Análisis aislado de las pantallas

1Alta persona

La pantalla se corresponde con un formulario vacío. Por ello, NO es necesario cargar ningún tipo de dato. (Carga ***NO NECESARIA***)

* 1. Añadir como integrante

(Integrantes [] -> lazy): En esta pantalla cargamos una única persona y todos los grupos existentes en la base de datos. En la pantalla solo mostramos los atributos básicos del estado de persona. También debemos cargar **todos** los grupos existentes en la base de datos para poder seleccionar en cual se insertará como nuevo integrante. Dado que vamos a hacer una modificación en la colección de integrantes esta colección debería estar en memoria. Pero habría que cargar la colección de todos los grupos existentes ya que nos sabemos en cual se va a insertar. Sin embargo, se encuentra del lado propietario por lo que se realiza una carga EAGER por defecto. (Carga ***LAZY***)

(reservas [] -> lazy): En esta pantalla **mostramos** únicamente atributos básicos del estado de la persona seleccionada y de los grupos existentes. No es necesario mostrar ni modificar NINGUNA información relativa a las reservas de la persona ni de los grupos. Tampoco vamos a hacer cambios ni modificar información básica de las reservas. Por tanto, no es preciso acceder a esta colección en ningún caso. (Carga ***LAZY***)

2Editar persona

(Integrantes [] -> lazy): En esta pantalla mostramos únicamente **nombre, apellidos, dni, email, teléfono**  y **tarjeta de crédito** deuna persona concreta, que son atributos básicos de su estado y los únicos que se pueden modificar. Modificamos el estado de la persona e invocamos la operación *merge*. No es necesario mostrar la colección de integrantes en la pantalla. Tampoco se realizarán cambios ni modificaciones en ella, por lo que no es preciso acceder a esta colección. (Carga ***LAZY***)

(reservas [] -> lazy): En esta pantalla mostramos únicamente **nombre, apellidos, dni, email, teléfono**  y **tarjeta de crédito** deuna persona concreta, que son atributos básicos de su estado y los únicos que se pueden modificar. No es necesario mostrar la colección de reservas en la pantalla. Tampoco se realizarán cambios ni modificaciones en ella, por lo que no es preciso acceder a esta colección. (Carga ***LAZY***)

3 Eliminar persona

4 Listar reservas persona

(RESERVAS [] -> eager): En esta pantalla solo **mostramos** información. Aparece el **nombre completo, dni, email** y **teléfono** de la persona seleccionada (C), que son atributos básicos de su estado, pero TAMBIÉN es necesario mostrar el código y la fecha de inicio de CADA reserva R1, R2… que la persona ha realizado y que son atributos básicos del estado de R1, R2… Cargar la colección de las reservas al cargar la persona **puede suponer una ventaja** en este caso. (Carga ***EAGER***).

(CLIENTE -> lazy): Debemos analizar este caso porque estamos suponiendo una carga EAGER de las reservas R1, R2… de la persona C. Teniendo esto en cuenta, ya estaríamos cargando C en memoria por lo que **todas** las reservas serán de la persona C. Por tanto, es **innecesario** inicializar las referencias desde R1, R2… hacia C. (Carga ***LAZY***)

4.1 Ver reserva

(Integrantes [] -> lazy): En esta pantalla mostramos únicamente atributos básicos del estado de la persona o grupo seleccionado (C). No es necesario mostrar NINGUNA información básica de los integrantes, por lo que no es necesario acceder a esa colección. (Carga ***LAZY***)

(Reservas [] -> lazy): En esta pantalla mostramos únicamente atributos básicos de la persona o grupo seleccionado (C), y datos básicos de UNA de sus reservas (R). Si establecemos una carga EAGER estaríamos cargando TODAS las reservas de esa persona/grupo. Como solo queremos una única reserva, no nos interesa. (Carga ***LAZY***)

(Cliente -> EAGER): Al descartar la carga EAGER para Cliente.reserva, y dado que analizamos las pantallas de forma aislada de las demás, consideramos la opción alternativa cargando la reserva R en memoria antes que la persona/grupo. De esta forma activamos una carga EAGER de Reserva.cliente consiguiendo cargar simultáneamente tanto la reserva R como su único cliente asociado (persona/grupo) C. (Carga ***EAGER***)

