



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

– **TELECOM** ESCUELA  
TÉCNICA **VLC** SUPERIOR  
DE INGENIERÍA DE  
TELECOMUNICACIÓN

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

School of Telecommunications Engineering

Study of the available options in robotic process automation  
and development of a use case

End of Degree Project

Bachelor's Degree in Telecommunication Technologies and  
Services Engineering

AUTHOR: Del Amo Alcañiz, Raúl

Tutor: González Ladrón de Guevara, Fernando Raimundo

ACADEMIC YEAR: 2022/2023



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

— **TELECOM** ESCUELA  
TÉCNICA **VLC** SUPERIOR  
DE INGENIERÍA DE  
TELECOMUNICACIÓN

Escuela Técnica Superior de Ingeniería de Telecomunicación  
Universitat Politècnica de València  
Edificio 4D. Camino de Vera, s/n, 46022 Valencia  
Tel. +34 96 387 71 90, ext. 77190  
[www.etsit.upv.es](http://www.etsit.upv.es)

**VLC/**  
**CAMPUS**  
VALENCIA, INTERNATIONAL  
CAMPUS OF EXCELLENCE



## Resumen

La automatización robótica de procesos (RPA, *Robotic Process Automation*) es una tecnología que permite reducir la carga de trabajo humana, llevando a cabo tareas repetitivas basadas en reglas de negocio específicas.

Esto se realiza secuencialmente, en base a navegación por diferentes pantallas, así como mediante la integración de varias tecnologías.

El interés en implementar esta tecnología surge con la necesidad de digitalización de procesos de negocio, así como la actualización a sistemas autónomos de las tareas más rígidas (aquellas que presentan un flujo de negocio más definido y cuya toma de decisiones requiere menor subjetividad) y que menos valor aporten al ser una persona la responsable de su realización.

Un primer objetivo del proyecto es analizar las diferentes tecnologías RPA disponibles en el mercado, encontrando sus características claves y realizando una comparativa sobre la que se pueda tomar una decisión de la herramienta a elegir según las capacidades y recursos de la empresa o individuo interesado. En base al análisis realizado, se busca hacer un desarrollo en UiPath, una de las herramientas líder en el desarrollo RPA, en el que podamos recoger información de una web de dominio público así como analizar el ciclo de vida de un proyecto.

Durante el proyecto se planteará el desarrollo siguiendo una metodología Ágil, que permite aportar la flexibilidad necesaria para este tipo de desarrollo, manteniendo al mismo tiempo la rapidez en la gestión y entrega.

## Resum

L'automatització robòtica de processos (RPA, *Robotic Process Automation*) és una tecnologia que permet reduir la càrrega de treball humà, duent a terme tasques repetitives basades en regles de negoci específiques.

Açò es realitza seqüencialment, mitjançant navegació per diferents pantalles així com la integració de diferents tecnologies.

L'interés a implementar aquesta tecnologia sorgeix amb la necessitat de digitalització de processos de negoci, així com l'actualització a sistemes autònoms de les tasques més rígides (aquelles que presenten un flux de negoci més definit i la seua presa de decisions requereix una menor subjectivitat) i que aporten menys valor al ser una persona la responsable de la seua realització.

Un primer objectiu del projecte és analitzar les diferents tecnologies RPA disponibles en el mercat, trobant les seues característiques claus i realitzant una comparativa sobre la qual es puga prendre una decisió de l'eina a triar segons les capacitats i recursos de l'empresa o individu interessat. Sobre la base de l'anàlisi realitzat, es busca fer un desenvolupament en UiPath, una de les eines líder en el desenvolupament RPA, en el qual puguem recollir informació d'una web de domini públic així com analitzar el cicle de vida d'un projecte.

Durant el projecte es planteja el desenvolupament seguint una metodologia Ágil, que permet aportar la flexibilitat necessària per a aquest tipus de desenvolupament, mantenint al mateix temps la rapidesa en la gestió i entrega.

## Abstract

Robotic Process Automation (RPA) is a technology that allows reducing the human workload, carrying out repetitive tasks based on specific business rules.



This is done sequentially, based on navigation through different screens as well as through the integration of different technologies.

The interest in implementing this technology arises from the need to digitize business processes, as well as updating the most rigid tasks (those with a more defined business flow and whose decision making requires less subjectivity) and the ones that provide less value when a person is responsible for their implementation.

A first objective of the project is to analyze the different RPA technologies available on the market, finding its key characteristics and making a comparison on which a decision can be made on the tool to choose according to the capabilities and resources of the company or interested individual. Based on the analysis carried out, it seeks to develop in UiPath, one of the leader tools in RPA development, a process in which we can collect public information from a domain website as well as analyze the life cycle of a project.

During the project, the development will follow an Agile methodology, which allows the necessary flexibility for this type of development, maintaining at the same time the speed in management and delivery.



## Índice

Capítulo 1.	Introducción .....	1
Capítulo 2.	Objetivos .....	2
Capítulo 3.	Metodología .....	3
Capítulo 4.	Desarrollo y resultados del trabajo .....	4
4.1	RPA .....	4
4.2	Estudio de herramientas en el mercado .....	5
4.2.1	UiPath .....	7
4.2.2	Automation Anywhere .....	9
4.2.3	SS&C BluePrism .....	11
4.2.4	Selenium .....	12
4.3	Comparativa entre las herramientas .....	13
4.3.1	Plataformas de desarrollo .....	13
4.3.2	Plataforma de control .....	16
4.3.3	Conclusiones .....	17
4.4	Caso de Uso .....	18
4.4.1	Definición del caso de uso .....	18
4.4.2	Desarrollo del caso de uso .....	19
Capítulo 5.	Conclusiones, limitaciones y propuesta de trabajo futuro .....	21
Capítulo 6.	Bibliografía .....	22
Capítulo 7.	Anexos .....	24
7.1	Anexo I – Desarrollo documentado .....	24
7.1.1	Initialization .....	24
7.1.2	Get Transaction Data .....	27
7.1.3	Process Transaction .....	29
7.1.4	End Process .....	32
7.2	Anexo II – Pliego del proyecto RPA .....	34
1.	Objetivos del proyecto .....	35
2.	Proceso a automatizar .....	35
2.1	Aplicaciones involucradas .....	35
2.1.1	Chrome .....	35
2.1.2	Access .....	35
2.1.3	Google .....	36
2.2	Ficheros utilizados .....	36
2.2.1	Fichero config interno .....	36



2.2.2	Fichero config en línea .....	36
2.2.3	Ficheros de mail con formato HTML.....	36
2.2.4	Fichero Access .....	36
2.3	Flujo del proceso .....	36
2.3.1	Initialization .....	37
2.3.2	Get Transaction Data.....	38
2.3.3	Process.....	38
2.3.4	End Process .....	39
3.	Alcance del proyecto.....	40
4.	Recursos .....	40
5.	Métricas de éxito .....	40
6.	Plan de implementación .....	41
7.	Plan de mantenimiento y soporte .....	41



## Índice de Figuras

Figura 1.- Ciclo en la clasificación de documentos.....	4
Figura 2.- Robots Atendidos vs Desatendidos .....	5
Figura 3.- Cuadrante Mágico RPA 2021, Gartner.....	6
Figura 4.- Cuadrante Mágico RPA 2022, Gartner.....	7
Figura 5.- Logo de UiPath.....	7
Figura 6.- Logo de Automation Anywhere .....	9
Figura 7.- Logo de SS&C BluePrism.....	11
Figura 8.- Logo de Selenium.....	12
Figura 9.- UiPath Studio.....	14
Figura 10.- UiPath Studio - Menú de depuración .....	14
Figura 11.- Automation 360 - Panel Automatización .....	14
Figura 12.- Automation 360 - Taskbot.....	15
Figura 13.- Automation 360 - Process .....	15
Figura 14.- BluePrism - Objetos .....	15
Figura 15.- BluePrism - Procesos.....	16
Figura 16.- UiPath Orchestrator - Folder .....	16
Figura 17.- UiPath Orchestrator - Tenant.....	16
Figura 18.- BluePrism - Control.....	17
Figura 19.- BluePrism - Analíticas.....	17
Figura 20.- Automation Anywhere - Gestión.....	17



## Capítulo 1. Introducción

Con la digitalización y evolución de los flujos de trabajo en los negocios, las empresas han entrado en un periodo en el que una de las estrategias clave para su desarrollo se centra en la mejora de los procesos que desarrollan. Así pues, los objetivos más centrados únicamente en la producción y en la creación de nuevos procesos quedan en un segundo plano para focalizarnos en la optimización de aquellos ya existentes.

Ya podíamos ver un interés en la mejora de los procesos y el apoyo a los trabajadores con la introducción de las herramientas de gestión y el desarrollo de diversas metodologías, sin embargo las tecnologías introducidas recientemente no solo se centran en el apoyo a los trabajadores sino que dan un paso más allá, extrayendo la dependencia de los procesos para con las personas.

En esta línea podemos encontrar diferentes tecnologías emergentes, como son la inteligencia artificial o el Internet de las cosas. Entre estas opciones encontramos la tecnología sobre la que nos centraremos en este documento: la automatización robótica de procesos.

La automatización robótica de procesos, también conocido como RPA [1] (*Robotic Process Automation*), es una vertiente de la automatización de procesos de negocio que se centra en la automatización de tareas repetitivas que previamente habrían sido realizadas por los trabajadores. Esto se realiza mediante robots de software (*bots*) o agentes de inteligencia artificial. Así pues, estos *bots* se encargarían de interactuar con diferentes aplicaciones, sitios web o sistemas siguiendo unas reglas definidas en el desarrollo de estos procesos.

El objetivo de esta tecnología no es reemplazar a los trabajadores, sino liberar a estos de las tareas más repetitivas y que menos valor aportan. Pese a esta afirmación es cierto que ha habido reticencia a la entrada de estas tecnologías por parte de trabajadores que consideraban sus trabajos en riesgo, sin embargo la aparición de nuevas tareas debería resultar en la creación de nuevos puestos de trabajos y el aumento del empleo, debido al efecto de restablecimiento [2].

Este documento pretende aportar una visión de que es el RPA, realizar una comparativa de los productos más utilizados para su desarrollo y finalmente documentar un caso de uso durante el ciclo de vida del mismo.



## Capítulo 2. Objetivos

Este proyecto se centrará en dos partes bien diferenciadas.

Inicialmente, el objetivo es presentar el RPA y ver el impacto que está teniendo en diferentes sectores. Así pues, se analizarán las diferentes tecnologías disponibles en el mercado, se abordará la evolución que han sufrido durante los últimos años y cuales son las opciones más conocidas. También veremos algunas soluciones gratuitas alternativas.

Al considerar las herramientas de pago se hará una comparativa para identificar los beneficios que se pueden obtener mediante ellas, el enfoque por el que han optado y las decisiones que se han tomado durante sus implementaciones.

Finalmente se realizará un caso de uso desarrollado en UiPath (una de las tecnologías mencionadas en el análisis) en la que se recogerá información de una web de dominio público y se realizarán comunicaciones en base a la información recogida, manteniendo un control sobre dichos datos. En la documentación realizada sobre el proyecto veremos las diferentes partes de este, la evolución del desarrollo y sus resultados.

Con este desarrollo, pretendemos mejorar la eficiencia al eliminar una de las tareas diarias de un trabajador. Dicha tarea repetitiva aportaría poco valor al ser realizada por el usuario y así este podría dedicar su tiempo a la realización de tareas con un mayor nivel de complejidad y valor aportado.

Este TFG pretende aportar una visión general sobre el RPA, una de las tecnologías que está transformando los negocios en la actualidad y mostrar un ejemplo de su implementación en los procesos de negocio de las empresas.



### Capítulo 3. Metodología

Durante el desarrollo de la automatización se seguirá una metodología Ágil [3], que permite aportar la flexibilidad necesaria para este tipo de desarrollos, mientras que mantenemos la rapidez en la gestión y entrega.

El marco de trabajo ágil utilizado será Scrum [4], en el que contaremos con periodos de trabajo cortos sobre los que se podrán tomar decisiones rápidas en el ciclo de vida del desarrollo, algo muy beneficioso en proyectos RPA. Estos ciclos, también llamados sprints, permitirán la involucración de todas las partes, facilitando la comunicación y colaboración.

Otra parte importante del uso de Scrum es la capacidad de adaptación y ajuste de la planificación. Esto es especialmente interesante en esta tecnología, puesto que a pesar de la minuciosidad con la que se pueda hacer el estudio inicial sobre el proceso a automatizar, es muy fácil encontrar nuevas variaciones o ajustes a medida que se comprende mejor el proceso y encontramos diferentes excepciones al flujo principal, así como comportamientos no habituales en las aplicaciones.

En cuanto a los recursos se trabajará con tres herramientas RPA. Se realizará una comparativa de las versiones de prueba de estas herramientas:

- UiPath [5].
- Automation Anywhere [6].
- BluePrism [7].

Estas serán revisadas en los apartados 4.2.1 a 4.2.3, en los que se hará una breve introducción de su historia y se revisarán sus productos principales, para posteriormente realizar una comparativa sobre estos en el apartado 4.3.

Para el proyecto se pretende hacer un desarrollo que no suponga costes adicionales, así pues se utilizará una licencia Microsoft 365 gratuita para estudiantes, programas y herramientas online sin coste añadido como son Google Chrome y Google Sheets, y el programa sobre el que se desarrollará, la versión Community de UiPath.

Se puede leer más al respecto las herramientas utilizadas en el segundo capítulo del anexo II, el pliego del proyecto RPA..

## Capítulo 4. Desarrollo y resultados del trabajo

Este capítulo pretende introducir el concepto del RPA, cómo funciona este tipo de tecnología, realizar una introducción de los productos disponibles en el mercado y hacer una comparativa entre ellos. Finalmente aplicar esta tecnología a un proceso de automatización de un caso de negocio.

Inicialmente hablaremos de qué es el RPA. En este punto indicaremos cuándo debería utilizarse este tipo de tecnología, los diferentes robots que se pueden emplear y qué beneficios conlleva.

Posteriormente se realizará un estudio de mercado entre las herramientas mejor valoradas en los últimos años (UiPath, Automation Anywhere y SS&CC BluePrism), iniciando con una breve explicación sobre su historia y sus productos principales, para finalizar mostrando las diferencias en la implementación de sus productos.

Finalmente se seguirá el ciclo de vida de un proyecto RPA realizado en UiPath para la descarga de licitaciones y envío de notificaciones.

### 4.1 RPA

Cuando hablamos de RPA nos referimos a la Automatización Robótica de Procesos, tecnología que automatiza tareas repetitivas y basadas en reglas dentro de las organizaciones mediante el uso de robots o *bots*. Este software se encarga de replicar las acciones humanas con sistemas digitales, como la navegación por interfaces, la introducción de información o su obtención [8], aunque no está limitada, pues se puede integrar con diferentes tecnologías que nos permiten crear desarrollos híbridos, aportando flexibilidad a su uso.

La elección de las tareas a automatizar se suele basar en aquellas actividades que consumen mucho tiempo, en las que su realización no aporta un valor significativo y existe un sistema muy claro en la toma de decisiones. Un buen ejemplo podría ser la gestión de la facturación en una empresa, especificando en un caso más concreto el acceso a una web bancaria, la descarga de cada uno de sus recibos y la posterior clasificación.

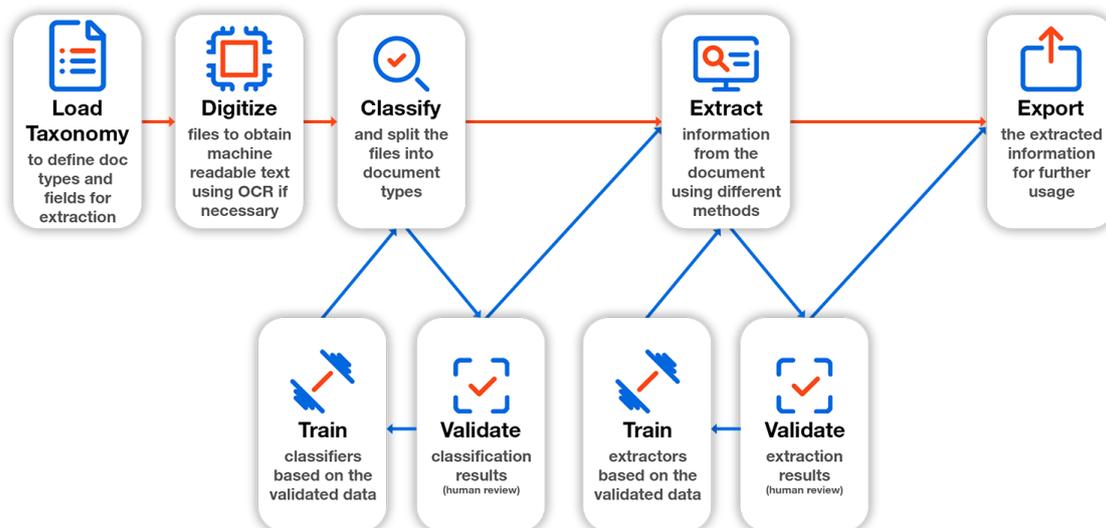


Figura 1.- Ciclo en la clasificación de documentos.

Otros ejemplos podrían ser la gestión de notificaciones de los clientes a través de correo electrónico y clasificación según temática, introducción de pedidos en aplicación interna en base a los datos obtenidos en aplicaciones de terceros o la gestión de pedidos.

