INFORME FUNDAMENTOS DE LENGUAJE DE PROGRAMACIÓN

RUBY

Icono

Descripción generada automáticamente con confianza media

**Profesor:**  
Esteban Miranda

**Integrantes:**

Gianfranco Pallero

Joel Reyes

Jorge Vera

Sebastián Carvajal

Noviembre de 2022

Contenido

[Introducción 3](#_Toc118734669)

[Descripción de la problemática 3](#_Toc118734670)

[Descripción de áreas a trabajar 3](#_Toc118734671)

[Descripción de objetivo 4](#_Toc118734672)

[Desarrollo 5](#_Toc118734673)

[Contenido del producto 5](#_Toc118734674)

[Ventajas del lenguaje 6](#_Toc118734675)

[Desventajas del lenguaje 6](#_Toc118734676)

[Usos en la vida real 6](#_Toc118734677)

[Base de datos utilizada 6](#_Toc118734678)

[Características de la base de datos 6](#_Toc118734679)

[Conclusión 6](#_Toc118734680)

[Anexos 6](#_Toc118734681)

[Esquemas 6](#_Toc118734682)

[Bibliografía 6](#_Toc118734683)

# Introducción

Para la correcta organización y funcionamiento de los programas computacionales es necesaria la consideración de la información pertinente a cada uno de sus procesos, datos y funciones. En antaño la programación se guiaba por un paradigma orientado a procesos debido al reciente descubrimiento del área computacional, el principal problema de tal paradigma era dar seguimiento al flujo del código debido a su constante iteración propia, como solución a esta problemática se dejaron de implementar funciones de regresión o saltos de línea, además del desarrollo del paradigma orientado a objetos que permite mayor control de datos.

El lenguaje Ruby es un claro ejemplo de lo dicho anteriormente, pues su desde su creación, su funcionamiento se ha desarrollado en base a la simplicidad y orientación a objetos, permitiendo así un alto entendimiento de las partes de sus códigos. Desde docencia en UNAP, se solicita para la asignatura “Fundamentos de lenguaje de programación” la realización de un programa en Ruby que sirva como sistema de gestión y evaluación de pacientes, dicho programa contará con los requisitos básicos de una conexión a base de datos, cuentas administradoras para los psicólogos y edición de encuestas con ponderación de preguntas, la interfaz visual es irrelevante para el motivo en cuestión.

# Descripción de la problemática

## Descripción de áreas a trabajar

La depresión clínica es un trastorno del estado anímico en el cual los sentimientos de tristeza, pérdida, ira y/o frustración interfieren con la vida diaria durante un largo periodo de tiempo, como tal, la depresión cuenta con distintas clasificaciones y síntomas que abarcan con mayor precisión la profundidad del caso.

Algunos de los síntomas generales de la depresión son:

* Anhedonia[[1]](#footnote-1)
* Ánimo irritable
* Insomnio
* Cambios de apetito
* Sentimientos continuos de desesperanza
* Dificultad para concentrarse
* Cansancio constante

La identificación de la depresión debe de ser realizada por profesionales en el área y tratada por los mismos, pero para su identificación existen distintos métodos y preguntas rutinarias que realizar, encuestas tales como el “Inventario de depresión de Beck”, “HRS-D” y “MADRS” son sólo las más conocidas en el área. Dentro de las preguntas que integran las distintas encuestas existe una jerarquía entre preguntas, pues dentro de la depresión existen identificadores claves como el insomnio o la anhedonia, estas preguntas y su valoración son administradas por el psicólogo en cuestión.

