

Proyecto Fin de Ciclo
Desarrollo de Aplicaciones Multiplataforma

**ERP para la gestión de
expulsiones en un Instituto de
Enseñanza Secundaria**

Alejandro M. Ogalla López

10-06-2022

Índice

1	Introducción	1
1.1	Motivación	1
1.2	Planteamiento inicial	1
1.3	Prototipo de la aplicación	2
1.4	Aplicaciones similares	3
1.5	Tecnologías utilizadas	4
2	Descripción del resultado	4
3	Instalación	5
3.1	Manual de instalación	6
4	Diseño funcional de la aplicación	6
5	Desarrollo	8
5.1	Ventajas e inconvenientes del software escogido	8
5.2	Dificultades afrontadas en el proyecto	9
6	Pruebas	10
6.1	Pruebas de volumen	11
6.2	Pruebas de estrés	11
6.3	Pruebas de usabilidad	11
7	Distribución	11
8	Manuales de uso de la aplicación	12
8.1	Activación y uso del bot de Telegram	12
9	Conclusiones	14
9.1	Posibles mejoras	14
10	Índice de imágenes	16
11	Bibliografía	17

1 Introducción

En el presente documento se describe el proceso realizado en el desarrollo y uso de una aplicación de gestión de un Instituto de Enseñanza Secundaria.

Para complementar algunos procesos, especialmente la comunicación de incidencias a la jefatura de estudios, se ha utilizado también la aplicación de mensajería [Telegram](#), para lo que ha sido necesaria una integración entre ambas aplicaciones.



Figura 1: Integración de Telegram en Odoo

1.1 Motivación

La idea de desarrollar un software de gestión como este surge de la experiencia personal de mi pareja en la tramitación de amonestaciones y expulsiones en un centro de Enseñanza Secundaria en Andalucía. En dicho centro, que suponemos no será el único, esta gestión se realiza fundamentalmente de forma manual y almacenando todos los registros en papel, por lo que, utilizando este centro como referencia, un programa como el planteado podría ser de gran ayuda en la manipulación de datos y en la automatización de procesos.

1.2 Planteamiento inicial

Inicialmente, la idea original consistía en la gestión de las matriculaciones de alumnos, del profesorado y de las asignaturas impartidas en el centro. En esta gestión se incluía la tramitación de amonestaciones y expulsiones de los alumnos y de la automatización de los procesos que conllevan dichas sanciones.

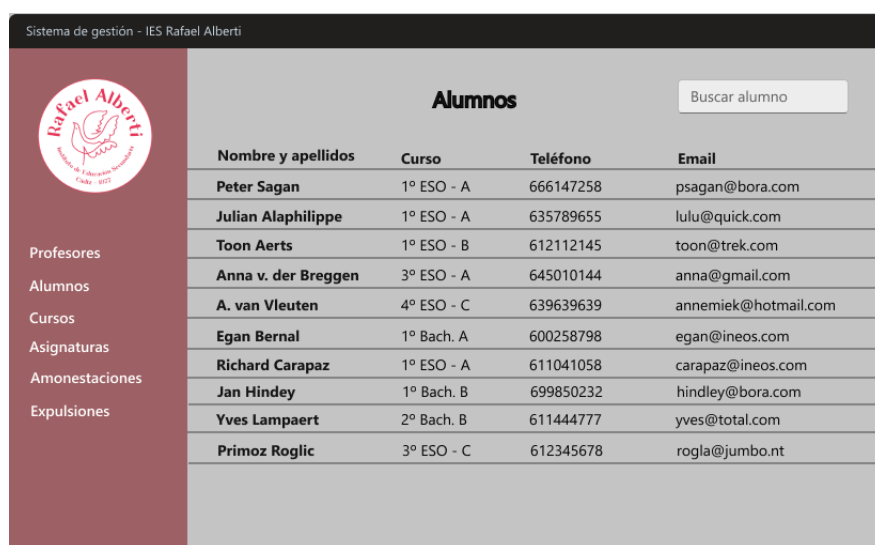
También se pretendía comunicar a los profesores y las familias de los alumnos expulsados, para que así todos los involucrados en dicha expulsión tuvieran conocimiento de ella. El aviso a las familias podía realizarse de diferentes formas: por SMS, por correo electrónico, por mensaje de Telegram, etc., mientras que el mensaje a los profesores se iba a realizar, también por cualquiera de dichos canales, incluyendo un formulario online en el que rellenar la tarea que el alumno debe

realizar durante su período de expulsión. La aplicación debía recibir la información enviada a través del formulario y recopilarla en un apartado ‘Tareas’ para enviársela al alumno.