Al implementar los procesos, las organizaciones pueden conseguir incrementar la eficiencia y la precisión en sus procesos, así como liberar a sus operarios de las tareas manuales más repetitivas [9].

El valor de esta tecnología recae en que, además de la posibilidad de trabajar con herramientas de terceros, no repercute en la infraestructura informática al trabajar con la interfaz de usuario, evitando el desarrollo de complejas integraciones de sistemas. Es decir, así como los desarrollos en diferentes tecnologías necesitan una integración con la aplicación, como puede ser una API, los robots trabajan sobre la pantalla, realizando lecturas y clicks, lo que nos permite no realizar modificaciones sobre las aplicaciones objetivo, pues se replica el modo de trabajo humano.

Cabe destacar que, aunque cuando se habla de automatización de las tareas nos planteamos una sustitución total de una tarea previamente realizada por una persona, este no siempre es el caso. De hecho podemos dividir los robots en dos tipos: **desatendidos** y **atendidos** [10].

Los robots desatendidos son los robots que se ejecutan en entornos virtuales de forma autónoma, sin la intervención humana. Su activación se realiza siguiendo un horario preestablecido o en base a señales externas y son más fáciles de controlar pues sus ejecuciones se realizan en un servidor controlado por lotes. Estos son beneficiosos al poder proporcionar un soporte las 24 horas del día, centrándose en tareas estructuradas, repetitivas, siguiendo reglas predefinidas y que cuentan con un gran volumen de trabajo.

Los robots atendidos son aquellos que trabajan junto a una persona en su puesto de trabajo, se activan por eventos de usuario como hacer click en una tecla y puede que necesiten intervención humana en varios puntos del proceso. Son beneficiosos en procesos que necesitan toma de decisiones o intervención humana, como pueden ser actividades de *front-office* y pueden mejorar el rendimiento individual de los trabajadores.

Cabe destacar que ambos robots no son excluyentes, y ambos pueden aportar un papel importante a la hora de realizar una implementación RPA.

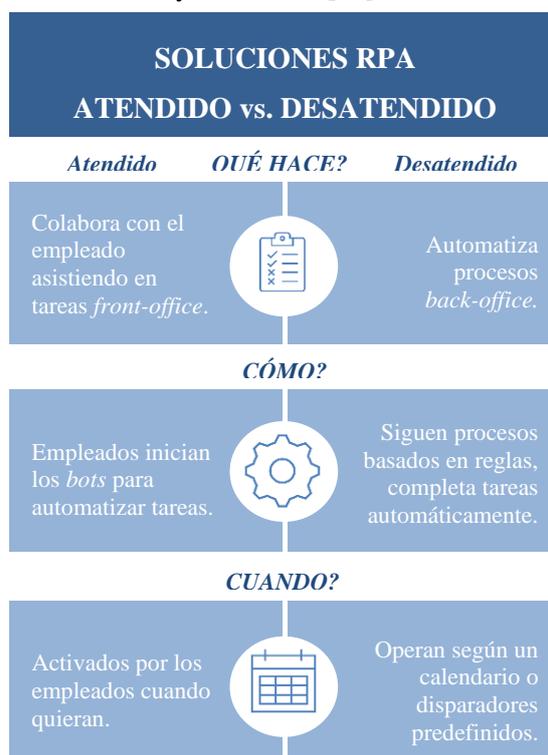


Figura 2.- Robots Atendidos vs Desatendidos

## 4.2 Estudio de herramientas en el mercado

Son muchas las herramientas que se han postulado como soluciones completas para el desarrollo de RPA. Así pues una de las preguntas a la que nos enfrentamos al plantear un desarrollo mediante esta tecnología es, qué herramienta utilizaremos.

Cabe destacar que para responder esta pregunta deberemos ser muy conscientes de los **requisitos** específicos de la automatización, así como del **entorno** en el que se implantará. Es importante también tener en cuenta cuales son los **objetivos** que se quieren alcanzar no solo a corto plazo, sino también a medio y largo plazo. Será en el apartado 4.3.3, en las conclusiones de la comparativa de las herramientas, donde podremos ver como se relacionan estas características para realizar una selección acertada entre las herramientas disponibles.

Debido a la gran cantidad de herramientas disponibles, nos centraremos en una comparativa entre las herramientas con una visión más completa y con una mayor capacidad para llevarla a cabo. Extraemos estas del análisis anual de Gartner de los años 2021 y 2022.

Para la aparición en estos cuadrantes, Gartner describe unas capacidades obligatorias [11] que estos softwares deben tener:

- Compatibilidad con el desarrollo de *scripts* con lenguajes de programación estándar.
- Capacidad de integración con otras aplicaciones.
- Posibilidad de ejecución atendida y desatendida.
- Capacidad de orquestación y administración (Usuarios, *scripts*, tiempos de ejecución, supervisión y seguridad entre otros).
- Capacidad para grabar una tarea y convertirla para su implementación.

Estos gráficos proporcionan orientación sobre el mercado de software de RPA, dividiendo en cuatro categorías a las empresas: Líderes, Competidores, Visionarios y Operadores Especializados [12].

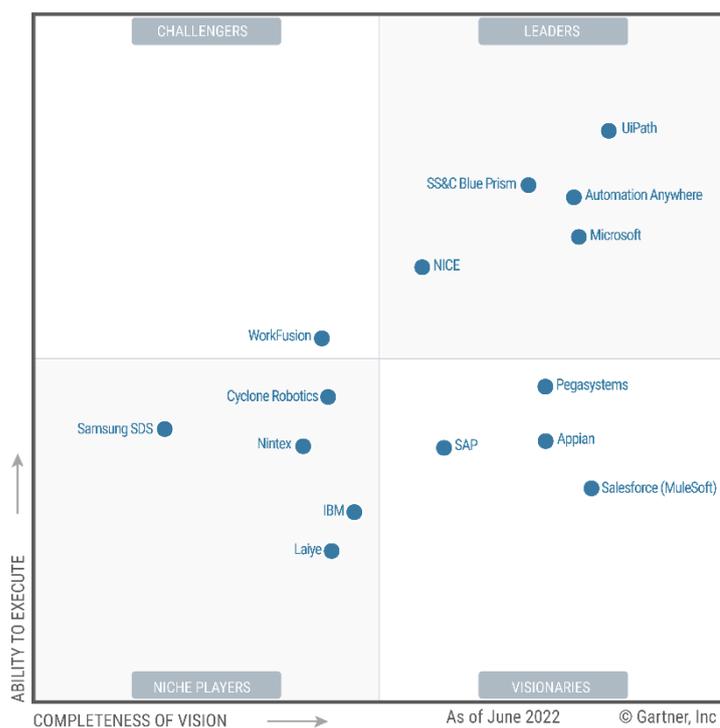
Los líderes son aquellos que ejecutan bien su visión actual y están bien posicionados para el futuro, los visionarios entienden el mercado o tienen una visión para cambiar sus reglas pero aún no han conseguido ejecutarlo, los competidores se desempeñan bien o dominan un segmento grande pero no demuestran una comprensión de la dirección que toma el mercado según los análisis de Gartner y los operadores especializados se centran con éxito en un segmento pequeño pero no innovan ni superan al resto.

En la figura 3 encontramos el cuadrante mágico de RPA para 2021, en el que podemos observar como UiPath se posiciona en cabeza de sus competidores, no muy alejado de Automation Anywhere mientras que, a pesar de estar entre los líderes, BluePrism y Microsoft quedan relegados a un segundo plano.



Source: Gartner (July 2021)

Figura 3.- Cuadrante Mágico RPA 2021, Gartner.



Source: Gartner (July 2022)

Figura 4.- Cuadrante Mágico RPA 2022, Gartner.

licencia o suscripción de 35 millones de dólares en el ejercicio fiscal anterior [13], y es por esto que en el apartado 4.2.4 incluimos también Selenium, como una opción de código abierto.

Sin embargo cuando observamos la figura 4 podemos observar cómo, aunque UiPath y Automation Anywhere se mantienen en cabeza, BluePrism entra como un competidor de las otras dos plataformas.

Entre estas podemos observar en el estudio que se realizará desde el apartado 4.2.1 hasta el 4.2.3, que comparten grandes similitudes en cuanto al alcance de sus plataformas y objetivos, sin embargo desglosaremos lo que aporta cada plataforma entre las tres nombradas.

Cabe destacar que uno de los criterios para poder aparecer en el cuadrante mágico de Gartner de RPA es la necesidad de tener unos ingresos mínimos por

#### 4.2.1 UiPath

UiPath fue creada en 2005 en Bucarest, Rumania, destinada a la búsqueda de soluciones de automatización. Posteriormente, en 2013 y tras la demanda por parte de sus clientes, sacaron al mercado un producto de escritorio para crear dichas automatizaciones. En 2019 ya habían creado una plataforma *end-to-end* de RPA y se habían posicionado a la cabeza del sector [14].



Figura 5.- Logo de UiPath

Entre las herramientas que nos ofrece UiPath [15] se puede mencionar:

- Studio**  
 El entorno de desarrollo integrado, en el que los usuarios son capaces de desarrollar y probar los procesos de automatización. Admite varios lenguajes de programación como VB.NET y C#, además de proveer de una interfaz visual de *drag-and-drop* para los procesos más sencillos. Además, permite al usuario elegir otro entorno de trabajo para trabajadores no técnicos, en los que se simplifican algunas de las utilidades pero también simplifican la forma de interaccionar.
- Orquestador**  
 Es una aplicación web que nos permite acceder a un panel centralizado desde el que podemos monitorizar y controlar los procesos, máquinas y robots. También permite definir disparadores para los procesos desplegados (conocidos como *jobs*), así como acceder a los logs de los procesos y obtener diversos reportes.

- **Robots**  
Es el agente que permite ejecutar y controla las automatizaciones creadas en Studio. Estos pueden trabajar de forma tanto atendida como desatendida, en máquinas virtuales o físicas. En las ejecuciones desatendidas, los robots trabajan de forma autónoma, mientras que las atendidas permiten a los usuarios trabajar conjuntamente con los robots o supervisarlos.
- **Marketplace**  
Comunidad destinada al intercambio de conocimiento entre usuarios. En esta se pueden encontrar flujos reutilizables, componentes y actividades creadas por la comunidad.

Respecto a los beneficios [16,17] en el uso de UiPath podemos encontrar:

- Fácil uso y aprendizaje, gracias a su editor visual e interfaz intuitiva. Así mismo cuenta con una herramienta dedicada a aquellos usuarios sin experiencia en el desarrollo de código, llamada UiPath Studio X.
- Puede alojarse tanto en terminales virtuales como en entornos basados en la nube, lo que permite su adaptabilidad y escalabilidad.
- Es compatible con múltiples tecnologías como Microsoft, Java, SAP, Oracle, etc., y puede integrarse con otros sistemas y aplicaciones.
- Permite la incorporación de inteligencia artificial a los robots para dotarlos de mayor autonomía y capacidad para crecer y adaptarse.
- La empresa ofrece soluciones preconstruidas para el desarrollo de procesos, una especie de *framework* al que llaman ReFramework.
- El manejo y ejecución de sus tareas es centralizado, lo que permite ejecuciones rápidas así como su programación y monitorización.
- Ofrecen una edición comunitaria bajo la cual permiten hacer una prueba gratuita antes de tomar cualquier decisión.

Entre las limitaciones [18] de UiPath identificamos:

- Requiere una conexión a internet, tanto para la activación y validación de licencias, así como en la descarga de los paquetes de actividades.
- Al utilizar la versión gratuita o de comunidad no puede ser utilizado para fines comerciales.
- Al estar protegidos por derechos de propiedad queda prohibida la modificación del código fuente de los proyectos de automatización.
- Los usuarios son los responsables con el cumplimiento de normas legales y éticas, por lo que no se garantiza la seguridad o privacidad de los datos almacenados o procesados por los robots.

El uso de UiPath puede conllevar ciertas desventajas:

- Al utilizar la versión comunitaria, pueden discontinuar o modificar las versiones en cualquier momento, resultando en una necesidad de modificación o rehacer el código.
- Se requiere una inversión inicial para adquirir las licencias e infraestructura necesarias.
- Requiere de una capacitación y mantenimiento constantes para adaptarse a los cambios, tanto en los procesos como en las tecnologías.

El coste que conlleva esta herramienta depende del tipo de licencia y de los componentes que se decidan adquirir. Los precios actuales son los siguientes para pequeños departamentos, aunque puede variar en el caso de negocios pues indican que establezcas comunicación con ventas [19].

- **Automation Developer**  
420\$ mensuales/ 4.400\$ anuales  
Licencia para el uso de UiPath Studio, el editor visual para el diseño de robots.
- **Robot Units Bundle - UiPath hosted**  
1.150\$ mensuales/ 12.000\$ anuales  
Unidad de consumo que permite una cantidad predefinida (5.000) de uso por mes para ejecutar automatizaciones utilizando robots en la nube (*VM* y *Serverless*).
- **Citizen Developer**  
250\$ mensuales/ 2.600\$ anuales  
Licencia de usuario para la creación de automatizaciones en UiPath Studio X.
- **Unattended Robot - Self hosted**  
960\$ mensuales/ 10.000\$ anuales  
Licencia de robot para la ejecución de automatizaciones desatendidas en máquinas independientes.
- **Attended User**  
150\$ mensuales/ 1.600\$ anuales  
Licencia de usuario individual para la ejecución mediante UiPath Assistant.

Además de cada licencia individual, la compra de cualquiera de ellas conlleva un soporte técnico 24 horas, posibilidad de creación de hasta 3 *Tenants* y permite almacenar tus datos en la región de tu elección.

#### 4.2.2 Automation Anywhere

Automation Anywhere fue fundada en 2003, en San José, California, Estados Unidos, estableciéndose con el objetivo de ofrecer soluciones de automatización robótica

Entre las principales herramientas que nos ofrece Automation Anywhere [20] podemos encontrar:



Figura 6.- Logo de Automation Anywhere

- **Control Room**  
Es el centro de control y administración de la plataforma, utilizado para la administración, supervisión e implementación de los bots, permitiendo a los usuarios el monitoreo de los robots, la generación de informes así como la programación y supervisión de tareas.
- **Bot Creator**  
El Bot Creator es el cliente de desarrollo de bots, tareas autocontenidas diseñadas para ejecutarse con poca o ninguna intervención humana. Este incluye el editor de tareas, visor de eventos, ejecutor de bots, autologin y el planificador local.
- **Bot Runner**  
El Bot runner es el entorno software utilizado para el despliegue y ejecución de los robots creados en el Bot Creator. Las instancias del Bot Runner descargan los bots del Control Room.
- **Bot Store**  
Biblioteca en línea, accesible desde el Control Room, que contiene bots preconstruidos o paquetes al repositorio del Control Room.

Entre los beneficios [16,17] que nos reporta trabajar con Automation Anywhere, encontramos:

- Funciones cognitivas emergentes basadas en *machine learning* para la automatización de tareas más complejas.
- *Multi-tenancy*
- Comandos OCR (*Optical character recognition*) para la lectura de documentos desde imágenes.
- Tecnología basada en la plataforma de Microsoft.
- Produce valiosos reportes de rendimiento para el análisis y mejora de los procesos.
- Oferta de servicios en la nube, por lo que no se necesita inversión en infraestructura.

Entre las limitaciones [6] podemos identificar:

- Al trabajar con la Community Edition solo se puede contar con una instancia de Bot Creator y Bot Runner, y las ejecuciones solo podrán ser atendidas. No se podrá importar robots desde la Bot Store ni exportarlos a la Control Room.
- Cuenta con una capacidad de integración con otras aplicaciones más baja que otras herramientas.
- La simplicidad en su plataforma puede conllevar dificultad a la hora de generar robots con una lógica más complicada, así como en la gestión de la escalabilidad en procesos.
- Pequeña cantidad de tipos de datos, lo que complica el desarrollo al necesitar construir lógicas adicionales.

Por parte de las desventajas [21] podemos encontrar algunas como:

- Migración entre las diferentes versiones del producto requiere esfuerzo y puede resultar complicado.
- Necesita de una inversión inicial alta para trabajar con la herramienta, además su versión de prueba no aporta una visión general de la herramienta al estar limitada.

El coste de la herramienta [22] varía en función del tamaño de negocio. Para negocios medianos y grandes ofertan el *Advanced Pack*, un pack con el que aseguran las herramientas para descubrir, automatizar y optimizar procesos de back-office y front-office, pero no aportan más información al respecto pues debes comunicarte con ventas. Para pequeños negocios y equipos ofrecen el *Cloud Starter Pack*, que incluye las siguientes herramientas a fecha de Mayo de 2020:

- **Pack inicial**  
750\$ mensuales  
Incluye un Bot Creator, un Control Room y un Bot Runner desatendido.
- **Bot Runner desatendido**  
500\$ mensuales
- **Bot Runner atendido**  
125\$ mensuales

### 4.2.3 SS&C BluePrism

SS&C BluePrism es una compañía de software que se creó en 2001 en Reino Unido, inicialmente bajo el nombre de BluePrism. Sus fundadores vieron el potencial de la automatización de procesos en un entorno empresarial y se propusieron la creación de una plataforma para automatizar las tareas repetitivas realizadas hasta el momento por las personas. Tras su salida en bolsa en 2016 y su adquisición por SS&C en 2022, la empresa se ha posicionado como uno de los principales proveedores de RPA [23].



