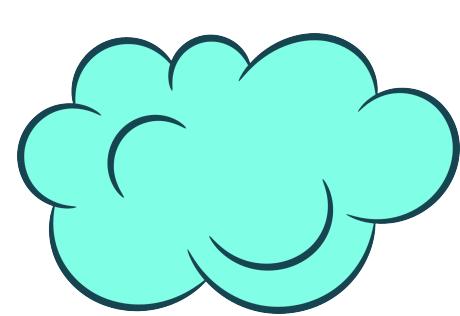


# **Staging in Cloud**

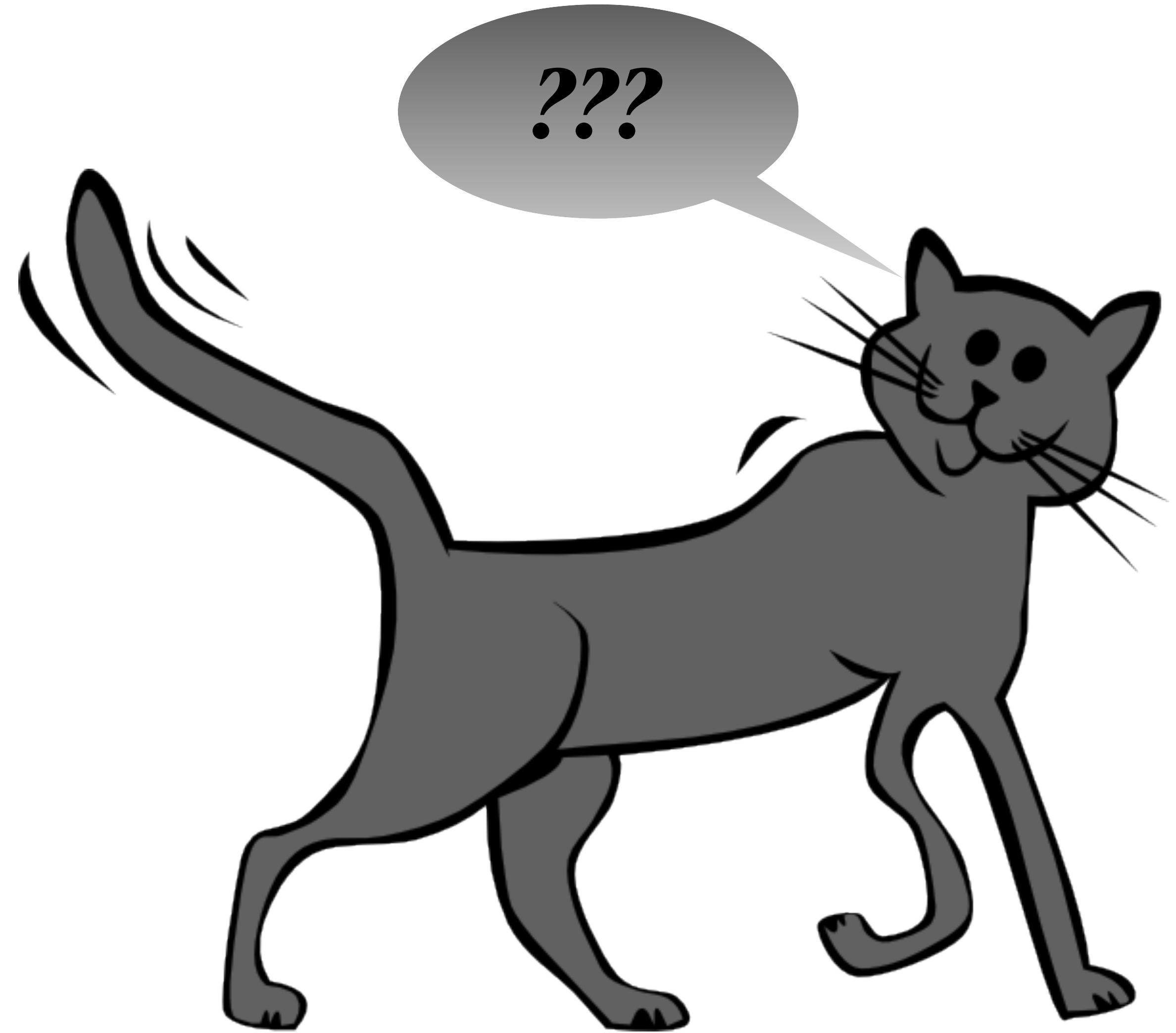
## **Based on AWS**

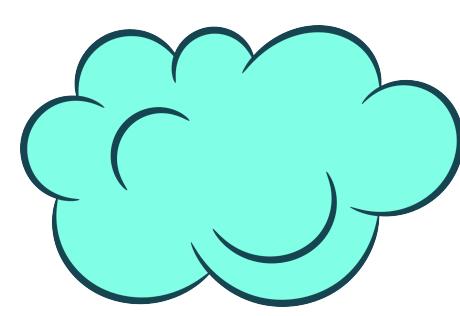
Nikita Kozlov, 2022



# Important Issue

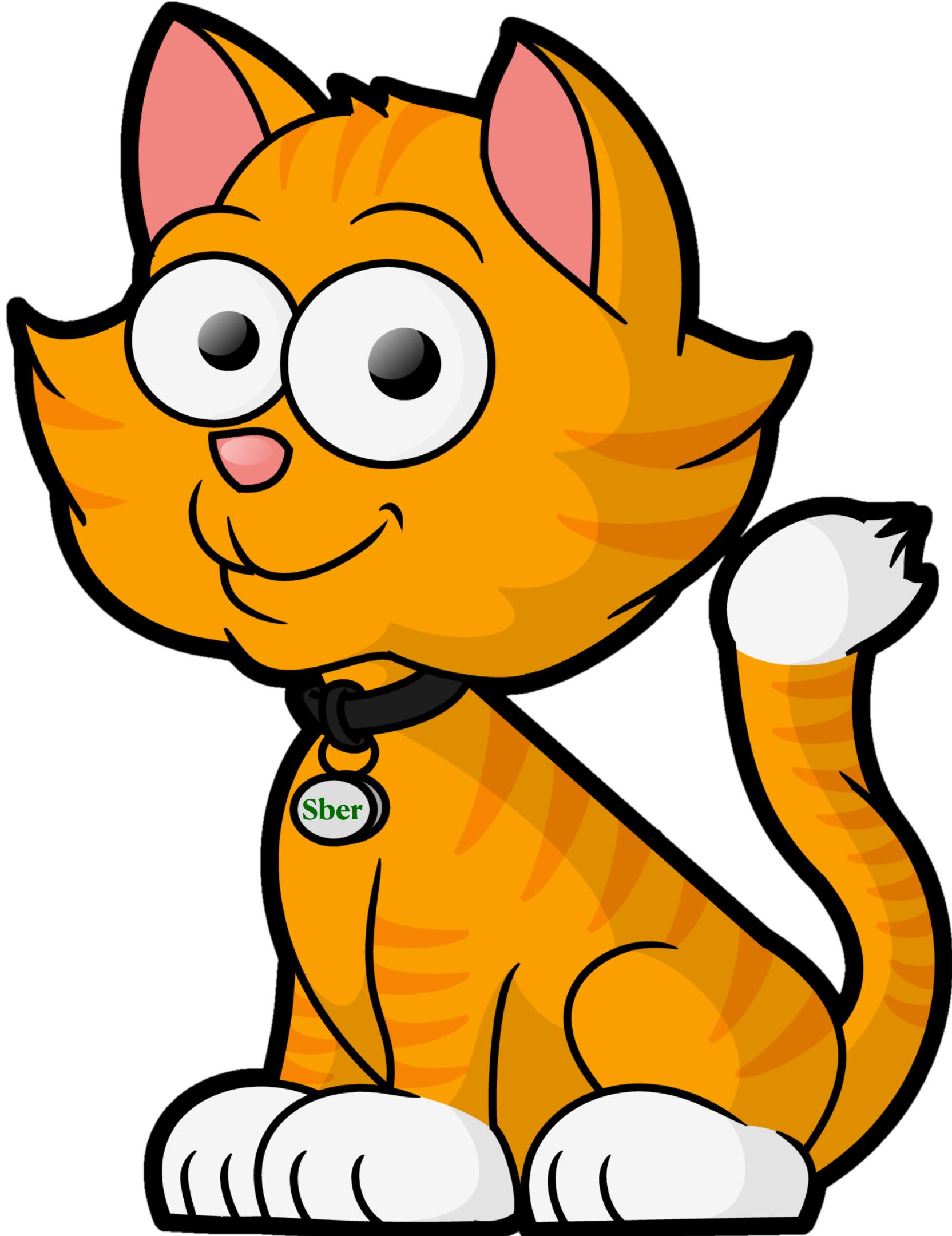
How to organise  
development based in  
AWS?





# How to make it?

**Main approaches**





# Что такое аккаунт AWS?

## Аккаунт

### Консоль управления биллингом и стоимостью

Просмотр информации о текущих расходах и работе аккаунта с разбивкой по сервисам. Доступны также счета за предыдущие месяцы.

### AWS Identity and Access Management

Создание множества пользователей и управление разрешениями для каждого из них из аккаунта AWS.

### Консолидированная оплата

Единый счет для всех ваших аккаунтов AWS с разбивкой по каждому аккаунту. Благодаря рассмотрению аккаунтов как единого целого вы быстрее достигнете уровня, на котором полагаются скидки от объемов использованных услуг.

### История DevPay

Просмотр отчетов о доходах и расходах ваших продуктов Amazon DevPay. Управление продуктами Amazon DevPay

