

Flashcard / Leerhulpmiddel

Probleemstelling

Als je als student flashcards gebruikt om te leren dan kan het zijn dat je bij grote vakken, of als je dit doet bij veel vakken, geen organisatie meer hebt tussen je flashcards en er soms een paar kwijtraakt. Dus zou een manier om flashcards te ordenen handig zijn.

Verder zijn flashcards vooral goed om theorie te leren, maar zijn ze minder goed om oefeningen te maken. Dit is omdat je de antwoorden van je geschreven oefeningen of theorie flashcards na verloop van tijd uit je hoofd leert, zonder dat je daadwerkelijk begrijpt of effectief weet wat het inhoudt. Een manier om een oefening in te geven en hier iedere keer wat variatie in te hebben (Bv. met variabele getallen bij een formule), zou ervoor zorgen dat je daadwerkelijk leert hoe je zo een type oefening moet oplossen. Je gaat dan niet het antwoord van die ene oefening van buiten leren.

Nog een nadeel aan fysieke flashcards is dat als je een vriend hebt die vraagt om je zelfgemaakte flashcard te mogen gebruiken, dan moet je ze afgeven of kopiëren. Beide zijn niet ideaal, maar met een digitale versie van je flashcards zou je deze zonder veel inspanning kunnen delen met meerdere vrienden.

Proefexamens maken is ook iets wat veel studenten doen om een gevoel te krijgen hoe goed ze de leerstof beheersen en als ze tijdig alle vragen zouden kunnen invullen. Maar het kan zijn dat je maar één of soms zelfs geen proefexamen krijgt. Dus een manier om zelf een proefexamen op te kunnen stellen is nooit nadelig om te hebben, want dan leer je beter omgaan met de tijdsdruk tijdens examens en kunt kijken als je de leerstof effectief beheerst. Natuurlijk, als je handmatig een proefexamen samenstelt neemt dit tijd in beslag, maar hierdoor verdwijnt ook de verrassing van welke vragen je gaat krijgen op dit proefexamen. Dus een manier om dit voor je te laten doen (op basis van je flashcards) zou beter zijn. Natuurlijk heb je zelf ooit je flashcards gemaakt en heb je dus al de vragen die op je gegenereerd examen komen ooit zelf gemaakt, maar als de student te veel flashcards heeft om ze moeilijk van buiten te leren of er heeft waar variabele elementen in zitten kan dit gegenereerd proefexamen toch nog een grote meerwaarde hebben. Het willekeurig kiezen van flashcards zorgt er ook voor dat het moeilijker is voor de student om het antwoord van buiten te weten.

Als we dan toch proefexamens integreren zou het natuurlijk ook goed zijn voor de professor of assistenten van een bepaald vak om zelf proefexamens op te stellen en deze dan ook vrij te geven op de software. Hier zou dan wel een manier moeten zijn om studenten en mensen van het onderwijsteam te kunnen onderscheiden.

URPS

Usability

- Kaart metafoor gebruikt om flash cards voor te stellen. Het metafoor zorgt ervoor dat de flashcards intuïtief en herkenbaar zijn voor gebruikers, zodat ze gemakkelijk begrijpen hoe ze de tool moeten gebruiken.
- Tooltips die mogelijke keuzes toelichten bij het maken van de flashcards. Dit kan mogelijk ook nog op andere plaatsen gehanteerd worden indien nodig.
- Diverse customizability (Kleuren, lettertypes)
- Consistente interface en terminologie, dit zorgt ervoor dat de gebruikers niet voor mogelijke verrassingen komen te staan als sommige elementen andere functionaliteiten hebben dan ze gedacht hadden.
- Interface moet aangenaam over komen en gebruikt worden voor lange tijd

Reliability

- Lage faalfrequentie van de server: 99% server uptime
- Er is een offline modus, indien de connectie verloren gaat
 - Online opgeslagen vragen of examens kunnen niet geraadpleegd worden
 - Nieuw gemaakte vragen kunnen enkel offline opgeslagen worden
 - Implementeer correcte foutafhandeling (zie hieronder) voor de gebruiker te verwittigen dat ze offline zijn
- Foutafhandeling: Implementeer duidelijke foutmeldingen die gebruikers helpen begrijpen wat er mis is gegaan en hoe ze het probleem kunnen oplossen.
- Herstelopties: Bied opties om per ongeluk verwijderde flashcards of groepen van flashcards te herstellen.
- Data-integriteit: Garandeer dat gegevens (flashcards) consistent en correct worden opgeslagen

Performance

- Responsieve interface met een maximum response tijd van 50ms

Supportability

- Gebruikers kunnen ondervragingen zelf configureren (Aantal vragen, tijdslimiet, ...)
- Consistente code en contracten (er is een vaste structuur) om onderhoudbaarheid te verhogen

Technologie

- GUI: Qt (C++)
- Programmeertaal: C++ , Python voor de web server
- Web Server: FastAPI
- Database: MySQL

Effort

GUI

- Flash card **(2 mandag)**
- Ondervraging **(1.5 mandag)**
 - Navigeren tussen vragen **(1 mandag)**
 - Input zekerheden **(0.5 mandag)**
- Ondervragingen beheren **(2 mandagen)**
- Customizability **(2 mandagen)**
 - Tijdslimiet vragen **(0.5 mandag)**
 - Aantal vragen **(0.5 mandag)**
 - Kleuren **(0.5 mandag)**
 - Lettertypes **(0.5 mandag)**
- Vragen delen **(2 mandagen)**
- Accounts (inloggen, beheren, ...) **(2 mandagen)**

Model

- Vragen kunnen creëren **(3 mandagen, kan uitgebreid worden met nieuwe vragen Bv. meerkeuze)**
- Ondervragingen genereren **(2.5 mandagen)**
- Systeem dat volgende vraag kiest **(2 mandagen, kan uitgebreid worden afhankelijk van complexiteit algoritme)**
- Vragen delen **(2.5 mandagen)**

Backend

- Database opzetten/configureren **(Worst case: 2 mandagen)**
- Server Backend (API en database communicatie) **(5 mandagen)**
- Accounts (registratie, inloggen, ...) **(1.5 mandag)**

Totaal aantal mandagen: 30 mandagen