# OpenSesame Developer Guide & Enterprise Integration Reference

## Architecture Summary

The OpenSesame system is composed of modular components:

- Frontend: React.js web interface

- Backend: FastAPI with PostgreSQL

- Edge AI: PyTorch model running on PYNQ-Z2

- Message Passing: Redis for real-time updates

## Extending the Platform

From sprint artifacts #17, #50, and #62, developers can:

- Build new facial recognition models using PyTorch

- Secure API endpoints with OAuth2-based token flows

- Add RESTful or MQTT-based integrations for external dashboards

- Streamline deployments with Docker Compose

- Automate updates via GitHub Actions or cron-based sync

## Hardware Development Insights

Our sprint work on the camera connection (artifacts #35, #48, #58) and housing integration (#46, #59) highlights the project's hands-on nature. Developers will find pre-built scripts for audio capture (#42), button integration (#53), and hardware diagnostics.

## Enterprise Use Case Scenarios

- Office visitor logging with secure alerts

- AI-powered door entry logging for rental property management

- Integration with company intranet or Slack for team notifications

- Ability to deploy a complete self-managed security system on a weekend