UNIVERSIDAD ARTURO PRAT

FACULTAD DE INGENIERÍA Y ARQUITECTURA|

INGENIERÍA CIVIL EN COMPUTACIÓN E INFORMÁTICA

**Logotipo

Descripción generada automáticamente**

**INFORME FUNDAMENTOS DE**

**LENGUAJE DE PROGRAMACIÓN:**

**FastAPI**

**INTEGRANTES:**

**Joel Reyes**

**PROFESOR:**

**Esteban Miranda**

IQUIQUE-CHILE

DICIEMBRE 2022

Contenido

[Introducción 3](#_Toc120088018)

[Descripción de la problemática 4](#_Toc120088019)

[Descripción de áreas a trabajar 4](#_Toc120088020)

[Análisis de proceso 5](#_Toc120088021)

[Descripción de objetivo 7](#_Toc120088022)

[Desarrollo 8](#_Toc120088023)

[Contenido del producto 8](#_Toc120088024)

[Preguntas implementadas 9](#_Toc120088025)

[Ventajas del lenguaje 10](#_Toc120088026)

[Desventajas del lenguaje 10](#_Toc120088027)

[Usos en la vida real 11](#_Toc120088028)

[Base de datos utilizada 11](#_Toc120088029)

[Características de la base de datos 12](#_Toc120088030)

[Conclusión 12](#_Toc120088031)

[Anexos 13](#_Toc120088032)

[Esquemas 13](#_Toc120088033)

[Bibliografía 13](#_Toc120088034)

# Introducción

# Descripción de la problemática

## Descripción de áreas a trabajar

## Análisis de proceso

## Descripción de objetivo

# Desarrollo

## Contenido del producto

## Lenguaje Usado

## Ventajas del lenguaje

* **Multiplataforma**, cuenta con soporte para diferentes sistemas operativos como Windows, Linux, Mac, Android, iOS.
* **Lenguaje de propósito general**, permite la programación de una alta gama de aplicaciones con distintas funcionalidades, como aplicaciones de servicio web, clientes de correo electrónico, aplicaciones de red.
* **Lenguaje interpretado**, el lenguaje interpretado puede modificar el código para hacer cualquier cosa en tiempo de ejecución, lo que agiliza el proceso a la hora de programar una aplicación en Ruby.
* **Meta programación**, Permite escribir programas que manipulan a otros programas, gracias a esto el programador puede ahorrar más código.
* **Open Source o código libre**, la comunidad puede generar sus propios módulos para el lenguaje (en el caso de Ruby mediante “gems”) y así extender su funcionalidad.
* **Versatilidad,** Ruby al contar con un Framework ofrece más opciones y esto hace que haya más variedad a la hora de empezar un proyecto, es decir, se puede elegir si se quiere trabajar con su Framework (Ruby On Rails) o con el propio Ruby.

## Desventajas del lenguaje

* **No tiene soporte técnico**, no hay compañías grandes soportando a Ruby, no tiene mucho apoyo al desarrollo del lenguaje, es mantenido solo por la comunidad de Ruby (no confundir con el soporte técnico de “Ruby on Rails”).
* **Lenguaje interpretado**, es de procesamiento lento y consume mucho más tiempo.
* **Alto uso de memoria,** para la ejecución del código almacena toda la base del código como representación en memoria.
* **No paralelismo (con Ruby),** Ruby es el intérprete por defecto, solamente podemos usar un Core a la vez, esto quiere decir que dos cosas no se pueden ejecutar al mismo tiempo.
* **Incompatibilidad,** algunas versiones de Ruby no son compatibles con su propio Framework (Ruby On Rails) lo que puede generar complicaciones en el desarrollo de aplicaciones.

## Usos en la vida real

## Framework Usado

## Ventajas del Framework

## Desventajas del Framework

## Usos en la vida real

## Base de datos utilizada

## Características de la base de datos

# Conclusión

# Anexos

## Esquemas

## Bibliografía