Continuing Education and Training

with

GCIT

Less Training More Application

Visit our website

ceta.gcit.edu.bt

CET Academy

AI Integration in Web Development

This intensive, two-phase program is designed to transform you from a beginner into a high-demand Full-Stack Web Developer equipped to build AI-integrated applications. You'll start by gaining foundational, hands-on expertise in the modern JavaScript stack (React, Node.js, and MongoDB). You'll then progress to mastering Python programming and the core concepts of Artificial Intelligence, culminating in the essential ability to integrate AI models and deploy intelligent, data-driven web solutions—the key skill for tomorrow's technology landscape.

CET Academy

Why This **PROGRAM?**

AI Integration in Web Development Phase I

This intensive hands-on course equips learners with the essential skills to build and deploy modern full-stack web applications using JavaScript across the stack. Participants will start with frontend development using React, progress to backend services with Node.js and Express, and integrate unstructured data storage using MongoDB.

Through daily coding exercises and a capstone project, learners will gain practical experience in Git-based collaboration, RESTful API design, and end-to-end application deployment —laying a strong foundation for careers in web development or further specialization in AI-integrated systems.

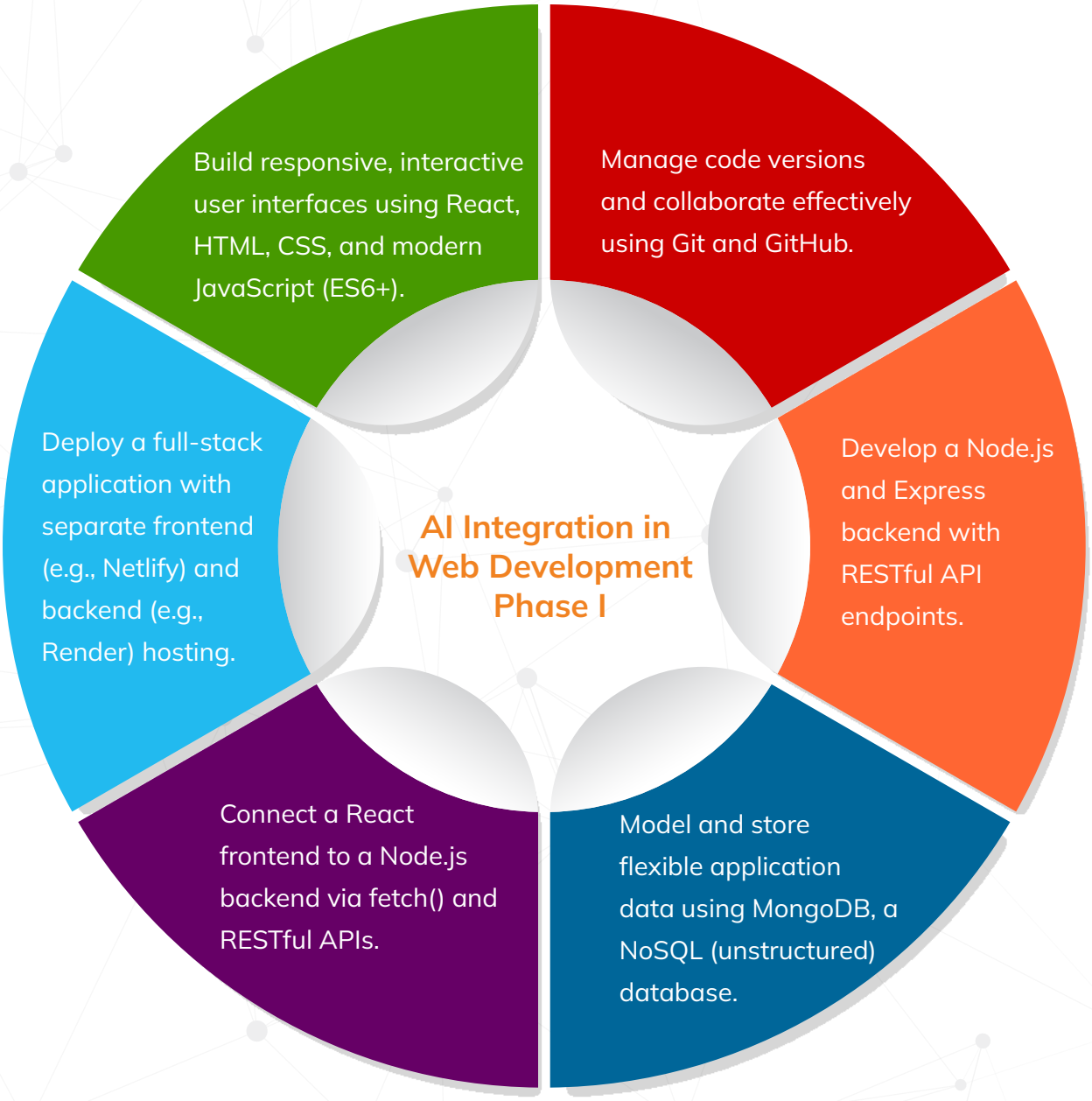
Target Audience

- Diploma, undergraduate, or graduate students in computer science or related fields
- Junior developers seeking to transition into full-stack roles
- Career switchers with basic coding exposure aiming to build deployable web applications
- Professionals preparing for technical interviews or portfolio development

