Unleash - Feature Flag Management Platform

# What is Unleash?

Unleash is an open-source feature flag and toggle management platform that enables teams to control feature rollouts, perform A/B testing, and manage feature flags across multiple environments. It provides a powerful API and user-friendly dashboard for managing feature flags in production applications.

# Key Concepts

Feature Flags (Toggles):

A feature flag is a boolean value that controls whether a feature is enabled or disabled. Features can be:

- Enabled: Feature is active and visible to users

- Disabled: Feature is inactive and hidden from users

- Gradual Rollout: Feature is enabled for a percentage of users

Strategies:

Unleash supports various rollout strategies:

- Default: Simple on/off toggle

- Gradual Rollout: Percentage-based rollout

- User Targeting: Enable for specific users

- Environment Targeting: Enable for specific environments

- Custom Strategies: User-defined rollout logic

# What is Unleash Used For?

1. Gradual Feature Rollouts

- Safely deploy new features to a percentage of users

- Monitor performance and user feedback

- Roll back quickly if issues arise

2. A/B Testing

- Test different versions of features

- Compare user engagement and performance

- Make data-driven decisions about feature releases

3. Kill Switch

- Quickly disable problematic features in production

- Prevent widespread issues from affecting all users

- Maintain system stability during incidents

4. Targeted Rollouts

- Release features to specific user segments

- Enable features for beta testers or premium users

- Control feature access based on user attributes

5. Environment Management

- Manage different feature states across environments

- Test features in development and staging

- Promote features safely to production

# How to Set Up Unleash for Local React-Based Application

Step 1: Set Up Unleash Server - Create a docker-compose.yml file.

Step 2: Start Unleash

docker-compose up -d

Step 3: Access Unleash Dashboard - http://localhost:4242

Step 4: Create Your First Feature Flag - Use dashboard

Step 5: Set Up React Application

- npm install @unleash/proxy-client-react

- Configure Unleash Client in src/unleash.js

- Wrap App in UnleashProvider

- Use feature flags with useFlag

Step 6: Get Client Key from Dashboard

Step 7: Test Your Setup - Toggle features and verify in React app

# Environment Configuration

Development Environment (.env.development):

REACT\_APP\_UNLEASH\_URL=http://localhost:4242/api/frontend

REACT\_APP\_UNLEASH\_CLIENT\_KEY=your-dev-client-key

REACT\_APP\_UNLEASH\_ENVIRONMENT=development

Production Environment (.env.production):

REACT\_APP\_UNLEASH\_URL=https://your-unleash-instance.com/api/frontend

REACT\_APP\_UNLEASH\_CLIENT\_KEY=your-prod-client-key

REACT\_APP\_UNLEASH\_ENVIRONMENT=production

# Key Benefits of Using Unleash

1. Risk Reduction

2. Faster Development

3. Better Testing

4. Improved User Experience

5. Operational Control

# Common Use Cases

- Feature Rollouts

- A/B Testing

- Kill Switches

- User Targeting

- Environment Management

# Troubleshooting

Common Issues:

- Unleash not starting: Check Docker logs

- React app not connecting: Check URL and key

- Feature flags not updating: Check refresh and connectivity

- Dashboard not accessible: Check port/firewall

Debug Commands:

curl http://localhost:4242/api/health

docker-compose logs -f unleash

docker-compose ps

# Conclusion

Unleash is a powerful and flexible platform for managing feature flags in React apps. It supports safe deployments, A/B testing, and full control of features across environments. Follow this guide for local setup and integration.