

Geachte Hilal,

Graag licht ik het idee achter mijn project ReadyRoad toe, samen met de motivatie voor de keuze van dit project en de gebruikte technische architectuur.

Het idee voor ReadyRoad is ontstaan na uitgebreid onderzoek en het testen van verschillende bestaande applicaties voor het theoretisch rijexamen. Tijdens deze analyse kwamen meerdere terugkerende problemen naar voren bij de meeste beschikbare toepassingen. Zo is de ondersteuning voor meerdere talen vaak beperkt, terwijl het theoretisch rijexamen in België officieel in verschillende talen wordt aangeboden. Daarnaast is de educatieve inhoud in veel gevallen oppervlakkig of grotendeels achter een betaalmuur geplaatst. Ook technische instabiliteit komt regelmatig voor, zoals applicaties die vastlopen, fouten bij het laden van vragen of onnauwkeurige resultaten tijdens proefexamens.

Op basis van deze vaststellingen, en gesteund door de kennis die ik tijdens de opleiding heb opgedaan, evenals uw begeleiding en opvolging, heb ik ervoor gekozen dit project te ontwikkelen met als doel kandidaten beter te ondersteunen bij het slagen voor het theoretisch rijexamen. Dit gebeurt door duidelijke en correcte uitleg te bieden over verkeersregels en verkeersborden, een technisch stabiel systeem te bouwen op basis van correcte logica en grondige tests, echte meertalige ondersteuning te voorzien, en het begrip van de leerstof centraal te stellen in plaats van louter memorisatie of gokken.

Daarom is ReadyRoad niet ontworpen als een eenvoudige vragenapplicatie, maar als een intelligente examenmotor die gericht is op leren, analyseren en een realistische voorbereiding op het theoretisch rijexamen.

### **Operationele samenvatting – Backend & Frontend**

#### **Backend (Spring Boot)**

De backend vormt het centrale brein van de applicatie. Alle kernlogica bevindt zich uitsluitend in de backend. Er is geen examenlogica aanwezig in de gebruikersinterfaces.

Gebruikte technologieën:

Java 21, Spring Boot, Maven, MySQL 8, Flyway, Spring Security met JWT-authenticatie.

#### **Belangrijkste verantwoordelijkheden:**

Het genereren van examens, het voorkomen van herhaling van vragen via een cooldown-mechanisme van 24 uur, dynamische moeilijkheidscontrole, automatische correctie van antwoorden, berekening van resultaten, opvolging van gebruikersvoortgang, foutenanalyse, afdwingen van de Belgische exameregels, en bescherming van gebruikersgegevens met volledige isolatie tussen accounts.

De backend stelt één uniforme REST API ter beschikking met stabiele JSON-responses en een duidelijk API-contract dat onafhankelijk is van interne herstructureringen.

Er zijn twee werkingsmodi:

Een ontwikkelmodus zonder JWT voor snelle ontwikkeling, en een beveiligde productiemodus met verplichte JWT-authenticatie en rolverdeling (USER en ADMIN).

De databank gebruikt één coherent schema voor examens, oefening, voortgang en analyses. Flyway wordt ingezet voor gecontroleerde, incrementale database-evolutie, aangevuld met database triggers en constraints ter bescherming van de data-integriteit.

## **Front-end**

De front-end lagen consumeren uitsluitend de API en bevatten geen enkele vorm van examenlogica. Hun verantwoordelijkheid beperkt zich tot weergave, interactie en gebruikerservaring.

### **Flutter (mobiele applicatie)**

Flutter wordt gebruikt voor de officiële mobiele applicatie op Android en iOS. Deze is bedoeld voor dagelijkse oefening, proefexamens, voortgangsopvolging en het weergeven van analyses. De applicatie ondersteunt meerdere talen, inclusief volledige RTL-ondersteuning voor Arabisch, en biedt consistente prestaties en een uniforme interface op alle toestellen.

### **React (webapplicatie)**

React (next.js) wordt ingezet voor een aanvullende webapplicatie die toegang biedt tot het platform via de browser. Deze is geschikt voor het bekijken van inhoud, het afleggen van examens, het opvolgen van voortgang en het gebruik van het systeem zonder mobiele applicatie. De webapplicatie biedt goede prestaties, SEO-ondersteuning en maakt gebruik van exact dezelfde API zonder afwijkingen in logica of resultaten.

## **Samenhang tussen Backend en Front-end**

Het systeem bestaat uit één backend en één API, met meerdere front-ends. Elke wijziging in regels, berekeningen of logica wordt automatisch doorgevoerd in zowel de mobiele applicatie als de webversie. Hierdoor blijven data, resultaten en examengedrag volledig consistent over alle platformen heen.

Ik bezorg u deze toelichting graag ter evaluatie en hoor met plezier eventuele opmerkingen of suggesties ter verdere verbetering van het project.

Met vriendelijke groet,

Haydar Tarek