# Marvel Rivals Stats Search

Afbeelding met pc-game, animatie, Cg-kunstwerk, fictie

Door AI gegenereerde inhoud is mogelijk onjuist.

Introductie

Ik heb als project gekozen om een Marvel Rivals stats tracker te maken. Dit wil ik uiteindelijk als een user friendly website hebben dat ook modderen en overzichtelijk er uit komt te zien. Voor dit project is het belangrijk dat ik de goed API gebruik zo dat ik de correcte gegevens makkelijk kan terug vinden. Voor de softwarepakket ga ik kiezen voor Javascript in Webstorm. In dit verslag zelf ge ik je meer vertellen over het ontwerp van het website, de realisatie van het website, de testen dat er zijn uitgevoerd, mijn motivatie voor het maken van het website en als last mijn reflectie van mijn project. Het doel was om een website te maken waar je de game-statistieken kunt zien van de hit game Marvel Rivals. Ik heb eerst gekeken wat belangrijk is voor de gebruiker. Daar heb ik user stories van gemaakt.

Calistus Michael 169268

Inhoudsopgave

[Marvel Rivals Stats Search 1](#_Toc198276420)

[B1-K1 Realiseert Software 3](#_Toc198276421)

[Eisen en wensen 3](#_Toc198276422)

[De eisen en wensen zijn verwerkt in de user stories. 3](#_Toc198276423)

[De user stories voldoen aan de criteria (wie, wat, waarom en realistisch). 3](#_Toc198276424)

[Ontwerp 4](#_Toc198276425)

[De user stories zijn vertaald naar een passend, eenduidig en volledig ontwerp (sluit aan op wensen en eisen). 4](#_Toc198276426)

[Er is gebruikgemaakt van relevante of toepasselijke schematechnieken (bijv. activiteitendiagram, klassendiagram, ERD, use case diagram). 4](#_Toc198276427)

[De gemaakte keuzes in het ontwerp zijn onderbouwd met steekhoudende argumenten, waarbij rekening is gehouden met bijv. ethiek, privacy en security. 4](#_Toc198276428)

[Realisatie 13](#_Toc198276429)

[Er is voldoende inhoud van de user stories gerealiseerd binnen de gestelde/geplande tijd. 13](#_Toc198276430)

[De opgeleverde functionaliteiten voldoen aan de eisen en wensen zoals omschreven in de betreffende user story. 13](#_Toc198276431)

[De kwaliteit van de code is goed. Dit uit zich onder andere in: OOP, objectstructuur, MVC, validatie, efficiëntie, foutafhandeling en terugkoppeling, security (veilig programmeren). 13](#_Toc198276432)

[De code is gestructureerd opgesteld volgens code conventions. 13](#_Toc198276433)

[De code is verzorgd, leesbaar, gestructureerd en voorzien van zinvol commentaar. 13](#_Toc198276434)

[Versiebeheer is effectief toegepast. 13](#_Toc198276435)

[De voortgang is bewaakt en de juiste keuzes/afwegingen zijn gemaakt op basis van prioriteiten. 13](#_Toc198276436)

[Testen 14](#_Toc198276437)

[De testcases in het testplan sluiten aan op de user stories en bevatten alle scenario's. 14](#_Toc198276438)

[De stappen, het gewenste resultaat en testdata zijn benoemd. Niet alleen het hoofdscenario, maar ook alternatieve scenario’s. 14](#_Toc198276439)

[Alternatieve paden zijn opgenomen in de scenario’s. 14](#_Toc198276440)

[Het testrapport bevat de juiste resultaten en conclusies. 14](#_Toc198276441)

[Het proces van het testen. 14](#_Toc198276442)

[Verbetervoorstellen 16](#_Toc198276443)

[De juiste verbetervoorstellen zijn gedaan vanuit het testen. 16](#_Toc198276444)

[De juiste verbetervoorstellen zijn gedaan vanuit de oplevering. 16](#_Toc198276445)

[De juiste verbetervoorstellen zijn gedaan vanuit de reflectie. 16](#_Toc198276446)

