



"Investing in Africa's Future"

COLLEGE OF ENGINEERING AND APPLIED SCIENCES (CEAS)

NCIS 403: WEB APPLICATION DEVELOPMENT

END OF SECOND SEMESTER EXAMINATIONS

APRIL/MAY 2025

LECTURER: MR BRAITON U. MUKHALELA

TIME: 3 HOURS

INSTRUCTIONS

1. Use the provided **API key**: EXAM2024-KEY-5678 for all API requests.
 2. Internet access is disabled. Use pre-installed tools (Node.js, React, Postman, VS Code).
 3. Submit your code as a ZIP file with the naming convention: STUDENT_ID_WEBDEV.zip.
-

Task 1: Backend Development (Node.js/Express)

(40 Marks)

1.1 Set Up a RESTful API

- Create an Express server (server.js) with the following endpoints:
 - POST /api/register: Register a user with username, email, and password.
 - POST /api/login: Authenticate a user using email and password.
 - GET /api/profile: Return user profile data (protected route requiring a valid API key).

1.2 Security & Data Handling

- Hash passwords using bcrypt (salt rounds: 10).
- Store user data in a JSON file (users.json).
- Validate input fields:
 - email must be in a valid format.
 - password must be at least 8 characters.

1.3 Error Handling

- Return appropriate HTTP status codes (e.g., 400 for bad requests, 401 for invalid API keys).
 - Handle cases where the users.json file is missing or corrupted.
-

Task 2: Frontend Development (React)

(30 Marks)

2.1 Build a Registration/Login Form

- Create a React component (AuthForm.js) with:
 - Input fields for username, email, and password.
 - Real-time validation (e.g., show red borders for invalid emails).
 - A submit button that sends data to the backend API.

2.2 Display User Profile

- Create a Profile.js component that fetches and displays user data from /api/profile.
- Include the API key in the request header as X-API-Key: EXAM2024-KEY-5678.

2.3 Error Feedback

- Display backend error messages (e.g., "Invalid API key") in the UI.

Task 3: RESTful API Testing (Postman)**(15 Marks)**

1. Create a Postman collection to test all backend endpoints.
2. Test the following scenarios:
 - Registering a user with an invalid email.
 - Logging in with an incorrect password.
 - Accessing `/api/profile` without an API key.

Task 4: Data Handling & Validation**(15 Marks)**

1. Add a PATCH `/api/profile` endpoint to update a user's username.
2. Validate that the new username is not empty and contains no special characters.
3. Update the `users.json` file accordingly.

END OF EXAMINATION