

**UNIVERSIDAD SIMÓN BOLÍVAR**

**Decanato de Estudios Profesionales**

**Departamento de Computación y Tecnología de la Información**

**CI-3715 Ingeniería del Software I**

**Septiembre-Diciembre 2017**

Informe Técnico:

Avanzómetro Sprint 3

Profesor: Alfonso Reinoza

**Integrantes:**

Miguel Canedo 13-10214

Rafael Cisneros 13-11156

Carlos Pérez 13-11089

Ritces Parra 12-11088

Yezabel Rincón 10-11005

Erick Flejan 12-11555

Andrés Buelvas 13-10184

José Bracuto 13-10173

Sartenejas, 01 de diciembre de 2017

Índice general

1. Introducción 3
2. Historias de Usuario y análisis de dominio 4

2.1. Animación varias cohortes 4

2.2. Granularidad de las gráficas 4

2.3. Gráfico de curvas 5

3. Documentación del software 6

3.1. Diseño 6

3.2. Interfaz 7

3.3. Clases

3.4. Métodos

3.5. Modelos

3.6. Vistas

4. Trabajo en equipo

5. Experiencia TDD

6. Manejo del repositorio de Github

7. Conclusiones y recomendaciones

2

Capítulo 1

Introducción.

En este documento se presentarán las historias de usuario del sprint 3 del Avanzómetro. El objetivo de las historias de usuario es la realización de una interfaz que permite mostrar gráficos dinámicos e interactivos sobre los créditos aprobados de los estudiantes de varias cohortes y cambiar la granularidad de dichas graficas

En el capítulo 2 de este informe hablaremos sobre las historias de usuario, que representan las funcionalidades de la aplicación y otras actividades necesarias para su desarrollo y para cada historia de usuario se explican el análisis de su dominio. En el capítulo 3 se especifica la documentación del software, su diseño, interfaz, sus clases, métodos, modelos y vistas. En el capítulo 4 se explicará la división del trabajo y el esfuerzo realizado por el equipo en la realización de cada actividad. El capítulo 5 presenta el desarrollo de las pruebas TDD realizadas para la aplicación y la experiencia del equipo en este aspecto, finalmente el capítulo 6 explica el manejo del repositorio Github de nuestra organización y como nos ayudó en el desarrollo del software. Finalmente en el capítulo 7 se presenta las conclusiones y recomendaciones para el proyecto.

3

Capítulo 2

Historias de usuario y análisis de dominio.

El prototipo del Avanzómetro cuenta con 13 historias de usuario para ser desarrolladas en 3 sprints. El tercer sprint consta de 3 historias referentes principalmente a la comparación de varias cohortes en la misma animación, y el cambio de la granularidad de las gráficas.

A continuación se explican las 3 historias de usuario del sprint 3 y sus respectivos análisis de dominio.

2.1. Animación varias cohortes

Se requiere visualizar la animación del flujo de estudiantes de dos o más cohortes. Para el desarrollo de esta historia de usuarios se tienen los siguientes criterios de aceptación:

a. Las cohortes deben tener colores bien diferenciados.

b. Las cohortes deben compartir el espacio ocupado por las coordenadas de los intervalos.

El dominio de esta historia de usuario es la cantidad de cohortes que se pueden mostrar en una misma gráfica. Por razones estéticas hemos colocado de 1 a 5 cohortes en una misma gráfica.

2.2. Granularidad de las graficas

Se requiere cambiar la granularidad de los diagramas de barra. Para esta historia de usuario se tienen los siguientes criterios de aceptación:

4

a. Se debe poder seleccionar la granularidad y las barras deben estar en concordancia con la base de datos.

El dominio de esta historia de usuario es el valor a seleccionar para la granularidad de las barras. Por razones estéticas, se acordó colocar la granularidad con un mínimo de 5 créditos. El máximo valor para la granularidad es de 240, ya que este es el número de créditos total en la mayoría de las carreras de la universidad.