### Управление аккаунтом

Просмотр сервисов, на которые вы подписаны, добавление новых сервисов, аннулирование подписки на сервисы.

### Метод оплаты

Просмотр и редактирование текущего метода оплаты, добавление новых методов оплаты.

### Личные данные

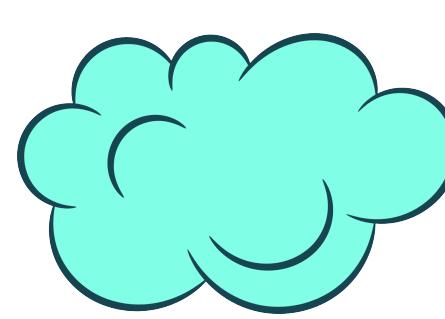
Просмотр и редактирование ваших контактных данных, таких как адрес и номер телефона. Изменение настроек подписки на рассылки.

### Данные, подтверждающие права доступа

В Amazon Web Services используются идентификаторы доступа для аутентификации запросов к сервисам AWS и идентификации отправителя запроса. Доступны идентификаторы трех типов: (1) идентификаторы ключей доступа AWS, (2) сертификаты X.509 и (3) пары ключей.

### Отчеты об использовании

Загрузка отчетов об использовании для каждого сервиса, на который вы подписаны. В отчетах можно настроить отображение типов использования и операций сервиса, задать отчетный период и многое другое.



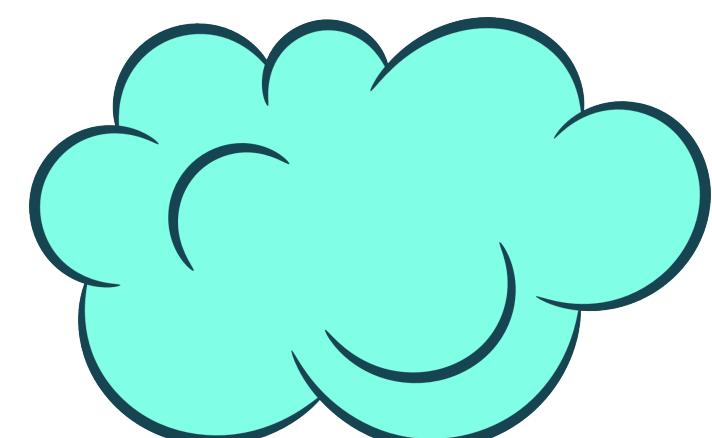
# Approaches

## Single Account

- One account for all stages
- Implemented by Resource Groups and staging feature into AWS resources
- Low level of resource isolation (quotas, permissions)

