4 de abril de 2025

Manual de formación y desarrollo de Yubiq Studio

Rev. 5.1

|  |  |  |
| --- | --- | --- |
| ESCRITO POR: |  |  |
| VERIFICADO POR: |  |  |
| APROBADO POR: |  |  |

# Registro de cambios en los documentos

|  |  |  |
| --- | --- | --- |
| Rev | Descripción | Fechas |
| 5.0.14 | Trabajo en curso | TBD |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Índice

[Registro de cambios en los documentos 3](#_Toc198300034)

[Índice 4](#_Toc198300035)

[Introducción 6](#_Toc198300036)

[La metodología ST.A.R. 7](#_Toc198300037)

[Etapas 7](#_Toc198300038)

[Acciones 7](#_Toc198300039)

[Funciones 7](#_Toc198300040)

[WorkItem (Práctica) 7](#_Toc198300041)

[Ejemplo de representación STAR 7](#_Toc198300042)

[Estudio Yubiq (YS) 10](#_Toc198300043)

[Instalación 11](#_Toc198300044)

[La caja de arena 12](#_Toc198300045)

[Proyecto de formación: Vuelo 13](#_Toc198300046)

[Descripción del proceso 13](#_Toc198300047)

[Los pasos necesarios 14](#_Toc198300048)

[Proyecto y flujo de trabajo 15](#_Toc198300049)

[Modelización de un proyecto 15](#_Toc198300050)

[Modelización de un flujo de trabajo 16](#_Toc198300051)

[Funciones 17](#_Toc198300052)

[Funciones estáticas 17](#_Toc198300053)

[Funciones dinámicas 17](#_Toc198300054)

[Modelización de roles estáticos y dinámicos 18](#_Toc198300055)

[Datos SandBox: Usuarios 19](#_Toc198300056)

[Objeto social 20](#_Toc198300057)

[Ejemplos de objetos de negocio 20](#_Toc198300058)

[Modelado de objetos empresariales 21](#_Toc198300059)

[Codificación en C# de Business Objects 22](#_Toc198300060)

[Objeto comercial (EnumValue) 24](#_Toc198300061)

[Modelado de objetos empresariales (EnumValue) 24](#_Toc198300062)

[Datos SandBox: Objetos empresariales 25](#_Toc198300063)

[SandBoxData/Modelado de objetos empresariales 26](#_Toc198300064)

[Elemento de trabajo 27](#_Toc198300065)

[Modelización del WorkItem 27](#_Toc198300066)

[WorkItem c# codificación 28](#_Toc198300067)

[Mapa del proceso 29](#_Toc198300068)

[Modelización del WorkItem 29](#_Toc198300069)

[Formularios 32](#_Toc198300070)

# Introducción

El propósito de este documento es complementar el Manual de Desarrollo de la Solución Yubiq BPM con el curso de formación correspondiente

Tras una visión general de la metodología STAR, que es la base del modelado BPM, tomaremos un flujo de negocio de ejemplo, como **una solicitud de billetes de avión**, y, aplicándolo paso a paso, entraremos en detalle en todos los aspectos del desarrollo.

Yubiq Studio (a partir de ahora YS) es una herramienta de desarrollo dirigida a dos categorías de especialistas:

**Analistas empresariales**.

YS, con su modo 'Modelador', permite la definición de estructuras básicas sin escribir código (No-Code) tales como: **BusinessObjects, WorkItems, Etapas, Acciones,**

**Desarrolladores de C#.**

El modo "Desarrollador" permite entonces completar las estructuras restantes, como **las CodeLibs**, que contienen código máquina para el tratamiento y la integración de datos, o **las CustomViews** para crear gr áficos personalizados.

Antes de adentrarnos en el desarrollo con YS, introduzcamos los conceptos básicos de la metodología STAR. No se trata de una notación o formalismo como BPMN, sino de una descomposición de un proceso de negocio en Etapas, Acciones y Roles.

# La metodología ST.A.R.

La metodología STAR (Stages, Action, and Roles) se desarrolló en Palo Alto, California, y es un enfoque de gestión de procesos empresariales (BPM) destinado a mejorar la eficiencia y eficacia de los procesos empresariales.

Representa un enfoque estructurado para implantar soluciones de Gestión de Procesos de Negocio (BPM) que hace hincapié en la importancia de la claridad de los pasos, las acciones a realizar y las funciones de los participantes. He aquí un resumen de los principales elementos de la metodología:

## Etapas

Las fases representan las diferentes etapas del proceso de gestión de procesos empresariales. Ejemplos válidos pueden ser:

* **Nuevas prácticas**
* **Prácticas a trabajar**
* **Prácticas erróneas**
* **Archivo**

## Acciones

Las acciones se refieren a las actividades específicas que deben realizarse en cada fase. Ejemplos válidos pueden ser:

* **Editar datos**
* **Pasar a la siguiente fase**
* **Asignar a otro usuario**

## Funciones

Definir claramente las funciones es crucial para el éxito de la metodología. Cada participante tiene responsabilidades específicas y su implicación es esencial para el éxito de la aplicación. Las funciones pueden ser:

* **Solicitante**
* **Aprobador**
* **Oficina de viajes**
* **Director**

## WorkItem (Práctica)

En la metodología STAR, también existe el concepto de **WorkItem**, que representa una unidad específica de trabajo asignada a un individuo o grupo dentro de un proceso. Un WorkItem puede representar cualquier práctica, actividad o tarea que deba completarse para que el proceso continúe o alcance sus objetivos.

## Ejemplo de representación STAR

Si tomamos el ejemplo de un proyecto BPM para "Solicitud de billetes de avión", un ejemplo de **Etapas** y **Acciones** podría ser el siguiente:

1. **Empezando**:
   * Creación de la solicitud de vuelo por el solicitante.
   * Recopilación de la información necesaria (datos del solicitante, detalles del vuelo, etc.).
2. **Verificación** preliminar:
   * Comprobación de la información introducida.
   * Comprobación de la idoneidad de la solicitud (por ejemplo, presupuesto, políticas de la empresa, disponibilidad).
3. **Aprobación**:
   * Remitir la solicitud al responsable designado (supervisor o director).
   * La solicitud está pendiente de aprobación.
   * El responsable de la decisión puede aprobar, rechazar o solicitar más información.
4. **Búsqueda de vuelos**:
   * Póngase en contacto con proveedores (agencias de viajes o sistemas de reservas) para buscar opciones de vuelo.
   * Recopilación de opciones de vuelo disponibles y sus costes.
5. **Opciones de presentación**:
   * Presentación de las opciones de vuelo al solicitante.
   * Recogida de comentarios o elección final por parte del solicitante.
6. **Reserva**:
   * Confirmación y reserva del vuelo seleccionado.
   * Emisión de billetes y comunicación de información al solicitante.
7. **Preparar el viaje**:
   * Envío de información sobre el viaje (itinerario, inscripción, documentación).
   * Verificación del cumplimiento de las normas de viaje (por ejemplo, documentación, equipaje, etc.).
8. **Ejecución del viaje**:
   * El solicitante emprende el viaje.
   * Seguimiento del progreso de la solicitud (cualquier problema o cambio).
9. **Cierre**:
   * Retorno del viajero.
   * Evaluación de la experiencia de viaje (comentarios del solicitante).
   * Cierre de la solicitud y registro de los gastos (reembolso, si procede).
10. **Análisis posterior al viaje**:
    * Recopilar datos sobre el gasto real en relación con el presupuesto.
    * Análisis del rendimiento del flujo de trabajo para su futura optimización.

