

# Versiebeheer hand-out 5

In de vorige hand-out zijn een aantal concepten langsgekomen zoals het maken van branches en hoe je deze kan benutten en het pullen en mergen en pull van branches. In hand-out 3 gaan we kijken naar het volgende:

- GitHub projecten
- Insights
- Issues

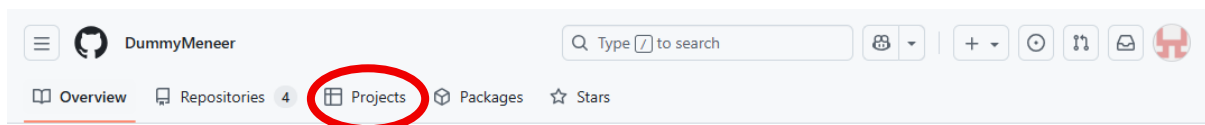
Deze hand-out bouwt voort op de vorige maar kan gebruikt worden in principe bij iedere Repository op GitHub.

## Inhoudsopgave

Opdracht 1: GitHub projecten bekijken en aanmaken (10 minuten) .....	2
Opdracht 2: Koppelen van een project aan een repo (5 minuten) .....	5
Opdracht 3: Zichtbaarheid voor projecten en groepsprojecten .....	6
Opdracht 4: Insight hebben op een repo.....	8
Opdracht 5: Issues schrijven .....	10
Opdracht 6: Eindopdracht project maken.....	13
Week 5 eindopdracht .....	13
Volgende week.....	13

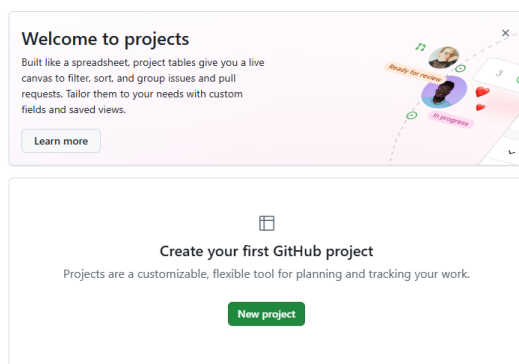
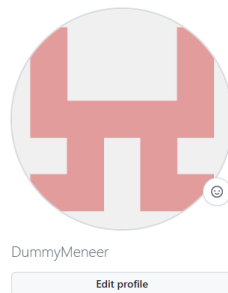
## Opdracht 1: GitHub projecten bekijken en aanmaken (10 minuten)

Er zijn veel tools die gebruikt kunnen worden om een project te ondersteunen. Een bugtracker kan een team overzicht geven in welke bugs nog in het systeem zitten. Een kanban kan taken overzichtelijk houden en voorkomen dat ze tussen de bezigheden vallen. Een teamplanner kan overzicht bieden met wie wat aan het doen is en hoever ze ermee zijn. Zo zijn er nog veel meer tools die gebruikt kunnen worden. Vaak wordt dit gedaan door meerdere programma's aan elkaar te koppelen echter is dat bij GitHub niet het geval, veel van de tools bevinden zich binnen de site onder het kopje projects.

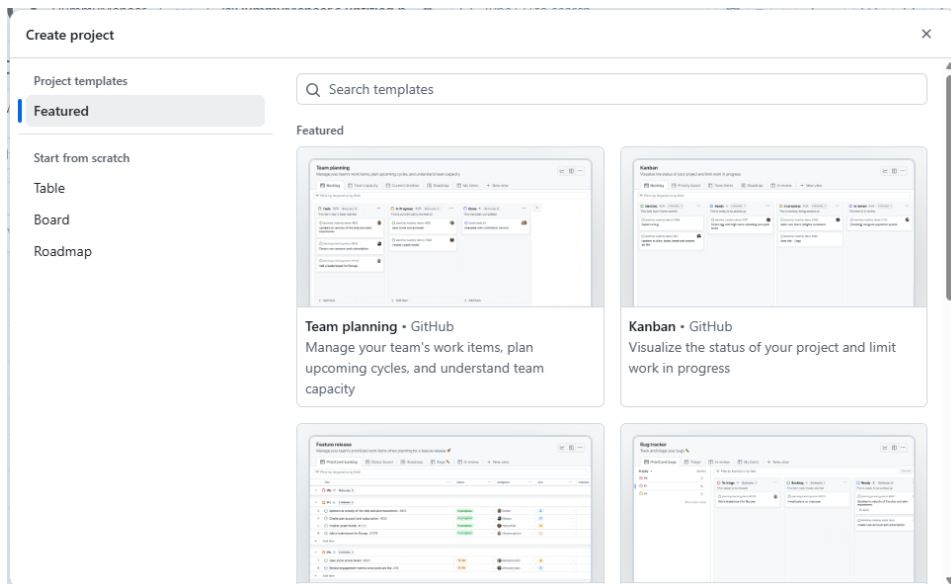


Op je hoofdpagina staan bij de balk bovenin de projects. Klik hierop dan krijg je het volgende scherm te zien.

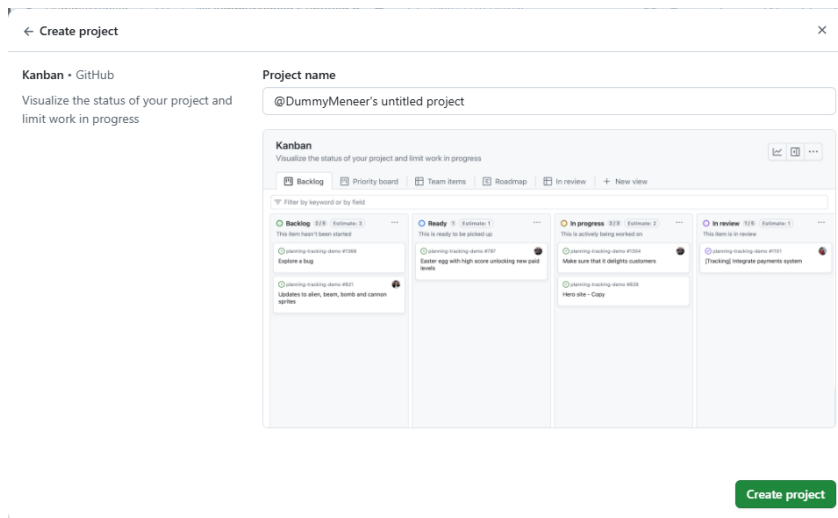
Hierbij zie je waarschijnlijk de welkom balk waar je gebruik van kan maken om meer over projects te leren. Gebruik dat vooral om meer diepgang te krijgen binnen dit onderwerp.



