

**Práctica Empresarial enfocada al Desarrollo Módulos de Data Visualization**

Nicolás Aguirre Espinosa

Director De La Práctica:

Alexander Molina Cabrera

Universidad Tecnológica De Pereira

Programa Ingeniería De Sistemas Y Computación

Pereira - Colombia

2021

## **Tabla de Contenido**

<b>1. Introducción</b>	<b>4</b>
1.1 Resumen	5
1.2 Abstract	6
1.3 Justificación	7
1.4 Objetivos	9
1.4.1 Objetivo General	9
1.4.2 Objetivos Específicos	9
<b>2. Marco Metodológico</b>	<b>10</b>
2.1 Herramientas de desarrollo	12
<b>3. Desarrollo del Software</b>	<b>15</b>
3.1 Requisitos	15
3.1.1 Requisitos no funcionales:	20
3.2 Desarrollo Conceptual	22
3.3 Ejecución y desarrollo del software	24
3.3.1 Análisis de requisitos	24
3.3.2 Diseño y arquitectura	25
3.3.3 Programación	25
3.3.4 Documentación	26
3.4 Interfaz de Usuario	27
3.4.1 Homepage	27
3.4.2 Población Estudiantil	31
3.4.3 Deserción	40
<b>4. Trabajos Futuros</b>	<b>44</b>
<b>5. Conclusiones</b>	<b>45</b>

**6. Recomendaciones**

**7. Bibliografía**

## 1. Introducción

Las prácticas empresariales surgen con el objetivo de fomentar una formación integral de los estudiantes, a través de un proceso que le permita aplicar los conocimientos y habilidades teóricas y técnicas adquiridas durante el desarrollo del programa académico de pregrado.

En la actualidad, cada día una masiva cantidad de información debe ser analizada de manera efectiva, por lo que esta práctica empresarial nace como respuesta a la necesidad de la Facultad de Ingenierías de la Universidad Tecnológica de Pereira de tener una herramienta que recolecta, analice y muestre en un contexto visual la información proporcionada por el sistema de inteligencia institucional, enfocados en los campos de deserción intersemestral, deserción interanual, análisis de cohorte y tendencia poblacional.

En este caso, el estudiante participó activamente en el desarrollo de módulos de Data Visualization basados en Ciencias de Datos para efectos de analítica enfocado principalmente en la tendencia poblacional de la facultad de ingenierías en variables como la deserción y el rendimiento académico.

## 1.1 Resumen

El presente documento tiene como objetivo documentar el proceso realizado durante la realización de las prácticas empresariales llevadas a cabo en la Facultad de Ingenierías de la Universidad Tecnológica de Pereira, la cual tuvo una duración de 5 meses, en los cuales se desarrollaron módulos de Data Visualization, enfocados al análisis de la deserción y tendencias poblacionales de la facultad a partir del año 2010 al presente.

Los módulos mencionados anteriormente, sirvieron como base para el desarrollo de un dashboard o tablero de mando, que se dividió en dos partes principales, el frontend y el backend; React y Django fueron los framework seleccionados, cada uno por sus características específicas que cumplían con los requisitos del proyecto.

**Palabras Clave:** Desarrollo de software, Backend, Frontend, Data Visualization, Tablero de Mando.

## 1.2 Abstract

The purpose of this document is to document the process carried out during the university practices in the Faculty of Engineering of the Technological University of Pereira, which lasted 5 months. Data Visualization modules were developed focused on the analysis of dropouts and population trends of the faculty from 2010 to the present.

The modules mentioned above served as the basis for the development of a dashboard, which was divided into two main parts, the frontend and the backend. React and Django were the selected frameworks, each one for its specific characteristics that met the project requirements.

**Keywords:** Software development, Frontend, Backend, Data Visualization, Dashboard.

### 1.3 Justificación

La deserción académica en el nivel universitario es un asunto que se observa a nivel internacional y nacional, que involucra a todas las carreras. “ Las causas son múltiples, pero entre ellas se cuentan la “escasez de recursos económicos, la desorientación vocacional, factores socioculturales, mala calidad en la educación y dificultades para rendir académicamente, cada una de las cuales se presenta bajo parámetros diferentes, según las circunstancias que se dan en el entorno social y cultural de cada individuo”[1].

La facultad de ingenierías en su búsqueda constante de nuevas estrategias para la disminución de la deserción estudiantil, se dio cuenta de la necesidad de una herramienta que permitiera el fácil y correcto análisis de la información proporcionadas por el sistema de inteligencia estudiantil, así mismo, que permitiera el fácil acceso y visualización de diferentes métricas como la cantidad de estudiantes por sexo, nivel de formación, tipo de colegio de procedencia, etc. Con lo anterior, se propuso el desarrollo de un dashboard o tablero de mando como respuesta a dicha problemática.

Por lo cual a través de las prácticas empresariales en la Convocatoria de Estado Joven con el objetivo de afianzar el proceso formativo e integral de los estudiantes, brindando la oportunidad de aplicar los conocimientos teóricos y metodológicos en los diferentes campos de intervención, así como diagnosticar, valorar y evaluar dentro de una situación real, las actividades planteadas durante el proceso de prácticas al practicante fueron:

- Entendimiento y análisis del funcionamiento y las fuentes de información del sistema de inteligencia institucional de la UTP.
- Definición y construcción de arquitectura de integración de fuentes de información.

- Conformación de base de datos para la facultad de ingenierías.
- Propuesta de tablero de mando basado en data visualization.



## **1.4 Objetivos**

### **1.4.1 Objetivo General**

Desarrollar módulos de Data Visualization basados en Ciencias de Datos para efectos para efectos de Analítica de la deserción y tendencia poblacional de los estudiantes de la facultad de ingenierías usando datos del sistema de inteligencia institucional.

### **1.4.2 Objetivos Específicos**

- Incrementar las capacidades del estudiante en práctica en el entendimiento, manejo y conceptualización de las estructuras de datos de diferentes sistemas de información.
- Desarrollar en el estudiante las habilidades asociadas a la construcción y planteamiento de arquitectura enfocada a la minería de datos.
- Fortalecer las competencias del estudiante el planteamiento, creación, diseño, implementación de una base de datos.
- Consolidar en el estudiante las competencias asociadas al desarrollo de software enfocado a la data visualization.

## 2. Marco Metodológico

Durante la práctica empresarial se aplicó el modelo incremental o creciente para llevar a cabo las actividades propuestas inicialmente en el plan de práctica, teniendo una reunión semanal o cada dos semanas con el docente guía de la práctica, para discutir los avances, correcciones y asignaciones de labores, agregando así nuevas funcionalidades con el fin de ver rápidamente los resultados de los requisitos propuestos durante cada reunión.

Se aplicó el modelo incremental, porque es un conjunto de tareas agrupadas en iteraciones, por lo que permite agregar nuevas especificaciones y funcionalidades durante el proceso de desarrollo del aplicativo final. Este modelo incremental entrega el software en partes pequeñas, pero utilizables llamadas (incrementos). En general, cada incremento se construye sobre la base de aquel que ya ha sido entregado.

El primer incremento que se desarrolla contiene los elementos básicos del proyecto hasta seguir con los siguientes incrementos, con el fin de mejorar la funcionalidad, se afrontan los requisitos básicos y algunas funciones suplementarias.

