

Pontificia Universidad Javeriana Cali  
Facultad de Ingeniería y Ciencias.  
Ingeniería de Sistemas y Computación.  
Anteproyecto de Grado.

Sistema para Ayudar en la Elección de una Carrera Universitaria  
en la Pontificia Universidad Javeriana-Cali, a Jóvenes Próximos a  
Ingresar

Sebastian Toro Franco  
Cristian Steven Osorio Paz

Director: Dr. Gerardo Mauricio Sarria

22 de Febrero de 2021





Santiago de Cali, 22 de Febrero de 2021.

Señores

**Pontificia Universidad Javeriana Cali.**

Dr. Gerardo Mauricio Sarria

Director Carrera de Ingeniería de Sistemas y Computación.

Cali.

Cordial Saludo.

Por medio de la presente me permito informarle que los estudiantes de Ingeniería de Sistemas y Computación Sebastian Toro Franco (cod: 8931567) y Sebastian Toro Franco(cod: 8931567) trabajan bajo mi dirección en el proyecto de grado titulado “Sistema para Ayudar en la Elección de una Carrera Universitaria en la Pontificia Universidad Javeriana-Cali, a Jóvenes Próximos a Ingresar”.

Atentamente,

---

Dr. Gerardo Mauricio Sarria

Santiago de Cali, 22 de Febrero de 2021.

Señores

**Pontificia Universidad Javeriana Cali.**

Dr. Gerardo Mauricio Sarria

Director Carrera de Ingeniería de Sistemas y Computación.

Cali.

Cordial Saludo.

Nos permitimos presentar a su consideración el anteproyecto de grado titulado “Sistema para Ayudar en la Elección de una Carrera Universitaria en la Pontificia Universidad Javeriana-Cali, a Jóvenes Próximos a Ingresar” con el fin de cumplir con los requisitos exigidos por la Universidad para llevar a cabo el proyecto de grado y posteriormente optar al título de Ingeniero de Sistemas y Computación.

Al firmar aquí, damos fe que entendemos y conocemos las directrices para la presentación de trabajos de grado de la Facultad de Ingeniería aprobadas el 26 de Noviembre de 2009, donde se establecen los plazos y normas para el desarrollo del anteproyecto y del trabajo de grado.

Atentamente,

---

Sebastian Toro Franco  
Código: 8931567

---

Cristian Steven Osorio Paz  
Código: 8931683

# Resumen

La educación es uno de los factores constituyentes y fundamentales para el avance y progreso de la sociedad. Es por esto que uno de los problemas que afronta un país como Colombia, es el hecho que el porcentaje de estudiantes bachiller que ingresan a la universidad sea bajo y la tasa de deserción alta en las carreras universitarias por motivos de mala elección y/o no saber que elegir. Según el Ministerio de Educación Nacional y su Sistema para la Prevención y Análisis de la Deserción en las Instituciones de Educación Superior, cerca de 504 mil estudiantes de grado 11 reportados en el año 2013, alrededor de 174 mil iniciaron un proceso de tránsito a educación superior inmediato [1] y de cada cien estudiantes que ingresaron a una institución educativa superior, cerca de la mitad no logra graduarse de sus respectiva carrera universitaria [2]. Esto nos lleva a preguntarnos, desde nuestra área de conocimiento y acción, ¿cómo desarrollar una aplicación web para ayudar a la elección de una carrera universitaria?

Este proyecto de grado busca la realización de una solución que permita apoyar al proceso de elección de una carrera universitaria de los jóvenes que acabaron su proceso escolar medio y piensan comenzar un estudio superior en ingeniería. El sistema de Información presentará datos de primera mano de las carreras universitarias, obtenida de personas estudiando en ellas en forma de comentarios. Así mismo, ofrecerá información acerca del mercado laboral colombiano, predicciones y estadísticas de los datos pertinentes al mercado laboral colombiano y el desempeño de la carrera universitaria en este. Sumado a lo anterior, se incluirá la relación de las aptitudes del joven usuario con ciertas carreras universitarias con base en su desempeño en la educación media y la correlación de los desempeños obtenidos de estudiantes en las carreras dentro de la universidad, encontrando afinidades del usuario con la carrera universitaria, permitiéndole a este poder tomar una decisión de la mejor manera ya que la información es asequible y específica a cada usuario.

**Palabras Clave:** Elección vocacional, estadística inferencial, mercado laboral, aptitudes profesionales.



# Índice general

<b>1. Descripción del Problema</b>	<b>11</b>
1.1. Planteamiento del Problema . . . . .	11
1.1.1. Formulación . . . . .	12
1.1.2. Sistematización . . . . .	12
1.2. Objetivos . . . . .	12
1.2.1. Objetivo General . . . . .	12
1.2.2. Objetivos Específicos . . . . .	13
1.3. Justificación . . . . .	13
1.4. Delimitaciones y Alcances . . . . .	14
1.4.1. Entregables . . . . .	15
<b>2. Desarrollo del Proyecto</b>	<b>17</b>
2.1. Marco de Referencia . . . . .	17
2.1.1. Áreas Temáticas . . . . .	17
2.1.2. Marco Teórico . . . . .	17
2.1.3. Trabajos Relacionados . . . . .	24
2.2. Metodología de Desarrollo . . . . .	25
2.2.1. Tipo de Estudio . . . . .	26
2.2.2. Actividades . . . . .	27
2.3. Cronograma . . . . .	29
2.4. Recursos . . . . .	29
2.4.1. Humanos . . . . .	29
2.4.2. Técnicos . . . . .	30
2.4.3. Presupuesto . . . . .	30
<b>3. Modelo Estadístico</b>	<b>33</b>
3.1. Introducción . . . . .	33
3.2. Estructura de un modelo estadístico Predictivo . . . . .	34
3.3. Componentes del modelo . . . . .	35
3.3.1. Nodos fuente . . . . .	35
3.3.2. Nodos de transformación de los datos . . . . .	35
3.3.3. Nodos de modelamiento . . . . .	38
3.3.4. Nodos contenedores de modelo . . . . .	39
3.3.5. Nodos de exportación . . . . .	40
3.4. Predicciones del Modelo . . . . .	41
3.5. Análisis de las Predicciones . . . . .	41

---

<b>4. Obtención y Análisis de Requisitos</b>	<b>43</b>
4.1. Introducción . . . . .	43
4.2. Descripción general del sistema . . . . .	43
4.3. Diagrama de casos de uso . . . . .	45
4.4. Requerimientos . . . . .	46
4.4.1. Requerimientos Funcionales . . . . .	46
4.4.2. Requerimientos No Funcionales . . . . .	47
<b>5. Diseño del Software</b>	<b>51</b>
5.1. Introducción . . . . .	51
5.2. Arquitectura del Aplicativo . . . . .	51
5.3. Modelo de datos . . . . .	55
<b>6. Tecnologías Involucradas en el Desarrollo del Sistema</b>	<b>57</b>
6.1. Introducción . . . . .	57
6.2. Frontend . . . . .	57
6.3. Backend . . . . .	59
6.4. Business Inteligence . . . . .	61
6.5. Bases de Datos . . . . .	62
6.6. Modelo Estadístico . . . . .	63
6.7. Generador de Información Aleatoria . . . . .	64
<b>7. Implementación del Software</b>	<b>65</b>
7.1. Prototipado de pantallas . . . . .	65
7.2. Modelo estadístico . . . . .	72
7.2.1. Análisis e integración de los resultados en el sistema . . . . .	75
7.3. Desarrollo del aplicativo . . . . .	82
7.4. Dificultades en la Implementación . . . . .	85
<b>8. Evaluación del Software</b>	<b>87</b>
8.1. Plan de Evaluación . . . . .	87
8.2. Pruebas . . . . .	88
8.2.1. Pruebas funcionales . . . . .	88
8.2.2. Pruebas de aceptación . . . . .	93
8.3. Resultados . . . . .	94
8.3.1. FAQ . . . . .	94
<b>9. Conclusiones</b>	<b>97</b>
<b>10.Trabajo Futuro</b>	<b>99</b>
<b>Bibliografía</b>	<b>101</b>

# Introducción

La orientación vocacional es un medio que permite explorar y caracterizar las creencias, emociones, pensamientos, preferencias y actividades que realizan los estudiantes durante su formación académica, personal, familiar y entre otras. Razón por la cual, se puede ayudar a definir la elección profesional del estudiante a través del asesoramiento o utilización de herramientas como las pruebas estandarizadas, donde se destacan las preferencias del individuo; también, permite explorar alternativas que ayuden a reflexionar al respecto de estas cuestiones y buscar posibilidades para reunir los recursos personales, económicos e intelectuales necesarios para lograr una elección de carrera universitaria adecuada [3].

Sin embargo, estas técnicas enfocadas al acompañamiento en la elección de orientación profesional, también llamadas pruebas psicotécnicas, no son asequibles para todos económicamente, además de no ser utilizables en todas las personas debido a la particularidad concerniente a cada una de ellas. Estas técnicas no presentan todos los aspectos que debería tener en cuenta una persona tanto de su entorno como de sí mismo para tomar una buena decisión, como lo son el mercado laboral y/o las aptitudes de la persona para una carrera universitaria [4]. Esto debido a que gran parte de las oportunidades de realizar estas pruebas, se enfocan en prácticas grupales no personalizadas, lo que no permite el esclarecimiento de sus necesidades o preferencias vocacionales [3].

Lo anterior invita al desarrollo de una herramienta tecnológica que sirva de apoyo para la elección vocacional o profesional, siendo este el objetivo de este trabajo de grado. Esta herramienta traerá a colación este tipo de ayudas, que auxilian al estudiante para que este conozca mejor el entorno de las carreras universitarias, mercado laboral y demás, tanto en el presente como a futuro por medio de predicciones estadísticas en los temas pertinentes. Sumado a esto, representa una ventana al auto conocimiento para una buena elección por medio de las aptitudes de la persona, logrando así que el estudiante tenga todo el conocimiento posible sobre las carreras para la toma de decisiones y la certeza de que hay una carrera para él, en caso de que no tenga contemplada una.



# Descripción del Problema

---

## 1.1. Planteamiento del Problema

La educación es uno de los factores fundamentales para el avance y progreso de la sociedad y las personas, esto se puede evidenciar en mejores niveles de bienestar social y crecimiento económico. Esta importancia, se manifiesta aún más en el mundo actual, un mundo en constante transformación y con un incremento de las necesidades de la humanidad por factores como el crecimiento poblacional, lo que provoca una mayor necesidad de recursos; por la ciencia misma y sus distintas aplicaciones en sus diferentes campos.

Uno de los problemas que enfrenta la educación y la economía en un país como Colombia, se puede caracterizar de manera general, como una inadecuada selección de una carrera universitaria de parte de los jóvenes cuando terminan sus estudios intermedios, es decir, el bachillerato [5]; según el Ministerio de Educación Nacional (MEN) cerca de 504 mil estudiantes de grado 11 reportados en el año 2013, alrededor de 174 mil iniciaron un proceso de tránsito a educación superior inmediato al término de sus estudios en lo que quedaba del año 2013 y parte del 2014 [1], sumado a esto según el MEN y su Sistema para la Prevención y Análisis de la Deserción en las Instituciones de Educación Superior (SPADIES), de cada cien estudiantes que ingresan a una institución educativa de nivel superior, cerca de la mitad no logra culminar su proceso académico y graduarse de sus respectivas carreras universitarias [2].

Las principales causas de una mala o ausente elección vocacional según, Luz Adriana Pineda Barón [6], son varias, se pueden encontrar clasificaciones de estas en los ámbitos personales, socioeconómicos, académicos y motivacionales; dentro de los factores personales podemos encontrar situaciones como la edad o género del individuo; en el ámbito socioeconómico, se observan los ingresos económicos, estrato, ocupación y tamaño del núcleo familiar, el hecho de que el estudiante este trabajando a la hora de ingresar a estudiar, la cantidad de hermanos en la familia, carecer de vivienda propia, un ambiente poco motivante socialmente, entre otros; en la parte académica podemos ver énfasis en las pruebas de estado presentadas y el colegio de donde provienen, sumado a esto la motivación salarial es uno de los factores para la elección de carrera universitaria, las expectativas salariales en relación con el mercado laboral influyen para la elección de carrera; además, a estos factores se pueden encontrar en los jóvenes situaciones como el bajo conocimiento de las habilidades que se presentan a la hora de elegir una carrera, falta de información concretamente asertiva de los distintos programas e instituciones educativas, entre otras.

Mientras tanto, los principales efectos de la mala elección vocacional se observan en distintos factores como por ejemplo en procesos sociales, económicos y políticos de desarrollo de un país, esto debido a que el capital humano proyectado para el desarrollo no se adapta a las necesidades del mismo, resultando en una fuerza de trabajo laboral poco calificada para suplir los objetivos de ciertos sectores del mercado laboral, creando sobre costos sociales y privados[5]. Sumado a esto, la no elección de una carrera o vocación causa un incremento en la desigualdad social, donde quienes están fuera del sistema educativo y profesional forman parte de grupos excluidos socialmente, grupos que tienden a formar parte de actividades delictivas y violentas, la falta de educación también puede terminar en que las personas no tengan una posibilidad tan alta de salir de la pobreza o no tener aptitudes para obtener un trabajo estable y bien remunerado [7], cabe destacar que estos son efectos comunes que se pueden ver en la deserción estudiantil, uno de los principales efectos de la mala elección de carrera. Por estas razones, en el presente trabajo se desarrollará una aplicación web que apoye el proceso de elección de la carrera universitaria en la Pontificia Universidad Javeriana Cali en jóvenes próximos a entrar.

### 1.1.1. Formulación

¿Cómo desarrollar una aplicación que apoye el proceso de elección de carrera universitaria en la Pontificia Universidad Javeriana Cali?

### 1.1.2. Sistematización

- ¿Que documentación sobre los temas a trabajar (técnicas predictivas, herramientas de analítica de datos, mercado laboral, orientación vocacional, entre otros), representan una necesidad de análisis y uso en el sistema?
- ¿Cómo utilizar estas técnicas predictivas en correlacion con las notas del próximo estudiante y las de estudiantes en la Pontificia Universidad Javeriana Cali y el análisis del mercado laboral?
- ¿Qué requisitos presentaría el sistema a desarrollar a nivel funcional, de diseño y no funcional?
- ¿Qué diseño debería tener el sistema en cuanto a arquitectura y componentes según sus requisitos y funcionalidades?
- ¿Qué tipos y/o casos de prueba se deben realizar para evaluar la correctitud y funcionalidad completa del sistema?

## 1.2. Objetivos

### 1.2.1. Objetivo General

Desarrollar una aplicación web que apoye a los jóvenes próximos a empezar estudios de nivel superior en el proceso de elección de carrera universitaria en la Pontificia Universidad Javeriana

Cali.

### 1.2.2. Objetivos Específicos

- Analizar y escoger herramientas que empleen técnicas predictivas predefinidas y modelos de analítica de datos (regresión lineal, campana de Gauss, red de Bayes, distribución normal, entre otros).
- Identificar y depurar las notas de estudiantes bachiller próximos a ingresar a la universidad, las notas de educación media de estudiantes universitarios y los datos del mercado laboral, para el correcto funcionamiento de los modelos predictivos, a través de una tecnología que permita su utilización.
- Realizar una arquitectura de software pertinente que permita su implementación y cumplimiento de los requisitos.
- Diseñar los casos de prueba necesarios, basados en las funcionalidades del sistema, incluyendo pruebas funcionales, de usabilidad y rendimiento.
- Evaluar y verificar a través de los casos de prueba y encuestas realizadas a estudiantes de grado once próximos a entrar a la universidad el cumplimiento del objetivo del sistema favoreciendo su usabilidad.

## 1.3. Justificación

Este sistema permitirá a los estudiantes obtener recomendaciones e información puntual y asertiva acerca de distintas carreras universitarias de la mano de estudiantes activos en ellas. Donde se presentará un listado de referencias y comentarios previamente analizados, obtenidos de encuestas realizadas a estudiantes activos de las carreras de manera anónima, sumado a esto, se mostrará el panorama del mercado laboral actual y un conjunto de predicciones estadísticas sobre este mismo para que en caso de que el estudiante decida basarse en la demanda laboral del país, este tenga los indicadores e información para tomar la decisión, obteniendo esta información de la mano del Observatorio Laboral y Ocupacional Colombiano [8], representando toda esta información y su análisis en un mismo sitio de fácil acceso y con una ambientación amigable al usuario. El sistema en cuestión presentará la posibilidad de anexar las notas escolares obtenidas en su educación media, para así, obtener datos y realizar su posterior análisis que se puedan convertir en información al medir la aptitud del joven con las distintas carreras y realizar una recomendación sobre datos históricos en la carrera, o en su defecto poder encontrar aspectos a mejorar y que así logre estar en sincronía con la carrera a buscar; todo lo anterior para poder guiar al joven de la actualidad en su proceso de elección de carrera de una manera consistente y asertiva.

El desarrollo de una herramienta por el estilo puede llegar a tener una gran trascendencia en la sociedad, más aún a nivel educativo, llegando a combatir la desinformación acerca de lo que significa

y acarrea estudiar una carrera de educación superior. Así mismo, permite que la demanda laboral del país se pueda ver suplida en gran medida, debido a que los estudiantes tendrán en cuenta esta información en su decisión, donde gracias al autoconocimiento y análisis de la información recibida, se puede saber de manera cuantitativa cómo está el perfil del estudiante en cada carrera. Esto permite que incluso las personas opten por una carrera de manera definitiva, ya que encuentran que realmente hay algo que pueden estudiar para mejorar su futuro, disminuyendo la cantidad de personas que no entran a estudiar por falta de motivación. Cabe destacar el gran apoyo que la guía vocacional de cualquier tipo significa para la lucha contra la deserción universitaria, si las personas son conscientes de sí mismos y las opciones que tienen, así sus decisiones pueden ser más definitivas.

El sistema descrito presentará una guía concisa y justificada al estudiante debido a que permitirá encontrar información real y de fácil acceso. Esta información tanto de las distintas carreras, las actividades y particularidades en ellas como el mercado laboral y demás opiniones se encuentran en un mismo sitio, lo que alienta a llegar a ella de manera más rápida y oportuna para la toma de una decisión. Permitirá al estudiante tener un conocimiento acerca de sí mismo en el ámbito académico, donde el estudiante encontrará proficiencia en ciertas y distintas áreas y su correlación con las necesarias o dictaminadas para ejercer las distintas carreras universitarias. Esto conlleva a realizar una reflexión más profunda sobre sus acciones y cómo marcará su futuro el tomar una decisión tan importante como la elección de carrera universitaria, también el estudiante se motivará al obtener una herramienta que le permita observar que, si existe una carrera para él, pues sólo precisaba de una guía que lo ayudará a encontrar esta.

#### 1.4. Delimitaciones y Alcances

La solución que se propone al problema es una aplicación que ayude a mejorar la situación sobre la mala elección vocacional universitaria en la Pontificia Universidad Javeriana Cali generada por desinformación general sobre la naturaleza de las carreras que se ofrecen, las alternativas financieras que hay o en general por una falta de motivación profesional y de autoconocimiento. Esta aplicación tiene unos objetivos específicos para poder garantizar que se obtendrá un beneficio al ponerla en práctica. Las delimitaciones y alcances, así como objetivos específicos son como siguen:

- Se proporcionará al estudiante de bachiller información sobre distintas carrera profesionales, donde se hará un análisis sobre la situación actual del país a nivel económico y laboral en el mercado y social.
- El análisis sobre la situación actual del país a nivel económico y social, será realizado a través de un modelo predictivo construido a partir de modelos estadísticos regresivos, en una herramienta que permita su desarrollo.
- El alcance de dicho análisis, será de carreras universitarias que estén contenidas dentro de la facultad de Ingenierías en la Pontificia Universidad Javeriana Cali. La razón es por la cercanía y familiaridad que se tiene con estas carreras respecto, a lo que ofrecen conceptual

y experimentalmente dentro del campus universitario o en el mundo laboral Colombiano e internacional.

- El tipo de estudiantes pertenecientes al conjunto de datos anónimo de la Pontificia Universidad Javeriana Cali, comprende estudiantes activos y estudiantes que han salido de cada carrera comprendida en la facultad de ingeniería y ciencias.
- En el transcurso del desarrollo del proyecto de grado se encuentra contenida la realización de encuestas para la obtención de comentarios acerca de cada carrera de manera anónima de parte de estudiantes activos en la Pontificia Universidad Javeriana Cali y una encuesta anonimizada a estudiantes de bachillerato a los cuales se les presentará la aplicación para evaluar el cumplimiento de su propósito.
- La aplicación de modelos estadísticos en el sistema está limitada a escogencia de estos y utilización en la herramienta de creación de modelos predictivos, no a la creación de un modelo estadístico.
- La seguridad y anonimización de los datos en el sistema, será un requisito del desarrollo del proyecto de grado, basados en la normativa colombiana de protección de datos personales (Ley 1581) [9] y normativas dadas por el Centro de Servicios Informático de la Pontificia Universidad Javeriana Cali para el uso de los datos del cuerpo estudiantil.
- La aplicación estará disponible y tomara forma de una aplicación web para obtener una cobertura de uso mayor con base a la tecnología presentada por los usuarios.
- La gestión, obtención de la autorización y normativa de uso de parte del Centro de Servicios Informáticos de la Pontificia Universidad Javeriana Cali para el uso de datos anonimizados de estudiantes del cuerpo estudiantil, sigue en trámite y es parte inicial del trabajo de grado mismo.

#### 1.4.1. Entregables

Para el momento de finalización de este proyecto se espera entregar:

- Aplicación web de código abierto, con información para usuarios a punto de ingresar a la universidad acerca del mercado laboral al que ingresan y recomendaciones con base en lo anteriormente mencionado.
- Desarrollo de un modelo predictivo que permite visualizar que tan apto es un estudiante en una carrera dentro de la facultad de ingeniería de la Pontificia Universidad Javeriana Cali.
- Documentación pertinente de la aplicación soportada por los estándares IEEE para su uso de manera abierta.



# Desarrollo del Proyecto

---

## 2.1. Marco de Referencia

### 2.1.1. Áreas Temáticas

- CCS - Information systems-Information retrieval-Retrieval tasks and goals-Recommender systems
- CCS - Information systems-Information systems applications-Collaborative and social computing systems and tools-Open source software
- CCS - Software and its engineering-Software organization and properties-Software system structures -Software architectures
- CCS - Software and its engineering-Software creation and management-Designing software
- CCS - Software and its engineering-Software creation and management-Software development techniques-Software prototyping

### 2.1.2. Marco Teórico

En esta sección se busca describir las técnicas de selección y análisis de datos de la mano de modelos estadísticos que serán usados durante todo el proceso teniendo como base los datos de notas de estudiantes en las carreras universitarias en sus educación media, correspondiente al rendimiento de estudiantes previos en carreras de la Facultad de Ciencias e Ingeniería. La evaluación de tendencias del mercado laboral colombiano, información particular perteneciente a cada carrera y el perfilamiento del usuario en una carrera pertinente de acuerdo con sus aptitudes, entre otros.

#### 2.1.2.1. Ingeniería de software

Esta área amplía los conceptos de ciertos elementos, técnicas y acercamientos importantes para el desarrollo de software; los cuales son importantes para plantear el horizonte de la solución a realizar y la manera para efectuar un desarrollo pertinente de acuerdo a los apéndices del Software Engineering Body of Knowledge(SWEBOK) [10], el cual describe los conocimientos, practicas y estrategias generalmente aceptados para la ingeniería de software.

- **Requerimiento:** un requerimiento de software es una propiedad que se debe exhibir en un sistema para así tener una completa funcionalidad o desarrollo. Estos requerimientos deben ser especificados, para obtener una completa interpretación del alcance, validaciones y maneras de verificación. Se deben incluir sus atributos y actualizarse con el avance del proyecto en caso de ser necesario. Poseen una clasificación a la que pertenecen, así como una prioridad, método de verificación o plan de aceptación e historial de cambio, guiados por un identificador único para poder diferenciarlos. Una propiedad importante de los requisitos es que deben de ser verificables como una funcionalidad individual en el sistema, o a nivel externo en el caso de los requerimientos no funcionales. Estos requerimientos se dividen de manera básica en dos categorías:
  - **Requerimientos funcionales:** describen las funciones que debe ejecutar el software, se conocen como características o capacidades del sistema. Se puede describir como aquel requerimiento del que se puede realizar una verificación de su comportamiento con un número finito de pasos. Para uso particular en este trabajo de grado, se decidió extender la clasificación a particularidades o componentes del sistema, como lo serían back-end, front-end y bases de datos.
  - **Requerimientos no funcionales:** actúan como una restricción o característica de calidad a tener en el sistema.
- **Elicitación de requisitos:** proceso que se centra en los orígenes y técnicas de recolección de los requisitos del software, en el cual se establecen los distintos grupos de interés y elementos relevantes del sistema a tener en cuenta para el desarrollo disponiendo de la definición de fuentes de requisitos y técnicas para el levantamiento correcto de estos.
  - **Fuente de requisitos:** las fuentes se obtienen con el análisis y entendimiento de las distintas relaciones en los elementos del sistema, además de los distintos grupos de interés del software a desarrollar y las metas que este debe cumplir como sistema funcional. Son fuentes de requisitos para este trabajo entonces, la meta o funcionalidad final del sistema como herramienta para el apoyo en la elección de una carrera universitaria en jóvenes y los distintos grupos de interés reconocidos como los próximos usuarios del sistema, así como el ambiente operacional en el que se desenvolverá el sistema.
  - **Técnicas de levantamiento de requisitos:** se entienden como técnicas en las cuales el ingeniero o desarrollador formula objetivos por medio de la obtención de información particular de las fuentes de requisitos. Las principales técnicas de levantamiento de requisitos son: las entrevistas con los distintos grupos de interés para determinar funcionalidades, el prototipado y verificación del sistema, visualización de escenarios o contextos por medio del uso de diagramas de casos de uso del sistema y el desarrollo de historias de usuario que se basa en los distintos puntos de vista que se pueden tener del sistema dependiendo del rol de la persona y sus deseos, como usuario, administrador, entre otros. Estas técnicas generan una retroalimentación que permite un correcto levantamiento de los distintos requisitos que puede tener el sistema.