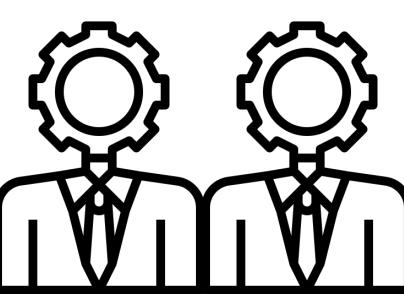
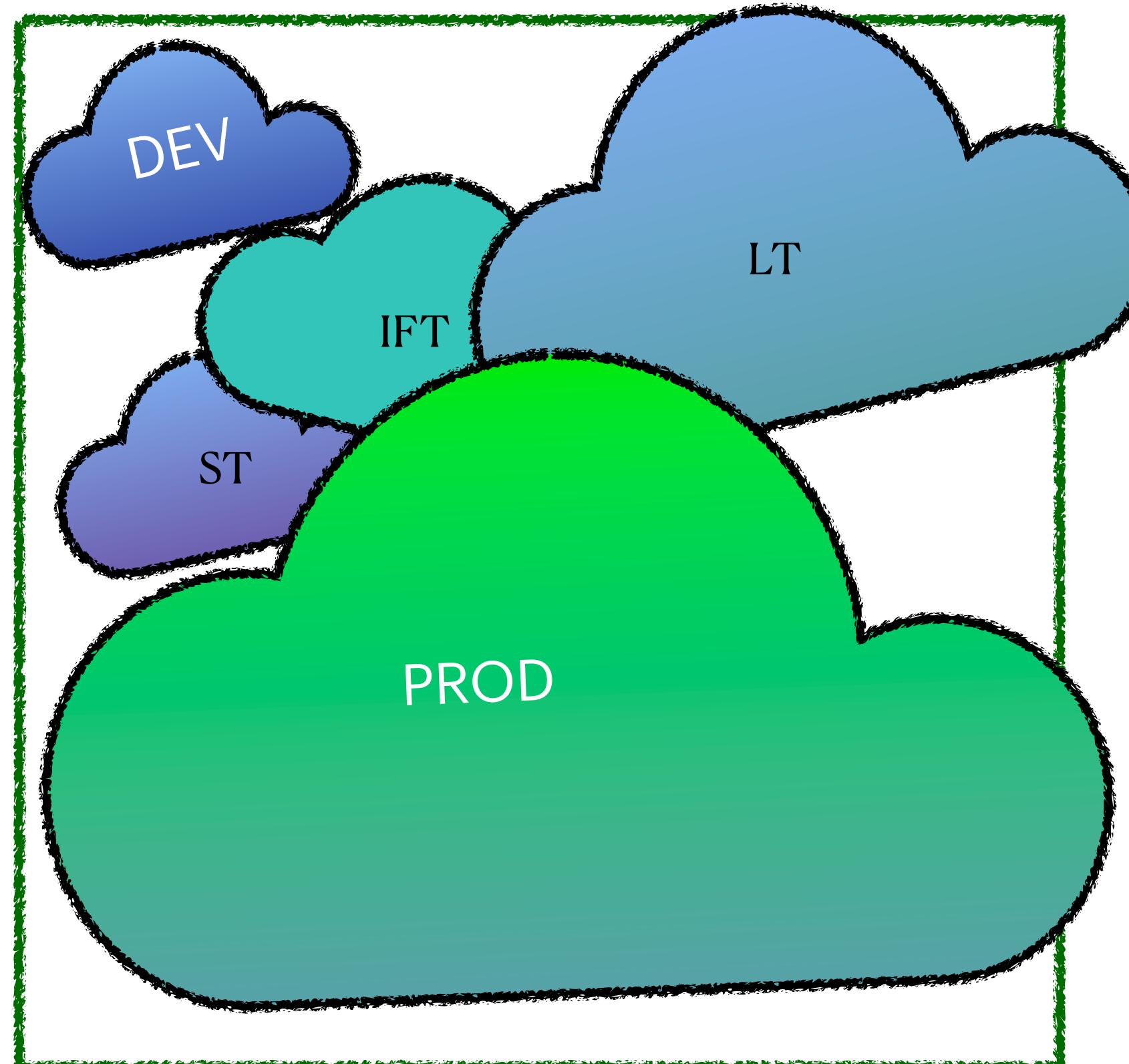
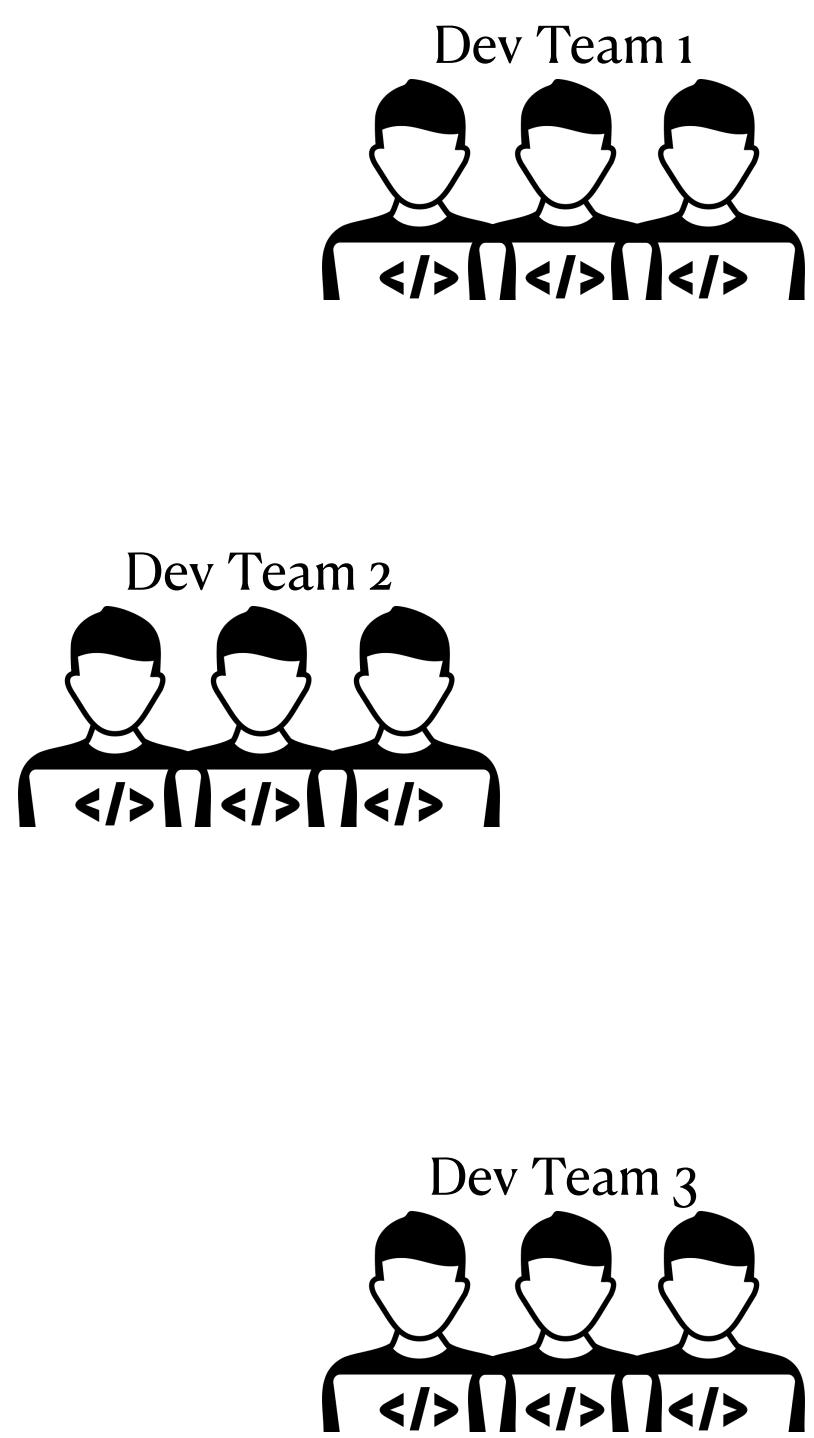
## Account per Stage

