

Werken in teamverband aan een Visual Studio project

Algemeen

Net zoals voor de projecten in het eerste jaar maken we gebruik van GIT als versiebeheersysteem.

We gaan er van uit dat de nodige software reeds geïnstalleerd is op je laptop. Indien dit niet zo is breng je een bezoekje aan <http://git-scm.com/downloads> en installeer je GIT. Verzorg je instellingen via Git Bash:

```
git config --global user.name "Voornaam Familienaam"
git config --global user.email voornaam.familienaam.r1234@student.hogent.be
git config --global push.default simple
git config --global core.autocrlf true (enkel Windows)
```

Ter opfrissing en ter verdieping van je kennis over Git raadpleeg je het boek 'Pro Git' op <http://git-scm.com/book/nl/v2/> Om de rest van dit verhaal te begrijpen moet je hoofdstukken 1, 2, 3 en 5 doornemen.

GitHub

Je vindt je repositories terug op <https://github.com/HoGentProjectenII>. Na toewijzing aan een team heb je toegang tot 2 repositories: 1 voor java, 1 voor .Net.

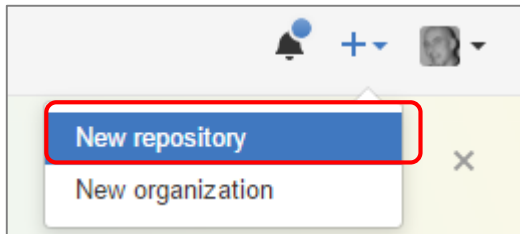
Als je klikt op de repository (hier groep00DotNet) krijg je onderstaande pagina te zien:

The screenshot shows the GitHub interface for the repository 'groep00DotNet' under the organization 'HoGentProjectenII'. The repository is marked as 'PRIVATE'. At the top, there are buttons for 'Unwatch', 'Star' (5), and 'Fork' (0). Below this is a navigation bar with tabs for 'Code', 'Issues' (0), 'Pull requests' (0), 'Wiki', 'Pulse', 'Graphs', and 'Settings'. The main content area shows the repository name 'Didactische Leermiddelen' and an 'Edit' button. Below this, there are statistics: '1 commit', '1 branch', '0 releases', and '1 contributor'. A 'New pull request' button is visible. The 'Branch: master' dropdown is shown. There are buttons for 'New file', 'Find file', 'HTTPS', and 'Download ZIP'. The 'HTTPS' button is highlighted with a red box, and a red callout bubble points to it with the text 'deze link heb je straks nodig...'. Below the buttons, there is a list of files: '.gitignore', 'README.md', and 'test'. Each file has a link to its commit history. The 'test' file's commit history is expanded, showing a commit by 'ssamyn' with the message 'Initial commit' and the hash 'f8cb9ef' from 'a minute ago'. The 'README.md' file is also shown with its commit history. At the bottom, there is a section for the 'README.md' file content, which includes the title 'groep00DotNet' and the subtitle 'Didactische Leermiddelen'.

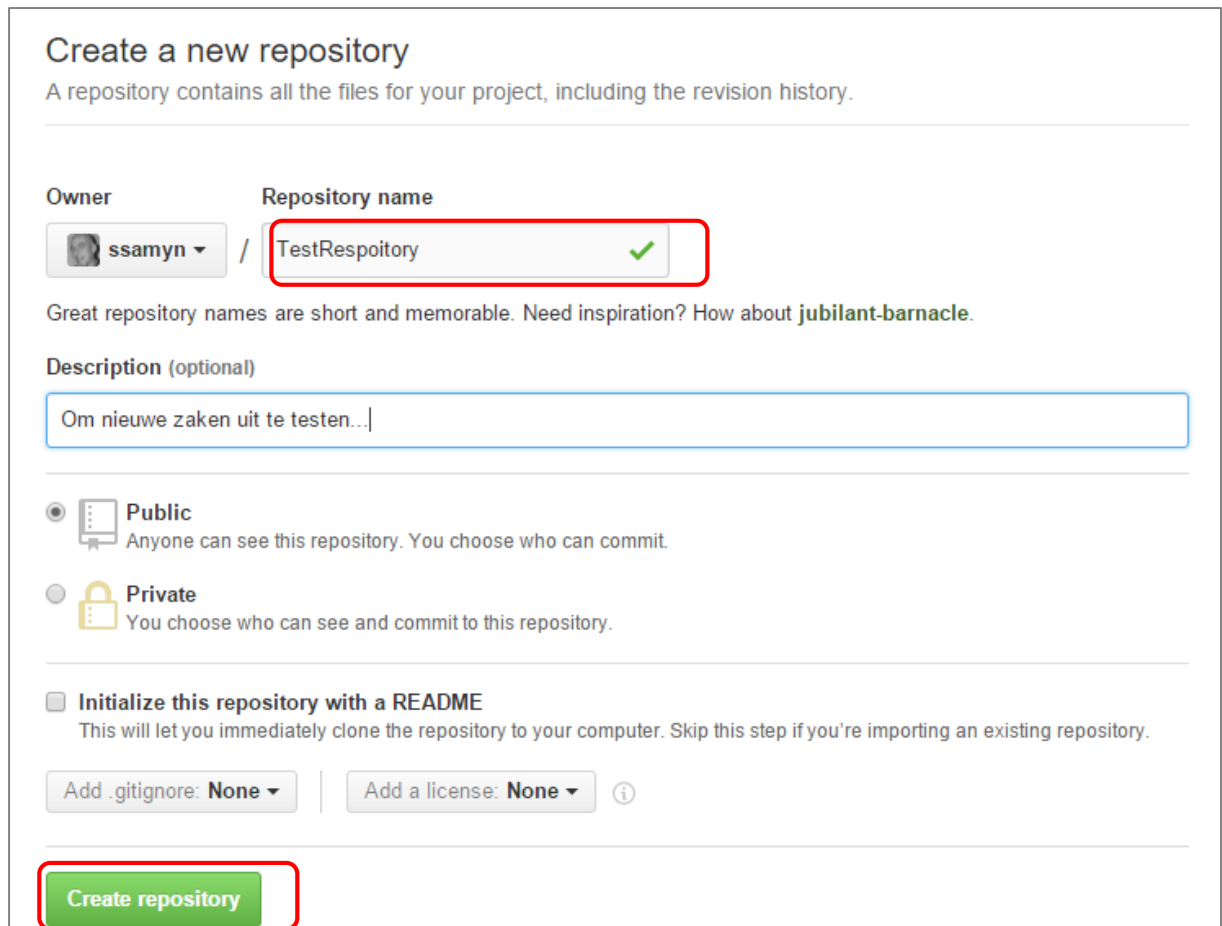
Vanaf nu kan je verderwerken in Visual Studio. Je maakt best eerst eens een eigen publieke repository aan om een aantal zaken eens uit te proberen zonder dat je je private project repository reeds bezoedelt.

Een eigen repository maak je als volgt aan

1. Klik naast je username op het + icoontje en kies New repository.



2. Kies een goeie naam voor de repository, vul eventueel de description in en klik op create repository

A screenshot of the 'Create a new repository' form in GitHub. The form has a title 'Create a new repository' and a subtitle 'A repository contains all the files for your project, including the revision history.' Below this, there are two main sections: 'Owner' and 'Repository name'. The 'Owner' section shows a dropdown menu with 'ssamyn' selected. The 'Repository name' section shows a text input field with 'TestRespoitory' entered, which is highlighted with a red rectangle and has a green checkmark next to it. Below these fields, there is a text input field for 'Description (optional)' with the text 'Om nieuwe zaken uit te testen...'. Further down, there are two radio button options: 'Public' (selected) and 'Private'. At the bottom, there is a checkbox for 'Initialize this repository with a README'. Below the checkbox, there are two dropdown menus: 'Add .gitignore: None' and 'Add a license: None'. At the very bottom, there is a green 'Create repository' button, which is highlighted with a red rectangle.

3. De aangemaakte repository wordt getoond. Je kan andere teamleden uitnodigen om mee te werken aan je project. Klik op Settings > Collaborators en voer de username van je teamleden in. De "clone Url "(groene kader) heb je straks nodig binnen VS.

ssamyn / TestRespoitory

Unwatch 1

Star 0

Fork 0

Code

Issues 0

Pull requests 0

Wiki

Pulse

Graphs

Settings

Quick setup — if you've done this kind of thing before

Set up in Desktop

 or

HTTPSSSHhttps://github.com/ssamyn/TestRespoitory.git

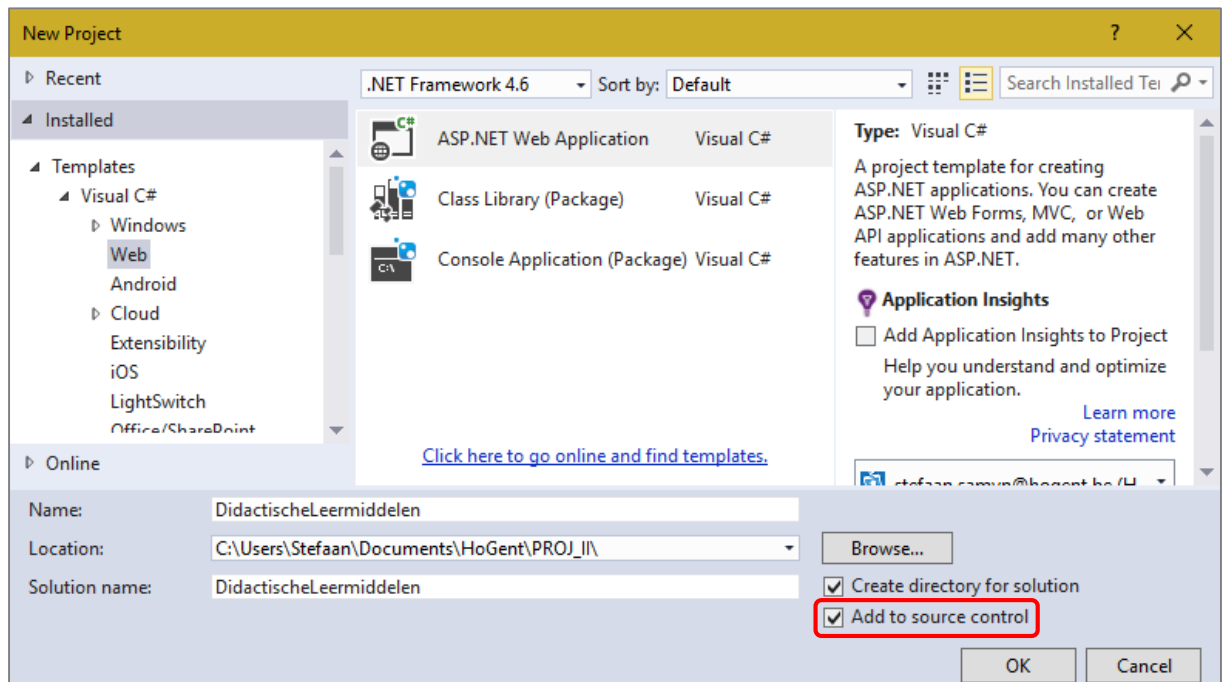
We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

