Epic: Implement a Comprehensive Inventory Management System on GCP Platform that ~~provides insights into resources deployed on GCP and publish the resource data to various~~ consumers (including IDEAs)

As a cloud operations and engineering team, we want to implement a system for tracking, managing and optimizing all cloud resources across GCP to ensure visibility and compliance with asset management requirements of Citi

User Story 1: Automatic Discovery of GCP Resources

- **As a** Cloud Engineer
- **I want to** automatically discover all resources (VMs, storage buckets, databases, etc.) across my GCP projects
- **So that I can** maintain an up-to-date inventory of resources without manual intervention

Acceptance Criteria:

- Automatic resource discovery is enabled across all GCP projects and regions.
- A centralized inventory list is generated showing all resources, their types, and locations.
- New resources are added to the inventory in real-time when created or deployed.

User Story 2: Centralized Inventory Dashboard

- **As a** Cloud Operations Manager
- **I want to** view all GCP resources across projects in a centralized dashboard
- **So that I can** quickly assess the status, configuration, and usage of resources in one place

Acceptance Criteria:

- A centralized dashboard lists all resources by type (Compute, Storage, Network, etc.).
- Resource details include project name, region, resource type, status, and configuration.
- The dashboard includes filter options by project, region, and resource type.

User Story 3: Resource Tagging and Categorization

- **As a** Cloud Administrator
- **I want to** tag and categorize GCP resources based on department, cost center, environment (prod, dev, test), and owner
- **So that I can** better organize resources and track ownership for management purposes

Acceptance Criteria:

- Resources can be tagged with custom labels (e.g., cost center, department, environment).
- The inventory management system supports bulk tagging for existing resources.

- Reports can be generated based on tags to track resource usage and costs by category.

User Story 4: Resource Configuration and Compliance Auditing

- **As a** Controls Engineer
- **I want to** track the configuration of GCP resources and ensure compliance with security and operational policies
- **So that I can** detect configuration drift and ensure that resources are securely configured

Acceptance Criteria:

- Configuration details for each resource are tracked and stored in the inventory.
- Compliance checks are run to ensure that resources adhere to security policies (e.g., encryption, firewall settings).
- Non-compliant resources are flagged, and alerts are generated for remediation.

User Story 5: Resource Dependency Mapping

- **As a** Cloud Operations
- **I want to** visualize the dependencies between GCP resources (e.g., VMs, databases, load balancers)
- **So that I can** understand how different resources are connected and optimize infrastructure design

Acceptance Criteria:

- The system generates a visual map showing dependencies between resources (e.g., compute instances connected to databases and storage).
- I can view resource relationships by project, region, and network configuration.
- The map is updated dynamically as resources are added, removed, or reconfigured.

User Story 6: Alerting for Resource Changes

- **As a** Cloud SRE/OPS
- **I want to** receive real-time alerts when critical resources are modified or deleted
- **So that I can** ensure resource changes are tracked and unauthorized modifications are detected

Acceptance Criteria:

- Alerts are triggered when significant changes are made to critical resources (e.g., firewall rules, VM shutdowns).
- Alerts can be configured based on resource type, change type, or severity.
- All resource changes are logged and linked to user actions for audit purposes.

User Story 7: Automated Inventory Feed to Drift

- **As a** Cloud Infrastructure Engineer
- **I want to** automatically sync my GCP inventory to IDEAS DRIFT System
- **So that I can** ensure effective service management and adherence to Citi inventory management policy

  Acceptance Criteria:

  - The system syncs GCP resource data to DRIFT System in a timely manner (according to SLO).
  - Differences between resource data in DRIFT and GCP asset are flagged.
  - Alerts are triggered if drift is detected between Google Asset Inventory and GCP resources in DRIFT.

User Story 8: Historical Resource Tracking

- **As a** Cloud Operations Manager
- **I want to** track the history of GCP resource creation, modification, and deletion
- **So that I can** analyze trends and investigate issues related to resource management

  Acceptance Criteria:

  - The system stores historical data on resource creation, modifications, and deletions.
  - I can generate reports showing resource changes over time (e.g., scaling up/down, reconfigurations).
  - The inventory system includes audit logs with details of who made changes and when.

User Story 9: Cloud Asset Inventory API

- **As a** Cloud Engineer
- **I want to** query and generate a report of various GCP resource information
- **So that I can** provide timely information to my stakeholders on the latest state of the GCP environment

  Acceptance Criteria:

  - The system stores latest snapshot of cloud resource information.
  - I can generate reports showing resource details and attributes.
  - The inventory api that have GET/POST routes to retrieve data in JSON format

User Story 10: Cloud CICD Inventory API

- **As a** Cloud CICD Pipeline Developer

- **I want to** query an api for CSI -> GCP project mapping
- **So that I can** validate that a pipeline can deploy resources to a specified GCP project

Acceptance Criteria:

  - The system stores latest snapshot of cloud resource information.
  - The inventory api that have GET/POST route to retrieve GCP project details for given CSI