

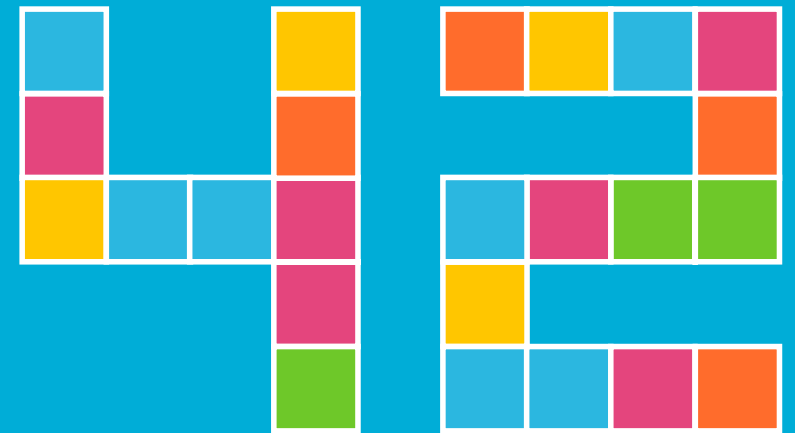
# Insights From a Spring Boot Leisure Project For Everyday Developer Life

Patrick Baumgartner

42talents GmbH, Zürich, Switzerland

@patbaumgartner

patrick.baumgartner@42talents.com



TALENTS

# Abstract

## **Insights From a Spring Boot Leisure Project for Everyday Developer Life**

New gadgets and slow mobile apps tempt many a developer to spend the Christmas holidays reverse engineering, hacking and coding to put their world back together.

Join a motivated developer on his journey to successfully implement his idea. Along the way, he is surprised at how much or how little effort is involved in each step, and what he learns along the way. We dive into the world of Android apps, Spring Boot, Buildpacks, GitHub Actions, Docker containers, GraphQL and API breaking changes.

A thriller in three acts.



# Insights

## From a Spring Boot Leisure Project for Everyday Developer Life

**Patrick Baumgartner**

**42talents GmbH, Zürich, Switzerland**

**@patbaumgartner**

**patrick.baumgartner@42talents.com**

## **WARNING:**

This is a **work of fiction**. Any similarities to real-life events or **persons** are purely **coincidental**. The information, names, characters, places and incidents are the product of the **author's imagination**. **Readers** should regard the story as **entertainment** and not as **representative** of any **real events or persons**.

# Introduction





## Patrick Baumgartner

Technical Agile Coach @ 42talents

My focus is on the development of software solutions with humans.

Coaching, Architecture, Development, Reviews, and Training.

Lecturer @ Zürcher Fachhochschule für Angewandte Wissenschaften ZHAW

[@patbaumgartner](#)

How do you learn new things for work or in general? 

Does it only happen during working hours? 

How do you prepare for future tasks? 

# Agenda

# Agenda

---

- Getting a new toy (device)
- The pain caused
- Trying to find a solution
- DIY
- More pain
- The (current) result
- Lessons learned (for work)

# Meet the Developer



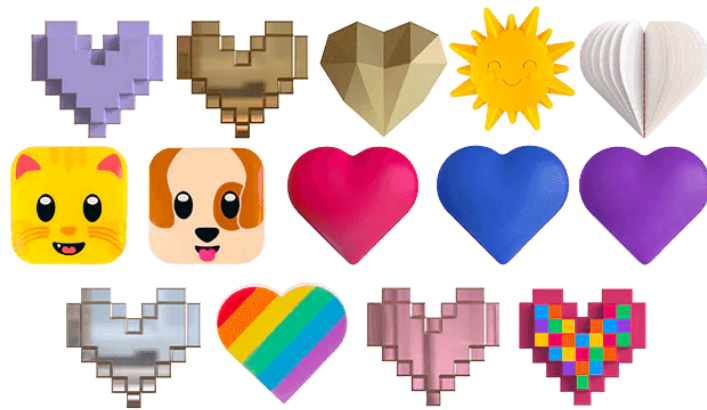
# Meet the Developer

---



- Problem solver, software developer and architect
- Loves to have a challenge and learn new things
- Has a family and a full-time job
- Tinkering around Christmas is possible
- Got a new toy (device) for Christmas 📺