4.2 Alta reserva

(integrantes [] -> lazy): En esta pantalla mostramos únicamente atributos básicos de la persona o grupo seleccionado (C). No es necesario mostrar ni modificar NINGUNA información relativa a los integrantes del grupo. Tampoco vamos a realizar ningún cambio en esta colección ni modificar información básica en ellos. Por lo tanto, no es preciso acceder a esta colección en ningún caso. (Carga ***LAZY***)

(reservas [] -> lazy): En esta pantalla cargamos una única persona/grupo. En la pantalla se mostrarán únicamente atributos básicos de su estado. Además, creamos un objeto que representará la nueva reserva R, que asociaremos con la persona/grupo, e invocaremos la operación *persist*.

Consideraciones:

* No es necesario mostrar en la pantalla NINGUNA información relativa a la colección de reservas de la persona/grupo seleccionada. Por ello, una carga EAGER no es necesaria.
* Como el nuevo objeto reserva está en el lado propietario, incluirá una referencia a la persona/grupo (clave foránea). (Carga ***LAZY***)

4.3 Modificar reserva

4.4 Eliminar reserva +- = 2.3.3 Editar artigo

5 Alta grupo

La pantalla se corresponde con un formulario vacío. Por ello, NO es necesario cargar ningún tipo de dato. (Carga ***NO NECESARIA***)

5.1 Añadir integrantes / 6.1.2.1 Añadir integrante

(integrantes [] -> lazy): En esta pantalla cargamos un único grupo. En la pantalla solo mostramos los atributos básicos de su estado. También debemos cargar **todas** las personas existentes en la base de datos para poder seleccionar las que formarán parte de los integrantes del grupo. Incluiremos en su colección de integrantes referencias a las personas que conforman el grupo. Como se encuentra del lado propietario (asociación unidireccional muchos a muchos persona -> grupo), el proveedor de persistencia creará las referencias necesarias. (Carga ***LAZY***)

(reservas [] -> lazy): En esta pantalla **mostramos** únicamente atributos básicos del estado del grupo seleccionado. No es necesario mostrar ni modificar NINGUNA información relativa a las reservas del grupo. Tampoco vamos a hacer cambios ni modificar información básica de las reservas. Por tanto, no es preciso acceder a esta colección en ningún caso. (Carga ***LAZY***)

6 Listar grupos  
(integrantes [] -> lazy): En esta pantalla solo se **muestra** información. Aparecen los atributos básicos de su estado de todos los grupos (C1, C2, …). No es necesario mostrar NINGUNA información relativa a los integrantes de dichos grupos, por ello, NO es necesario acceder a esta colección en ningún caso. (Carga ***LAZY***)

(reservas []-> lazy): En esta pantalla solo se **muestra** información. Aparecen los atributos básicos de su estado de todos los grupos (C1, C2, …). No es necesario mostrar NINGUNA información relativa a los integrantes de dichos grupos, por ello, NO es necesario acceder a esta colección en ningún caso. (Carga ***LAZY***)

6.1 Ver grupo

(integrantes [] -> lazy): En esta pantalla solo se **muestra** información. Aparecen los atributos básicos de su estado del grupo C seleccionado. No es necesario mostrar NINGUNA información relativa a los integrantes de dichos grupos, por ello, NO es necesario acceder a esta colección. (Carga ***LAZY***)

(reservas [] -> lazy): En esta pantalla solo se **muestra** información. Aparecen los atributos básicos de su estado del grupo C seleccionado. No es necesario mostrar NINGUNA información relativa a las reservas de dichos grupos, por ello, NO es necesario acceder a esta colección. (Carga ***LAZY***)

6.1.2 Listar integrantes

(integrantes [] -> eager): En esta pantalla solo se **muestra** información. Aparecen los atributos básicos de su estado del grupo C seleccionado, pero TAMBIÉN es necesario mostrar el **nombre completo** de CADA integrante perteneciente al grupo y que son atributos básicos del estado de dichos integrantes. Cargar la colección de los integrantes **puede suponer una ventaja** en este caso (e.g añadir un nuevo integrante). (Carga ***EAGER***)

(reservas [] -> lazy): En esta pantalla solo se **muestra** información. Aparecen los atributos básicos de su estado del grupo C seleccionado y de su colección de integrantes. No es necesario mostrar NINGUNA información relativa a las reservas de dichos grupos y/o integrantes, por ello, NO es necesario acceder a esta colección. (Carga ***LAZY***)

6.1.2.2 Eliminar integrante

6.2 Editar grupo

(integrantes [] -> eager): En esta pantalla mostramos los datos básicos de un grupo, pero también necesitamos saber que integrantes lo componen. En este caso podemos modificar información básica del grupo, pero también su colección de integrantes (añadiendo o eliminando integrantes). Por ello se aplicará una operación *merge* sobre el grupo.