- Estructura de Software y Arquitectura: Conjunto de estructuras requerido para razonar sobre un sistema que contiene elementos de software [10]. También comprende conceptos sobre el diseño de software en diferentes niveles de abstracción que pueden ser usados durante las etapas de diseño de arquitectura, así como el diseño detallado del sistema [10].
  - Estilo de Arquitectura: Es usado para la especialización de un elemento y tipos de relación, así como sus restricciones y como estas pueden ser usadas.
  - Patrón de Diseño: Son soluciones frecuentes a problemas que se presenten en el diseño de un sistema de software. Pueden contener estilos de arquitectura para describir una organización de alto nivel del software [10].
  - Diseño de interfaz de usuario: Se encarga de garantizar una efectiva interacción entre el humano y la máquina, manteniendo unos principios como el aprendizaje, la familiaridad del usuario y la consistencia, entre otros.

#### 2.1.2.2. Selección de datos

Es un área amplia la cual consta de diferentes técnicas cuyo objetivo es definir claramente una problemática específica [11]; estas buscan que se pueda aplicar un pensamiento estadístico descriptivo, sobre un conjunto de datos recolectado [11]. Donde los registros u observaciones efectuados proporcionan una serie de datos que necesariamente deben ser ordenados y presentados de una manera inteligible. La estadística descriptiva desarrolla un conjunto de técnicas cuya finalidad es presentar y reducir los diferentes datos observados [11].

- Identificar el problema en una situación crítica de interés: esta técnica dentro de la estadística es usada para definir la problemática de enfoque, la cual es mencionada en el objetivo general 1.2.1. También identificada como ciencia a través de la cual se utiliza un conjunto de datos numéricos para obtener a partir de ellos inferencias (deducciones) basadas en cálculos de probabilidades [11].
- Identificar los conceptos, métodos para estimar y/o las técnicas estadísticas de procesamiento de datos necesarias para dar solución al problema encontrado: su uso es estrictamente para facilitar a los autores de este trabajo de grado el establecer las herramientas a usar antes de iniciar el estudio comprendido en este proyecto.
  - Población, colectivo o universo: Es necesario definir cuál es el colectivo al que se quiere someter a una observación estadística [11]. Este se define como cualquier conjunto de personas objetos, ideas o acontecimientos que se someten a la observación estadística comprendida en este trabajo de grado [11]. Dicha observación es de una o varias características que comparten los elementos del colectivo y que permiten diferenciarlos [11].
  - Tamaño de la población: normalmente denominado con la letra  $n$ , el número de elementos de una población puede ser finito o infinito. Para el caso de este trabajo será finito [11].



Figura. 2.1: Matriz de dispersión [13].

- Muestra: Es la parte seleccionada de una población en la que los elementos que la componen no tienen ninguna característica esencial que los distinga de los restantes [11]. Se utiliza cuando es necesario disponer de una parte representativa de la población. Puede ser elegida inspirándose en el azar (muestreo aleatorio) o con reglas fijadas (muestreo no aleatorio) [11].
- Parámetro o estimador: punto de referencia cuantitativo o cualitativo que permite estudiar la evolución de las características al relativizar los valores de diferentes magnitudes con uno de referencia [11].
- Caracteres: es cada una de las propiedades, rasgos o cualidades que poseen los elementos de una población [11].
- Clasificación del objeto de estudio: esta técnica se utiliza para definir una población, una muestra, un parámetro o un estimador y con base a estos se analizarán las notas estudiantiles para determinar su aptitud en las distintas carreras.
- Matrices de dispersión: esta técnica permite ver la relación existente entre todas las variables que se evalúan pero de forma pareada [12]. Ayuda a una visualización gráfica para guiarnos sobre si realmente las notas estudiantiles de su educación media y las aptitudes del estudiante han o no influenciado en la elección de la carrera.
- Base de datos relacional: un tipo de base de datos que almacena y proporciona acceso a puntos de datos relacionados entre sí. Las bases de datos relacionales se basan en el modelo relacional, una forma intuitiva y directa de representar datos en tablas.



Figura. 2.2: Ejemplo gráfico de base de datos relacional [11].

### 2.1.2.3. Análisis e interpretación

Esta técnica permite darle sentido a los datos recolectados, esto es, relacionarlos con el contexto específico del problema y organizarlos de acuerdo con el objetivo planteado. Estas herramientas que serán mencionadas podrían ser de utilidad para el desarrollo del trabajo de grado en la recolección, análisis y procesamiento de datos, a continuación se mencionan aquellas herramientas que entran en dicha consideración:

- Estadística descriptiva: implica resumir y organizar los datos para que puedan entenderse fácilmente en relación con el objetivo general y los específicos. Aquí buscamos explorar sin restricción alguna los datos en busca de regularidades que llamen la atención[11]. Dentro de las herramientas que podrían ser usadas para esto, tenemos las siguientes:
  - Muestreo: Esta técnica es utilizada para la selección de una muestra a partir de la población de interés, donde la población son los estudiantes próximos a empezar estudios de nivel superior. Se usa para ahorrar en costos, fácil ubicación de los elementos y optimización de tiempo de estudio debido a que la población es muy grande.

Valoración	Productos
30 - 39	1
40 - 49	3
50 - 59	11
60 - 69	21
70 - 79	43
80 - 89	32
90 - 99	9

Figura. 2.3: Ejemplo de muestreo para puntos de productos en un supermercado [11].

- Teoría de probabilidades: se ocupa fundamentalmente de definir y manejar la probabilidad indispensable para cuantificar la aleatoriedad, indicando que tan factible es que ocurra o no

un evento; para nuestro caso este evento sería el elegir correctamente una carrera universitaria tal y como se explica en la sección 1.2.1 [14].

- Estadística inferencial: se estudian las técnicas y procedimientos con el objetivo de extender o generalizar la información obtenida de una muestra de la población.
  - Intervalos de confianza, tolerancia y predicción: Son intervalos que se usan en esencia para predecir un rango de valores entre los que se moverá un parámetro, a través del uso de estimadores. Estos intervalos normalmente acaban en un número (límite superior) que es el mismo con el que empieza el siguiente (límite inferior) [11]. Para el caso particular se analiza la muestra (estudiantes cursando una carrera universitaria) para determinar entre qué valores ronda la afinidad de un estudiante entre los estudiantes para la carrera seleccionada según las notas de los estudiantes en su educación media [15].
  - Pruebas de hipótesis: procedimiento estadístico que a través del estudio de una muestra nos permite determinar el cumplimiento de una hipótesis planteada sobre la importancia de las notas de los estudiantes en su educación media en la población definida haciendo uso de las diferentes técnicas estadísticas aquí descritas. Son conjeturas que se hacen sobre la realidad que aún no se conoce y que se formula para llegar a conocimientos nuevos [15].
  - Análisis de regresión lineal simple: Esta técnica se utiliza para investigar y modelar la relación entre dos variables continuas de interés en la población. El objetivo es mirar la relación existente entre dos variables de modo que se pueda obtener información sobre una de ellas, en nuestro caso, el de la elección vocacional y perfil del estudiante antes de iniciar el periodo universitario. [11]. Dicha relación definida como correlación lineal se evalúa de manera implícita en la herramienta de escogencia a través de la fórmula de covarianza definida como:

$$Cov(X, Y) = \sum = \frac{\sum_{i=1}^n (x^i - x)(y^i - y)}{N - 1} \quad (2.1)$$

Donde  $x$  e  $y$  = media de cada variable; y  $x^i$  e  $y^i$  = valor de las variables para la observación  $i$  [16].

- Modelo predictivo: Modelo de datos basado en estadística inferencial que se utiliza para predecir una determinada respuesta a una promoción de marketing o a una determinada inversión. Su base es la ciencia de datos que a través de estadísticas predice resultados. Para este trabajo de grado se usará como base estadística la regresión lineal anteriormente mencionada que describirá de forma implícita a través de la herramienta de escogencia, la siguiente ecuación:

$$Y = \beta_0 + \beta_1 X_1 + \varepsilon \quad (2.2)$$

Donde  $B_0$  = la ordenada en el origen; y  $B_1$  = pendiente y  $E$  = el error aleatorio [16].

- Análisis de regresión múltiple: se usa para incluir múltiples variables regresivas, es decir, se busca relacionar varias variables. Aquí se determina que tan relacionados están los

resultados de las notas en educación media de los estudiantes con las diferentes carreras de la universidad. Para representar este análisis de forma gráfica a través de la herramienta de escogencia, se escoge el formato de matriz de dispersión ejemplificada en 2.1

- Distribución de probabilidad estadística o modelo estadístico: lista que proporciona todos los valores resultantes que pueden presentarse en un acontecimientos y que son descritos por el parámetro estadístico [15].
- Distribución Normal Estándar: Distribución de probabilidad que nos permite reducir cualquier distribución normal al formato de la normal estándar ya que esta tiene una media aritmética y desviación estándar únicas, cuyos valores son cero y uno respectivamente [15]. Su aplicación estará de forma implícita en la herramienta de escogencia a través de la ecuación:

$$Z = \frac{X - \mu}{\sigma} \quad (2.3)$$

Donde  $X$  = valor de variable a estandarizar; y  $\mu$  = media aritmética de la desviación estándar de la variable [15].

- Gráficos de frecuencias simples: se representan mediante histogramas, que están constituidos por tantos rectángulos como clases se consideran en una distribución. El área de cada rectángulo debe ser igual o proporcional a la frecuencia simple del correspondiente intervalo. En cada uno se verifica implícitamente a través de la herramienta de escogencia lo siguiente:

$$A(i) = n(i) = c(i) * h(i) \quad (2.4)$$

Donde  $h(i)$  = altura del rectángulo; y  $c(i) = L(i) - L(i - 1)$  = amplitud del intervalo [11].

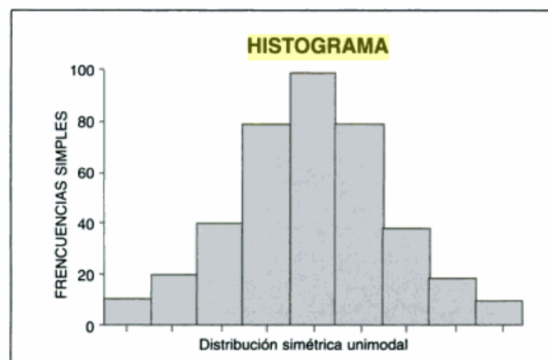
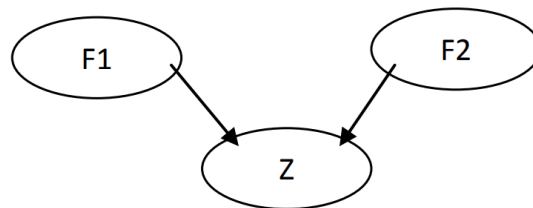


Figura. 2.4: Ejemplo de histograma con distribución normal simétrica [11].

- Frecuencia absoluta simple: Es el número de veces que se presenta un determinado dato de un carácter en los diferentes elementos de una población. Se representa por  $n(i)$  [11].

- Red de Bayes: Las redes bayesianas o red de Bayes organizan un problema mediante un conjunto de variables y las relaciones de dependencia entre ellas. Dado este modelo se puede realizar una estimación de la probabilidad posterior. Más específicamente, es un modelo probabilístico que relaciona un conjunto de variables aleatorias mediante un grafo dirigido, son redes gráficas sin ciclos en el que se representan variables aleatorias y las relaciones de probabilidad que existan entre ellas que permiten conseguir soluciones a problemas de decisión en casos de incertidumbre.



**Figura 1 : Grafo**

Figura. 2.5: Ejemplo de Red de Bayes con una variable  $Z$  y dos factores  $F1$  y  $F2$  [11].

- Parámetro estadístico: cantidad numérica calculada sobre una población y resume los valores que esta toma en algún atributo. También intenta resumir toda la información presente en una población en unos pocos números (parámetros) [15].

#### 2.1.2.4. Validación de los resultados

Al finalizar la etapa de desarrollo del sistema se realizará una encuesta anónima a un grupo de estudio conformado por distintos estudiantes bachiller próximos a salir de la educación media y con la autorización de la institución educativa en la que se encuentren para la realización de distintas preguntas de control que verifiquen la finalidad del sistema, que es apoyar su proceso de elección vocacional.

#### 2.1.3. Trabajos Relacionados

- Buscando carrera-Colombia aprende [17]

En esta plataforma web propuesta por el MEN, en conjunto con el Sistema Nacional de Información de la Educación Superior (SPADIES) [18], para ayudar a elegir la carrera de parte de los estudiantes, se presentan datos tales como los salarios profesionales de una carrera, implementación de una prueba vocacional basado en intereses personales, muestra de las distintas carreras que hay en Colombia y claves para escoger bien una carrera todo esto para facilitar su transición a la educación superior.

- Distintas pruebas vocacionales estandarizadas [19]

A lo largo de la historia de la orientación vocacional, nos podemos encontrar con numerosas pruebas estandarizadas que permiten al usuario valorar desde un punto auto reflexivo las profesiones, donde usualmente se realizan acciones como expresar la opinión del usuario frente a la carrera, observar la motivación de parte del usuario frente a la carrera, aproximaciones frente a los instrumentos de las carreras, entre otros. Estas pruebas permiten tener un acercamiento a la realidad del sujeto y sus intereses, sin embargo, no se deben utilizar como resultados definitivos del proceso, en ejemplo nos encontramos con muchos como el cuestionario de intereses profesionales(CIPSA) [20] y el inventario de intereses profesionales (IPP) [21], entre otros.

- Entrevistas con el involucrado [19]

Se trata de entrevistas que involucran al participante del proceso de elección vocacional, que van desde personales y semiestructuradas solo con la ayuda de un experto en el área a una con completo acompañamiento de los seres significativos de la persona. En la entrevista semiestructurada, se busca conocer la historia formativa del usuario, sus preferencias y expectativas laborales. Las entrevistas que involucran a los seres mas cercanos al usuario como la familia y demás tratan de suplir información, de manera usual la información dada por el usuario puede ser vaga, por eso es importante el hecho de que la familia y demás aporten respecto a las capacidades de la persona en su pasado formativo, llegando a una información completa.

- Técnicas basadas en la discusión grupal [22]

Se trata de distintas técnicas que abarcan un desarrollo grupal y cambia el enfoque de una sola persona a un conjunto de estas buscando elegir su vocación, usualmente estas técnicas se realizan a grupos de estudiantes conocidos a lo largo de la educación media donde cumplen con objetivos específicos dependiendo de la actividad a realizar. aquellas que están basadas en la discusión grupal cumplen objetivos comunes como conocer las opiniones de los participantes en tiempo mínimo, incrementar su información por medio de los aportes grupales y promoviendo la libre expresión de sus opiniones e ideas. En estas técnicas podemos encontrarnos con el uso de debates dirigidos, foros, lluvias de ideas, juegos de rol, entre otros, como actividades a realizar, donde por medio de un tema central se despierta el debate y las características para ayudar al proceso vocacional de los jóvenes.

## 2.2. Metodología de Desarrollo

Scrum se usará como un marco de trabajo y metodología ágil de desarrollo de productos en el cual se emplean distintos procesos y técnicas para poder lograr el desarrollo de productos complejos, donde se necesitan obtener resultados prontamente para poder realizar ajustes con base a los requisitos cambiantes o necesidades emergentes, esto permite tener un énfasis en la flexibilidad y productividad al desarrollar un producto mientras se le entrega al cliente lo que el necesita. Sumado a esto, esta metodología cuenta con una definición de parámetros de trabajo o lineamientos

por llamarlos de una manera, los cuales están ejemplificados y especificados en su guía [23].

- **Equipo Scrum:**  
Consiste en asignar roles a cada parte del equipo para aumentar la funcionalidad de este, entre los roles nos encontramos con el de desarrollador, que se encarga de desarrollar el sistema, incluyéndose en un equipo de desarrollo el cual al ser de pocas personas permite disminuir la complejidad del equipo y aumentar la productividad; sumado a esto nos encontramos con el Scrum master, el cual se encarga de entender las necesidades del producto y ayudar al equipo a entenderlas para maximizar el valor del producto.
- **Incremento:**  
Al hablar de las iteraciones hablamos de la forma fijada en un marco temporal que se encarga de la planificación de lo que se hará en el sprint, las respectivas sincronizaciones y la retrospectiva sobre lo obtenido en cada evento, reunión y demás como un todo.
- **Sprint:**  
Para regular las reuniones para revisiones del progreso, se tienen eventos denominados sprint, los cuales se observan como una ventana de tiempo en el que se deben llegar a completar objetivos del desarrollo del producto, constituyendo una oportunidad para la inspección y adaptación de algún aspecto referente al proyecto en los cuadros temporales obtenidos, minimizando la pérdida de tiempo en el proceso.
- **Artefacto:**  
Al establecerse la meta a cumplir en el sprint, nos encontramos con un cumulo de trabajo cuyo resultado se reconoce como un artefacto, objeto sobre el cual se realizará la inspección y verificación necesaria, de acuerdo con el product backlog, el cual es la lista de requisitos, funcionalidades, características, mejoras y correcciones que se deben hacer sobre el desarrollo del proyecto.
- **Definición de producto terminado:**  
Un producto o parte del sprint backlog se encuentra terminado solo en caso de que las evaluaciones pertinentes se hayan realizado, que se encuentre listo para poder entregar un incremento de funcionalidad para su liberación, integración, o deseo que se tenga con el, integrándolo de manera rigurosa con los demás elementos que se encuentren en esta categoría

### 2.2.1. Tipo de Estudio

El tipo de estudio a utilizar será de carácter exploratorio al ser una investigación que se realiza específicamente cuando el tema ha sido poco trabajado o valga la redundancia, poco explorado,

buscando aumentar el grado de conocimiento de un fenómeno. A pesar de que la orientación vocacional es un espectro ampliamente trabajado en el área de la psicología, el uso de estadísticos y criterios particulares entre el usuario a estudiar y la carrera, además de un sistema de información centralizada para la toma de decisiones, tiene pocos precedentes o no se encuentra explicado de esta manera con una herramienta computacional accesible para todos.

### 2.2.2. Actividades

#### 1. Búsqueda y Revisión de la documentación y modelos estadísticos:

- Buscar los distintos artículos, manuales, libros y demás documentación sobre la orientación vocacional utilizable para el desarrollo del sistema.
- Obtener y analizar la información relacionada a las técnicas predictivas para su selección y posterior utilización.
- Obtener y catalogar la información relacionada con el mercado laboral, y el rendimiento académico en la educación media de los estudiantes que hayan cursado al menos un semestre de una carrera de ingeniería.
- Realizar las encuestas anónimas para el conjunto de datos de comentarios de cada carrera, suministradas a estudiantes de cada carrera.
- Obtener las autorizaciones y normativas pertinentes para el uso de datos del cuerpo estudiantil de la universidad y realización de encuestas en la institución de educación media y la universidad.
- Obtener, leer y analizar la información importante relacionada.

#### 2. Interiorización y uso de la documentación encontrada:

- Integración y anonimización de los datos entregados por la Pontificia Universidad Javeriana Cali de las notas de educación media de estudiantes pertenecientes a la institución.
- Automatización de la obtención de los datos del mercado laboral dada por los entes pertinentes al público.
- Elegir los modelos estadísticos a utilizar en el sistema en los distintos conjuntos de datos (notas de educación media estudiantiles, mercado laboral).
- Elección del software para la construcción de los modelos predictivos.
- Utilización de los modelos predictivos para mostrar el estado del mercado laboral y la afinidad del estudiante próximo en la carrera.

#### 3. Planificación del desarrollo del sistema:

- Levantar y especificar los requisitos funcionales, no funcionales, restricciones, de interfaz, entre otros, necesarios para la elaboración del sistema bajo estándares de la IEEE.

- Realizar la documentación pertinente restante, basados en los estándares IEEE y el lenguaje de modelado de sistemas de software UML 2.X(Diseño de la arquitectura lógica y física del sistema, Diagrama de casos de uso, de clase, relacional, de paquetes y componentes, de contexto, de secuencia, entre otros), además de la definición de los casos de prueba para evaluación del sistema y la correlación con su flujo.
- Definir las etapas de desarrollo por medio de sprints, sprint backlogs y demás elementos presentes en la metodología de desarrollo ágil Scrum.

#### 4. Implementación del sistema:

- Desarrollar el CRUD de la base relacional y establecimiento de esta para su utilización en el sistema teniendo en cuenta los requisitos especificados del sistema.
- Implementar los modelos predictivos en la herramienta escogida anteriormente y los conjuntos de datos adecuados(mercado laboral colombiano y notas de educación media de estudiantes en las carreras).
- Desarrollar el apartado de back-end de la aplicación web para el consumo de las apis de datos externos y el manejo de la base de datos del sistema para ser consumidas en el sistema.
- Desarrollar el apartado de front-end de la aplicación web para el funcionamiento visual del sistema, donde se consumirán las apis correspondientes para su funcionamiento y definirá la usabilidad del sistema finalmente.

#### 5. Evaluación del sistema

- Realizar y documentar los distintos casos de prueba definidos para los requerimientos en el sistema para poder encontrar fallos o cosas a mejorar.
- Realizar la implementación y/o mejora de los detalles encontrados para el despliegue del sistema.
- Realizar las encuestas pertinentes con el grupo de estudio de estudiantes bachilleres de grado 11 próximos a entrar a la educación superior para evaluar que la aplicación ayuda en la elección vocacional del estudiante.

#### 6. Realización del documento final:

- Escribir el documento final.
- Ejecutar por parte del director de tesis
- Efectuar la corrección del documento.
- Realizar la entrega y sustentación del documento final.

## 2.3. Cronograma

Cronograma por semana incluyendo las actividades descritas en la Sección 2.2.2, ver Figura 2.6, de las cuales muchas, como lo son el desarrollo del sistema, la evaluación y mejoramiento del mismo se realizan de manera paralela y por un largo tiempo, por efectos de la metodología de desarrollo escogida(2.2), sumado a esto se encuentra el hecho de que la actividad de escritura del texto se realizará de manera constante a lo largo del desarrollo del proyecto.

Figura. 2.6: Cronograma de actividades

Actividades	Mes 1				Mes 2				Mes 3				Mes 4				Mes 5				Mes 6			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Buscar información relevante al proyecto	■																							
Realización y análisis de encuestas sobre las carreras		■																						
Análisis y elección de las técnicas predictivas a usar			■																					
Implementación del modelo predictivo y métodos de obtención de datos				■																				
Levantamiento de requisitos					■																			
Realización de la documentación pertinente						■																		
Definición de los sprints							■																	
Desarrollo a nivel de bases de datos								■																
Desarrollo de back-end									■															
Desarrollo de front-end										■														
Evaluación de los casos de prueba											■													
Mejora del sistema												■												
Evaluación de resultados por encuestas																				■				
Escritura del documento final																								
Revisión del documento y corrección																								
Entrega y sustentación del documento																								■

## 2.4. Recursos

### 2.4.1. Humanos

1. Cristian Steven Osorio Paz: estudiante de pregrado Pontificia Universidad Javeriana Cali.
2. Sebastián Toro Franco: estudiante de pregrado Pontificia Universidad Javeriana Cali.
3. Dr. Gerardo Mauricio Sarria: Como Director del proyecto de grado, Docente del Departamento de Electrónica y Ciencias de la Computación y Director de la carrera de ingeniería de sistemas y computación de la facultad de ingeniería en la Pontificia Universidad, perteneciente al grupo Destino de esta misma universidad, con títulos de Doctorado en ingeniería, con énfasis en ciencias de la computación en la Universidad del Valle.

4. Grupo de estudiantes voluntarios y anónimos por carrera que participen en las encuestas para los comentarios de estas.
5. Grupo de estudiantes bachilleres voluntarios y anónimos que participen en las encuestas para evaluar el cometido del sistema.

### 2.4.2. Técnicos

- Autorización y normativa para el manejo de datos pertenecientes a las notas de educación media de los estudiantes de la Pontificia Universidad Javeriana Cali por parte del Centro de Servicios Informáticos(CSI) de la institución.
- Autorización de una institución educativa media para la obtención de voluntarios para evaluación de resultados en la aplicación.
- Computadores personales de los estudiantes encargados del proyecto de grado con especificaciones básicas.
- Computadores de la universidad con especificaciones avanzadas, con un abanico de aplicaciones y membresías utilizables para el desarrollo del proyecto.
- Conjunto de notas de educación media de estudiantes en las carreras de ingeniería en la Pontificia Universidad Javeriana Cali, además de aquellos que abandonaron cada carrera.

### 2.4.3. Presupuesto

Se incluye la descripción del presupuesto de recursos humanos (ver Cuadro 2.2) y técnicos ( ver Cuadro 2.3), basados en el salario mínimo legal vigente de Colombia para el año en que se desarrolla el proyecto (ver Cuadro 2.1), dado el presupuesto total para el proyecto en si (ver Cuadro 2.4).