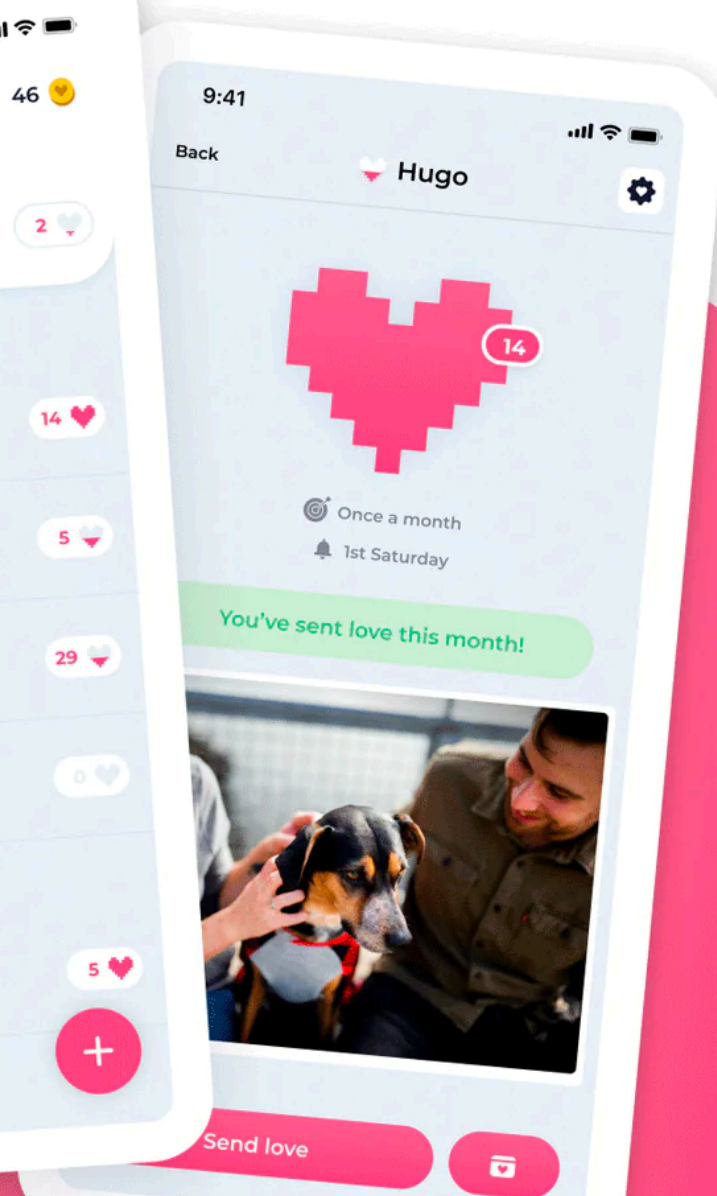
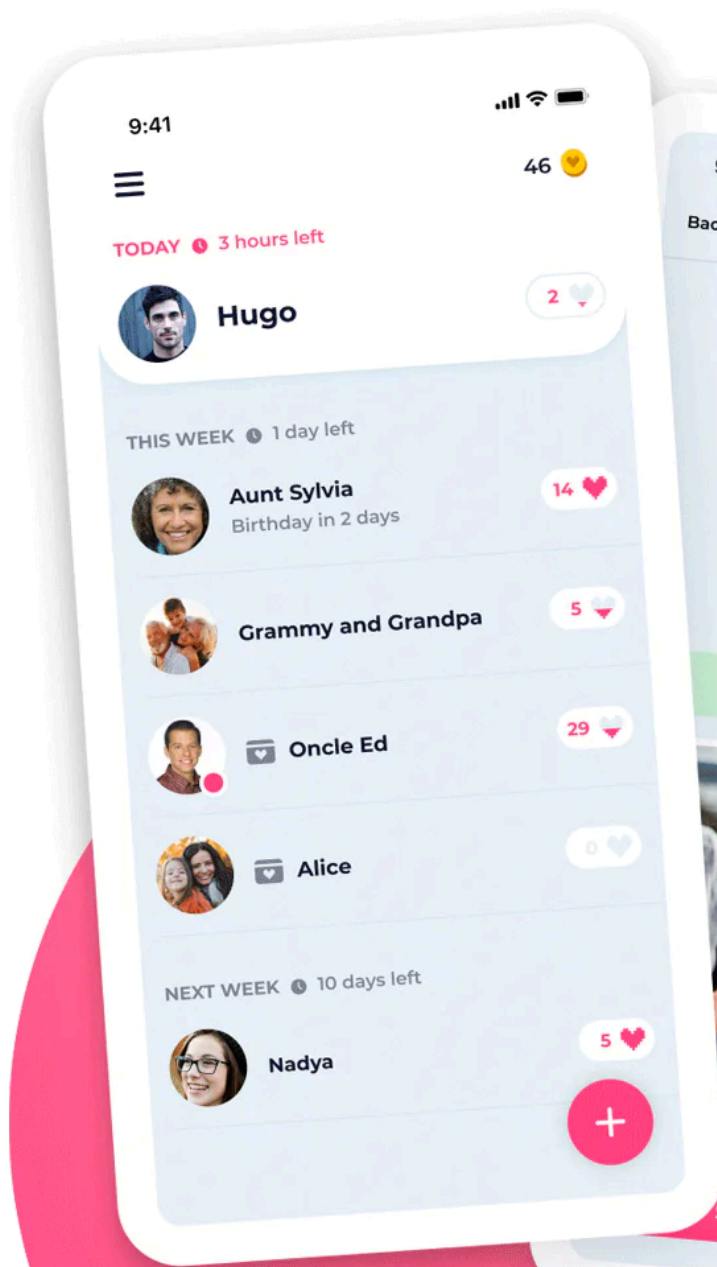


