

EDUFAST

Versión:

V 1.00

|  |
| --- |
| Queda prohibido cualquier tipo de explotación y, en particular, la reproducción, distribución, comunicación pública y/o transformación, total o parcial, por cualquier medio, de este documento sin el previo consentimiento expreso y por escrito de la Junta de Andalucía. |

HOJA DE CONTROL

|  |  |  |  |
| --- | --- | --- | --- |
| **Organismo** | SENA | | |
| **Proyecto** | EDUFAST | | |
| **Entregable** |  | | |
| **Autor** | Dylan Santiago Herrera Espinosa | | |
| **Aprobado por** |  | **Fecha Aprobación** | DD/MM/AAAA |
|  |  | **Nº Total de Páginas** |  |

REGISTRO DE CAMBIOS

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Causa del Cambio** | **Responsable del Cambio** | **Fecha del Cambio** |
| 0100 | Versión inicial | Dylan Santiago Herrera Espinosa | 28/05/2025 |
|  |  |  |  |
|  |  |  |  |

CONTROL DE DISTRIBUCIÓN

|  |
| --- |
| **Nombre y Apellidos** |
| <Nombre Apellido1 Apellido2> |
|  |
|  |
|  |
|  |

ÍNDICE

[1 INTRODUCCIÓN 5](#__RefHeading__1984_1977503599)

[1.1 Objeto 5](#__RefHeading__1986_1977503599)

[1.2 Alcance 5](#__RefHeading__1988_1977503599)

[2 DESCRIPCIÓN DEL SISTEMA 6](#__RefHeading__1990_1977503599)

[2.1 Antecedentes y descripción funcional del sistema 6](#__RefHeading__1992_1977503599)

[2.2 Componentes fundamentales 6](#__RefHeading__1996_1977503599)

[2.3 Relación con otros sistemas 6](#__RefHeading__1998_1977503599)

[3 RECURSOS HARDWARE 7](#__RefHeading__2000_1977503599)

[3.1 Servidores 7](#__RefHeading__2002_1977503599)

[3.2 Estaciones cliente 7](#__RefHeading__2004_1977503599)

[3.3 Conectividad 8](#__RefHeading__2006_1977503599)

[3.4 Restricciones 8](#__RefHeading__2008_1977503599)

[4 RECURSOS SOFTWARE 9](#__RefHeading__2010_1977503599)

[4.1 Matriz de certificación 9](#__RefHeading__2012_1977503599)

[4.2 Restricciones técnicas del sistema 9](#__RefHeading__3655_1276270192)

[4.3 Requisitos de otros sistemas 9](#__RefHeading__2014_1977503599)

[5 INSTALACIÓN Y CONFIGURACIÓN DEL SOFTWARE BASE 11](#__RefHeading__3657_1276270192)

[6 CONFIGURACIÓN DEL SISTEMA 12](#__RefHeading__2022_1977503599)

[6.1 Configuración del sistema 12](#__RefHeading__20382_1181642420)

[6.2 Configuración de otros sistemas 13](#__RefHeading__20384_1181642420)

[7 COMPILACIÓN DEL SISTEMA 14](#__RefHeading__3659_1276270192)

[8 INSTALACIÓN DEL SISTEMA 15](#__RefHeading__2040_1977503599)

[8.1 Requisitos previos 15](#__RefHeading__9176_1255198503)

[8.2 Procedimiento de instalación 15](#__RefHeading__9178_1255198503)

[9 VERIFICACIÓN DEL PROCESO DE INSTALACIÓN 17](#__RefHeading__37000_790639373)

[10 MARCHA ATRÁS DE LA INSTALACIÓN Y CONFIGURACIÓN 18](#__RefHeading__2044_1977503599)

[10.1 Requisitos previos 18](#__RefHeading__2046_1977503599)

[10.2 Marcha atrás del sistema 18](#__RefHeading__2048_1977503599)

[10.3 Marcha atrás del software base 19](#__RefHeading__2050_1977503599)

[11 ANEXOS 20](#__RefHeading__2052_1977503599)

[11.1 Resumen de tareas de configuración 20](#__RefHeading__9180_1255198503)

[12 GLOSARIO 21](#__RefHeading__2054_1977503599)

[13 BIBLIOGRAFÍA Y REFERENCIAS 22](#__RefHeading__2056_1977503599)

# INTRODUCCIÓN

## Objeto

El propósito de este manual es proporcionar una guía clara, detallada y accesible para la instalación del software **Edufast** una aplicación desarrollada con fines académicos y educativos. Este documento está dirigido a estudiantes, docentes, desarrolladores y administradores de sistemas que requieran instalar y configurar el entorno necesario para el funcionamiento del sistema.

El manual describe paso a paso el proceso de instalación, incluyendo los requisitos del sistema, la configuración de dependencias, Su objetivo es garantizar una implementación exitosa, minimizar errores durante la instalación y facilitar el uso autónomo del sistema por parte de los usuarios en entornos educativos.

## Alcance

### Unidades Organizativas y Responsabilidades

Este documento está dirigido a las siguientes unidades y perfiles organizativos:

 **Responsable Funcional del Proyecto**  
Unidad o persona encargada de definir los objetivos funcionales del software, así como de coordinar su uso en el ámbito académico. Supervisa que el documento responda a las necesidades de los usuarios finales y valida su contenido funcional.

 **Responsable Técnico del Proyecto**  
Encargado de supervisar los aspectos técnicos del software, incluyendo su instalación, configuración y despliegue. Participa activamente en la redacción del documento técnico, asegurando la corrección de los procedimientos descritos.

 **Equipo de Desarrollo**  
Unidad responsable de la implementación del software. Participa en la elaboración inicial del manual de instalación y en la documentación técnica asociada, describiendo los requisitos, pasos y configuraciones necesarias.

 **Unidad de Soporte Técnico o TIC**  
Encargada de asistir a los usuarios finales en la instalación del sistema. Utiliza este documento como referencia para resolver incidencias y realizar instalaciones en entornos controlados (aulas, laboratorios, servidores institucionales, etc.).

 **Unidad de Calidad o Aseguramiento de la Información** (si aplica)  
Revisa la documentación desde el punto de vista de la calidad, la claridad, la coherencia y la conformidad con los estándares del modelo MADEJA. Aprueba la versión final del documento antes de su publicación oficial.

 **Usuarios Finales**  
Principal público objetivo del manual. Incluye a docentes, estudiantes y personal administrativo como profesores, secretaria y coordinación que hará uso del sistema y que necesita una guía clara para su instalación.

 **Unidad de Coordinación del Proyecto / Secretaría Técnica**  
Se encarga del registro, archivo y publicación del documento en el repositorio documental correspondiente, garantizando la trazabilidad y control de versiones de acuerdo con la política de gestión documental de la organización.

# DESCRIPCIÓN DEL SISTEMA

## Antecedentes y descripción funcional del sistema

Previo al desarrollo del sistema, la gestión de información relacionada con jornadas académicas, actividades formativas, materias, cursos, grados y publicaciones se realizaba de forma manual o con herramientas aisladas, lo que dificultaba la organización, el acceso a los datos y la trazabilidad de la información. Esta situación afectaba directamente a los administradores del sistema, quienes enfrentaban dificultades para centralizar, consultar y actualizar los datos de manera eficiente.

El nuevo sistema de información fue concebido para dar respuesta a estas necesidades, proporcionando una solución tecnológica que permite gestionar de forma centralizada:

* **Jornadas académicas**: programación, seguimiento y registro de eventos institucionales con hora, fecha y tipo de jornada.
* **Actividades**: **Materias**: administración de la oferta académica, contenidos como id de materia y su nombre.
* **Cursos**: organización y gestión de cursos impartidos, incluyendo inscripción y seguimiento de los estudiantes.
* **Grados**: administración de los programas académicos, su estructura y requisitos.
* **Publicaciones**: registro y gestión de documentos y trabajos académicos relacionados con la institución.

Gracias a esta herramienta, los administradores del sistema pueden realizar tareas de gestión con mayor agilidad, fiabilidad y seguridad, reduciendo errores y mejorando el control sobre los procesos académicos. Además, el sistema estandariza los flujos de trabajo y mejora el acceso a la información, lo que repercute en una administración más eficaz y alineada con los objetivos institucionales como un apoyo.

