SOFTWARE DEVELOPMENT METHODOLOGY

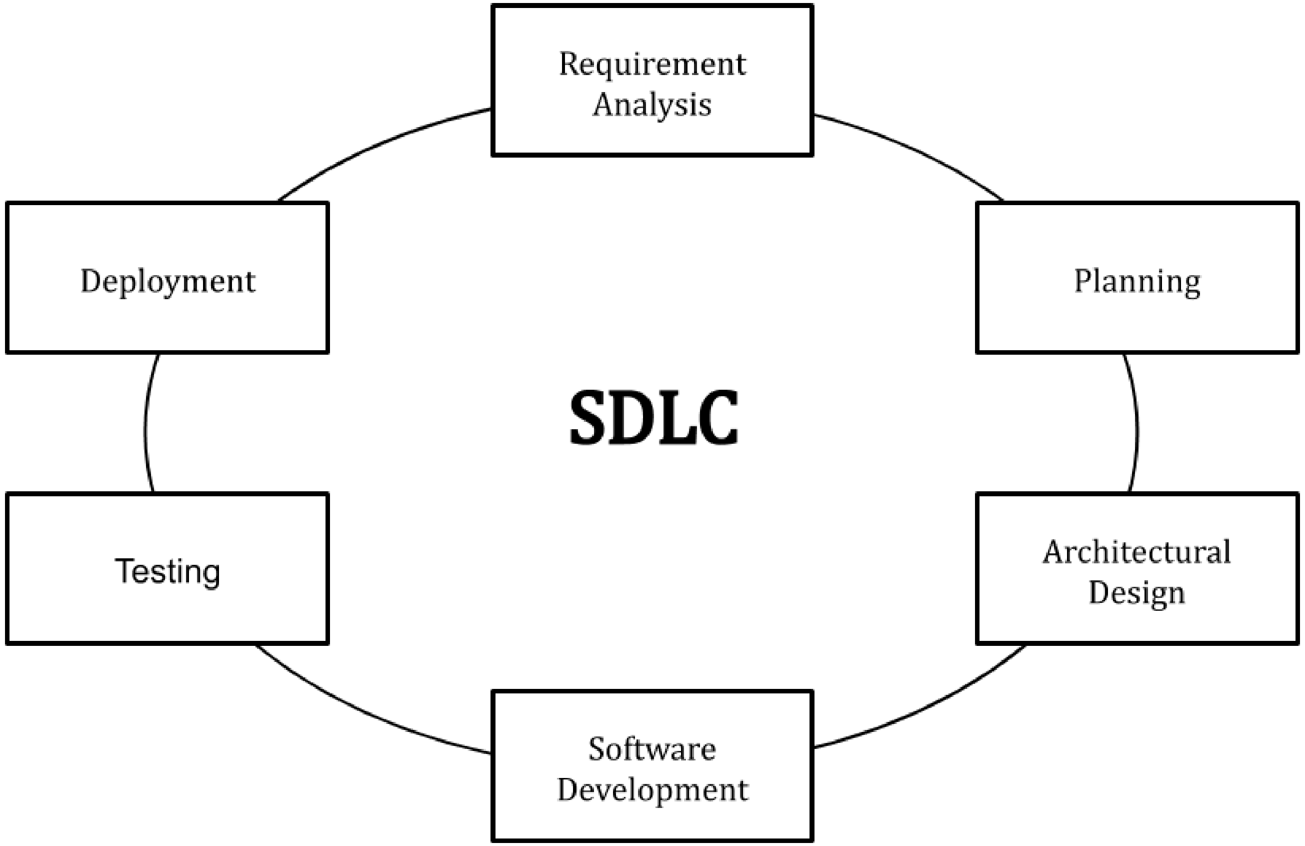
SUKRU CAKMAK

ATEZ SOFTWARE TECHNOLOGIES

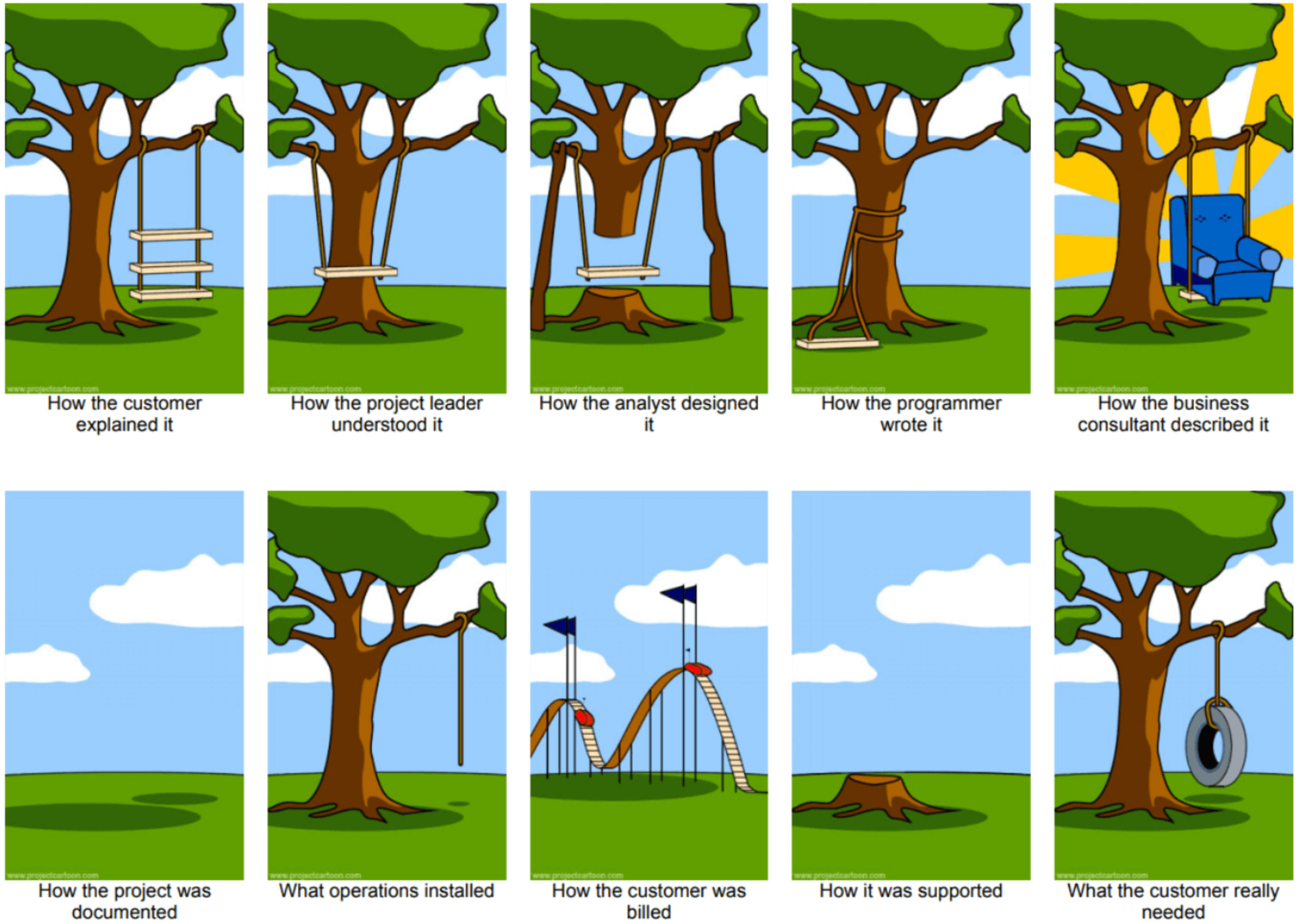
- ATEZ
- Vakıfbank, R&D and Innovation
- Turkcell, Vodafone, Turk Telekom, i2i Systems
- Anadolu Efes
- Lecturer at Kadir Has Univ and Altinbas Univ

- PhD C. Computer Engineering
- MSc. Electrical and Computer Engineering
- MSc. Software Engineering
- BSc. Computer Engineering