Como resultado de la constante evaluación, se desarrolla un plan para cada incremento, teniendo como núcleo que se va a realizar modificaciones sobre el primer incremento planteado. Este proceso se repite siguiendo la entrega de cada incremento, hasta que se elabore el producto completo en este caso de la plataforma web, la estructura que se siguió se puede ver representada en el siguiente esquema:

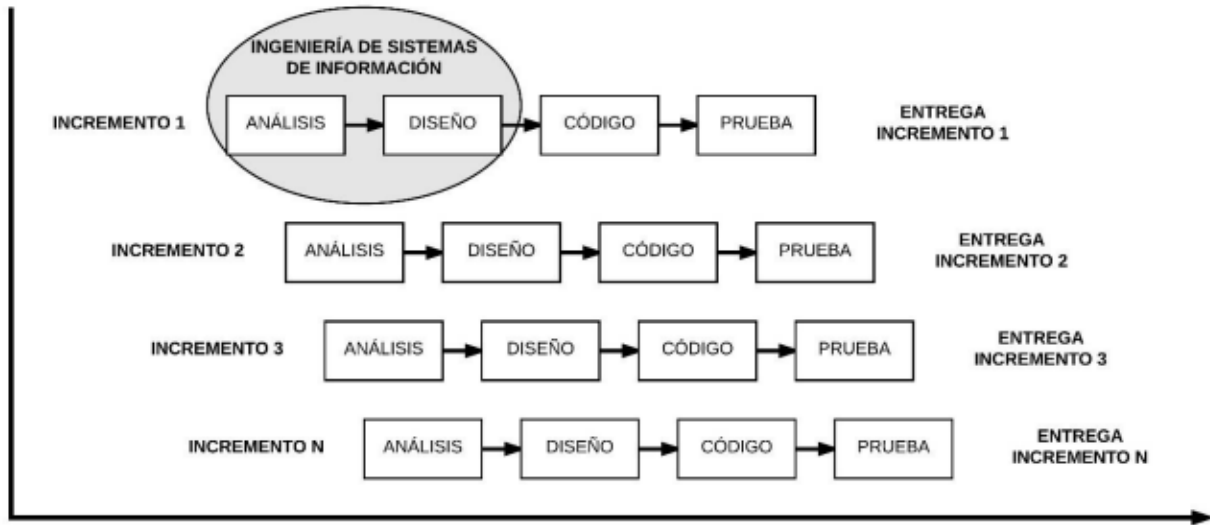


Figura No.1 Esquema del Modelo Incremental (Pressman, 2001)

## 2.1 Herramientas de desarrollo

En el presente capítulo se definen las herramientas y tecnologías utilizadas durante el desarrollo e implementación de los módulos de data visualization de la práctica empresarial en cuestión:

- **JavaScript:** Es uno de los lenguajes de programación más populares en la actualidad, inicialmente fue creado para interactuar con los diversos elementos que puede poseer un página web, por ejemplo con JavaScript se puede agregar un evento a un clic para que muestre una alerta entre muchas otras cosas.

Se caracteriza por que se escribe y se compila dinámicamente JIT(Just in Time), es decir, el código se compila cuando se ejecuta.

- **Python:** Python es un lenguaje de programación de código abierto y multiplataforma. Con el pasar del tiempo Python ha ganado popularidad debido a sus facilidad tanto de uso como de aprendizaje, así mismo, por la amplia variedad de posibilidades que ofrece al permitir trabajar en múltiples ramas del conocimiento como la inteligencia artificial, machine learning, deep learning entre muchos otras.
- **React:** También llamada React.js o ReactJS, es una biblioteca del lenguaje de programación JavaScript de código abierto diseñada con el objetivo de crear interfaces de usuario y así facilitar el desarrollo de aplicaciones web, SPA (Single Page Application) y hasta aplicaciones para dispositivos móviles.

En los últimos años su auge ha ido aumentando por parte del mundo del desarrollo de software y especialmente del desarrollo web, gracias a que ofrece grandes beneficios en rendimiento, modularidad, y facilidad de planeación y desarrollo de apps complejas.

- **Django:** Es un framework de código abierto y escrito en Python que está basado en el patrón de diseño MTV (Model Template View) para el desarrollo de páginas webs, aplicaciones y APIs (con Django REST Framework).
- **Django REST Framework:** Django Rest Framework es una aplicación Django que permite la reutilización de una gran cantidad de código, fue utilizado en este proyecto dado que permite construir proyectos software bajo la arquitectura REST, Por otro lado, Django REST Framework trae incluido una interfaz administrativa que permite realizar pruebas y peticiones sobre el protocolo HTTP.
- **PostgreSQL:** También llamado Postgres, es un sistema de gestión de bases de datos relacional orientado a objetos y es de código abierto.  
  
Las bases de datos PostgreSQL se han convertido en unas de las más utilizadas por los usuarios gracias a que ofrecen una serie de ventajas como son: la disponibilidad multiplataforma, la fácil configuración, sistemas de fiabilidad, control de concurrencias entre muchas otras.
- **GitHub:** Es una plataforma que ofrece a los desarrolladores la posibilidad de crear repositorios de código y guardarlos en la nube de forma segura, usando un sistema de control de versiones, llamado Git.  
  
Su principal objetivo en este proyecto es llevar registro de los cambios realizados durante el desarrollo del aplicativo Web.

### 3. Desarrollo del Software

#### 3.1 Requisitos

##### 3.1.1 Requisitos funcionales:

<b>Nombre:</b> Barra de navegación					
<b>Código:</b>	RF-1	<b>Versión:</b>	1.0	<b>Fecha:</b>	26/08/2021
<b>Fuente</b>	Nicolás Aguirre Espinosa	<b>Prioridad</b>	Alta		
<b>Descripción</b>					
El tablero de mando deberá permitir a los usuarios visualizar el menú de opciones de la página web , así como permitir el acceso a dicho menú y navegación en el mismo. Por otro lado, el menú tendrá dos opciones principales: población estudiantil y deserción.					
<b>Observaciones</b>					
Se debe incluir subdivisiones de población estudiantil y deserción					

Tabla No.1 Requisito funcional()

<b>Nombre:</b> Vista de información general anual					
<b>Código:</b>	RF-2	<b>Versión:</b>	1.0	<b>Fecha:</b>	26/08/2021
<b>Fuente</b>	Nicolás Aguirre Espinosa	<b>Prioridad</b>	Alta		
<b>Descripción</b>					
El sistema debe permitir al usuario una correcta visualización de la información general comprendida entre los años 2010 y el presente , así como por medio de un gráfico circular presentar la información de la cantidad de estudiantes según las siguientes métricas: matriculados, inscritos, primer curso, cancelados y graduados.					
<b>Observaciones</b>					
Se debe poder visualizar en el gráfico circular la información por semestre.					

Tabla No.2 Requisito funcional()

<b>Nombre:</b> Vista de información general					
<b>Código:</b>	RF-3	<b>Versión:</b>	1.0	<b>Fecha:</b>	26/08/2021
<b>Fuente</b>	Nicolás Aguirre Espinosa	<b>Prioridad</b>	Alta		
<b>Descripción</b>					
El dashboard debe implementar un gráfico de barras que muestre la cantidad de estudiantes a través del tiempo teniendo en cuenta métricas como matrícula, inscripción, cancelación, primer curso y graduación.					
<b>Observaciones</b>					

Tabla No.3 Requisito funcional()

<b>Nombre:</b> Vista de índice de deserción interanual					
<b>Código:</b>	RF-4	<b>Versión:</b>	1.0	<b>Fecha:</b>	26/08/2021
<b>Fuente</b>	Nicolás Aguirre Espinosa	<b>Prioridad</b>	Alta		
<b>Descripción</b>					
El tablero de mando debe brindar la posibilidad de realizar un gráfico de barras de la deserción interanual entre el año 2010 y el presente, teniendo en cuenta diferentes métricas como estudiantes graduados, estudiantes que permanecen en el programa, estudiantes que cambiaron de programa y estudiantes matriculados, así como el programa académico que el usuario elija.					
<b>Observaciones</b>					

Tabla No. 4 Requisito funcional()

<b>Nombre:</b> Vista de índice de deserción intersemestral					
<b>Código:</b>	RF-5	<b>Versión:</b>	1.0	<b>Fecha:</b>	26/08/2021
<b>Fuente</b>	Nicolás Aguirre Espinosa	<b>Prioridad</b>	Alta		
<b>Descripción</b>					
El tablero de mando debe brindar la posibilidad de realizar un gráfico de barras de la deserción intersemestral entre el año 2010 y el presente, teniendo en cuenta diferentes métricas como					

estudiantes graduados, estudiantes que permanecen en el programa, estudiantes que cambiaron de programa y estudiantes matriculados, así como el programa académico que el usuario elija.
<b>Observaciones</b>

Tabla No. 5 Requisito funcional()

<b>Nombre:</b> Vista de inscritos					
<b>Código:</b>	RF-6	<b>Versión:</b>	1.0	<b>Fecha:</b>	26/08/2021
<b>Fuente</b>	Nicolás Aguirre Espinosa	<b>Prioridad</b>	Alta		
<b>Descripción</b>					
El sistema debe contar con un vista destina a la métrica de estudiantes inscritos en la que se pueda visualizar a través de tablas y gráficos la información histórica agrupada por periodo académico, sexo biológico, colegio de procedencia y tipo de estrato socioeconómico.					
<b>Observaciones</b>					

Tabla No.6 Requisito funcional()