En la práctica de una solicitud de vuelo, un **WorkItem** puede definirse con varios atributos para garantizar una gestión eficaz de la solicitud. He aquí una lista de los atributos más comunes que podrían asociarse a un WorkItem en este contexto:

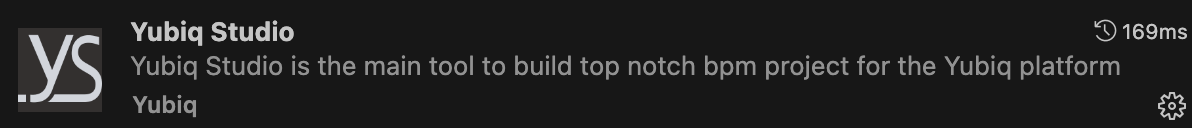
1. **WorkItem ID**: Un identificador único para cada solicitud de vuelo.
2. Fecha de **creación**: Fecha y hora en que se generó la solicitud de vuelo.
3. **Solicitante**: Nombre y datos de contacto de la persona que presenta la solicitud.
4. **Tipo de vuelo**: Especifica si se trata de un vuelo nacional, internacional, de ida y vuelta, etc.
5. **Destino**: Lugar al que desea volar.
6. Fecha y hora de salida: La fecha y hora de salida deseadas.
7. **Fecha y hora de devolución** (si procede): La fecha y hora de devolución deseadas.
8. **Número de pasajeros**: Cantidad de personas que viajarán.
9. **Clase de vuelo**: Clase de viaje preferida (turista, business, primera clase).
10. **Motivo del viaje**: Descripción del motivo de la solicitud (por ejemplo, negocios, vacaciones, urgencia).
11. **Estado de la solicitud**: Estado actual del elemento de trabajo (por ejemplo, "Pendiente de aprobación", "Aprobado", "Rechazado", "Completado").
12. **Fecha de vencimiento**: Fecha límite en la que debe tramitarse la solicitud.
13. **Notas adicionales**: Espacio para comentarios o información relevante enviada por el solicitante.
14. **Asignatario**: Persona o equipo responsable de gestionar la solicitud de vuelo.
15. **Historial de cambios**: Registro de los cambios realizados en la solicitud (cambios de estado, actualizaciones, etc.).

En conclusión, la metodología STAR representa un enfoque innovador y eficaz de la gestión de procesos empresariales. Al definir claramente los pasos, las acciones y las funciones, es posible optimizar el flujo de trabajo, mejorar la eficacia operativa y garantizar una mayor transparencia y responsabilidad dentro de la organización. Su aplicación en proyectos BPM como la reserva de salas de reuniones o la solicitud de billetes de avión demuestra la versatilidad y utilidad de este método. Con Yubiq Studio y la metodología STAR, las empresas pueden simplificar y automatizar sus procesos, facilitando la gestión de las actividades cotidianas y promoviendo una cultura de mejora continua. La integración de estas herramientas y metodologías ofrece una importante ventaja competitiva, contribuyendo al éxito y al crecimiento sostenible de la organización.

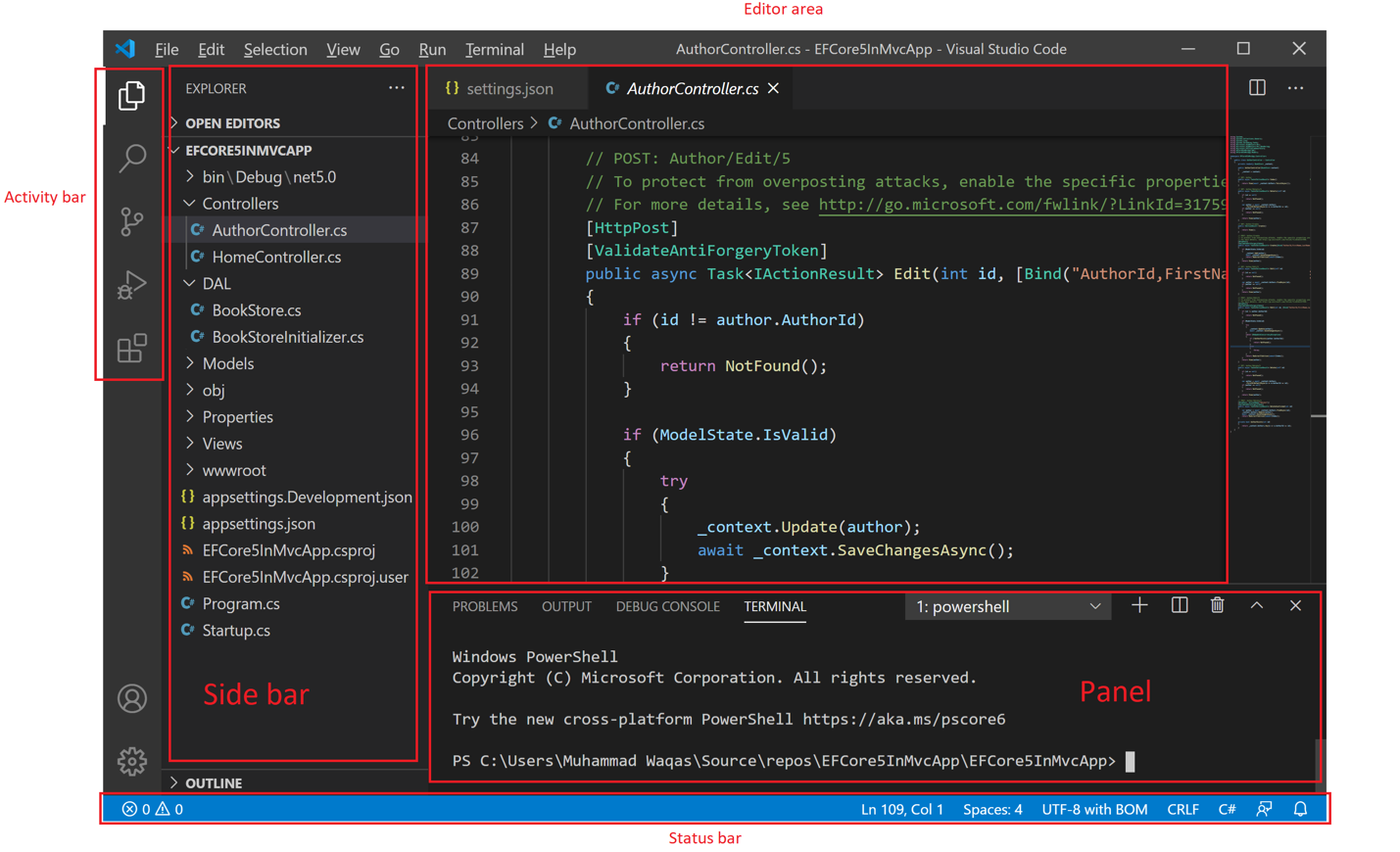
## Estudio Yubiq (YS)