2.3. Gráfico de curvas

Se requiere la opción de ver el flujo de estudiantes de una o más cohortes en un gráfico de curvas.

a. Las cohortes deben tener colores bien diferenciados.

b. Las cohortes deben compartir el espacio ocupado por las coordenadas de los intervalos.

El dominio de esta historia de usuario es la granularidad de los créditos para las barras, que va de 5 a 240; el % en la vertical de la gráfica, que va de% 0 a 100%; la cantidad de milisegundos que dura una instantánea para un trimestre especifico, que va de 500 a 3000 milisegundos; y la cantidad de cohortes graficadas, que son de 1 a 5 cohortes.

5

Capítulo 3

Documentación del software

3.1. Diseño

Para la realización del Avanzómetro se utilizaron las siguientes herramientas:

- Django como framework para desarrollo web

- Python como lenguaje de programación

- Postgres como manejador de base de datos con Psycopg2 para compatibilidad con python

- Html como lenguaje de desarrollo web

- Javascript para realizar animaciones y validaciones de las páginas web

- D3 y Dimple para realizar las gráficas mostradas en el software

- Selenium y geckoDriver para la realización de las pruebas en el navegador Mozilla Firefox

Para poder hacer uso del Avanzómetro es necesario tener todas estas herramientas instaladas en el ordenador, a excepción de Selenium y geckodriver.

Para instalar estas librerías se usan los siguientes comandos (se debe tener instalado pip de python):

python -m pip install django --upgrade

python -m pip install psycopg2 --upgrade

python -m pip install selenium –upgrade

La instalación de geckodriver se encuentra en el siguiente link:

<https://www.obeythetestinggoat.com/book/pre-requisite-installations.html#firefox_gecko>

Como estructura de datos se usó Json para tomar los arreglos del controlador de Django y pasarlo al html.

6

3.2. Interfaces

Al iniciar el avanzómetro se muestra la interfaz de login, aquí se facilitan los campos (inputs) para el nombre de usuario (correo) y la contraseña. Además se tiene la opción de registro, que al seleccionarla te redirecciona a la página del registro que cuenta con un formulario de datos para introducir nombres, apellidos, correo y contraseña con los cuales se ingresaran posteriormente al sistema.

Una vez conectado, se muestra la interfaz para generar instantánea, en esta página se observa en primera instancia una gráfica con una distribución normal de datos que sirve como muestra de las gráficas que genera el sistema. En la parte superior de todas las páginas que visualiza un usuario conectado se encuentra el navbar de color azul rey, en él se muestran las diferentes opciones del sistema: registrar nuevos usuarios, desconectarse, cargar archivo de estudiantes, generar instantáneas y generar animaciones. La interfaz para registrar usuarios es la misma usada en la redirección que aparece en la página de login. La página de cargar archivo cuenta con un botón de selección, al darle click se le solicita al usuario que busque dentro de su directorio el archivo que desea cargar, el archivo debe seguir un formato estricto para que pueda ser aceptado; una vez cargado un archivo, se indica si la carga fue exitosa o si se produjo un error. La interfaz para generar gráficas cuenta con los campos cohorte, trimestre, y carrera en forma de dropdown con los datos que se encuentran en la base de datos, además se tiene el campo año que se usa para relación el trimestre con un año específico, también se tiene el botón “graficar” que al hacerle click toma los datos de los inputs, realiza la búsqueda en la base de datos y muestra en pantalla la gráfica correspondiente. La interfaz para generar animaciones cuenta con los campos cohorte y carrera para los datos de la gráfica, y el campo milisegundos que es la duración del tiempo de muestra de una imagen en pantalla, y un el botón “animar”, una vez se hace click sobre el botón.

3.3. Clases

Para el tercer sprint no se agregó ninguna clase extra, se mantuvo la única clase usada: RegistroUsuario, esta clase es utilizada por Django para poder crear la vista correspondiente al registro de usuarios.

