Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Mejora al sistema de seguridad de una empresa mediante gestión de identidades

Trabajo Fin de Grado

Grado en Ingeniería Informática

Autor: Javier Cutillas Zárate

Tutor: Carlos Miguel Tavares de Araujo Cesariny Calafate

Curso 2020-21

Agradecimientos:

A mi familia por el apoyo recibido durante el desarrollo del proyecto.

A Jaime, gracias por ayudarme a llegar donde estoy.

Resumen

Esta memoria demuestra que la gestión de identidades es la base de la seguridad de cualquier empresa hoy en día. Gracias a sus numerosos beneficios pueden reducirse los riesgos de ciberataques. En este trabajo se realiza una prueba de concepto sobre un caso ficticio en la Universidad Politécnica de Valencia usando la tecnología de One Identity. El resultado es un sistema capaz de automatizar el ciclo de vida de estudiantes y empleados, y asegurar que las personas tengan los accesos y permisos adecuados.

Palabras clave: gestión de identidades, ciberataques, One Identity.

Abstract

The aim of this report is to demonstrate that identity management is the heart of any company's security system. Thanks to its many benefits, the risks of cyber-attacks can be reduced. In this work, a proof of concept is carried out on a fictitious case with the Technical University of Valencia using One Identity technology. The result is a system able to automatize the lifecycle of students and employees, and of ensuring that people have the appropriate access and permissions.

Keywords : identity management, cyber-attacks, One Identity.

Índice de Contenidos

Índice de Ilustraciones

Índice de tablas

Índice de algoritmos

Capítulo 1: Introducción

La digitalización ha transformado el modo en el que las personas interactúan. En la época actual, ésta se ha visto incrementada debido a la pandemia, afectando directamente a las empresas, las cuales se han visto obligadas a adoptar un nuevo modelo de trabajo (1).

Actualmente las organizaciones de todo el mundo utilizan cuentas de usuarios para que sus empleados puedan hacer su trabajo. Estas organizaciones varían en tamaño desde pequeñas y medianas a grandes; además, se encuentran en todo tipo de industrias (servicios financieros, salud, gobernanza, educación superior, etc). Estas cuentas de usuario forman parte de la identidad digital de un empleado.

La Real Academia de Lengua Española define el concepto de identidad como un "*conjunto de rasgos propios de un individuo o de una colectividad que los caracterizan frente a los demás*" (RAE). Teniendo esto en cuenta, podríamos definir una identidad digital como: información sobre una entidad utilizada por los sistemas informáticos para representar a un agente externo que se diferencie a los demás.

La identidad digital es la base de la ciberseguridad. Permite la gestión y el acceso basado en la autenticación y autorización en los servicios para proteger los datos y los recursos [1].

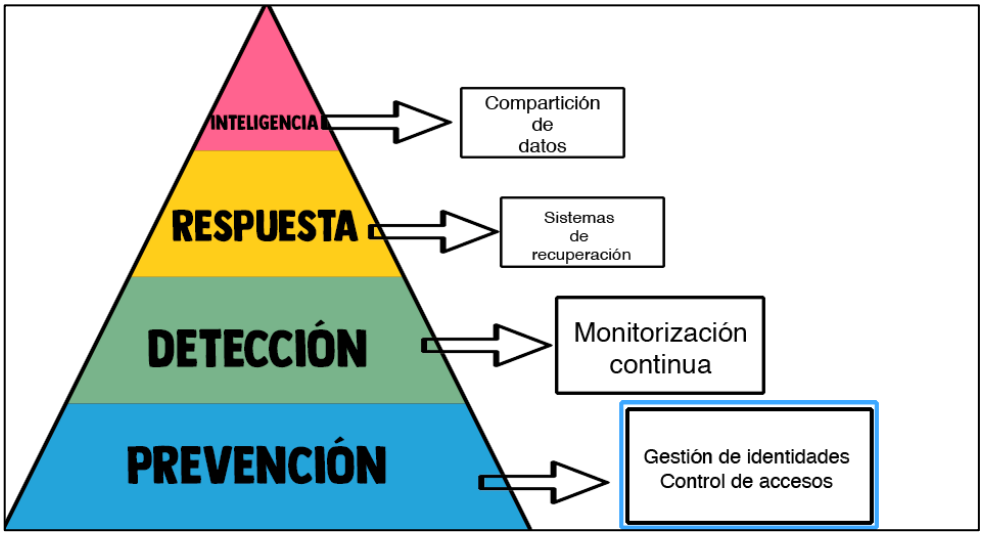

Ilustración 1: Pirámide ciberseguirdad. Fuente: Telefónica

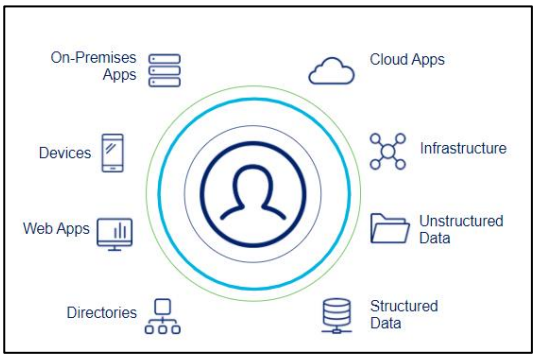

Ilustración 2: Acceso a sistemas de una identidad. Fuente: Sailpoint

Dentro de una organización, los empleados usan identidades digitales todo el tiempo ya sea en una aplicación, en un sistema, o en una red, con el fin de dar el acceso correcto a los recursos que el empleado necesita en su trabajo [2].

La IAM[1] se considera un campo de la ciberseguridad que garantiza que la información sensible sólo puede ser accedida por usuarios seleccionados en determinados momentos. Gracias a la gestión de identidades y accesos, se reduce en gran medida la posibilidad de que se produzcan infracciones y ciberataques, además de reducir el riesgo de error humano, el cual representa la principal causa de violaciones de datos [3].

1.1 Motivación

La necesidad de la creación de este proyecto nace con la transformación digital y la nueva industria, también conocida como industria 4.0 o "smart factory". El panorama tecnológico en la empresa es cada vez más complejo y heterogéneo. Para gestionar el cumplimiento y la seguridad de este entorno, la IAM permite que las personas adecuadas accedan a los recursos adecuados en el momento adecuado y por las razones adecuadas.

Además, no es solo una herramienta para la mejora de la seguridad, también puede sernos de utilidad en la creación de auditorías. Esto supone que la empresa pueda tener informes personalizados como, por ejemplo, un listado de personas que accedieron a un recurso en una determinada fecha [4].

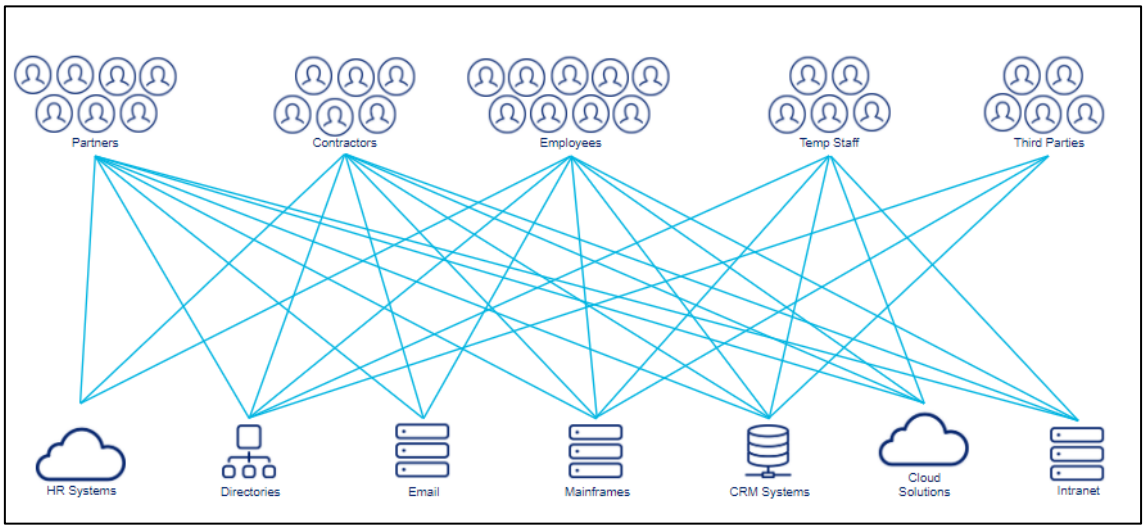

Ilustración 3: Complejidad de una empresa. Fuente: Sailpoint

1.2 Objetivos

Con este trabajo lo que se pretende conseguir es mostrar la importancia de la gestión de identidades, y enseñar como implementar esta tecnología partiendo desde el inicio, lo cual incluye:

- Analisis y Consultoría.
- Presupuesto.
- Administración de las tareas.
- Implementación.
- Pruebas.

[1] IAM : Identity and Access Management : Gestión y acceso de identidades.

Para ello realizaremos una PoC[2] (prueba de concepto) utilizando a la Universidad Politécnica de Valencia como referencia. El objetivo final es que tanto alumnos de cualquier campus o facultad y empleados puedan ser gestionados desde el mismo entorno con diferentes accesos y con un ciclo de vida automatizado.

1.3 Impacto esperado

El impacto esperado es poder aplicar todos los beneficios de la gestión de identidades a la UPV:

- **Aprovisionamiento automático de cuentas de usuario - LifeCycle Management:**

  El ciclo de vida de una identidad consta de las siguientes fases:

  1. Creación del usuario
  2. Asignación de los recursos adecuados (onboarding)
  3. Suspensión temporal
  4. Petición de nuevos recursos
  5. Desactivación del usuario (offboarding)

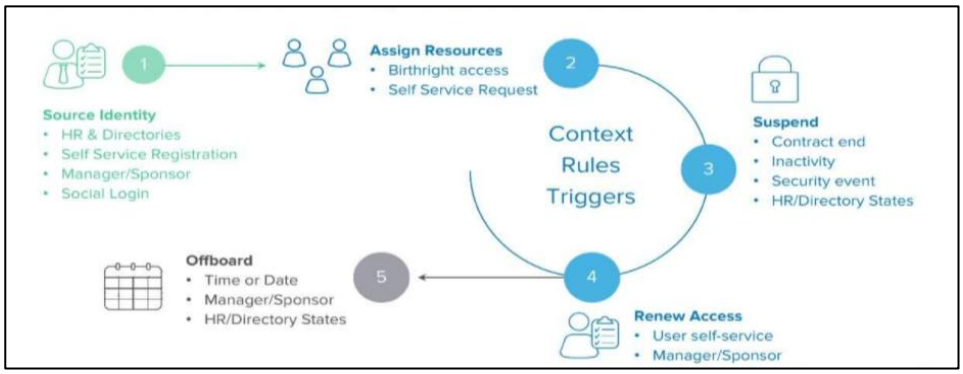

Ilustración 4: Ciclo de vida. Fuente: Okta [5]

- **Gestión de flujos de trabajo y autoservicio:** Habilitación de un portal donde los empleados de la UPV podrán realizar peticiones a ciertos recursos. Estas peticiones tendrán un flujo de aprobación automático.

- **Manejo de contraseñas:** Creación de políticas de contraseñas para cada uno de los sistemas conectados con la tecnología IAM.

- **Inicio de sesión único (SSO):** Una sola identidad por usuario conectada a múltiples sistemas.

- **Control de acceso basado en roles (RBAC) / Gobernanza de acceso.**

- **Auditoría y Cumplimiento**: Creación de reportes e informes personalizados.

[2] PoC : Prueba de concepto (PoC – Proof of Concept) Es una implementación, a menudo resumida o incompleta, de un método o de una idea, realizada con el propósito de verificar que el concepto o teoría en cuestión es susceptible de ser explotada de una manera útil.

1.4 Estructura

Esta memoria se compondrá de un total de 7 capítulos:

1. Capítulo 1: Introducción: Qué es, y por qué es necesaria la gestión de identidades.
2. Capítulo 2: Estado del arte: Situación actual de las tecnologías.
3. Capítulo 3: Análisis del problema: Planteamiento y Análisis de la gestión de la Universidad Politécnica de Valencia.
4. Capítulo 4: Identificación y análisis de la solución propuesta: En este capítulo se desarrolla una solución para el problema presentado en el capítulo 3.
5. Capítulo 5: Diseño de la solución: Se describe y se desarrolla toda la implementación. En este capítulo se mostrarán una serie de imágenes como prueba de lo implementado.
6. Capítulo 6: Validación y análisis de despliegue: Metodología empleada, y presupuesto.
7. Capítulo 7: Conclusiones: Análisis del resultado obtenido, relación con los estudios cursados y trabajo futuro.

Capítulo 2: Estado del arte

La gestión de identidades (IAM) no es monolítica, esta se puede ramificar en subcategorías únicas, cada una con sus propios objetivos y capacidades. Esto incluye la gestión de accesos privilegiados (PAM), el gobierno y la administración de identidades (IGA) [6].

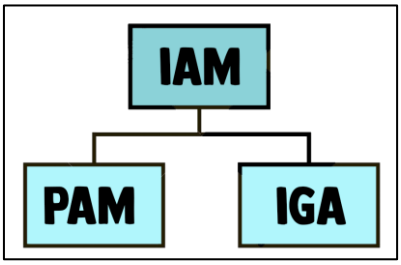

Ilustración 5: Esquema IAM. Fuente: Elaboración propia.

2.1. PAM: Gestión de accesos privilegiados

- La Gestión de Cuentas Privilegiadas o PAM (Privileged Access Management) consiste en asegurar que la persona adecuada dispone del acceso adecuado al recurso adecuado, en el momento adecuado, y por razones adecuadas [7].

- Requisitos:
  - Una solución tecnológica sólida.
  - Los procesos necesarios.
  - Combinados y utilizados de la manera correcta.

- Dificultades en un proyecto PAM:

  - Problemática para descubrir y gestionar los accesos privilegiados por parte de las empresas.
  - Existen pocas organizaciones capaces de implementar el principio de Acceso de Mínimos o "Least Privileged Access" lo que conlleva unos riesgos de seguridad excesivos.

Dos de las tecnologías mas usadas en este campo son Okta y CyberArk.

1. Okta

Los principales objetivos a destacar de okta incluyen:

- Ofrecer un Acceso seguro.
- Mejorar la experiencia del usuario.
- Eliminar tareas manuales.
- Mejorar los procesos de negocio.

La figura 6 ilustra los principales productos del Identity Cloud que ofrece okta.

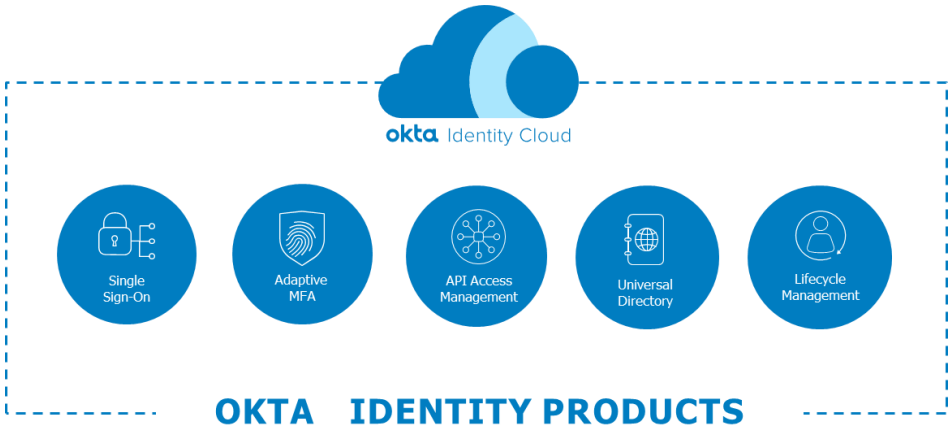

Ilustración 6: Productos en Okta. Fuente: Okta.

Entre sus principales características son de destacar:

- Single Sign-On: Basta con autenticarse una vez para poder usar cualquier aplicación federada con Okta.
- MFA[3] Adaptativo (basado en contexto: dispositivo, localización y red).
- API Access Management: Creación, mantenimiento y audición de las políticas de acceso por API.
- Universal Directory: Customización, organización y gestión de los múltiples atributos de las cuentas de los diferentes sistemas fuente.
- Lifecycle Management: Automatización de "onboarding" y del "offboarding", asegurando la comunicación con los diferentes sistemas externos [8].

2. CyberArk

Las principales características de CyberArk son las siguientes:

- Ofrece visibilidad de los riesgos en permisos en ambientes multinube.
- Implementa Mínimos Privilegios a través de diferentes ambientes Cloud.
- Opera los permisos de una manera segura y eficiente.
- Proactivamente reduce el riesgo y mide el progreso.
- Describe controles básicos que todo programa PAS[4] debe abordar a lo largo del tiempo [9].

[3] MFA : Multifactor de autenticación: Un factor en la autenticación es una forma de demostrar que usted es quien dice que es al intentar iniciar sesión.
[4] PAS : Pro-Active Support : La atención proactiva consiste en identificar y resolver los incidentes de los agentes externos antes de que se conviertan en problemas.

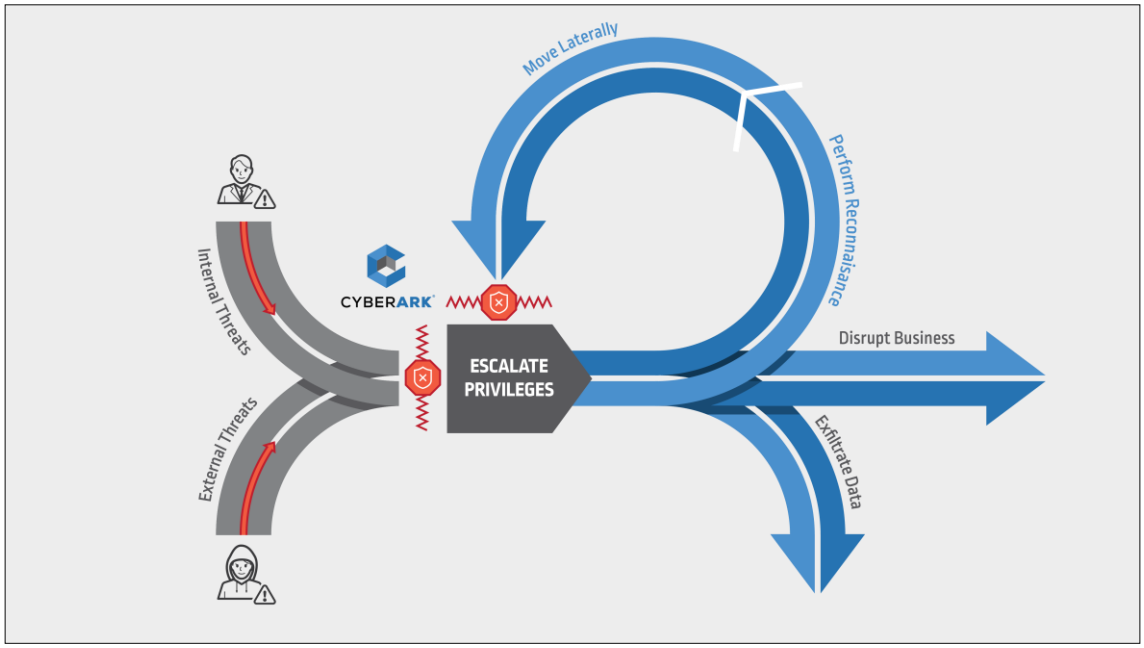

Ilustración 7: "Rompiendo la cadena de ataque". Fuente: Cyberark

CyberArk es nombrada líder en el "**Gartner 2018 Magic Quadrant** " para la gestión de acceso privilegiado.

