

# **Business Requirement Analyzer**

Presented by Prerna Kumari

### The Challenge of Manual Analysis

Business requirement analysis in software development is often manual, leading to significant challenges:

### **Time-Consuming**

Manual processes cause delays in project timelines.

#### **Error-Prone**

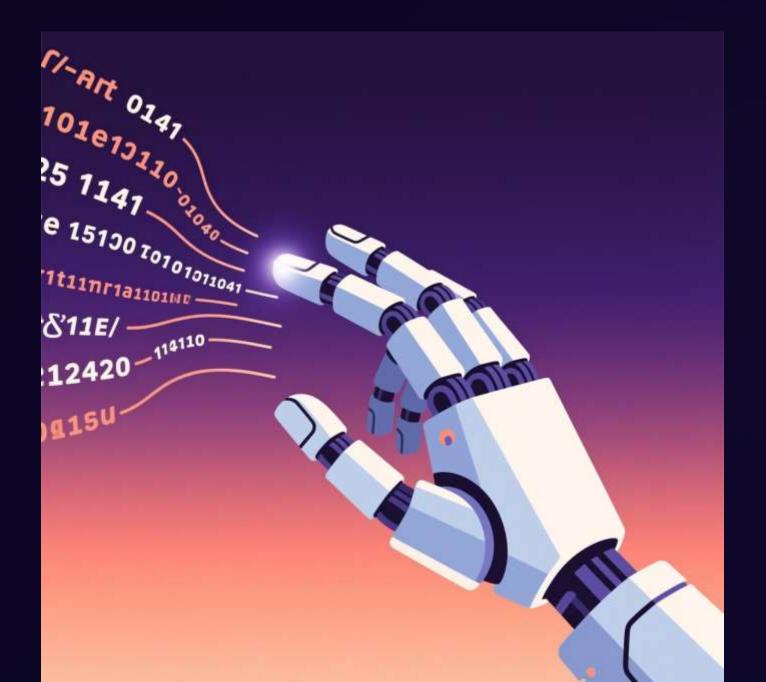
High potential for inconsistencies and ambiguities.

### **Scalability Issues**

Difficult to manage large-scale projects effectively.

### **Automating Requirements with NLP**

Our solution is an automated tool leveraging Natural Language Processing (NLP) to streamline business requirement analysis.



#### **Faster Analysis**

Automates categorization and keyword extraction.

#### **Improved Consistency**

Reduces human error and ensures uniformity.

#### **Enhanced Traceability**

Provides clear links between requirements.

### NLP's Role: Classification & Extraction

NLP automates key tasks like requirement classification and keyword extraction. This project primarily uses a rule-based approach for simplicity and efficiency.



### Requirement Classification

Categorizes requirements based on predefined keywords.



### **Keyword Extraction**

Identifies important terms for summarisation.



### **Future: Machine Learning**

Potential for more advanced and accurate analysis.

### System Architecture: Three Layers

The system follows a client-server model, structured into three distinct layers for robust functionality.

**Presentation Layer** User-friendly web interface (HTML, CSS, JavaScript). **Application Layer** 2 Core logic (Python, Flask) processes NLP and interacts with the database. **Data Layer** Persistent storage of requirements using SQLite.

### Core Logic & Database Design

A simple SQLite database stores requirements, categorised using keyword matching. Keyword extraction uses spaCy's part-of-speech tagging.

#### **Database Fields**

- Raw text
- Category
- Keywords

### **Categorisation Example**

Terms like "feature" or "process" classify a requirement as "Functional."

### **Keyword Extraction**

spaCy identifies important nouns and proper nouns for summarisation.



127.0.0.1:5000

Category: Technical

Keywords: system, user, edit, product, price, description

#### Saved Requirements

ID	Description	Category	Keywords
1	A daily ETL pipeline should aggregate transaction data from multiple sources, clean the data for inconsistencies, and store the results in a centralized data warehouse for business intelligence reporting. The system must support generating ad hoc reports with filters for date range, customer segments, and product categories. All raw and processed data must be retained for at least five years to comply with regulatory requirements.	Functional	ETL, pipeline, transaction, data, sources, data, inconsistencies, results, data, warehouse, business, intelligence, reporting, system, ad, filters, date, range, customer, segments, product, categories, data, years, requirements
2	The application needs to integrate with a third-party payment gateway API to process transactions securely	Functional	application, party, payment, gateway, API, transactions
3	The application shall generate a monthly sales report showing total revenue and product sales data for each quarter.	Data	application, sales, report, revenue, product, sales, data, quarter
4	The system must allow a logged-in user to create, edit, and save a new product listing with a price and description.	Technical	system, user, edit, product, price, description

### Intuitive Web Interface

The web interface enables users to input, analyse, and save requirement descriptions. Results are displayed instantly, with options to view all saved requirements.

### **Input & Analyse**

Users can easily submit requirement descriptions.

### **Instant Display**

Categorised results and extracted keywords appear immediately.

### **Data Management**

View all saved requirements or clear data as needed.

### Future Enhancements & Conclusion

This project provides a strong foundation, with potential for significant future improvements.

### **Key Future Work**

- Machine learning for accuracy.
- Semantic understanding (word embeddings).
- Scalable databases (PostgreSQL/MySQL).
- Richer summaries and integration APIs.

#### Conclusion

The Business Requirement Analyzer automates initial analysis, boosting speed and consistency. While current limitations exist, it forms a solid base for future enhancements.

## Thank You

Prerna Kumari