[B1-K2: Werkt in een ontwikkelteam 17](#_Toc198276447)

[Overleggen 17](#_Toc198276448)

[De kandidaat neemt actief deel waarbij relevante onderwerpen worden ingebracht en de juiste vragen worden gesteld. 17](#_Toc198276449)

[De kandidaat stemt regelmatig en tijdig af met projectteamleden en opdrachtgever over de voortgang en eventuele knelpunten. 17](#_Toc198276450)

[De gemaakte afspraken zijn eenduidig vastgelegd. 17](#_Toc198276451)

[De kandidaat houdt zich aan gemaakte afspraken. 17](#_Toc198276452)

[(Deel)oplevering 18](#_Toc198276453)

[De kandidaat presenteert een overtuigend, duidelijk, beargumenteerd verhaal, afgestemd op de doelgroep. 18](#_Toc198276454)

[De kandidaat stelt gerichte vragen om te controleren of de betrokkenen goed geïnformeerd zijn over het opgeleverde werk. 18](#_Toc198276455)

[De kandidaat reageert adequaat op feedback. 18](#_Toc198276456)

[Reflecteren 19](#_Toc198276457)

[Reflecteert op het proces en resultaat. 19](#_Toc198276458)

# B1-K1 Realiseert Software

## Eisen en wensen

### De eisen en wensen zijn verwerkt in de user stories.

|  |
| --- |
| user stories |
| * Als gebruiker wil ik mijn wins en losses kunnen zien.   + Prio: Hoog, 4 dagen * Als gebruiker wil ik mijn kills en deaths per game kunnen zien zodat ik weet hoe goed ik speel.   + Prio: Hoog, 4 dagen * Als gebruiker wil ik zien welke Hero ik heb gespeelt voor die game.   + Prio: Laag, 1 dagen * Als gebruiker wil ik zien hoelang ik een game hebt gespeeld.   + Prio: Laag, 1 dagen * Als een gebruiker wil ik een zoekbalk hebben waar in ik iedereen kan opzoeken.   + Prio: Hoog, 2 dagen |

### De user stories voldoen aan de criteria (wie, wat, waarom en realistisch).

De user stories zijn duidelijk en haalbaar. In elke user story staat wie iets wil, wat die persoon wil doen, en waarom. Ze zijn ook niet te groot, zodat ik ze makkelijk kon inplannen en uitvoeren. Een voorbeeld: "Als gebruiker wil ik mijn kills en deaths kunnen zien zodat ik weet hoe goed ik speel." Deze was haalbaar binnen de tijd en belangrijk voor de speler.

## Ontwerp

### De user stories zijn vertaald naar een passend, eenduidig en volledig ontwerp (sluit aan op wensen en eisen).

Op basis van de user stories heb ik drie belangrijke pagina’s ontworpen. De homepage bevat een zoekbalk waarmee je de speler kunt zoeken. Als je de speler niet kunt vinden, kun je via een aparte UID-pagina jouw unieke code opvragen. Als je een speler hebt gevonden, ga je naar een pagina met de statistieken van die speler. Deze indeling sluit goed aan bij wat de gebruiker nodig heeft.

### Er is gebruikgemaakt van relevante of toepasselijke schematechnieken (bijv. activiteitendiagram, klassendiagram, ERD, use case diagram).

Ik heb een ontwerp gemaakt in Krita. Krirta is een gratis programma waar in je tekeningen kan maken, je kan het vergelijken met Photoshop. Daarin heb ik laten zien hoe de pagina’s eruit zouden zien. Dit hielp mij bij het bouwen van de website. Ook heb ik feedback gevraagd aan een vriend. Op basis van zijn tips heb ik het ontwerp duidelijker gemaakt, bijvoorbeeld door knoppen groter te maken en de zoekbalk beter zichtbaar te plaatsen.

Afbeelding met tekst, Post-it-briefje, Lettertype, handschrift

Door AI gegenereerde inhoud is mogelijk onjuist.

### De gemaakte keuzes in het ontwerp zijn onderbouwd met steekhoudende argumenten, waarbij rekening is gehouden met bijv. ethiek, privacy en security.

