

<b>Nombre del autor</b>	Ivan David Rey Rueda
<b>Fecha creación</b>	30/01/2021

<b>Propósito</b>	Desarrollar una aplicación de escritorio para Aprendizaje de Máquina supervisado en el sistema operativo Windows 10 x64, que permita a los usuarios la selección del modelo, hiperparámetros, entrenamiento y predicción con base al conjunto de datos suministrado.
<b>Alcance</b>	Desarrollar una aplicación de escritorio para Aprendizaje de Máquina supervisado para el uso de estudiantes en sus primeros años de formación profesional, en el transcurso de ocho meses desde el mes de Agosto del año 2020
<b>Características del usuario</b>	<ol style="list-style-type: none"> <li>1. Estudiante universitario de Ingeniería de software, sistemas, informática o Computación</li> <li>2. Estudiante en sus primeros dos años de formación académica</li> <li>3. Carece de las habilidades necesarias para generar sus propios modelos de aprendizaje de máquina en código fuente</li> <li>4. Requiere aprender la teoría para la creación de modelos de aprendizaje de máquina supervisado</li> <li>5. No tiene los recursos económicos para invertir en aplicaciones web para generar automáticamente modelos de aprendizaje de máquina</li> </ol>
<b>Entorno operativo</b>	Windows 10 Home 64 bits. Versión 1903
<b>Requerimientos mínimos del sistema</b>	RAM: 4Gb Procesador: Intel o AMD a 1.6Ghz con dos núcleos Espacio libre disco: 4Gb SO: Windows 10 64 bits Home 1903

Requerimientos funcionales			
Código	Descripción	Prioridad	Requerimientos asociados
RF-01	La aplicación debe permitir al usuario la selección entre entrenamiento paso a paso o automático	Alta	
RF-02	La aplicación posibilitará al usuario elegir entre Clasificación, Regresión y Agrupamiento, cuando el	Media	RF-01

	modelo sea seleccionado paso a paso		
RF-03	La aplicación debe admitir seleccionar conjuntos de datos con extensión txt, csv o tsv	Baja	
RF-04	La aplicación debe permitir seleccionar entre diferentes métodos para la selección de hiperparámetros, cuando el modelo sea seleccionado paso a paso	Media	RF-02
RF-05	La aplicación debe ejecutar los algoritmos únicamente en el equipo del usuario	Alta	
RF-06	La aplicación debe mostrar botones de ayuda con la documentación apropiada en cada opción que el usuario pueda seleccionar	Media	
RF-07	La aplicación deberá generar los resultados finales en un archivo de marcado que se guardará en el dispositivo de almacenamiento secundario del usuario	Baja	
RF-08	La aplicación deberá mostrar información en tiempo real del estado actual cuando se realice el entrenamiento del modelo	Baja	
RF-09	La aplicación deberá limitar al usuario a ejecutar una única instancia al mismo tiempo	Baja	

RF-10	La aplicación deberá permitir al usuario regresar a un formulario anterior a excepción cuando se realiza el entrenamiento del modelo	Baja	
RF-11	La aplicación deberá permitir al usuario cancelar el proceso de entrenamiento en cualquier momento	Media	
RF-12	Se debe redirigir al usuario al formulario inicial cuando la opción “inicio” sea seleccionada en el formulario de resultados	Media	
RF-13	Se debe cerrar la aplicación cuando la opción “cancelar entrenamiento” con respuesta afirmativa sea seleccionada durante el entrenamiento del modelo	Media	RF-11
RF-14	Se debe mostrar una ventana emergente cuando la opción “cancelar entrenamiento” sea seleccionada cuando el entrenamiento del modelo se encuentra en curso	Media	RF-13


Requerimientos no funcionales			
Código	Descripción	Prioridad	Requerimientos asociados
RNF-01	La aplicación debe de ser desarrollada	Alta	

	principalmente para Windows de 64bits		
RNF-02	El idioma de la aplicación debe ser Español (latinoamérica)	Bajo	
RNF-03	La aplicación deberá contar con una interfaz de usuario sencilla, intuitiva y cómoda a la vista	Medio	
RNF-04	La interfaz de usuario debe diseñarse con base a las normas ISO/IEC 11581-10:2010 e ISO 9241-210:2019	Medio	
RNF-05	El diseño de la interfaz de usuario se debe realizar con el editor Figma	Media	RNF-02, RNF-03, RNF-04
RNF-06	El implementación de la interfaz de usuario debe ser en el framework QT designer	Media	RNF-04
RNF-07	Se debe garantizar la integridad del conjunto de datos suministrado por el usuario	Media	
RNF-08	El back-end de la aplicación debe ser desarrollado con Python 3.7.0	Alta	
RNF-09	El back-end de la aplicación debe ser desarrollado siguiendo los estándares PEP, PEP20, PEP257, PEP3131, PEP 484 y PEP 526.	Alta	RNF-08
RNF-10	La metodología de desarrollo ágil para el desarrollo del proyecto debe ser Scrumban	Media	

RNF-11	El ejecutable de la aplicación debe generarse por medio del módulo Pyinstaller	Media	
RNF-12	La fecha límite de entrega es 01/03/2021	Alta	

Control de Cambios			
Versión	Fecha Aprobación	Descripción del Cambio	Participantes del Cambio
03	4/02/2021	Cambios en requerimientos del sistema, funcionales y no funcionales	Iván David Rey Rueda

#### Aprobación de requerimientos



Firma del director de proyecto

**Nombre y Apellidos:** Nydia Paola Rondón Villarreal

**Cargo:** Docente Directora Proyecto