Aanmaken van een project in Visual Studio in een repository

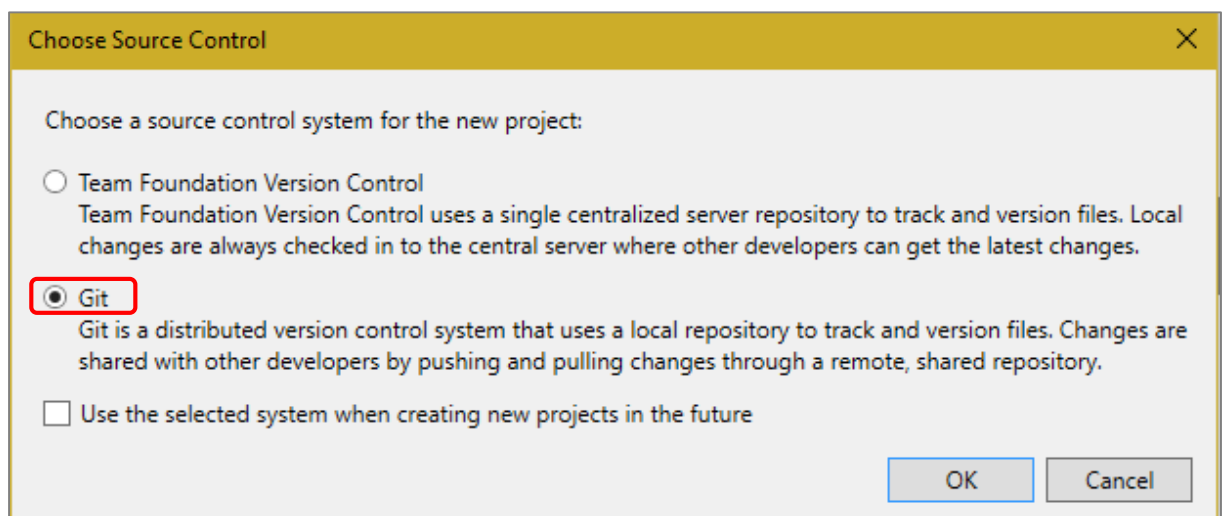
Deel 1

Dit dient **door 1 student uit het team** te worden uitgevoerd. De andere teamleden volgen deel 2.

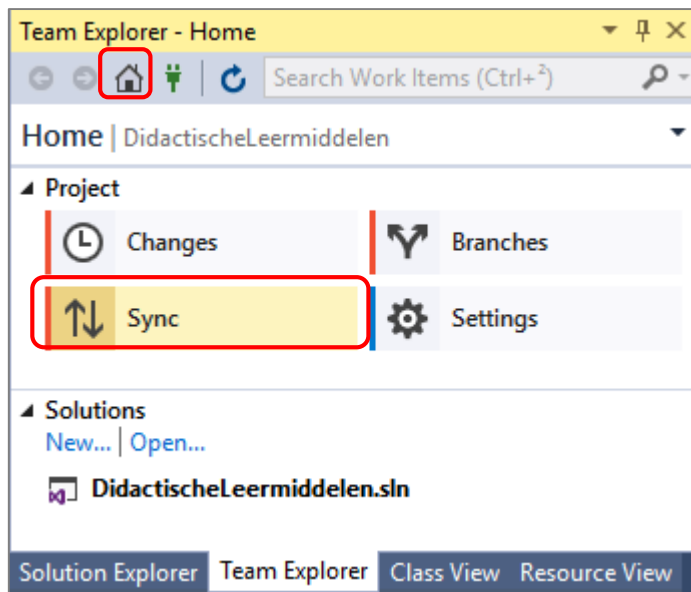
1. Start Visual Studio
2. Maak een nieuw project aan binnen Visual Studio. File > New Project. Kies Visual C#, Web, ASP.NET Web application. Geef een naam in en selecteer een locatie. De naam is belangrijk. Dit wordt je Repository. Vink Add to source control aan. Klik OK



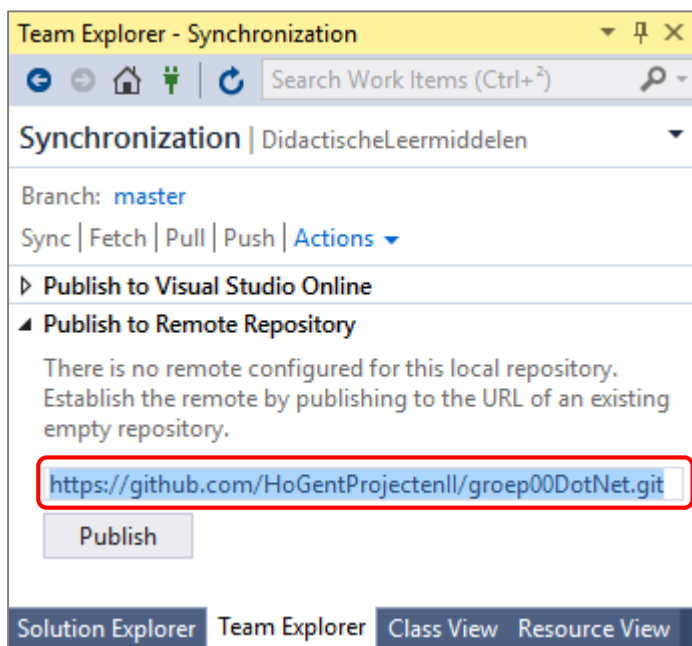
3. Kies vervolgens het MVC template en vink unit tests aan.
4. Vervolgens zal je moeten kiezen voor het gewenste versiebeheersysteem. Kies Git.



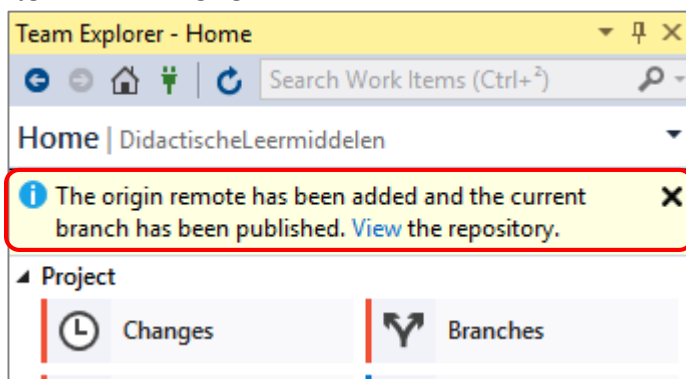
5. Open de Team Explorer (View > Team Explorer)
6. Klik op het Home > Sync.



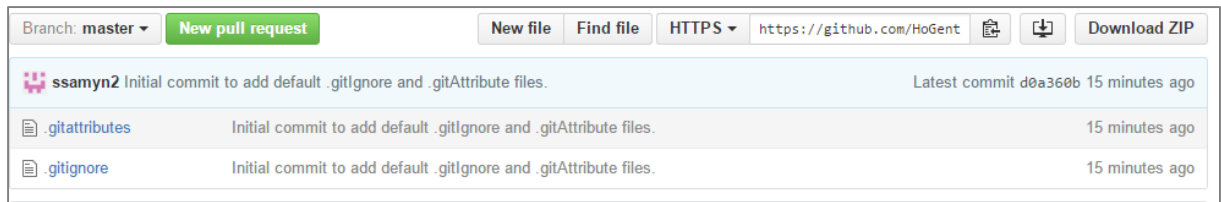
7. Kies 'Publish to remote repository', vul de clone url uit de vorige sectie in en klik Publish



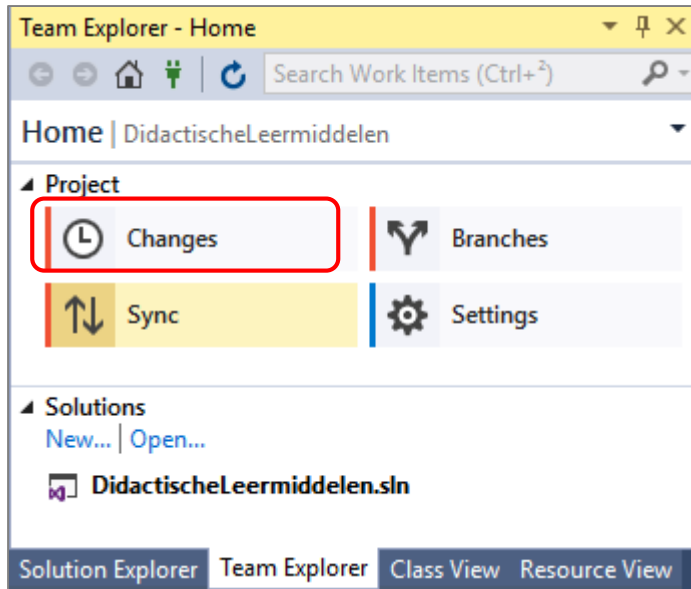
8. Je krijgt een bevestiging.



