Software Design Document

# 1. Introduction

## 1.1 Purpose

This Software Design Document (SDD) outlines the architecture and design details of the ICD Creation Tool, a hybrid software system that combines Python/Django and C++ modules to automate the generation and validation of ICD (Interface Control Document) data for defense or aerospace applications.

## 1.2 Scope

The system automates the creation, processing, and validation of ICD tables. It supports both a GUI/backend (built in Python using Django) and low-level validation and processing routines implemented in C++.

## 1.3 Identification

Project Title: ICD Creation Tool  
Version: 1.0 (April 8 Release)  
Technologies: Django, C++, SQLite, PyInstaller

# 2. System Overview

A modular architecture with a Django backend and C++ processing engine. Key components include APIs, C++ modules, SQLite database, and file interfaces.

# 3. Architectural Design

The ICD Creation Tool uses a layered architecture: Django views handle input, validation, and data storage; C++ executables process the data; SQLite handles storage.

# 4. Data Design

Data is modeled in Django ORM, validated using custom logic, and stored in SQLite3. ICD entries consist of fields like parameter\_id, bit\_length, data\_type, etc.

# 5. Component Design

Python components: views.py, urls.py, models.py, validation.py, etc. C++ components: main.cpp, union.cpp. These are linked through launcher.py or icd\_creation\_tool.py.

# 6. Interface Design

Includes REST APIs, command-line interfaces, file input/output. APIs allow posting ICD data, triggering processing, and getting results.

# 7. Testing Strategy

Unit and integration testing for Django modules, and manual/output-based testing for C++ executables. test.py includes Django tests.

# 8. Appendices

Glossary, file list, development setup, and future improvements such as frontend UI, enhanced error handling, and more export formats.