Por último, al tener registrados y asociados en la base de datos tanto los cursos como los alumnos y sus amonestaciones y expulsiones, una buena idea sería realizar un estudio estadístico de las sanciones que se tramitan en el centro, con el objetivo de detectar las clases más problemáticas, los alumnos más conflictivos, etc.

1.3 Prototipo de la aplicación

Antes de iniciar el desarrollo de la aplicación, se ha realizado un prototipo de la interfaz de la misma. En la siguiente imagen se muestra el prototipo de la vista general en el que se ha basado el diseño final:



Sistema de gestión - IES Rafael Alberti				
Alumnos				Buscar alumno
Nombre y apellidos	Curso	Teléfono	Email	
Peter Sagan	1º ESO - A	666147258	psagan@bora.com	
Julian Alaphilippe	1º ESO - A	635789655	lulu@quick.com	
Toon Aerts	1º ESO - B	612112145	toon@trek.com	
Anna v. der Breggen	3º ESO - A	645010144	anna@gmail.com	
A. van Vleuten	4º ESO - C	639639639	annemiek@hotmail.com	
Egan Bernal	1º Bach. A	600258798	egan@ineos.com	
Richard Carapaz	1º ESO - A	611041058	carapaz@ineos.com	
Jan Hindley	1º Bach. B	699850232	hindley@bora.com	
Yves Lampaert	2º Bach. B	611444777	yves@total.com	
Primoz Roglic	3º ESO - C	612345678	rogla@jumbo.nt	

Figura 2: Prototipo de la vista general

Además, la funcionalidad en la que se ha centrado este proyecto trataba de crear amonestaciones y expulsiones a los alumnos y generar los documentos imprimibles de ambos. Se muestra también el prototipo de la gestión de un alumno determinado, donde se podría generar amonestaciones y expulsiones:

Sistema de gestión - IES Rafael Alberti

Alumno

Nombre: Peter Sagan
 Curso: 1º ESO - A
 Teléfono: 666147258
 Email: psagan@bora.com

Amonestar Expulsar

Figura 3: Prototipo de la vista de un alumno

1.4 Aplicaciones similares

Para la realización de esta aplicación, se ha tomado como referencia la herramienta que la Junta de Andalucía tiene a disposición de los centros educativos: [Séneca](#).



Figura 4: Logo de Séneca

En dicha herramienta, además de toda información referente a alumnos y profesores, existe un formulario para añadir expulsiones como se muestra a continuación:

CONDUCTAS CONTRARIAS/GRAVES DE UN ALUMNO/A

Año académico: 2021-2022 Curso: 1º de E.S.O.
 Unidad: 1º ESO A
 Alumno/a:
 Primer/a tutor/a DNI/Pasaporte del primer/a tutor/a
 Segundo/a tutor/a DNI/Pasaporte del segundo/a tutor/a
 Dirección:
 Teléfono: Teléfono urgencias:
 Incidente:
 Fecha:
 Profesional que ha comunicado el incidente:
 Tramo horario:
 Incidente:
 Descripción detallada:

Figura 5: Formulario de expulsiones en Séneca

Tomando esta aplicación web como punto de partida, se ha intentado mantener los aspectos positivos y añadir otros que no se contemplan, como la incorporación de forma automática de las amonestaciones que han provocado la expulsión o la inclusión de las tareas que el alumno debe realizar durante su período de sanción.

1.5 Tecnologías utilizadas

Para llevarlo a cabo se ha decidido utilizar el software de gestión ERP de [Odoo](#), ya que reúne todas las condiciones para satisfacer las principales necesidades del proyecto.

Con respecto a la programación del bot, existen librerías de varios lenguajes de programación diferentes para desarrollar bots, como puede verse en [este listado](#). Se ha decidido realizar este desarrollo en *Python* con objeto de unificar el lenguaje utilizado tanto en Odoo como en Telegram, y mantener así el mismo esquema de desarrollo en ambos.



Figura 6: El bot de Telegram se ha desarrollado en Python

Una vez determinado el lenguaje de programación, de las tres librerías principales existentes para *Python*, se ha decidido utilizar [Python-telegram-bot](#), dado que es la más actualizada y la que mayor comunidad y soporte tiene en la actualidad.

El sistema está alojado en un servidor virtual privado (VPS) provisto por [Digital Ocean](#), con lo que nos aseguramos que el ERP esté en continuo funcionamiento. Asimismo, el bot también se aloja y ejecuta en el mismo servidor, y en este caso se ha utilizado el programa *Screen* para poder ejecutar en segundo plano y de forma ininterrumpida el script, incluso una vez cerrada la sesión remota con el servidor.