Salario mínimo Colombia 2021
\$908.526

Cuadro 2.1: Salario mínimo para Colombia en el año 2021

Recurso humano	Salario minimo		Costo mensual	Meses	Sub total
Sebastian Toro Franco	1.5 x S.M.L.V		\$1.362.787	6	\$8.176.725
Cristian Steven Osorio Paz	1.5 x S.M.L.V		\$1.362.787	6	\$8.176.725
Dr. Gerardo Mauricio Sarria	Horas x semana 2	Costo x hora \$71.299	\$570.392	6	\$3.422.352
<b>Total</b>					\$19.775.802

Cuadro 2.2: Presupuesto para recursos humanos

Recurso técnico	Costo mensual de uso	Meses	Sub total
Servidor web	\$30.000	6	\$180000
Servicio de internet del equipo desarrollador	\$151.900	6	\$911.400
Servicio de energia eléctrica del equipo desarrollador	\$150.900	6	\$905.400
<b>Total</b>			<b>\$1.996.800</b>

Cuadro 2.3: Presupuesto para recursos técnicos

Recurso	Costo a lo largo del proyecto
Recursos humanos	\$19.775.802
Recursos técnicos	\$1.996.800
<b>Total</b>	<b>\$21.772.602</b>

Cuadro 2.4: Presupuesto total necesario para el proyecto



# Modelo Estadístico

---

## 3.1. Introducción

J. Aitchison y Diane Sculthorpe definen la predicción estadística como el uso de los datos de un experimento informativo cualquiera definido como  $E$  para hacer declaraciones sobre el posible resultado de un experimento futuro cualquiera definido como  $F$  [24]. Dichas declaraciones predictivas generalmente son de carácter inferencial y cuyo objetivo es: indicar el resultado probable de  $F$  o, sugerir un subconjunto de posibles resultados en el que el resultado real de  $F$  está contenido [24].

Por otro lado, la aplicación de los modelos estadísticos en la sugerencia de información relevante a un individuo específico es una técnica ya utilizada antes. Según TechTarget, en los últimos años las redes Bayesianas son usadas por los filtros de spam en los cuales mediante modelos predictivos se establece la probabilidad de que un mensaje sea spam [25]. Esto permite la adaptación de un modelo predictivo según el objetivo del proyecto y cómo se desea manejar la información, abriendo la posibilidad de que información relevante del ámbito laboral en Colombia con respecto a una carrera llegue al destinatario correcto [25].

Oracle NetSuite define el modelamiento predictivo como una técnica estadística que utiliza la minería de datos y/o Machine Learning para predecir y pronosticar resultados futuros probables a través de datos históricos ya existentes [26]. Esta técnica analiza la información que se posee y deposita lo que aprende en un modelo generado para predecir estos resultados esperados [26].

Cada modelo predictivo es diferente dependiendo de lo que se quiera pronosticar, sin embargo, todos concuerdan en el hecho de que su funcionamiento no es 'reparado'. Por el contrario, es validado o revisado regularmente para incorporar cambios en el flujo de datos [26]. Es decir, no es una predicción de una única vez y que no debe realizarse más; constantemente se deben realizar asunciones con base en lo que pasó y lo que está ocurriendo, y si la nueva información altera de alguna forma el presente entonces se debe recalcular cómo esto impacta el resultado futuro [26].

El uso de modelos predictivos permite el manejo de la información de forma adecuada y que a su vez esta sea desplegada correctamente al destinatario intencionado. Su aplicación en el ámbito académico permite que el individuo, es decir el estudiante, pueda acceder a la información más relevante para este, permitiendo que su apropiación del conocimiento sea más amena [25]. SEASIA contempla estos logros dentro de los grandes beneficios para cualquier tipo de industria que use los modelos predictivos de la siguiente manera [27]:

1. Muy útil en la contemplación de pronósticos de demanda.
2. Planificación de la fuerza laboral y análisis de la rotación de clientes.
3. Profundo análisis de la competencia.
4. Pronóstico de factores externos que pueden afectar el flujo de trabajo.
5. Mantenimiento del recurso humano.
6. Identificación de riesgos financieros o de cualquier tipo.

### 3.2. Estructura de un modelo estadístico Predictivo

Cuando se habla de la estructura de un modelo estadístico predictivo se hace referencia a las partes generales y a grosso modo, las cuales contienen los diferentes componentes [25]. La estructura define la naturaleza y en últimas el objetivo del modelo predictivo; es la encargada de garantizar que el modelo cumpla con las predicciones que se desean obtener [27]. Según Joseph M. Carew, la estructura de un modelo está compuesta por las siguientes etapas y/o zonas [25]:

1. Obtención de la información: normalmente la parte más desafiante y validada en un modelo predictivo es la obtención de datos en cantidad adecuada y organización la información de manera correcta para el uso y desarrollo de los algoritmos internos del modelo; definiendo si se hará un acercamiento paramétrico o no paramétrico en el modelo predictivo [25].
2. Limpieza de la información: una vez obtenida la información y terminada su organización inicial, hacer un limpieza de esta comprende no solo deshacerse de datos mal registrados que puedan causar ruido, sino también garantizar que cada registro tenga sentido y pueda ser usado dentro del modelo [25].
3. Mantenimiento de la información: hace referencia a cómo evitar que la información produzca modelos incorrectos. Es decir, no sobreajustar la información y evitar hacer pruebas en exceso que puedan generar modelos engañosos. Modelos que pueden demostrar ser increíblemente precisos, pero en últimas solo han memorizado ciertos patrones más que generalizar como deberían [25].
4. Uso de la información: hacer referencia al preprocesamiento de la información a una forma adecuada para el algoritmo de modelado escogido, a su vez que especificar un subconjunto de datos a ser usados en dicho modelo [28].
5. Validación del modelo: antes de empezar a usar los datos en un entorno real con el modelo escogido, es necesario validar que dicho modelo presenta comportamientos esperados. Dichos comportamientos definen si el modelo fue construido correctamente, si la información fue obtenida adecuadamente o si ambos elementos en conjunto fueron utilizados de manera esperada para así determinar que parte debe ser corregida. Dentro de estas validaciones encontramos:

entrenar o estimar parámetros del modelo sacados del conjunto de datos. Realizar una prueba al rendimiento del modelo o prueba de bondad de ajuste para validar que tan adecuado es el modelo. Finalmente, es necesario verificar que tan preciso es el modelo en datos que no han sido usados y así calibrarlo [28].

6. Uso del modelo con la información: si los resultados del modelo son satisfactorios en su rendimiento, se procede a usarlo para generar las predicciones.
7. Obtención de los resultados: comprende la etapa donde las primeras predicciones son obtenidas con la información ya filtrada y acomodada al modelo escogido.
8. Validación de los resultados: se verifica que las predicciones generadas estén dentro de los rangos esperados y tengan sentido al ser comparadas con los datos y las predicciones anteriores. Es decir, que a pesar de que se hayan generado predicciones, los resultados no sean tan alejados de los obtenidos anteriormente garantizando así que el presente no cambie y por ende el futuro se mantenga estable [27].

### 3.3. Componentes del modelo

Los componentes del modelo predictivo construido para este proyecto constan de aquellas partes que permiten obtener los resultados esperados; garantizando también que la información tenga un flujo correcto y que, además, con cada predicción nueva que se genera, el resultado no afecte la precisión del modelo. Dentro de los componentes están:

#### 3.3.1. Nodos fuente

Son aquellos nodos encargados de recibir toda la información en crudo que luego se limpiará y adecuará al modelo particular. Este nodo puede recibir distintos tipos de fuentes incluyendo una base relacional que, para el propósito del proyecto funciona perfecto. Cabe aclarar que solo realiza funciones de lectura. A través del nodo origen (Véase la figura 3.1) el modelo predictivo recibe todos los datos referentes a los estudiantes con aspiración a entrar en la Pontificia Universidad Javeriana Cali y los almacena en una estructura de tabla, donde cada línea equivale a un registro con sus respectivas variables.

#### 3.3.2. Nodos de transformación de los datos

Son aquellos nodos que permiten modificar los registros o la información de manera general, retornando los datos modificados de forma que el manejo de estos sea mucho más fácil y que su lectura sea sencilla. Permiten adaptar los datos al modelo específico para obtener el objetivo buscado [29].



Figura. 3.1: Nodo origen dentro del modelo predictivo [29].

### 3.3.2.1. Nodo de agregación

El nodo de agregación (Véase la figura 3.2) es aquel que permite agrupar registros de acuerdo a los valores que estos presentan. De esta forma se puede visualizar mucho más fácil que grupos de datos hay dentro del conjunto de información. Sirve para catalogar de manera rápida la información, ejemplo: *M: masculino. F: femenino* [29]. Dentro del modelo predictivo este nodo se encarga de agrupar los registros de acuerdo a características de: género, edad, ciudad de procedencia, carrera a la que aplican y de esta forma poder determinar cuantos registros hay por cada grupo. Esto permite identificar de forma rápida medidas de tendencia central descritas en 2.1.2.3

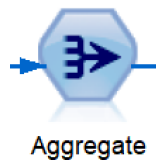


Figura. 3.2: Nodo agrupación dentro del modelo predictivo [29].

### 3.3.2.2. Nodo de derivación

Este nodo permite generar nuevos campos a partir de los ya existentes. Estos campos pueden ser de tipo: fórmula, bandera, nominal, estado, recuento o condicional [29]. Este nodo (Véase la figura 3.3) dentro del modelo predictivo permite mirar de forma gráfica y cuantitativa cuál es el promedio académico por estudiante próximo a ingresar, cuál es la ciudad de procedencia más repetida y cuál es la carrera dentro de la facultad de ingenierías que más es escogida por los aspirantes (Véase 2.1.2.3).

### 3.3.2.3. Nodo de tipado

El nodo de tipado (Véase la figura 3.4) sirve para modificar los tipos de datos en cada campo, las medidas que manejan y definir el rol que tendrán dentro del modelo predictivo [29]. Este nodo

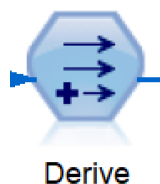


Figura. 3.3: Nodo derivación dentro del modelo predictivo [29].

cumple dos funciones principales en el flujo del modelo predictivo: por un lado permite indicarle al modelo cuál será la variable objetivo a predecir, es decir, la afinidad del estudiante aspirante con la carrera seleccionada. Por otro lado, facilita la modificación al tipo de dato de variables como el género o la edad debido a una mala clasificación automática de parte del software.

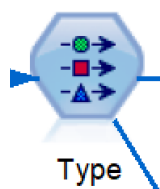


Figura. 3.4: Nodo de tipado dentro del modelo predictivo [29].

#### 3.3.2.4. Nodo de ordenamiento

Este nodo (Véase la figura 3.5) permite ordenar los registros de acuerdo con el campo que se indique [29]. Esto hace más fácil identificar que registro tiene los mejores puntajes o valores en la variable objetivo. Dentro del modelo predictivo permite ordenar todos los registros de los estudiantes aspirantes; inicialmente de acuerdo con su promedio en la educación media y posteriormente, con la afinidad o *esperanza* que presenta cada estudiante con la carrera elegida.



Figura. 3.5: Nodo de ordenamiento dentro del modelo predictivo [29].

### 3.3.3. Nodos de modelamiento

Son aquellos nodos que aplican modelos estadísticos previamente contruidos (Véase 2.1.2.3) en la herramienta sobre los datos obtenidos mediante el nodo origen (Véase 3.3.1). Internamente aplican el modelo estadístico escogido con sus diferentes técnicas probabilísticas sobre la información, mostrando el resultado (si se desea observar) o puntuación en la variable objetivo de cada registro [29]. Dentro del modelo estadístico predictivo procesan la información y por cada estudiante aspirante o registro determinan el porcentaje de afinidad entre el estudiante y la carrera escogida.

#### 3.3.3.1. Regresión lineal

Este modelo (Véase 2.1.2.3) definido internamente en el nodo de modelamiento de regresión lineal (Véase la figura 3.6) es usado para predecir el valor de una variable basándonos en los valores de otra variable. Dentro del modelo el nodo solo recibe los datos ya transformados por los nodos anteriormente descritos y genera un calculo de afinidad o *esperanza* (Variable dependiente) con cada estudiante aspirante y su respectiva carrera escogida; es esencial para observar el primer resultado de las 3 formas de predicción escogidas. Para que este modelo sea aplicado correctamente sobre los datos se deben garantizar 6 cosas [30]:

1. Ambas variables, tanto la dependiente (objetivo) como la independiente (predictora) deben ser continuas.
2. Debe haber una relación lineal entre ambas variables. Esto se puede comprobar de varias maneras pero la más asequible viene siendo un diagrama de dispersión.
3. No debe haber datos atípicos en la información. Entendidos como aquellos registros dentro de la variable dependiente que no siguen el patrón establecido por la fórmula de regresión.
4. Cada registro u observación debe ser independiente del otro.
5. Debe haber homocedasticidad en la información. Significa que la variación entre cada observación de la variable dependiente se mantiene igual o similar a lo largo de todo el experimento.
6. Finalmente, se debe verificar que los errores residuales de la línea de regresión se distribuyen de manera aproximadamente normal.



Figura. 3.6: Nodo modelo regresión lineal dentro del modelo predictivo [29].

### 3.3.3.2. Regresión logística

Es un modelo estadístico (Véase 2.1.2.3) que predice la probabilidad de que una observación caiga en una de dos categorías de una variable dependiente basado en una o más variables independientes que pueden ser continuas o categóricas [30]. El nodo de regresión logística (Véase la figura 3.7) dentro del modelo predictivo recibe los datos transformados y limpiados para un fácil manejo y así generar la segunda forma de predicción sobre la variable dependiente; todo esto lo realiza internamente el nodo. Para que las predicciones sean efectivas y esta parte del modelo trabaje correctamente se debe asumir y emplear lo siguiente [29]:

1. La variable dependiente o a predecir debe ser dicotómica. Ejemplo: *Femenino o Masculino. Aprobó o No aprobó.*
2. Se tienen una o más variables independientes que pueden ser continuas o categóricas.
3. Cada registro u observación debe ser independiente del otro.
4. Debe haber una relación lineal entre cualquier variable continua independiente y la transformación logarítmica de la variable dependiente. Esto se puede probar de muchas formas, para el caso de este proyecto se hará con la prueba de cajas y bigotes [30].



Figura. 3.7: Nodo modelo regresión logística dentro del modelo predictivo [29].

### 3.3.3.3. Modelo carma

El nodo CARMA (Véase la figura 3.8) utiliza un algoritmo de descubrimiento de reglas de asociación para encontrar reglas de asociación existentes en los datos. Las reglas de asociación son instrucciones del tipo: *Si antecedente, entonces consecuencia* [29]. De esta forma, dentro del modelo predictivo internamente establece escenarios posibles y determina el resultado posible con la variable dependiente para cada registro, es decir, cada estudiante aspirante y completando así la 3 tercera forma de predicción.

### 3.3.4. Nodos contenedores de modelo

Este tipo de nodo (Véase la figura 3.9) contiene las predicciones, es decir, los resultados obtenidos de aplicar cada modelo escogido (Véase 3.3.3.1, 3.3.3.2 y 3.3.3.3) sobre los datos fuente. Incluye los

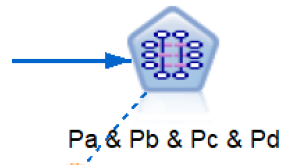


Figura. 3.8: Nodo modelo carma dentro del modelo predictivo [29].

datos iniciales, así como predicciones realizadas al principio del proceso de cada modelo, pero que al momento de consultar el nodo solo sirven como casos de entrenamiento para las futuras predicciones; así mismo, muestra los resultados de forma ordenada de manera que se puedan comparar con anteriores [29].

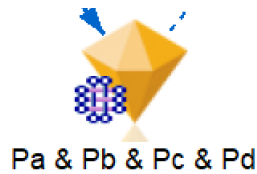


Figura. 3.9: Nodo contenedor de un modelo estadístico dentro del modelo predictivo [29].

### 3.3.5. Nodos de exportación

Es un nodo (Véase la figura 3.10) que permite almacenar toda la información resultante del modelo predictivo, junto con las predicciones trabajadas sobre la información en una estructura de dato estandarizada para fácil acceso [29].



Figura. 3.10: Nodo de exportación dentro del modelo predictivo [29].

### 3.4. Predicciones del Modelo

Una vez completado el flujo de los datos a través del modelo, los resultados obtenidos son guardados en una estructura de datos de fácil acceso a través del nodo de exportación (Véase la figura 3.10) para posteriormente ser accedido por el sistema. En el caso del sistema construido en este proyecto la estructura de datos es una tabla. Estas predicciones son las que determinan el grado de afinidad del estudiante con cada carrera, las cuales fundamentan su veracidad en toda la información previamente almacenada a través de técnicas de aprendizaje [31].

### 3.5. Análisis de las Predicciones

- El primer modelo que debe ser analizado viene siendo el CARMA. La forma para analizar los resultados de este modelo es a través del porcentaje de confianza y de soporte a la regla de asociación entre en las diferentes variables, es decir las combinaciones de antecedentes para dar lugar al consecuente. Aquí lo que se busca en la asociación con mayor porcentaje de soporte para tener cierta certeza de que con los predictores escogidos se llega al objetivo deseado. Este modelo es como el punto de partida para saber que el modelo va por buen camino y, sobre todo, que el resultado esperado es posible de obtener. Se observa cual regla de asociación tiene mayor soporte y esa opción es la que se asume como mejor resultado cercano al esperado.
- Posteriormente se analizan los diferentes resultados arrojados por el modelo de regresión lineal. Aquí a través de un histograma o gráfico de barras (Véase 2.1.2.3) se visualiza cuáles son las variables independientes más contribuyentes dentro del modelo para que el resultado esperado se produzca. Entre más grande el histograma, mayor la contribución de la variable al modelo. Para confirmar esta respuesta se procede a pasar a la pestaña de resumen donde esta la tabla de los coeficientes. Aquí numéricamente se valida lo que se vio gráficamente en los histogramas anteriores, donde el valor debajo de la columna  $t$  indica el grado de significancia o contribución de la variable al modelo. Entre más cercano el valor a 0, mayor la contribución y si se obtiene un cero absoluto entonces la variable es muy significativa para el modelo.
- Finalmente, el modelo de regresión logística valida la capacidad predictiva del modelo. Es decir, que con los predictores escogidos o variables independientes incluidas en la ecuación de regresión se pueden generar predicciones sobre el resultado esperado. Este modelo está reservado para análisis interactivo, es decir que requiere de un monitoreo constante del desarrollador y en vez de ser entregado a las demás partes del sistema, se queda en este nivel para uso exclusivo de los desarrolladores.



# Obtención y Análisis de Requisitos

---

## 4.1. Introducción

Este capítulo se enfoca en la planeación del sistema, pasando por la definición general de la funcionalidad del sistema, describiendo sus módulos y/o componentes y arquitectura, luego realizando la generalización y especificación de los casos de uso del sistema y contexto de este. Después de estas importantes definiciones para el desarrollo del sistema se describen los distintos requerimientos encontrados para el desarrollo en cuestión.

## 4.2. Descripción general del sistema

A continuación se presenta un diagrama de contexto del sistema a desarrollar (Véase en al Figura 4.1). Es preciso señalar que este sistema apoya la elección vocacional en lo que se constituye como el grupo objetivo de este sistema, estudiantes de alta secundaria próximos a entrar a la educación superior; de manera que se puedan apoyar en información activa del Observatorio Laboral Nacional Colombiano sobre el mercado laboral con sus distintas carreras, esto implica que es una funcionalidad base del sistema el permitir a los usuarios visualizar los valores históricos de salario para las distintas carreras y/o labores y su predicción para el periodo adyacente.

El hecho de que el sistema debe permitir apoyar la toma de decisiones implica que se debe permitir realizar comparaciones entre las distintas carreras en todos los ámbitos que se hace trabajo sobre los datos para crear información, como el comportamiento del mercado laboral entre distintas carreras o el perfilamiento que el mismo usuario pueda tener entre distintas carreras por poner un ejemplo.

El sistema debe permitir una administración sencilla de los datos de entrada para su regulación y visualización, lo que implica que el sistema debe tener una interfaz que permita observar los distintos datos presentes en la aplicación además de distintos apartados que permitan el seguimiento de eventos y el estado general de la arquitectura y las peticiones manejadas entre los distintos servicios.

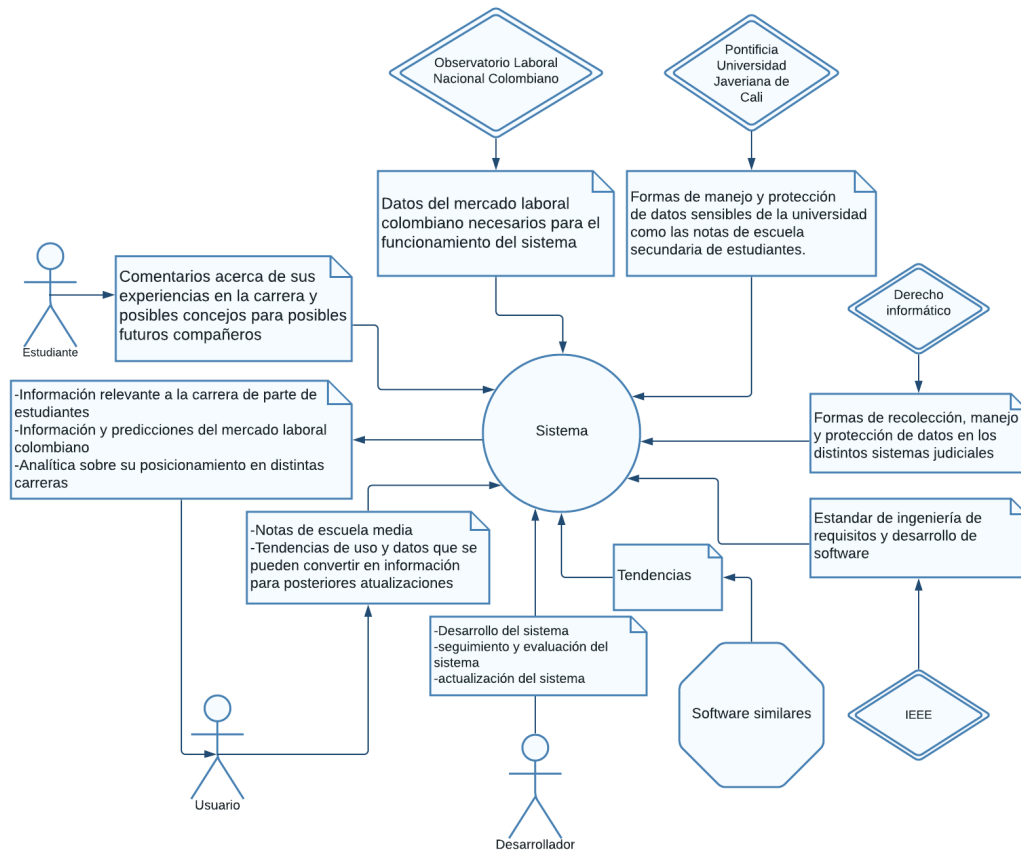


Figura 4.1: Diagrama de contexto del sistema.

### 4.3. Diagrama de casos de uso

Este diagrama muestra el comportamiento de todos los objetos involucrados, su estructura y relaciones entre sí. A través de este se pretende representar el sistema de la aplicación para usuarios estudiantes (Véase en la Figura 4.2).

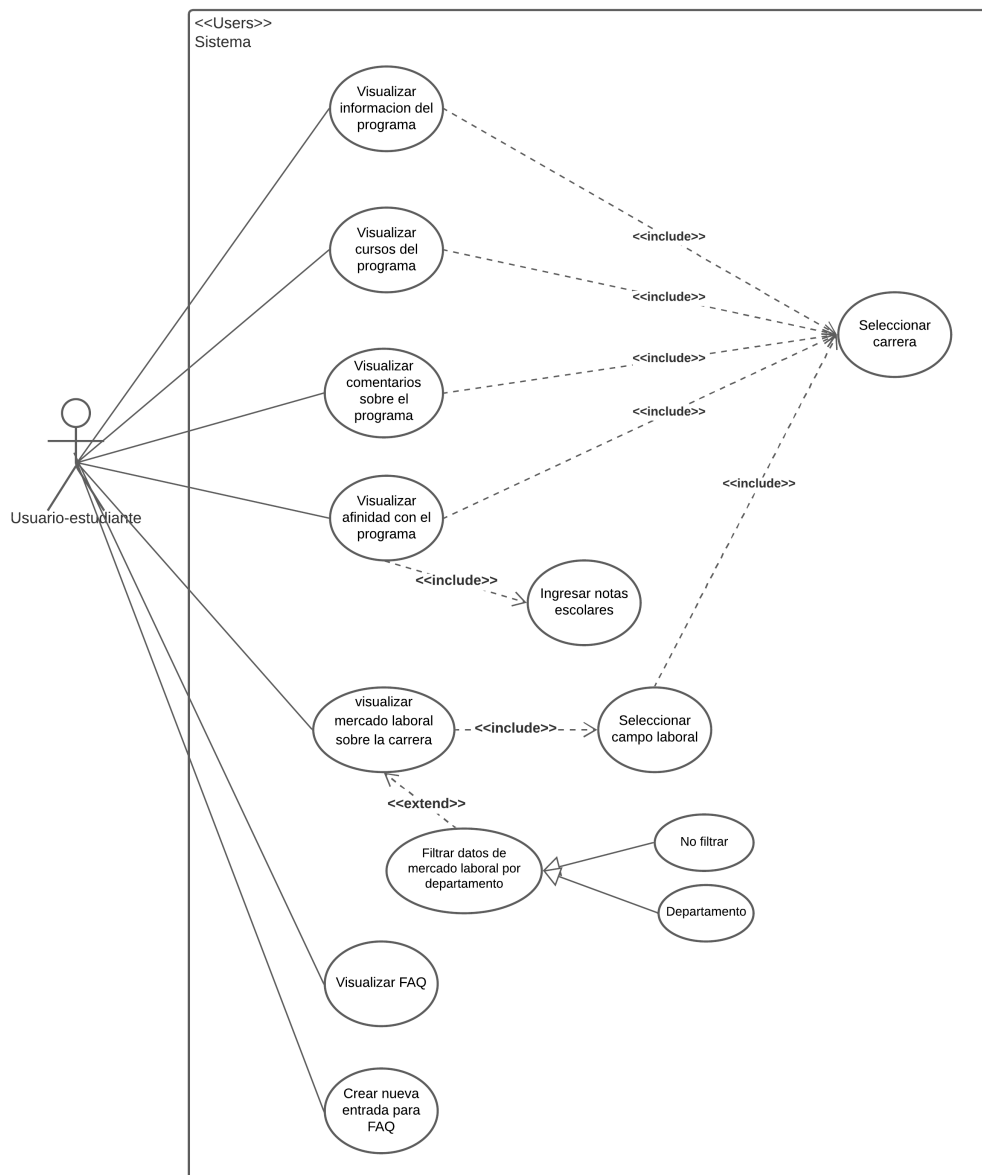


Figura 4.2: Diagrama de casos de uso del sistema.