Happy Mother's Day

# The Pain

1997

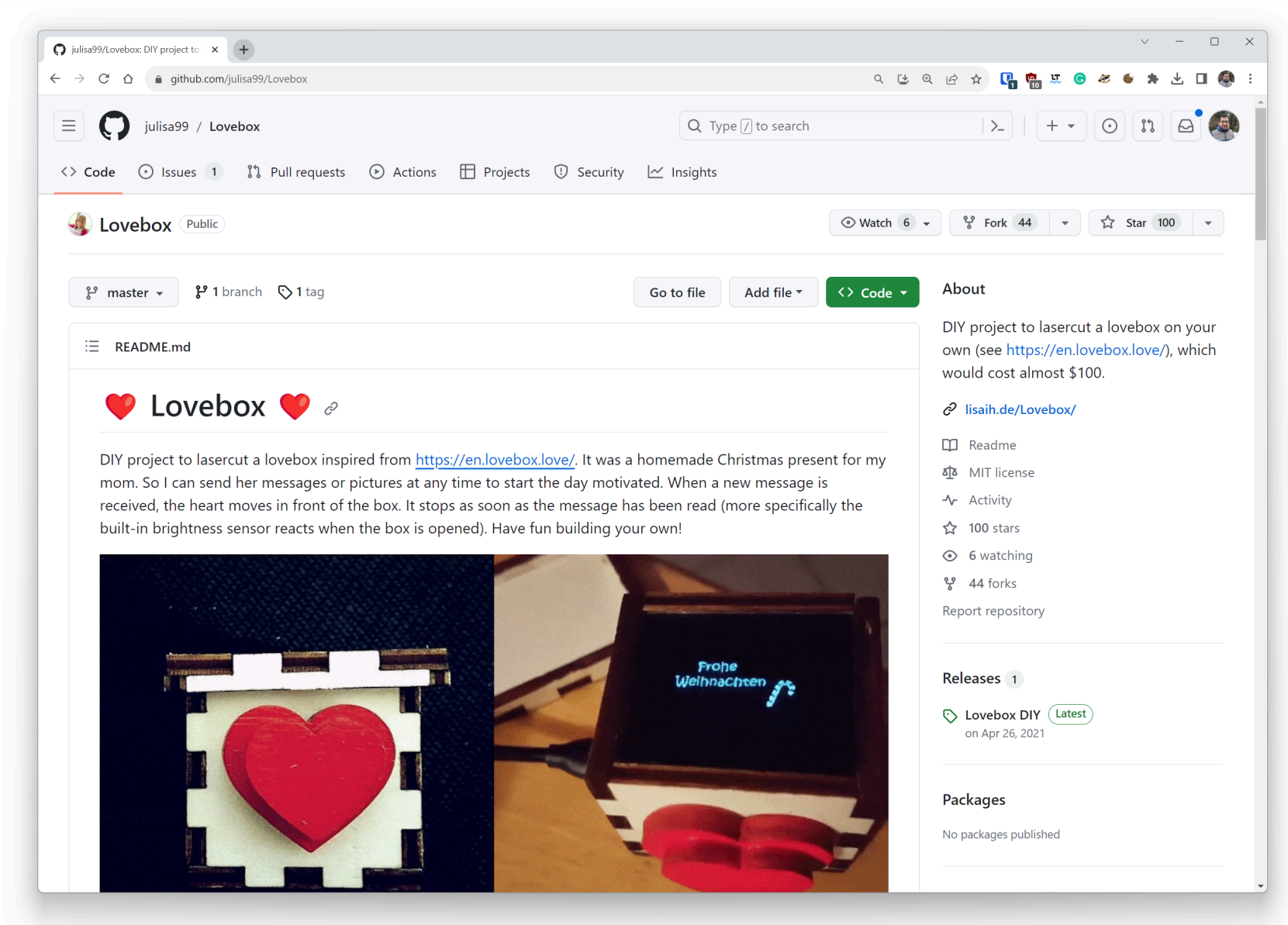
1998

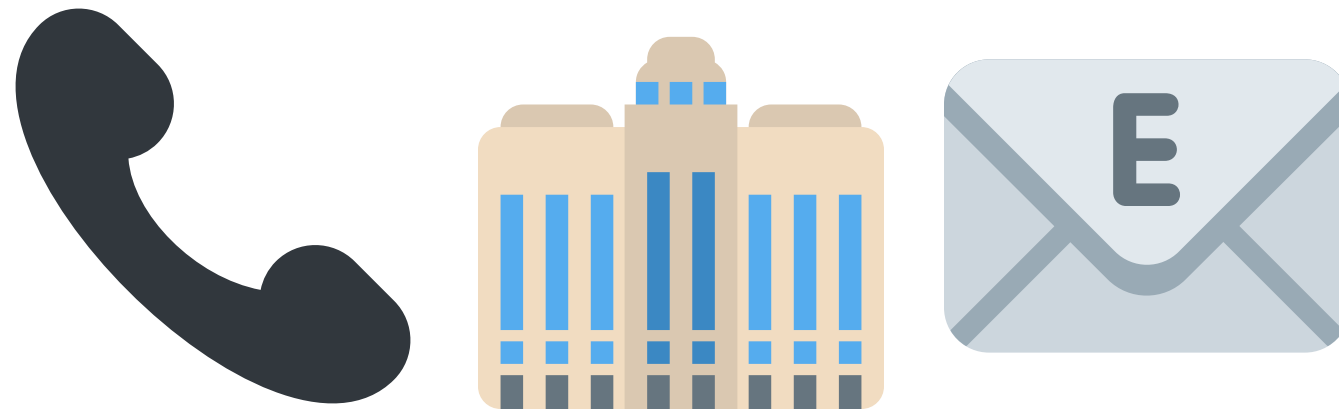


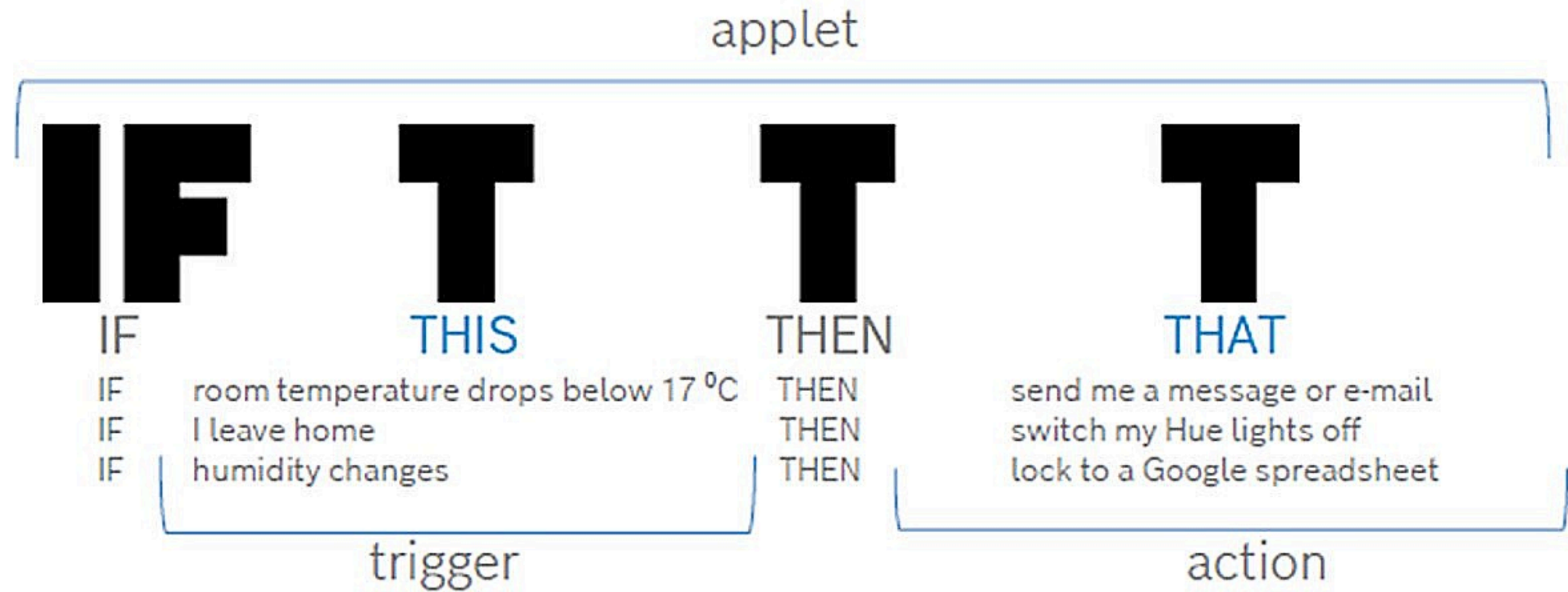












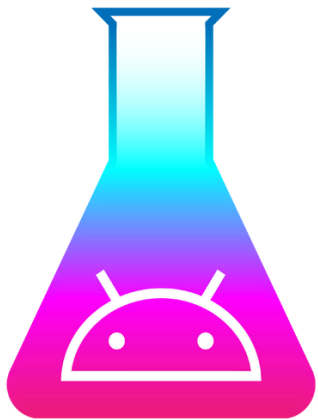




