

VeriGen - NLP Requirements Analysis System

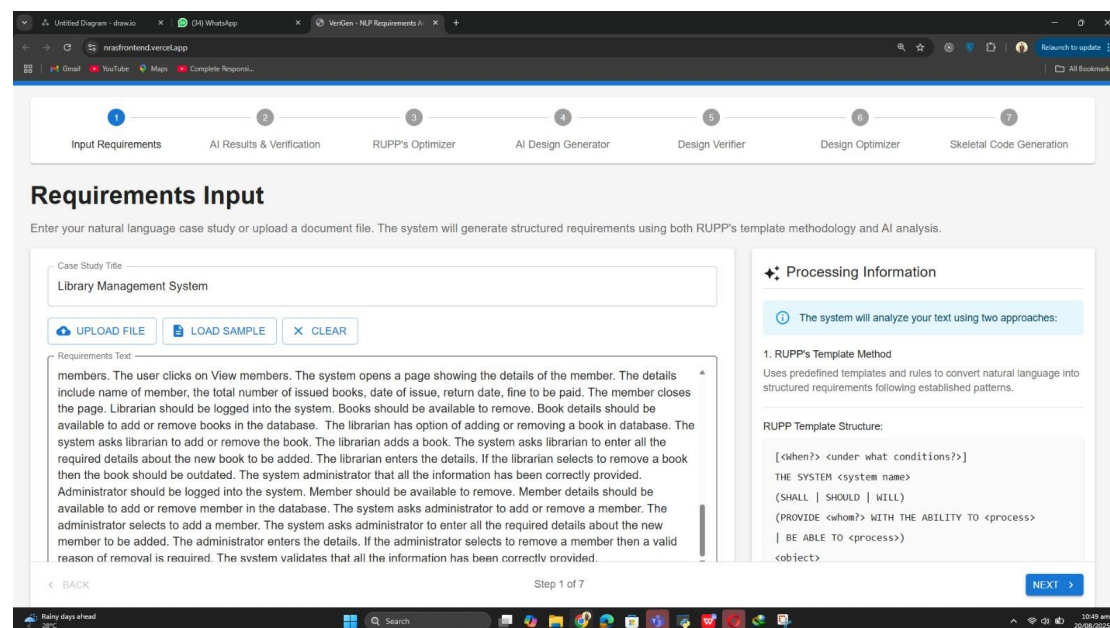
Quick User Guide

STEP 1: Input Requirements

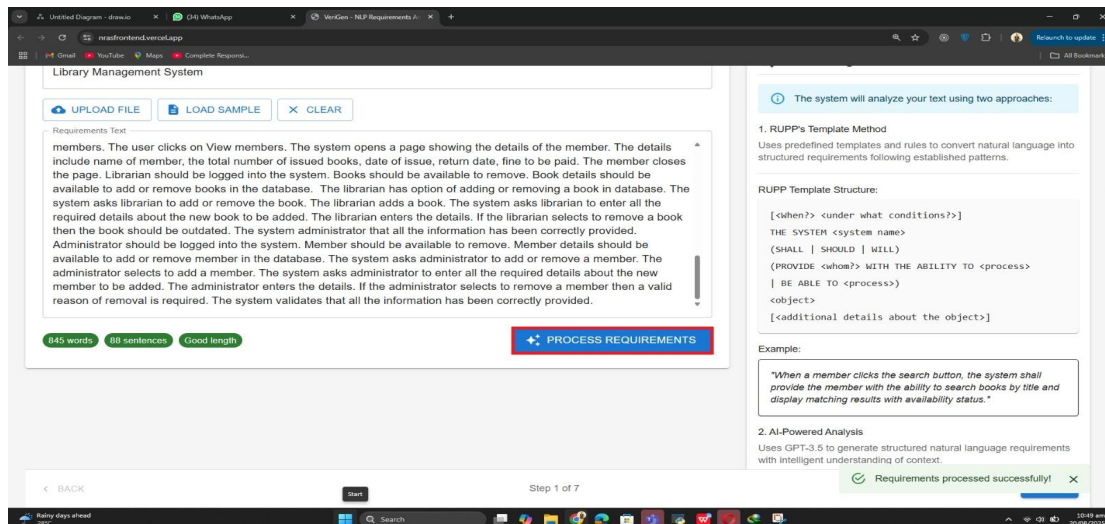
1. Enter your **Case Study Title** in the text field (e.g., "Library Management System")
2. Type or paste your requirements description in the **Requirements Text** area
3. Alternatively, click [**UPLOAD FILE**] to upload a document or [**LOAD SAMPLE**] to load an example
4. Click the [⚡ **PROCESS REQUIREMENTS**] button

Result: System will analyze your text and show "✓ Requirements processed successfully!"

- Shows the Requirements Input screen with text area and process button.



