



**Universidad Nacional Autónoma  
de México**



**Facultad de Ingeniería**

**División de Ingeniería Eléctrica**

**Inteligencia Artificial**

Proyecto Final (Primer avance)

*ínor* Computation

**Profesor:** Dr. Guillermo Gilberto Molero Castillo

**Grupo:** 3

**Alumno:** Iñaky Ordiales Caballero

**Semestre:** 2022 - 2

**Fecha de entrega:** 24 de marzo del 2022.

# *ínor* Computation

## **Idea general:**

Para la elaboración del proyecto final de la asignatura de Inteligencia Artificial se elaborará una aplicación web que permita a un usuario utilizar diferentes algoritmos para analizar un conjunto de datos. El objetivo del proyecto es demostrar los conocimientos de las diferentes clases de algoritmos de inteligencia artificial revisados a lo largo del curso. Esto de una manera práctica, aplicada y formal que pueda servirnos como un desarrollo que aporte a nuestro currículum profesional.

La forma en la que se aterrizó esto en este primer avance y acercamiento al proyecto fue mediante una aplicación con múltiples roles de usuarios y en la cual se seleccione un archivo local que contenga los datos en formatos csv y después de aplicarles algún algoritmo y procesamiento (reducción de dimensiones, clasificaciones, etiquetados, etc.) se puedan consultar y almacenar los resultados de manera local.

## **Requerimientos:**

### **Funcionales.**

- Se deben implementar los algoritmos que se marquen en la bitácora.
- El usuario podrá seleccionar cuál algoritmo utilizar.
- El usuario podrá escoger y cargar su conjunto de datos en formato csv.
- El usuario deberá poder añadir un análisis de los resultados del algoritmo que proceso los datos.
- El usuario podrá guardar el resultado del procesamiento en la computadora.

### **No funcionales.**

- La interfaz con el usuario debe ser gráfica e interactiva.
- Se debe de ver limpia y su utilización debe ser sencilla.
- Se utilizará un framework para la elaboración de la interfaz.

## Lenguajes, frameworks y herramientas:

### Lenguaje back-end: Python 3.



Para toda la lógica de programación de la aplicación, así como el totalidad de su funcionalidad se utilizará el conocido lenguaje de programación Python. Éste además de ser muy flexible y completo, tiene la gran ventaja de todas las bibliotecas existentes que nos permite utilizar. Como hemos visto a lo largo de las primeras prácticas de la asignatura, se tienen muy buenas bibliotecas para el análisis y representación de datos, pero también se cuenta con otras que tienen los algoritmos de inteligencia artificial ya implementados. Es por esto que Python se posiciona como uno de los lenguajes más populares y comunes a la hora de trabajar con inteligencia artificial.

### Framework de aplicación / front-end: Streamlit



Streamlit es una biblioteca escrita en Python y para Python la cual fue desarrollada en específico para los ingenieros de Machine Learning. Streamlit convierte los scripts de datos en aplicaciones web de manera muy sencilla e integrada en su totalidad con el lenguaje Python. Este marco de trabajo permite desarrollar una aplicación web con una pequeña curva de aprendizaje y sin necesitar tener gran experiencia en el desarrollo de front-end. Tiene las ventajas de que es código abierto, se refresca la vista de la aplicación automáticamente cada vez que se guarda el script, permite agregar widgets con facilidad y se despliega instantáneamente.

En un planteamiento inicial se vieron otras opciones de framework para utilizar, algunas de las que más se consideraron fueron PyWeblo (descartado por ser muy simple y poco gráfico), WebPy (descartado por su necesidad de trabajar con otros sistemas como PostgreSQL) y Flask. Éste último fue revisado más a profundidad ya que era más flexible y robusto que Streamlit. Sin embargo, debido a estas características también era más complejo y con una curva de aprendizaje mucho mayor. Al no haber trabajado en el desarrollo de aplicaciones web con anterioridad se decidió quedarnos con la opción que pareciera más sencilla y que se pudiera aprovechar al máximo para no sólo hacer un trabajo superficial. Streamlit ofrece todas las características necesarias para la creación e implementación de una aplicación web. Además, al estar diseñado específicamente para aplicaciones de datos, las herramientas de despliegue de datos están muy bien desarrolladas. Y nos ofrece templates para guiarnos.