2 Descripción del resultado

El resultado final es una aplicación de gestión en la que podemos gestionar las matriculaciones de los alumnos y los profesores asignados a cada curso, así como tramitar amonestaciones y expulsiones de los alumnos del centro. Es en estos últimos

trámites en los que se ha centrado la atención, ya que, como se ha comentado en [1], es la principal carencia encontrada en el centro que se utiliza como referencia. En definitiva, el resultado final es un ERP con las siguientes funcionalidades:

- Creación de amonestaciones y expulsiones a los alumnos de forma manual.
- Descarga en formato PDF de ambas sanciones con los datos cumplimentados de forma automática.
- Aviso por mensaje de Telegram al jefe de estudios cuando un alumno alcance o supere tres amonestaciones. Dicho aviso contiene dos botones: **‘Expulsar alumno’** e **‘Indultar alumno’**, y si se pulsa el primero de ellos, se genera una expulsión en el ERP.
- Al crear una expulsión en borrador, se avisa por Telegram a todos los profesores que imparten aquellas asignaturas en las que está matriculado el alumno propuesto para expulsión.
- Aviso automatizado, también por Telegram, a la familia del alumno expulsado, una vez validada dicha sanción.
- Posibilidad de añadir las tareas asociadas a una expulsión de un alumno.
- Registro de todas las acciones llevadas a cabo en el ERP, incluidas las realizadas desde Telegram.

3 Instalación

El ERP desarrollado en este proyecto es una aplicación que no necesita ser instalada para utilizarse. Para ello, basta con acceder en el navegador a 159.223.221.8:8069 para entrar a la aplicación y, una vez identificados con usuario y contraseña, se podrá utilizar con normalidad.

Importante

Si se opta por desarrollar un ERP propio, tal como se indica en [7], el host deberá cambiarse por la IP del servidor en el que esté instalado el sistema.

3.1 Manual de instalación

Para incluir el módulo ‘Jefatura’ desarrollado en este proyecto en otro ERP desarrollado en Odoo, se deberá añadir a la carpeta que esté configurada para alojar los módulos personalizados (por defecto es el directorio `/opt/odoo15/odoo/addons`). Dicho módulo puede descargarse [aquí](#).



Importante

El funcionamiento de este módulo sólo está probado en la versión 15 de Odoo, para versiones anteriores puede haber errores de dependencias.

Una vez incluido el directorio, debemos acceder en modo desarrollador al menú **Aplicaciones**, actualizar la lista de aplicaciones disponibles, y en ese momento, en el listado aparecerá ‘Jefatura’, tal como se muestra en la siguiente imagen:

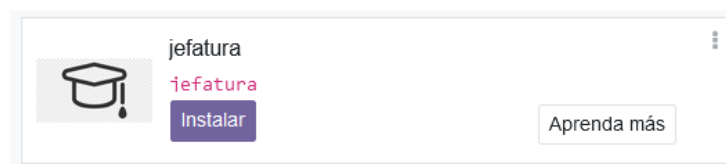


Figura 7: Instalación del módulo Jefatura

Al finalizar la instalación, ‘Jefatura’ aparecerá como ‘Instalada’ y en el menú principal se podrá acceder a dicho módulo:

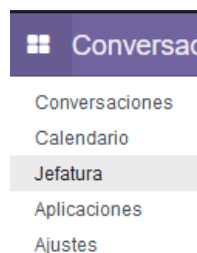


Figura 8: Menú tras instalar el módulo Jefatura

Tras acceder al módulo recién instalado, ya tendremos acceso a los todos los menús.

4 Diseño funcional de la aplicación

Para llevar a cabo este proyecto, se ha realizado un diseño de la base de datos en la que se basa el ERP. Dicha base de datos ha sufrido varias modificaciones a lo largo de la labor de desarrollo, siendo este el diagrama definitivo:

