

# Informe Final del Reto de ASOCIO

Samuel Rico Gómez  
Gregorio Bermúdez Ocampo  
Valeria Cardona Velásquez  
Universidad EAFIT

30 de Junio de 2025

## 1. Introducción

El presente trabajo aborda el problema de asignación de puestos de trabajo en un esquema híbrido de presencialidad y teletrabajo. Este desafío surge como resultado de la transformación organizacional que se consolidó tras la pandemia del COVID-19, cuando el teletrabajo demostró ser no solo viable, sino también beneficioso para la calidad de vida de los colaboradores y para la optimización del uso de los espacios físicos disponibles.

El objetivo principal del reto es diseñar una herramienta de apoyo a la toma de decisiones que permita asignar de manera óptima los puestos de trabajo presenciales a los colaboradores, considerando múltiples criterios: días de asistencia, compatibilidades técnicas, preferencias individuales, cohesión de equipos de trabajo y restricciones físicas del espacio. En concreto, se deben determinar tres componentes clave:

1. El día de reunión presencial de cada grupo de trabajo.
2. Los días asignados para la presencialidad de cada colaborador.
3. El escritorio específico que se le asigna en dichos días.

El problema fue abordado mediante una estrategia basada en optimización y formulación matemática, lo cual permitió modelar rigurosamente las múltiples restricciones y objetivos involucrados en la asignación de puestos de trabajo. En particular, se desarrolló un modelo de programación entera, adecuado para resolver problemas de asignación con múltiples restricciones, preferencias y criterios de agrupamiento espacial y temporal. Esta aproximación posibilita encontrar soluciones eficientes y coherentes con la estructura organizativa de la institución, maximizando la satisfacción de condiciones deseables dentro de un entorno híbrido de trabajo. Este modelo prioriza la asignación óptima de los puestos disponibles, maximizando la cantidad de empleados con una ubicación válida y cumpliendo la mayor cantidad de condiciones deseables posible, como la cercanía entre miembros de un mismo grupo y la asignación de escritorios compatibles con sus requerimientos.

Para evaluar la aplicabilidad de la solución propuesta, se trabajó sobre 10 instancias distintas del problema, cada una con diferentes configuraciones de colaboradores, escritorios, zonas, equipos de trabajo y preferencias. La metodología adoptada se centra en una estrategia estructurada que permite escalar la solución a diferentes escenarios, maximizando el número de empleados correctamente asignados, cumpliendo la mayor cantidad posible de restricciones y garantizando un tiempo de cómputo razonable para su implementación práctica.

## 2. Estrategia de Solución

### 2.1. Enfoque General

Para abordar el problema de asignación de puestos de trabajo, se utilizó un enfoque de programación lineal entera con estructura lexicográfica. Esta elección responde a la necesidad de manejar múltiples objetivos jerárquicos y restricciones estructurales del problema, manteniendo una formulación rigurosa y con garantías de optimalidad dentro de un marco computacional razonable.

El modelo se dividió en dos etapas secuenciales:

#### 1. **Primera etapa:** Maximización de la satisfacción individual.

Se formuló un modelo de programación lineal entera cuyo objetivo principal fue **maximizar la cantidad de empleados que trabajan en sus días preferidos**. Para ello, se consideraron las siguientes restricciones:

- Cada empleado debe contar con un escritorio compatible y adecuado a sus necesidades.
- Un escritorio no puede ser asignado a más de un empleado por día.
- Los empleados solo pueden ocupar escritorios compatibles con su perfil.
- Cada equipo debe tener al menos un día de reunión presencial a la semana.
- Un empleado debe asistir a la oficina al menos dos días por semana.
- La relación entre los días de reunión de los equipos y la asistencia individual debe ser coherente.

#### 2. **Segunda etapa:** Maximización de la cohesión de los equipos.

Luego de resolver el problema de la etapa 1, nos predispusimos a determinar una nueva función objetivo: **maximizar la cohesión de los empleados de un mismo equipo en las mismas zonas de trabajo**. La idea principal es que, durante los días de reunión grupal, los integrantes de un equipo trabajen en escritorios ubicados dentro de la misma zona, promoviendo la colaboración y evitando que algún miembro quede completamente aislado.

Utilizando la solución obtenida en la primera etapa como base, se redefinió la función objetivo para maximizar esta cohesión espacial. Para garantizar la jerarquía entre los objetivos del modelo, se impuso como restricción que la solución en esta segunda etapa no redujera significativamente el valor óptimo alcanzado en la etapa anterior en términos de satisfacción de los días preferidos por los empleados.

Este enfoque permitió capturar y balancear dos dimensiones clave del problema: la **satisfacción individual** de los empleados y la **eficiencia organizacional** a través del trabajo colaborativo por zonas. Asimismo, la formulación permite escalar el modelo a diferentes tamaños de instancia, manteniendo un tiempo de cómputo razonable gracias al uso de solvers modernos como **Gurobi**.

Para lograr estos objetivos, se añadieron las siguientes restricciones al modelo:

- **Tracking de zonas:** Se introdujeron variables que permiten asociar a cada empleado con una zona en cada día, a partir del escritorio que le fue asignado. Esta restricción es fundamental para evaluar la cohesión de los equipos, ya que permite identificar en qué zona se encuentra cada colaborador.

- Definición de cohesión: Se añadieron variables binarias que indican si un empleado se encuentra cohesionado con su equipo, es decir, si está ubicado en la misma zona junto con al menos otro miembro del mismo grupo durante el día de la reunión. Esta restricción permite computar la nueva función objetivo de cohesión de forma lógica y cuantificable.
- Restricción de optimalidad: Se estableció una cota inferior sobre el valor obtenido en la primera etapa, exigiendo que el número de días preferidos asignados no disminuya significativamente. Esto preserva el enfoque lexicográfico adoptado: primero se prioriza la satisfacción individual y luego se mejora la cohesión, sin comprometer lo ya logrado.