## Componentes fundamentales

### Principales Módulos del Sistema

El sistema está estructurado en varios módulos principales que permiten gestionar de forma integral la información académica y administrativa. A continuación, se describen brevemente cada uno de ellos:

|  |  |
| --- | --- |
| **Módulo** | **Descripción** |
| **Módulo de Jornadas Académicas** | Permite la creación, programación y seguimiento de jornadas y eventos académicos, facilitando la gestión de fechas, horas y tipo de jornadas. |
| **Módulo de Actividades** | planificación, documentación y evaluación de actividades formativas internas o externas con acceso a información como id, materias y el profesor asignado. |
| **Módulo de Materias** | Administra la oferta académica, incluyendo la definición de contenidos, con código, nombre, asignación a docentes y seguimiento del progreso. |
| **Módulo de Cursos** | Facilita la organización de cursos impartidos, la inscripción de estudiantes y el seguimiento de su desarrollo con código y nombre. |
| **Módulo de Grados** | Gestión de programas académicos, estructura curricular, requisitos y seguimiento de estudiantes con contenido de código, nombre, grado y nivel educativo. |
| **Modulo de Publicaciones** | Registro y administración de documentos, trabajos académicos y publicaciones institucionales con título fecha y imágenes. |
| **Administración del Sistema** | Gestión de usuarios, roles, permisos y configuración general del sistema con acceso a registrar, actualizar y eliminar la información de los módulos,. |

## Relación con otros sistemas

## 6. Sistemas Relacionados y Naturaleza de las Relaciones

EduFast es un sistema enfocado en la **gestión académica institucional**. Su diseño contempla funcionalidades internas como:

* Gestión de **grados, cursos y jornadas por parte del docente**.
* Administración de **materias** y **logros académicos por parte el docente**.
* Registro de **actividades** y **publicaciones institucionales**. Por el rol de administrador y coordinador
* Seguimiento de **asistencias** y **notas** por parte de los docentes.

Aunque no depende directamente de sistemas externos complejos, existen ciertas relaciones que pueden establecerse para complementar o facilitar el trabajo institucional.

### Sistema: Plataforma de Gestión Documental Institucional

**Descripción:** Sistema interno o compartido donde se almacenan documentos como boletines, informes de logros y reportes de notas.

|  |  |
| --- | --- |
| **Elemento** | **Detalle** |
| **Relación** | Exportación de archivos PDF o Excel generados desde EduFast. |
| **Funcionalidad usada** | Generación de reportes de notas, listados de asistencia, logros. |
| **Requisito técnico** | Compatibilidad con el formato (PDF/Excel). |
| **Requisito administrativo** | Definición de ruta o carpeta compartida dentro del servidor. |

### Sistema: Servidor Institucional o Hosting Web

**Descripción:** Espacio donde se aloja EduFast para ser accedido por los usuarios de la institución.

|  |  |
| --- | --- |
| **Elemento** | **Detalle** |
| **Relación** | EduFast debe estar desplegado en un entorno web (servidor LAMP/WAMP). |
| **Funcionalidad usada** | Acceso a través de navegador por roles: administrador, docente, etc. |
| **Requisito técnico** | Soporte para PHP y MySQL, |
| **Requisito administrativo** | Asignación de dominio o subdominio institucional. |

### Sistema: Plataforma de Comunicación Interna (opcional)

**Descripción:** Puede ser un grupo de WhatsApp, correo institucional o sistema de mensajería con el que se comuniquen eventos o publicaciones registradas en EduFast.

|  |  |
| --- | --- |
| **Elemento** | **Detalle** |
| **Relación** | Publicaciones registradas pueden ser notificadas manualmente o integradas. |
| **Funcionalidad usada** | Copia de mensajes o exportación de publicaciones. |
| **Requisito técnico** | (Opcional) Desarrollo de una API para envío automático. |
| **Requisito administrativo** | Autorización institucional para enviar comunicaciones oficiales. |

⚠️ **Nota importante:** EduFast **no se conecta actualmente a sistemas de inventario, firma digital, cifrado ni calendarios externos.** Cualquier integración adicional deberá evaluarse según necesidades futuras.

# RECURSOS HARDWARE

## Requisitos de los Servidores para la Implantación

La correcta implantación del sistema **EduFast** requiere la habilitación de distintos servidores, ya sea físicos o virtuales, que permitan el alojamiento, operación y acceso eficiente al sistema por parte de los diferentes roles administrador, rector, secretaria, docentes y estudiantes.

A continuación, se presentan los requisitos de cada uno de los servidores necesarios, diferenciando entre **requisitos mínimos y recomendados**.

### Servidor de Aplicaciones

Este servidor aloja el sistema EduFast (frontend y backend), siendo responsable de servir la aplicación al navegador de los usuarios.

|  |  |
| --- | --- |
| **Recurso Azure** | **Recomendación** |
| **Servicio** | [Azure App Service (Linux o Windows)](https://azure.microsoft.com/en-us/services/app-service/) |
| **Plan** | B1 (básico) para pruebas  P1v3 (Premium) para producción |
| **Lenguaje soportado** | PHP 8.x |
| **Almacenamiento** | 10 GB incluidos |
| **Escalabilidad** | Automática por demanda |
| **Sistema operativo** | Linux (recomendado) o Windows |
| **Disponibilidad** | Alta disponibilidad (99.95% SLA) |

### Base de Datos

Responsable de gestionar y almacenar toda la información estructurada del sistema: usuarios, cursos, asistencias, notas, etc.

|  |  |
| --- | --- |
| **Recurso Azure** | **Recomendación** |
| **Servicio** | [Azure Database for MySQL - Flexible Server](https://learn.microsoft.com/en-us/azure/mysql/flexible-server/) |
| **Versión** | MySQL 8.x |
| **Plan mínimo** | Burstable B1MS (para desarrollo/pruebas) |
| **Plan recomendado** | General Purpose D2ds\_v4 (para producción) |
| **Almacenamiento** | 20 GB (ampliable hasta 1 TB) |
| **Seguridad** | Autenticación SSL, firewall por IP, conexión privada (VNet) |
| **Backups** | Automáticos, configurables hasta 35 días |

### Justificación: Uso de Servidores Virtuales

Se recomienda encarecidamente el uso de **servidores virtuales (VMs)** para facilitar la escalabilidad, las copias de seguridad, la administración remota y el mantenimiento de cada componente por separado. En caso de que se opte por una instalación en servidores físicos, debe justificarse por alguna de estas razones:

* Existencia previa de servidores dedicados en la institución.
* Limitaciones de infraestructura de virtualización.
* Presupuesto restringido.
* Baja carga esperada y centralización deseada.

Para ello, se deberá haber completado previamente la **plantilla de solicitud de creación de máquina virtual** correspondiente y entregarla al área técnica.

|  |  |  |
| --- | --- | --- |
| **Servidor 1** | | |
| **Dato** | **Valor mínimo** | **Valor recomendado** |
| Procesador |  |  |
| Memoria RAM |  |  |
| Tamaño Almacenamiento |  |  |
| Otros |  |  |
|  |  |  |

## Estaciones cliente

## Requisitos Hardware de Estaciones Cliente para Edufast

|  |  |  |
| --- | --- | --- |
| **Componente** | **Requisitos Mínimos** | **Requisitos Recomendados** |
| **Procesador (CPU)** | Intel Core i3 (4ª generación o superior) AMD Ryzen 3 | Intel Core i5 (8ª gen o superior) AMD Ryzen 5 o superior |
| **Memoria RAM** | 4 GB | 8 GB o más |
| **Disco Duro / SSD** | 128 GB (HDD o SSD) | 256 GB SSD (para mayor velocidad) |
| **Resolución de Pantalla** | 1366 x 768 | 1920 x 1080 (Full HD) |
| **Sistema Operativo** | Windows 10 / Linux Ubuntu 20.04 / macOS Catalina | Windows 11 / Linux Ubuntu 22.04 / macOS Monterey o superior |
| **Navegador Web** | Google Chrome, Mozilla Firefox, Microsoft Edge (últimas versiones) | Google Chrome actualizado (recomendado por compatibilidad) |
| **Conectividad** | Conexión a Internet de al menos 5 Mbps | Conexión estable de 10 Mbps o más (preferiblemente por cable o Wi-Fi 5/6) |
| **Periféricos** | Teclado, ratón, parlantes o auriculares | Webcam y micrófono si se integran videollamadas |
| **Software adicional** | Adobe Reader (PDF), Extensión PHP viewer (si se generan reportes) | Office (Web o local) para apertura de reportes, hojas de cálculo, etc. |

### ✅ Notas adicionales

* **EduFast es un sistema basado en navegador**, por lo tanto, no requiere instalación en las estaciones cliente.
* Compatible con **sistemas operativos de escritorio comunes** (Windows, Linux, macOS).
* El rendimiento será óptimo si se accede desde navegadores actualizados con extensiones mínimas activas como recomendación para que funcione perfectamente.