Ik gebruik alleen data uit een openbare API. Ik sla geen persoonlijke gegevens op. De gebruiker hoeft geen account aan te maken en er wordt niets opgeslagen. Ook controleer ik invoer in de zoekbalk, zodat er geen verkeerde data of scripts kunnen worden ingevoerd. Dit zorgt voor meer veiligheid.

|  |
| --- |
| User story  Als een gebruiker wil ik een mooi modern website hebben dat ook overzichtelijk is met duidelijke navigatie en zoekbalk. |
| Ontwerpkeuzes  Ik wilde een achtergrond met veel van de heros die in-game zijn en een achtergrond die ook officieel door Marvel Rivals gemaakt is. Ik heb voor dit achtergrond gekozen omdat ik met dit achtergrond goede keuzen kan maken voor de kleuren voor de design van de website. De keuren dat ik ga gebruiken voor de design zijn verschillende blauw, omdat het goed de achtergrond complementeert HEX #87CEEB, #6CB4EE. Ik heb als een accent kleur voor zwart gekozen omdat het goed neutraal keur is ik gebruik het accent kleur voor de navigatiebar en knoppen. |
|  |

|  |
| --- |
| User story  Als een gebruiker wil ik een UID invullen van een player en dan op search klikken. Waar ik vervolgens de informatie van de player zie. |
| Ontwerpkeuzes  Voor het zoek balk moet je UID gebruiken dit is een uniek ID dat gekoppeld is aan jouw account. Voor de API dat ik gebruik je moet dit gebruiken om je gegevens terug te vinden. Omdat het niet duidelijk is dat je dat moet gebruiken heb ik voor dat je iets invult duidelijk uitgelegd dat je een UID moet invullen.  Het normaal zoek bal vond ik niet mooi dus ik heb voor mijn zoekbalk Bootstrap gebruikt om het te maken als hoe ik het wil hebben. Ik heb de knop dat bij de search bar is verandert zodat het meer overeen komt met de website design. |
| Afbeelding met tekst, Animatie, fictie, schermopname  Door AI gegenereerde inhoud is mogelijk onjuist. |
| Ontwerpkeuzes  Het kan zijn dat iemand hier niet op let en zijn in-game naam invult. In dit geval heb ik een error code op scherm laten zien zodat het meer duidelijk is dat de persoon UID moet invullen en waar de persoon dit kan vinden. |
| Afbeelding met tekenfilm, fictie, schermopname, Animatie  Door AI gegenereerde inhoud is mogelijk onjuist. |

|  |
| --- |
| User story  De gebruiker kan zijn UID niet vinden. Wat moet ik nu doen? |
| Ontwerpkeuzes  Ik heb naast de error code ook een extra pagina gemaakt met foto’s dat duidelijk laat zien waar de player zijn UID kan vinden. Ik heb ook een witte achtergrond toegevoegd om de tekst meer leesbaar te maken. Ik heb bij de achtergrond een opacity toegevoegd zodat je de oorspronkelijke achtergrond ook ziet. Je kan op dit pagina terecht komen door te klikken op Try this of bij de navbar te klikken op Find your UID. |
|  |

