CONTENIDO

[1. Arquitectura software 2](#_Toc126047314)

[1.1 Frameworks 2](#_Toc126047315)

[1.2 Componentes 3](#_Toc126047316)

[1.3 Tipo de software 4](#_Toc126047317)

[1.4 Arquitectura 4](#_Toc126047318)

[1.5 Diagrama de componentes 4](#_Toc126047319)

[1.6 Tecnologías 4](#_Toc126047320)

[1.7 Diagrama de despliegue 5](#_Toc126047321)

[1.8 Proceso de desarrollo 5](#_Toc126047322)

[2. Arquitectura de seguridad 5](#_Toc126047323)

[2.1 Desarrollo seguro 5](#_Toc126047324)

[2.2 Ethical Hacking 5](#_Toc126047325)

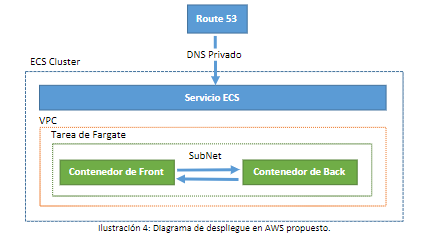
[Arquitectura de integración 5](#_Toc126047326)

[2.3 Mecanismos de integración 5](#_Toc126047327)

[3. Soporte técnico 6](#_Toc126047328)

# Arquitectura software

La solución Datlas de Bancoldex esta soportada por contenedores de AWS y está conformada por 2 capas front y back como se muestra en el siguiente diagrama:



## Frameworks

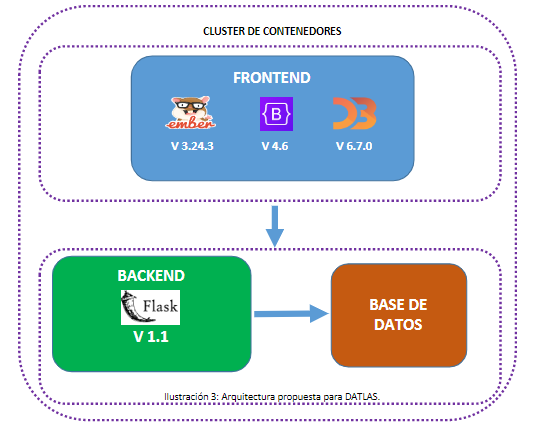
Datlas se encuentra construido en los siguientes frameworks.

Front

* Ember.js–v3.24.3
* Node.js–v6.13.1
* Ember-cli–v1.13.1

Backend

* Python–v3.4.3
* Flask–v1.1
* Atlas\_core–v0.2.8 (Dependencia de git)
* Linnaeus–v0.0.96 (Dependencia de git)
* Reckoner–v0.0.1 (Dependencia de git)
* Flask-sqlalchemy–v2.0 (Dependencia fork en git)

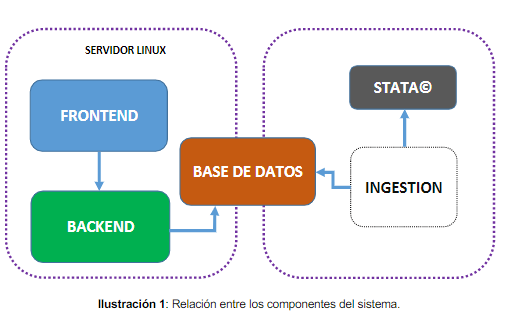


## Componentes

El sistema Datlas está compuesto un proceso back que alimenta la BD, un componente backend que expone la información por ApiRest y un componente frontend que recibe la información y la muestra al usuario.

El Backend está desarrollado en Python y se hace uso del framework Flask que añade una capa extensible de funcionalidades web al lenguaje nativo, el Frontend está basado en JavaScript y en el framework Ember.js.

El sistema de gestión de bases de datos es SQLITE y se encuentra almacenado en un archivo sobre el componente de Backend, este archivo de base de datos se construye mediante el proceso de ingestión (Procesos de ETL a través del paquete Pandas de Phyton) de los archivos de datos generados por el software STATA©.