Ilustración 8: Ranking Gartner 2018 Magic Quadrant. Fuente: Gartner 2018 [10].

2.2 IGA: Gobierno de la Identidad

- El Gobierno de la Identidad (IGA), también llamado Gestión de Identidad (IDM), combina la administración y el gobierno sobre la recopilación, el uso y la eliminación de la información de identidad. Requiere un mecanismo de gobierno que permita a los gestores certificar los derechos que se han concedido a su personal. Además, el IGA suele incluir funciones de supervisión y elaboración de informes para los servicios de identidad que, a su vez, respaldan los requisitos corporativos [11].

- IGA nos permite:
  - Automatizar los procesos de aprovisionamiento de cuentas de las personas pertenecientes a una organización.
  - Adoptar un enfoque Least Privilege, de forma que garanticemos la asignación del mínimo número de permisos para realizar un trabajo.
  - Facilitar el gobierno de la identidad, permitiendo delegar en negocio la certificación de accesos.
  - Integrarse en los procesos de negocio para mejorarlos.

- Dificultades en IGA:
  - En un programa IGA se necesita conocimiento de negocio y conocimiento técnico de manera proporcional. En la mayoría de las organizaciones, los responsables de la ciberseguridad no tienen desarrollado el perfil de negocio y consultoría técnica necesario.
  - Cuando hablamos de proyectos de Gestión de Identidad se suele pensar en la automatización del ciclo de vida del empleado, pero es importante entender que IGA es mucho más complejo. También se incluye el aprovisionamiento de las cuentas tipo IOT o funcionalizades avanzadas (gestión del riesgo, certificación, segregración de funciones…).
  - Muchas de las iniciativas IGA no aprovechan al máximo las capacidades de esta tecnología. Para tener un proyecto sólido es necesario utilizar todas sus funciones tanto internas como externas, que apoyen las operaciones de negocio, siendo el propio negocio el elemento más relevante en este tipo de proyectos.

  Una de las tecnologías mas usadas para IGA es One Identity by Quest.

Ilustración 9: Logotipo One Identity. Fuente: One Identity.

One Identity Manager sincroniza identidades, suscripciones, planes de servicio, grupos y roles de administración. Esto hace posible el uso de los procesos de gobierno de identidad y acceso, incluyendo la atestación, la auditoría de identidad, la gestión de cuentas de usuario y los derechos del sistema, el portal web, o las suscripciones de informes para los inquilinos de Azure Active Directory. Además, One Identity cuenta con un módulo de PAM [12].

Esta será la tecnología que utilizaremos para el desarrollo del proyecto. En los próximos capítulos se detallarán sus herramientas y funcionalidades.

Capítulo 3: Análisis del problema

Para el desarrollo de este estudio recrearemos una situación hipotética en la Universidad Politécnica de Valencia. En este caso ficticio, la UPV no tiene habilitado un gestor de identidades adecuado. Las características de su arquitectura de sistemas incluyen:

- Una base de datos relacional donde almacenan todos los datos, tanto de los estudiantes como los trabajadores.
- Un dominio de Directorio Activo (AD) organizado mediante unidades organizativas (OUs). Cada una de estas unidades organizativas ofrecen licencias o accesos a recursos.
- El uso de una aplicación de recursos humanos externa (HR) por parte de la UPV. Esta app lleva la gestión de los empleados (alta, modificación y baja).
- Gestión financiera, la cual se lleva a cabo en un sistema externo ERP[5] (enterprise resource planning).

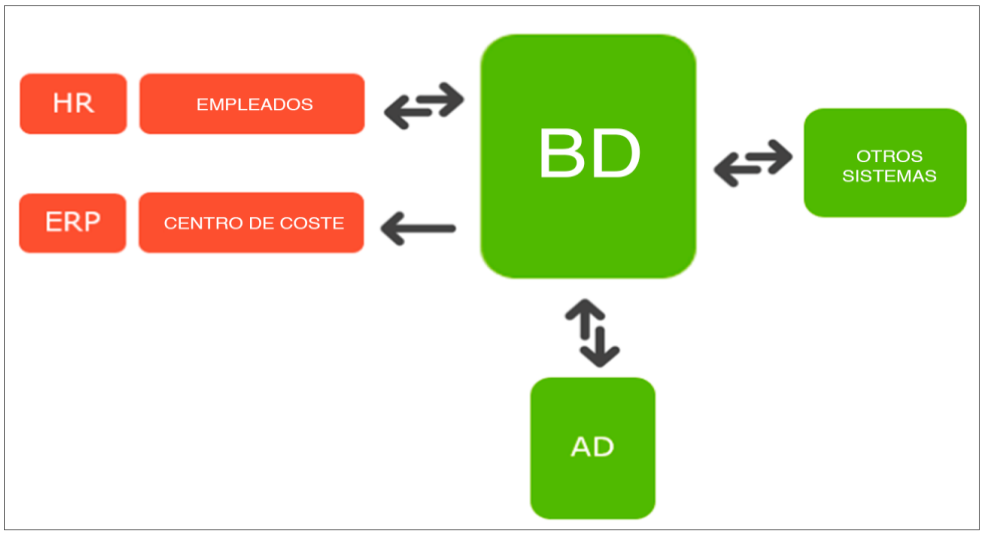

Ilustración 10: Estructura previa UPV. Fuente: Elaboración propia

A continuación se analizan todos los problemas que conlleva esta estructura.

Tabla 1:Análisis de los problemas de la UPV sin gestor de identidades. Fuente: Elaboración propia.

TIPO	PROBLEMA	DETALLES
Seguridad	El ciclo de vida del usuario no es automático.	• Para una nueva alta se crea la cuenta y manualmente se asigna los grupos de AD (sobretrabajo). • Para realizar una baja se realiza de manera manual en la aplicación RRHH[6]. El usuario queda dado de baja en la app, pero todas las asignaciones que se han realizado externamente no son eliminadas, dejando una

[5] ERP : sistemas de planificación de recursos empresariales : son sistemas de información gerenciales que integran y manejan muchos de los negocios asociados con las operaciones de producción.
[6] RRHH: Recursos Humanos.

		brecha en el sistema la cual puede ser usada por los ciberdelincuentes.
Seguridad	No existen reglas o roles en la base de datos.	Si un usuario cambia de puesto, habrá que introducir manualmente sus nuevos accesos. Además, los antiguos accesos no son revocados.
Funcionalidad	Añadir nuevos permisos a un empleado.	Si un empleado necesita tener acceso a un recurso en un momento determinado, deberá abrir una incidencia con el equipo de IT para que se lo puedan dar. Esto conlleva sobretrabajo y tiempo.
Funcionalidad	Conflicto de Contraseñas	Si las políticas de contraseña de los diferentes sistemas no coinciden entre ellos, pueden generar errores. Por ejemplo, si desde RRHH introducimos una contraseña que cumple con la política de la aplicación, pero luego no cumple con la política de AD, no se mapeará la contraseña.
Funcionalidad	Arquitectura no modular	Si en un futuro se quisiera introducir un nuevo sistema necesitaríamos adaptarlo y volver a configurarlo.
Auditorías	Complejidad de realizar reportes.	Dificultad de crear históricos ya que la arquitectura está divida en multitud de sistemas. Esto es un inconveniente a la hora de realizar auditorías, ya que no tienes un registro claro con las personas y sus permisos.

Capítulo 4: Identificación y análisis de la solución

Para resolver los problemas descritos en el anterior capítulo, será necesario implantar una tecnología capaz de gestionar todas las identidades y controlar los accesos. Para ello, unificaremos todos los sistemas en un solo entorno:

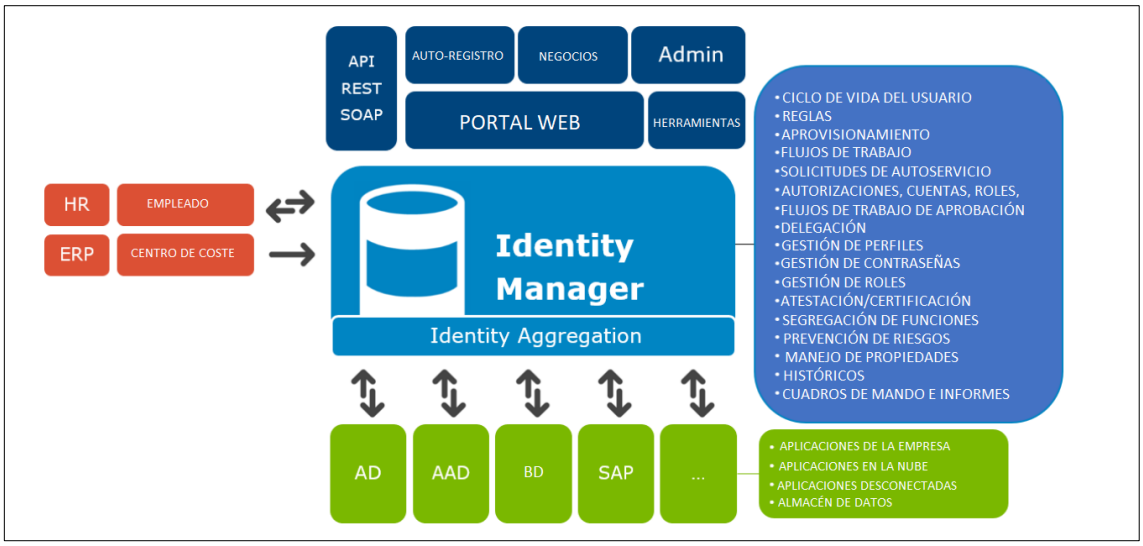

Ilustración 11: Estructura resultado de Sistemas UPV. Fuente: Elaboración propia.

Para realizar esta implementación utilizaremos la tecnología *One Identity* de la empresa Quest. Esta solución establece una estrategia de seguridad centrada en la identidad, con una combinación única de gestión de identidades (IGA).

En esta memoria se documenta la creación del sistema para una cantidad reducida de estudiantes, empleados y facultades, para que el cliente, en este caso, la UPV, pueda valorar si invertir en este tipo de proyecto.

El sistema reducido cuenta con tres campus (Vera, Alcoy y Gandía) donde se ubicarán los empleados (profesores, empleados externos, directores, administrativos, etc) y tres facultades ubicadas en el campus de Vera (Informática ETSINF, Arquitectura ETSAT y Telecomunicaciones ETSIT)

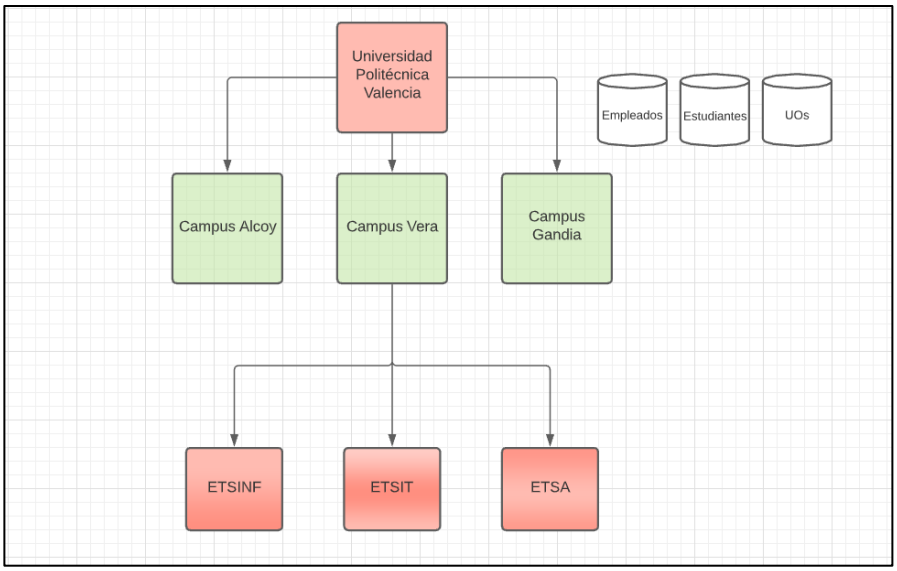

Ilustración 12: Esquema organizativo para la PoC. Fuente: Elaboración propia.

Las identidades que se manejarán en la PoC son las siguientes:

Tabla 2 : Identidades usadas para la PoC. Fuente: Elaboración propia.

Estudiante ETSINF	Estudiante ETSA	Estudiante ETSIT	Empleado vera	Empleado Alcoy	Empleado Gandia
Javier Cutillas Zárate	Verónica Perea Ruiz	Gonzalo Ayala Ponce	Roberto García Zamora	Jaime Cuesta Zamora	Imanol Manzano Mártinez
Naiara Manzano Mártinez	Victor Francisco Antolín Marín	David Iglesias Ortiz	Daniel Castro Medina	Patricia Ramirez Alonso	Jonathan García García
Raul Monleón Mártinez		Aida García Garciía	Josep Mars Blanco		Luis Mariano García García
Judith Díaz Sánchez			Nuria Gil Serrano		
Argie Escobal Leal					
David Romero Torres					

Además, Roberto, Patricia y Jonathan son becarios, lo que significa que tendrán un recurso adicional respecto a los empleados que no lo son.

También se añadirá un nuevo empleado llamado Jaime Caravaca desde la aplicación de recursos humanos.

El sistema tiene que ser capaz de:

- Crear un proceso que genere un identificador único para estudiantes y empleados que utilice la misma nomenclatura.
- Establecer un proceso de creación de correos electrónicos según su facultad (estudiante) o su campus (empleado).
- Automatizar un proceso que importe estudiantes y empleados al sistema periódicamente.
- Conectar la aplicación de HR para que, cuando se realice una nueva alta, se introduzca automáticamente en el sistema.
- Para cada identidad se creará automáticamente una cuenta de Directorio Activo; para los estudiantes dentro del grupo "UPV - Alumno", y para los empleados dentro del grupo "UPV - Empleados".
- Los alumnos que pertenezcan a la facultad de informática, se les aprovisionará de las siguientes licencias: "Aplicaciones de Google", "Office 365", "Visual Studio"," VmWare" y "Aplicaciones Java".
- A los becarios se le asignará automáticamente la licencia: "Aplicaciones de Google" o también conocido como "G-Suite".
- Los empleados que pertenecen al campus de Alcoy trabajarán solo con la licencia de "Office 365", y los empleados de Gandía trabajarán con las "Aplicaciones de Google".
- Se habilitará un portal web donde un empleado de Gandía pueda pedir la licencia de Office 365, y viceversa. Para que esta petición sea aprobada, será el manager del campus al que pertenece el empleado quien deba aceptarla o reclinarla.

4.1 Programa IGA propuesto para la Universidad Politécnica de Valencia

Un programa IGA va más allá de la solución tecnológica seleccionada e incide, sobre todo, en los procesos de negocio a los que da soporte. Para la PoC inicial seguiremos el siguiente programa:

Ilustración 13: Programa propuesto. Fuente: Elaboración propia.

4.1.1 Diseño del plan de ejecución

La implementación de la prueba de conceptos tendrá una duración de 14 semanas.

Tabla 3: Diseño del plan PoC. Fuente: Elaboración propia.

	Semana													
Fase	1	2	3	4	5	6	7	8	9	10	11	12	13	14
2. PoC implementación														
2.1 Instalación														
2.1.1 Instalación cliente														
2.1.2 Instalación AD														
2.1.3 Instalación IAM Server														
2.2 Estructura organizativa														
2.2.1 Creación de las facultades y campus														
2.3 Ciclo de vida de estudiantes y empleados														
2.3.1 Configuración login y mail														
2.3.2 Ciclo de vida desde CSV														
2.3.3 Ciclo de vida desde HR														
2.4 Implantación de Roles														
2.4.1 AD Conector														
2.4.2 Configuración definición de cuenta														
2.4.3 Roles estáticos														
2.4.4 Roles dinámicos														
2.5 Habilitar Web														
2.5.1 Configuración back-end														
2.5.2 Configuración de los aprobadores														
2.5.3 Despliegue front-end														
3. Pruebas														
3.1 Realización de pruebas														

Capítulo 5: Implementación de la solución IGA

5.1 Herramientas de One Identity

La tecnología que hay detrás de One Identity es una gran base de datos relacional en la cual se guardan datos, permisos, relaciones, etc. Para poder gestionar todas las funcionalidades la solución nos ofrece un amplio catálogo de aplicaciones. En este apartado vamos a comentar las aplicaciones comunes que se han usado para la realización de la PoC:

- **Manager**: Esta herramienta nos permite gestionar los objetos de una forma visual. Podemos ver los atributos de un objeto, sus asignaciones, si pertenece a otros objetos, etc.

- **Designer**: Gestiona todas las tablas de la base de datos, lo cual incluye librería de scripts y todo lo relacionado con procesos (creación, automatización, etc.).

- **JobQueue**: Proporciona información de todos los procesos. Podemos seguir la ejecución de un proceso, averiguar qué procesos están en cola, o ver el histórico de los procesos.

- **SchemaExtension**: Nos permite tanto crear como actualizar tablas existentes en la base de datos.

- **SynchronizationEditor**: Encargado de establecer conexiones con otros sistemas, como por ejemplo Directorio Activo. Además, desde él podemos configurar los mapeos de los datos.

- **WebDesigner**: Aplicación encargada de configurar el portal Web para las peticiones.

- **DBCompiler**: Compilador del sistema.

One Identity está construido bajo el lenguaje de Visual Basic .Net. Este mismo lenguaje también es el usado para realizar "scripts" y procesos. La parte enfocada a bases de datos utilizaremos "Sql Server".

5.2 Arquitectura del Sistema

Para este proyecto vamos a necesitar 3 servidores. En el servidor "Client" vamos a instalar herramientas básicas para que el cliente pueda consultar los datos del sistema, pero sin tener permisos de administrador. El servidor "IAMSERVER" será el principal encargado de gestionar todo el sistema, y desde donde haremos las configuraciones; por últimon el servidor "AD" se instalará el controlador de directorio activo que estará sincronizado con One Identity.

