

Opdracht Gegevensbanken - deel 3: DB creatie en front-end

Tussentijds verslag

TEAMLEDEN

1. Zimcke Van de Staey
2. Ruben Van Looy
3. Minh Tri Tran

SESSIE

1. Welke moeilijkheden ondervonden jullie tijdens deel 1 van de opdracht (d.w.z., de modellering van het domein in ER)?

- Verdedigen van slechts één voorstellingswijze wanneer er meerdere, alternatieve voorstellingen mogelijk zijn.
- Beslissen of een entiteit zwak moet zijn of niet.
- Beslissen welke informatie uit de opgave relevant is voor het model.

2. Hoe verschilde jullie oplossing van de modeloplossing? Welke bijkomende veronderstellingen maakten jullie?

- Wij hadden een ‘loads’ relatie, deze was echter niet relevant voor het model.
- We hadden geen ‘Country’ entiteit, we gebruikten country als attribuut.
- We hadden geen ‘Trip’ entiteit, zelfs de attributen ‘Departure_date’ en ‘Arival_date’ hebben wij niet gebruikt.
- De plaatsing van de afgeleide attributen is verschillend. Er zijn ook enkele gevallen waar wij geen afgeleid attribuut hebben waar dat wel zou moeten.
- Het attribuut ‘Size’ is bij ons niet opgesplitst in meerdere onderdelen.
- Door het gebrek aan de attributen ‘Departure_date’ en ‘Arival_date’, maakten wij de veronderstelling dat een schip slechts één route kan hebben op een gegeven tijdstip.
- We veronderstelden ook dat een ‘Customer’ of een ‘Ship_broker’ slechts één adres kan opgeven.

3. Welke moeilijkheden ondervonden jullie tijdens deel 2 van de opdracht (d.w.z., de omzetting van het ER-diagram naar een relationeel model)? Wat was de reden voor deze moeilijkheden?

- We gebruikten de ‘Goes On’ relatie, dit is echter redundant omdat deze relatie de entiteit ‘Trip’ identificeert. Alle informatie in die relatie wordt dus ook weergegeven in de entiteit.

4. Welke moeilijkheden ondervonden jullie tijdens deel 3 van de opdracht (d.w.z., het ontwikkelen van een web-based front-end voor de database in PHP/HTML)?

- Het heeft wat tijd nodig om te wennen aan de – voor ons nieuwe – syntax van PHP en het debuggen verliep moeizaam.
- Het was niet evident om met meer dan twee mensen aan dezelfde code te programmeren.

5. Wat zijn de sterke punten van jullie design (het ER-model) en de implementatie?

- De mapper houdt interface en controller logica gescheiden.
- Door gebruik te maken van de web interface kan het scheepvaart programma op elke browser gebruiken en is dus platformonafhankelijk.

6. Hoe zou je het *design* en/of de *implementatie* kunnen verbeteren, vanuit het oogpunt van de database? Welke technieken uit de hoorcolleges zouden hierbij kunnen helpen?

- Toevoegen van zoeken op meerdere attributen in plaats van enkel op 'ssn'.

7. De opdrachten uit het tweede deel introduceerden gradueel de verschillende concepten voor het bevragen van een database vanuit PHP. Bedenk en beschrijf een mogelijke (creatieve) uitbreiding van de web-interface voor de scheepsvaartdatabase, die de verschillende concepten gebruikt en gegevens uit de database combineert.

- Om de database efficiënter toegankelijk te maken op mobiele systemen, zou een aparte app kunnen uitgewerkt worden.
- Extra multimedia toevoegen die de gegevens uit de database visualiseert: een kaart die de klanten localiseert, aangeduide scheepsroutes...