Learning
OUTCOME

AI Integration in Web Development Phase I

By the end of this phase, you will be able to:



Course **STRUCTURE**

Module 1: Frontend Foundations with HTML, CSS, JavaScript & React Intro

- Core frontend triad: HTML (semantic structure), CSS (styling, Flexbox/Grid), JavaScript (DOM, events)
- Introduction to React: Why component-based UI? Virtual DOM, declarative syntax
- Setting up a React app with Vite or Create React App
- JSX, components, props, and useState for interactivity
- Building a static React app: e.g., a portfolio page or product showcase
- Developer workflow: browser DevTools, live reload

Module 2: Git, Collaboration & Project Workflow

- Git fundamentals: repositories, commits, branching, merging
- GitHub: remote repos, pull requests, collaboration
- .gitignore for node_modules, .env, build folders
- Integrating Git into React projects from day one
- Best practices: meaningful commits, branch naming (feat/, fix/)
- Hands-on: Initialize, commit, and publish a React app to GitHub

Module 3: Backend with Node.js & Express + NoSQL Databases

- Introduction to Node.js: JavaScript on the server
- Building a REST server with Express.js
- Routing, middleware, request/response handling
- Unstructured databases: Why NoSQL for modern apps?
- MongoDB fundamentals:
- Documents, collections, BSON/JSON structure
- Connecting Express to MongoDB Atlas (cloud) using the MongoDB Node.js Driver or Mongoose
- Creating a simple backend: e.g., /api/users that stores and retrieves user profiles as documents

Course
STRUCTURE

Module 4: RESTful APIs & Frontend-Backend Integration

- Designing RESTful endpoints: GET /tasks, POST /tasks
- Sending and receiving JSON data
- Consuming APIs in React: fetch() and useEffect
- Displaying dynamic data from MongoDB in React components
- Handling loading states and errors
- CORS setup in Express (cors middleware)
- Project: React frontend + Node.js backend that reads/writes to MongoDB

Module 5:Full-Stack Project with Modern Workflow

- Planning a full-stack app: data model API UI
- Building a MERN-like stack (MongoDB, Express, React, Node.js)
- Backend: Express server with CRUD routes
- Database: MongoDB Atlas (free tier)
- Frontend: React app with state management (useState, useEffect)
- Implementing key features:
- Fetch and display NoSQL documents
- Add new items via POST request
- Responsive, clean UI with CSS 0
- Version control: Separate frontend/backend repos or monorepo with clear structure
- Deployment (optional but encouraged):
- Frontend - Netlify/Vercel
- Backend - Render/ Railway

Why This
PROGRAM?

AI Integration in Web Development Phase II

This course equips learners with the foundational skills in Python programming, web technologies, database systems, and artificial intelligence, culminating in the ability to integrate AI models into full-stack web applications. Through hands-on exercises and projects, participants will elevate from understanding basic programming concepts to deploying intelligent, data-driven web solutions.

