



CHRIST
(DEEMED TO BE UNIVERSITY)
BANGALORE · INDIA

Phonix: AI-Powered Smartphone Recommendation Assistant

Specialization Project (AI/ML Project)

Project Presentation

by

**JOEL JOSEPH MOTH(2448521), SYED MOHAMMED LUQMAAN(2448552),
V.B.S.R.VARMA(2448556)**

IV Trimester MSAIM

Project Guide

Dr. Bhuvana

Department of Computer Science
CHRIST(Deemed to be University), Bengaluru-29

MISSION

CHRIST is a nurturing ground for an individual's holistic development to make effective contribution to the society in a dynamic environment

VISION

Excellence and Service

CORE VALUES

Faith in God | Moral Uprightness
Love of Fellow Beings
Social Responsibility | Pursuit of Excellence

OUTLINE

- Introduction
- Alignment with SDG Goals
- Specialization Concepts to be Applied
- Existing Systems
- Proposed System
- Proposed System (Architecture Design)
- Key Features of the Proposed System
- Functional and Non-Functional Requirements
- UI Design
- Database Structure
- Feasibility Analysis
- Benefits of Proposed Work
- Anticipated Outcomes
- Plan of Work
- Module / Algorithm Design
- References

INTRODUCTION

- Phonix is an AI-powered smartphone recommendation chatbot that helps users find the best mobile phone based on their preferences and budget.
- Focuses on key parameters like price, RAM, storage, brand, and feature preferences to generate accurate recommendations.
- Highly feasible due to readily available phone data (via CSV or scraping), Python-based implementation, and plug-and-play chatbot deployment.
- Unlike basic filters, this system uses AI/ML for intelligent recommendations and model comparisons.
- Aligns with current tech trends in personalized e-commerce and consumer behavior optimization.

ALIGNMENT WITH SDG GOALS

SDG 9 – Industry, Innovation and Infrastructure: Promotes innovation in digital sales and AI integration.

SDG 12 – Responsible Consumption and Production: Helps consumers make informed, need-based purchases.

SDG 17 – Encourages integration with e-commerce platforms and manufacturers.

SPECIALIZATION CONCEPTS TO BE APPLIED

- **Rule-Based Filtering** – Based on keyword/condition mapping
- **NLP** – For interpreting user queries
- **Web Technologies** – Chat UI using Streamlit
- **Database** – Storing smartphone specs and queries in a csv file
- **Future ML Integration** – For learning user preferences over time

EXISTING SYSTEM

Current Market Options:

Amazon, Flipkart, GSM Arena: Filter-based UI

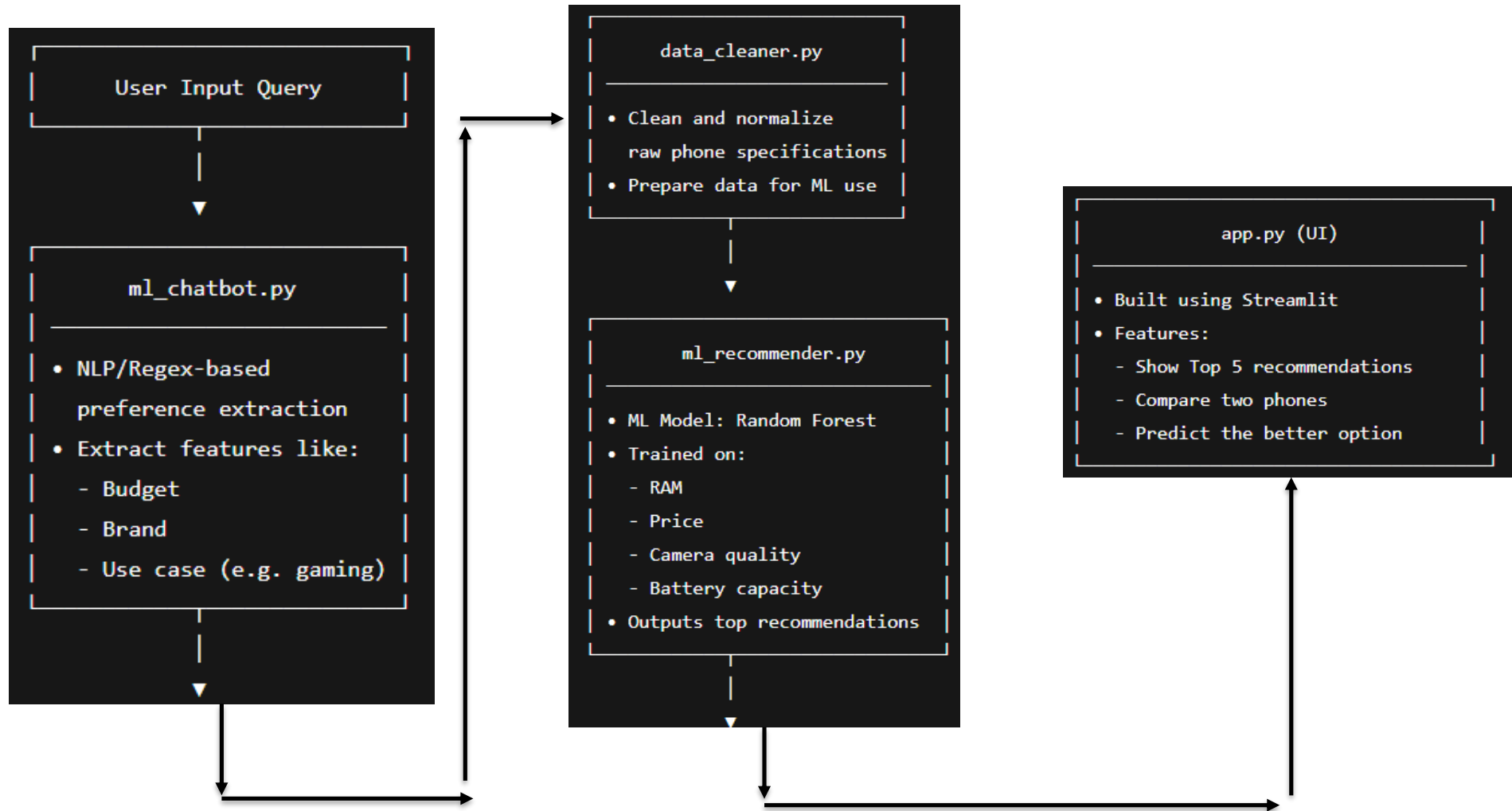
Limitations:

- No conversational input
- Overwhelming filter lists
- Lacks personalization
- Requires product knowledge to filter correctly

PROPOSED SYSTEM

- **Functional Description:** Accepts user input → AI processes data → returns phone recommendations and comparisons.
- **Solution Architecture:** Data Preprocessing → AI Engine → Recommendation Module → UI Chatbot → Output.
- **Software Requirements:** Python, Pandas, Scikit-learn, Streamlit/Flask, NLTK, CSV/Database support.
- **Hardware Requirements:** 8 GB RAM, i5 processor or better, 256 GB storage.

PROPOSED SYSTEM (ARCHITECTURE DESIGN)



KEY FEATURES OF THE PROPOSED SYSTEM

- Conversational phone search
- Rule-based accurate filtering
- Cross-platform UI
- Lightweight backend
- Query logging for ML pipeline
- Scalable for large datasets

FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

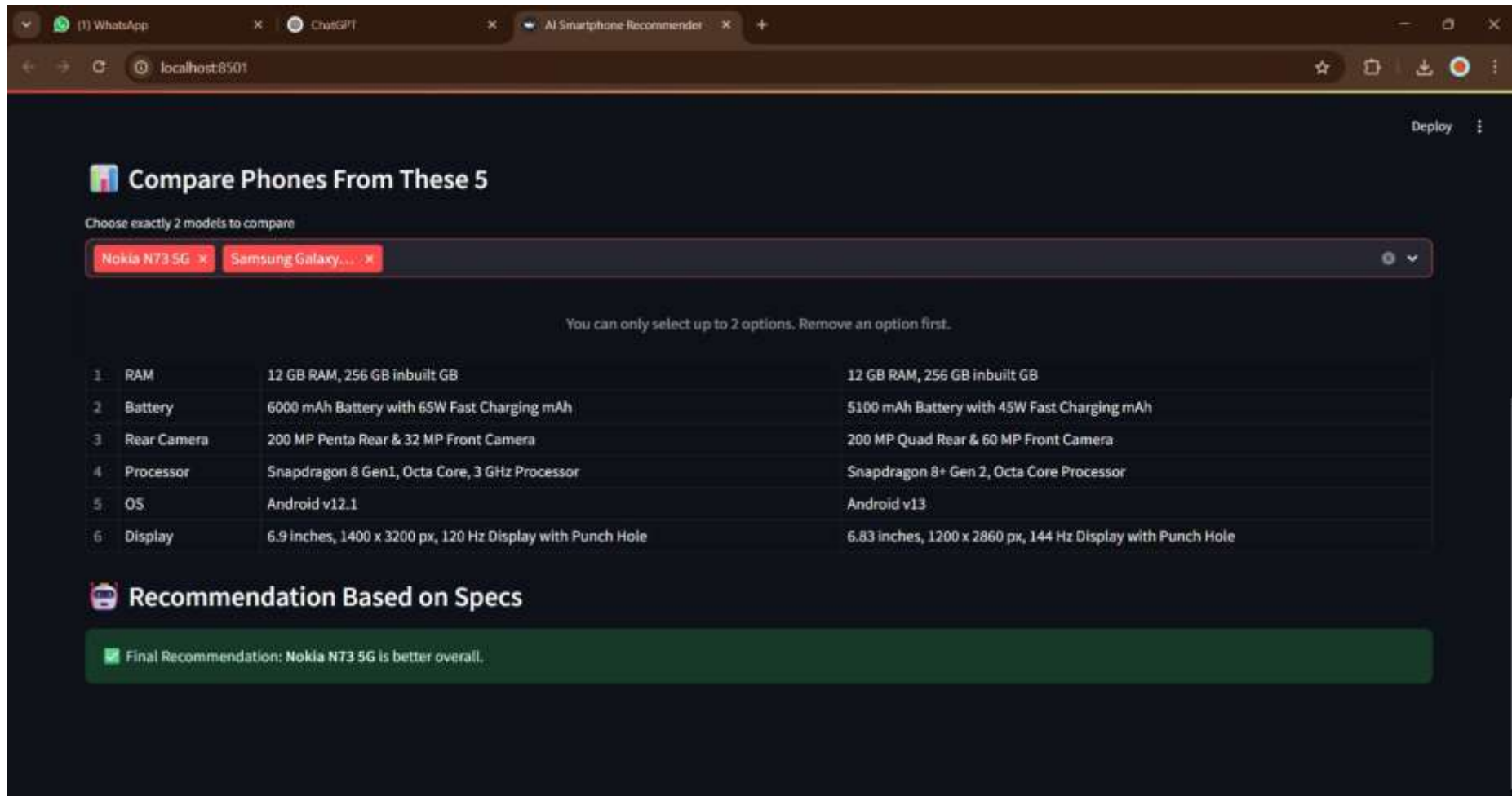
Functional:

- Extract preferences from user input
- Match phones to rules
- Display ranked results
- Log user interactions

