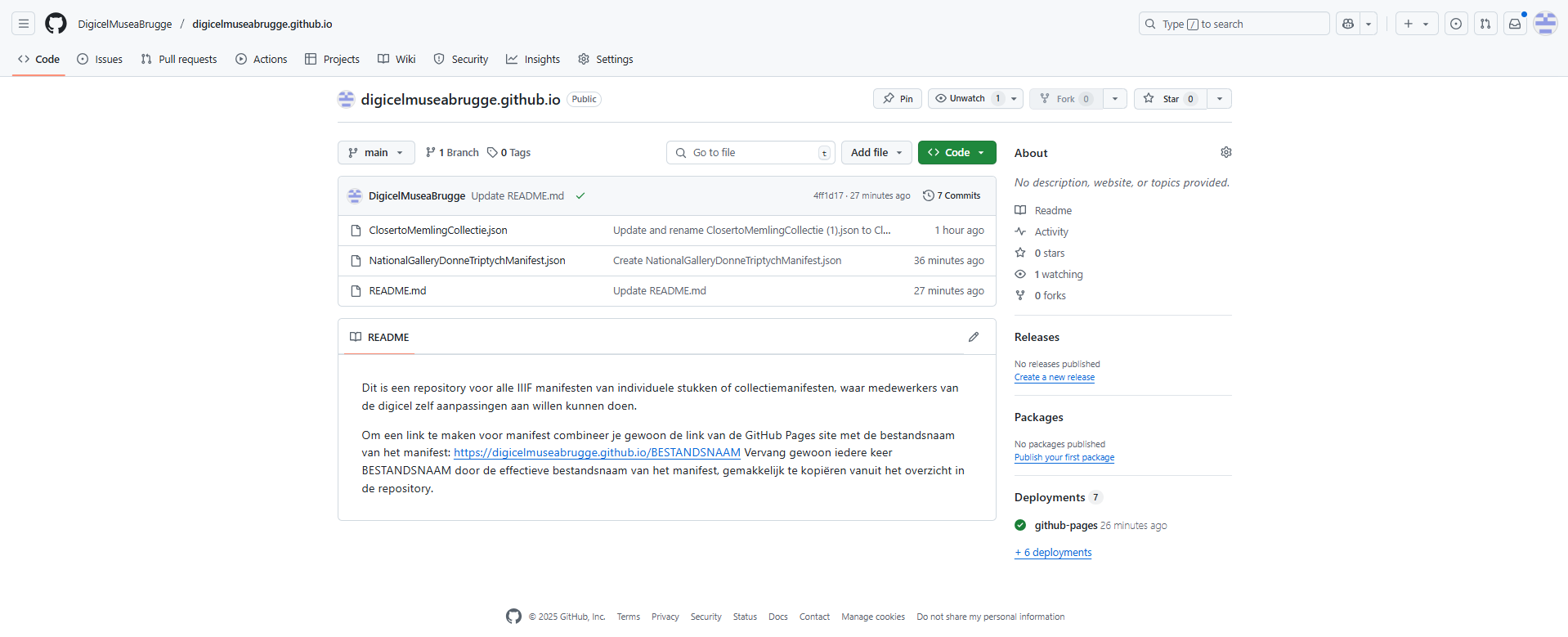
Github gebruiken als server voor het hosten van IIIF manifesten of collecties

Soms willen we op korte termijn iets uittesten of aanpassen voor een toepassing van IIIF. Het is daarom niet altijd handig om te wachten op de externe leverancier. Als we zelf aan de slag willen gaan met het bewerken of creëren van manifesten is het echter noodzakelijk dat we de manifesten online kunnen hosten op een server waar we zelf toegang tot hebben. De meeste IIIF toepassingen kunnen namelijk niets aanvangen met lokaal opgeslagen bestanden. **Github** is een gratis en gebruiksvriendelijke toepassing met uitgebreide documentatie, die ons toelaat om manifesten en meer te uploaden, bewerken en gebruiken in IIIF toepassingen.

In deze handleiding toon ik eerst hoe je toegang kan krijgen tot de Github**Repository** van Musea Brugge. Daarna leg ik uit hoe we de Repository kunnen gebruiken om manifesten toegankelijk te maken voor IIIF toepassingen. Daaronder valt het uploaden en bewerken van bestanden in de Repository en de twee manieren waarop een link naar het gehoste bestand kunnen genereren. Ten slotte beschrijf ik hoe je zelf een Repository kunt aanmaken en de bestanden hosten met Github *Pages*.

# Toegang krijgen tot de GitHubRepository van Musea Brugge

Repositories zijn het basiselement van GitHub. We kunnen ze zien als een hoofdmap of schijf waar we een bepaald project op bewaren. De repository bevat alle bestanden en mappen die deel uitmaken van dat ene project. In ons geval zullen dat dus vooral (collectie)manifesten, thumbnails en logo’s zijn. De repository houdt verder ook de geschiedenis van alle bewerkingen die toegepast zijn op de bestanden in de repository bij. De voertaal op GitHub is Engels.