- Shows the processing complete with word count statistics and success message

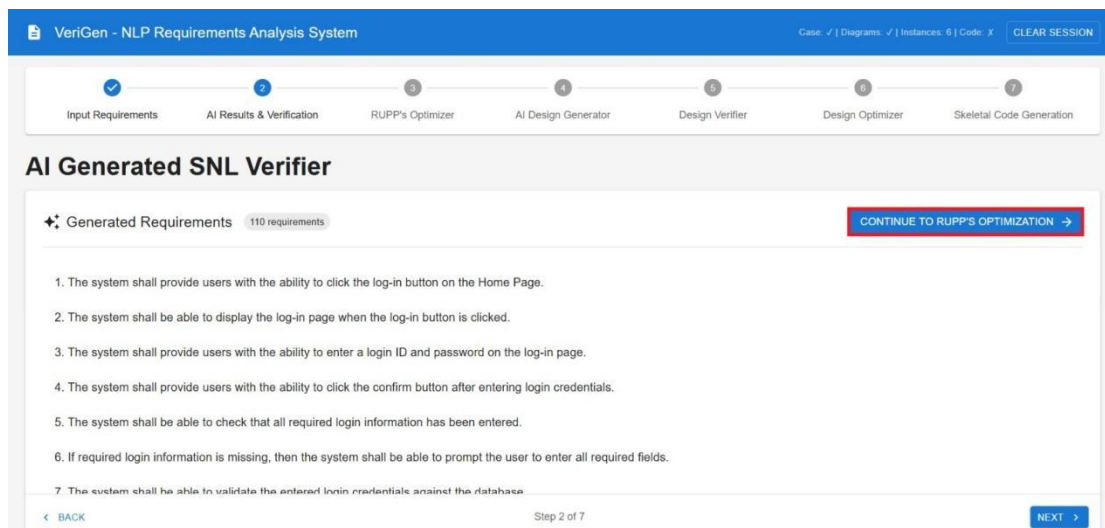


STEP 2: AI Results & Verification

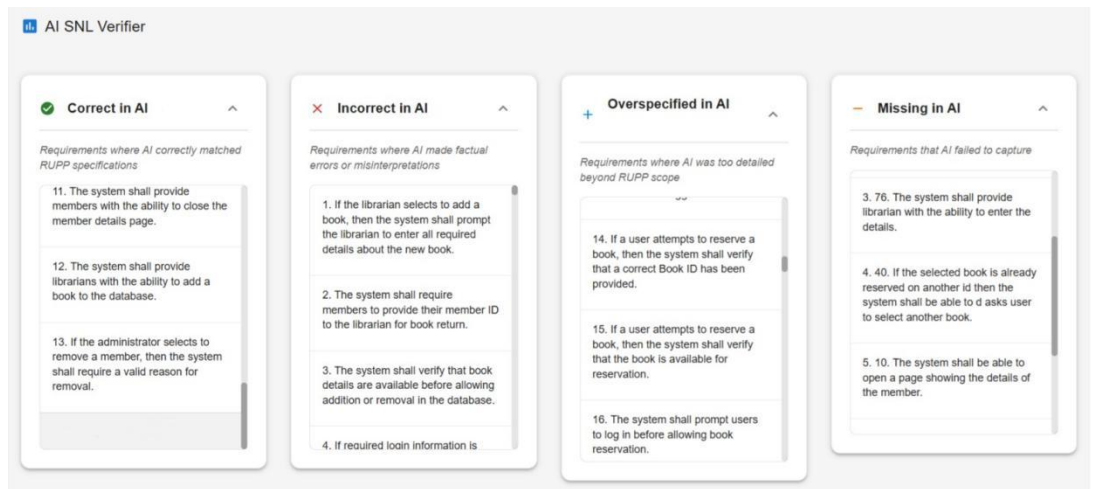
1. Review the AI-generated structured requirements list (110 requirements)
2. Click the [CONTINUE TO RUPP'S OPTIMIZATION →] button

Result: System displays AI Generated SNL Verifier with categorized requirements (Correct, Incorrect, Overspecified, Missing)

- Shows AI Generated SNL Verifier with requirements categories



- Shows detailed view of AI SNL Verifier categories (Correct in AI, Incorrect in AI, Overspecified, Missing)