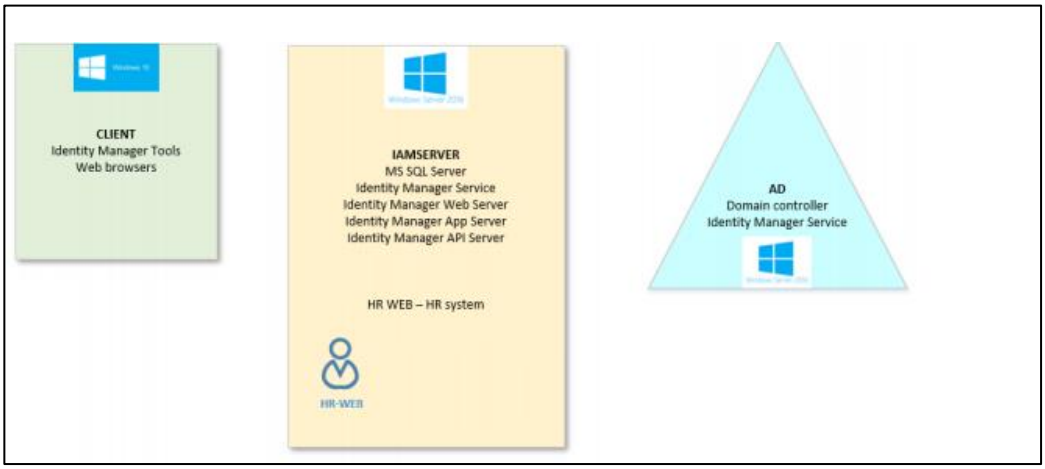

Ilustración 14: Arquitectura PoC. Fuente: Elaboración propia.

Como hemos comentado anteriormente, One Identity esta formado por una base de datos. Esta base de datos ya cuenta con unas tablas por defecto. En esta PoC vamos a utilizar las siguientes:

- Person: tabla principal donde se registran las identidades y todos sus atributos.
- ADSAccount: identidades que tienen una cuenta en AD
- Department: utilizadas comúnmente para establecer la jerarquía de la organización. En este caso, guardaremos los campus y las facultades.
- Org: tabla en la cual se guardan los roles estáticos y dinámicos.
- QERResource: usadas para guardar recursos adicionales; en este proyecto crearemos como un recurso nuevo ser becario.

  Para guardar la relación entre una tabla y otra, OneIdentity tiene tablas relacionales, por ejemplo:

- PersonHasResource: una identidad tiene un recurso.
- PersonHasOrg: una identidad tiene un rol.

  Además, podemos añadir más tablas o más columnas desde la aplicación: "SchemaExtension"

5.3 Desarrollo PoC

5.3.1 Primeras configuraciones

Desde el servidor IAMSERVER abrimos la aplicación "Designer". Desde su interfaz abriremos la tabla "Person" y seleccionaremos el atributo IdentityType.

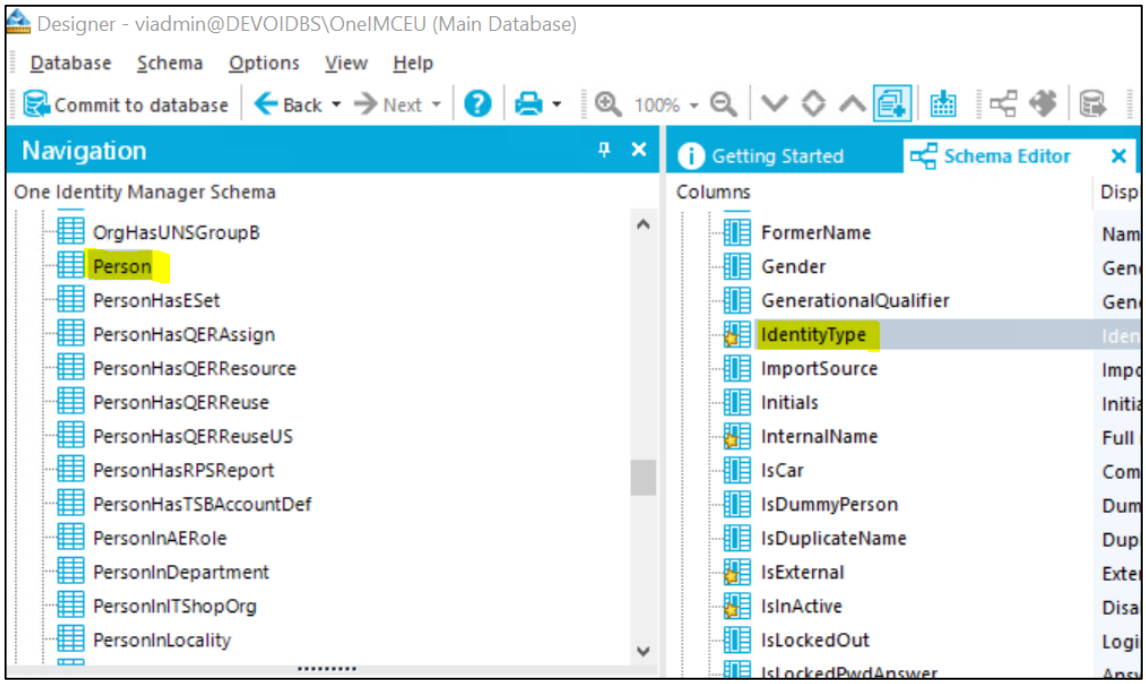

Ilustración 15: Interfaz aplicación "designer". Fuente: Elaboración propia.

Esta columna es original de One Identity. Nos permite diferenciar las identidades a través de un "String". Lo configuraremos para que el sistema sea capaz de distinguir entre empleados y estudiantes.

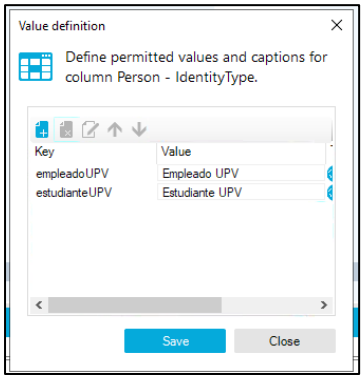

Ilustración 16: Valores en la columna "IdentityType". Fuente: Elaboración propia.

El siguiente paso será crear un identificador que distinga las identidades. El identificador seguirá la misma nomenclatura que tiene la actual UPV. Se formará a través de las dos primeras letras del nombre, las dos primeras letras del primer apellido, y las dos primeras letras del segundo apellido. Si ese identificador ya está en uso, se agregarán de forma secuencial números al final del ID.

La columna prestablecida por One Identity para gestionar el identificador se llama "CentralAccount", que pertenece a la tabla "Person".

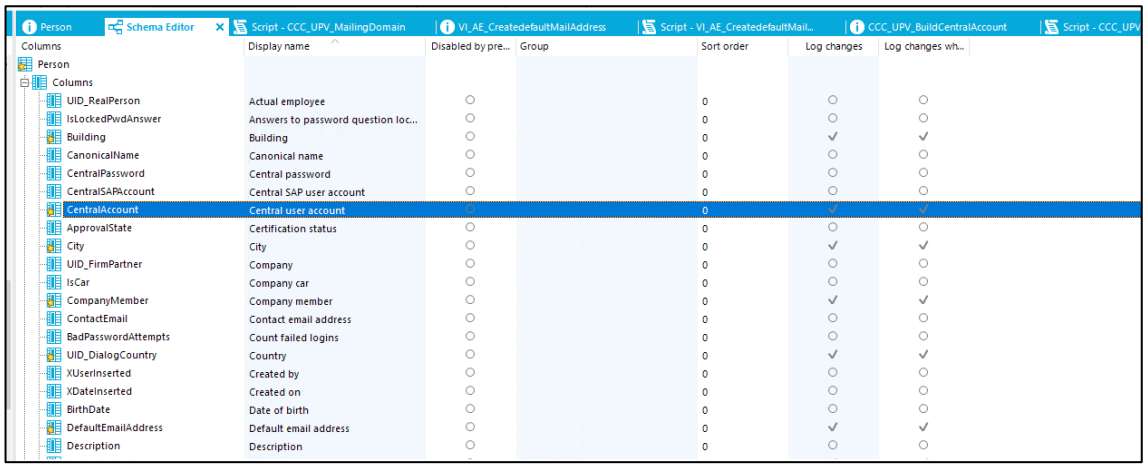

Ilustración 17: Columna "CentralAccount" en la aplicación "designer" . Fuente: Elaboración propia.

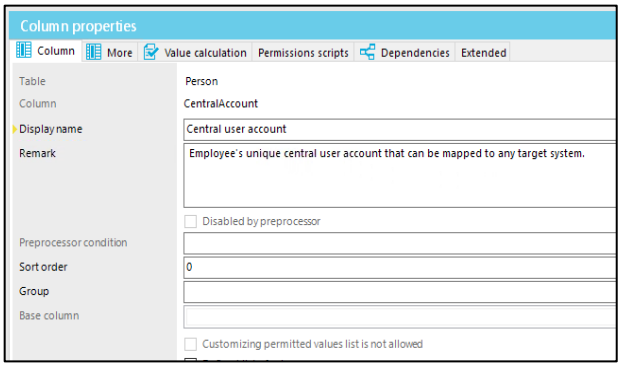

Ilustración 18: Propiedades de "CentralAccount". Fuente: Elaboración propia.

Para cada columna de cualquier tabla de One Identity podemos configurar "Triggers". Un "Trigger" lo podríamos definir como un cálculo que se realiza cuando la columna o sus relaciones tienen algún cambio. También se desencadena el cálculo cuando se crea un nuevo objeto.

En nuestro caso, si una identidad es del tipo empleado o estudiante, llamará a un "script" que calcule dicho identificador:

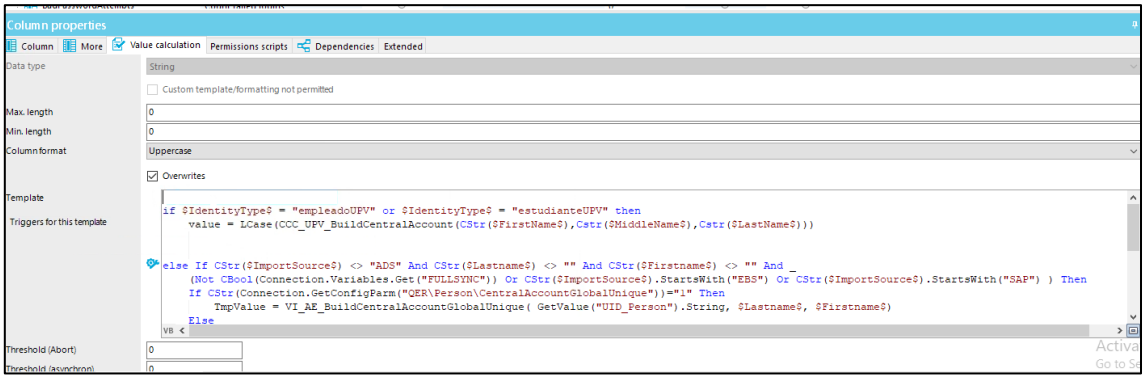

Ilustración 19: Cálculo de "CentralAccount. Fuente: Elaboración propia.

El check "Overwrittes" significa que, aunque el campo tenga un valor, si se produce algún cambio se volverá a recalcular. Para calcular el identificador ("centralAccount") necesitamos el nombre y los apellidos; estos campos se los podemos pasar al script con la siguiente nomenclatura: *$nombreColumna$*

```
Public Function CCC_UPV_BuildCentralAccount(ByVal firstName As String, ByVal
middleName As String, ByVal lastName As String) As String

        'Creación de CentralAccount a través de los atributos: Nombre y apellidos

        Dim vCentralAccount As String = LSet(firstName,2)+ LSet(lastName,2) +
LSet(middleName,2)

        'Comprueba si existe en el sistema un centralAccount con la misma combinación

        Dim query As Query = Query.From("person").Where(String.Format("centralaccount =
'{0}'",vCentralAccount)).SelectAll

        Dim collection As IEntityCollection = Session.Source().GetCollection(query,
EntityCollectionLoadType.Slim)



        If collection.Count = 0 Then

                Return vCentralAccount

        Else

                'Encuenta que ya existe esa combinación

                'Por tanto, añade un numero al final del centralAccount

                'y vuele a comprobar

                Dim auxCentalAccount As String = String.Empty

                For contador As Integer = 1 To 9

                        auxCentalAccount = vCentralAccount + contador.ToString

                        query =
Query.From("person").Where(String.Format("centralaccount =
                '{0}'",auxCentalAccount)).SelectAll

                        collection = Session.Source().GetCollection(query,
EntityCollectionLoadType.Slim)

                        If collection.Count = 0 Then

                                Return auxCentalAccount

                        End If

                        auxCentalAccount = String.Empty

                Next
```

Algoritmo 1: Cálculo del ID de las identidades. Fuente: Elaboración propia.

El primer paso de este "script" es generar el identificador y comprobar si está en uso. Si lo está, entrará en un bucle sumando un numero al final hasta que encuentre una combinación libre.

Para generar la cuenta de correo, utilizaremos el "centralAccount" como nombre del buzón. Para crear el dominio replicaremos una lógica parecida a la actual de la UPV. Si es un estudiante se calculará en base a la facultad, si es un empleado en base al campus; por ejemplo, si un estudiante pertenece a la ETSINF su dominio será @alumno.etsinf.es, y si un empleado pertenece al campus de VERA será @empleado.vera.es.

Este proceso será similar al utilizado en el centralAccount. Se trata de un campo llamado "defaultEmailAddress" de la tabla "Person", que está prestablecido en OneIdentity. Crearemos un "trigger" con la condición de que, si es empleado o estudiante UPV, llame a un "script" con los siguientes parámetros: "centralAccount", "identityType", "UID_Department").

Cada objeto en One Identity está identificado a través de un UID ("unique identificator"). Para establecer que una persona es miembro de un departamento (facultad o campus), se guarda el "UID_Department" en una columna en la tabla de "Person". Si la identidad no es miembro de un departamento no se le asignará cuenta de correo.

```
Public Function CCC_UPV_MailingDomain(ByVal centralAccount As String, ByVal identityType As
String, ByVal uidDepartamento As String) As String

        Dim dominio As String = String.Empty

        Dim nombreDepartamento As String = String.Empty

        Dim email As String = String.Empty

        'Busca en la tabla departamento el nombre del departamento con condicion de que el UID sea
el UID que pasamos por parametro y lo guarda en nombreDepartamento

        If session.Source.TryGetSingleValue(Of
String)("Department","departmentname",String.Format("UID_Department =
        '{0}'",uidDepartamento),nombreDepartamento) Then

                If identityType.Contains("empleado") Then

                        dominio = "@empleado." + nombreDepartamento + ".es"

                Else If identityType.Contains("estudiante") Then

                        dominio = "@alumno." + nombreDepartamento + ".es"

                Else

                        dominio = "@UPV.es"

                End If

        End If

        email = centralAccount + dominio

        Return email.ToLower


End Function
```

Algoritmo 2: Cálculo del correo electrónico. Fuente: Elaboración propia.

5.3.2 Ciclo de vida

El ciclo de vida empieza cuando una nueva persona es creada en el sistema. Esto puede ser realizado en diferentes maneras que satisfagan las necesidades de negocio del cliente.

El camino común para crear personas es sincronizarla desde uno o más sistemas externos. En el caso de esta PoC utilizaremos dos caminos: desde un CSV, y desde la aplicación HR externa.

Una vez la identidad está creada, el siguiente paso del ciclo de vida es aprovisionarla. Mediante este proceso empiezan una serie de asignaciones de permisos y accesos a otros sistemas.

La configuración del aprovisionamiento es muy flexible. Puede venir dada por roles, por departamentos, localizaciones o centros de costes asignados, por recursos, o incluso por procesos automatizados.

Antes de poder importar las identidades debemos tener establecida la estructura departamental de nuestro sistema. Esta estructura también se puede sincronizar e importar, pero en nuestro caso la vamos a agregar manualmente.

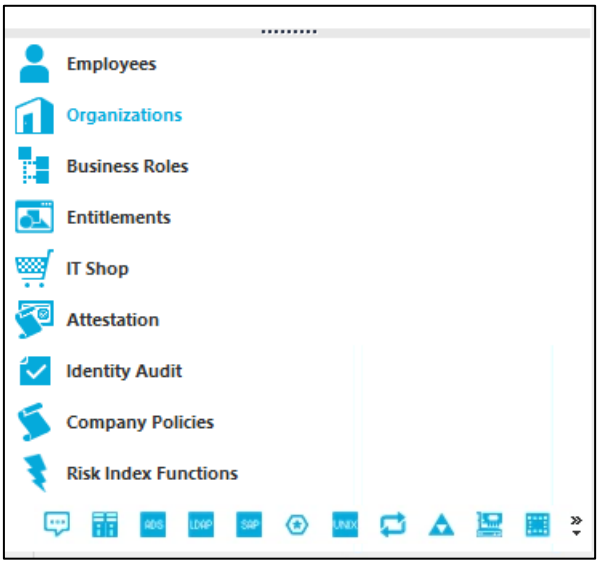

Ilustración 20: Interfaz aplicación "manager". Fuente: Elaboración propia.

Dentro de la aplicación "manager" nos dirigimos a "Organizations". Dentro de "organizations", tenemos tres tipos de estructuras: "Departments", "Locations", "Cost Centers". Para el "back-end" estas tres estructuras trabajan igual. Para recrear los campus y facultades las crearemos bajo "Departments". Esto conlleva nuevas entradas en la tabla "Departments" en la base de datos.

Los atributos básicos para crear un departamento son el nombre del departamento, un alias ("short name"), el padre del departamento en la jerarquía, y la ruta; estos datos los podemos ver en la ilustración 21.

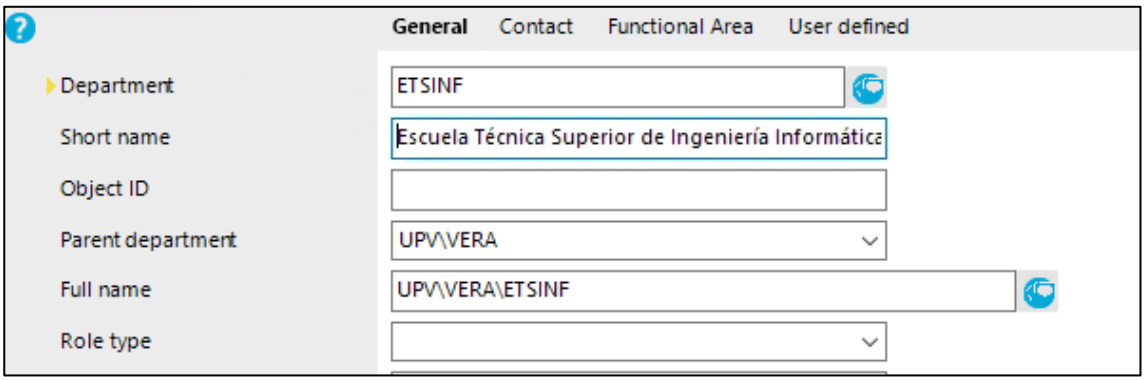

Ilustración 21: Atributos de un departamento. Fuente: Elaboración propia.

