The below tabular column is the best assumption list of services that can be used based on the requirement of applications.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S No** | **Components** | **Staging** | **UAT** | **Production** |
| 1 | User Interface  API Gateway  Authentication API  Public Endpoints  Internal APIs | Apigee | Apigee | Apigee |
| 2 | Long Running Workers/Jobs | EC2 Instance (With Services Installed) | AWS EKS | AWS EKS |
| 3 | Queues (RabbitMQ) | EC2 Instance (With Services Installed) | CloudAMQP (RabbitMQ as a service) | CloudAMQP (RabbitMQ as a service) |
| 4 | Database |  |  |  |
|  | Non-Relational | EC2 Instance (With Services Installed) | MongoDB Atlas | MongoDB Atlas |
|  | Relational | EC2 Instance (With Services Installed) | Amazon Aurora | Amazon Aurora |
|  | Data Warehouse | EC2 Instance (With Services Installed) | Snowflake | Snowflake |
|  | Graph Database | EC2 Instance (With Services Installed) | Neo4j | Neo4j |
| 5 | Redis | EC2 Instance (With Services Installed) | AWS ElastiCache | AWS ElastiCache |
| 6 | OAuthorization Server | Okta | Okta | Okta |

Alerts and Monitoring across all the above-mentioned components can be done by **New Relic**.

New Relic helps to monitor entire resources and components in a single dashboard. It provides deep insights and real time data.

Provisioning Infrastructure can be done by **Terraform.** Its declarative approach provides good results in maintaining complex infrastructure and it is cloud agnostic.

Configuration management can be done by **Ansible**. Define our playbooks with necessary tasks and use modules to automate the patch management and installation of required services.

Deployment can be done by using **CircleCI.** Deploy each service in all cloud providers by using **ORBs** in CircleCI. Kubernetes deployment can also be automated by combination of **HELM** charts and Orb of CircleCI.