PICK MY DISH

Your Personal Cooking Companion

- Time-aware
- Mood-based
- Ingredient-aware

Intelligent Recipe Recommendation System

Team Members:

Kamdeu Yamdjeuson Neil Marshall (Backend & DevOps)
Tuheu Tchoubi Pempeme Moussa Fahdil (Frontend & UI/UX)

⚠ THE CHALLENGE

💡 OUR SOLUTION

THE CHALLENGE		OUR SOLUTION
Decision Fatigue	→	AI Recommendations
Limited Resources	→	Smart Filtering
Time Constraints	→	Quick Planning
No Personalization	→	Mood-Aware

Intelligent Features for Seamless Cooking

Mood Filtering

Smart Ingredient Matching

Time-Aware Planning

Recipe Sharing

Admin Dashboard

Personal Favorites

Offline Access

User Profiles
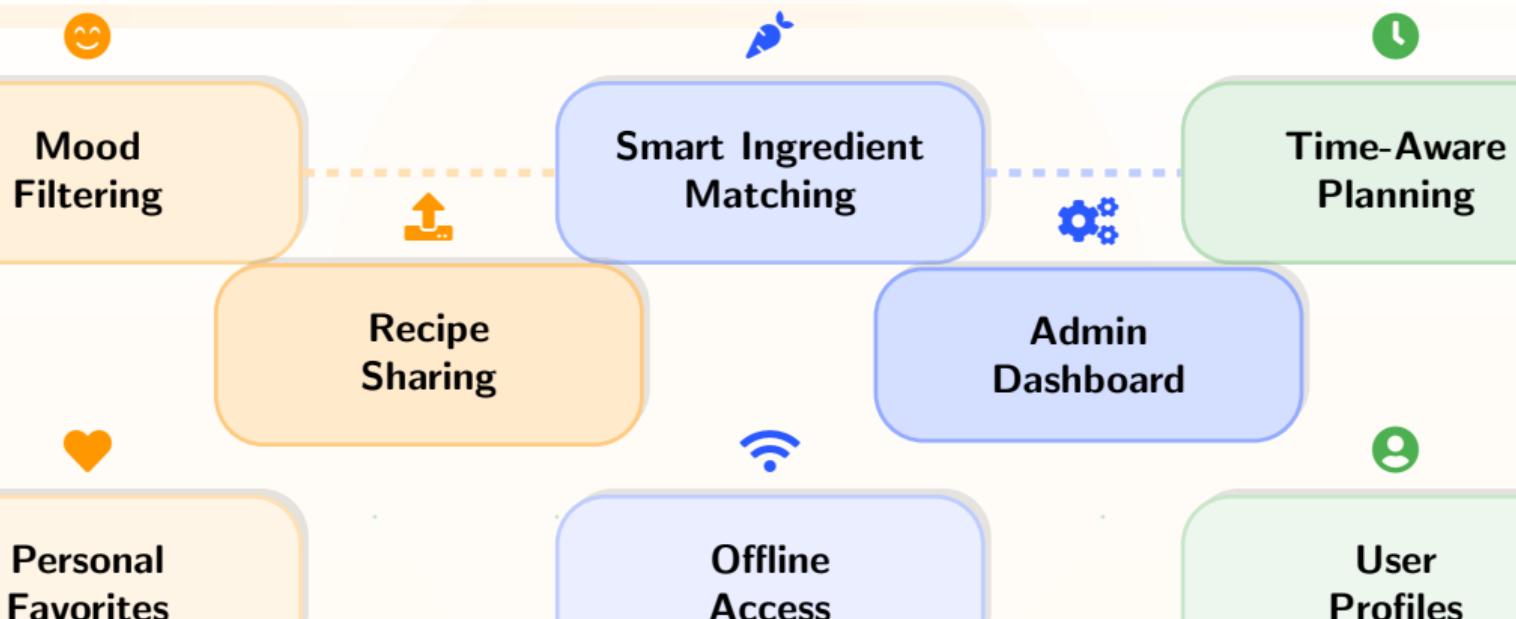
Modern Stack for Scalable Solutions