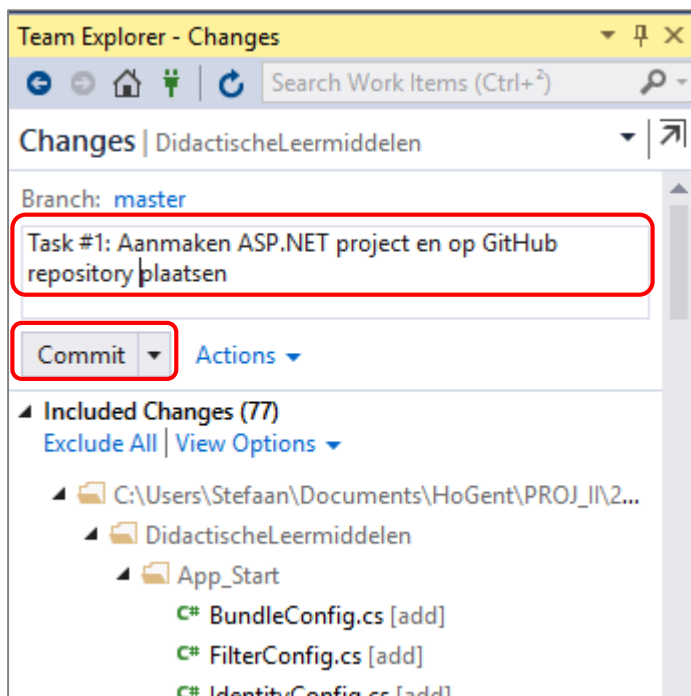
9. Op de remote repository werden de .gitignore en .gitattributes op de master geplaatst.



10. Nu dien je het project er nog op te plaatsen. In Team Explorer. Klik op het Home icoontje. Kies Changes.



11. Dan zie je een lijst van je project en alle bestanden die nog aan de commit moeten worden toegevoegd. Committen **gebeurt lokaal**. Included Changes toont de lijst (normaalgezien de wijzigingen sinds de laatste commit). Hier kan je beslissen wat je wenst te committen (je kan wijzigingen excluden). Vul een commit message in. Gebruik het taaknummer uit de sprint backlog en geef dan wat extra uitleg. Klik vervolgens op commit.



If it's too hard to write good commits, just follow this template:

- Each commit should have a purpose. Is it fixing a bug, or adding a new feature, or removing an existing feature?
- Only one change per commit. Commit when you solve issue #127, and only that issue.
- Include issue number in the commit message.
- Include a description of the change in commit description. This depends on the project contributing guidelines but usually you mention what was causing the bug, and how you fixed it, and how to test the change.
- Write sensible commit message:

fixed cache being reset on every insert caused by missed access after write. fixes #341

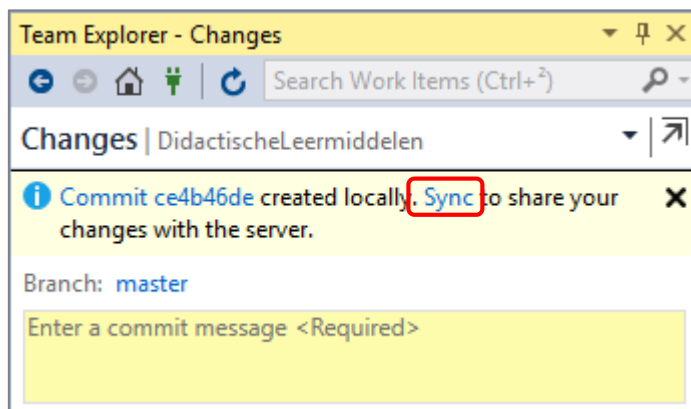
or...

added a new form in header to make it easier to collect leads. close #3

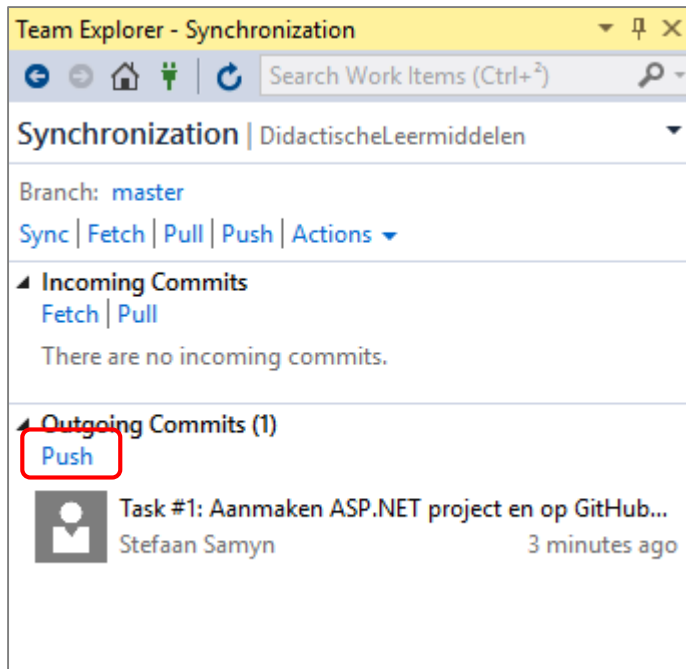
and definitely not

remove stuff because why not.xoxo

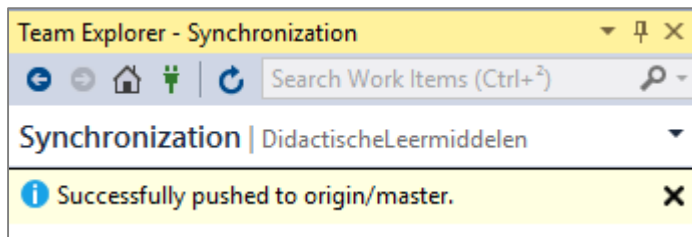
12. Je komt op onderstaand scherm. Nu moet je nog de commits naar de remote repository doorsturen (push-en). Klik op de Sync hyperlink.



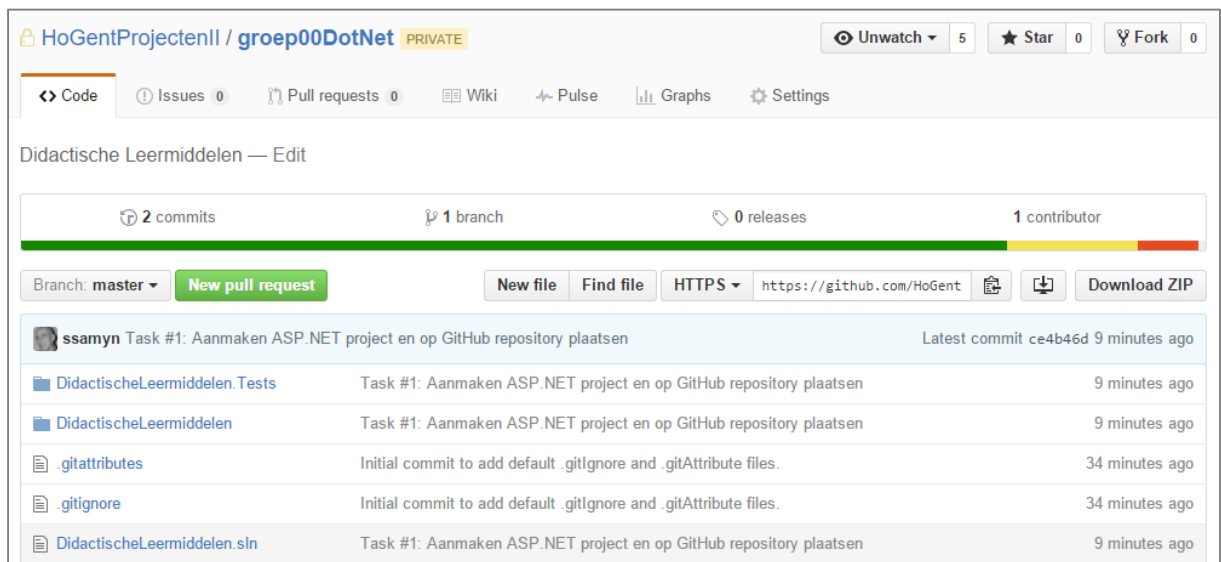
13. Je komt op onderstaande. Klik dan Push.



14. Je krijgt terug een melding.



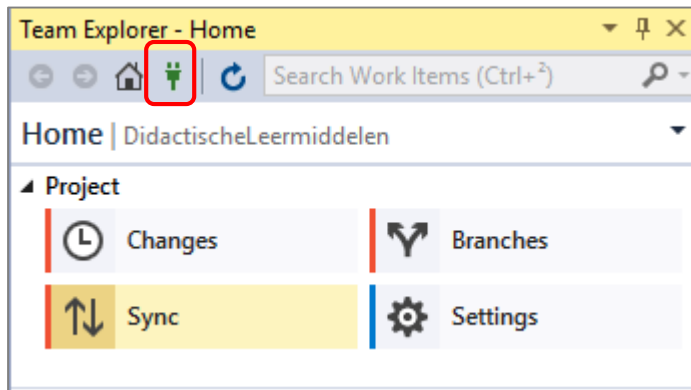
15. Als je nu naar github gaat en je klikt terug op je repository krijg je onderstaande te zien.