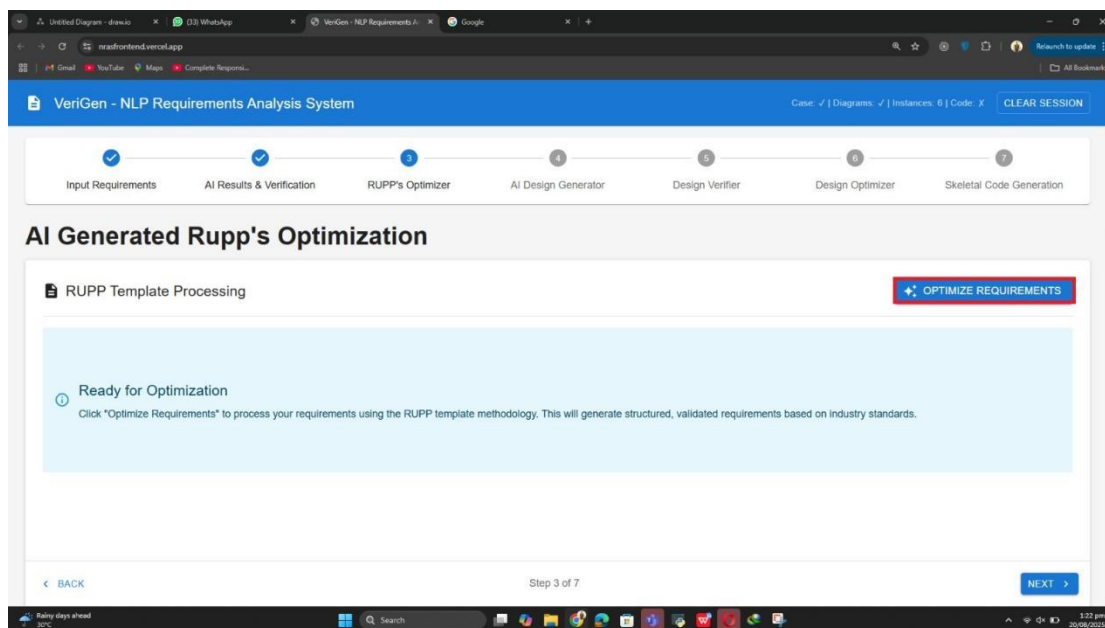


STEP 3: RUPP's Optimizer

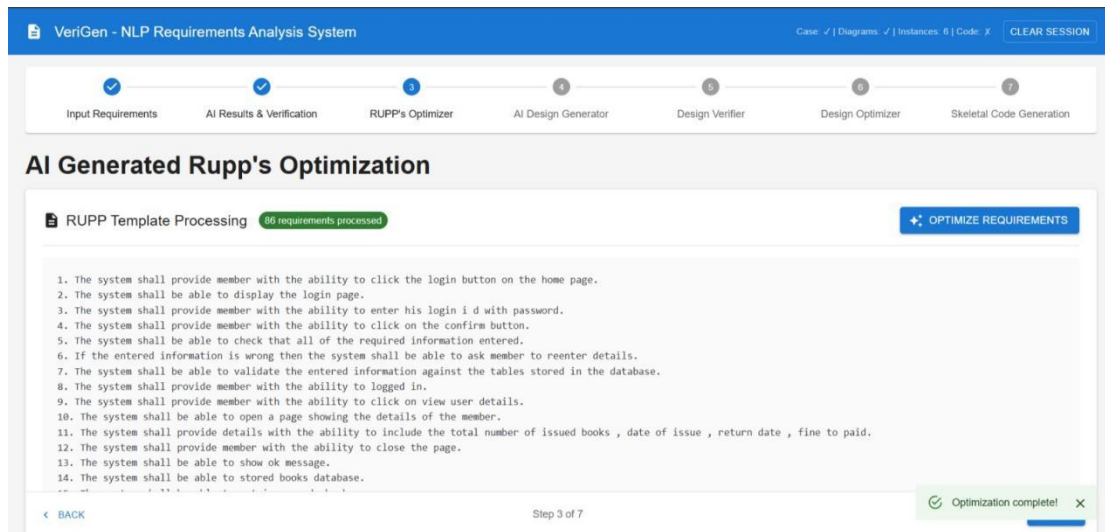
1. Click the [↗ OPTIMIZE REQUIREMENTS] button
2. Wait for optimization to complete
3. Click the [NEXT →] button

Result: System shows "✓ Optimization complete!" with 86 optimized requirements in RUPP template format