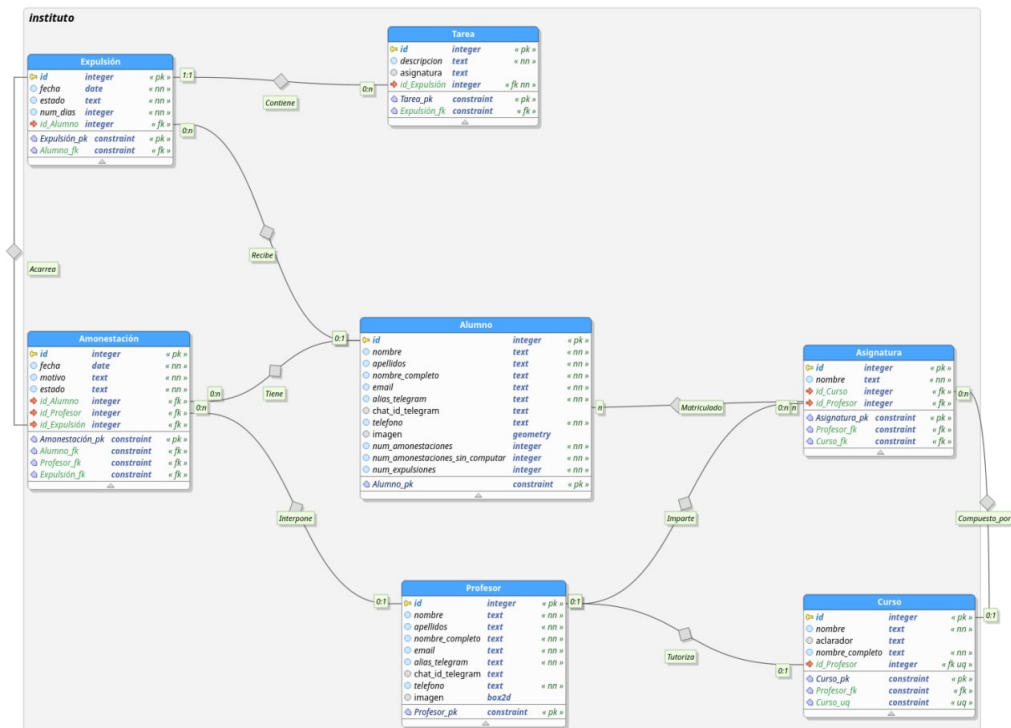


Figura 9: Diagrama de la base de datos

Por otra parte, se ha realizado una enumeración de casos de uso que llevarían a cabo los trabajadores de un instituto, con el objetivo de servir de guía en cuanto a las funcionalidades más importantes a desarrollar. Estos casos de uso son los siguientes:

- Dar de alta alumnos, profesores, cursos y asignaturas.
- Generar amonestaciones y expulsiones desde el ERP.
- Generar expulsiones desde Telegram.
- Darse de alta en el bot que sirve como canal de notificaciones.
- Introducir las tareas a realizar por un alumno en su período de expulsión.

El caso de uso que ha centrado la mayor parte de la atención ha sido el de generar expulsiones desde Telegram, ya que conlleva el envío de información desde Odoo al bot y viceversa, incluyendo la inserción de datos (expulsión) y la modificación de campos en registros ya existentes (las amonestaciones asociadas pasan a estar computadas). A continuación se muestra el diagrama de las operaciones llevadas a cabo en este proceso:

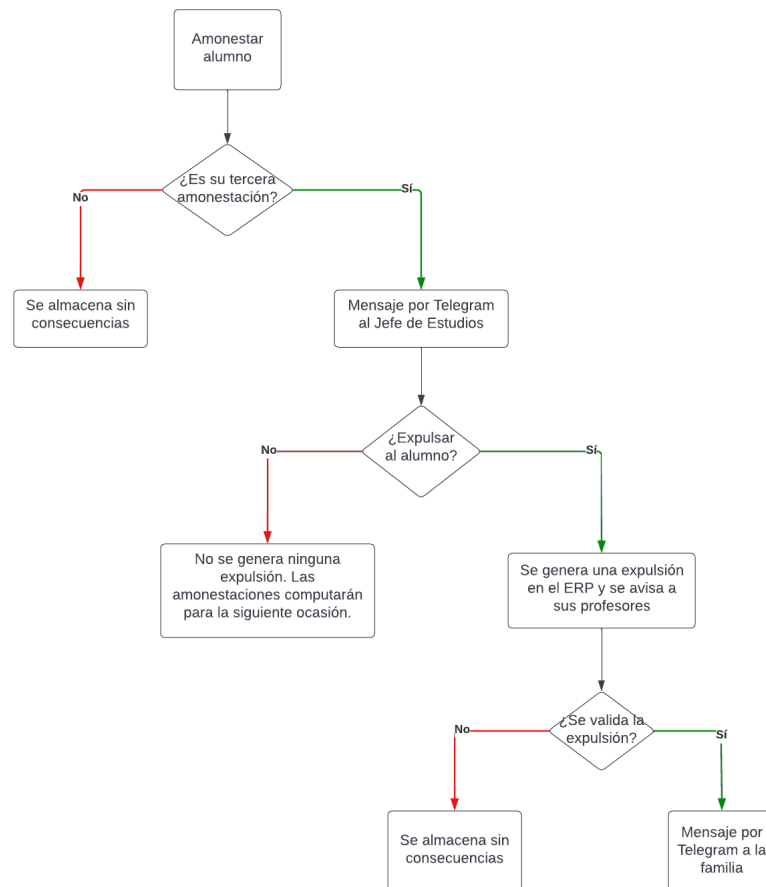


Figura 10: Diagrama de caso de uso: creación de expulsión desde Telegram.