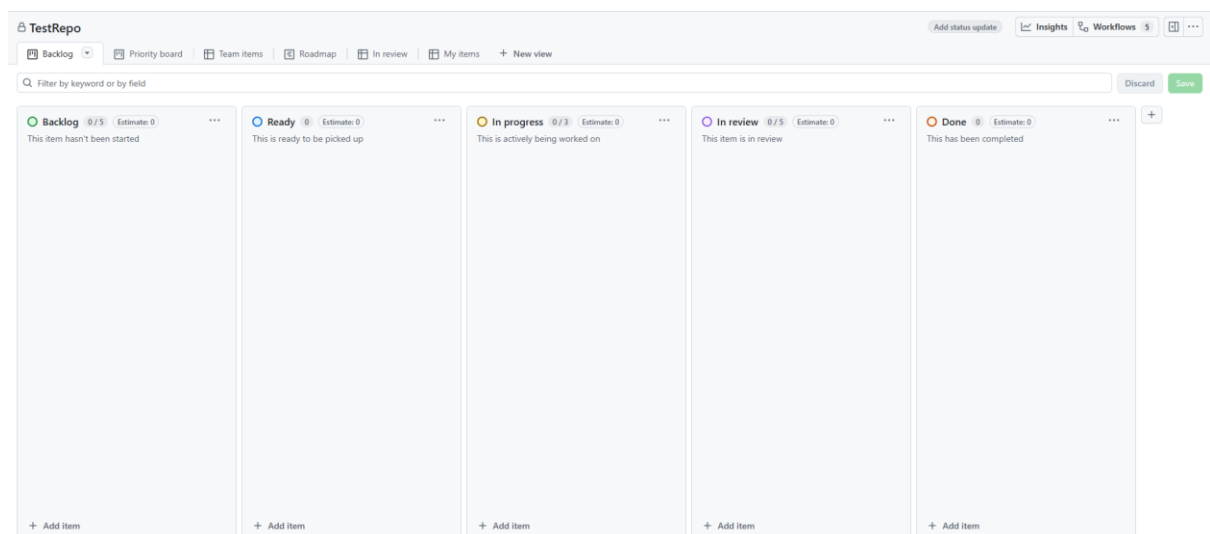
Waar we voor nu naar gaan kijken is het onderste venster, hier komen je projecten te staan. Op dit moment staat er niks dus we drukken op de knop “New project”.



We zien hier een menu waaruit we kunnen kiezen wat voor soort project we willen starten. Je kan ervoor kiezen om een template te kiezen of om eentje zelf op te bouwen. Voor nu kiezen we voor Kanban aangezien dit een vaak voorkomende tool is voor projecten. Dit maakt ook de andere tools aan maar daar leggen we nu geen focus op.



Dit project moeten we nog een naam geven. Dit is heel belangrijk aangezien het mogelijk is dat er meerdere projecten verbonden zijn aan een repository. Voor deze reden is het verstandig om de reponaam te vernoemen. Dus in dit geval “TestRepo” en dan klikken we op “Create project”



Dit is het scherm wat er dan te zien is. Bovenin zien we meerdere tabs staan, deze leiden naar andere gegenereerde tools maar onze focus blijft op de backlog/kanban.

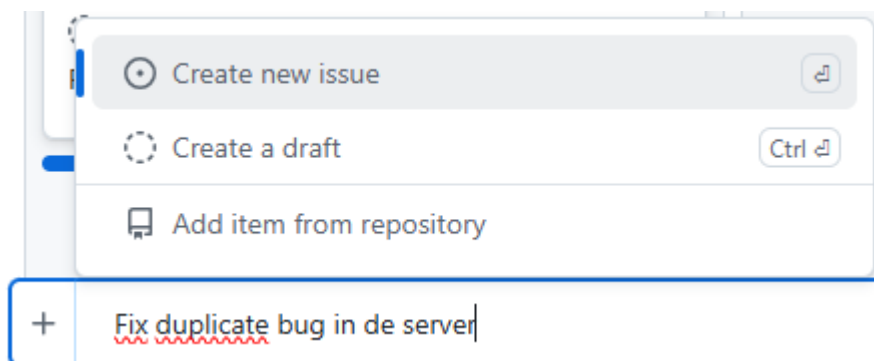
We zien hier een kanban board met daarin 5 vakken. Deze 5 vakken zijn bedoeld om de taken overzichtelijk in te verwerken en bij te houden.

- De backlog is waar je geplande taken in komen, deze taken zijn zo klein mogelijk gemaakt, dit betekent dat de Taak maak server er niet op staat maar de taken “init server applicatie”, “connect to functie”, “response test” er wel in komen.
- Ready is het vak waar taken in komen die eventueel klaar zijn om opgepakt te worden en die geen voorgaande taken nodig hebben. Bijvoorbeeld je kan een Naamwijziging van een gebruiker pas maken als de gebruiker is gemaakt.

- In progress is het vak waar taken die nu worden gedaan komen te staan, dit voorkomt dat 2 mensen niet dezelfde taak gaan doen.
- In review is het vak waar afgeronde taken in staan maar die nog getest en goedgekeurd moeten worden.
- Done is het lijstje van taken die zijn afgerond.

Het is aangeraden om het board bij te houden aangezien je met een groep dan makkelijker zonder onnodige vragen verder kan werken. Het is ook handig om je te helpen herinneren aan welke onderdelen nog gemaakt moeten worden.

Om een item aan te maken klik je onderin op de “+ add item” knop. Dit opent het input veld en hier schrijven we in wat het item is. Bijvoorbeeld: “Fix Duplicate bug in de server”



Dit opent een aantal opties, we gebruiken voor nu create a draft, dit betekent dat het niet gekoppeld is aan een issue of een repository, dit doen we in de volgende opdrachten.

