

Generador de Horarios

Paula Hernández Góchez B53375
Escuela de Ingeniería Eléctrica
Universidad de Costa Rica, San José, Costa Rica

Resumen—Descripción del desarrollo del proyecto Generador de Horarios para la Universidad de Costa Rica.

DESCRIPCIÓN DEL PROBLEMA

En el proceso de matrícula de la Universidad de Costa Rica los estudiantes deben de realizar su horario, incurriendo frecuentemente en errores al momento de realizarlos. Esto ocasiona que durante todo un semestre se tengan "huecos" en el horario o cursos en el cual su horario choca.

SOLUCIÓN PLANTEADA

Se plantea el desarrollo de un programa que ingrese a e-matrícula a descargar los cursos faltantes y posteriormente, en base a los cursos disponibles y a la carga académica que el estudiante decida llevar, se generan distintas opciones de horario para el estudiante.

DESARROLLO DEL PROYECTO

-A. Ingreso a la plataforma de e-matrícula

Primeramente se buscó las herramientas a disposición en C++ para el ingreso a una página web y el parseo de html. Entre dichas soluciones se encontró proyectos que utilizaban *Selenium WebServer*, sin embargo se encontraban con paquetes en conflicto o sin seguimiento desde hace 3 años.

Es por esto que se decide utilizar python y así mismo continuar con un proyecto iniciado a finales del semestre pasado. Este proyecto utiliza *SeleniumWebDriver* el cual realiza llamadas directas al navegador utilizando el soporte nativo de cada navegador para la automatización. La forma en que se realizan estas llamadas directas y las características que admiten depende del navegador que está utilizando. [1]

Al ser un proyecto que ya había sido iniciado, se contaba con un programa inicial que ingresaba a la plataforma, por lo que para este proyecto se continuó con eso agregando navegación por la plataforma a cursos pendientes del plan y una vez realizado esto se generaba un archivo .txt con los cursos faltantes del plan.

En C++ este llamado de ingreso a la plataforma se hace con la siguiente función, la cual el sistema hace un llamado a python para generar el archivo necesario.

```
system("python3 ./bin/web_main.py USUARIO  
CONTRASEÑA")
```

-B. Obtención de horario

Se utiliza la herramienta de linux *pdftotext* en la cual se utilizan los flags *-layout* para obtener el formato justo como se encuentra en el horario y poder tenerlo en un archivo de texto plano sobre el cual pueda operarse desde c++.

```
pdftotext -layout -f 170 -l 180 guiaCursos.pdf cursos
```

-C. Estructuras definidas en C++

Se definió una estructura en la cual se guarda cada materia con su sigla, cantidad de créditos y nombre, como puede verse en el siguiente código.

```
struct subject  
{  
    string sigla;  
    string nombre;  
    int créditos;  
};
```

Y una en la cual se guardaran los elementos para generar horarios

```
struct timeTable  
{  
    string sigla;  
    string curso;  
    string grupo;  
    string horario;  
    timeTable *ptrOption= NULL; // if there's 2 or more  
    possible courses, points to the next one, if not to  
    NULL ;  
};
```

-D. Funciones definidas en C++

Las funciones definidas se ven en el siguiente cuadro:

```
vectorsubject * > trimSubjects(string txtName)  
vectortimeTable * > trimFile(string txtName,  
vectorstring > targetSub)  
vectortimeTable >  
timeTableGenerator(vectortimeTable * > schedules )
```

En la cual *trimSubjects*, recibe por parámetro el archivo .txt con el cual se obtienen los cursos restantes del plan y se almacena en la estructura antes definida para las materias.

La función *trimFile*, recibe por parámetro el archivo que contiene los horarios publicados en la página de la escuela de

ingeniería eléctrica de la oferta de cursos del semestre y la lista de cursos para la cual se debe hacer el horario y devuelve un vector de estructuras timeTable que son los horarios de los cursos deseados.

REFERENCIAS

- [1] Selenium WebDriver . Obtenido de :
http://www.seleniumhq.org/docs/03_webdriver.jsp#introducingwebdriver