📱 Frontend	🗄 Backend	🔧 DevOps & Tools
</> Flutter	⚡ Node.js	○ Git/GitHub
📦 Provider State	🚀 Express.js	☰ Jenkins
🖌 Material UI	🛢 MySQL	🌐 Nginx
	🔒 JWT + Bcrypt	☁ Contabo VPS
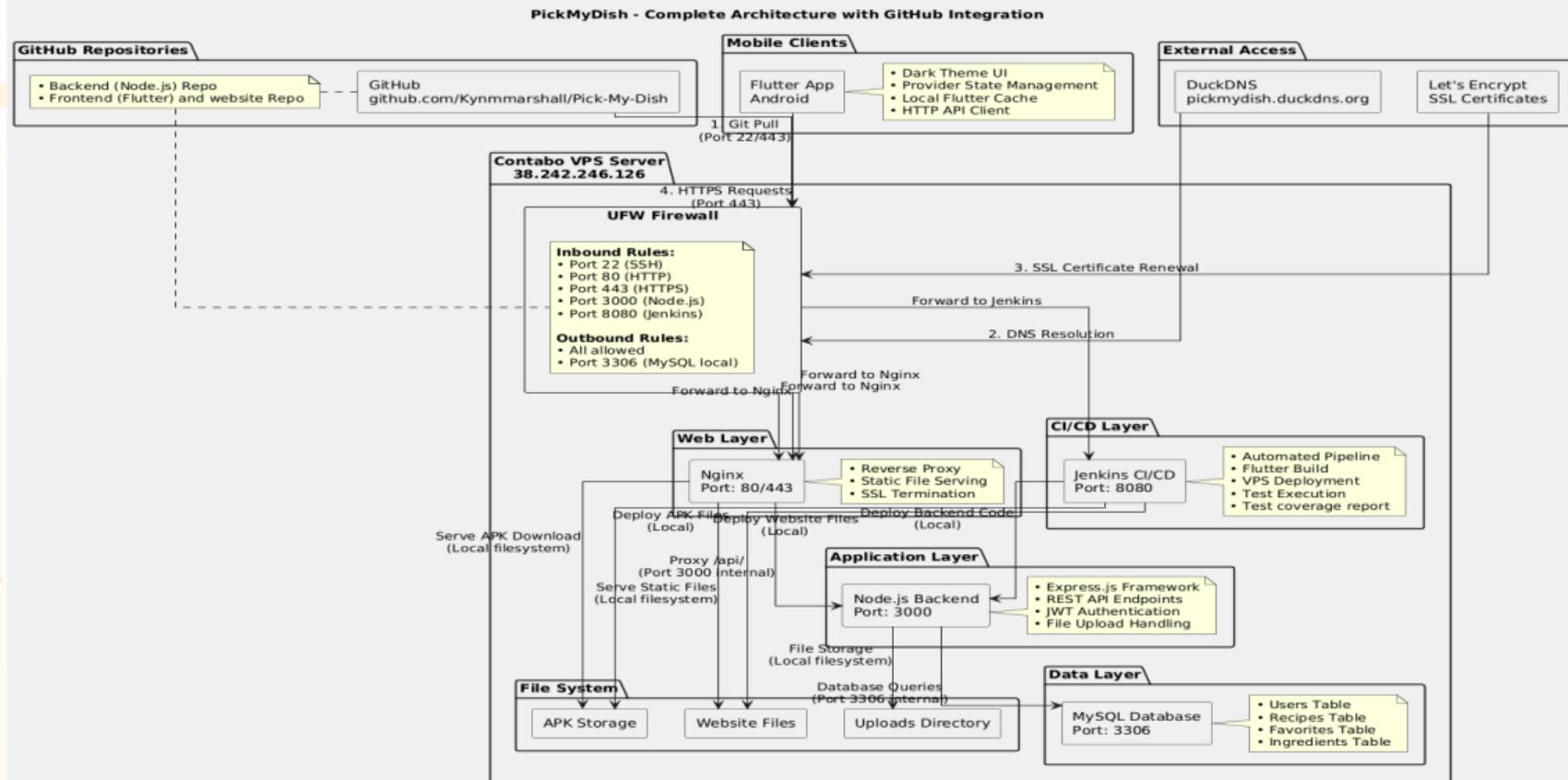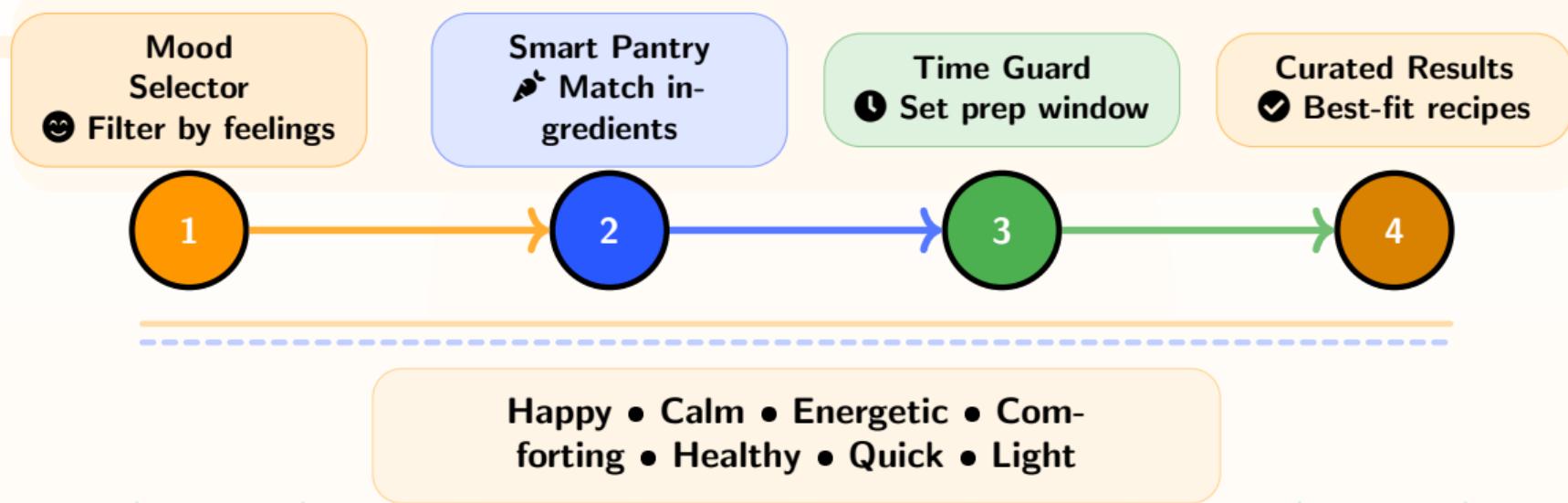
System Architecture