Figura 7.- Logo de SS&C BluePrism

Entre sus principales herramientas [24] podemos encontrar:

- **Process Studio**  
Entorno de desarrollo para el diseño y configuración de procesos de automatización mediante el sistema de *drag-and-drop*, permitiendo el desarrollo a usuarios sin un extenso conocimiento en lenguajes de programación.
- **Control Room**  
Una consola centralizada de gestión para proporcionar control, monitorización y analíticas de los robots desplegados. Este permite la monitorización y priorización de tareas, así como la observación y gestión de los procesos a un nivel general.
- **DX (Digital Exchange)**  
Una plataforma online en la que los usuarios pueden encontrar lo que son conocidos como “*Skills*” o “*Connectors*”, integraciones ya preparadas con otros sistemas, aplicaciones o tecnologías para la aceleración del desarrollo de los procesos.

Los beneficios [17,25] que obtenemos al trabajar con SS&C BluePrism son:

- Alta adaptabilidad a cualquier plataforma o aplicación.
- Estructura basada en creación de procesos y objetos.
- Al almacenar todos los procesos en una base de datos central permite la comparación de versiones entre el código desarrollado.
- La herramienta BluePrism es altamente estable y escalable, sobre todo al realizar comparaciones con sus competidores con una carga mayor.

Entre las limitaciones [25] que tiene esta herramienta:

- Solo permite el desarrollo de automatizaciones de *back-office*.
- Todas las aplicaciones se encuentran en el mismo dispositivo local, lo que no aporta opciones de control desde una consola online.
- Necesidad de conexión a Internet para descargar y activar, así como para instalar y configurar su infraestructura de base de datos.

El uso de esta puede conllevar algunas desventajas [25] como pueden ser:

- Necesidad de vasto conocimiento en programación por parte del usuario.
- OCR se empieza a incluir en la versión 6 pero no aporta actividades de fácil uso y es muy compleja.

En cuanto al precio de las licencias BluePrism ofrece un modelo de precios estándar por licencia de robots con cuotas anuales, además de cuotas adicionales por soporte o mantenimiento. Sin

embargo en 2021 incorporó el concepto de pago por uso, para así abastecer a los clientes que no podían responder a los picos inesperados de demanda de automatizaciones [26].

La compra de su licencia se hace en base a trabajadores digitales, y a pesar de no ofrecer esta información sin contactar a sus servicio de ventas, sus usuarios estiman [27] que el precio por un trabajador digital esta alrededor de los 13.000\$, anualmente una infraestructura está sobre los 100.000\$ y que el primer año por hardware, la implementación y coste de personas se encuentra alrededor de los 30.000\$.

Los usuarios que comparan esta herramienta con sus competidoras aceptan que el precio de esta es sustancialmente mayor.

#### 4.2.4 Selenium

Selenium fue desarrollado inicialmente en 2004, como una herramienta interna en una compañía consultora de software, debido a la necesidad de pruebas frecuentes en una aplicación web. Así pues se quiso crear una forma automatizada de validar el comportamiento de esta.

La herramienta creada se basó en JavaScript y fue llamada *JavaScriptTestRunner*. Posteriormente se fusionó con otro proyecto interno llamado Depot, creando finalmente el proyecto conocido como Selenium.

Finalmente en 2006 Selenium fue publicado como una herramienta de código abierto y la comunidad empezó a contribuir activamente en su desarrollo y mejoras.

Actualmente Selenium no es solo una herramienta o API, sino que es un proyecto marco conformada por varias herramientas y librerías, que actualmente incluye apoyo en diferentes lenguajes como Java, Python, C# y demás [28].

Cabe destacar que, aunque se incluye en la comparativa, Selenium no es una herramienta de RPA, sino una herramienta de automatización de pruebas pero ha sido utilizada en ambos ámbitos.

A continuación vemos algunas de estas herramientas [29]:

- **WebDriver**  
WebDriver es una API y protocolo que define un lenguaje para el control del comportamiento del navegador, llamando driver al componente que se encarga de la comunicación entre Selenium y el navegador, proveídas por estos para controlarlos y ejecutar pruebas, como si un usuario real estuviera utilizándolos.
- **IDE**  
IDE (*Integrated Development Environment*) es una herramienta para el desarrollo de casos de uso. Es una extensión de Chrome y Firefox y ha sido definida como una de las formas más eficientes de desarrollo. Graba las acciones del usuario en el navegador, usando comandos de Selenium, con parámetros definidos por el contexto de ese elemento.
- **Grid**  
Grid te permite la ejecución de tus casos de uso en diferentes máquinas entre diferentes plataformas. La activación se realiza localmente, permitiendo la ejecución en remoto.

Selenium puede aportar los siguientes beneficios [29] al trabajar con ella:

- Herramienta de código abierto.



Figura 8.- Logo de Selenium

- Soporta múltiples lenguajes de programación (C#, Java, Python, JavaScript, etc).

También tiene una serie de limitaciones [30], como:

- Normalmente utilizado como software para hacer pruebas de estrés en aplicaciones web.
- Al no poder cambiar a otro programa en la ejecución suele requerir scripts y herramientas adicionales para cubrir todo el flujo de trabajo.

Las desventajas [31] que encontramos en esta son:

- Requiere conocimientos en alguno de los lenguajes que soporta.
- Solo soporta aplicaciones web
- No permite automatizar

La herramienta no tiene costes asociados al ser de código abierto por lo que puede ser utilizada sin ningún reparo a cualquier tipo de escala, ya seas un particular, un departamento o una empresa de cualquier tamaño.

### 4.3 Comparativa entre las herramientas

Cuando comparamos las herramientas previamente descritas vemos que el enfoque es muy parecido entre las diferentes herramientas, cubriendo las necesidades dentro de una arquitectura RPA. Tenemos los *bots* o robots, software que ejecuta la automatización, tenemos los entornos de desarrollo y finalmente un entorno para el control de las ejecuciones y equipos. Adicionalmente tenemos las librerías digitales que contienen componentes reutilizables. Todo ello bajo la nomenclatura específica de cada empresa.

Así pues vamos a hacer una comparativa entre estas herramientas para ver las diferencias en sus implementaciones. Cabe destacar que esta se realizará sobre las versiones *Community* de las aplicaciones, versiones de prueba o para personas físicas.

UiPath nos ofrece una versión *community* que cuenta con Studio, Robot y Orquestador de prueba. No conlleva límite de tiempo, aunque es renovada anualmente y está destinada a desarrolladores.

Automation Anywhere ofrece la versión *community* de su Automation 360 a estudiantes y desarrolladores, también ampliando a pequeños negocios.

Blueprism nos ofrece una prueba gratuita por 30 días para hasta 1 trabajador digital y 15 procesos.

Cada una de ellas ha optado por una estrategia diferente en la creación de su herramienta.

Mientras que Blueprism opta por una aplicación de escritorio que contiene todas las funcionalidades, Automation Anywhere trabaja con una aplicación web para su interacción con el usuario, teniendo un agente en la máquina para ejecución y grabación de las tareas. UiPath por su parte divide las plataformas, teniendo una aplicación de escritorio para el Studio (entorno de desarrollo), un agente con el que ejecutar el código y contando con una aplicación web para el despliegue y control de máquinas y robots (Orquestador).

#### 4.3.1 Plataformas de desarrollo

Tras realizar pruebas en dichos entornos podemos afirmar que UiPath cuenta con el entorno más intuitivo y con mayores facilidades a la hora de desarrollar (Como son el acceso a su plataforma de contenido propio y frameworks ya preparados para diferentes tipos de desarrollo, máquinas de estado, secuenciales, etc...).

Nos permite la creación de proyectos, contenidos en carpetas y en los que se puede almacenar diferentes tipos de documentos, siendo los workflows en los que están definidos el código que se refleja de los diagramas realizados mediante actividades de arrastrar.

Los principales flujos a utilizar son secuencias, diagramas de flujo y máquinas de estado.

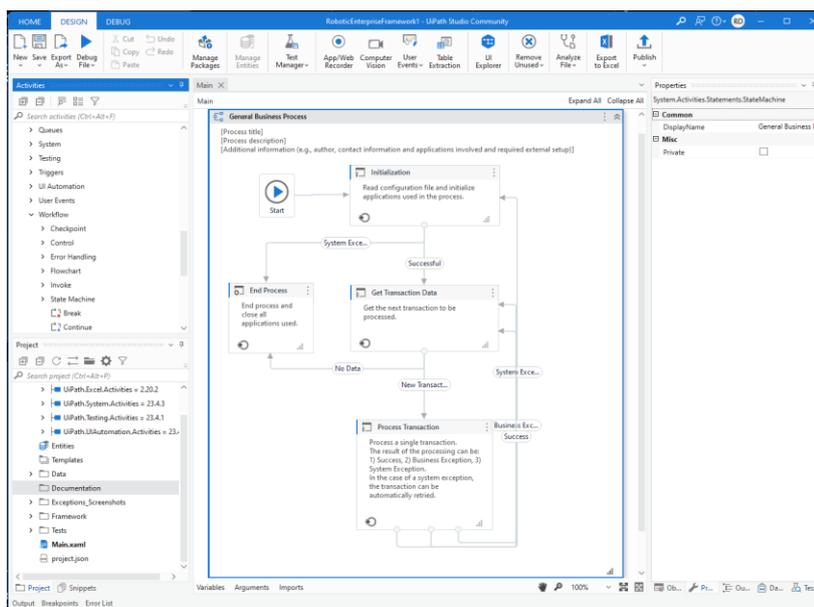


Figura 9.- UiPath Studio

Cuenta con una gran cantidad de actividades predefinidas así como, y a diferencia de sus competidores, un gestor de actividades incluido en la propia aplicación, lo que nos permite una búsqueda rápida y sin acceso al navegador, algo bastante útil cuando los desarrollos interactúan directamente con estos y en muchos casos se ejecutan procesos que los cierran.

Cuenta con una consola configurable con muchas opciones, tanto para el desarrollo, como para su depuración, lo que la convierte en una aplicación muy completa.

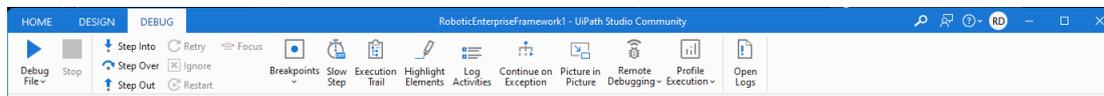


Figura 10.- UiPath Studio - Menú de depuración

Automation Anywhere sigue de cerca, con un enfoque intuitivo pero un poco simplista, se pueden realizar desarrollos rápidamente y cuenta con una serie de actividades predefinidas muy útiles, sin embargo cuenta con solo tres objetos base que, aunque se pueden relacionar y crear flujos más complejos, no lo permite con facilidad.

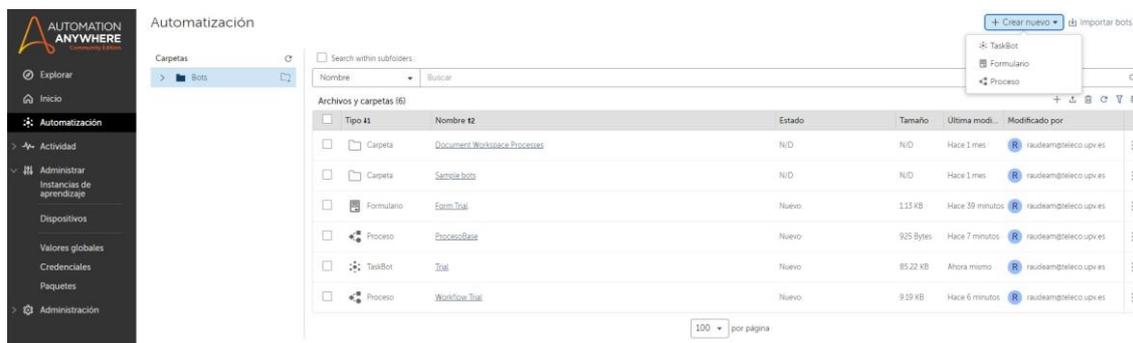


Figura 11.- Automation 360 - Panel Automatización

Estos tres objetos son:

- **TaskBots**  
Secuencias lineales que contienen las acciones, actividades preparadas para realizar las tareas básicas en una automatización, desde la navegación web hasta la gestión de

ficheros.

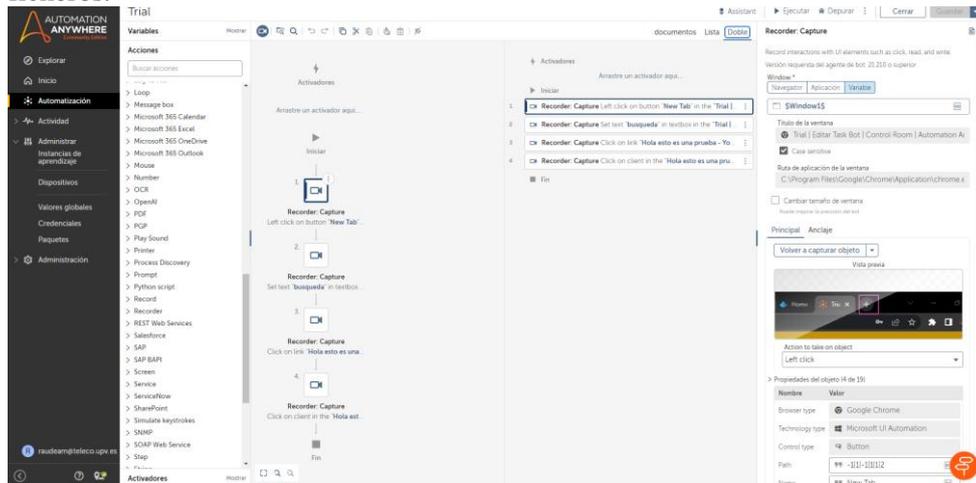


Figura 12.- Automation 360 - Taskbot

- **Formulario**  
Permite la interacción con el usuario para toma de decisiones, definición de variables y demás.
- **Proceso**  
Flujo visual que permite la organización del flujo de ejecución del proceso, con toma de decisiones y llamadas a TaskBots, Formularios y otros procesos.

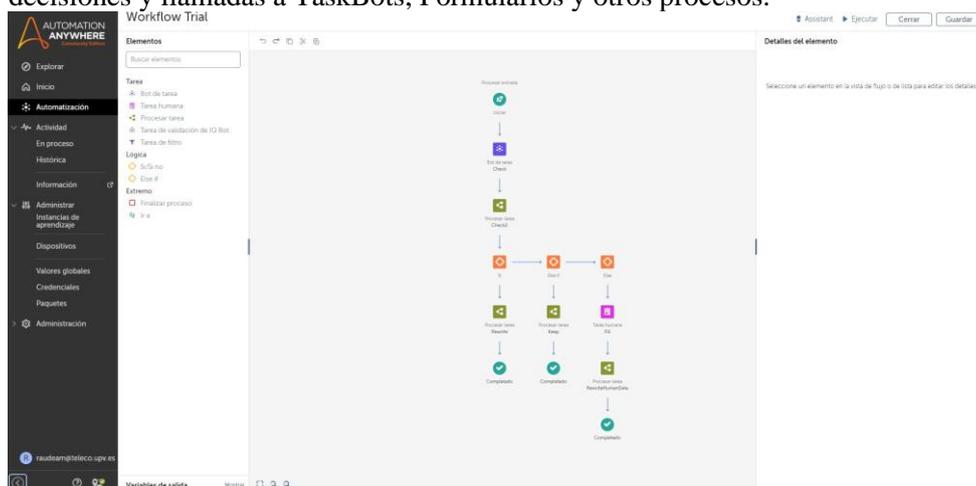


Figura 13.- Automation 360 - Process

SS&C BluePrism es el que mayor curva de aprendizaje necesita, dispone de una creación de flujos que da cierta flexibilidad a la hora de definir procesos más complejos pero carece de una buena cantidad de actividades predefinidas, lo que hace que sus desarrollos sean mucho más arduos y consuman mucho tiempo. Las actividades predefinidas en el Digital Exchange tampoco conllevan una facilidad a la hora de instalar.

En este caso contamos con dos características:

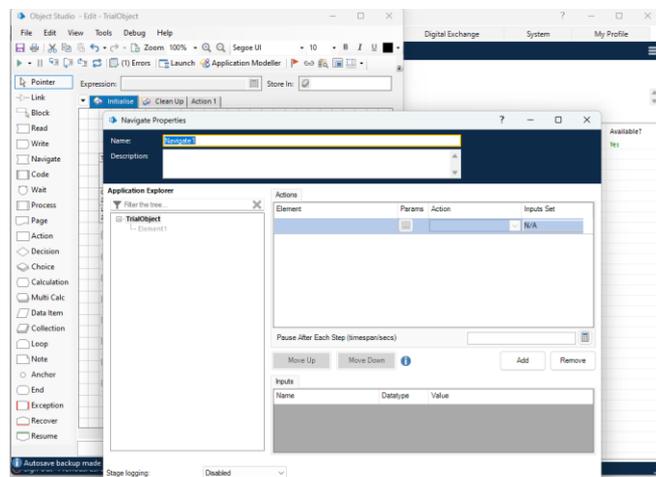


Figura 14.- BluePrism - Objetos

- **Objetos**  
Los objetos nos permite la creación de objetos de negocio mediante diagramas de flujo, utilizados para automatizar aplicaciones heredadas de Windows. También se pueden crear otro tipos de objetos como los objetos de negocio COM, utilizados para la comunicación con aplicaciones que cuentan con API.
- **Procesos**  
Diagrama del flujo de proceso completo, desde el cual se pueden llamar a los objetos y a otros procesos. Permite la creación del código de automatización mediante su representación gráfica.