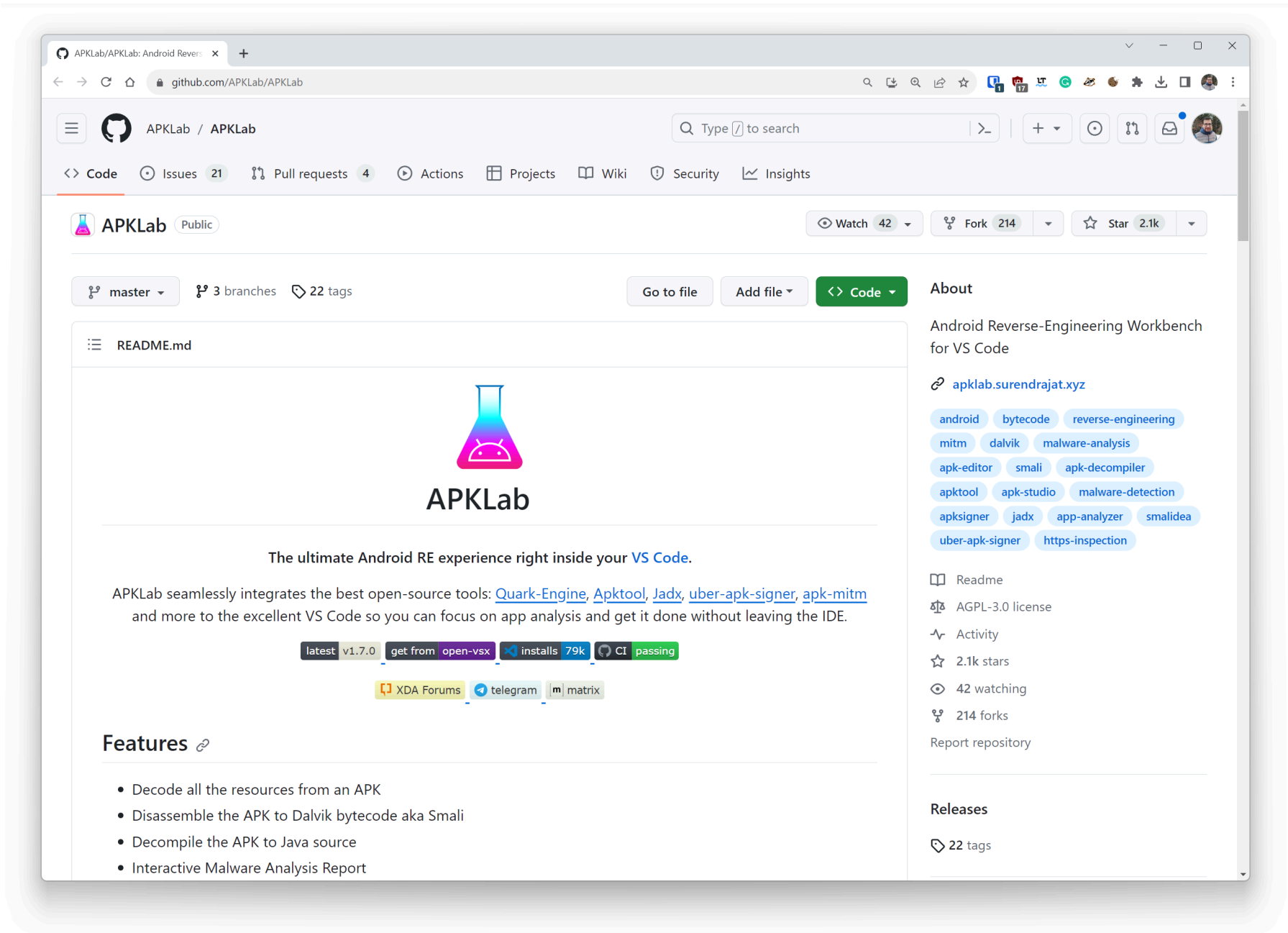
# Cracking the Mobile App

## (Reverse Engineering)

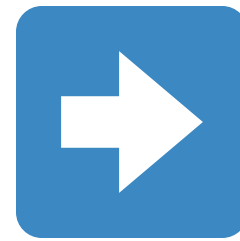
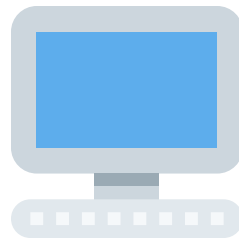
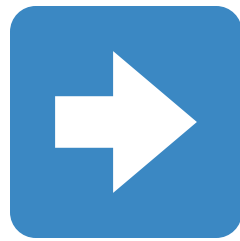
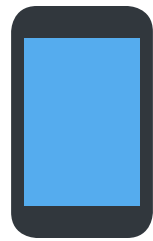
---