## Conectividad

### Estructura de Conectividad del Sistema EduFast

El sistema **EduFast** está compuesto por una arquitectura en la nube que permite la interacción eficiente entre sus componentes mediante una infraestructura de conectividad distribuida. Esta arquitectura está diseñada para ofrecer alta disponibilidad, seguridad en el acceso y un rendimiento fluido en la gestión académica.

A continuación se describe cómo se encuentra estructurada la conectividad:

#### 1. Estaciones Cliente

Las estaciones cliente (usuarios finales: administradores, rectores, profesores, estudiantes y secretarías) se conectan a **EduFast** a través de navegadores web, estableciendo comunicación con el sistema mediante **protocolo HTTPS**. Este acceso se realiza desde redes locales o remotas, conectando primero a través de **Internet pública**.

#### 2. Perímetro de Seguridad (Firewall / Gateway)

Toda solicitud de los clientes pasa por un **firewall o gateway perimetral**, encargado de filtrar, permitir y redirigir las conexiones hacia los servidores correspondientes. Este componente actúa como primera línea de defensa para el acceso a la infraestructura alojada en la nube.

#### 3. Plataforma en la Nube (Azure)

El sistema está desplegado en **Microsoft Azure**, donde se han configurado las máquinas virtuales necesarias:

* **Servidor de Aplicaciones (Apache + PHP)**: Es el servidor que ejecuta la lógica del sistema EduFast. Aquí residen los módulos que gestionan cursos, grados, jornadas, materias, asistencias, actividades, publicaciones y logros.
* **Servidor de Base de Datos (MySQL)**: Almacena toda la información académica y administrativa. La base de datos está dimensionada para manejar un volumen aproximado de **5 GB de datos**, entre registros históricos y actuales.

Ambos servidores se encuentran en la misma **red virtual interna de Azure**, lo que garantiza una comunicación rápida y segura entre ellos. Esta red está configurada para evitar exposición directa a Internet, excepto por el servidor web, el cual está debidamente protegido mediante políticas de acceso y certificados SSL.

#### 4. Canal de Comunicación Interno

Dentro de la infraestructura de Azure, se establece una red de alta velocidad entre los servidores. La comunicación entre el servidor de aplicaciones y el servidor de base de datos local y en la nube se realiza usando **conexiones internas privadas**, sin exponer puertos de la base de datos al exterior, lo que mejora la seguridad y el rendimiento.

#### 5. Protección de Datos

Todas las transmisiones de datos entre el cliente y los servidores están cifradas mediante **protocolo TLS (HTTPS)**. Además, la infraestructura en la nube incorpora servicios de monitorización, cortafuegos, respaldo y recuperación, lo que asegura la integridad y disponibilidad de los datos.

## Restricciones

### ⚠️ Limitaciones del Sistema EduFast

A pesar de estar diseñado para ser un sistema académico funcional, modular y escalable, **EduFast** presenta algunas limitaciones propias de su estructura actual, su funcionalidad y los entornos donde se implementa. Estas restricciones deben tenerse en cuenta tanto en el uso diario como en futuras ampliaciones o integraciones.

#### 1. ****Limitaciones Funcionales****

* El sistema **no gestiona inventarios, pagos ni procesos administrativos financieros**. Su enfoque está centrado exclusivamente en la gestión académica (grados, jornadas, cursos, publicaciones, asistencias, logros, materias y actividades).
* No incluye un sistema interno de mensajería entre usuarios, aunque admite publicaciones visibles por roles.
* Las estadísticas y reportes generados son básicos y no cuentan aún con un módulo avanzado de analítica educativa.

#### 2. ****Limitaciones de Acceso****

* **No cuenta con integración nativa con sistemas externos** como @firma, @ries, LDAP, o sistemas de autenticación de terceros, por lo que actualmente el inicio de sesión se realiza de forma local y directa.
* El acceso depende de una **conexión activa a Internet**, ya que todo el sistema reside en la nube (Azure). Esto impide el uso offline.
* Las sesiones de usuarios no pueden mantenerse abiertas por tiempo prolongado sin actividad, por políticas de seguridad, lo que puede forzar cierres inesperados si se deja inactivo por mucho tiempo.

#### 3. ****Limitaciones Técnicas****

* **No hay soporte oficial multilingüe**. El sistema está desarrollado en español, lo que puede ser una barrera en instituciones con múltiples idiomas.
* La escalabilidad está condicionada a los recursos configurados en Azure. Aunque puede ampliarse, requiere intervención técnica para ajustar máquinas virtuales, ancho de banda o capacidad de almacenamiento.
* Aún no dispone de un sistema de **backup automático programado**, por lo que las copias de seguridad deben configurarse manualmente o desde Azure.

#### 4. ****Dependencia del Software Base****

* Depende de tecnologías específicas como **Apache, PHP y MySQL**, lo cual puede limitar su portabilidad inmediata a entornos con arquitecturas distintas (por ejemplo, entornos basados en .NET o PostgreSQL).
* El sistema está diseñado para navegadores modernos. Puede presentar problemas de compatibilidad o visualización en versiones antiguas de navegadores como Internet Explorer

# RECURSOS SOFTWARE

## Matriz de certificación

### Compatibilidad de Versiones

El sistema **EduFast**, en su versión actual, ha sido desarrollado y probado para asegurar una correcta operatividad con versiones específicas del software base y componentes externos. A continuación, se detalla la compatibilidad del sistema con cada uno de los elementos de su entorno:

|  |  |  |
| --- | --- | --- |
| **Componente** | **Versión Requerida** | **Compatibilidad Confirmada** |
| **Sistema Operativo** | Windows Server 2019 / Ubuntu 20.04+ | Compatible con ambas plataformas, siempre que se respeten los requisitos de entorno. |
| **Servidor Web** | Apache HTTP Server 2.4.x | Totalmente compatible. Configurado para interpretar scripts PHP. |
| **Lenguaje de servidor** | PHP 8.1.x | Versión mínima compatible. No se recomienda el uso de versiones anteriores. |
| **Servidor de base de datos** | MySQL 8.0.x | Compatible. Se recomienda configuración con usuarios específicos y roles. |
| **Sistema de despliegue en la nube** | Microsoft Azure for students | Compatible. Actualmente desplegado con App Service + MySQL flexible server oficialde . |
| **Navegadores** | Chrome 100+, Firefox 100+, Edge 100+ | Totalmente compatibles. No se garantiza funcionamiento en Internet Explorer. |
| **LDAP / Autenticación externa** | No aplica | El sistema actualmente no realiza integración con directorios externos. |
| **Máquina Virtual Java** | No aplica | No se requiere para el funcionamiento del sistema EduFast. |
| **Sistema de backups** | Azure Backup / Manual | Compatible con herramientas de respaldo de Azure. |

#### Notas Adicionales:

* Las pruebas de compatibilidad se realizaron en entornos controlados utilizando máquinas virtuales estándar en Azure.
* En caso de actualización de alguno de los componentes base (PHP, MySQL, Apache), se recomienda realizar pruebas de regresión para asegurar la continuidad operativa del sistema.
* No se garantiza la compatibilidad con tecnologías distintas a las especificadas (por ejemplo, servidores basados en IIS, PostgreSQL, o lenguajes como Java).

## Restricciones técnicas del sistema

## Recursos Software Necesarios

A continuación se presenta un listado detallado de todos los recursos software requeridos para la **compilación, instalación y configuración** del sistema **EduFast**, teniendo en cuenta su arquitectura basada en tecnologías web, despliegue en la nube (Azure) y gestión académica.

|  |  |  |  |
| --- | --- | --- | --- |
| **Recurso** | **Descripción** | **Versión Recomendada** | **Ubicación / Acceso** |
| **Sistema Operativo** | Sistema base sobre el que se ejecutan los servidores. | Windows Server 2019 o Ubuntu 20.04+ | Azure Virtual Machines / App Service |
| **Servidor Web** | Encargado de servir las páginas web. | Apache HTTP Server 2.4.x | Instalado en la máquina virtual o App Service |
| **Lenguaje de servidor** | Lenguaje principal de desarrollo backend del sistema. | PHP 8.1.x | Repositorio oficial: <https://www.php.net> |
| **Servidor de base de datos** | Almacena toda la información del sistema (usuarios, notas, grados, etc.). | MySQL 8.0.x | Azure Database for MySQL Flexible Server |
| **Cliente MySQL** | Herramienta de gestión para conectarse a la base de datos. | MySQL Workbench / Adminer | Cliente local o vía navegador web |
| **Servicios de Red** | Permiten la comunicación entre clientes y servidores. | HTTP/HTTPS, DNS | Gestionados por Azure |
| **Editor de código** | Utilizado por los desarrolladores para la modificación del código fuente. | Visual Studio Code | Local |
| **Compilador PHP** | No aplica (PHP es interpretado). | N/A | N/A |
| **Máquina Virtual Java** | No requerida por el sistema. | No aplica | No aplica |
| **Sistema de Backup** | Utilizado para crear copias de seguridad de los datos. | Azure Backup | Integrado en Azure Management |
| **Control de versiones** | Gestión de versiones del código fuente. | Git | GitHub / Azure Repos |

### Ubicación y Acceso a Servidores

* **Servidor Web + Aplicación**: Desplegado en **Azure App Service**, accesible mediante URL interna con permisos restringidos.
* **Servidor de Base de Datos (MySQL)**: Utiliza **Azure Database for MySQL Flexible Server**, accesible solo desde IPs autorizadas y cifrado TLS habilitado.
* **Backups**: Gestión desde el portal de Azure con política diaria de respaldo configurada.