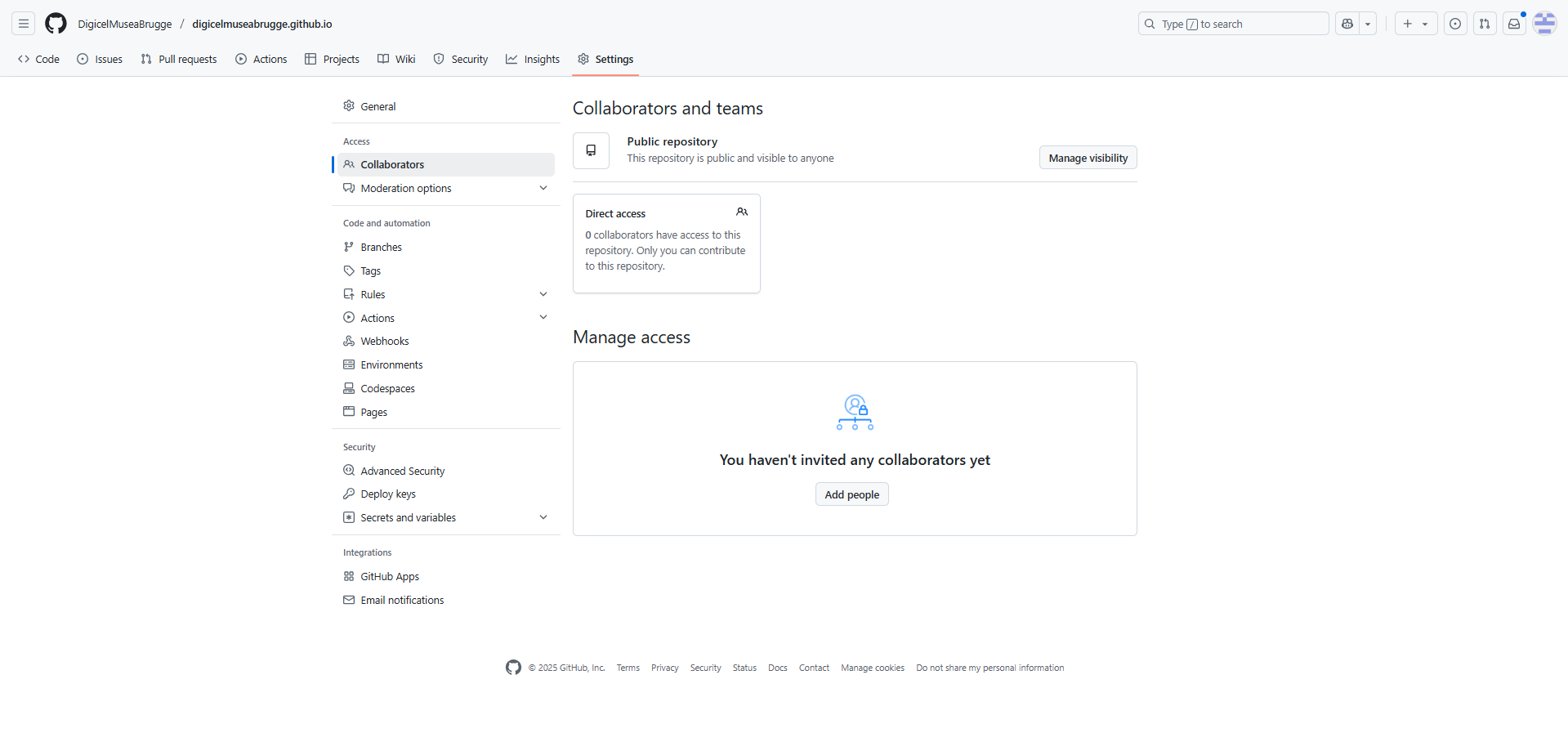


Repositories kunnen publiek of privaat zijn. Voor onze doeleinden is een publieke repository voorlopig voldoende, maar als de digicel of Musea Brugge van plan zou zijn om GitHub op grotere schaal of voor gevoelige data te gebruiken is het geen slecht idee om een abonnement te nemen op een betalende versie en een private repository te gebruiken.

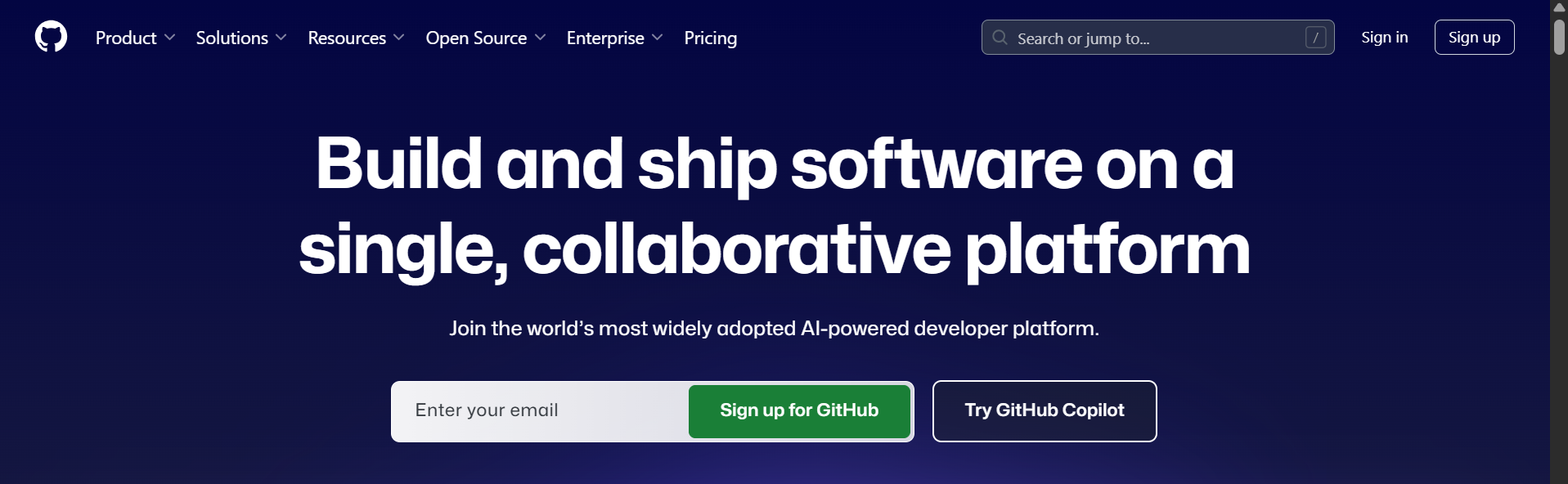
Er zijn twee manieren om toegang te krijgen tot een repository. Bij een publieke repository kan standaard iedereen de repository opzoeken en bestanden uploaden of bewerken. Je kan als normale gebruiker dus gewoon de naam van onze repository intypen in de zoekbalk bovenaan de pagina in GitHub en er op klikken om toegang te krijgen. De naam van de repository van Musea Brugge is: **digicelmuseabrugge.github.io**.

Om te voorkomen dat er ongeoorloofde veranderingen gemaakt worden aan de bestanden in de repository heb je als eigenaar van de repository echter verschillende mogelijkheden om andere gebruikers beperkingen op te leggen in wat ze kunnen bewerken of uploaden.

Als eigenaar van een repository kan je ook direct nieuwe gebruikers uitnodigen met hun gebruikersnaam of emailadres. Je kan dit wanneer je in je repository zit via **settings** > **collaborators** > **manage** **access** > **add** **people**.



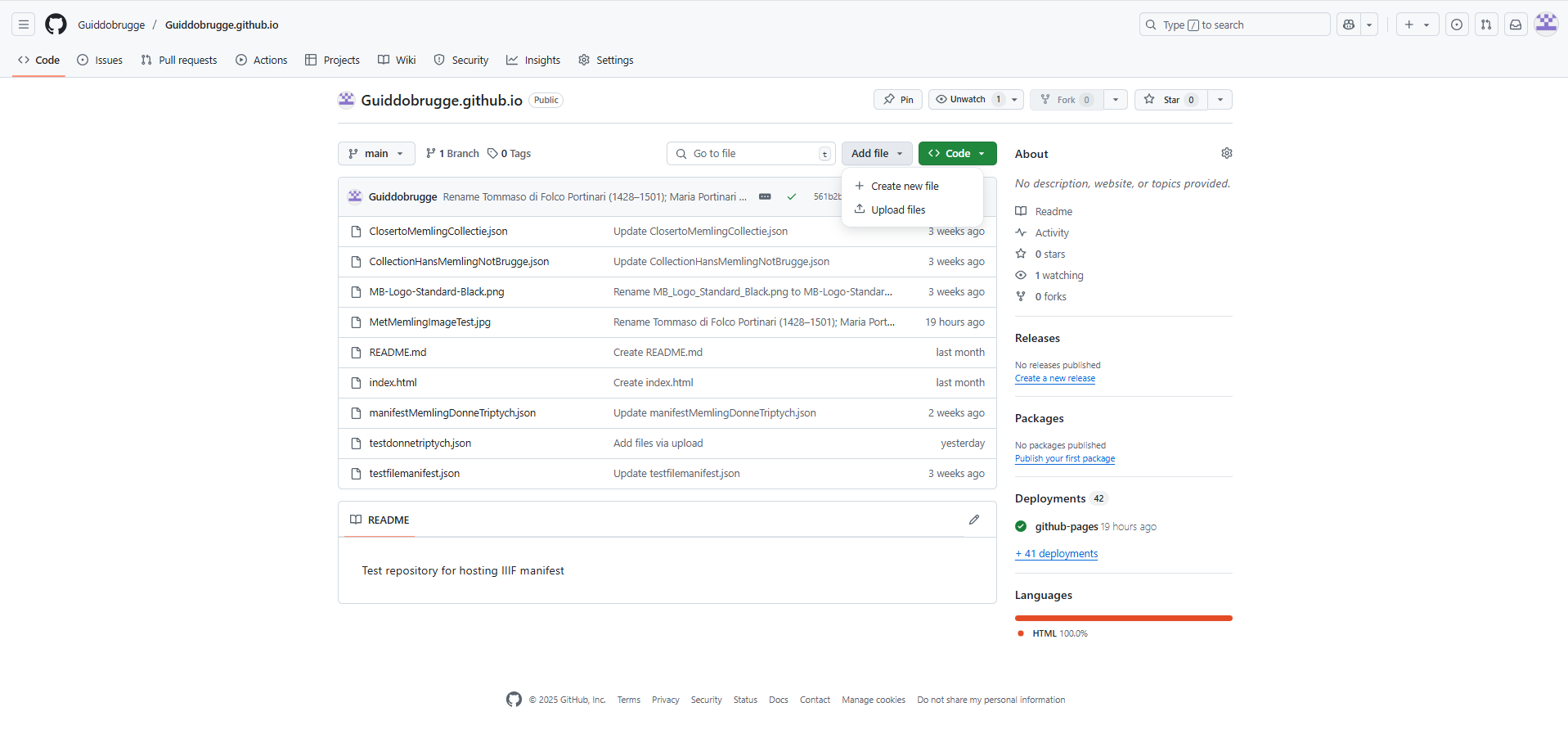
Voor beide manieren heeft iedere gebruiker een eigen account nodig. Een account maakt je gemakkelijk aan via de **sign-up** knop op de homepagina van GitHub.