Consideraciones:

* La carga EAGER es beneficiosa porque el grupo ya viene con su colección de integrantes inicializada.
* Hay que tener en cuenta que necesitaremos tener todas las personas cargadas en caso de querer añadir más integrantes. Amortizamos el uso del EAGER
* Asociaciones: Cualquier cambio en las personas asociadas como integrantes deberían reflejarse en esta colección. En este caso estamos en el lado propietario por lo que los cambios llegarán a la base de datos. Tener precargados los datos de persona nos será útil en caso de añadir integrantes. Justifica el uso del EAGER
* Propagación: No hay propagación desde grupo a persona, por lo que no vamos a modificar información propia de persona. En este caso no se justifica la carga EAGER. Sin embargo, hacer la carga anticipada no representa un problema y nos brinda más beneficios que inconvenientes, por lo que elegimos carga ***EAGER***.

(reservas [] -> lazy): En esta pantalla mostramos los datos básicos de un grupo, pero también necesitamos saber que integrantes lo componen. No es necesario mostrar ni modificar NINGUNA información relativa a las reservas del grupo seleccionado. Tampoco se realizarán cambios ni modificaciones de información básica de las mismas. Por tanto, no es necesario acceder a esta colección en ningún caso. (Carga ***LAZY***)

6.3 Eliminar grupo

6.4 Listar reservas grupo

(INTEGRANTES [] -> lazy): En esta pantalla solo **mostramos** información. Aparece únicamente el **código, email** y **teléfono** asociados al grupo seleccionado (C), que son atributos básicos de su estado. No es necesario mostrar NINGUNA información relativa a los integrantes del grupo, por lo que no es necesario acceder a esa colección en ningún caso. (Carga ***LAZY***)

(RESERVAS [] -> eager): En esta pantalla solo **mostramos** información. Aparece únicamente el **código, email** y **teléfono** asociados al grupo seleccionado (C), que son atributos básicos de su estado, pero TAMBIÉN es necesario mostrar el **código** y la **fecha de inicio** de CADA reserva R1, R2… que la persona ha realizado y que son atributos básicos del estado de R1, R2… Cargar la colección de las reservas al cargar la persona **puede suponer una ventaja** en este caso. (Carga ***EAGER***).

(CLIENTE -> lazy): Debemos analizar este caso porque estamos suponiendo una carga EAGER de las reservas R1, R2… del grupo C. Teniendo esto en cuenta, ya estaríamos cargando C en memoria por lo que **todas** las reservas serán de ese grupo C. Por tanto, es **innecesario** inicializar las referencias desde R1, R2… hacia C. (Carga ***LAZY***)

7 Listar reservas

(cliente -> eager): En esta pantalla solo **mostramos** información. Aparecen únicamente atributos básicos del estado de reserva. Sin embargo, en este caso nos interesa traerlo todo pues estamos listando TODAS las reservas de TODOS los clientes. (Carga ***EAGER***)

7.1 Filtro reservas por fecha

(cliente -> lazy): En esta pantalla solo **mostramos** información. Aparecen únicamente atributos básicos del estado de reserva. Sin embargo, a diferencia del caso *7 Listar reservas*, no nos interesa cargar todas las reservas, sino solo las que cumplan la condición del filtro. (Carga ***LAZY***)

7.2 Filtro reservas por nombre albergue

(cliente -> lazy): En esta pantalla solo **mostramos** información. Aparecen únicamente atributos básicos del estado de reserva. Sin embargo, a diferencia del caso *7 Listar reservas*, no nos interesa cargar todas las reservas, sino solo las que cumplan la condición del filtro. (Carga ***LAZY***)