Server-Factory-Additional-Components

Server Factory प्रोविज़निंग टूलचेन के सहायक घटक

सारांश

Mail Server Factory और कोर फ्रेमवर्क के अलावा, सर्वर-फैक्ट्री संगठन में कई छोटे घटक शामिल हैं: प्रति-सेवा "फैक्ट्रियाँ" (वेब सर्विस, SonarQube, कैशिंग प्रॉक्सी), घोषणात्मक कॉन्फ़िगरेशन पैक (Docker/स्टैक/सॉफ़्टवेयर परिभाषाएँ), और साझा यूटिल्स। यह संकलित पृष्ठ उन्हें ईमानदारी से प्रस्तुत करता है—कई प्रारंभिक चरण में हैं या अधूरे दस्तावेज़ीकरण वाले—न कि पूर्णतः निर्दिष्ट उत्पादों के रूप में।

संक्षिप्त विवरण

Server Factory के सहायक रिपॉज़िटरी का समूह: वेब-सर्विस-फैक्ट्री, SonarQube-फैक्ट्री, और कैशिंग-प्रॉक्सी-फैक्ट्री (प्रति-सेवा प्रोविज़निंग टूल, अधिकतर प्रारंभिक चरण में); Docker/स्टैक/सॉफ़्टवेयर-परिभाषाएँ (फ्रेमवर्क द्वारा उपयोग किए जाने वाले घोषणात्मक कॉन्फ़िगरेशन पैक); और यूटिल्स (SSH-एक्सेस सहायक और सामान्य टूलिंग)। सभी कोर फ्रेमवर्क पर आधारित हैं।

विस्तृत विवरण

यह पृष्ठ शेष सर्वर-फैक्ट्री रिपॉज़िटरी को एक साथ प्रस्तुत करता है क्योंकि व्यक्तिगत रूप से अधिकांश छोटे हैं या जानबूझकर कम दस्तावेज़ीकृत हैं, और प्रत्येक को एक पूर्ण उत्पाद के रूप में प्रस्तुत करना उनकी परिपक्वता को बढ़ा-चढ़ाकर दिखाएगा। ये तीन समूहों में विभाजित हैं। **सेवा फैक्ट्रियाँ** अन्य सर्वर भूमिकाओं के लिए Mail Server Factory पैटर्न का अनुसरण करती हैं: **कैशिंग-प्रॉक्सी-फैक्ट्री** ("अपना खुद का कैशिंग प्रॉक्सी सर्वर चलाएँ") कैशिंग प्रॉक्सी, सेल्फ-साइन्ड सर्टिफ़िकेट, और सुरक्षा-सर्टिफ़िकेट प्राप्ति के HTTP एंडपॉइंट को प्रमुख विशेषताओं के रूप में सूचीबद्ध करती है; **SonarQube-फैक्ट्री** ("अपना खुद का SonarQube सर्वर चलाएँ") सॉफ़्टवेयर-विकास उपयोग के लिए लक्षित है; और **वेब-सर्विस-फैक्ट्री** वेबसाइटों और माइक्रो-सर्विसेज़ जैसे लक्ष्यों को तैनात करने के लिए वेब सर्वर को स्थापित और कॉन्फ़िगर करती है। ये तीनों Kotlin प्रोजेक्ट हैं जो कोर फ्रेमवर्क पर निर्मित हैं, लेकिन उनके सार्वजनिक README अधिकतर प्लेसहोल्डर हैं ("Tbd." संगतता, विनिर्देश, सेटअप और उपयोग के लिए)—इसलिए बताए गए उद्देश्य से परे उनकी ठोस क्षमताएँ **अपुष्ट** हैं। **परिभाषा पैक—Docker-परिभाषाएँ**, **स्टैक-परिभाषाएँ**, और **सॉफ़्टवेयर-परिभाषाएँ**—घोषणात्मक कॉन्फ़िगरेशन रिपॉज़िटरी हैं जिन्हें फ्रेमवर्क Docker छवियों, स्टैक और सॉफ़्टवेयर के निर्माण/तैनाती के लिए उपयोग करता

है; ये संस्करण-नियत डेटा पैक हैं, न कि एप्लिकेशन। **यूटिल्स** परिवार के लिए सामान्य सहायक प्रदान करता है, जिसमें `init_ssh_access.sh` स्क्रिप्ट शामिल है जो SSH कुंजी उत्पन्न करती है और इसे दूरस्थ होस्ट पर स्थापित करती है ताकि बाद के प्रोविज़निंग के लिए पासवर्ड-रहित रूट एक्सेस सक्षम हो सके। ये घटक मिलकर Mail Server Factory के आसपास प्रोविज़निंग टूलचेन को पूर्ण करते हैं।