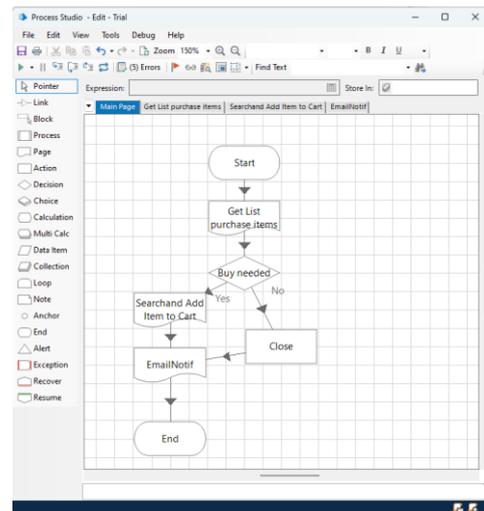


Figura 15.- BluePrism - Procesos

#### 4.3.2 Plataforma de control

En UiPath contamos con el Orquestador, la aplicación web desde la cual podemos gestionar a dos niveles, desde el Tenant y a nivel de carpeta.

En las carpetas trabajamos a nivel de proceso, encontramos todo lo relativo a ellos:

Los procesos creados, las ejecuciones realizadas, sus logs, la definición de los *triggers*, un pequeño panel de monitoreo de los procesos incluidos en dicha carpeta, así como las colas y los *assets* (variables accesibles desde los scripts), entre otros...

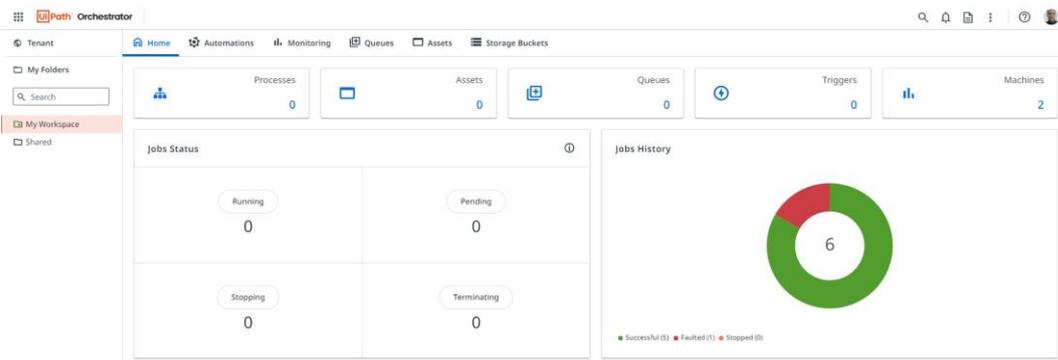


Figura 16.- UiPath Orchestrator - Folder

Cuando accedemos a la pestaña del Tenant podemos trabajar a un nivel más técnico, desde el que podemos gestionar recursos como las conexiones establecidas a equipos, las máquinas conectadas, usuarios, paquetes, licencias y demás aspectos.

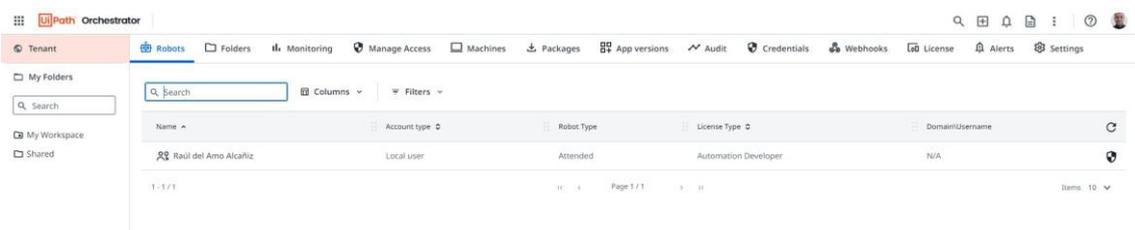


Figura 17.- UiPath Orchestrator - Tenant

Desde BluePrism tenemos control sobre los equipos, colas y el programador, lo que nos aporta una pequeña posibilidad de gestión.

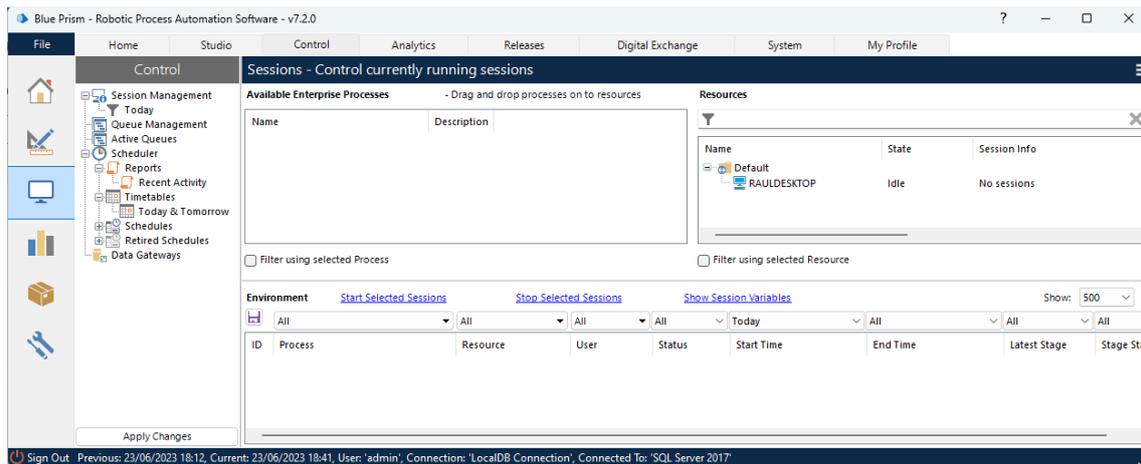


Figura 18.- BluePrism - Control

Sin embargo la parte interesante que nos puede ofrecer Blueprism es la posibilidad de crear paneles, pues aunque en el resto de herramientas contamos con algunos paneles, no tenemos la posibilidad de modificarlos como tenemos en este caso.



Figura 19.- BluePrism - Analíticas

En este caso es Automation Anywhere la que menos características parece poseer. De nuevo una interfaz muy limpia y sencilla pero desde la que solo podemos ver las ejecuciones, gestionar dispositivos, manejar los paquetes y crear valores globales y credenciales.

### 4.3.3 Conclusiones

Así pues podemos observar que cada empresa, aunque cubriendo los mismos puntos, ha tenido un visión diferente en la forma de llevarlo a cabo, tanto en plataforma como en el enfoque a puntos clave.

Desde mi punto de vista y tras la comparativa y pruebas realizadas, concuerdo con el cuadrante Gartner que revisamos el Estudio de herramientas en el mercado. Parece que UiPath es la herramienta más realizada y que cubre el mayor número de necesidades y facilidades en un proyecto RPA.

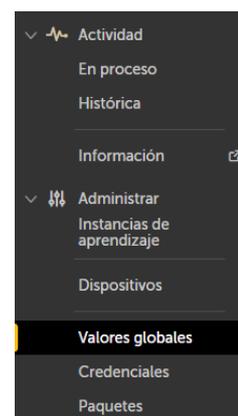


Figura 20.-  
Automation  
Anywhere - Gestión

Características	UiPath	Automation Anywhere	BluePrism	Selenium
Lenguaje base	Drag-and-Drop, .NET, C#	Drag-and-Drop, Java, .NET	Drag-and-Drop, .NET	Java, Python, C#
Curva de aprendizaje	Baja	Baja	Alta	Media
Coste Licencias	Media	Baja	Alta	No aplica
Capacidad de integración	Alta	Media	Media	Baja
Aplicación	Nube, Local	Nube	Local	Local

**Tabla 1. Comparativa de herramientas.**

Sin embargo, las necesidades específicas de cada empresa pueden variar y afectar a la elección sobre la herramienta. Selenium es una opción para aquellas empresas que tengan desarrolladores expertos, no quieran realizar un gasto en la generación de la infraestructura y solo necesiten aplicar automatizaciones en aplicaciones web o puedan completar con otro software.

Aquellas empresas que aspiren a realizar automatizaciones de procesos sencillos por un bajo coste, con cortos plazos y con empleados con conocimientos medios en programación deberían decantarse por Automation Anywhere.

En el caso de buscar una herramienta más robusta para empleados con conocimientos altos en programación la mejor opción sería BluePrism, aunque cabe destacar que el equipo RPA no debería ser muy grande o descentralizado, pues corremos el riesgo de recurrir en costes muy altos.

Finalmente si nuestro objetivo es tener una herramienta con una gran capacidad de integración con otros sistemas, con un equipo grande o descentralizado UiPath sería la opción más segura.

#### 4.4 Caso de Uso

Situándonos en la situación previa a la implementación del robot, nos encontramos con un departamento de una consultoría dedicado a realizar proyectos rápidos de desarrollo con diferentes tecnologías. Está conformado por un equipo pequeño, compuesto en su mayoría por desarrolladores. Hay dos personas encargadas en la gestión de proyectos y marketing.

Estas dos personas se han visto desbordadas por picos de trabajo, pero no cubren el suficiente para contratar a una tercera. Es en este momento en el que deciden revisar sus tareas diarias y ver si alguna aplica para implementar una automatización de la misma.

El objetivo es identificar aquellas tareas repetitivas con reglas de negocio rígidas que puedan implementarse con la tecnología RPA, liberando a los trabajadores y permitiéndoles dedicarse a tareas que aportan mayor valor

##### 4.4.1 Definición del caso de uso

El primer paso en un caso de uso es identificarlo, esto puede quedar fuera del ciclo de vida del proyecto como tal, pero es un paso de gran importancia. A la hora de plantear que tareas podemos automatizar deberemos ser conscientes del alcance, de la complejidad del proceso, de si conlleva una toma de decisiones flexible o de los sistemas a automatizar.

Es recomendable que en esta toma de decisiones se involucre una persona del equipo RPA, idealmente un arquitecto RPA, que podrá definir, identificar y guiar en esta toma de decisiones.

Una vez que se ha identificado el proceso, podemos pasar a revisar el alcance. Aunque es una posibilidad la ampliación posterior, siempre facilita el desarrollo y la cohesión del código el haber

sido previsores a la hora de plantear que queremos cubrir, que debe quedar fuera o que podemos ampliar posteriormente.

La realidad es que muchas de estas decisiones van relacionadas con el nivel de aceptación de la tecnología y con la comprobación de que los desarrollos realmente aportan valor en el equipo. Tanto es así que podemos ver cierta tendencia en los desarrollos para los diferentes departamentos o equipos, iniciando con proyectos parciales o de poca dedicación y ampliándolos con el tiempo u optando por desarrollos más ambiciosos.

En nuestro caso optamos por un proceso con unos límites bastante definidos, en el que la persona encargada dedicaba una cantidad importante de su tiempo, pero que a su vez tiene poco nivel de toma de decisiones y es una tarea muy repetitiva.

Este proceso cubriría el acceso a una aplicación web estatal que contiene la información pública de las licitaciones presentadas por órganos de gobierno, el recorrido por todas estas licitaciones y su recogida de datos. Posteriormente almacenaría dichos datos y enviaría dicha información a los usuarios definidos.

Tras la identificación del proceso a automatizar se realiza el pliego del proyecto (Anexo II – Pliego del proyecto RPA), en el que se recoge el alcance de la automatización, las herramientas a utilizar, los plazos y otra información clave. Con esto pretendemos tener una buena comprensión del proceso y poder definir los costes y plazos iniciales, aunque cabe destacar que este es un documento vivo, pues en este tipo de proyectos se actualizan durante todo el desarrollo, ya sea por decisiones en el desarrollo o por modificaciones por parte de negocio en el flujo a implementar.

Esto se puede realizar gracias a la metodología seguida. Gracias al uso de la metodología Ágil y su marco de trabajo Scrum (Capítulo 3), podemos contar con la flexibilidad necesaria para aplicar cambios en el ciclo de vida del proyecto.

#### **4.4.2 Desarrollo del caso de uso**

El desarrollo del caso de uso se realizó de acuerdo a lo estipulado en el pliego del proceso (7.2 Anexo II) y se puede ver su implementación y código en el “Desarrollo documentado” (7.1 Anexo D).

En este último podemos ver como el desarrollo se realizó basándose en el *ReFramework*, una plantilla para el desarrollo de procesos secuenciales y utilizando las actividades aportadas en los paquetes de UiPath, pues a diferencia de otras versiones de comunidad estos estaban disponibles a los desarrolladores de dicha versión.

En dicho anexo podemos ver el detalle del desarrollo, centrándonos en las diferentes partes del proyecto, pudiendo revisar las decisiones tomadas y los ajustes realizados según la respuesta de la aplicación.

Lamentablemente en lo que respecta a los plazos superó ligeramente el tiempo inicial planteado al encontrar algunas dificultades. En el caso de la web el funcionamiento del buscador mostraba diferentes errores, algunos de estos siendo:

- Muestreo de opciones descatalogadas  
En el buscador, en un campo de selección desplegable, mostraba una opción que no devolvía ningún resultado y a su vez bloqueaban la aparición de ciertos campos. Esto podía ocurrir de forma aleatoria en el primer acceso a la web y al buscar por este desaparecía, desbloqueando el resto de valores. Así pues en el inicio de la aplicación se implementó la búsqueda por este valor en caso de que existiera la etiqueta en cuestión en dicho campo.
- Reseteo del filtro en la iteración de una búsqueda



Tras la implementación se detectaron ejecuciones inusualmente largas y tras unas pruebas se detectó que al filtrar e iterar las páginas podía resultar ocasionalmente en que el filtro se reseteara en uno de los campos de búsqueda (el estado) y el proceso intentara leer todos los casos sin dicho filtrado.

Respecto a las aplicaciones utilizadas también se encontraron algunos retos.

En cuanto al almacenamiento de las opciones se decidió por optar por una base de datos.

Se podría haber realizado en base a un Excel para poder mantenerlo localmente o en una carpeta compartida, pero se decidió por la base de datos en base a sus bajos tiempos de respuesta y a la menor posibilidad de encontrar errores durante su ejecución.

Así pues se realizó el desarrollo inicialmente en base a una base de datos SQLite, pues permitía mantener el proyecto sin costes adicionales y además mantener la base de datos en una carpeta compartida del equipo, dando la posibilidad de realizar búsquedas adicionales al funcionamiento base del robot sin necesitar grandes conocimientos, ni una gestión de usuarios adicional.

Sin embargo en la mitad del desarrollo se descubrió que dicha base generaba problemas al contener campos con caracteres especiales, apareciendo estos incluso en campos esenciales para la identificación de la licitación.

Así pues y para mantener la no necesidad de gestión de usuarios de una base de datos online, así como el poder tener un fichero local, se optó por una base de datos de Access. En esta decisión también influyó el coste, pues entraba dentro del paquete de aplicaciones de la universidad.

En esta línea también se decidió inicialmente el trabajar con todas las herramientas de Microsoft 365, creando un espacio de trabajo en Sharepoint en el que almacenar la información respecto al proyecto, así como los ficheros de configuración online del proceso. Sin embargo para acceder a estos se necesitaba una configuración de Azure solo realizable por los administradores así que se decidió adaptar a tener los ficheros almacenados en Google Sheets.

Este tipo de decisiones rápidas son respaldadas en proyectos como este mediante la metodología Agile, y gracias a la modularidad de UiPath y sus actividades predefinidas se pueden realizar estos cambios de forma rápida, incluso cuando el proyecto ya está implementado.

## Capítulo 5. Conclusiones, limitaciones y propuesta de trabajo futuro

La automatización de procesos es un importante recurso que permite la reasignación de tareas al personal, permitiendo robotizar aquellas partes más repetitivas de su trabajo, revalorizando el tiempo de estas personas.

Cabe siempre ser precavidos con las elecciones de los procesos a automatizar y valorar si un desarrollo en *back-end* con otras tecnologías podría ser la solución ideal previamente a automatizar directamente con RPA; pero también se ha podido observar que por el tiempo de desarrollo puede ser una herramienta bastante valorada, pues incluso puede ser utilizada para tareas puntuales o como prueba previa a un desarrollo más ambicioso.

Como se ha mencionado en el estudio de herramientas, a día de hoy hay muchas posibilidades a nuestra elección. A pesar de sus grandes similitudes debemos hacer un estudio, tanto de nuestras necesidades como de sus posibilidades. Muchas de estas empresas están trabajando en diferentes líneas como pueda ser la implicación de la inteligencia artificial en la toma de decisiones de los procesos o la lectura de documentos para su clasificación automática.

En cualquier caso el RPA es, a día de hoy una herramienta muy reconocida y utilizada en todo tipos de negocio.

Respecto al caso de uso desarrollado se debe tener en cuenta que, al ser un caso ficticio, en lo que respecta a los integrantes del equipo, no tenemos una visión clara de si su implementación bastaría o podría ser mejorada ampliando, por ejemplo, los diferentes tipos de contrato que podemos encontrar en la web. Sin embargo podemos ver claramente como una tarea a la que se dedicaba gran parte de la jornada pasa a ejecutarse de forma autónoma y en un horario nocturno, permitiendo al trabajador pasar a un puesto de revisión de los resultados del robot. Puesto que la mayoría del tiempo se dedicaba a la navegación por la página podemos decir que es un proyecto ideal para su implementación.