<b>Nombre:</b> Vista de matriculados					
<b>Código:</b>	RF-7	<b>Versión:</b>	1.0	<b>Fecha:</b>	26/08/2021
<b>Fuente</b>	Nicolás Aguirre Espinosa	<b>Prioridad</b>	Alta		
<b>Descripción</b>					
El sistema debe contar con un vista destina a la métrica de estudiantes matriculados en la que se pueda visualizar a través de tablas y gráficos la información histórica agrupada por periodo académico, sexo biológico, colegio de procedencia y tipo de estrato socioeconómico.					
<b>Observaciones</b>					

Tabla No. 7 Requisito funcional()

<b>Nombre:</b> Vista de primer curso
--------------------------------------



<b>Código:</b>	RF-8	<b>Versión:</b>	1.0	<b>Fecha:</b>	26/08/2021
<b>Fuente</b>	Nicolás Aguirre Espinosa		<b>Prioridad</b>	Alta	
<b>Descripción</b>					
El sistema debe contar con un vista destina a la métrica de estudiantes de primer curso en la que se pueda visualizar a través de tablas y gráficos la información histórica agrupada por periodo académico, sexo biológico, colegio de procedencia y tipo de estrato socioeconómico.					
<b>Observaciones</b>					

Tabla No.8 Requisito funcional()

<b>Nombre:</b> Vista de cancelados					
<b>Código:</b>	RF-9	<b>Versión:</b>	1.0	<b>Fecha:</b>	26/08/2021
<b>Fuente</b>	Nicolás Aguirre Espinosa		<b>Prioridad</b>	Alta	
<b>Descripción</b>					
El sistema debe contar con un vista destina a la métrica de estudiantes cancelados en la que se pueda visualizar a través de tablas y gráficos la información histórica agrupada por periodo académico, sexo biológico, colegio de procedencia y tipo de estrato socioeconómico.					
<b>Observaciones</b>					

Tabla No. 9 Requisito funcional()

<b>Nombre:</b> Vista de graduados					
<b>Código:</b>	RF-10	<b>Versión:</b>	1.0	<b>Fecha:</b>	26/08/2021
<b>Fuente</b>	Nicolás Aguirre Espinosa		<b>Prioridad</b>	Alta	
<b>Descripción</b>					
El sistema debe contar con un vista destina a la métrica de estudiantes graduados en la que se pueda visualizar a través de tablas y gráficos la información histórica agrupada por periodo académico, sexo biológico, colegio de procedencia y tipo de estrato socioeconómico.					
<b>Observaciones</b>					

--

Tabla No. 10 Requisito funcional()

<b>Nombre:</b> Vista de análisis cohortes					
<b>Código:</b>	RF-11	<b>Versión:</b>	1.0	<b>Fecha:</b>	26/08/2021
<b>Fuente</b>	Nicolás Aguirre Espinosa	<b>Prioridad</b>	Alta		
<b>Descripción</b>					
El sistema debe contar con un vista destina a la métrica de análisis por cohorte en la que se pueda visualizar a través de tablas y gráficos la información histórica agrupada por periodo académico, sexo biológico, colegio de procedencia y tipo de estrato socioeconómico, así mismo, se debe tener en cuenta que lo anterior se debe realizar tanto para programas académico de pregrado como para posgrado.					
<b>Observaciones</b>					

Tabla No.11 Requisito funcional()

<b>Nombre:</b> Vista de deserción interanual					
<b>Código:</b>	RF-12	<b>Versión:</b>	1.0	<b>Fecha:</b>	26/08/2021
<b>Fuente</b>	Nicolás Aguirre Espinosa	<b>Prioridad</b>	Alta		
<b>Descripción</b>					
El sistema debe contar con un vista destina a la métrica de deserción interanual en la que se pueda visualizar a través de tablas y diferentes gráficos la información histórica, la información anual y la información por programa académico					
<b>Observaciones</b>					

Tabla No.12 Requisito funcional()

<b>Nombre:</b> Vista de intersemestral					
<b>Código:</b>	RF-13	<b>Versión:</b>	1.0	<b>Fecha:</b>	26/08/2021
<b>Fuente</b>	Nicolás Aguirre Espinosa	<b>Prioridad</b>	Alta		

<b>Descripción</b>
El sistema debe contar con un vista destina a la métrica de deserción intersemestral en la que se pueda visualizar a través de tablas y diferentes gráficos la información histórica, la información anual y la información por programa académico
<b>Observaciones</b>

**Tabla No.13 Requisito funcional()**

### **3.1.1 Requisitos no funcionales:**

En esta sección se presentarán requerimientos no funcionales del sistema explicando cada uno detalladamente.

- **RNF -1** : El sistema debe ser capaz de procesar N cantidad de transacciones por segundo.
- **RNF -2** : El nuevo sistema debe desarrollarse aplicando patrones de diseño, recomendaciones y buenas prácticas de programación que incrementen la seguridad de datos y la fiabilidad del proyecto.
- **RNF -3** : El software podrá ser utilizado en los sistemas operativos como: Windows, Linux y MAC OS X.
- **RNF - 4:** Todos los componentes web y elementos de frontend deben ser válidos con W3C.
- **RNF -5** : Debe ser un software de arquitectura web y los usuarios deben poder acceder a él por Internet, usando navegadores Internet Explorer 10 o superior, Firefox 23 o superior, Google Chrome 28.0 o superior y Opera.
- **RNF -6** : Las opciones disponibles para el sistema serán visualizadas de manera ordenada y clara, y estarán organizadas.

- **RNF -7** : El sistema debe poder utilizarse sin necesidad de instalar ningún software adicional además de un navegador web.

## 3.2 Desarrollo Conceptual

### 3.2.1 Arquitectura y generalidades de la implementación:

**Arquitectura:** Para la práctica empresarial se hizo uso de una arquitectura orientada a servicios separada en 2 nodos principales, el primero de ellos será un servicio de hosting que temporalmente fue Heroku con una base de datos Postgresql y un backend en Django.

El segundo de los nodos es un servicio de hosting temporalmente en firebase que aloja la parte del frontend y se servirá del mismo, cabe mencionar que cualquier navegador web con acceso a internet tendrá acceso para hacer uso de él.

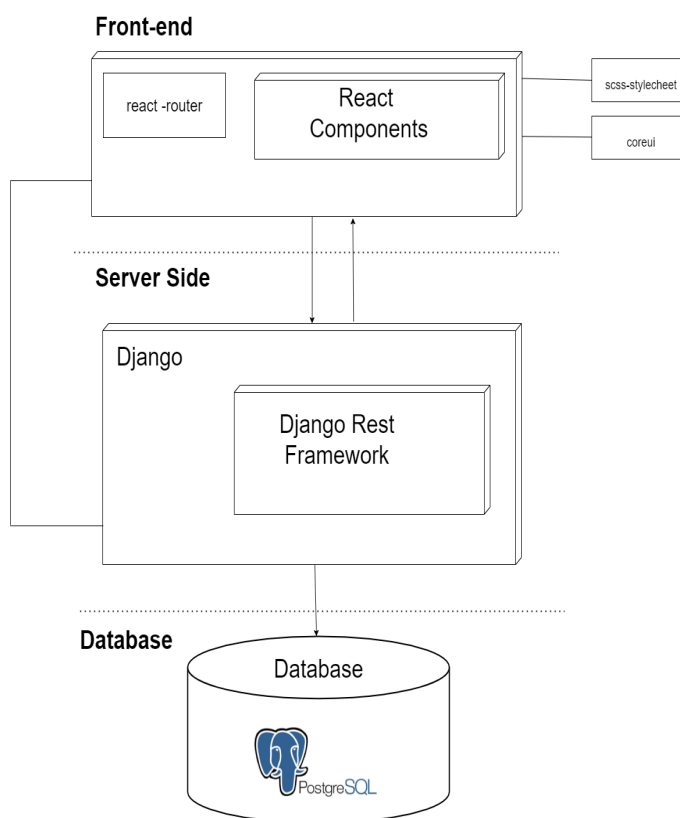


Figura No.2 Arquitectura del Sistema

- **Tecnologías:**
  - **Bases de datos:** La base de datos que se usó en el proyecto fue una base de datos relacional en postgresql inicialmente en elephant SQL para la etapa de desarrollo y posteriormente migrada a Heroku.
  - **Backend:** El backend se desarrolló en el framework Django sobre el motor de Python sobre el motor de Python 3.8.10
  - **Frontend:** El frontend se desarrolló en el framework ReactJS en su versión 16.3.1 y se utilizó la plantilla proporcionada por la herramienta gratuita de código abierto CoreUI en su versión 3.2.1.
- **Comunicación:** La comunicación entre el backend y el frontend se realizó de una API REST proporcionada por el Framework Django que hace parte del backend, desde el navegador se realizarán solicitudes al API y está proporcionará una respuesta permitiendo una comunicación bidireccional.

