in a mini project.

Unit	Content	No. of Lectures
1	**1.1 Introduction to Algorithmic Thinking and Problem-Solving** **1.2 Introduction to Python** **1.3 Control Structures** **1.3 Logic Puzzles and Basic Programs**	**15**
2	**2.1 Strings**: **2.2 Class:**	**15**
3	**3.1 Introduction to Functions** **3.2 Recursive Functions and Examples** 3**.3 Modular Programming and Code Reusability** **3.4 Exception Handling Basics** **3.5 Introduction to File Handling** **3.6 Application-Oriented Problem Solving using Functions**	**15**

SLE Topics

Difference Between Flowchart and Pseudocode Simple Calculator Using Python (if-else + operators) Palindrome Checker Using String Slicing Create and Use a Python List for a To-Do App File Handling: Read and Write a Simple Text File

Online Resources

https://onlinecourses.nptel.ac.in/noc19_cs41/preview **https://onlinecourses.swayam2.ac.in/cec22_cs20/preview** **https://www.w3schools.com/python/**

References

1. Get hands-on with Python Programming and Django web development ,Fabrizio Romano , Gastón C. Hillar , Arun Ravindran
2. Python Pocket Reference, 5th Edition,Mark Lutz,ISBN: 1449357016,O'Reilly Media, February 2014

Practicals: Logic, Structures and Procedures with Python

Practical No.	Practical
1.	**Basic Input, Output, and Arithmetic Programs** *(e.g., unit converter, bill calculator, simple interest)*

2.	**Conditional Logic-Based Programs** *(e.g., grade calculator, leap year, number classification)*
3.	**Loops and Pattern Printing Programs** *(e.g., star/number patterns, sum of digits, prime check)*
4.	**String Manipulation Programs** *(e.g., palindrome check, frequency counter, string reverse)*
5.	**List and Dictionary Handling Programs** *(e.g., sort list, remove duplicates, mark list using dictionary)*
6.	**Function-Based Problem Solving** *(e.g., modular billing system, reusable tax calculator)*
7.	**Recursive Function Programs and Class** *(e.g., factorial, Fibonacci, binary to decimal)*
8.	**Simple File Handling Programs** *(e.g., read/write student records, file summary, word count)*
9.	**Exception Handling in User Input and File Processing** *(e.g., handling division by zero, file not found errors)*
10.	**Mini Project: Logic-Based Application Using Modular Code** *(e.g., quiz system, expense tracker, to-do list manager)*

MINOR SUBJECT

Prototyping and Development in Unity

Course Objectives

- Understand the fundamentals of vectors, transformations, and 3D graphics.
- Develop proficiency in using industry-standard tools and technologies for 3D game development.
- Implement advanced graphics techniques, including lighting, shading, and texturing, to create visually appealing game environments.
- Apply principles of game design and create engaging and immersive gaming experiences.

Course Outcomes

- Apply vector manipulation techniques and geometric transformations to construct and control 3D objects.
- Use industry-standard tools such as Unity and DirectX for developing interactive 3D games.
- Implement lighting, shading, and texturing techniques to enhance visual realism in game environments.
- Integrate fundamental principles of game design to build engaging and immersive gameplay experiences.
- Develop and debug complete 3D games with interactive elements and optimized performance.

- Deploy and present 3D games across multiple platforms, demonstrating technical and creative proficiency.
- Implement object translation, rotation, scaling, and color manipulation in Unity using keyboard and mouse events.
- Develop interactive 3D game mechanics including jumping, collision detection, object destruction, and random object generation using Unity scripting.

