Nayna Koju

kojunayna@gmail.com | (+977) 9803574387 | Thimi, Nepal

GitHub: https://github.com/NaynaKoju
LinkedIn: https://www.linkedin.com/in/nayna-koju/

EDUCATION

BSc. (HONS), Computer Systems Engineering — **Graduated 2023**
International School of Management and Technology (ISMT) — Second Class Honors
Affiliated with **University of Sunderland**

+2 Science — **Graduated 2019**
Khwopa Higher Secondary School(KHSS) — 2.64

SEE — **Graduated 2016**
Lord Buddha English Secondary School — 3.45

KEY COMPETENCIES

Programming Languages	Python, JavaScript,
Technogies	HTML, CSS, React.js, Node.js, MongoDB, MySQL
Development Tools	Git, Postman, Anaconda

Projects

https://github.com/NaynaKoju/cli-project-python

Command Line Interface

Database Management:

- Developed and managed SQLite database using Python's sqlite3 library.
- Created tables and performed CRUD (Create, Read, Update, Delete) operations, and handled user data.

CSV Integration:

- Implemented functionality to import user data from a CSV file into the SQLite database, enabling bulk data management and integration.

User Interaction:

- Designed a console-based user interface to interact with the database, allowing users to execute various operations such as adding, querying, updating, and deleting records based on user input.

https://github.com/NaynaKoju/Heartstroke-Prediction

Heart-stroke Prediction

Model Selection:

- Evaluated various classification algorithms from scikit-learn for heart stroke prediction, including Logistic Regression, K-Nearest Neighbors (KNN), Random Forest Classifier, Gaussian Naive Bayes, and Support Vector Classification (SVC) and chose appropriate one.

Model Evaluation Techniques:

- Implemented methods like Train-Test Split and Cross-Validation to assess model performance and ensure consistency across different data subsets.

Hyper parameter Tuning:

- Applied RandomizedSearchCV and GridSearchCV to optimize model parameters and improve performance.

Performance Metrics: Used metrics such as Confusion Matrix, Classification Report, Precision, Recall, F1-Score, and ROC Curve to evaluate and compare the models, helping in selecting the most effective algorithm for prediction.

https://github.com/NaynaKoju/Web-Scrapping

Web scraping project

Web Scraping:

- Developed a Python based web scraper using BeautifulSoup and requests,Pandas, JSON, Regex, OOP to fetch and parse stock price data from the Sharesansar website.
- The scraper retrieves HTML content and extracts relevant data from tables.

Data Processing:

- Implemented functions to clean and format the scraped data, including converting string values to numerical types and handling missing or non-numeric data. The data is organized into a pandas DataFrame for easier manipulation

Data Storage:

- Saved the processed stock price data into a JSON file, making it easily accessible and structured for further use. This involves converting the cleaned data from a DataFrame to a JSON format and writing it to a file.

https://github.com/NaynaKoju/Flutter-Login-DB

Fully Funded login/registration

Login and Registration System:

- Designed and implemented a secure login and registration system featuring captcha verification, email verification, a password strength meter, password hashing, and encryption.
- Utilized the Django framework along with Python, SQLite, HTML, CSS, and JavaScript to build a robust and secure system.

https://github.com/NaynaKoju/BankApplication-JavaFunctions

Bank application

Account and User Management:

- Created fully functional bank application where I used Java classes for managing bank accounts and user profiles.
- CRUD Operations: Created handler for performing CRUD operations on accounts and users.
- **User Interface:** Developed the graphical user interface for user interaction and account management

https://github.com/NaynaKoju/Baby-buy

Baby Buy (Mobile App)

User Authentication and Management:

- Users can to log in, starting their session, and log out, which ends the session and redirects them to the login screen.

Product Management:

- Users can add new products, update existing ones, delete products, and view detailed information about each product.

Shopping Cart Functions:

- Can add products to the cart, update cart contents, view the cart's total cost, and check out to finalize purchases with payment and shipping details.

Additional Features:

- Users can send and receive messages within the app and use swipe gestures for navigation and interactions.

https://github.com/NaynaKoju/funolympic

Fun-Olympic

Developed a Live Streaming Website:

- Enabled users to register accounts, watch live games, participate in live chats, and explore additional features.
- Technologies Used: Leveraged the Django framework, Python, SQLite, HTML, CSS, and JavaScript to build and integrate the site's functionalities**.**

https://github.com/NaynaKoju/React-projects

School Website

School Website with Custom CMS (React.js, Node.js, Express, MySQL, MongoDB)
Developed a full-stack school website with an integrated custom CMS. Built the frontend using React.js and implemented a backend with Node.js, Express, MySQL, and MongoDB. Key features include JWT-based admin authentication, dynamic content management for multiple pages (Home, About Us, Socials, Students, Admission, Facilities, Fees), editable sections with rich-text editor (Tiptap), and file uploads. Designed a secure admission form submission system with auto-

incremented IDs, storing records in both MySQL and MongoDB. Ensured seamless admin dashboard functionality for adding, editing, and deleting page sections, enabling complete control over site content.

https://github.com/NaynaKoju/blog-api-nayna-koju

Blog-Backend

Built a secure and scalable RESTful API for a blogging platform with JWT-based authentication and role-based authorization.
Designed MongoDB schemas for users, posts, comments, and tags to support dynamic content management.
Implemented CRUD operations, advanced filtering, searching, sorting, and pagination for efficient blog retrieval.
Developed a comment system with ownership-based update/delete permissions.
Ensured security with password hashing and protected admin routes.
Documented and tested all APIs using Postman, ensuring reliable and consistent performance.

Certifications and Training

Python with Data Science (Certification Earned)	Broadway Infosys (11Aug, 2024-11thNov, 2024)