La representación jerárquica sería la siguiente:

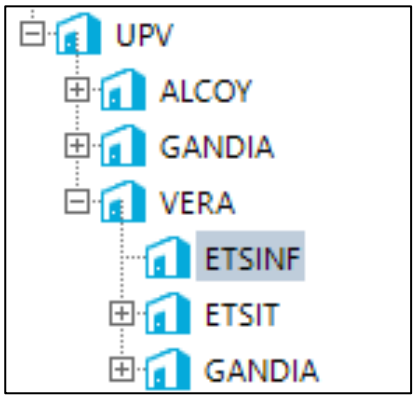

Ilustración 22: Jerarquía vista desde la aplicación "manager". Fuente: Elaboración propia.

Además de la creación de los campus y facultades, vamos a añadir un campo customizado a la tabla "Person" para guardar el valor del documento de identidad de una persona. Para ello abrimos el "SchemaExtension" y elegimos la opción de expandir tabla y configuramos una nueva columna: CCC_Document.

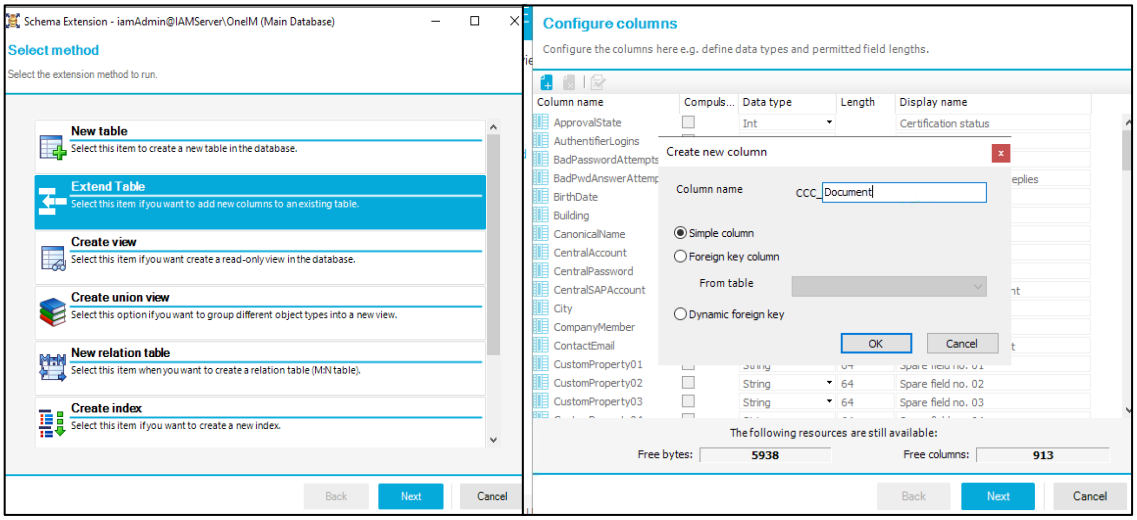

Ilustración 23: Creación de una nueva columna en la tabla "person". Fuente: Elaboración propia.

En los siguientes dos apartados se mostrarán las dos formas usadas para importar personas:

5.3.2.1 Desde CSV

Para importar debemos generar un CSV delimitando los campos con ";" y decidir qué valores ponemos en cada campo, en nuestro caso:

nombre;apellido1;apellido2;género;tipo;departamento;EsBecario;Documento

"Género" es un atributo prestablecido en One Identity donde 1 es Masculino y 2 es Femenino. Cuando un empleado sea becario, agregaremos el texto "becario" en la columna "esBecario".

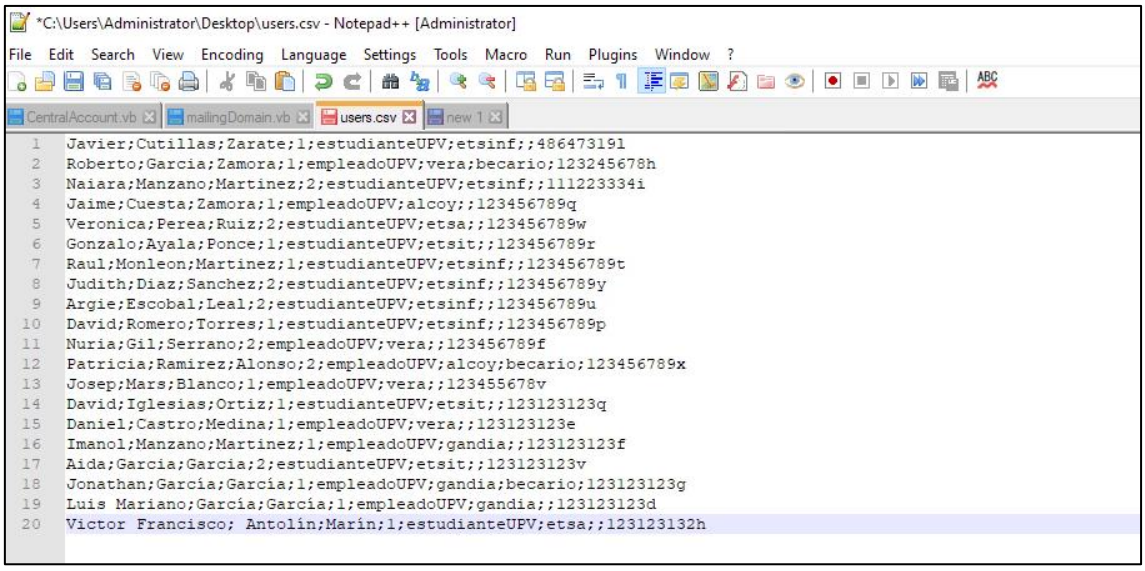

Ilustración 24: Fichero para importar identidades. Fuente: Elaboración propia.

One Identity cuenta con una aplicación "launchpad" que tiene acceso directo al resto de apliaciones. Desde el launchpad ejecutamos "Configure a data import".

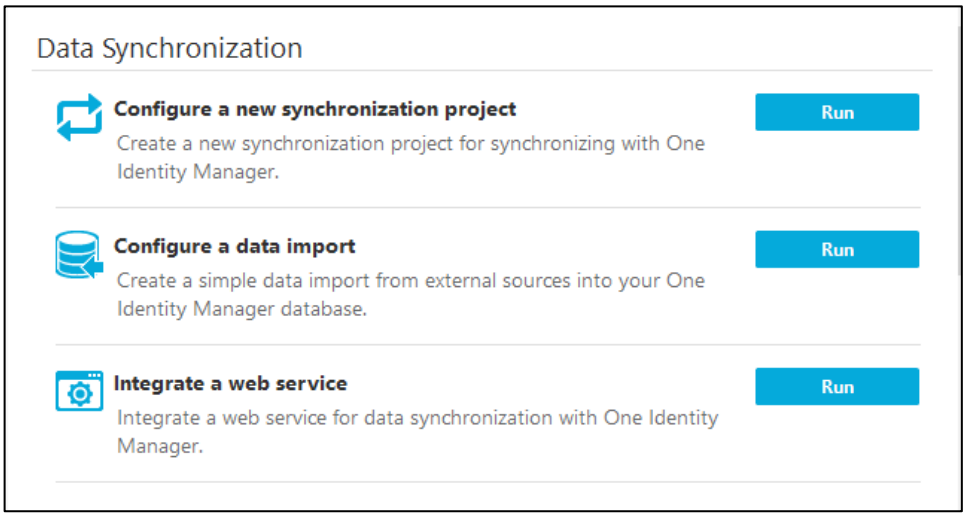

Ilustración 25: Interfaz launchpad one identity. Fuente: Elaboración propia.

Se abrirá un asistente que facilita la importación de valores. Seleccionamos "Import CSV file" para importar desde CSV.

Desde la configuración debemos establecer el enlace de los valores con los campos de la tabla "Person" y establecer una clave primaria para que no haya duplicados, en este caso el documento de identidad.

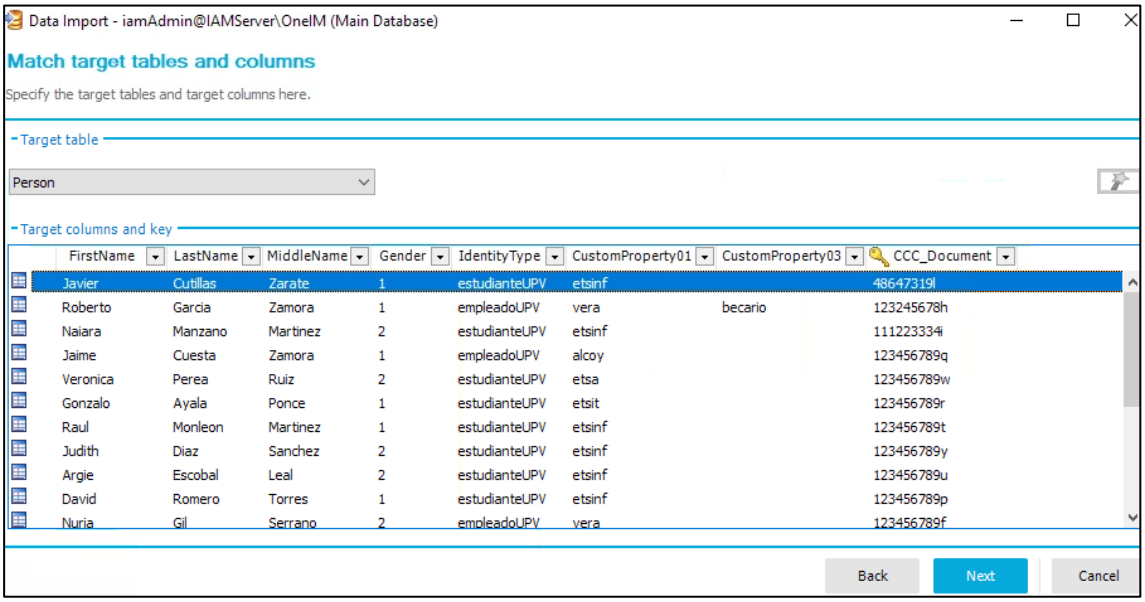

Ilustración 26: Sincronización fichero CSV. Fuente: Elaboración propia.

En la fase final de la importación seleccionaremos generar un "script". Este código va a popular los campos mencionados anteriormente, pero no va a crear la relación con los departamentos. Por este motivo necesitaremos modificar el "script" posteriormente

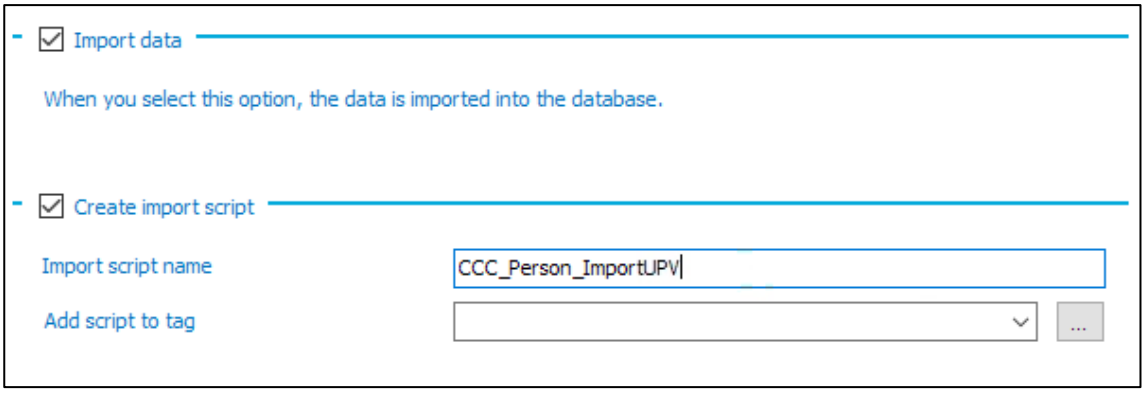

Ilustración 27: Creación automática de Script que importa las identidades desde CSV. Fuente: Elaboración propia.

Una vez finalizado el proceso de importación, nos notificará si ha habido errores. En nuestro caso todos los datos han sido importados correctamente.

Hay 3 roles prestablecidos que asigna One Identity automáticamente. Estos sirven para mantener la cuenta activa, y que pueda ser asignada a recursos y permisos. La caja roja es un acceso automático de One Identity para que el usuario pueda acceder al portal web. En nuestro caso deshabilitaremos esta opción para los estudiantes, pues serán solo los empleados quienes tengan acceso. Podemos apreciar como la columna "Central user account" se ha calculado correctamente:

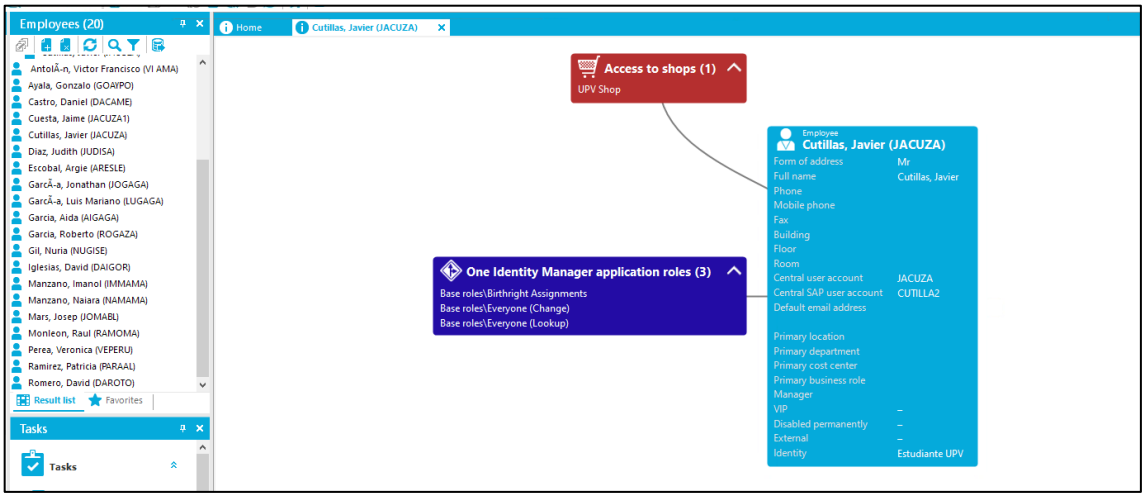

Ilustración 29: Vista de un usuario desde "manager". Fuente: Elaboración propia.

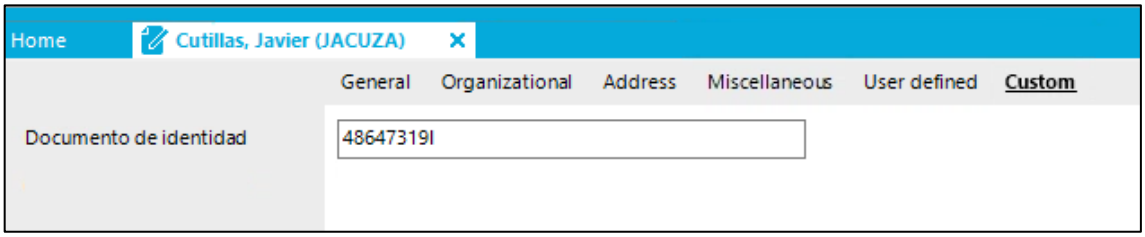

Ilustración 28: Vista del atributo "CCC_Document". Fuente: Elaboración propia.

En la esquina superior izquierda de la ilustración 30, podemos ver que hay 20 identidades, que son las que hemos añadido, con sus respectivos datos. Además, observamos que para Javier Cutillas Zárate ha generado "JACUZA", y para Jaime Cuesta Zamora ha generado "JACUZA1".

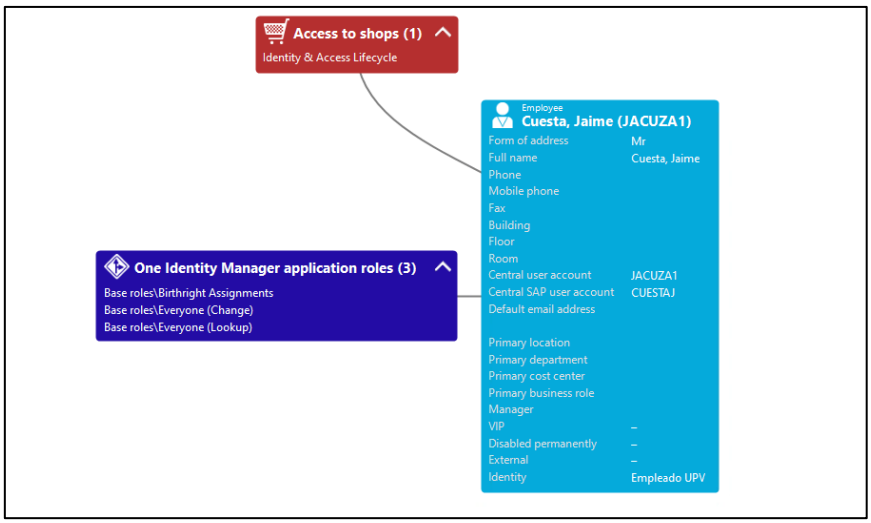

Ilustración 30: Vista del usuario jacuza1 desde "manager". Fuente: Elaboración propia.

Como podemos apreciar en la ilustración 30, no se ha generado una cuenta de correo. El cálculo del correo, como hemos explicado anteriormente, depende de la facultad o el campus al que estes asignado. Por lo tanto, al no ser miembro de ningun departamento (facultad o campus) la cuenta de correo no se ha generado. Para crear una relación entre la persona y el departamento debemos modificar el proceso de importación.

La tabla "Person" cuenta con 10 campos preestablecidos que van desde "CustomProperty01" a "CustomProperty10". En este primer campo, "CustomProperyo1", hemos guardado el nombre del campus o de la facultad, en el "CustomProperty03" hemos guardado si el empleado es becario o no. Estos campos los vamos a utilizar en la modificación del "script" generado por la importación. La nueva modificación se basa en la membresía del departamento al que pertenece. El proceso obtendrá el texto almacenado en "CustomProperty01", después buscará en la tabla "Departments" si algún departamento (campus o facultad) coincide con el texto mencionado anteriormente. Finalmente, si existe dicho departamento, se guardará en la persona el valor "UID_Department" que consecuentemente creará una relación del tipo persona-departamento que se guardará en la tabla "PersonHasDepartment".

```
Select d.uid_department from department as d join person as p where p.customproperty01 = d.departmentname;
```

Algoritmo 3: Consulta SQL para la obtención del identificador del departamento

Lo que conseguiremos con este cambio es automatizar la asignación de una identidad a su departamento. La próxima vez que el cálculo se ejecute, todas las identidades tendrán su departamento correspondiente asignado. Para automatizar este cálculo vamos a crear un proceso en la aplicación "Designer":

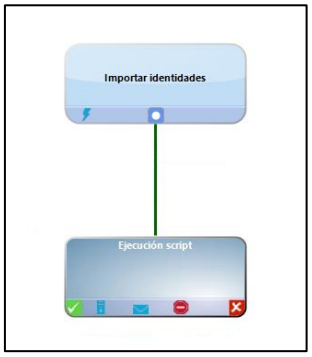

Ilustración 31: Pasos del proceso de importar identidades.
Fuente: Elaboración propia.