El resto de la aplicación se realizó en métodos, por lo que no fue necesaria la creación de más clases.

3.4. Métodos

El desarrollo del software se dividió en diferentes aplicaciones de Django. Cada aplicación cuenta con sus propios métodos. A continuación se explican los métodos de cada aplicación.

Para la aplicación de login se usan los siguientes métodos:

index: este método direcciona a la página principal del avanzómetro.

logout: es el método utilizado para desconectar a un usuario del sistema, y lo redirecciona a la página de login del avanzómetro.

Para la aplicación instantánea se usan los siguientes métodos:

getcreditsbytrandct: este método consigue en la base de datos los créditos de cada estudiante de una cohorte, según el trimestre señalado, para la generación de instantáneas.

Instantánea: genera una instantánea básica cuando no tiene especificados datos de entrada, y una instantánea correspondiente a una cohorte según los datos dados.

A continuación se explican los métodos usados por la aplicación de carga:

comprobar\_entero: verifica que no se introduzcan notas negativas ni elementos distintos a un entero positivo.

separar\_estudiantes: lee el archivo csv e introduce en la base de datos los estudiantes indicados en el archivo con sus respectivos datos usando el método read\_csv. El archivo csv usado por este método es una simulación del expediente de DACE.

separar\_estudiantes\_V2: similar al método anterior, lee un archivo csv e introduce en la base de datos los estudiantes con sus respectivos datos usando el método read\_csv. En este método el archivo csv contiene datos más sencillos para realizar pruebas sobre las gráficas y animaciones.

read\_csv: esta función es la encargada de introducir los datos en de los archivos csv en la base de datos.

CargarArchivo: este es el método usado para Django para invocar la vista creada.

A continuación se explican los métodos usados por la aplicación animación:

obtenerMatriz: este método consigue en la base de datos los créditos de cada estudiante de una cohorte, para todos los trimestre cursados por dichos estudiantes, para la generación de las instantáneas usadas en las animaciones.

animación: genera una animación básica cuando no tiene especificados datos de entrada, y una instantánea correspondiente a una cohorte según los datos dados.

Se realizó en el tercer sprint una nueva aplicación llamada “multiGraph”, esta aplicación cuenta con las siguientes funciones:

3.5. Modelos

La base de datos usada cuenta con 4 tablas para los datos usados en las gráficas: estudiantes, trimestre, cohorte y cursa que es una tabla para relacionar un estudiante con un trimestre y los créditos aprobados por ese estudiante en ese trimestre; para el login y registro de usuarios se usan tablas que provee Django donde se implementan todas las medidas de seguridad referentes a los usuarios.

3.6. Vistas

Capítulo 4

Trabajo en equipo

Para el desarrollo del proyecto se identificó las diferentes necesidades por cada historia de usuario. Se logra identificar las siguientes actividades:

Capítulo 5

Experiencia TDD

Aún el equipo no se ha acostumbrado a trabajar bajo la modalidad de TDD, por lo que comenzamos a realizar el software sin las pruebas, pero en medio del desarrollo necesitábamos saber cuál sería el resultado final del sprint, por tanto se hizo diferentes pruebas sobre algunas de las historias para poder tener una mejor idea de cómo se culminan.

Capítulo 6

Manejo del repositorio Github

Para el desarrollo de este sprint se creó una nueva rama llamada “desarrollo3”, al final del desarrollo del sprint se hizo merge con la rama “master”. Se turnaron los diferentes commits que se hicieron en la rama desarrollo. No se usaron más ramas ya que se realizó una historia de usuario a la vez y hasta que no se terminó no se procedió con la siguiente historia.

Capítulo 7

Conclusiones y

Con este tercer sprint se completó el primer prototipo funcional para presentar al cliente y decidir si se prosigue con el desarrollo del Avanzómetro para uso de la Universidad Simón Bolívar.

Se recomienda continuar con el desarrollo del Avanzómetro ya que puede ser de una gran utilidad para mejor el desempeño de los estudiantes de la universidad