

**PROYECTO DE ALGORITMIA Y PROGRAMACIÓN**

**ENTREGA 1**

**PRESENTADO A:**

### **CAMILO LLERENA RIASCOS**

**(Docente)**

**INTEGRANTES:**

**MATEO BOTERO**

**ALEXANDRA VELEZ**

**UNIVERSIDAD DE ANTIOQUIA**

**DEPARTAMENTO DE INGENIERÍA INDUSTRIAL**

**ALGORITMIA Y PROGRAMACIÓN**

**14/09/2024**

**TABLA DE CONTENIDO**

Descripción general del software............................................................................................. 3

Objetivo general......................................................................................................................3

Objetivos especificos............................................................................................................. 3

Beneficios.............................................................................................................................. 4

Requisitos del software.......................................................................................................... 4

Actividades y Cronograma tentativo...................................................................................... 6

Presupuesto ........................................................................................................ 6

Plan versionado.................................................................................................. 7

Flujograma..................................................................................................... 8

Descripción del código........................................................................................ 9

**Descripción general del software**

Este proyecto se enfoca en desarrollar un sistema automatizado en Python para la planificación académica semestral del programa de Ingeniería Industrial de la Universidad de Antioquia. El software gestionará la asignación de 1000 estudiantes a los cursos correspondientes, respetando la capacidad de las aulas y la secuencia de la malla curricular. Generará listados de estudiantes por curso en formatos Excel y CSV, organizados por semestre y asignatura, almacenados en una estructura de carpetas clara. Además, simulará el rendimiento académico, asegurando que al menos el 70% de los estudiantes aprueben, optimizando así la gestión académica y administrativa.

**Objetivo General**

Desarrollar un sistema automatizado en Python que optimice la planificación académica semestral del programa de Ingeniería Industrial de la Universidad de Antioquia, gestionando de manera eficiente la asignación de estudiantes a las diferentes asignaturas, y generando listados organizados para su uso administrativo.

**Objetivos Específicos**

* Diseñar un algoritmo que asigne a los 1000 estudiantes de manera equitativa a las distintas asignaturas y grupos, maximizando la capacidad de las aulas y asegurando una distribución balanceada.
* Implementar un proceso que automatice la creación de listados de estudiantes por curso en formatos Excel y CSV, organizados por semestre y asignatura, el código del curso, nombre de la asignatura, cantidad de estudiantes
* Incorporar un módulo que permita simular y analizar el rendimiento académico de los estudiantes, generando notas de manera aleatoria para evaluar el progreso y asegurar la calidad educativa.
* Establecer una estructura clara y eficiente para el almacenamiento de los datos académicos, asegurando un fácil acceso y organización de la información para su posterior análisis.
* Hay que asegurar que el software desarrollado sea compatible e integrable con otros sistemas administrativos y académicos de la universidad, facilitando una gestión unificada y coherente de los procesos educativos.

**Beneficios**

* Minimizar significativamente el tiempo dedicado a la planificación semestral, optimizando así el flujo de trabajo del personal académico.
* Promueve un ambiente educativo más innovador al liberar a los docentes y administradores de tareas repetitivas, permitiéndoles concentrarse en el desarrollo de nuevas metodologías y enfoques educativos que beneficien a los estudiantes.
* Distribución justa de estudiantes en los cursos, previniendo la saturación de aulas y asegurando una experiencia educativa equilibrada.
* Mejora el flujo de información entre diferentes departamentos de la universidad, lo que permite una gestión más coherente de los datos académicos y administrativos, optimizando la colaboración interdepartamental.
* Facilita la implementación de ajustes rápidos ante cambios en la cantidad de estudiantes o modificaciones en el currículo, asegurando que el sistema se mantenga relevante y efectivo en distintas circunstancias académicas.
* Permite el seguimiento continuo del desempeño de los estudiantes, ayudando a identificar aquellos que necesitan apoyo adicional y asegurando que se tomen medidas a tiempo para mejorar su éxito académico.