- Using separate account for each stage
- Each developer or team have their own accounts
- Each stage (ST, IFT, UAT, LT, PROM) has separate account
- High level of isolation resources
- Permissions can be fine grained
- Accounts can be combined into an Organisation  
it allows to centralise rule them and have one payment method
- Different SLA



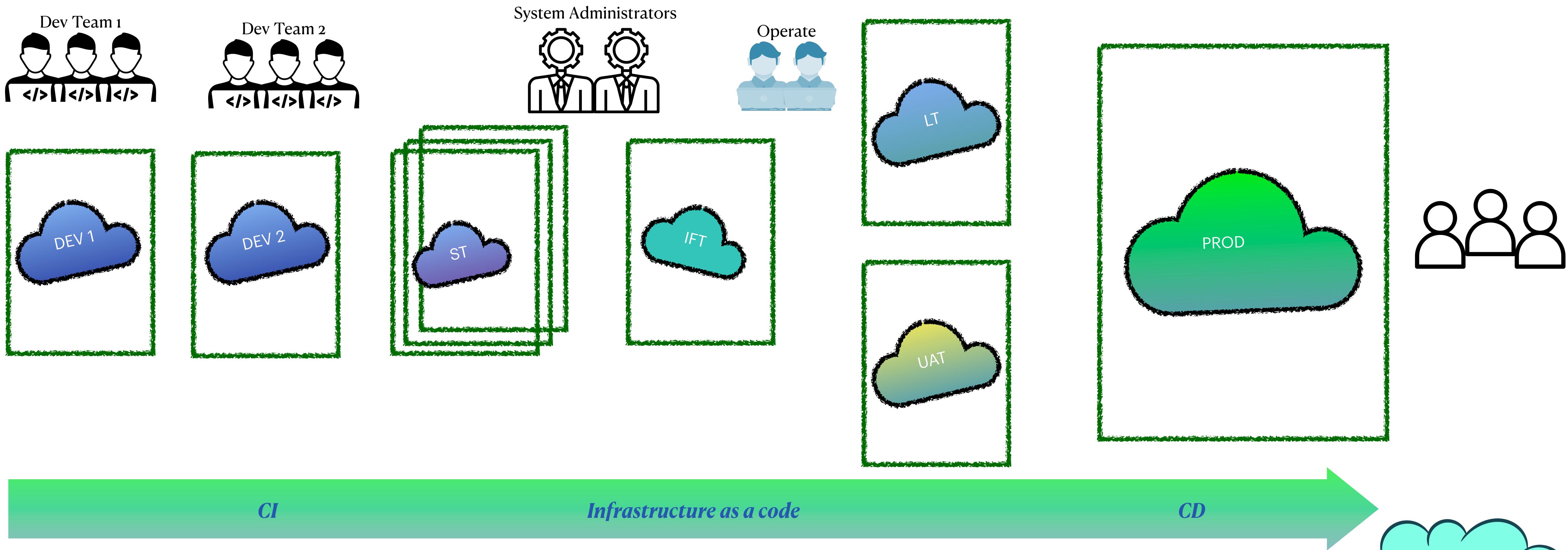
# Approaches

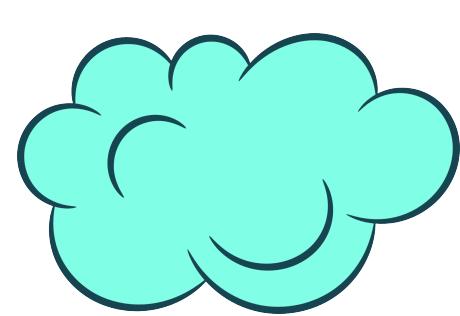
## Single Account



# Approaches

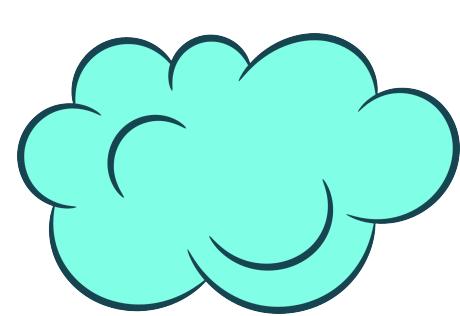