5 Desarrollo

Como se ha visto en la sección [1], se ha decidido realizar este proyecto utilizando la herramienta ERP de Odoo en su versión 15, debido a que las necesidades de la aplicación requerían una integración con una base de datos completa y centralizada para la gestión del instituto.

5.1 Ventajas e inconvenientes del software escogido

La utilización de Odoo ofrece determinadas ventajas, como la posibilidad de crear varios usuarios y roles diferentes para el acceso a la aplicación, o bien la posibilidad de crear **acciones automatizadas** en función de unos eventos determinados (triggers). Estas acciones se definen en *Técnico* → *Automatización* → *Acciones automatizadas*, y se han utilizado para cumplimentar campos de manera automática, enviar mensajes de Telegram cuando se generen determinadas acciones, o para al-

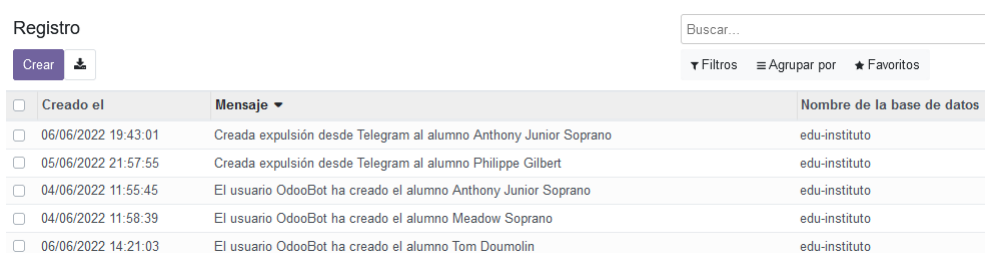
macenar registros de toda actividad realizada desde el ERP, como veremos más adelante.

Entre los inconvenientes de Odoo respecto a otro tipo de software diferente, además de la configuración tan específica y problemática del sistema y sus dependencias, destacan las dificultades técnicas derivadas de las particularidades de Odoo, debido en gran parte al modo en que internamente se modeliza la base de datos y, en consecuencia, a la forma de acceder a la información de sus tablas.

5.2 Dificultades afrontadas en el proyecto

A continuación, se enumeran las principales dificultades encontradas en el desarrollo de la aplicación:

- Una de las funcionalidades esenciales de la aplicación es la de poder generar expulsiones desde Telegram y, en lugar de utilizando la API de Odoo, se ha decidido realizar por medio de inserciones directas a la base de datos desde el correspondiente script de Python, haciendo uso de la librería **psycopg2**. Esto, además del tratamiento directo a la base de datos, y los problemas que eso conlleva, implica que no se ejecutan las funciones asociadas a las acciones automatizadas en Odoo (triggers), con lo cual el registro de logs, que se realiza automáticamente en cada inserción realizada manualmente desde Odoo, no se ejecuta al hacer la inserción desde Telegram. Esto se ha tenido en cuenta de forma que al insertar una expulsión a la base de datos, se realiza también una inserción a la tabla **ir.logging**, de modo que así forzamos a que quede un registro siempre, independientemente de dónde provengan los datos insertados.



<input type="checkbox"/>	Creado el	Mensaje	Nombre de la base de datos
<input type="checkbox"/>	06/06/2022 19:43:01	Creada expulsión desde Telegram al alumno Anthony Junior Soprano	edu-instituto
<input type="checkbox"/>	05/06/2022 21:57:55	Creada expulsión desde Telegram al alumno Philippe Gilbert	edu-instituto
<input type="checkbox"/>	04/06/2022 11:55:45	El usuario OdooBot ha creado el alumno Anthony Junior Soprano	edu-instituto
<input type="checkbox"/>	04/06/2022 11:58:39	El usuario OdooBot ha creado el alumno Meadow Soprano	edu-instituto
<input type="checkbox"/>	06/06/2022 14:21:03	El usuario OdooBot ha creado el alumno Tom Doumolin	edu-instituto

Figura 11: Logs de acciones registradas en Odoo

- El uso de la [API de Telegram](#), a pesar de estar muy bien documentada, ha sido uno de los principales obstáculos a la hora de desarrollar este proyecto. Por ejemplo, el mensaje de aviso por Telegram al Jefe de Estudios sobre una posible expulsión debe almacenar toda la información necesaria para que el bot (en su

script externo y ajeno a Odoo) tenga la información completa y pueda realizar la inserción de la expulsión. Esto, tras valorar varias opciones, se ha decidido realizar mediante un *string* almacenado en el atributo **callback_query** del mensaje, aprovechando sus características, que podemos encontrar en su [documentación](#). Dicho atributo, a pesar de ser un *string*, ha sido construido convenientemente con estructura de diccionario para ser transformado con posterioridad de forma directa y poder trabajar de forma sencilla con él.