- Decode APK
- ... some other steps
- Certificate Unpinning
- ... some other steps
- Repackaging APK







# Login Request

---

```
curl --location --request POST 'https://app-api.loveboxlove.com/v1/auth/loginWithPassword' \
--header 'accept: application/json' \
--header 'content-type: application/json' \
--header 'host: app-api.loveboxlove.com' \
--data-raw '{
    "email": "my@email.com",
    "password": "mySecret"
}'
```

```
{
  "_id": "42c61f261f399d0016350b7f",
  "firstName": "FirstName",
  "email": "my@email.com",
  "token": "exJhbGcp0iJIVzI1NipsInR5cCI6IkpXVDj9
.eyJ1c2xySwFiOiI2MWM5LWYzNjFhMzk5YzAwMTYzNTBhN2YiLCJpYXQiOiFjE22DAVNmQwNTL9
.q1svp_roqCu4MFwBMwNZu2eyImFGjogvNeR4tkoTLPe"
}
```

# Creating a Server App

---



- Spring Boot
- Spring Cloud Open Feign Starter
- Telegrambots Spring Boot Starter
- Imgscalr
- OCI (Docker) Container
- Using Lovebox GraphQL API

# Spring Boot Application

---

```
@EnableScheduling @EnableFeignClients @SpringBootApplication
@EnableConfigurationProperties({LoveboxRestClientProperties.class, LoveboxBotProperties.class})
@Import(TelegramBotStarterConfiguration.class)
public class LoveboxTelegramSenderApplication {

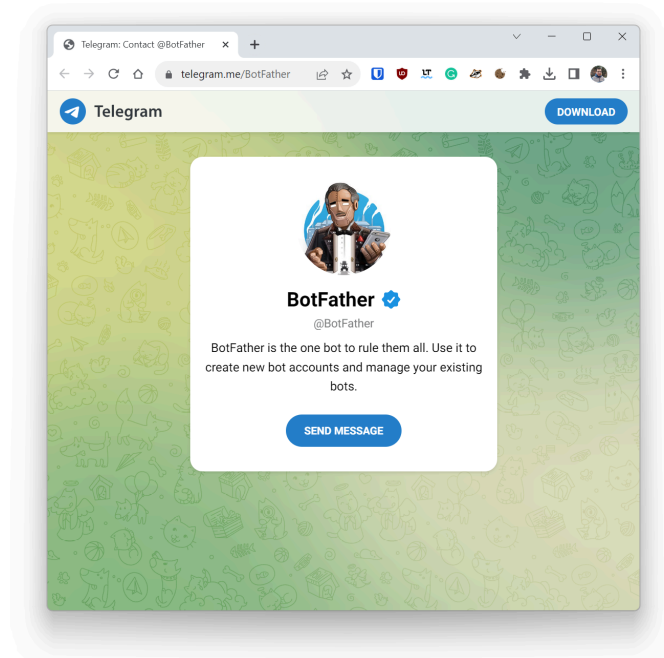
    public static void main(String[] args) {
        SpringApplication.run(LoveboxTelegramSenderApplication.class, args);
    }
}
```

```
@Component
public class YourBotName extends TelegramLongPollingBot { ... }
```

```
<dependency>
  <groupId>org.telegram</groupId>
  <artifactId>telegrambots-spring-boot-starter</artifactId>
  <version>6.8.0</version>
</dependency>
```

# Setting Up the Telegram Bot

---



- Contact @Botfather
- Create a chatbot with /newbot
- Get the bot username, and token

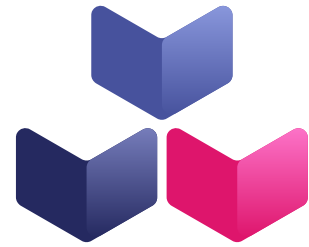




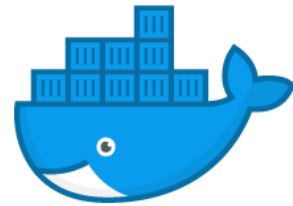








Buildpacks.io



docker



# 2 Nights of Debugging





# Patching Buildpack Run-Image

---

```
FROM paketobuildpacks/run:base-cnb

USER root

## Idea from: https://blog.adoptopenjdk.net/2021/01/prerequisites-for-font-support-in-adoptopenjdk/
RUN apt-get update \
    && apt-get install -y --no-install-recommends \
        wget \
        fontconfig \
    && rm -rf /var/lib/apt/lists/*

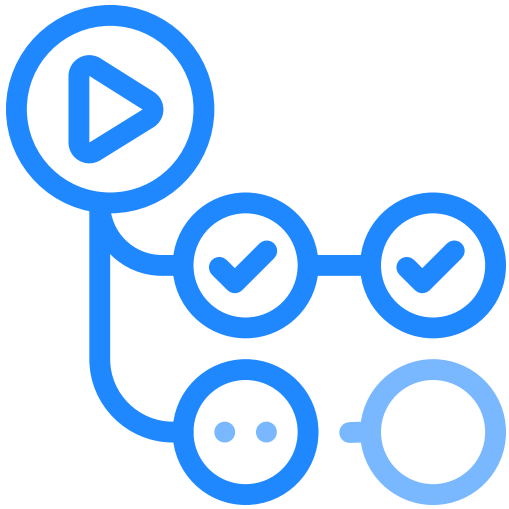
# Install Noto Emoji Manually
RUN mkdir -p /usr/share/fonts/truetype/noto \
    && cd /usr/share/fonts/truetype/noto \
    && wget https://github.com/googlefonts/noto-emoji/raw/9a5261d871451f9b5183c93483cbd68ed916b1e9/fonts/NotoEmoji-Regular.ttf \
    && fc-cache -fv

USER cnb
```



