

PROYECTO I

Integrantes: Juliana Yepes Valencia, Vincent Castro, Susana Aguirre Franco

El presente documento contiene la información requerida solicitada para la primera entrega del trabajo final de algoritmia y programación 2024 - 1. No es de menos aclarar que el trabajo se llevará a cabo a la par con el proceso de aprendizaje, consideramos que hay muchos aspectos por mejorar.

El objetivo es adquirir el conocimiento y las bases necesarias para seguir llevando a cabo el proyecto y obtener resultados positivos en la materia al final del semestre.

A continuación se encontrará cada parte del proceso para esta primera entrega:

1. Visión

El proyecto lo llamaremos "Proyecto i", este es un software diseñado para automatizar la planeación del programa para el semestre siguiente de la carrera de ingeniería industrial en la universidad de Antioquia.

El objetivo principal es alcanzar la distribución óptima de los estudiantes por semestre y asignatura, teniendo en cuenta el cupo máximo por aula. Esto se logrará mediante la implementación de los algoritmos adecuados que le permitirán al estudiante observar las materias y en el grupo que fue asignado. Los beneficios de este software incluyen la eficiencia en la planeación, la precisión y la facilidad de uso para los administradores del programa. Además, al automatizar este proceso se reduce la posibilidad de encontrar errores "humanos" en el proceso y permitirá ahorrar tiempo para dedicarlo a otras tareas.

2. Especificación de requisitos

Requisitos funcionales:

- El software será capaz de leer los datos de entrada que incluya la lista de estudiantes y la malla curricular. Estos datos pueden estar en diferentes formatos (CSV o Excel), este debe tener la capacidad de extraer la información relevante respecto a estos.

- Una vez que los datos se extraen, el programa tendrá que calcular la distribución de los estudiantes por semestre y asignatura.
- El software debe generar códigos únicos para cada asignatura, estos se utilizarán con el fin de identificar cada asignatura en el sistema.
- Este debe crear grupos de estudiantes para cada asignatura, respetando el cupo máximo que se tiene por aula.
- Finalmente, el programa debe generar un documento con cada registro representando un grupo de una asignatura y los estudiantes que acudirán a esta. Este documento será el resultado y se utilizará para la planeación del semestre.

Requisitos no funcionales:

El software debe ser fácil de usar y entender, esto significa que la interfaz del usuario debe ser intuitiva, mientras la documentación debe ser clara y completa. Debe ser eficiente en el proceso de planeación del semestre, ya que es un proceso que se realiza con frecuencia y cualquier error puede generar un impacto negativo.

3. Plan del proyecto

Actividades:

- Diseño y desarrollo del software: Esta es la primera etapa del proyecto donde se definirán las funcionalidades y se comenzará a escribir el código.
- Pruebas y correcciones: Una vez que el plan está desarrollado se realizarán las pruebas adecuadas para asegurar que funcione correctamente, además se ejecutarán las correcciones pertinentes para mejorar el rendimiento.
- Documentación: Esta es la última etapa del proyecto, donde se registrará: descripción de las funcionalidades, instrucciones de uso y detalles técnicos.

Cronograma:

Este tendrá una duración de 4 semanas que se distribuirán de la siguiente manera:

- Semana 1-2: Diseño y desarrollo del software

- Semana 3: Pruebas y correcciones
- Semana 4: Documentación.

4. Plan versionado

El flujograma que es el próximo punto, presentará la primera versión del plan. Teniendo en cuenta que todavía no contamos con la cantidad correcta de recursos para llevar a cabo la actividad, consideramos que podrán haber varios cambios en el proceso, por eso según la relevancia se enumeran de acuerdo al avance hecho según el cronograma, ejemplo: Nuestra primera versión será la 1.0, de la cual si se hacen cambios en esta semana de entrega (Número 1) pasarán a ser en orden las versiones 1.1, 1.2... de acuerdo con su relevancia, para por último tener una versión 5.0, la cual será el entregable al finalizar el proyecto.

A medida de que avance la ejecución será de mucha ayuda elaborar retroalimentaciones, para que cada idea que debamos descartar sea registrada y cometer el menor número de errores. Cada que haya un avance o cambio, será importante registrarlo junto con esta con sus debidas actualizaciones en este documento.

Para llevar un registro claro decidimos elaborar una tabla con sus elementos principales:

Versión	Retroalimentación	Añadir	Consideraciones
Número de la versión	¿Qué funciona de manera correcta e incorrecta? ¿En qué se ha fallado? ¿Qué hace falta?	Son los elementos nuevos o cambios hechos en el programa.	Dudas que surjan a partir de esta versión, comentarios y posibles avances que se puedan hacer más adelante

Después de haber hecho la primera entrega del trabajo con la respectiva versión piloto comenzará a desarrollarse el proyecto con los pasos anteriormente propuestos.

5. Pseudocódigo

Plan piloto (Versión 1.0):

Función para generar el listado de estudiantes

generar_listado_estudiantes(nombres, apellidos):

Leer archivos "nombres.csv" y "apellidos.csv"

Generar 1000 estudiantes con datos personales y semestre

Repartir estudiantes proporcionalmente por semestres

Función para planear el programa

planear_programa(malla_curricular):

Leer la malla curricular del programa

Para cada semestre:

Calcular el número total de estudiantes por semestre

Para cada asignatura:

Calcular las horas de trabajo docente y horas de trabajo independiente

Calcular el número de grupos necesarios

Generar un código único para la asignatura

Registrar la información en la planeación del programa

Función para crear las listas de cursos

crear_listas_cursos(planeacion_programa):

Para cada semestre:

Para cada grupo de cada asignatura:

Crear un archivo Excel y un archivo CSV con la información del curso y la lista de estudiante

Función para generar el log de eventos

generar_log_eventos(operaciones):

Registrar en un archivo de texto la fecha, hora, instrucción y tiempo de ejecución de cada operación

Incluir información sobre el usuario, sistema operativo y plataforma

Función principal

Generar listado de estudiantes

estudiantes = generar_listado_estudiantes("nombres.csv", "apellidos.csv")

Planear el programa

planeacion_programa = planear_programa(malla_curricular)

Crear listas de cursos

crear_listas_cursos(planeacion_programa)

Generar log de eventos

generar_log_eventos(operaciones)

Ejecutar la función principal