- Por último, es importante destacar que Telegram sólo es capaz de enviar mensajes por medio de la ID del chat de un usuario, no por el alias, por lo que la base de datos de Odoo debe almacenar de alguna manera dicha ID. Este es un proceso delicado porque un usuario no conoce su ID de la aplicación, y, aunque es cierto que existen herramientas que la informan (por ejemplo, bots como [@RawDataBot](#)), esto implicaría un proceso manual de obtención de información que posteriormente debe ser introducida individualmente por cada profesor y cada familia de alumnos. Por lo tanto, para evitar este proceso manual, con lo que ello conlleva (errores humanos, olvidos, desconocimiento, etc.), se ha optado por definir el uso de la aplicación de la siguiente manera:

1. Se introduce en el ERP el alias de Telegram de los profesores y alumnos. Este es un campo obligatorio en la matriculación, al igual que el número de teléfono o el correo electrónico.
2. El usuario, una vez que su alias está incluido en la base de datos, podrá iniciar el bot con el comando **/start**. Si en el ERP existe un profesor y/o alumno con el alias de Telegram del usuario que ha escrito el comando, se almacena su ID de chat en el campo **chat_id_telegram**. Si no existe, el bot responderá con un mensaje de error.
3. En cualquiera de los dos casos, tanto si se trata de un alumno como si es un profesor, una vez almacenado dicha ID, ésta se utilizará para conformar la URL adecuada para lanzar la petición por medio de la API de Telegram.

6 Pruebas

Este proyecto, a lo largo de su desarrollo ha experimentado una serie de pruebas para validar tanto la utilidad de la aplicación como su intuitividad a la hora de realizar las principales funciones. Las pruebas llevadas a cabo se detallan a continuación.

6.1 Pruebas de volumen

Se ha probado la inserción de un número elevado de registros en la base de datos, concretamente en la tabla **ir.logging**, que es, además, una tabla con bastantes campos. Tratándose de la gestión de un instituto no se espera que haya más de 100 usuarios trabajando e insertando datos simultáneamente, por lo que consideramos una cantidad suficiente para la prueba de 10.000 inserciones. Dicha prueba se ha repetido en varias ocasiones sin detectar ninguna incidencia en las inserciones. El código del test puede encontrarse [aquí](#).

<input type="checkbox"/>	Creado el	Mensaje ▼	Nombre de la base de datos	Nivel	Función	Tipo
<input type="checkbox"/>	08/06/2022 20:50:23	Añadido log en prueba de volumen	edu-instituto	info	Log de prueba	Servidor
<input type="checkbox"/>	08/06/2022 20:50:26	Añadido log en prueba de volumen	edu-instituto	info	Log de prueba	Servidor
<input type="checkbox"/>	08/06/2022 20:50:22	Añadido log en prueba de volumen	edu-instituto	info	Log de prueba	Servidor
<input type="checkbox"/>	08/06/2022 20:50:22	Añadido log en prueba de volumen	edu-instituto	info	Log de prueba	Servidor
<input type="checkbox"/>	08/06/2022 20:50:25	Añadido log en prueba de volumen	edu-instituto	info	Log de prueba	Servidor
<input type="checkbox"/>	08/06/2022 20:50:24	Añadido log en prueba de volumen	edu-instituto	info	Log de prueba	Servidor
<input type="checkbox"/>	08/06/2022 20:50:25	Añadido log en prueba de volumen	edu-instituto	info	Log de prueba	Servidor

Figura 12: Logs creados en prueba de volumen

6.2 Pruebas de estrés

Del mismo modo que en las pruebas de volumen vistas anteriormente, con [este código](#) se ha probado a realizar la obtención de 10.000 datos de la tabla **jefatura_alumno**, también sin incidencias. Se considera la prueba superada y damos el programa por apto para ser utilizado en un centro de enseñanza.

6.3 Pruebas de usabilidad

Por último, quizás el test más importante al que se ha sometido la aplicación es la prueba de usabilidad con una persona que se ha dedicado en años anteriores a la gestión de amonestaciones y expulsiones en un centro de Enseñanza Secundaria. Gracias a estas pruebas se ha podido corregir detalles a medida que se ha ido desarrollando el programa, y de igual modo, una vez completado el proyecto, se ha vuelto a realizar una prueba completa de los principales casos de uso (definidas en [4]), siendo su opinión positiva con respecto a la funcionalidad y sencillez de uso.