Unit	Content	No. of Lectures
1	**1.1 Introduction And Setup** **1.2 Introduction to Scripting**	**15**
2	**2.1 Simple Movement and Input** **2.2 Managing Game Objects**	**15**
3	**3.1Simple Movement, Operations and Object Oriented Concepts:** **3.2 Advanced Programming:** **3.3Sound Effects** **3.4Publishing Games**	**15**

SLE Topics

What are GameObjects and Components in Unity?
Simple Movement Script Using C#
Creating and Using a Prefab
Difference Between 2D and 3D Game Projects in Unity
Creating a Simple Start Menu using UI elements (Buttons, Text)
Destroying an Object on Mouse Click using Script
Using Physics: Rigidbody vs Collider – What's the Difference?
How to Build/Export a Game Project for PC (Demo with .exe)

Online Resources

https://www.youtube.com/playlist?list=PL2VtxRp2QzRrwd8LOLt1IKB9LfTmmOIT9
https://onlinecourses.nptel.ac.in/

References

1. Game Development with Unity by Michelle Menard, Unity 3D Game Development by Ryan Henson Creighton.

2. Game Programming: Developing with Unity in C# for Beginners by Ortus Publishing,Let Us C# by Yashavant P.Kanetkar.
3. Mobile Game Development with Unity by Jon Manning, Paris Buttfield addison,Working environment with Holistic Game Development with Unity by Penny deByl.

Practicals: Prototyping and Development in Unity

Practical No-1-A Translation of Game Object Using Arrow Keys.
Practical No-1-B Translation of Game Object Using Arrow Keys
Practical No-1-C Translation of Game Object Using Keys A,S,W,D.
Practical No-1-D Translation of Game Object Using Velocity.
Practical No-2-A Rotation of Game Object Using and without using R key.
Practical No-2-B Changing Color of Game Object.
Practical No-3-A Handling 3D game with Mouse Event.
Practical No-3-B Collision Detection in Unity.
Practical No-4-A Jump Player in Unity.
Practical No-4-B Jump Player with Ground Checking in Unity.
Practical No.5 Double Jump Player with Ground Checking in Unity.
Practical No-6 Generate Random Cube with Color.
Practical No-7 Destroying game object on Mouse Click.
Practical No.8 Scaling (Up and down) of game object by pressing Key.

SYLLABUS FOR VOCATIONAL COURSE

Semester I
Digital Typing & Office Suites

Course Objectives

- Develop proficiency in digital typing and document formatting using word processors for professional communication.
- Enable students to create structured and data-driven documents, spreadsheets, and presentations for academic or professional use.
- Foster collaborative skills using cloud-based tools like Google Docs/Sheets for real-time editing, commenting, and feedback.

Course Outcomes

- Demonstrate the ability to create and format professional documents such as reports, resumes, and project files with headers, tables, and hyperlinks.
- Apply spreadsheet functions and visualization tools to perform calculations and represent data using charts.
- Create engaging presentations and collaborate effectively using cloud tools by utilizing commenting, version history, and shared editing features.

Practical	Content
1	**Touch Typing Practice with Speed and Accuracy Monitoring** *Practice English typing using tools like TypingClub or 10FastFingers and record progress (WPM, accuracy).*
2	**Typing and Formatting Formal Letters and Reports** *Create structured content with font styles, alignments, bulleting, spacing, and indentation in Word/Docs.*
3	**Create a Professional Resume Using MS Word or Google Docs** *Design a resume using sections, formatting styles, and tables or templates.*
4	**Prepare a Document with Header, Footer, Table, and Hyperlinks** *Simulate a project report or assignment with formatted elements and navigable links.*
5	**Create and Format a Spreadsheet with Basic Formulas** *Use formulas like SUM(), AVERAGE(), and formatting for tasks like mark sheets or budgets.*
6	**Generate Charts from Spreadsheet Data** *Create bar, pie, or line charts and analyze them visually within the spreadsheet.*
7	**Prepare a 5-Slide Presentation on a Web Technology Topic** *Use transitions, images, bullet points, and speaker notes in PowerPoint or Google Slides.*

8	**Collaborative Editing and Commenting in Google Docs/Sheets** *Share a document or sheet with peers and use comments, suggestions, and version history features.*

References

1. https://elearn.nptel.ac.in/shop/nptel/digital-skilling/?v=c86ee0d9d7ed
2. https://www.coursera.org/projects/introduction-microsoft-excel
3. https://www.coursera.org/learn/microsoft-powerpoint-work-smarter

Semester II
Canva for Beginners

Course Objectives

- Introduce students to basic principles of visual design and branding through hands-on practice with Canva for various digital content formats.
- Enable students to design professional documents and promotional materials such as resumes, social media posts, business cards, and certificates.
- Encourage creativity and digital literacy by developing personalized designs like logos, infographics, and portfolio cover pages using templates and visual elements.