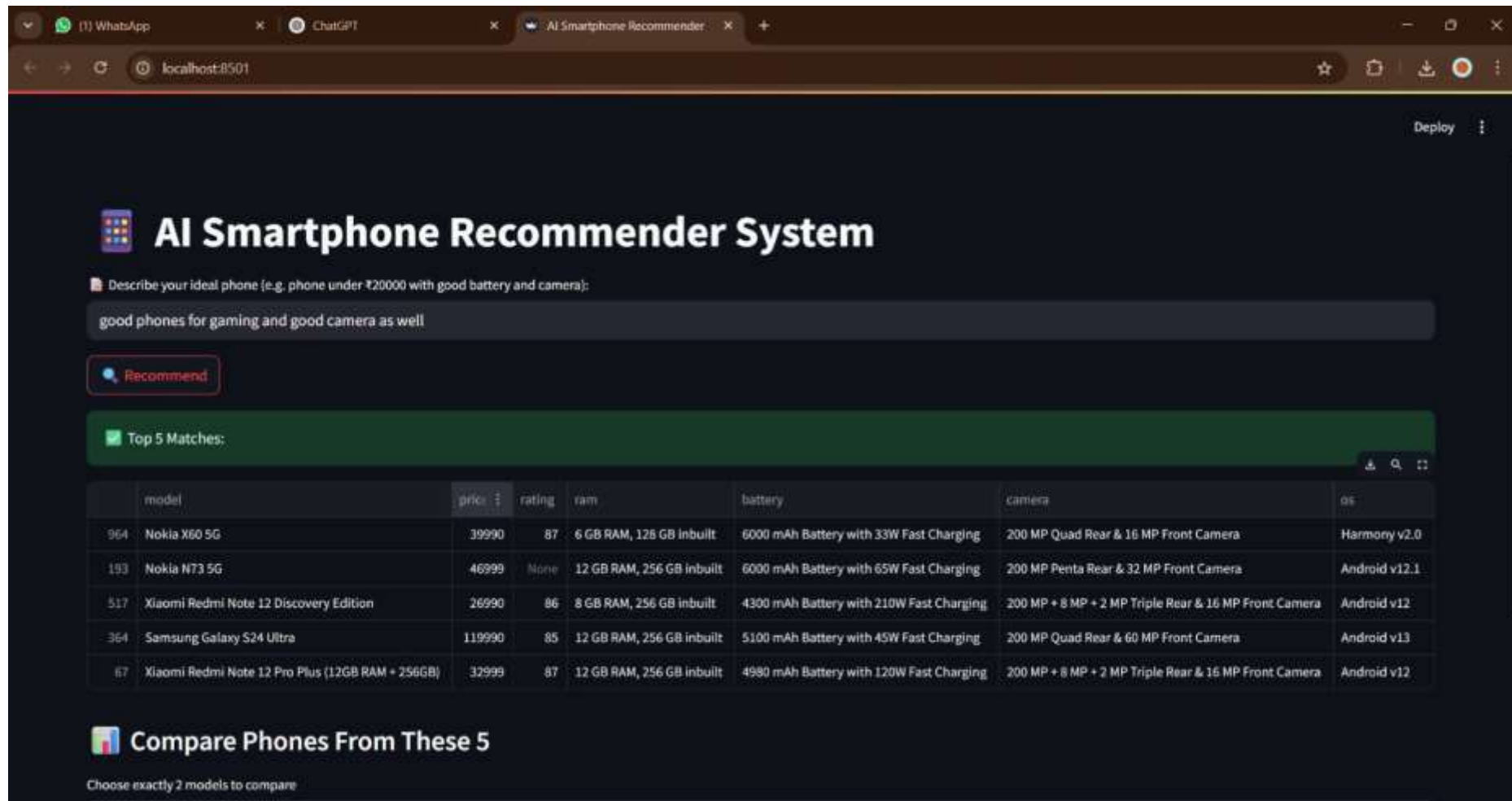
Non-Functional:

- Low latency responses
- Responsive UI
- Scalable backend logic
- Modular rule management

UI DESIGN



UI DESIGN



DATABASE STRUCTURE

- **model** – Name of the smartphone model.
- **price** – Selling price of the phone.
- **rating** – User rating or expert score for the device.
- **processor** – Type/brand of the chipset used.
- **cpu cores** – Number of processing cores in the CPU.
- **processor speed** – Clock speed of the CPU in GHz.
- **ram** – Amount of Random Access Memory (RAM) available.
- **storage** – Internal storage capacity of the device.
- **memory card support** – Indicates if external memory cards are supported.
- **os** – Operating system (e.g., Android, iOS).
- **display size** – Physical size of the phone's screen (in inches).

