

# Python Projects to Land Your First Software Engineering Role in London

## What Employers Want:

1. Strong Python and software engineering fundamentals.
2. Problem-solving and real-world usefulness.
3. Local relevance (finance, tech, data-driven fields).

## Top-Tier Projects That Impress London Employers:

### 1. Full-Stack Web App with Django + React

- Example: A budgeting app with user login and expense tracking.
- Skills: Django REST API, PostgreSQL, React, Docker, CI/CD.
- Why: Demonstrates full-stack development.
- CV Line: Built and deployed a full-stack budgeting app with Django REST API and React.

Dockerized and deployed via Heroku.

### 2. Finance or Crypto Tracker with APIs

- Example: Crypto dashboard using public APIs.
- Skills: requests, pandas, matplotlib, Dash or Streamlit.
- Why: Fintech is big in London.
- CV Line: Developed a dashboard with Streamlit to track real-time crypto prices using CoinGecko

API.

### 3. Automated Job Application Bot

- Example: Scrapes job boards and sends applications.
- Skills: Selenium, BeautifulSoup, Google Sheets API.

- Why: Shows automation skills and initiative.
- CV Line: Created a bot using Selenium to scrape jobs and log them in Google Sheets.

#### 4. Machine Learning Mini Projects

- Examples: Spam classifier, house price predictor.
- Why: Good for tech/data roles.
- CV Line: Trained a spam classifier with 94% accuracy, deployed with Flask on AWS EC2.

#### 5. Open Source Contributions

- Example: Fixing bugs in Python libraries.
- Why: Proves real-world collaboration and skill.
- CV Line: Contributed fixes and docs to requests Python library.

#### Bonus/Niche Ideas:

- Real-time London Underground Tracker (TfL API)
- Rental Price Analysis App (web scraping)
- AI CV Screening Tool
- Book club or review app
- Personal portfolio with Flask/Django

#### What Makes a Project Great:

- Hosted on GitHub with live demo
- Clean README, instructions
- Includes basic tests
- Well-structured commits
- Optional blog post

Tools to Learn:

- Git + GitHub
- Django, Flask, FastAPI
- Docker, CI/CD
- Frontend basics (HTML/CSS/JS or React)
- Deployment (Heroku, Render, AWS)

Tip: Want tailored ideas? Let me know your interests or stack.