7 Distribución

Como se ha visto en [3.1], es posible instalar el módulo ‘Jefatura’ en otro ERP desarrollado en Odoo. Si además se desea continuar el desarrollo de este mismo

ERP al completo, se debe descargar una copia de la base de datos íntegra en [este enlace](#), y posteriormente importarla.

Por otra parte, en cuanto al desarrollo del bot de Telegram, el código se encuentra disponible en [este directorio de GitHub](#). Únicamente será necesario modificar el fichero sustituyendo el TOKEN y el ID del chat por los datos correspondientes al bot y al chat que se hayan creado para ello. También será necesario realizar la conexión a la base de datos de Odoo, indicando la IP, usuario y contraseña para realizar la conexión a través del paquete **psycopg2**. Esto se ha realizado por medio de un fichero llamado **acceso_postgres**, que se ha ocultado en el repositorio de GitHub por razones de seguridad.

8 Manuales de uso de la aplicación

En la sección [6] se destaca fundamentalmente la intuitividad de la aplicación a la hora de realizar las operaciones básicas. A continuación se muestra cómo se crean registros en el ERP, aunque un manual más detallado de todos los procedimientos que se pueden llevar a cabo tanto en la aplicación como en Telegram puede encontrarse en la [wiki del proyecto](#).

Por ejemplo, para dar de alta una asignatura en el sistema, debemos dirigirnos al menú ‘Asignaturas’ y pulsar en el botón ‘Crear’. Una vez hecho esto, cumplimentamos todos los campos del formulario y pulsamos en ‘Guardar’, tal como vemos en la siguiente imagen:

Figura 13: Creación de una asignatura en el ERP

8.1 Activación y uso del bot de Telegram

Para comenzar a usar el bot, se debe iniciar la aplicación de Telegram y abrir un chat con **@JefaturaIES_bot**. Este será el canal de comunicaciones establecido para profesores y familiares de alumnos.

Al abrir el chat, nos aparecerá el siguiente mensaje, y deberá pulsarse el botón ‘Iniciar’ para arrancar la ejecución del bot.

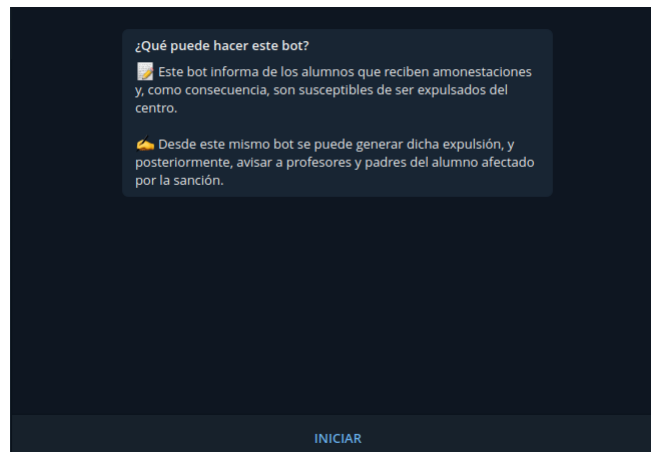


Figura 14: Inicio del bot

Si el usuario desde el que se inicia el bot no figurase ni como profesor ni como familiar de ningún alumno, la respuesta obtenida nos indicará un mensaje de error, y dicha cuenta no se utilizará en ningún momento para las comunicaciones. De esta forma, se protege información sensible como nombres, apellidos, y datos sobre comportamiento de los alumnos a terceras personas.

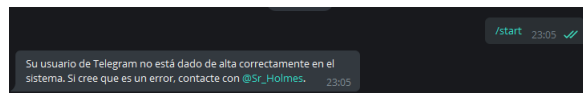


Figura 15: Mensaje de Telegram si no está dado de alta correctamente

Una vez corregido el error, o bien si desde el primer momento el usuario sí está dado de alta correctamente, se obtendrá un mensaje de bienvenida. Este es un ejemplo para el caso del profesor Silvio Dante:

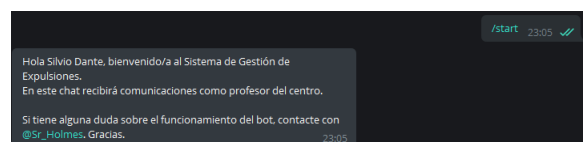


Figura 16: Mensaje de bienvenida del bot a un profesor

A partir de este instante, tanto profesores como familiares habrán activado el servicio de notificaciones y se les comunicará las situaciones correspondientes en cada caso a través del mismo canal.