# GitHub Actions

---



- Checkout
- Login to DockerHub
- Build and push Buildpack run-image
- Setup JDK 17
- Build app artifact with Maven
- Build and push the container image for the app



# Dependency Updates with Dependabot

---



- Updating GitHub Actions
- Updating Maven dependencies

# Automated Deployment with Watchtower

---

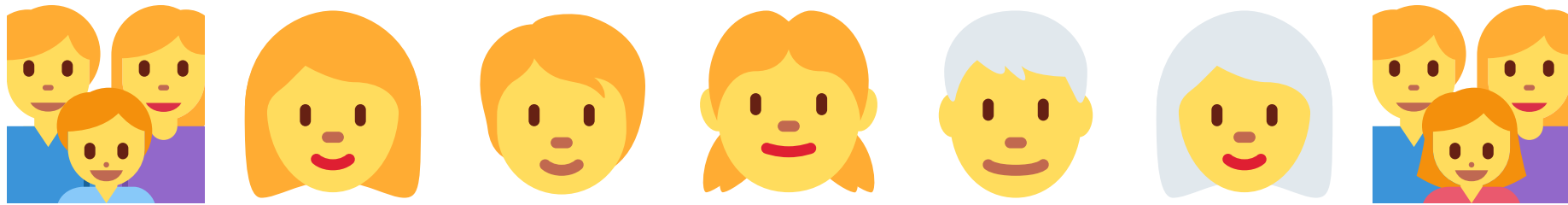


- Polling container updates from DockerHub
- Cron runs by default every 24 hours
- Update images and restart containers
- Everything runs on my NAS!

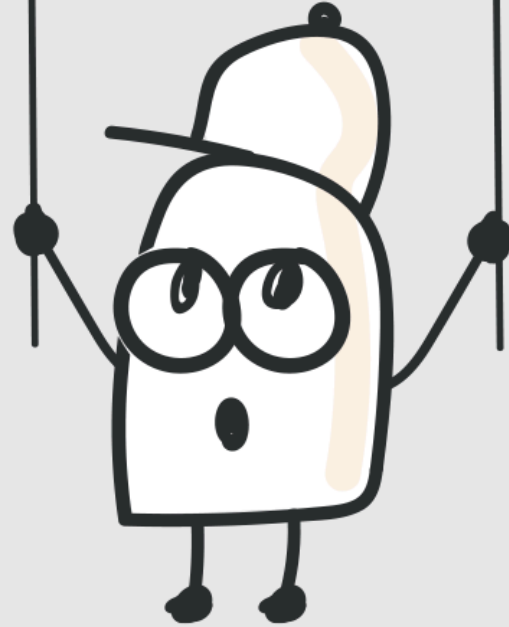
**There is absolutely no need to have Kubernetes! 🏹**

