

TWITTER SCHEDULER

TRABAJO FIN DE MASTER
DICIEMBRE 2021



AUTOR: DAVID ROJO ANTONA

TUTOR: FRANCISCO GORTÁZAR BELLAS



 @davidrojoa

QUIÉN SOY



spring®

JUnit 5



MariaDB

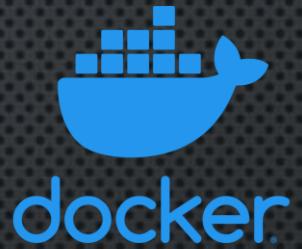


kafka

BACKEND
DEVELOPER



Grafana



OPENSHIFT



TWITTER SCHEDULER

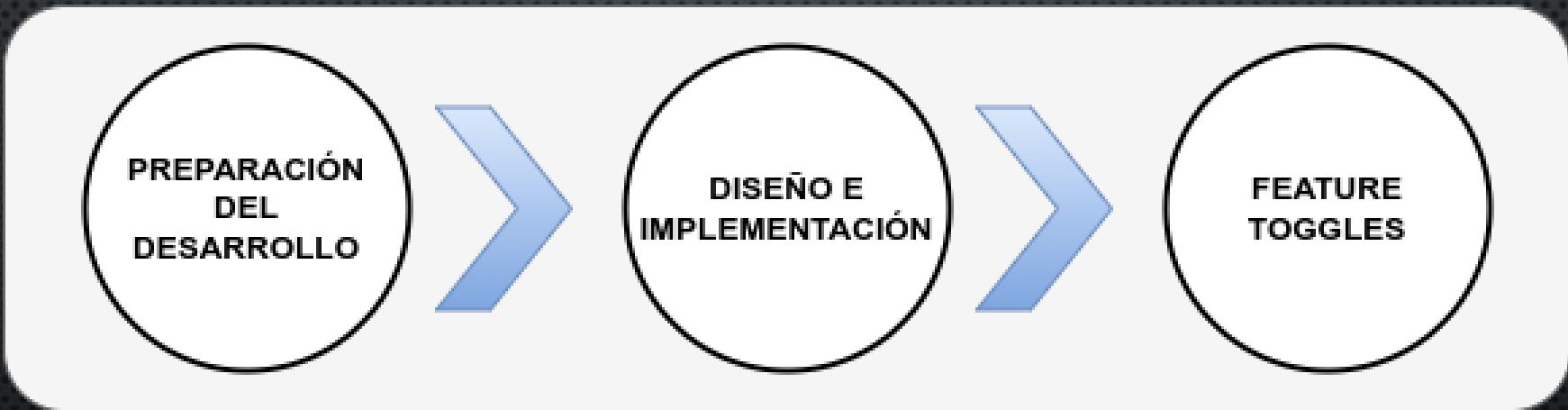
OBJETIVO:

APLICAR LAS TÉCNICAS DE **FEATURE TOGGLS** Y
BRANCH BY ABSTRACTION EN UN PROYECTO
JAVA CON **TRUNK-BASED DEVELOPMENT**.

EL PROYECTO ESTÁ DESARROLLADO SIGUIENDO
UNA ARQUITECTURA **HEXAGONAL** Y SE HA
APLICADO **DOMAIN DRIVEN DESIGN** (DDD).