## Requisitos de otros sistemas

En esta sección se describen los **requisitos necesarios para que la institución educativa pueda integrar EduFast** con otros sistemas relacionados para automatizar sus procesos académicos.

Los requisitos pueden ser:

**Administrativos**: registros, solicitudes de acceso, autorizaciones.

**Técnicos**: configuraciones, conexiones, restricciones de red.

### Sistema: Sistema de Gestión de Personal Institucional

**Descripción:** Sistema interno utilizado para la administración del personal docente y administrativo.  
**Funcionalidad utilizada:** Consulta y sincronización de datos del personal asociado a asignaturas, jornadas, cursos y grados.

* **Requisito administrativo 1:** Solicitud de credenciales de acceso para EduFast por parte del área TIC.
* **Restricción técnica 1:** El sistema EduFast debe tener conexión directa a la base de datos o servicios web del sistema de personal.
* **Restricción técnica 2:** Acceso restringido a través de una red segura o VPN institucional.

### Sistema: Sistema de Inventario Académico

**Descripción:** Sistema que gestiona los recursos físicos asignados a aulas y laboratorios.  
**Funcionalidad utilizada:** Consulta del inventario vinculado a materias o actividades académicas.

* **Requisito administrativo 1:** Alta de EduFast como sistema con permisos de solo lectura.
* **Restricción técnica 1:** API habilitada solo desde IPs internas de la institución.
* **Restricción técnica 2:** Validación mediante token institucional por cada solicitud.

### Sistema: Plataforma de Firma Digital

**Descripción:** Sistema que permite la firma electrónica de documentos institucionales.  
**Funcionalidad utilizada:** Firma digital de boletines, certificados y documentos oficiales.

* **Requisito administrativo 1:** Solicitud formal para el uso del sistema de firma por parte de la dirección de la institución.
* **Restricción técnica 1:** EduFast debe conectarse mediante los servicios web proporcionados por la plataforma.
* **Restricción técnica 2:** Los documentos deben cumplir con el formato PDF/A y firmarse con certificados válidos.

### Sistema: Sistema de Gestión de Personal Institucional

**Descripción:** Sistema interno utilizado para la administración del personal docente y administrativo.  
**Funcionalidad utilizada:** Consulta y sincronización de datos del personal asociado a asignaturas, horarios y grupos.

* **Requisito administrativo 1:** Solicitud de credenciales de acceso para EduFast por parte del área TIC.
* **Restricción técnica 1:** El sistema EduFast debe tener conexión directa a la base de datos o servicios web del sistema de personal.
* **Restricción técnica 2:** Acceso restringido a través de una red segura o VPN institucional.

### Sistema: Plataforma de Firma Digital

**Descripción:** Servicio para la firma electrónica de documentos oficiales generados por EduFast.

|  |  |
| --- | --- |
| **Elemento** | **Detalle** |
| **Funcionalidad usada** | Firma digital de boletines, certificados y otros documentos académicos. |
| **Req. administrativo 1** | Solicitud de uso formal firmada por la dirección de la institución. |
| **Restricción técnica 1** | Integración vía servicios web proporcionados por la plataforma. |
| **Restricción técnica 2** | Uso obligatorio del formato PDF/A con certificados digitales válidos. |

# INSTALACIÓN Y CONFIGURACIÓN DEL SOFTWARE BASE

### Instalación y Configuración del Proyecto con GitHub

En este apartado se describen los pasos necesarios para obtener el código fuente del sistema desde un repositorio GitHub, así como su correcta configuración previa al despliegue.

#### Requisitos Previos

|  |  |
| --- | --- |
| **Requisito** | **Descripción** |
| **Cuenta GitHub** | Tener una cuenta en <https://github.com> |
| **Acceso al repositorio** | Permisos de lectura (o escritura si se va a colaborar activamente) sobre el repositorio del proyecto |
| **Git instalado** | Tener instalado Git en el entorno local o en el servidor de despliegue |
| **Conexión a internet** | Acceso a la red para clonar el repositorio y descargar dependencias |

#### Procedimiento de Instalación

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **1** | Acceder a [GitHub](https://github.com) e iniciar sesión |
| **2** | Ubicar el repositorio del proyecto (público o privado) |
| **3** | Clonar el repositorio en local: |
|  | ```bash |
| **git clone** <https://github.com/Yisedcasti/edufast> |  |

### 1. ****Servidor Web – Apache****

|  |  |
| --- | --- |
| **Elemento** | **Descripción** |
| **Nombre del software** | Apache HTTP Server |
| **Función** | Servidor web para alojar la aplicación EduFast |
| **Dónde obtenerlo** | <https://httpd.apache.org/> o a través de XAMPP / paquetes del sistema |
| **Sistemas compatibles** | Linux, Windows |

#### Procedimiento de instalación

**En Ubuntu/Linux:**

sudo apt update

sudo apt install apache2

sudo systemctl enable apache2

sudo systemctl start apache2

**En Windows:**

* Descargar XAMPP desde <https://www.apachefriends.org/>
* Instalar seleccionando “Apache” en los servicios
* Iniciar Apache desde el panel de control de XAMPP
* Descomprimir el proyecto y mover la carpeta edufast dentro de xampp/localhost
* Usar la ruta de archivos para abrir el proyecto (de forma local).

#### Parámetros y configuración

|  |  |
| --- | --- |
| **Parámetro** | **Valor recomendado** |
| **Puerto** | 80 (o 8080 si hay conflicto) |
| **Document Root** | /var/www/html/edufast o C:\xampp\htdocs\edufast |
| **Módulos necesarios** | mod\_rewrite, mod\_ssl |

### 2. ****Servidor de Base de Datos – MySQL****

|  |  |
| --- | --- |
| **Nombre del software** | **MySQL Server** |
| **Función** | Gestión de base de datos relacional |
| **Dónde obtenerlo** | <https://dev.mysql.com/downloads/mysql/> |
| **Sistemas compatibles** | Linux, Windows |

#### Procedimiento de instalación

**En Ubuntu/Linux:**

bash

sudo apt install mysql-server

sudo systemctl enable mysql

sudo systemctl start mysql

**En Windows:**

* Instalar XAMPP esto incluye MySQL
* Iniciar desde el panel de XAMPP

#### Parámetros y configuración

|  |  |
| --- | --- |
| **Parámetro** | **Valor** |
| **Usuario** | Root |
| **Contraseña** | N/A |
| **Puerto** | 3306 |
| **Codificación** | utf8mb4 |
| **Base de datos usada** | edufast\_db |

### 3. ****PHP (Lenguaje de servidor)****

|  |  |
| --- | --- |
| **Nombre del software** | **PHP** |
| **Función** | Lenguaje backend que interpreta la lógica del sistema |
| **Dónde obtenerlo** | <https://www.php.net/downloads> |
| **Versión recomendada** | PHP 8.1 o superior |

#### Instalación

**En Ubuntu/Linux:**

bash

sudo apt install php libapache2-mod-php php-mysql

**En Windows:**

* XAMPP incluye PHP
* También puede instalarse por separado desde el instalador oficial

### ****PHPMailer (para envío de correos)****

|  |  |
| --- | --- |
| **Nombre del software** | **PHPMailer** |
| **Función** | Librería para envío de correos a través de SMTP en PHP |
| **Dónde obtenerlo** | LINK:<https://github.com/PHPMailer/PHPMailer> |

#### Instalación

* Descomprímelo y guárdalo dentro de la dirección de la carpeta raíz del proyecto.

