Poredak, Tu Organizador de Horarios

**Juan Diego Plaza, Andrés Medina, Leonardo Pacheco**

**No. de Equipo Trabajo: 10**

# INTRODUCCIÓN

El propósito de este documento es explicar de forma clara el proyecto que se desarrollará durante el semestre para la asignatura Estructuras de Datos. Se especificarán diferentes aspectos del proyecto, desde el problema a resolver hasta la forma en que se espera que el programa lo solucione.

# DESCRIPCIÓN DEL PROBLEMA A RESOLVER

Al momento de organizar horarios, muchos de los estudiantes de la Universidad Nacional de Colombia tienen inconvenientes porque no pueden cuadrar fácilmente sus horarios porque tal vez no conocen de forma muy profunda su malla curricular y pueden terminar teniendo huecos indeseables en sus agendas o viendo materias en un orden que podría no ser coherente. En ocasiones también pasa que un grupo en específico no está ofertado para todas las carreras, generando también confusiones al momento de programar un horario. El programa a desarrollar tiene el propósito de ayudar a este tipo de estudiantes a organizar un horario óptimo y útil según las condiciones que le sean dadas (materias aprobadas, horarios disponibles, carrera respectiva y carga académica deseada).

# usuarios DEL PRODUCTO DE SOFTWARE

La aplicación está dirigida a estudiantes activos de la Universidad Nacional de Colombia Sede Bogotá (aunque se podría extender a otras instituciones educativas cuya metodología de inscripción sea similar) al momento de inscribir materias, es decir, al inicio de cada semestre.

# REQUERIMIENTOS FUNCIONALES DEL SOFTWARE

## **Ingreso Materias a Inscribir**

## El usuario debe ser ingresar un nombre de búsqueda para las materias que desea inscribir, una por una. El sistema mostrará las opciones disponibles.

* *Acciones iniciadoras y comportamiento esperado*:

En la pantalla se mostrará un buscador y el usuario debe ingresar el nombre de la materia que desea inscribir. El buscador establecerá una conexión con la base de datos del SIA y se mostrarán las materias relacionadas, de las cuales se debe seleccionar la adecuada. Este proceso se debe repetir por cada materia que se desee ingresar.

*Requerimientos funcionales*:

* La materia debe estar en la base de datos de la universidad.
* El usuario puede seleccionar alguna de una serie de opciones.
* Una vez seleccionada una materia, esta será almacenada en una cola. Las materias almacenadas en esta cola se verán en pantalla. De esta forma, si se eligió una materia de forma errónea es posible eliminarla de la cola.
* En caso de no ingresar un nombre válido, el sistema lo notificará mediante un mensaje en pantalla.
* El número de materias que se pueden inscribir está limitado dependiendo del número de créditos.
* Una vez seleccionadas todas las materias deseadas, se debe clickear en un botón llamado “Generar Horario”.

## **Generar Horario**

## Es la opción final del usuario. Una vez ingresadas las materias que desea ver solo debe hacer click en esta opción para mostrar el horario óptimo generado por el programa.

* *Acciones iniciadoras y comportamiento esperado*:

En la pantalla se mostrará un horario convencional con las materias ordenadas junto con el nombre de la materia, el profesor, al código de la materia y el grupo respectivo. Si se desea devolverse en el proceso es posible hacerlo mediante un botón “atrás”.

*Requerimientos funcionales*:

* Se deben elegir mínimo la cantidad de créditos permisibles (10). En caso de no ingresar los suficientes, el sistema lo notificará mediante un mensaje en pantalla.
* En caso de que dos o más diferentes materias tengan un grupo único en el mismo horario, se mostrará un mensaje de error explicando que no es posible inscribir las asignaturas al tiempo.
* Para la generación del horario se tendrá en cuenta la cola anteriormente creada. Si se desea un horario diferente puede cambiarse esta cola cuantas veces se quiera.

## **Atrás**

## En cada actividad en desarrollo es posible devolverse a la etapa inicial del programa.

* *Acciones iniciadoras y comportamiento esperado*:

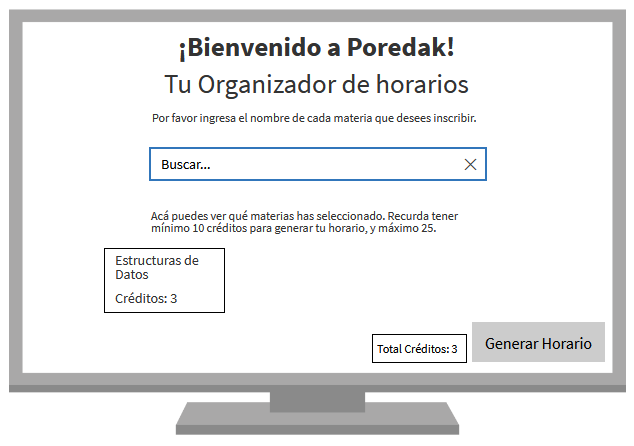
Se espera que el programa este dividido en etapas. Cada vez que se oprima el botón se llevará a la etapa anterior. En el caso de ser la etapa inicial, el botón no se mostrará.