Yubiq Studio se basa por completo en **Visual Studio Code**, (VSC o vscode), un editor de código fuente gratuito desarrollado por Microsoft para **Windows, macOS y Linux**. Sus funciones incluyen depuración, resaltado de sintaxis, completado inteligente de código, fragmentos, refactorización de código y control de versiones integrado con Git. Con casi **un 75% de cuota de mercado y 14 millones de usuarios**, VSCode es el IDE más popular del mercado (fuente: Wikipedia).

Dado que VSC es un IDE genérico, en su tienda hay más de 30.000 extensiones que permiten personalizar prácticamente todo en el IDE. YS es formalmente una extensión de Visual Studio Code, que puede descargarse e instalarse desde la tienda de Microsoft.



Yubiq Studio es una herramienta de desarrollo versátil y avanzada diseñada para optimizar los procesos empresariales y mejorar la eficacia operativa. Su interfaz fácil de usar e intuitiva permite a los desarrolladores crear, gestionar y automatizar WorkFlows sin tener que recurrir a soluciones o scripts externos. Con sus Etapas, Acciones, Roles y WorkItems perfectamente estructurados e interrelacionados, Yubiq Studio ofrece una experiencia de desarrollo integrada y coherente.



Durante la formación con Yubiq Studio, los usuarios pueden explorar y utilizar un exclusivo "árbol funcional" que facilita la navegación y la interacción con los objetos del Proyecto. Los formularios WorkFlow son totalmente realizables con YS, utilizando tanto objetos gráficos estándar, denominados CoreViews, como el mecanismo CustomView para crear objetos gráficos personalizados. Esto convierte a Yubiq Studio en una herramienta potente y flexible para satisfacer las necesidades específicas de cada organización.

Además, Yubiq Studio admite la integración con extensiones como Microsoft .NET C# y GitHub Copilot, que añaden capacidades adicionales de desarrollo y soporte de IA, haciendo que el proceso de desarrollo sea aún más eficiente y productivo.

Con Yubiq Studio, las empresas pueden simplificar y automatizar sus procesos, facilitando la gestión de las actividades cotidianas y fomentando una cultura de mejora continua.

## Instalación

Una vez que haya instalado VSC en un entorno Windows o MAC (Linux no es compatible por el momento), proceda a la instalación del SDK .NET 8: <https://dotnet.microsoft.com/en-us/download/dotnet/8.0>

Y, por lo tanto, de las siguientes ampliaciones del VSC

**Microsoft .**NET C#: Las extensiones básicas del desarrollo en .NET C#.

Immagine che contiene testo, schermata, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.Immagine che contiene testo, schermata

Il contenuto generato dall'IA potrebbe non essere corretto.

**GitHub Copilot**: Para añadir IA durante el desarrollo. Muy recomendable, especialmente si tienes una licencia de usuario completa.

Immagine che contiene testo, schermata, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.

**Extensiones de utilidad:** También recomendamos instalar las siguientes extensiones que ofrecen funcionalidades útiles durante el desarrollo:

**Immagine che contiene testo, schermata, Carattere, software

Il contenuto generato dall'IA potrebbe non essere corretto.**

**Yubiq Studio: ¡Instala por fin** la extensión para Yubiq Studio!

Immagine che contiene testo, schermata, Carattere, nero

Il contenuto generato dall'IA potrebbe non essere corretto.

Immagine che contiene testo, schermata, Carattere, design

Il contenuto generato dall'IA potrebbe non essere corretto.En este punto VSC está listo para crear un nuevo proyecto

Durante tu formación con Yubiq Studio, además de **Etapas**, **Acciones**, **Roles** y el **WorkItem**, descubrirás otros objetos perfectamente estructurados, interrelacionados y explorables desde un único 'árbol funcional'.

Todo el desarrollo de un **proyecto** lleva a cabo exclusivamente con YS sin recurrir a soluciones o scripts externos.

Todos los **formularios** de un **WorkFlow** se realizan con YS y se pueden utilizar todos los objetos gráficos estándar llamados **CoreViews** disponibles. Si necesita crear un objeto gráfico personalizado, el mecanismo **CustomView** dentro de los formularios estándar viene al rescate.

## La caja de arena

SandBox es un entorno de desarrollo aislado que permite simular y probar un proyecto sin afectar al sistema de producción real. Esta herramienta es esencial para la fase inicial de desarrollo, ya que permite trabajar en diversos aspectos del proyecto, como el análisis, el modelado de flujos, la creación de estructuras de datos accesorias y el diseño de formas de interacción. Además, todo el código que no depende del entorno de producción de Yubiq puede escribirse y probarse en la SandBox.

Immagine che contiene schermata, software, testo, Icona del computer

Il contenuto generato dall'IA potrebbe non essere corretto.Con la SandBox, cerca del 90% del diseño puede realizarse localmente, lo que garantiza una mayor eficacia y rapidez en la fase de desarrollo. La demostración del prototipo al cliente también puede hacerse a través de la SandBox, que puede hacerse "portátil" para facilitar las presentaciones en distintos contextos.

Otra ventaja del entorno SandBox es la posibilidad de utilizar BusinessObjects simulados para probar el funcionamiento del proyecto. Esto permite verificar la correcta integración e interacción entre los distintos componentes del sistema antes de pasar a la fase de producción.

La SandBox ofrece así un entorno seguro y controlado para la optimización y verificación de proyectos.

YS lo permite:

* Inicie la SandBox localmente en su navegador en cualquier momento durante el desarrollo de la solución. Ideal para pruebas constantes y depuración local
* Exporte un archivo ZIP con el SandBox para lanzarlo en cualquier cliente Windows o Mac.
* Publica el SandBox en el 'Servidor SandBox' para que cualquiera que posea el enlace pueda ejecutarlo.

# Proyecto de formación: Vuelo

El proyecto Vuelo coordina el proceso de solicitud de billetes de avión, incluidas las distintas aprobaciones y la emisión de billetes por parte de la oficina de viajes.