Este proyecto permitiría a su vez, demostrar el potencial de la herramienta e iniciar un una etapa de identificación y automatización de otros procesos.

En lo que respecto al desarrollo del proyecto podemos concluir con que ha sido un proceso fácil de automatizar pero aun así, y como ocurre con gran parte de los procesos RPA, se han incluido capacidades posteriormente a la definición, lo que puede afectar a los plazos iniciales. En este aspecto contar con una metodología como es la ágil nos permite tener la agilidad y comunicación para ampliar o modificar la hoja de ruta inicial.

Se han encontrado limitaciones en lo que respecta a las aplicaciones a utilizar, esto se debe a que en las cuentas utilizadas el nivel de derechos de los usuarios no permitía utilizar algunas de las capacidades de integración, aún así cabe destacar que la herramienta RPA utilizada disponía de varias opciones permitiendo el desarrollo de otras soluciones.

Los resultados han sido los esperados, concluyendo en un proyecto exitoso que permitiría la inclusión de la tecnología y resolvería el problema inicialmente planteado, un punto de apoyo para un equipo de trabajo pequeño desbordado por periodos con mayores volúmenes de trabajo.

## Capítulo 6. Bibliografía

- [1] Aguirre, S., & Rodriguez, A. (2017). Automation of a business process using robotic process automation (RPA): A case study. In *Applied Computer Sciences in Engineering: 4th Workshop on Engineering Applications, WEA 2017, Cartagena, Colombia, September 27-29, 2017, Proceedings 4* (pp. 65-71). Springer International Publishing.
- [2] OECD (2023), OECD Employment Outlook 2023: Artificial Intelligence and the Labour Market, OECD Publishing, Paris, <https://doi.org/10.1787/08785bba-en>.
- [3] <https://www.atlassian.com/es/agile> [Online].
- [4] <https://www.atlassian.com/es/agile/scrum> [Online].
- [5] <https://www.uipath.com/> [Online].
- [6] <https://www.automationanywhere.com/la> [Online].
- [7] <https://www.blueprism.com/es/> [Online].
- [8] Wewerka, J., & Reichert, M. (2020). Robotic Process Automation--A Systematic Literature Review and Assessment Framework. *arXiv preprint arXiv:2012.11951*.
- [9] Kaya, C. T., Türkyılmaz, M., & Birol, B. (2019). Impact of RPA technologies on accounting systems. *Muhasebe ve Finansman Dergisi*, (82).
- [10] Kokina, J., & Langmann, C. The Role of Governance and Attended and Unattended Bot Use in Robotic Process Automation (Rpa) Performance—an Exploratory Study. *Available at SSRN 4295732*.
- [11] <https://www.gartner.com/en/documents/4004033> [Online].
- [12] <https://www.gartner.com/en/research/methodologies/magic-quadrants-research#:~:text=Niche%20Players%20focus%20successfully%20on,an%20understanding%20of%20market%20direction>. [Online]
- [13] <https://harpia-software.com/cuadrante-magico-de-gartner-rpa-2022/> [Online].
- [14] UiPath, “About us: Our story,” <https://www.uipath.com/company/about-us> [Online].
- [15] <https://docs.uipath.com/es>. [Online].
- [16] Jovanović, S. Z., Đurić, J. S., & Šibalija, T. V. (2018). Robotic process automation: overview and opportunities. *International Journal Advanced Quality*, 46(3-4), 34-39.
- [17] Beneficios de implementar RPA UiPath - Rootstack. <https://rootstack.com/es/blog/beneficios-de-implementar-rpa-uipath> [Online].
- [18] UiPath Legal Terms, <https://www.uipath.com/legal/trust-and-security/legal-terms> [Online].
- [19] Plans and Pricing, <https://www.uipath.com/pricing> [Online].
- [20] <https://docs.automationanywhere.com/> [Online].
- [21] <https://www.peerspot.com/products/automation-anywhere-aa-pros-and-cons> [Online].
- [22] <https://www.automationanywhere.com/company/press-room/rpa-cloud-packages-for-business-continuity> [Online].
- [23] <https://www.computing.es/mercado-ti/la-fintech-ssc-compra-blue-prism/> [Online].
- [24] <https://bpdocs.blueprism.com/en-us/home.htm> [Online].
- [25] <https://www.linkedin.com/pulse/comparison-two-rpa-leaders-blueprism-uipath-jakub-va%C5%88hara/> [Online].



- [26] <https://www.fortra.com/es/blog/comparacion-de-soluciones-rpa-blue-prism-vs-automate-y-como-pueden-utilizarse-en-conjunto> [Online].
- [27] <https://www.peerspot.com/products/blue-prism-reviews> [Online].
- [28] <https://www.selenium.dev/history/> [Online].
- [29] <https://www.selenium.dev/documentation/> [Online].
- [30] Yatskiv, S., Voytyuk, I., Yatskiv, N., Kushnir, O., Trufanova, Y., & Panasyuk, V. (2019, June). Improved method of software automation testing based on the robotic process automation technology. In *2019 9th International Conference on Advanced Computer Information Technologies (ACIT)* (pp. 293-296). IEEE.
- [31] Santos, F. D. S. (2019). *Using a test automation tool for robotic process automation: An empirical study* (Master's thesis).

## Capítulo 7. Anexos

### 7.1 Anexo I – Desarrollo documentado

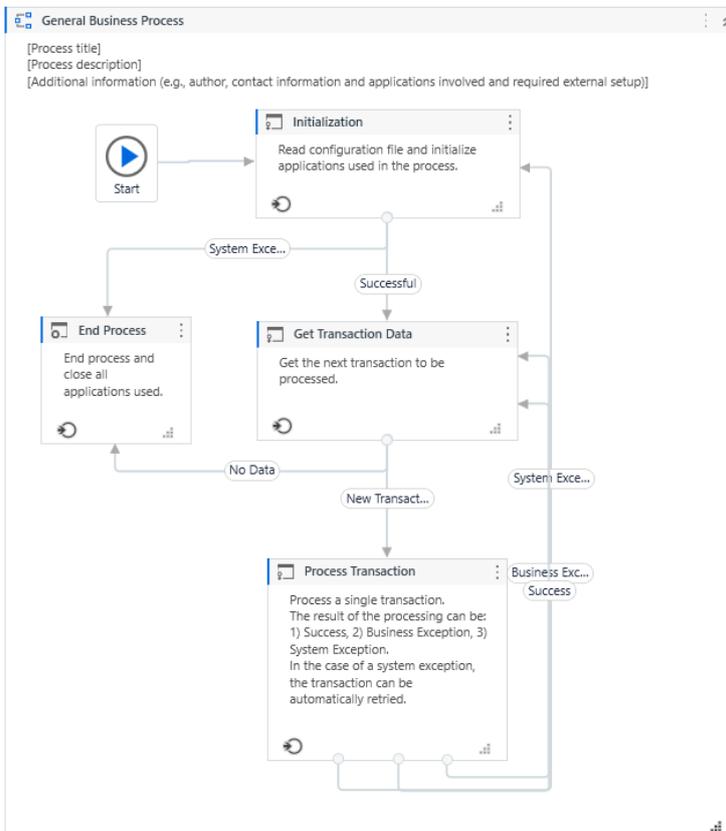
El desarrollo se ha realizado utilizando el REFramework de UiPath como base.

El REFramework (*Robotic Enterprise Framework*) es una plantilla para el desarrollo de procesos RPA de UiPath, especialmente indicada para la implementación de procesos transaccionales, debido a la posibilidad de la gestión de errores que nos ofrece la propia plantilla.

Así pues en esta encontramos como base una máquina de estados, que cuenta con cuatro estados principales:

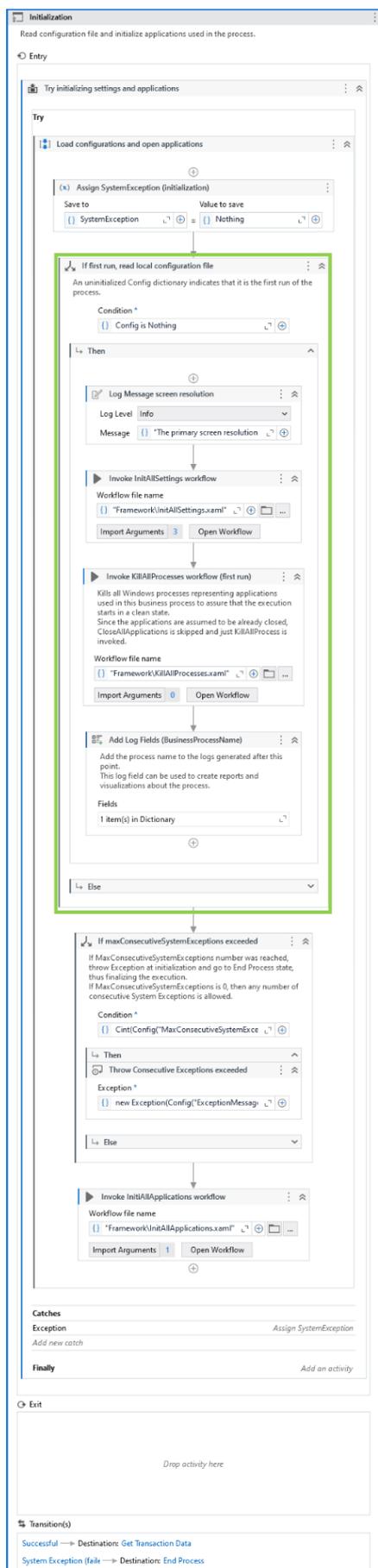
- **Initialization**  
Lee el fichero Excel de configuración e inicializa las aplicaciones usadas en el proceso. En caso de encontrar una excepción finaliza la ejecución accediendo al estado “End Process” mientras que si no la hubiera procede al “Get Transaction Data”.
- **Get Transaction Data**  
Obtiene la siguiente transacción a procesar, en caso de no funcionar en base a una cola de UiPath también se utilizará en la primera iteración para obtener la variable sobre la que iteraremos. Esto se puede ajustar para que utilice una tabla, una lista, etc.
- **Process Transaction**  
Ejecuta una transacción y posteriormente en base al resultado (Correcto, excepción de negocio o excepción del sistema) puede reintentar o proseguir, según se haya indicado en la configuración.
- **End Process**  
Finaliza el proceso y cierra las aplicaciones en uso, también se puede utilizar cualquier tarea adicional como pueda ser el envío de resultados o flujo fuera del nivel transaccional.

Entre los ficheros provistos por la plantilla encontramos diferentes workflows ya preparados para las funcionalidades base como son la lectura de la configuración, la captura de pantalla, la gestión de la transacción en función del resultado y el reintento de transacción. Además también existen diferentes ficheros preparados para diferentes tareas que se encuentran vacíos y es el desarrollador el encargado de rellenarlos.



#### 7.1.1 Initialization

En este estado se realiza en su primera ejecución la lectura de la configuración, que será almacenada en un diccionario de texto, objeto para su posterior uso. Además se matan los procesos de aquellas



aplicaciones que vayan a ser utilizadas en la ejecución, para así poder tener un inicio limpio. Esta parte se puede encontrar reflejada en la imagen del código, remarcada por un cuadrado verde.

En nuestro proceso el fichero InitAllSettings, encargado de la lectura del fichero de configuración, ha sido ampliado para abrir la conexión con la base de datos, obtener el diccionario almacenado en esta con las equivalencias de los campos en PCSP y finalmente obtener las configuraciones de los usuarios desde el fichero almacenado en GoogleSheets.

En el caso del diccionario de equivalencias se crea un diccionario de texto, texto que se almacena en el diccionario de configuración.

En el caso de la lectura del fichero de configuración de usuarios en línea, dependerá de la página que leamos.

La página Settings contiene algunos datos a incluir en el diccionario de configuración inicial, al que serán añadidos.

La página DistributionLists contiene una tabla con tema de búsqueda, usuario de envío y palabras claves de la búsqueda, por lo que la almacenará como tabla.

Finalmente la página ExecutionControl contiene una tabla que indicará los días de ejecución de las funcionalidades adicionales de envío de notificación y revisión de expedientes ya almacenados sin fecha de presentación, así pues se almacenaran en el diccionario inicial dos listas de números enteros que contendrán los días en función de de la propiedad DateTime.DayOfWeek.

Day	checkExpedientsWithoutDate	NotificationSent
L	x	x
M		
X		
J	x	x
V		
S		
D		

Finalmente y ya aplicable a todas las veces que se llegue a este estado, se inicializan las aplicaciones necesarias. En nuestro caso se realiza el acceso a la PCSP.

Se accede mediante Chrome, accediendo a la dirección indicada en la configuración.

Maximizamos la pantalla y navegamos mediante click al buscador de licitaciones.

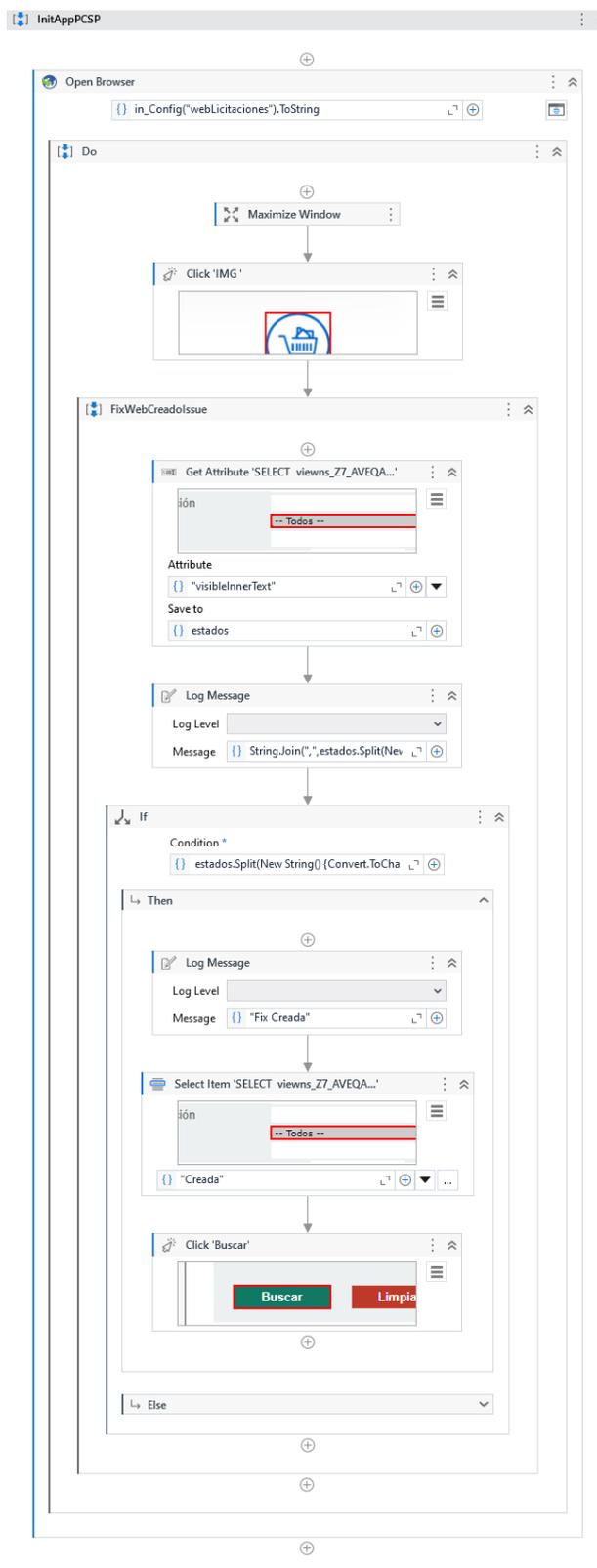
Ya en esta página, obtenemos las opciones del desplegable de estados mediante su atributo “visibleInnerText” y en caso de que aparezca el estado “Creada” se realizara una búsqueda con este.

Esto se realiza tras encontrar que la aparición de dicho estado enmascara otros dos estados que si son necesarios para la ejecución y que tras buscar por el mismo, este desaparece dando paso al resto.