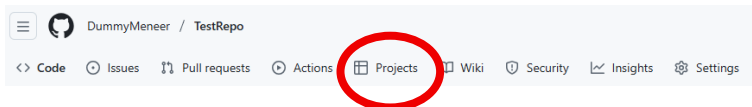
Dit zorgt ervoor dat je een item in het vak waar je de knop hebt ingedrukt. Deze kan dan eventueel gesleept worden naar het correcte vak.



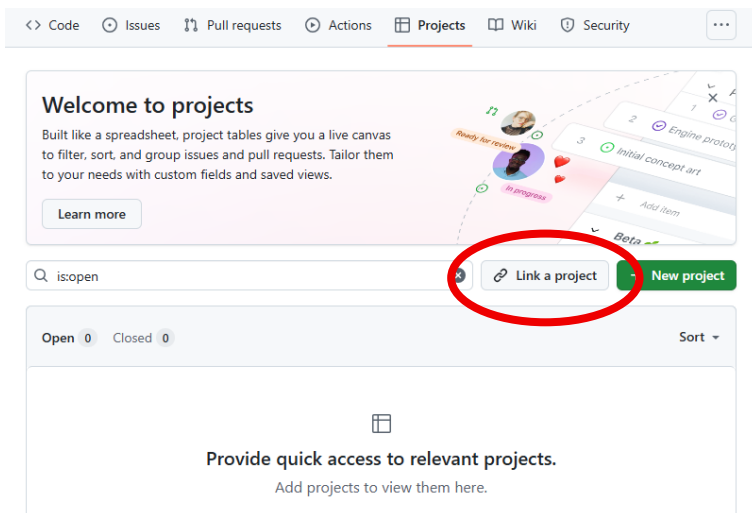
De opdracht van deze opdracht hangt samen met die van opdracht 2: Maak en link een project, met daarin minstens 3 items op een template naar keuze, voor een repository die je al hebt staan.

## Opdracht 2: Koppelen van een project aan een repo (5 minuten)

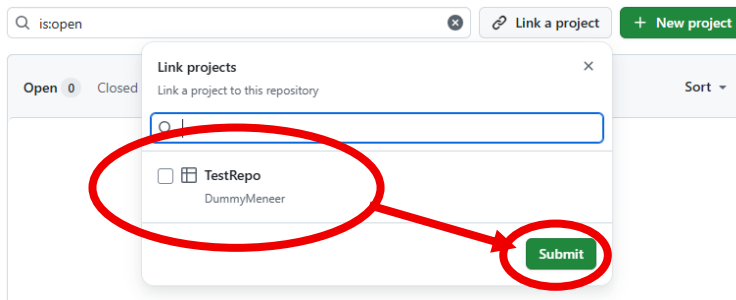
In opdracht 1 hebben we een losstaand project gemaakt, dit kan heel nuttig zijn als je vooruit wilt plannen voor een project maar nog niet in het stadium bent van programmeren. Als we dat al wel zijn kunnen we het project gaan koppelen aan de GitHub Repository. Hiervoor gaan we naar het overzicht van de repo en klikken op de Projects knop bovenin.



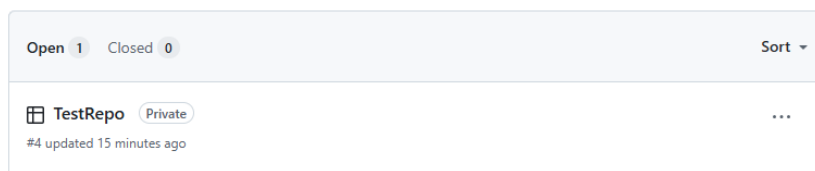
Dit opent een overzicht van de projecten die gekoppeld zijn aan de repository. Op dit moment zijn er nog geen projecten gekoppeld maar dat gaan we nu doen.



We klikken op “Link a Project” en kiezen het project met dezelfde naam.



Dit koppelt het project aan de repository en kan het makkelijk terug gevonden worden.



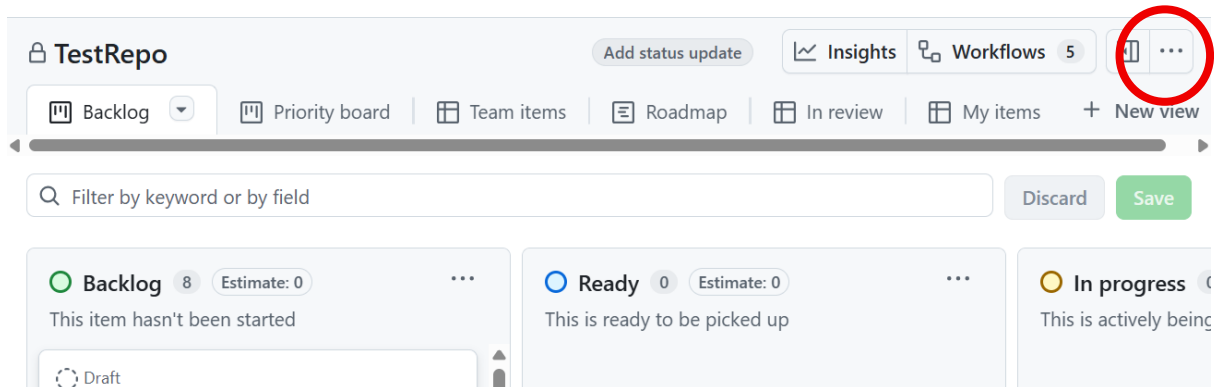
Zoals ook beschreven in opdracht 1: Maak en link een project, met daarin minstens 3 items op een template naar keuze, voor een repository die je al hebt staan.

## Opdracht 3: Zichtbaarheid voor projecten en groepsprojecten

Standaard zijn projecten op private gezet. Dit kan je zo laten staan om de stand van een repository geheim te houden echter heeft dit als effect dat ook je groepsgenoten het niet kunnen zien ondanks het feit dat ze bij de repository kunnen.