PickMyDish - Complete Architecture with GitHub Integration

GitHub Repositories
- Backend (Node.js) Repo
- Frontend (Flutter) and website Repo

GitHub
github.com/Kynmmarshall/Pick-My-Dish

Mobile Clients

Flutter App
Android
- Dark Theme UI
- Provider State Management
- Local Flutter Cache
- HTTP API Client

External Access

DuckDNS
pickmydish.duckdns.org

Let's Encrypt
SSL Certificates

1. Git Pull
(Port 22/443)

Contabo VPS Server
38.242.246.126

4. HTTPS Requests
(Port 443)

UFW Firewall

Inbound Rules:
- Port 22 (SSH)
- Port 80 (HTTP)
- Port 443 (HTTPS)
- Port 3000 (Node.js)
- Port 8080 (Jenkins)

Outbound Rules:
- All allowed
- Port 3306 (MySQL local)

3. SSL Certificate Renewal

Forward to Jenkins

2. DNS Resolution

Forward to Nginx
Forward to Nginx
Forward to Nginx

Web Layer

Nginx
Port: 80/443
- Reverse Proxy
- Static File Serving
- SSL Termination

CI/CD Layer

Jenkins CI/CD
Port: 8080
- Automated Pipeline
- Flutter Build
- VPS Deployment
- Test Execution
- Test coverage report

Serve APK Download
(Local filesystem)

Deploy APK Files
(Local)

Deploy Website Files
(Local)

Deploy Backend Code
(Local)

Proxy /api/
(Port 3000 internal)

Serve Static Files
(Local filesystem)

Application Layer

Node.js Backend
Port: 3000
- Express.js Framework
- REST API Endpoints
- JWT Authentication
- File Upload Handling

File Storage
(Local filesystem)

File System

APK Storage

Website Files

Uploads Directory

Database Queries
(Port 3306 internal)

Data Layer

MySQL Database
Port: 3306
- Users Table
- Recipes Table
- Favorites Table
- Ingredients Table

