



Tecnológico de Monterrey

Campus Monterrey

Reflexión

Ana Elisa Celaya Montalvo | A01287120

Ethiel Favila Alvarado | A00844789

Catherine González Díaz | A00845539

Profesor David Alonso Cantú Delgado
Pensamiento computacional orientado a objetos
Grupo 313

03 de diciembre del 2025

El proyecto AutoMatch (versión Toyota) se creó utilizando los conocimientos adquiridos en clase así como nuevos por medio de investigación y uso de la Inteligencia Artificial. Se analizó una base de datos csv, por lo que se utilizaron metodologías similares (pero más compuestas) a las aplicadas en clase. Además de estas funciones, quisimos hacer algunos extras. El equipo se encargó totalmente de la creación de imágenes ASCII. Después, investigamos la nueva librería sstream, para poder crear archivos de salida. Este concepto fue nuevo para el equipo y aunque se programó a mano, surgieron errores. Aquí recurrimos a la inteligencia artificial para poder encontrar los errores y corregirlos. Además, tuvimos un problema con el formato de las imágenes ASCII en la computadora de una compañera, por lo que igual solicitamos el guión de la IA. El proyecto fue creado mayormente por nuestro conocimiento, pero fue útil tener a la Inteligencia Artificial como consultor.

La base de datos cuenta con 6,739 entradas, y todas las opciones imprimen las opciones filtradas. Desde simple vista se puede ver que las opciones mostradas son menos de seis mil, por lo que podemos ver que los distintos filtros funcionan. Nuestro filtro más extenso “Ideal Car Quiz”, produce un archivo de texto txt. Comparando las opciones elegidas, el archivo resultante y el csv original, podemos ver que nuestros resultados funcionaron.

Desde principio a fin AutoMatch fue un proceso de aprendizaje. Desde la selección de la base de datos, se tuvo que tener una lluvia de ideas sobre la relevancia de los datos y que podríamos hacer con ellos. Por otro lado, la creación del archivo colaborativo usando GitHub fue un reto nuevo para nosotras. Habíamos utilizado GitHub individualmente antes, pero al hacerlo de manera colaborativa creamos un conflicto de fusión. Se nos borró un main y agregamos dos “Readme” por accidente. Conforme avanzamos en este proyecto aprendimos a usar GitHub y a enfocarnos cada quien en un archivo. Conforme se volvía más complejo el proyecto aprendimos nuevos conceptos como los archivos de salida. Finalmente, cuando creímos ya tener todo nuestro código, le pedimos a amigos ser testers. Dadas sus recomendaciones agregamos nuevas funciones y modificamos algunas. Fue una buena lección tener alguien que pruebe el código antes de entregarlo.

El programa tiene el objetivo de ser interactivo en todo momento, por lo que las decisiones de los usuarios es un riesgo. Mucho del código depende del input de dichos usuarios usando cin <<, ya sean strings o números enteros. Para que el código no colapsara tuvimos que adaptar las entradas permitidas a cosas como “N/A”, “NA”, “na”, etc. Además, para que el código no tronara, tuvimos que implementar varios try-catch que permitieran el flujo del código. Otro riesgo, si esta fuera una aplicación realmente usada por Toyota, tendríamos el problema de la actualización de datos. Por el momento la base de datos es fija, por lo que incluso si

se vende un carro permanece en el inventario. Para el objetivo de la situación problema el código funciona, pero sería una oportunidad interesante trabajar con una base de datos que se actualizará acordemente.