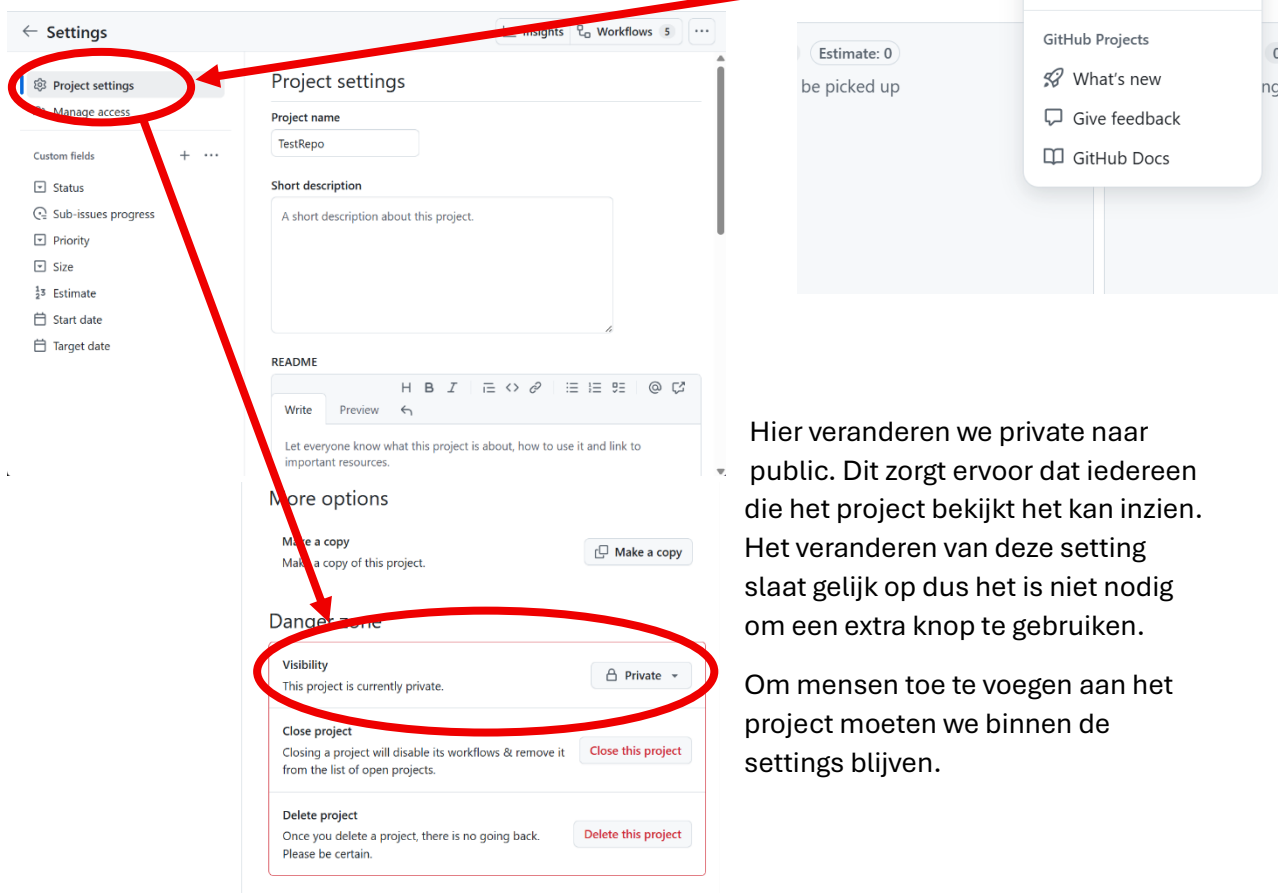
In deze opdracht gaan we 2 zichtbaarheid en toegangsonderdelen bekijken en toepassen.

De eerste is de zichtbaarheid van het project aanpassen: Dit laat gebruikers toe om het project te zien maar niet aan te passen. Om dit te bereiken gaan we naar de settings van het project. Die is te vinden rechts bovenin onder de 3 puntjes.



Dit opent een uitklap menu, hier selecteren we settings.

Binnen de settings zijn er enkele tabbladen, te vinden links. Om de zichtbaarheid aan te passen blijven we op het hoofdtabblad en scrollen we naar beneden

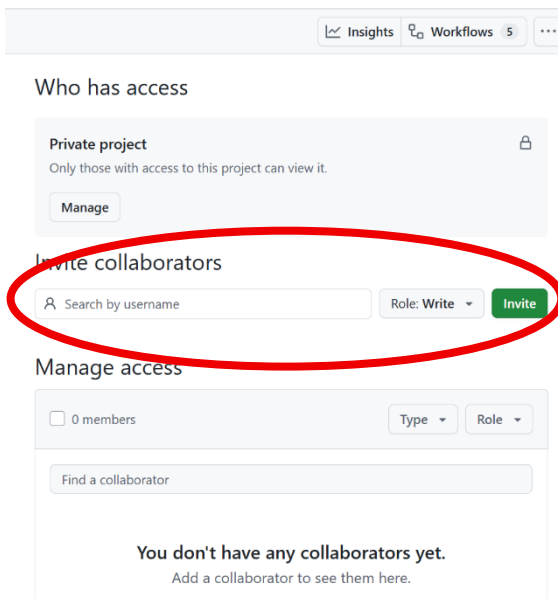
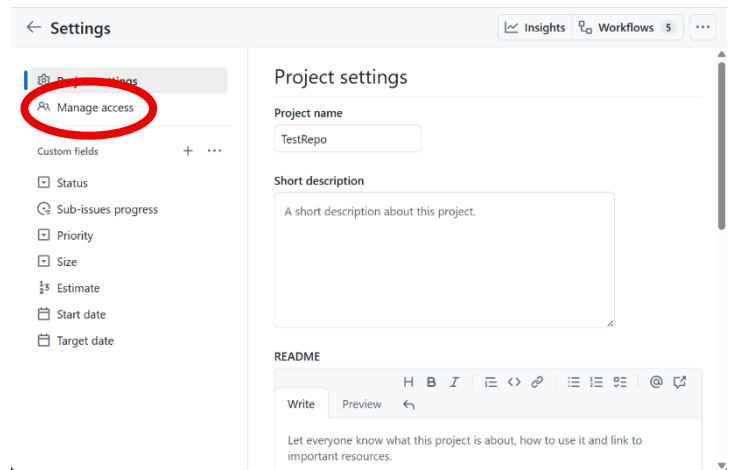


Hier veranderen we private naar public. Dit zorgt ervoor dat iedereen die het project bekijkt het kan inzien. Het veranderen van deze setting slaat gelijk op dus het is niet nodig om een extra knop te gebruiken.

