

El coche de Peblo: MenuApp



Equipo:

- Jesús Calle Lozano -> Programador//Documentación
- Javier Juarez Alhama -> Programador//Jefe
- Enrique Garrido Sánchez -> Tester//Jefe
- Pablo Díaz García -> Tester//Jefe
- Francisco Fernández Montiel -> Documentación//Programador
- Fernando Astorga Cobos -> Documentación//Programador
- Martin Daglio Cabrera -> Documentación// Tester



E.T.S. INGENIERÍA INFORMÁTICA

Índice



- Introducción – El problema
- La solución
- Actividades de Ing. Sw
 - Requisitos
 - Arquitectura
 - Modelos
 - Patrones/Principios
 - Pruebas
- Desarrollo/Despliegue
 - Estrategias y herramientas
 - Modelo de Implementación (qué es y qué no es real)
 - Despliegue
- Resultados
- Conclusiones

Introducción: El problema



- Falta de información del tema.
- Algo útil.
- De existir la gente lo usaría.
- Ayuda a gente con dietas especiales
- Bueno para las cafeterías.
- Incentiva a llevar una dieta equilibrada.

La solución



- App simple.
- Interfaz intuitiva y fácil de entender.
- Con acceso a toda la información por facultad y por plato.
- Posibilidad de filtrar información.

req [Package] 3.1 Functional requirements [15a Pantallas]

```
graph TD
    Vistas["«functionalRequirement»  
Vistas  
Id = \"25\"  
Text = \"Explicacion de las diferentes vistas que encontraremos en la aplicacion\""]
    AdminReg["«functionalRequirement»  
Pantalla de registro/inicial  
Id = \"30\"  
Text = \"En esta pantalla encontraremos el registro, un boton para acceder de modo usuario y un login con usuario contraseña para acceder como administrador.\""]
    AdminMod["«functionalRequirement»  
Admin  
Id = \"29\"  
Text = \"Entrar modo admin-> Las pantallas seran iguales que las de usuario pero con las modificaciones aqui indicadas.\""]
    AdminBusq["«functionalRequirement»  
Pantalla de busqueda/Admin  
Id = \"36\"  
Text = \"Pasara a una pantalla donde saldran dos botones. 'Añadir plato' 'Añadir universidad'\""]
    AdminPlato["«functionalRequirement»  
Añadir plato  
Id = \"61\"  
Text = \"Sera una nueva pantalla en la que podras introducir por teclado los datos del nuevo plato\""]
    AdminUniv["«functionalRequirement»  
Añadir Universidad  
Id = \"62\"  
Text = \"Sera una nueva pantalla en la que podras introducir por teclado los datos de la nueva universidad\""]
    AdminModUniv["«functionalRequirement»  
Modificar/Eliminar Universidades  
Id = \"58\"  
Text = \"Al realizar la busqueda de la universidad, nos aparecera un boton de 'Modificar' o 'Eliminar' para la universidad el cual al pinchar en el nos lleva a una nueva pantalla que nos permite modificar los datos de la universidad en cuestion o eliminarla de la base de datos \""]
    AdminModPlato["«functionalRequirement»  