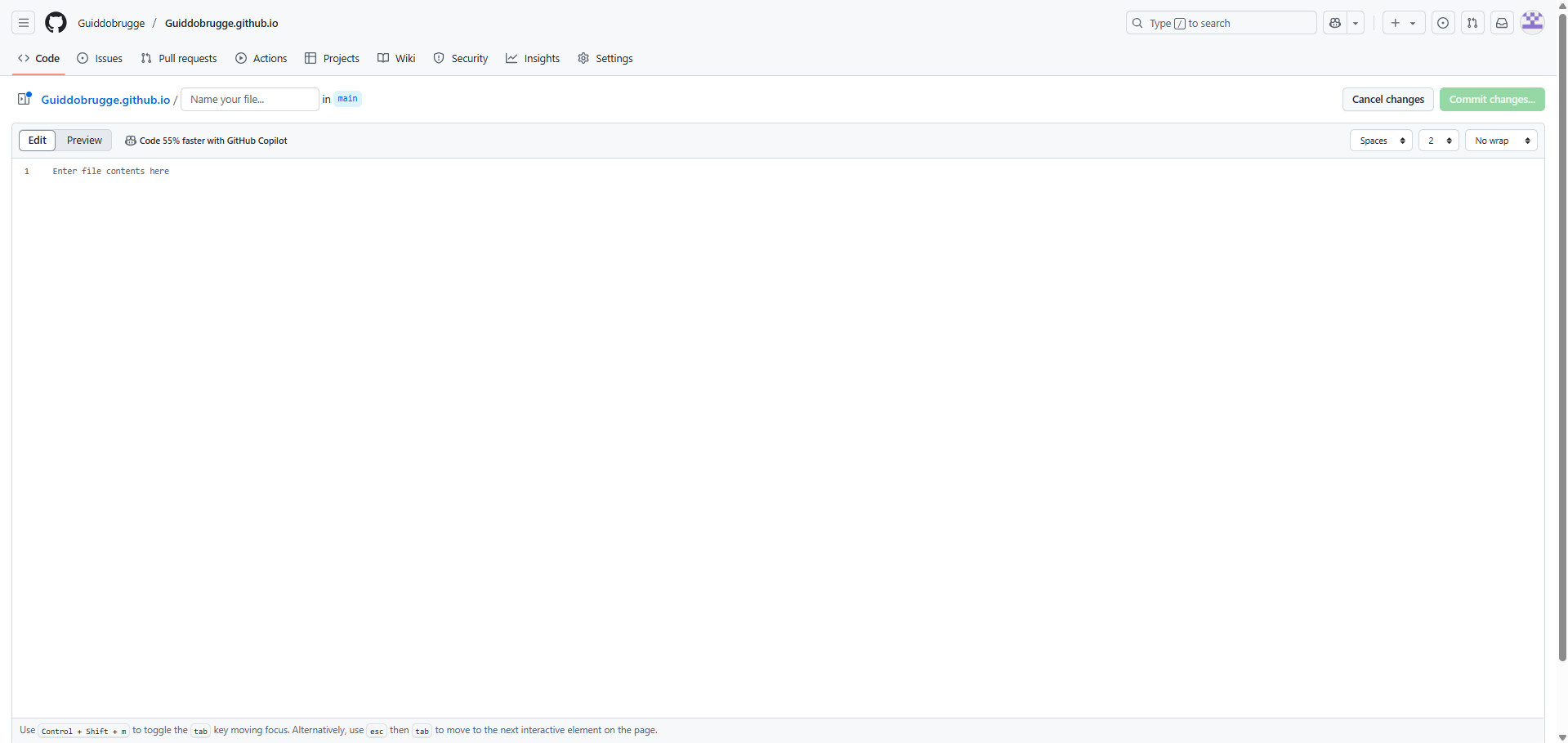
# Bestanden uploaden en bewerken

## Bestanden uploaden

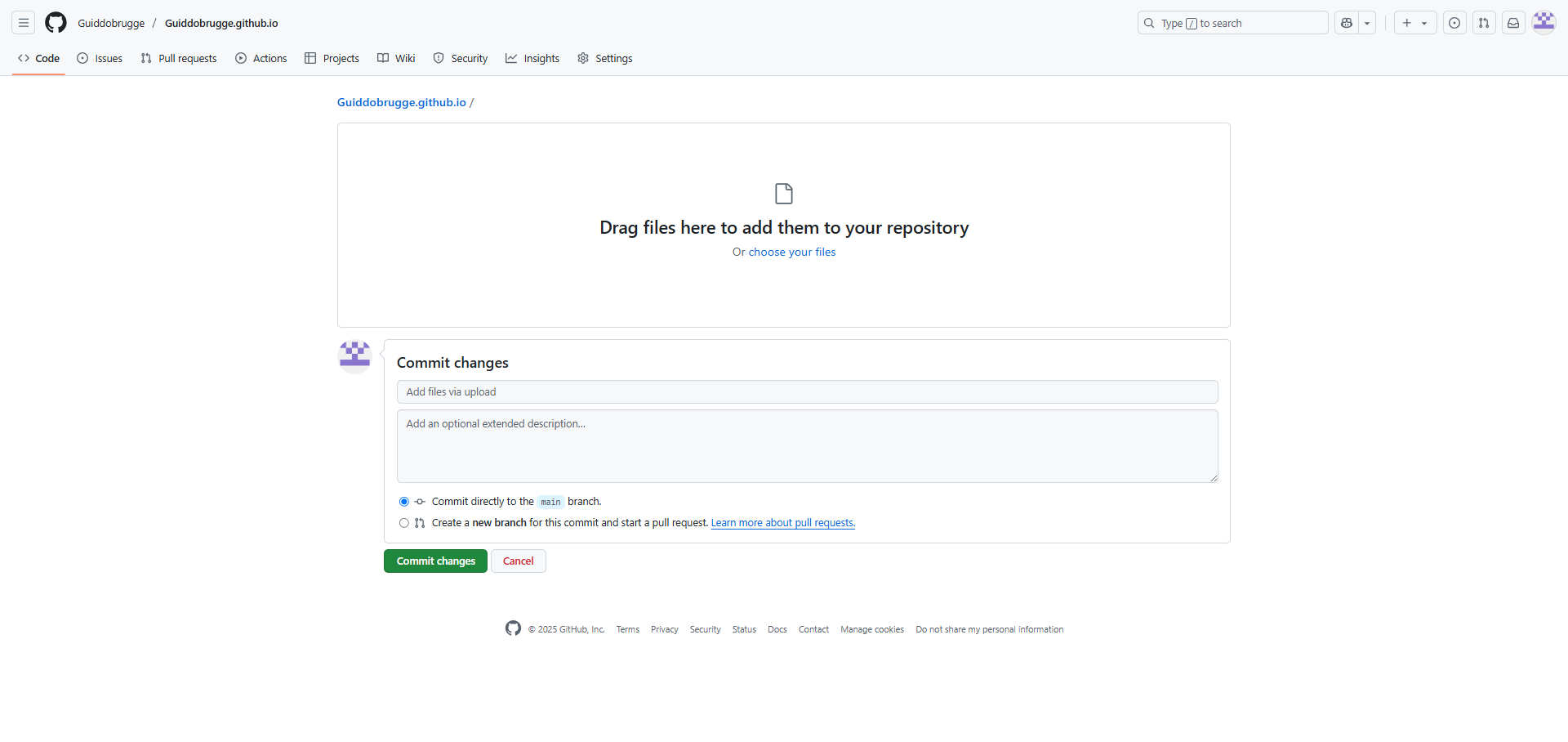
Het uploaden en bewerken van bestanden is behoorlijk gebruiksvriendelijk. Als je een nieuw bestand wil uploaden ga je gewoon naar de **repository**. In de repository klik je op **code** in de werkbalk bovenaan de pagina. Als je klikt op code zie je een overzicht van alle bestanden die in onze GitHub repository zitten met daarnaast heel wat extra informatie. Je ziet ook een knop genaamd **add file**. Als je daarop klikt krijg je twee opties om een bestand toe te voegen.



