

Informe Desafío II

Por:

Isabela Aguirre Ceballos

Juan Felipe Higueta Pérez

Informática II

Universidad de Antioquia

2025

Análisis del problema

En este desafío se plantea el desarrollo de una aplicación llamada **UdeAStay**, orientada a facilitar el alquiler temporal de alojamientos hogareños entre usuarios, de manera similar al funcionamiento de plataformas existentes como Airbnb. El sistema debe permitir que dos tipos de usuarios interactúen: los **huéspedes**, quienes buscan un lugar para alojarse, y los **anfitriones**, quienes ofrecen propiedades en arriendo.

Cada tipo de usuario accede a funcionalidades específicas mediante un menú adaptado a su perfil. Los huéspedes pueden buscar alojamientos según criterios como la fecha de entrada, el municipio de interés y la cantidad de noches que desean hospedarse. Opcionalmente, pueden aplicar filtros como precio máximo por noche o puntuación mínima del anfitrión, y posteriormente realizar una reserva si encuentran una opción que se ajuste a sus preferencias.

Por su parte, los anfitriones pueden visualizar todas las reservaciones activas de sus alojamientos y administrar el histórico de reservaciones a través de una fecha de corte, que les permite mover las reservas pasadas a un archivo histórico para mantener el sistema organizado y eficiente.

El sistema debe permitir guardar la información más importante como los datos de usuarios, alojamientos y reservaciones de forma permanente, usando archivos externos simples. Estos archivos deben tener un formato claro y fácil de leer y escribir, para que el programa pueda recuperar los datos al iniciar y actualizar los cambios cuando sea necesario. Aunque no se busca construir una plataforma profesional, esta solución básica simula las funciones esenciales de un sistema de reservas, con el fin de aplicar los conceptos de programación orientada a objetos en un contexto cercano a la vida real.

Consideraciones para la alternativa de solución propuesta

La solución propuesta para este desafío es desarrollar una aplicación en C ++ usando programación orientada a objetos (POO).

Se crearán varias clases que representen las partes principales del sistema: **huéspedes**, **anfitriones**, **alojamientos**, **reservaciones**, y también clases de apoyo como **Fecha** y **Pago**, para organizar mejor la información y facilitar las validaciones. Cada clase tendrá sus propios atributos y funciones. Todas estarán diseñadas usando encapsulación, de modo que los datos se mantengan protegidos y solo se pueda acceder a ellos a través de funciones públicas.

- La clase **Huesped** tendrá un documento y una lista de códigos de reservaciones hechas.
- La clase **Anfitrión** almacenará el documento y una lista de códigos de alojamientos que administra.
- La clase **Alojamiento** tendrá su propio código, tipo (casa o apartamento), ubicación, precio por noche, amenidades y un registro de fechas reservadas.
- La clase **Reservacion** incluirá el código de reserva, fecha de inicio, duración, código del alojamiento, documento del huésped, método de pago, monto y anotaciones de un máximo de 1000 caracteres.
- La clase **Fecha** permitirá representar una fecha con día, mes y año, además de incluir funciones para comparar fechas, sumar días y validar rangos.
- La clase **Pago** incluirá el método de pago (PSE o tarjeta de crédito), la fecha del pago (usando también la clase Fecha) y el monto, separando así claramente los datos financieros de la reservación.

Finalmente, se incluirá una clase **UdeAStay** encargada de controlar el menú del sistema, realizar las validaciones necesarias, coordinar las funcionalidades según el rol del usuario que ingrese a la aplicación (Huésped o Anfitrión), y manejar la lectura y escritura de archivos para guardar los datos de manera permanente.