Om mensen toe te voegen aan het project moeten we binnen de settings blijven.

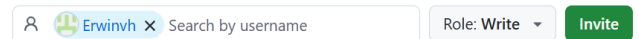
In plaats van project settings gaan we naar het tabblad eronder, Manage access.

Dit opent de accessibility tabblad. Hierin wordt aangegeven welke gebruikers welke acties kunnen doen. Het toevoegen van een project aan een groepsrepository voegt niet gelijk mensen toe dus ieder groepslid moet toegevoegd worden.



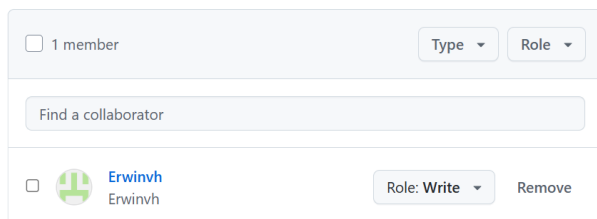
Dit doen we door de gebruikersnaam in te vullen en te bepalen welke rechten de gebruiker hoort te hebben. Voor groepsgenoten die gaan programmeren is **write** een verstandige recht om te hebben. Een begeleider zal bijvoorbeeld alleen een **read** functie hebben. In grote groepen zijn managers degene die admin rollen hebben. De **admin** rol samen met de eigenaar kunnen als enige nieuwe gebruikers toevoegen en rollen wijzigen. Enkel de eigenaar kan het project verwijderen en de zichtbaarheid aanpassen.

#### Invite collaborators

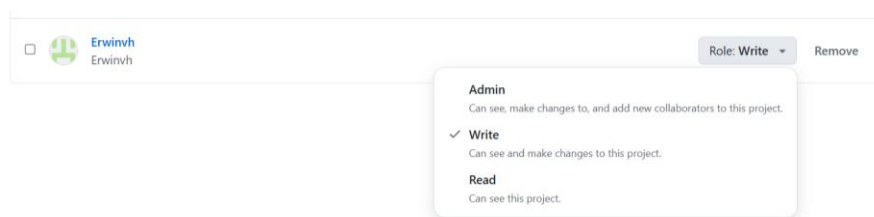


Na het toevoegen van een gebruiker komt deze in de lijst terecht. De gebruiker krijgt GEEN email dat ze zijn toegevoegd dus attendeer ze hier op.

#### Manage access

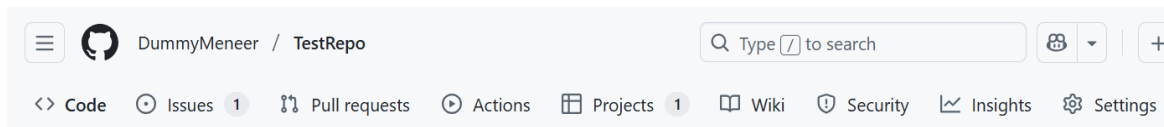


Mocht het zo zijn dat een rol verkeerd is aangewezen dan kan dat aangepast worden in het overzicht door admins of de eigenaar.

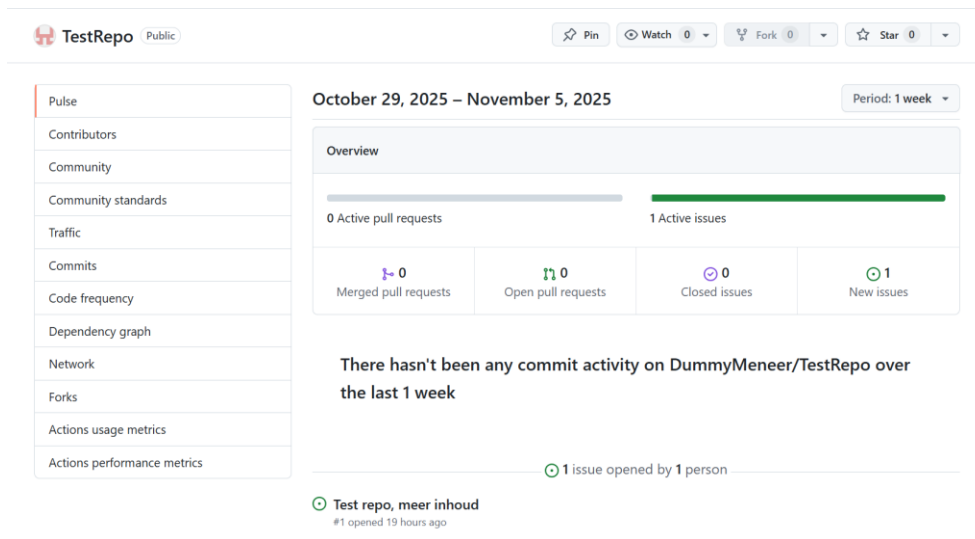


## Opdracht 4: Insight hebben op een repo

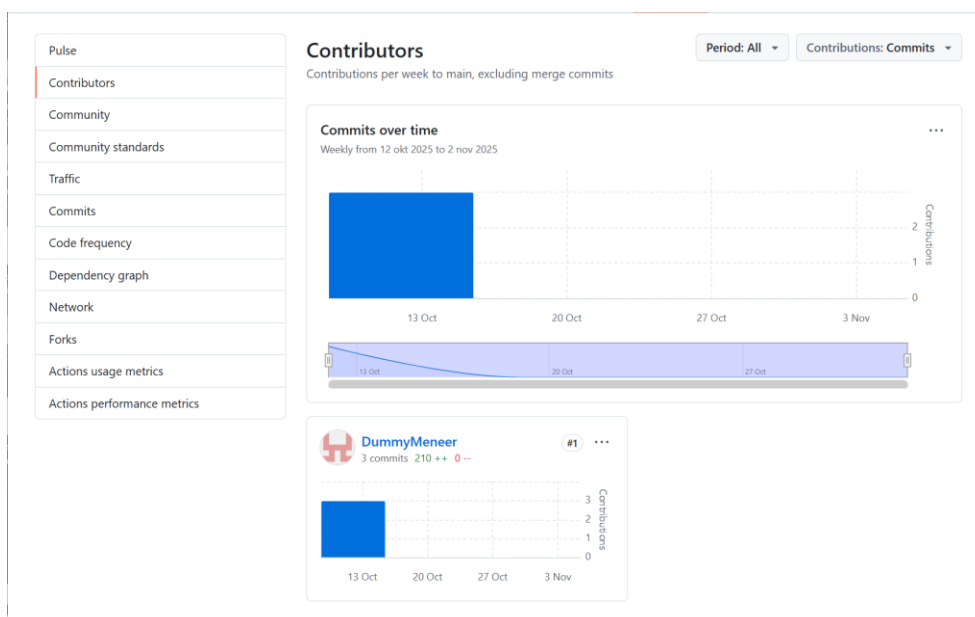
Insight het tabblad van een repo waarin inzicht tot, code, commits en samenwerking zichtbaar wordt. Bij deze opdracht doorloop je door enkele van de meest gebruikte en nuttige inzichten.