**ESPECIFICACIÓN DE REQUISITOS DEL SOFTWARE DE PLANIFICACIÓN ACADÉMICA**

**Requisitos funcionales**

• **Generación automática de listas de clases**: Cuando comienza el proceso de planificación de un nuevo semestre, el sistema debe generar automáticamente listas de clases para todas las materias de la carrera de Ingeniería Industrial, asignando a los estudiantes a cursos específicos según el porcentaje proporcional establecido.

**• Asignación equitativa de estudiantes a los cursos**: Si una materia tiene un límite de 30 estudiantes por aula y hay 90 estudiantes matriculados, el sistema debe distribuir a los estudiantes en tres grupos de manera equitativa, evitando sobrecargar un grupo sobre los demás, y si hay Si es un grupo con menos estudiantes de lo esperado, el sistema permitirá realizar ajustes manuales a la tarea.

**• Registro de eventos y auditoría de procesos:** El sistema debe registrar cada acción realizada por los usuarios, como crear listas de clases, que serán accesibles para los desarrolladores y administradores, modificar las tareas de los estudiantes y generar informes de auditoría para revisarlos más adelante.

* **Tipo de usuario**: El sistema debe de tener tipos de usuario para cada requerimiento, que los administradores sean quienes tienen acceso a los listados csv, los cuales contienen los datos de todos los estudiantes y capacidad de modificar los cursos y las notas de dichos estudiantes, y la interfaz de usuario donde el estudiante puede ver sus notas y a que grupo fue asignado.
* **Manejo de casos especiales:** El sistema debe tener como manejar casos especiales, como por ejemplo los estudiantes inscritos tarde o que el sistema los haya inscrito en un grupo que no los debía inscribir y los administradores y desarrolladores puedan tener acceso a ello.
* **Validación de datos:** El sistema deberá de reconocer cuando se equivoca al introducir información errada como tipo de dato o extensión de un número de registro.

**Requisitos no funcionales:**

* **Interfaz de usuario**: El sistema debe tener una interfaz gráfica amigable con los usuarios, una interfaz que sea intuitiva y fácil de utilizar.

**• Rendimiento**: El sistema debe ser capaz de generar listas de clases para un número variable de estudiantes y materias, adaptándose a las necesidades cambiantes de la carrera de Ingeniería Industrial sin comprometer el rendimiento.

• **Seguridad y privacidad de los datos**: El sistema debe implementar medidas de seguridad como autenticación de usuarios, cifrado de datos y control de acceso para proteger la información confidencial de los estudiantes.

**Plan de proyecto**

**Actividades Y Cronograma tentativo**

**(Adjuntado en Excel)**

**PRESUPUESTO**

Al momento de la primera entrega se llevan acumuladas 27 horas de trabajo (9 cada integrante).

El Salario Mínimo Legal Vigente en Colombia es **$1,300,000 COP**

***Cálculo del valor por hora de práctica profesional:***

El salario mínimo legal vigente cubre 48 horas semanales, lo cual equivale a 192 horas al mes (4 semanas).

Entonces, el valor por hora sería:

El presupuesto total, basado en las 27 horas de trabajo invertidas por los tres integrantes, sería $182,812.41 COP en tiempo de práctica profesional.

**PLAN VERSIONADO**

***Versión 1.0***

* Creación de la estructura base del programa en Python iniciando con la capacidad de cargar los datos de 1000 estudiantes con sus respectivos nombres

#### **Versión 1.1**

* Ahora, se tiene la capacidad de asignar a cada estudiante a los respectivos cursos según el semestre que se encuentra

#### **Versión 1.2**

* En esta nueva versión, el programa es capaz de asignar a los estudiantes en cada aula, ahora, teniendo en cuenta no sobrepasar la capacidad de cada una

#### **Versión 1.3**

* Con los grupos armados, el programa puede generar el listado según el aula, para así, simular y asignar notas aleatorias a cada estudiante