Learning outcomes

- Create and customize visually appealing digital designs such as logos, social media posts, posters, and presentations using Canva tools and templates.
- Apply design principles—such as alignment, typography, color theory, and layout—to produce professional resumes, business cards, and certificates.
- Develop a digital portfolio showcasing creative work, demonstrating proficiency in using Canva for self-branding and communication.

Practical	Content
1	**Create a Personal Logo or Monogram:** Design a simple logo representing your initials, name, or personal brand using icons, typography, and color palette.
2	**Design a Social Media Post (Instagram/Facebook):** Create an eye-catching post promoting an event, product, or motivational quote using Canva templates.
3	**Make a Professional Resume in Canva:** Choose a resume template, insert personal details, and customize design, fonts, and layout. Export as PDF.
4	**Design a Poster or Flyer for a College Event:** Create an A4 or digital flyer for an upcoming event like a tech fest, seminar, or cultural show.
5	**Create a Business Card Design:** Design front and back of a business card with personal or fictional business details.
6	**Make a Presentation (5–6 Slides):** Use Canva's presentation templates to create a visually appealing slide deck on a web technology topic.

7	**Design a YouTube Thumbnail or Blog Banner:** Create an engaging thumbnail or banner for a video/blog with good use of typography and visuals.
8	**Create an Infographic Using Canva Elements:** Design an infographic (e.g., "5 Tips for Web Security" or "Steps in UI Design") using icons, shapes, and text boxes.
9	**Design a Certificate or Appreciation Template:** Make a certificate layout for participation or excellence using frames, borders, and watermark logos.
10	**Prepare a Digital Portfolio Cover Page / Poster:** Create a portfolio front page or one-page personal showcase with photo, skills, links, and design style.

References

https://www.youtube.com/watch?v=mQLJ2rF4QnQ
https://www.youtube.com/watch?v=rXLvN1FEkOE

The Scheme of Teaching and Examination for Major and Minor Course:

The performance of the learners shall be evaluated in two components: Internal Assessment with 40% marks by way of continuous evaluation and by Semester End Examination with 60% marks by conducting the theory examination.

INTERNAL ASSESSMENT: - It is defined as the assessment of the learners on the basis of continuous evaluation as envisaged in the credit based system by way of participation of learners in various academic and correlated activities in the given semester of the program.

A). Internal Assessment– 40% **40 marks**

1. **For Theory Courses**

Sr. No.	Particulars	Marks
1	Self-Learning Evaluation – Active participation in routine class instructional deliveries Overall Performance – Attendance Record	15

2. **For Courses with Practicals -**

Practical Assessment– 25 Marks (50 Marks converted into 25 Marks) Duration – 2 Hours

Sr. No	Evaluation type	Marks
1	Two Best Practical a. Evaluation of One Program	20
	b. Evaluation of Second Program	20
2	Journal	05
3	Viva	05

B) Theory Assessment– 60% **60 marks**

Theory Question Paper Pattern

The semester end examination (external component) of 60 % for each course will be as follows:

i) Theory Component Duration – 2 Hours

Theory Question Paper Pattern: -

Q-No.	Particulars	Marks
Q-1	All Units	15 Marks
Q-2	Unit-I	15 Marks
Q-3	Unit-II	15 Marks
Q-4	Unit-III	15 Marks

The marks will be given for all examination and they will be converted into grade (quality) points. The semester-end, final grade sheets and transcripts will have only credits, grades, grade points, SGPA and CGPA.

Evaluation Pattern for Vocational course for both the semesters (1 Credit)

Practical Exam will be taken of 50Marks which will be converted to 25

50Marks Practical Exam Pattern

Sr. No	Evaluation type	Marks
1	Two Best Practical a. Evaluation of One Program	20
	b. Evaluation of Second Program	20
2	Journal	05
3	Viva	05