|  |
| --- |
| User story  Als een gebruiker wil ik zien   * hoeveel matches je hebt gespeelt, * hoeveel je er van hebt gewonnen, * hoeveel je er van hebt verloren. * je win rate per % is van de player dat ik hebt opgezocht. * welke Hero de persoon hebt gespeelt, * of de player hebt gewonnen of verloren, * KDA van de player, * scoren van de player, * welke map er op gespeeld is, * hoelang een potje hebt geduurd. * als laats een foto van de map. |
| Ontwerpkeuzes  Om het duidelijk en overzichtelijk te maken heb ik er voor gekozen om de gegevens in 2 delen te splitsen de eerste deel zie je hoeveel matches de player hebt gewonnen, hoeveel matches de player hebt verloren en je win rate per %.  Voor het tweede deel zie je welke Hero de persoon hebt gespeelt, of de player hebt gewonnen of verloren, KDA van de player, scoren van de player, welke map er op gespeeld is, hoelang een potje hebt geduurd en als laats een foto van de map. |
| Ontwerpkeuzes  Voor het eerste deel van het gegevens heb ik gekozen om het in een goede volgorden te zetten het eerste wat je gelijk ziet is Player Information hier in zie je jouw in-game naam en op welke season we momenteel zitten. Daarna zie je Unranked hierin zie je hoeveel potjes je hebt gespeelt, hoeveel je hier van hebt gewonnen, hoeveel je hier van hebt verloren en als last de winrate per %. Vervolgens zie je Ranked met de zelfde informatie als Unranked maar de informatie voor Ranked. |
|  |
| Ontwerpkeuzes  Bij de tweede deel van het gegevens heb ik er voor gekozen om de matches in verschillende blokjes te zaten zien in plaats van een groot block waar alle informatie op staat. Dit heb ik gedaan met het gebruiken van forEach.  Border kleur voor win en loss  Ik heb een border toe gevoegd aan de Player Informatie om makkelijker te zien of je hebt gewonnen of verloren dit zien je aan de kleur bij de border. Dit vond ik een mooi en modern oplossing. Ziens ik al via de data kan zien of je hebt gewonnen of verloren was het makkelijk om dit er bij toe te voegen.    Problemen tijden het maken van Player Informatie  De opmerkelijk probleem dat ik tegen kwam bij Player Informatie was het laten zien van hero en de mapen omdat bij de data geen informatie hier over was. Dus ik heb lokaal een archief gemaakt voor elke hero en map. Bij de hero gedeelte is er dus hero\_id, naam en de icon opgeslagen in een Javascritp. Bij de map is de ModeID, ModeName, MapName, SubMapID opgeslagen in een Javascritp. Bij de data dat ik request was er Hero\_id en ModeID die mee kon ik de hero en de map laten zien. ik wist niet hoe je dit moest opslaan zodat je het later terug kan vinden en kan gebruiken dus ik heb de help van ChatGPT hier voor gebruikt. Dit was een goede keuzen want ik was bang dat ik een Database in SQL voor dit moest gaan gebruiken maar na het uitleggen aan ChatGPT heb ik gevonden dat ik het kan opslaan in Javascript en dit kan requesten om de gegevens weer terug te halen.    Volgorder van informatie in Player  De volgorder van informatie in Player Informatie heb ik gekozen om het in een goed en leesbaar volgorden te maken. Wat je al eerst merkt is of je de potje hebt gewonnen of verloren dit merk je door de border. Vervolgens zie je welke hero je voor dat potje hebt gespeelt. Daarna voor de verzekering als de border niet werkt zie je of je het potje hebt gewonnen of verloren. Dan zie je jouw KDA voor hoe je in het potje hebt gedaan. Daarna de score, mode en hoelang het potje was. Als laatst zie je op welke map je hebt gespeelt.  Kleine animatie op Matches  Om de Matches Informatie uniek te maken heb ik een kleine animatie toegevoegd wanneer je er met je muis er over gaat. Dit is waar ik op de idee kwam om animatie toe te voegen om alle Matches Informatie te laten zien.    Kleuren  De keuren voor de design voor de Player Informatie heb ik kozen voor (HEX #87CEEB, #6CB4EE ) blauwe kleuren omdat het goed met de achtergrond past. Om het design ook modern te houden heb ik voor een ronde border gekozen voor hero’s en een rechthoek voor de map’s. niet alle foto’s hadden de zelfde grootte dus ik heb er een width en height toegevoegd. Als een tweede kleur voor Player Informatie heb ik gekozen voor wit dit gebruik ik bij de border van de foto’s.  Animatie  Omdat ik forEach heb gebruikt had ik later problemen gehad met de animatie.  De probleem met animatie is dat alles Player Informatie de zelfde hoogte gebruikt dus je ziet alleen jouw laste match omdat de andere matches achter de laste match zijn. Dit is een probleem dat ik heb opgelost met behulp van ChatGPT. Dit was op gelost na dat ik dit hebt toegevoegd aan de code.  Afbeelding met tekst, Lettertype, schermopname  Door AI gegenereerde inhoud is mogelijk onjuist.  De probleem met CSS is dat het informatie soms buiten het scherm was. Dit lag aan de grootte van je scherm. Dit heb ik kunnen oplossen met het toevoegen van min-width. Hier door had het Player Informatie de correcte breedte. |
|  |
| Animatie voor Player Information en Matches  Tijden het maken van Matches gedeelte deed ik een kleine animatie om het uniek te maken. Hier door heb ik het idee gekregen om dit ook te doen wanneer je de Matches wilt zien. Ik heb de volgende stappen genomen eerste heb ik een animatie toegevoegd om de Player Information naar links te brengen.    Vervolgens wilde ik ook animatie toevoegen voor Matches. Hiervoor wilde ik dat alle Matches naar boven komen. Maar omdat ik forEach had gebruikt had ik hier wat moeite mee zoals uitgelegd hier boven. Na dat het op gelost was, was het best simple.    Nu had ik ook een back button toegevoegd en moest ik er voor zogen dat het Player Information weer naar rechts gaat zodat het weer als oorspronkelijk in zijn positie staat. Hier mee had ik was moeite want ik moest er ook voor zogen dat Matches wordt als hidden worden gezet en dat het de animatie triggert. Niet alleen dat ik moest er ook voor zogen als iemand Matches weer laat zien en weer weg wilt halen dat het animatie het weer triggert. Na een beetje testen was het uiteindelijk gelukt. De problemen dat ik hier tegen kwam was voor namelijk dat de style het niet hellemaal opsloeg zo als ik het wilden. |
|  |