## 4.4. Requerimientos

La obtención de los requisitos del sistema requiere de un análisis contextual, para lo cual se toma como base la información contenida en el marco de referencia:

- Ingeniería de Software, más específicamente la elicitación de requisitos.
- Selección de los datos.
- Análisis e interpretación.
- Validación de los resultados.

También, se analizaron no solo trabajos relacionados, sino también soluciones similares. Primeramente, *Buscando Carrera-Colombia aprende* [17] es una propuesta del MEN en conjunto con el Sistema Nacional de Información de la Educación Superior (SPADIES) [18], que presenta datos que pueden ser relevantes a la hora de elegir una carrera profesional en el contexto colombiano, así como la oferta que presenta el ámbito académico en este mismo. Segundamente, están las ya mencionadas pruebas psicotécnicas o vocacionales. Donde diferentes organizaciones logran acercarse más a la realidad de cada sujeto, al igual que conocer sus intereses en un punto de vista auto-reflexivo. Más específicamente, se toman como referencia el CIPSA [20] y el IPP [21].

Finalmente, se toma como base la información suministrada por los mismos usuarios finales. Esta información es obtenida a través de preguntas generales pero que permiten conocer un poco más la realidad de cada sujeto. De esta información se destacan los siguientes temas:

- Datos personales del involucrado.
- Realidad familiar del involucrado.
- Realidad extra familiar del involucrado.
- Intereses del involucrado.
- Rendimiento escolar del involucrado.

### 4.4.1. Requerimientos Funcionales

Partiendo del diagrama de casos de uso realizado, los Requerimientos se dividen de acuerdo a los módulos del sistema. Estos grupos de requisitos, son los que documentan la estructura fundamental de cada módulo y dependiendo de su pertenencia, poseen una designación particular (Véase desde la Figura 4.3 hasta la Figura 4.4.1). Estas designaciones son como siguen:

- APP: Aplicativo Web.

- CU : Cursos.
- PR : Predicciones.
- CIG : Centro Informativo General.
- CIAC : Centro Informativo Académico.

Aplicativo Web				
Id	Nombre	Descripción	Dependencias	Prerrequisitos
APP-1	Visualizar información sobre los cursos	El sistema debe permitir al estudiante visualizar información general sobre los cursos de parte de estudiantes universitarios	-	-
APP-2	Visualizar carreras	El sistema debe permitir visualizar las carreras disponibles dentro de la Javeriana Cali en la Facultad de Ingenierías	CU-01	Deben existir cursos
APP-3	Vincular carrera	El sistema debe permitir al estudiante vincular una carrera de la facultad de Ingenierías con su perfil	CU-02	Deben existir carreras universitarias
APP-4	FAQ	El sistema deb contar con una sección destinada a la solución de preguntas que puedan surgir antes, durante y después del uso del aplicativo.	-	-
APP-5	Visualizar información sobre las carreras	El sistema debe permitir al estudiante visualizar información general sobre las carreras de parte de estudiantes universitarios	CU-03	Deben existir carreras universitarias

Figura 4.3: Requerimientos generales del aplicativo módulo principal, fuente propia.

Cursos y Carreras				
Id	Nombre	Descripción	Dependencias	Prerrequisitos
CU-01	Visualizar información sobre los cursos	El sistema debe permitir al estudiante visualizar información general sobre los cursos de parte de estudiantes universitarios	-	-
CU-02	Visualizar carreras	El sistema debe permitir visualizar las carreras disponibles dentro de la Javeriana Cali en la Facultad de Ingenierías	CU-01	Deben existir cursos
CU-03	Vincular carrera	El sistema debe permitir al estudiante vincular una carrera de la facultad de Ingenierías con su perfil	CU-02	Deben existir carreras universitarias
CU-04	Visualizar información sobre las carreras	El sistema debe permitir al estudiante visualizar información general sobre las carreras de parte de estudiantes universitarios	CU-03	Deben existir carreras universitarias

Figura. 4.4: Requerimientos del módulo Cursos, fuente propia.

#### 4.4.2. Requerimientos No Funcionales

Primero, desde la eficiencia, se pretende resolver la gestión de información puntual y asertiva acerca de distintas carreras universitarias de la mano de estudiantes activos en ellas; segundo, la recopilación masiva de información acerca del panorama del mercado laboral actual; por último, la administración de las notas escolares obtenidas en la educación media para una posterior recomendación. Por otro lado, al manejar información privada de los estudiantes y universidades, es

Predicciones				
Id	Nombre	Descripción	Dependencias	Prerrequisitos
PR-01	Crear las predicciones	El sistema debe permitirle al estudiante crear una nueva predicción con una carrera.	CU-01, CU-02	Deben existir cursos y carreras
PR-02	Mostrar las predicciones	El sistema debe poder mostrar la predicción realizada en la carrera escogida por el estudiante.	PR-01	Deben existir predicciones
PR-03	Modificar las predicciones	El sistema debe permitir al estudiante realizar una nueva predicción en una carrera ya escogida y modificar su resultado si así lo desea.	PR-01	Deben existir predicciones
PR-04	Visualizar nivel de afinidad a una carrera	El sistema debe permitir al estudiante visualizar que tan afín es con una carrera vinculada	CU-03	Deben existir carreras universitarias

Figura. 4.5: Requerimientos del módulo Predicciones, fuente propia.

Centro informativo general				
Id	Nombre	Descripción	Dependencias	Prerrequisitos
CIG-01	Almacenar información del mercado laboral	El sistema debe permitir al estudiante visualizar información relevante sobre los cursos de parte de estudiantes universitarios	-	-
CIG-02	Visualizar información del mercado laboral	El sistema debe permitir visualizar las carreras disponibles dentro de la Javeriana Cali en la Facultad de Ingenierías	CU-01	Deben existir cursos
CIG-03	Catalogar información del mercado laboral	El sistema debe permitir al estudiante vincular una carrera de la facultad de Ingenierías con su perfil	CU-02	Deben existir carreras universitarias

Figura. 4.6: Requerimientos del módulo Centro Informativo General, fuente propia.

importante la seguridad lógica y de datos al momento de administrar el acceso a estos. Finalmente, desde la usabilidad, la aplicación debe ser cómoda e intuitiva al usuario y, de esta manera, visualizar indicadores acerca del mercado laboral actual; segundo, visualizar el listado de referencias y comentarios de otros estudiantes previamente analizados; por último, visualizar las recomendaciones sobre datos históricos de alguna carrera o aspectos a mejorar del estudiante.

Estos requerimientos comprenden aquellas características del sistema que no son funciones pero que cumplen parte crucial dentro del sistema para poder ofrecer una experiencia óptima al usuario final, el estudiante. Estos se ven presentes en cada acción dentro del sistema por defecto y completan la esencia del producto mínimo viable a modo de aplicativo web. Se dividen de acuerdo a la categoría no funcional que abarcan de la siguiente manera:

- Usabilidad.
- Escalabilidad.
- Eficiencia.
- Seguridad, integridad y lógica de datos.

Centro informativo académico				
Id	Nombre	Descripción	Dependencias	Prerrequisitos
CIAC-01	Almacenar información académica general sobre estudiantes previos	El sistema debe permitir al estudiante visualizar información relevante sobre los cursos de parte de estudiantes universitarios	-	-
CIAC-02	Almacenar información académica general sobre cursos dentro de una carrera	El sistema debe permitir visualizar las carreras disponibles dentro de la Javeriana Cali en la Facultad de Ingenierías	-	-
CIAC-03	Almacenar información académica general sobre las carreras dentro de la facultad	El sistema debe permitir al estudiante vincular una carrera de la facultad de Ingenierías con su perfil	-	-
CIAC-04	Almacenar información académica general sobre la Pontificia Universidad Javeriana Cali	El sistema debe permitir al estudiante visualizar información relevante sobre las carreras de parte de estudiantes universitarios	-	-
CIAC-05	Mostrar información académica general sobre cursos dentro de una carrera	El sistema debe permitir al estudiante visualizar qué tan afín es con una carrera vinculada	CIAC-02	Debe existir información sobre los cursos de una carrera de la facultad de Ingenierías en el sistema
CIAC-06	Mostrar información académica general sobre las carreras dentro de la facultad	El sistema debe permitir al estudiante visualizar qué tan afín es con una carrera vinculada	CIAC-03	Debe existir información sobre las carreras de la facultad de ingenierías en el sistema
CIAC-06	Mostrar información académica general sobre la Pontificia Universidad Javeriana Cali	El sistema debe permitir al estudiante visualizar qué tan afín es con una carrera vinculada	CIAC-04	Debe existir información sobre la institución educativa en el sistema

Figura. 4.7: Requerimientos del módulo Centro Informativo Académico, fuente propia.

Requerimientos No Funcionales			
Id	Nombre	Descripción	Categoría
RNF-01	Tiempo de aprendizaje	El tiempo de aprendizaje del sistema por un usuario, deberá ser menor a 1 hora	Usabilidad
RNF-02	Documentación del sistema	El sistema debe contar con manuales de usuario estructurados o secciones FAQ (Frequently Asked Questions)	Usabilidad
RNF-03	Notificaciones del sistema	El sistema debe proporcionar mensajes informativos y orientados al usuario	Usabilidad
RNF-04	Presentación del sistema	El sistema debe presentar interfaces gráficas intuitivas y amables a la vista	Usabilidad
RNF-05	Tiempo de adaptación al crecimiento	El tiempo empleado en adaptar el sistema a nuevas tecnologías para su crecimiento debe ser menor al doble empleado su desarrollo inicial	Escalabilidad
RNF-06	Costo de adaptación al crecimiento	El costo empleado en adaptar el sistema a nuevas tecnologías para su crecimiento debe ser al menos 25% que el empleado en su desarrollo inicial.	Escalabilidad
RNF-07	Interoperabilidad de las partes del sistema	Los módulos del sistema manejan un grado de interoperabilidad aceptable una vez adaptado a una nueva arquitectura	Escalabilidad
RNF-08	Concurrencia del sistema	adecuadamente con hasta 100 peticiones de usuario.	Eficiencia
RNF-09	Rapidez del sistema	Toda funcionalidad del sistema, sin contar las predicciones debe responder al usuario en menos de 5 segundos	Eficiencia
RNF-10	Seguridad del sistema	Toda información almacenada por el sistema puede ser vista, modificada y/o eliminada por el administrador de la página	Seguridad, integridad y lógica de datos

Figura. 4.8: Requerimientos No Funcionales del sistema.

# Diseño del Software

---

## 5.1. Introducción

Este capítulo explicará el diseño del software construido para este proyecto. Empezando con la arquitectura del aplicativo web, pues representa una parte fundamental del sistema completo; argumentando la escogencia de su tipo y comparándola con un contexto actual donde se verán proyectos similares, así como entes organizacionales que compartan estas preferencias. Continuando con el diagrama de la base datos, el uso que se le da en el software, su naturaleza y, por qué fue escogida para este proyecto.

## 5.2. Arquitectura del Aplicativo

Para que este proyecto cumpla con los requerimientos funcionales y no funcionales anteriormente mencionados, la arquitectura del software que se diseñó fue del estilo API REST. Basada en un entorno enteramente web. De forma general, en la Figura 5.1 se puede observar la arquitectura general de todo el sistema de este proyecto. Esto anterior contemplando y diferenciando cada módulo necesario dentro del sistema.

Además, ejemplifica la manera en cómo se maneja la información entre el *Cliente* y el *Servidor*, la cual consta del uso de *APIs*. Estas se encargarán de gestionar el flujo de los datos mediante todo el sistema, es decir, que información le retorna el sistema al *Cliente* y garantizar que esta siempre sea coherente con lo que se esta solicitando. A su vez, esto permite el desarrollo separado de la parte *Back* y la parte *Front*. Más aún, es necesario el uso de un *API GATEWAY* con funcionalidad de interfaz para de esta manera redireccionar las diferentes peticiones realizadas por el estudiante al aplicativo web.

Para permitir una fácil escalabilidad del proyecto a nuevas demandas de rendimiento futuras, procurando que sea siempre óptimo a su vez que se mantiene un bajo acoplamiento, surge como primera opción el uso de una arquitectura de software por microservicios. Esta aproximación en primera instancia parece ofrecer todas las características necesarias, alta cohesión entre los diferentes módulos aún cuando estos manejan un bajo acoplamiento. Sin embargo, todo esto viene al precio de un alto costo de tiempo y recursos a la hora del despliegue en comparación con otras alternativas [32]. Es por esta razón que, si se desea implementar un sistema óptimo, escalable y que cumpla con todas las características anteriormente mencionadas pero que no presente alta complejidad en su

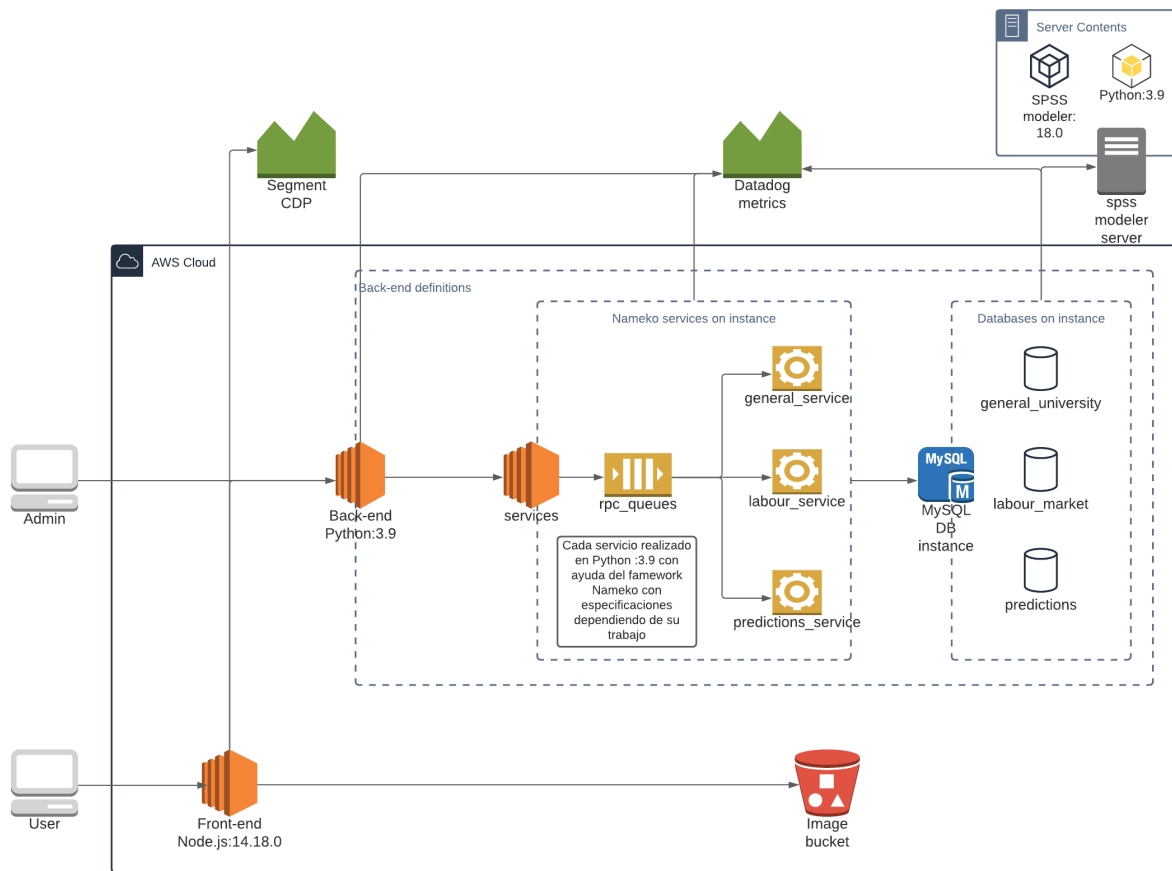


Figura 5.1: Diagrama de arquitectura y despliegue del sistema.

despliegue y distribución de recursos, la arquitectura monolítica modular es la mejor aproximación.

A diferencia de la arquitectura por microservicios, la monolítica modular permite emplear un proyecto con un equipo de trabajo pequeño, manejar una lógica de negocio simple como en el caso de este proyecto [32]. Más aún, su implementación en un solo archivo o directorio facilita las pruebas, depuración de errores y disminuye su tiempo de despliegue; todo esto sin la necesidad de emplear los microservicios [32].

Su diferenciación modular de la arquitectura tradicional monolítica permite la obtención de módulos independientes e intercambiables que a su vez dan lugar a una alta cohesión con un bajo acoplamiento. Además, al ser una elección estándar en la construcción de aplicaciones, cualquier equipo de ingenieros debe estar en capacidad de implementarla y mantenerla a través del tiempo [32]. Esto incrementa el soporte que puede brindar la comunidad internacional debido a su popularidad en el mercado en los últimos años, tal y como lo argumenta AppDynamics evidenciado en las figuras 7.5 y 5.3 [33].

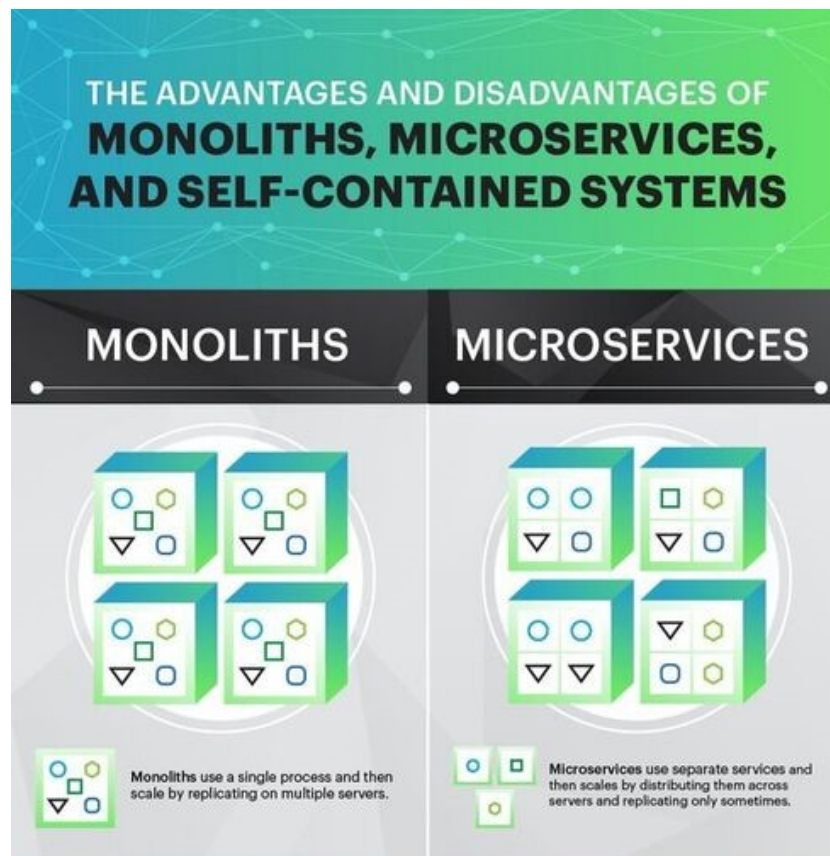


Figura 5.2: Comparación entre la arquitectura monolítica y por microservicios (parte 1) [33].



Figura 5.3: Comparación entre la arquitectura monolítica y por microservicios (parte 2) [33].

En su discusión sobre cómo migrar de una arquitectura monolítica a microservicios, AppDynamics empieza primero estableciendo ventajas y desventajas de cada una (Véase la figura 7.5). Aquí empiezan mencionando qué una arquitectura monolítica utiliza un único proceso y su forma de escalar es replicarlo en múltiples servidores. Por otro lado, la de microservicios como su nombre lo indica usa diferentes servicios y su escalamiento es distribuirlos en diferentes servidores, replicando solo ocasionalmente.

Después AppDynamics procede a mencionar las distintas cualidades (Véase la figura 5.3). Estas cualidades van desde los aspectos negativos hasta los positivos, empezando con el hecho de que la arquitectura monolítica permite: empezar un desarrollo rápido, evitar complicaciones a la hora de implementar funcionalidades que afectan todo el sistema y permitirle al desarrollador trabajar los problemas de desarrollo una única vez al ser de fácil manejo. Por otro lado, según AppDynamics la arquitectura por microservicios posibilita: manejar un bajo costo de mantenimiento y un bajo acoplamiento de sus componentes. Finalmente, se mencionan las desventajas de ambas arquitecturas; La arquitectura monolítica es difícil de cambiar y de escalar, mientras que la de microservicios requiere una alta automatización en la infraestructura y se le dificulta transferir la información entre

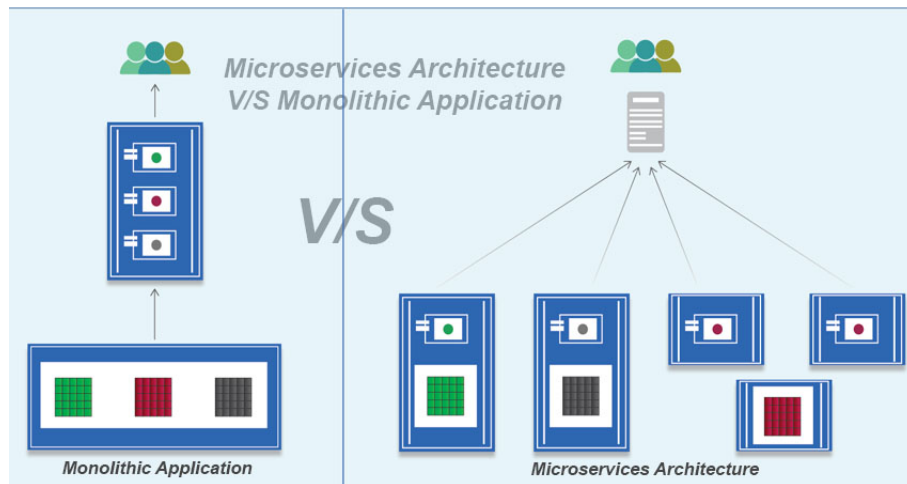


Figura 5.4: Comparativa en la distribución física entre arquitectura monolítica y por microservicios [34].

servicios de forma consistente.

Debido a lo anterior mencionado y a la complejidad inicial del proyecto, su objetivo es fácilmente alcanzable con la arquitectura de software monolítica modular. Así, en la búsqueda de permitir un sistema que pueda ir escalándose conforme pase el tiempo cada módulo del sistema puede ser asumido por un microservicio, permitiendo su adaptación a una arquitectura de microservicios modular de forma parcial o total en caso de que la situación así lo requiera [32] (Véase en la figura 5.4).

Para saber cómo se puede migrar de una arquitectura a la otra, es necesario entender como se distribuyen físicamente ambas. Tal y como lo muestra Dmitriy Nefedov (Véase la figura 5.4), las funcionalidades o módulos dentro de la arquitectura monolítica podrían distribuirse en diferentes microservicios con su respectivo flujo garantizando así un bajo acoplamiento y una migración paulatina [34].

### 5.3. Modelo de datos

La forma en como se manejaron los datos en el sistema de este proyecto fue mediante el uso de una base de datos relacional. Esto debido a que el flujo de la información era gestionado por este tipo de base de datos de forma óptima. La poca o nula existencia de datos vacíos, así como la naturaleza de cada tipo de dato son fácilmente almacenados por un gestor de datos relacional como lo es MySQL. La Figura 5.5 muestra el modelo relacional dentro la base de datos del sistema.

Dentro de la base de datos relacional, las principales tablas son:

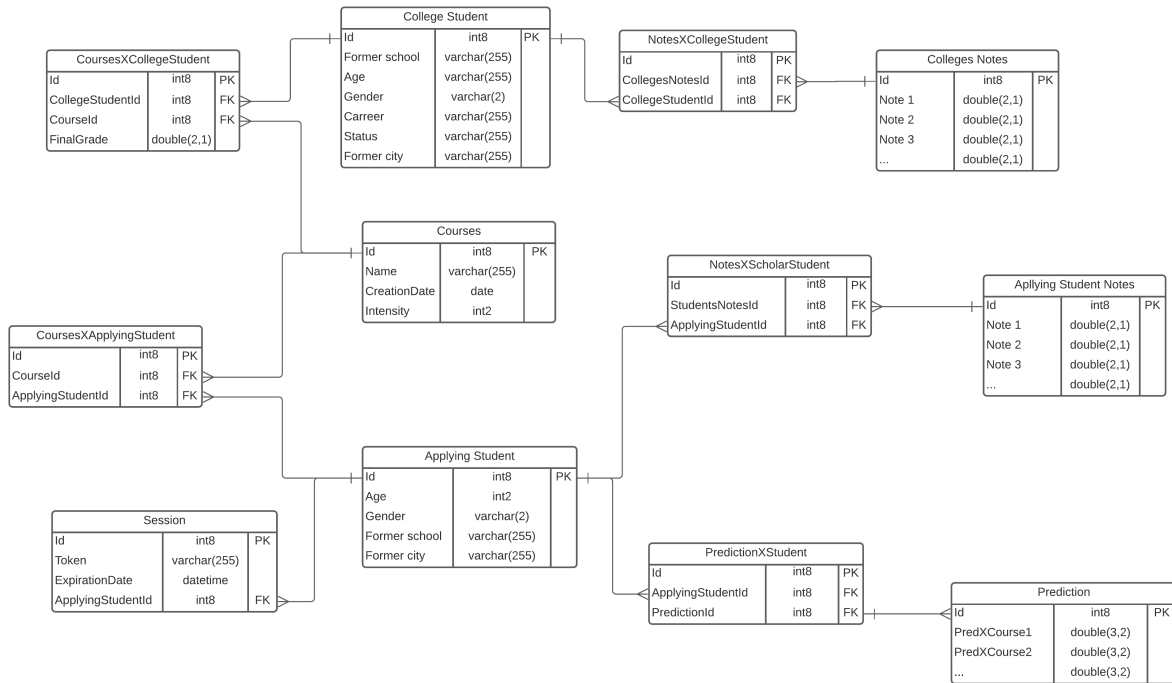


Figura 5.5: Diagrama de base de datos relacional del sistema.

