Examenafspraken

Proeve van Bekwaamheid

|  |  |  |
| --- | --- | --- |
| **Algemene informatie** | | |
| Examenvorm | Proeve van bekwaamheid – Praktijkexamen in de beroepspraktijk | |
| Kwalificatiedossier en cohort | Software development | 2020 en verder |
| Profiel, niveau en crebocode | P1: Software developer,  niveau 4 | 25604 |
| Examencode | SD\_SD20-PE1\_B1-K1 & B1K2 | |
| Kerntaak B1K1 | Realiseert software | |
| Werkprocessen | B1-K1-W1 Plant werkzaamheden en bewaakt de voortgang B1-K1-W2 Ontwerpt software B1-K1-W3 Realiseert (onderdelen van) software B1-K1-W4 Test software B1-K1-W5 Doet verbetervoorstellen voor de software | |
| Kerntaak B1K2 | B1-K2 Werkt in een ontwikkelteam | |
| Werkprocessen | B1-K2-W1 Voert overleg B1-K2-W2 Presenteert het opgeleverde werk B1-K2-W3 Reflecteert op het werk | |
| Vaststellingsdatum | 27-10-2020 | |

|  |  |
| --- | --- |
| **Persoonsinformatie** | |
| Datum | 9/24/2024 |
| Naam kandidaat | Bjorn Molendijk |
| Studentnummer | 9019120 |
| Klas/groep | SOD3B |
| Praktijkbeoordelaar | Iemand van een andere school |
| Schoolbeoordelaar | Rob Wigmans |

# Algemeen

Dit is het document *Examenafspraken*. Gebruik voor het concretiseren van dit document de *Invulinstructie voor examenafspraken*.

|  |
| --- |
| **Referentieopdrachten** |
| Het examen neemt 40 uur in beslag. Voor dit examen worden de opdrachten door een team van zes personen uitgevoerd. In deze 40 uur wordt precies één sprint gedaan.  Als referentieopdrachten zijn de drie door TCR geleverde voorbeelden van examenafspraken gebruikt.  De kandidaat moet laten zien dat hij userstories kan schrijven, een gedeelte van een ontwerp kan maken, OOP kan programmeren en testen. Verder moet de kandidaat kunnen laten zien dat hij kan overleggen, presenteren en reflecteren met betrekking op het geleverde werk.  Voor het examen houden we deze dagverdeling aan:  Afbeelding met tafel  Automatisch gegenereerde beschrijving  De kandidaat wordt individueel beoordeeld. |