## Realisatie

### Er is voldoende inhoud van de user stories gerealiseerd binnen de gestelde/geplande tijd.

Ik heb de belangrijkste user stories kunnen uitvoeren, maar ik was wel iets te laat klaar. In het begin had ik moeite met het vinden van een goede API voor de game Marvel Rivals. Daarna had ik problemen met het stijlen van de website. Hierdoor duurde het allemaal wat langer dan gepland. Toch heb ik alle hoofdfuncties kunnen maken.

### De opgeleverde functionaliteiten voldoen aan de eisen en wensen zoals omschreven in de betreffende user story.

De functies werken zoals ik in de user stories had bedacht. Je kunt een speler zoeken, zijn statistieken bekijken en via de UID-pagina je unieke code vinden. De gebruiker kan op een makkelijke manier bij zijn gegevens komen, precies zoals de bedoeling was.

### De kwaliteit van de code is goed. Dit uit zich onder andere in: OOP, objectstructuur, MVC, validatie, efficiëntie, foutafhandeling en terugkoppeling, security (veilig programmeren).

De code is verdeeld in duidelijke stukken. Ik gebruik functies om taken te scheiden, en ik controleer altijd of de invoer goed is. Als iets niet goed werkt, bijvoorbeeld als een speler niet gevonden wordt, krijgt de gebruiker een foutmelding. Ik heb gelet op veiligheid en duidelijke opbouw.

### De code is gestructureerd opgesteld volgens code conventions.

Ik heb de code netjes gehouden door vaste namen te gebruiken voor bestanden en functies. Ook heb ik gewerkt met comments om uit te leggen wat stukjes code doen. Alles staat op een logische plek, zodat ik of iemand anders het later makkelijk kan begrijpen. Alleen wat iemand niet gelijk kan op is mijn CSS.

Afbeelding met tekst, schermopname, software, Multimediasoftware

Door AI gegenereerde inhoud is mogelijk onjuist.

Afbeelding met tekst, schermopname, software

Door AI gegenereerde inhoud is mogelijk onjuist.

### De code is verzorgd, leesbaar, gestructureerd en voorzien van zinvol commentaar.

Ik heb geprobeerd om de code zo duidelijk mogelijk te schrijven. Ik gebruik witregels, goede inspringing en commentaar om de structuur leesbaar te houden. Dat maakt het makkelijker om fouten te vinden en aanpassingen te doen.

