

## Descripción del proyecto final

Una agencia de viajes desea contar con un software para administrar la información sobre destinos, paquetes turísticos, clientes, reservas y guías turísticos. En cuanto a los clientes, se desea guardar su identificación, nombre completo, correo electrónico, número de teléfono y dirección de residencia. Para los destinos, la información clave abarca el nombre, ciudad, descripción, imágenes representativas y clima.

En el contexto de paquetes turísticos, la agencia busca conformar conjuntos de destinos. Cada paquete puede incorporar diversos destinos, y a su vez, un destino podría formar parte de varios paquetes. De un paquete turístico se requiere su nombre, duración, servicios adicionales y precio. Tenga en cuenta que cada paquete está disponible con un cupo máximo de personas y solo está disponible en fechas específicas.

Adicionalmente, la agencia posee registros sobre guías turísticos, incluyendo su identificación, nombre completo, lenguajes en los que se comunica y experiencia acumulada en tiempo.

Por último, como las reservas constituyen un componente esencial de los datos, se debe registrar su información. Cada reserva deberá tener su fecha de solicitud, fecha planificada de viaje, el cliente involucrado, la cantidad de personas que viajan y el paquete turístico seleccionado. Además, es opcional designar un guía turístico. Se debe indicar también el estado de la reserva, que puede ser pendiente, confirmada o cancelada.

A continuación se hace un listado de los requerimientos, cuando se mencione gestionar se refiere a crear, actualizar, eliminar y listar.

### Administrador:

- Gestionar destinos.
- Gestionar paquetes turísticos.
- Gestionar guías turísticos.
- Mostrar estadísticas sobre los datos guardados. Para esto debe hacer uso de gráficas estadísticas donde se evidencie los destinos más reservados, los que más se buscan, los guías mejor puntuados, los paquetes que más se suelen reservar \*\*.

### Cliente:

- Registrarse y loguearse.
- Modificar su perfil.
- Consultar paquetes de la agencia. Hacer búsquedas dado el nombre del destino, ciudad, paquete, precio, clima, fechas, etc. Se debe implementar filtros de búsqueda.
- Crear reservas. Para esto se debe validar el cupo máximo de personas y las fechas.
- Cancelar reservas.
- Listar sus propias reservas (pasadas y futuras).
- Ver el detalle de cada paquete (se debe mostrar sus destinos y la información de cada uno, incluyendo las imágenes).
- Calificar los destinos visitados por medio de estrellas y comentarios.

- Calificar el servicio del guía una vez se haya cumplido el viaje.
- El software le debe recomendar destinos usando los datos de búsqueda del cliente.

### Requerimientos adicionales:

- Diseñar diagrama de clases para representar cada entidad en el sistema: Cliente, Destino, Paquete Turístico, Guía Turístico, Reserva, etc
- El proyecto se debe implementar usando Java en su versión 1.
- Hacer la implementación de la GUI. La navegación por medio de la GUI debe ser lo suficientemente clara para no requerir un manual de usuario.
- Investigar qué es Lombok e implementarlo en el proyecto.
- Debe hacer uso del patrón Singleton para la gestión de la información entre todas las ventanas del proyecto.
- Debe haber una pantalla de inicio con la información de paquetes, destinos, el buscador, y los menús de login y registro.
- Implementa un sistema de control de acceso para garantizar que solo los administradores puedan acceder a ciertas funciones administrativas.
- Todos los recorridos a las listas y consultas a los datos se deben hacer usando recursividad.
- Implementar un chat con un asesor para preguntas rápidas. Debe hacer uso de sockets. Si pasa un minuto y nadie responde al chat, se debe cerrar instantáneamente \*\*.
- Cuando un cliente hace una reserva se le debe enviar un correo electrónico con los detalles de la misma.
- Debe haber una funcionalidad que permita deshacer o rehacer una acción (crear, borrar, editar, etc).
- Para que un cliente pueda calificar o comentar un destino (o guía), su reserva ya tuvo que haber pasado, por lo tanto se debe validar las fechas.
- Los administradores ya están registrados, por lo tanto, no se les debe implementar un formulario de registro.
- Hacer uso de excepciones propias para los posibles errores o advertencias de validaciones que pueda arrojar el software.
- Hacer uso de persistencia en archivos para guardar y leer toda la información asociada a clientes, guías, reservas, paquetes, lugares, etc.
- Usar sockets para crear un servidor donde se guarde la información del software (persistencia en archivos serializados o de texto plano), es decir, la persistencia debe estar gestionada por un servidor usando sockets.
- Implementar hilos para admitir múltiples conexiones al servidor central de persistencia.
- El proyecto se debe subir a un repositorio en GitHub. Solo se recibirá el código fuente de la entrega del proyecto por este medio.

**IMPORTANTE:** Con su grupo de trabajo debe pensar en dos funcionalidades adicionales a los que están acá detallados para que los incluya en la implementación de su proyecto. Estas funcionalidades no deben ser triviales.

\*\* Este punto es opcional y tendrá una bonificación extra de realizarse.