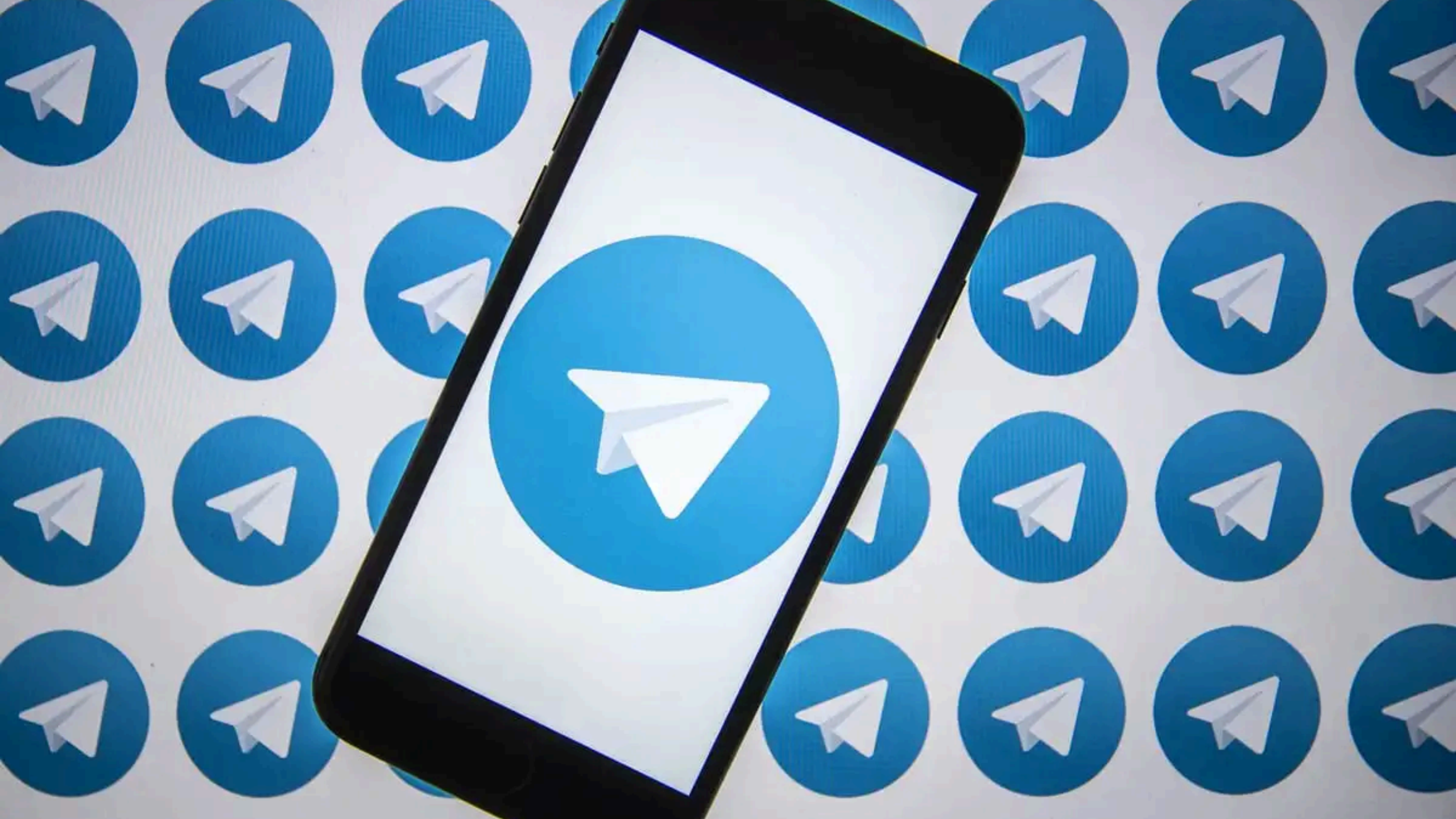
# The NEW Product

NEW Product



# FEATURES







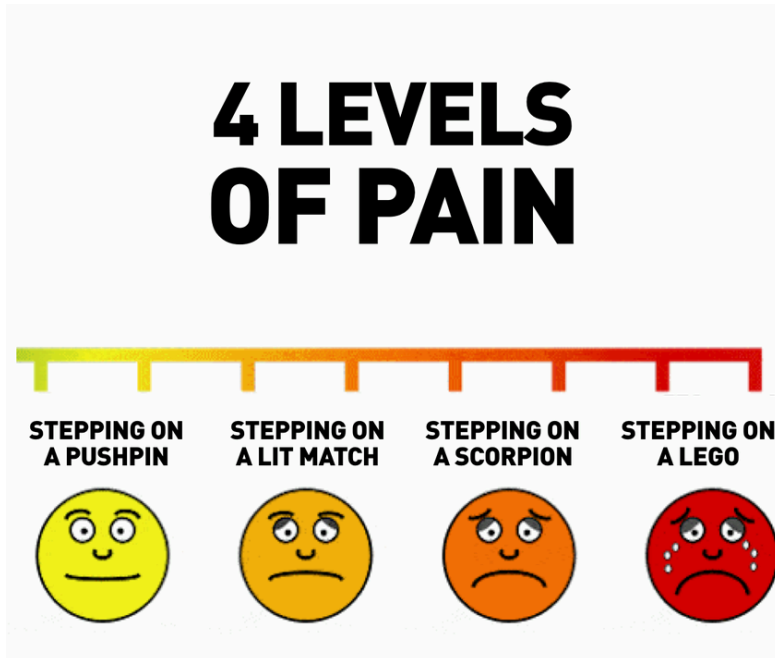
UPDATE...

More Pain



# More Pain ...

---



- No contribution accepted (Telegram starter)
- API changes (Lovebox GraphQL)
- Java and Spring Boot Updates
- New & better ways to do things

# Product Improvements

# Product Improvements - What's Next?

---



- Producer-Driven Contract Testing
- Better Error Handling
- Logging and Monitoring improvements

# Lessons Learned

# *Manifesto for Software Craftsmanship*

Raising the bar.

As aspiring Software Craftsmen we are raising the bar of professional software development by practicing it and helping others learn the craft. Through this work we have come to value:

Not only working software,  
but also **well-crafted software**

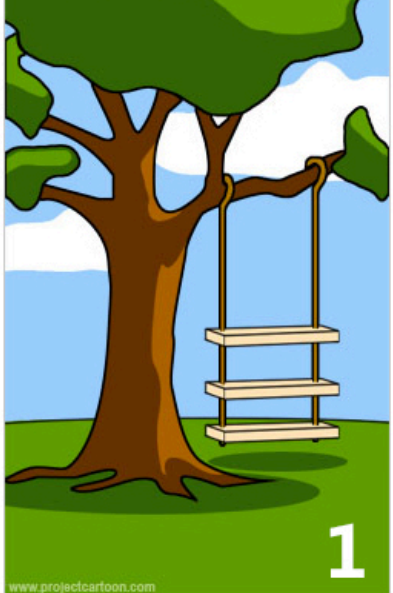
Not only responding to change,  
but also **steadily adding value**

Not only individuals and interactions,  
but also **a community of professionals**

Not only customer collaboration,  
but also **productive partnerships**

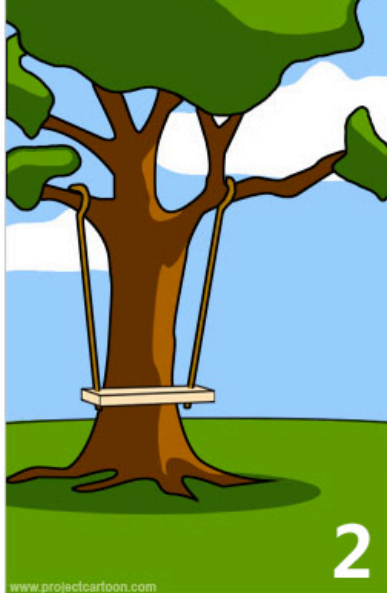
That is, in pursuit of the items on the left we have found the items on the right to be indispensable.





