# Casus 1 – Bouw een webapplicatie

Bouw een 3-tier Java webapplicatie waar gebruikers de beschikbare voertuigen kunnen inzien. De data bestaan uit een *naam*, *beschikbaar* en *totaal*. Let op dat het niet mogelijk is dat het aantal beschikbaar hoger kan zijn dan het totaal. ALLEEN een beheerder kan deze data wijzigingen. Tot slot moeten alle events gelogd worden.

PVA:

String Name  
Boolean isBeschikbaar  
double Totaal.  
Totaal moet denk geen List zijn maar via Rest in repo komen.  
Daar wellicht een methode die uitrekend wat het totaal is? Of misschien een get request waar het totaal te zien is. Daar dan op inspelen wat het totaal beschikbaar en onbeschikbaar is. Eventueel kan ik een extra klasse maken met Beschikbaar of onbeschikbaar.

Als een User een voertuig wilt reserveren dan moeten andere users in de wachtrij worden geplaatst om zodat dubbele boekingen tegen worden gegaan. **(Monitoring)**

Er moet een log komen wie welk voertuig heeft gereserveerd.

Het programma moet kunnen refreshen zonder dat de server gereset dient te worden **(Dynamic service/ configuration)**

Ik moet unit tests schrijven.

## Voorbeeld stack

* Java 8 of hoger;
* Maven;
* Spring Boot;
* PostgreSQL;
* Hibernate;
* Angular 8;

## Gewenste aandachtsgebieden

* Microservice; /Restful API
* Logging;
* Monitoring; /
* Autorisatie en Authenticatie;
* Dynamic Configuration;
* Security;
* Testing;
* Dockerized;

*\* let op indien u al een omgeving heeft waar de gewenste aandachtsgebieden voorkomen kan deze ook besproken worden.*

# Casus 2 – Bouw een ontwikkelstraat voor een Java Webapplicatie

Bouw een MVP ontwikkelstraat waar ik mijn Java Webapplicatie kan beheren. Ga verder met de applicatie van Casus 1 OF gebruik een elders Java Spring Boot applicatie om met casus 2 aan de slag te gaan,

## Voorbeeld stack

* GitLab;.
* SonarQube;
* ELK / EFK;
* Grafana / prometheus; check kafka
* Oath2.0 + OIDC;
* Consult;
* Keycloak / OCTA;
* Vault;
* Kubernetes (EKS/GKE) / Minikube / docker-compose

## Gewenste aandachtsgebieden

* Versiebeheer;
* CI/CD;
* Logging;
* Monitoring;
* Autorisatie en Authenticatie;
* Dynamic Configuration;
* Security;
* Quality checks;

*\* let op indien u al een omgeving heeft waar de gewenste aandachtsgebieden voorkomen kan deze ook besproken worden.*