|  |  |
| --- | --- |
| **Team samenstelling** | |
| Bjorn Molendijk | Designer |
| Erim Koçak | Front-end & Back-end Developer |
| Lucas Tanis | Front-end & Back-end Developer |
|  |  |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
| **Concrete invulling examenopdrachten** | |
| *Beschrijf hier de gekozen invulling van de opdrachten, de context waarbinnen het examen plaatsvindt en de afnamecondities.* | |
| *Beschrijving situatie bedrijf & development team* | *Akkoord* |
| **Ministerie van Binnenlandse Zaken**  Voor het ministerie moeten verschillende mensen kunnen inloggen. Hierbij zijn er verschillende onderdelen die het ministerie moet kunnen:  • Het ministerie moet aangemelde partijen kunnen goedkeuren. Hier gelden natuurlijk wettelijke regels voor, maar dat hoeft niet in de applicatie te komen.  • Het ministerie kan een verkiezing uitschrijven. Alle verkiezingen die plaats vinden worden door het ministerie ingepland, zelfs de regionale verkiezingen. Bij het uitschrijven van de verkiezing worden de datums wanneer gestemd kan worden van tevoren bepaald. Als het regionale verkiezingen betreft krijgen de gemeenten een melding dat ze het proces voor uitnodigen moeten starten.  • Het ministerie moet het gehele proces hoe het stemmen gaat netjes in de applicatie kunnen weergeven.  • Zodra de datum van het stemmen voor een landelijke verkiezing is verlopen, kan de uitslag bekend worden gemaakt. Dit gaat niet automatisch, maar zal iemand van het ministerie zal deze moeten publiceren.  Voor de gemeente moeten verschillende mensen kunnen inloggen. Zodra het ministerie een verkiezing uitschrijft is de gemeente verantwoordelijk voor de uitnodigingen hiervoor. Dit geld voor landelijke en regionale verkiezingen. Zodra de datum van het stemmen voor een regionale verkiezing is verlopen, kan de uitslag bekend worden gemaakt. Dit gaat niet automatisch, maar iemand van de gemeente zal deze moeten publiceren Stemgerechtigden Een stemgerechtigde moet kunnen inloggen.  Vervolgens krijgt deze een overzicht van verkiezingen voor zich die actief zijn waarvoor nog een stem uitgebracht kan worden. Op dit scherm staan ook de 5 laatste verkiezingen die niet meer actief zijn. Er wordt wel geregistreerd dat er een stem is uitgebracht of niet, maar niet op wie je hebt gestemd.  Naast het overzicht kan je ook voor actieve verkiezingen je stem uit brengen, indien dit nog niet is gebeurd. Verkiesbare / Partijen Een partij kan zich altijd aanmelden in de applicatie.  Zodra er een partij is gemaakt moet er een koppeling zijn met de partij en de verkiesbare. Er is altijd 1 aanvrager voor een partij, die dan ook de administratie bevoegdheden krijgt over de partij.  Deze bevoegdheden moeten ook aan een andere verkiesbare gegeven kunnen worden.  Je kan je als verkiesbare aanmelden bij een partij, waarbij de administrator van de partij je kan accepteren als lid van de partij.  De administrator van de partij kan je ook weer verwijderen als lid van de partij. Voor elke verkiezing kan een volgorde bepaald worden welke verkiesbare op plek 1, 2, 3 etc. Bij de volgorde kunnen uitsluitend de verkiesbare van dat moment meegenomen worden voor de volgorde. De volgorde dient ook na de verkiezing beschikbaar te zijn. |  |
| *Beschrijving wat vooraf het examen al gemaakt is (eerdere sprints indien van toepassing)* | *Akkoord* |
| n.v.t |  |
| Beschrijving wat het development team in de sprint van het examen gaat ontwikkelen | *Akkoord* |
| • Registratie van alle verkiezing soorten • Registratie van politieke partijen per verkiezing soort • Registratie van verkiesbare • Koppeling van een verkiesbare aan een politieke partij • Volgorde bepalen van de stemlijst van een politieke partij • Registratie van stemgerechtigden • Het goedkeuren van een politieke partij door het ministerie. • Aanmaken van een verkiezing van een bepaalde verkiezingssoort, met daarbij de datum(s) wanneer er gestemd mag worden • Informatie pagina’s aanmaken en in het menu plaatsen • Uitslagen van een verkiezing publiceren door het ministerie (landelijke) of door gemeente (regionale) • Een inlog pagina waar elke gebruiker kan inloggen. Hierbij zit ook de optie om uit te loggen en wachtwoord te resetten. • Uitnodigingen versturen naar de stemgerechtigden • Percentages/hoeveelheden zien hoeveel van de stemgerechtigden van een stad/regio hebben gestemd.  • Het stemmen in een verkiezing door een stemgerechtigde. Je mag maximaal maar 1 keer per verkiezing kunnen stemmen. |  |
| *Welke apparatuur / tools worden gebruikt tijdens het examen* | *Akkoord* |
| Laptop en computers, speciale servers en accounts. We gebruiken als tools als git,visual studio code, framework laravel, teams voor communicatie. En voor het ontwerp gebruiken we figma. |  |