### **3.3 Ejecución y desarrollo del software**

Para el proceso de desarrollo se realizó un proceso incremental, en el cuál se realizaban reuniones semanales, en las que se planteaban las diferentes funcionalidades y actividades. Cada iteración se considera un incremento que dará como resultado un producto de software completo y que cumpla con los requerimientos planteados.

Para cualquier desarrollo de software se hace evidente la necesidad de tener una planificación clara del producto que se quiere obtener, sin embargo, al hacer uso de la metodología incremental dicha planificación debe ser flexible y adaptable, es decir, durante el desarrollo de la práctica se tomaron en cuenta las etapas planteadas por el modelo en cascada o convencional, pero se adaptaron a las necesidades y requerimientos del proyecto.

#### **3.3.1 Análisis de requisitos**

El análisis de requisitos es el primer paso para un desarrollo de software de alta calidad, dado que, se establecen las bases del software que se comenzará a desarrollar. Se investiga el problema al que se le quiere dar solución y se plantean las pautas de manera clara del sistema que cumplirá con dichos requerimientos. Así mismo, durante cada ciclo de desarrollo se realizó una evaluación constante de los requerimientos funcionales y no funcionales del proyecto, se hizo imprescindible la constante comunicación entre el practicante y el guía de la práctica, dando como resultado los requisitos planteados anteriormente en el apartado 3.1.1.

### **3.3.2 Diseño y arquitectura**

En este punto se determinó de forma general el funcionamiento del tablero de mando. Se realizaron consideraciones sobre la red, el hardware, el software, la base de datos, etc. La arquitectura sirvió como base sobre la estructura interna del software. Para ello, se descompuso y organizó el sistema en diferentes elementos que podían elaborarse por separado. Así mismo, se realizó el esquema en la figura No.2.

### **3.3.3 Programación**

Para la etapa de programación, se hizo uso del editor de código Visual Studio por su facilidad y compatibilidad con la herramienta de versionamiento seleccionada para el proyecto como lo fue Github.

Inicialmente para la recopilación de información se hizo uso del sistema de inteligencia institucional de la Universidad, pero se dejó de lado por su poca eficacia a la hora de exportar los archivos necesarios para la práctica. Por lo anterior, por gestiones del tutor de la práctica se pudo acceder a los archivos de la base de datos en formato EXCEL, los cuales se convirtieron al formato CSV para posibilitar su uso y extracción desde el lenguaje de programación Python.

Dichos archivos sirvieron como fundamento para el planteamiento de la Base de Datos, posteriormente se planteó el backend en Django Rest Framework, ya que permitía hacer filtrado desde los endpoints y facilitaba su consumo desde el frontend a través de React.



### **3.3.4 Documentación**

La documentación realizada del proyecto es el presente documento, que tiene los diferentes esquemas y explicaciones necesarias para el entendimiento de todo el proceso de la práctica. Así mismo en los diferentes repositorios de Github destinado para el despliegue, se dejó una breve explicación y guía de instalación.

### **3.4 Interfaz de Usuario**

Para el aplicativo web desarrollado no es necesario ningún tipo de instalación previa ya que se ejecuta en el navegador.

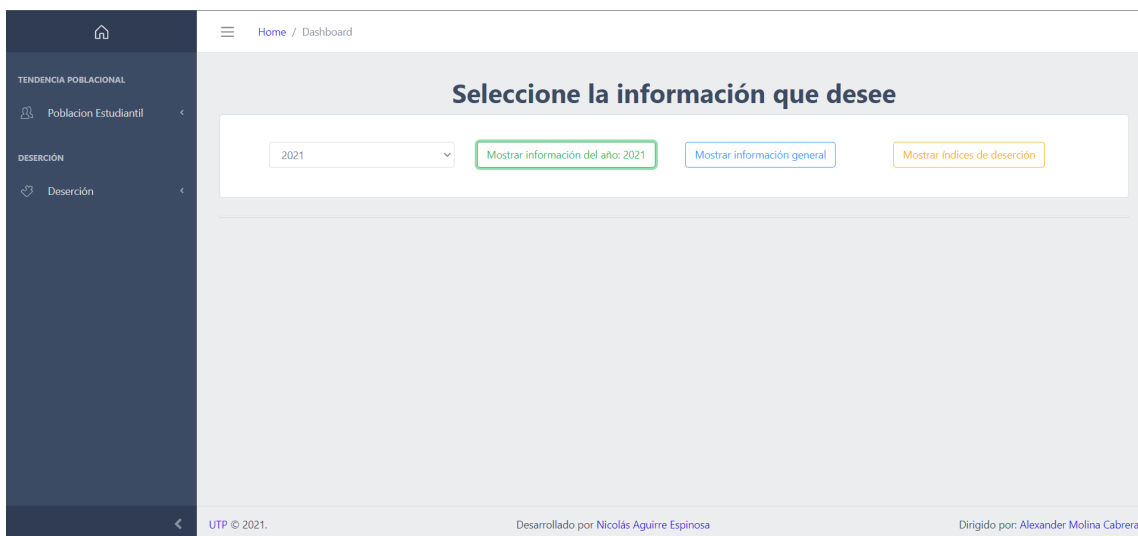
Dando más claridad a las características lógicas y visuales de las funcionalidades y servicios que se ofrecen en la plataforma y el cómo interactúa con los usuarios, se mostrarán a continuación diferentes imágenes y textos explicativos sobre cada una de las funcionalidades desarrolladas.

El dashboard cuenta con tres secciones principales las cuales son la vista de la homepage, la vista de población estudiantil y la vista de deserción, cada una de estas con diferentes subcategorías.

#### **3.4.1 Homepage**

La homepage es la página principal e inicial a la hora de acceder al aplicativo, tiene un resumen de las diferentes métricas que se manejan en todo el dashboard.

Tiene tres subdivisiones: mostrar información por año, mostrar información general y mostrar índices de deserción, cada una cuenta con un botón que permite su visualización, como se muestra en la siguiente imagen:



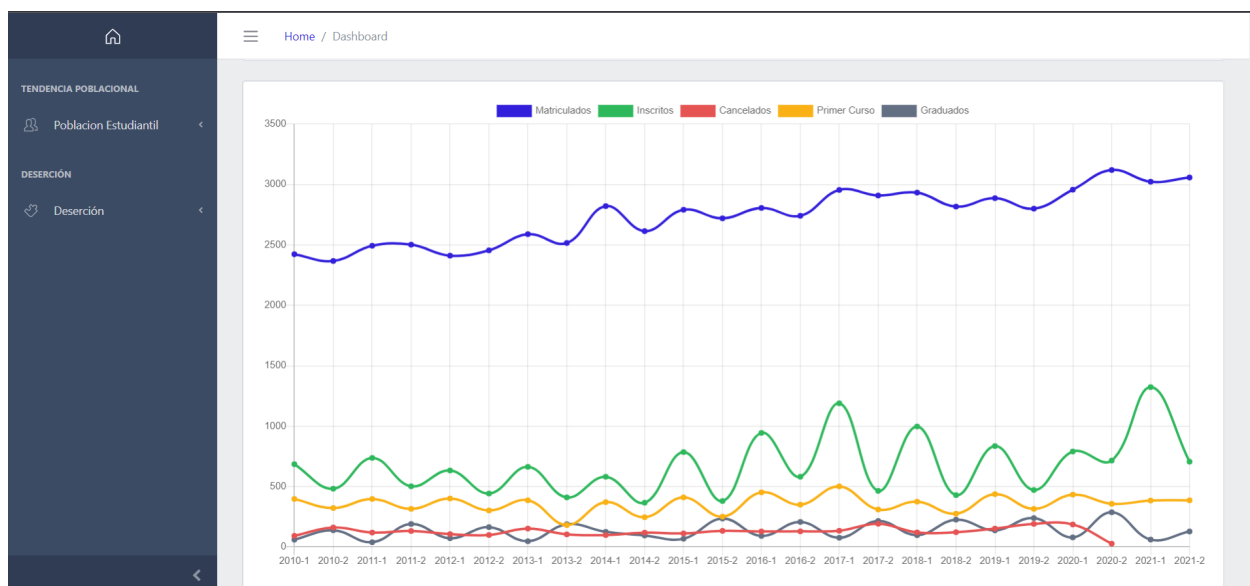
**Figura No.3 Homepage**

La sección de información anual tiene un botón con el intervalo de años que se maneja en la base de datos, como lo fue 2010 al presente año que muestra las métricas de la cantidad de estudiantes inscritos, matriculados, de primer curso, cancelados y graduados, así como dos gráficos circulares con las métricas anteriormente mencionadas por semestre.