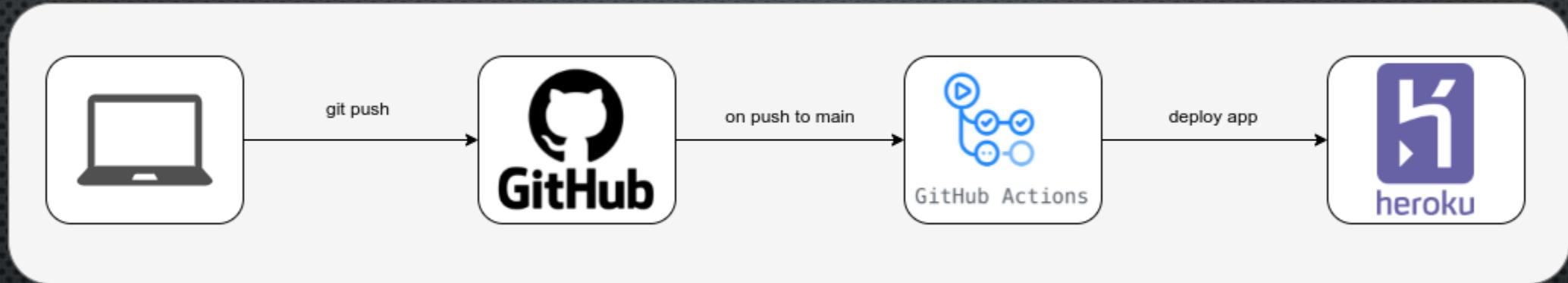


FASES DE DESARROLLO

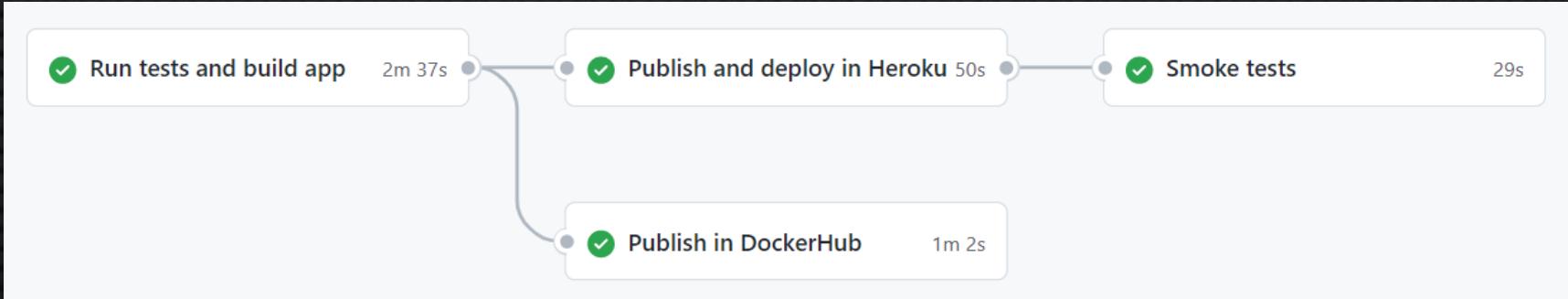




CI/CD

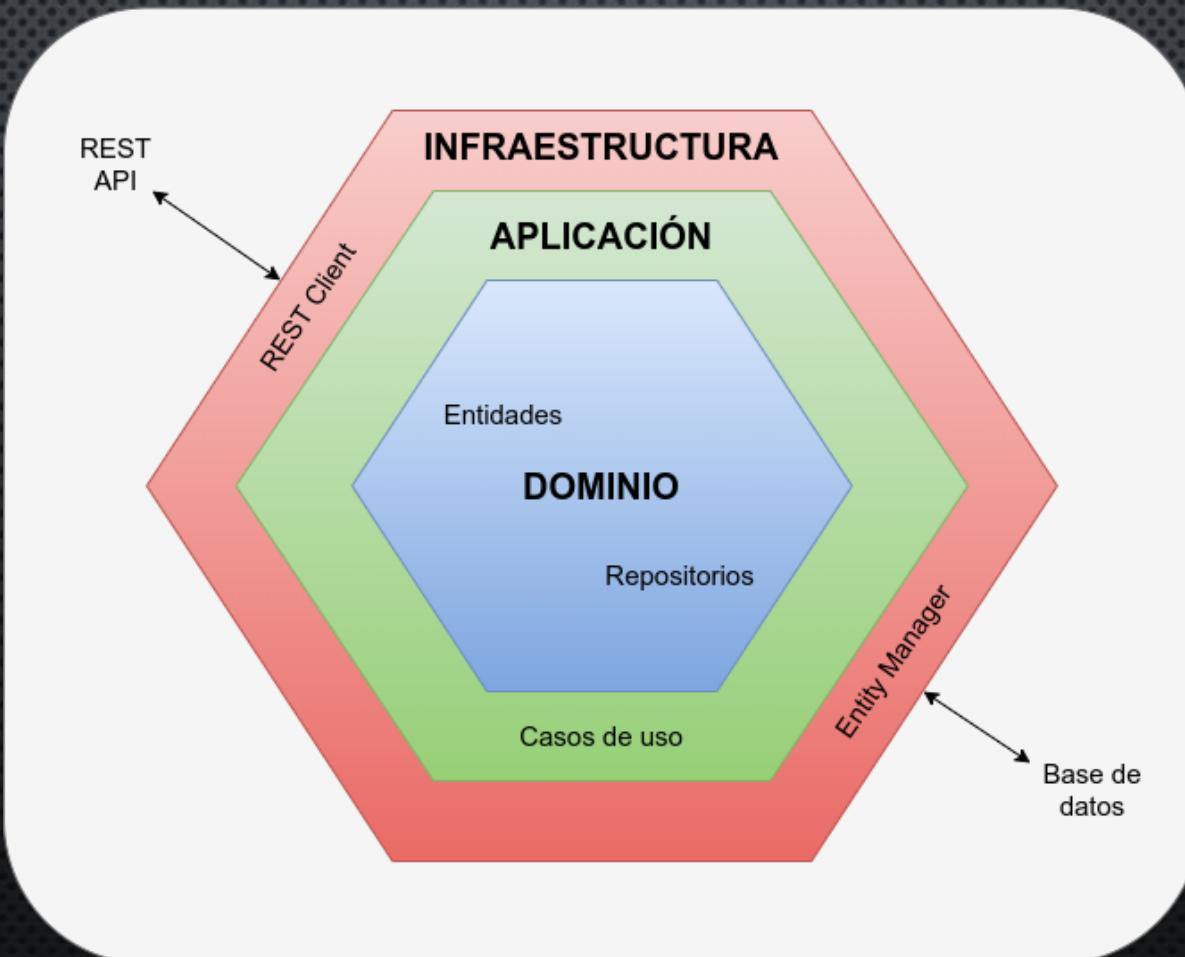


WORKFLOW



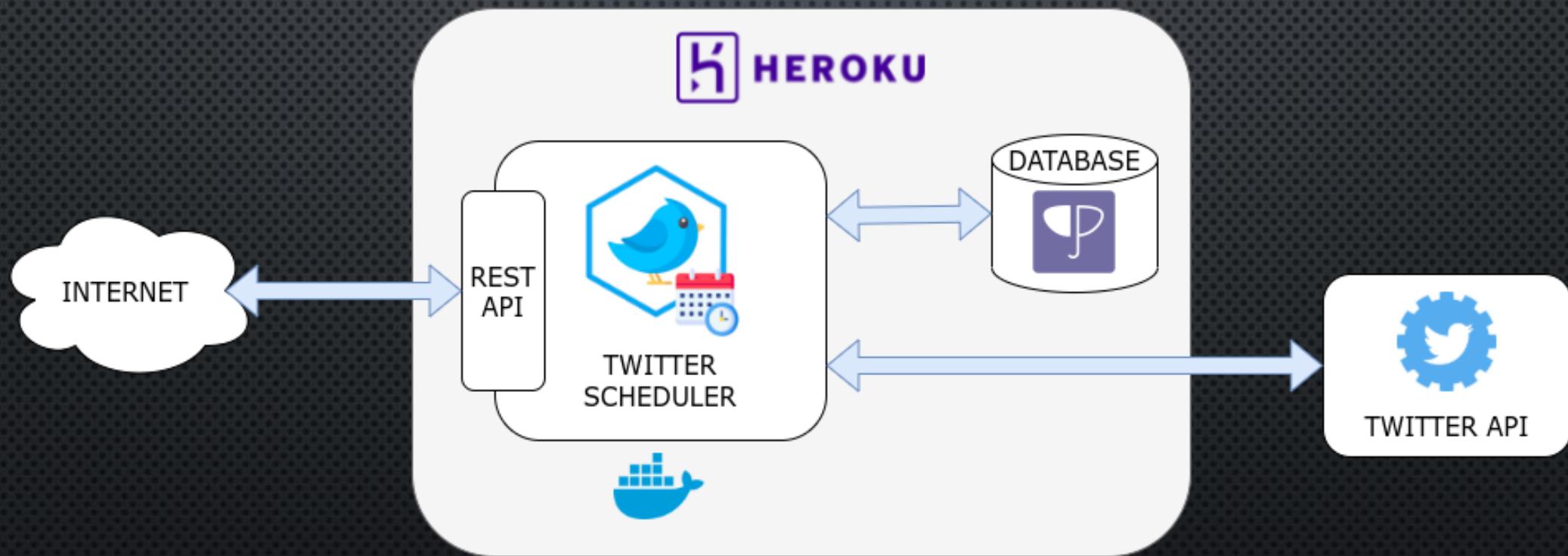


ARQUITECTURA





PRODUCCIÓN





DOMINIO

PendingTweet

- + pendingTweetId: Long
- + message: String
- + publicationDate: Instant
- + createdAt: Instant

Tweet

- + tweetId: Long
- + message: String
- + url: String
- + requestedPublicationTime: Instant
- + publishedAt: Instant
- + createdAt: Instant



REST API

index			
GET	/	Get application welcome message	▼ 🔒
pending			
GET	/api/pending	Find all pending tweets available in app	▼ 🔒
POST	/api/pending	Add a new pending tweet	▼ 🔒
GET	/api/pending/{id}	Get the information of a specific pending tweet and its images	▼ 🔒
DELETE	/api/pending/{id}	Deletes a pending tweet and its images	▼ 🔒
scheduler			
GET	/api/scheduler/info	Get scheduler info	▼ 🔒
tweets			
GET	/api/tweets	Find all tweets available in the account	▼ 🔒