### Otros

|  |  |  |
| --- | --- | --- |
| **Software** | **Función** | **Estado** |
| **Java Virtual Machine** | No necesaria para EduFast | No requerido |
| **Composer** | Gestor de dependencias PHP | Recomendado |
| **Git** | Control de versiones | Recomendado |

### ****1. Crear un Servidor MySQL en Azure****

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **1.1** | Inicia sesión en [Azure Portal](https://portal.azure.com). |
| **1.2** | Busca y selecciona **"Azure Database for MySQL - Flexible Server"**. |
| **1.3** | Haz clic en **"Crear"**. |
| **1.4** | Completa los campos requeridos: |
|  | - **Suscripción / Grupo de recursos**: selecciona o crea uno. |
|  | - **Nombre del servidor**: por ejemplo, edufast-db. |
|  | - **Región**: la más cercana al público objetivo. |
|  | - **Usuario admin**: por ejemplo, adminuser. |
|  | - **Contraseña segura**. |
|  | - **Versión**: selecciona **MySQL 8.0**. |
| **1.5** | Selecciona el tamaño: puedes comenzar con B1ms (para pruebas) o configuración personalizada. |
| **1.6** | Configura las **reglas de red**: activa "Acceso público" y permite tu IP actual (puedes editar luego). |
| **1.7** | Haz clic en **"Revisar + crear"** y luego en **"Crear"**. |

### ****2. Crear la Base de Datos Vacía****

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **2.1** | Una vez el servidor esté desplegado, accede al recurso. |
| **2.2** | En el menú lateral, selecciona **"Consola de consultas"** o usa un cliente como MySQL Workbench. |
| **2.3** | Conéctate al servidor con las credenciales definidas. |
| **2.4** | Ejecuta: |

sql

CREATE DATABASE edufast;

### ****3. Importar los Datos (desde un archivo**** .sql****)****

#### Usando ****MySQL Workbench****

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **3.1** | Abre **MySQL Workbench** y conecta con tu servidor Azure. |
| **3.2** | En el menú, selecciona **Server > Data Import**. |
| **3.3** | Marca **"Import from Self-Contained File"** y selecciona tu archivo .sql. |
| **3.4** | En **Default Target Schema**, selecciona edufast. |
| **3.5** | Haz clic en **Start Import**. |

### ****4. Verificación y Ajustes****

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **4.1** | Conéctate a la base de datos y verifica que las tablas estén presentes: |

sql

USE edufast;

SHOW TABLES;

| 4.2 | Comprueba los roles, cursos, jornadas y demás entidades cargadas correctamente. |

### ****Parámetros a Verificar en Azure****

|  |  |
| --- | --- |
| **Parámetro** | **Valor recomendado** |
| **Versión de MySQL** | 8.0.x |
| **TLS** | Habilitado (obligatorio en Azure) |
| **Tamaño de almacenamiento** | Al menos 5 GB (escalable hasta 100 GB o más) |
| **Backup automático** | Activado (por defecto, 7 días) |
| **Firewall de acceso** | IPs permitidas específicas |
| **Autenticación SSL** | Requerida (usar certificado de Azure si es local) |

# CONFIGURACIÓN DEL SISTEMA

### ****Parámetros de conexión a la base de datos (Datasource)****

|  |  |
| --- | --- |
| **Elemento a Configurar** | **Efecto** |
| **Datasource** | Permite que la aplicación se conecte correctamente con la base de datos. |
| **Fase** | **Ubicación** |
| **Configuración final después del despliegue** | /var/www/edufast/config/bd.properties (o ruta en tu contenedor o servidor web) |

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **1º** | Asegurarse de que la aplicación esté desplegada y el servidor web (Apache, Nginx, etc.) o backend esté detenido. |
| **2º** | Abrir el archivo bd.properties ubicado en el directorio indicado. |
| **3º** | Establecer los siguientes valores: |
|  | - cad\_conexion: jdbc:mysql://<host>.mysql.database.azure.com:3306/edufast |
|  | - driver: com.mysql.cj.jdbc.Driver |
|  | - nombre: edufast |
|  | - usuario: adminuser@edufast-db |
|  | - password: \*\*\*\*\*\*\*\* (contraseña segura) |
| **4º** | Guardar y cerrar el archivo. |
| **5º** | Reiniciar el servicio de backend o el contenedor que corre el backend para aplicar los cambios. |

### ****Configuración: Variables de entorno para entorno de producción****

|  |  |
| --- | --- |
| **Elemento a Configurar** | **Efecto** |
| **Variables de entorno** | Definen parámetros sensibles como claves, rutas o entorno de ejecución. |
| **Fase** | **Ubicación** |
| **Instalación y despliegue** | Archivo .env o configuración de contenedor |

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **1º** | Crear o editar el archivo .env en el directorio raíz del backend. |
| **2º** | Establecer los siguientes valores: |
|  | - APP\_ENV=production |
|  | - DB\_HOST= db-edufast.database.windows.net |
|  | - DB\_PORT=3306 |
|  | - DB\_USER=admin01 |
|  | - DB\_PASS=edufast01. |
|  | - DB\_NAME=edufast |
| **3º** | Guardar el archivo. |
| **4º** | Asegurarse de que los servicios de backend lean este archivo o estén configurados para ello. |

### ****Configuración: Permisos iniciales del sistema****

|  |  |
| --- | --- |
| **Elemento a Configurar** | **Efecto** |
| **Roles y permisos** | Define qué puede hacer cada tipo de usuario en el sistema. |
| **Fase** | **Ubicación** |
| **Inicial** | Panel de administración o script SQL |

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **1º** | Acceder al panel de administración como usuario administrador. |
| **2º** | Crear los roles: Administrador, Docente, Estudiante. |
| **3º** | Asignar permisos según el módulo (asistencias, cursos, logros, notas...). |
| **4º** | Guardar cambios. |
| **5º** | Alternativamente, ejecutar script SQL para preconfigurar roles. |

## Configuración de otros sistemas

### ****Sistema no base: PHPMailer (Envío de correos)****

|  |  |
| --- | --- |
| **Elemento** | **Descripción** |
| **Nombre** | PHPMailer |
| **Función** | Librería para el envío de correos electrónicos desde el backend del sistema. |
| **Localización** | <https://github.com/PHPMailer/PHPMailer> |

#### 🔧 Procedimiento de Instalación

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **1º** | Descargar PHPMailer desde el repositorio oficial o instalar con Composer: composer require phpmailer/phpmailer |
| **2º** | Incluir en el proyecto PHP con: use PHPMailer\PHPMailer\PHPMailer; require 'vendor/autoload.php'; |
| **3º** | Verificar que el servidor permite salida por SMTP (puertos abiertos y configurados). |

#### ⚙️ Procedimiento de Configuración

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **1º** | Configurar el host SMTP del proveedor de correo (por ejemplo, smtp.gmail.com) |
| **2º** | Establecer credenciales seguras (correo remitente y contraseña) en variables de entorno o archivo de configuración |
| **3º** | Configurar el puerto (587 o 465) y el tipo de cifrado (TLS o SSL) |
| **4º** | Probar el envío con un correo de prueba |

#### ⚙️ Parámetros a configurar

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Ubicación** | **Valor Ejemplo** |
| **SMTP Host** | /config/email\_config.php | smtp.gmail.com |
| **SMTP User** | /config/email\_config.php | usuario@gmail.com |
| **SMTP Password** | /config/email\_config.php | \*\*\*\*\*\*\*\* |
| **SMTP Port** | /config/email\_config.php | 587 |
| **Encryption** | /config/email\_config.php | tls |

**Descripción:**  
Sistema de respaldo automático de la base de datos utilizado para generar copias de seguridad periódicas del sistema en formato .sql, exportadas localmente o a almacenamiento en la nube.