## Account per Stage





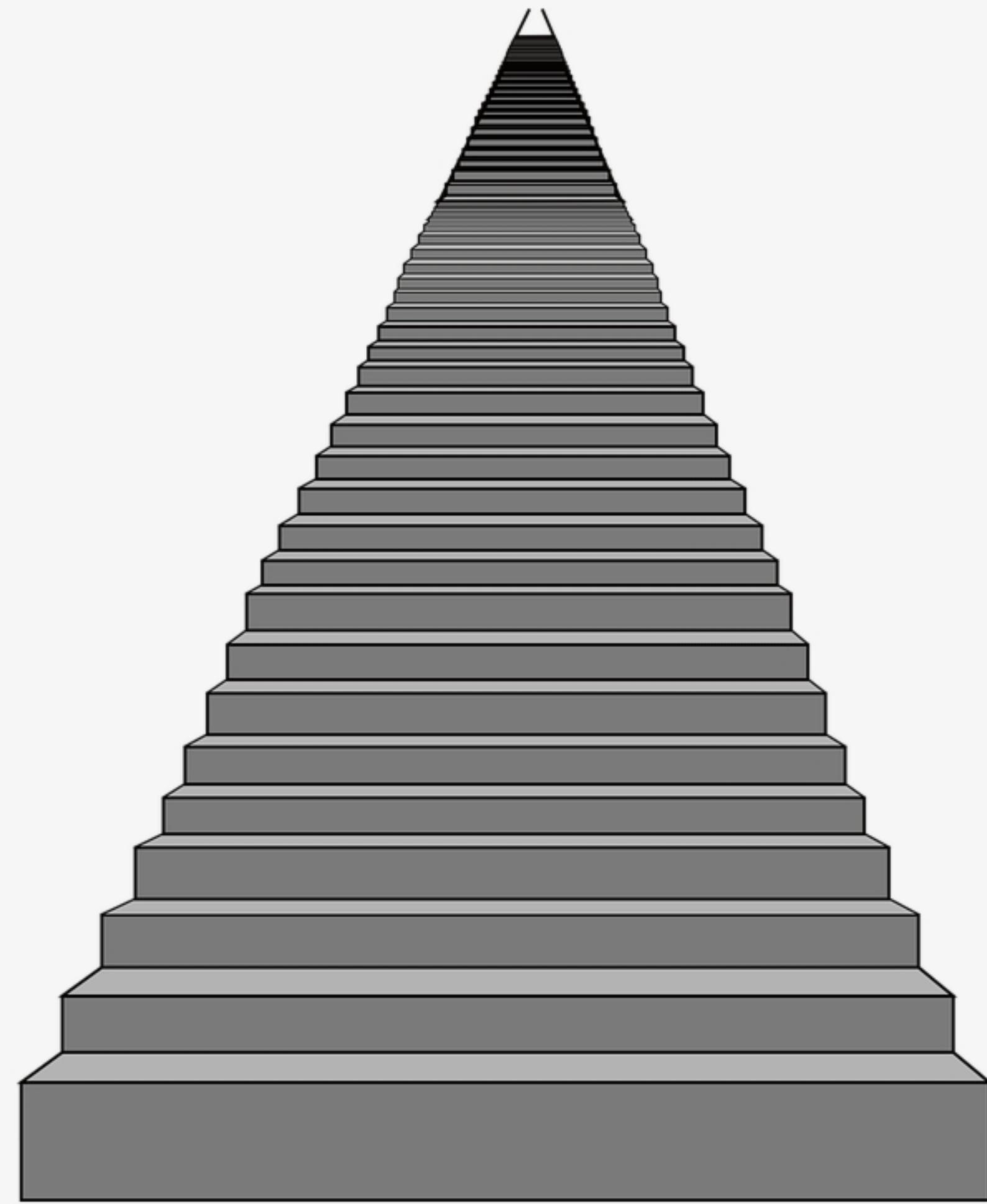
# Account per Stage

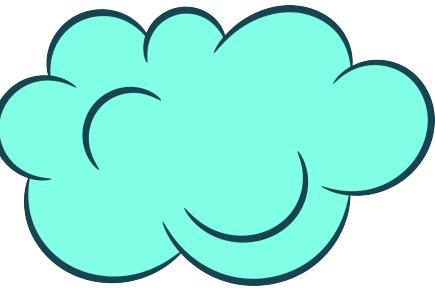
**Right choice**



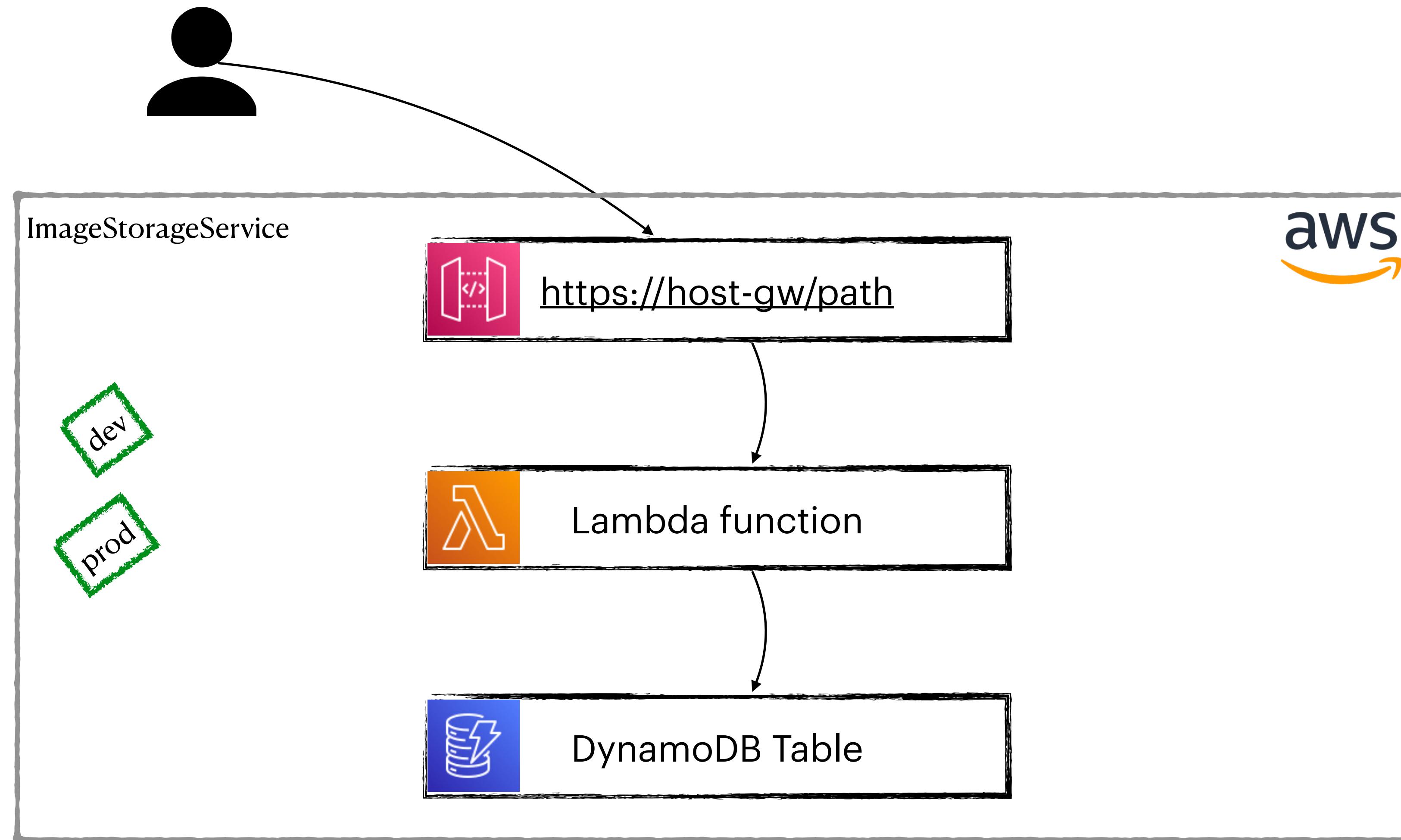
# Staging in Lambda

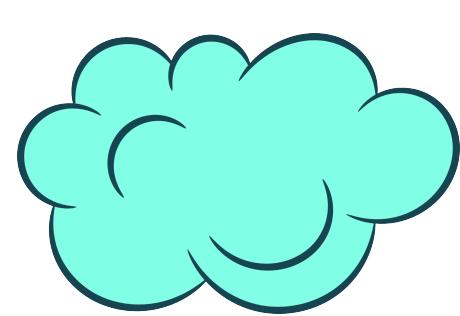