हमने इसे क्यों बनाया

Server Factory मॉडल को सामान्यीकृत करने के लिए बनाया गया है: एक बार जब आप घोषणात्मक विवरण से मेल सर्वर प्रोविज़न कर सकते हैं, तो उसी इंजन से वेब सर्वर, कैशिंग प्रॉक्सी और कोड-क्वालिटी सर्वर भी प्रोविज़न किए जा सकने चाहिए—जिन्हें पुनः उपयोग योग्य परिभाषा पैक और साझा यूटिलिटीज़ द्वारा संचालित किया जाए, न कि प्रति भूमिका के लिए विशेष तर्क द्वारा। ये रिपॉज़िटरी उसी सामान्यीकरण की प्रक्रिया में हैं, जो सिद्ध पैटर्न को नए सर्वर प्रकारों तक विस्तारित करती हैं। यहाँ इनका मूल्य मॉडल की पहुँच के प्रमाण के रूप में है; उनकी परिपक्वता भिन्न है, और यह पृष्ठ जानबूझकर स्पष्ट करता है कि कौन से दिशा-निर्देश हैं और कौन से पूर्ण हैं।

सामग्री

यह क्यों एक गेम-चेंजर है (संतुलित दृष्टिकोण)

समग्र रूप में, ये इस बात का प्रमाण हैं कि कोर फ्रेमवर्क विभिन्न सर्वर प्रकारों में पुनः उपयोगी है और घोषणात्मक डेटा (परिभाषाएँ) को निष्पादन (फैक्ट्रियाँ) से अलग करता है। व्यक्तिगत रूप से, सेवा फैक्ट्रियाँ प्रारंभिक चरण में हैं और इन्हें पूर्ण उत्पाद के रूप में नहीं, बल्कि दिशा-निर्देश के रूप में प्रस्तुत किया जाना चाहिए।

क्या है नवीन

- एक ही प्रोविज़निंग फ्रेमवर्क जो मेल/वेब/कैशिंग-प्रॉक्सी/SonarQube भूमिकाओं में सामान्यीकृत है।
- घोषणात्मक परिभाषा पैक (Docker/स्टैक/सॉफ़्टवेयर) जो निष्पादन इंजन से अलग हैं।
- साझा यूटिल्स (जैसे, एक-कमांड पासवर्ड-रहित SSH बूटस्ट्रैप) जो फैक्ट्रियों में पुनः उपयोगी हैं।

चुनौतियाँ और समाधान

- **एक ही इंजन का विभिन्न सर्वर भूमिकाओं में पुनः उपयोग:** कोर फ्रेमवर्क पर प्रत्येक फैक्ट्री बनाकर हल किया गया।
- **कॉन्फ़िगरेशन को कोड से अलग करना:** परिभाषा रिपॉज़िटरीज़ को संस्करण-संलग्न डेटा पैक के रूप में उपयोग करके हल किया गया।
- **(असत्यापित):** सेवा फैक्ट्रियों के README प्लेसहोल्डर हैं; सार्वजनिक दस्तावेज़ों से उनकी पूर्णता का सत्यापन संभव नहीं — इन्हें प्रारंभिक चरण के रूप में प्रस्तुत करें।

तकनीकी स्टैक (क्यों + कैसे)

- **Kotlin** — वेब-सर्विस-फैक्ट्री, SonarQube-फैक्ट्री, कैशिंग-प्रॉक्सी-फैक्ट्री (कोर फ्रेमवर्क पर निर्मित)।
- **Shell** — यूटिल्स और परिभाषा पैक (स्क्रिप्ट्स/कॉन्फ़िग)।
- **Gradle** — `./gradlew test` फैक्ट्रियों में बिल्ड/टेस्ट प्रवाह।
- **Docker** — Docker-परिभाषाओं द्वारा वर्णित लक्षय रनटाइम।
- **SSH / OpenSSH** — यूटिल्स का पासवर्ड-रहित एक्सेस बूटस्ट्रैप।
- **SonarQube** — वह सर्वर जिसे SonarQube-फैक्ट्री प्रोविज़न करती है (और जिसके विरुद्ध Mail Server Factory एक स्वच्छ गेट की रिपोर्ट करता है)।

> ईमानदारी का नोट: इनमें से अधिकांश रिपॉज़िटरी संगठन के भीतर फ़ोर्क हैं; सेवा फैक्ट्रियाँ प्लेसहोल्डर-दस्तावेज़ीकृत हैं और संविधान §11.4.6 के अनुसार असत्यापित चिह्नित हैं। इन्हें स्पष्ट रूप से Mail Server Factory और कोर फ्रेमवर्क से निम्न श्रेणी में रखा गया है।