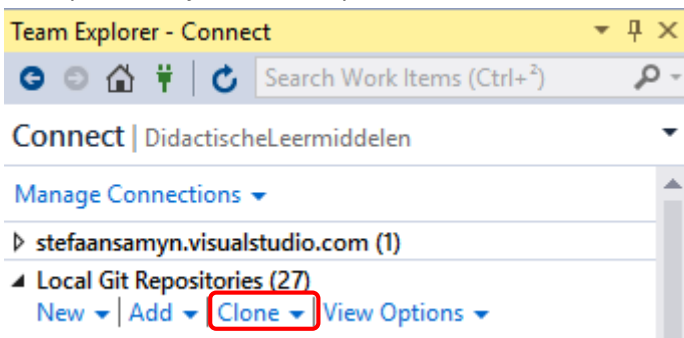
Deel 2

Dit deel is voor de teamleden die deel 1 niet uitvoerden.

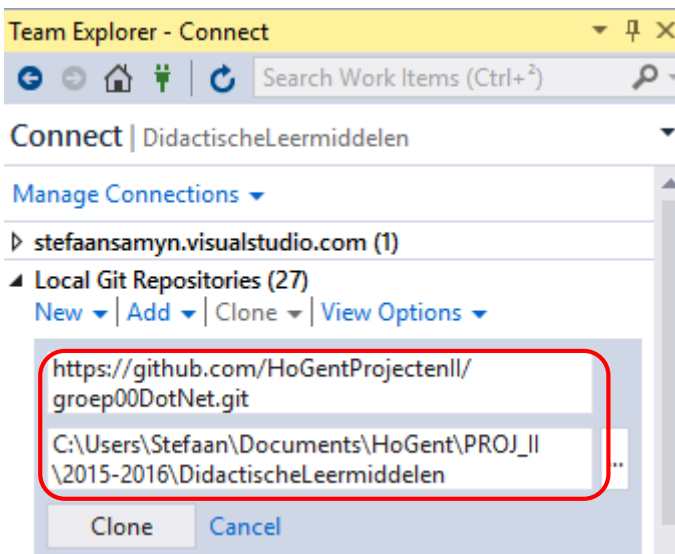
1. Open Visual Studio en open de Team Explorer (View > Team Explorer). Klik op de connect icon.



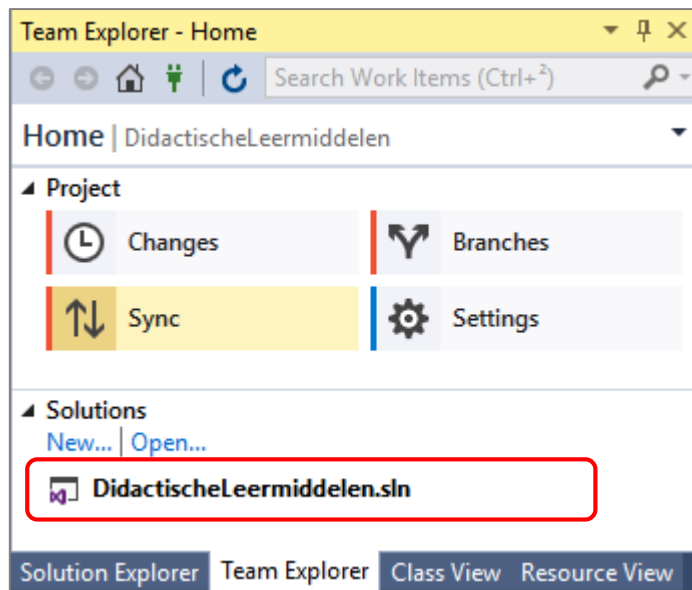
2. Klik op Clone bij Local Git Repositories



3. Vul de "clone Url" uit (zie vorige sectie) in in het eerste tekstvak. In het tweede tekstvak browse je naar een folder waar je de repository lokaal zal aanmaken. Dit is ook de naam van je solution folder. Klik op Clone.



4. Dubbelklik op de aangemaakte lokale repository, je krijgt een venster met de solution te zien. Dubbelklik op de solution om ze te openen...



5. **Build** het project zodat de referenties goed komen. Nu kan je aan het project werken.

Een workflow voor werken in team

Een mogelijke workflow

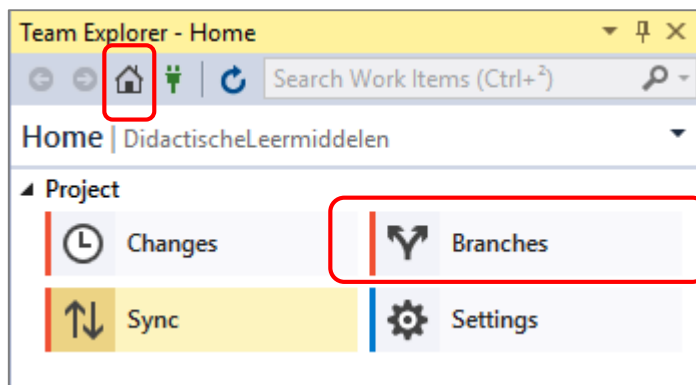
- 1 student maakt bij het begin van een sprint een nieuwe **branch** aan en publiceert deze
- De andere studenten halen de branch lokaal binnen
- Binnen een sprint werkt elke student aan zijn toegewezen taken
 - Elke student werkt lokaal aan zijn taak. Als de taak volledig is afgewerkt en getest, commit hij/zij de taak lokaal. In de commit boodschap wordt naar de taak verwezen via het taaknummer. Bvb Task #12: repositories added.
 - Nu kan de student ook beslissen om de commit te synchroniseren met de server. Mogelijks dient eerst een conflict te worden opgelost.
- Op het einde van de sprint, als de branch volledig werkt, merged een student de branch met de master. Zorg ervoor dat de **master branch alleen volledig stabiele code bevat**.

Aanmaken van een branch.

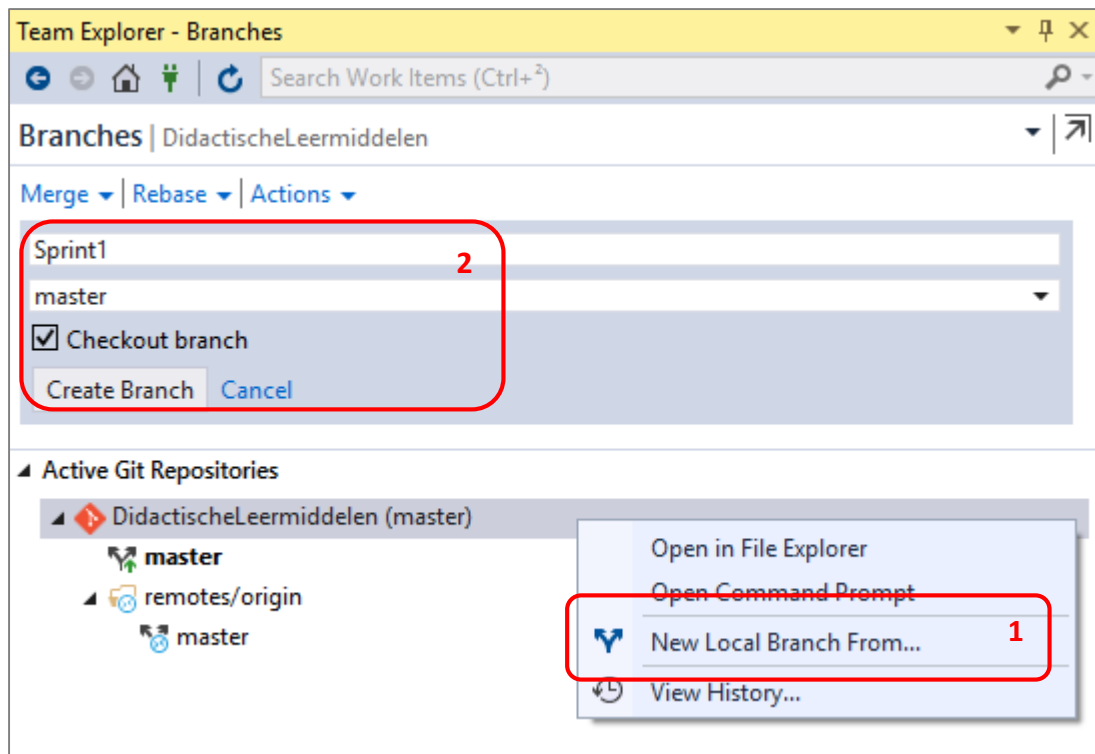
Wat is een branch ? <http://git-scm.com/book/nl/v2/Branchen-in-Git-Eenvoudig-branchen-en-mergen>

Voor het aanmaken van een branch in Visual Studio ga je als volgt te werk:

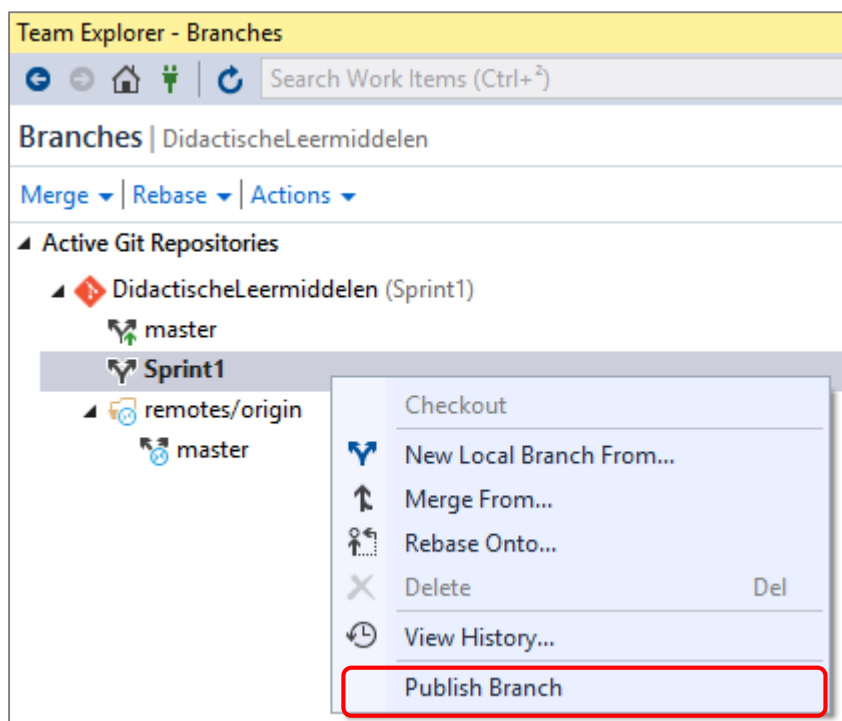
1. Klik op Home > Branches.