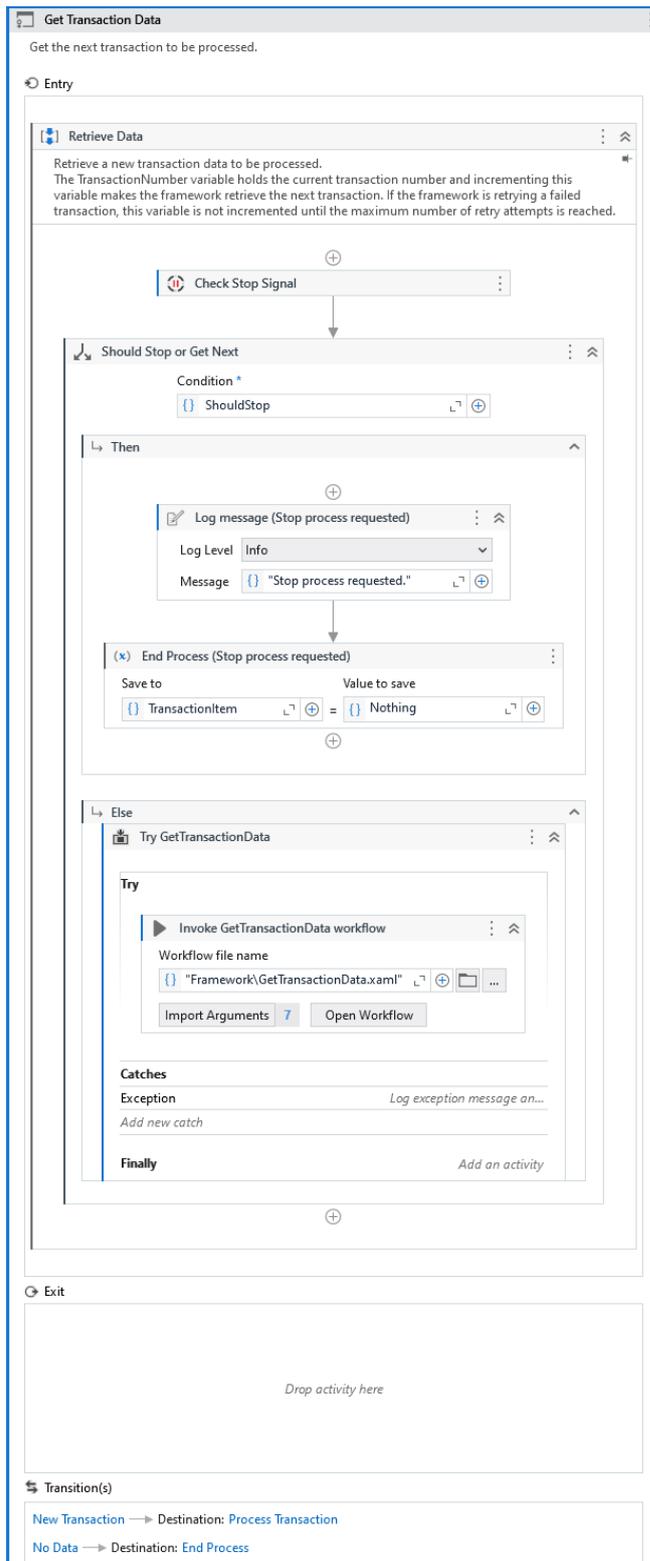
Tras dicho punto hemos inicializado la aplicación correctamente y podemos proceder con la descarga de transacciones y su posterior lectura y actualización en la base de datos.

Si todos estos pasos se realizan correctamente el proceso continuara a GetTransactionData, mientras que si un fallo aparece finalizaremos accediendo a EndProcess.

Los posibles fallos que podemos encontrar serían un fallo en la conexión a internet, bloqueando nuestros intentos de conexión a Google Sheets y a Chrome, así como un fallo en la página web que accedemos, como podría ser una caída del servidor o un mantenimiento del mismo o de la página en sí.



### 7.1.2 Get Transaction Data



El objetivo de este estado es obtener la transacción a ser ejecutada. Inicialmente este está preparado para la lectura desde una cola del orquestador (Normalmente respaldado por otro proceso, siendo estos dos los robots conocidos como Dispatcher y Performer), en cuyo caso simplemente revisa si existe alguna transacción en cola.

Sin embargo este proceso esta preparado para, en lugar de utilizar una cola, utilizar una tabla que será descargada desde este punto, y utilizar filas de tabla como su ítem de transacción, en lugar de utilizar ítems de cola.

Así pues se modifica la obtención de un ítem de cola de orquestador por la iteración en la tabla de datos, así como se añade la descarga de la misma desde la PCSP.

Además el estado revisa si se hubiera enviado una señal de parada al robot para dejar de procesar transacciones y acabar la ejecución, pasando al estado final de manera controlada.

Es en este punto en el que se añade la lectura de la tabla en caso de no estar inicializada su variable.

Si es así accedemos al fichero "PCSP\getTransactionsPCSP.xaml", el cual necesitara un argumento de tipo texto que le indicará cuáles serán las tablas en base a los estados indicados en el fichero de configuración, su otro argumento será de salida, inicializando la tabla de licitaciones a leer.

La idea de este fichero es obtener tablas de licitaciones a las que acceder posteriormente para la lectura de datos, para ello y desde el buscador

realizará búsquedas en base a los estados indicados y extraerá la información estructurada que este devuelva, navegando por las diferentes páginas que pueda mostrar el resultado de la búsqueda.

Para ello también hará uso del fichero "PCSP\filterLicitaciones.xml", que se encargará de rellenar el buscador y comprobar que aparezcan datos.

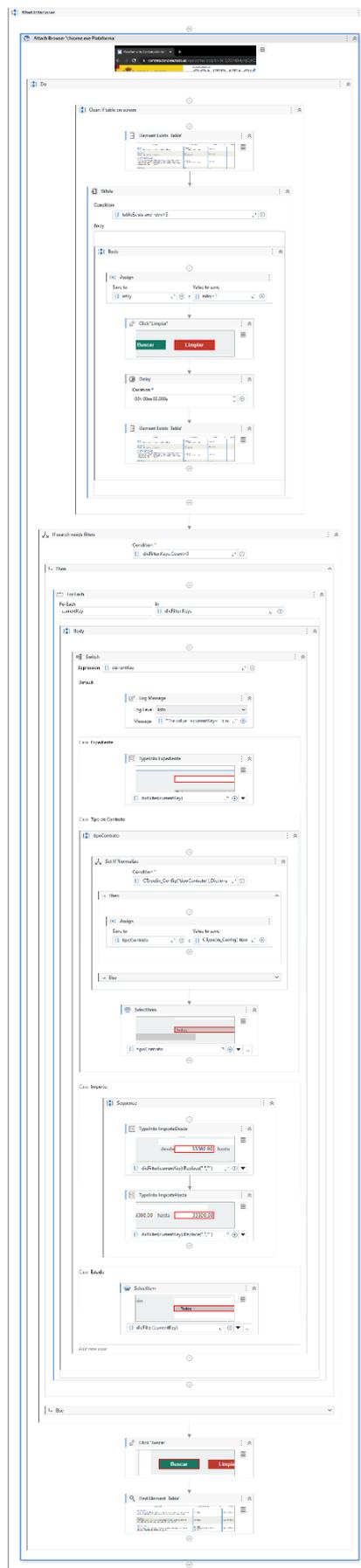
En este inicialmente se revisa si el elemento que contiene la tabla de resultados existe en pantalla, en cuyo caso procede a limpiar la tabla, reintentándolo hasta 3 veces y con un tiempo de espera de 3 segundos entre cada intento, lo que además de eliminar la tabla de resultados limpia todos los campos del buscador, para así poder realizar una búsqueda sin ningún dato cruzado de una búsqueda anterior.

Posteriormente procede a iterar sobre el diccionario de filtrado, proveniente de un argumento de entrada, en el que escribirá los valores o elegirá el valor esperado de los desplegables. Los campos permitidos son Expediente, Tipo de Contrato, Importe (Desde y Hasta) y Estado, utilizando únicamente el campo estado en su búsqueda de las licitaciones para su descarga inicial y el resto a la hora de la búsqueda transaccional de cada licitación identificada inicialmente. Cabe destacar que en el caso del tipo de contrato se utilizará un diccionario para identificar el posible valor, ya que existen diferencias en algunos valores entre el texto obtenido del resultado de la búsqueda y el campo del desplegable del buscador.

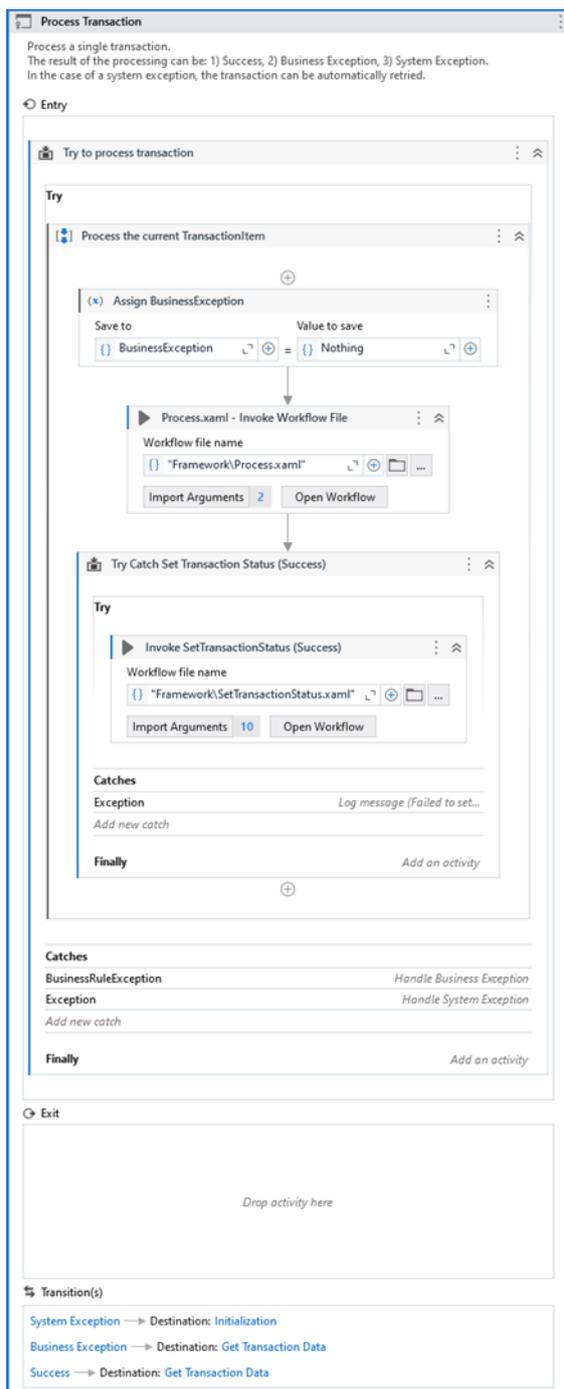
Finalmente hará click en el botón "Buscar" y procederá a buscar el elemento de la tabla resultante de la búsqueda, que esperará hasta 30 segundos hasta conseguirlo.

Así pues de vuelta a nuestro fichero "getTransactionsPCSP.xml", almacenaremos todos los resultados de las diferentes búsquedas según el estado de las licitaciones y lo almacenaremos en el argumento de salida out\_dtLicitaciones, el cual será nuestra DataTable de referencia sobre la cual iteraremos y realizaremos las búsquedas y extracción de información de cada licitación, almacenándolo en nuestra base de datos.

Todo esto se realizará en el siguiente estado, el Process Transaction.



### 7.1.3 Process Transaction



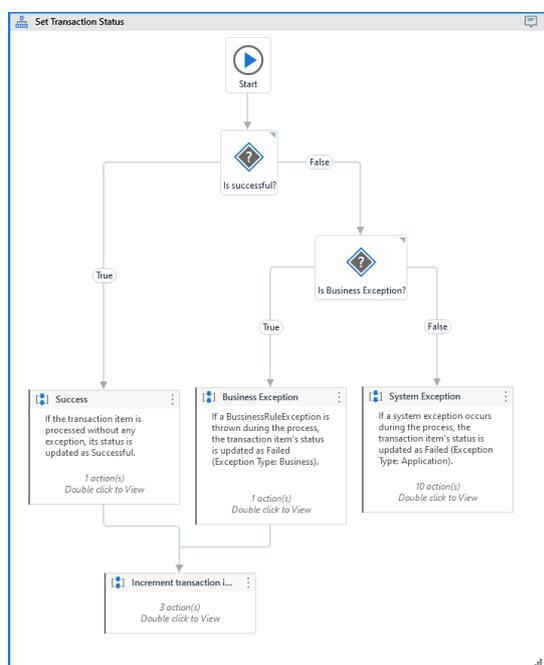
Este estado contiene el grueso de la ejecución, reconociendo en el la llamada al fichero “Framework\Process.xaml” en el que se encuentra la secuencia de actividades que decidirá el funcionamiento de cada transacción.

A su vez encontramos en este estado una gestión de errores y la llamada al “Framework\SetTransactionStatus.xaml”, que establece el estado de la transacción en función al resultado de esta.

Los tres posibles estados son: procesado correctamente, excepción por regla de negocio o excepción desconocida.

Este archivo se encarga de pasar a la siguiente transacción o, en caso de dar una excepción desconocida realizar una captura de pantalla y revisar si debe reintentar la transacción actual.

Además ha sido modificado para realizar una escritura en Excel de cada transacción, para tener ficheros de control diarios.



De vuelta al “Framework\Process.xaml”, encontramos el flujo general en el que realizamos una consulta a la base de datos en base al expediente y el órgano contratante de la transacción.

En principio esta combinación debería resultar en un identificador único, así pues si apareciera más de un caso en la base de datos se marca la excepción como una excepción por regla de

negocio. Posteriormente tenemos dos posibilidades que es acceder al caso para recoger la información relativa a la licitación o saltar a la siguiente transacción.

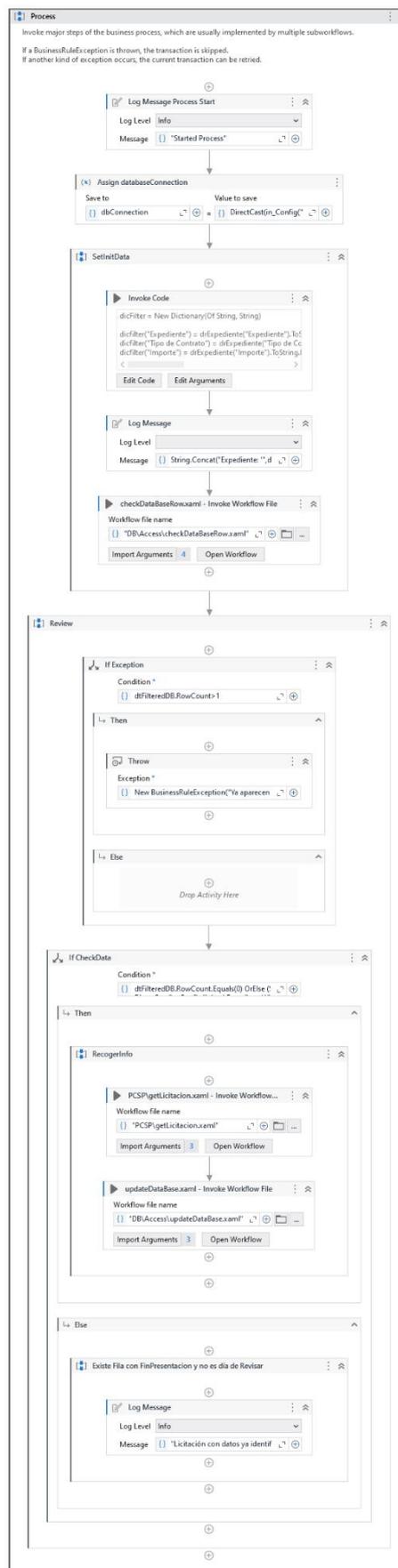
Esto se decidirá en base a si se obtiene algún caso de la query en la base de datos y de si, en caso de obtener alguno, el campo “FinPresentación” de este viene informado y el día de la semana. Así pues si no existe ningún caso se recogerá la información, si hay alguno se revisará que la fecha fin de presentación de la licitación este ya indicada ya que, en caso de estarlo la licitación ya fue recogida de forma total. En caso de que exista y el campo en cuestión no este relleno significa que aunque la licitación ya fue recogida en anteriores ejecuciones, la información aún era parcial, pues se encontraría en algún estado previo a la presentación de la misma. En este caso se revisará si el día de ejecución es uno de los indicados en los documentos en línea con los usuarios, si esto fuera así se revisará la licitación para actualizar el caso encontrado en base de datos, si no saltaremos a la siguiente transacción.

En caso de recoger la información, ubicándonos en el fichero "PCSP\getLicitacion.xaml", filtraremos en el buscador mediante el fichero "PCSP\filterLicitaciones.xaml", el cual ya ha sido utilizado en el Get Transaction Data. Lo único que cabe la pena indicar es que en este caso se filtrará mediante expediente, tipo de contrato e importe. La decisión de no hacerlo en base al expediente y órgano contratante (definidos previamente como combinación de campos de identificador único) es porque el buscador de la página puede mostrar valores diferentes para el órgano contratante entre el campo en la lista de licitaciones y en el detalle de la licitación, resultando en búsquedas vacías si este era utilizado. Al poder contar con diferentes tipo de contrato y este ser un campo desplegable (de elección única) también conllevo en errores en la búsqueda durante las pruebas de impacto.

Tras filtrar accederemos a la línea con el expediente en el título (ya que se ha podido comprobar que a veces, con importes genéricos pueden aparecer varias líneas en las que el expediente es parecido o el indicado aparece parcialmente en el buscado).

