

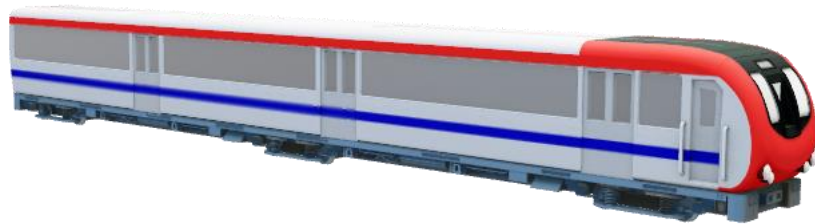
INFRA-IMPROVER

Yves Jegge
Beat Weisskopf
Sandrine Ngo-Dinh



Idee

- App zur Meldung von Infrastruktur-Schäden
 - Idee kam aus eigener Erfahrung – wo kann man so was melden?



Use Case: Melden

Situation:

- Bürger ist genervt von einem Schaden (z. B. Schlagloch, defekte Lampe)

Aktion:

- Verwendet sein Handy
- Prüft in der App, ob der Schaden schon gemeldet wurde
- Erstellt einen Schaden
- Lädt Fotos hoch

Ergebnis:

- Schaden ist dokumentiert und kann von anderen Benutzern angesehen und priorisiert werden



Use Case: Abstimmen

Situation:

- Bürger ist genervt von einem Schaden (z. B. kaputter Gehweg, Laterne defekt)

Aktion:

- Prüft in der App, ob der Schaden schon gemeldet wurde
- Findet den Eintrag
- Votet für den Schaden (→ Priorität steigt)

Ergebnis:

- Schaden gewinnt an Sichtbarkeit und Dringlichkeit
- Gemeinde erkennt, welche Probleme am dringendsten sind



Use Case: Analyse

Situation:

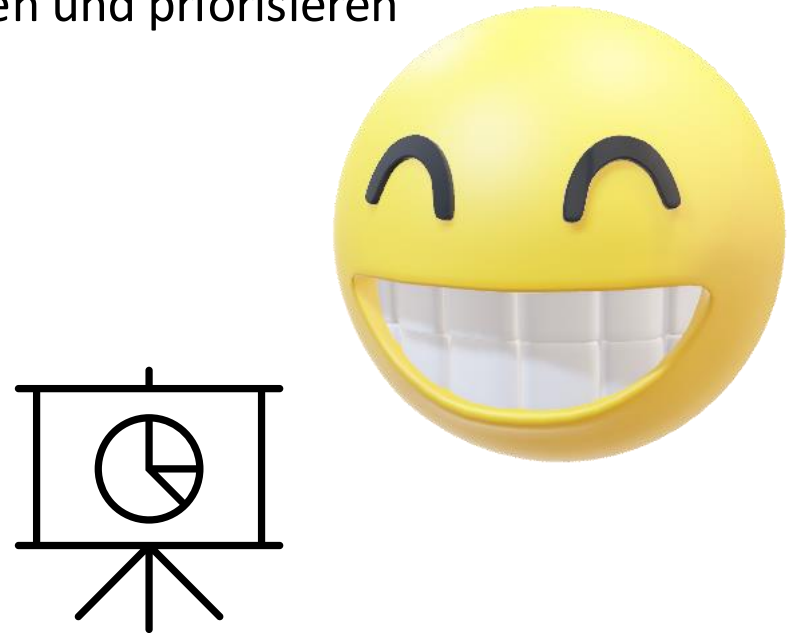
- Gemeinde möchte Infrastruktur-Schäden schneller erkennen und priorisieren

Aktion:

- Prüft in der App die gemeldeten Schäden
- Erkennt den dringendsten Schaden
- Behebt den dringendsten Schaden

Ergebnis:

- Priorisierung von Infrastruktur-Problemen



Use Case: Beheben

Situation:

- Gemeinde hat den Schaden behoben

Aktion:

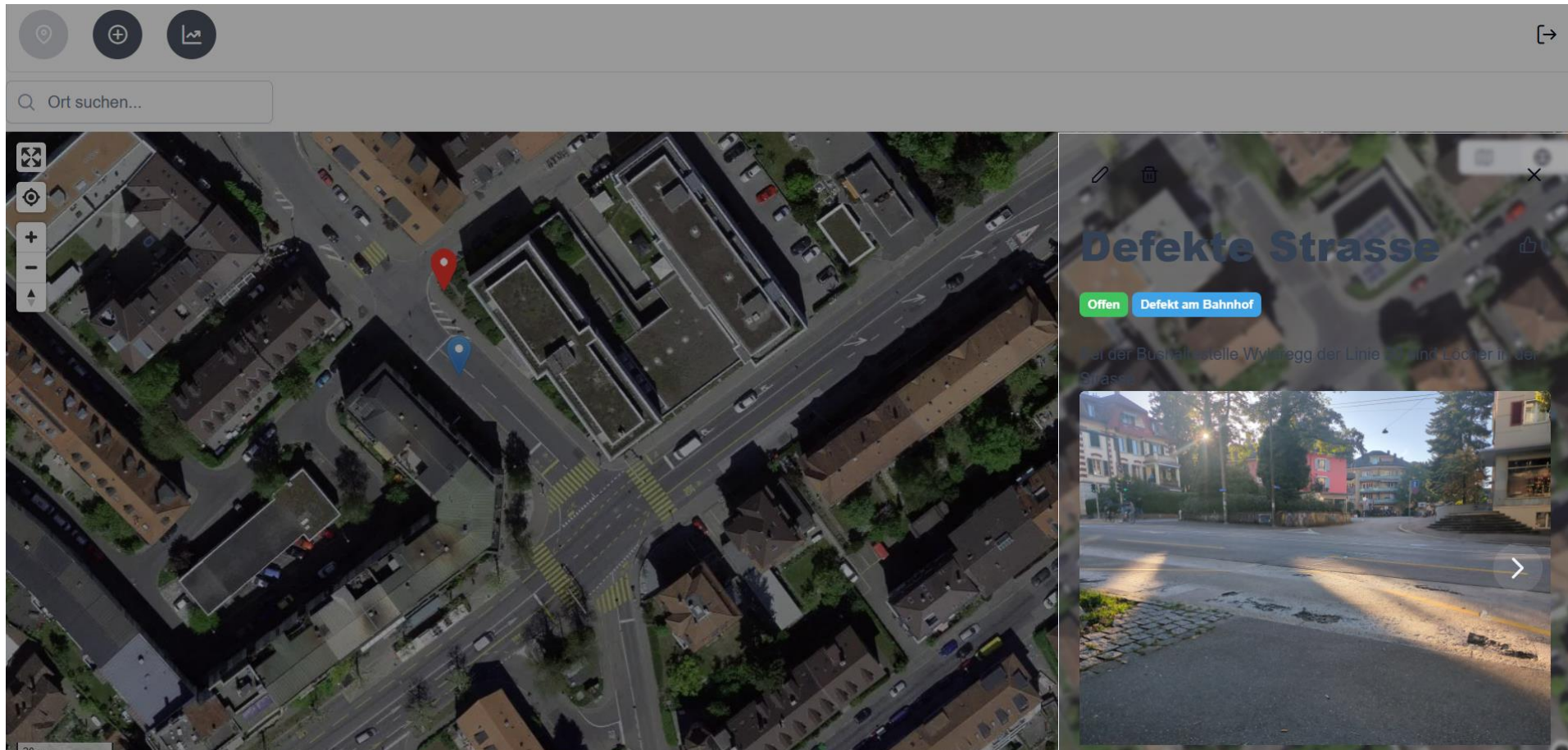
- Markiert den Schaden als behoben
- Benachrichtigung bei Status-Wechsel

Ergebnis:

- Zufriedene Bürger



Demo



Backend Technologien

- **Supabase PostgreSQL:** Datenbank
 - **Supabase Auth:** Authentifizierung
 - **Supabase RPC:** Remote Procedure Calls
 - **Supabase Storage:** Blob-Speicher
 - **Postgre PostGIS Extension:** → eignet sich gut für Geolocation
- **photon:** Reverse GeoCoding API
- **SendGrid:** Email Versend- API
- **maptiler:** Provider für Kartenmaterial (swisstopo/satellite)