Modificar/Eliminar Platos  
Id = \"63\"  
Text = \"Al realizar la busqueda de la universidad, nos aparecera un boton de 'Modificar' o 'Eliminar' para el plato el cual al pinchar en el nos lleva a una nueva pantalla que nos permite modificar los datos del plato en cuestion o eliminarlo de la base de datos donde podras elegir si eliminarlo exclusivamente de esta facultad o de todas.\""]
    AdminRep["«functionalRequirement»  
Opciones->Reportes  
Id = \"60\"  
Text = \"Apareceran una lista de reportes, que al clickar en ellos nos mostrará el contenido de dicho reporte. Existira tambien un boton de 'Atras'\""]
    UsuarioMod["«functionalRequirement»  
Usuario  
Id = \"26\"  
Text = \"Entrar modo usuario\""]
    UsuarioBusq["«functionalRequirement»  
Pantalla de busqueda  
Id = \"27\"  
Text = \"En esta pantalla nos aparecera el buscador de universidades, además, aparecera un boton de opciones.\""]
    UsuarioBusqReal["«functionalRequirement»  
Buscar  
Id = \"31\"  
Text = \"\""]
    UsuarioBusqRealReal["«functionalRequirement»  
Pantalla de busqueda realizada  
Id = \"32\"  
Text = \"Al realizar la busqueda se recarga la vista y se ofrece una lista con los platos y sus características, aparte de un boton de 'Atras' para volver a la pantalla anterior y realizar otra busqueda\""]
    UsuarioIdiom["«functionalRequirement»  
Pantalla idioma  
Id = \"34\"  
Text = \"Aparecera una lista de los idiomas disponibles para la aplicacion, al seleccionarlos y confirmar el cambio se recargar la vista en ese idioma, tambien en esta pantalla habra un boton de 'Atras' \""]
    UsuarioOpciones["«functionalRequirement»  
Opciones  
Id = \"28\"  
Text = \"\""]
    UsuarioOpcionesReal["«functionalRequirement»  
Pantalla opciones  
Id = \"33\"  
Text = \"Mostrara tres botones: -Idioma -Reportes -Atras\""]
    UsuarioRep["«functionalRequirement»  
Pantalla reportes  
Id = \"35\"  
Text = \"En esta vista se podra escribir por teclado con la finalidad de mandar a los administradores los errores que hayan experimentado \""]

    Vistas -.-> AdminReg
    AdminReg -.-> AdminMod
    AdminMod -.-> AdminBusq
    AdminBusq -.-> AdminPlato
    AdminBusq -.-> AdminUniv
    AdminBusq -.-> AdminModUniv
    AdminBusq -.-> AdminRep
    AdminPlato -.-> AdminModPlato
    AdminUniv -.-> AdminModPlato
    AdminModUniv -.-> AdminModPlato
    AdminRep -.-> AdminModPlato
    UsuarioMod -.-> AdminReg
    UsuarioMod -.-> UsuarioBusq
    UsuarioMod -.-> UsuarioOpciones
    UsuarioBusq -.-> UsuarioBusqReal
    UsuarioBusqReal -.-> UsuarioBusqRealReal
    UsuarioBusqRealReal -.-> UsuarioIdiom
    UsuarioBusqRealReal -.-> UsuarioOpcionesReal
    UsuarioOpcionesReal -.-> UsuarioRep
```