Los procesos en One Identity se organizan por pasos. Cada paso puede tener una función distinta. El primer paso es el evento que dispara el proceso, el segundo paso, en esta ocasión, es de tipo "ScriptComponent - DataImport" el cual ejecuta un "script" que coge como parámetro un fichero.

Como parámetros indicaremos el nombre del "script" y la localización del fichero a importar. Una vez creado el proceso se puede automatizar para que lo haga cada cierto tiempo, en nuestro caso vamos a lanzarlo a través del evento que hemos creado:

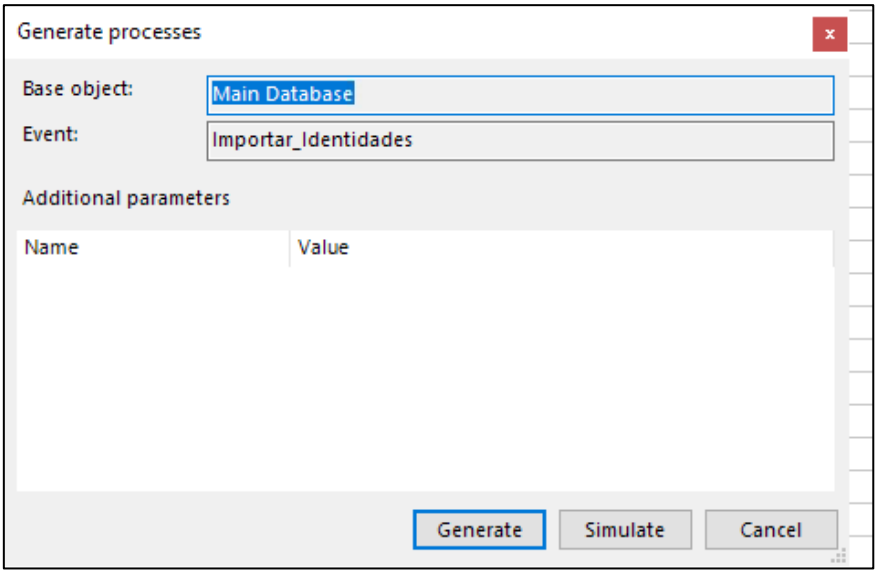

Ilustración 32: Interfaz para lanzar un evento. Fuente: Elaboración propia.

Desde la aplicación JobQueueInfo podemos monitorizar la ejecución del proceso:

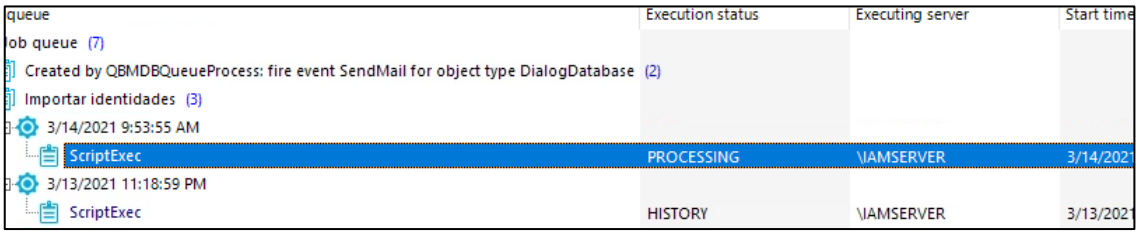

Ilustración 33: Monitorización del proceso. Fuente: Elaboración propia.

Esta aplicación te permite ver el estado del sistema. En su interfaz aparecen unas gráficas que muestran cuantos procesos se están ejecutando, cuantos se han quedado congelados y cuantos han terminado. Además puedes conocer el estado de todos los servidores sincronizados con One Identity. Es importante saber que cuando un cambio se realiza en la base de datos, el sistema bloquea la ejecución de todos los procesos hasta que el sistema sea compilado desde la aplicación : "DbCompiler". Una vez finalizado se quedará en el historial de procesos.

Si regresamos a la aplicación "Manager" podemos revisar que los cambios se han originado:

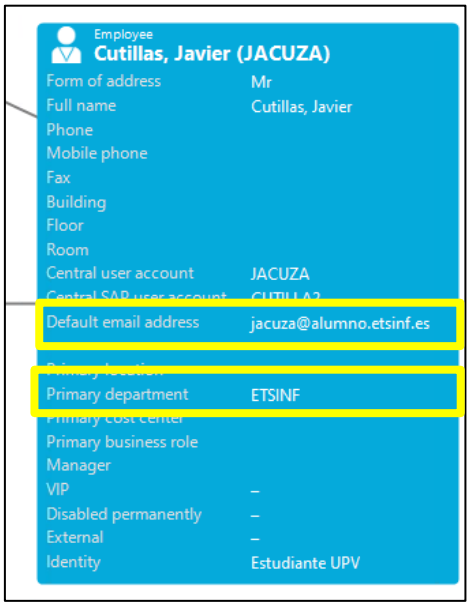

Ilustración 34: Vista del usuario jacuza desde "manager". Fuente: Elaboración propia.

Uno de los principales cambios que podemos observar es que ahora se ha calculado la cuenta de correo. Si entramos dentro del objeto y vamos a la pestaña "Organizational" podemos confirmar que tiene un departamento asignado.

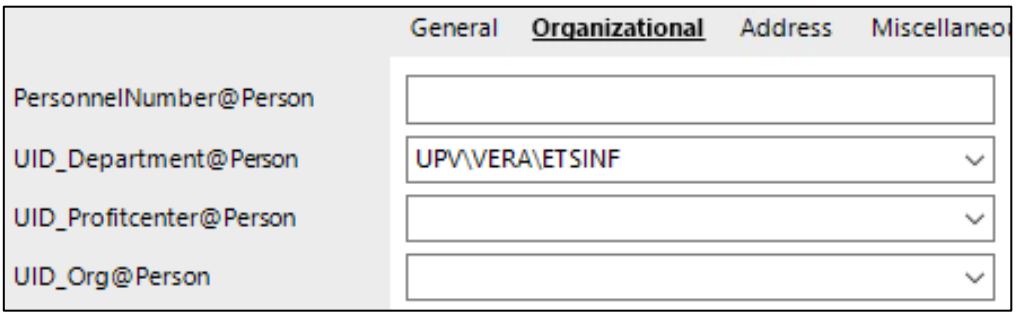

Ilustración 35: Atributos del usuario jacuza.Fuente: Elaboración propia.

Si cambiamos la vista a "Organizations" podemos ver las personas que están asignadas a los distintos departamentos. En el caso de la Facultad de Informática:

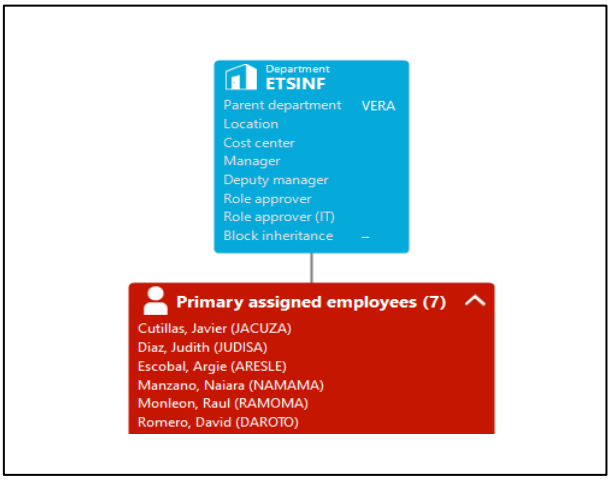

Ilustración 36: Vista membresía de la facultad "ETSINF".
Fuente: Elaboración propia.

Como dato adicional, el servidor del cliente cuenta con una aplicación llamada "Object Browser" que permite a los trabajadores del sistema realizar consultas de datos y tener una vista esquematizada de todas las tablas. Además, podemos añadir una opción para consultas sql. En este caso hemos generado una consulta que devuelve todos los datos de los empleados del sistema. Esto puede ser muy útil para generar informes periódicamente.

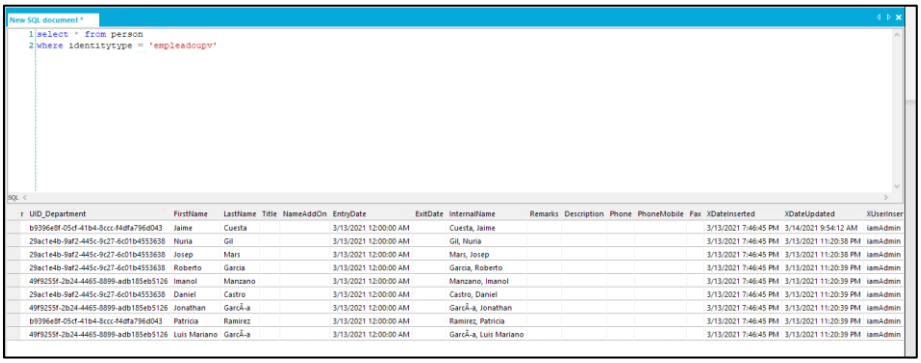

Ilustración 37: Consulta desde la aplicación "Object Browser".
Fuente: Elaboración propia.

5.3.2.2 Desde aplicación de recursos humanos

Existen numerosos softwares que utilizan los negocios y que ayudan al personal a realizar tareas de recursos humanos. En ellas se pueden gestionar el alta de nuevos empleados. Para esta PoC utilizaremos una aplicación DEMO muy sencilla y totalmente customizada. La única función que tiene es un formulario que da de alta a un nuevo empleado.

Lo primero que debemos hacer es sincronizar esta aplicación con One Identity. Para ello deberemos usar el synchronizationEditor. Concretamente, utilizaremos el conector SCIM para poder conectarnos a una aplicación customizada. Se abrirá un asistente que nos ayudará con la sincronización. Debemos vincular los datos del formulario con los campos de la tabla "Person" de One Identity.

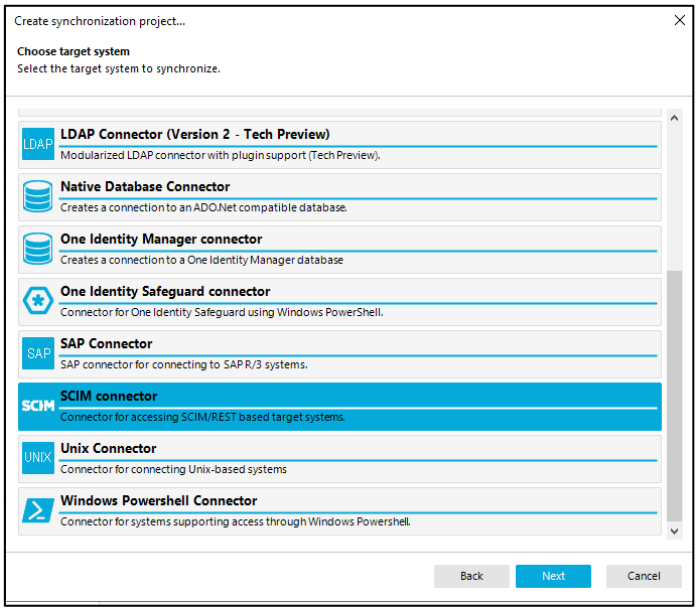

Ilustración 38: Interfaz synchronizationEditor.
Fuente: Elaboración propia.

Desde la aplicación HR podemos rellenar el formulario para dar de alta a un nuevo trabajador de la UPV.

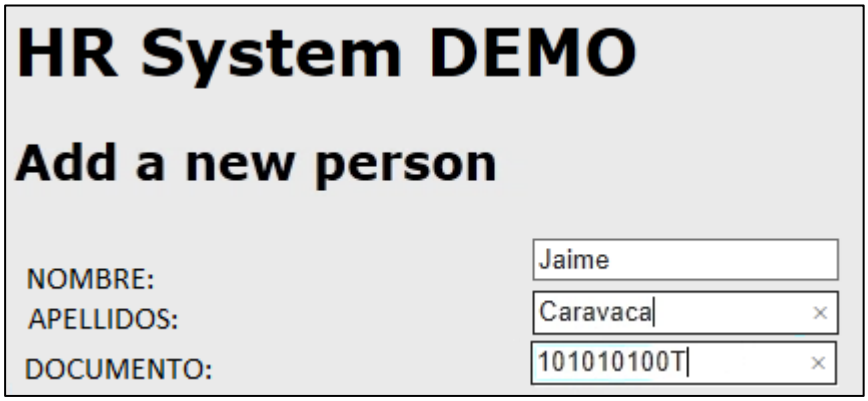

Ilustración 39: Aplicación HR. Fuente: Elaboración propia.

El siguiente paso es establecer unos horarios en la cual esta sincronización se va a llevar a cabo. Para agilizar, podemos ejecutar manualmente la sincronización desde One Identity. Como podemos observar, el usuario ha sido generado correctamente.

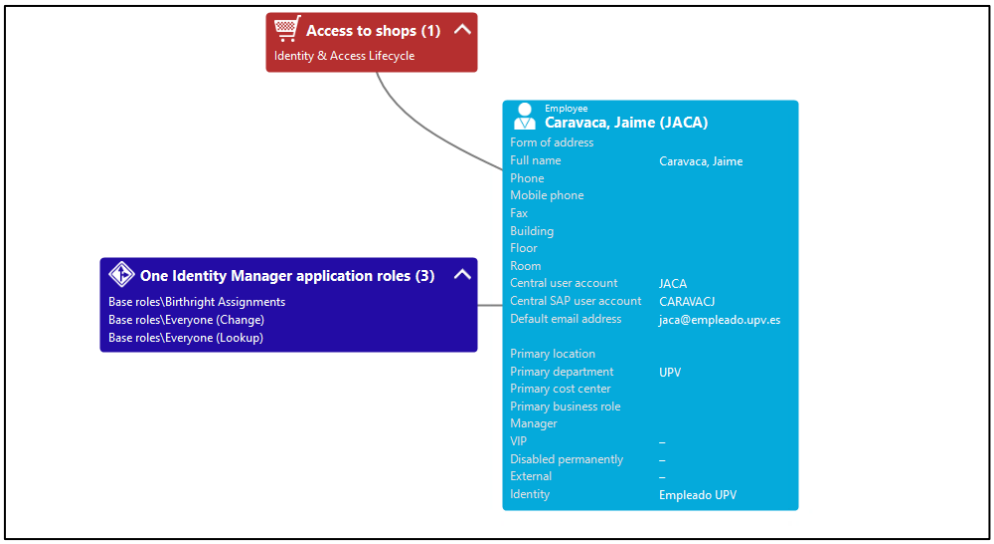

Ilustración 40: Visualización usuario jaca desde "manager". Fuente: Elaboración propia.

5.3.3 Implementación de Roles y sincronización AD

El Gestor de identidades proporciona un modelo RBAC [7] completo para gestionar las asignaciones de acceso. Este modelo de seguridad es muy flexible dentro de One Identity, ya que podemos elegir un comportamiento de herencia de derechos descendente o ascendente.

Los roles son un recurso que permiten facilitar el manejo del aprovisionamiento de una identidad. La asignación de la pertenencia a los roles puede hacerse manualmente, a través de una solicitud con aprobación y comprobación del cumplimiento, o automáticamente mediante reglas. Las reglas pueden basarse en el valor de los atributos del perfil de la identidad o en las asignaciones a otras estructuras (departamentos, localizaciones, etc.). Todas las asignaciones pueden modificarse en respuesta a eventos del ciclo de vida como: alta de una cuenta, baja de una cuenta, y reorganización.

Para esta PoC vamos a diseñar una estructura que nos permita generar una cuenta AD por cada identidad. Esta cuenta de AD se creará en grupos distintos dependiendo del atributo "identityType" de cada identidad de la tabla "Person". Para ello crearemos roles dinámicos que comprobarán cada cierto tiempo el valor del atributo. Si el atributo tiene el valor "empleadoUPV" se le generará una cuenta en directorio activo en el grupo "UPV - Empleados" y viceversa. Para realizar este proceso usaremos una funcionalidad de One Identity llamada "account definition". Cuando una definición de cuenta AD está asignada a una identidad, esta crea una cuenta en directorio activo.

[7] RBAC (Role based access control) : Control de accesos basados en roles.

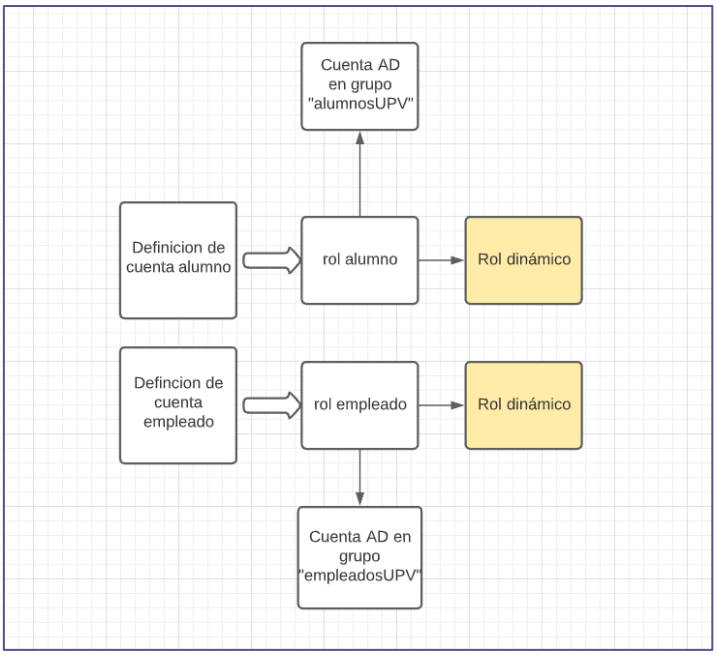

Ilustración 41: Esquema de roles. Fuente: Elaboración propia.

Por tanto, cuando una identidad rellene su campo "identityType", automáticamente se le asignará un rol dinámico. Este rol asignará a la identidad una definición de cuenta de AD, y finalmente se generará la cuenta en Directorio Activo.

Antes de empezar con la creación de dicha estructura debemos sincronizar One Identity con directorio activo desde el SynchronizationEditor.

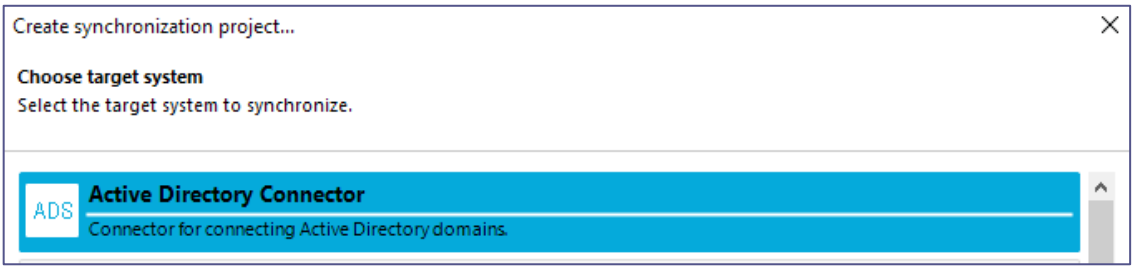

Ilustración 42: Conector AD en synchronizationEditor. Fuente: Elaboración propia.

