**INFORME**

**Desafío II:** **Mercado de estadías hogareñas UdeAStay**

**1. Análisis del Problema**

El desafío plantea la creación de un sistema para administrar un mercado de estadías hogareñas, similar a Airbnb, con conceptos claves como:

**Alojamiento:** Cada alojamiento tiene características como nombre, código, anfitrión, ubicación, tipo, precio. También tiene un registro de las fechas en que está reservado, los históricos de cancelaciones y reservas.

**Reservación:** Una reservación incluye información como fecha de entrada, duración, código, alojamiento reservado, huésped que la hizo, método de pago, fecha de pago y monto. Los huéspedes pueden agregar anotaciones a sus reservaciones.

**Anfitrión:** Un anfitrión tiene un número de documento, antigüedad en la plataforma, puntuación y una lista de los alojamientos que administra.

**Huésped:** Un huésped también tiene un número de documento, antigüedad, puntuación e información de sus reservas.

**Otros:** Manejo de memoria, como consumo de recursos y numero de iteraciones.

El desarrollo de esta aplicación es un reto bastante complejo ya que tiene muchas cosas a tener en cuenta, la cantidad de funciones que tiene, el nivel de complejidad que se maneja, y fuera de eso el tiempo para desarrollarlo, ya que se debe implementar en 15 días, por lo menos creo que me tome entre 70 a 80 horas en desarrollar la aplicación, ese es uno de los problemas más relevantes, también el tener que reforzar los conceptos básicos C++ y llevarlos a un nivel más complejo.

De acuerdo a mi análisis del programa y de los requerimientos, debo crear objetos como Alojamiento, reserva, anfitrión, huésped y creería que también un objeto que guarde y actualice los archivos. Los más fácil es crear archivos de texto de manera que se guarden los datos en forma de tabla para escribir de manera fácil y se pueda acceder a la información. Desarrollare el programa de manera modular de manera organizada para encontrar problemas y no afectar el resto de código.

**Diagrama UML**

**CLASEALOJAMIENTO**

- nombre: string

- codigoIdenti: string

- precioPorNoche: float

+ ~Alojamiento ()

+ getNombre(): string

+ setNombre(string): void

**CLASEANFITRION**

- numeroDocumento: string

- puntuacion: float

+ Anfitrion(string, float)

+ ~Anfitrion()

+ getNumeroDocumento(): string

+ setNumeroDocumento(string): void

+ getPuntuacion(): float

+ setPuntuacion(float): void

1— 1

0..\*— 1

**CLASEHUESPED**

- numeroDocumento: string

- puntuacion: float

+ Huesped(string, float)

+ ~Huesped()

+ getNumeroDocumento(): string

+ setNumeroDocumento(string): void

+ getPuntuacion(): float

**CLASERESERVACION**

- codigoReserva: string

- fechaEntrada: string

- alojamiento: Alojamiento\*

- huesped: Huesped\*

+ Reservacion(string, Fecha, Alojamiento\*, Huesped\*)

+ ~Reservacion()

+ getCodigoReserva(): string

1..\* — 1