The diagram illustrates the functional requirements for the 'Pantallas' package, organized into two main user paths: Admin and Usuario.

Admin Path:

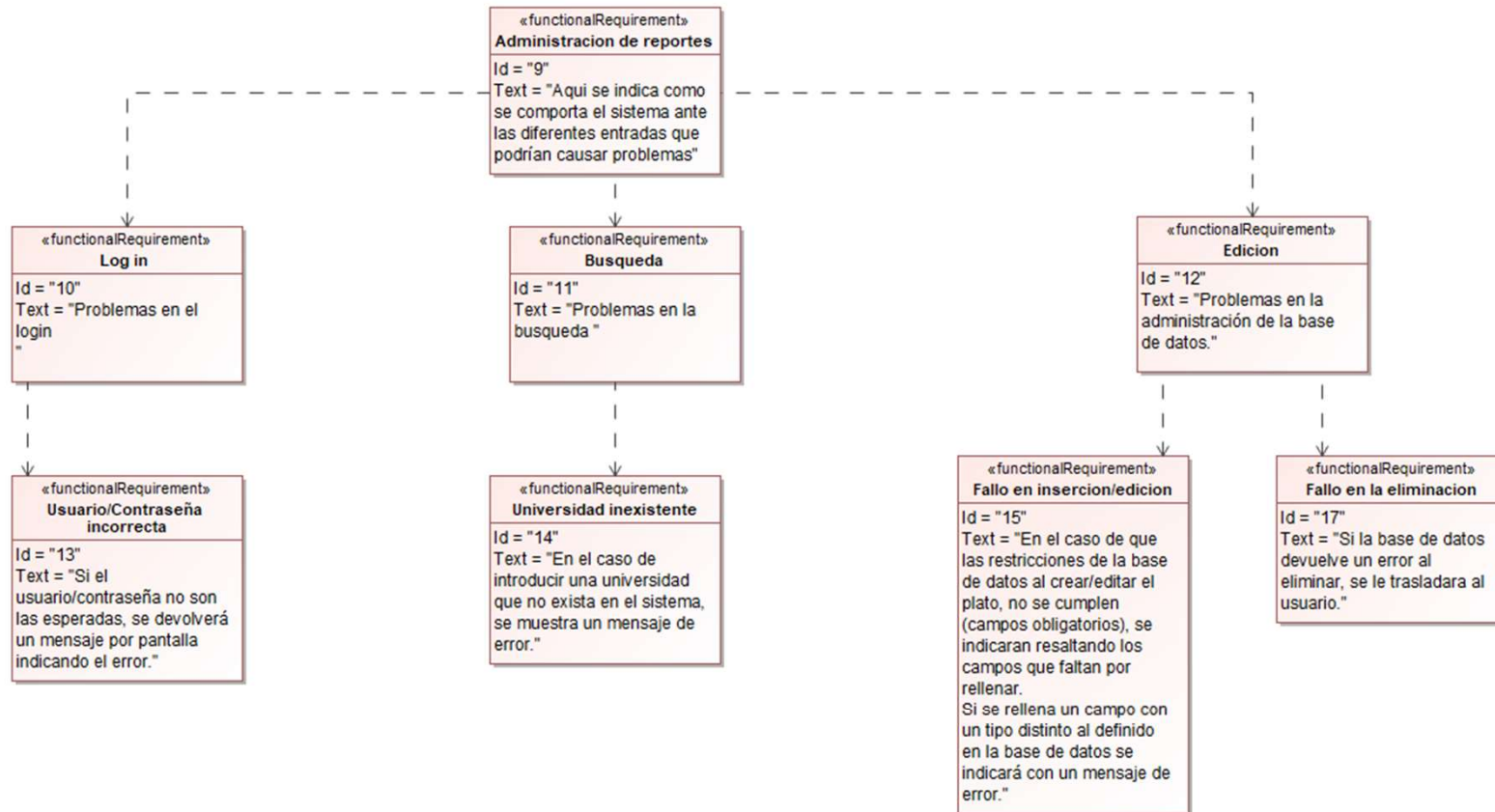
- «functionalRequirement» Vistas** (Id = "25"): Explains the different views found in the application.
- «functionalRequirement» Pantalla de registro/inicial** (Id = "30"): Contains the registration, login (with username/password), and admin access buttons.
- «functionalRequirement» Admin** (Id = "29"): Allows entering admin mode, where all screens are identical to the user mode but with specific modifications.
- «functionalRequirement» Pantalla de busqueda/Admin** (Id = "36"): Displays two buttons: "Añadir plato" and "Añadir universidad".
- «functionalRequirement» Añadir plato** (Id = "61"): A screen for adding a new dish by typing its data.
- «functionalRequirement» Añadir Universidad** (Id = "62"): A screen for adding a new university by typing its data.
- «functionalRequirement» Modificar/Eliminar Universidades** (Id = "58"): After searching for a university, it shows "Modificar" or "Eliminar" buttons to edit or delete the university's data from the database.
- «functionalRequirement» Modificar/Eliminar Platos** (Id = "63"): After searching for a university, it shows "Modificar" or "Eliminar" buttons to edit or delete the dish's data from the database (specifically for this faculty or all).
- «functionalRequirement» Opciones->Reportes** (Id = "60"): Displays a list of reports, showing their content when clicked. Includes an "Atras" button.

Usuario Path:

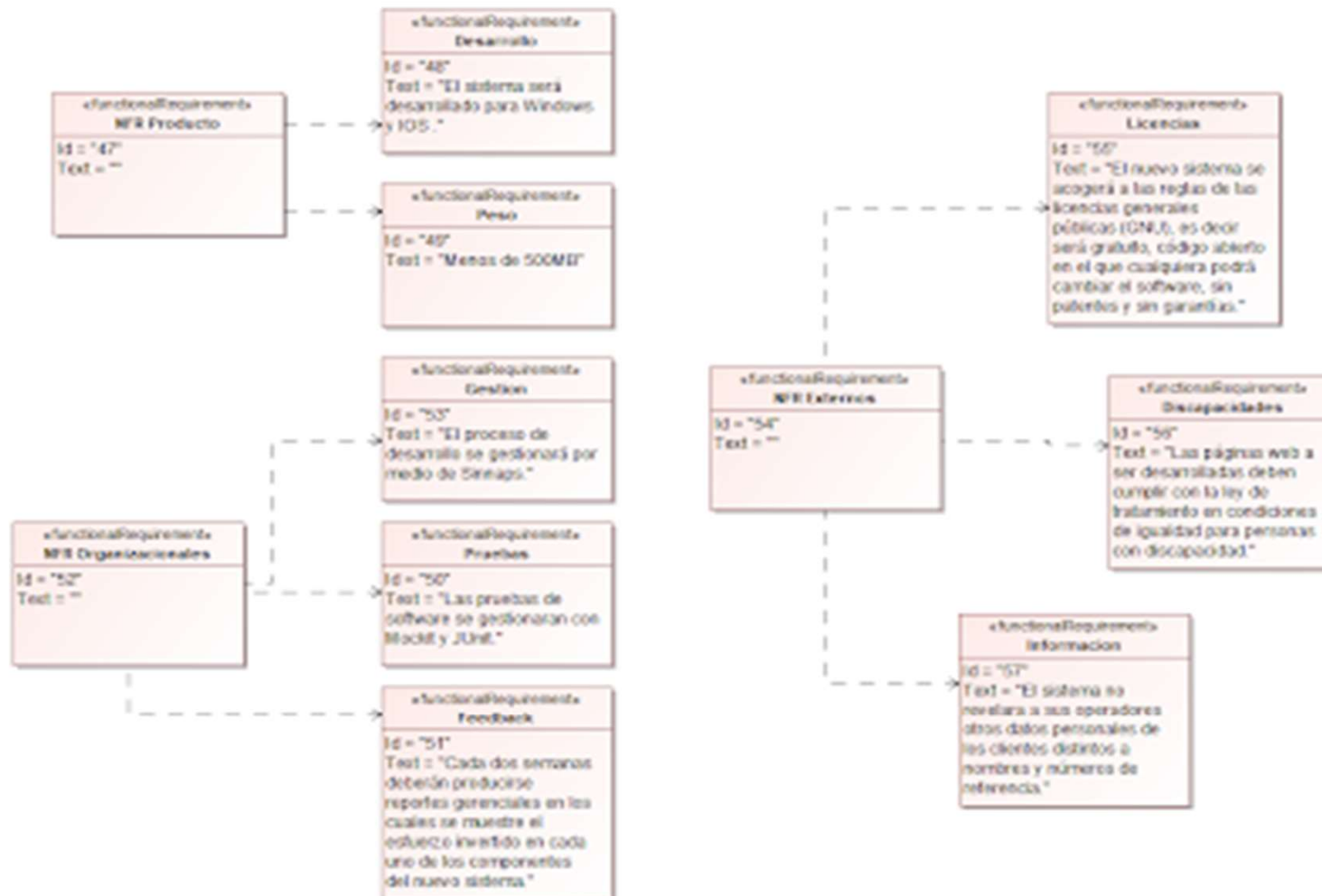
- «functionalRequirement» Usuario** (Id = "26"): Allows entering user mode.
- «functionalRequirement» Pantalla de busqueda** (Id = "27"): Displays the university search bar and an "Opciones" button.
- «functionalRequirement» Buscar** (Id = "31"): The search button.
- «functionalRequirement» Pantalla de busqueda realizada** (Id = "32"): After searching, it displays a list of dishes with their characteristics, an "Atras" button to return to the previous screen, and the ability to perform another search.
- «functionalRequirement» Pantalla idioma** (Id = "34"): Displays a list of available languages for the application. After selection and confirmation, the view is reloaded in that language. Includes an "Atras" button.
- «functionalRequirement» Opciones** (Id = "28"): The options button.
- «functionalRequirement» Pantalla opciones** (Id = "33"): Displays three buttons: "Idioma", "Reportes", and "Atras".
- «functionalRequirement» Pantalla reportes** (Id = "35"): A view where users can type errors to report to administrators.