9 Conclusiones

Con respecto a la idea original del proyecto, a grandes rasgos la solución aportada cumple con la mayoría de los requisitos, aunque ha quedado pendiente la elaboración de informes estadísticos sobre amonestaciones y expulsiones en los diferentes cursos.

En la parte contraria, el planteamiento inicial no contemplaba el hecho de almacenar registros (logs) de cada acción realizada sobre la base de datos del ERP. Esto se ha decidido hacer aprovechando la estructura de Odoo, que ya tiene de forma predefinida una tabla específica para estos registros.

9.1 Posibles mejoras

A pesar de contar con una aplicación funcional, a la finalización de este proyecto hay detalles mejorables, por lo que se proponen las siguientes futuras mejoras:

- Creación de usuarios diferentes para cada profesor, diferenciados del Jefe de Estudios, que tengan roles diferenciados y permisos de acceso a los menús en función de su categoría. Por ejemplo, no se debe permitir que un usuario con rol de profesor pueda crear ni validar una expulsión.
- Creación de otro módulo personalizado para Secretaría, con objeto de para gestionar, entre otras cosas, los retrasos en la entrada por parte de alumnos. La acumulación de estos retrasos es motivo de amonestación, por lo que el desarrollo sería similar al módulo Jefatura. Asimismo, se podría hacer otro bot para informar a la persona encargada de Secretaría, o bien modificar el existente para gestionar las comunicaciones en función de qué persona deba recibir un aviso u otro.
- Actualmente, una expulsión creada desde Telegram se genera para un período de 3 días. Siempre existe la posibilidad de modificarlo manualmente en el ERP antes de validarlo, pero se propone como mejora el poder establecer esta cantidad desde Telegram.
- Una idea para automatizar la selección del número de días en una expulsión sería la de añadir un campo ‘Gravedad’ a cada amonestación, de forma que al crear una expulsión, en función de la importancia de las amonestaciones asociadas, el ERP proponga de forma automática una duración de la expulsión.
- El uso de la API de Odoo. En el desarrollo actual del bot, éste utiliza una conexión a PostgreSQL realizando inserciones y consultas directamente a la

base de datos y, aunque la aplicación es sencilla y no ha habido ningún inconveniente, este método es menos seguro y más propenso a errores. Igualmente, a medida que la base de datos vaya creciendo, las actuaciones directas a la misma serán más difíciles de controlar, necesitando en muchos casos el uso de procedimientos para los que la API supone una mejoría tanto en eficiencia como en claridad de código.

- Creación de una base de datos para pruebas. En el transcurso de este proyecto, cada variación realizada en el ERP, por mínima que haya sido, ha implicado la actualización del módulo (si se modifican las vistas), el reinicio del servicio (si se modifican los modelos) o ambas acciones en conjunto. Al no ser este un proyecto con uso en tiempo real, esto no ha supuesto ningún inconveniente, pero si se tratase de una aplicación de uso real y continuo, esto no puede llevarse a cabo de esta manera, porque conllevaría errores en el funcionamiento si hubiera algún usuario utilizando la aplicación en esos momentos de reinicio. Para ello, la propuesta es crear una base de datos de prueba en la que realizar el desarrollo y, una vez que se ha probado y se considera definitivo, realizar un reinicio programado y notificado con antelación a los usuarios.

10 Índice de imágenes

1	Integración de Telegram en Odoo	1
2	Prototipo de la vista general	2
3	Prototipo de la vista de un alumno	3
4	Logo de Séneca	3
5	Formulario de expulsiones en Séneca	3
6	El bot de Telegram se ha desarrollado en Python	4
7	Instalación del módulo Jefatura	6
8	Menú tras instalar el módulo Jefatura	6
9	Diagrama de la base de datos	7
10	Diagrama de caso de uso: creación de expulsión desde Telegram.	8
11	Logs de acciones registradas en Odoo	9
12	Logs creados en prueba de volumen	11
13	Creación de una asignatura en el ERP	12
14	Inicio del bot	13
15	Mensaje de Telegram si no está dado de alta correctamente	13
16	Mensaje de bienvenida del bot a un profesor	13

11 Bibliografía

1. Documentación oficial de [Odoo](#), versión 15.
2. Grupo de Telegram [Odoo Developers](#).
3. Documentación oficial de [Python-telegram-bot](#), versión 20.0a1.
4. Documentación oficial de [Telegram Bot API](#), versión actualizada a fecha 16/04/2022.
5. Artículo [Primeros pasos creando un bot de Telegram con Python](#), 04/04/2021.
6. Sitio web de [Moldeo Interactive](#).
7. Artículo [Cómo conectarse a PostgreSQL usando Python](#), Edu Lázaro, 27/05/2019.