Utilizaremos este proceso de ejemplo para ilustrar cómo crear una solución BPM con Yubiq. Le aconsejamos que mantenga la solución Flight completa abierta en una ventana de VSC y en la otra su solución que creará paso a paso siguiendo esta guía.

En primer lugar, los empleados interesados deben presentar una solicitud con información detallada sobre el destino, las fechas y los motivos del viaje. A continuación, esta solicitud se somete a la aprobación de los directivos, que evalúan la necesidad y la conformidad de los gastos con la política de la empresa. Una vez obtenida la aprobación, la oficina de viajes procede a la selección de las opciones de vuelo más convenientes y adaptadas a las necesidades de tiempo de los viajeros. Por último, una vez que el empleado ha confirmado la elección, se emiten los billetes y se entregan al solicitante, con lo que finaliza el proceso.

## Thumbnail Image Crea un'immagine che rappresenta un processo di business per la richiesta di biglietti aerei. Includi elementi come un richiedente che compila una richiesta di volo, un manager che approva la richiesta, un ufficio viaggi che prenota il biglietto e un viaggiatore che intraprende il viaggio. Mostra le fasi del processo come creazione della richiesta, verifica preliminare, approvazione, ricerca volo, presentazione opzioni, prenotazione, preparazione al viaggio, esecuzione del viaggio, chiusura e analisi post-viaggio. Utilizza uno stile professionale e chiaro, con icone rappresentative per ogni fase e ruoli definiti.Descripción del proceso

1. Tanto los usuarios Empleados como Directores tienen la funcionalidad de solicitar un billete de avión e introducir la información necesaria.
2. Si es un Empleado:
   1. su solicitud está sujeta a la aprobación de su Gestor
   2. El directivo puede hacer preguntas al empleado para que se aclare, y el empleado puede renunciar al vuelo si es necesario.
   3. Una vez obtenido el visto bueno, el expediente pasa a la oficina de viajes
3. Si es un Director, el expediente va directamente a la Oficina de Viajes
4. La oficina viajes reserva el billete y transmite el expediente al solicitante.
5. El solicitante puede pedir un cambio de vuelo directamente en la Oficina de Viajes dentro de los tres días anteriores a la salida.
6. El solicitante, una vez obtenido, podrá:
   1. dar indicios de haber realizado el vuelo
   2. Renunciar al viaje
   3. Si transcurre más de una semana después de la fecha de salida, el billete se "renuncia" automáticamente.

Comencemos por definir los objetos básicos de este proyecto. Es decir, los **Roles**, los **Business Objects** y los **WorkItem**, y también tendremos que crear los datos simulados para el **SandBox** para los **Usuarios** y los distintos **BusinessObjects (BO)**.

### Los pasos necesarios

Para realizar el proceso descrito, tendremos que llevar a cabo los pasos que se ilustran en la tabla. Para cada paso, indicamos la figura profesional más adecuada entre **Modelizador** y **Desarrollador**.

|  |  |  |
| --- | --- | --- |
| **Objeto** | **Modelador** | **Desarrollador** |
| Proyecto y flujo de trabajo | Badge Tick1 con riempimento a tinta unita |  |
| Funciones | Badge Tick1 con riempimento a tinta unita |  |
| Objetos empresariales | Badge Tick1 con riempimento a tinta unita |  |
| Elemento de trabajo | Badge Tick1 con riempimento a tinta unita  Atributos básicos | Badge Tick1 con riempimento a tinta unita  Atributos avanzados |
| Datos SandBox | Badge Tick1 con riempimento a tinta unita | Badge Tick1 con riempimento a tinta unita |
| Proceso (etapas y funciones) | Badge Tick1 con riempimento a tinta unita  Definición | Badge Tick1 con riempimento a tinta unita  Código |
| Formularios | Badge Tick1 con riempimento a tinta unita  Definición básica | Badge Tick1 con riempimento a tinta unita  Diseños, búsquedas, validación |
| CodeLibs | Badge Croce con riempimento a tinta unita | Badge Tick1 con riempimento a tinta unita |
| Vista personalizada | Badge Croce con riempimento a tinta unita | Badge Tick1 con riempimento a tinta unita |

En los capítulos siguientes entraremos en detalle en cada uno de los pasos enumerados, siempre procurando describir primero el ejemplo práctico del proyecto "Vuelo" y luego los aspectos de la aplicación.

## Thumbnail Image Crea un'immagine che rappresenti un workflow molto semplice. Il workflow dovrebbe includere fasi chiare e ben definite come 'Inizio', 'Verifica Preliminare', 'Approvazione', 'Ricerca Volo', 'Presentazione Opzioni', 'Prenotazione', 'Preparazione al Viaggio', 'Esecuzione del Viaggio', 'Chiusura', e 'Analisi Post-viaggio'. Utilizza icone e grafici per rappresentare ogni fase e azione, con frecce che mostrano il flusso da una fase all'altra. Usa colori chiari e uno stile minimalista per rendere l'immagine facile da comprendere. Includi anche i ruoli principali come 'Richiedente', 'Approvatore', 'Ufficio Viaggi', e 'Direttore' in modo che siano chiaramente identificabili.Proyecto y flujo de trabajo

La creación de un Proyecto es el primer paso para crear la estructura inicial de la solución. Una vez creado, podemos pasar a definir uno o varios Flujos de Trabajo que compondrán la solución.

Por lo tanto, primero crearemos la **Vuelo de** Proyecto y luego la **Vuelo** de Flujo de Trabajo.

### ****Modelización de un proyecto****

**Immagine che contiene Personaggio immaginario, Cartoni animati, illustrazione, Animazione

Il contenuto generato dall'IA potrebbe non essere corretto.**

|  |  |  |
| --- | --- | --- |
| **Paso** | **Descripción** | **Ejemplo** |
| **1** | **Seleccione la extensión YS en la barra de actividades y haga clic en "Nuevo proyecto".**  **Se le pedirá que indique una carpeta padre donde crear la nueva carpeta de Proyecto**  **A continuación, introduzca el nombre del nuevo proyecto que llamaremos Vuelo**  **Al mismo tiempo, se creará una carpeta con el mismo nombre bajo la carpeta seleccionada** |  |
| **2** | **Localiza el árbol de YUBIQ STUDIO en la Barra Lateral, que debería aparecer como en la figura.**  **Una de las primeras ramas es Workflows**  **Tenga en cuenta que no tiene un flujo de trabajo definido** |  |

### ****Modelización de un flujo de trabajo****

**Immagine che contiene Personaggio immaginario, Cartoni animati, illustrazione, Animazione

Il contenuto generato dall'IA potrebbe non essere corretto.**

|  |  |  |
| --- | --- | --- |
| **Paso** | **Descripción** | **Ejemplo** |
| **1** | **Localice la rama Workflows y haga clic en el icono**  **Introduzca el nombre del flujo de trabajo sin espacios Vuelo** |  |
| **2** | **YS generará la estructura inicial del flujo de trabajo con:**   * **La acción de crear** * **Una primera etapa** * **La fase final de Archivo** * **Una forma minimalista vacía**   **Este aspecto permite la ejecución casi inmediata del SandBox** |  |
| **4** | **Simultáneamente a la generación del código necesario, también se genera el diseño del proceso.**  **Al hacer clic en el icono de la rama de flujo de trabajo recién creada, se muestra lo siguiente** |  |

Continuaremos con la definición del proceso después de crear los principales objetos básicos.