- Shows RUPP's Optimizer ready state with Optimize Requirements button



- Shows completed optimization with structured requirements list

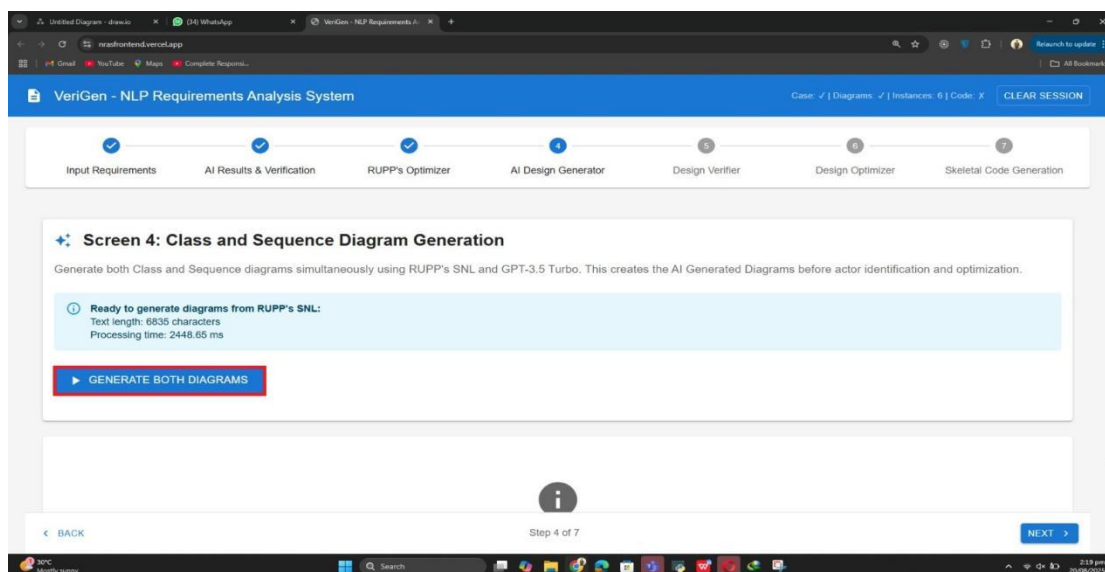


STEP 4: AI Design Generator

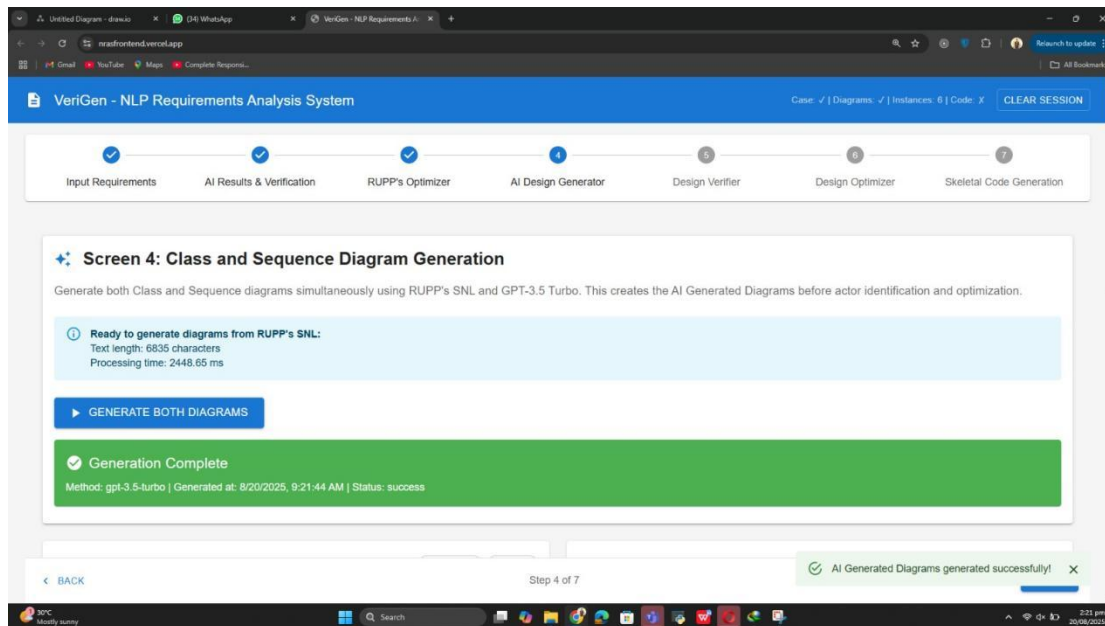
1. System displays "Screen 4: Class and Sequence Diagram Generation"
2. Click the [▶ GENERATE BOTH DIAGRAMS] button
3. Wait for generation to complete

Result: System shows "✓ AI Generated Diagrams generated successfully!" with Class Diagram (13 elements, 127 lines) and Sequence Diagram (16 elements, 141 lines)

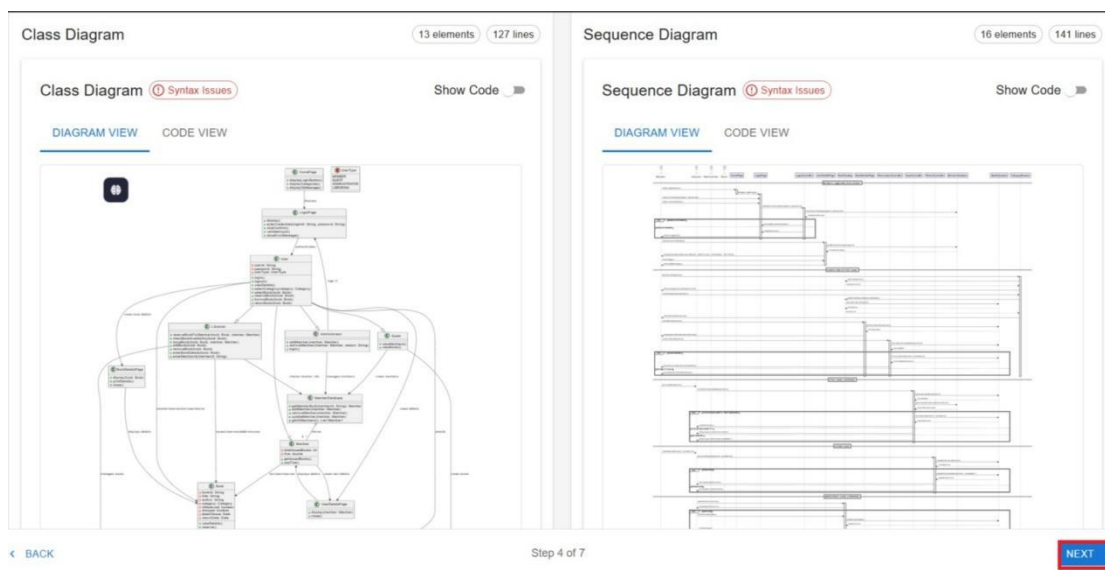
- Shows Screen 4 with Generate Both Diagrams button ready.



