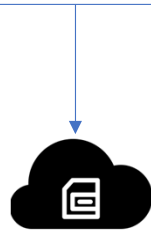
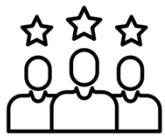




Scalability

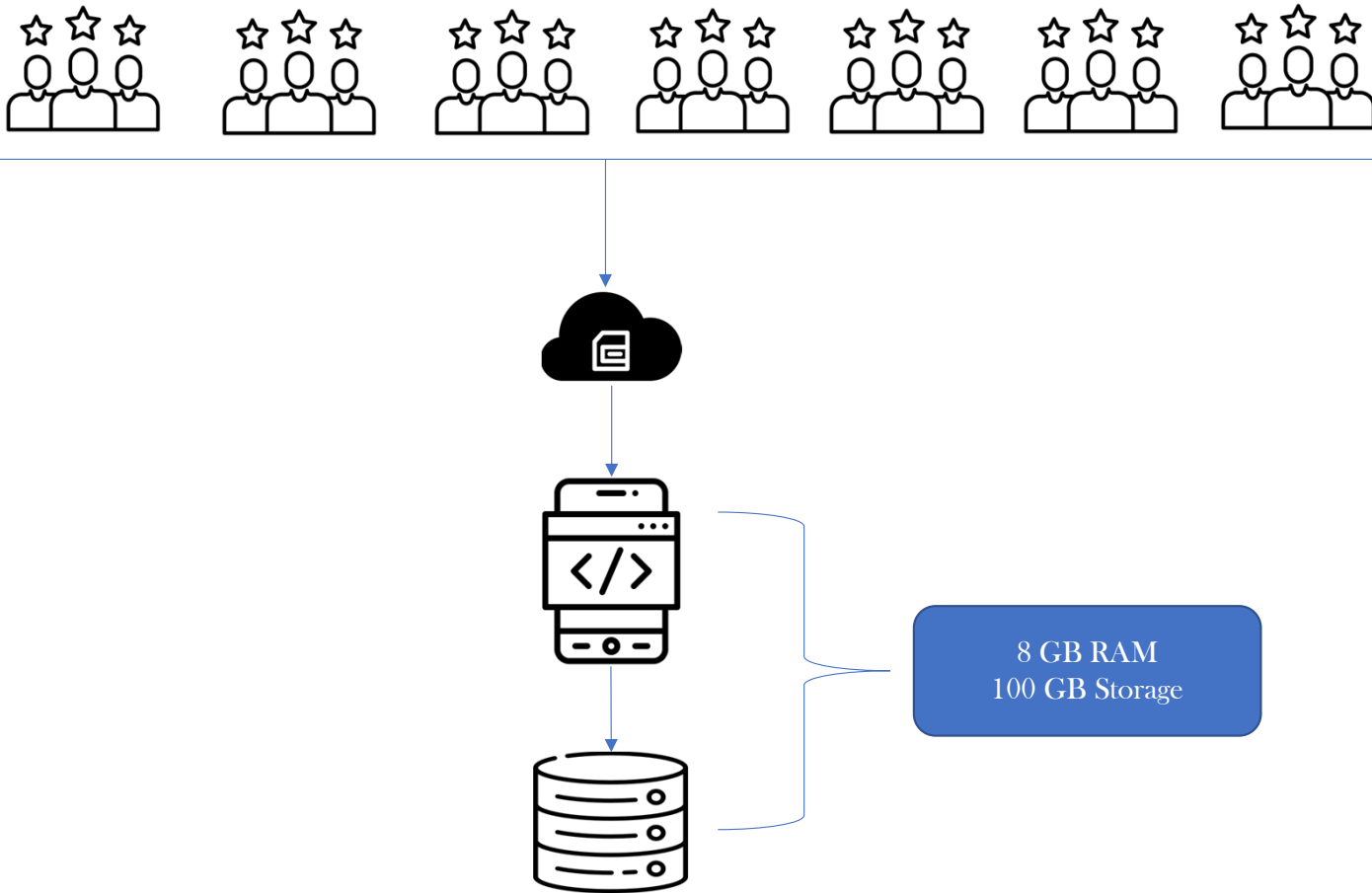
- 3 Aspects that's crucial to any application development and deployment :
 - Scalability
 - Security
 - Monitoring and Alerting
- Scalability applies to any piece of software running on a server :
 - ETL Pipelines
 - Machine learning Model
 - Client serving web applications
 - Data-warehouses
- 2 Types of scaling mechanisms :
 - Vertical Scaling
 - Horizontal Scaling