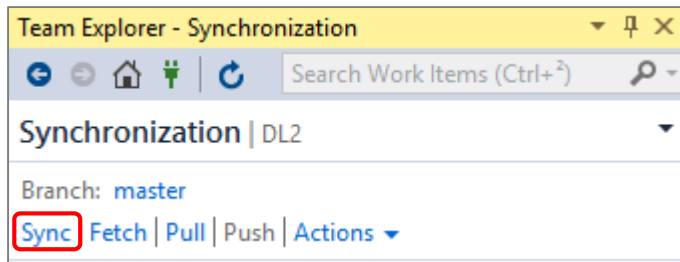
2. Rechtermuisklik op de repository, of op een branch en kies je voor New Local Branch From.... Geef een naam in voor de nieuwe branch, bvb de naam van de sprint, en geef aan van welke branch je deze branch wil maken. Vink Checkout Branch aan (hierdoor schakel je over naar deze branch). Klik Create Branch.



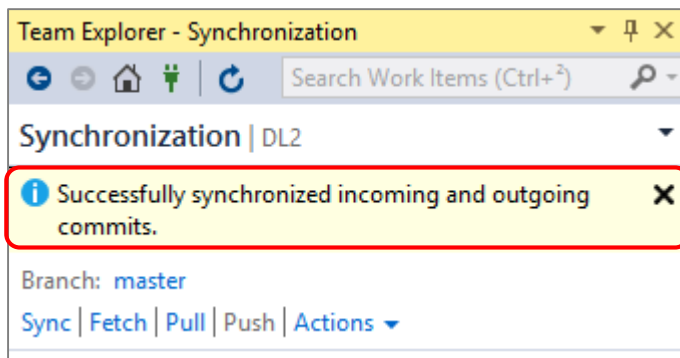
3. Nu moet je de Branch nog publish-en naar de remote repository. Rechtermuisklik op de branch en kies Publish



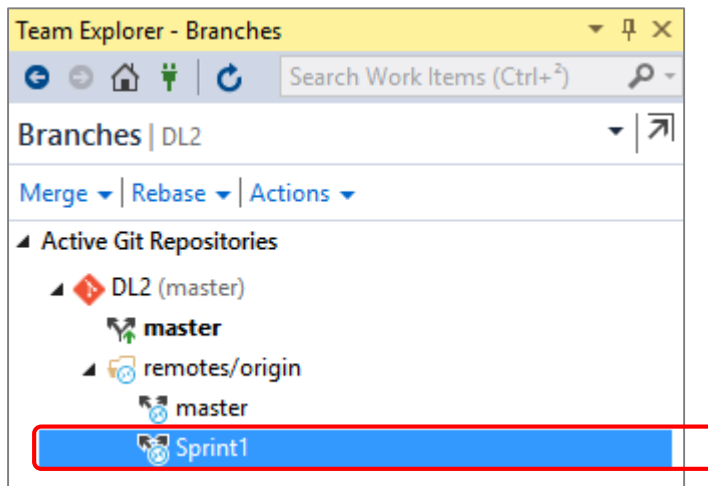
4. De andere groepsleden kunnen nu de branch lokaal binnenhalen. Klik Home > Sync en voer een sync uit...



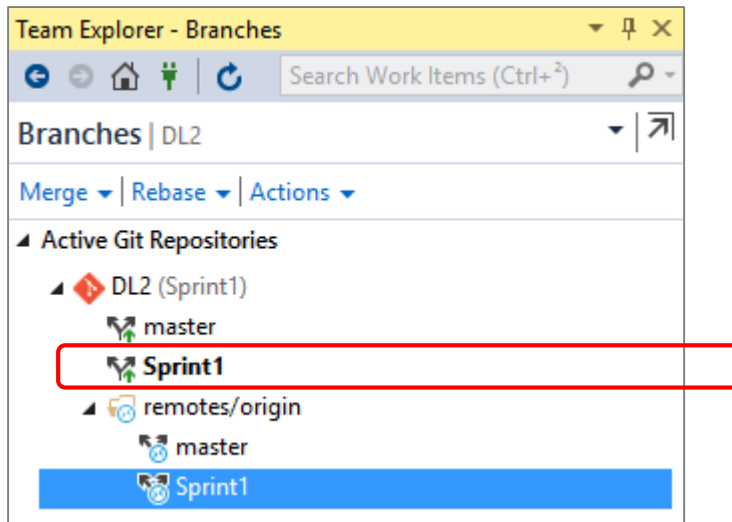
Je krijgt een boodschap dat dit succesvol werd uitgevoerd:



5. Klik Home > Branches. Je ziet dat de nieuwe branch op de remote server werd opgepikt.



6. Rechtermuiklik op de remote branch en doe een checkout. Nu heb je de branch ook lokaal en kan je commits uitvoeren in deze branch...



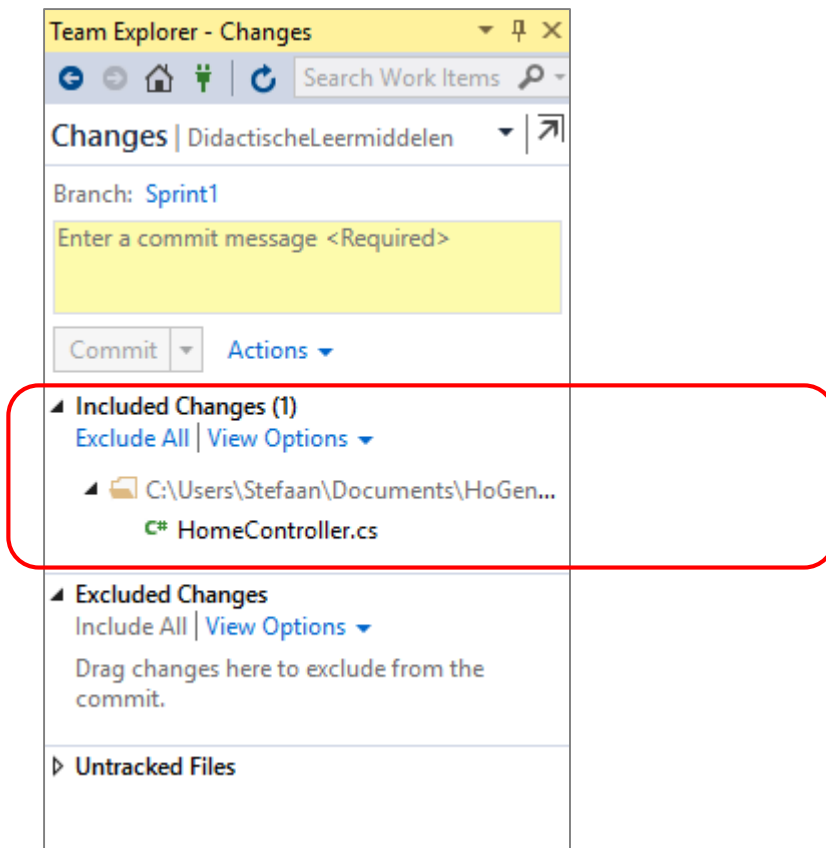
7. Vanaf nu kan het team samenwerken in dezelfde branch.
Meer info op <https://msdn.microsoft.com/en-us/library/jj190809.aspx>

Aanbrengen van wijzigingen

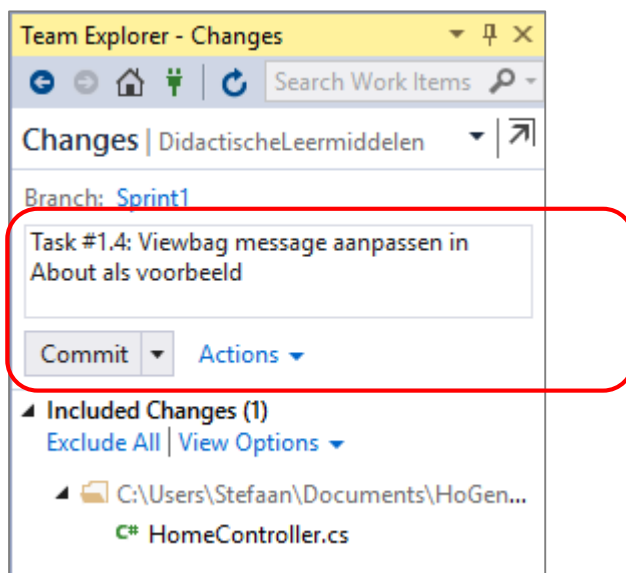
Breng eerst alle wijzigingen lokaal aan. Commit enkel als je een werkende versie hebt. In de commit boodschap kan je naar de taak verwijzen via het taaknummer. Bvb #12: Repo added.

Zorg dat je in de juiste branch werkt (Klik Home icon > Branches > Kies de juiste branch)

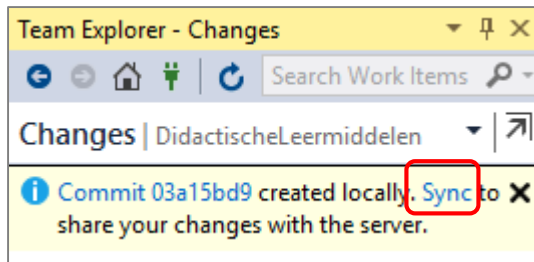
1. Open de Solution explorer en breng wijzigingen aan in je project. In dit voorbeeld gaan we enkel de ViewBag message in de About methode van de HomeController aanpassen. In realiteit werk je je volledige taak af.
2. Als alle wijzigingen werken en getest zijn, commit je je werk (lokaal!). In de Team Explorer > Home icon > Changes: toont alle wijzigingen die lokaal nog niet gecommited geweest zijn.