Bij **Create new file** kan je direct in GitHub zelf een nieuw bestand creëren in de ingebouwde editor voor code, deze heeft verschillende behulpzame functies en is handig als we zelf een .json manifest willen maken. Zorg er ook voor dat je de juiste extensie toevoegt na de naam die je kiest voor het bestand. De gewenste naam voor het bestand kan je typen in het tekstvak boven de grijze balk. Voor een manifest zou de naam bijvoorbeeld kunnen zijn: “VoorbeeldManifest.json”



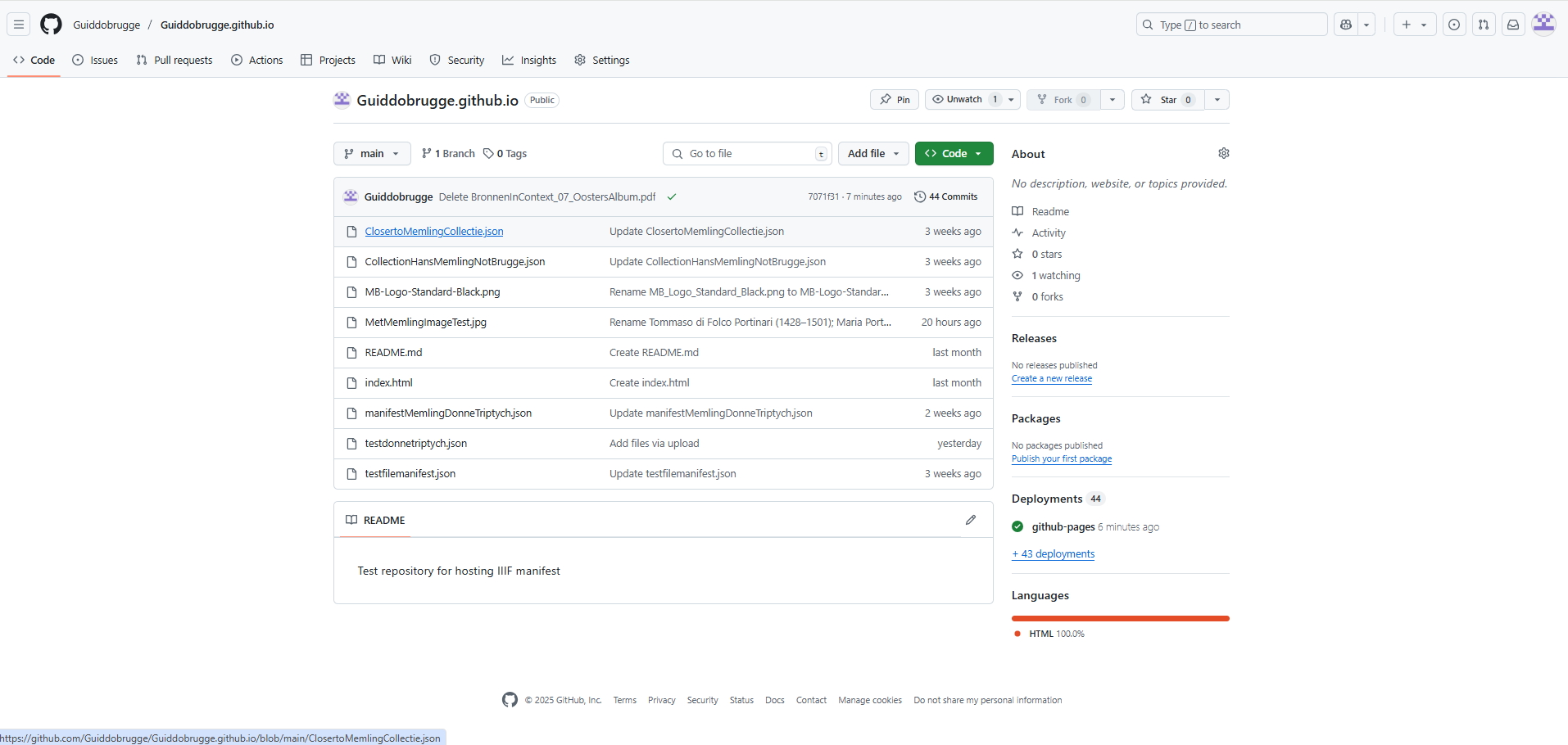
Als we al een manifest hebben kunnen we dat direct uploaden vanop onze computer via **Upload files**.



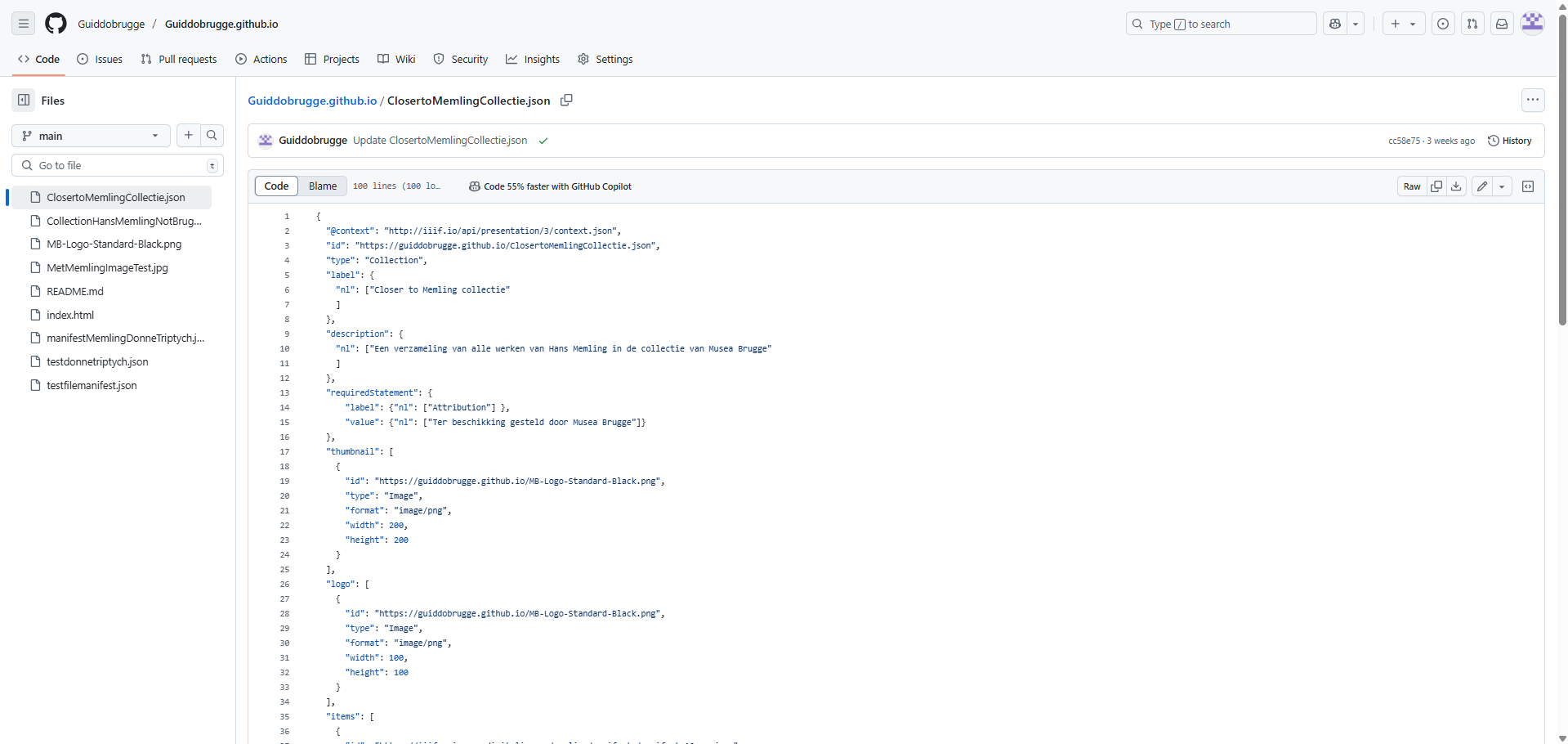
Eens je tevreden bent met het bestand dat je gemaakt hebt of wil uploaden klik je op **Commit changes** om je keuzes te finaliseren en toe te voegen aan de repository. GitHub houdt ook de hele geschiedenis bij van welke gebruiker wat aangepast of toegevoegd heeft in de repository.