De eerste is wat GitHub noemt een pulse overzicht. Hierin is in 1 opslag zichtbaar welke issues en pull requests open staan en welke zijn afgerond. (Pull requests in hand-out 5) Deze statistieken kunnen helpen met een stand van het project en repo weergeven.

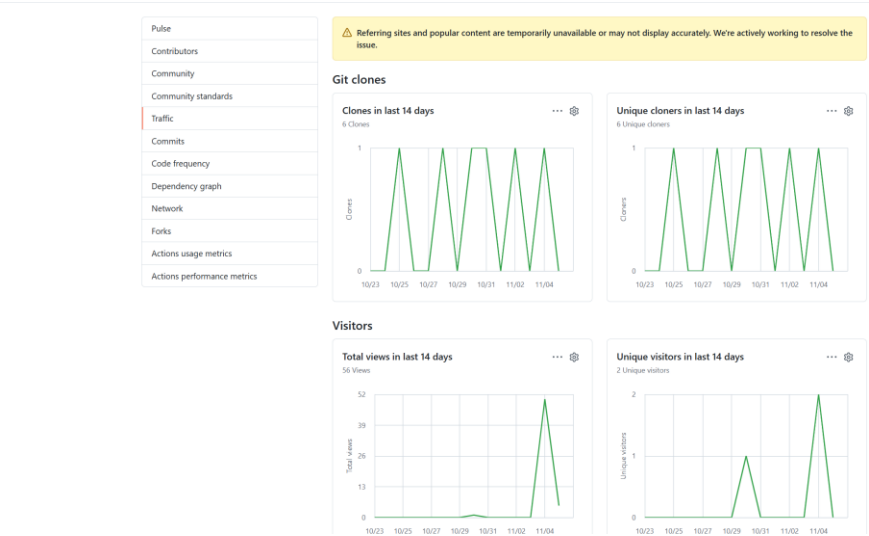


Het tabblad van contributors geef inzicht van welke groepsgenoten hoeveel hebben gedaan en hoeveel regels zijn aangepast. Dit tabblad helpt met het groepswerk verdelen maar moet niet leidend zijn. Iemand die veel code verwijderd en weinig schrijft kan vooral bezig zijn met optimalisatie en niet een slechte groepsgenoot, aan de andere kant een groepsgenoot die veel code schrijft hoeft niet een goede programmeur/groepsgenoot per definitie.

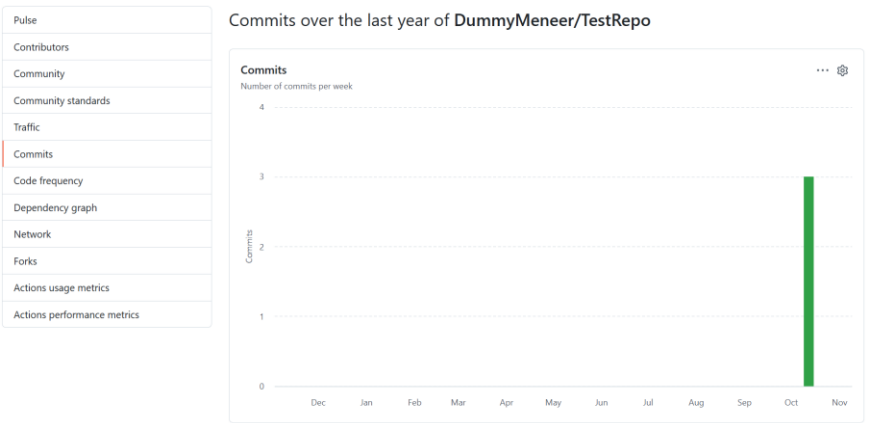




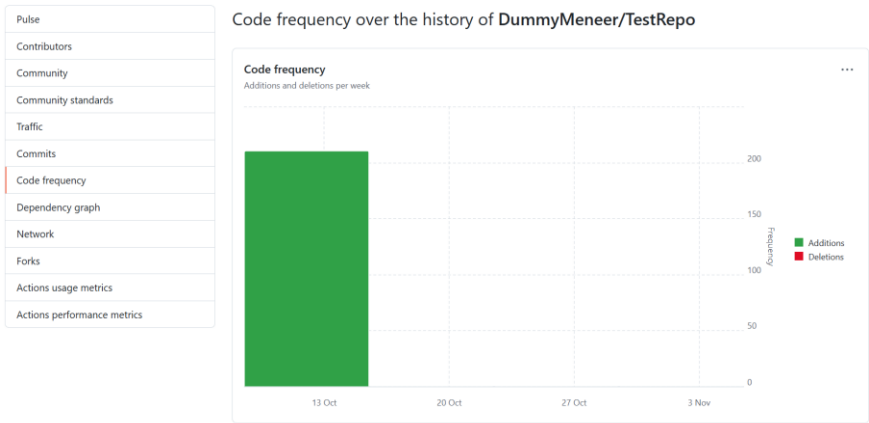
Traffic is het volgende tabblad waarin zichtbaar wordt hoeveel clones van je repo zijn gemaakt en hoeveel bezoekers naar de Repo zijn geweest. Hou er rekening mee dat bij de clones GitHub's eigen repo ook mee telt en dat jij zelf ook mee telt als bezoeker.



Het commits tabblad is een overzicht van hoe verspreid de commits zijn over een periode. Dit geeft aan hoe de werkdruk is verdeeld.