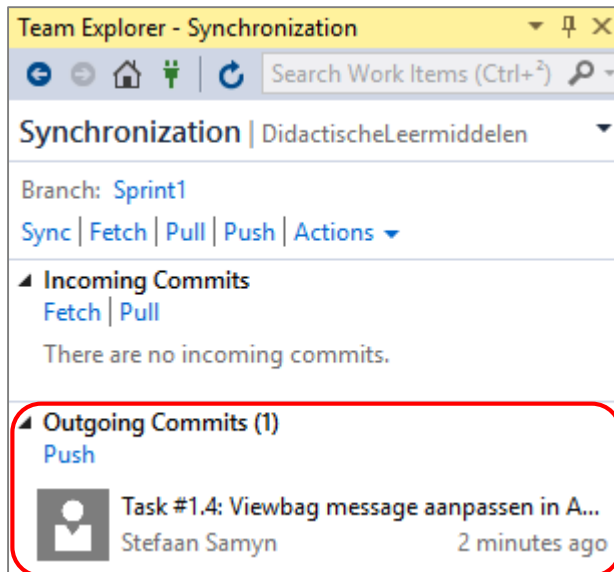
3. Je kan eventueel een aantal changes excluden zodat je een werkende versie kan committen. Geef bovenaan een commit message in (taaknummer - omschrijving) en klik commit



4. Nu moet je de commit nog doorvoeren op de remote repository Visual Studio suggereert direct om te Sync-en. Klik Home icon > Unsynced Commits. Klik Sync.



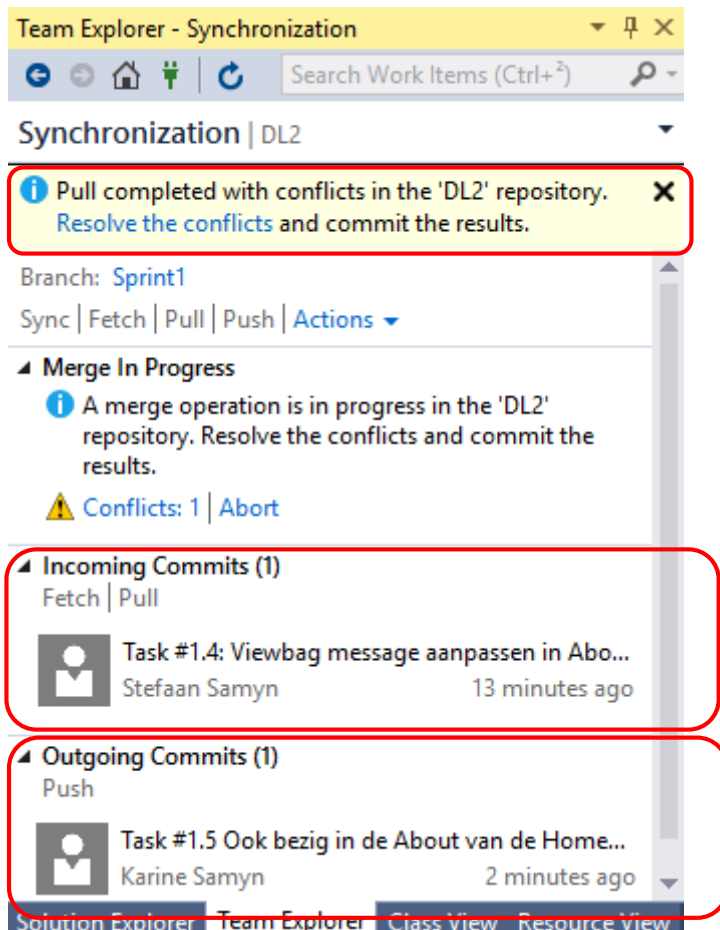
Je kan ook steeds de commits zien die nog niet remote werden gemaakt via Home > Sync bij de Outgoing Commits. Klik op Sync of op Push...



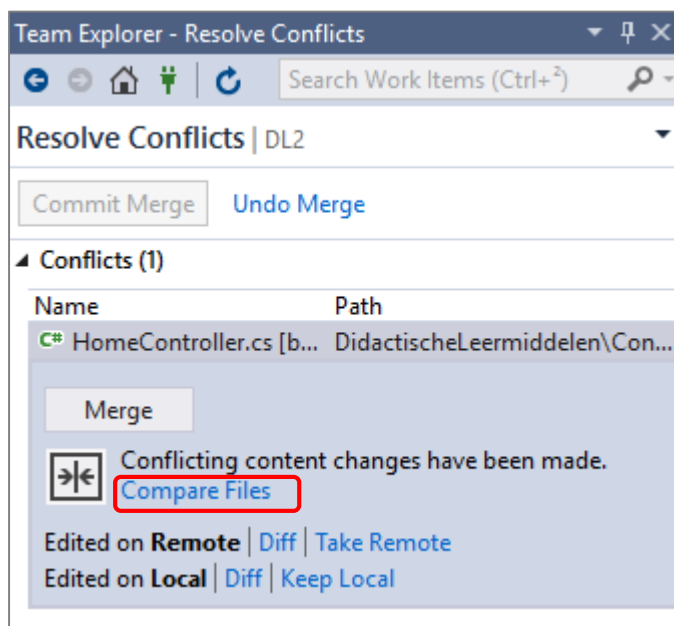
5. Het kan zijn dat je wijzigingen conflicteren met wat er al op de server staat. Dan dien je eerst het conflict op te lossen (zie puntje conflicten oplossen)

Conflicten oplossen.

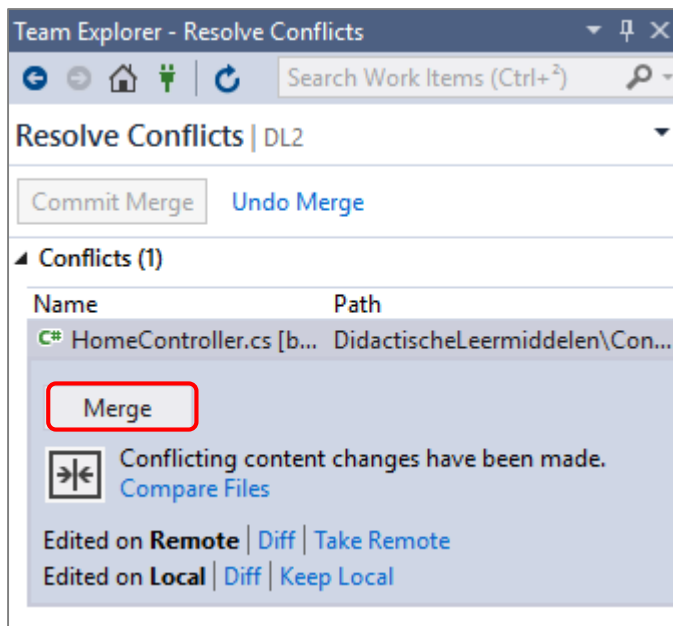
Wanneer je je commits synchroniseert met de server, kan het zijn dat er conflicten optreden. In onderstaand voorbeeld probeert de gebruiker te Sync-en maar er is een conflict opgetreden. Het conflict moet opgelost worden en dan moet er gecommitt worden...



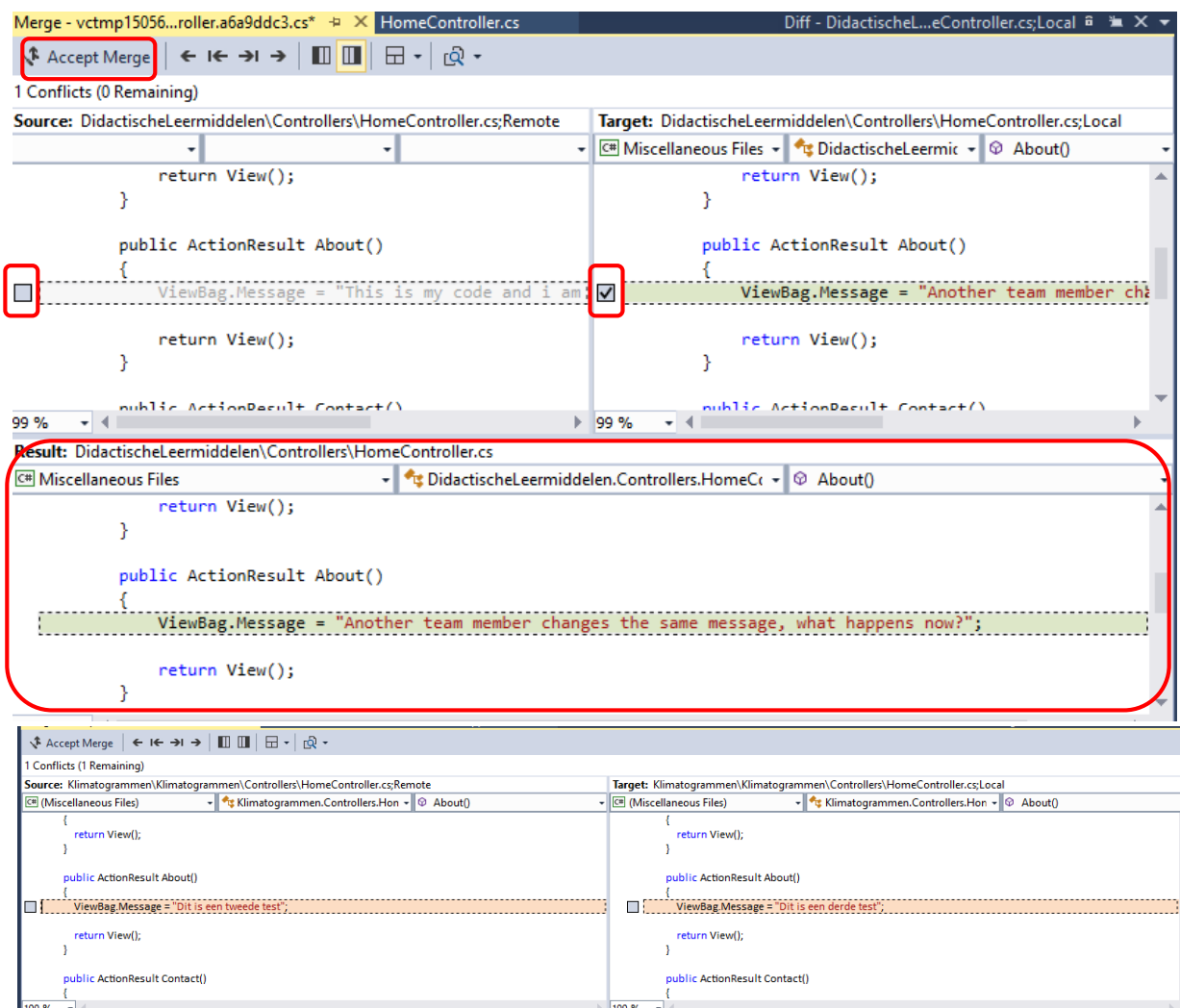
1. Klik op de link "Resolve the conflicts" bovenaan dit venster. De files met conflicten worden getoond. Als je op een file klikt opent het blauwe kader. Klik op Compare files.



