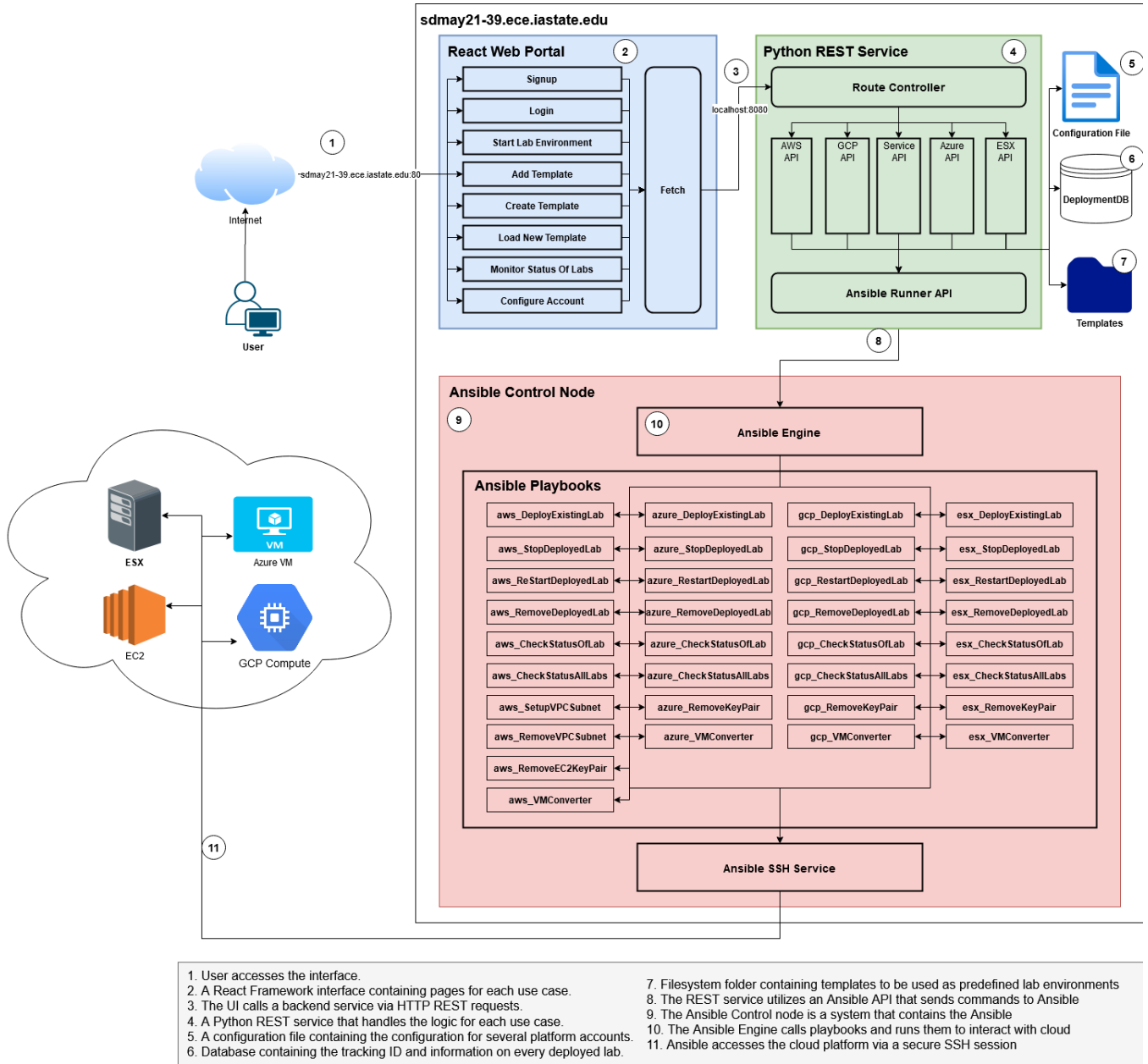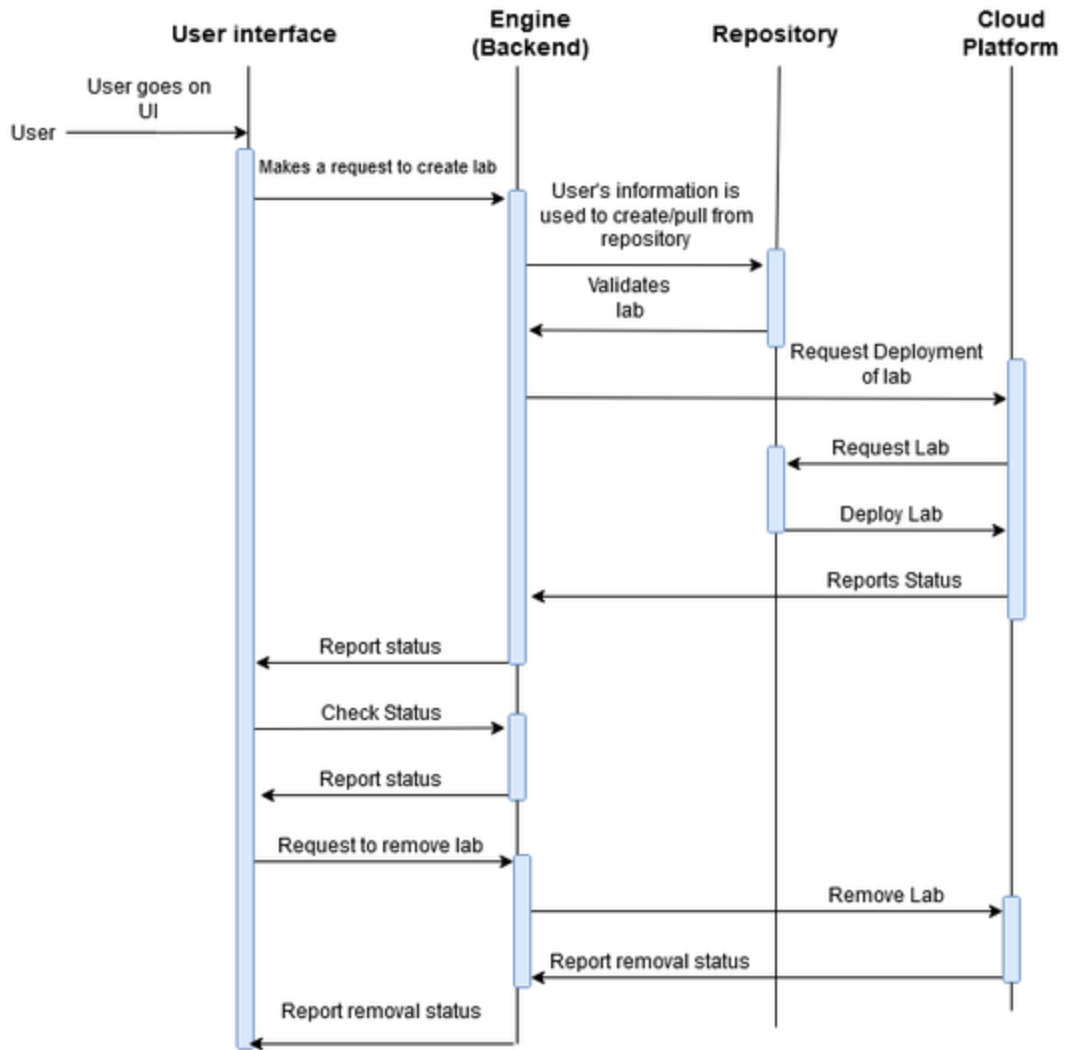# Overall Design

