

Informe Final Desafío II

Marx Eusdav Lopez Montero, Daniel Antonio Londoño Godoy Facultad

De Ingeniería, Universidad De Antioquia

Informática II: 2598521

28 de mayo de 2025

Introducción.

El presente informe documenta el desarrollo de un sistema de gestión de alojamientos, reservas, anfitriones y huéspedes llamado *UdeAStay*, basado en el paradigma de la Programación Orientada a Objetos. El objetivo principal del proyecto radica en una solución funcional y modular que permita reservar alojamientos turísticos, permitiendo a su vez consultar o eliminar reservaciones previamente existentes.

El sistema se construyó sin usar STL, organizando el código en múltiples archivos que separan la lógica implementada, a su vez, clases como: Alojamiento, Reserva, Huesped, Anfitrión y Fecha, cada uno con sus propios atributos y métodos, lo que permite una representación concisa del problema.

Los datos se almacenan en archivos de texto, y su lectura y escritura se realiza mediante flujos de entrada y salida. Además, se optó por el uso de estructuras dinámicas, organizando la información en arreglos de punteros a entidades como anfitriones, alojamientos y huéspedes y una matriz dinámica de punteros para el caso específico de las reservas, lo que facilita la gestión tabular y la manipulación eficiente de registros.

Metodología.

Análisis Previo.

Se delimitaron cinco clases principales: Anfitrión, Huésped, Alojamiento, Reserva y Fecha. Aunque tanto los huéspedes como los anfitriones comparten atributos como la identificación y el tiempo de permanencia en la plataforma, sus funciones son esencialmente diferentes, y es por esto que se decidió hacerlos en clases separadas.

El anfitrión es responsable de la administración de los alojamientos mientras que el huésped se enfoca en reservar y eliminar sus reservas. La clase Alojamiento concentra toda la información relevante de una propiedad, incluyendo su ubicación, precio, el conjunto de amenidades e información de las reservas asociadas a este, lo cual permite evitar duplicación de datos en otras clases.

Por otro lado, la clase Reserva se encarga de almacenar la información de las reservas, tales como las fechas de inicio, fin y pago, la duración de la reserva e información asociada.

Finalmente, la clase Fecha actúa como soporte para el manejo de operaciones temporales, facilitando el cálculo de periodos de reservas y disponibilidad sin comprometer la lógica interna de las clases Reserva o Alojamiento.

De acuerdo con el análisis anterior, se diseñó el diagrama de clases de la siguiente

manera  UML.pdf

[illegible]

Implementación

Se recalcó el trabajo modular, por consiguiente, se describe brevemente el propósito de cada archivo principal:

- **main.cpp:** Contiene la función principal del programa y coordina la interacción entre las diferentes clases. Gestiona el menú de opciones para anfitriones y huéspedes.
- **huesped.h / huesped.cpp:** Representa al usuario que realiza reservas en los alojamientos. Contiene información personal básica tal como su número de documento, su antigüedad en la plataforma y su puntuación. El huésped puede gestionar sus reservas, permitiendo la interacción con el sistema para crear o eliminar sus estadías.
- **anfitrión.h / anfitrión.cpp:** Gestiona alojamientos y consulta reservas asociadas a los alojamientos que administra; esta clase contiene información personal básica tal como su número de documento, su antigüedad en la plataforma y su puntuación.
- **alojamiento.h / alojamiento.cpp:** Representa propiedades con su dirección, amenidades y precios, además de manejar la relación con las reservas.
- **reserva.h / reserva.cpp:** Contiene la lógica de la clase Reserva, que permite a los huéspedes reservar alojamientos disponibles, almacenando fechas de inicio, fin, pago, monto pagado, método de pago, alojamiento al que pertenece y su respectivo huésped.
- **fecha.h / fecha.cpp:** Implementa la clase Fecha la cual es utilizada como apoyo para comparar fechas y calcular la fecha de fin de una reserva.
- **miscelaneos.h / miscelaneos.cpp:** Incluye funciones auxiliares para manejo de archivos, tales como cargarlos al sistema o guardarlos, validar la información o

presentar datos en pantalla.

Los archivos del sistema están organizados en formato .txt con campos separados por comas y guiones. Esta elección se debe a su simplicidad y facilidad para ser procesados, sin requerir librerías adicionales o estructuras muy complejas. Las comas funcionan como separadores principales entre campos, lo que permite una lectura y escritura ordenada de cada registro. Por otro lado, los guiones se utilizan para delimitar subcampos o listas dentro de un mismo campo.

Además, el manejo de datos del proyecto se dividió en cinco archivos principales: alojamientos, huéspedes, anfitriones, reservas y un archivo histórico. Hacerlo de forma independiente reduce la complejidad de las operaciones de lectura y escritura, mejora la claridad del almacenamiento, y permite que cada módulo del programa acceda únicamente a la información necesaria, optimizando el uso de memoria y facilitando la escalabilidad futura del sistema.

Se tomó la decisión de almacenar en arreglos unidimensionales para los alojamientos, huéspedes y anfitriones dado que no van a cambiar en tiempo de ejecución y facilita el acceso a la información. Las reservas se almacenan en una matriz ya que optimiza el consumo de memoria a la hora de tener que redimensionarlo, pues la cantidad de estas si cambian en tiempo de ejecución, además de que permite una estructura organizada y facilita el acceso.

Flujo Del Programa.

A continuación se detalla la lógica o flujo del programa.

Al iniciar, el sistema se carga la información desde los archivos de texto y se almacena en la estructura de datos descrita previamente en este informe

Ingreso Y Autenticación

El programa presenta un menú inicial que pide la credencial del usuario para distinguir entre anfitrión o huésped.

Flujo Para Huésped

El huésped puede consultar los alojamientos disponibles para realizar reservas y eliminar sus reservas disponibles. Para ello, el sistema

- Solicita los parámetros de búsqueda
- Recorre la matriz de alojamientos y verifica la disponibilidad usando la clase Fecha y el calendario de reservas de cada alojamiento.
- Permite seleccionar un alojamiento disponible y registra una nueva reserva, actualizando la matriz de reservas y las fechas ocupadas del alojamiento.
- Calcula el costo total basado en el precio por noche y duración, registrando el pago.

Flujo Para Anfitrión

El anfitrión administra sus alojamientos y puede revisar las reservas existentes, para esto el sistema

- Consulta su lista de alojamientos a través de su arreglo dinámico
- Visualiza detalles como ubicación, precio y fechas reservadas.
- Consulta reservas en un rango de fechas para evaluar ocupación y gestionar disponibilidad.

Operaciones Comunes

La clase Fecha se utiliza para comparar, validar y calcular intervalos de tiempo entre reservas, facilitando la verificación de solapamientos y el control de disponibilidad. La clase Reserva gestiona la información de pago y duración de la reserva, asegurando que los registros

se mantengan coherentes.

Manejo De Actualizaciones

Después de realizar todas las operaciones pertinentes los arreglos de punteros y la matriz de reservas se vuelcan a los archivos de texto mediante las funciones de guardado. Esto garantiza la persistencia y permite recuperar la información en futuras ejecuciones.