Supabase



Frontend Technologien

Framework: Angular 20

Supabase JS: Backend-Access

PrimeNG: Component Library

Chart.js: Diagramm Library

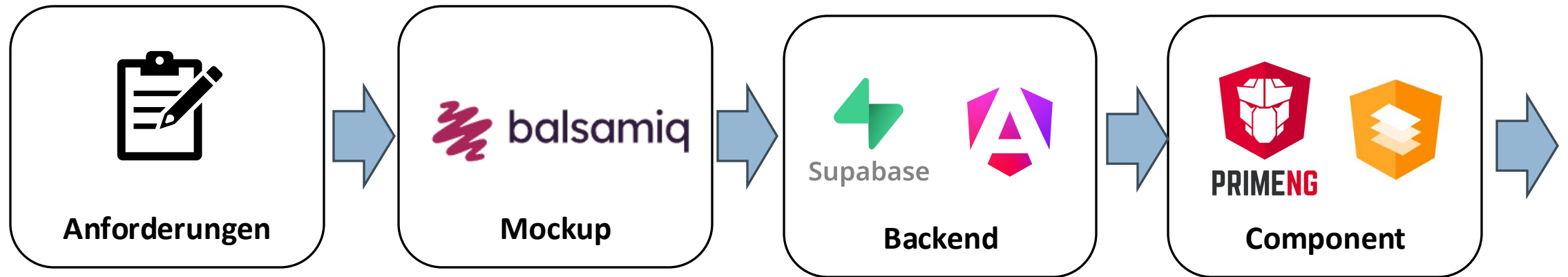
Tailwind: SCSS/CSS framework

ngx-mapLibre: Weltkarten Library

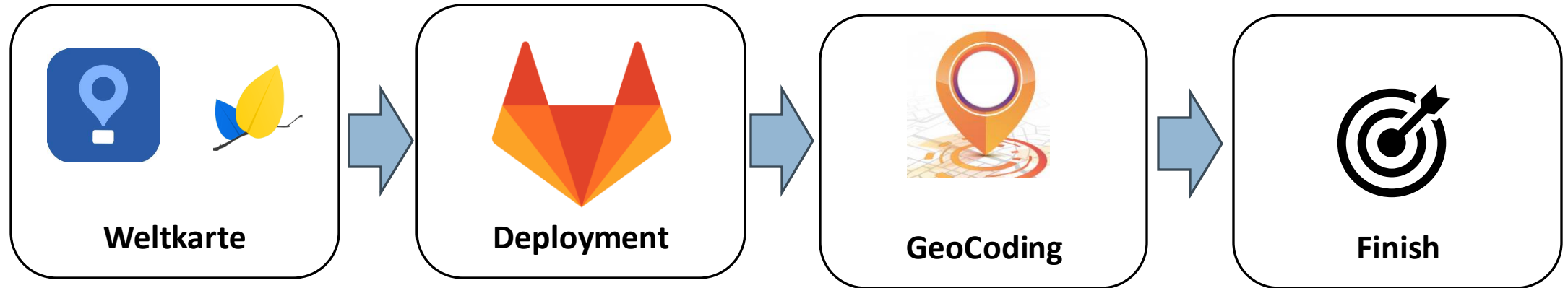
exifreader: Foto GPS Bestimmungs Library