GET	/api/tweets/{id}	Get the information of a specific tweet and its images	▼ 🔒



FEATURE TOGGLES & BRANCH BY ABSTRACTION



TWEETS BAJO DEMANDA

OBJETIVO:

INCLUIR LA FUNCIONALIDAD QUE PERMITA LA PUBLICACIÓN INMEDIATA DE UN TWEET PENDIENTE DE PUBLICAR YA EXISTENTE



TWEETS BAJO DEMANDA

PendingTweet

- + pendingTweetId: Long
- + message: String
- + publicationDate: Instant
- + createdAt: Instant

Tweet

- + tweetId: Long
- + message: String
- + url: String
- + requestedPublicationTime: Instant
- + publishedAt: Instant
- + createdAt: Instant
- + publicationType: PublicationType



TWEETS BAJO DEMANDA

A screenshot showing two panels of developer activity. The left panel is a GitHub commit history for a repository, dated November 24, 2021. It lists 14 commits by user "david-rojo" over a 7-day period, all marked with a green checkmark. The right panel is a Heroku deployment history for the same period, showing 14 deployments by user "davidrojotech@gmail.com", each with a "Roll back to here" link.

Commits on Nov 24, 2021

- delete PUBLISH_ON_DEMAND feature toggle
david-rojo committed 7 days ago ✓
- show publish on demand endpoint in openapi
david-rojo committed 7 days ago ✓
- enable feature toggle PUBLISH_ON_DEMAND
david-rojo committed 7 days ago ✓
- add onDemand publish controller unitary tests
david-rojo committed 7 days ago ✓
- add publishOnDemand controller implementation
david-rojo committed 7 days ago ✓
- add PublishPendingTweetOnDemand service implementation
david-rojo committed 7 days ago ✓
- add PublishPendingTweetOnDemand useCase definition
david-rojo committed 7 days ago ✓
- add publication type to domain and adapt current implementation
david-rojo committed 7 days ago ✓
- add publicationType to Tweet jpa entity
david-rojo committed 7 days ago ✓
- add rest endpoint, hide from openapi, return not allowed
david-rojo committed 7 days ago ✓
- add publication type column to db
david-rojo committed 7 days ago ✓
- add new feature toggle PUBLISH_ON_DEMAND
david-rojo committed 7 days ago ✓