Afbeelding met tekst, schermopname

Door AI gegenereerde inhoud is mogelijk onjuist.

### Versiebeheer is effectief toegepast.

Ik heb geen versiebeheer zoals Git gebruikt. In plaats daarvan heb ik steeds lokale back-ups gemaakt van mijn project. Als er iets fout ging, kon ik terug naar een vorige versie.

### De voortgang is bewaakt en de juiste keuzes/afwegingen zijn gemaakt op basis van prioriteiten.

Ik had wat vertraging doordat ik eerst moeite had met het vinden van een goede API. Daarna liep ik vast op de vormgeving van de website. Toen heb ik besloten om me eerst te richten op de belangrijkste functies. Daarna heb ik gewerkt aan de vormgeving. Op die manier had ik in elk geval een werkend product

## Testen

### De testcases in het testplan sluiten aan op de user stories en bevatten alle scenario's.

Tijdens het bouwen heb ik nagedacht over verschillende situaties die gebruikers kunnen tegenkomen. Bijvoorbeeld: wat gebeurt er als iemand een verkeerde spelernaam of een verkeerd nummer invoert? Ook heb ik getest of de UID-pagina goed laat zien waar je je speler-ID kunt vinden. Deze testgevallen hielpen mij om fouten te vinden en op te lossen.

### De stappen, het gewenste resultaat en testdata zijn benoemd. Niet alleen het hoofdscenario, maar ook alternatieve scenario’s.

Ik heb geen aparte documenten gemaakt van de testen, maar ik heb tijdens het bouwen steeds getest of alles werkte. Als ik een fout vond, probeerde ik die meteen te verbeteren. Zo hield ik het project werkend terwijl ik eraan werkte.

### Alternatieve paden zijn opgenomen in de scenario’s.

Ik heb getest met juiste en onjuiste invoer. Als iemand een verkeerde speler-ID invult, krijgt hij een foutmelding en wordt hij doorgestuurd naar de UID-pagina. Daar kan hij zien waar hij zijn echte speler-ID kan vinden. Ook heb ik getest of de zoekfunctie werkt met verschillende soorten invoer. Verder heb ik ook getest of het zoekfunctie zal werken als je op enter drukt in plaats van het zoek knop. Je hebt internet connectie nodig om mijn website te gebruiken. Las je de tabblad sluit terwijl je zoekt moet je weer vanaf het begin zoeken. Dit komt omdat je niet hoeft in te loggen en omdat er niet wordt op geslagen.

### Het testrapport bevat de juiste resultaten en conclusies.

Tijdens het testen heb ik fouten direct opgelost. Een voorbeeld: bij een verkeerde invoer kreeg je eerst geen reactie. Ik heb toen een foutmelding toegevoegd en ervoor gezorgd dat je automatisch wordt doorgestuurd naar de uitlegpagina over de speler-ID.

### Het proces van het testen.

Ik heb aan het begin eerst getest of ik gegevens van het API zelf krijg dit ik heb laten zien in mijn console om te controleren of de gegevens wel kloppen. Vervolgens heb ik heel simpel maar verwarrend de gegevens in een blok gezet in het startscherm.

Nu dat ik zeker wist dat mijn API werkt ben ik begonnen aan het maken en testen van mijn zoekbalk. Ik heb wat je zocht op de console laten zien zo dat ik makkelijk kan kijken of het werkt.

Na dat het zoekbalk werkte heb ik dat verbonden met het API zelf. De gegevens dat je naar zoekt wordt laten zien in je console en je ziet niets op het scherm zelf. Na dat het werkte heb ik een code gemaakt als je zoek term correct is maakt het een blok aan en laat hij het gegevens daar in zien.

Bij het zoekbalk heb ik ook getest met een naam invoegen in plaats van UID invullen. Omdat er toen geen error code was heb ik een toegevoegd en dit ge koppelt met het zoekbalk. De error code zie je alleen wanneer er geen data krijgt van het API zelf. Dus ook als je verzonnen nummers invoert krijg je een error code te zien. In dit error laat ik zien waar je de UID kan vinden zodat iemand het makkelijk kan vinden.