Vertical Scaling



4 GB RAM
50 GB Storage

Vertical Scaling





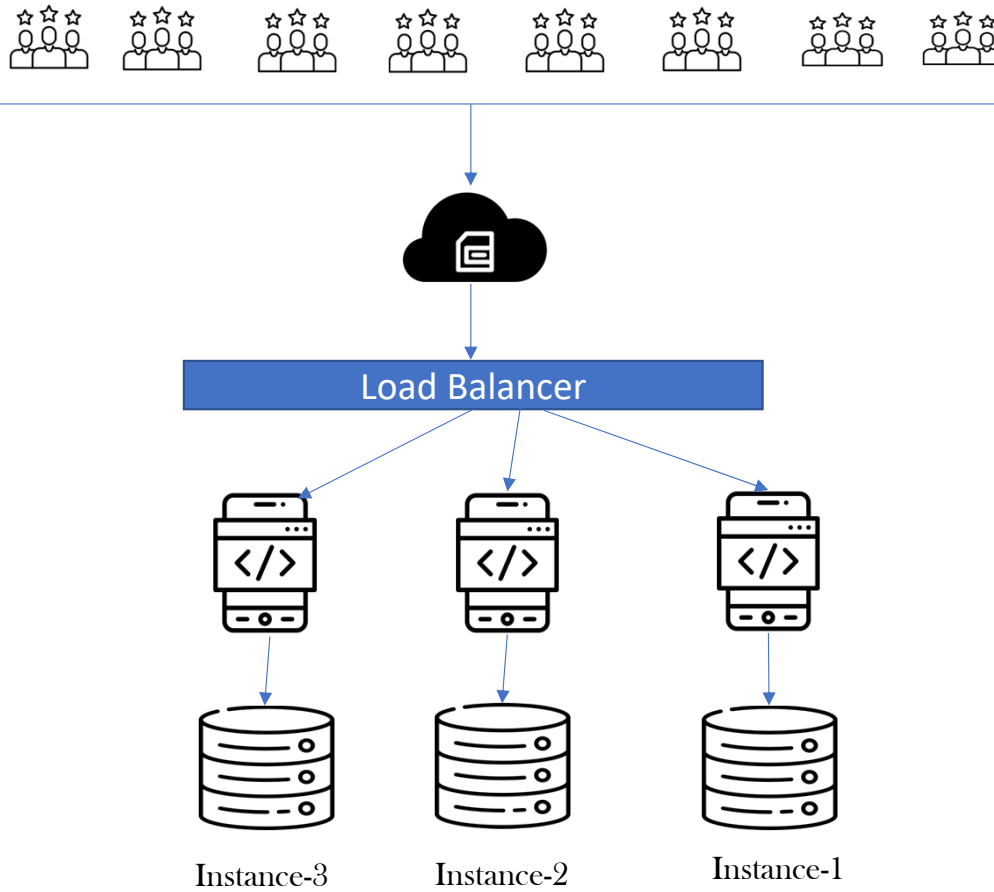
Vertical Scaling Advantages/Limitations

- Single Point of Failure
- Costs are relatively higher because of high-end servers
- Limit to which servers can scale
- Downtime or interruptions during maintenance or migrations or any application level changes

Advantages :

- Easier to maintain
- Data lives in a single node

Horizontal Scaling





Horizontal Scaling Advantages/Limitations

- No single Point of Failure
- You are not limited to the capacity of a single server
- No limitations to scaling and concurrency
- No downtime or interruptions during maintenance or migrations or any application-level changes
- Scaling is easier

Disadvantages : Pricing / Cost control



GCP – Serverless Computing

- Serverless is a cloud-native development model that allows to build and run applications without having to manage servers
- There are still servers in serverless, but they are abstracted away from app development
- Serverless applications are like any other applications which can be invoked via :
 - Events
 - Endpoints/URLs
 - Internally from other cloud components
- Servers that run the application have ephemeral ip addresses.

On-Demand Horizontal Scaling

- Once deployed, serverless apps respond to demand and automatically scale up and down as needed
- As a result, when a serverless function is sitting idle, it doesn't cost anything