RF Reportes

req [Package] 3.1 Functional requirements [Reportes]



Requisitos no Funcionales



Casos de Uso

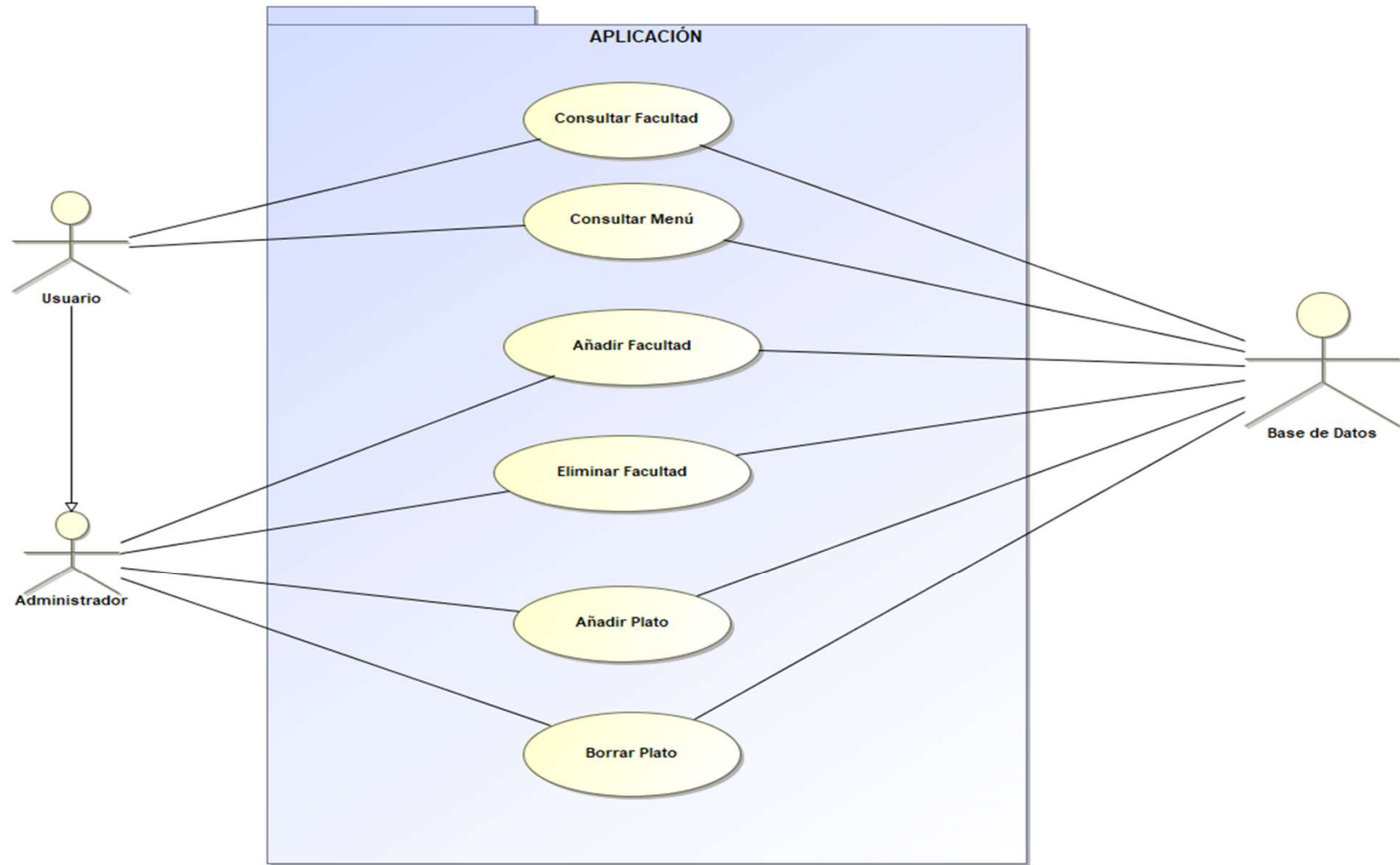