## Bestanden bewerken

Als we een bestaand bestand in de repository willen bewerken moeten we terug beginnen bij het tabblad **code** in de grijze bovenbalk**.** Daar klikken we het **bestand** dat we willen bewerken gewoon aan. In dit geval wil ik bijvoorbeeld “ClosertoMemlingCollectie.json” bewerken, een collectiemanifest waar alle schilderijen van Memling uit de collectie van Musea Brugge in zitten.



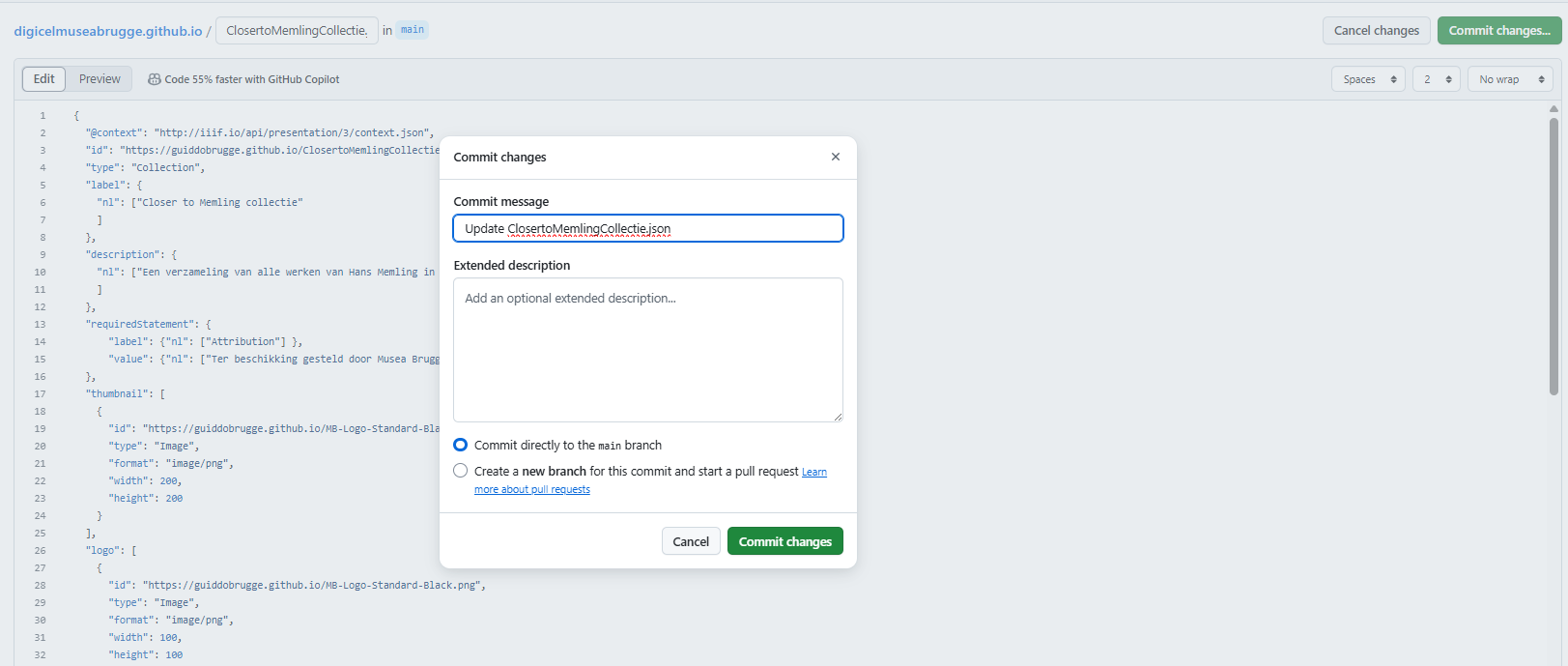
Eens we op het bestand geklikt hebben krijgen we een nieuw scherm met de lijnen code in het bestand. Dit is heel gelijkaardig aan het scherm dat we te zien krijgen wanneer we op Create new file klikken.



GitHub geeft ons hier verschillende opties in de lichtgrijze balk boven de lijnen met code. Als je de code wil bewerken moet je gewoon op de **Edit** knop met het potlood icoontje klikken. We kunnen zo ook de naam aanpassen.

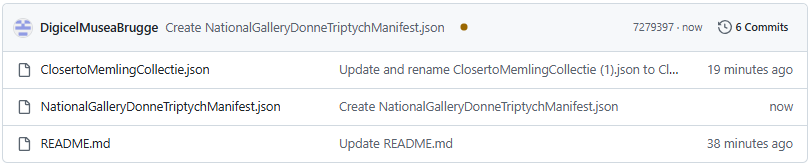
Boven het potlood icoontje zien we ook **History** staan. Als je daarop klikt kan je de hele geschiedenis van de verschillende versies van het bestand zien, wat soms van pas kan komen, zeker als je met meerdere mensen in de repository zou werken.

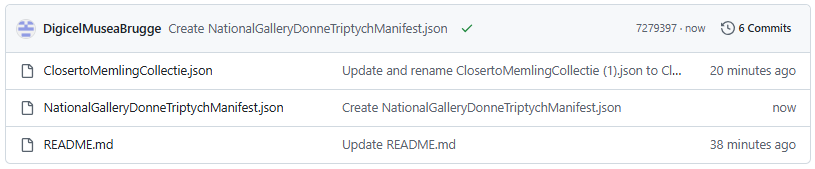
Als je klaar bent met bewerken van het bestand klik je gewoon terug op **Commit changes** om de veranderingen toe te passen.



Het kan tot een paar minuten duren voor de nieuwe versie van het bestand beschikbaar is op de GitHub server. Dit geldt ook voor het uploaden van geheel nieuwe bestanden.

We kunnen de voortgang van het synchroniseren van de aanpassingen met de server volgen door te kijken naar het bruine bolletje of het groene vinkje op verschillende plaatsen in de repository. Als er een bruin bolletje staat is GitHub nog bezig met de aanpassingen online beschikbaar te maken op de **GitHub Page**. Als het bolletje een groen vinkje wordt is de aanpassing doorgevoerd en kan je het manifest beginnen gebruiken in viewers en andere IIIF toepassingen.