GCP – Servers & Containers

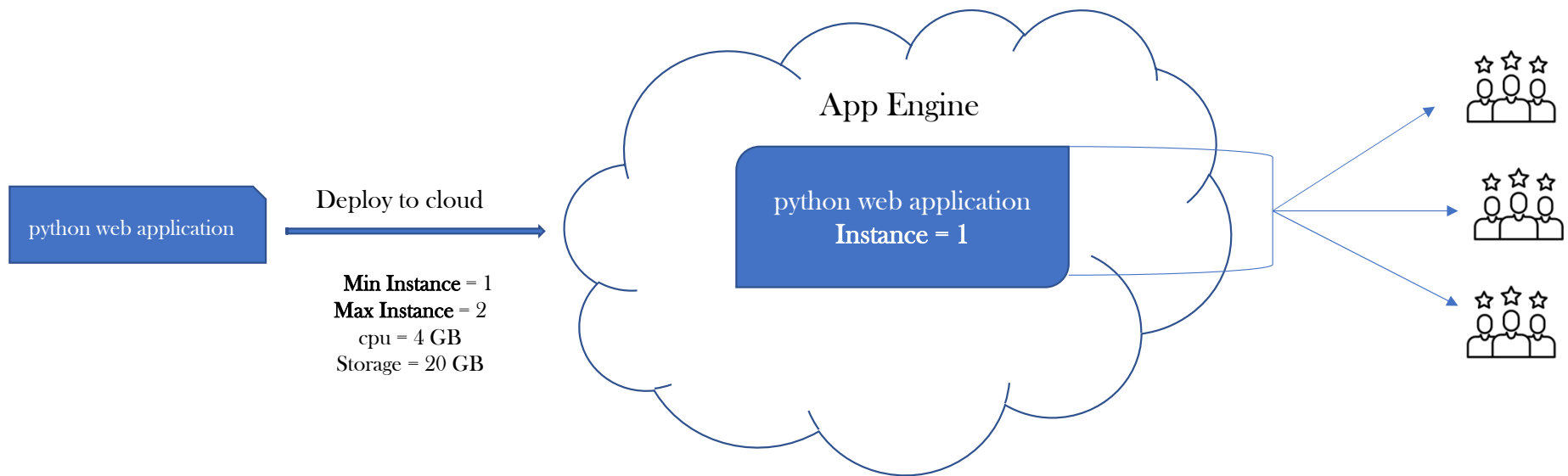
Servers

- You spin up clusters with a pre-defined number of nodes
- Maintenance and management is taken care of by the cloud service provider
- Comes with a Cost to ownership

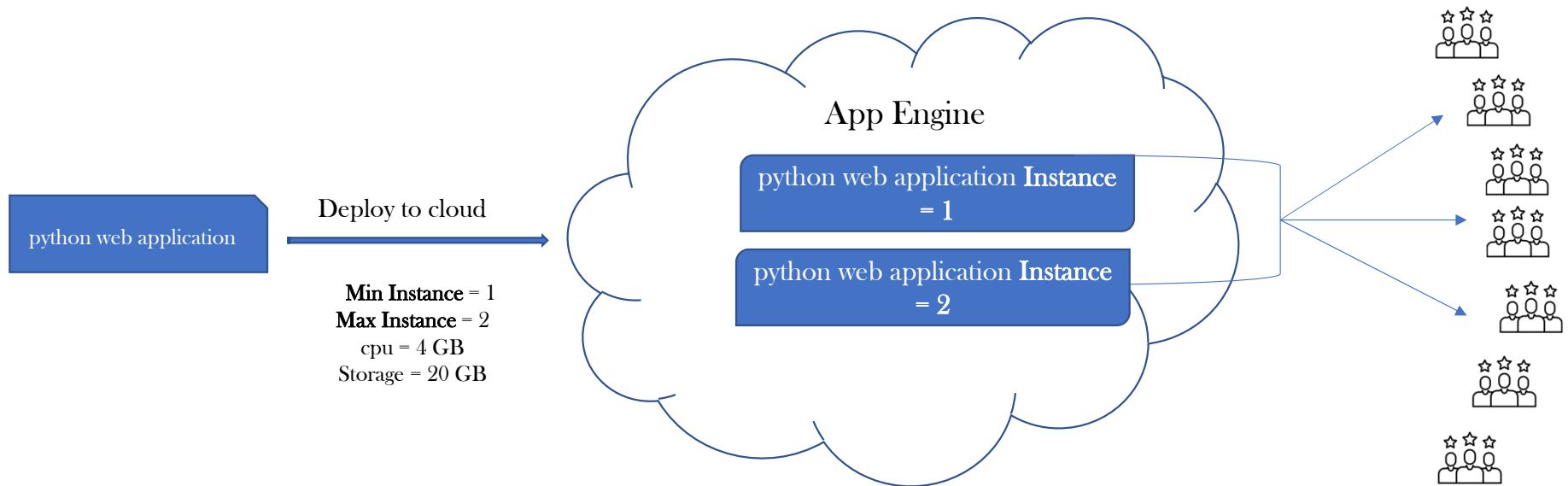
Containers

- Containerized applications are applications running inside a container , For Eg : Docker
- Containerized applications can be serverless or running inside servers
- Scaling applies to containerized applications as well
- To orchestrate and handle the communication between different containers , you need a container orchestration tool like Kubernetes .

GCP - Serverless & On-demand Horizontal Scaling



GCP - Serverless & On-demand Horizontal Scaling





GCP - Cloud Components

Serverless

Data processing

- Cloud Run
- App Engine
- Dataproc Serverless
- Cloud Functions
- Cloud Composer/Airflow

Datawarehouse

- Bigquery
- Hive - Dataproc

Machine Learning

- Vertex AI
- Bigquery ML

RDBMS

CloudSql - Mysql , Postgresql

Servers

Data processing

- Dataproc
- Compute Engine/Virtual Machines
- Cloud Composer/Airflow

Machine Learning

Vertex AI

Containers

Data processing

- Cloud Run
- App Engine
- Dockers - Artifact and container registry

Container Orchestration

GKE - Kubernetes