PATRÓN DE ARQUITECTUA MVC DE FLASK - ELEGANZA ELEGANZA sigue el patrón de arquitectura Modelo-Vista-Modelo Controlador (MVC) implementado con Flask y Marshmallow. Data Transfer Domain Models Esta arquitectura permite una clara separación de responsabilidades, manteniendo la lógica de negocio UserSchema Usuario (Controllers) separada del manejo de datos (Models) y la (schemas_usuario.py) (modelo usuario.py) presentación (Templates), facilitando el mantenimiento y la escalabilidad del sistema. ProductSchema Producto (schemas_producto.py) (modelo producto.py) Autores: Petición / Controlador Arcia Osorio Jairo Andrés Respuesta Daniela Isabel Villa Bastidas Jorge Junior Solano Romero OrderSchema Pedido Sara Margarita Sierra Oliveros Controllers (schemas_pedido.py) (modelo_pedido.py) UserController user_controller.py Petición / Petición / Respuesta Respuesta ProductController Data access product_controller.py Database Config (config.py) OrderController order_controller.py Run _init__.py ServerApp Flask App + Marshmallow (run.py) Routes UserRoutes **Vista** user_routes.py Templates User Views ProductRoutes (login.html, profile.html) product_routes.py Petición HTTP (GET. **Product Views** POST, PUT, DELETÉ). OrderRoutes (catalog.html, detail.html) order_routes.py Petición / Respuesta Order Views (cart.html, checkout.html) Cliente Static Files **CSS Styles** JS Scripts Images