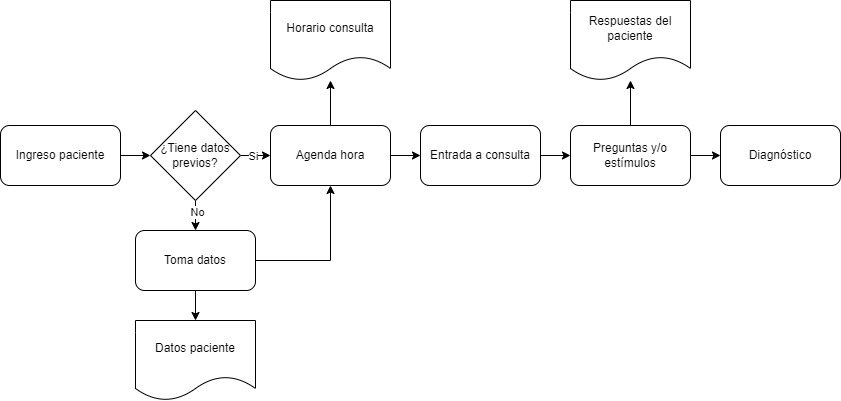
## Descripción de objetivo

Se busca el diseño e implementación de un sistema programado en Ruby que permita la automatización de los registros por parte de los psicólogos, además los psicólogos contarán con cuentas exclusivas para su uso que permitan la gestión de pacientes, desarrollo de encuestas y personalización de estas.

El sistema deberá contener una conexión a una base de datos anexa que será desarrollada en SQL, dicha base de datos será actualizada por el psicólogo sin la necesidad de programarla, se le otorgarán las funciones necesarias para añadir, eliminar y modificar sus distintas tablas[[2]](#footnote-2).

A continuación, el proceso general de diagnóstico

Ilustración 1: Proceso general de diagnóstico



El proceso abordado consta de 2 distintos subprocesos, el agendado y la consulta, para efectos prácticos se simplificará el proceso de consulta por la baja relevancia de este para el sistema.

Para el proceso de selección de hora se requieren solo dos pasos importantes, el registro de datos del paciente y la selección de una hora disponible, para el proceso de consulta existen distintos procesos de obtención de datos, pero en general se basan en estimulo y respuesta, esta respuesta se almacena para posterior evaluación y diagnóstico.

# Desarrollo

## Contenido del producto

El sistema realizado cuenta con 3 secciones relevantes, Menú de inicio, gestión de pacientes y gestión de encuestas, cada sección cumplirá una función distinta y tendrá distintas características que se detallaran a continuación.

**Menú**

El menú será lo primero que se visualizará al entrar a la aplicación, contará con 3 funciones.

* Ingresar: El psicólogo ingresará su correo y contraseña, dichos datos serán verificados por el sistema y permitirán el ingreso a un segundo menú de selección de funciones, tal menú dará la elección entre gestión de pacientes y gestión de encuestas.
* Registrarse: El nuevo usuario registrará sus datos en el sistema y se le permitirá el acceso.
* Salir: cerrar la aplicación.

**Gestión de pacientes**

La Gestión de pacientes se seleccionará en el puente entre ingresar y gestión, contará con 4 funciones.

* Ingreso de nuevo paciente: se registrarán los datos de un nuevo paciente.
* Realización de encuesta: se seleccionará el paciente al que se le desea realizar una encuesta, y se seleccionará la encuesta que se desea realizar, tras esto comenzará la encuesta y el psicólogo se encargará de anotar la valoración de las respuestas, en caso de desearlo podrá realizar observaciones que podrá observar luego.
* Información de pacientes: se mostrará toda la información del paciente seleccionado, incluyendo observaciones.
* Eliminar pacientes: Tras seleccionar un paciente, se eliminará de la base de datos (borrado inteligente).

**Gestión de encuestas**

La Gestión de encuestas se seleccionará en el puente entre ingresar y gestión, contará con 3 opciones

* Ingresar nueva encuesta: Se creará una encuesta desde 0, definiendo la cantidad de preguntas con un mínimo de 5 y máximo de 20, se ingresarán las preguntas y la posibilidad de valoración, y por último la ponderación de las preguntas.
* Modificar encuesta existente: Se permitirá modificar una encuesta existente.
* Eliminar encuesta: Se permitirá eliminar una encuesta existente.

Durante toda la aplicación, marcar “X” generará un retroceso al sector anterior sin mantener ningún cambio realizado.