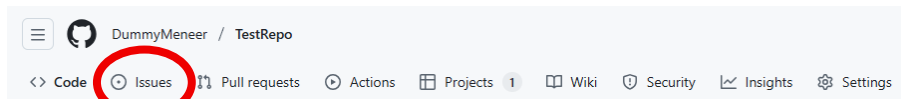


Als laatste is het code frequency tabblad waarin je inzicht kan krijgen hoeveel code in bepaalde periode is toegevoegd of verwijderd. Dit kan je inzicht geven over hoe het project verloopt maar mag niet leidend zijn. Om dat beeld te krijgen is een stand-up of inspectie van de code nodig.

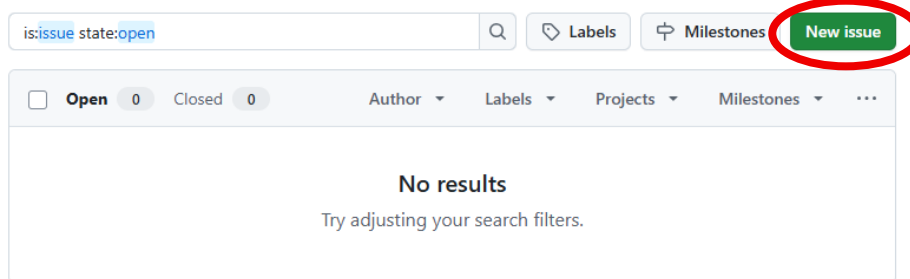


## Opdracht 5: Issues schrijven

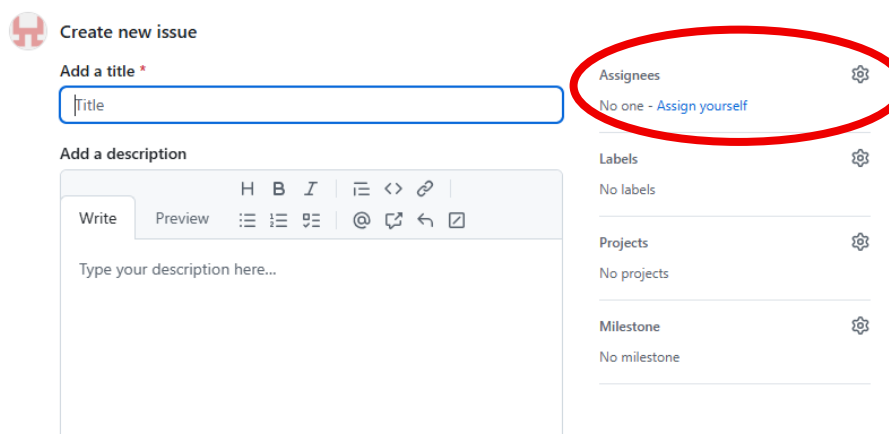
Issues zijn origineel gesproken bedoeld om aan te geven dat er een probleem is met de repository of de code erin, tegenwoordig wordt het gebruikt voor van alles. Een issue kan gemaakt worden om onder andere een functionaliteit te plannen of een idee te delen of om een discussie te houden. Er zijn een aantal manieren om een issue aan te maken. Eentje die al voorbij is gekomen is dat je een issue kan opmaken tijdens het maken van taken in een project. Een andere manier is om een issue aan te maken via de repository hoofdpagina. Klik op het tweede vakje van de bovenbalk.



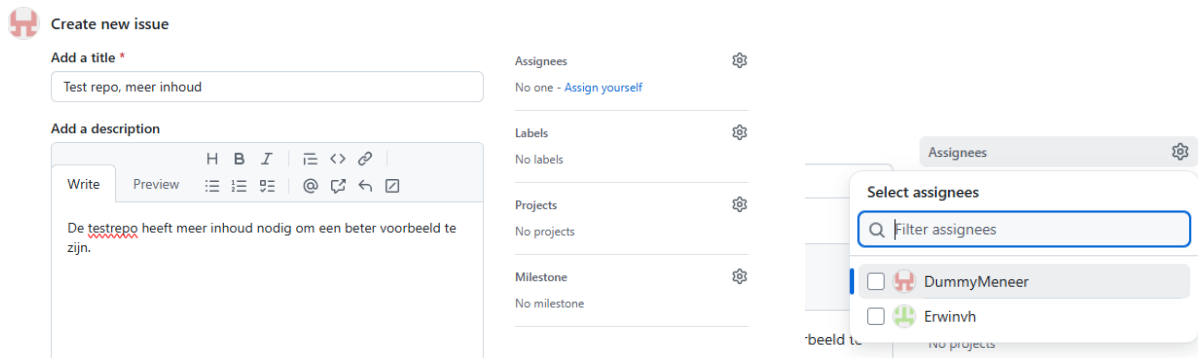
Dit opent het overzicht van alle issues die zijn gemaakt voor deze repository. Projecten kunnen overkoepelend zijn over verschillende repositories, al is dat niet verstandig, issues kunnen dit niet. Om een nieuwe issue te maken klik je op de groene knop rechtsboven in.



Het invulformulier wordt hierdoor geopend. Een Issue heeft een titel nodig en een optionele beschrijving. Dit vullen we in het linker deel van het formulier, het rechterdeel is om specificaties te benoemen.

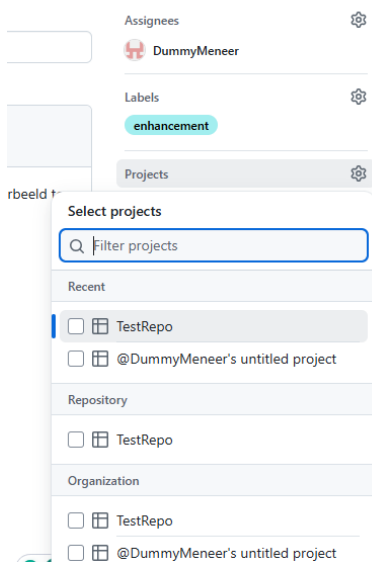
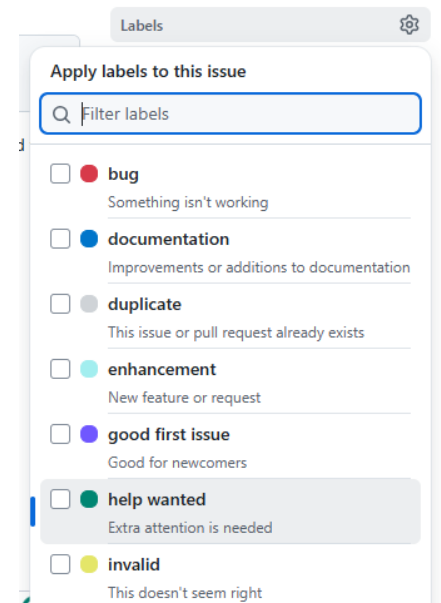