*Requerimientos funcionales*:

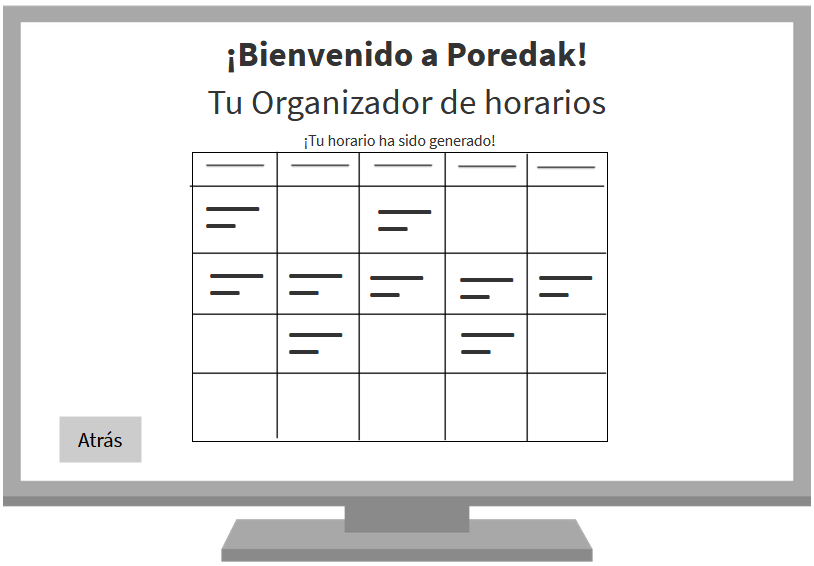
* El botón no se debe mostrar en la etapa inicial porque no habría a dónde devolverse, por lo que solo aparecerá en la funcionalidad de “Generar Horario”.
* El botón llevará al usuario otra vez a la etapa del buscador de materias.

# DESCRIPCIÓN DE LA INTERFAZ DE USUARIO PRELIMINAR

Se espera una interfaz de fácil entendimiento y uso. En la siguiente imagen se muestra como se espera que la etapa inicial del programa se vea.



En esta primer etapa el usuario será capaz de buscar las materias que desea cursar durante el semestre teniendo en cuenta los límites establecidos anteriormente. Después, cuando el usuario de le click a “Generar Horario”, si se tienen las condiciones iniciales, aparecerá el horario de una forma similar a la siguiente:



# Entornos de desarrollo y de operación

A pesar de que la principal parte de programación de desarrollará en Python, la implementación del programa tendrá que ser en html para que cualquier estudiante de la Universidad sea capaz de acceder a él sin problema alguno. Cada dato ingresado se guardará localmente y no afectará la base de datos de la nube. Es debido a esto que no hay un sistema operativo en específico para la operación del programa, sino que se podrá acceder a él desde cualquier dispositivo con navegador. Sin embargo, se enfocará en el desarrollo para computadores.

# PROTOTIPO DE SOFTWARE INICIAL

El software se puede encontrar fácilmente en el repositorio del siguiente link:

<https://github.com/amedinae/ProyectoDS>

Se usaron listas para el almacenamiento de materias a inscribir y generar así un horario pertinente. Siempre se inserta de a un dato y se puede eliminar cuando se necesite. Así mismo se muestra constantemente los datos guardados. Estos de guardan también en archivos tipo JSON dentro del programa.

# PRUEBAS DEL PROTOTIPO

Para probar el prototipo se ingresarán un número n de materias cada una con un número indefinido de cursos e información de profesores y horarios. Para cada número de materias se medirá el tiempo de ejecución:

* 1 materia: 1.6 s
* 2 materias: 2 s
* 3 materias: 1.6 s
* 4 materias: 2.4 s
* 5 materias: 2.6 s

La cantidad de datos generados es igual a la productoria de los diferentes grupos de cada materia diferente, dejando así para cada caso los siguientes datos:

* 4
* 4\*8
* 4\*8\*4
* 4\*8\*4\*10
* 4\*8\*4\*10\*1