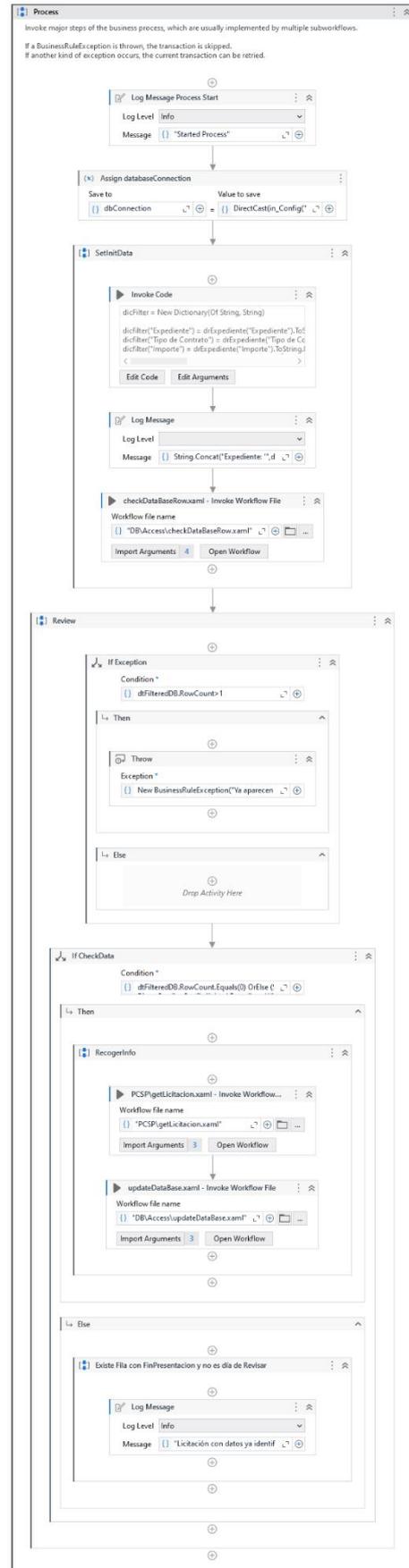
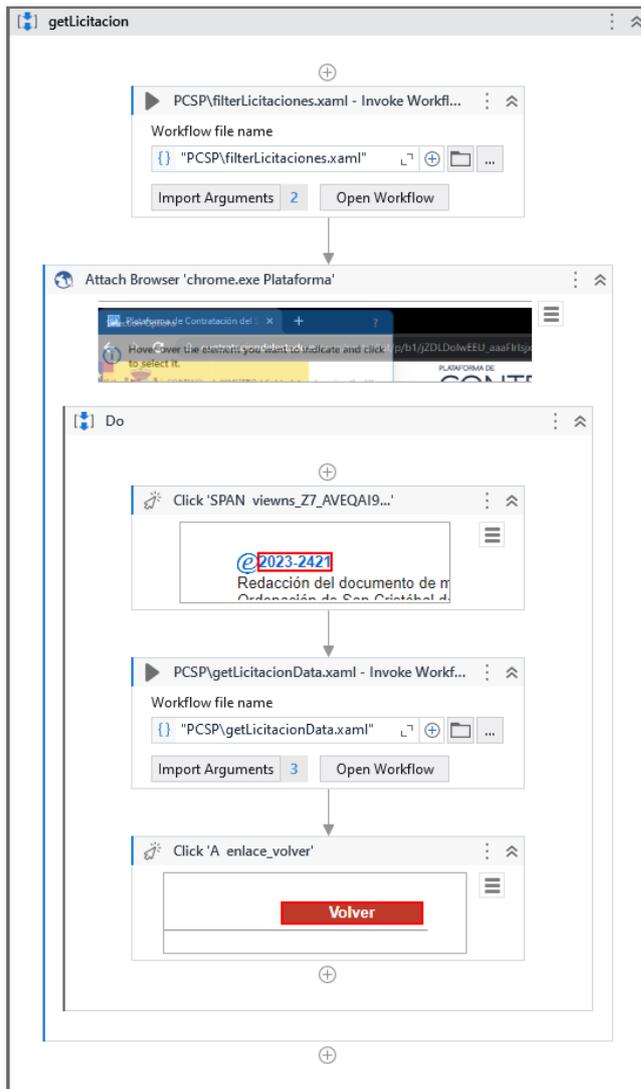
En este punto y accediendo a "PCSP\getLicitacionData.xaml" se almacena la información de la licitación en un diccionario, guardando el expediente y el tipo de contrato de la búsqueda inicial, mientras que el resto de campos se recogen en el detalle de la licitación.

Esto se debe a razones de continuidad y coincidencia en la búsqueda en la base de datos para el expediente y

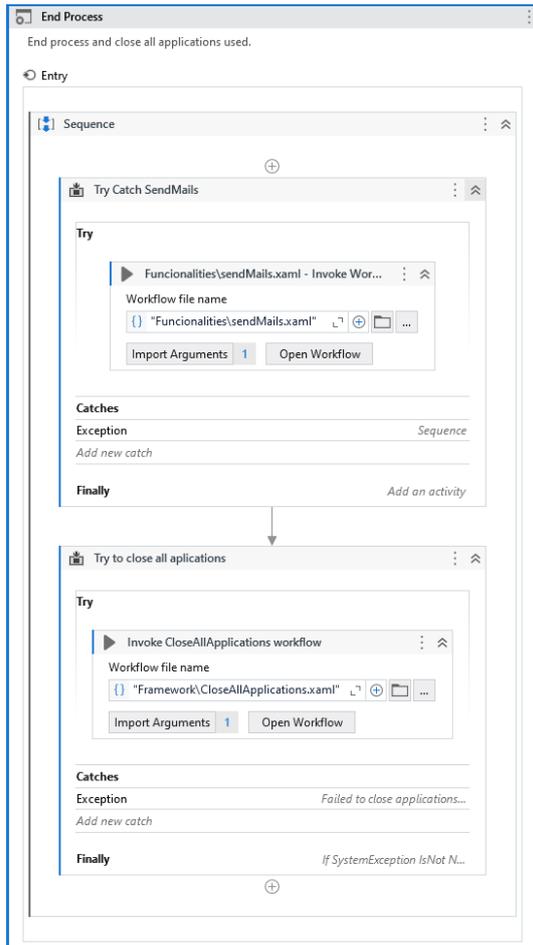


porque el tipo de contrato no aparece en esta pantalla.  
El resto de campos se recogen de la pantalla en cuestión.

Finalmente y ya volviendo a  
"PCSP\getLicitacion.xml" se hace click en el botón  
volver para acceder de nuevo a la ventana del buscador.



### 7.1.4 End Process



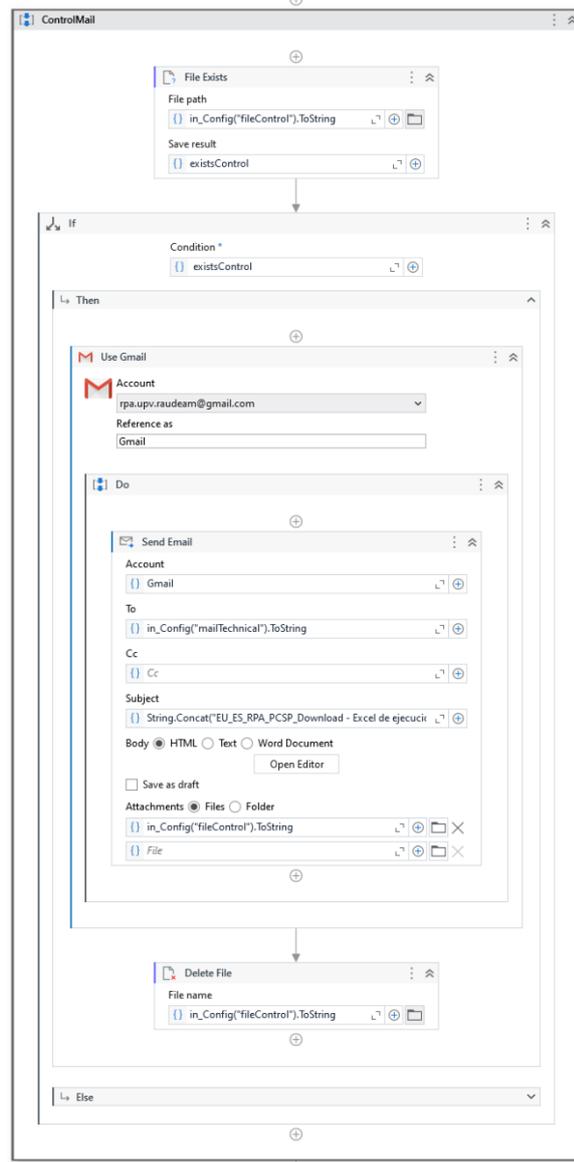
El envío del fichero de control se realiza en cada ejecución, enviando dicho fichero excel escrito en el SetTransactionStatus al correo del responsable técnico, y borrándolo después de la carpeta interna.

Respecto al envío de notificaciones y borrado de registros se revisará si el día es uno de los identificados para dicha tarea desde la configuración en línea del usuario. Si así es se iterará sobre la tabla de notificaciones, se hará una consulta a la base de datos en función de unas palabras claves definidas para cada tema y se filtrará dicha búsqueda en función de las fechas de lectura del registro. Si el registro ha sido recogido por primera vez desde la última notificación o ha sido actualizado (lectura en base a un registro que no contaba con fecha fin de presentación previamente) el registro aparecerá en la notificación a usuarios.

Este estado finaliza el proceso, en el framework está preparado para cerrar las aplicaciones que se han utilizado, pero hemos ampliado este para poder hacer las acciones necesarias al final de la ejecución, como son el envío de notificaciones y el borrado de registros de la base de datos.

Como en nuestros estados anteriores ya viene implementado con una gestión de errores, en la que si no consigue cerrar las aplicaciones de forma natural, realizará un cierre forzado de estas matando la instancia de la aplicación.

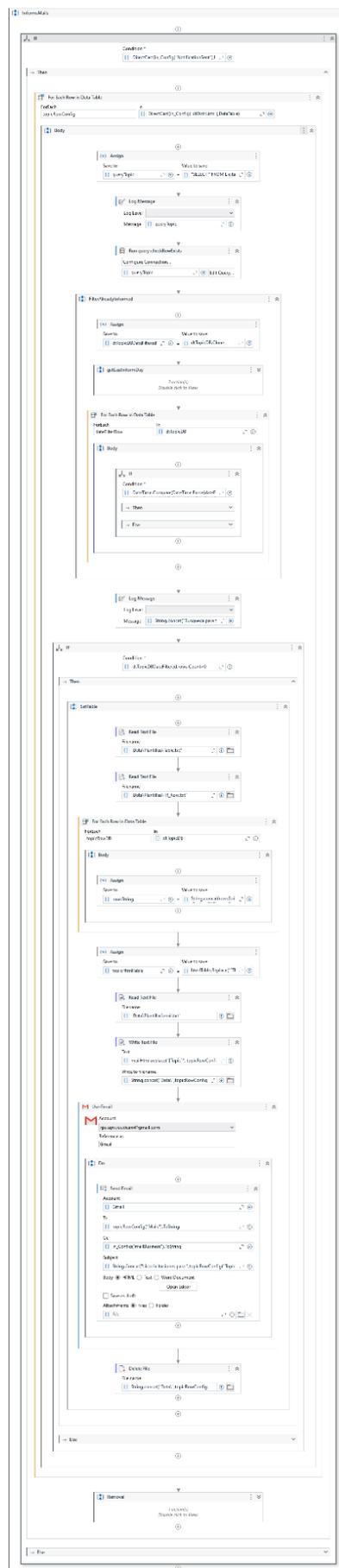
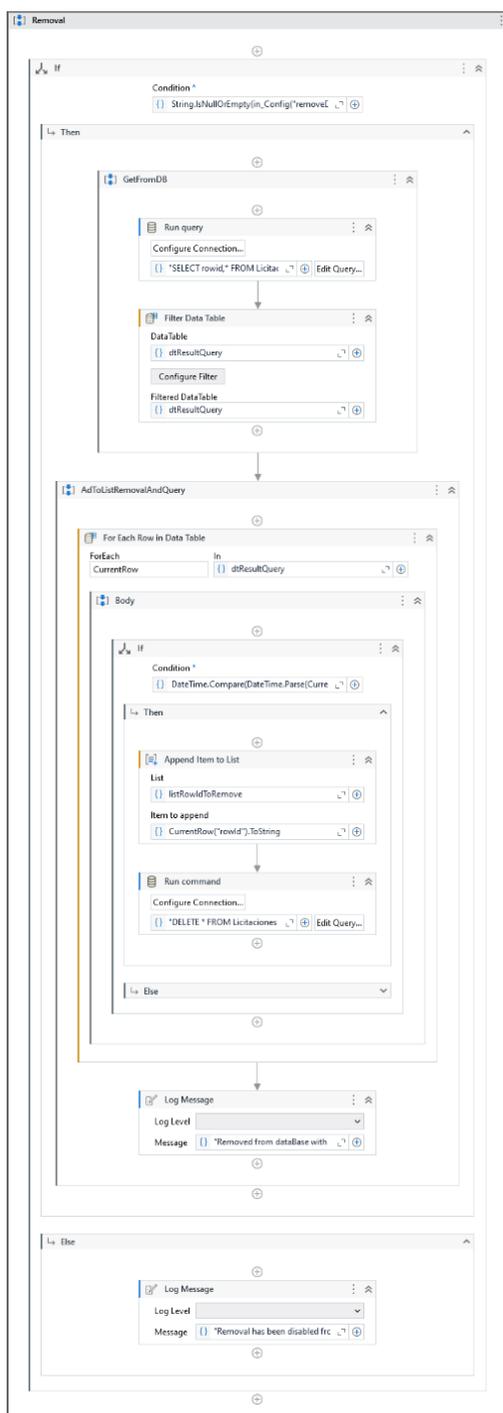
Adicionalmente se ha añadido la secuencia “Funcionalities\sendMails.xaml”, en la que nos encargamos del envío de notificaciones así como del borrado de registros que hemos comentado previamente.



Finalmente si tras la búsqueda y filtrado hay alguna licitación de la que informar, el robot realizará un envío de dicha información en un correo electrónico, en el que mostrará la tabla en cuestión.

Posteriormente y en función también de la configuración de usuario realizará un borrado de los registros cuya fecha fin de presentación haya pasado la fecha actual. También se puede controlar el tiempo sobre el que se realizará el borrado, dando un plazo de ciertos días indicado en dicha configuración.

Finalmente el proceso cerrará el navegador, así como la conexión con la base de datos.





## **7.2 Anexo II – Pliego del proyecto RPA**

**DESCARGA DE LICITACIONES DE LA PLATAFORMA DE  
CONTRATACIONES DEL ESTADO**

## 1. Objetivos del proyecto

Este proceso pretende reducir el esfuerzo en la búsqueda de licitaciones, así como mejorar la regularidad con la que se realizaba dicha tarea.

El proceso se ejecutará todos los días, revisando las licitaciones que se encuentran en la web “Plataforma de contratación del sector público” y actualizará la base de datos, incluyendo las nuevas licitaciones, actualizando en los casos necesarios y obviando en caso de haber concluido con la recogida de datos.

Posteriormente en los días indicados en la configuración se enviarán reportes con la información recogida durante el periodo previo de recogida desde la última notificación.

La implementación de este proyecto pretende liberar a los usuarios de esta tarea, mejorar la eficiencia así como definir un proceso normalizado y estructurado en la revisión de licitaciones.

## 2. Proceso a automatizar

El proceso a automatizar es la búsqueda de licitaciones en base a ciertos parámetros en la página web “Plataforma de Contrataciones del estado”, así como el envío de informes regulares en base a los datos encontrados y los términos de búsqueda indicados por los usuarios.

### 2.1 Aplicaciones involucradas

Se definen las siguientes aplicaciones sobre las que el proceso trabajará:

#### 2.1.1 Chrome

Se utilizará Chrome como el navegador de referencia con el que acceder a la “Plataforma de Contrataciones de Estado”

#### 2.1.2 Access

Se define Access como la base de datos en la que se almacena la información encontrada en la plataforma de contrataciones del estado

Además de la información sobre las licitaciones, también se definirá una tabla diccionario, en la que encontrar las relaciones entre un campo y sus posibles formatos en los formularios de la página web.

Las tablas definidas serán las siguientes:

- **Licitaciones**  
Tabla en la que se almacenará la información referente a las licitaciones sobre las que itera el proceso, así como información relacionada, definida para la gestión de estas a la hora del envío/iteración.

- **Normalizado**

A la hora de navegar en la web encontraremos caso en los que, a pesar de hacer referencia al mismo campo, el texto mostrado no es el mismo, así pues definimos esta tabla en la que se incluirán los diferentes posibles valores que el robot utilizará para encontrar la coincidencia correcta.

En esta se definen tres columnas. Campo nos indica sobre el campo en el que estamos trabajando y hará referencia a la variable con la que el robot trabaja, valor es cada posible texto que podamos encontrar y normalizado será el valor normalizado de cada posible texto.

Nombre del campo	Tipo de datos
rowid	Autonumeración
Expediente	Texto corto
OrganoContratante	Texto corto
TipoContrato	Texto corto
Estado	Texto corto
ObjectoContrato	Texto largo
PresupuestoSinImp	Número
Presupuesto	Número
LugarEjecucion	Texto corto
Procedimiento	Texto corto
FinPresentacion	Fecha/Hora
EnlaceLicitacion	Hipervínculo
fechaPrimeraLectura	Fecha/Hora
fechaActualizacionPresentacion	Fecha/Hora

Id	campo	valor	normalizado
----	-------	-------	-------------

### 2.1.3 Google

Se define Google Sheets como una plataforma clave para la gestión de la información entre los usuarios y el proceso RPA. En este se almacenarán ficheros para la configuración que podrán ser modificados por los usuarios de negocio permitidos e impactarán directamente en la ejecución del proceso.

Además el proceso realizará envíos de información recurrente en base al contenido de la base de datos. Se utilizará Gmail para realizar dicha tarea.

## 2.2 Ficheros utilizados

### 2.2.1 Fichero config interno

Es el fichero excel ya implementado en la base del ReFramework (framework para automatizaciones de UiPath) en el que se almacenarán constantes y configuraciones que el robot utilizará en su ejecución. Se indican en este valores que son necesarios para la ejecución y sensibles a cambios a nivel de desarrollo, así pues solo serán modificados por el desarrollador en función de los cambios aplicables en próximas modificaciones sobre el flujo.