Nu dat hij de gegevens laat zien heb ik mijn gegevens vergelijkt met de gegevens van andere website die game stats laat zien van Marvel Rivals of het wel klopt.

Ik heb rond welke 1 of 2 dagen een back-up gemaakt, maar soms moest ik er een eerder maken omdat ik veel liep te veranderen. Op dit manier had ik een extra verzekerdheid als ik iets had verwijdert en het niet meer terug kon gaan.

Bij het testen zelf liep ik best vaak tegen error codes aan. Gelukkig zegt het meestal in welke regel het code niet goed is dus kon ik het goed oplossen en als het niet lukte heb ik ChatGPT gebruikt om het error optelossen.

Test resultaat van User story

• Als gebruiker wil ik mijn wins en losses kunnen zien.

* Prio: Hoog, 4 dagen

Omdat ik probleem had met een goede API vinden is het langer geduurd dan verwacht. Na dat ik het API te werkend had gekregen was dit de eerste informatie dat ik heb laten zien. Periode 6 dagen

• Als gebruiker wil ik mijn kills en deaths per game kunnen zien zodat ik weet hoe goed ik speel.

* Prio: Hoog, 4 dagen

Omdat ik de API eerder werkend had gekregen was dit makkelijk te coderen. Het was niet heel mooi dus heb ik wat tijd gewerkt aan het verbeteren. Niet alleen dit maar ik heb ook een animatie toegevoegd aan dit, waardoor ik hier mee 12 dagen aan gewerkt hebt.

• Als gebruiker wil ik zien welke Hero ik heb gespeelt voor die game.

* Prio: Laag, 1 dagen

In het data dat ik van het api krijg zie je niet welke hero jij hebt gekozen maar ik kreeg wel een hero\_id. ik heb zelf de data in een Javascript gezet waar in je de gegevens kan terug halen dit kosten 4 dagen aan tijd.

• Als gebruiker wil ik zien hoelang ik een game hebt gespeeld.

* Prio: Laag, 1 dagen

Omdat dit al in de data van het API is was het makkelijk te laten zien. dit heeft minder dan 1 dag geduurd.

• Als een gebruiker wil ik een zoekbalk hebben waar in ik iedereen kan opzoeken.

* Prio: Hoog, 2 dagen

Bij het zoekbalk heb ik ook getest met een naam invoegen in plaats van UID invullen. Omdat er toen geen error code was heb ik een toegevoegd en dit ge koppelt met het zoekbalk. De error code zie je alleen wanneer er geen data krijgt van het API zelf. Dus ook als je verzonnen nummers invoert krijg je een error code te zien. In dit error laat ik zien waar je de UID kan vinden zodat iemand het makkelijk kan vinden. Voor dit gedeelte heb ik 4 dagen over gedaan.

## Verbetervoorstellen

### De juiste verbetervoorstellen zijn gedaan vanuit het testen.

De gegevens van een Player is niet goed te lezen zet het netjes en in een goede volgorden neer zodat iemand het goed kan lezen.

Afbeelding met tekst, schermopname

Door AI gegenereerde inhoud is mogelijk onjuist.

### De juiste verbetervoorstellen zijn gedaan vanuit de oplevering.

### De juiste verbetervoorstellen zijn gedaan vanuit de reflectie.

Achterafgezien had ik het beter met GitHub kunnen werken. Ook al was GitHub niet handig als ik grootte veranderingen in een keer wilde maken zoals het startscherm van de website vanaf het begin wilden bouwen en een oude back-up wilden hebben zonder kleine veranderingen. Hier voor zou ik toch het best kunnen gebruik maken van lokaal back-up voor grote veranderingen en GitHub voor kleine veranderingen.

# B1-K2: Werkt in een ontwikkelteam

## Overleggen

Om het project te beginnen moest ik een voorstel maken aan mijn docent hier in heb ik uitgelegd dat ik graag een stats search wil maken voor de gloed nieuwe hit game Marvel Rivals. Er was gevraagd wat ik hier in wil laten zien dit was dan de basis gegevens je kan denken aan wins, loss en hoe de players game is gegaan. Ik wilde het project met iemand anders maken maar omdat de genen naar stage ging kwam het helaas niet goed uit.

