

Leeruitkomst 1: Je baseert je keuzes op feedback van **stakeholders** en **onderbouwt** ze op een **heldere** en **professionele** wijze.

Deze leeruitkomst is te behalen door professioneel met de stakeholders in gesprek te gaan en hun eisen en feedback te noteren. Het is belangrijk dat je op basis van de feedback onderbouwde keuzes kunt maken binnen het project. De argumentatie moet ook met bronnen te onderbouwen zijn.

Vorig semester vond ik het lastig om de documentatie van het individuele project goed bij te houden, hier moet ik dus extra op letten. In de proftaak lukte dit wel goed.

Leeruitkomst 2: Je **werkt samen** en **communiceert** met anderen op **constructieve** en **professionele** wijze.

Het is belangrijk om initiatieven te tonen in de groep. Voor elke oplevering moet ik ervoor zorgen dat de deelproducten volledig zijn en af is wat met de opdrachtgever gecommuniceerd is. Ook is het handig om feedback van andere groepsleden te krijgen. Hierdoor weet ik mijn sterke en zwakke punten en kan ik mijn gedrag in de groep eventueel verbeteren.

De samenwerking in het vorige semester met de vorige proftaak groep ging goed en de communicatie was duidelijk. Tijdens deze proftaak wil ik proberen meer initiatief te tonen en vaker het woord te nemen tijdens meetings. Vorige keer vond ik deze dingen wat lastiger.

Leeruitkomst 3: Je documenteert **gevalideerde gebruikersspecificaties** voor applicaties en vertaalt deze in **correcte softwareontwerpen**.

Het is belangrijk om requirements en/of userstories op te stellen en deze te bespreken met de stakeholders. Zo is te bepalen welke requirements de hoogste prioriteit krijgen. Er moeten diagrammen en modellen gemaakt worden die relevant zijn voor het product. Ook is het belangrijk om het systeem goed te testen, zodat ik zeker weet dat het naar behoren werkt.

In het vorige semester heb ik requirements en userstories opgesteld en hierbij verschillende diagrammen en modellen gemaakt. Ik heb systeem niet getest. Ik moet ervoor zorgen dat ik hier dus nog wat extra aandacht aan ga besteden.

Leeruitkomst 4: Je bouwt, ontwerpt en **levert herhaaldelijk veilige** en **onderhoudbare** applicaties **op** (waarvan er tenminste één web-gebaseerd is) die verbinding maken met een database en gebruik maken van **OO-principes** en **standaard technieken** gebaseerd op gevalideerde gebruikerseisen.

Ik moet meerdere applicaties maken die kunnen verbinden met een database. Hiervan moet er minimaal een web gebaseerd zijn. De applicaties moeten gebaseerd zijn op standaard technieken en OO-principes. Hierbij kan ik rekening houden met S.O.L.I.D.. De applicatie moet getest zijn op fouten door middel van acceptatietesten.

Deze leeruitkomst heb ik vorig semester niet voldoende aangetoond. Ook hier heb ik geen acceptatietesten uitgevoerd. Ik moet ervoor zorgen dat ik dit nu beter onder de knie krijg.

Leeruitkomst 5: Je redeneert over **computationale uitdagingen** en implementeert **algoritmisches complexe problemen** in software.

Ik moet veelvoorkomende uitdagingen oplossen met verschillende technieken. Belangrijk is dat ik kan onderbouwen waarom ik een bepaalde techniek heb gebruikt om deze uitdaging op te lossen. Ik moet zorgen dat ik algoritmes implementeer zodanig dat ze rekening houden met de randvoorwaarden van de stakeholders.

Vorig semester heb ik voor deze leeruitkomst de Circustrein gemaakt. Dit semester komt hiervoor een nieuwe opdracht in de plaats en deze zal ik ook moeten maken. Vorige keer hielp het mij om van tevoren eerst een flowchart te maken, dus dat ga ik nu weer doen.

Leeruitkomst 6: Je **ontwerpt**, bouwt en **bevecht** een relationeel databasesysteem en integreert dit met een applicatie.

Er moeten database en software ontwerpen gemaakt worden. Hierbij moet duidelijk onderscheid gemaakt worden. De database moet veel op veel relaties bevatten. Er worden CRUD-operaties uitgevoerd op de data in de database.

Vorig semester heb ik een SQL database verbonden aan mijn webapplicatie. Hierbij heb ik entity framework gebruikt. Dit semester wil ik dit niet gebruiken, zodat ik het nog beter ga begrijpen.

Leeruitkomst 7: Je **verbetert** en **toont** de kwaliteit van je software **continu aan**, gebruikmakend van **standaard technieken en hulpmiddelen**.

Veranderingen kunnen bijgehouden worden met versiebeheer. Op die manier kun je altijd teruggaan naar een oudere versie van de software, wanneer er iets fout is gegaan. Een systeem waarbij dit handig kan is bijvoorbeeld Git. Voor alle veranderingen kan het handig zijn om een logboek bij te gaan houden.

Vorig semester heb ik Git gebruikt voor versiebeheer, zowel in het individuele project als in de proftaak. Dit werkte erg fijn en ik ben van plan dit weer te gebruiken.