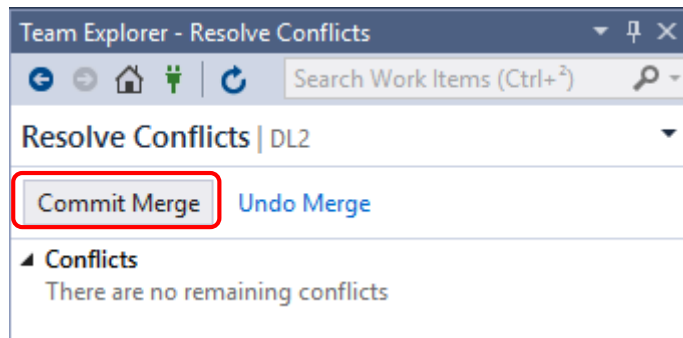
2. Compare files toont de 2 versies naast mekaar. Klik op "Merge" in het blauwe kader.



3. Er verschijnen checkboxen voor de verschillende lijnen code. Vink de versie aan die je wenst te behouden (je kan het resultaat in het onderste venster bekijken) en klik bovenaan op Accept Merge wanneer je het conflict hebt opgelost...



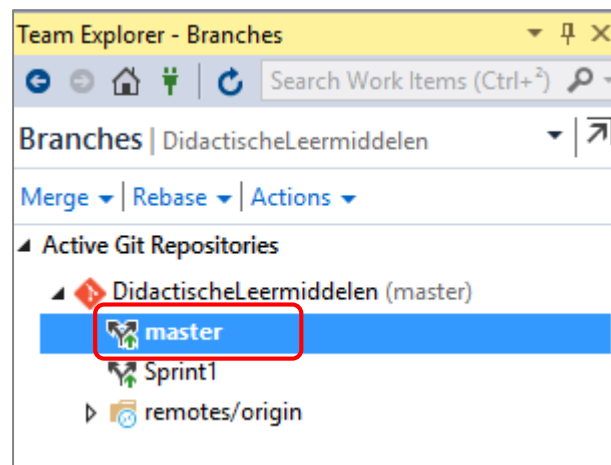
4. Het team explorer venster wordt aangepast. Klik nu op Commit Merge. Geef een commit boodschap in. Sync vervolgens de wijziging naar de server.



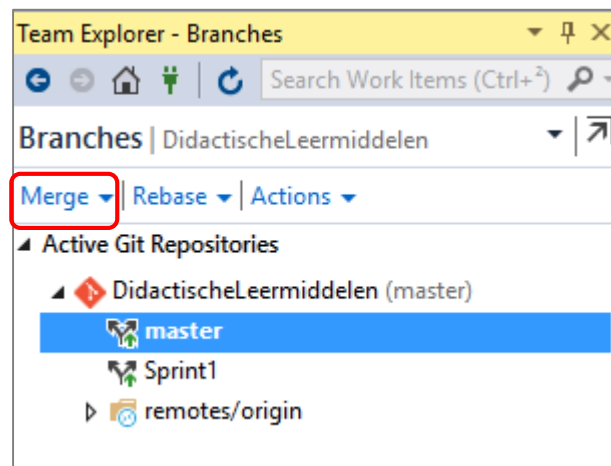
Meer info op <https://msdn.microsoft.com/en-us/library/dd286559.aspx>

Mergen van een branch met de master

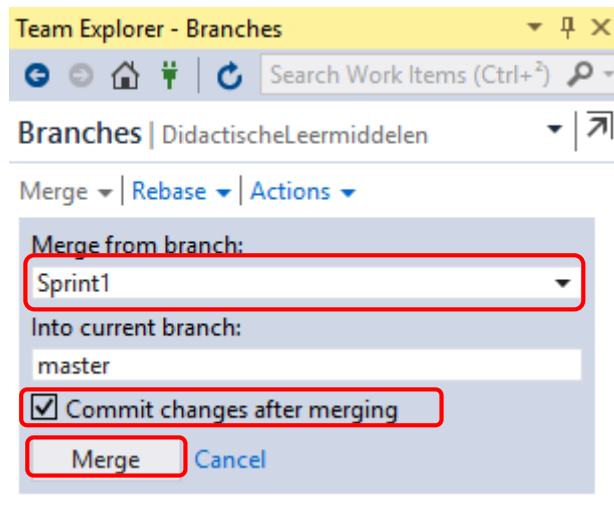
1. Op het einde van de sprint, als de code volledig werkt en alle teamleden akkoord gaan moet 1 teamlid de Branch mergen met de master. In dit voorbeeld gaan we Sprint1 mergen in de master.
2. Ga naar Home > Branches. Zorg dat de branch waarin je een andere branch wil mergen de actieve branch is (hier is dit master; dubbelklik op master).



3. Klik op Merge...



4. Nu selecteer je Sprint1 als Merge from branch en zorg dat commit changes after merging aangevinkt is. Klik op dan op Merge



5. Ga naar Home > Sync en klik op sync om de merge ook op de remote repository door te voeren.
6. De andere teamleden moet je hiervan op de hoogte stellen, zodat ze terug naar de master kunnen overschakelen en een sync kunnen doen om de laatste versie van de master binnen te halen.

Revert van een commit

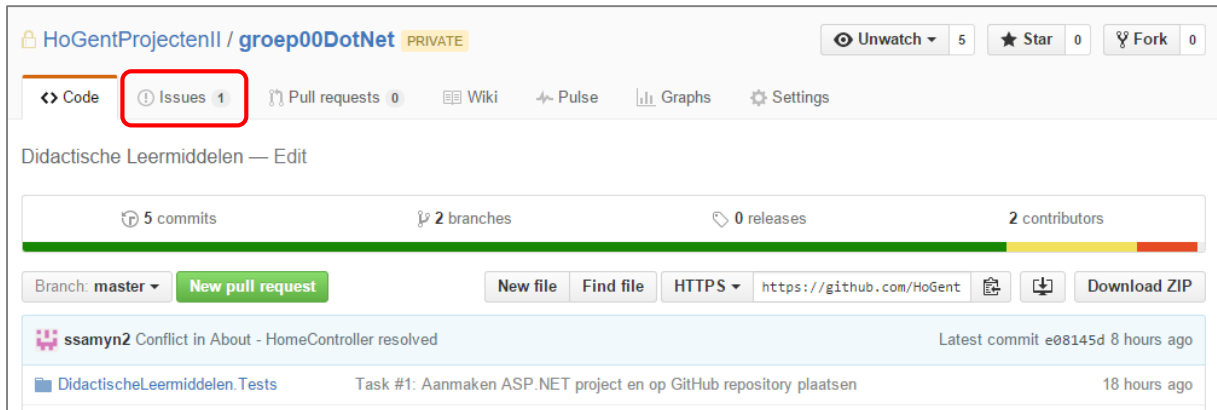
1. Ga naar unsynced commits
2. Doubleklik op de commit die je wenst terug te draaien. Dan heb je link Revert.

GitHub issues

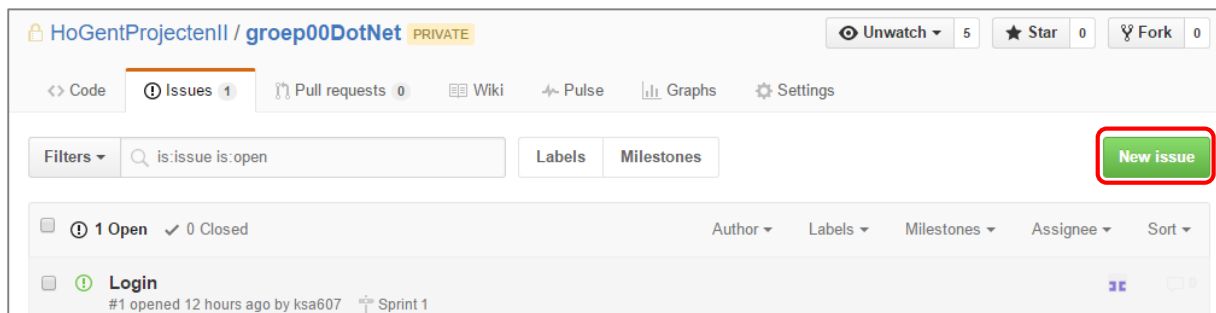
Het gebruik van Issues in GitHub, eventueel gekoppeld aan een collaboration tool zoals Trello, kan een grote hulp zijn om taken, voorgestelde verbeteringen, bugs, enz. op te volgen. Elke repository in GitHub heeft een Issues-section.