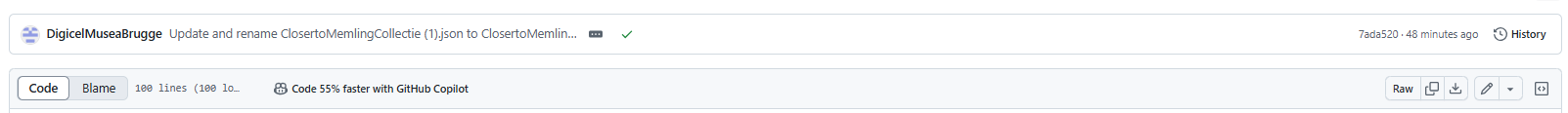




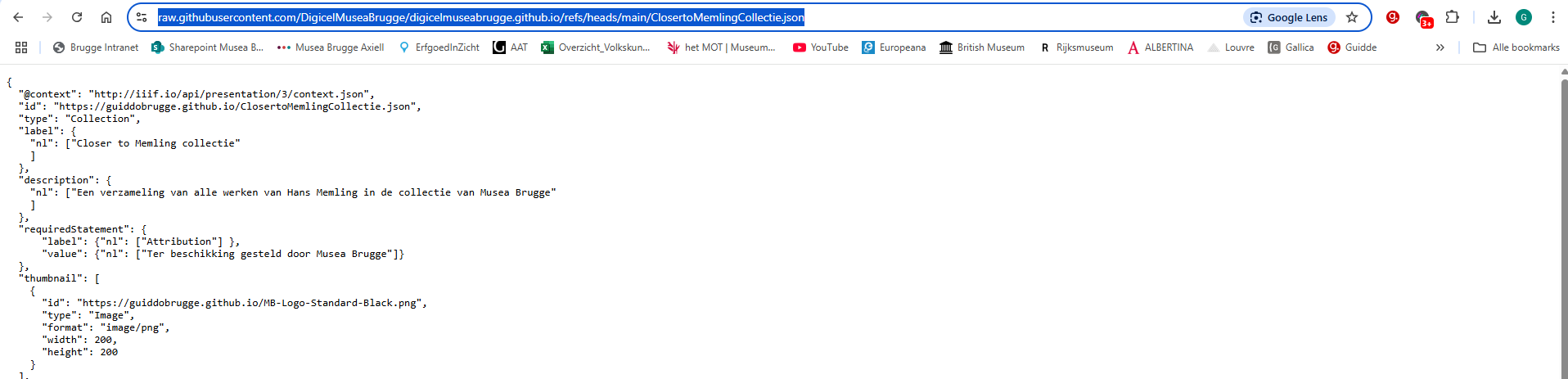
## Link genereren voor het manifest: Raw of GitHub page site

Onze manifesten worden nu online gehost op de server gekoppeld aan onze GitHub repository. De laatste stap om de manifesten te kunnen gebruiken is om link te genereren voor ons manifest, dat we kunnen gebruiken in IIIF toepassingen zoals de Mirador Viewer. We kunnen dit op twee manieren doen.

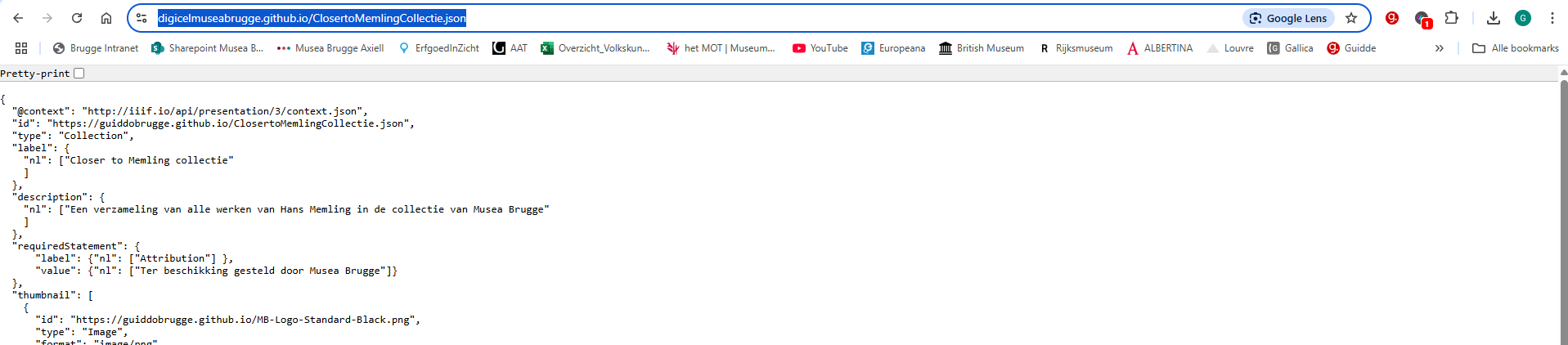
De eerste manier is om gewoon op het manifest in de repository te klikken en een **Raw URL** te genereren. Als je op het manifest klikt kom je terug op dezelfde pagina waar we eerder al bestanden leerden bewerken. In dezelfde grijze balk als de **edit** knop staat er nog een knop met **Raw** in geschreven.



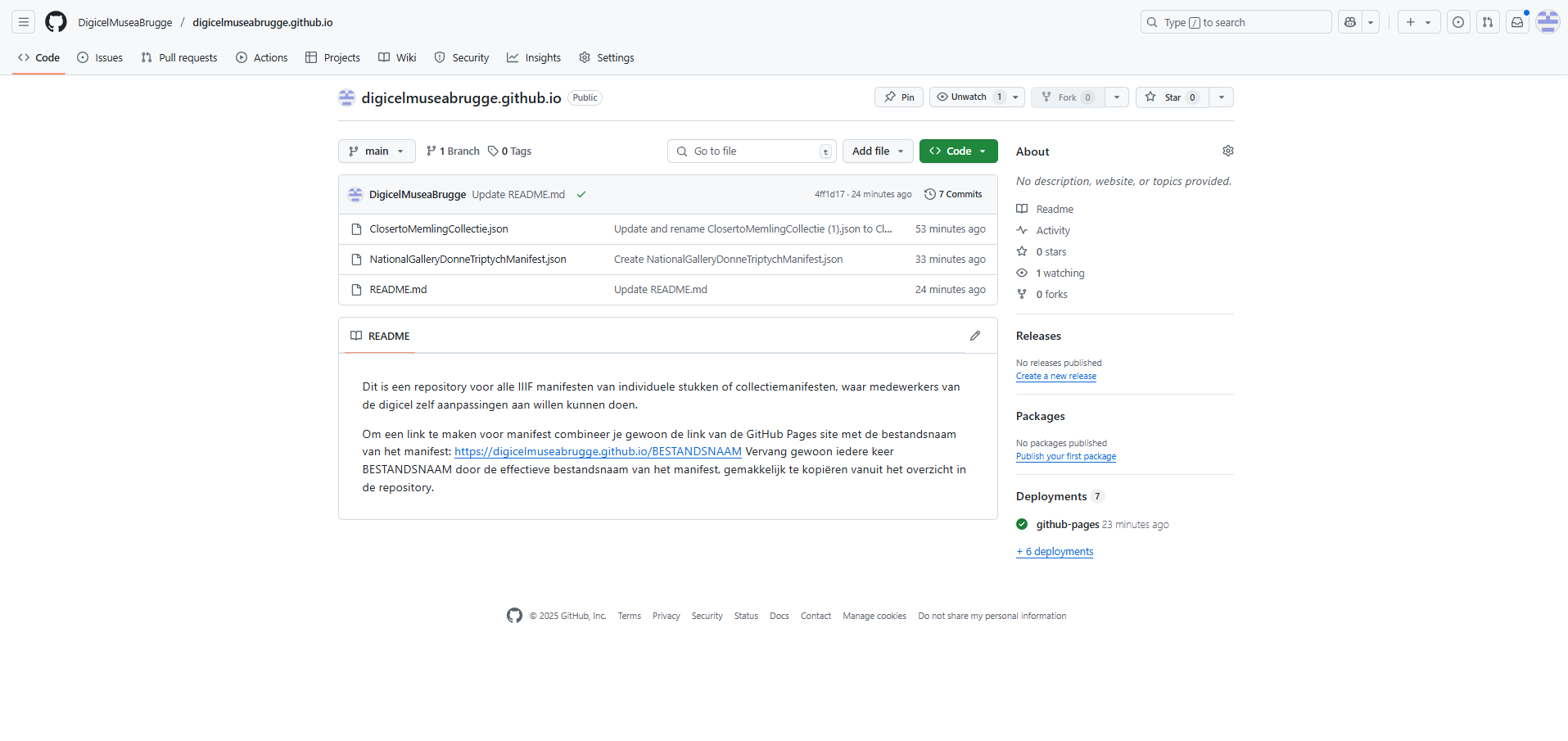
Als je hierop klikt opent GitHub een webpagina met enkel de code van je manifest op. De URL van die pagina kan je gewoon kopiëren en die kan perfect dienst doen als de manifest URL. De link die we op deze manier genereren is echter behoorlijk lang, gecompliceerd en onoverzichtelijk.