## Ventajas del lenguaje

* **Multiplataforma**, significa que se puede ejecutar en diferentes sistemas operativos como Windows, Linux, Mac, Android, iOS.
* **Lenguaje de propósito general**, se puede hacer todo tipo de aplicaciones diferentes, desde 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**, el programa modifica su estructura usando más código, se puede escribir menos código y generar muchas funcionalidades.
* **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.
* **Lenguaje interpretado**, es de procesamiento lento y consume mucho más tiempo.
* **Alto uso de memoria,** utiliza toda la representación de nuestra base de código 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

El principal uso de Ruby es el de hacer desarrollo de aplicaciones web que son aplicaciones sustanciales de CPU, hay varias aplicaciones que usan el lenguaje de Ruby como los sitios web de comercio como Airbnb, de música como SoundCloud y de películas como Hulu, no obstante, el lenguaje Ruby puede utilizarse para hacer programas de escritorio y maneja un gran desarrollo de aplicaciones Back-End en el mundo.

Las páginas web creadas con Ruby funcionan como herramientas que tienen la capacidad de interactuar con grandes bases de datos y realizar operaciones complejas.

En general Ruby se utiliza para el desarrollo de cualquier tipo de sistema/aplicaciones debido a que cuenta con una alta versatilidad, ya que al ser multiplataforma, multipropósito y Open Source permite a nivel práctico una alta gama de posibilidades a sus usuarios, además de ello es un lenguaje enfocado en simplicidad y productividad que permite a sus usuarios la rápida familiarización con el mismo, por cuanto Ruby es, a nivel práctico, una de las mejores opciones de lenguaje a utilizar en proyectos. No obstante, Ruby no cuenta con una con soporte técnico al igual que varias de sus librerías, por lo que la mejor opción de uso es su Framework “Ruby on rails”, el cual aún cuenta con soporte técnico y está en constante actualización.

## Base de datos utilizada

PostgreSQL se caracteriza por ser un sistema gestor de base de datos relacionales, orientado principalmente a objetos, de código abierto y multiplataforma, además de ser extensible ya que se pueden añadir funcionalidades y escalable ya que este funciona bajo licencia libre y puede tratar con grandes cantidades de datos. Además, cuenta una gran comunidad de colaboradores en todo el mundo que mantienen esta base de datos actualmente vigente. Dos detalles que destacar de PostgreSQL es que posee data types (tipos de datos) avanzados y permite ejecutar optimizaciones de rendimiento avanzadas, que son características que por lo general solo se ven en sistemas de bases de datos comerciales, como por ejemplo SQL Server de Microsoft u Oracle.

## Características de la base de datos

Este sistema gestor de base de datos también provee de forma nativa soporte para ciertos tipos de datos como lo son: números de precisión arbitraria, texto de largo ilimitado, direcciones IP, etc... Además de permitir al usuario crear sus propios tipos de datos gracias a la infraestructura que posee.

Algunos de los principales usos de este sistema gestor de base de datos son:

* Almacenamiento de datos
* En servicios web de distintas empresas
* Para procesar datos y que otros servicios puedan conectarse también
* Aplicaciones de escritorio

Ruby al tener ciertas librerías sin mantenimiento por parte de la comunidad no cuenta con muchas opciones para conectarse a una base de datos, una de las que se encontró y que permitió el acceso fue PostgreSQL por lo cual decidimos utilizarla para el desarrollo de este programa.

# Conclusión

Con programa propuesto en el siguiente informe se pretende solucionar lo solicitado proporcionando un sistema que se enfoca en la usabilidad y comodidad del usuario, sin perjudicar la seguridad, integridad y privacidad del sistema y sus archivos, presentando así una solución factible y fiable para el problema presentado.

# Anexos

## Esquemas

## Bibliografía

<http://www.acmcb.es/files/425-2363-DOCUMENT/Perez-Pigdemont-42-20oct11.pdf>

<https://www.psi.uba.ar/academica/carrerasdegrado/psicologia/sitios_catedras/obligatorias/070_psicoterapias1/material/inventario_beck.pdf>

1. Incapacidad para experimentar placer. [↑](#footnote-ref-1)
2. Entidades almacenadas [↑](#footnote-ref-2)