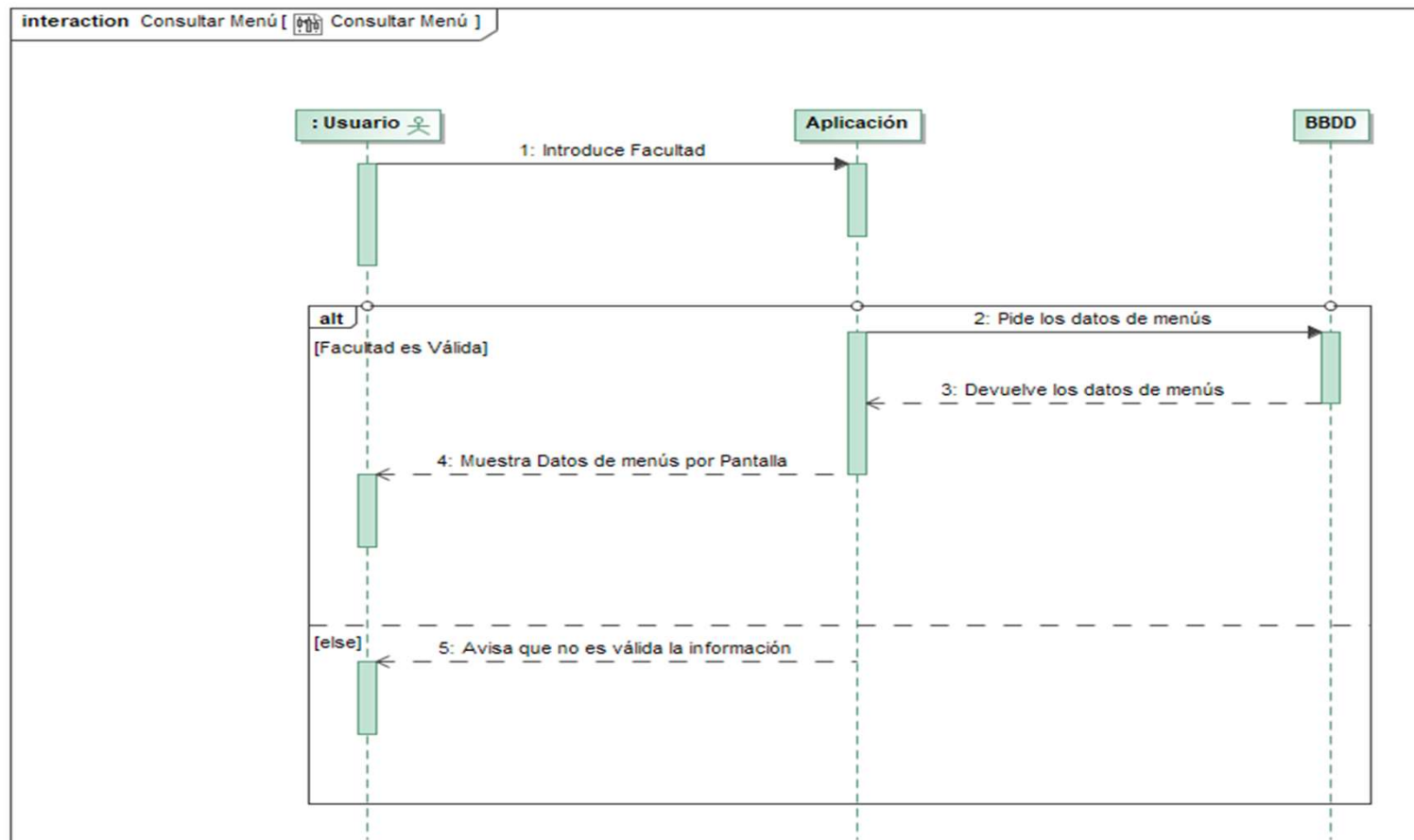
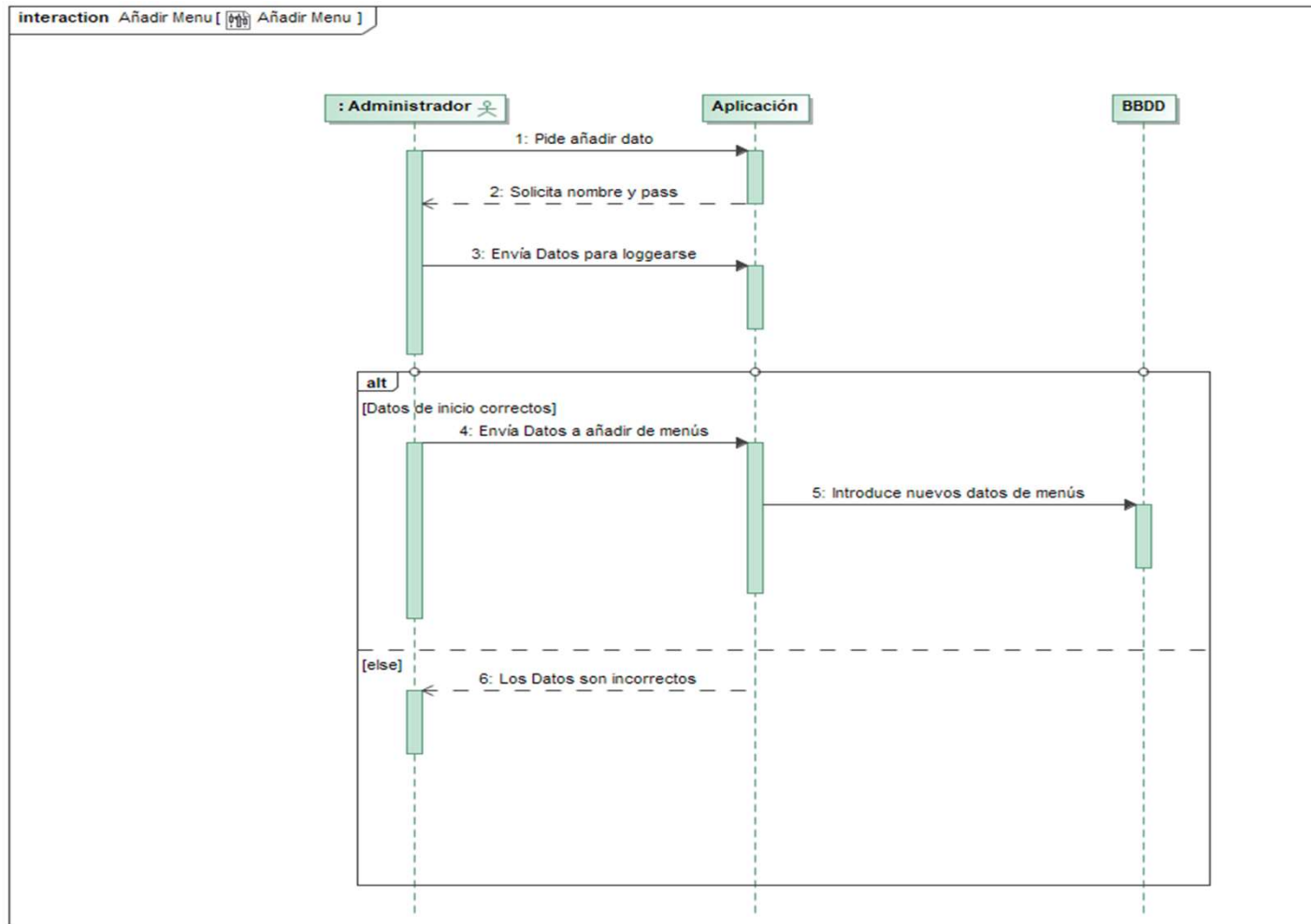