## Example





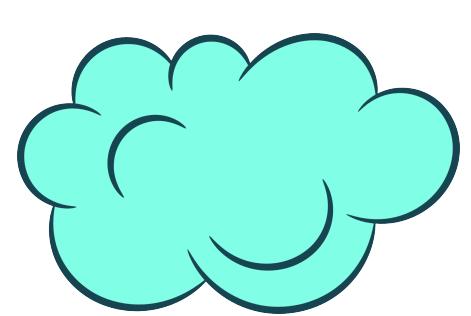
# Staging Application





# Staging Tools

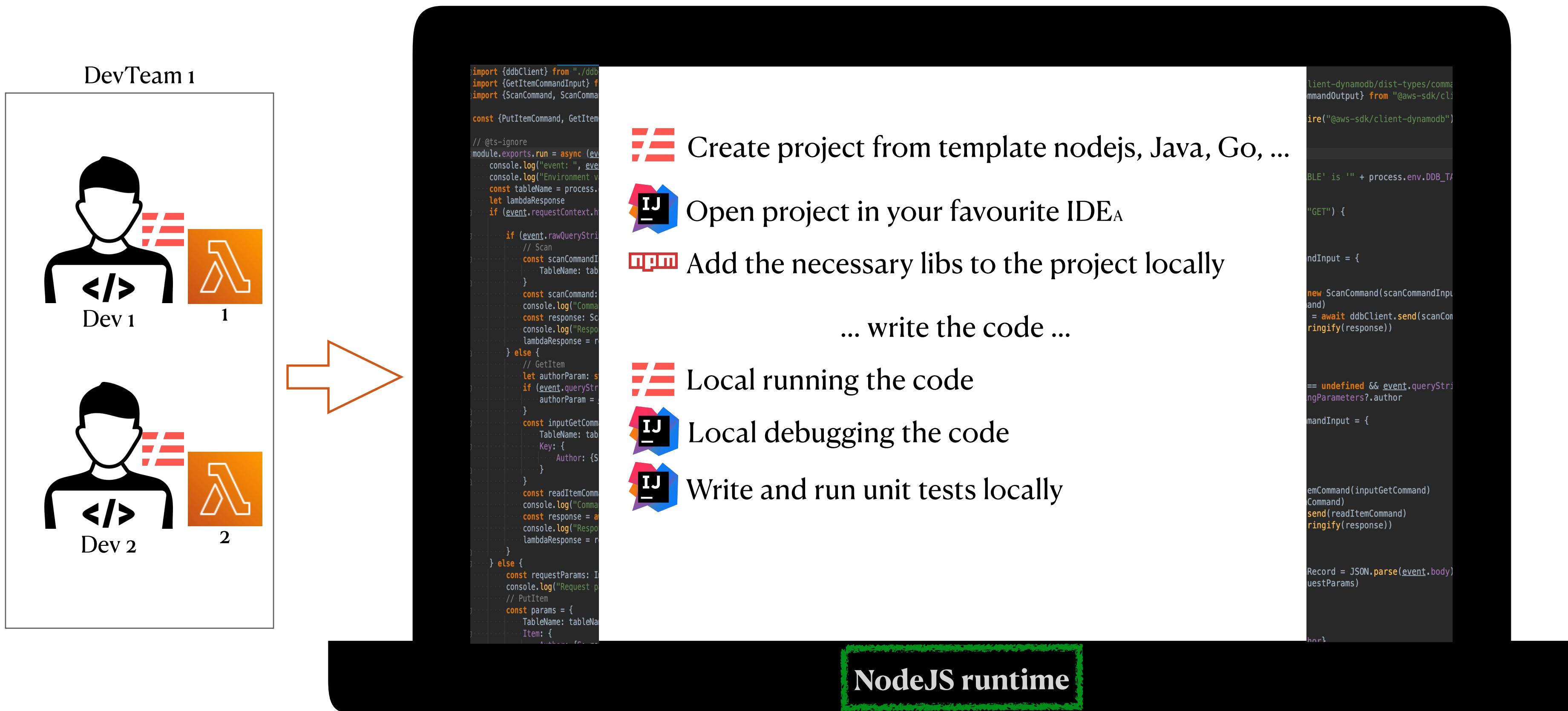
- AWS CLI
- Serverless Framework

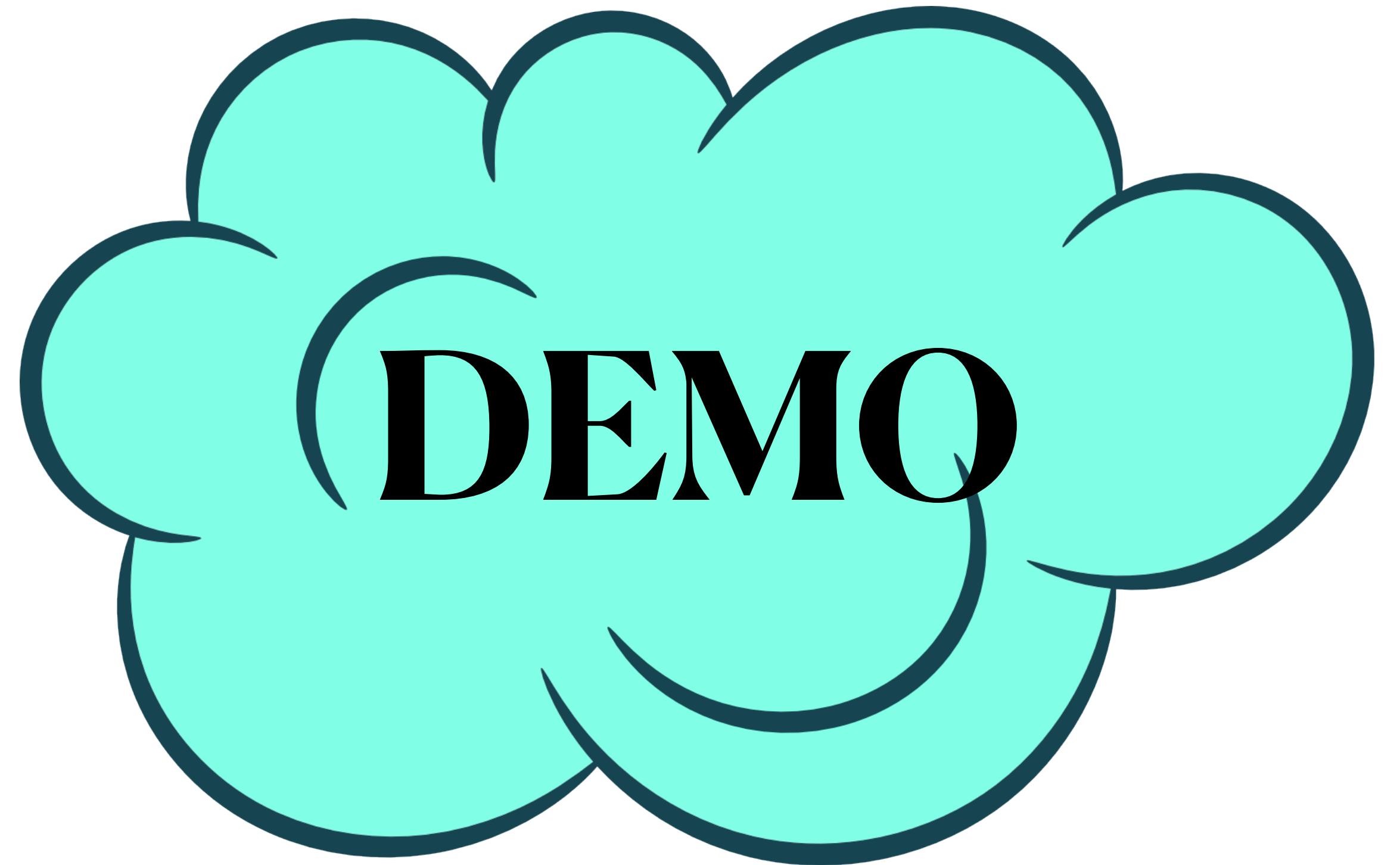