- **Courses:** almacena información de los cursos brindados por la Universidad Javeriana Cali en la facultad de Ingeniería.
- **Applying Student:** almacena la información personal y datos relevantes a estudiar del estudiante próximo a ingresar en la Universidad Javeriana Cali.
- **College Student:** almacena la información personal y datos relevantes a estudiar del estudiante que cursó materias en la Universidad Javeriana Cali.
- **Applying Student Notes:** almacena las notas obtenidas en la educación media por el estudiante próximo a ingresar en la Universidad Javeriana Cali.
- **Colleges Notes:** almacena las notas obtenidas por el estudiante de la Universidad Javeriana Cali en los diferentes cursos.
- **Prediction:** almacena la predicción sobre si el estudiante próximo a entrar aprobará en cada curso de la de la carrera a elegir en la universidad.
- **Session:** almacena información general relacionada con la sesión de cada usuario dentro del aplicativo web.

# Tecnologías Involucradas en el Desarrollo del Sistema

---

## 6.1. Introducción

En este capítulo se argumentarán las decisiones tomadas con respecto a la escogencia de las diferentes tecnologías para el desarrollo del sistema. A su vez, lo conforman cinco partes: frontend, backend, bases de datos, analítica de datos e inteligencia empresarial. Dichas elecciones se justifican con la naturaleza que presentan tanto la arquitectura como el enfoque del proyecto.

## 6.2. Frontend

En el desarrollo de las diferentes vistas con las que interactúa el usuario (las interfaces web del usuario) se optó por una tecnología capaz de cumplir con requisitos funcionales, de calidad y de usuario especificados en sistema. Por esta razón se optó por el framework de JavaScript creado por Facebook, ReactJS [35].

Al ser un framework con naturaleza “Data Driven” que enfoca sus esfuerzos en el usuario y sus características en la resolución de necesidades de este mismo con sus diferentes estructuras de datos, es de esperarse que dicha tecnología pueda adaptarse a los requerimientos de gran cantidad de sistemas y sobre todo, al sistema construido para este proyecto al permitir un relativamente fácil flujo de información consistente entre los diferentes módulos [35]. Adicionalmente, ReactJS demuestra ser un framework bastante competitivo y popular en la comunidad desarrolladora, llegando a ocupar el puesto número dos en el ranking a nivel mundial [36]. Esto anterior se debe a que tanto su naturaleza declarativa, como su anatomía basada en componentes permiten la creación de interfaces complejas de una manera muy amigable, permitiendo así una vista modular completamente cohesiva y dejando una experiencia agradable en el desarrollo del sistema, particularmente el mencionado en este documento potenciando un arranque fácil y rápido en su construcción. [35].

La popularidad del framework en la comunidad de programadores debido a su gran curva de aprendizaje, así como su rendimiento excepcional en aplicaciones grandes posibilita el tener un fácil acceso a la comunidad de soporte en caso de algún inconveniente durante el desarrollo del aplicativo web y así poder dar con una solución en un tiempo óptimo [36]. Este constante soporte se eviden-

cia a través de la continua escogencia de esta tecnología para diferentes desarrollos en los últimos años tal y como se evidencia en la figura 6.1, así como su uso a nivel corporativo en la figura 6.2 [37].

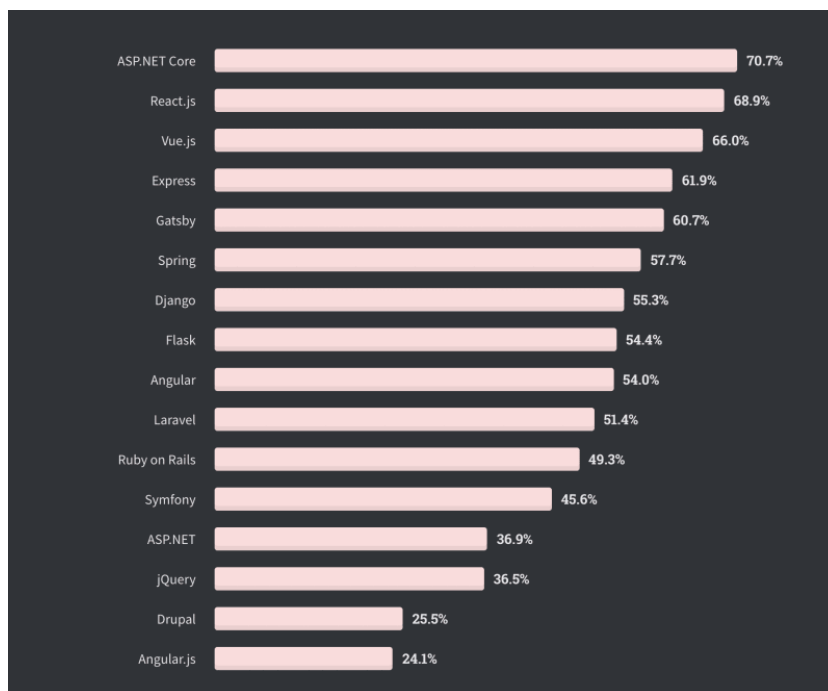


Figura 6.1: Popularidad del framework ReactJS en la comunidad desarrolladora [36].

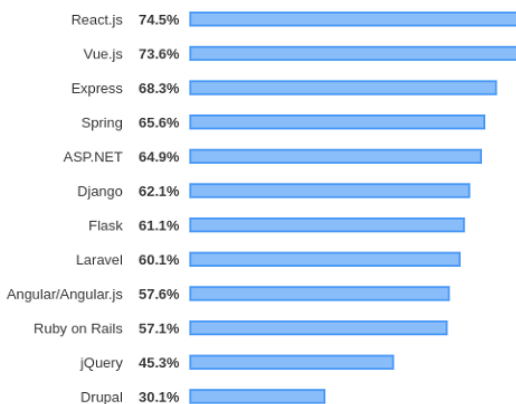


Figura 6.2: Demanda del framework ReactJS en el sector corporativo frente a otras alternativas [37].

## 6.3. Backend

Observando la naturaleza que presenta este aplicativo de crecer con facilidad debido a los volúmenes de información con los que puede llegar a trabajar en un futuro, es necesario elegir un lenguaje de programación que se encuentre en capacidad de manejar esto. Es por esta razón principal que se optó por usar Python. Un lenguaje de programación de código abierto desarrollado, probado y avalado por una amplia comunidad multicultural de programadores que apoyan la fácil distribución de este mismo a través de cualquier plataforma para facilitar su uso e interoperabilidad [38].

Su curva de aprendizaje rápida frente a otros lenguajes de programación se fundamenta en su característica diferenciadora: ser un lenguaje interpretado. Esto le permite tener una naturaleza de alto nivel y ser orientado a objetos, así como manejar semánticas de alto nivel [38]. Esto se traduce en la capacidad de asimilar en un alto grado el lenguaje humano para la construcción rápida de aplicativos, a través de estructuras de datos de alto nivel incorporadas en el lenguaje, generando una mayor atracción por su uso [38]. La Figura 6.3 muestra como Python impulsa a los diferentes desarrolladores a buscar Python después de tener un primer acercamiento básico en comparación con otros lenguajes de programación [39].

### Most Loved, Dreaded, and Wanted Languages



Figura. 6.3: Preferencia de los diferentes lenguajes de programación en los últimos años [39].

STXNext muestra muy claramente (Véase la figura 6.3) cuál es la preferencia en los lenguajes de programación y sobre todo cuál es el ranking de Python como tecnología de backend frente a sus competidores debido a las cualidades mencionadas anteriormente en esta sección. En esa figura se evidencian fuertes competidores por sus diferentes cualidades, sin embargo, para requisitos de aplicativos webs, Python sobresale con creces [39]; esto fue crucial a la hora de su escogencia para el desarrollo pues vuelve y surge la necesidad de poder acceder a un soporte constante en caso de que algún problema surja y deba ser escalado.

Con aplicativos similares al que se presenta en este trabajo, es necesario mantener el software a través del tiempo y garantizar que esto no sea una tarea compleja. Python cuenta con una sintáxis fácil de aprender que trata de asimilar en un alto grado la del vocabulario humano, aumentando su legibilidad y reduciendo así los costos de mantenibilidad. Permite el uso de módulos y paquetes lo cual da pie a la programación modular y reutilización de código [38]. Más aún, al ser un lenguaje interpretado el paso de compilación es prácticamente omitido a vista del programador, permitiendo así que el ciclo de programación, prueba y depuración sea increíblemente rápido. De esta manera, resuelve la preocupación de un costo elevado de mantenimiento [38].

Adicionalmente, es necesario procurar el mantenimiento no solo entre los desarrolladores iniciales del aplicativo designado, sino también a través de las diferentes generaciones de programadores que puedan llegar a appyar el proyecto. En esencia, es necesario lograr una fácil migración de conocimiento entre los diferentes equipos. Para este caso, Python garantiza un apoyo continuo a través de su depurador propio, desplegando a vista del programador errores de una forma más amigable y, en caso de que no se logre identificar la causa permite rastrear el error desde los último pasos hasta los que el programa ejecutó correctamente [38].

En la Figura 6.4 se evidencia el acompañamiento de parte de la comunidad al lenguaje de programación Python a través de la resolución de dudas frente a su competencia al finalizar el año 2019.

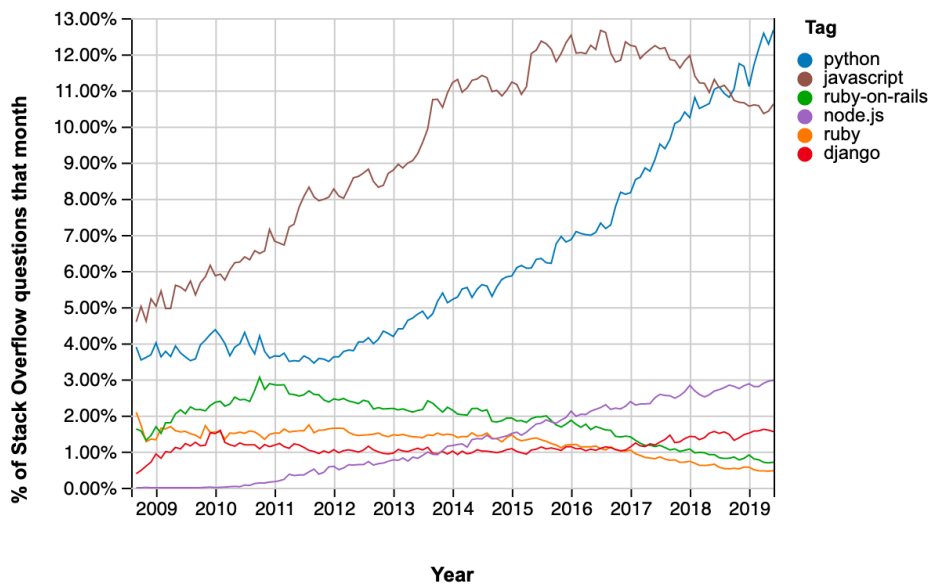


Figura. 6.4: Tendencias en soporte técnico sobre los diferentes lenguajes de programación [40].

## 6.4. Business Intelligence

En el desarrollo y mantenimiento de todo sistema se presenta la necesidad de tener un control acerca del rendimiento de cada proceso que hace parte de este, esto nos permite caracterizar la experiencia de usuario, trazar y diagnosticar cuáles pueden ser las fallas de nuestro sistema, para esto sirven los Application Performance Management, herramientas que nos permitirán el sistema a la necesidad del cliente y los requisitos de este sistema de una mejor manera, optimizando muchos procesos necesarios para el desarrollo. Datadog[41] es un servicio de monitorización en la nube que proporcionan distintas características importantes que ayudan en el proceso de desarrollo, solución de crisis y mantenimiento de infraestructuras de una manera completa por medio de distintos dashboards que nos permiten observar todo la infraestructura en un mismo lugar, permitiéndonos obtener información en un lugar centrado como trazas de logs entre todos los componentes de la arquitectura, base de datos, servicios, entre otros. Básicamente nos permite analizar peticiones desde su inicio a fin a través de servicios distribuidos como lo sería una arquitectura de microservicios. Así mismo, nos permite hacer seguimiento sobre el rendimiento de todos los servicios pertenecientes al sistema por medio de vistas personalizadas donde se resume su rendimiento bajo parámetros de aceptación predefinidos como latencia, porcentaje de errores, entre otros. Además de esto, nos permite realizar pruebas o obtener métricas acerca de la usabilidad del servicio de front end y su rendimiento, verificando rutas críticas de la usabilidad del sistema, visualizando tiempos de carga de recursos, páginas, y ver correlación con usuarios concurrentes y sus interacciones con cualquier

flujo dentro del sitio.

Este servicio de business intelligence o básicamente gestor de rendimiento para nuestro sistema nos ayuda básicamente en la realización de múltiples pruebas de rendimiento y usabilidad como se puede observar con la descripción de sus capacidades, sumado a la necesidad de tener un lugar centralizado para la administración y desarrollo de un sistema con una arquitectura basada en microservicios, donde el manejo del contexto para tratar bugs se puede ver desdibujado por la distribución del sistema, el uso de esta tecnología permitirá a su vez obtener métricas acerca de la usabilidad del sitio y demás pruebas necesarias para asegurar el cumplimiento de muchos requisitos no funcionales del sistema y posibles desarrollos entorno a la escalabilidad del sistema cuando se agreguen nuevos módulos al sistema.

## 6.5. Bases de Datos

Con respecto al almacenamiento de los datos con los que trabajará el aplicativo se optó por usar MySQL, el cual es un sistema de gestión de bases de datos relacional. Esta herramienta manejará:

- Datos básicos del perfil del estudiante que ingresa al aplicativo
- Notas de educación media del estudiante que ingresa al aplicativo
- Notas de educación media de los diferentes estudiantes activos o inactivos dentro de las diferentes carreras
- Comentarios respecto a una carrera de los diferentes estudiantes activos o inactivos
- Resultados finales en asignaturas específicas dentro de la carrera de los diferentes estudiantes activos o inactivos

Su naturaleza de código abierto permite que el mantenimiento de los datos a través del tiempo no resulte costoso, pues es posible usar y modificar los datos a través de dicho motor sin mayor complicación ni costos adicionales [42]. Su gran compatibilidad con los entornos que manejan una dinámica cliente-servidor, así como sistemas embebidos permite con facilidad un funcionamiento multihilos garantizando a su vez un rendimiento eficiente con grandes volúmenes de datos [42]. Más aún, para un sistema como el que se muestra en este trabajo donde el volumen de los datos puede aumentar de forma acelerada con el pasar del tiempo, se vuelve necesario guardar toda esta información en una ubicación remota donde se encuentren seguros. Para estas situaciones la alternativa es MySQL Server que al ser rápido, confiable, escalable y fácil de usar tanto en un computador de escritorio como en un portátil, aumentando así su portabilidad [42].

Adicionalmente, MySQL demuestra como gestor de bases de datos demuestra ser la segunda preferencia a nivel de la comunidad desarrolladora, donde el 53% de casi 42.000 programadores prefiere MySQL sobre otros motores (Véase en la Figura 6.5).

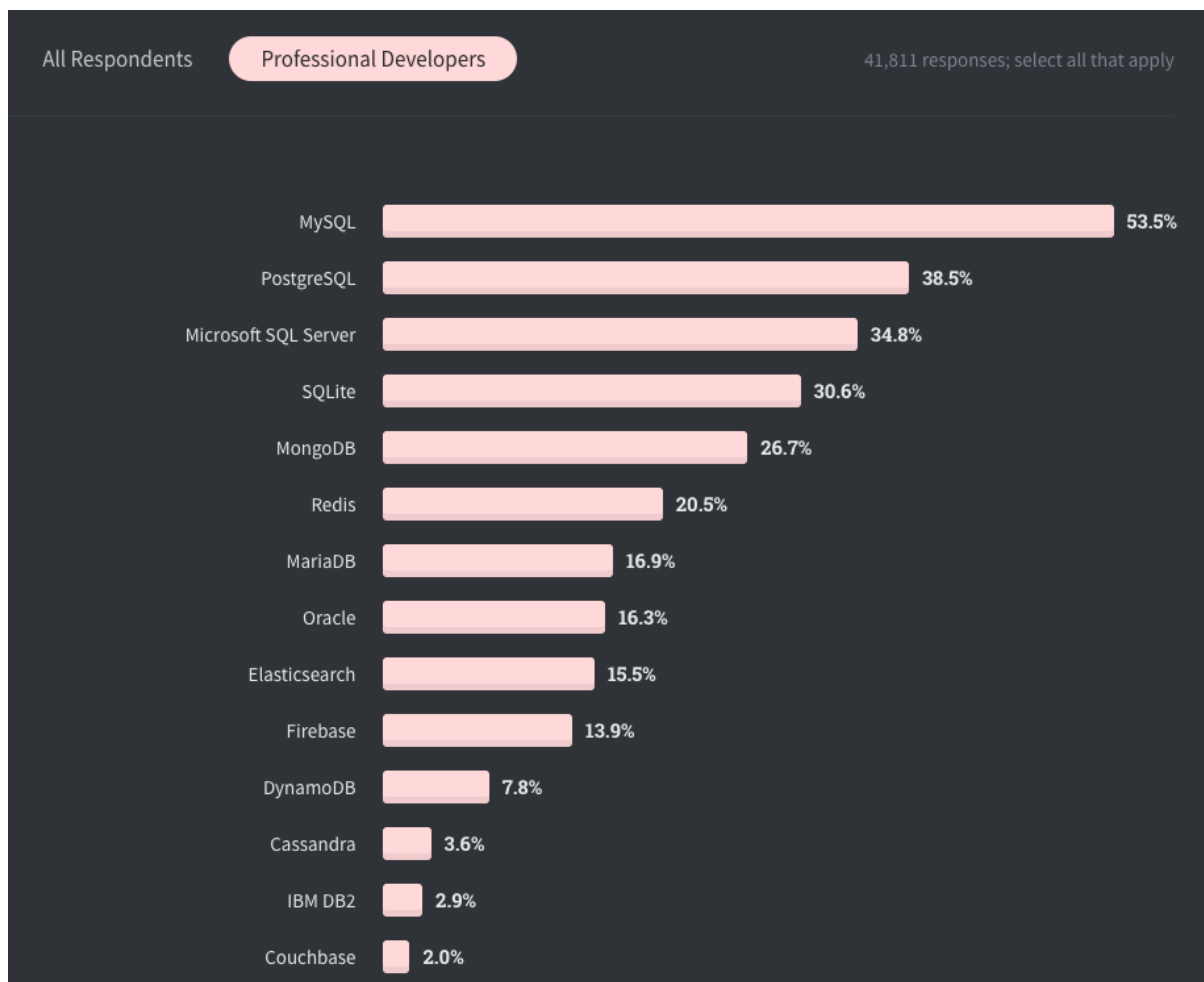


Figura. 6.5: Top de los gestores de bases de datos preferidos en la comunidad desarrolladora [43].

## 6.6. Modelo Estadístico

Para el modelamiento de los datos a través de técnicas estadísticas se decidió utilizar la herramienta IBM SPSS MODELER. Esta herramienta cuenta múltiples características que la diferencia de otras como Spark, en que cuenta con soporte continuo una vez adquirido el software. Cuenta con una alta conectividad a la nube y sobre todo un alto nivel de seguridad, integridad, fidelidad y confidencialidad en los datos. De esta forma no solo mantienen los datos protegidos, sino que también se garantiza una conectividad continua a través de cada entorno creado, permitiendo también una alta modularidad [44].

## 6.7. Generador de Información Aleatoria

De acuerdo con la recomendación del director Gerardo Sarria sobre trabajar con información simulada, y en consideración de la necesidad de garantizar credibilidad, confiabilidad, coherencia y continuidad en la información, se optó por utilizar Mockaroo [45]. Este aplicativo web contaba con todas las características mencionadas, además de especificar el tipo de dato de cada variable e incluso permitía generar datos para entrenamiento de modelos a diferencia de otras alternativas. Adicionalmente, contaba con la particularidad de ser código abierto lo cual facilitaba su uso [45].

# Implementación del Software

## 7.1. Prototipado de pantallas

Continuando con el desarrollo del sistema, se pasa al desarrollo de prototipos de pantalla y flujos, sobre los cuales los participantes del proyecto tuvieron opinión en todo momento para mejoras y definición de los mismos. para la realización de estos prototipos y/o mock ups se hizo uso de la herramienta de MarvelApp[46] la cual permitió un desarrollo sencillo y tener una visualización de cada parte de estas de una manera mucho más fácil para los integrantes. así mismo esta herramienta permite realizar un flujo directo de la aplicación entre botones y pantallas, dando un sentido de funcionalidad y prototipo funcional.

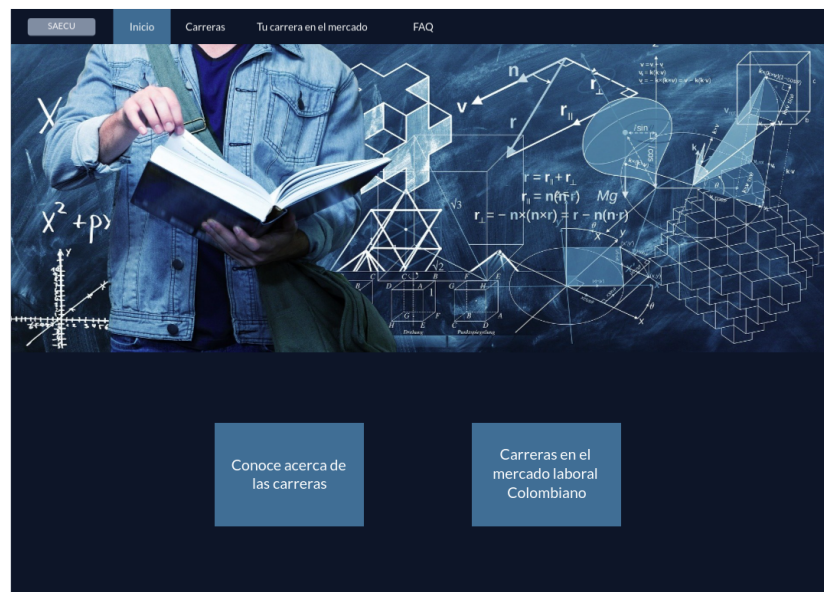


Figura 7.1: Vista de pantalla principal del aplicativo.



Figura 7.2: Vista introductoria al apartado de carreras.



Figura 7.3: Expansión de información sobre el programa.

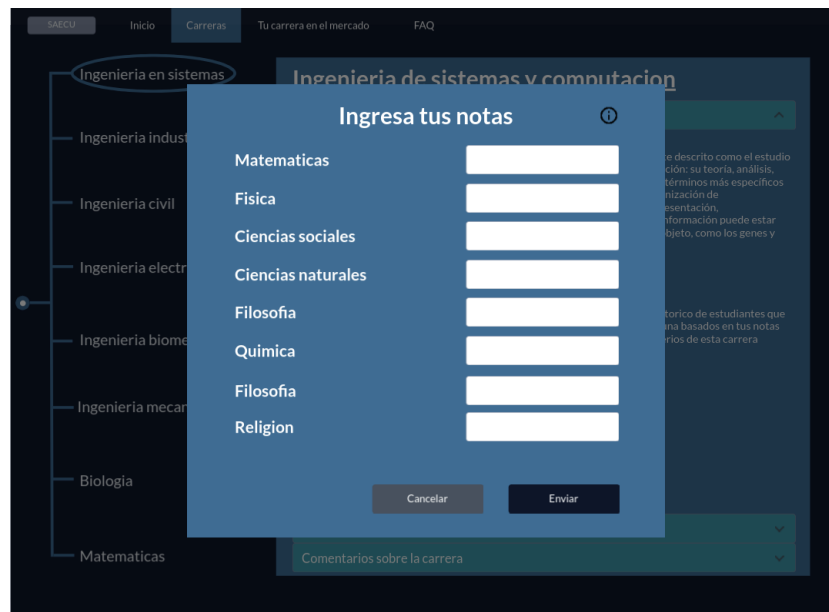


Figura 7.4: Expansión sobre agregar notas para predicción.



Figura 7.5: Expansión de información sobre el programa con predicción realizada.



Figura 7.6: Expansión de cursos pertenecientes al programa.



Figura 7.7: Expansión de comentarios pertenecientes al programa.