DATABASE STRUCTURE

- **display resolution** – Number of pixels in the screen (e.g., FHD+).
- **refresh rate** – Screen refresh rate (in Hz).
- **rear camera** – Resolution and configuration of the rear camera(s).
- **front camera** – Resolution of the front/selfie camera.
- **battery** – Battery capacity in mAh.
- **charging speed** – Maximum charging power (e.g., 67W).
- **charging port** – Type of port used (e.g., USB-C).
- **sim slots** – Number and type of SIM card slots.
- **network support** – Supported network technologies (e.g., 5G).
- **nfc** – Indicates presence of NFC (Near Field Communication) support.

FEASIBILITY ANALYSIS

- **Time:** 6–8 weeks to develop and test MVP.
- **Cost:** Minimal – open-source tools and libraries used.
- **Implementation Issues:** Depends on regular dataset updates; initial version uses static data (CSV).
- **Risk:** Low, with high scalability potential.

BENEFITS OF PROPOSED WORK

- Personalized, AI-based suggestions for better user experience.
- Businesses can easily integrate it into their platform.
- Simplifies decision-making for customers.
- Enables comparison and insight-driven purchases.

ANTICIPATED OUTCOMES

- A working AI chatbot that suggests the best phones.
- A scalable system that can integrate with real-time data sources.
- Increased efficiency and satisfaction in mobile shopping.
- An adaptable solution for businesses to improve sales and service.

PLAN OF WORK

Methodology:

Data Collection → Preprocessing → Model Training → UI Design → Integration → Testing

Timeline:

- Week 1: Data Prep
- Week 2–3: AI Model Development
- Week 4: Interface Building
- Week 5: Integration & Testing
- Week 6: Final Review & Deployment

MODULE/ALGORITHM DESIGN

1. `data_cleaner.py` – Data Preprocessing

- Cleans raw smartphone dataset.
- Removes unwanted symbols (₹, commas) from price.
- Extracts numeric values from fields like RAM, storage, battery, camera.
- Outputs a cleaned DataFrame for machine learning.

2. `ml_recommender.py` – ML Recommendation Engine

- Loads cleaned data for training.
- Uses features like price, RAM, battery, and camera.
- Trains a Random Forest model to compute a suitability score.
- Returns top-N phone recommendations based on user preferences.

MODULE/ALGORITHM DESIGN

3. ml_chatbot.py – Preference Extraction

- Extracts user preferences from natural language input.
- Uses rule-based logic or TF-IDF for MVP version.
- Parses key details like budget, usage focus (e.g., gaming, photography).

4. app.py – Streamlit UI

- Provides a user-friendly web interface.
- Accepts user queries via text input.
- Displays top 5 recommended phones.
- Allows comparison between two selected phones.
- Predicts the better phone using the trained model.

REFERENCES

- **Smartphone Dataset (GitHub)** – <https://github.com/koyarqasm/SmartPhone-dataset/blob/main/smartphones.csv>
- **GSM Arena** – <https://www.gsmlarena.com>
- **Scikit-learn Documentation** – <https://scikit-learn.org>
- **Streamlit Documentation** – <https://docs.streamlit.io>
- **spaCy** – <https://spacy.io>
- **Natural Language Toolkit (NLTK)** – <https://www.nltk.org>
- **United Nations SDG Portal** – <https://sdgs.un.org/goals>