Como en anteriores sincronizaciones, se abrirá un asistente que nos ayude con toda la configuración. El siguiente paso es gestionar que atributos queremos que se envien hacia el directorio activo y viceversa. En nuestro caso, lo vamos a dejar por defecto, el cual envía la información básica de las cuentas.

Antes de sincronizar con el sistema, en el servidor AD creamos los dos grupos correspondientes: UPV - Empleados y UPV - Alumnos.

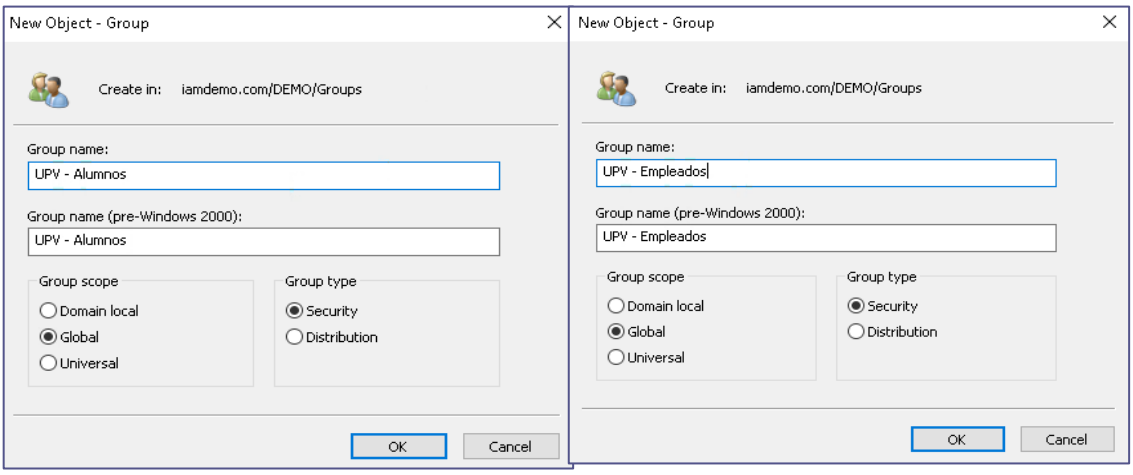

Ilustración 43: Creación grupos en AD. Fuente: Elaboración propia.

Cuando ejecutemos la sincronización , podremos ver estos dos grupos creados en la tabla "ADSGroup" de One Identity. Para ello podemos verlo desde la aplicación "manager":

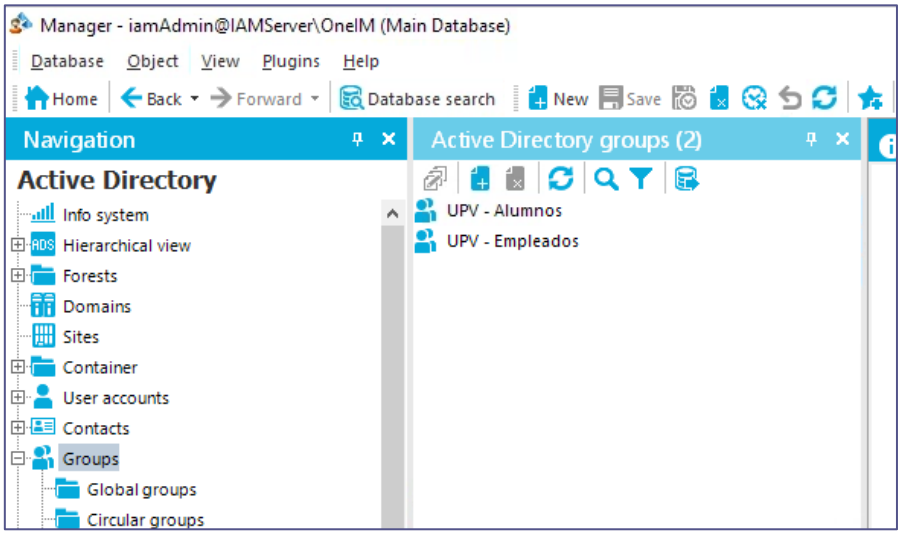

Ilustración 44: Vista AD en One Identity. Fuente: Elaboración propia.

El siguiente paso es generar las definiciones de cuenta AD. Para ello indicamos el nombre (una por tipo), la tabla donde se genera esta cuenta, el servidor y el grado de manejo, en nuestro caso "full managed", el cual nos permite modificar todos los atributos de una cuenta AD.

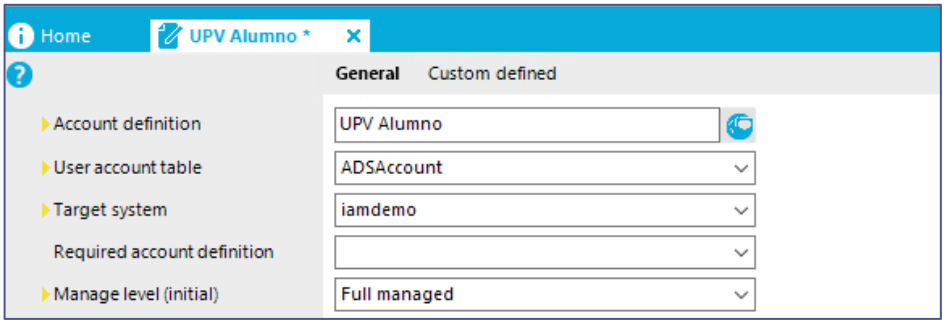

Ilustración 45: Creación definición de cuenta. Fuente: Elaboración propia.

Si asignáramos en este momento la definición de cuenta a una identidad cualquiera, crearía una cuenta en directorio activo que colgaría directamente desde el dominio, ya que todavía no hemos configurado el grupo donde deben pertenecer.

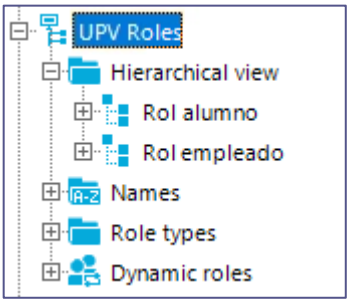

Ilustración 46: Jerarquía roles. Fuente: Elaboración propia.

El siguiente paso es crear los dos roles. Como se pueden generar roles para otras acciones, la mejor práctica es tener organizada la pestaña de roles. En este caso vamos a crear un rol padre llamado "UPV Roles", y sus hijos serán los roles que usaremos para asignar la definición de cuenta. Para cada rol que hemos creado es necesario generarles un rol dinámico asociados a ellos para que compruebe el atributo "identitytype" de cada identidad.

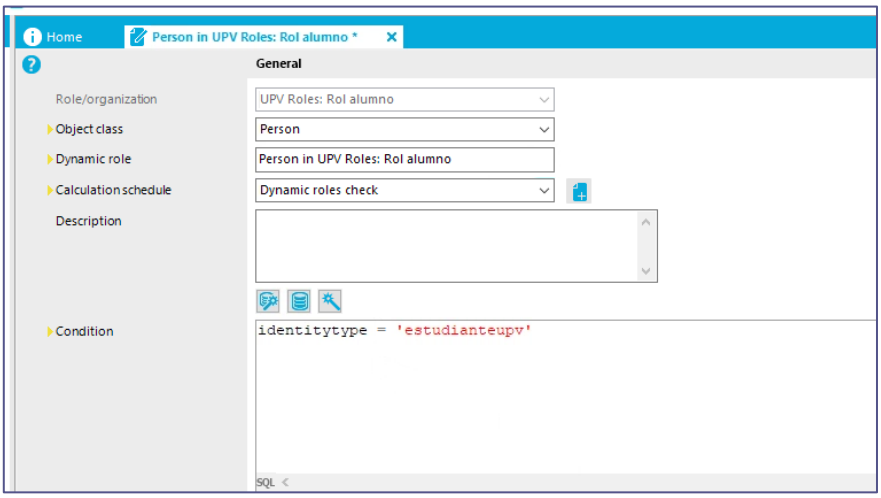

Ilustración 47: Creación rol dinámico. Fuente: Elaboración propia.

Como se puede apreciar en la imagen superior, para el rol alumno la condición es que el atributo "identitytype" de la tabla "Person" sea estudianteupv. El último paso para terminar esta configuración es asignarle al rol la definición de cuenta, y el grupo de directorio activo en la que quiere que se genere la cuenta. Este sería el resultado final:

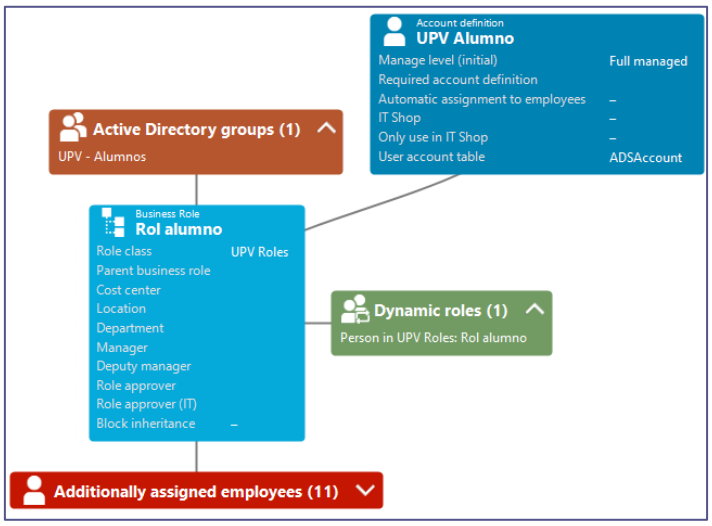

Ilustración 48: Vista Rol alumno. Fuente: Elaboración propia.

Como se observa, se han asignado adicionalmente 11 empleados a este rol. Esto ha sido gracias al rol dinámico que hemos configurado. Si nos dirigimos al servidor de directorio activo podemos ver que efectivamente se han sincronizado las cuentas en la ubicación correcta:

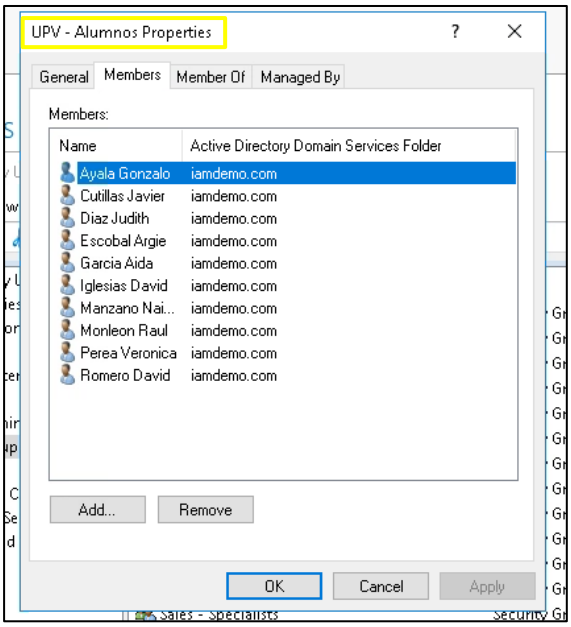

Ilustración 49: Vista AD membresía grupo. Fuente: Elaboración propia.

Si cambiamos la vista al gestor de identidades, podemos apreciar que se han asignado de forma automática la definición de cuenta y una cuenta AD.

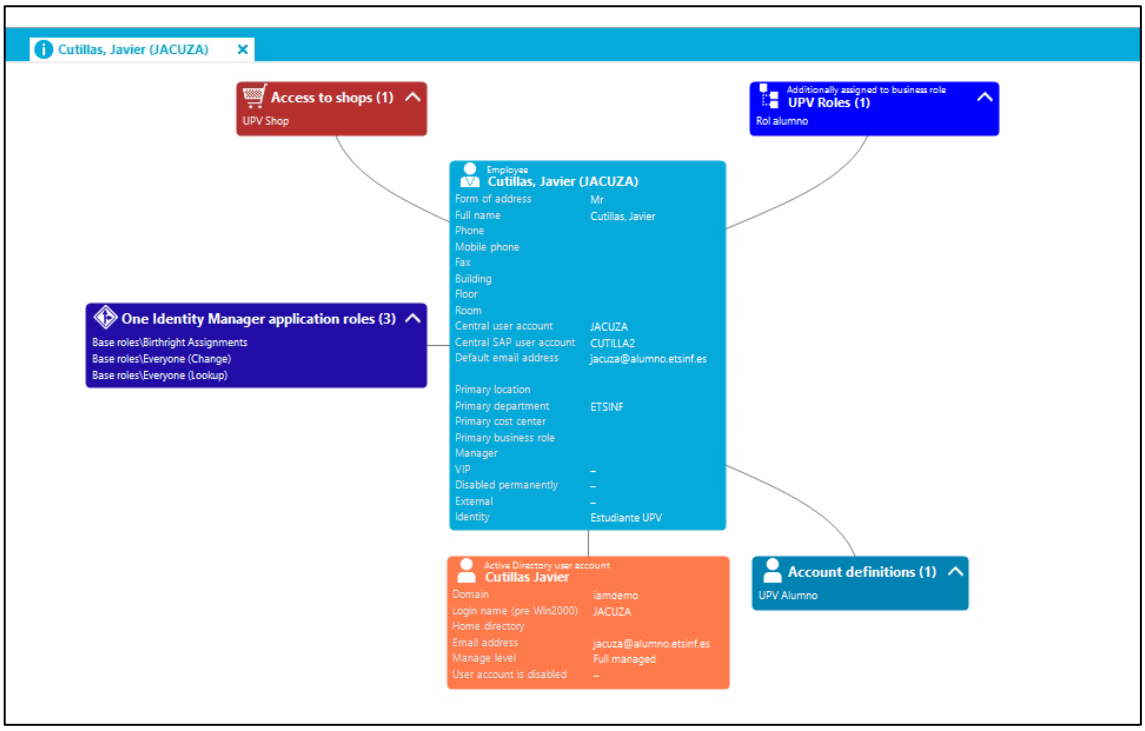

Ilustración 50: Vista usuario jacuza. Fuente: Elaboración propia

El resultado conseguido es el aprovisionamiento de la cuenta automático, sin que ninguna persona intervenga en ella.

Como se comenta previamente, no solo gracias a los roles podemos conseguir este aprovisionamiento. Podemos asignar grupos AD de licencia basándonos en la membresía del departamento. En nuestro caso, todos los alumnos de la facultad de informática necesitan licencias de aplicaciones de Google, Office 365, Visual Studio, VmWare y de Java. Para ello, desde la aplicación "manager" podemos asignar estos grupos de AD sobre el departamento determinado. Esto producirá que todos las personas que sean miembros de ese departamento, y que además cuenten con una cuenta de Directorio Activo, se les ubicará dentro de ese grupo en AD.

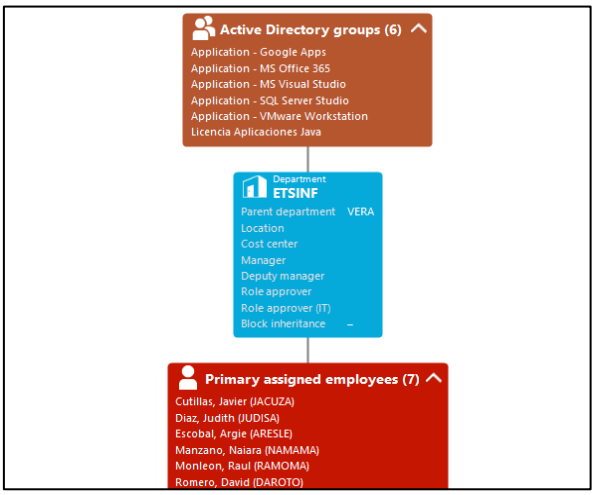

Ilustración 51: Vista del departamento ETSINF. Fuente: Elaboración propia.

Si observamos la cuenta de Directorio Activo de JACUZA en One Identity observamos que las asignaciones a los grupos se han realizado automáticamente:

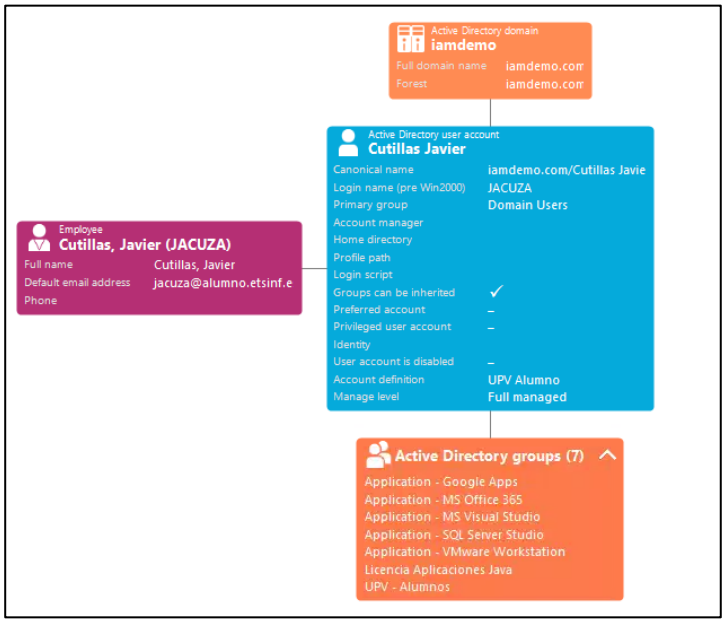

Ilustración 52: Cuenta del usuario AD en "manager". Fuente: Elaboración propia.

Realizamos el mismo proceso con el resto de departamentos y sus licencias determinadas.

5.3.4 Recursos adicionales : Becarios

El gestor de identidades cuenta con una funcionalidad llamada "resources". Estos recursos se pueden asignar a las identidades. En esta PoC se crea un recurso denominado "becario". Su objetivo es aprovisionar a todos los empleados que son becarios con la licencia de aplicaciones de Google.

Para saber si un empleado es becario debemos fijarnos en el atributo "CustomProperty03" de la tabla "Person" de cada identidad. Para ello generaremos un rol dinámico que compruebe que el valor de ese campo sea igual a "becario".

El siguiente paso es crear un recurso. Todos los recursos en One Identity pertenecen a la tabla "QERResource". Cuando una identidad tiene un recurso, se guarda en una tabla denominada "PersonHasQERResource". Dicha tabla contiene dos columnas únicamente: el UID de la tabla "Person", y el UID de la tabla "QERResource".