Openen van een Issue

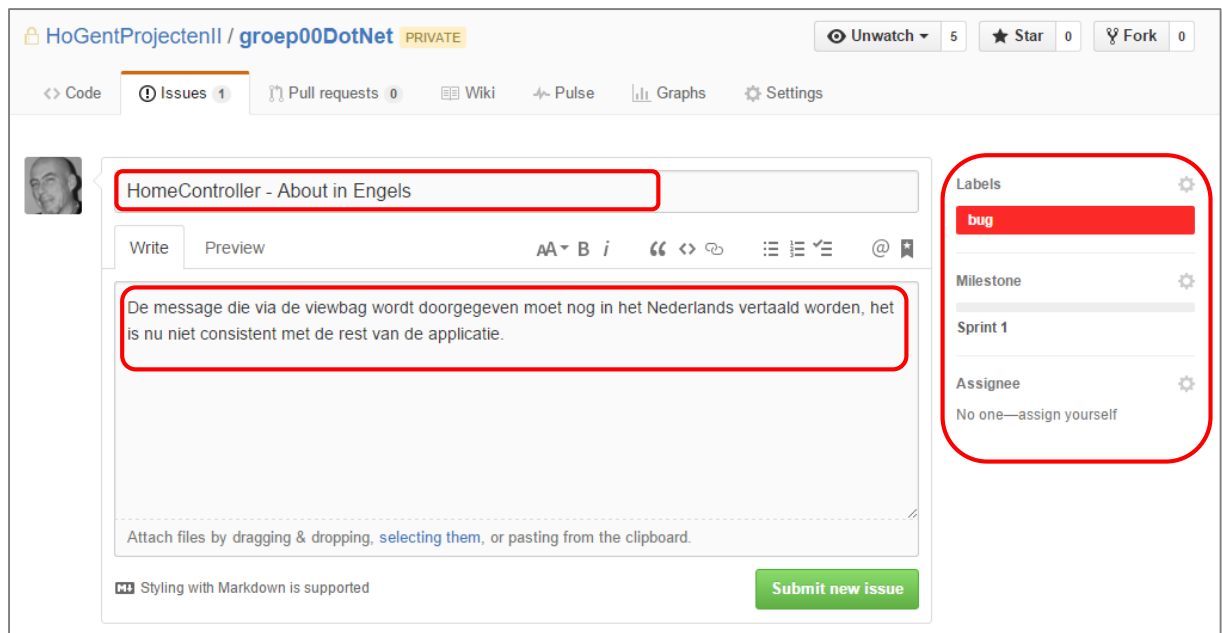
1. Ga naar de issues sectie van je repository



2. Klik op New issue.



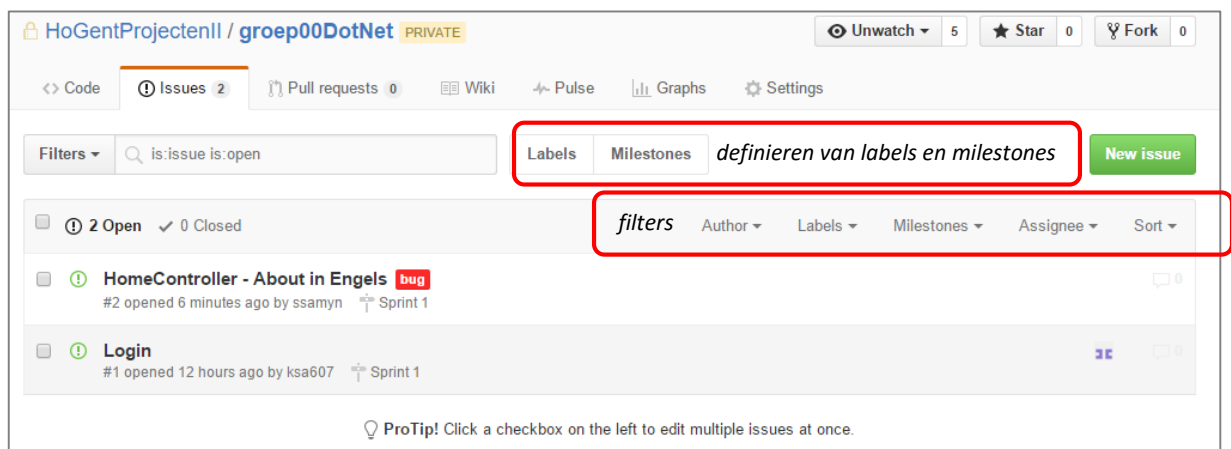
3. Vervolledig de Issue: vul de titel en description in. Geef een gepast label, en geef eventueel milestone aan. Desgewenst kan je de Issue ook aan een teamlid (dat is inclusief jezelf) toekennen.



4. Je kunt nu de issue submitten, ze komt nu voor in de lijst van issues en via een e-mail worden de betrokkenen op de hoogte gebracht van de issue.

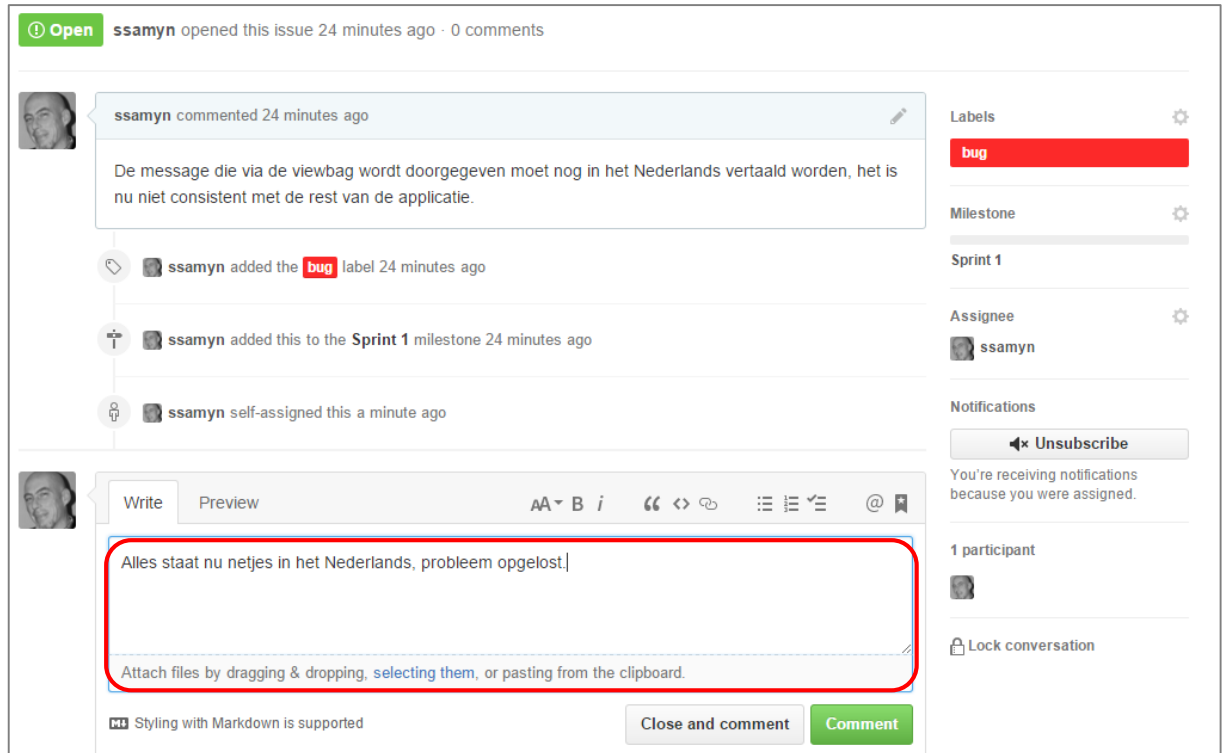
Merk op dat er een standaardlijst van Labels aanwezig is maar je kan deze volledig naar je hand zetten (klik op Labels). Klik op Milestones om je milestones te definiëren.

De lijst van issues kan gefilterd worden op auteur, label, milestone, ...

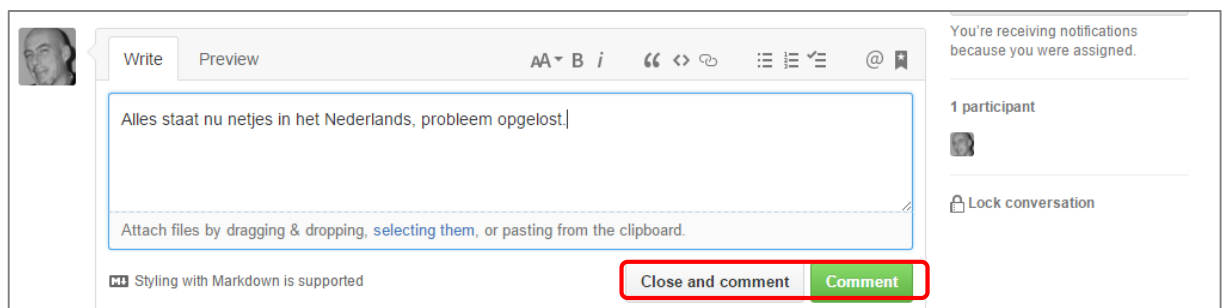


Opvolgen van een Issue

1. Wanneer je op een open issue klikt heb je de mogelijkheid om commentaar te geven.



2. Je kan klikken op 'Comment' om het commentaar te posten of op Close and comment. Deze laatste sluit de Issue af. Betrokken personen worden via e-mail van dit commentaar op de hoogte gebracht.



Documentatie

Boek Pro Git: <http://git-scm.com/book/nl/v2>

Use Visual Studio with Git: <https://msdn.microsoft.com/en-us/library/hh850437.aspx>

Op Pluralsight (e-learning)

1. Git Fundamentals
2. Git for Visual Studio Developers: enkel de onderdelen
 - a. Git vs Centralized Version Control
 - b. Working with Visual Studio: enkel het stukje Demo: Working with github (in VS 2015 zijn de Tool for Git reeds volledige geïntegreerd)