User Flow: Mood-Based Recipe Discovery

Mood Selector 😊 Filter by feelings

Smart Pantry 🥕 Match ingredients

Time Guard 🕐 Set prep window

Curated Results ✔ Best-fit recipes

1 → **2** → **3** → **4**

Happy • Calm • Energetic • Comforting • Healthy • Quick • Light
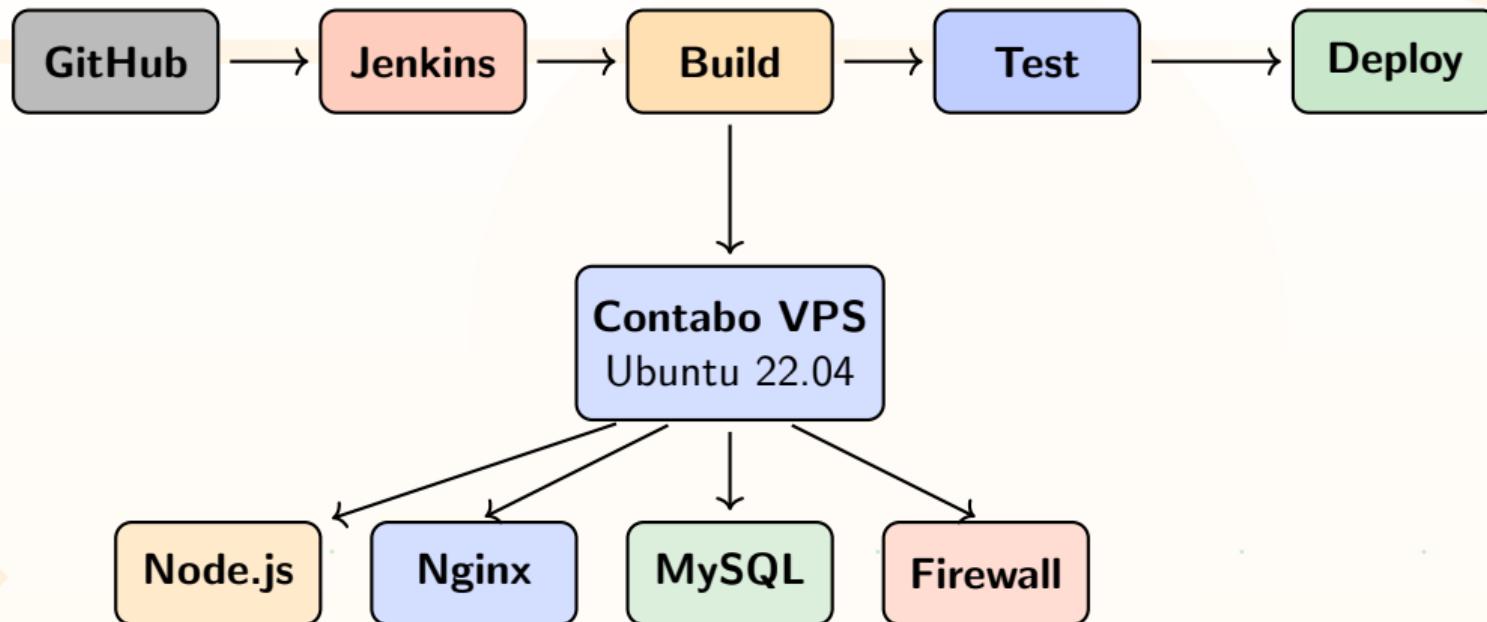
Why it impresses: Adaptive filtering blends mood, pantry items, and prep time to present a shortlist that feels tailored. Real-time scoring ranks recipes by relevance, freshness, and dietary fit.

Design Patterns

- **Provider Pattern**
  State management

- **Repository Pattern**
  Data abstraction

- **Singleton Pattern**
  Shared instances

- **Factory Pattern**
  Object creation
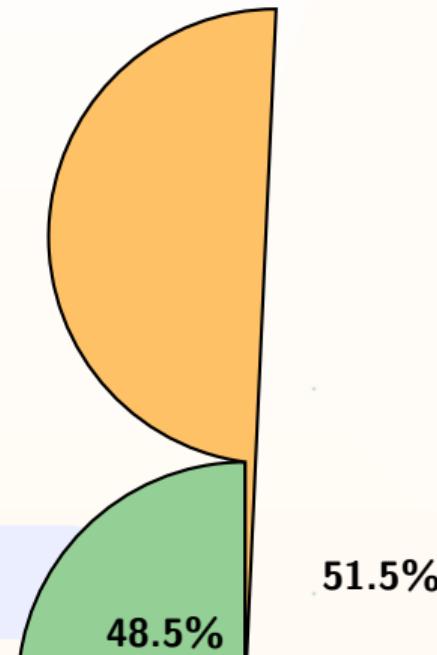
Deployment & DevOps Pipeline

99.8% Uptime — CI/CD Automation — Real-time Monitoring

Test Coverage

51.5%

48.5%

Performance

Metric	Target	
Recipe Load Time	¡2s	
API Response	¡500ms	
Database Query	¡300ms	
Uptime	99.8%	

Conclusion & Future Roadmap

Achievements

- Fully functional cross-platform app

- Secure authentication system

- Offline-first architecture

- Fast, responsive UI

- Automated deployment

- Comprehensive testing

Future Enhancements

- AI-powered recommendations

- Multi-language support

- Advanced analytics

- Smart notifications

- Nutrition tracking

- Social sharing

Thank You!

Questions?