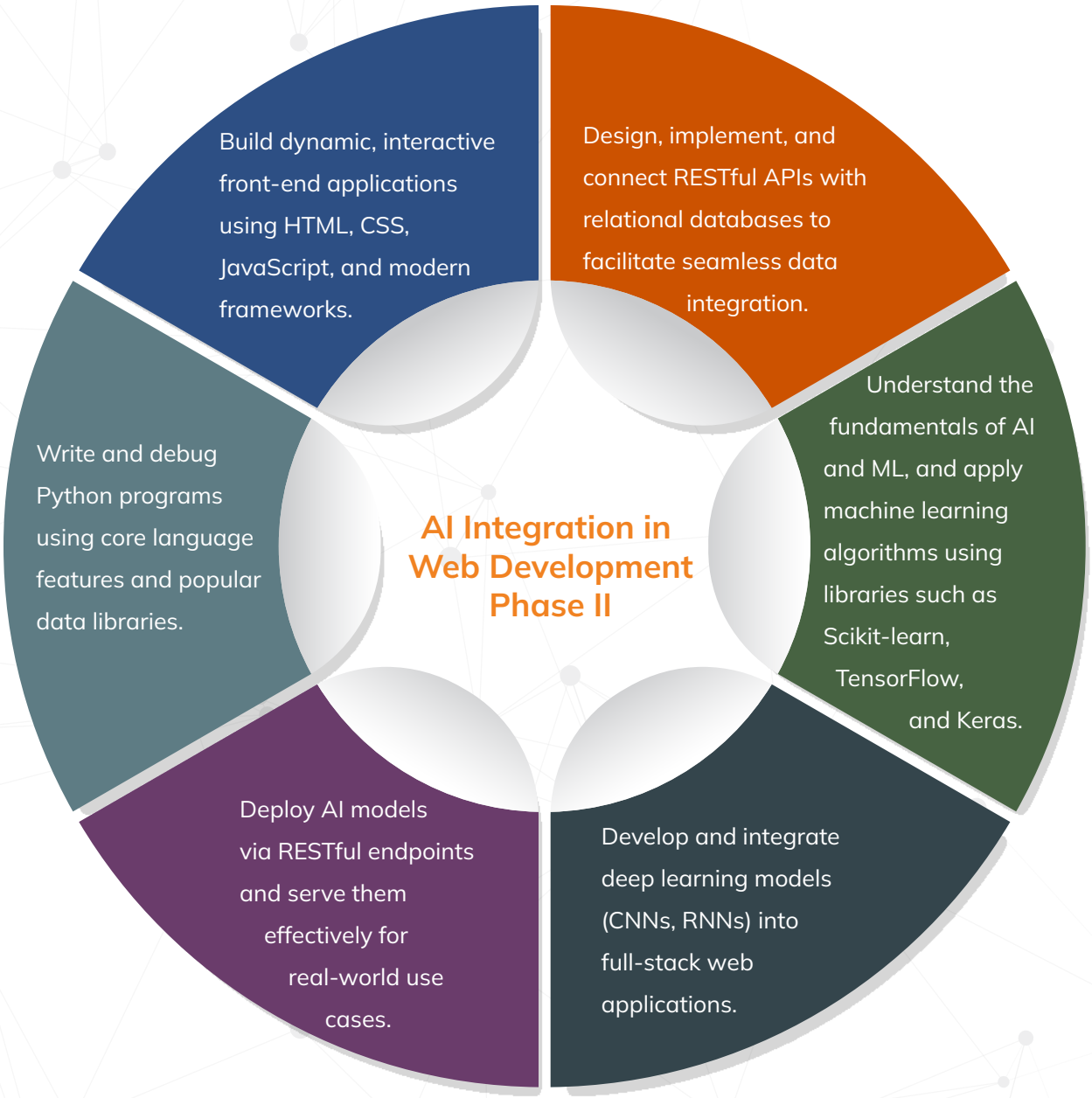
Target Audience

- Diploma, undergraduate or graduate in computer science or related fields
- Developers seeking to add AI capabilities to their web projects
- Industry professionals transitioning to AI-driven development

Learning
OUTCOME

AI Integration in Web Development Phase II

By the end of this phase, you will be able to:



Course **STRUCTURE**

Module 1: Introduction to Python

- Variables, Data Types, and Operators in Python
- Basic Input and Output in Python
- Conditional Control Flow with Python
- Iterative Control Flow with Python
- Python Functions
- Basic Data Structures in Python
- Introduction to Numpy package
- Data Processing with NumPy ndarray
- Introduction to Pandas
- Data Preprocessing using Pandas
- Data Visualisation using Matplotlib

Module 2: Introduction to Artificial Intelligence and Machine Learning

- Overview of AI and ML
- Types of Machine Learning
- Real-world Applications of AI and ML
- ML algorithms and implementation
- Overview of Python Libraries: Sklearn, Tensorflow, PyTorch, and Keras.

Module 3: Introduction to Deep Learning

- Introduction to Deep Learning
- Convolutional Neural Networks (CNNs) for Image Data with Keras Example
- Recurrent Neural Networks (RNNs) for Text Data with Keras Example
- Building a simple image classifier using CNN

Module 4: Integrating AI Model into Full Stack Application

- Saving and Loading scikit-learn Models
- Saving and Loading Keras Models
- Serving Models via RESTful API Endpoints
- Consuming RESTful API Endpoints
- Serving a Model using MLflow

Things To **KNOW**

Prerequisite:

- Basic programming knowledge (variables, loops, functions) in any language (JavaScript or Python preferred)
- Familiarity with fundamental web concepts (e.g., what a browser or website is)
- No prior experience with React, Node.js, or databases required

Workshop Details:

- Duration: 1 week (5 days), 8 hours per day (total 40 hours) for each Phase
- Delivery Mode: In-Person

Minimum Computer Specifications:

- Windows 10 / 11 Operating System or later version **(MAC OS is also supported)**
- Dual-Core CPU at 1.4 GHz or higher (Intel i4 / i6 or equivalent)
- 15 GB disk space available
- 64-bit Processor
- 8 GB RAM
- Microsoft 2010 or above
- Connection to Internet with Wi-Fi access

Capable. Elevated. Thriving.

CET Academy

ceta.gcit.edu.bt

info.ceta@gcit.edu.bt

+975 77 132 432

Thimphu, Bhutan