**Figura No.4 Homepage sección anual**

La sección de información general muestra la tendencia poblacional a través de una gráfica lineal desde el año 2010 hasta el presente año según las métricas de la cantidad de estudiantes inscritos, matriculados, de primer curso, cancelados y graduados.

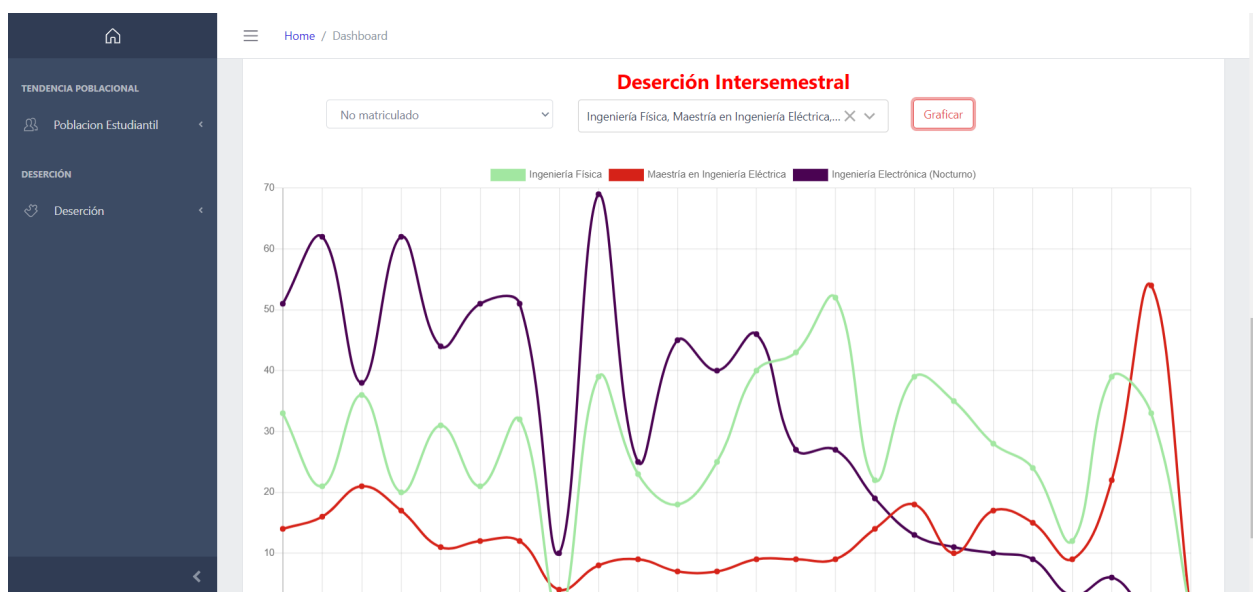


**Figura No.5 Homepage sección general**

La sección de índices de deserción se divide en dos: en deserción interanual e intersemestral, cada una consta de tres botones, el primero permite seleccionar según el estado( graduado, permanece programa, cambio de programa y no matriculado), el segundo es para elegir el o los programas académicos que se quieren analizar y el tercero es para controlar la visualización del gráfico que se despliega.



**Figura No.6 Homepage deserción interanual**



**Figura No.7 Homepage deserción intersemestral**

### 3.4.2 Población Estudiantil

La sección de población estudiantil tiene 6 subdivisiones: inscritos, matriculados, primer curso, cancelados, graduados y análisis cohortes.

La subsección de inscritos, matriculados, primer curso, cancelados y graduados consta de 5 divisiones: tabla general, según sexo, según estrato, según colegio y según inscripción, cada una de las anteriores muestra a través de gráficos o estadísticas la cantidad de estudiantes según la métrica donde el usuario se encuentre ubicado.

The screenshot shows a web application interface for 'Inscritos'. On the left is a dark sidebar with a home icon at the top. Below it, the section 'TENDENCIA POBLACIONAL' is expanded to show 'Poblacion Estudiantil', which includes sub-items: 'Inscritos', 'Matriculados', 'Primer Curso', 'Cancelados', 'Graduados', and 'Análisis Cohortes'. Below this is the 'DESERCIÓN' section with 'Deserción'. The main content area has a breadcrumb 'Home / Inscritos' and a title 'Inscritos'. A text box contains the following text: 'Para tener en cuenta: En la siguientes tablas y gráficos se muestran la información de Inscritos de forma histórica agrupado por periodo académico, sexo biológico, colegio de procedencia y tipo de estrato socioeconómico.' Below this text are five buttons: 'Tabla General' (purple border), 'Según Sexo' (green border), 'Según Estrato' (orange border), 'Según Colegio' (red border), and 'Según Inscripción' (blue border). The footer contains 'UTP © 2021.', 'Desarrollado por Nicolás Aguirre Espinosa', and 'Dirigido por: Alexander Molina Cabrera'.

**Figura No.9 Población estudiantil inscritos**

Home / Matriculados

## Matriculados

Para tener en cuenta:

La **matricula** es el acto que formaliza la vinculación del estudiante al servicio educativo, el cual se renueva cada periodo académico.

El **régimen especial** hace referencia a las minorías desplazados, comunidad indígena, negritudes y deportistas de alto rendimiento

En la siguientes tablas y gráficos se muestran la información de Matriculados de forma histórica agrupado por periodo académico, sexo biológico, colegio de procedencia y tipo de estrato socioeconómico.

[Tabla General](#)
[Según Sexo](#)
[Según Estrato](#)
[Según Colegio](#)
[Según Inscripción](#)

UTP © 2021. Desarrollado por Nicolás Aguirre Espinosa Dirigido por: Alexander Molina Cabrera

**Figura No.10 Población estudiantil matriculados**

Home / Primer Curso

## PrimerCurso

Para tener en cuenta:

En la siguientes tablas y gráficos se muestran la información de PrimerCurso de forma histórica agrupado por periodo académico, sexo biológico, colegio de procedencia y tipo de estrato socioeconómico.

[Tabla General](#)
[Según Sexo](#)
[Según Estrato](#)
[Según Colegio](#)
[Según Inscripción](#)

UTP © 2021. Desarrollado por Nicolás Aguirre Espinosa Dirigido por: Alexander Molina Cabrera

**Figura No.11 Población estudiantil primer curso**

Home / Graduados

## Graduados

Para tener en cuenta:

Un **graduado** es cualquier persona natural que, previa culminación del programa académico y cumpliendo los requisitos de ley y los exigidos por la institución de educación superior, recibe el título académico. En los siguientes gráficos se muestra la información de los graduados agrupados por periodo académico, por nivel (Pregrado y Posgrado), subnivel de formación, por facultad, programa académico, área de conocimiento, sexo biológico. El reporte es interactivo seleccionado una facultad las gráficas cambian en función a la facultad seleccionada. .

En la siguientes tablas y gráficos se muestran la información de Graduados de forma histórica agrupado por periodo académico, sexo biológico, colegio de procedencia y tipo de estrato socioeconómico.

[Tabla General](#)
[Según Sexo](#)
[Según Estrato](#)
[Según Colegio](#)
[Según Inscripción](#)

UTP © 2021. Desarrollado por Nicolás Aguirre Espinosa Dirigido por: Alexander Molina Cabrera

**Figura No.12 Población estudiantil graduados**

Home / Cancelados

## Cancelados

Para tener en cuenta:

En la siguientes tablas y gráficos se muestran la información de Cancelados de forma histórica agrupado por periodo académico, sexo biológico, colegio de procedencia y tipo de estrato socioeconómico.

