

Zoals besproken is het de bedoeling dat je een werkend stuk software end-2-end gaat maken. De software geeft invulling aan de opdracht zoals verwoord in Technische opdracht Cas.pdf.

## **Stap 1 - REST API**

- Design en bouw een REST API met Java waarbij je de ingelezen / gemodificeerde data opslaat in een datastore.
  - Bouw YAK story 1 t/m 3 (muv van command line interface zoals beschreven in YAK1)
  - Gebruik Jakarta EE voor maken van REST API
    - Implementeer hierbij status codes, error handling, model validatie, verbs (HTTP Methods) en routes
  - Gebruik JPA voor data retrieval/opslag richting Postgres
  - Schrijf paar unittesten voor core domein code
  - Neem je code op in GitHub

## **Stap 2 - UI**

- Maak een UI met Vanilla.js of een framework (bijv. JSF). Deze pagina roept de REST API aan zodat de gegevens worden opgehaald of opgeslagen in de database.
  - Zoek uit wat CORS is en implementeer dat in de API (tip: lees officiële CORS spec <https://www.w3.org/TR/2020/SPSD-cors-20200602/>)

## **Stap 3 - Gebruik Docker voor bouwen, testen en lokaal deployen van je software**

- Deploy Postgres
- Deploy de REST API
- Deploy de UI
- Laat totale oplossing met bovenstaande 3 componenten werken

## **Verder raad ik je aan om je in te lezen over het volgende:**

\* SCRUM: methodiek die wordt toegepast

(<https://www.infoq.com/minibooks/scrum-xp-from-the-trenches-2/> & Official SCRUM Guide van SCRUM.org)

Heel veel succes, maar bovenal ook heel plezier met het bovenstaande. Het doel is om zo veel mogelijk leren en je goed voor te bereiden op je volgende gesprek bij Triodos. Zodat je indruk zult maken. Bij vragen kun je me altijd contacteren. Werk niet te hard, het is natuurlijk niet de bedoeling dat dit je uitput.