# B1-K1 & B1-K2

|  |  |
| --- | --- |
| **Examenopdracht 1** | |
| **B1-K1-W1: Plant werkzaamheden en bewaakt de voortgang** | |
| Beschrijf waar de Definition of Done staat | *Akkoord* |
| De Definition of Done staat in ons werkzaamheden en voortgangs map, deze kan bereikt worden door onze map in git,github |  |
| Beschrijf hoe je de eisen en wensen verwerkt in userstories. Hoe is de opbouw van je userstories | *Akkoord* |
| In een Github omgeving word een scrumboard gebruikt waarop alle userstories op de product backlog komen, de userstories hebben een opbouw in de vorm van “Als [gebruikersgroep], wil ik [functionaliteit], zodat ik [reden]”. |  |
| Beschrijf hoe en met wat je een planning maakt. Hoe geef je de prioriteiten aan. | *Akkoord* |
| Op ons scrumboard worden de userstories die in de sprint uitgewerkt worden naar de sprintbacklog verplaatst. Hierbij worden prioriteiten gesteld waarbij de bovenste userstory de meeste prioriteit heeft. |  |
| Beschrijf hoe je de voortgang bewaakt en hoe je dit bewijst. | *Akkoord* |
| Elke dag word er een stand-up gehouden, deze zullen wij elke dag bijhouden door middel van screenshots in de planning. |  |
| **B1-K1-W2 Ontwerpt software** |  |
| Beschrijf hoe je de userstories vertaald naar een ontwerp. (sitemap, wireframe) | *Akkoord* |
| Voor alle pagina’s die gemaakt gaan worden wordt er een wireframe gemaakt voor de pc versie. De wireframes die we maken worden in het ontwerpdocument opgeslagen. |  |
| Beschrijf welke schematechnieken gebruikt worden. (use-case diagram, klasse diagram, activiteitendiagram, flowchart, erd) | *Akkoord* |
| Om alles goed te begrijpen gaan we een klassendiagram maken, en voor de databases gebruiken wij een erd ontwerp. Deze worden uiteraard opgeslagen in ons ontwerpdocument. |  |
| Beschrijf hoe je de gemaakte keuzes onderbouwt, rekening houdend met ethiek, privacy en security | *Akkoord* |
| Een onderbouwing van de gemaakte keuzes wordt toegevoegd aan het ontwerpdocument. Hierbij geeft ik aan welke onderdelen te maken hebben met ethiek, privacy en security en hoe hier rekening mee wordt gehouden |  |
| **B1-K1-W3 Realiseert (onderdelen van) software** |  |
| Beschrijf met welke programmeertaal en/of framework je de software gaat realiseren. Geef aan welke technieken hierbij gebruikt worden (functioneel programmeren/object georiënteerd) | *Akkoord* |
| Voor de backend gebruiken wij het php framework Laravel. Voor de front-end gebruiken wij css. |  |
| Beschrijf welke code conventies je gaat gebruiken |  |
|  **PascalCase** - voor class- en modelnamen.   **camelCase** - voor methoden- en functienamen.   **snake\_case** - voor route-, database tabel- en kolomnamen.   **PSR-12** - voor code style.   **Meervoudige namen** - voor database tabellen.   **snake\_case** - voor view-bestandsnamen.   **Vier spaties** - voor inspringen. |  |
| Beschrijf hoe je gaat bewijzen dat versiebeheer goed is toegepast | *Akkoord* |
| Wij gebruiken GIT, hierin kunnen wij allemaal onze bestanden pushen en daarbij zetten wat we aangepast hebben. |  |
| **B1-K1-W4 Test software** |  |
| Beschrijf hoe je je testplan gaat maken op basis van de gemaakte userstories. Neem hierin mee hoe je aan testdata komt en omgaat met alternatieve scenario’s | *Akkoord* |
| We werken voor de testen van het project met acceptatie formulieren. |  |
| Beschrijf wat er in het testrapport komt | *Akkoord* |
| We vullen de acceptatie test in. En daar komt de test van elke scenario van onze codes |  |
| **B1-K1-W5 Doet verbetervoorstellen voor de software** |  |