Desde el "manager" podemos generar dicho recurso y asignarle el rol dinámico y el producto de la licencia de Google:

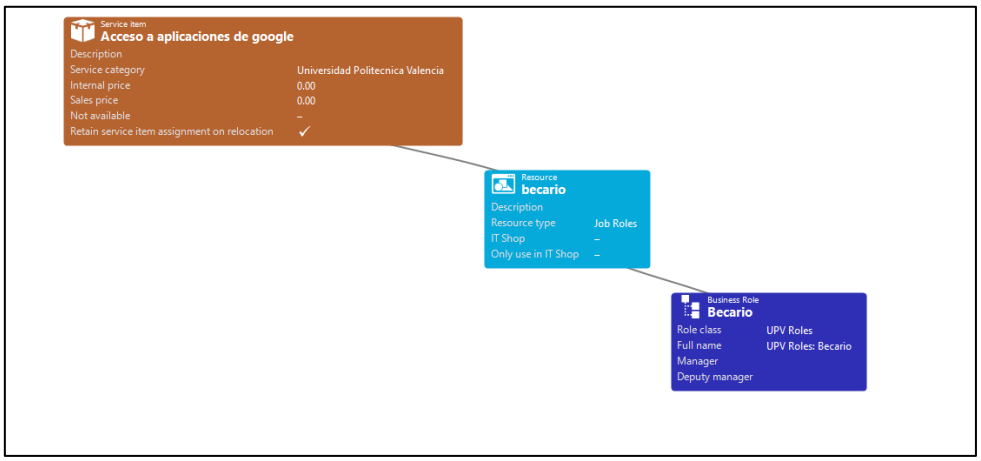

Ilustración 53: Vista recurso becario. Fuente: Elaboración propia.

y automáticamente el aprovisionamiento es calculado:

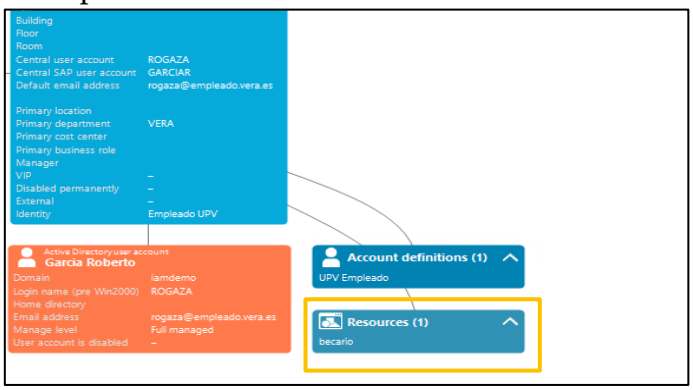

Ilustración 54: Vista usuario rogaza. Fuente: Elaboración propia.

5.3.5 Portal Web

El objetivo de este último apartado es generar un portal web exclusivo para los empleados. Este portal te permite pedir accesos a recursos, entre otros.

Los empleados del campus de Gandía trabajan en un entorno Google, y los empleados de Alcoy en un entorno de Office 365. La finalidad de la creación del portal es que, si un trabajador del campus de Gandía se siente más cómodo usando Office 365, pueda pedirlo desde la web. Esta petición será aprobada por el manager del departamento al que pertenece.

Para empezar esta configuración vamos a asignar managers a los campus. En ambos campus asignaremos como manager a Jaime Caravaca (usuario importado desde la aplicación HR)

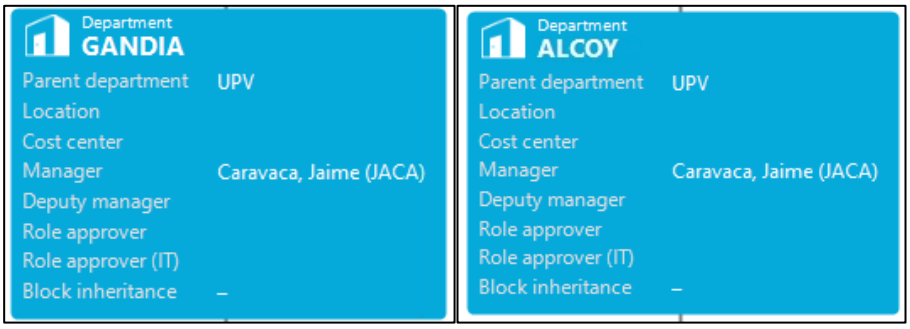

Ilustración 55: Vista atributos de los campus. Fuente: Elaboración propia.

En One Identity se pueden crear diversas "tiendas" donde poder crear productos que posteriormente pueden ser pedidas por las personas que pertenecen a dicha tienda.

Para ello, desde la aplicación "manager", nos dirigiremos a IT SHOP y crearemos una nueva tienda con la condición de que solo pueden acceder a ella los empleados utilizando el atributo identitytype, como hemos visto anteriormente.

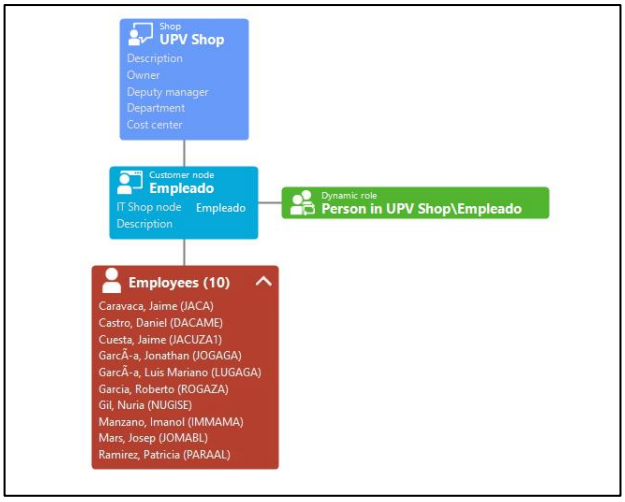

Ilustración 56: Vista de los accesos al portal web.
Fuente: Elaboración propia.

Como observamos en la ilustración 56, automáticamente se le han dado acceso a las personas que cumplían dicha condición. El siguiente paso es configurar un flujo de aprobación.

El gestor de indentidades permite customizar los flujos de aprobación, pudiendo añadir más de un paso y más de un cálculo. En nuestro caso, el flujo contendrá un paso, y será el manager del campus quien apruebe la petición.

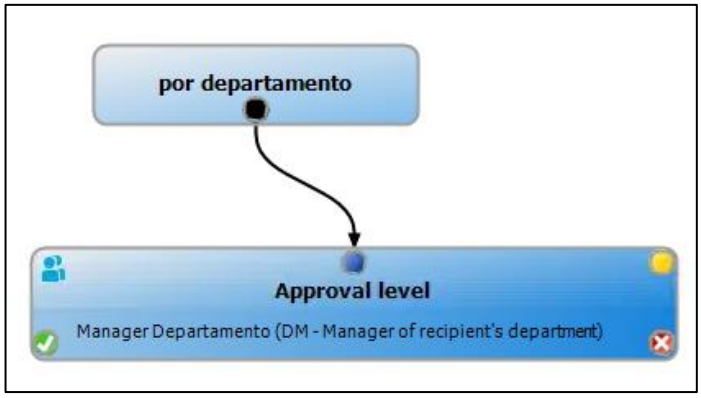

Ilustración 57: Flujo de aprobación. Fuente: Elaboración propia.

Una vez creado el flujo, podemos asignárselo a la política de aprobación:

Ilustración 58: Propiedades de la política de aprobación. Fuente: Elaboración propia.

El último paso es crear un producto dentro de la tienda que hemos creado. El producto estará asignado al grupo de licencia correspondiente. A este producto deberemos asignarle el flujo de aprobación. Este sería el resultado:

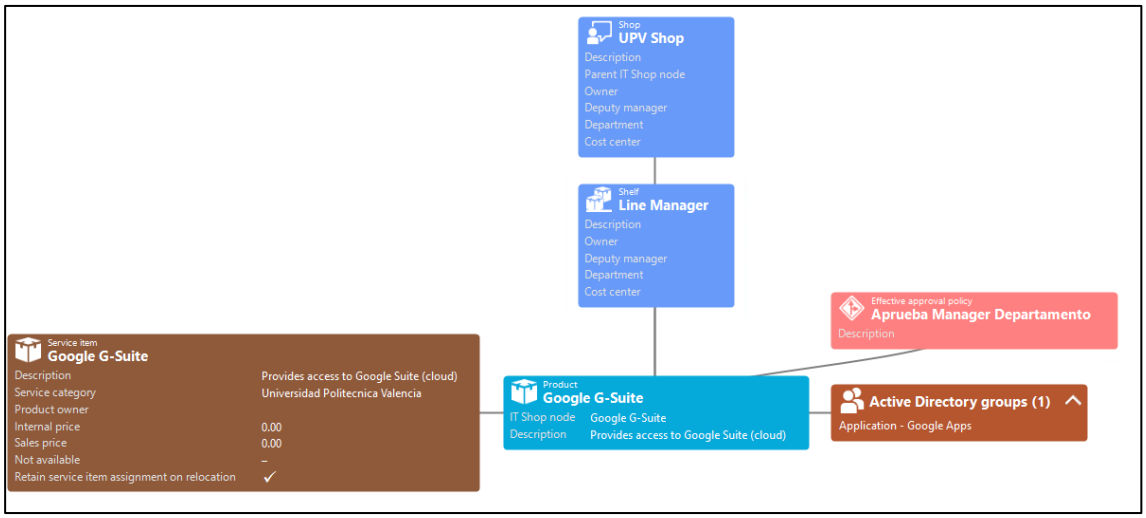

Ilustración 59: Producto del portal web. Fuente: Elaboración propia.

Para poder acceder a la web, debemos definir una contraseña en la tabla "Person":

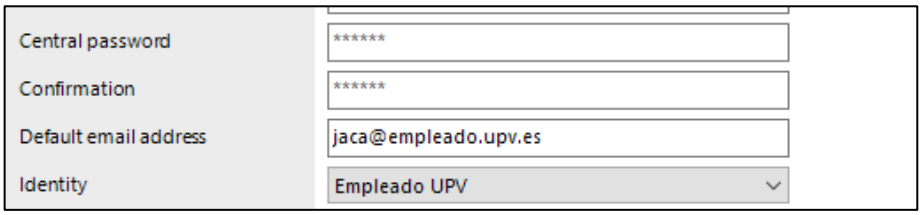

Ilustración 60: Propiedades de jaca. Fuente: Elaboración propia.

Para terminar, vamos a comprobar su funcionamiento y como se visualizaría en el sistema tras la petición del producto. Para ello utilizaremos al empleado Jaime Cuesta que pertenece al campus de Alcoy.

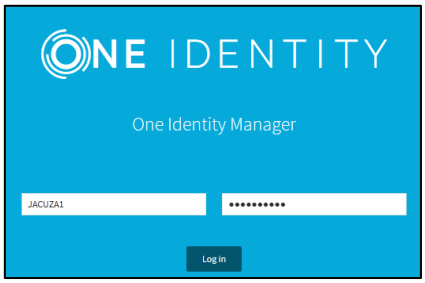

Ilustración 61: Portal web One Identity. Fuente: Elaboración propia.

Dentro de la pantalla principal deberemos seleccionar "Start a new request":

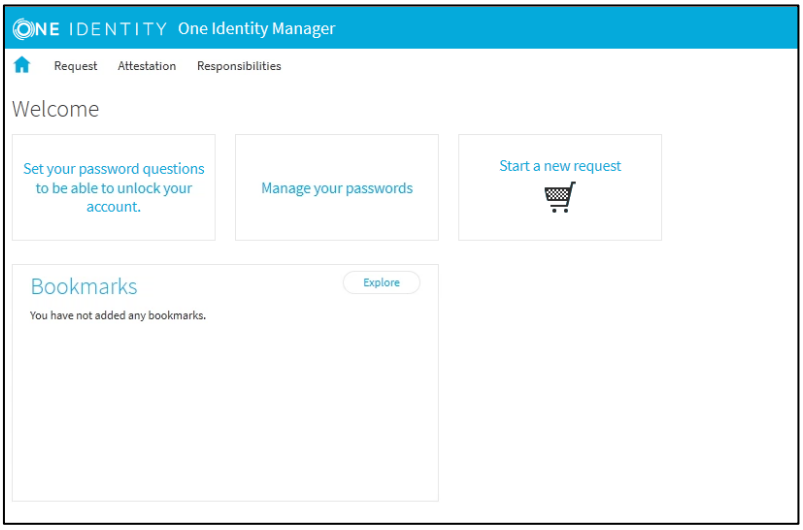

Ilustración 62: vista "Home" portal web. Fuente: Elaboración propia.

Seguidamente nos aparecerán las tiendas donde este usuario tiene acceso, y seleccionaremos la denominada como "Universidad Politécnica de Valencia". Una vez dentro nos aparecerán todos los productos y seleccionaremos el que hemos creado llamado "Google G-Suite":

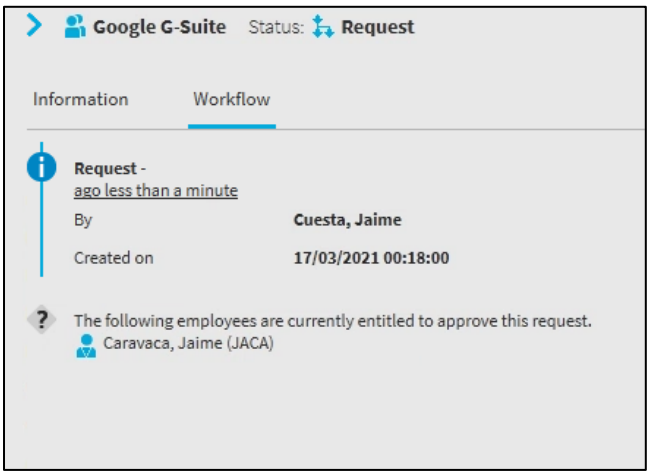

Ilustración 63: Información de la petición. Fuente: Elaboración propia.

Como indica la imagen superior, podemos ver la información del proceso de aprobación.

Ahora entraremos a la web como Jaime Cuesta:

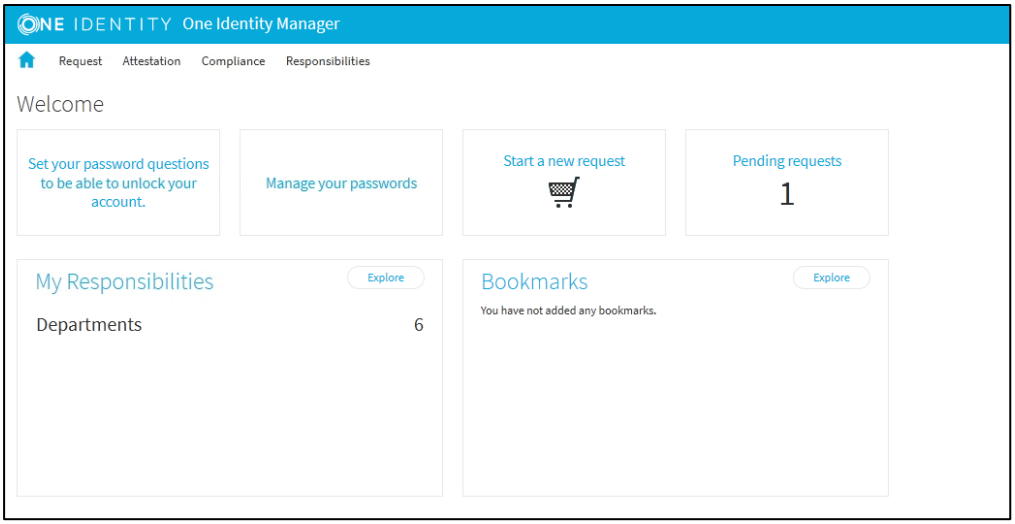

Ilustración 64: vista "Home" portal web. Fuente: Elaboración propia.

Jaime, al ser manager, tiene más opciones habilitadas en el portal. Podemos ver que tiene una petición pendiente. Ahora podemos decidir si aprobarla o declinarla.

Ilustración 65: Aprobación de la petición. Fuente: Elaboración propia.

Una vez aprobada, One Identity guarda todo el procedimiento en una tabla. Si entramos a la aplicación "manager" podemos ver la asignación adicional del producto (Active product requests).

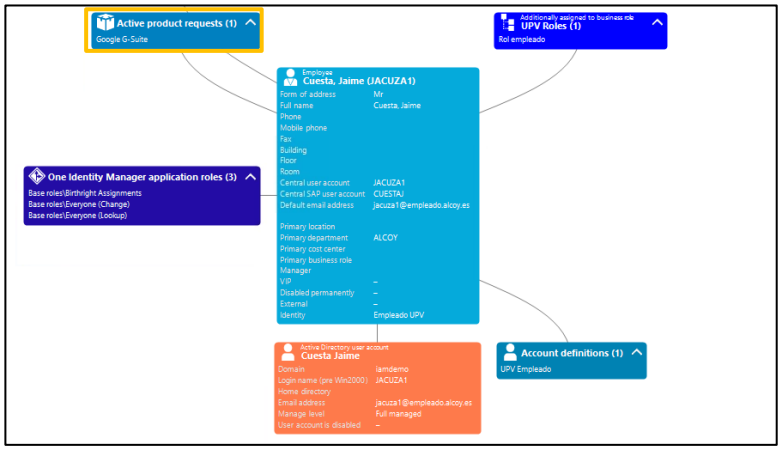

Ilustración 66: vista jacuza1 en "manager". Fuente: Elaboración propia.

Si entramos dentro de esa asignación obtendremos más detalles:

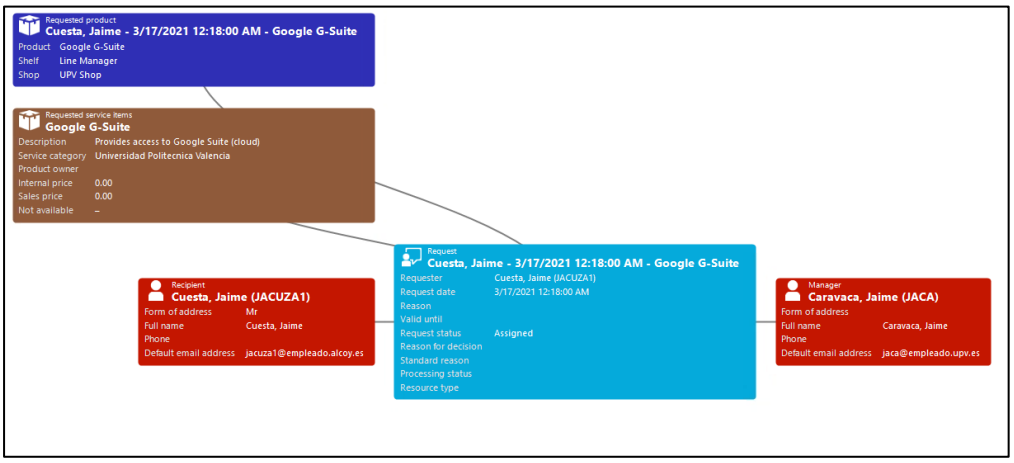

Ilustración 67: Detalles de la petición. Fuente: Elaboración propia.

