



Universidad de Antioquia
Facultad de ingeniería
Algoritmia y programación
Semestre 2024-2

Nombre:	Juan Diego Ocampo Correa	Identificación:	1011392687
Nombre:	Sharon Dailly Parra López	Identificación:	1037120253
Nombre:	Valerie Domínguez Toro	Identificación:	1099739017

Nombre del proyecto : InduSurvivors

Descripción general del software:

El proyecto InduSurvivors consiste en el desarrollo de un software creado en Python, diseñado para proporcionar una solución robusta y eficiente para la gestión académica del programa de Ingeniería Industrial en la Universidad de Antioquia, garantizando una planificación y distribución precisa de los estudiantes en sus cursos a lo largo de los semestres.

Esta aplicación deberá generar listados de estudiantes y asignarlos a los cursos correspondientes según las reglas predefinidas del currículo, como el semestre que cursan y las materias que deben tomar. Lo anterior con el fin de facilitar y automatizar la gestión de los cursos académicos y el proceso de matrícula tanto para los estudiantes como para la universidad. Además, el software permitirá la gestión eficiente de la información estudiantil, la asignación de grupos por aula, la generación de calificaciones, reportes y listas de manera automática, almacenándolos en formatos Excel y CSV.

Objetivo general:

Desarrollar un software de gestión académica que permita planificar, organizar y automatizar la distribución de estudiantes en las asignaturas del programa de Ingeniería Industrial de la Universidad de Antioquia, gestionando eficientemente la asignación de grupos, el rendimiento académico y la generación de reportes en formatos Excel y CSV.

Objetivos específicos:

- Implementar un sistema que asigne de forma automática a los 1000 estudiantes en las diferentes asignaturas y grupos según los criterios establecidos.
- Crear un sistema que permita exportar los listados de estudiantes asignados a cada asignatura en archivos Excel y CSV, organizados en carpetas por semestre y asignatura.
- Desarrollar una funcionalidad que genere calificaciones aleatorias para cada estudiante, calcule los promedios por semestre y determine si el estudiante aprueba o no con base en un promedio mínimo de 3.0, asegurando que el 70% de los estudiantes aprueben.
- Diseñar un sistema de codificación único para cada asignatura y grupo, que facilite la identificación y gestión de los diferentes cursos.

- Garantizar que el software sea fácil de usar y que pueda adaptarse a una mayor cantidad de estudiantes, cursos o semestres en el futuro sin comprometer su rendimiento o funcionalidad.

Beneficios:

- **Para la universidad:** Se simplificará la gestión académica, optimizando la asignación de estudiantes en cursos y la administración de calificaciones.
- **Para los docentes:** Podrán contar con listas organizadas y actualizadas por curso y grupo, lo que les permitirá manejar la información de manera más eficiente.
- **Para los estudiantes:** Se beneficiarán de una inscripción y distribución automática y justa en los grupos de clase, lo que reducirá errores y problemas de registro.
- **Automatización en general:** El proceso será más rápido y menos propenso a errores, reduciendo el tiempo implementado.

Especificación de requisitos

Requisitos funcionales:

- **Gestión de Estudiantes:**
 - El sistema debe ser capaz de leer un archivo de datos ("datos.csv") que contenga la información de los 1000 estudiantes o más.
 - Debe asignar automáticamente a los estudiantes a las asignaturas correspondientes de acuerdo con el semestre.
- **Gestión de Asignaturas y Grupos:**
 - El software debe asignar a los estudiantes a grupos sin exceder un límite de cupos por aula según el semestre.
 - Para cada asignatura, se deben crear grupos con un código único y una cantidad específica de estudiantes.
- **Generación de Calificaciones:**
 - El sistema debe generar notas aleatorias para cada asignatura, asegurando que el 70% de los estudiantes aprueben con una calificación promedio igual o mayor a 3.0.
- **Exportación de Datos:**
 - El sistema debe generar listas de los estudiantes asignados a cada grupo, guardando esta información en archivos Excel y CSV.
 - Los archivos generados deben estar organizados en carpetas según el semestre y la asignatura, siguiendo una estructura clara.
- **Códigos de Curso y Asignatura:**
 - Cada asignatura debe tener un código único basado en las reglas descritas.
 - Cada grupo de una asignatura debe recibir un código consecutivo único dentro de la misma asignatura.
- **Cálculo de Promedio por Estudiante:**

- El sistema debe calcular el promedio de notas por estudiante en cada semestre y determinar si aprueban o no el semestre.

Requisitos no funcionales:

- Rendimiento: El sistema debe procesar y asignar a los 1000 o más estudiantes a sus respectivos cursos y grupos en un tiempo razonable, sin afectar la experiencia del usuario.
- Seguridad: Los datos de usuario se almacenan en una base de datos protegida para la seguridad y privacidad del usuario.
- Usabilidad: El software debe ser intuitivo y fácil de usar para usuarios con poca experiencia, con una interfaz clara para cargar archivos, generar listas y acceder a reportes.
- Fiabilidad: El sistema debe ser confiable y evitar fallos durante la asignación de estudiantes y la generación de notas, garantizando que los datos procesados sean correctos.
- Compatibilidad: El software debe ser compatible con archivos en formato CSV y Excel, tanto para la lectura de datos como para la exportación de los resultados.
- Escalabilidad: El sistema debe ser capaz de adaptarse a un mayor número de estudiantes o cursos en el futuro, sin afectar el rendimiento o la funcionalidad.

