**REPUBLIC OF CAMEROON REPUBLIQUE DU CAMEROUN**

**Peace – Work – Fatherland**   **Paix – Travail – Paitrie**



**FACULTY OF ENGINEERING AND TECHNOLOGY**

**DEPARTMENT OF COMPUTER ENGINEERING**

**SPECIALIZATION: SOFTWARE ENGINEERING**

**COURSE TITLE: DATA WAREHOUSING**

HYBRID 3-TIER ARCHITECTURE MODEL FOR THE FLAVORIDE WEBAPP

|  |  |  |
| --- | --- | --- |
|  | **NAMES** | **MATRICULE NUMBERS** |
| **1** | **QUINUEL TABOT NDIP-AGBOR** | **FE21A300** |
| **2** | **NGUEDIA JEATSA JOYCE GRACE** | **FE21A263** |
| **3** | **SIRRI THERESIA ANYE** | **FE21A306** |
| **4** | **NANGLEFACK LEODIA FIETSOP** | **FE21A244** |

**COURSE INSTRUCTOR: DR. SOP LIONEL**

**ABSTRACT**

This document presents a comprehensive analysis and design for a hybrid 3-tier architecture to support the Flavoride cosmetic web application. Leveraging the strengths of both cloud-based services and on-premises infrastructure, this architecture aims to deliver scalability, cost-efficiency, security, and performance for the application.

**INTRODUCTION**

The Flavoride cosmetic web application requires a robust and scalable architecture to support its current and future needs. This architecture must be able to handle increasing user traffic, ensure data integrity, and provide a seamless user experience. A hybrid 3-tier architecture, combining cloud-based services and on-premises infrastructure, is proposed to meet these requirements

2. Architecture Overview:

This hybrid 3-tier architecture comprises three distinct layers:

2.1. Tier 1: Presentation Layer:

* Technologies: ReactJS (frontend framework), Next.js (server-side rendering), Bootstrap (CSS framework)
* Responsibilities: Handles user interface and interaction, including product information, shopping cart, checkout, and user account management.
* Benefits:
  + Responsive design for optimal viewing across devices
  + Fast loading times for improved user experience
  + SEO optimization for increased visibility in search results
  + Single page application (SPA) functionality for seamless navigation
* Deployment: AWS Amplify or Vercel for global reach and auto-scaling.

2.2. Tier 2: Business Logic Layer:

* Technologies: Python (backend language), Django (web framework), FastAPI (REST API framework)
* Responsibilities:
  + Business logic processing, including product management, order processing, authentication, and user management
  + Secure and scalable data handling
  + REST API for mobile app development
* Benefits:
  + Highly scalable and secure
  + Modular design for easy maintenance and extensibility
  + REST API enables mobile app integration
* Deployment: Docker containers hosted on AWS Elastic Beanstalk or Google Kubernetes Engine for auto-scaling and resource optimization.

2.3. Tier 3: Data Layer:

* Technologies: PostgreSQL (relational database), Redis (key-value store)
* Responsibilities: Stores and manages application data, including product information, user data, orders, and other relevant information.
* Benefits:
  + ACID compliance ensures data integrity and consistency
  + PostgreSQL offers high performance for read/write operations
  + Redis provides caching for frequently accessed data, improving application responsiveness
* Deployment:
  + PostgreSQL hosted on AWS RDS or Google Cloud SQL for scalability and security
  + Redis hosted on AWS ElastiCache or Azure Cache for Redis for high availability and performance

3. Hybrid Architecture Advantages:

This hybrid architecture offers several advantages for the Flavoride web application:

* Scalability: Cloud-based services can dynamically scale resources up or down based on actual demand, ensuring smooth performance even during peak traffic periods.
* Cost-efficiency: Utilizing pay-as-you-go models for cloud resources eliminates unnecessary upfront investments and optimizes operational costs.
* Security: Cloud providers offer robust security features and compliance certifications, protecting sensitive data and application assets.
* Flexibility: The hybrid approach allows for customization and integration with existing systems, enabling future growth and adaptation.
* Data Protection: Sensitive data remains securely stored and processed on-premises, providing additional control and compliance.

4. Additional Considerations:

* API Gateway: Implementing an API Gateway like AWS API Gateway or Azure API Management centralizes API management and security.
* Load Balancing: Distributing traffic across multiple servers using a load balancer improves application performance and reliability.
* Monitoring and Logging: Comprehensive monitoring and logging systems identify potential issues and ensure optimal application performance.
* Security: Implementing robust security measures, including encryption, access controls, and vulnerability scans, is crucial for data protection.

5. Conclusion:

This hybrid 3-tier architecture provides a comprehensive and well-defined framework for the Flavoride cosmetic web application. By combining the scalability, cost-efficiency, and security of cloud-based services with the control and data protection of on-premises infrastructure, this architecture ensures a robust foundation for the application to thrive and evolve in the digital marketplace.

6. Next Steps:

* Conduct a detailed cost analysis and resource estimation for the proposed architecture.
* Develop a comprehensive deployment plan outlining the implementation process for each tier.
* Establish security protocols and procedures to ensure data and application protection.
* Implement monitoring and logging systems to track application performance and identify potential issues.
* Regularly review and update the architecture to keep pace with changing requirements and technologies.

This in-depth analysis and design provide a solid foundation for confidently constructing and deploying the Flavoride web application. By adhering to best practices and continuously optimizing the infrastructure,