Podemos observar información del solicitante y de su manager. Además, vemos fecha y hora, en qué tienda se ha realizado, qué flujo de aprobación se ha llevado a cabo, y las características del producto que ha sido asignado.

El objetivo de esta petición era que Jaime pudiera tener acceso al entorno de Google. Si entramos dentro de su cuenta de directorio activo en el gestor de identidades, podemos observar como esta asignación se ha generado satisfactoriamente:

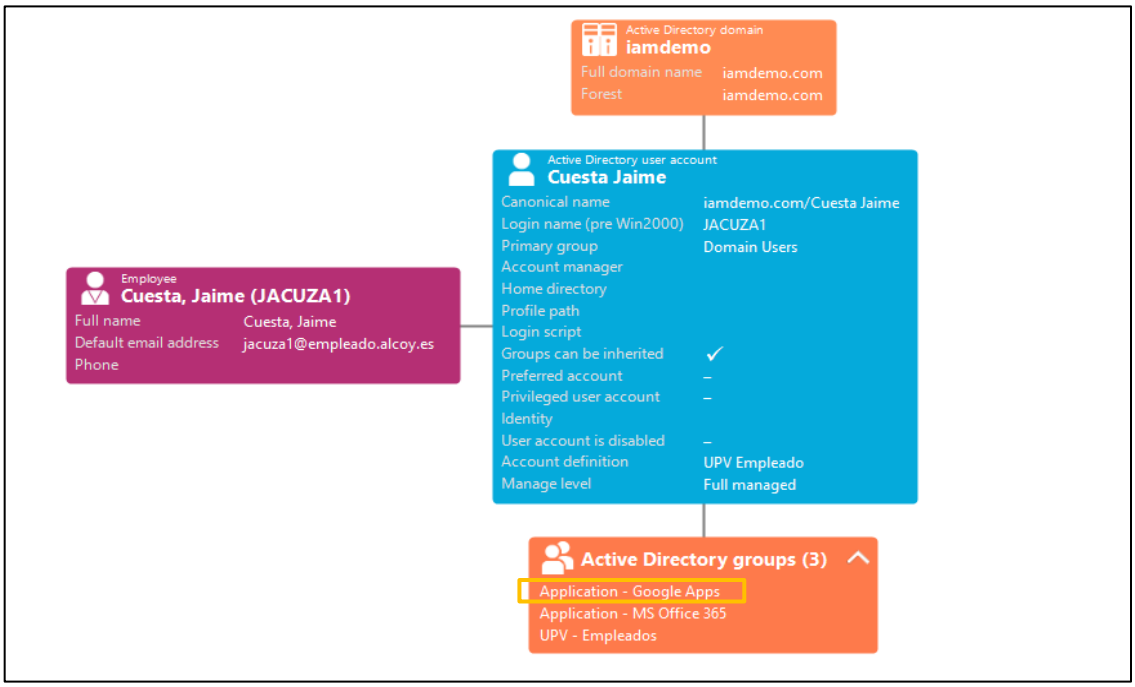

Ilustración 68: Vista de la cuenta AD del usuario jacuza1. Fuente: Elaboración propia.

Ahora Jaime puede utilizar tanto un entorno de Office 365 como de Google.

Capítulo 6: Validación y análisis de despliegue

Realizada la demostración de la prueba de concepto, en este capítulo se analiza el despliegue de One Identity como gestor de identidades de la Universidad Politécnica de Valencia.

Es recomendable la disposición de dos equipos: un esquipo enfocado a las incidencias de tercer nivel y un segundo equipo que se encargará de las tareas de desarrollo y evolutivo del gestor.

El primer paso del despliegue consistirá en la instalación de la solución en los entornos de desarrollo y producción. También se aconseja instalar un entorno de test o de calidad donde validar las nuevas evoluciones previo paso a producción:

- **Entorno de desarrollo**: En este servidor es donde se llevarán a cabo gran parte de las tareas de configuración de la iniciativa IGA. Aquí se realizarán las pruebas y validaciones para no tener un impacto imprevisto en producción. Hay que tener en cuenta que estas herramientas pasan a gobernar el entorno y, por lo tanto, cualquier error de configuración puede tener un alto impacto en la producción de la UPV.
- **Entorno de producción**: Este es el servidor principal encargado de gestionar las identidades de la UPV y de todos los procesos que conlleva.
- **Entorno de test o de calidad**: Este servidor sirve para la realización de pruebas y validación previo paso al entorno de producción, que será lo más se asemeje al entorno real de producción.

El siguiente paso es la carga de las distintas aplicaciones que utiliza la universidad. Una vez realizada la carga, se procederá a realizar un análisis de las cuentas, lo que incluye un análisis de riesgos inicial, detectar duplicados, detección de cuentas huérfanas, cuentas con información incompleta, etc. Este análisis permitirá simplificar la operativa posterior de proyecto y, por ende, sentar las bases de una correcta ejecución de proyecto.

El último paso sería realizar la configuración del ciclo de vida de las cuentas tal y cómo se ha detallado en la PoC.

Para terminar, una iniciativa IGA debe ser estable en el tiempo, ya que se debe avanzar en su implementación de forma recurrente, por lo que debe tener una fase evolutiva, que será sobre la que se asiente el crecimiento y madurez del Programa IGA

9.1: Presupuesto

En este capítulo se realiza un estudio del presupuesto que conllevaría toda esta implementación final en la Universidad Politécnica de Valencia. Para hacer una estimación, utilizamos los datos ofrecidos por la Wikipedia:

- Administrativos: 1458 personas.
- Profesores: 2646 personas.
- Estudiantes: 28528 personas.

Podemos dividir el presupuesto en dos: Licencia de One Identity y coste de la empresa especializada que realiza la implementación.

One Identity cuenta con diversas licencias que se adaptan a las necesidades del cliente. El precio de las licencias parte desde los $20 por identidad [15]. En nuestro caso, esta licencia se queda escasa para realizar todas las configuraciones que queremos. La licencia apropiada para esta implementación tiene un coste de 38€ por identidad aproximadamente.

Tabla 4: Presupuesto licencias One Identity. Fuente: Elaboración propia.

Profesores	Total
2646	100548
Total	100.548,00 €

Administrativos	Total
1458	55404
Total	55.404,00 €

Estudiantes	Total
28528	1084064
Total	1.084.064,00 €

Descripción	Meses de soporte	Coste unitario total
IDENTITY MANAGER PER MANAGED PERSON 24X7 TERM LICENSE/MAINT	36	38

Total global	1.240.016,00 €

La licencia dura un total de 36 meses y tendría un coste de 1.240.016€. Por tanto, sería un pago a realizar periódicamente cada 3 años.

Tabla 5: Presupuesto empresa promedio enfocada a IAM. Fuente: Elaboración propia.

Fase	Tiempo real (semanas)	Duración (días)	IAM Consultor	
			Tarifa diaria	Total
Consultoría Inicial + PoC	14	70	200 €	14.000 €
Proyecto Implantación	33	165	200 €	33.000 €
Servicio de Mantenimiento	32	160	200 €	32.000 €
Servicio de Evolución y mantenimiento (tiempo de 1 persona full time técnicamente)	115	575	390 €	224.250 €

El siguiente presupuesto corresponde al coste de una empresa promedio enfocada a realizar este tipo de implantaciones:

- Pago primer año: 303.250€
- Pago segundo año: Mantenimiento + evolutivo : 256.250€
- Pago tercer año: Mantenimiento + evolutivo : 256.250€

El coste total para los tres primeros años sería 1.240.016€ (licencia) + 815.750€ (Empresa especializada en gestión de identidades). Esto supone un total de 2.055.766€.

Capítulo 7: Conclusiones

Este proyecto ha demostrado que la base de la seguridad de una organización o empresa es la gestión de identidades. Existen numerosos estudios que demuestran que más del 90% de los incidentes de ciberseguridad se producen por error humano. Muchos de los empleados tienen accesos y permisos que no necesitan; sumado al desconocimiento de ellos, supone una puerta abierta para el ciberdelincuente. Todos estos riesgos aumentan cuando la empresa tiene una estructura de organización compleja [14].

El objetivo de este proyecto es demostrar como un gestor de identidades puede resolver y simplificar muchas de estas dificultades de las organizaciones.

En la prueba de concepto hemos cumplido con todos los requisitos previos:

1. Automatización del ciclo de vida de los empleados y de los estudiantes.
2. Gestión homogénea de estudiantes y empleados. No se necesitan dos sistemas distintos para poder gestionar cada uno de ellos.
3. Sincronización con diferentes sistemas, los cuales hemos podido automatizar.
4. Automatización y división de roles, con el objetivo de asignar los recursos determinados a las personas determinadas.
5. Configuración del portal web para pedir nuevos recursos.

La mayor parte de estas configuraciones corresponden a la parte IGA, comentada en el capítulo 2.

Gracias a esta implementación podemos:

- Reducir la superficie de ataque y mejorar el cumplimiento normativo limitando el acceso a recursos y permisos, así como desactivar cuentas obsoletas.
- Aplicar controles a las identidades para evitar cuentas huérfanas, pérdida de derechos y permisos excesivos.
- Generar alertas e informes sobre identidades, grupos, accesos, membresías, etc.
- Simplificar tareas humanas y optimizar los procesos de negocio.
- Mitigar el error humano y, por lo tanto, el riesgo de ciberataque, gracias a la automatización del ciclo de vida, el cual supone un refuerzo al sistema de seguridad de cualquier organización.

7.1 Relación del trabajo desarrollado con los estudios cursados

Este proyecto está relacionado con el Grado en Ingeniería Informática de la Universidad Politécnica de Valencia. Dentro de un proyecto IAM hay 3 campos principales involucrados:

- Ingeniería de sistemas
- Programación
- Base de datos

Este proyecto está orientado a la rama de Tecnologías de la Información, enfocada a almacenar, recuperar, transmitir y manipular datos, con frecuencia utilizado en el contexto de los negocios u otras empresas.

La administración de dominios en AD (usuarios, grupos, políticas, etc.), automatización y sincronización con One Identity ha sido posible gracias a los conocimientos adquiridos en la asignatura " Administración de Sistemas" .

El desarrollo de procesos y scripts ha sido posible gracias a las siguientes asignaturas:

- "Introducción a la informática y a la programación".
- "Programación".
- "Estructuras de datos y algoritmos".
- "Ingeniería del software".

Todas ellas me han dado los conocimientos para diseñar y desarrollar un software de calidad, y que cumplan con estas características: mantenibilidad, confiabilidad, eficiencia y usabilidad.

Por último, las asignaturas "Bases de datos" y " Tecnología de bases de datos" han sido indispensables para el entendimiento e implementación de este trabajo. El núcleo de un gestor de identidades como One Identity es una gran base de datos compuesta por numerosas tablas de todo tipo. La creación o actualización de tablas, las consultas y la combinación de SQL con Visual Basic han sido imprescindibles para este proyecto.

7.2 Trabajo futuro

En la prueba de concepto no se han realizado muchas configuraciones que son requeridas en un futuro proyecto IGA:

- La creación de los departamentos debe estar sincronizada con un sistema. Si en el futuro una facultad es creada, eliminada o actualizada, el sistema debe estar preparado para realizar el cambio.
- Toda la parte organizativa, como centros de coste, oficinas y departamentos deben seguir la misma sincronización.

- El algoritmo del cálculo del login sólo contempla que se repita el mismo login 9 veces. En un entorno productivo, es necesario optimizarlo para que no haya límite.
- En la PoC no se contempla que un profesor pueda ser estudiante, y viceversa. Para resolver esta logística es necesario el uso de subidentidades. La idea principal sería mantener una identidad principal por persona con los datos básicos que no dependen de su puesto, y que pueda tener tantas subidentidades como posiciones tenga dentro de la universidad. Por ejemplo, para un profesor que imparte clases, que es director de una facultad y estudia una carrera, esta sería su estructura de identidades:

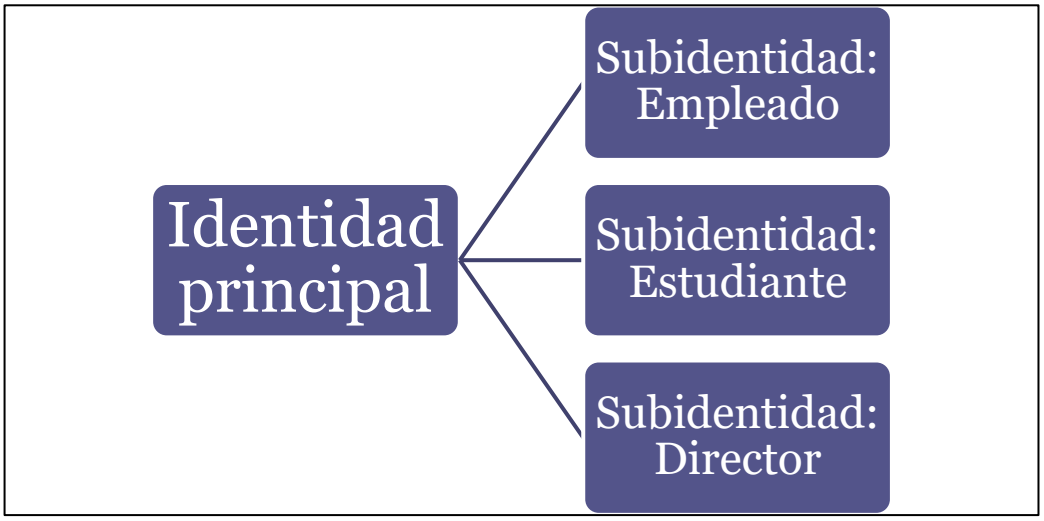

Ilustración 70: Identidad principal y subidentidades. Fuente:Elaboración propia

- Para el portal web, sería recomendado establecer un tiempo de caducidad al producto ofertado. Muchas veces los usuarios acceden a recursos que, a la larga, dejan de utilizar, y esta es una medida más para restringir ese acceso.
- Para el entorno productivo es necesario realizar un manejo de las políticas de contraseña, las cuales deben coincidir con las políticas de los sistemas sincronizados para que no se produzca un conflicto.

Bibliografía

[1] Carbonell, J. (2016, noviembre). La ciberseguridad ante los nuevos desafíos de internet. La importancia del Big data y la colaboración entre entidades. Telefónica. Obtenido de: https://www.telefonica.com/es/web/public-policy/blog/articulo/-/blogs/la-ciberseguridad-ante-los-nuevos-desafios-de-internet-la-importancia-del-big-data-y-la-colaboracion-entre-entidades Fecha de acceso: 1 de mayo de 2021.

[2] Gaedke, M., Meinecke, J., & Nussbaumer, M. (2005, mayo). A modeling approach to federated identity and access management. In *Special interest tracks and posters of the 14th international conference on World Wide Web* (pp. 1156-1157). Obtenido de: https://dl.acm.org/doi/pdf/10.1145/1062745.1062916 Fecha de acceso: 5 de mayo de 2021.

[3] (2018, marzo). ¿QUÉ ES LA GESTIÓN DE IDENTIDADES Y ACCESOS (IAM)?. SYNNEX Corporation. Obtenido de: http://digital.la.synnex.com/que-es-la-gestion-de-identidades-y-accesos-iam Fecha de acceso: 1 de febrero de 2021.

[4] Blanchard, B., Doherty, R., Buck, A., Dahlbom, J., Renshaw, J., & Wendel,S. (2020, junio). Identity and access management. Microsoft. Obtenido de: https://docs.microsoft.com/en-us/azure/cloud-adoption-framework/ready/enterprise-scale/identity-and-access-management#why-we-need-identity-and-access-management Fecha de acceso: 21 de febrero de 2021.

[5] About provisioning. Okta Help Center. Obtenido de: https://help.okta.com/en/prod/Content/Topics/Provisioning/lcm/con-okta-prov.htm#:~:text=Using%20Okta%20to%20provision%20user%20account%20information%20combines,to%20a%20single%20corporate%20user%20ID%20and%20password. Fecha de acceso: 1 de noviembre de 2020.

[6] IGA and PAM: How Identity Governance Administration Connects with Privileged Access Management. Cybersecurity Simplified WALLIX. Obtenido de: https://www.wallix.com/blog/iga-and-pam-how-identity-governance-administration-connects-with-privileged-access-management/#:~:text=Unifying%20IGA%20and%20PAM%20enables%20a%20central%20locus,are%20part%20of%20a%20single%20access%20control%20chain. Fecha de acceso: 1 de junio de 2021.

[7] Mathers, B. (2021, junio). Privileged Access Management for Active Directory Domain Services. Microsoft. Obtenido de: https://docs.microsoft.com/en-us/microsoft-identity-manager/pam/privileged-identity-management-for-active-directory-domain-services Fecha de acceso: 15 de febrero de 2021.

[8] Grasmayer, M. (2019, agosto). What is Okta and what do you use it for?. Workspace365. Obtenido de: https://workspace365.net/en/what-is-okta-and-what-do-you-use-it-for/#:~:text=Okta%20is%20a%20platform%20in%20the%20Identity-as-a-Service%20%28IDaaS%29,Okta%20is%20always%20within%20reach%2C%20but%20safely%2C%20 Fecha de acceso: 1 de noviembre de 2020.

[9] Savaram, R. (2021, mayo). CyberArk Tutorial. mindmajix. Obtenido de: https://mindmajix.com/cyberark-tutorial Fecha de acceso: 1 de noviembre de 2020.

[10] Budnik, M. (2018, mayo). CyberArk Named a Leader in Gartner's Inaugural 2018 Magic Quadrant for Privileged Access Management. Cyberark. Obtenido de: https://www.cyberark.com/resources/blog/cyberark-named-a-leader-in-gartner-s-inaugural-2018-magic-quadrant-for-privileged-access-management Fecha de acceso: 15 de noviembre de 2020.

[11] Cameron, A., & Williamson, G. (2020). Introduction to IAM Architecture. IDPro Body of Knowledge, 1(2). Obtenido de: https://bok.idpro.org/article/38/galley/46/view/ Fecha de acceso: 1 de diciembre de 2020.

[12] About One Identity. One Identity. Obtenido de: https://www.oneidentity.com/company/ Fecha de acceso: 15 de diciembre de 2020.

[13] Abele, H. (2020, febrero). How to use Identity Manager. Youtube (10 videos). Obtenido de: https://www.youtube.com/playlist?list=PL242czeZwlAkfobKXhLV9cjkU_UW9YX-y Fecha de acceso: 15 de diciembre de 2020.

[14] El error humano, principal aliado de los ciberdelincuentes. aiwin. Obtenido de: https://aiwin.io/es/blog/2020/10/05/error-humano-ciberdelincuentes/ Fecha de acceso: 1 de julio de 2021.

[15] Lavi, S. (2021, marzo). What is Quest One Identity's cost rating?. Itqlick. Obtenido de: https://www.itqlick.com/quest-one-identity-manager/pricing Fecha de acceso: 1 de julio de 2021.