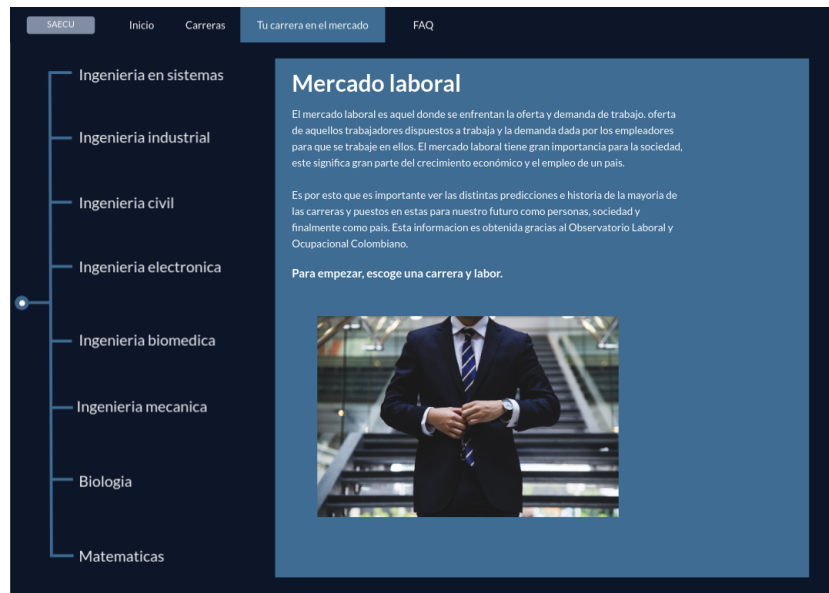


Figura 7.8: Vista introductoria al apartado de mercado laboral.

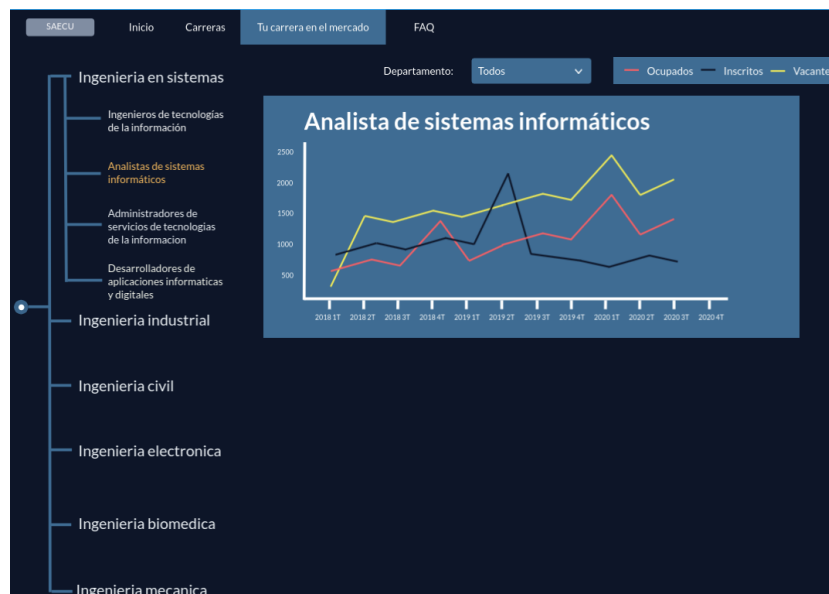


Figura 7.9: Expansión de profesiones asociadas a carreras y datos después de su selección (una seleccionada).

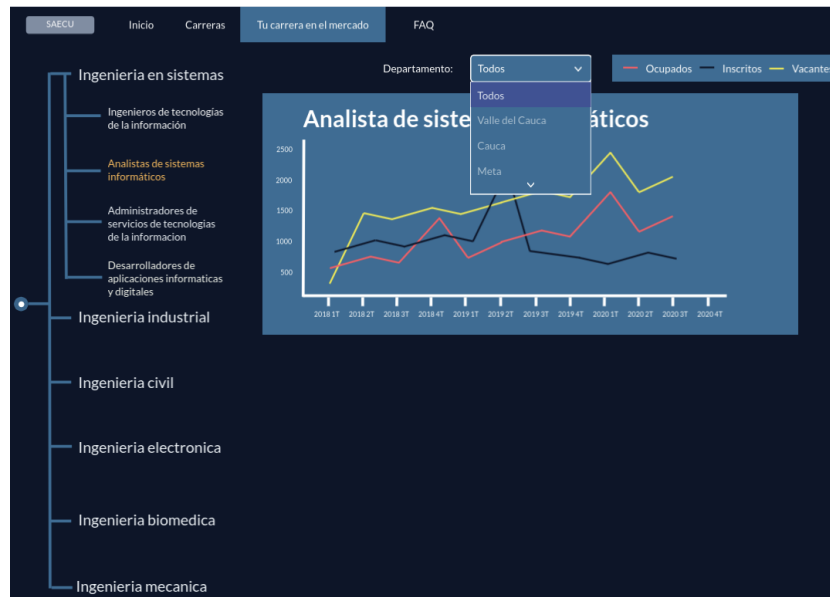


Figura 7.10: Expansión de filtro para estadística de datos seleccionados.



Figura 7.11: Expansión de profesiones asociadas a carreras y datos después de su selección(dos seleccionadas).



Figura 7.12: Vista del apartado de preguntas frecuentes.

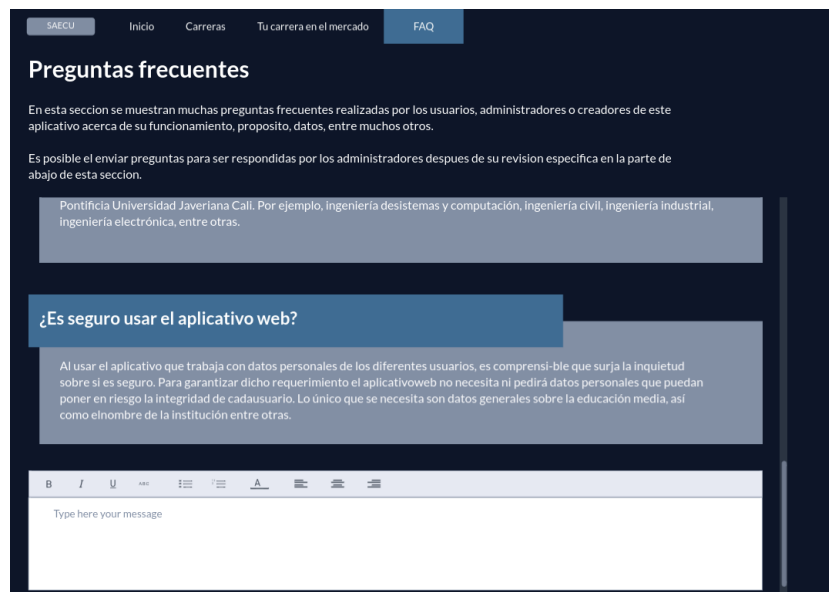
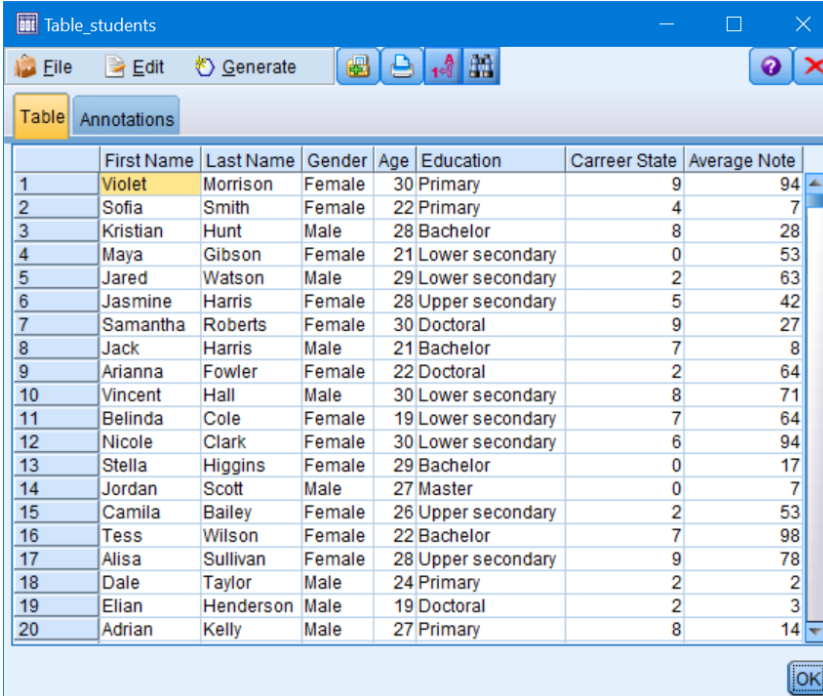


Figura 7.13: Expansión sobre envío de una pregunta frecuente.

## 7.2. Modelo estadístico

Continuando con la metodología de desarrollo y una vez finalizada la recolección y delimitación de la información necesaria para el correcto funcionamiento del modelo, en esta etapa se empieza el desarrollo del modelo estadístico predictivo. Esto anterior, haciendo uso de la tecnología de IBM SPSS Modeler definida previamente en la etapa de planificación (Véase 6.1). A continuación se presenta de forma más detallada los diferentes componentes del modelo definidos en 3.1.



	First Name	Last Name	Gender	Age	Education	Career State	Average Note
1	Violet	Morrison	Female	30	Primary	9	94
2	Sofia	Smith	Female	22	Primary	4	7
3	Kristian	Hunt	Male	28	Bachelor	8	28
4	Maya	Gibson	Female	21	Lower secondary	0	53
5	Jared	Watson	Male	29	Lower secondary	2	63
6	Jasmine	Harris	Female	28	Upper secondary	5	42
7	Samantha	Roberts	Female	30	Doctoral	9	27
8	Jack	Harris	Male	21	Bachelor	7	8
9	Arianna	Fowler	Female	22	Doctoral	2	64
10	Vincent	Hall	Male	30	Lower secondary	8	71
11	Belinda	Cole	Female	19	Lower secondary	7	64
12	Nicole	Clark	Female	30	Lower secondary	6	94
13	Stella	Higgins	Female	29	Bachelor	0	17
14	Jordan	Scott	Male	27	Master	0	7
15	Camila	Bailey	Female	26	Upper secondary	2	53
16	Tess	Wilson	Female	22	Bachelor	7	98
17	Alisa	Sullivan	Female	28	Upper secondary	9	78
18	Dale	Taylor	Male	24	Primary	2	2
19	Elian	Henderson	Male	19	Doctoral	2	3
20	Adrian	Kelly	Male	27	Primary	8	14

Figura 7.14: Representación en formato tabla de la información usada dentro del modelo, fuente propia.

En la figura 7.14 se puede observar la información usada dentro del modelo predictivo. Para el caso específico se visualizan los datos que hacen de sustitos a los originalmente esperados de parte del CSI. Los diferentes nombres de las columnas representan las variables que identifican la información general de cada estudiante que pasó por la carrera de la facultad de Ingeniería y Ciencias. Estas son: Nombre, Apellido, Género, Edad, Educación, Estado de la Carrera, Promedio de Notas respectivamente. Cabe aclarar que variables como el estado de la carrera y el promedio de notas son modificadas en un paso posterior para una fácil interpretación.

En la figura 7.15 se puede apreciar cómo se realiza una de las primeras modificaciones a la información dentro del modelo. A través del nodo filtro, se busca evitar que ciertos campos que se consideran innecesarios al momento de visualizar mejor la información continúen en el flujo; esto

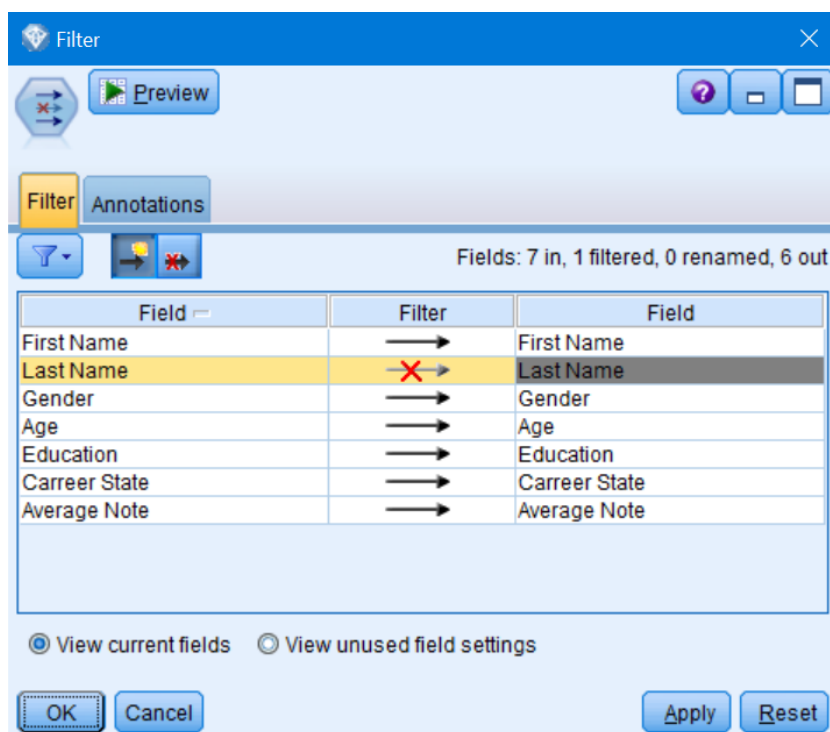


Figura 7.15: Filtrado del campo *Last Name* (Apellido en español) a través del nodo filtro, fuente propia.

debido a su poca relevancia dentro del objetivo final que vienen siendo las predicciones. En este caso específico se filtra el campo Apellido (*Last Name* en inglés) pues su contribución a las predicciones es nula. El resto de los campos continúan dentro del flujo y por tanto si ícono no cambia.

Una vez obtenidos los campos principales para el funcionamiento correcto del modelo se procede a verificar el tipo de datos que maneja cada uno. Esto se hace principalmente para validar si la naturaleza de cada variable es fácilmente interpretable o no. En la figura 7.16 por ejemplo se deriva la variable original que maneja un tipo de dato estilo nominal a una nueva variable que maneje datos binarios o estilo bandera (1 y 0); de esta forma el modelo puede diferenciar el umbral de cada categoría, donde 1 significa *Hombre* y 0 significa *Mujer*. Este procedimiento se repite con las variables *Education*, *Career State* y *Average Note* (Nivel de educación, estado de la carrera y nota promedio, respectivamente) hasta que todas queden con tipo de dato binario para diferenciar fácilmente el estado de cada campo o variable.

Posterior a la generación de nuevos campos con tipos de datos adecuados al modelo a partir de los ya existentes, es necesario filtrar las variables originales debido a su irrelevancia dentro del modelo y su posible afectación en las predicciones. Se procede entonces a filtrarlas, es decir, evitar su aparición futura dentro del flujo. La figura 7.17 muestra cómo se realiza esto, manteniendo las

nuevas variables y cancelando las originales; la equis sobre la flecha simboliza esta acción.

Para este punto todos los campos con los que se cuenta dentro del modelo son los necesarios, sin embargo, aún falta validar que la información dentro de cada variable es aceptable; esto es, que cada registro tenga información legible, entendible e interpretable. En la figura 7.18 se visualiza la forma por la cual aquellos valores que podrían afectar las predicciones del modelo son descartados. Estos descartes son hechos después de revisar minuciosamente todos los registros, dividiéndolos en categoría según los valores que toman en las variables principales que son: GENDER, EDUCATION, STATE y AVG NOTE. Para el caso específico de la figura 7.18 se tomó EDUCATION y se determinó que valores como ‘Doctoral’ ‘Máster’ y ‘Primary’ son irrelevantes en el modelo, no obstante, el proceso se repite con los 4 campos principales antes mencionados.

Para el primer modelo estadístico denominado CARMA (Véase 3.1), se utilizan 3 variables principales: EDUCATION, STATE y AVG NOTE. Estas variables son las que se pueden observar en la figura 7.19 donde la palabra *Input* ordena que sean las únicas a recibir o entradas, mientras que aquellas que dicen *None* son ignoradas para efectos de tener mejores predicciones.

La figura 7.20 muestra los resultados obtenidos por el primer modelo estadístico CARMA; estos resultados son obtenidos dependiendo de las variables entradas definidas en la figura 7.19. Con este modelo se intenta visualizar cómo y qué tanto se afectan entre si los campos principales. Se puede apreciar entonces que STATE que significa el estado de la carrera de un estudiante cualquiera, se ve afectado por EDUCATION, es decir, el nivel de educación de este. De ahí que el estado en la carrera aparezca como consecuente (Consecuente en inglés) y su nivel de educación como antecedente (Antecedent en inglés). ‘Support %’ Significa que tanta probabilidad hay de que todos los antecedentes ocurran y ‘Confidence %’ hace referencia al nivel de certeza o confianza que se tiene de este resultado o también la probabilidad de que ocurra el consecuente tal que ocurrió el antecedente. Finalmente, ‘Lift’ vendría siendo y ‘Confidence %’ sobre la probabilidad total de que ocurra el consecuente.

Para el segundo modelo estadístico denominado REGRESIÓN LINEAL (Véase 3.1), se utilizan 4 variables principales predictoras: GENDER EDUCATION, STATE y AVG NOTE y 1 variable objetivo. Cabe aclarar que las 5 variables son continuas para efectos de que la REGRESIÓN LINEAL realice sus predicciones correctamente. Estas variables son las que se pueden observar en la figura 7.21 donde la palabra *Input* ordena que sean las únicas a recibir o entradas, mientras que aquellas que dicen *None* son ignoradas para efectos de tener mejores predicciones. De igual manera la variable que contiene de Rol ‘Target’ es la que se visualizará como objetivo dentro del modelo.

En la figura 7.22 se evidencia el resultado del modelo de regresión lineal. Este gráfico lo que indica es el grado de importancia de cada variable importante sobre cada predicción, demostrando que, para efectos de los datos tomados, la variable que más influye en el estado de la carrera de un estudiante es el género. O, dicho de otra forma, de los estudiantes que aprobaron una carrera, gran

parte pertenece a uno de los 2 géneros. A su vez, se evidencia el grado de importancia de las otras variables predictoras como lo son la edad, el promedio de notas y su nivel de educación.

En la figura 7.23 se observa la importancia o significancia de cada variable predictora dentro del modelo. Como las variables predictoras escogidas que son EDAD, GÉNERO, ESTADO EN LA CARRERA, EDUCACIÓN Y PROMEDIO DE NOTAS puede que no contribuyan mucho en el modelo, es necesario realizar una prueba de su significancia dentro del modelo y así corroborar que en efecto fueron escogidas correctamente. Para esto se utiliza el modelo de regresión lineal en la figura 7.23; de esta forma se asegura que las predicciones serán acertadas.

Para el tercer modelo estadístico denominado REGRESIÓN LOGÍSTICA (Véase 3.1), se utilizan 3 variables principales predictoras: GENDER, AGE y AVG NOTE y 1 variable objetivo. Cabe aclarar que las 4 variables son continuas para efectos de que la REGRESIÓN LOGÍSTICA realice sus predicciones correctamente. Estas variables son las que se pueden observar en la figura 7.24 donde la palabra *Input* ordena que sean las únicas a recibir o entradas, mientras que aquellas que dicen *None* son ignoradas para efectos de tener mejores predicciones. De igual manera la variable que contiene de Rol 'Target' es la que se visualizará como objetivo dentro del modelo.

Finalmente, para poder determinar que la variable predictora *Educación* (EDUCATION en inglés) permitiera que el estado en la carrera de un estudiante fuese (Aprobado), se encontró que la edad fue la segunda variable predictora más importante en el modelo y la más influyente en la *Educación*, tal y cómo se evidencia en la figura 7.25. El objetivo principal de este modelo final es dar validez a la escogencia de la educación como variable predictora, así como visualizar bajo que circunstancias cae la variable objetivo sobre la categoría deseada. De esta forma se garantiza que las decisiones tomadas sean las más acertadas y sobre todo que aún con datos simulados el modelo cumple con su objetivo sin sesgo alguno.

### 7.2.1. Análisis e integración de los resultados en el sistema

Tomando como referencia lo mencionado en 3.5 inicialmente en el modelo CARMA 7.20, se toma como predicción definitiva, aquella que tenga el porcentaje de soporte más alto (Véase 3.5) que para el caso viene siendo el registro con el valor de 55.89. Sin embargo, como puede haber varios registros con el mismo valor en el porcentaje entonces se usa como segundo filtro el *Lift*. De esta forma se garantiza un único resultado que viene siendo el que tiene como consecuente el estado y de antecedente el promedio de las notas.

No obstante, estos resultados pueden no ser del todo correctos y aquí es donde entran a juego los otros dos modelos: regresión lineal y regresión logística. Cabe resaltar que dichos modelos cumplen el papel único y exclusivo de dar a los desarrolladores, validez y seguridad sobre el resultado escogido. Es decir, son modelos que dan soporte a la elección y están para dar mejor entendimiento sobre la predicción, más no ingresan al sistema como tal. Son de uso exclusivo y por tal podría decirse

que son independientes a las demás partes del sistema. El análisis que se les da se describe más detalladamente en 3.5, pero en esencia a través de estos se busca verificar cuales son las variables más influyentes en el modelo y si el modelo tiene veracidad en las predicciones como tal, esto último se puede verificar en 7.23.

Finalmente, la integración a las demás partes del sistema la asume Python desde Django iniciando el modelo a través de un comando en una consola de servidor Windows y retomando los resultados desde la base de datos. Estos resultados han sido almacenados ahí previamente por el mismo modelo a través de una funcionalidad interna que permite exportar directamente en SQL y, como cada registro tiene un número de identificación único (asociado al token de sesión de cada estudiante), creando así un índice en la base de datos entonces cada predicción es fácilmente asequible y relacionable con cada usuario [47].

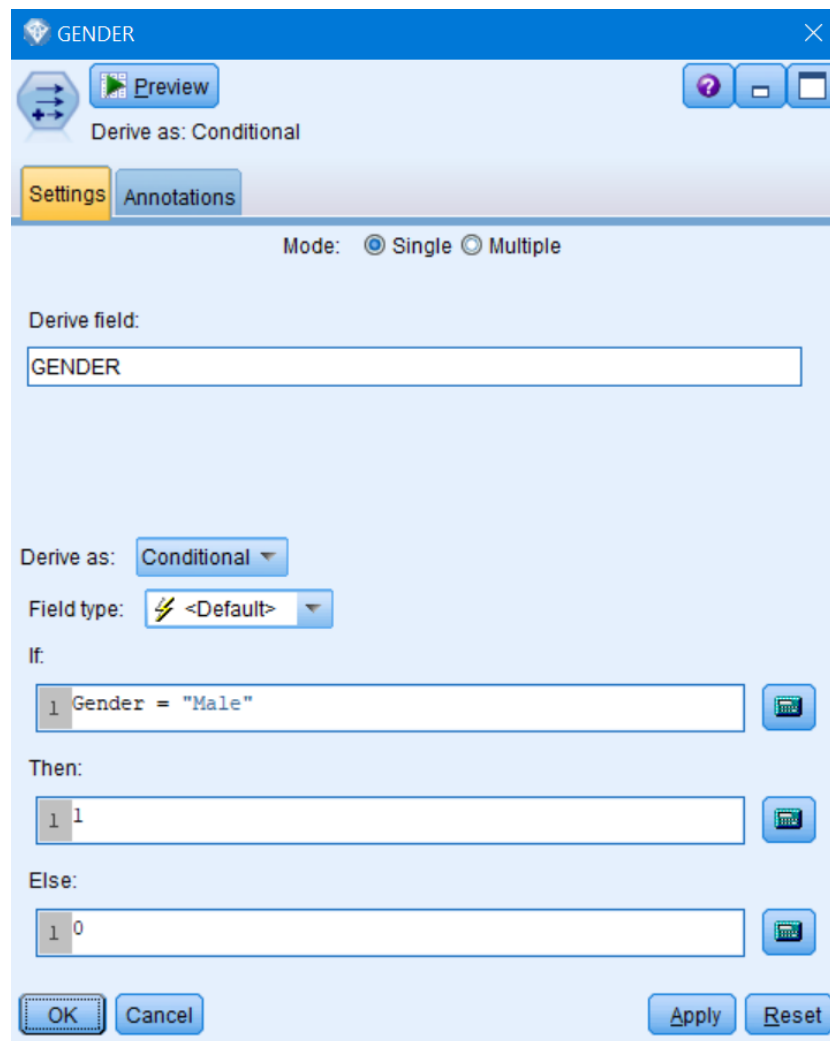


Figura 7.16: Derivación del campo original *Gender* (Género en español) en un nuevo campo binario 'GENDER', fuente propia.

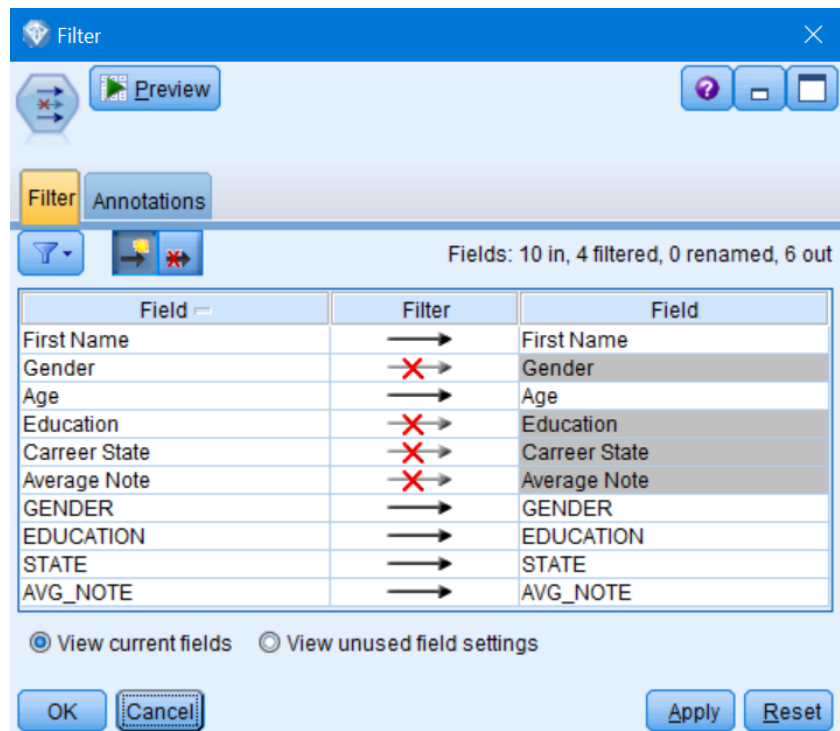


Figura 7.17: Filtrado de campos ya irrelevantes (*Education*, *Career State* y *Average Note*) a través del nodo filtro, fuente propia.