- Shows generation complete with success message.



- Shows both generated diagrams side-by-side (Class Diagram on left, Sequence Diagram on right)



STEP 5: Design Verifier

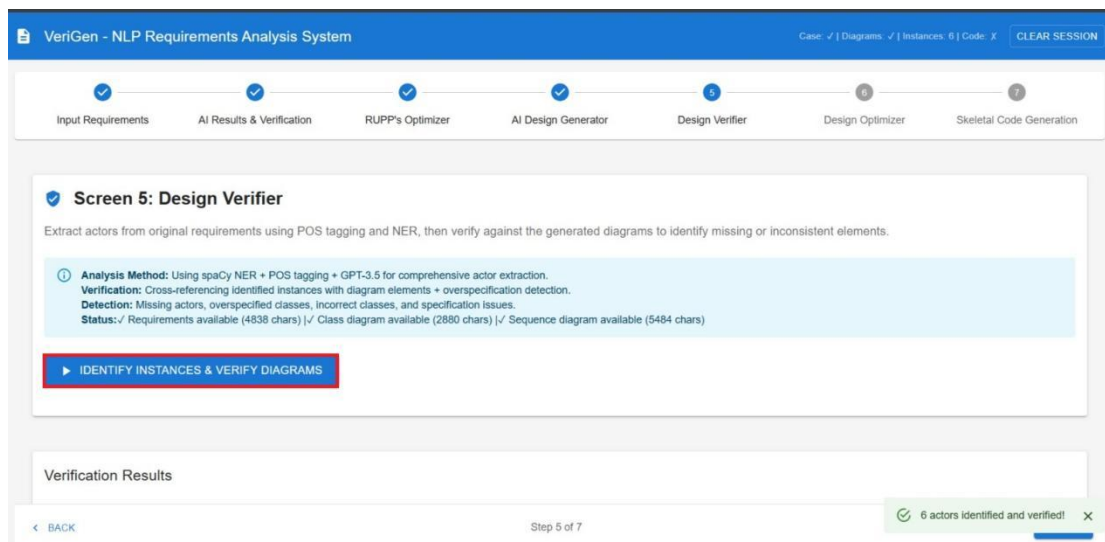
1. Click the [► IDENTIFY INSTANCES & VERIFY DIAGRAMS] button
2. Review the Verification Results showing:

- Verification Score
- Coverage Statistics
- Missing Instances
- Overspecified Classes (Librarian, Guest, Book, Administrator, Member)
- Incorrect Instances (HomePage, Category)
- Extra Instances (LoginPage, UserDetailsPage, MemberDatabase, BookDatabase, BookDetailsPage)
- Present Instances (User)

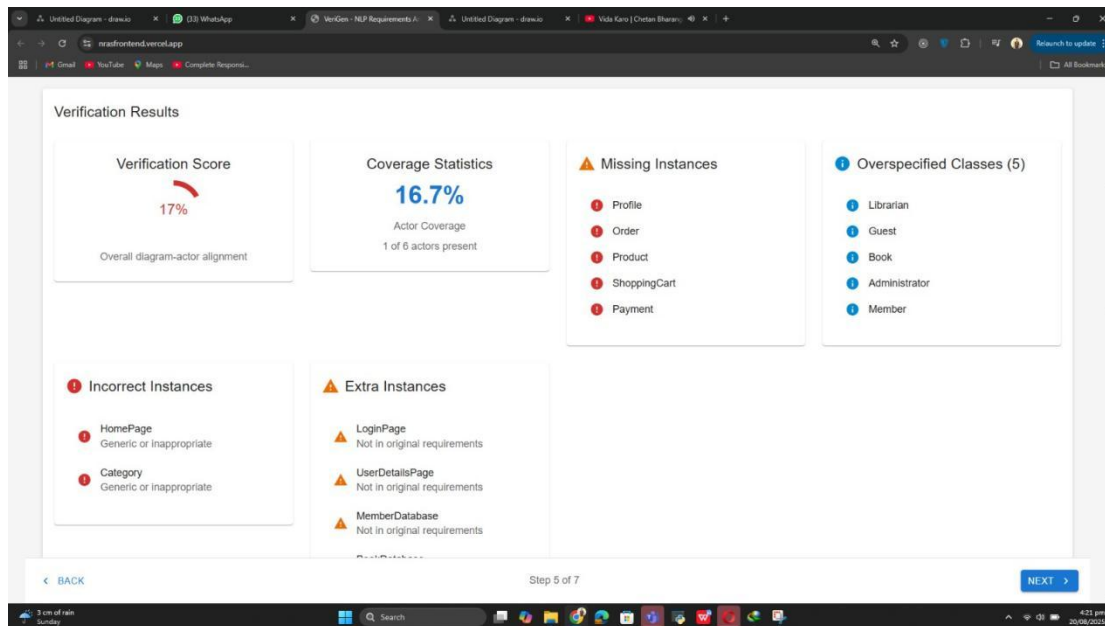
3. Click the [NEXT →] button

Result: System shows "✓ 6 actors identified and verified!" with detailed analysis