Na het invullen van de gaan we naar de specificaties. De eerste is de assignees, dit zijn de mensen die gekoppeld moeten worden aan de issue. Dit kan zijn omdat deze persoon een probleem heeft veroorzaakt of omdat iemand de functie gaat maken. Je kan het snel koppelen aan jezelf door "assign yourself" te klikken maar als je een groepsgenoot wil koppelen eraan dan klik je op het tandwiel en selecteert de groepsgenoot.

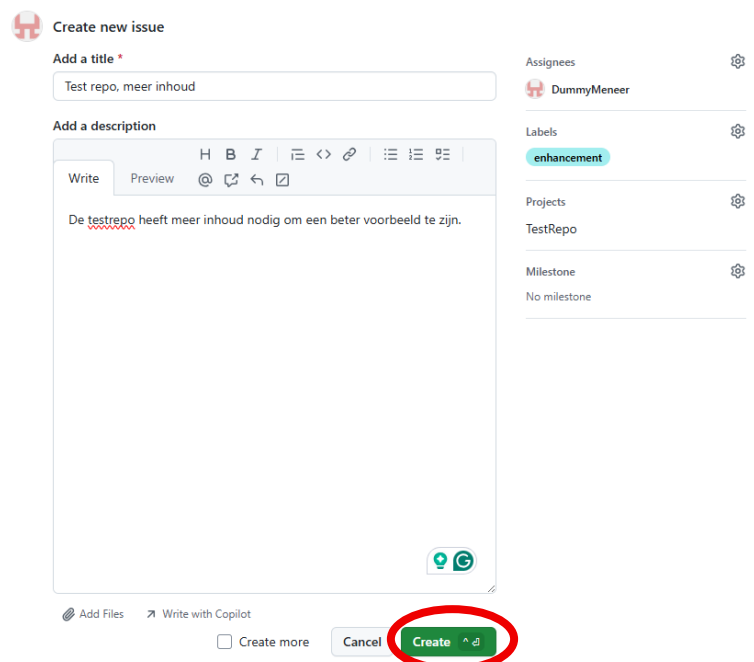


Als je iemand koppelt aan een issue krijgt deze persoon ook notificaties over de issue als er berichten in gestuurd worden of wijzigingen zijn plaatsgevonden.

Nadat de assignees zijn gekoppeld moeten de labels geplaatst worden. Labels zijn bedoeld om onderscheid te maken tussen de verschillende soort issues die geschreven kunnen worden. Dit maakt zoeken tussen de issues makkelijker en bevordert teamwerk.



Daarna gaan we het project definiëren en koppelen aan de issue indien het nodig is. Dit doen we als het bijvoorbeeld een taak is die gedaan moet worden en dus op de kanban moet. Het plaatsen van de issue op de kanban wordt dan automatisch gedaan als het gekoppeld is aan dat project. Een idee of discussie wordt niet gekoppeld aan een project omdat het geen taak is.



Hierna zijn de milestones/mijlpaal die toegekend kunnen worden. Dit doe je om mijlpalen te vullen met criteria die nodig zijn voor de mijlpaal. Bijvoorbeeld: de mijlpaal UI bouwen kan bestaan uit de issues “Kleurenpalet kiezen”, “knoppen maken”, “Teksten schrijven” en “Alles samenvoegen”

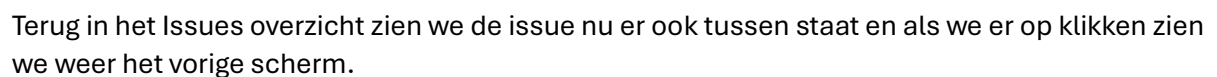
In dit vak gaan we niet verder op in hoe je milestones maakt, dit mag je zelf uitzoeken als je dat leuk vindt maar is niet vereist.

Hierna kunnen we op de create knop klikken om de issue te maken.

Test repo, meer inhoud #1

[Edit](#) [New issue](#) 

[Open](#)



## Opdracht 6: Groepsopdracht project maken

Voor de groepsopdracht moet er een groepsproject in jullie GitHub repository zitten. Dit hoeft niet een complex project te zijn maar wel een die een deel van het proces ondersteunt. Dit kan een kanban zijn maar kan ook een takenoverzicht zijn of een overzicht van wie waar mee bezig is, van alles mag zolang er over na gedacht en nuttig is voor het project. Ga goed nadenken met je groepje welke projecttemplates, als je er een gaat gebruiken, je wil gaan gebruiken en hoe.

Zorg er ook voor dat het volgende ook gedaan is:

- Gekoppeld is aan jullie groepsrepo
- Zorg ervoor dat het project publiek is of dat iedere deelnemer van het project als collaborator is toegevoegd aan het project.
- Zorg ervoor dat ieder groepslid minstens schrijfrechten heeft.
- Indien het een privé repo en project is, zorg ervoor dat de docent toegang heb maar geen schrijfrechten.

## Week 5 eindopdracht

Stuur de link van de repo en invite naar de docent toe

Wees je ervan bewust dat dit de laatste praktijk les is en in week 8 en 9 jullie je groep repo en eigen repo zo goed als klaar moeten hebben voor beoordeling. Denk alvast vooruit wat je kan voorbereiden. Check ook de matrix om te controleren hoever jullie zijn met de repository.

## Volgende week

Deze week is de laatste praktijk les, de volgende twee lessen zijn niet nodig voor het halen van het vak. De informatie is bedoeld als extra informatie en kan nuttig zijn.

Volgende week worden de volgende onderwerpen behandeld:

- Gitkraken
- Lokaal en remote werken
- Stashen en poppen