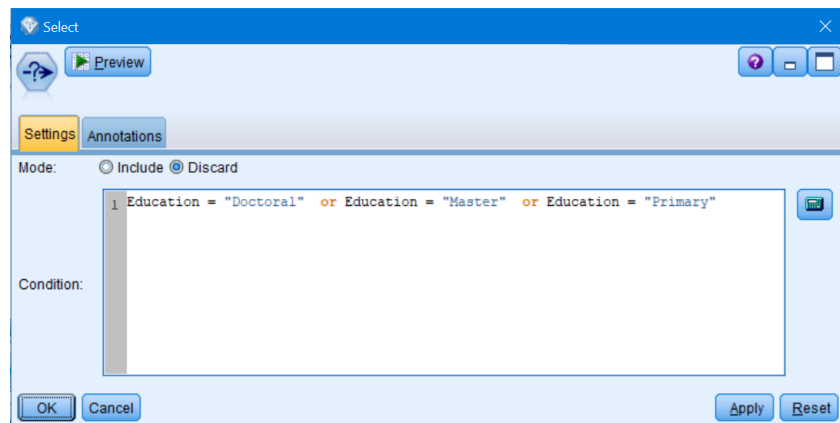


Figura 7.18: Filtrado de campos ya irrelevantes (*Education*, *Career State* y *Average Note*) a través del nodo filtro, fuente propia.

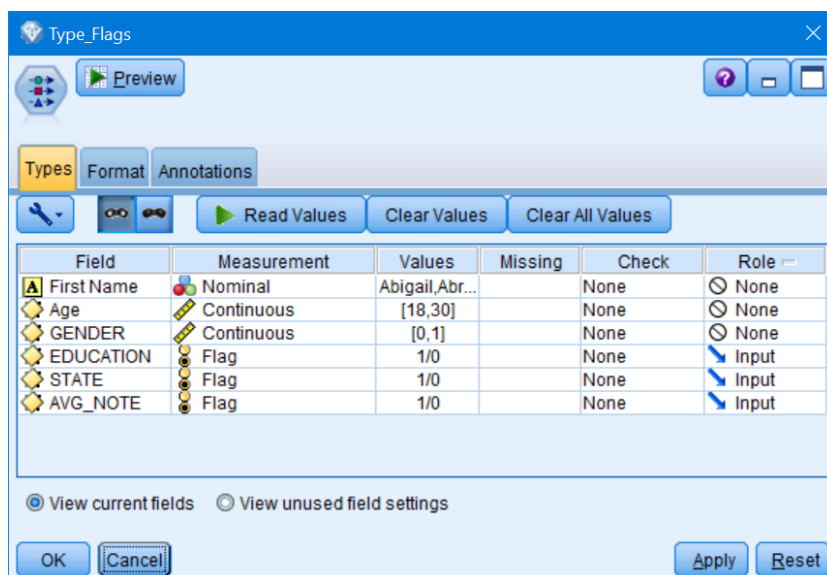


Figura 7.19: Definición de roles para cada variable dentro del primero modelo estadístico a través del nodo tipo, fuente propia.

Consequent	Antecedent	Support %	Confidence %	Lift
STATE	EDUCATION	41.644	40.789	0.788
EDUCATION	STATE	51.781	32.804	0.788
AVG_NOTE	STATE	51.781	43.915	0.786
STATE	AVG_NOTE	55.89	40.686	0.786
EDUCATION	STATE	51.781	12.169	0.766
STATE	EDUCATION	15.89	39.655	0.766
AVG_NOTE	EDUCATION	41.644	38.158	0.683
EDUCATION	AVG_NOTE	55.89	28.431	0.683
EDUCATION	STATE	22.74	27.711	0.665
STATE	EDUCATION	41.644	15.132	0.665
EDUCATION	AVG_NOTE	55.89	11.275	0.664
STATE	EDUCATION	16.986	37.097	0.664

Figura 7.20: Resultados obtenidos por el primero modelo estadístico CARMA (Véase 3.1), fuente propia.

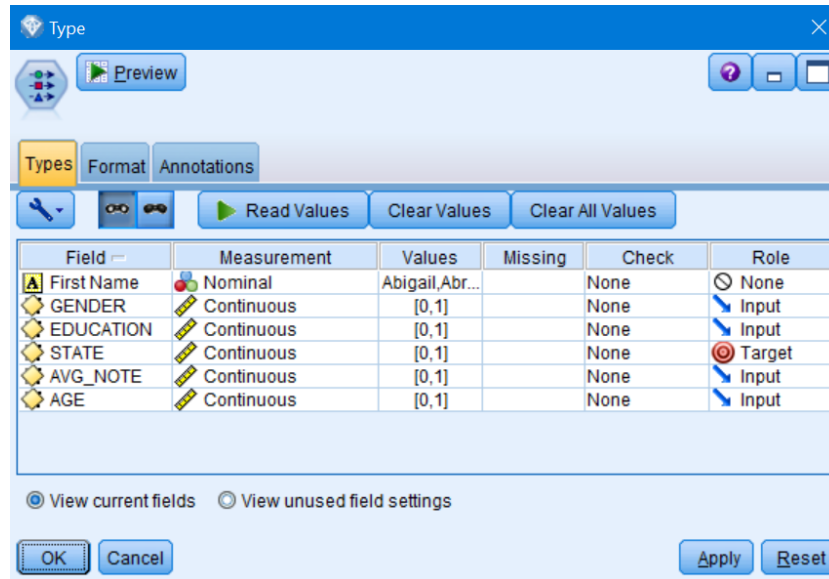


Figura 7.21: Definición de roles para cada variable dentro del segundo modelo estadístico a través del nodo tipo, fuente propia.

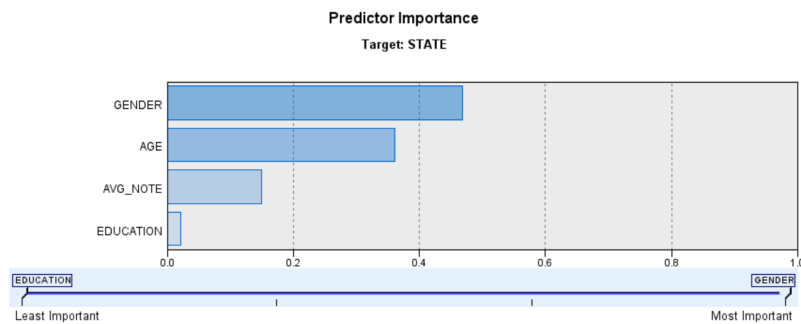


Figura 7.22: Grado de importancia de cada variable predictora dentro del modelo, fuente propia.

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	GENDER	.215	.040	.250	5.399	.000
	EDUCATION	.126	.049	.113	2.562	.011
	AVG_NOTE	.184	.043	.191	4.277	.000
	AGE	.148	.041	.175	3.585	.000

Figura 7.23: Resultados obtenidos por el segundo modelo estadístico Regresión lineal (Véase 3.1), fuente propia.

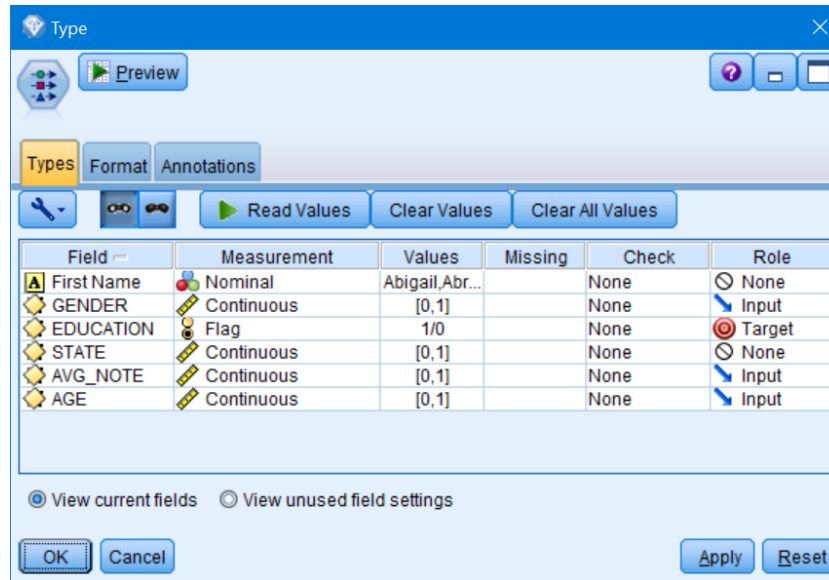


Figura 7.24: Definición de roles para cada variable dentro del tercer y último modelo estadístico a través del nodo tipo, fuente propia.

Variables in the Equation						
	B	S.E.	Wald	df	Sig.	Exp(B)
Step 1 <sup>a</sup> AGE	.431	.199	4.695	1	.030	1.539
Constant	-1.034	.152	46.580	1	.000	.355

a. Variable(s) entered on step 1: AGE.

Figura 7.25: Resultados obtenidos por el tercer modelo estadístico Regresión logística (Véase 3.1), fuente propia.

### 7.3. Desarrollo del aplicativo

Desde este punto, se empezó el desarrollo del aplicativo web, comenzando por el frontend, siguiendo con el backend y posteriormente definiendo el middleware para gestionar las diferentes peticiones. Esta etapa empezó de manera normal y según lo estipulado por cronograma, de forma relativamente independientes una de la otra, evolucionando periódicamente y siendo sus avances constantemente revisados con el fin de obtener una retroalimentación permanente. Ya después se inició con la integración de ambos extremos.

Al manejar un API-REST en la parte del backend, se procedió a desarrollar en primer lugar todas las interacciones con las bases de datos, teniendo en consideraciones cada una de las funcionalidades requeridas por cada uno de los módulos del proyecto. Además de ello, al hacer uso de información tan personal, se requería un especial cuidado en el manejo, cómo se almacenaba y la forma en la que movilizaba con el fin de contar con el mayor nivel de confidencialidad e integridad posible sin perder entonces compatibilidad dentro del aplicativo.

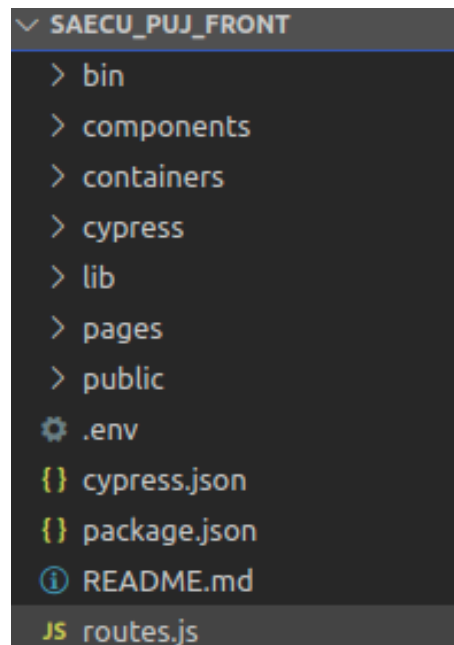


Figura 7.26: Estructura del proyecto en el apartado de front-end, fuente propia.

Adicionalmente, entre la amplia gama de retos que tuvieron lugar en la implementación del backend, se destacó la necesidad de modificar la estructura de los componentes visuales y los modelos manejados desde el backend al cambiar la información con la que se trabajaría (Véase 7.4). Como

se evidenció, esto generó un incremento en el tiempo destinado a la etapa de desarrollo, atrasando así siguientes etapas en el ciclo de la metodología.

En la figura 7.26 se evidencia la forma en cómo se manejó la estructura del frontend, es decir, la parte que evidencia el usuario. Para esta parte se utilizó una aproximación por **components** que es la especialidad de ReactJs como framework de TypeScript. Los diferentes componentes genéricos consultan y consumen las rutas que maneja el *middleware* del backend para luego mostrar de forma amable al usuario la información necesaria. Todas las carpetas evidenciadas a excepción de **components**, **containers** y **public** son creadas por defecto.

En el apartado de back como se muestra en la figura 7.27, nos encontramos con dos submódulos el submódulo de sufijo app y el submódulo de sufijo services, esta estructura se da gracias a la definición que tenemos de api-rest construida en Django y uso de micro servicios de parte de Nameko.

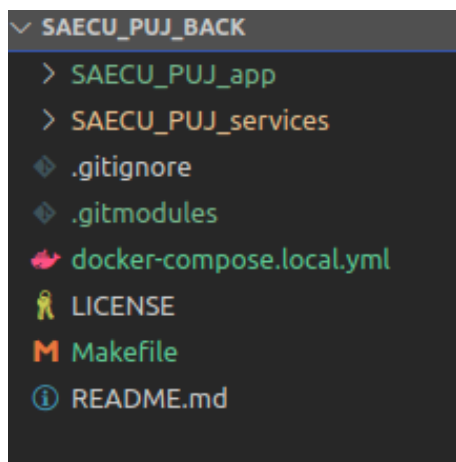


Figura 7.27: Estructura del proyecto por submodulos en el apartado de back-end, fuente propia.

En el submódulo de api como se muestra en la figura 7.28, cuenta con las siguientes carpetas y definiciones importantes:

- Apps: donde se define toda la estructura en modelos servicios de django para el funcionamiento del flujo de cada proceso y panel administrativo, además de los llamados a los microservicios pertinentes al siguiente submódulo de servicios.
- Deployment: lugar en que se definen distintos scripts que son utilizados para el despliegue en la nube del repositorio y cambios como tal.
- Project: se manejan definición de rutas, middlewares y conexiones a microservicios fundamentales además de definición del logging para datadog.

- Tests: lugar en que se definen los distintos tests para asegurar funcionalidad de cada endpoint con cada construcción posible.

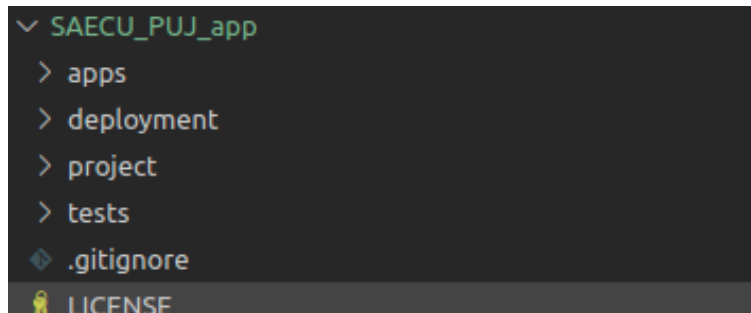


Figura 7.28: Estructura del proyecto en el apartado de back-end por parte del submodulo de api, fuente propia.

En el submódulo de services como se muestra en la figura 7.29, cuenta con las siguientes carpetas y definiciones importantes:

- configs: donde se define toda configuracion y secretos del aplicativo para hacer conexion con demas servicios como fron o datadog.
- Dependencies: lugar en que se definen distintos scripts que son utilizados para construccion y/o actualizacion de servicios necesarios para los microservicios como lo serian construccion de base de datos.
- Services: lugar donde se definen los distintos servicios por carpeta, modelos y metodos de cada una de estos.
- Tests: lugar en que se definen los distintos tests para asegurar funcionalidad de cada servicio con cada construcción posible, se realizan tests para cada servicio particularmente.

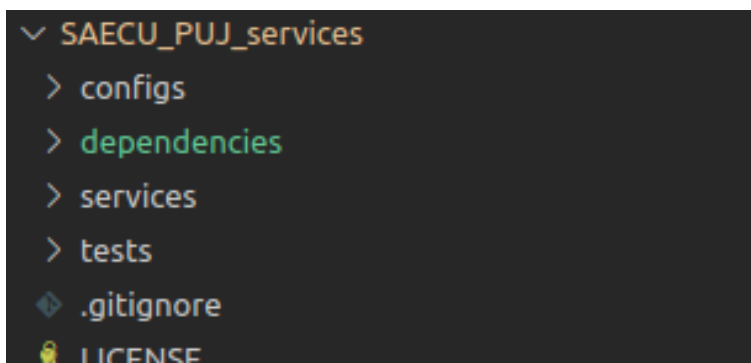


Figura 7.29: Estructura del proyecto en el apartado de back-end por parte del submodulo de servicios, fuente propia.

## 7.4. Dificultades en la Implementación

Finalizada la implementación del modelo estadístico y antes de continuar con el desarrollo del aplicativo, es pertinente mostrar cuáles fueron las complicaciones principales en la etapa de implementación.

Para la parte del modelo estadístico es claro que la dificultad presente fue la necesidad de modificarlo para que trabajara con datos simulados y así obtener predicciones aterrizadas. Esto se desembocó en más tiempo de desarrollo en su estructura para luego implementarlo en el MVP. Adicionalmente, fue necesario encontrar una herramienta que pudiera simular esta información, garantizando lealtad de datos para las predicciones, lo cual empleó tiempo de desarrollo adicional extendiendo la etapa.

Al no contar con información real sobre la que se basara el sistema, realizar las encuestas de satisfacción con el usuario perdía toda justificación pues las predicciones no serían completamente confiables. Por esta razón no se realizaron encuestas y no fueron implementadas para efectos de esta versión del MVP. Debido a esto no se logró evidenciar el nivel de satisfacción con el usuario final y queda definido para un trabajo futuro.

Finalmente, al contar con un modelo diferente al original, fue necesario modificar la estructura del código de forma tal que pudiera presentar de forma correcta la información al usuario final. Esto se desencadenó en más tiempo de desarrollo, lo que extendió el cronograma pensado inicialmente.



# Evaluación del Software

---

## 8.1. Plan de Evaluación

Para que la metodología SCRUM escogida para el desarrollo de este proyecto permitiera la obtención de soluciones aceptables a la hora de evaluar el software y, que a su vez estos resultados tuvieran sentido dentro del sistema con los recursos limitados que se disponían, fue necesario acotar la forma en cómo se evaluarían estos resultados. Estas limitaciones en esencia permiten observar la eficiencia del sistema desde una perspectiva bastante específica, la cual vendría siendo la del fabricante.

Esto anterior debido a la crisis y restricciones sanitarias consecuencia del COVID-19. Razón por la cual la interacción con la Pontificia Universidad Javeriana Cali fue bastante limitada a la hora de solicitar los datos requeridos para la obtención de resultados dentro del sistema predictivo, así como la organización de reuniones o encuentros siguiendo una modalidad formal para realizar dicha solicitud. De igual manera, la comunicación con el Centro de Servicios Informáticos (CSI) debió ser efectuada a través del director de tesis respetando un protocolo administrativo, cosa que frenó la rapidez en la obtención de dichos datos en posesión de dicho departamento dentro de la universidad. Todo esto conllevó a tomar la decisión en conjunto con el director de tesis de trabajar con datos simulados y controlados, reemplazando los datos originales que no se obtuvieron, pero garantizando un comportamiento aceptable y unos resultados que permitieron seguir el flujo de vida del mínimo producto viable (MVP).

El plan de evaluación quedó entonces definido como un plan centralizado o, mejor dicho, un plan de evaluación de fábrica. Este tipo de pruebas consisten en proveer un procedimiento que establezca los lineamientos generales para la realización y documentación de todas las pruebas de aceptación en fábrica, que abarcan cada una de las demostraciones, inspecciones físicas, mediciones y análisis de datos que permitirán validar y verificar el correcto funcionamiento de los equipos bajo prueba, de acuerdo con las especificaciones técnicas de ingeniería del fabricante [48]. En esencia, son pruebas controladas por los desarrolladores, eligiendo los escenarios que consideran más críticos o primordiales para garantizar un funcionamiento mínimo del sistema.

Nuevamente, este plan de evaluación fue realizado de la manera descrita por las limitaciones encontradas a lo largo del desarrollo del proyecto y porque de esta forma como desarrolladores pudimos definir un ambiente controlado de pruebas, limitado a funcionar bajo ciertas directrices y garantizando resultados esperados aún sin tener el producto completo sino más bien un mínimo

producto viable.

## 8.2. Pruebas

### 8.2.1. Pruebas funcionales

La sección presente contiene las pruebas realizadas para verificar el correcto funcionamiento e implementación del sistema desarrollado con todos los requerimientos para el MVP del aplicativo web. Para cada requerimiento se realizaron casos de prueba que tenían en cuenta diferentes parámetros, abarcando así las diferentes situaciones que se puedan presentar.

Como se mencionó anteriormente en esta misma sección, hubo complicaciones con los datos a usar dentro del modelo. Esto conllevó a la necesidad de modificar el modelo inicialmente desarrollado y a trabajar con otro tipo de datos; resultando en esencia en la necesidad de limitar las pruebas.

Estas pruebas son controladas para efectos de garantizar resultados aterrizados sin comprometer la integridad del sistema al tratar con datos que bien podrían dar resultados erróneos. El control que se realiza en el sistema es el de escoger previamente los datos con los cuales se trabajará contrario a recibir datos completamente inesperados. Adicionalmente, estos datos escogidos para ser trabajados por el modelo predictivo son entregados ya limpios y con un formato correcto al modelo, acelerando el proceso de prueba y evitando que el modelo de resultados inesperados (Véase 8.1). Dichas pruebas son documentadas en un cuadro de fuente propia. A continuación, se presentan los casos de prueba más importantes.

