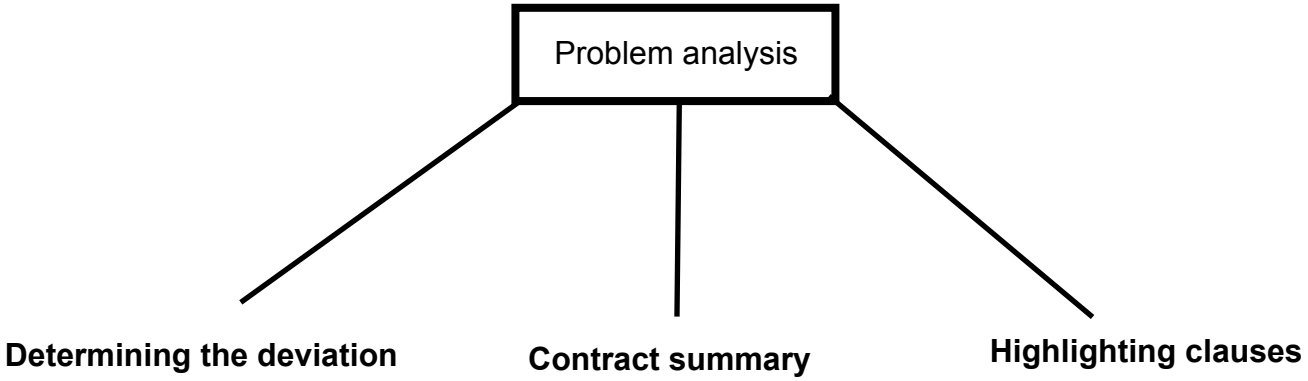
Problem Statement

Business Contract Validation-

To classify content within the Contract Clauses & to determine deviations from Template & highlight them



Unique Idea Brief (Solution)

Automated Clause Classification:
Implement NLP techniques like Named Entity Recognition (NER) and text classification to automatically classify clauses into predefined categories (e.g., Payment Terms, Confidentiality).
Standard Template Creation:

Develop a comprehensive standard contract template that includes all necessary clauses and terms.
Deviation Detection and Comparison:

Use semantic analysis and rule-based systems to compare extracted clauses from contracts against the standard template.
Employ string matching and similarity measures (e.g., cosine similarity) to identify and quantify deviations.
Highlighting and Visualization:

Visualize deviations in contract clauses through color coding or annotations to facilitate easy identification.
Automated Report Generation:

Automatically generate detailed reports highlighting identified deviations, their nature, and potential impact, ensuring transparency and compliance

Features Offered

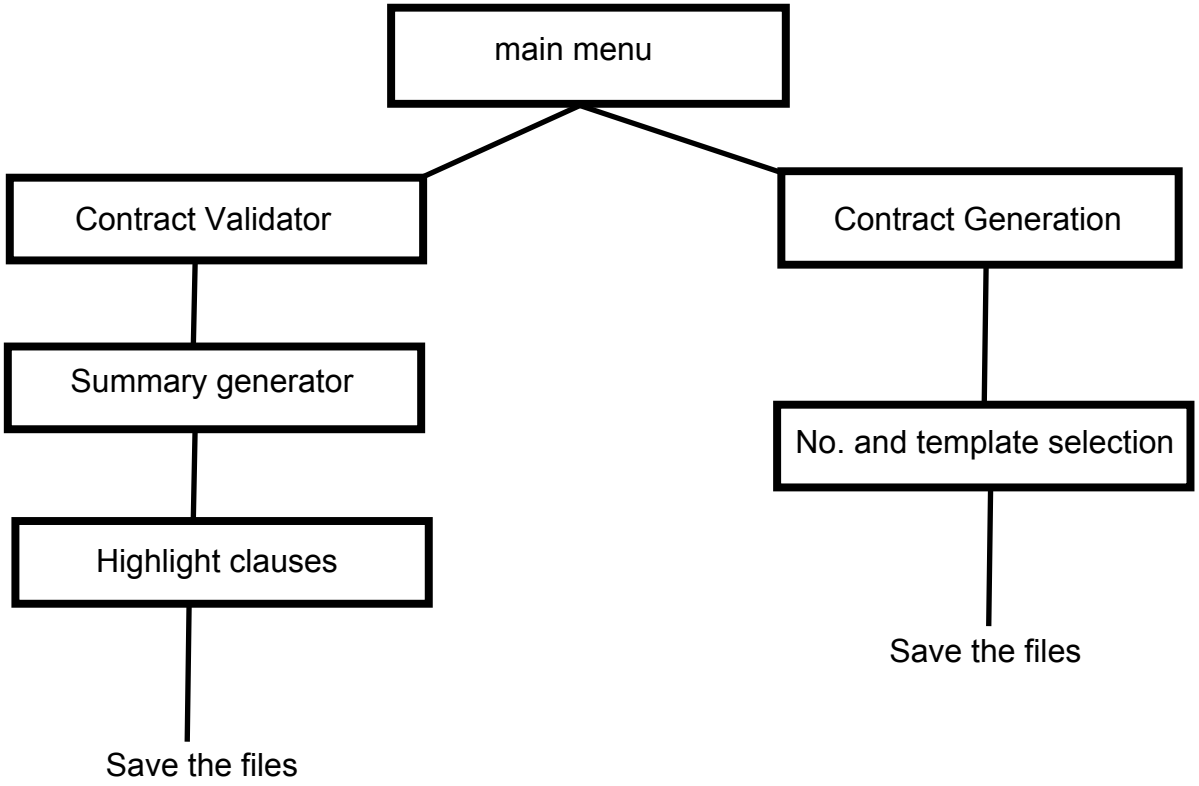
Contract Generation:

Functionality: Users can generate standard template contracts based on predefined categories and clauses.
Implementation: Utilize form inputs or dropdowns to select clause types (e.g., Payment Terms, Confidentiality, Termination) and dynamically generate contracts using pre-defined templates or user-provided inputs.
Benefits: Provides a streamlined way to create consistent, legally compliant contracts tailored to specific business needs.
Contract Validation:

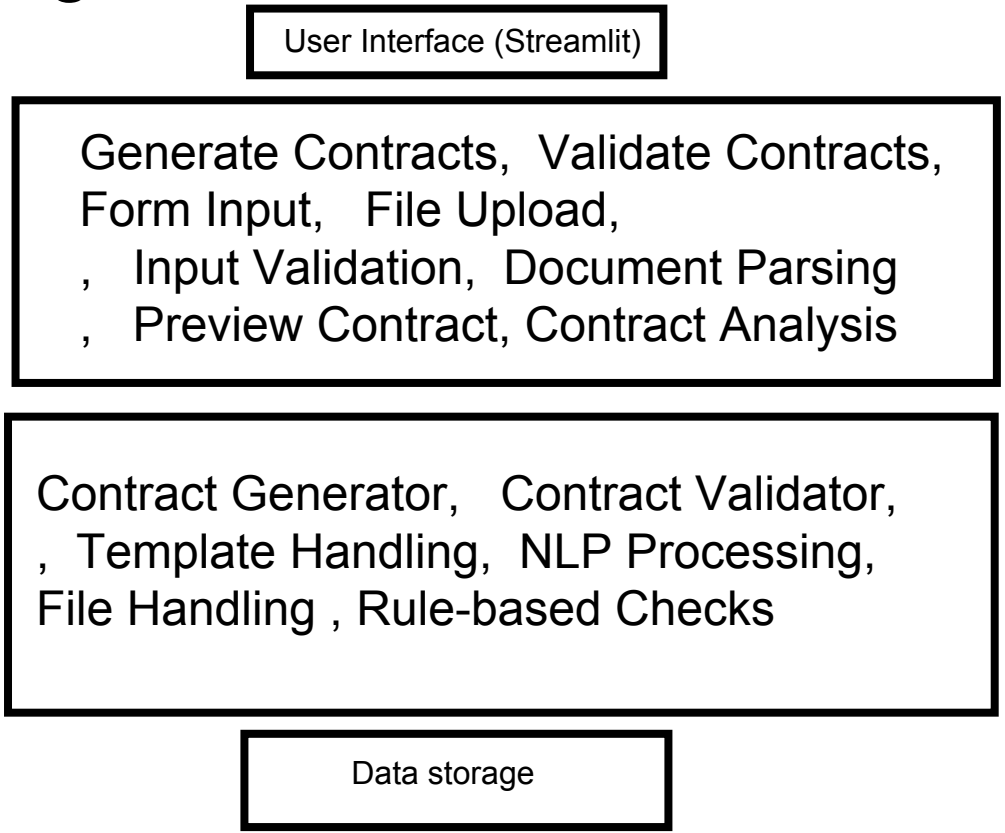
Functionality: Validate contracts against predefined standard templates or regulatory requirements.
Implementation: Implement NLP techniques (e.g., NER, text classification) to classify clauses and identify deviations from the standard template.
Benefits: Enables automated checking for compliance, highlighting discrepancies and potential risks before finalizing contracts.
Interactive User Interface:

Functionality: Provide a user-friendly interface with interactive elements for contract generation and validation.
Implementation: Use Streamlit's widgets and components (e.g., forms, buttons, sliders) to create an intuitive UI for users to input contract details, select options, and view validation results.
Benefits: Enhances user experience by simplifying complex tasks such as contract creation and validation, improving usability and efficiency.

Process flow



Architecture Diagram



Technologies used

Python 3.9 Colab notebook

Python libraries
streamlit
fitz
fpdf
re
faker
random
tempfile
difflib
SequenceMatcher

Conclusion

Developing a Python Streamlit application for contract management integrates user-friendly interfaces with robust backend capabilities to streamline contract lifecycle processes effectively. The application allows users to generate standardized contracts via intuitive form inputs and validates existing contracts by uploading documents for analysis.

Backend services include a Contract Generator for dynamic contract creation based on user inputs or predefined templates, alongside a Contract Validator that utilizes NLP techniques to extract and classify clauses, comparing them against standard templates. This ensures accurate identification of deviations and compliance issues, supporting regulatory adherence and operational reliability.

Centralized data storage for contracts and templates enables efficient data management, ensuring consistency and accessibility across operations. The application's modular architecture facilitates scalability and updates, accommodating evolving business needs. By automating validation and highlighting discrepancies, it enhances compliance management and operational efficiency, making it a crucial tool for modern contract management practices.