Full-stack web application map

DJANGO BACK-END

Django / Python is the heart of the whole application. It will include serialized data models for DRF with relevant view functions / routes. It is also the source for machine learning integration, deployment and content.

DJANGO SERVE APP

After running build scripts, the serve app will funnel all build dist to the browser. Django will serve the entire app as one static application.

REDIS

BACK-END DJANGO FRAMEWORK

PYTHON / DJANGO

DRF

«Interface back-end»
DJANGO

«Machine Learning»
APP

Endpoints
REST API

DJANGO SERVE APP

SCSS   CSS

JAVASCRIPT

STATIC ASSETS

«interface»
djserve

MEDIA

«Cloud Storage»
Cloudinary

PRODUCTION

App will be containerized with Docker and deployed to Heroku.

docker

OR

aws

DATABASES

PostgreSQL

HTTP

PLATFORMS

IOS

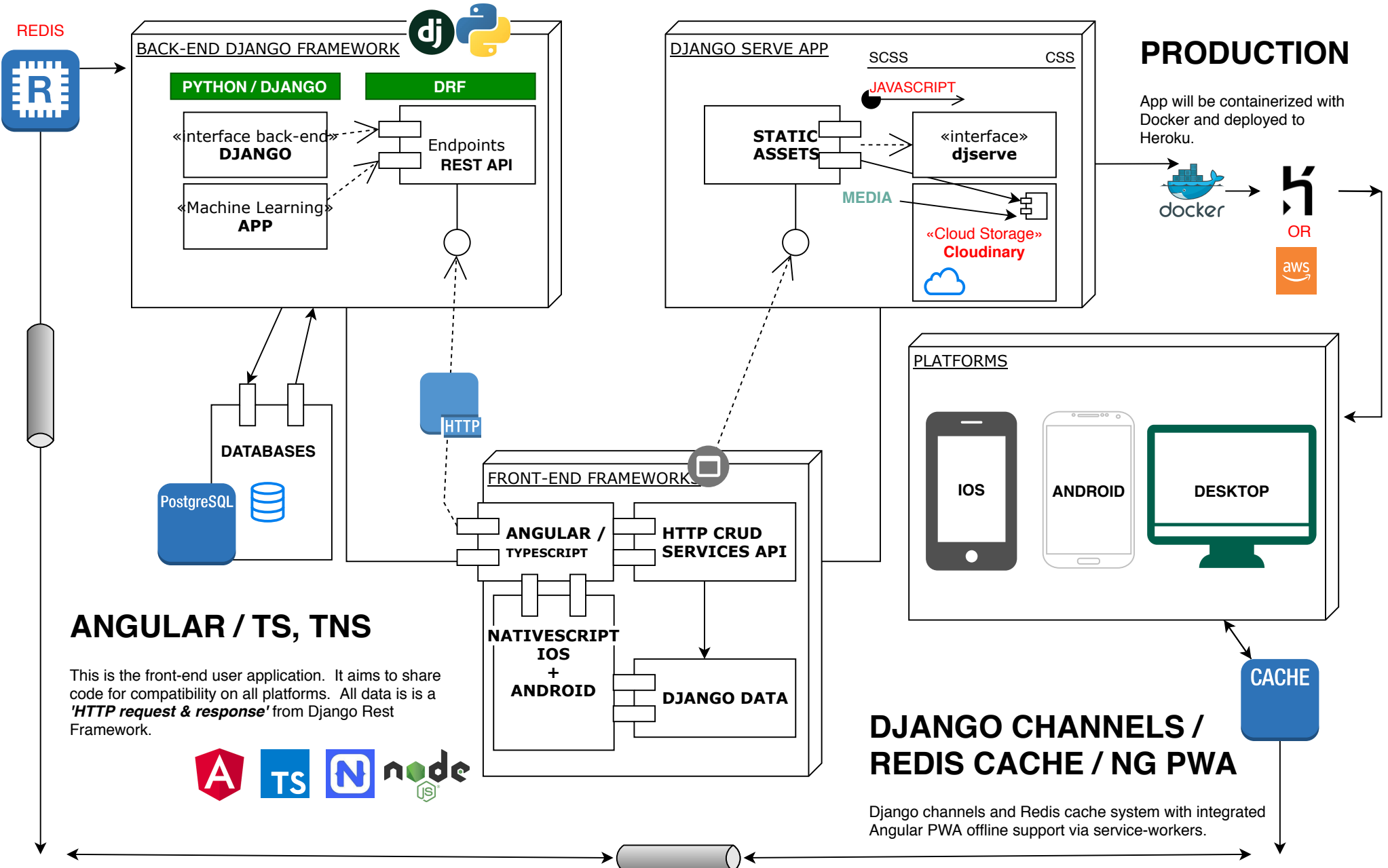**ANDROID**

DESKTOP

ANGULAR / TS, TNS

This is the front-end user application. It aims to share code for compatibility on all platforms. All data is is a **'HTTP request & response'** from Django Rest Framework.

FRONT-END FRAMEWORK

ANGULAR /
TYPESCRIPT

HTTP CRUD
SERVICES API

NATIVESCRIPT
IOS
+
ANDROID

DJANGO DATA

CACHE

DJANGO CHANNELS / REDIS CACHE / NG PWA

Django channels and Redis cache system with integrated Angular PWA offline support via service-workers.

Scaling

AMAZON WEB SERVICES

Instead of HEROKU for deployment and CLOUDINARY for cloud storage.

CELERY

Distribution system to process vast amounts of messages queue tasks.

GOOGLE SERVICES API

Communication with Google Services and their integration to other services.

PAYMENT GATEWAYS

Integrated payment gateways.

Design

WEBFLOW

UX design and animation.

HTML

Complaint to W3C standards.

CSS

High standard use of CSS / SCSS