Diagrama de Secuencia



Diagramas de secuencia



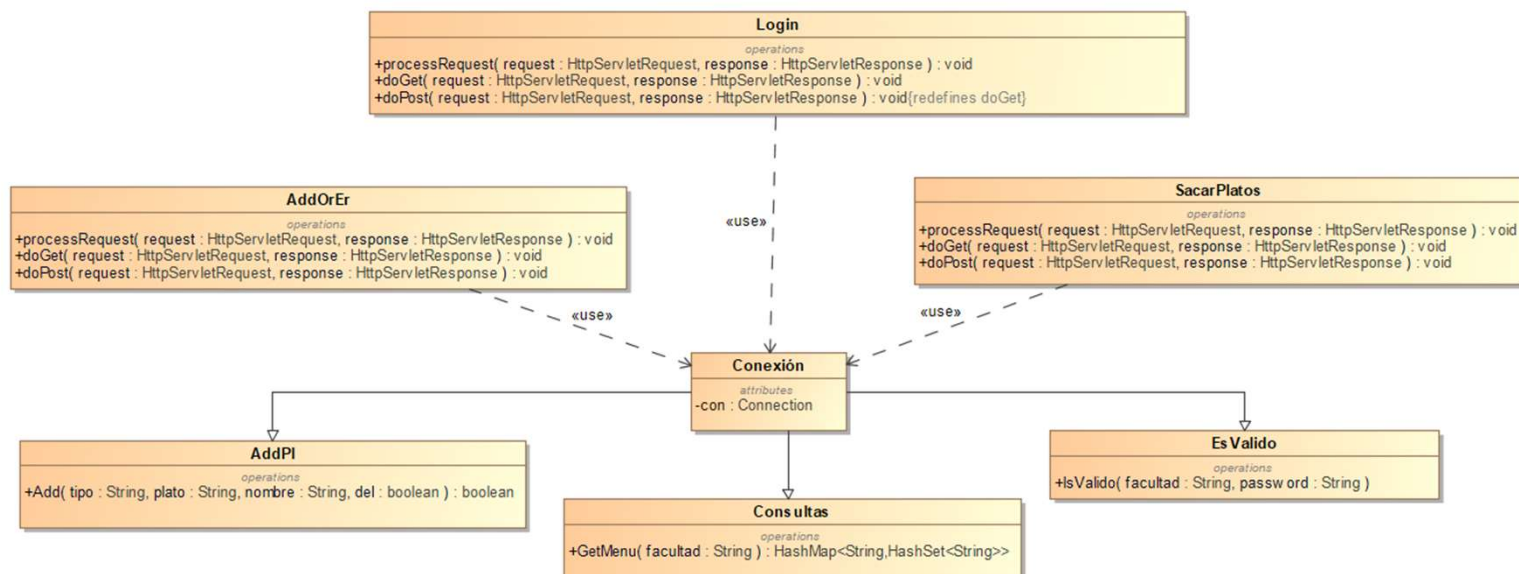
Actividades de ingeniería del Software: Arquitectura

Nuestro Diagrama
de clases
original :



Actividades de ingeniería del Software: Arquitectura

Nuestro Diagrama de clases implementado:



Patrones/Principios

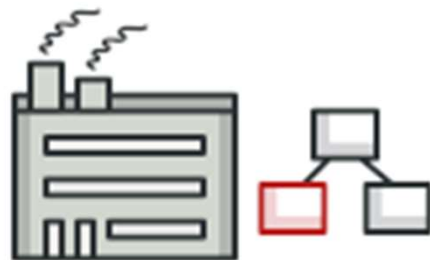


- El patrón a seguir para realizar al app ha sido el método de fábrica, que es un patrón de diseño creacional.
- Nos proporciona una interfaz para crear objetos en una superclase, permitiendo a las subclases modificar los tipos de objeto.

Patrones/Principios



- Esto nos permite expandir nuestra aplicación en el caso de que quisieramos añadir mas componentes como desayunos o actividades de la facultad añadiendo subclases y sin modificar el código principal.



Factory Method

Desarrollo: Herramientas



- MagicDraw
- NetBeans
- Java
- Sinnaps



NetBeans



Despliegue



- Despliegue Actual: El despliegue consistiría en compilar el archivo java con un compilador.
- Despliegue deseable: Queremos que sea una aplicación funcional para Android/iOS que puede ser descargada desde la appStore

Modelo de implementación



El modelo implementado es vista-controlador.

La vista: html que se muestra

Controlador: servlets (uno por cada método)

Modelo: la conexión con la BBDD

Los datos controlados mediante sentencias sql y la BD es tipo derby.

Estrategia



1. Investigar los recursos que utilizaremos más adelante
2. Buscar recursos online para no empezar a trabajar de cero (librerías, botones, etc)
3. Modificar este código para que se acomode a nuestro modelo
4. Agregar las funciones que faltan para acabar de perfilar el código final
5. (Por Hacer) Mejorar el aspecto visual

Resultado



Conseguimos obtener un programa que:

- Tiene un buscador
- Posee opción de admin para introducir datos y eliminarlos
- Requiere autenticación para modificarla
- Es funcional
- Tiene la mayoría de la idea original implementada

Conclusiones



- Hemos conseguido llegar a un primer acercamiento de App.
- Problemas con la programación web por desconocimiento.
- Problemas con las pruebas.
- No hemos conseguido implementar toda la app.

FIN