SOFTWARE DEVELOPMENT LIFECYCLE



WHAT THE CUSTOMER REALLY NEEDED



How the customer explained it

How the project leader understood it

How the analyst designed it

How the programmer wrote it

How the business consultant described it

How the project was documented

What operations installed

How the customer was billed

How it was supported

What the customer really needed

THE TWELVE-FACTOR APP

THE TWELVE-FACTOR APP

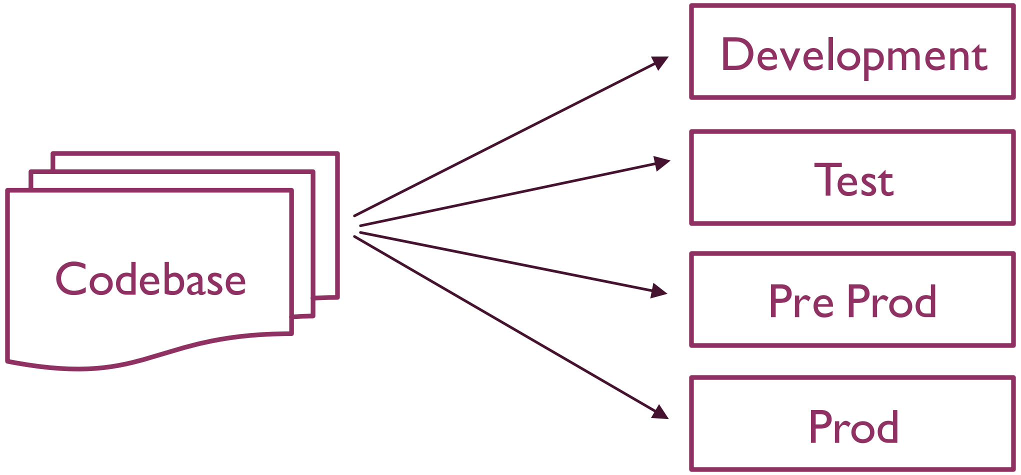
- Use **declarative** formats for setup automation
- Have a **clean contract** with the underlying operating system
- Are suitable for **deployment** on modern **cloud platforms**
- **Minimize divergence** between development and production
- And can **scale up** without significant changes to tooling, architecture, or development practices.

FACTORS

- Codebase
- Dependencies
- Config
- Backing Services
- Build, Release, Run
- Processes
- Port Binding
- Concurrency
- Disposability
- Dev/Prod Parity
- Logs
- Admin Processes

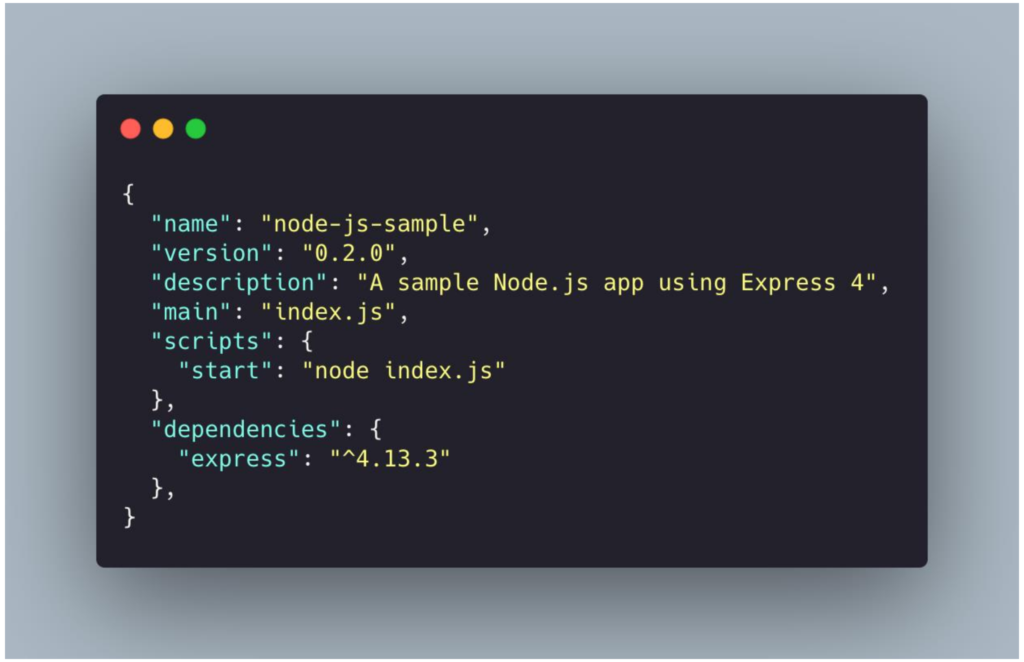
FACTOR I - CODEBASE

- A twelve-factor app is always tracked in a version control system
- One codebase, one application
- Git, TFS, SVN



FACTOR II - DEPENDENCIES

- Explicitly declare and isolate dependencies
- **Never rely on implicit existence of system-wide packages**
- NPM for Node.js, Maven for Java, Rubygems for Ruby



```
{
  "name": "node-js-sample",
  "version": "0.2.0",
  "description": "A sample Node.js app using Express 4",
  "main": "index.js",
  "scripts": {
    "start": "node index.js"
  },
  "dependencies": {
    "express": "^4.13.3"
  },
}
```

