

Analysis Report



Álvaro Navarro Rivera (alvnavriv@alum.us.es)
Mario Sánchez Naranjo (marsannar2@alum.us.es)
Gabriel Vacaro Goytia (gabvacgoy@alum.us.es)
Ignacio Warleta Murcia (ignwarmur@alum.us.es)

<https://github.com/DP2-L1/ACME-L3-D03>

Diseño y Pruebas II
Grupo C1.02.01

20/04/2023

Tabla de contenido

Sumario..... 2

Historial de versiones 3

Introducción..... 4

Contenido..... 5

Conclusiones..... 8

Bibliografía..... 8

Sumario

En este documento de análisis se lleva a cabo el análisis de los requisitos del tercer entregable de la asignatura.

Historial de versiones

Fecha	Versión	Descripción	Sprint
20/04/2023	1.0	Se ha analizado el documento de análisis del tercer entregable.	e

Introducción

En este documento se ha llevado a cabo el segundo documento de análisis correspondiente al tercer entregable del 21 de abril. En este se realiza un análisis de los requisitos planteados en el documento de requisitos del grupo.

Este escrito se ha organizado según se indica en el anexo de la asignatura y en orden creciente numérico de los requisitos, aunque en este caso solo se trata de uno.

Requisito 21

[MANDATORY] Operations by any principals on peeps:

- List the peeps and show their details.
- Publish a peep. By default, the nick must be kept blank if the principal is anonymous; otherwise, it must be filled with the principal's full name; anyway, it can be changed at will.

Para este requisito he creado el repositorio, los servicios (listAll, show, create), el controlador y las vistas tanto con el rol Anonymous como con el rol Authenticated ya que nos piden que haga una función distinta dependiendo de si esta autenticado o no. El nick es un atributo que no puede dejarse en blanco, debido a esto interpretamos que a lo que se refiere el requisito es que al entrar en el formulario como anónimo el Nick aparezca vacío y si entras como autenticado aparece directamente el nombre completo del usuario. En ambos casos se puede cambiar el Nick.

Complicaciones Encontradas:

Al principio no sabíamos cómo interpretar la parte de que el nick en anónimo debía ser blanco, pero ya que en el entregable anterior se nos especificó que no podía ser blanco (señalado como @NotBlank) solo había una interpretación posible. También podríamos haber puesto un nick por defecto a los anónimos y que en el formulario el nick apuntarlo como readonly, pero como no se especifica que se deba guardar de ninguna manera pues hemos pensado que la otra interpretación era más correcta.

Requisito 24

Operations by authenticated principals on offers:

- List the offers in the system and show their details.

Para este requisito he creado el repositorio, los servicios (ListAll y show), el controlador y las vistas con rol Authenticated y objeto Offer.

Lista las offers y se pueden ver los detalles.

Complicaciones Encontradas:

Ninguna complicación encontrada en este requisito

Requisito 27

[MANDATORY] Operations by any principals on banners:

- Display any of them randomly, as long as its display period are active; the banners must be shown in every page as little intrusively as possible.

Para este requisito he creado, en el paquete componen, he creado el repositorio y el advisor del objeto Banner, además de modificar el footer.

Complicaciones Encontradas:

Para lograr implementar el requisito de que siempre devuelva un banner dentro de fecha, hemos añadido el atributo derivado isAvailable el cual nos dice si estábamos o no dentro del periodo del banner, tardamos bastante en conseguir lo que buscábamos, intentamos varias cosas entre ellas validación intentamos poner la validación en la query directamente, pero al tener como dato de entrada una PageRequest saltaba error en la query al añadirla.

Finalmente utilizando un while (isAvailable == false) conseguimos que devolviera solo los banners que estén dentro de la fecha.

Requisito 28

[MANDATORY] Operations by administrators on banners:

- List the banners and show their details.
- Create, update, or delete a banner.

Para este requisito he creado, de nuevo, el repositorio, los servicios (ListAll, show, create, delete y update), el controlador y las vistas.

Un administrador puede ver todos los banners, ver sus detalles, crear banners, editarlos y borrarlos.

