Bytexl's guided projects

Build job relevant skill sets by developing solutions to practical use cases

Bytexl's educators have created specialised guided projects so you can practice current technology languages / softwares such as <……..Python, Java, Nodejs, ….>

Educators should create a Guided Project for students to execute on the Bytexl App. Students should be able to complete the project in a short duration of time <Example : 2 hours > and the use case chosen should enable them to attend interviews with confidence.

Educators should create a project scenario which will enhance the job relevant skills as they guide the project through with a specially created hands-on experience available on Bytexl's app.

Note: Placeholders have been created for educators to appropriately fill in the relevant details

Guided projects should be created with the following content:

Project based learning course overview:

About the project:

Prerequisites:

What will you learn?:

Skills you will practice:

How to execute? Your learning platform:

- Practice new skills by completing job-related tasks
- No downloads or installation required. Use your Nimbus access to access all the tools.
- Practice on your desktop or laptop. This cannot be developed on your mobile phones.

Use Nimbus on Bytexl's platform:

Learn, practice and enhance job relevant skills in just <2 hours>

- Receive detailed instructions from instructors

- Gain hands-on experience solving real-world case studies
- Enhance your confidence with solutions developed on Nimbus using the latest tools and technologies

Learn step-by-step:

In this guided project, you will find your educator giving you a walk-through to complete your project in 20 hours.

Structure for educators:

How to create the use cases for students to practice?
Instructions:

Welcome to <Title of the Project>. This is a guided project which will take about 60 hours to complete.
Here are the course objectives and structure:

Course Objectives:
In this project we will focus on the following objectives:

- Objective 1: **mpower Users with Financial Insights and Control**
  Enable users to take control of their personal finances by providing easy-to-understand insights, budgeting tools, and automated financial tracking. The platform should help users set and achieve financial goals, track spending in real-time, and visualize their financial progress, enhancing overall financial literacy and decision-making.
- Objective 2: **Ensure Data Security and Privacy Compliance**
  Protect user data with high security and privacy standards, implementing robust authentication, data encryption, and compliance with regulations such as GDPR and CCPA. By prioritizing data security, the platform aims to build user trust and ensure confidentiality in managing personal financial information.
- Objective 3: **Provide Personalized Financial Recommendations Using AI and Data Analytics**
  Use AI and data analytics to deliver personalized budgeting advice, spending insights, and investment suggestions based on individual user profiles and financial behavior. By offering tailored recommendations, the platform will drive better financial habits, improve user engagement, and help users reach their unique financial objectives effectively.

1. By the end of this project you will be able to Effectively Track and Manage Your Finances
   Gain comprehensive control over your financial health with tools for budgeting, expense tracking, and spending analysis, allowing you to make informed financial decisions with real-time data.
2. **Set and Achieve Financial Goals**
   Establish personal financial goals, monitor your progress toward them, and receive actionable insights to stay on track, whether you're saving for a major purchase, paying down debt, or planning investments.

3. **Access Personalized Financial Guidance and Insights**
   Benefit from AI-driven recommendations tailored to your unique financial habits, helping you optimize savings, manage expenses, and explore investment options confidently.

You will deploy the project on the Nimbus Platform using :

You will deploy the project on the Nimbus Platform using the following tools and platforms:

1. **Docker**: For containerization, enabling consistent deployment across different environments and simplifying dependency management.

2. **Kubernetes**: To orchestrate and manage containers, ensuring scalability, load balancing, and fault tolerance for the application.

3. **GitLab CI/CD**: For continuous integration and deployment, automating testing, building, and deployment workflows on the Nimbus Platform.

4. **Prometheus and Grafana**: To monitor the application's performance and gather insights on metrics like resource usage, response times, and error rates, enabling proactive maintenance.

5. **NGINX**: As a reverse proxy and load balancer, managing incoming traffic and optimizing performance for user requests.

6. **PostgreSQL (or MongoDB)**: As the primary database service, handling structured (or semi-structured) data storage and retrieval within the Nimbus ecosystem.

These tools and platforms will ensure a reliable, scalable, and secure deployment of the project on Nimbus.

Course Structure:
This course is divided into 3 parts:

Course overview: This is the introductory reading material.

Course Overview: Building a Personal Finance Management Platform on Nimbus
This course will guide participants through the end-to-end development of a personal finance management platform, from initial concept to deployment on the Nimbus cloud platform. Designed for developers, data analysts, and IT professionals, this course combines practical knowledge of frontend and backend development, security, data management, and deployment using industry-standard tools and JavaScript-based technologies.

Course Objectives
By the end of this course, participants will:

- Build a comprehensive personal finance management application that includes budgeting, expense tracking, and financial insights.
- Integrate AI-driven recommendations for personalized user experiences.
- Implement robust data security and privacy measures.
- Deploy the application on the Nimbus Platform using Kubernetes, Docker, and CI/CD pipelines.