<b>Formato de caso de prueba</b>		
<b>Objetivo del caso de prueba</b>	Validar visualización de carreras	
<b>Identificador</b>	TC_carreras_0001	
<b>Nombre del caso</b>	Visualización de carreras	
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>• Encontrarse en la pantalla principal</li> </ul>	
<b>Paso</b>	<b>Resultado esperado</b>	<b>Resultado real</b>
1) Ir a la opción “Conoce acerca de las carreras” y hacer click	<ul style="list-style-type: none"> <li>- Se debe mostrar la interfaz de carreras con la descripción de carreras de ingeniería.</li> <li>- La pestaña superior “Carreras” debe aparecer sombreada</li> </ul>	OK

Figura 8.1: TC\_carreras\_0001

Formato de caso de prueba		
<b>Objetivo del caso de prueba</b>	Validar visualizacion de “informacion de programa” de una carrera	
<b>Identificador</b>	TC_carreras_0002	
<b>Nombre del caso</b>	Visualizacion de informacion del programa	
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>Ejecutar TC_carreras_0001</li> </ul>	
<b>Paso</b>	<b>Resultado esperado</b>	<b>Resultado real</b>
1) Acercarse al lineamiento izquierdo de la pagina y presionar el nombre “Ingenieria de sistemas”	- Debe verse un cambio en la interfaz mostrando el acordeon con opcion seleccionada “Sobre el programa”, donde se dara una descripcion del programa de ingenieria de sistemas.	OK

Figura 8.2: TC\_carreras\_0002

Formato de caso de prueba		
<b>Objetivo del caso de prueba</b>	Validar visualizacion de cursos de una carrera	
<b>Identificador</b>	TC_carreras_0003	
<b>Nombre del caso</b>	Visualizacion de cursos del programa	
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>Ejecutar TC_carreras_0002</li> </ul>	
<b>Paso</b>	<b>Resultado esperado</b>	<b>Resultado real</b>
1) ir a la opcion del acordeon de combre “Cursos” bajo el lineamiento del programa	- Se debe visualizar cada componente que hace referencia a cursos demostrando su carga academica y pertenencia dentro del pensum del programa.	OK

Figura 8.3: TC\_carreras\_0003

Formato de caso de prueba		
<b>Objetivo del caso de prueba</b>	Validar visualizacion de comentarios sobre una carrera	
<b>Identificador</b>	TC_carreras_0004	
<b>Nombre del caso</b>	Visualizacion de comentarios del programa	
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>Ejecutar TC_carreras_0002</li> </ul>	
<b>Paso</b>	<b>Resultado esperado</b>	<b>Resultado real</b>
1) ir a la opcion del acordeon de combre “Cursos” bajo el lineamiento del programa	- Se debe visualizar cada componente que hace referencia a comentarios del programa datando la informacion pertinente a fecha y tipo de persona como egresado o estudiante	OK

Figura 8.4: TC\_carreras\_0004

Formato de caso de prueba		
<b>Objetivo del caso de prueba</b>	Validar que al añadir notas al aplicativo se responda con la prediccion pertinente sobre las carreras	
<b>Identificador</b>	TC_carreras_0005	
<b>Nombre del caso</b>	Obtener prediccion de confiabilidad de carrera	
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>Ejecutar TC_carreras_0002</li> <li>No haber agregador notas</li> </ul>	
<b>Paso</b>	<b>Resultado esperado</b>	<b>Resultado real</b>
1) Ir a la opcion “Sobre el programa” en el apartado derecho de la pantalla.		OK
2) Ir a la opcion titulada “Agrega tus notas” y hacer click	Un modal debe ser desplegado permitiendo llenar los datos correspondientes a las materias	OK
3) Ingresar datos correspondientes de 1 a 5 para cada campo de nota.		OK
4) Ir a la opcion que lleva por nombre “Enviar” y hacer click	El modal debe desaparecer y un un numero indicador debe aparecer en el lugar donde el boton “Agrega tus notas” se encontraba.	OK

Figura 8.5: TC\_carreras\_0005

Formato de caso de prueba		
<b>Objetivo del caso de prueba</b>	Validar visualizacion de la pantalla introductoria al mercado laboral.	
<b>Identificador</b>	TC_mercado_0001	
<b>Nombre del caso</b>	Visualizacion de seccion de mercado laboral	
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>• Encontrarse en la pantalla principal</li> </ul>	
<b>Paso</b>	<b>Resultado esperado</b>	<b>Resultado real</b>
1) ir a la opcion "Carreras en el mercado laboral Colombiano" y hacer click	<ul style="list-style-type: none"> <li>- Se debe mostrar la interfaz de mercado laboral con la descripcion de el mercado laboral y su importancia.</li> <li>- La pestaña superior "Tu carrera en el mercado" debe aparecer sombreada</li> </ul>	OK

Figura 8.6: TC\_mercado\_0001

Formato de caso de prueba		
<b>Objetivo del caso de prueba</b>	Validar visualizacion de mercado laboral para cargo seleccionado.	
<b>Identificador</b>	TC_mercado_0002	
<b>Nombre del caso</b>	Visualizacion de seccion de mercado laboral	
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>• Ejecutar TC_mercado_0001</li> </ul>	
<b>Paso</b>	<b>Resultado esperado</b>	<b>Resultado real</b>
1) Acercarse al lineamento izquierdo de la pagina y presionar el nombre "Ingenieria de sistemas"	Se debe desplegar un sublistado de cargos relacionados a esta carrera	OK
2) Acercarse a la primera opcion de esta sublista y hacer click en su nombre	<ul style="list-style-type: none"> <li>-Se debe desplegar un componente con la grafica pertinente a los datos y el nombre del cargo en la parte superior de este</li> <li>-Se debe mostrar seleccionado el cargo al cual se accedio</li> </ul>	OK

Figura 8.7: TC\_mercado\_0002

Formato de caso de prueba		
<b>Objetivo del caso de prueba</b>	Validar filtro de datos para mercado laboral de un cargo.	
<b>Identificador</b>	TC_mercado_0003	
<b>Nombre del caso</b>	Visualizacion de seccion de mercado laboral	
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>Ejecutar TC_mercado_0002</li> </ul>	
<b>Paso</b>	<b>Resultado esperado</b>	<b>Resultado real</b>
1) Acercarse al lineamento superior de la estadística mostrada cerca al seleccionador que es apuntado de manera “Departamento”	Se debe desplegar un sublistado de departamentos colombianos para filtrar los datos	OK
2) Acercarse a la primera opción diferente a “Todos” de esta sublista y hacer click en ella	-Se debe notar un cambio en los datos mostrados actualmente en estadísticas de acuerdo lo anterior encontrado sin el nuevo filtro	OK

Figura 8.8: TC\_mercado\_0003

Formato de caso de prueba		
<b>Objetivo del caso de prueba</b>	Validar visualizacion de la pantalla de FAQ.	
<b>Identificador</b>	TC_FAQ_0001	
<b>Nombre del caso</b>	Visualizacion de seccion de FAQ	
<b>Precondiciones</b>		
<b>Paso</b>	<b>Resultado esperado</b>	<b>Resultado real</b>
1) Posicionarse en la barra de navegacion del aplicativo.		OK
2) Acercarse a la opción que lleva por título FAQ y hacer click.	<ul style="list-style-type: none"> <li>- Se debe mostrar la interfaz de la sección de FAQ mostrando todos los componentes de descripción y preguntas frecuentes.</li> <li>- La pestaña superior “FAQ” debe aparecer sombreada</li> </ul>	

Figura 8.9: TC\_FAQ\_0004

Formato de caso de prueba		
<b>Objetivo del caso de prueba</b>	Validar creacion de pregunta en la seccion de FAQs.	
<b>Identificador</b>	TC_FAQ_0002	
<b>Nombre del caso</b>	Creacion de preguntas en seccion de FAQ	
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>Ejecutar TC_FAQ_0001</li> </ul>	
<b>Paso</b>	<b>Resultado esperado</b>	<b>Resultado real</b>
1) Ir al final de la pagina acercandose al cuadro de texto para realizar el comentario.		OK
2) Registrar lo conveniente para el comentario.		OK
3) Ir a la opcion nombrada "Enviar" en el mismo formulario y hacer click.	Se debe observar una alerta que lleva por contenido "La pregunta ha sido enviada"	OK

Figura 8.10: TC\_FAQ\_0005

### 8.2.2. Pruebas de aceptación

Las pruebas de aceptación del usuario son importantes porque permiten verificar si el producto resultante cumple con los estándares, necesidades y requerimientos del usuario objetivo quien va a hacer uso de este producto. Para efectos de implementación de estas pruebas se tenía pensado como público objetivo los estudiantes próximos a entrar a la universidad, específicamente los provenientes de un colegio ubicado en el municipio de Yumbo; esto por la cercanía, facilidad y confianza depositada de parte de la institución para poder realizar dichos procedimientos con sus estudiantes; al ser ellos la razón principal por la cual surge la idea de este proyecto de grado, consideramos natural realizar las pruebas con ellos.

Al usar datos simulados en el sistema, fue necesario modificar la estructura del modelo de tal forma que pudiera trabajar con datos replica que probablemente estuviesen incompletos al ser asunciones de lo que se pudo haber recibido de parte del CSI y cuyo único propósito era el de para reemplazar la inexistencia de los originales. Por ende, no se consideró verificar la satisfacción del usuario sobre las predicciones, pues estas podrían estar sesgadas y presentar resultados erróneos sin tener garantía de que en un escenario real el sistema funcionaría aún con datos simulados. Por estas razones no se pudieron programar dichas pruebas y por lo tanto no hubo una retroalimentación completa sobre el MVP.

### 8.3. Resultados

Se propuso un mínimo producto viable en forma de aplicativo WEB que ayude a mejorar la situación sobre la mala elección vocacional universitaria en la Pontificia Universidad Javeriana Cali generada por la desinformación sobre la naturaleza de las carreras que se ofrecen en la institución, las alternativas financieras que hay o en general por una falta de motivación profesional y de autoconocimiento.

A través de la innovación y el desarrollo de las TIC, se implementó el 60% de los requisitos funcionales del software. Así mismo, se realiza una entrega de un producto mínimo viable a modo de aplicativo web que apoya en la elección vocacional de estudiantes de último año próximos a entrar a la universidad. Adicionalmente, se efectúan y aprueban más del 80% de los casos de prueba y se entrega la documentación del software que incluye: análisis, diseño, implementación y una sección explicativa del uso de la aplicación web y funciones, denominada FAQ (Frequently Asked Questions).

#### 8.3.1. FAQ

La sección de Preguntas Frecuentes (FAQ por sus siglas en inglés) es aquella que esta destinada a responder inquietudes que pueden surgir antes, durante y después del uso del sistema. A continuación, se presentan las posibles interrogantes más importantes que pueden surgir.

- **¿A quién va dirigido el aplicativo?**

El aplicativo va dirigido a todo aquel estudiante que haya culminado sus estudios de educación media (Bachillerato en el sistema educativo colombiano) y que desea ingresar a la Pontificia Universidad Javeriana Cali escogiendo una carrera comprendida dentro de la facultad de Ingeniería y Ciencias.

- **¿Cuál es el objetivo del aplicativo?**

El aplicativo web pretende apoyar a los estudiantes en la selección de su carrera universitaria a través de predicciones (porcentaje de afinidad y porcentaje de seguridad sobre la afinidad proporcionada) que muestran de forma muy general y resumida cómo se desempeñaría cada uno de los estudiantes, basándose en su rendimiento previo en la educación media y tomando en cuenta variables adicionales, así como cientos de ejemplos previos guardados.

- **¿Cómo acceder al aplicativo?**

Para hacer uso del sistema producto de este proyecto de grado es necesario tener acceso a un ordenador, celular inteligente o laptop que a su vez cuenten con internet o cobertura móvil en caso de los celulares. Como su naturaleza lo indica, al ser un aplicativo web su acceso es mediante la web exclusivamente y por ello la necesidad de algún dispositivo que cuente con esta tecnología.

- **¿En qué carreras universitarias se puede visualizar el apoyo mencionado?**

El aplicativo web comprende todas las carreras que se encuentran dentro de la facultad de

Ingeniería y Ciencias en la Pontificia Universidad Javeriana Cali. Por ejemplo, ingeniería de sistemas y computación, ingeniería civil, ingeniería industrial, ingeniería electrónica, entre otras.

- **¿Es seguro usar el aplicativo web?**

Al usar el aplicativo que trabaja con datos personales de los diferentes usuarios, es comprensible que surja la inquietud sobre si es seguro. Para garantizar dicho requerimiento el aplicativo web no necesita ni pedirá datos personales que puedan poner en riesgo la integridad de cada usuario. Lo único que se necesita son datos generales sobre la educación media, así como el nombre de la institución entre otras.

- **¿Cómo utilizar el aplicativo?**

El aplicativo web esta diseñado para trabajar con cierta información del usuario. Esta información se reparte en variables con las que trabaja el modelo por debajo para poder presentar ante el estudiante las predicciones en la página del aplicativo. Lo único que debe realizar el usuario es brindar cada uno de los datos que solicita el aplicativo, introducirlos mediante la opción brindada y elegir sobre que carrera desea visualizar su pronóstico. Para una descripción más detallada de la información necesaria para que el sistema funcione correctamente ver 7.2.

- **¿De donde provienen los datos previos?**

Los datos previos ya presentes en el sistema para poder realizar las predicciones fueron sacados de un generador aleatorio de datos en internet. Esto debido a la falta de respuesta de parte del CSI (Véase 8.1) sobre los datos solicitados necesarios para el desarrollo de este proyecto. Sin embargo, dichos datos aleatorios hacen de sustitos al tener información general previa y actual de cada estudiante (Colegio, edad, notas en el colegio, etc.) que pasó por la facultad de Ingeniería y Ciencias en la Pontificia Universidad Javeriana Cali. La idea a futuro es utilizar los datos que brinde el CSI para poder realizar las predicciones con total seguridad de que sus resultados son verídicos.



# Conclusiones

---

Partiendo del hecho de que el objetivo de este proyecto era desarrollar una aplicación que apoye el proceso de elección de carrera universitaria en la Pontificia Universidad Javeriana Cali, en conclusión, se puede decir que se realizó el proceso indicado para diseñar y desarrollar un aplicativo web, que apoye en el proceso de elección de una carrera universitaria dentro de la facultad de ingenierías en la Pontificia Universidad Javeriana Cali. Lo anterior logrado mediante la implementación del MVP para el aplicativo web pensado. Adicionalmente, como conclusiones generales en el desarrollo de este proyecto de grado y siguiendo los objetivos planteados, se concretó lo siguiente:

- Tener una metodología de desarrollo es crucial en la ejecución de un proyecto, porque sirve para primero organizar las diferentes etapas que se deben realizar y que son cruciales para crear un producto como de este proyecto; segundo para tener planes de contingencia en caso de que alguna de las etapas no se ejecute de acuerdo con lo planeado y así poder tomar otros caminos que generen resultados similares o por lo menos aceptables comparados con los originalmente planeados. El tipo de metodología designado para el proyecto (Ágil o tradicional), depende de cómo esta visualizada dicha planificación. Una metodología ágil se presta para ideas que no están bien definidas aún y que presentan un tiempo de vida corto. Por otro lado, las metodologías tradicionales requieren de proyectos bien definidos, donde los cambios no son muchos y el entorno no es tan volátil. Precisamente por estas razones es que se decidió escoger una metodología ágil, porque su continuo seguimiento y sus herramientas para contrarrestar cualquier inconveniente en un tiempo reducido permiten seguir el desarrollo sin mayor problema; su fácil adaptación era el componente ideal para trabajar con un proyecto que podría cambiar a mitad de camino debido a la incertidumbre de los datos recibidos, qué tan completos estarían y si los resultados eran mínimamente precisos.
- Las tecnologías usadas para el desarrollo de este proyecto, como lo son el lenguaje de programación Python designado para el backend, el framework Angular para el frontend y la base de datos relacional MySQL, son de gran ayuda a la hora de desarrollar aplicaciones pensadas para ser accedidas desde cualquier dispositivo de forma web. No solo ahorran tiempo en el desarrollo por su fácil implementación, también cuentan con la capacidad de ser ampliamente soportados y corregidos al recibir apoyo de una gran comunidad desarrolladora distribuida en diferentes foros que manifiestan sus aportes a través de las muchas librerías existentes que extienden sus funcionalidades. Combinadas, estas tecnologías y usadas en el desarrollo del MVP

facilitaron en gran medida la implementación en un ambiente web volviendo la experiencia para el desarrollador para nada tediosa.

- El desarrollo inicial pensado para el proyecto de grado descrito en este documento, no se pudo realizar en su totalidad debido a las diferentes situaciones ocasionadas por la pandemia COVID-19 y por los diferentes sucesos consecuencia de las manifestaciones ocurridas durante el paro nacional que tuvo lugar en el primer semestre del 2021. Esto ocasionó que la comunicación con el Centro de Servicios Informáticos (CSI) de la Pontificia Universidad Javeriana Cali se dificultara, puesto que aunque se estableció un mecanismo comunicativo que respetara los lineamientos administrativos a través del director del presente proyecto, nunca se obtuvo una respuesta imposibilitando la obtención de los datos en posesión de dicho organismo y necesarios para solventar la completitud del aplicativo web a través de por ejemplo, pruebas de aceptación de usuario e información temporal previa con la cual se hubiese obtenido mejor retroalimentación del MVP resultante, obligando a recurrir.
- Finalmente, realizar un diseño apropiado permite contar con componentes del proyecto lo suficientemente dinámicos como para ser adaptados a la situación que se presente independientemente de si era o no la esperada. Para este proyecto fue de gran ayuda tener una metodología ágil debido a que, al no contar con los datos necesarios para completar el aplicativo web, fue necesario recurrir a datos simulados que reemplazarán los originalmente pensados y modificar componentes como el modelo de datos de tal forma que pudieran retornar resultados aceptables aún con datos simulados y controlados. Debido a esta metodología y la naturaleza del sistema fue posible contrarrestar la situación imprevista de manera rápida, permitiendo un flujo normal hasta lograr el MVP.

# Trabajo Futuro

---

Como trabajo a futuro es necesario establecer un plan de acción que permita implementar las funcionalidades que no pudieron ser plasmadas en el sistema por las complicaciones mencionadas en 8.1. Este plan de acción esta conformado por las siguientes etapas:

1. **Obtención de los datos:** obtener los datos necesarios para el funcionamiento adecuado del modelo predictivo encargado de entregar las predicciones de afinidad de cada estudiante con la carrera escogida para cursar. Estos vienen siendo los datos que están en posesión del centro de servicios informáticos de la Pontificia Universidad Javeriana Cali.
2. **Modificación del modelo:** una vez obtenidos los datos reales y de forma aleatoria, es necesario modificar el modelo a su estado original, es decir, la estructura que tenía designada para funcionar con datos reales. Esta estructura contiene internamente las pruebas que garantizan la veracidad de las predicciones y el funcionamiento del modelo.
3. **Uso de los datos en el modelo:** posteriormente, teniendo ya tanto el modelo como los datos reales listos, se puede empezar a usar el modelo en conjunto con los datos, esto es, realizar las predicciones para cada estudiante y la carrera que desea cursar.
4. **Pruebas de aceptación:** finalmente, una vez completado todo el flujo anteriormente descrito de forma correcta y cómo se tenía pensado desde el inicio de este proyecto de grado, se pueden validar las pruebas de aceptación con el usuario objetivo para verificar si el producto resultante cumple con los estándares, necesidades y requerimientos del usuario objetivo quien va a hacer uso de este producto.



# Bibliografía

- [1] “¿qué porcentaje de nuestros bachilleres ingresa de manera inmediata a la educación superior?.” [https://www.mineduccion.gov.co/1621/articles-350451\\_recurso\\_11.pdf](https://www.mineduccion.gov.co/1621/articles-350451_recurso_11.pdf), 2015. [Online accessed 13-march-2020].
- [2] “Deserción estudiantil en la educación superior: Metodología de seguimiento, diagnóstico y elementos para su prevención.” [https://www.mineduccion.gov.co/sistemasdeinformacion/1735/articles-254702\\_libro\\_desercion.pdf](https://www.mineduccion.gov.co/sistemasdeinformacion/1735/articles-254702_libro_desercion.pdf), 2009. [Online accessed 13-march-2020].
- [3] J. O. C. Restrepo, “¿qué es la orientación vocacional?.” <https://www.ucn.edu.co/virtualmente/bienestarC-blog/Lists/EntradasDeBlog/Post.aspx?ID=5>, 2015. [Online accessed 30-may-2020].
- [4] L. J. B. Sánchez, “Orientación vocacional y profesional en la juventud colombiana..” <https://repository.unimilitar.edu.co/bitstream/handle/10654/14245/BetancourthSElnchezLinaJhulieth.pdf;jsessionid=B6BB38F405CB578360FC36EB04E81C84?sequence=1>, 2016. [Online accessed 11-may-2021].
- [5] S. D. Escobar, “Elección profesional y deserción universitaria. del redireccionamiento del plan de vida,” *Revista Electrónica Psyconex*, vol. 5, oct. 2014. [Online accessed 11-may-2020].
- [6] L. A. Pineda, *Factores que afectan la elección de carrera: caso Bogotá*, vol. 15. Pontificia Universidad javeriana, 2015.
- [7] P. M. Barrios, “Impacto de la universidad en la sociedad: un análisis desde la financiación de la educación superior en colombia..” <https://revistas.unilibre.edu.co/index.php/saber/article/view/713/561>, 2016. [Online accessed 11-may-2021].
- [8] “Tendencia de las ocupaciones.” <https://observatorio.sena.edu.co/Tendencia/Informes>. [Online accessed 24-jan-2021].
- [9] “Ley estatutaria 1581 de 2012.” <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=49981>. [Online accessed 27-jan-2021].
- [10] P. Bourque and R. Fairley, *Guide to the Software Engineering Body of Knowledge*. IEEE Computer Society, version 3.0 ed., 2014.
- [11] S. F. Fernández, J. M. C. Sánchez, and A. Córdoba, *Estadística Descriptiva*. Escuela Superior de Gestion Comercial y Marketing, 2 ed., 2002.
- [12] “Estadística descriptiva multivariante.” <http://halweb.uc3m.es/esp/Personal/personas/jmmarin/esp/AMult/tema2am.pdf>. [Online accessed 21-may-2020].

- [13] F. Maureira Cid, *Estadística Avanzada para Educación Física*. Editorial Académica Española, 03 2016.
- [14] H. J. Larson, *Introducción a la Teoría de Probabilidad e Inferencia Estadística*. Limusa, 1 ed., 1978.
- [15] A. G. Peña, M. V. Buitrago, and M. P. Curietes, *Estadística Basica*. Instituto Tecnológico Metropolitano, 2 ed., 2010.
- [16] J. A. Rodrigo, “Correlación lineal y regresión lineal simple.” [https://www.cienciadedatos.net/documentos/24\\_correlacion\\_y\\_regresion\\_lineal](https://www.cienciadedatos.net/documentos/24_correlacion_y_regresion_lineal), 2016. [Online accessed 03-may-2021].
- [17] “Portal buscando carrera.” <http://aprende.colombiaaprende.edu.co/es/buscandocarrera>. [Online accessed 30-may-2020].
- [18] “Descripción del portal de buscando carrera.” <https://www.mineducacion.gov.co/sistemasinfo/Informacion-a-la-mano/214354>. [Online accessed 30-may-2020].
- [19] V. Galilea, “Orientación vocacional.” [https://www.sie.es/crl/archivo\\_pdf/ORIENTACION%20VOCACIONAL.pdf](https://www.sie.es/crl/archivo_pdf/ORIENTACION%20VOCACIONAL.pdf), Dec 2001. [Online accessed 30-may-2020].
- [20] J. L. F. Seara and F. A. García, “Cuestionario de intereses profesionales.” [http://www.web.teaediciones.com/Ejemplos/CIPSA\\_extracto\\_manual.pdf](http://www.web.teaediciones.com/Ejemplos/CIPSA_extracto_manual.pdf), 2013. [Online accessed 30-may-2020].
- [21] M. De La Cruz, “Inventarios de intereses personales profesionales-revisado.” <https://www.studocu.com/co/document/universidad-de-antioquia/psicologia/apuntes/ipp-r-apuntes-test-vocacional-aplicacion-calificacion/2652633/view>, 2015. [Online accessed 30-may-2020].
- [22] A. M. D. Mori and C. Santiviago, “Conceptos y herramientas para aportar a la orientación vocacional ocupacional de los jóvenes.” <http://www2.compromisoeducativo.edu.uy/sitio/wp-content/uploads/2013/10/MANUAL-CONCEPTOS-Y-HERRAMIENTAS-OVO.pdf>. [Online accessed 30-may-2020].
- [23] K. Schwaber and J. Sutherland, “The scrum guide.” <https://www.scrumguides.org/scrum-guide.html>. [Online accessed 30-may-2020].
- [24] J. Aitchison and D. Sculthorpe, *Some Problems of Statistical Prediction*. Oxford University Press, 2 ed., 1965.
- [25] J. M. Carew, “predictive modeling.” <https://searchenterpriseai.techtarget.com/definition/predictive-modeling>, 2020. [Online accessed 12-october-2021].

- [26] R. Ali, “Predictive modeling: Types, benefits, and algorithms.” <https://www.netsuite.com/portal/resource/articles/financial-management/predictive-modeling.shtml>, 2020. [Online accessed 12-october-2021].
- [27] A. Rana, “Predictive modeling - what it is and why you need it for your business.” <https://www.seasiainfotech.com/blog/what-is-predictive-modeling-why-you-need-it-for-your-business/>, 2020. [Online accessed 12-october-2021].
- [28] “Predictive modeling.” <https://www.mathworks.com/discovery/predictive-modeling.html>. [Online accessed 18-october-2021].
- [29] “Overview of nodes.” [https://www.ibm.com/docs/en/spss-modeler/SaaS?topic=SS3RA7\\_sub/modeler\\_mainhelp\\_client\\_ddita/clementine/clef\\_nodes\\_intro.html](https://www.ibm.com/docs/en/spss-modeler/SaaS?topic=SS3RA7_sub/modeler_mainhelp_client_ddita/clementine/clef_nodes_intro.html). [Online accessed 19-october-2021].
- [30] “Linear regression analysis using spss statistics.” <https://statistics.laerd.com/spss-tutorials/linear-regression-using-spss-statistics.php>. [Online accessed 21-october-2021].
- [31] J. Brownlee, “What is the difference between test and validation datasets?.” <https://machinelearningmastery.com/difference-test-validation-datasets/>, 2020. [Online accessed 27-october-2021].
- [32] R. Gnatyk, “Microservices vs monolith: which architecture is the best choice for your business?.” <https://www.n-ix.com/microservices-vs-monolith-which-architecture-best-choice-your-business/>, 2018. [Online accessed 26-september-2021].
- [33] “What is the difference between monolithic and microservices?.” <https://www.appdynamics.com/topics/microservices-vs-monolithic>. [Online accessed 27-october-2021].
- [34] D. Nefedov, “Monolithic vs. microservices architecture: How the software structure can influence the efficiency.” <https://magora-systems.com/monolithic-architecture-vs-microservices/>, 2018. [Online accessed 16-november-2021].
- [35] “React: A javascript library for building user interfaces.” <https://reactjs.org/>. [Online accessed 16-august-2021].
- [36] C. Sev, “React popularity when not to use react.” <https://www.better.dev/react-popularity>, 2021. [Online accessed 26-august-2021].
- [37] Phillip, “Why to choose reactjs.” <https://appwrk.com/why-businesses-are-preferring-reactjs-development-services/>, 2019. [Online accessed 26-august-2021].

- [38] “What is python? executive summary.” <https://www.python.org/doc/essays/blurb/>. [Online accessed 26-august-2021].
- [39] “Python vs. other programming languages.” <https://www.stxnext.com/python-vs-other-programming-languages/>. [Online accessed 26-august-2021].
- [40] “What is python and why is it so popular?.” <https://www.monterail.com/blog/what-is-python-and-why-is-it-so-popular>. [Online accessed 26-august-2021].
- [41] datadoghq, “Datadog website.” <https://www.datadoghq.com/>, 2010. [Online accessed 27-october-2021].
- [42] “1.2.1 what is mysql?.” <https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html>. [Online accessed 26-august-2021].
- [43] O. Valin, “Most popular databases in 2020 and new trends.” <https://www.eversql.com/most-popular-databases-in-2020/>, 2020. [Online accessed 26-august-2021].
- [44] IBM, “Ibm spss modeler advantage.” <https://www.ibm.com/docs/en/sdm/18.0.0?topic=overview-spss-modeler-advantage>, 2021. [Online accessed 17-december-2021].
- [45] TONIC, “Why mockaroo?.” <https://www.mockaroo.com/>, 2021. [Online accessed 22-october-2021].
- [46] marvelapp, “Marvelapp, rapid prototyping.” <https://marvelapp.com/>, 2021. [Online accessed 27-october-2021].
- [47] IBM, “Export node (export client node information.” <https://www.ibm.com/docs/en/spectrum-protect/8.1.9?topic=commands-export-node-export-client-node-information>, 2021. [Online accessed 27-october-2021].
- [48] J. Ferrer, “Procedimiento: Pruebas de aceptación en fabrica (fat).” <https://www.linkedin.com/pulse/procedimiento-de-aceptación-pruebas-en-fabrica-sat-josé-ferrer-1e/?originalSubdomain=es>, 2019. [Online accessed 04-november-2021].