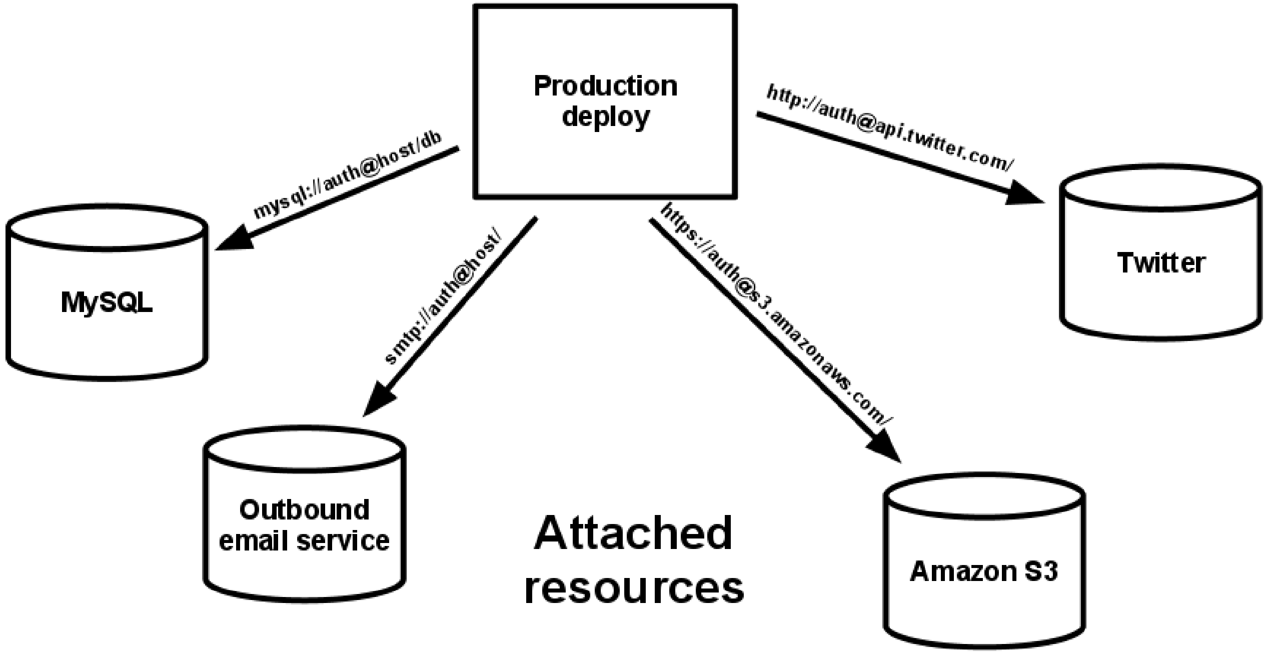
FACTOR III - CONFIG

- Resource handles to the database and other backing services
- Credentials to external services
- **The twelve-factor app stores config in *environment variables***

- https://docs.aws.amazon.com/lambda/latest/dg/configuration-envvars.html
- https://docs.spring.io/spring-boot/docs/current/reference/html/features.html#features.external-config

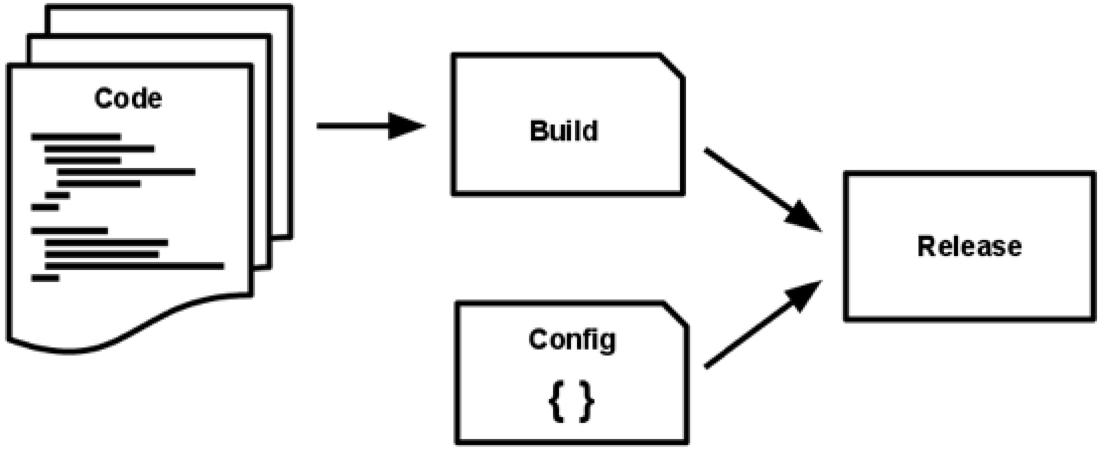
FACTOR IV – BACKING SERVICES

- A *backing service* is any service the app consumes over the network as part of its normal operation.
- The current production external app could be detached, and the new app attached – all without any code changes.