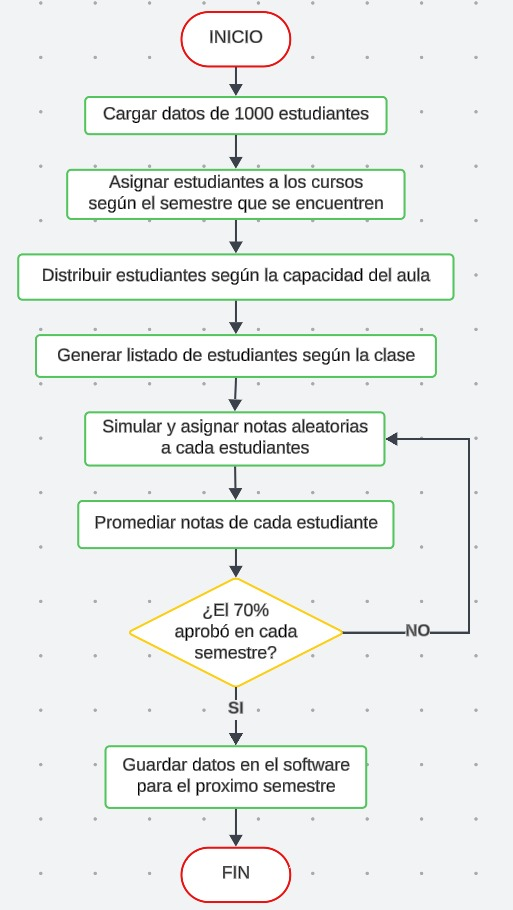
#### **Versión 1.4**

#### Según las notas de cada estudiante en sus respectivas materias, se tiene la capacidad de sacar el promedio y determinar si el estudiante aprobó o reprobó el semestre

#### **Versión 1.5**

* Ahora para organizar mejor los datos para posteriores semestres, los cursos y grupos tendrán asignado un código para identificarlos mejor, de igual forma, las listas de estudiantes de cada asignatura y su situación académica serán guardadas en archivos de Excel y en un archivo plano tipo CSV.

**FLUJOGRAMA**

****

**Inicio**

**Cargar datos de 1000 estudiantes:**

* Se cargan o registran los datos de 1000 estudiantes que ingresaron a la universidad, estos pueden incluir información como el nombre y semestre en que se encuentra.

**Asignar estudiantes a los cursos según el semestre que se encuentren:**

* Se agrupan a los estudiantes de acuerdo con el semestre en el que están matriculados, estos serán asignados a las clases correspondientes según el plan de estudios para su nivel académico.

**Distribuir estudiantes según la capacidad del aula:**

* Después de la asignación a los cursos, se distribuyen a los estudiantes en aulas, respetando la capacidad máxima permitida en cada salón. En este caso, si un aula tiene una capacidad de 30 estudiantes y hay 60 estudiantes en un curso, se dividirán en dos grupos de 30.

**Generar listado de estudiantes según la clase:**

* Luego, se genera una lista que contiene los estudiantes asignados a cada clase, permitiendo una gestión adecuada por parte de los profesores o administradores.

**Simular y asignar notas aleatorias:**

* En esta etapa, el sistema simula el rendimiento académico de los estudiantes, asignando notas aleatorias, para así, evaluar su desempeño hipotéticamente en las clases a las que fueron asignados.

**Promediar notas de cada estudiante:**

En esta etapa, el sistema calculará el promedio de las notas asignadas a cada estudiante para obtener su rendimiento académico global en las materias asignadas. Esto permitirá evaluar de manera más precisa si el estudiante ha cumplido con los requisitos para aprobar el semestre o el curso.

**¿El 70% aprobó?:**

* Aquí, se realiza una verificación para determinar si al menos el 70% de los estudiantes aprobó el curso. Esta condición señala:

- Si no aprueba el 70%, el proceso regresa a la etapa de simulación, es decir, se vuelven a generar nuevas notas aleatorias hasta que el 70% de los estudiantes haya aprobado.

- Si aprueba el 70%, se continúa al siguiente paso del proceso.

**Guardar datos en el software para el próximo semestre:**

* Los resultados obtenidos, como los estudiantes que aprobaron y los que no, se guardan en el sistema para preparar las asignaciones y seguimiento en el próximo semestre.

**Fin**