General idea is to have a UI that interacts with a backend engine. The backend will have functions to validate labs, maintain deployed lab, and track status. There will be a repository that holds the labs that PwC uses that will be accessed from the backend and from the cloud platform.

Architecture diagram

1. User accesses the interface.
2. A React Framework interface containing pages for each use case.
3. The UI calls a backend service via HTTP REST requests.
4. A Python REST service that handles the logic for each use case.
5. A configuration file containing the configuration for several platform accounts.
6. Database containing the tracking ID and information on every deployed lab.
7. Filesystem folder containing templates to be used as predefined lab environments
8. The REST service utilizes an Ansible API that sends commands to Ansible
9. The Ansible Control node is a system that contains the Ansible
10. The Ansible Engine calls playbooks and runs them to interact with cloud
11. Ansible accesses the cloud platform via a secure SSH session

Sequence Diagram

https://app.diagrams.net/#G1n4mc4DX7A2t3n4u9iJBTlBu8qg6QWiVJ

General Flow

AWS   GCP   Azure

Lab
1 .. N

Lab
1 .. N

Lab
1 .. N

AWS

Back-End
Orchestrator
(ECS)

Lab Image
Repository
(ECR)

Back-End
Load Balancer

Front-End
User Interface
(ECS)

Front-End
Load Balancer

Route 53

Admin