**About Us:**

At MyStreeT , we're passionate about sneakers and dedicated to redefining your shopping experience. Our platform offers a curated selection of the latest and iconic sneakers, along with accurate sizing guidance. Join us as we innovate, delivering a seamless sneaker shopping adventure.

**Problem Statement:**  
  
The Sneaker Shopper Portal project aims to address the current challenges faced by sneaker enthusiasts when shopping for their favorite footwear online. These challenges include limited access to comprehensive information, difficulty in finding the right fit, a lack of payment flexibility, and a dearth of subscription alerts.   
  
The goal is to create a user-friendly platform that enhances the sneaker shopping experience, ensuring customers can easily discover, evaluate, and purchase sneakers that suit their style, preferences, and needs.   
  
The project seeks to incorporate accurate sizing guidance, a user-friendly interface, multiple payment modes, and subscription alerts to address these issues and create a more enjoyable sneaker shopping experience. Users will receive notifications when they subscribe to alerts for their favorite sneakers, enhancing their overall shopping experience.  
  
**Roles:**  
1. Customers who want to buy sneakers.  
2. Admin who can add/remove products to catalog.  
  
  
  
Green represents MLP scope

**Key Features:**

* **User Registration and Authentication:**
  + Customers can create accounts, sign in, and reset passwords.
  + Admins have full access to manage user accounts.
* **Browsing and Searching:**
  + Customers can browse sneaker categories, brands, styles, and apply filters.
  + Users can search for specific sneaker types, sizes, or brands.
* **Product Catalog:**
  + Display a variety of sneaker products, each with images, descriptions, prices, and available sizes.
  + Customers can add/or and view product ratings and reviews.
* **Product Details:**
  + Customers can view detailed product information, including materials, color options, and customer reviews.
  + Admins can edit or update product information.
* **Shopping Cart:**
  + Customers can add and remove items from their shopping carts.
  + Calculate the total cost of selected items.
* **Checkout and Payment:**
  + Enable customers to provide shipping information.
  + Support various payment methods, such as credit card, PayPal, or other options.
* **Order Management:**
  + Customers can track the status of their orders.
  + Admins can view and manage orders, update order statuses, and communicate with customers.
* **User Reviews and Ratings:**
  + Customers can leave reviews and ratings for products they've purchased.
  + Admins can moderate and manage reviews.
* **Inventory Management:**
  + Automatically update product quantities when an order is placed.
  + Send notifications to admins when stock is low.
* **User Notifications:**
  + Notify customers about ‘new shoes alert’, optional: [order confirmation, shipment, and delivery updates.]
  + Send promotional notifications or newsletters to registered customers.
* **Returns and Refunds:**
  + Provide a process for customers to initiate returns or request refunds.
  + Admins can manage return requests and process refunds.
* **Admin Dashboard:**
  + Admins have access to a dashboard to manage products, user accounts, ratings, reviews and orders.
  + Generate reports on sales, revenue, and customer engagement.
* **Customer Support:**
  + Offer customer support through chat, email, or a ticketing system.
  + Assist customers with inquiries, complaints, and issues.
* **Security and Privacy:**
  + Ensure secure handling of customer data, including payment information.
  + Protect customer privacy and comply with data protection regulations.

**Flow of Events:**

* Customers access the web application and either sign in or browse as guests.
* They search or navigate through the product catalog to find desired sneakers.
* Customers add items to their shopping carts and proceed to checkout.
* At checkout, they provide shipping details and choose a payment method.
* After successful payment, the order is confirmed, and customers receive order confirmation notifications.
* The admin manages the order, updates its status, and oversees inventory levels.
* Customers receive notifications about the status of their orders, from shipment to delivery.
* Customers can leave reviews and ratings for the purchased products.
* If needed, customers can request returns or refunds, which are managed by the admin.

**SLA and NFRs:**

* Design documentation
* Microservice response time < 200 ms
* Secure access over HTTPS/TLS (Encryption over wire)
* Secure data store for PII information (Encryption at rest)
* Page view times < 3s
* Throughput - Req/Sec – 100 req/s
* Availability - 99.99
* Concurrent user session – 3000
* Responsive web design - supported view port (Mobile and Tablet and Desktop)
* CI&CD Pipelines to be deployed via Jenkins | Docker.
* Release wise pipeline configuration for hot fixes and feature releases
* Proper Error page for 4xx – 5xx
* Test Data preparation, management and data cleanup should be there
* Continuous failures shall be communicated at priority via alerts to the support teams to be acted upon
* Proper error handling should be there for microservice.
* Logging and Monitoring services
* Alerting

**Engineering PoV:**

* 100% method coverage
* 95% block/line coverage
* Code Quality - 0 Blocker, Critical & Major issues reported by Sonarqube, ESLint
* Vulnerability scanning for code and dependencies.
* Code Complexity
* Memory profiling
* Unit Test Reports
* Release cadence
* Feature Release
* Subsequent bug-fix release
* Subsequent Prod Release
* Subsequent patch release (if any)

**Testing**

* Unit Testing & Regression
* Unit Test coverage (Front End and Microservices) > 95%