

Especificación de Requerimientos: Software para la Resolución Gráfica de Modelos de Programación Lineal

Juan Carlos Barrera Guevara, Diego Alejandro Machado Tovar
Facultad de Ciencias Básicas e Ingeniería, Universidad de los Llanos
Villavicencio, Colombia
jc.bguevara@unillanos.edu.co
damachado@unillanos.edu.co

1. Introducción

1.1. Propósito

El presente documento tiene como propósito especificar los requerimientos funcionales y no funcionales del software desarrollado para la resolución de problemas de programación lineal mediante el método gráfico. Este sistema está orientado a servir como herramienta de apoyo en la enseñanza y en la práctica de la investigación de operaciones, facilitando tanto el ingreso de datos manuales como el procesamiento automatizado de restricciones a partir de imágenes.

1.2. Alcance

El software consiste en una aplicación de escritorio implementada en Python, con una interfaz gráfica interactiva construida sobre PySide6/Qt y capacidades de visualización mediante Matplotlib. Su alcance abarca la interpretación de restricciones lineales, la construcción de la región factible, la identificación de los puntos extremos y el cálculo de la solución óptima de la función objetivo. Adicionalmente, el sistema incluye integración con un módulo OCR que permite leer restricciones desde imágenes y un componente de inteligencia artificial para el manejo de errores. El producto final está dirigido principalmente a estudiantes y docentes de investigación de operaciones, así como a profesionales interesados en la resolución práctica de problemas lineales.

1.3. Definiciones, acrónimos y abreviaturas

- **OCR (Optical Character Recognition):** Reconocimiento óptico de caracteres.
- **Qt:** Framework multiplataforma para el desarrollo de interfaces gráficas.
- **Matplotlib:** Biblioteca de Python para visualización de datos y generación de gráficos.

2. Descripción General

2.1. Perspectiva del sistema

El sistema se concibe como una aplicación autónoma que combina un módulo de interfaz gráfica con un motor de cálculo matemático. El front-end está diseñado para brindar al usuario un entorno interactivo, mientras que el back-end gestiona los cálculos numéricos, el análisis de restricciones y la integración de entradas provenientes de OCR. La estructura modular del software permite su mantenimiento y futura ampliación sin comprometer la estabilidad de la aplicación.

2.2. Funcionalidades principales

El software ofrece la posibilidad de ingresar manualmente problemas de programación lineal, procesar restricciones a partir de imágenes mediante OCR, graficar la región factible en un plano cartesiano, calcular la función objetivo y presentar la solución óptima al usuario. La interfaz proporciona herramientas de visualización que permiten explorar interactivamente la gráfica generada y obtener información precisa sobre los puntos de interés.

2.3. Restricciones y supuestos

El sistema está diseñado para resolver problemas de programación lineal de dos variables, dado que el método gráfico no es aplicable a dimensiones superiores. Se asume que las restricciones son lineales y que los datos ingresados cumplen con el formato esperado. Para el funcionamiento del OCR se requiere que el usuario instale previamente el motor Tesseract-OCR, y para el módulo de inteligencia artificial se requiere acceso a la API de Groq con la configuración adecuada.

3. Requerimientos específicos

3.1. Requerimientos funcionales

El sistema debe cumplir con las siguientes funciones observables:

RF1. El sistema debe permitir el ingreso manual de restricciones lineales y de la función objetivo a través de la interfaz gráfica.

RF2. El sistema debe procesar las restricciones ingresadas y representar gráficamente la región factible en un plano cartesiano.

RF3. El sistema debe identificar los vértices de la región factible y calcular la solución óptima de la función objetivo.

RF4. El sistema debe permitir la carga de restricciones desde imágenes utilizando reconocimiento óptico de caracteres (OCR).

RF5. El sistema debe mostrar de manera interactiva la gráfica generada, permitiendo al usuario visualizar y analizar la solución.

3.2. Requerimientos no funcionales

Además de las funcionalidades, el sistema debe cumplir con las siguientes características de calidad y restricciones técnicas:

RNF1. El sistema debe estar desarrollado en Python 3 y ser compatible con entornos Windows y Linux.

RNF2. La interfaz gráfica debe estar implementada con PySide6/Qt, incorporando Matplotlib para la representación visual.

RNF3. El sistema debe garantizar la concurrencia entre los cálculos y la interfaz gráfica mediante hilos, evitando bloqueos en la ejecución.

RNF4. El sistema debe gestionar sus configuraciones mediante archivos externos, facilitando su adaptación y despliegue en diferentes entornos.