**Localización:**

* En sistemas Linux: /usr/bin/mysqldump
* En sistemas Windows: Incluido en la instalación de MySQL o XAMPP, ruta: C:\xampp\mysql\bin\mysqldump.exe

### Procedimiento de instalación

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **Paso 1** | Verificar que mysqldump esté instalado. En Linux: which mysqldump. En Windows: comprobar ruta. |
| **Paso 2** | Crear un script .sh (Linux) o .bat (Windows) con el comando de respaldo. |
| **Paso 3** | Establecer permisos de ejecución en Linux: chmod +x backup.sh. |
| **Paso 4** | Programar el respaldo diario/semanal con crontab (Linux) o "Tareas programadas" en Windows. |

### Procedimiento de configuración

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **Paso 1** | Editar el script para incluir credenciales, nombre de base de datos y ruta de salida. |
| **Paso 2** | En Linux, abrir el crontab con crontab -e y añadir: 0 2 \* \* \* /ruta/backup.sh (ejecuta diario a las 2am). |
| **Paso 3** | En Windows, crear una tarea programada que ejecute el .bat en la frecuencia deseada. |
| **Paso 4** | Validar que se esté generando el archivo .sql correctamente. |

### Parámetros a configurar

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Ubicación (BBDD, fichero xml, properties...)** | **Valor ejemplo** |
| **Usuario BDD** | /scripts/backup.sh o .bat | root |
| **Contraseña BDD** | /scripts/backup.sh o .bat | 123456 |
| **Nombre base de datos** | /scripts/backup.sh o .bat | sistema\_educativo |
| **Ruta de salida del backup** | /scripts/backup.sh o .bat | /home/backups/backup\_$(date +%F).sql (Linux) |
| **Frecuencia** | crontab o “Tareas programadas” | Diaria a las 2:00 AM |

# COMPILACIÓN DEL SISTEMA

### ****Requisitos de compilación****

|  |  |
| --- | --- |
| **Requisito** | **Descripción** |
| **Ubicación Fuentes** | Repositorio Git ubicado en: https://dev.azure.com/miempresa/proyecto/\_git/sistema-gestion  Estructura estándar Maven: /src/main/java, /src/main/resources, /src/main/webapp. |
| **Configuración 1** | Configuración de pom.xml:  - Uso de maven-compiler-plugin para compilar con Java 17.  - Profile activo: azure-deploy. |
| **Configuración 2** | En el pom.xml se incluyen los plugins: - maven-war-plugin para empaquetar como .war - maven-surefire-plugin para pruebas unitarias. |
| **Repositorio 1** | Repositorio central Maven: https://repo.maven.apache.org/maven2 |
| **Repositorio 2** | Azure Artifacts (privado, autenticación mediante token PAT) |
| **Dependencia 1** | Spring Boot Starter Web: org.springframework.boot:spring-boot-starter-web:2.7.0 |
| **Dependencia 2** | Conector MySQL: mysql:mysql-connector-java:8.0.33 |
| **Dependencia 3** | Apache Commons IO: commons-io:commons-io:2.11.0 |
| **Producto final** | Archivo .war: sistema-gestion.war, generado en el directorio target/ tras compilación exitosa. |

### ****Procedimiento de compilación****

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **1** | **Clonar el repositorio del sistema** desde Azure DevOps o GitHub: git clone <https://dev.azure.com/miempresa/proyecto/_git/sistema-gestion> |
| **2** | **Verificar la instalación de Java y Maven**: - Asegúrate de tener JDK 17 configurado (java -version) - Maven 3.8+ instalado (mvn -v) |
| **3** | **Acceder al directorio del proyecto**: cd sistema-gestion |
| **4** | **Editar (si es necesario) el archivo pom.xml** para activar el profile adecuado (azure-deploy, por ejemplo) y verificar las dependencias. |
| **5** | **Ejecutar la limpieza y compilación del proyecto**: mvn clean install -Pazure-deploy |
| **6** | **Verificar el producto final generado** en el directorio target/: sistema-gestion.war debe estar disponible y sin errores de empaquetado. |

# INSTALACIÓN DEL SISTEMA

#### Requisitos Previos

|  |  |
| --- | --- |
| **Requisito** | **Descripción** |
| **Sistema operativo** | Windows Server 2022 o Ubuntu Server 22.04 (compatible con Java y Tomcat) |
| **Java Development Kit (JDK)** | JDK 17 instalado y configurado en la variable de entorno JAVA\_HOME |
| **Servidor de aplicaciones** | Apache Tomcat 10 instalado y configurado |
| **Servidor de base de datos** | Azure SQL Database con una instancia activa y configurada |
| **Acceso a internet** | Requerido para clonar repositorio, descargar dependencias y acceder a servicios en la nube |
| **Maven** | Apache Maven 3.8+ para compilar el proyecto |
| **Cuenta de Azure activa** | Para desplegar recursos, base de datos y configurar servicios |

#### Procedimiento de instalación

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **1** | Clonar el repositorio del sistema desde Azure DevOps o GitHub:  git clone https://dev.azure.com/... |
| **2** | Acceder al directorio del proyecto:  cd sistema-gestion |
| **3** | Compilar el sistema utilizando Maven:  mvn clean install -Pazure-deploy |
| **4** | Verificar que el archivo .war generado se encuentra en la carpeta target/ |
| **5** | Copiar el archivo .war al directorio webapps/ de Tomcat |
| **6** | Iniciar el servidor Tomcat:  startup.bat (Windows) o ./startup.sh (Linux) |
| **7** | Acceder al sistema a través del navegador web:  http://localhost:8080/sistema-gestion |
| **8** | Verificar conexión a la base de datos configurando correctamente el archivo bd.properties |
| **9** | Realizar pruebas básicas de funcionamiento (login, carga de módulos, etc.) |

## Instalación en Azure (App Service) con Azure for Students

### Requisitos Previos

|  |  |
| --- | --- |
| **Requisito** | **Descripción** |
| **Cuenta de Azure for Students** | Debes tener una [cuenta activa](https://azure.microsoft.com/free/students/) con tu correo institucional. Incluye 100 USD en crédito y servicios gratuitos. |
| **Grupo de recursos** | Puedes crearlo desde el portal o con Azure CLI. Agrupa los recursos relacionados al proyecto. |
| **Azure App Service** | Crea un plan gratuito o de bajo costo (F1 o B1) si estás limitado por créditos. Asegúrate de que sea compatible con Java. |
| **Azure SQL Database** | Puedes usar el servicio gratuito incluido (250 MB de almacenamiento). Crea una base de datos y servidor lógico. |
| **Stack Java configurado** | En App Service, selecciona **Java 17** y **Tomcat 10** para aplicaciones tipo .war. |
| **Azure CLI** | Descárgalo en [Azure CLI](https://learn.microsoft.com/cli/azure/install-azure-cli) para automatizar tareas desde consola. |
| **Maven y JDK local** | Requerido para compilar tu aplicación Java en tu máquina local. |
| **IDE opcional** | Puedes usar VS Code, IntelliJ o Eclipse con extensiones de Azure si prefieres entorno gráfico. |

### ⚙️ Procedimiento de Instalación

|  |  |
| --- | --- |
| **Paso** | **Descripción** |
| **1** | **Compila el proyecto localmente**: mvn clean package -Pazure-deploy |
| **2️** | **Accede al portal de Azure**: <https://portal.azure.com> |
| **3️** | **Crea un App Service**: - Tipo: Web App  - Stack: Java 17 con Tomcat 10  - Plan: F1 (gratuito) o B1  - OS: Linux o Windows según necesidades |
| **4️** | **Ve a "Centro de implementación"** dentro del recurso App Service |
| **5️** | **Selecciona método de despliegue**: - Local Git - FTP - Automatizado con Azure CLI |
| **6️** | **Sube el archivo .war**: az webapp deploy --resource-group <grupo> --name <nombre-app> --src-path target/miapp.war |
| **7️** | **Agrega las variables de entorno** (desde "Configuración > Configuración de aplicación"): DATABASE\_URL, DATABASE\_USER, DATABASE\_PASSWORD |
| **8️** | **Configura el firewall de la Azure SQL Database** para permitir conexiones desde el App Service. |
| **9️** | **Verifica la URL del App Service**: https://Appedufast.azurewebsites.net |
| **10** | **Valida la conexión a base de datos** y prueba las funcionalidades principales. |

### ✅ Observaciones Específicas para Azure for Students

* El **Plan Gratuito (F1)** en App Service tiene limitaciones de recursos (1 GB RAM, sin SSL personalizado, CPU compartida).
* Puedes usar **Azure SQL gratuito (250 MB)** si tu base de datos es pequeña.
* Los créditos gratuitos son limitados (100 USD por 12 meses), así que **optimiza recursos**: elimina servicios no utilizados y apaga recursos cuando no estén en uso.
* También puedes integrar fácilmente con **GitHub Actions** sin costo adicional para CI/CD.