## Thumbnail Image Crea un'immagine che rappresenta i diversi ruoli utente nella metodologia STAR di Yubiq Studio. Includi i ruoli di Richiedente, Approvatore, Ufficio Viaggi e Direttore. Utilizza icone o figure stilizzate per ogni ruolo, con colori distinti per differenziarli. Mostra una struttura gerarchica o un flusso di lavoro che evidenzi come i ruoli interagiscono tra loro. Lo stile dell'immagine dovrebbe essere professionale e moderno, con un design pulito e chiaro. Utilizza font leggibili e un layout che faciliti la comprensione visiva dei ruoli e delle loro responsabilità.Funciones

Describimos los Roles que deben tener los usuarios distinguiendo entre estáticos y dinámicos. Recordemos que por rol estático indicamos un grupo fijo de usuarios mientras que un usuario obtiene un rol dinámico en función de las reglas del proceso.

### Funciones estáticas

Los roles estáticos se asignan a un grupo fijo de usuarios dentro de un sistema o solución. Estos roles no cambian en función de las acciones o circunstancias del usuario, sino que permanecen constantes a lo largo del tiempo. Por ejemplo, dentro de una solución empresarial, podríamos tener dos roles estáticos principales:

* Empleado: Todos los empleados de la empresa pertenecen a este rol. Estos usuarios tienen acceso a determinados recursos y funcionalidades definidos por su posición de empleado.
* Director: Los directores de empresa pertenecen a este rol. Tienen mayores privilegios y acceso que los empleados, en consonancia con sus responsabilidades de gestión.

Además, los usuarios de la oficina de viajes corporativos pueden pertenecer al rol estático **TravelDept**, que les otorga acceso específico para la gestión de viajes corporativos.

### Funciones dinámicas

Los roles dinámicos, en cambio, se asignan a los usuarios en función de determinadas condiciones o acciones dentro del proceso. Estos roles pueden cambiar y no son fijos, en función de las necesidades operativas. Por ejemplo:

* Viajero: Cuando un usuario realiza una nueva solicitud de billete, adquiere automáticamente el rol dinámico de Viajero. Este rol permite al usuario acceder a funcionalidades específicas relacionadas con la gestión de viajes.
* **Gestor**: Cuando un usuario adquiere el rol de Viajero, se identifica a su gestor directo, que adquiere el rol dinámico de Gestor. Para implementar esta lógica, la definición del usuario se amplía con el atributo ManagedBy, que permite identificar al gestor.

**Immagine che contiene testo, schermata, Carattere, design

Il contenuto generato dall'IA potrebbe non essere corretto.**

### **Modelización de roles estáticos y dinámicos**

**Immagine che contiene Personaggio immaginario, Cartoni animati, illustrazione, Animazione

Il contenuto generato dall'IA potrebbe non essere corretto.**

|  |  |  |
| --- | --- | --- |
| **Paso** | **Descripción** | **Ejemplo** |
| **1** | **Vaya al árbol YS y seleccione la rama 'Roles'. Aparecerán las ramas Estática y Dinámica. Seleccione la rama que le interese** | **Immagine che contiene testo, schermata, Carattere, Elementi grafici  Il contenuto generato dall'IA potrebbe non essere corretto.** |
| **2** | **Seleccione la subrama que le interese. Por ejemplo,**  **A continuación, haga clic en el icono y en la barra superior se le pedirá el nombre que desea darle**  **El procedimiento para los roles dinámicos es el mismo** | **Immagine che contiene testo, schermata, Carattere  Il contenuto generato dall'IA potrebbe non essere corretto.** |
| **4** | **Al pasar el ratón por encima de cada una de las funciones, aparecen cuatro iconos diferentes.**  **que permiten respectivamente**   * **Mostrar todas las referencias de código a ese elemento** * **Muestra la parte del código relacionada con la definición del elemento** * **Permite editar el nombre** * **Supresión de elementos (previa confirmación)** | **Immagine che contiene testo, Carattere, schermata  Il contenuto generato dall'IA potrebbe non essere corretto.**  **Immagine che contiene testo, schermata, Carattere  Il contenuto generato dall'IA potrebbe non essere corretto.**  **Immagine che contiene testo, schermata, Carattere  Il contenuto generato dall'IA potrebbe non essere corretto.**  **Immagine che contiene testo, Carattere, schermata  Il contenuto generato dall'IA potrebbe non essere corretto.** |

## Thumbnail Image Crea un'immagine di dipendenti in piedi che guardano in camera con lo sfondo di un ufficio open space. L'ufficio dovrebbe avere un design moderno con ampie finestre, scrivanie ordinate e attrezzature tecnologiche. I dipendenti devono essere vestiti in abbigliamento professionale, mostrando diversità in termini di età, genere e etnia. L'atmosfera dovrebbe essere luminosa e accogliente, con colori neutri e dettagli in verde per rappresentare un ambiente di lavoro produttivo e collaborativo.Datos SandBox: Usuarios

Para poder ejecutar la SandBox localmente y así poder probar el flujo, es necesario crear los datos de ejemplo en forma de archivo Json.

Immagine che contiene testo, schermata, menu

Il contenuto generato dall'IA potrebbe non essere corretto.El primer archivo a editar es el **users.json** cuyo contenido debe ser consistente con los Roles estáticos definidos y los atributos que debe tener un usuario para que la lógica del flujo de trabajo funcione correctamente.

En el lateral se ve un ejemplo de elementos.

Llaves estándar

* UserName**:** Identificador único de usuario
* FullName**:** Nombre del usuario que aparece en la interfaz
* Roles: Roles suplantados por el usuario separados por coma
* CompanyName: Nombre de la empresa que aparece en la interfaz;
* **Email**: Email del usuario que está siendo utilizado por el código.

Llaves personalizadas:

Bajo la clave **Profile**, se pueden añadir otros atributos que luego se pueden leer en tiempo de ejecución. Para nuestro ejemplo, crearemos la clave **ManagedBy** con el UserName del usuario responsable.