Este se encuentra en el sistema de carpetas del robot y es volcado en un diccionario de texto, objeto.

### 2.2.2 Fichero config en línea

Así como su contraparte, este fichero de configuración será un fichero Excel en el que se podrán indicar ciertos valores. Sin embargo este se encuentra en línea, almacenado en un drive al que se le otorgará acceso a los responsables del proceso, y que contendrá información actualizable, de esta forma se podrán modificar ciertos parámetros sin necesidad de modificación del propio proceso.

Valores que podemos encontrar en este fichero son listas de distribución, valores de filtrado, días de envío y demás información relevante para los usuarios.

### 2.2.3 Ficheros de mail con formato HTML

Se preparan ficheros txt con el formato esperado por los usuarios para el envío de correos.

Estos tendrán tanto el cuerpo del correo como las tablas en las que se verterá la información de las licitaciones.

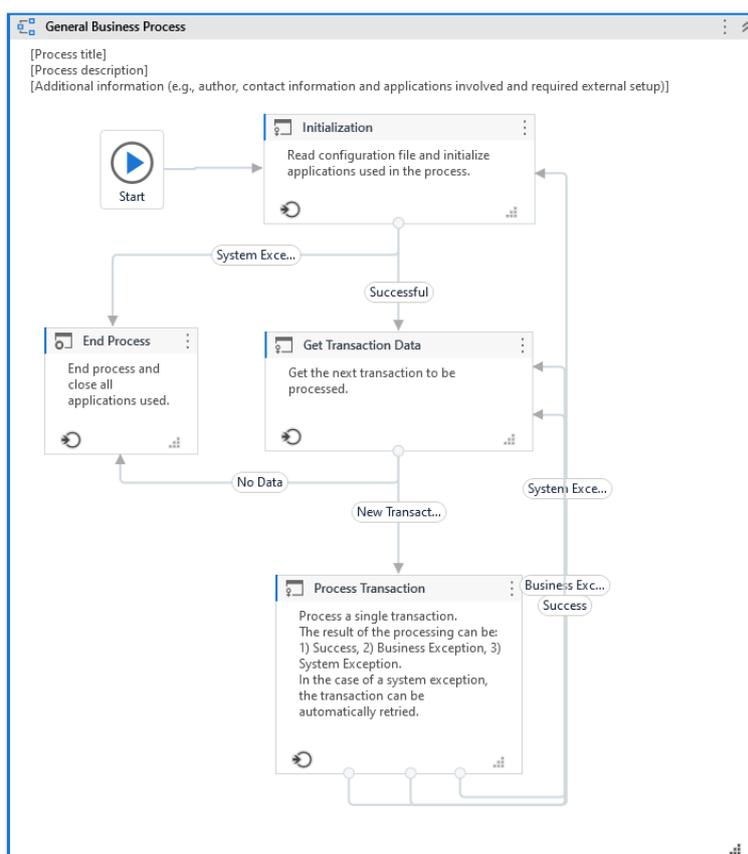
### 2.2.4 Fichero Access

Se utiliza como base de datos para almacenar la información sobre las licitaciones y tablas con diccionarios para normalizar los campos encontrados en la web a automatizar.

## 2.3 Flujo del proceso

El proceso accederá a la web “Plataforma de contratación del sector público”, de la que recogerá información de las licitaciones abiertas y la almacenará en una base de datos, para así poder configurar avisos que serán enviados mediante correo electrónico.

Se utilizará el ReFramework de UiPath como base para la automatización. Este nos provee con un framework preparado con estados predefinidos, control de errores y reintentos. Sobre este se ha desarrollado el flujo que el proceso seguirá y veremos a continuación.



### 2.3.1 Initialization

El inicio del proceso es en el que se leen los ficheros de configuración y se inicializan las aplicaciones que van a ser usadas.

Es en este punto es en el que leemos los ficheros de entrada y configuración comentados en el punto anterior, a excepción del fichero Access pues será actualizado a lo largo del proceso y servirá tanto de entrada como de salida.

También se matan las aplicaciones a utilizar para prevenir cualquier posible estado previo y se inicializan. En nuestro caso accederá a la web “Plataforma de contratación del sector público”, entrando al formulario de licitaciones.



En este formulario se ha detectado que inicialmente puede aparecer un estado llamado “Creado”, el cual no devuelve ninguna licitación y hace que no aparezcan algunos estados utilizados. Sin embargo al buscar por este, aparecen el resto de opciones y a su vez desaparece esta.

Así pues aprovechamos en este punto para realizar la búsqueda de dicho estado y preparar la página para que el proceso se ejecute con un funcionamiento normalizado.

### 2.3.2 *Get Transaction Data*

Este estado recoge la información sobre la que iterará el proceso, inicialmente esta preparado para utilizar las colas del orquestador de UiPath pero se ha modificado para trabajar con una DataTable. Es en la primera iteración con este estado en el que recogeremos la información de la página web, mientras que en las siguientes servirá para iterar sobre la tabla.

Para recoger la información se extraerá la información estructurada que se encuentra al buscar en el formulario, debido a la cantidad de registros que aparecen al buscar sin filtrar y que muchos de estos no aplicarían al objetivo del proceso se decide filtrar por el campo “Estado”, iterando entre los estados permitidos en el fichero de configuración y extrayendo la información devuelta.

Formulario de Búsqueda  
Licitaciones

Expediente:  País:

Tipo de Contrato:  Lugar de Ejecución:

Código CPV:  Añadir Selección CPV

CPV Seleccionados:  Quitar

Organización contratante:

Nombre O. Contratación:  Presentación:

Estado:  Anuncio Previo:  Procedimiento:

Adjudicatario:  Fecha publicación entre:

Importe: desde  hasta

Buscar Limpiar

Búsqueda avanzada (Solo Perfiles del Contratante en la Plataforma del Sector Público)

Expediente	Tipo de Contrato	Estado	Importe	Presentación	Órgano de Contratación
0923ETHC0004I0000062E Servicio de fisioterapia en la Base El Goleto	Concesión de Servicios	Anuncio Previo	0.00		Jefatura de Intendencia de Asuntos Económicos Centro
1124/2023 Renovación de la electrónica de red del sistema de mando y control nacional (CCN) para los nodos de la Armada	Servicios Servicios de mantenimiento y reparación	Anuncio Previo	482 194.31		Dirección de Gestión Económica de la Jefatura de Apoyo Logístico de la Armada
168/2023 Reposición de beiramaús del Centro Médico á Estrada da Feira	Obras Construcción de autopistas, carreteras, campos de aterrizaje, vías férreas y centros deportivos	Anuncio Previo	50 946.46	23/06/2023	Junta de Gobierno Local del Ayuntamiento de Moeche
1146/2023 Adquisición equipamiento de red para renovar el NODO FMN de SPMARFOR en los nodos desplegables de los buques.	Servicios Servicios de transporte por vía terrestre, incluidos los servicios de furgones blindados y servicios de mensajería, excepto el transporte de correo	Anuncio Previo	260 851.20		Dirección de Gestión Económica de la Jefatura de Apoyo Logístico de la Armada
1128/2023 Suministro de equipos respiratorios con destino a los buques de la Armada	Suministros Adquisición	Anuncio Previo	1 543 550.00		Dirección de Gestión Económica de la Jefatura de Apoyo Logístico de la Armada

### 2.3.3 *Process*

Tras la recogida de la tabla en el GetTransactionData, se iterará a través de todas las filas. El procesamiento de cada una de estas se realizará en este estado.

En nuestro caso se seguirán los siguientes pasos:

#### 2.3.3.1 *Búsqueda de registro en base de datos*

Se realiza una búsqueda en la base de datos para ver si el expediente ya existe en esta.

Al no ser el nombre del expediente un valor único se opta por identificar mediante dicho valor combinado con el órgano contratante.

#### 2.3.3.2 *Gestión según existencia*

Tras la búsqueda del registro en base de datos y en función del resultado de esta se realiza la búsqueda en la web, accediendo al expediente en cuestión.

Las tres posibilidades son:

- **No existe registro**  
Es un expediente nuevo, así pues se realiza la búsqueda, accede al expediente y procede a recoger los datos de la licitación.

Expediente: Asimpl-13-2023

ENTIDADES LOCALES>Comunidad Valenciana>Alicante>Diputación Provincial de Alicante

Órgano de Contratación	Presidencia del Organismo Autónomo de la Excma. Diputación Provincial de Alicante Instituto Alcantino de Cultura Juan Gil-Albert
Estado de la Licitación	Publicada
Objeto del contrato	Servicio agencia de viaje para la asistencia y organización de acciones culturales programadas por el Instituto Alcantino de Cultura Juan Gil-Albert.
Presupuesto base de licitación sin impuestos	16.200,00 Euros
Valor estimado del contrato:	48.600,00 Euros
Tipo de Contrato:	Servicios
Código CPV	63510000-Servicios de agencias de viajes y servicios similares.
Lugar de Ejecución	España - Alicante/Alacant - Alicante
Procedimiento de contratación	Abierto simplificado

#### Información

Fecha fin de presentación de oferta	16/06/2023 15:00
-------------------------------------	------------------

#### Resumen Licitación

Publicación en plataforma	Documento	Ver documentos
04/06/2023 17:53:59	Anuncio de Licitación	<a href="#">Html</a> <a href="#">Xml</a> <a href="#">Pdf</a> <a href="#">Sello de Tiempo</a>
04/06/2023 17:55:04	Pleigo	<a href="#">Html</a> <a href="#">Xml</a> <a href="#">Pdf</a> <a href="#">Sello de Tiempo</a>

Enlace a la licitación

<https://contrataciondeestado.es/wps/wcm/connect/2fXIPmpxkZPzrdqOdhWq%3D%3D>

Tras la recogida de estos valores vuelve a la pantalla del buscador e inserta una nueva fila en la base de datos con los valores recogidos y la fecha de inserción del registro para el posterior informe.

- **Existe registro sin fecha de presentación**

Esta situación se da cuando, con anterioridad, hemos recogido la información de un expediente pero en este no aparecía la fecha fin de presentación de la oferta.

En este caso el proceso contará con ciertos días, gestionados desde los ficheros de configuración en línea, en los que accederá de nuevo al expediente y actualizará la información que se encuentra en la base de datos.

En caso de que la ejecución no coincida con las fechas de revisión designadas, el proceso saltará al siguiente expediente.

- **Existe registro con fecha de presentación**

Esto significa que el expediente ya ha sido recogido por el robot y este no debería ser actualizado, así pues saltamos al siguiente caso sin realizar ninguna actualización en la base de datos.

### 2.3.4 End Process

Finalmente, tras haber recorrido todos los expedientes encontrados en la búsqueda inicial, el robot procede a cerrar las aplicaciones utilizadas. Tras esto solo queda el envío de las notificaciones en base a los datos indicados en los ficheros de configuración en línea.

En caso de que el día de ejecución coincida con los días de envío de notificaciones el proceso procederá a realizar las notificaciones por correo electrónico. Para esto revisará las listas de distribución, que contarán con destinatarios y elementos de búsqueda, y enviará un correo por cada categoría, en el que enviará todos los expedientes cuya fecha de lectura o fecha de actualización de la presentación se encuentre entre dicha y su última fecha de ejecución.

En estos días el robot también revisará la base de datos para borrar aquellos registros cuya fecha de fin de presentación haya pasado tras un tiempo indicado en los ficheros de usuario.

Finalmente y para todas las ejecuciones se realizará un envío de un Excel a los encargados técnicos. En este se encontrarán los resultados de cada transacción, indicando si esta ha sido ejecutada con éxito, si hay alguna excepción de negocio o si ha habido un fallo, en cuyo caso también se indicará la captura de pantalla realizada.

### 3. Alcance del proyecto

Se ha realizado un análisis inicial del proyecto y se ha optado por contar con un desarrollo prudencial, en el que se realizarán las tareas básicas de recogida de información e informes detallados de estas, así como el envío de información y métricas por correo.

En función de las métricas de éxito del proyecto se plantean las siguientes ampliaciones:

- Actual búsqueda restringida a “Licitaciones”, ampliar para buscar en “Contratos Menores” y “Encargos a medios propios”.
- Modificación del envío de métricas actual para la utilización de un panel en Kanban.

Actualmente hay una persona dedicada parcialmente para realizar dicha tarea, esta será la encargada a nivel de negocio de revisar la información, así como informar al equipo en caso de necesidad de ampliaciones, correcciones o ejecuciones adicionales.

### 4. Recursos

Los recursos necesarios para la implementación de este proyecto recaerán directamente sobre el equipo RPA, entre estos podemos encontrar recursos de infraestructura como son:

- Orquestador  
Entorno para todos los procesos en el que se tendrá control sobre las versiones subidas, ejecuciones realizadas y *triggers* de los procesos.
- Máquina virtual de Azure  
Máquina en la que se ejecutará el proceso, esta será propiedad del equipo RPA y será compartida con diferentes procesos y áreas de la empresa. En caso de necesidad de una máquina individual para este proceso o los procesos de un departamento, se permite el disponer de una máquina dedicada para ello, pero los gastos de esta serán repercutidos al departamento en cuestión. Cabe destacar que aun contando con una máquina dedicada, el control de esta será realizada por el equipo RPA, así como se verá supeditada a las normas dispuestas por estos.

Entre las aplicaciones o ajustes necesarios para esta encontramos:

- Licencia de Microsoft (Excel y Access)
- Chrome

En este caso todas las aplicaciones necesarias se encuentran entre las aplicaciones base definidas en los estándares, así que no existe necesidad de instalación de ninguna aplicación por parte de este proceso.

### 5. Métricas de éxito

Para revisar el éxito del proceso se utilizarán KPI's del tiempo destinado a la tarea y se compararán con la dedicación previa por parte del equipo a dicha tarea.

Cabe destacar que deberá ser calculado el desempeño actual del robot a la tarea actual, es decir, cuanto tiempo del total se dedica a la tarea original, pues el proceso inicial a automatizar ha sido ampliado en la implementación.

Así pues también se revisarán los casos ejecutados correctamente y el número de fallos, tanto a nivel de transacción como a nivel de ejecución completa. Esto nos permite conocer si la aplicación automatizada es dada a cambios que impacten en los selectores del robot, así como en el flujo del proceso.

Finalmente también se tendrá en cuenta el nivel de satisfacción del departamento que recibe el respaldo del RPA.

## 6. Plan de implementación

Se define el siguiente cronograma para ver el tiempo destinado a la implementación del proyecto, así como las personas responsables de cada una de las etapas del caso de uso.

Tareas	Responsable	Dias	Febrero							Marzo							
			6	7	13	14	20	21	27	1	6	7	13	14	20		
Planificación																	
Análisis y definición proceso	Script Owner	3	■	■	■												
Reunión inicial	Script Owner	0	■														
Análisis																	
Documentación proceso	Script Owner	0,5		■													
Análisis viabilidad	Scripter	0,2		■													
Diseño proceso	Scripter	0,2		■													
Desarrollo																	
Desarrollo	Scripter	13,2			■	■	■	■	■	■	■	■	■	■	■	■	■
Pruebas unitarias	Scripter	2			■	■											
Documentación script	Scripter	0,375															
Mejoras y arreglos	Scripter	2															
Pruebas																	
Pruebas exhaustivas	Scripter	0,5															
Validación resultados	Script Owner	2															
Comprobación técnica	RPA Engineer	0,5															
Implementación																	
Preparación entorno	Admin RPA	0,25															
Configuración de programación	RPA Engineer	0,175															
Monitoreo inicial	Scripter/ SO	2															

Cabe destacar que este cronograma corresponde con los plazos en el supuesto de que las personas implicadas, principalmente el desarrollador, tengan una dedicación total al proyecto, no hayan grandes tiempos de comunicación fuera de lo común entre las partes involucradas y no surjan grandes dificultades en el desarrollo o cambios en el proceso por parte del usuario.

El tiempo de desarrollo ya contiene un plazo margen para posible gestión de errores o complicaciones pero hay casos en los que este no es suficiente o necesario. El importe final será computado en base a la dedicación real y no al análisis inicial.

## 7. Plan de mantenimiento y soporte

A partir del punto de la puesta en producción, pasado el plazo de monitoreo inicial y tras la comprobación del correcto funcionamiento del script, el mantenimiento del robot será realizado por el Business Owner y Script Owner.

Estos serán, a través de los correos de estado y las lecturas que puedan hacer desde el orquestador, los encargados de revisar y evaluar el correcto funcionamiento del robot.

A su disposición se encontrará el equipo de ingenieros y administradores RPA, a los que podrán contactar en cuestión de ruegos o dudas. Estas son algunas de las funciones que tienen dichos equipos:

- **Ingenieros RPA**  
Solicitud de ejecuciones adicionales, comunicaciones en referencia a comportamiento del proceso o dudas generales, cambio en la programación del proceso, petición de grabación del comportamiento...
- **Administradores RPA**  
Arreglos de fallos en las máquinas, necesidad de instalación de software, modificación de los assets propios en el entorno de producción...

Estas tareas están libres de recargo.



En caso de que el equipo de usuarios crea necesario una modificación en el proceso, tanto por una necesidad técnica como por un cambio en el flujo o modificación del comportamiento, se deberá abrir una petición de cambio que será sujeta a un proceso parecido al desarrollo inicial, que se cobrará y deberá ser aceptada por el Script Owner.