De tweede manier genereert een compactere **URL via de GitHub Pages site**. De GitHub Page is al geconfigureerd in het geval van de repository voor DigicelMuseaBrugge. Om een link te creëren naar een manifest in de repository moet je gewoon de **URL van de GitHub Page** combineren met de **bestandsnaam van het manifest**.

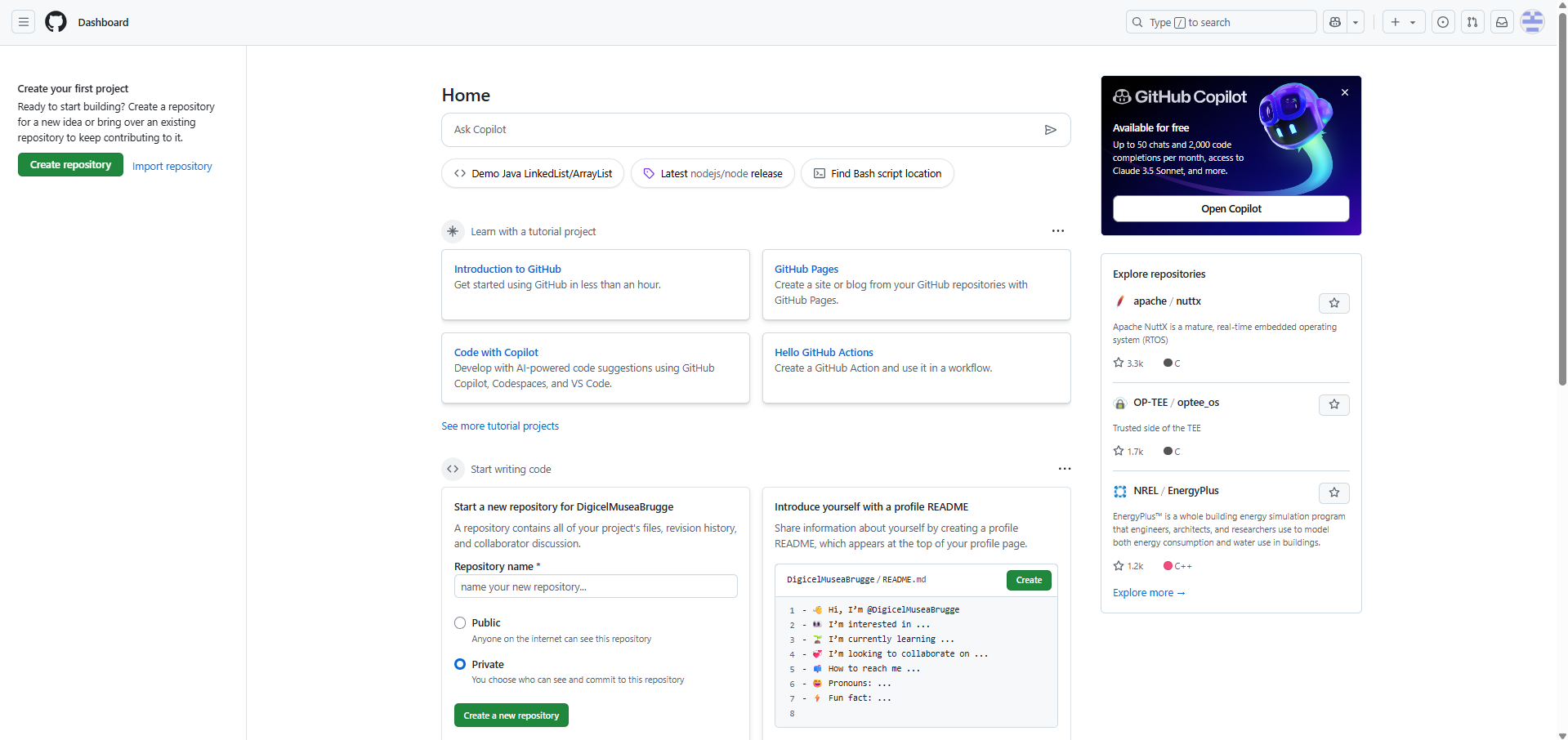


In ons geval is dat dus **https://digicelmuseabrugge.github.io/BESTANDSNAAM**. Bestandsnaam vervang je door de naam van het manifest. Deze uitleg staat ook nog een keer in de **README** onder de GitHub Repository zodat je gemakkelijk een link kan maken voor gelijk welk bestand uit de repository.