## Thumbnail Image Crea un'immagine che rappresenti tabelle di dati di un database relazionate fra loro. Mostra diverse tabelle con colonne e righe, con linee che collegano le chiavi primarie e le chiavi esterne tra le tabelle. Utilizza uno stile professionale e pulito, con colori neutri come il blu e il grigio. Includi etichette chiare per ogni tabella e colonna, e mostra esempi di dati realistici. L'immagine dovrebbe trasmettere un senso di organizzazione e struttura.Objeto de negocio

Como su nombre indica, los objetos de negocio (BO) representan entidades del mundo real utilizadas en la gestión de datos y los procesos empresariales. Estas entidades pueden ser clientes, productos, pedidos y muchas otras figuras esenciales para el funcionamiento de una empresa.

Los Objetos de Negocio son modelos de datos que definen las características y propiedades de las entidades de negocio. Cada BO tiene un conjunto de atributos que describen sus propiedades y que pueden variar en función de la naturaleza de la entidad. Por ejemplo, un BO de Cliente puede incluir atributos como nombre, dirección, número de teléfono y correo electrónico.

### Ejemplos de objetos de negocio

Cliente: Este BO puede tener atributos como:

* ID de cliente (único)
* Nombre del cliente
* Dirección
* Número de teléfono
* Correo electrónico

Producto: Este BO puede incluir:

* ID del producto (único)
* Nombre del producto
* Categoría
* Precio
* Stock disponible

Orden: Los atributos de este BO podrían ser:

* ID del pedido (único)
* Identificación del cliente
* Identificación del producto
* Fecha del pedido
* Cantidad
* Estado del pedido

Para el flujo de trabajo de Vuelo, identificamos tres BO diferentes:

Compañía aérea: Directorio de compañías aéreas. Se utiliza en los formularios de elección de compañía aérea. El WorkItem hace referencia a una instancia del mismo

|  |  |  |  |
| --- | --- | --- | --- |
| Línea aérea | | | |
| ID | Int | 1,2,3,4,5 |
| Código | Cadena | "AA", "BA", "DL" |
| Nombre | Cadena | American Airlines", "British Airlines", "Delta". |

Aeropuerto: Lista de aeropuertos. Se utiliza en los formularios para seleccionar el aeropuerto de salida y de regreso. El elemento de trabajo hace referencia a dos instancias de esta lista

|  |  |  |
| --- | --- | --- |
| Aeropuerto | | |
| ID | Int | 1,2,3,4,5 |
| Código | Cadena | "CTA", "FCO", "LIN" |
| Nombre | Cadena | "Catania Fontanarossa", "Roma Fiumicino", "Milán Linate". |

**CostCenter: Lista de CostCentres para la imputación de costes de billetes. Cada usuario pertenece a un CostCenter. El WorkItem hace referencia a una instancia del mismo.**

|  |  |  |
| --- | --- | --- |
| Centro de costes | | |
| ID | Int | 1,2,3,4,5 |
| Código | Cadena | "HR", "MK", "FN" |
| Nombre | Cadena | "Recursos Humanos", "Marketing", "Finanzas". |

### ****Modelado de objetos empresariales****

**Immagine che contiene Personaggio immaginario, Cartoni animati, illustrazione, Animazione

Il contenuto generato dall'IA potrebbe non essere corretto.**

|  |  |  |
| --- | --- | --- |
| **Paso** | **Descripción** | **Ejemplo** |
| **1** | **Vaya al árbol YS y seleccione la rama 'Business Objects'.** |  |
| **2** | **A continuación, haga clic en el icono y en la barra superior se le pedirá el nombre que desea darle.**  **Introduzca el nombre Aerolínea**  **El nuevo BO se creará automáticamente, lo que incluye:**   * **Una lista de atributos donde el principal es ID que nos da unicidad** * **Uno o varios Formularios que representan a la BO individual** |  |
| **4** | **Al pasar el ratón por encima del nombre del BO, aparecen cuatro iconos diferentes.**  **que permiten respectivamente**   * **Mostrar todas las referencias de código a ese elemento** * **Inserta un nuevo atributo WorkItem** * **Permite editar el nombre** * **Supresión de elementos (previa confirmación)** |  |
| **5** | **Creemos ahora los atributos restantes de BO Airline. A saber, Nombre y Código de tipo cadena.**  **Haciendo clic en el icono aparece una lista de todos los tipos posibles de atributos que se pueden crear.**  **Los veremos más adelante cuando hablemos del WorkItem. Limítense ahora a elegir el tipo "cadena** | Immagine che contiene testo, schermata, software, Software multimediale  Il contenuto generato dall'IA potrebbe non essere corretto. |
| **6** | **Una vez completada la tarea, el resultado debería ser como el de la figura.**  **Tenga en cuenta que si está en modo Desarrollador, el código C# autogenerado también aparece en el lateral.**  **¡Manipular con cuidado!** |  |
| **7** | **Completamos el negocio creando también BO Airport y CostCenter** |  |

### **Immagine che contiene testo, schermata, software Il contenuto generato dall'IA potrebbe non essere corretto.**Codificación en C# de Business Objects

**Una vez definido el Objeto de Negocio, se puede pasar a la fase de codificación en C# para manejar aspectos más técnicos de su funcionamiento.**

**A un lado se muestra un BO recién generado. En la primera línea, identifica la decoración DbBo, que indica que el BO es de tipo base de datos. YS actualmente sólo soporta este tipo de BO, mientras que otros se añadirán más adelante (Web Service, InMemory, Custom)**

**Los posibles atributos de DbBo son los siguientes**

[DbBo(DescriptionProperties= new string[] { "Código", "Nombre" },ConnectionName= "ExternalDB", TableName= "LÍNEA AÉREA", IdPropName= "AL\_ID")]

**Immagine che contiene testo, schermata, Carattere, numero

Il contenuto generato dall'IA potrebbe non essere corretto.DescriptionProperties**: es del tipo array de cadenas e indica los distintos atributos que componen la parte descriptiva del elemento. Se suele utilizar en las listas desplegables de los formularios, proporcionando una descripción clara y concisa de los elementos seleccionables.

**ConnectionName**: es de tipo cadena y especifica el nombre de la conexión a la base de datos utilizada. Este campo es esencial para garantizar que el BO pueda acceder correctamente a los datos almacenados en la base de datos una vez en producción.

"dBConnections": [

{

"name": "YubikDB",

"connectionString": '"

},

{

"nombre": "ExternalDB",

"connectionString": " Servidor=localhost;Base de datos= ExternalDB;Id\_usuario=sa;Contraseña= tu\_contraseña;"

}

],

**TableName**: es de tipo string y define el nombre de la tabla de la base de datos asociada al BO. A través de este atributo, el BO sabe dónde encontrar los datos que necesita manipular.

**IdPropName**: es de tipo cadena e identifica el campo único de la tabla de la base de datos. Este campo se utiliza para garantizar la unicidad de los elementos dentro de la tabla, facilitando las operaciones de búsqueda y gestión de datos.