La fecha de inicio debe ser después de la fecha de instanciación/actualización.

La fecha de final debe ir después que la fecha de inicio.

La promoción debe durar al menos 1 semana.

Entendemos como momento de modificación en el momento en el que se están realizando los cambios, es decir, el momento exacto en el que estas cambiando los datos.

Complicaciones Encontradas:

Tardamos un poco en entender cómo funciona el tiempo al lanzarse la aplicación.

Requisito 29

[MANDATORY] Moments, money amounts, and Booleans must be internationalised when they are shown. Other kinds of data are not expected to be internationalised. This requirement must be fulfilled in this and every subsequent group or individual deliverable.

Este requisito nos pide que internacionalicemos el formato de las fechas, el dinero y los booleanos. Lo cual hemos implementado a lo largo del entregable.

Complicaciones Encontradas:

No entendíamos muy bien a lo que se refería el requisito, pero al entrar en el foro vimos uno de los hilos en el cual el profesor explicaba que la función de internacionalización ya lo hace el framework automáticamente si has utilizado correctamente las etiquetas para cada atributo.

Adjunto imagen de la respuesta (no encuentro un enlace directo a este hilo):



RAFAEL CORCHUELO GIL

Hace 6 días

RE: [D&T] Implementación de requisito grupal 29

Estimado Olegario:

Gracias por seguir los consejos en relación con cómo plantear una petición de ayuda.

El requisito sobre el que pregunta es un tanto atípico dado que realmente no hay que hacer nada para cumplirlo. El framework Acme ya se encarga automáticamente de i18n los momentos y cantidades de dinero tanto en la entrada de datos como a la hora de mostrarlos. Los Booleanos (*) pueden gestionar mediante etiquetas de tipo "acme:input-checkbox", estas etiquetas muestran los booleanos como un cuadrado en el que Ud. puede hacer click para activarlo (true) o desactivarlo (false); al tratarse de una representación gráfica no requiere de i18n específica alguna dado que tanto en español como en inglés dicha representación tiene el mismo significado.

Pese a tratarse de un requisito que se cumple trivialmente, requiere de un poco más de análisis. Los Booleanos no son más que un tipo enumerado predefinido; se pueden considerar al mismo nivel que un enumerado definido por el usuario como "ApplicationStatus" en el caso de Acme Jobs o el enumerado que representa el tipo de curso, clase, actividad, etcétera en el caso de Acme L3 (THEORETICAL, HANDS ON o BALANCED). Por desgracia, el framework no proporciona un buen soporte para la i18n de los valores enumerados, por lo que en sus proyectos no influye para nada en la evaluación si i18n o no esos valores. Por ejemplo: si en un listado muestran Udes. un Booleano, sus valores saldrán siempre como "true" o "false" independientemente de que el locale sea "en" o "es". Ocurre lo mismo cuando muestran la naturaleza de un curso: siempre saldrá "THEORETICAL" o "HANDS-ON" o "BALANCED" con independencia al locale seleccionado.

El problema de la i18n de los valores enumerados NO es un problema específico de Acme Jobs o de Acme L3, por lo que requiere de una solución general a nivel de framework, pero, por desgracia, aún no se ha podido diseñar una solución genérica suficientemente buena. Tampoco se considera éste un issue prioritario de corregir. Por lo tanto, mientras no tengamos una buena solución no es necesario buscar ninguna solución ad hoc en sus proyectos.

Dicho de otra forma: tan sólo hay que marcar el requisito como satisfecho, no hay que hacer nada especial para cumplirlo.

Saludos. RC

PS: Tenga en cuenta que el adjetivo "Booleano" se escribe en mayúscula dado que hace referencia a George Boole, que fue uno de los pioneros del tratamiento automático de datos. En java, la diferencia entre escribir "boolean" y "Boolean" es que en el primer caso se trata de un tipo primitivo que tan sólo admite los valores "true" y "false", mientras que el segundo es un tipo wrapper que admite los valores "true", "false" y "null".

Bibliografía

Intencionalmente vacío.