
TRABAJO PRÁCTICO INTEGRADOR

SOPORTE A LA GESTIÓN DE DATOS CON PROGRAMACIÓN VISUAL

Lisandro Ferroni
Legajo 44161 - Comisión 401
lisandro.ferroni@gmail.com

Martín Balletero
Legajo 45463 - Comisión 401
martin.balletero@hotmail.com

May 14, 2020

ABSTRACT

En el siguiente informe se detallará el trabajo realizado usando de guía la consigna del trabajo práctico número 8. La selección del sistema para desarrollar consiste en realizar una sucesión de informes y gráficos en base al sistema de transporte público de la ciudad de Rosario recabando información del mismo y simulando la utilización que tienen los pasajeros. La tecnología seleccionada es GraphQL.

Contents

1	Introducción	3
2	Información	3
3	Reportes	3

1 Introducción

El objetivo seleccionado para el trabajo práctico integrador establecido por la Cátedra de Soporte a la Gestión de Datos con Programación Visual es el desarrollo de un sistema que permite al usuario obtener reportes y gráficos acerca del funcionamiento del transporte urbano de pasajeros de la ciudad de Rosario.

El lenguaje de programación para construirlo es Python 3.x, respetando el modelo de capas. Se utilizará una base de datos diseñada en MySQL, mediante el ORM llamado SQLAlchemy. Se utilizará la librería GraphQL de Python llamada Graphene. Para el servidor web se seleccionó Flask como framework de aplicación web.

2 Información

Parte de la información para el desarrollo del sistema es obtenida desde el sitio web del sistema "¿Cuándo llega?" de la Municipalidad de Rosario. A partir del mismo se pueden identificar todas las líneas de colectivo que se encuentran activas en la ciudad, junto con sus recorridos, paradas identificadas por intersecciones de calles y horarios que deben cumplir las unidades.

Toda esta información es recopilada desde el código HTML que se inspecciona cuando se visita el sitio del ETR (Ente de la Movilidad de Rosario). Se monitorizan las peticiones HTTP que realiza el mismo al interactuar con los selectores del formulario y se comprende la estructura de la información disponible públicamente. Esta se almacenará la base de datos del sistema, previamente diseñada con tablas relacionadas que harán posible las diferentes consultas para generar reportes que se detallarán más adelante.

Para completar la información necesaria para cumplir con los objetivos del sistema a desarrollar, es necesario conocer la cantidad de pasajeros que ascienden en cada parada a cada unidad, y en qué momento. Esto nos permitirá estimar la densidad de personas en los colectivos, y a partir de allí continuar con el análisis establecido por el trabajo. Dichos datos pueden ser obtenidos mediante una fuente interna de la municipalidad, ya que han sido medidos en el pasado para determinadas unidades la cantidad de boletos que se marcan a lo largo del recorrido. En caso de no contar con esta fuente, se procederá a simular los datos.

3 Reportes

Se confeccionarán reportes en forma de gráficos con el objetivo de visualizar diferentes características sobre el funcionamiento del sistema de transporte, y a partir de allí analizar posibles medidas para la optimización del servicio. Se le permitirá al usuario ingresar los parámetros mediante los cuales se crearán las consultas a la base de datos mediante GraphQL, tales como el área geográfica para conocer la densidad de pasajeros por barrio, franja horaria, días de la semana, líneas de colectivo, etc. Los mismos podrán ser utilizados para auditar el servicio, destacar qué zonas de la ciudad tienen mayor flujo de pasajeros, detectar cuándo la demanda excede la oferta, estimar el horario en el que pasa un colectivo por una parada sin servicio de GPS, entre otros.

References

- [1] Cameron Moss. *GraphQL server up and running with 50 lines of python*. Para Medium.com, Agosto 25, 2017.
- [2] Jonatas Baldin. *graphql-python Tutorial - Introduction* Para Howtographql.com.