De momento sólo vamos a definir el atributo 'DescripciónPropiedades' con los campos 'Código' y Nombre'.

## Objeto comercial (EnumValue)

[Banderas]

public enum Clase

{

Ecomomía= 1,

Negocios= 2,

Primero= 3,

Premium= 4,

}

También existe un tipo de BO más sencillo que consiste únicamente en una lista de valores fijos, como la clase de un vuelo (Turista, Business, ...). Esta estructura es muy cómoda de utilizar, ya que está vinculada a las listas desplegables de los formularios.

### ****Modelado de objetos empresariales (EnumValue)****

**Immagine che contiene Personaggio immaginario, Cartoni animati, illustrazione, Animazione

Il contenuto generato dall'IA potrebbe non essere corretto.**

|  |  |  |
| --- | --- | --- |
| **Paso** | **Descripción** | **Ejemplo** |
| **1** | **Vaya al árbol YS y seleccione la rama "Business Objects"🡪 "EnumValue"** |  |
| **2** | **A continuación, haga clic en el icono y en la barra superior se le pedirá el nombre que desea darle.**  **Introduzca el nombre Clase**  **El nuevo BO vacío de tipo EnumValue se creará automáticamente** |  |
| **4** | **Ahora introduzca la lista de valores de texto dándoles un número de identificación como en el ejemplo mostrado** |  |

## Immagine che contiene testo, Carattere, schermata, design Il contenuto generato dall'IA potrebbe non essere corretto.Datos SandBox: Objetos empresariales

Una vez definidos los diferentes BOs de la solución, es posible crear dentro de la sección SendboxData/BO los ficheros JSON pertinentes que persistirán los datos. Para ello, hay que seguir una serie de pasos bien definidos:

Immagine che contiene testo, schermata, menu, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.Cada uno de los ficheros JSON debe ser un array de objetos. Estos objetos deben tener todos los atributos definidos en el BO. Estos atributos representan propiedades y características específicas del BO, como el nombre, el tipo y otros detalles relevantes.

Además de los atributos específicos del BO, debe añadirse un atributo adicional denominado "BusinessObjectType". Este atributo tendrá como valor el nombre del BO relacionado. Esto es crucial porque permite a YS identificar fácilmente el tipo de BO al que pertenece cada objeto dentro del archivo JSON.

Los archivos JSON creados se utilizarán para la persistencia de datos durante las primeras fases de desarrollo sin tener que acceder a recursos externos como servidores de bases de datos.

### ****SandBoxData/Modelado de objetos empresariales****

**Immagine che contiene Personaggio immaginario, Cartoni animati, illustrazione, Animazione

Il contenuto generato dall'IA potrebbe non essere corretto.**

|  |  |  |
| --- | --- | --- |
| **Paso** | **Descripción** | **Ejemplo** |
| **1** | **Vaya al árbol YS y seleccione la rama "SandBoxData"🡪 "BO"** |  |
| **2** | **A continuación, haga clic en el icono y en la barra superior se le pedirá el nombre que desea darle.**  **Introduzca el nombre Aerolínea**  **El nuevo archivo de datos BO vacío se creará automáticamente** | **(Función aún no disponible)** |
| **4** | **Para la inserción de los elementos BO, se debe crear un array de objetos Json. Cada elemento contendrá todos los campos BO definidos como claves y "BusinessObjectType" debe introducirse como última clave. : "Aerolínea"** |  |

## Elemento de trabajo

El Workitem constituye una de las estructuras fundamentales de una solución, ya que incluye todas las propiedades relevantes relacionadas con la práctica. Cada Workitem representa una instancia de proceso que agrega datos y documentos relevantes, permitiendo una gestión eficiente e integrada de las actividades.

**Immagine che contiene testo, schermata, software, Software multimediale

Il contenuto generato dall'IA potrebbe non essere corretto.**Un elemento de trabajo consta de diferentes tipos de propiedades:

* **Tipos de base**: Campos estándar individuales
* String, Integer, Decimal, DateTime, DateOnly, TimeOnly, Boolean. Todos en la excepción nullable o non-nullable
* **Core Typed**: Campos de estructura del sistema
* Identificador del documento
* Atributo de soporte de un rol dinámico (DynamicRole)
* Identificador de un usuario (Usuario)
* **Business Objects**:
* Creación de instancias únicas de un BO específico (Aerolínea
* Instancias individuales de un Enum (Clase)
* Colecciones de instancias de un BO específico (iCollection< Airline>)

### ****Modelización del WorkItemImmagine che contiene Personaggio immaginario, Cartoni animati, illustrazione, Animazione Il contenuto generato dall'IA potrebbe non essere corretto.****

|  |  |  |
| --- | --- | --- |
| **Paso** | **Descripción** | **Ejemplo** |
| **1** | **Vaya al árbol YS y seleccione la rama 'WorkItem' del proyecto Flight** |  |
| **2** | **A continuación, haga clic en el icono y en la barra superior se le pedirá que seleccione el tipo de atributo que desea crear de entre los que aparecen en la lista. Por ejemplo cadena**  **Una vez seleccionado el tipo, se le pide que defina su nombre, que debe ser único.** |  |
| **4** | **Repita la operación para todas las propiedades que desee crear. Al final, debería obtener una lista como la de la figura** |  |

### **Immagine che contiene testo, schermata, software, Sistema operativo Il contenuto generato dall'IA potrebbe non essere corretto.**WorkItem c# codificación

**Una vez definido el Objeto de Negocio, se puede pasar a la fase de codificación en C# para manejar aspectos más técnicos de su funcionamiento.**

**En esencia, lo único que se necesita es especificar el tamaño de los caracteres de los campos de cadena.**

**En el ejemplo de al lado, establecimos la cadena 'RequestNote' a 1000 caracteres e inicializamos algunos campos.**

**También es posible inicializar los distintos campos del WorkItem con un valor inicial.**

## Mapa del proceso

A continuación se muestra una posible representación STAR del proceso. Se invita al alumno a modelar el mismo proceso con YS utilizando los mismos nombres de Etapa y Acción.

Para modelar el mapa de la forma más eficaz, te recomendamos abrir el árbol de Yubiq Studio de la forma que ves para que puedas arrastrar fácilmente al mapa los elementos que ves con los rectángulos blancos.

Si te sitúas en la esquina inferior derecha, encontrarás en la barra la opción de cambiar del modo "Modelador" al modo "Desarrollador".

Immagine che contiene schermata, testo, diagramma, Software per la grafica

Il contenuto generato dall'IA potrebbe non essere corretto.