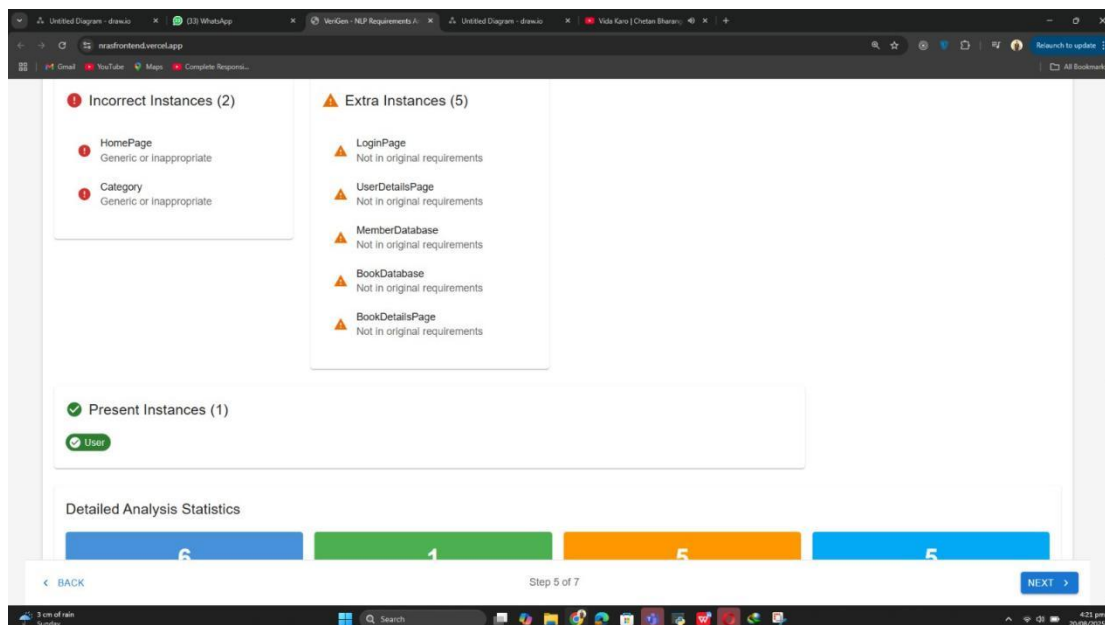
- Shows Design Verifier screen with Identify Instances button.



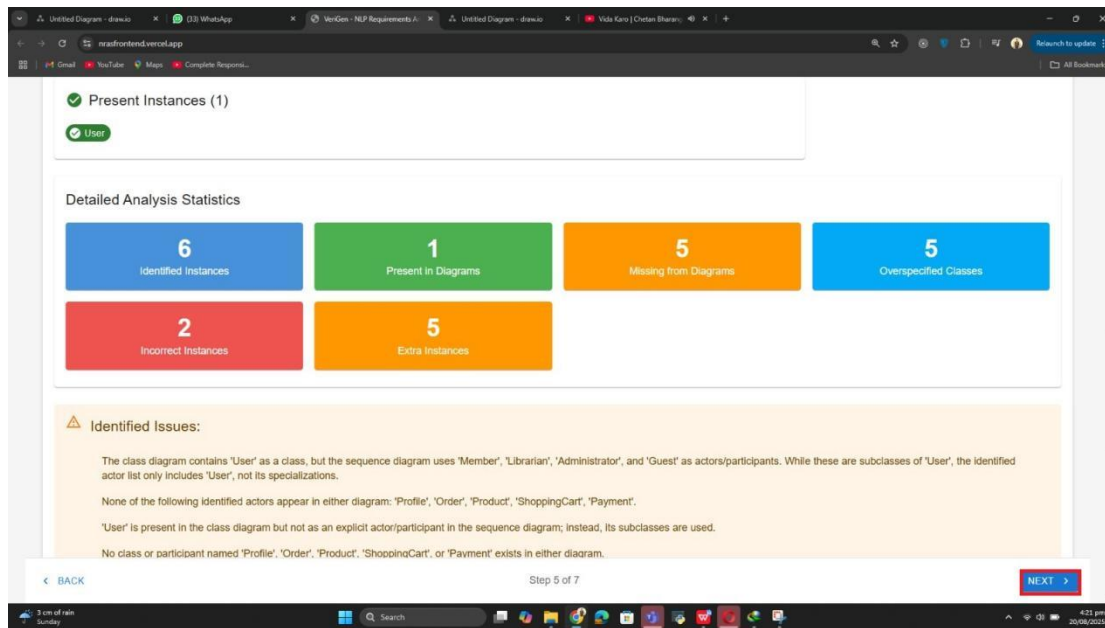
- Shows Verification Results dashboard with scores and statistics



- Shows detailed breakdown of Incorrect, Extra, and Present Instances.