[Tabla General](#)
[Según Sexo](#)
[Según Estrato](#)
[Según Colegio](#)
[Según Inscripción](#)

UTP © 2021. Desarrollado por Nicolás Aguirre Espinosa Dirigido por: Alexander Molina Cabrera

**Figura No.13 Población estudiantil cancelados**

Como mencioné anteriormente las subsecciones de inscritos, matriculados, primer curso, cancelados y graduados comparten el mismo diseño por lo que no se considera necesario mostrar evidencias de cada una de ellas. En este caso se tomará la sección de inscritos.



En primer lugar se tiene la subdivisión de tabla general que consta de un botón para elegir el año que desea visualizar y dos más que permiten visualizar a través de tablas tanto el primer semestre como el segundo del año anteriormente elegido. Así mismo, se puede hacer filtrado en por cada una de las columnas (codigo periodo, estudiantes, estrato, sexo,tipo de inscripción y tipo de colegio.)

COD PERIODO	ESTUDIANTES	ESTRATO	SEXO	TIPO INSCRIPCIÓN	TIPO COLEGIO
2021-1	1	III	Masculino	General	Oficial
2021-1	3	null	Masculino	null	Na
2021-1	1	I	Femenino	Desplazado	Oficial
2021-1	1	I	Femenino	Negritudes	Oficial
2021-1	1	III	Femenino	null	Oficial

**Figura No.14 Población estudiantil tabla general**

En segundo lugar se tiene la subdivisión según sexo, que permite visualizar a través de gráficos circulares la cantidad de estudiantes según el año que seleccione el usuario.



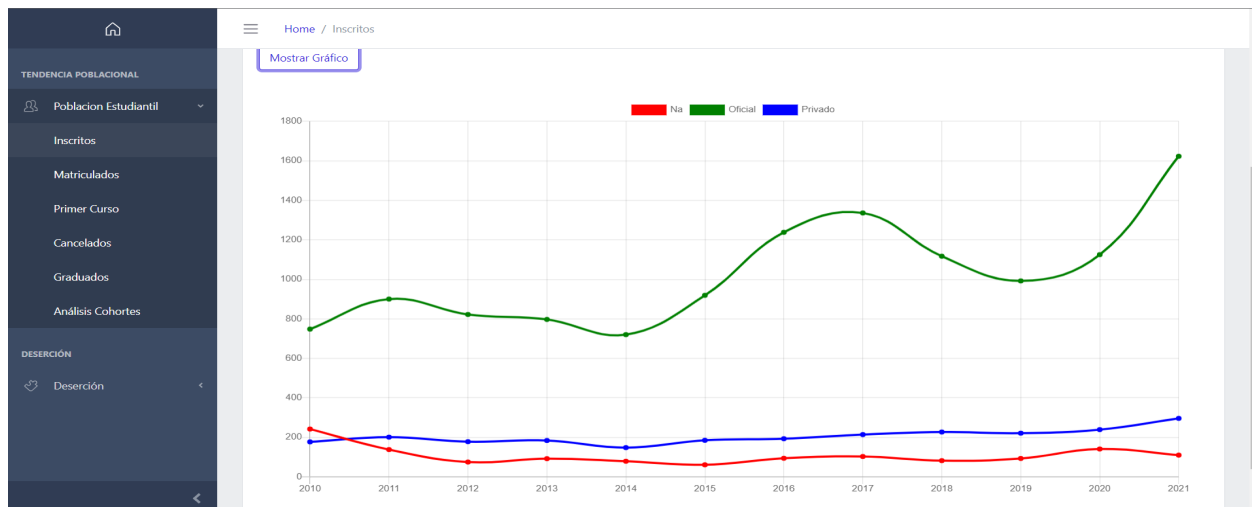
**Figura No.15 Población estudiantil segun sexo**

En tercer lugar se encuentra la sección según estrato, que permite visualizar a través de un gráfico de barras la cantidad de estudiantes en un año determinado tanto del primer semestre como del segundo semestre en los diferentes estratos.



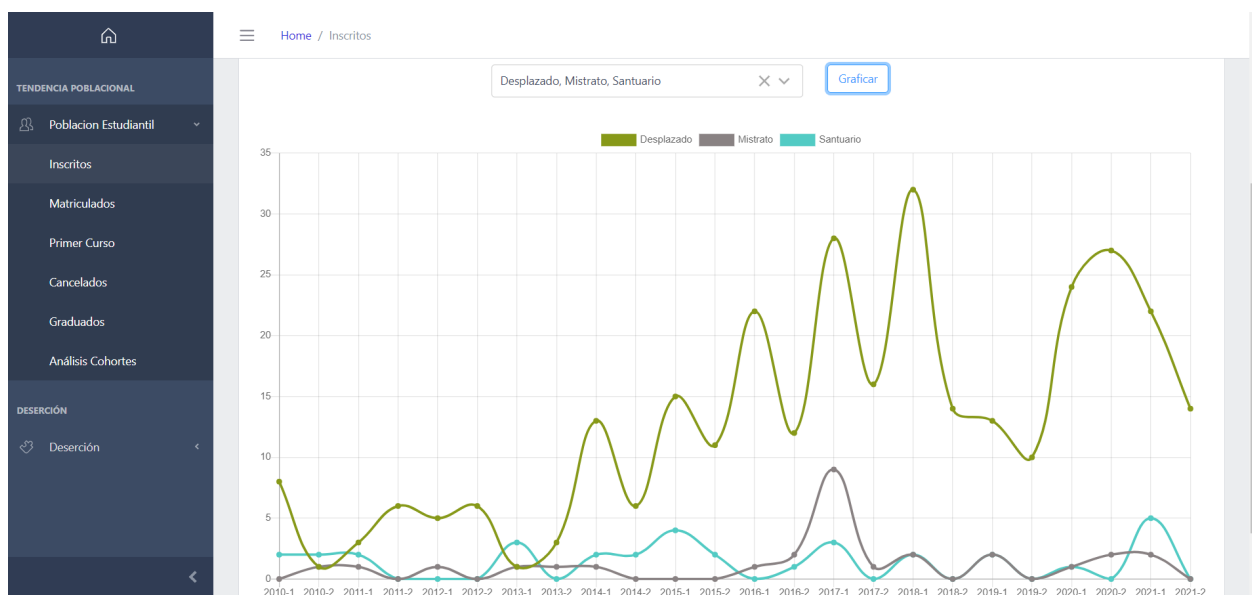
**Figura No.16 Población estudiantil según estrato**

En cuarto lugar se encuentra la sección según colegio, que permite visualizar a través de un gráfico lineal la cantidad de estudiantes en un intervalo de tiempo en este caso de 2010 al presente año teniendo en cuenta métricas según el colegio de procedencia como no aplica(Na), colegio oficial y colegio privado.



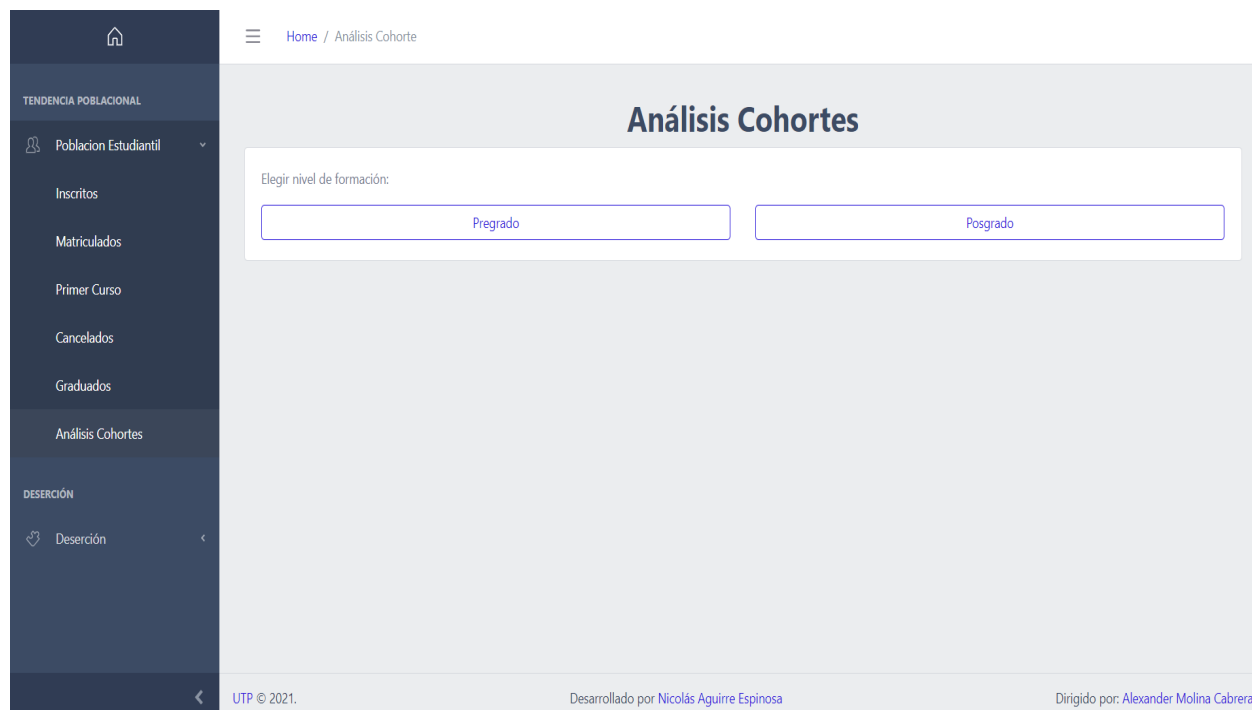
**Figura No.16 Población estudiantil según colegio**

