Exercise 2.6: Writing a Domain-specific ADR

Using what we learned so far, write an ADR for our Compliance domain, documenting Compliance at the Point of Change with, giving the context of our compliance domain and how it interrelates with the Continuous Deployment, Security, and Self-service domains.

Solution

At PE Tech, our Control Plane platform supports developers across multiple continents and potentially multiple cloud providers, like AWS and GCP. This demands a highly available and globally distributed system, including databases. However, achieving this while maintaining a developer-friendly experience and accommodating different cloud provider requirements presents challenges. Additionally, the acquisition of a company using GCP complicates our requirement for multi-cloud support.

Here are the key challenges

1. **Global Availability:** The Control Plane needs to be resilient, fast, and replicated across regions for global availability.
2. **Developer Experience:** Ensuring the development environment is not overly complex or cumbersome.
3. **Multi-Cloud Support:** Supporting platforms like AWS DynamoDB while accommodating new GCP-based services.

We will adopt an architecture that abstracts the database layer through a decoupled service and repository pattern. The key components will include:

- **Service Layer:** Implements business logic and interacts only with the repository layer.
- **Repository Layer:** Interacts with the datastore interface, providing a consistent way to interact with data.
- **Datastore Interface:** Abstracts the database implementation, allowing different database technologies to be used without affecting higher layers.

Fitness functions will be used to ensure this architecture remains consistent over time, specifically:

- **Layered Architecture:** Enforce boundaries between Service, Repository, and Datastore layers.
- **Compliance Verification:** Ensure repository layer interacts with the datastore interface, preventing service layer access to datastore functions directly.
- **Interface Implementation:** Verify that all Datastore implementations adhere to the Datastore interface.

Based on the ADR, we see the consequences as follows

- **Flexibility:** This architecture allows us to change databases without impacting the Control Plane logic, supporting various databases across cloud providers.
- **Compliance:** Fitness functions ensure the architecture integrity and compliance across changes, preventing architectural violations.
- **Scalability:** This approach supports global distribution and replication, while ensuring the system is scalable.