- Shows Detailed Analysis Statistics and Identified Issues.

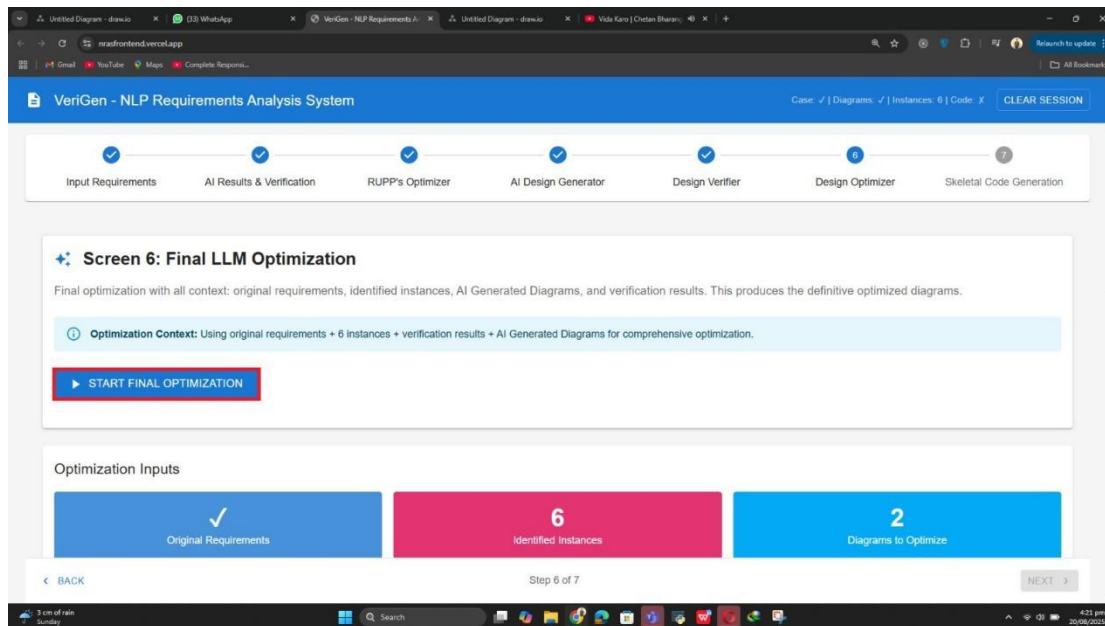


STEP 6: Final LLM Optimization

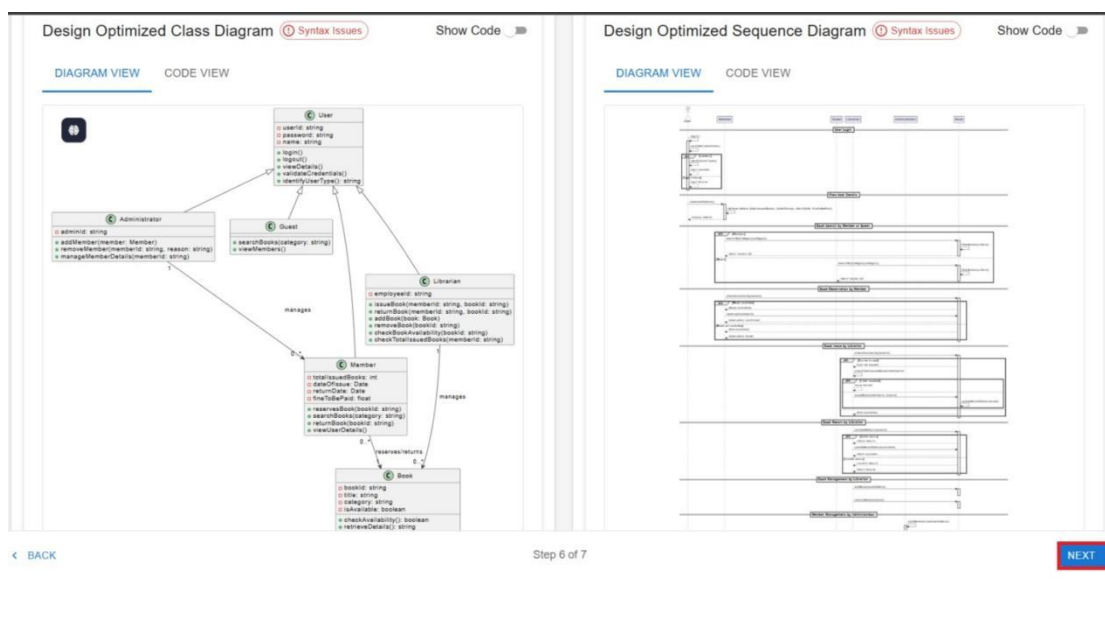
1. System displays "Screen 6: Final LLM Optimization"
2. Click the [▶ **START FINAL OPTIMIZATION**] button
3. Review the optimized diagrams:
 - Design Optimized Class Diagram
 - Design Optimized Sequence Diagram
4. Click the [NEXT →] button

Result: System generates final optimized diagrams with all corrections applied

- Shows Screen 6 with optimization inputs and Start Final Optimization button.