Se toma como supuesto que si un empleado sólo tenía un día preferido, nosotros le asignamos el otro un día cualquiera.

## 2.2. Herramientas Utilizadas

- Lenguaje: Python 3.10.6
- Librerías: PuLP, Flask, Flask-CORS, Gurobi
- Solver: Gurobi.

## 3. Resultados Obtenidos

Para evaluar la efectividad de la herramienta desarrollada, se resolvieron más de 10 instancias del problema de asignación de puestos de trabajo en modalidad híbrida. Las primeras 10 instancias brindadas por el Challenge de ASOCIO y el resto las creamos nosotros para fines de validación de casos extremos, por lo tanto, no las añadimos en los resultados. La resolución se llevó a cabo mediante el modelo de optimización con enfoque lexicográfico, implementado en Python utilizando la librería PuLP y resuelto con el solver Gurobi. Los resultados obtenidos se resumen en la siguiente tabla:

### 3.1. Tabla Resumen

Instancia	Tiempo de ejecución (s)	Días preferidos asignados	% Equipos con reunión	% Equipos cohesionados
inst01	0.30	38/44 (86.4 %)	4/4 (100 %)	7/8 (87.5 %)
inst02	0.34	41/45 (90.9 %)	4/4 (100 %)	7/8 (87.5 %)
inst03	0.41	83/100 (83.0 %)	5/5 (100 %)	15/19 (75.0 %)
inst04	0.52	87/100 (87.0 %)	5/5 (100 %)	15/18 (77.8 %)
inst05	1.03	151/173 (86.8 %)	8/8 (100 %)	17/38 (45.0 %)
inst06	1.02	154/174 (88.5 %)	8/8 (100 %)	15/35 (40.5 %)
inst07	1.00	180/200 (90.0 %)	12/12 (100 %)	14/64 (22.2 %)
inst08	1.98	177/200 (88.5 %)	12/12 (100 %)	15/66 (23.4 %)
inst09	2.81	199/225 (88.4 %)	18/18 (100 %)	13/86 (11.1 %)
inst10	3.04	196/225 (87.1 %)	18/18 (100 %)	7/94 (7.5 %)

- **Tiempo de ejecución (s):** Tiempo total, en segundos, que el modelo tardó en encontrar una solución factible, optimizada para cada instancia y renderizar la solución.
- **Días preferidos asignados:** Porcentaje de las asignaciones de días presenciales que coincidieron con las preferencias explícitas de cada colaborador. Esta métrica mide el nivel de satisfacción individual de los empleados respecto a su jornada laboral.

- **% Equipos con reunión:** Porcentaje de los equipos que lograron reunirse presencialmente al menos un día a la semana, con todos sus miembros presentes. Esta métrica garantiza el cumplimiento de una restricción crítica del problema.
- **% Equipos cohesionados:** Porcentaje de los equipos cuya mayoría de miembros fueron ubicados en la misma zona física durante su reunión presencial. Esta métrica refleja el grado de cohesión espacial logrado por el modelo, fomentando la colaboración entre compañeros del mismo grupo.

## 4. Instrucciones de Ejecución

### 4.1. Requisitos

- Python 3.7 o superior
- Entorno virtual configurado.
- Instalar dependencias: `pip install -r requirements.txt`

### 4.2. Estructura del proyecto

A continuación se presenta la estructura de carpetas y archivos del proyecto implementado para desplegar la solución mediante una aplicación web:

```
FlaskDeploy/
  app.py                # Aplicación principal Flask
  requirements.txt       # Dependencias de Python
  README.md             # Documentación
  .gitignore            # Archivos a ignorar por Git
  models/               # Modelos de optimización
    __init__.py
    lexicographic.py
    lexicographicv1.py  # Implementación ppal lexicográfico.
    modelo_adaptado_web.py
  templates/            # Plantillas HTML
    index.html
  venv/                 # Entorno virtual (no incluido en Git)
```

### 4.3. Ejecución Local

La herramienta desarrollada permite ejecutar el modelo de asignación con enfoque lexicográfico en un entorno web mediante el uso de Flask. Esta implementación resuelve de forma automatizada distintas instancias del problema de asignación de escritorios en modalidad híbrida, priorizando en primer lugar la satisfacción individual de los empleados y, en segunda instancia, la cohesión de los equipos de trabajo.

El aplicativo puede ser desplegado localmente y, una vez iniciado, es posible acceder a la interfaz web a través del puerto correspondiente para cargar el archivo de la instancia deseada. La herramienta permite al usuario visualizar los resultados de manera interactiva. Adicionalmente, si se desea una integración con otras aplicaciones, también es posible consumir el endpoint de resolución

como se describe en la documentación del repositorio.

Para una comprensión adecuada de la ejecución local y el uso completo de la herramienta, se recomienda consultar el archivo `README.md` disponible en el repositorio del proyecto: **Challenge ASOCIO**.

#### 4.4. Formato de Entrada/Salida

Breve explicación de las estructuras esperadas por el programa:

- Entrada: Recibe un archivo `JSON` con la estructura de datos del problema planteado, y devuelve la solución del modelo. También se puede llamar al endpoint como se explica en el `README.md` del repositorio de GitHub.
- Salida: Retorna un objeto `JSON` con:
  - Estado de la solución.
  - Tiempo de resolución.
  - Tasa de satisfacción.
  - Tasa de cohesión de equipos.
  - Horarios asignados a cada empleado.
  - Asignación de escritorios.
  - Días de reunión por equipo.
  - Análisis de proximidad por zonas.