## Otras herramientas front-end: HTML5 y CSS.



El mundo del internet, de las aplicaciones web y de los navegadores está moldeado mediante el lenguaje de marcado HTML y el estilo CSS. A pesar de que Streamlit nos ayudará a desplegar de manera sencilla nuestra aplicación, es necesario la utilización de estas dos herramientas para definir su forma y estilo. Dentro del mismo framework de Streamlit se tiene que trabajar con HTML para mejores resultados. Y si queremos ensario definir un diseño deberemos utilizar CSS para que se vea bien. Igualmente se ha trabajado poco con ambas herramientas, pero se conoce la sintaxis básica por lo que no deberían de presentar dificultades

## Javascript

Al no tener mucha experiencia con el desarrollo de aplicaciones web, no sé si se utilizará el lenguaje Javascript, el cual también es de lo más común en las páginas web. Por lo que más adelante en el proyecto se revisará su necesidad de uso.

## Estructura general y arquitectura:

La idea general es que el usuario pueda acceder a su perfil y dependiendo de su rol tenga diferentes opciones. Todos podrán analizar datos y generar resultados a través de la interacción mediante botones y widgets con la aplicación web.

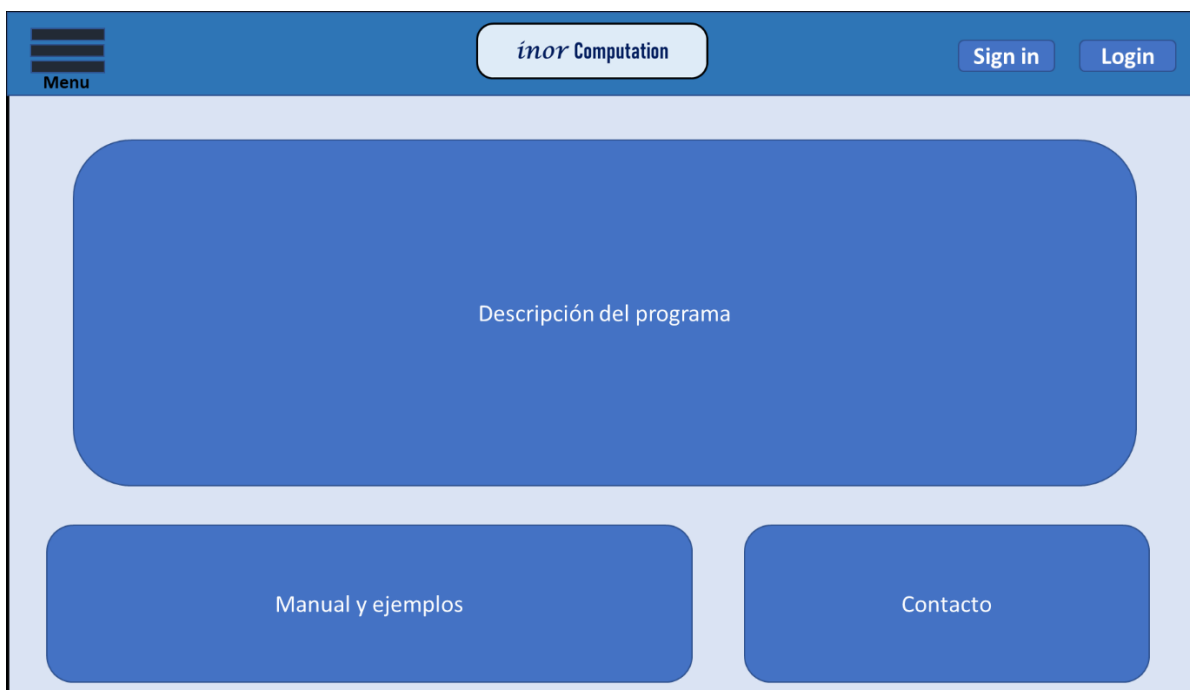
## Arquitectura de la aplicación web

La arquitectura de una aplicación web describe las interacciones entre aplicaciones, bases de datos y demás sistemas en la web. Se asegura de que el usuario pueda trabajar de manera simultánea en varias aplicaciones de manera correcta. La forma en que el usuario interactuará con la aplicación depende de la arquitectura con la que este elaborada el sistema. La arquitectura deberá incluir todos los subcomponentes de la aplicación, debiendo lidiar con la eficiencia, confianza, escalabilidad, seguridad y robustez.

La arquitectura seleccionada es la de **Single Page Architecture (SPA)**. Las SPA únicamente se componen de HTML, CSS y JavaScript. Su arquitectura se basa en cargar todo lo necesario en una única página, e ir actualizándola dinámicamente con los recursos necesarios y/o añadiéndolos si hicieran falta, a medida que el usuario interactúa con la aplicación o site. Como resultado, la experiencia del usuario es mucho más rápida y fluida. Las SPA's basan su arquitectura en las siguientes características: *chunking*, *controllers*, *templating*, *routing*, *real-time communication*, *local storage*. También se revisó la arquitectura de microservicios como opción si se complicara la SPA.