Por último, se encuentra la sección según tipo de inscripción consta de dos botones, el primero para elegir el o los tipos de inscripción que se quieren analizar y el segundo es para controlar la visualización del gráfico que se despliega.



**Figura No.17 Población estudiantil según tipo de inscripción**

Por otro lado, se tiene la sección de análisis de cohorte que subdivide en dos: pregrado y posgrado como se muestra en la figura No.18:



**Figura No.18 Población estudiantil análisis cohortes**

En la sección de análisis de cohorte, al seleccionar tanto el nivel de formación de pregrado como de posgrado, se desplegará un sección para ver información general o para ver información por programa académico.

En la subdivisión de información general se puede visualizar la cantidad de estudiantes según año, semestre y permite visualizar toda esta información en una tabla que permite filtrar por columna, todo esto se puede observar en la figura No.

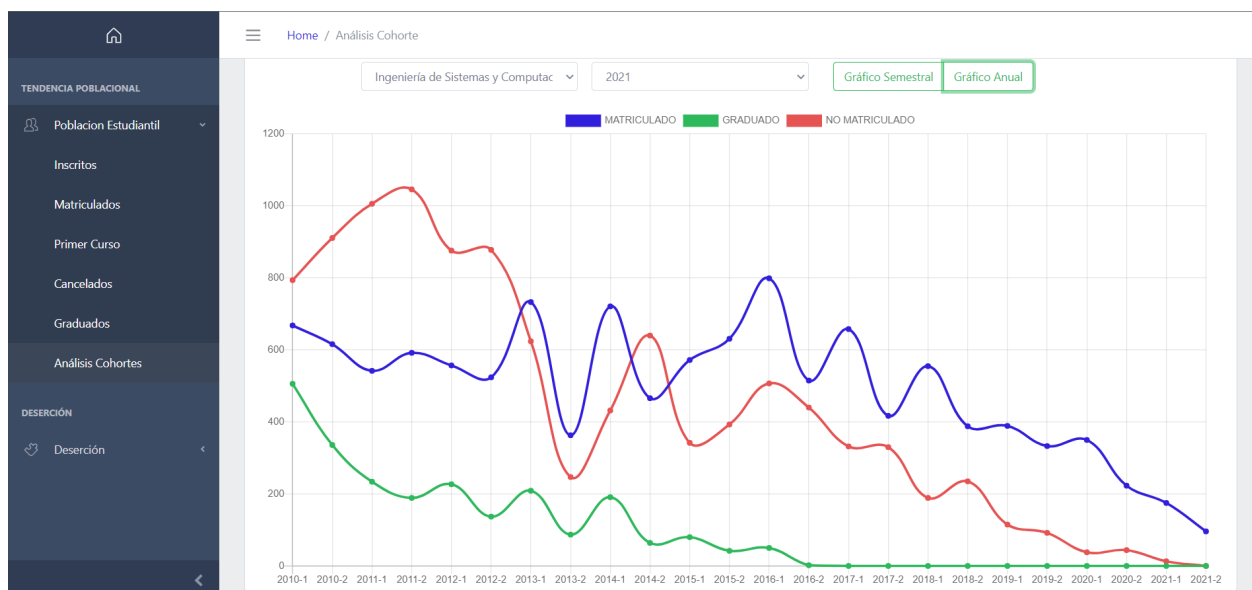
**Figura No.18 Población estudiantil pregrado**

**Figura No.19 Población estudiantil análisis cohorte general**

En la subdivisión de información por programa académico existen dos botones primarios, en el primero el usuario selecciona el programa del cual quiere saber la información y el segundo el año. Luego los gráficos se dividieron en dos por semestre como en la figura No.20 y anualmente como en la figura No.21.



**Figura No.20 Población estudiantil análisis cohorte semestral**



**Figura No.21 Población estudiantil análisis cohorte anual**

### 3.4.3 Deserción

La página de deserción se divide en deserción intersemestral y deserción interanual, cada una de las cuales tiene una breve explicación y definición de los diferentes términos que se pueden encontrar en esta sección, así mismo, agrupan las estadísticas y gráficos por información histórica, información anual e información por programa, como se muestra en la figura No.22.

Home / Interanual

## Deserción Interanual

Para tener en cuenta:

**Definición:** La deserción interanual representa el porcentaje de estudiantes matriculados en un periodo académico que se ausentan de la institución durante dos periodos consecutivos. El ministerio de Educación Nacional define este porcentaje: **No matriculados / (Matricula total del semestre)** teniendo en cuenta las siguientes consideraciones:

**Período base (n):** Período para el cual se está calculando la deserción.

**Graduado:** Estudiante que esta matriculado en el período base (n) que obtiene su título en el mismo período base (n), o en alguno de los semestres siguientes (n+1 o n+2) Nota: Este valor no coincide con el total de graduados en un periodo, porque se toman 3 periodos de observación (n, n+1 y n+2).

**Permanece Programa:** Estudiantes que estaba matriculado en n y está matriculado en el periodo n+1 o n+2 sin cambiar de programa académico.

**Cambio de Programa:** Estudiantes que estaba matriculado en n y está matriculado en el periodo n+1 o n+2 en otro programa académico

**No matriculado:** Estudiantes que estaba matriculado en n y no aparece con estado Graduado o Permanece programa o Cambio programa.

Seleccione una de las siguientes opciones:

Mostrar Información Histórica    Mostrar Información Anual    Mostrar Información por Programa

Figura No.22 Deserción

Cuando el usuario se dirige a información histórica se encuentra un tabla que contiene todos los datos desde el año 2010 hasta el presente, también al igual que las demás tablas del dashboard permite hacer búsquedas por cada columna, en este caso por código de periodo, nombre de programa académico, duración en semestres del programa, estado y cantidad de estudiantes, lo anterior se muestra en la figura No.23. Por otro lado permite visualizar a través de un gráfico de

barras diferentes métricas como estudiantes graduados, estudiantes que cambian de programa, estudiantes no matriculados y estudiantes que permanecen en el programa como se muestra en la

figura No.24



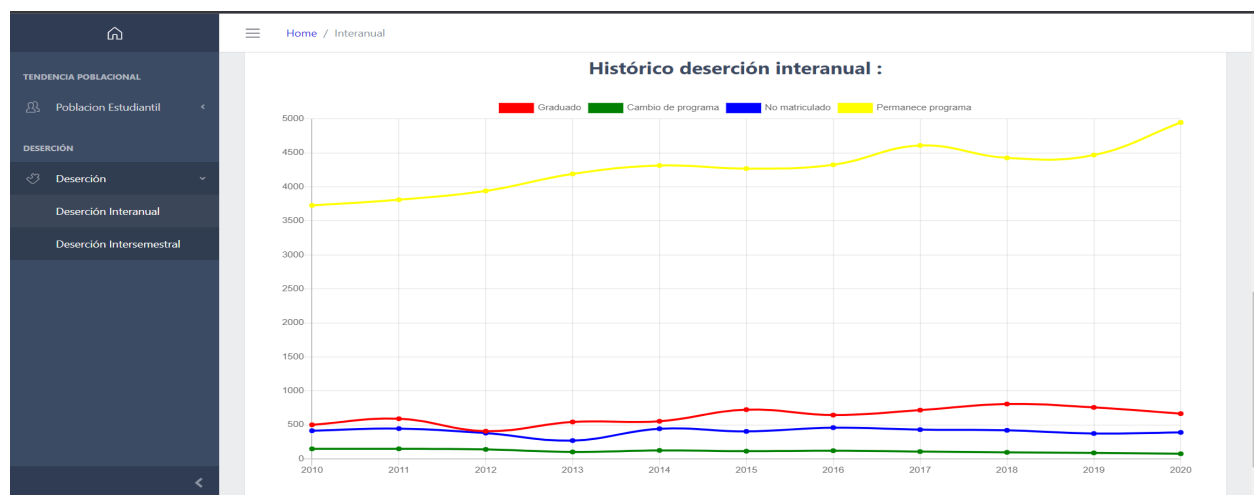
Home / Interanual

Mostrar Información Histórica    Mostrar Información Anual    Mostrar Información por Programa