Plan de proyecto

Cronograma:

https://www.canva.com/design/DAGRuf2PJxU/_VxtYffgnrltd2as62ESjw/view?utm_content=DAGRuf2PJxU&utm_campaign=designshare&utm_medium=link&utm_source=editor

Plan de actividades:

Tarea 1: Reunión para conocer el trabajo, construir propuestas de software y asignar las siguientes tareas.

Tarea 2: Inicio del plan de proyecto: crear cronograma, presupuesto y realizar el documento de visión describiendo las funcionalidades, uso e implementación del sistema.

Tarea 3: Llevar a cabo el plan de actividades y diseñar el flujograma del algoritmo.

Tarea 4: Reunión para revisar la entrega 1, luego realizar la entrega.

Tarea 5: Crear un repositorio en GitHub, subir los archivos del proyecto, e incluir el archivo README.md con la información del equipo y las instrucciones.

Tarea 6: Desarrollar funciones para guardar las listas de estudiantes en archivos Excel y CSV, con el formato y la estructura especificados y programar el algoritmo para la creación de códigos únicos para cada asignatura y grupo, siguiendo las reglas establecidas.

Tarea 7: Implementar el algoritmo que distribuye a los 1000 estudiantes según los grupos y los límites de aula.

Tarea 8: Generar calificaciones y promedios. Implementar la lógica que calcula el promedio de cada estudiante, asegurando que el 70% apruebe.

Tarea 9: Repetir el proceso siguiendo la malla curricular para los alumnos aprobados semestre a semestre.

Tarea 10: Exportar datos a Excel y CSV. Desarrollar las funciones para exportar las listas de cada grupo en formato Excel y CSV, organizadas por carpetas siguiendo la ruta establecida.

Tarea 11: Reunión de revisión general y realización de pruebas iniciales. Ejecutar pruebas para validar el correcto funcionamiento del sistema.

Tarea 12: Crear la sección que genera archivos automáticos del rendimiento de los estudiantes y la asignación de los mismos a las aulas y los grupos, dependiendo del semestre académico.

Tarea 13: Optimizar el rendimiento del sistema.

Tarea 14: Realizar las pruebas finales.

Hacer una prueba completa del sistema con los 1000 estudiantes, asegurándose de que todo funcione correctamente antes de la entrega final.

Presupuesto:

El presupuesto total para llevar a cabo el desarrollo del proyecto se presenta en la tabla a continuación:

Horas de trabajo total	Presupuesto por hora	Presupuesto por integrante de equipo (3)	Presupuesto personal	Presupuesto equipo	Presupuesto total
60	\$10.000	\$600.000	\$1'800.000	\$1'700.000	\$3'500.000

Plan versionado

- **Versión del software 1.0 -> Documentación**

Se definió el nombre y alcance del proyecto, se han detallado los objetivos que buscan la correcta distribución de estudiantes en asignaturas, la generación automática de calificaciones y la exportación de reportes en formatos Excel y CSV. Se documentaron las funcionalidades requeridas del sistema, se estableció un plan de proyecto con las actividades que se deben desarrollar y el tiempo estimado por actividad. Por último, se realizó un presupuesto basado en el tiempo invertido por los integrantes del equipo, equivalente al valor de una práctica profesional.

- **Versión del software 1.1 -> Desarrollo inicial**

En esta versión del software de gestión académica, se completaron varios avances clave. Se creó un repositorio en GitHub con un README.md que incluye información del equipo y las instrucciones de uso. Se desarrolló un algoritmo para la asignación automática de estudiantes en grupos, generando códigos únicos para cada grupo. Además, se implementó un sistema que distribuye a los 1000 estudiantes respetando los límites de cupo por aula y otro para la generación de calificaciones y promedios, asegurando que el 70% de los estudiantes aprueben. El proceso de asignación se repitió para todos los semestres, siguiendo la malla curricular.

- **Versión del software 1.2 -> Entrega del software**

Se desarrollaron las funciones para exportar las listas de estudiantes por grupo en formato Excel y CSV, organizadas en carpetas según la ruta establecida por semestre y asignatura. Además, se creó la sección que genera automáticamente archivos del rendimiento académico de los estudiantes y la asignación a grupos y aulas según el semestre.

Posteriormente, se llevó a cabo una reunión de revisión general, se ejecutaron pruebas iniciales para validar el correcto funcionamiento del sistema y se corrigieron los errores encontrados.

Finalmente, se optimizó el rendimiento del sistema, asegurando que la asignación y generación de informes sea más eficiente; se realizaron las pruebas finales con los 1000 estudiantes del sistema, verificando que todo el flujo funcione correctamente antes de la entrega final.

Diagrama de flujo

https://miro.com/welcomeonboard/RINtUVI6eTBGV0d5N2pwaWIWc0xPVk1TYkdialdSbHVvVzNWanp4MGt1OEpHRkdKNnNLWFBSZDUweXJFajV4NnwzNDU4NzY0NTk3OTkwOTI2Mjk3fDI=?share_link_id=958921661044