### 📌 Ejemplo de Configuración de Variables de Entorno

DATABASE\_URL=jdbc:sqlserver://<server>.database.windows.net:1433;database=miBD

DATABASE\_USER=adminuser

DATABASE\_PASSWORD=\*\*\*\*\*\*\*\*

## Requisitos previos

Este apartado recoge todos los elementos necesarios antes de iniciar el proceso de instalación del sistema. Algunos de estos requisitos se detallan con mayor profundidad en apartados posteriores (por ejemplo, compilación o configuración), por lo que aquí se referencian si corresponde.

|  |  |
| --- | --- |
| **Requisito** | **Descripción** |
| **Ficheros y recursos** | Artefacto .war o .jar generado mediante Maven (target/miApp.war). Archivo bd.properties para configuración de base de datos. |
| **Ubicación de servidores** | - App Service en Azure: https://edufast.azurewebsites.net  - Base de datos:db\_edufast.database.windows.net |
| **Modo de acceso a servidores** | - Acceso al portal de Azure: credenciales de usuario con permisos sobre el grupo de recursos.  - Azure CLI o Visual Studio Code. |
| **Software preinstalado** | - Azure App Service debe tener configurado Java 17 + Tomcat 10.  - Azure SQL Database preconfigurada. |
| **Requisitos de comunicación** | - El App Service debe tener acceso de red a la base de datos (configuración del firewall de Azure SQL).  - Permitir tráfico HTTP/HTTPS. |
| **Altas necesarias** | - Creación del usuario administrador en la base de datos.  - Alta del App Service con su identidad para posibles integraciones con Azure AD. |
| **Paradas o conflictos** | - Verificar que no exista otra instancia de la aplicación corriendo en el mismo nombre de dominio.  - Detener versiones anteriores. |
| **Repositorios Maven accesibles** | - Acceso a https://repo.maven.apache.org/maven2 desde el entorno local para descargar dependencias. |
| **Claves y tokens necesarios** | - Variables de entorno: DATABASE\_URL, DATABASE\_USER, DATABASE\_PASSWORD.  - Token de despliegue Git o FTP en Azure. |

## Procedimiento de instalación

## Instalación del Sistema – Detalle de Tareas (Software Base)

En este caso, la instalación parte del software base, por lo que **no se requiere una instalación completa desde cero**. El producto se entrega ya compilado y listo para desplegar. Las acciones a realizar se resumen en:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Paso** | **Tipo de Acción** | **Descripción** | **Componentes Implicados** | **Permisos Necesarios** |
| **1** | Verificación | Verificar que el software base se encuentra correctamente instalado (consultar apartado 5) | N/A | Acceso a la máquina o entorno donde está desplegado |
| **2** | Descarga | Descargar el artefacto (archivo .war o .jar) del sistema entregado | Archivo .war o .jar | Permisos de lectura sobre el repositorio o medio de entrega |
| **3** | Despliegue local | Ejecutar el archivo .jar en localhost o desplegar el .war en un servidor como Tomcat o Azure | Artefacto entregado | Usuario local con permisos para ejecutar aplicaciones |
| **4** | Configuración mínima | (Opcional) Ajustar archivo de configuración (por ejemplo application.properties) si cambia entorno | Archivo de configuración | Acceso de escritura al archivo |
| **5** | Verificación | Abrir el navegador y verificar que el sistema está funcionando | App ejecutada | Usuario tester o administrador |

# VERIFICACIÓN DEL PROCESO DE INSTALACIÓN

## ✅ Comprobaciones tras el Despliegue

Una vez finalizada la instalación o despliegue del sistema, se deben realizar una serie de **comprobaciones mínimas** para garantizar que todo está correctamente configurado y funcionando. Estas pruebas aseguran la operatividad del sistema, la comunicación con servicios externos y la integridad del despliegue.

|  |  |  |
| --- | --- | --- |
| **Comprobación** | **Descripción** | **Resultado Esperado** |
| **Acceso al sistema** | Acceder a la URL del sistema (localhost o entorno remoto como Azure) | Página de inicio visible sin errores |
| **Inicio de sesión (si aplica)** | Iniciar sesión con credenciales válidas | Acceso permitido al sistema |
| **Funcionalidad básica** | Navegar por las principales secciones del sistema (menús, formularios, vistas, etc.) | Carga y navegación sin errores |
| **Conexión a base de datos** | Validar que el sistema se conecta correctamente a la base de datos | Datos cargados desde BBDD |
| **Variables de entorno** | Verificar que las configuraciones definidas (ej. cadena JDBC, claves API) funcionan correctamente | Aplicación responde con datos correctos |
| **Logs del servidor** | Revisar que no existan errores críticos en los logs de despliegue o ejecución | Logs sin errores importantes |
| **Comunicación con servicios externos** | (Si aplica) Validar que los servicios externos (API REST, SMTP, etc.) están disponibles | Respuestas válidas desde los servicios |
| **Rendimiento inicial** | Comprobar el tiempo de carga y respuesta en el primer uso | Respuesta fluida, sin bloqueos |

### Observaciones

* En caso de que alguna comprobación falle, se recomienda revisar los logs de la aplicación o consola del navegador (en frontend).
* Si se desplegó en Azure App Service, se pueden usar las herramientas de diagnóstico incluidas en el portal (por ejemplo, "Diagnóstico y resolución de problemas").
* Para asegurar la estabilidad, puede ejecutarse un test funcional completo o una prueba manual por parte del equipo QA.

# MARCHA ATRÁS DE LA INSTALACIÓN Y CONFIGURACIÓN

## Requisitos previos

## Requisitos Previos para la Marcha Atrás

Antes de realizar cualquier procedimiento de marcha atrás (rollback) del sistema, es fundamental tener en cuenta ciertos **requisitos previos** que aseguren que el proceso pueda realizarse de manera segura, controlada y sin pérdida irreversible de datos. A continuación se detallan los aspectos más relevantes:

|  |  |
| --- | --- |
| **Requisito** | **Descripción** |
| **Copia de seguridad completa** | Antes de iniciar la marcha atrás, debe haberse generado una copia de seguridad completa del sistema, incluyendo base de datos, archivos de configuración y artefactos desplegados (.war, .jar, etc.). |
| **Plan de reversión documentado** | Se debe contar con un procedimiento detallado y probado para revertir a la versión anterior del sistema o estado estable. |
| **Registro de cambios** | Deben estar documentados todos los cambios realizados durante la instalación o actualización, para saber qué elementos deben deshacerse. |
| **Accesos y permisos** | El usuario que ejecuta la marcha atrás debe tener permisos de administrador sobre los servidores, base de datos y entorno de despliegue (por ejemplo, Azure App Service). |
| **Disponibilidad del entorno previo** | Asegurarse de que el entorno anterior (software base o versión anterior del sistema) esté disponible para restauración inmediata si se requiere. |
| **Revisión de conflictos** | Verificar si existen procesos en ejecución, archivos bloqueados o dependencias activas que puedan interferir en la reversión. |
| **Coordinación con otros sistemas** | Si el sistema se comunica con otros entornos o servicios, validar que la reversión no genere inconsistencias o errores de integración. |

## Marcha atrás del sistema

## Desinstalación y Marcha Atrás de la Instalación

En este apartado se detalla la secuencia ordenada de tareas a realizar en caso de que se requiera **desinstalar el sistema** y revertir la instalación realizada previamente. El objetivo es devolver el entorno a un estado estable anterior, asegurando que no queden rastros ni configuraciones residuales que puedan afectar futuras instalaciones.

### Tareas de Reversión

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Paso** | **Tipo de Tarea** | **Descripción** | **Componente Implicado** | **Permisos Necesarios** |
| **1** | Validación previa | Asegurar que existe una copia de seguridad actualizada de la base de datos y archivos del sistema | Sistema completo | Administrador del servidor, propietario BBDD |
| **2** | Eliminación del artefacto desplegado | Eliminar el archivo .war o .jar desplegado en el servidor de aplicaciones o Azure App Service | App Service o servidor local | Permisos de administrador sobre el entorno |
| **3** | Eliminación de configuración | Eliminar variables de entorno, cadenas de conexión y configuraciones específicas realizadas para el sistema | App Service (Azure) / archivo application.properties / entorno local | Administrador |
| **4** | Reversión de cambios en la base de datos | Restaurar el esquema de base de datos desde una copia previa o ejecutar scripts de reversión si los cambios fueron parciales | Base de datos (SQL Server, MySQL, etc.) | Propietario del esquema o DBA |
| **5** | Limpieza de archivos temporales | Eliminar directorios de logs, archivos temporales o configuraciones residuales generadas durante la ejecución del sistema | Sistema de archivos del servidor / contenedor | Permisos de escritura/administrador |
| **6** | Eliminación del recurso | eliminar el recurso App Service o el grupo de recursos si ya no es necesario | Azure Resource Group / App Service | Permisos de administrador en Azure for students |
| **7** | Verificación final | Confirmar que todos los componentes han sido eliminados correctamente y que no quedan procesos en ejecución ni dependencias activas | Todos los componentes | Administrador |

## Desinstalación y Marcha Atrás de la Instalación (Despliegue en Azure)

### Requisitos Previos

Antes de iniciar la marcha atrás, asegúrate de cumplir con los siguientes puntos:

* Haber realizado una copia de seguridad de la base de datos (Azure SQL Database).
* Tener permisos suficientes en el portal de Azure para eliminar recursos (App Service, grupos de recursos, configuraciones, etc.).
* Notificar al equipo de desarrollo o usuarios sobre la interrupción del servicio (si aplica).