A continuación encontrará un cuadro que le guiará en la asignación de diferentes funciones a las prácticas.

|  |  |  |  |
| --- | --- | --- | --- |
| Objeto | Funciones estáticas | Funciones dinámicas | Motivación |
| CreationAction: Solicitud de vuelo | Empleado, Director |  | Permite la creación de un nuevo WorkItem para los roles dados. Dado que cada usuario tiene al menos uno de los dos roles, cualquiera puede solicitar un ticket |
| Prácticas: MyRequests |  | Viajero | El rol de Viajero dinámico se asigna al WI al mismo tiempo que su con la acción anterior. por lo que indicamos que cada Viajero accederá a sus propias peticiones en el Stage.  Dado que las 4 acciones de esta etapa son todas ejecutables por aquellos que tienen acceso a esta etapa, no tenemos que especificar roles sino sólo indicar que la ejecución está permitida a aquellos que tienen la WI en la lista ToDo |
| Prácticas: OnHold  Prácticas: MyTakenflight |  | Viajero | En estas Etapas, el viajero accede a sus demandas. |
| Prácticas: SolicitudAprobación |  | Director | El rol dinámico de Gestor también se asigna al WI en el momento de su creación. Por lo tanto, cada Gestor sólo tiene acceso a esta Etapa para aprobar sus propias solicitudes. |
| Prácticas: TicketBooking | TravelDept |  | A esta etapa sólo pueden acceder los usuarios con el rol TravelDept |
|  |  |  |  |

Nota: A todas las etapas y acciones siempre accede la persona que tiene el rol estático del sistema 'Admin'.

### ****Modelización del proceso****

**Immagine che contiene Personaggio immaginario, Cartoni animati, illustrazione, Animazione

Il contenuto generato dall'IA potrebbe non essere corretto.**

|  |  |  |
| --- | --- | --- |
| **Paso** | **Descripción** | **Ejemplo** |
| **1** | **Vaya al árbol YS y seleccione la rama 'Workflows' y luego 'Flight'.** |  |
| **2** | **A continuación, haga clic en el icono para mostrar el mapa inicial y proceda a modelar** | Immagine che contiene schermata, testo, diagramma, Software per la grafica  Il contenuto generato dall'IA potrebbe non essere corretto. |
| **3** | **Para crear etapas, debe ir a Proceso, hacer clic en el icono e introducir un nombre de etapa único. A continuación, aparecerá en el Mapa** |  |
| **4** | **Las acciones se crean directamente en el mapa. Para ello, seleccione la etapa inicial con un clic y, a continuación, haga clic en la etapa final mientras mantiene pulsada la tecla "Mayúsculas".**  **A continuación, se le pedirá el nombre de la Acción y el tipo de Acción entre las disponibles.**  **Si se elige la acción Tipo de usuario, se pregunta al usuario por el formulario a mostrar de entre los disponibles.**  **Si todavía no ha creado el formulario deseado, le recomendamos que elija temporalmente el que está creado por defecto al crear el Flujo de Trabajo FlightDisplay** |  |
| **5** | **También es posible cambiar el destino de una Acción. Para ello, haga clic en una Acción para que la flecha de destino aparezca punteada y, a continuación, mientras mantiene pulsada la tecla "Mayúsculas", haga clic en la etapa de destino deseada.** |  |
| **6** | **Para añadir los roles deseados a las etapas y acciones, puede arrastrar los roles del árbol al mapa.**  **Seleccione la función "Director" y arrastre el elemento al escenario o a la acción.**  **A continuación, mantenga pulsada la tecla "Mayúsculas" y suéltela cuando haya llegado a la Etapa o Acción a la que desea asociarla.** |  |
| **7** | **Del mismo modo, es posible asociar formularios a etapas o acciones.** |  |

### Proceso de codificación c#

Una vez modelado el proceso y asignados adecuadamente los roles de usuario, pasamos a escribir el código en las diferentes etapas y acciones.

**Prácticas:**

Immagine che contiene testo, schermata, Carattere, linea

Il contenuto generato dall'IA potrebbe non essere corretto.Una etapa intercepta dos eventos específicos en los que se debe introducir el código:

**OnEnter**: Cuando el elemento de trabajo llega al escenario desde una acción conectada a él.

Immagine che contiene testo, Carattere, schermata

Il contenuto generato dall'IA potrebbe non essere corretto.**OnExit**: Cuando el elemento de trabajo se reinicia desde el escenario con una acción adjunta.

Estos dos eventos deben crearse explícitamente pulsando '+' si se desea insertar código c#. Al pulsar '+' se crea el método en la clase correspondiente donde se va a insertar el código, asistido por AI.

**Acción:**

Una Acción también intercepta eventos, pero éstos también dependen del tipo de Acción.

**Acción del usuario:**

**Immagine che contiene testo, Carattere, schermata

Il contenuto generato dall'IA potrebbe non essere corretto.**Una Acción de Usuario intercepta tres eventos específicos donde insertar el código:

Immagine che contiene testo, Carattere, schermata

Il contenuto generato dall'IA potrebbe non essere corretto.**CanView**: YS llama a este método antes de mostrar la Acción entre las disponibles para el usuario con el fin de ejecutar el código que puede devolver True o False y, a continuación, mostrar la Acción en consecuencia.

**OnPrepare**: YS llama a este método para preparar el Formulario asignado a la Acción.

**Immagine che contiene testo, Carattere, schermata, Elementi grafici

Il contenuto generato dall'IA potrebbe non essere corretto.OnExecute**: YS llama a este método cuando el usuario ha dado la Confirmación del Formulario para ejecutar el código correspondiente

**Acción condicional:**

### Formularios

Immagine che contiene testo, schermata, schermo, numero

Descrizione generata automaticamente

**Flightrequest**: Formulario para solicitar y modificar los datos del billete. En modo ReadOnly contiene la información de la solicitud. Deberá solicitar el documento de identidad del solicitante

Immagine che contiene testo, schermata, schermo, numero

Descrizione generata automaticamente

**FlightReservation**: Formulario para introducir y editar los datos del billete. En modo ReadOnly muestra la información del billete. Deberá solicitar una foto del billete

Immagine che contiene testo, schermata, Rettangolo, schermo

Descrizione generata automaticamente

**VPApprouval**: Formulario de respuesta de aprobación del Vicepresidente

Immagine che contiene testo, schermata, schermo, numero

Descrizione generata automaticamente

**Flightrequest**: Formulario para solicitar y modificar los datos del billete. En modo ReadOnly contiene la información de la solicitud. Deberá solicitar el documento de identidad del solicitante

Immagine che contiene testo, schermata, schermo, numero

Descrizione generata automaticamente

**FlightReservation**: Formulario para introducir y editar los datos del billete. En modo ReadOnly muestra la información del billete. Deberá solicitar una foto del billete

Immagine che contiene testo, schermata, Rettangolo, schermo

Descrizione generata automaticamente

**VPApprouval**: Formulario de respuesta de aprobación del Vicepresidente