# Finite Element Analysis (FEA) Application – User Requirements Document

Version: 1.0  
Date: 2025-03-07  
Author: Muhanned Samir Yacoub Ma’ayeh

# 1. Introduction

## 1.1 Purpose

This document outlines the functional, non-functional, and technical requirements for a Finite Element Analysis (FEA) application developed as a hobbyist project. The software is intended for educational purposes and personal experimentation, providing basic FEA functionality.

## 1.2 Scope

The FEA application will:  
- Provide basic finite element analysis capabilities (e.g., static structural analysis).  
- Include a Graphical User Interface (GUI) for user interaction.  
- Use HDF5 as a data storage layer for input and output data.  
- Implement a solver engine capable of handling simple 1D, 2D, and 3D elements.  
- Provide predefined lookup tables for materials and cross-sections.  
- Be non-commercial and open-source, with licensing restrictions preventing professional or commercial use.

# 2. User Requirements

## 2.1 Target Audience

This application is designed for:  
- Hobbyists & Enthusiasts interested in learning about FEA.  
- Students exploring basic FEA concepts.  
- Developers & Researchers experimenting with FEA solvers and algorithms.

## 2.2 Use Cases

The following use cases describe how users will interact with the system:  
- UC-01: Define and input mesh data.  
- UC-02: Define and assign material properties.  
- UC-03: Run finite element solver.  
- UC-04: Visualize analysis results.  
- UC-05: Save and load sessions.

# 3. Functional Requirements

## 3.1 User Interface (GUI)

The system shall provide a GUI for mesh generation, material assignment, solver execution, and result visualization.

## 3.2 Mesh Handling

The system shall support importing and generating finite element meshes.

## 3.3 Material & Cross-Section Database

The system shall store predefined materials and cross-sections in an HDF5-based database.

## 3.4 Solver Engine

The system shall support static structural analysis and solve the displacement vector using numerical methods.

# 4. Non-Functional Requirements

The system shall optimize memory usage and perform efficiently for medium-scale FEA problems.

# 5. System Architecture

The system consists of a GUI, a middleware layer, an FEA solver, and an HDF5 data storage module.

# 6. Data Storage & HDF5 Structure

All input and result data will be stored in an HDF5 file with structured datasets and attributes.

# 7. Constraints & Limitations

The software is not certified for engineering applications and is limited to small-to-medium FEA problems.

# 8. Licensing & Legal Considerations

The software will be licensed under Creative Commons BY-NC 4.0 and includes a disclaimer to limit liability.