Luis Aries Meza Castillo

Python & Full-Stack Developer

ABOUT

Programmer with a background in Computational Technology Engineering. A responsible team player with strong logical thinking and a commitment to continuous learning. Currently enhancing skills in React, Power BI and Docker.

EDUCATION

Degree in engineering in computer technology from the Universidad Autonóma de Baja Califoria Sur.
Science Mayor

CONTACT

Phone: +52 612 1696 226
Location: Guadalajara
Email: luisaries1998@gmail.com
Github: https://github.com/Ariiies
Linkedin: https://www.linkedin.com/in/luis-aries-meza-castillo-a53727263

CERTIFICATES

PCAP-Certified Associate in Python Programing | 2021
Python Institute

Programing Essentials in Python | 2021
Cisco Networking Academy

Complete Web Development Course | 2021
Udemy

Python Masterclass: Python, Django, Flask, and TKinter | 2021
Udemy

Introduction to Data Science | 2024
Santander Open Academy & IE University

Curso de PHP Moderno | 2025
Udemy

Java Programer Course | 2021
Edutyn Academy

CCNA R&S: Introduction to Networks | 2019
Cisco Networking Academy

CCNA R&S: Routing and Switching Essentials | 2029
Cisco Networking Academy

CCNA R&S: Scaling Networks | 2020
Cisco Networking Academy

Excel Course | 2024
Santander Open academy

JavaScript Essentials 1 | 2025
Cisco Networking Academy

Laravel 12: de 0 a experto | 2025
Udemy

HABILITIES

TECNICAL SKILLS

- Programming Languages: Python, JavaScript, Java, HTML, CSS.
- Frameworks & Libraries: Flask, Django, FastAPI, Bootstrap, jQuery, React.
- Tools: Git, GitHub, Power BI (in progress).
- Cloud and Networking: CCNA certifications.
- Databases: SQL.
- English Proficiency: B2 level.

STRENGTHS

- Strong logical and mathematical skills.
- Disciplined and detail-oriented.
- Problem-solving mindset
- Team player and cooperative.
- Quick learner with a passion for technology

EXPERIENCE/ PROJECTS

Programmer at Maestros Joyeros (1 year)

- Developed and maintained a custom ERP system using Laravel + Livewire framework.
- Designed new modules and refactored existing ones to improve functionality and performance.
- Automated PDF and Excel report generation to streamline business processes.
- Planned and optimized database management strategies.
- Created Python scripts for:
  - Data extraction and analysis.
  - Bulk data imports from Excel files into the database.

FlaskNotes

Developed a web app for note management with secure session handling (Flask-Session) and efficient pagination for large datasets. Enabled users to create, edit, and delete notes smoothly.

Built a custom search engine using TF-IDF, cosine similarity, and QuickSort for efficient and relevant results. Overcame challenges like session consistency and paginated search.

Tech stack: Python, Flask, HTML, CSS, SQL.

CorpusClassifier

Designed a text classification project to predict categories within a corpus using machine learning and natural language processing techniques. Key features include:

- Corpus Processing: Utilized the Brown corpus with three categories: "lore," "learned," and "belles-lettres." Cleaned and vectorized text data using TF-IDF matrices.
- Classification: Labeled text samples and applied a neural network model built with TensorFlow and Keras to classify and predict categories accurately.
- Evaluation: Assessed model accuracy and fine-tuned predictions by iterating over preprocessing steps and model parameters.
- Challenges Overcome:
  - Managing text data for optimal input into the TensorFlow and Keras models.
  - Integrating text processing with machine learning workflows efficiently.

Tech stack: Python, pandas, scikit-learn, numpy, spacy, Keras, TensorFlow.

DetectFrameAI

A computer vision system that detects and classifies moving objects from video frames. The project involves:

- Frame Processing: Extracts frames from a video and segments them to identify moving objects by comparing differences between frames.
- Object Detection: Utilizes recursive functions and the skimage.measure library to isolate and extract the region of interest (ROI) for each detected object. Implements size thresholds to manage multiple objects in a frame.
- Image Preprocessing: Adapts cropped images for input into the VGG16 neural network for classification.
- Object Annotation: Draws bounding boxes around detected objects with their classification details.
- Video Reconstruction: Rebuilds the video from the processed and annotated frames.
- Challenges Overcome:
  - Identifying and extracting clean ROI data from segmented frames using recursive techniques and skimage.measure.
  - Managing the detection of multiple objects while ensuring high accuracy.

Tech stack: Python, OpenCV (cv2), matplotlib, numpy, skimage, keras, VGG16.