Development Prozess I

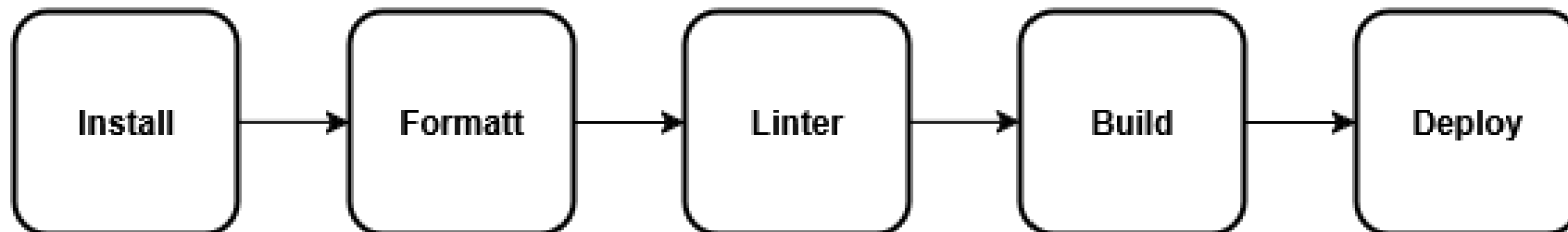


Development Prozess II



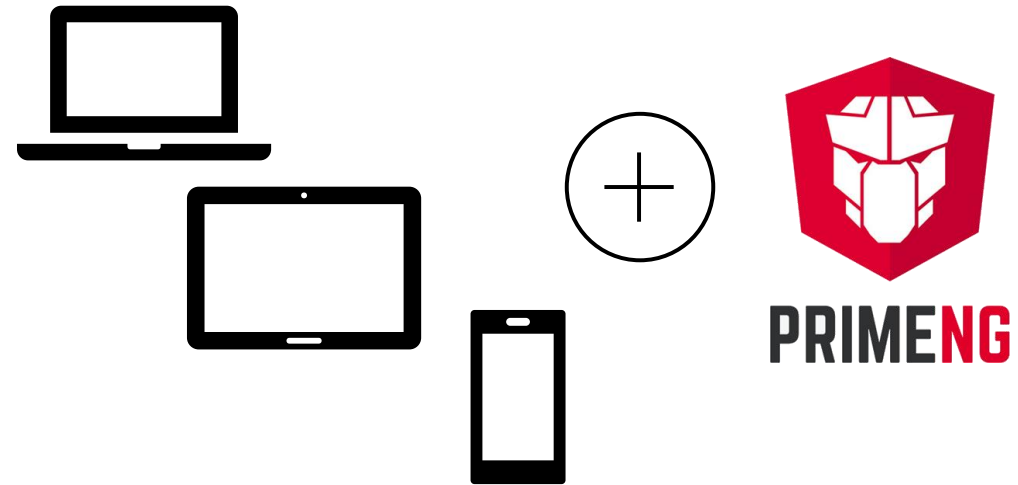
Deployment

- GitLab Build Pipeline
- GitLab Pages



Herausforderungen

- Map-Entwicklung (Technologie, Features)
- Responsives Design mit PrimeNg
- Nicht jedes Gerät verhält sich gleich
- Mockup vs Realität
- CORS Policy beim Reverse GeoCoding



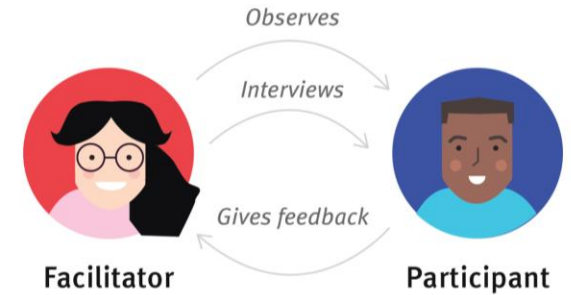
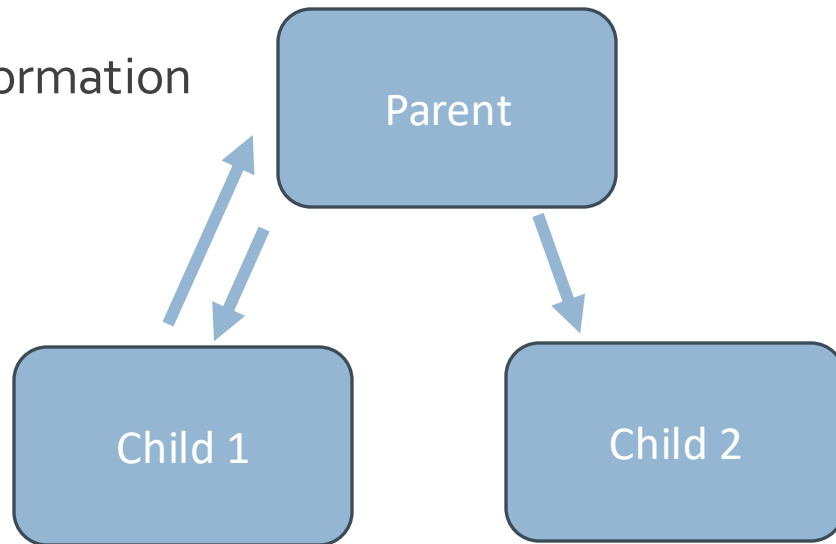
Learnings I

- Technologien besser evaluieren



Learnings II

- Responsives Styling ist aufwändig
- Datenflüsse/States besser konzipieren
- Usability früher beginnen
- Linter war sehr hilfreich
- Image Transformation



{css}



tailwindcss

Vielen Dank fürs Zuhören!