## (Deel)oplevering

### De kandidaat presenteert een overtuigend, duidelijk, beargumenteerd verhaal, afgestemd op de doelgroep.

De game Marvel Rivals is een gloednieuw en heel populair game waar miljoenen mensen spelen. Tijden het maken van het project waren/zijn er niet zo veel website om jouw in-game stats te zien. Ik wilden een mooie website maken waar de gegevens duidelijk zichtbaar zijn. Ik wilde ook dat iemand niet hoeft in te loggen om gegevens te kunnen zien. Verder wilde ik ook dat een gebruiker iedereen dat hij maar wilde kan opzoeken.

### De kandidaat stelt gerichte vragen om te controleren of de betrokkenen goed geïnformeerd zijn over het opgeleverde werk.

In overleg met de docent kwam ik bij de volgende vraag: welke informatie wil ik laten zien. Welke programeer taal ik zou hebben gebruikt.

### De kandidaat reageert adequaat op feedback.

Voor mijn feedback heb ik de meningen van mijn meden student gevraagd. Omdat ik meerdere feedbacks had was het makkelijk om een keuzen te maken of ik het wel echt wilden toepassen. Hier door heb ik mijn website meer user friendly kunnen maken. Want door de feedback kwam ik op het idee om de gegevens op een goed en leesbaar volgorden neer te zetten. Niet alleen dat hierdoor kwam ik ook op het idee om een terug knop te maken. Het aller belangrijkst feedback was waar iemand zijn UID kan terug vinden. Hier door heb ik een nieuwe pagina gemaakt met duidelijke uitleg over waar je het kan vinden.

## Reflecteren

### Reflecteert op het proces en resultaat.

Ik kijk tevreden terug op het eindresultaat van mijn project. De belangrijkste functies werken goed en de website doet wat hij moet doen: game-statistieken tonen van *Marvel Rivals*. Het proces zelf ging niet altijd makkelijk. Ik had moeite met het vinden van een goede API en ik liep vast op de vormgeving van de site. Daardoor was ik later klaar dan ik wilde. Maar ik heb doorgezet en steeds oplossingen gezocht. Uiteindelijk heb ik alles werkend gekregen.

Geeft aan wat goed ging en wat beter kon.

Wat goed ging, was het technisch werkende maken van de functies. De API koppelen en de statistieken tonen lukte uiteindelijk goed. Ook was mijn code netjes en overzichtelijk. Wat beter kon, was mijn planning. Ik had meer tijd moeten nemen om van tevoren de juiste API te kiezen. Ook had ik eerder hulp kunnen vragen bij het stijlen van de website, zodat ik minder tijd kwijt was aan dat deel.

Geeft aan wat hij/zij geleerd heeft.

Ik heb geleerd hoe ik data uit een API haal en hoe ik dat toon op een website. Ook heb ik geleerd hoe belangrijk het is om op tijd te beginnen en een goede planning te maken. Verder heb ik geleerd om problemen zelf op te lossen, maar ook dat het soms beter is om hulp te vragen. Als een voorbeeld voor dit is toen het Show Match Info het niet goed deed omdat ik forEach had gebruikt. Ik had hier voor de hulp nodig gehad van een docent.

Geeft aan wat hij/zij de volgende keer anders zou doen.

De volgende keer zou ik meer tijd nemen voor het zoeken van de juiste API. Ook zou ik eerder beginnen met het testen van de vormgeving. Verder wil ik dan meteen met versiebeheer zoals Git werken, zodat ik makkelijker kan teruggaan als iets fout gaat. En ik wil sneller feedback vragen als ik ergens op vastloop.

Samenvatting

Als conclusie van dit project is dat ik een website hebt gemaakt waarin een gebruiker een player uit het hit game Marvel Rivals kan opzoeken. Verder was er uitgelegd welke stappen ik had ge nomen voor dit project. Ik hoop dat in met dit verslag jullie genoeg geïnformeerd te hebben.