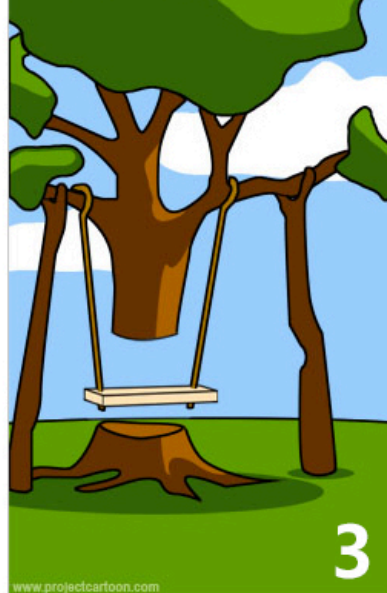
1

How the customer explained it



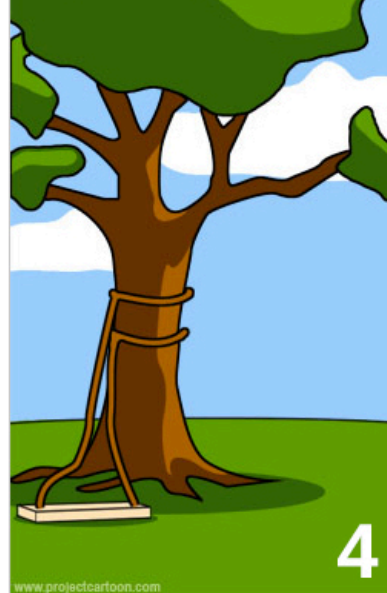
2

How the project leader understood it



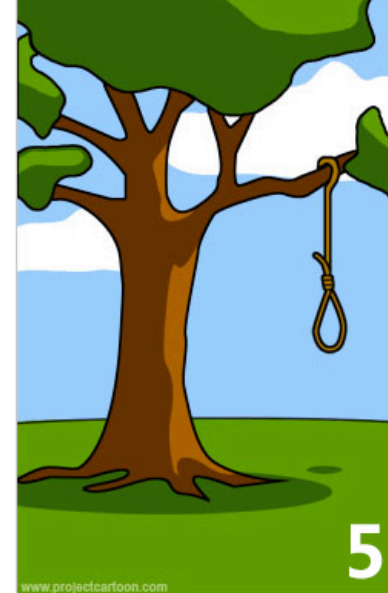
3

How the analyst designed it



4

How the programmer wrote it



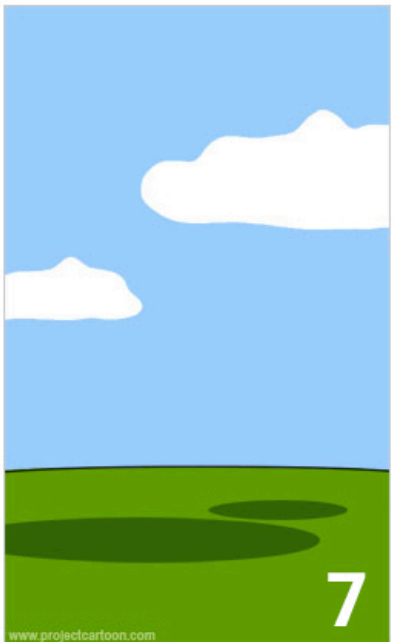
5

What the beta testers received



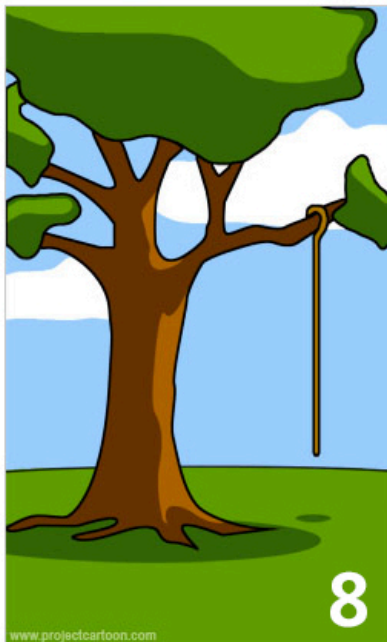
6

How the business consultant described it



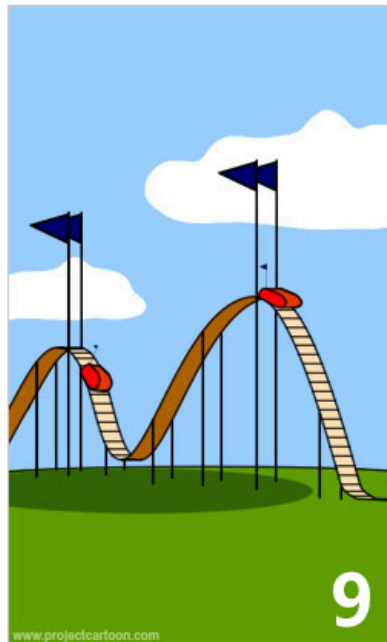
7

How the project was documented



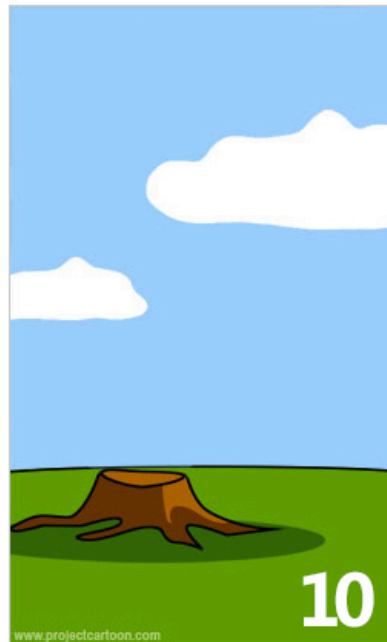
8

What operations installed



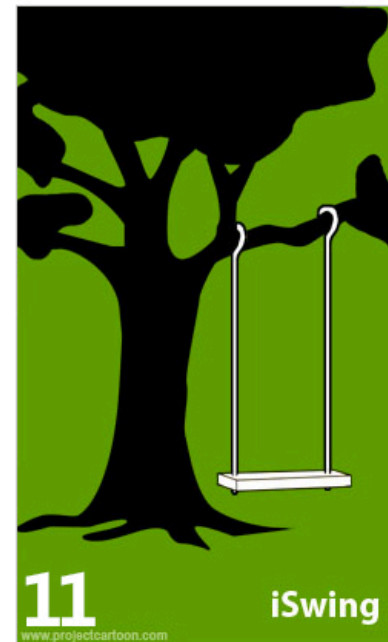
9

How the customer was billed



10

How it was supported



11

iSwing

What marketing advertised



12

What the customer really needed

**Every developer** should have a **leisure project** as a learning experience and a **testing ground for new ideas**.  
But **that's not the product** that goes into production and is used by customers!

**Happy Coding!** 



# Insights

## From a Spring Boot Leisure Project for Everyday Developer Life

**Patrick Baumgartner**

**42talents GmbH, Zürich, Switzerland**

**@patbaumgartner**

**patrick.baumgartner@42talents.com**