## Borradores del diseño:

### Portada.



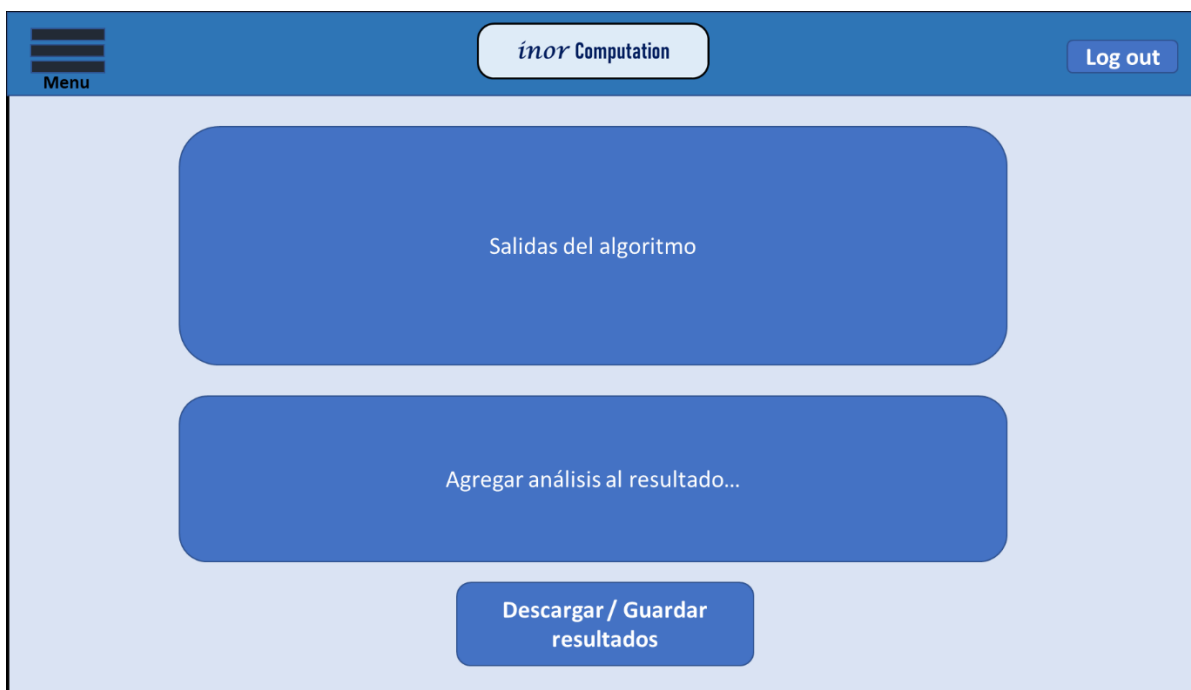
### Menú analista.



### Menú administrador / gerente.



### Presentación resultados.



**Siguientes pasos:**

Para continuar con el desarrollo del proyecto los siguientes pasos a seguir serían el empezar a familiarizarnos con el framework y las herramientas de front-end para que junto con las demás herramientas y lenguajes se pudiera crear una primera versión semi funcional de la aplicación. Como ya se ha tenido experiencia con la programación en Python creemos que la implementación de los algoritmos de inteligencia artificial se hará con relativa facilidad, pero la construcción de toda la parte gráfica es lo que puede tomarnos más tiempo.

En las siguientes semanas se debe comenzar a crear y usar los diseños casi finales para la aplicación, y manejar de forma general la interacción de los diferentes elementos de la arquitectura y el diseño. El objetivo del siguiente avance sería tener la funcionalidad gráfica de la página y los primeros algoritmos revisados en el curso ya funcionando. Esta será la parte más pesada del proyecto ya que consiste en comenzar a construir y crear para poder modificar después. Con las prácticas realizadas ya tenemos la sensación de cómo se debe trabajar y manejar con los algoritmos, desde su selección de características y estandarización de los datos hasta el análisis de los resultados que nos arrojan. Estos conocimientos se verán aplicados para el siguiente avance.

The logo consists of the text "ínor Computation" in a dark blue, sans-serif font. The word "ínor" is in a lighter blue color, while "Computation" is in a darker blue. The text is centered within a light blue rounded rectangle with a dark blue border.

*ínor* Computation