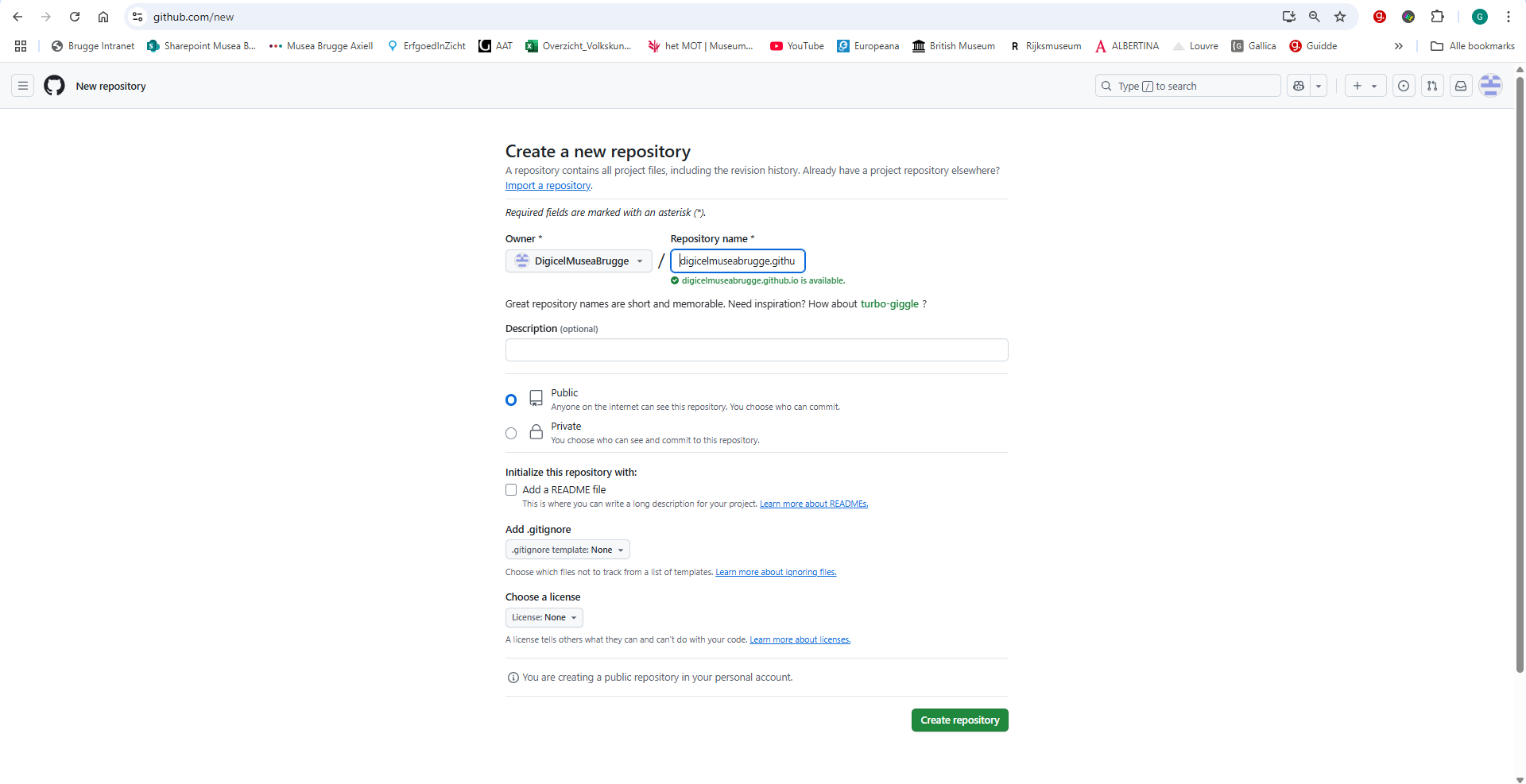


# Zelf een nieuwe repository met GitHub Pages site aanmaken

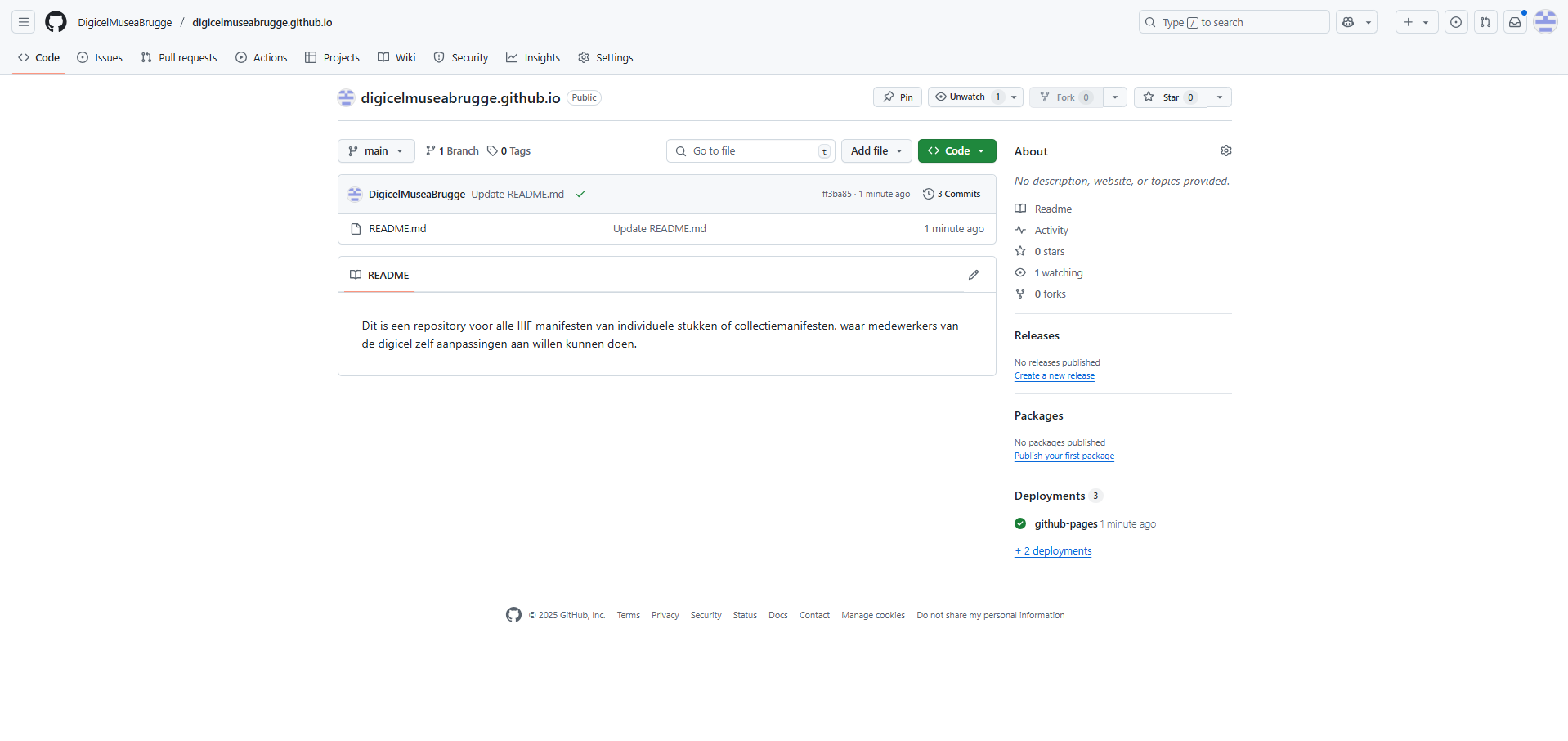
Als er iets zou mislopen of als je op een nieuwe account vanuit een repository bestanden online wil hosten kan het handig zijn om te weten hoe je zelf een repository met een actieve GitHub Pages site opzet. We zullen er bij deze handleiding vanuit gaan dat je een volledig nieuwe repository wil aanmaken. Klik eerst op **Create repository** in je **Dashboard** als je account al een repository heeft kan je een nieuwe repository maken via het **plusteken** rechts bovenaan de pagina.



Eens je op de pagina zit om een nieuwe repository aan te maken moet je een **eigenaar** selecteren en een **naam** geven. De selectie van eigenaar zal in de meeste gevallen jezelf als enige keuze hebben. Voor de naam combineer je je **gebruikersnaam** met “.github.io”. Belangrijk hierbij is ook dat je alleen maar **kleine letters** gebruikt. Voor DigicelMuseaBrugge wordt de juiste naam dus **digicelmuseabrugge.github.io**.



Aangezien we een gratis account gebruiken is het belangrijk dat we bij **Visibility** kiezen voor **Public**. Als je hier een fout maakt kan je dit altijd later nog aanpassen bij Settings**.** Vink vervolgens **Add a README file** aan. In de Readme kan je dan later een korte beschrijving van de repository toevoegen, dit maakt het gemakkelijk voor jezelf en andere gebruikers om te weten waar de repository voor dient. Daarnaast is het ook nodig voor de GitHub Page om te functioneren. Klik ten slotte op **Create repository** om je creatie te voltooien.



Het is nu tijd om onze **GitHub Pages site** te configureren. Ga naar **Settings** in de grijze balk bovenaan. Ga vervolgens naar **Pages.** Als alles goed gegaan is staat daar nu een **link naar je GitHub Pages site**. Controleer of bij de instellingen **Source: Deploy from branch** is en **Branch: Main > /(root)**. Als dit nog niet het geval is pas dit dan aan. Als je meer ervaring hebt of met complexere repositories werkt kan je hier met andere instellingen beginnen werken.