FACTOR V - BUILD, RELEASE, RUN

- The *build stage* is a transform which converts a code repo into an executable bundle known as a *build*.
- The *release stage* takes the build produced by the build stage and combines it with the deploy's current config.
- The *run stage* (also known as "runtime") runs the app in the execution environment.



FACTOR VI - PROCESSES

- **Twelve-factor processes are stateless and share-nothing.** Any data that needs to persist must be stored in a stateful backing service, typically a database.
- Execute the app as one or more stateless processes.

FACTOR VII – PORT BINDING

- **The twelve-factor app is completely self-contained** and does not rely on runtime injection of a webserver into the execution environment to create a web-facing service. The web app **exports HTTP as a service by binding to a port**, and listening to requests coming in on that port.
- Local deployment http://localhost:8080/
- Test deployment https://test-tariff.singlewindow.io/

FACTOR VIII - CONCURRENCY

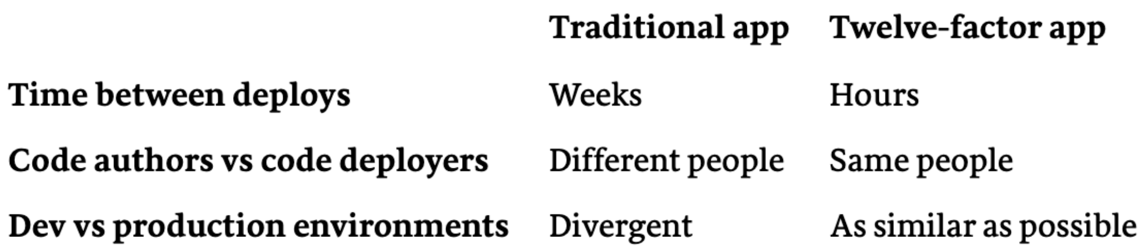
- Using this model, the developer can architect their app to handle diverse workloads by assigning each type of work to a *process type*. For example, HTTP requests may be handled by a web process, and long-running background tasks handled by a worker process.
- https://adam.herokuapp.com/past/2011/5/9/applying_the_unix_process_model_to_web_apps/

FACTOR IX - DISPOSABILITY

- **The twelve-factor app's processes are *disposable*, meaning they can be started or stopped at a moment's notice.**
- Processes should strive to **minimize startup time**.
- Processes **shut down gracefully when they receive a SIGTERM** signal from the process manager.
- Processes should also be **robust against sudden death.**

FACTOR X – DEV/PROD PARITY

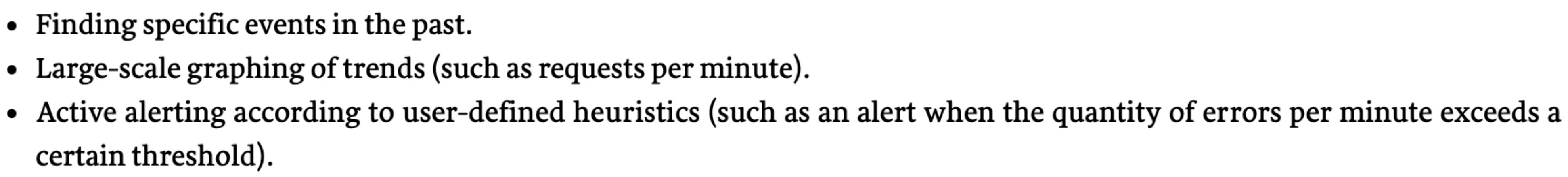
- Keep development, staging, and production as similar as possible
- **The twelve-factor app is designed for continuous deployment by keeping the gap between development and production small.**



	Traditional app	Twelve-factor app
Time between deploys	Weeks	Hours
Code authors vs code deployers	Different people	Same people
Dev vs production environments	Divergent	As similar as possible

FACTOR XI - LOGS

- *Logs* provide visibility into the behavior of a running app.
- Logs are the stream of aggregated, time-ordered events collected from the output streams of all running processes and backing services.
- These systems allow for great power and flexibility



- Finding specific events in the past.
- Large-scale graphing of trends (such as requests per minute).
- Active alerting according to user-defined heuristics (such as an alert when the quantity of errors per minute exceeds a certain threshold).

FACTOR XII - ADMIN PROCESSES

- Run admin/management tasks as one-off processes
- Db migrations
- One-time scripts

- https://12factor.net/

- sukru.cakmak@atez.com.tr