### Tareas de Reversión

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Paso** | **Tipo de Tarea** | **Descripción** | **Componente Implicado** | **Permisos Necesarios** |
| **1** | Copia de seguridad | Descargar o exportar los datos de la base de datos si es necesario conservarlos | Azure SQL Database | Permisos de administrador en SQL y Azure |
| **2** | Detener la aplicación | Detener temporalmente el App Service para evitar nuevas conexiones | Azure App Service | Permisos de administrador del App Service |
| **3** | Eliminar archivo desplegado | Si fue cargado vía FTP o CLI, eliminar el archivo .war o .jar desde el centro de implementación | Azure App Service | Permisos de escritura/deploy |
| **4** | Eliminar configuraciones del entorno | Quitar las variables de entorno (DATABASE\_URL, DATABASE\_USER, etc.) desde la pestaña **Configuración > Configuración de aplicación** | Azure App Service | Permisos de edición del App Service |
| **5** | Restaurar base de datos (opcional) | Ejecutar scripts de rollback o restaurar backup si hubo cambios en el esquema | Azure SQL Database | DBA o propietario del esquema |
| **6** | Eliminar App Service | Borrar el App Service si no será reutilizado | Azure Portal / Azure CLI | Propietario del recurso |
| **7** | Eliminar grupo de recursos (opcional) | Si todo fue contenido dentro de un mismo Resource Group y ya no se necesita | Azure Resource Group | Permisos de propietario sobre el recurso |
| **8** | Verificación final | Verificar que la URL del App Service no responde y que no hay recursos asociados activos | Azure Portal / CLI | Administrador |

### Comprobaciones Finales

* Confirmar que el App Service ha sido eliminado correctamente.
* Verificar que no existen bases de datos residuales si también se eliminaron.
* Validar que no quedan logs ni artefactos en el entorno de compilación.
* Revisar el historial del portal de Azure o de Azure DevOps (si se usó CI/CD) para confirmar que la reversión fue completa.

## Marcha atrás del software base

## Desinstalación y Marcha Atrás del Software Base

Este apartado detalla los pasos necesarios para desinstalar y revertir la instalación y configuración de los componentes que forman el **software base** del sistema. Las acciones están organizadas por tipo de software y localización.

### 🗄️ MySQL Server (Local)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Paso** | **Tipo de Tarea** | **Descripción** | **Componente Implicado** | **Permisos Necesarios** |
| **1** | Backup | Exportar la base de datos (si aplica) con mysqldump | MySQL | Usuario con permisos DBA |
| **2** | Detención del servicio | sudo systemctl stop mysql o desde el panel de servicios (Windows) | Servicio MySQL | Sudo/Admin |
| **3** | Desinstalación | sudo apt purge mysql-server o desinstalar en Windows | MySQL Server | Admin |
| **4** | Limpieza | Eliminar carpetas residuales /var/lib/mysql o archivos de configuración | Archivos de configuración | Sudo |
| **5** | Verificación | Confirmar que el puerto 3306 está libre y MySQL no está activo | Terminal | Usuario |

### 🌐 Servidor Web / Contenedor Local

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Paso** | **Tipo de Tarea** | **Descripción** | **Componente Implicado** | **Permisos Necesarios** |
| **1** | Detener servicio o contenedor | systemctl stop tomcat o docker stop contenedor | Tomcat / Docker | Admin/Sudo |
| **2** | Eliminar archivos | Borrar directorios de instalación (webapps/, etc.) | Sistema de archivos | Admin |
| **3** | Eliminar imagen/contenedor (Docker) | docker rm y docker rmi | Docker | Admin local |
| **4** | Verificación | Acceder a localhost:8080 y comprobar que no responde | Navegador o curl | Usuario |

### 🔐 Observaciones Generales

* Es fundamental contar con permisos de **administrador local o sudo** para eliminar software base instalado a nivel de sistema.
* Se recomienda realizar **copias de seguridad de configuración y datos** antes de proceder con cualquier eliminación.
* En caso de que la instalación base esté **en un servidor compartido o remoto**, coordinar la desinstalación con el administrador de sistemas correspondiente.

# ANEXOS

### Información adicional para la elaboración y validación del Manual de Instalación

Este apartado recoge toda la información complementaria que puede resultar de utilidad tanto para la **elaboración** como para la **validación** del presente Manual de Instalación. Incluye consideraciones, recomendaciones y buenas prácticas que deben tenerse en cuenta para asegurar una correcta comprensión, ejecución y mantenimiento del procedimiento de instalación descrito.

#### Puntos relevantes:

* **Claridad y precisión:** Es fundamental que todas las instrucciones, comandos y requisitos estén claramente especificados y no den lugar a interpretaciones ambiguas.
* **Referencias cruzadas:** Siempre que sea posible, debe hacerse referencia a los apartados anteriores del documento (por ejemplo, configuración, requisitos previos, software base) para evitar duplicidades y mantener la coherencia.
* **Actualización del contenido:** Cualquier cambio en los componentes del sistema (versiones, dependencias, servicios en la nube, etc.) debe reflejarse en este manual para garantizar su vigencia.
* **Validación del proceso:** Se recomienda realizar pruebas de instalación en un entorno controlado para verificar que todos los pasos permiten la correcta puesta en marcha del sistema.
* **Registro de incidencias:** Durante la instalación, cualquier error, advertencia o comportamiento inesperado deberá documentarse para su posterior análisis y mejora del proceso.

## Resumen de tareas de configuración

### Resumen de elementos de configuración para la puesta en marcha

Para garantizar la correcta instalación y operación del sistema en el entorno deseado, es necesario actualizar y verificar los siguientes elementos de configuración:

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción / Detalle** | **Momento / Fase** |
| **Sistema operativo** | Windows Server 2022 o Ubuntu Server 22.04, compatible con Java y Tomcat | Antes de la instalación |
| **Java** | JDK 17 instalado y variable de entorno JAVA\_HOMEconfigurado | Antes de compilar y ejecutar |
| **Servidor de aplicaciones** | Apache Tomcat 10 instalado | Antes de desplegar la aplicación |
| **Servidor de base de datos** | Conversión de la base de datos SQL de Azure | Antes de configurar la conexión |
| **Archivo de configuración de base de datos (bd.properties)** | Parámetros de conexión (URL, usuario, contraseña) actualizados para el entorno correspondiente | Durante configuración post-despliegue |
| **Variables de entorno en Azure App Service** | DATABASE\_URL, DATABASE\_USER, DATABASE\_PASSWORD configuradas en el portal de Azure | Al configurar App Service |
| **Fuego** | Regla para permitir conexiones desde App Service | Alabama |
| **Experto** | Apache Maven 3.8+ para compilar el proyecto con el perfil adecuado (azure-deploy) | Antes de compilar |
| **Repositorio Git** | URL correcta para clonar el repositorio del sistema | Inicio del procedimiento |
| **Archivo .war** | Ubicación correcta en target/ y despliegue en webapps/ de Tomcat o despliegue en Azure | Durante despliegue |
| **Cuenta de Azure activa** | Cuenta con permisos para crear recursos, App Service y base de datos | Al desplegar en Azure |
| **Configuración de App Service** | Stack Java 17 y Tomcat 10 seleccionados, plan adecuado (F1 o B1), configuración OS | Durante creación del recurso |
| **Conexión a internet** | Disponible para clonar repositorios, descargar dependencias y acceder a servicios en la nube |  |

# GLOSARIO

<Insertar comentario y eliminar cuadro>

Este punto contendrá la definición de todos los términos utilizados en el presente documento.

|  |  |
| --- | --- |
| **Término** | **Descripción** |
|  |  |
|  |  |
|  |  |

# BIBLIOGRAFÍA Y REFERENCIAS

<Insertar comentario y eliminar cuadro>

En este punto se incluirán las referencias a la documentación utilizada para la elaboración del presente documento.

|  |  |  |
| --- | --- | --- |
| **Referencia** | **Título** | **Código** |
| Ref. 1 |  |  |
|  |  |  |
|  |  |  |
|  |  |  |