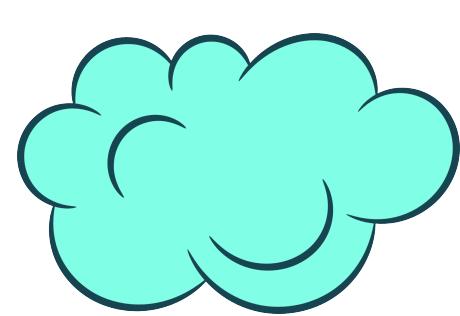
# Staging

## Stage DEV

# Local Dev

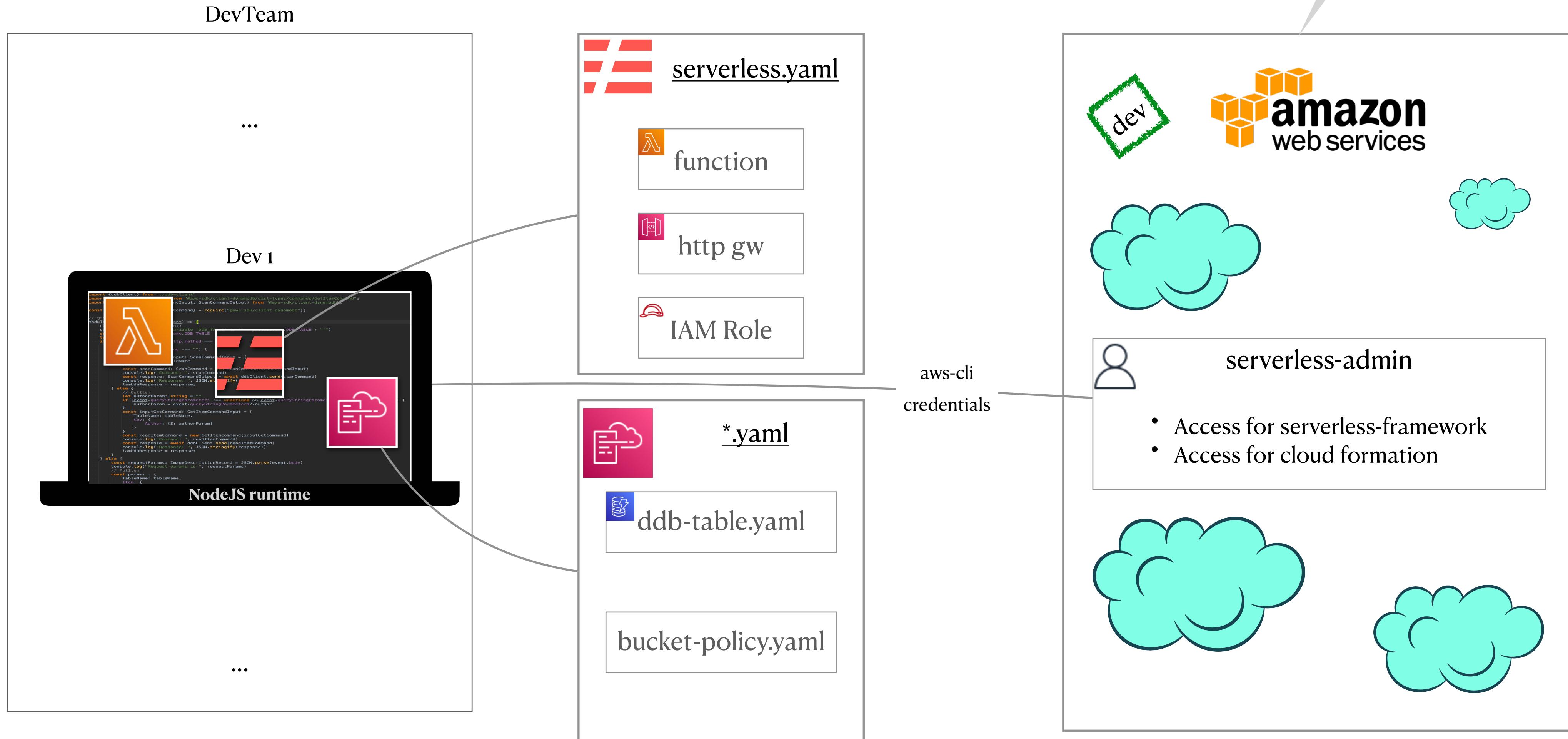






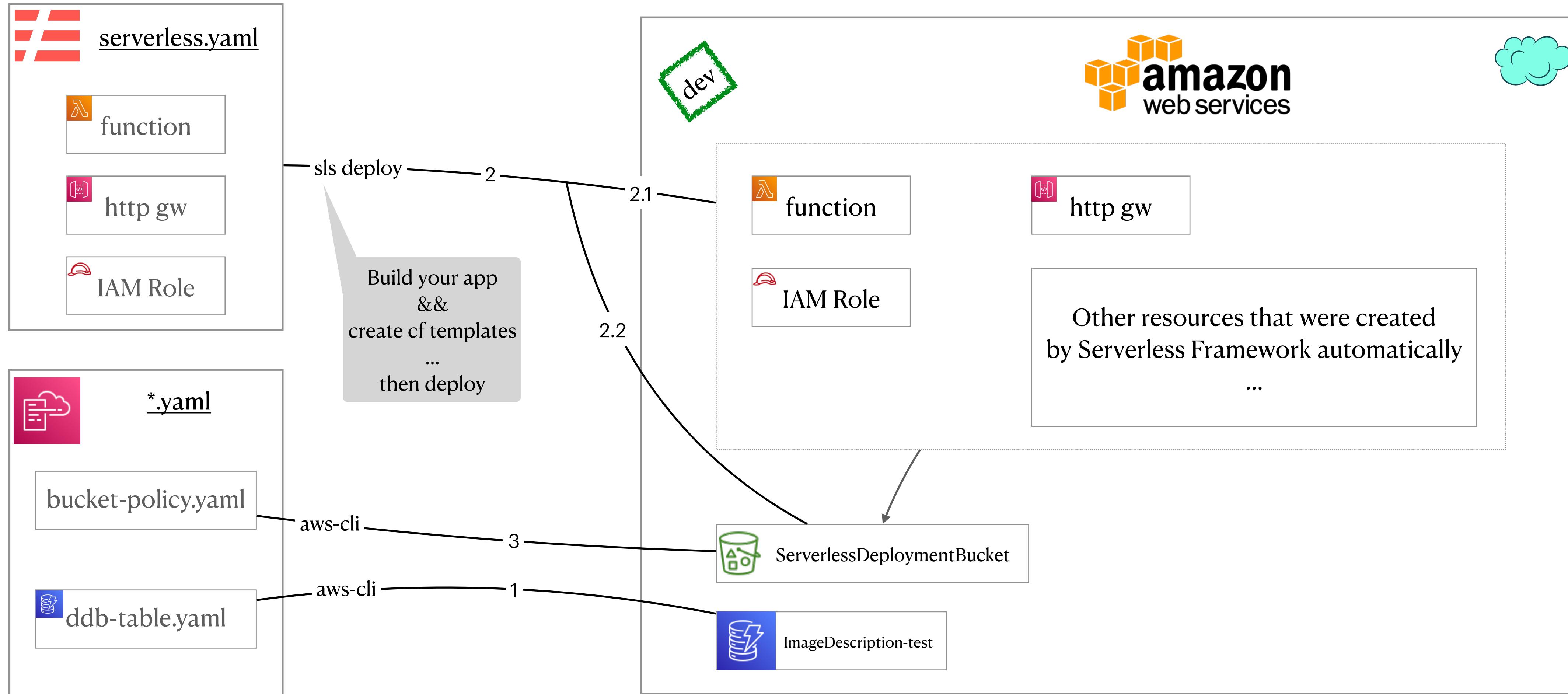
# Staging

## Stage DEV



# Staging

## Stage DEV



# Staging

## Stage DEV → PROD