- Shows final optimized Class Diagram and Sequence Diagram side-by-side.



STEP 7: Skeletal Code Generation

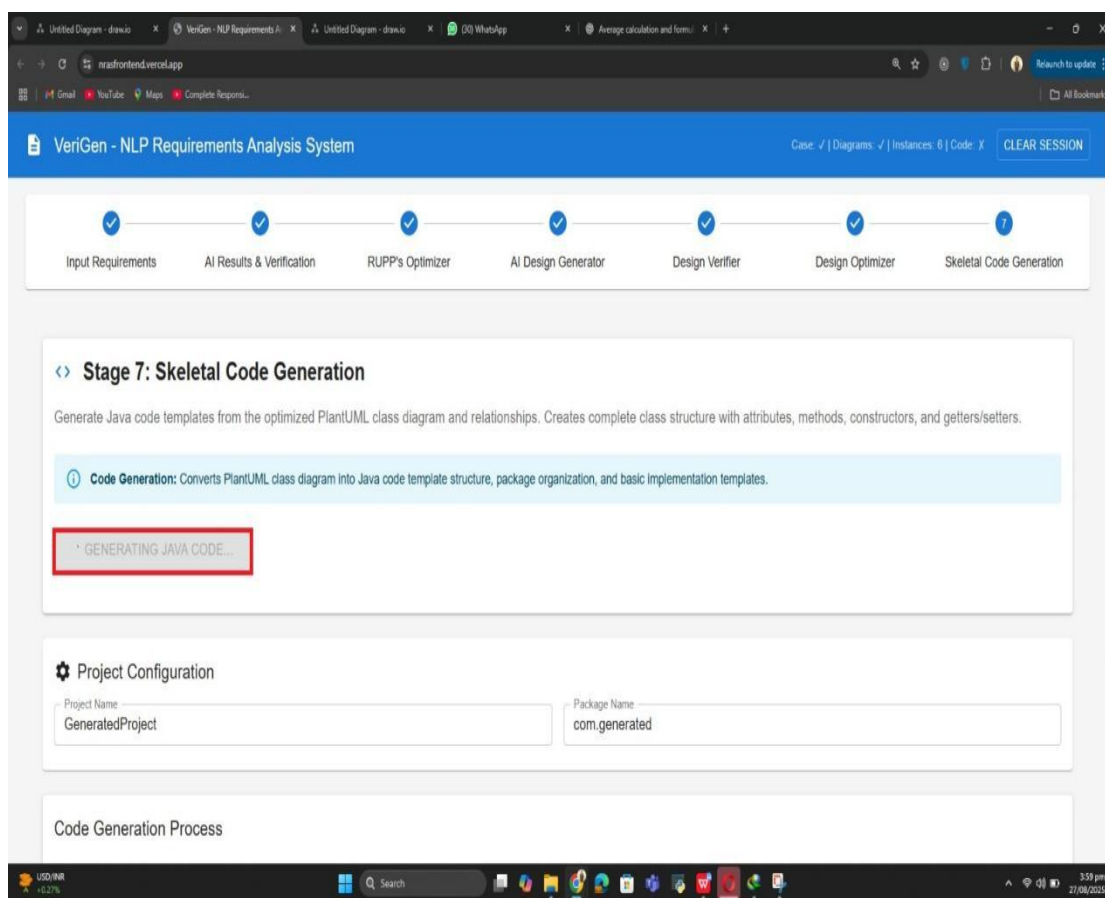
1. System displays "Stage 7: Skeletal Code Generation"
2. Configure Project Name and Package Name (optional)
3. System automatically starts [**GENERATING JAVA CODE...**]
4. Review generated Java files:
 - USER.JAVA
 - ADMINISTRATOR.JAVA

- GUEST.JAVA
- LIBRARIAN.JAVA
- MEMBER.JAVA
- BOOK.JAVA

5. Click [**DOWNLOAD PROJECT**] to download complete project as ZIP file

Result: System generates 6 Java class files with skeletal code (package, attributes, constructors, getters/setters)

- Shows Stage 7 with code generation process and project configuration.



- Shows Generated Java Files with code preview and download options.

<> Generated Java Files (6)

📄 DOWNLOAD PROJECT

USER.JAVA ADMINISTRATOR.JAVA GUEST.JAVA LIBRARIAN.JAVA MEMBER.JAVA BOOK.JAVA



```
1 package com.generated.model;
2
3 public class User
4 {
5
6     // Attributes
7     private String userId;
8     private String password;
9     private String name;
10
11     // Default constructor
12     public User() {
13         // TODO: Initialize default values
14     }
15
16     // Parameterized constructor
17     public User(String userId, String password, String name) {
18         this.userId = userId;
19         this.password = password;
20         this.name = name;
21     }
22
23     // Getters and Setters
```