GitHub

heroku

- davidrojotech@gmail.com: Deployed web (1bf0cc3596a9) Nov 24 at 3:21 AM · v76 · [Roll back to here](#)
- davidrojotech@gmail.com: Deployed web (3f623d3c8826) Nov 24 at 2:52 AM · v75 · [Roll back to here](#)
- davidrojotech@gmail.com: Deployed web (a13f976608f4) Nov 24 at 2:44 AM · v74 · [Roll back to here](#)
- davidrojotech@gmail.com: Deployed web (2cbd32085b76) Nov 24 at 2:40 AM · v73 · [Roll back to here](#)
- davidrojotech@gmail.com: Deployed web (3a25df45a114) Nov 24 at 2:35 AM · v72 · [Roll back to here](#)
- davidrojotech@gmail.com: Deployed web (2e7af6ad4ee2) Nov 24 at 2:30 AM · v71 · [Roll back to here](#)
- davidrojotech@gmail.com: Deployed web (87b1744df0f9) Nov 24 at 2:20 AM · v70 · [Roll back to here](#)
- davidrojotech@gmail.com: Deployed web (a37a5efbc4f5) Nov 24 at 2:17 AM · v69 · [Roll back to here](#)
- davidrojotech@gmail.com: Deployed web (065d0874dcff) Nov 24 at 1:49 AM · v68 · [Roll back to here](#)
- davidrojotech@gmail.com: Deployed web (a814e9fc6f88) Nov 24 at 1:39 AM · v67 · [Roll back to here](#)
- davidrojotech@gmail.com: Deployed web (ac3986410c8a) Nov 24 at 1:17 AM · v66 · [Roll back to here](#)
- davidrojotech@gmail.com: Deployed web (b7ab726c6299) Nov 24 at 1:15 AM · v65 · [Roll back to here](#)



TWEETS CON IMÁGENES

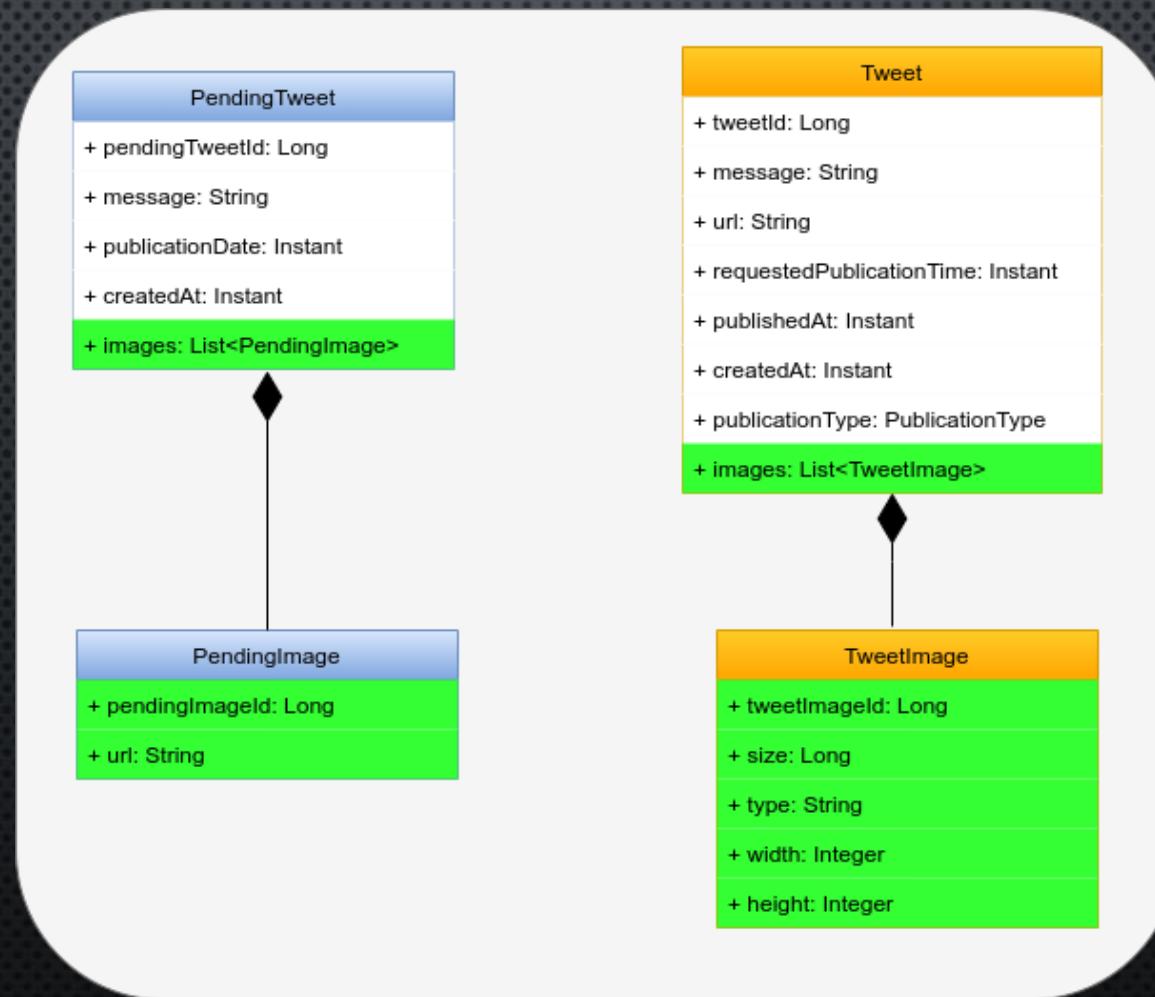
OBJETIVO:

INCLUIR LA FUNCIONALIDAD PARA QUE LOS TWEETS A PUBLICAR PUEDAN TENER COMO CONTENIDO UNA O MAS IMÁGENES

DE CADA IMAGEN SE PROPORCIONARÁ LA URL EN LA QUE ESTÁ DISPONIBLE



TWEETS CON IMÁGENES





DEMO



CONCLUSIONES

- **CI/CD FIRST**
- **TESTS:**
 - UNITARIOS, INTEGRACIÓN, SMOKE TESTS
 - IMPORTA LA CALIDAD QUE TENGAN
- TRABAJAR EN **MAIN** → ENTORNO TEST



CONCLUSIONES

- **TRUNK-BASED DEVELOPMENT:**
 - NECESARIA INFRAESTRUCTURA DE DESARROLLO
 - DOMINIO DE TÉCNICAS AVANZADAS:
 - DISEÑO (FEATURE TOGGLS, BRANCH BY ABSTRACTION)
 - CONTROL DE VERSIONES (CHERRY PICK)



CONCLUSIONES

- **ARQUITECTURA HEXAGONAL Y DDD:**
 - COMPENSA EN GRANDES PROYECTOS
 - SI ESTÁ BIEN DISEÑADO:
 - AYUDA A ESCALAR
 - FACILITA LOS CAMBIOS



CONCLUSIONES

- CAPA GRATUITA HEROKU:
 - LÍMITE DE MEMORIA: 512 MB
 - APP HIBERNA SI NO TIENE TRÁFICO WEB EN 30 MIN
 - LÍMITE CONEXIONES SIMULTÁNEAS A BASE DE DATOS POSTGRES: 20

INTERESA EN PRUEBAS DE CONCEPTO, APPS QUE NECESITEN POCOS RECURSOS Y NO CRÍTICAS



TRABAJOS FUTUROS

- IMPLEMENTACIÓN EN NODE
- NUEVAS APIs: GRAPHQL, GRPC, COLAS
- IMPLEMENTACIÓN CQRS
- ALMACENAMIENTO DE RECURSOS EN LA NUBE
(AMAZON S3, GOOGLE CLOUD STORAGE)



PREGUNTAS





¡GRACIAS!