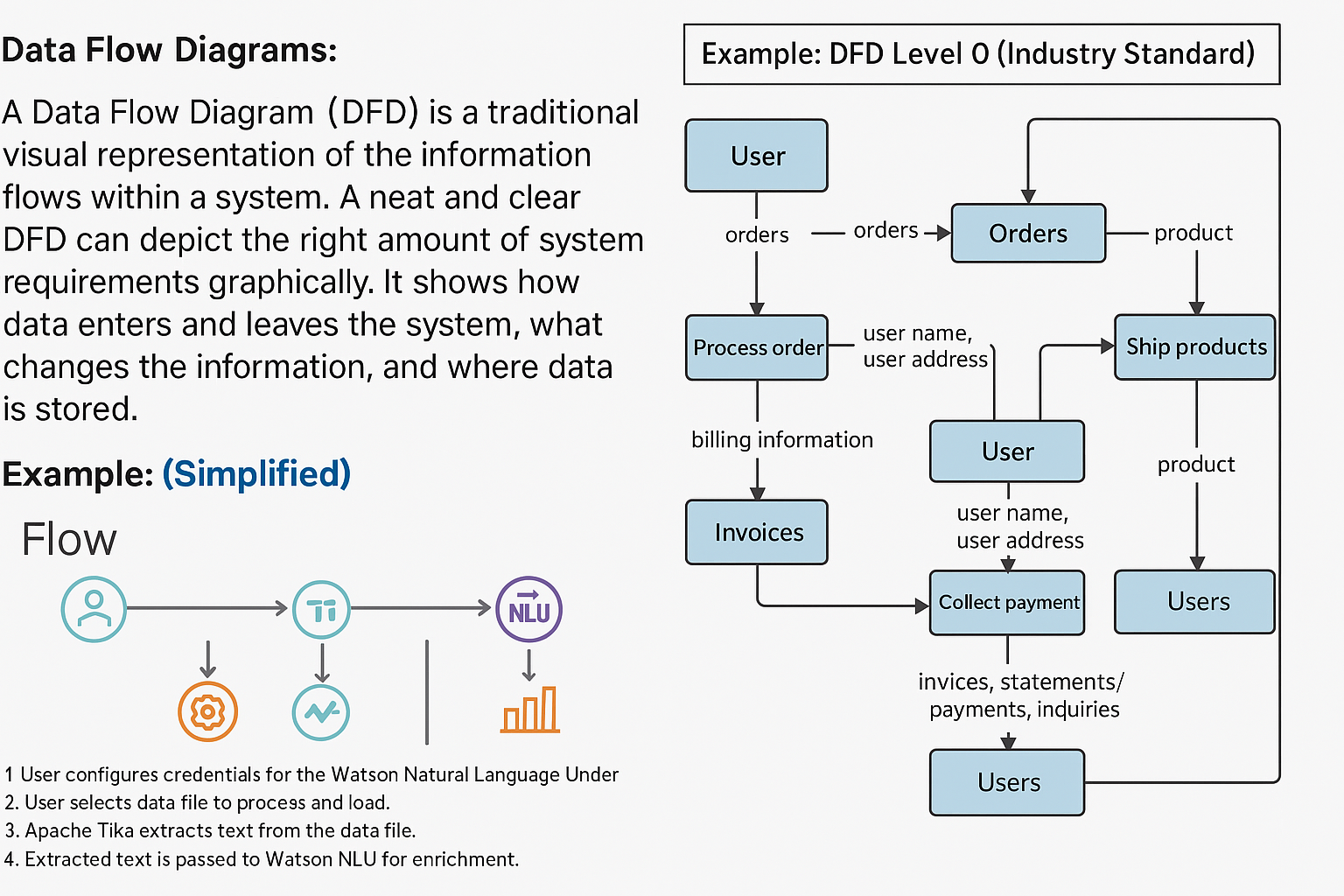
**Project Design Phase-II**

**Data Flow Diagram & User Stories**

|  |  |
| --- | --- |
| Date | 31 January 2025 |
| Team ID | LTVIP2025TMID32342 |
| Project Name | SmartSDLC – AI-Enhanced Software Development Lifecycle |
| Maximum Marks | 4 Marks |

**Data Flow Diagrams:**

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

****

**User Stories**

| **User Type** | **Functional Requirement (Epic)** | **User Story Number** | **User Story / Task** | **Acceptance Criteria** | **Priority** | **Release** |
| --- | --- | --- | --- | --- | --- | --- |
| Developer / Analyst | Requirement Upload & Classification | USN-1 | As a user, I can upload a PDF with unstructured requirements and have it classified into SDLC phases. | System extracts text using PyMuPDF, classifies each sentence using Watsonx Granite-20B into SDLC phases, and stores them for further processing. | High | Sprint-1 |
| Developer / Analyst | Requirement Upload & Classification | USN-2 | As a user, I can view structured user stories grouped by SDLC phase. | Frontend displays classified content in a readable, phase-grouped format with clear traceability. | High | Sprint-1 |
| Developer | AI Code Generator | USN-3 | As a developer, I can input a prompt or user story and receive production-ready code. | Watsonx generates syntactically correct, context-aware code; code is displayed with syntax highlighting and is ready for use or enhancement. | High | Sprint-2 |
| Developer | Bug Fixer | USN-4 | As a developer, I can submit buggy code and receive a corrected version. | System detects both syntactic and logical errors; returns optimized code; frontend shows original and fixed versions side-by-side. | Medium | Sprint-2 |
| QA Engineer | Test Case Generator | USN-5 | As a QA, I can input code or requirements and receive auto-generated test cases. | AI generates test cases using unittest or pytest; test cases cover expected functionality and edge cases; output is ready for automation. | High | Sprint-3 |
| Technical Writer / Dev | Code Summarizer | USN-6 | As a user, I can input source code and receive a human-readable summary. | Watsonx analyzes code logic and generates a concise summary explaining purpose, logic, and usage; summary is displayed in readable format. | Medium | Sprint-3 |