**Tabla histórica de estudiantes por estado:**

COD PERIODO	NOMBRE	DURACION SEMESTRES	ESTADO	CANTIDAD
2013-1	Ingeniería de Sistemas y Computación	10	Cambio de programa	19
2013-2	Ingeniería Física	10	Graduado	26
2014-2	Maestría en Ingeniería Eléctrica	4	Graduado	28
2015-1	Ingeniería Electrónica (Nocturno)	10	Permanece programa	218
2015-2	Ingeniería Eléctrica	10	Cambio de programa	17
2016-1	Doctorado en Ingeniería	7	Permanece programa	32

**Figura No.23 Deserción histórica tabla**



**Figura No.24 Deserción histórica gráfico**

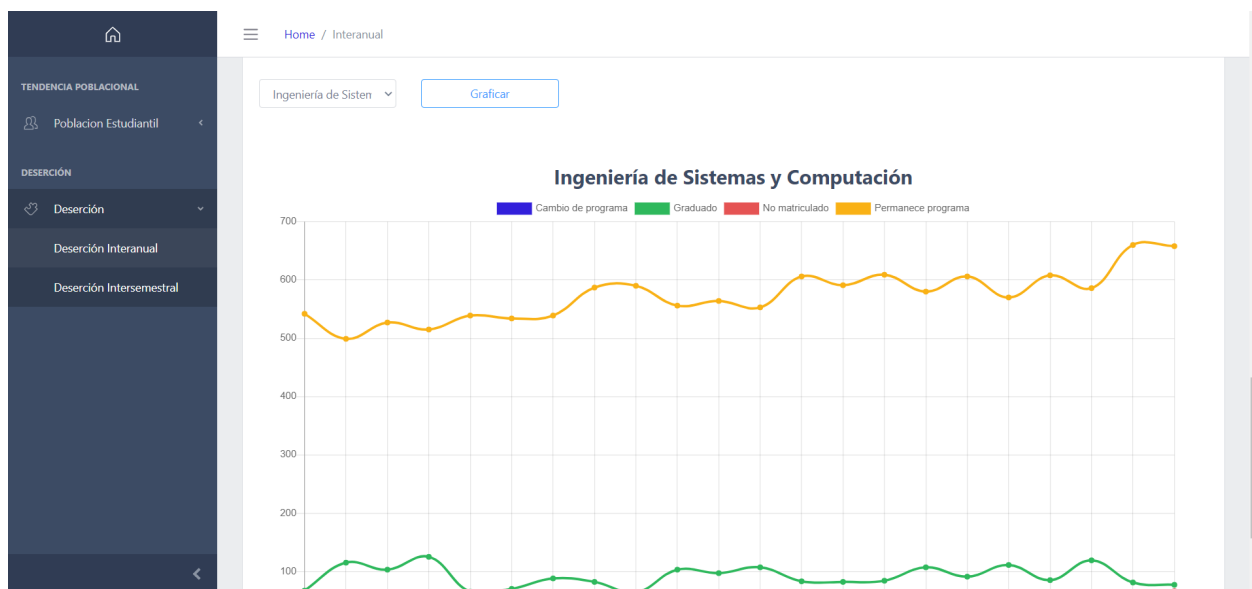
Por el lado de información anual permite visualizar información general según nivel de formación que puede ser pregrado y posgrado.





**Figura No.25 Deserción anual**

La subdivisión de información por programa académico divide su información en pregrado y posgrado cada uno de ellos permite visualizar un tabla general de todos los programas académicos asociados al nivel formación seleccionado y abajo de esto permite seleccionar al programa académico al que se graficara en el tiempo.



**Figura No.25 Deserción por programa académico**

## **4. Trabajos Futuros**

Como continuación al proyecto de software, es decir, el tablero de mandos o dashboard realizado durante la práctica se pretende mejorar la usabilidad de los usuarios en dispositivos móviles, principalmente teniendo en cuenta las limitaciones que ofrece la librería CoreUI a la hora de manejar el responsive en la graficación de los datos.

Por otro lado, se propone la posibilidad en el backend haciendo uso de las facultades y ventajas proporcionadas por el lenguaje de programación Python, principalmente de la biblioteca de código abierto TensorFlow, el desarrollo de un algoritmo predictivo enfocado a la deserción estudiantil intersemestral e interanual.

También se pretende agregar un sistemas de registro y autenticación que se comunique con la base de datos de la universidad.

## **5. Conclusiones**

El tablero de mando o dashboard permite apoyar a la Facultad de Ingeniería a través de diferentes gráficos en el análisis de la deserción y tendencia poblacional definiendo múltiples métricas como el tipo de inscripción, sexo, tipo de colegio, estrato, programa académico, etc.

Este trabajo de grado recoge toda la evidencia de todo el proceso de la práctica empresarial llevada a cabo en la Facultad.

La metodología utilizada permite un constante y correcto seguimiento de las actividades realizadas por el practicante así como la integración de las diferentes funcionalidades sugeridas tras cada ciclo de desarrollo.

## **6. Recomendaciones**

Se recomienda desarrollar una herramienta de consulta y descarga automática de los sistemas de inteligencia institucional que permita exportar las diferentes consultas en formato CSV para su posterior uso en la base de datos.

Así mismo, se sugiere buscar un servidor con la capacidad pertinente para albergar tanto la base de datos como los servicios de frontend y backend.

## 7. Bibliografía

[1] Vélez, A. y López, D. (2004). Estrategias para vencer la deserción universitaria. Revista Educación y Educadores, 7, 177-203.

<http://educacionyeducadores.unisabana.edu.co/index.php/eye/article/view/555>

[2] Estrategias para vencer la deserción universitaria | Educación y Educadores. (s. f.). Educación y Educadores. <http://educacionyeducadores.unisabana.edu.co/index.php/eye/article/view/555>

[3] Pressman, R. S. (2001). Software Engineering: A Practitioner's Approach. New York: McGraw-Hill Higher Education.

[4]T, Neha. 2020. «What Is an Incremental Development Model? Characteristics, Use, Types, Advantages & Disadvantages – Binary Terms». Binary Terms. <https://binaryterms.com/incremental-development-model.html>.

[5] *JavaScript*|MDN.(s.f). MDN Web Docs. <https://developer.mozilla.org/es/docs/Web/JavaScript>

[6] ¿Qué es JavaScript? - Aprende sobre desarrollo web | MDN. (s. f.). MDN Web Docs. [https://developer.mozilla.org/es/docs/Learn/JavaScript/First\\_steps/What\\_is\\_JavaScript](https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/What_is_JavaScript)

[7] JavaScript | MDN. (s. f.). MDN Web Docs. <https://developer.mozilla.org/es/docs/Web/JavaScript>

[8] What Is PostgreSQL? (s. f.). PostgreSQL Documentation. <https://www.postgresql.org/docs/14/intro-what-is.html>

[9] Empezando – react. (s. f.). React – Una biblioteca de JavaScript para construir interfaces de usuario. <https://es.reactjs.org/docs/getting-started.html>

[10] Qué es React.js y cómo funciona. (s. f.-b). Platzi. <https://platzi.com/blog/react-js-de-javascript/>

[11]Introducción a Django - Aprende sobre desarrollo web | MDN. (s. f.). MDN Web Docs. <https://developer.mozilla.org/es/docs/Learn/Server-side/Django/Introduction>

[12] Tablado, F. (2020, 9 de septiembre). Qué es PostgreSQL y sus principales ventajas <https://ayudaleyprotecciondatos.es/bases-de-datos/que-es-postgresql-ventajas/>