## Tipo de software

El banco desea una solución bajo un esquema de desarrollo evolutivo, donde la propiedad del software será del banco y la instalación con sus componentes la infraestructura actual Nube AWS en la cuenta del banco.

## Arquitectura

Se deberá describir la arquitectura de solución que el proponente ofrece en su solución, confirmar si se conserva la arquitectura actual o si esta es modificada con sus respectivas capas a nivel de software. El proponente debe presentar un blueprint de servicios del sistema donde describa los cambios a desarrollar los servicios que van a soportar las necesidades funcionales hacia los clientes del banco.

## Diagrama de componentes[[1]](#footnote-2)

Se deberá actualizar el diagrama de alto nivel donde se describan claramente cada una de las capas, componentes y/o módulos de cómo va a quedar la solución del sistema.

## Tecnologías

Se deberá presentar un diagrama actualizado donde se mapean las tecnologías de software y de infraestructura de cada una de las capas del sistema.

## Diagrama de despliegue[[2]](#footnote-3)

Se deberá presentar la actualización del diagrama de despliegue donde se explique claramente como la arquitectura del sistema será desplegada.

## Proceso de desarrollo

Se deberá presentar la descripción del proceso de desarrollo que se implementará para la actualización de la solución. Se debe mantener el esquema GitHub de despliegue continuo. Importante mencionar cuales son los mecanismos que garantizan el desarrollo de software seguro desde el proceso de construcción y evolución del sistema.

# Arquitectura de seguridad

## Desarrollo seguro

El proponente debe confirmar si cuenta con un proceso de desarrollo seguro y hacer la descripción de como hace la implementación de este, detallando las herramientas o tareas que realiza para cumplir con este requerimiento. Así mismo deberá entregar un informe de la ejecución de pruebas de desarrollo seguro donde se evidencie que el software desarrollado no tiene fallas de seguridad.

Adicionalmente por favor responder las siguientes preguntas:

1. ¿Cuenta con procedimientos de control de cambios?
2. ¿Cuenta con funcionalidad para el control de versiones de software?
3. ¿Cuenta con procedimientos y controles para el paso de programas a producción y el software en operación es catalogado?
4. ¿Cuenta con un esquema backup donde se garantiza el respaldo de la información del Banco?

## Ethical Hacking

El proveedor debe garantizar pruebas de Ethical Hacking sobre el desarrollo del sistema para lo cual deberá entregar los informes al banco, si se presentan hallazgos el proveedor debe emprender las acciones requeridas para hacer la remediación de estos. El banco podrá ejecutar pruebas adicionales de este tipo y si se presentan hallazgos estos también deben ser remediados por el proveedor.

# Arquitectura de integración

## Mecanismos de integración

El desarrollo del sistema debe contemplar esquemas de integración a través de archivos planos, Web Services o API’s, el proponente deberá dar la recomendación a utilizar con base en su experiencia.

# Soporte técnico

El proponente deberá describir en la propuesta cómo se va a prestar el soporte técnico para la solución o para los componentes que presente en su propuesta (software base, aplicaciones, servicios, componentes, módulos, etc.). Presentar el esquema de atención, plan de comunicaciones y el equipo de soporte, si es subcontratado, describa el nombre del subcontratista. Los tiempos de atención deben estar basados en Acuerdos de Niveles de servicio los cuales se deben especificar en la propuesta, el incumplimiento de estos dará lugar a la penalización en los costos que el Banco pague por el servicio contratado. Estos tiempos podrán ser sometidos a ajustes por las partes, según la criticidad y urgencia que se presente. Se deberá adjuntar el proceso de gestión de incidentes que se tiene implementado o que se implementará para la operación del sistema describiendo claramente los canales de atención, tiempos de respuesta y las actividades generales del proceso.

1. <https://en.wikipedia.org/wiki/Component_diagram> [↑](#footnote-ref-2)
2. <https://en.wikipedia.org/wiki/Deployment_diagram> [↑](#footnote-ref-3)