Modules

1. **Introduction to Personal Finance Management Tools**
    - Overview of personal finance applications and their relevance
    - Project objectives, architecture planning, and defining user requirements
2. **Frontend Development with React**
    - Building responsive, interactive UIs using React
    - Managing state with Redux, using D3.js for data visualizations
    - Creating an intuitive user experience for financial insights and goal setting
3. **Backend Development with Node.js and Express**
    - Setting up a RESTful API with Node.js and Express for handling financial data
    - Managing data with MongoDB/PostgreSQL and setting up secure data connections
    - Implementing authentication and authorization with Auth0 or Passport.js
4. **Data Analysis and AI Integration**
    - Using AI and analytics to provide spending insights and budget recommendations
    - Training models for personalized financial advice based on user data
5. **Data Security and Compliance**
    - Applying data encryption, secure data storage, and user authentication
    - Compliance with GDPR and other data privacy regulations
6. **Deployment on Nimbus Platform**
    - Setting up Docker containers and managing container orchestration with Kubernetes
    - Using GitLab CI/CD for automated testing, building, and deploying the application
    - Monitoring and performance tuning with Prometheus and Grafana
7. **Performance Testing and Troubleshooting**
    - Testing for application load, stress, and security
    - Troubleshooting common deployment issues and optimizing performance

Project structure:

The hands on project on Budget Buddy is divided into following tasks:

Task 1: **Frontend Development with React and UI Design**

Objective: Build the user interface for the personal finance management platform, ensuring an intuitive and responsive design.

- **Steps**:
    - Set up the React application and configure essential libraries (e.g., React Router, Redux for state management).
    - Develop reusable components (e.g., budget overview, transaction list, goal tracker) and ensure they are responsive.
    - Integrate D3.js or Recharts for data visualizations, creating graphs and charts for expenses, savings, and investment trends.
    - Implement authentication flows (login, sign-up) with Auth0 or Firebase.
    - Connect the frontend with the backend API (to be developed in Task 2) for real-time data display.
- **Deliverables**:
    - Component library (modular and reusable UI elements).
    - Authentication and authorization UI flows.

- Financial data visualizations and interactive charts.

Task 2: **Backend Development with Node.js and Express**
Objective: Set up the backend server to handle requests, manage data, and provide secure API endpoints.

- **Steps**:
  - Initialize a Node.js project with Express to handle RESTful API requests.
  - Define API endpoints for core features:
    - **User Management**: Register, login, profile updates.
    - **Financial Data**: CRUD operations for transactions, budgets, goals.
    - **Analytics**: Endpoints to fetch aggregated financial insights.
  - Set up MongoDB or PostgreSQL for persistent data storage, with schemas for user data, transactions, and financial goals.
  - Implement middleware for authentication (JWT tokens) and validation to secure API endpoints.
  - Write unit tests for critical endpoints (Jest or Mocha) to ensure reliability and minimize errors.
- **Deliverables**:
  - RESTful API documentation.
  - Authenticated backend server with secure endpoints.
  - Database models for users, transactions, budgets, and goals.

Task 3: **Data Analytics and AI Integration**
Objective: Implement data analytics to provide personalized financial insights.

- **Steps**:
  - Design and train a basic AI model (or rules-based algorithm) to analyze spending patterns and suggest budgeting tips.
  - Integrate data analytics libraries to calculate insights on users' expenses, income, and savings patterns.
  - Implement functions for generating personalized reports and goal-tracking suggestions based on historical data.
  - Develop endpoints in the backend to serve these insights and recommendations to the frontend.
- **Deliverables**:
  - AI-based financial recommendation system.
  - API endpoints for accessing personalized insights.
  - Code documentation for the analytics engine.

Task 4: **Deployment on Nimbus Platform**
Objective: Deploy the application with containerization, CI/CD, and monitoring.

- **Steps**:
  - Containerize the application using Docker for consistent deployment across environments.

- Set up Kubernetes configurations to deploy and manage containers on the Nimbus Platform.
- Implement GitLab CI/CD pipelines to automate testing, building, and deployment of both frontend and backend.
- Set up Prometheus and Grafana for monitoring application performance, including real-time data on usage, errors, and resource consumption.
- Finalize NGINX configuration for load balancing and reverse proxy setup.

- **Deliverables**:
  - Docker images and Kubernetes manifests.
  - CI/CD pipelines for automated deployment.
  - Monitoring dashboards for application health and performance metrics.

…

Meet your educator:
Hi I am Narvin Patidar and I will be your instructor for your course. I have about 5 years of experience in DSA and MERN stack. I have worked in multiple companies to create product end to end. I also work with different business organizations Regex Software Services etc. I have a <Bachelor's degree in Computer Science. When I am not teaching I enjoy coding and movies. I also love travelling.

Welcome to the Guided Project!

About the Nimbus Platform:

Earn a Certificate: After you have completed the **Budget Buddy** hands-on project, you should complete the Quiz to assess your knowledge. You will earn a certificate if you score 80 % or more

References:
Educators can access the links given below or any other resource for sample use cases and a basic understanding of how to create the project scenarios

Sl. No.	Some References for educators
1	https://leetcode.com/problem-list/du693s/
2	https://www.geeksforgeeks.org/coding-projects-for-beginners/#8-quiz-game
3	https://neetcode.io/courses
4	https://www.designgurus.io