

Entrega de avances 21/06

Integrantes del proyecto

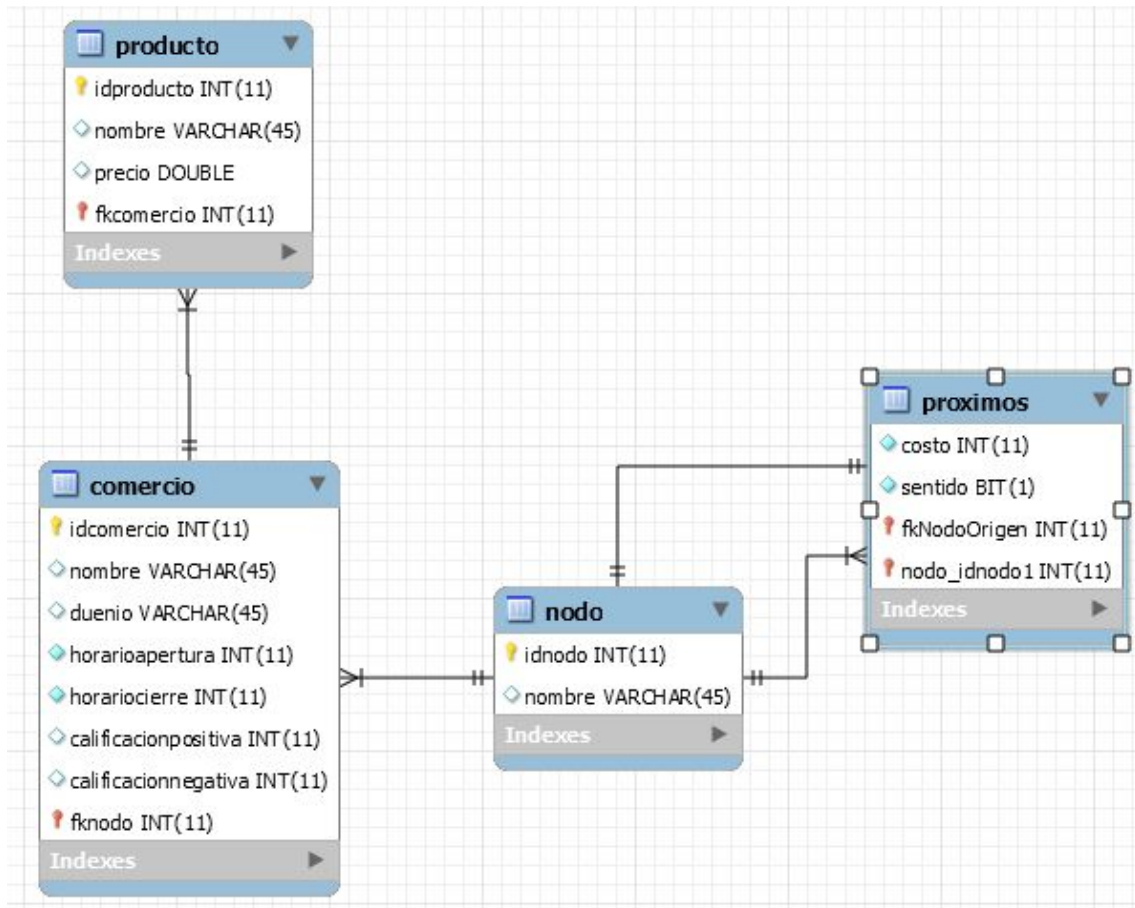
- **Estefanía Lucía Betelú** - Diagrama de clases, desarrollo del código
- **John David Molina** - Desarrollo de código, DER, base de datos
- **Juan José Martínez** - Base datos, DER, diagrama de clases
- **Lautaro Lopez** - Diseño visual, desarrollo de código, tester
- **Lucas Martinez** - Diseño visual, desarrollo de código
- **Luciana Andrea Echevarría** - Base de datos, DER, diagrama de clases
- **Luciano Quiroga** - Tester, desarrollo de código.
- **Mariela Toscano** - Codificación, Diagrama de clases
- **Marina Binando** - Desarrollo de código, DER, diagrama de clases, tester
- **Mario J. Rupertus** - Desarrollo de código, DER, presentación
- **Maximiliano Jarque** - Desarrollo de código, DER, diagrama de clases, presentación
- **Yanina Rojo** - Diagrama de clases, desarrollo, tester

Modificaciones respecto a la segunda entrega

Base de datos:

- **Se elimina la tabla producto_has_comercio**; en primera instancia pensamos que la relación de producto y comercio era de muchos a muchos y creamos la tabla producto_has_comercio para relacionarlas. Pero en realidad cada producto pertenece a un solo comercio. Por lo tanto se elimina la tabla auxiliar. Un comercio puede tener varios productos pero un producto tiene un solo comercio. Producto contiene FK de comercio.
- **En clase Comercio, se reemplaza el atributo Horario tipo Date por dos atributos tipo int horarioApertura y horarioCierre**: Para simplificar filtro de comercios abiertos. No se tienen en cuenta minutos, sólo horas.
- **Tabla próximo, se elimina campo idproximo**: En virtud de que cada Proximo tiene un nodo de origen y un nodo de destino y para simplificar la cantidad de datos guardados, se elimina la id automática y en su lugar se utiliza la PK doble conformada por las FK nodoOrigen y nodoDestino. Así mismo, se cambió el nombre de la clase Ubicación por Nodo a fin de unificar criterio con la clase respectiva.

DER Actual



Estado de avance

- **Código:** Están implementadas todas las clases del diagrama. Falta desarrollar la clase Servicios y a Dijkstra agregar comportamiento para filtrar recorrido si lo hacemos en automóvil. Además de métodos necesarios que requiera la interfaz gráfica.
 - **Git:** Se creó y compartió repositorio para facilitar trabajo en equipo.
- **Base de datos:** Se crearon tablas y relaciones de acuerdo a DER. Se alojó DB en servidor disponible 24x7 y conexión a internet.
 - **Maven:** En proceso
 - **Carga de datos:** En proceso
- **Visual:** Se implementaron en Swing las pantallas principales. Pendiente implementación y conexión de botones y funciones con las clases relacionadas.
- **Testing:** Se ha testado correcto funcionamiento del algoritmo de búsqueda de recorrido.