| Beschrijf waar je de verbetervoorstellen vanuit de test vastlegt | *Akkoord* |
| Nadat conclusies zijn getrokken uit de test zal ik verbetervoorstellen vastleggen in het document verbetervoorstellen. De userstories of taken die eruit voortkomen worden vastgelegd op het scrumboard |  |
| Beschrijf waar je de verbetervoorstellen vanuit de presentatie vastlegt | *Akkoord* |
| Via een power point |  |
| Beschrijf waar je de verbetervoorstellen vanuit de reflectie (retrospective) vastlegt | *Akkoord* |
| Vanuit de reflectie worden verbetervoorstellen vastgelegd in het document verbetervoorstellen. Indien nodig kunnen aanpassingen gedaan worden aan de definition of done, definition of fun of het scrumboard |  |
| **B1-K2-W1 Voert overleg** |  |
| Beschrijf wanneer de overlegmomenten zijn | *Akkoord* |
| Elke les en sowieso na elke belangrijke push ook. |  |
| Beschrijf hoe de afspraken uit de overleggen worden vastgelegd | *Akkoord* |
| De afspraken uit de overleggen worden verwerkt op het scrumboard. Van de toegevoegde afspraken maak ik screenshots. |  |
| **B1-K2-W2 Presenteert het opgeleverde werk** |  |
| Beschrijf wanneer de presentatie is voor de opdrachtgever en welke (hulp)middelen je nodig hebt | *Akkoord* |
| 22 oktober, wij gebruiken PowerPoint voor het presenteren van onze applicatie |  |
| **B1-K2-W3 Reflecteert op het werk** |  |
| Beschrijf wanneer de reflectie(retrospective) plaats vindt en welke (hulp)middelen je nodig hebt | *Akkoord* |
| Plan: getekende examenafspraken Producten/documenten per werkproces: B1-K1-W1: Plant werkzaamheden en bewaakt de voortgang 1. Planning of Trello-board. 2. Logboek of Scrumboard (t.b.v. daily standup en vastleggen van afspraken) 3. Defintion of Done B1-K1-W2: Ontwerpt software 4. Functioneel ontwerp B1-K1-W3: Realiseert (onderdelen van) software 5. Broncode en export database B1-K1-W4: Test software 6. Testplan 7. Testrapport B1-K1-W5: Doet verbetervoorstellen voor de software 8. Verbetervoorstellen (uit test, presentatie en retrospective) B1-K2: Werkt in een ontwikkelteam B1-K2-W1: Voert overleg . 9. Logboek met een verslag van de daily standup en de gemaakt afspraken. B1-K2-W2: Presenteert het opgeleverde werk 10. Video-opname (mp4) B1-K2-W3: Reflecteert op het werk 11. Verslag reflectiemeeting (retrospective) |  |
| Uiteindelijk Resultaat Als resultaat van deze opdracht lever je de volgende producten en/of diensten op. | |
| **Aanvullende afspraken** | |
| De bedrijfsexaminator heeft tijdens het examen de observatieformulieren ingevuld en indien mogelijk daarna ook de beoordelingsformulieren. Daarna bel ik de schoolexaminator op in Teams (of ik stuur hem een mail) om hem te vertellen dat hij een afspraak met de bedrijfsexaminator kan maken voor de eindbeoordeling. Ik maak een zip-bestand van al mijn ingeleverde werk (voor zover het BPV-bedrijf daarmee akkoord gaat) en lever dat op de met de schoolbeoordelaar afgesproken manier in. Tijdens de eindbeoordeling laat ik aan beide beoordelaars zien wat ik gemaakt en beantwoord ik hun vragen | |
| **Periode, beschikbare tijd en afgesproken beoordeelmomenten** | |
| Eind van het jaar en we hebben vernomen dat de beoordelaar af en toe meekijkt maar beoordeelt word aan het einde van het school jaar | |

|  |
| --- |
| ***Beoordeling van examen*** |
|  |

# Start en einde examen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **B1-K1 & B1-K2** | | | | |
| Examenduur | 24-09-2024 | 12:00 | 22-10-2024 | 12:00 |
| Handtekening Kandidaat | nvt | | Nvt | |
| Handtekening  Praktijkbeoordelaar |  | |  | |
| Handtekening Schoolbeoordelaar |  | |  | |