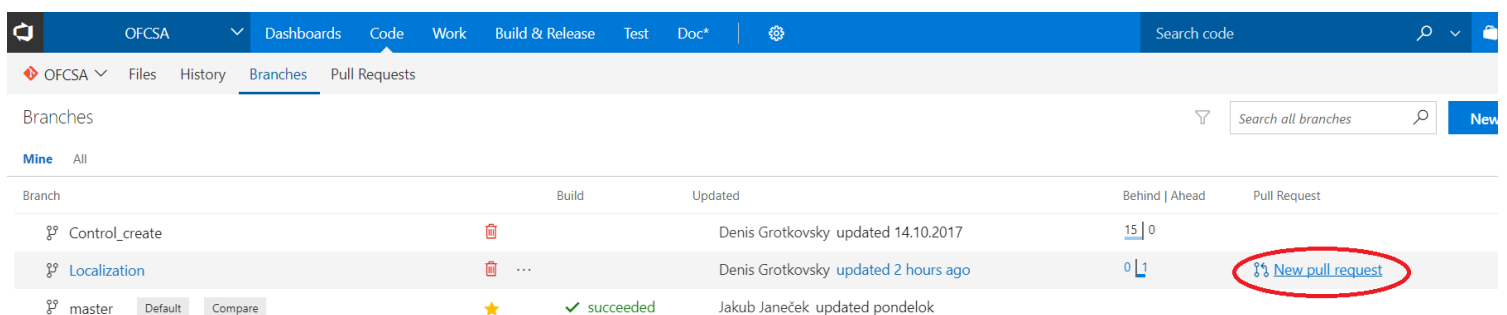
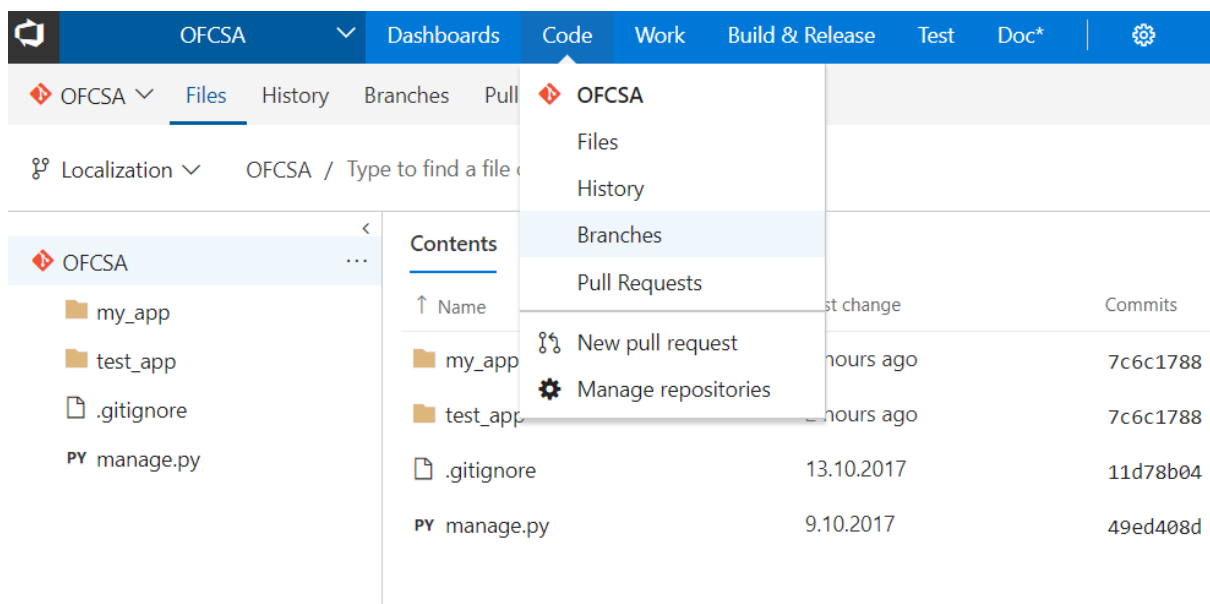


Metodika na code review

Pokyny pre autora:

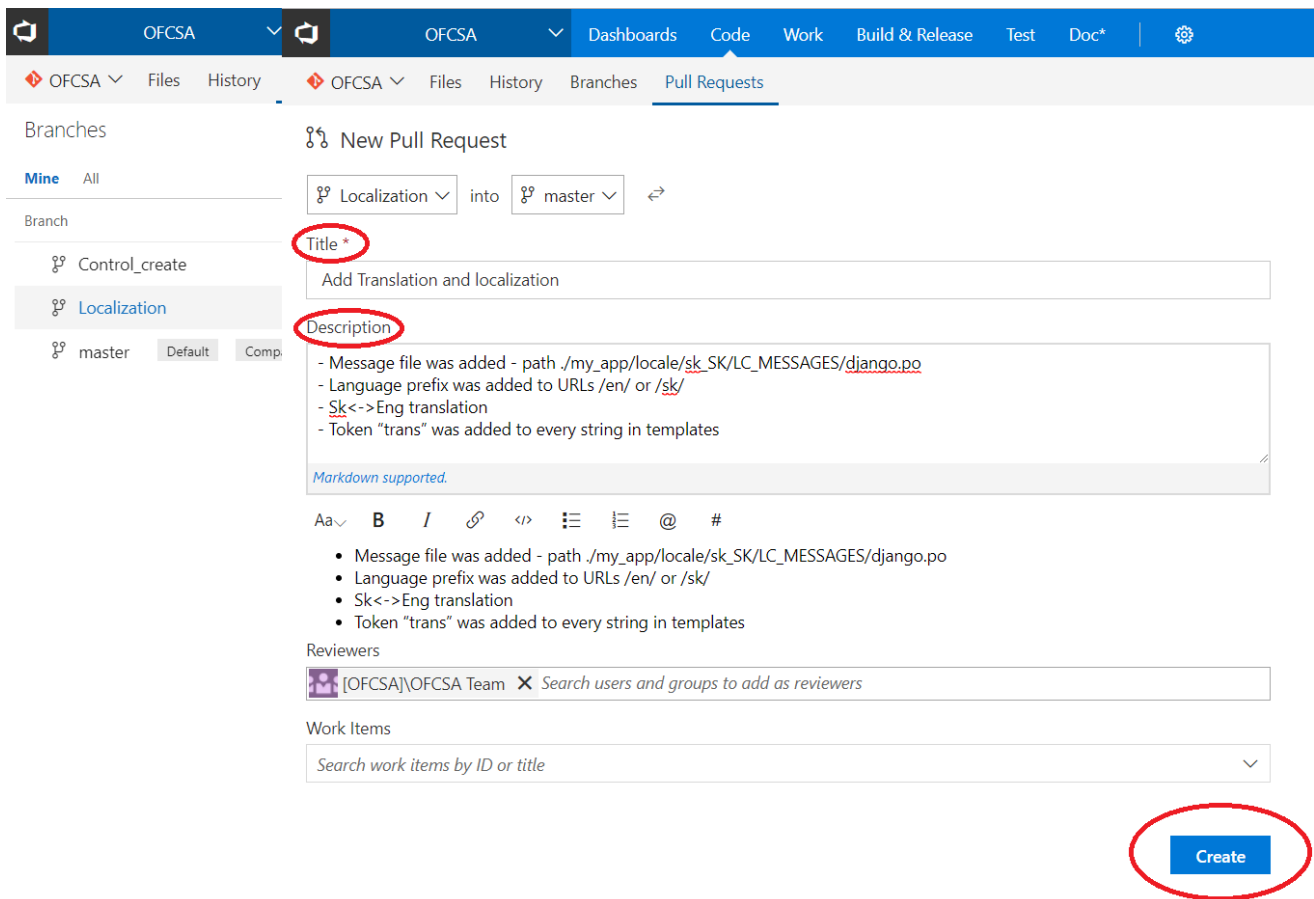
- po tom ako si vytvoril svoj kód, commitniho a vytvor novú branchu
- nasledovne túto branchu pushni na git
- ďalej budeš chcieť vytvoriť z tejto branche pull-request:

1. Na stránke TFS choď na Code -> Branches.

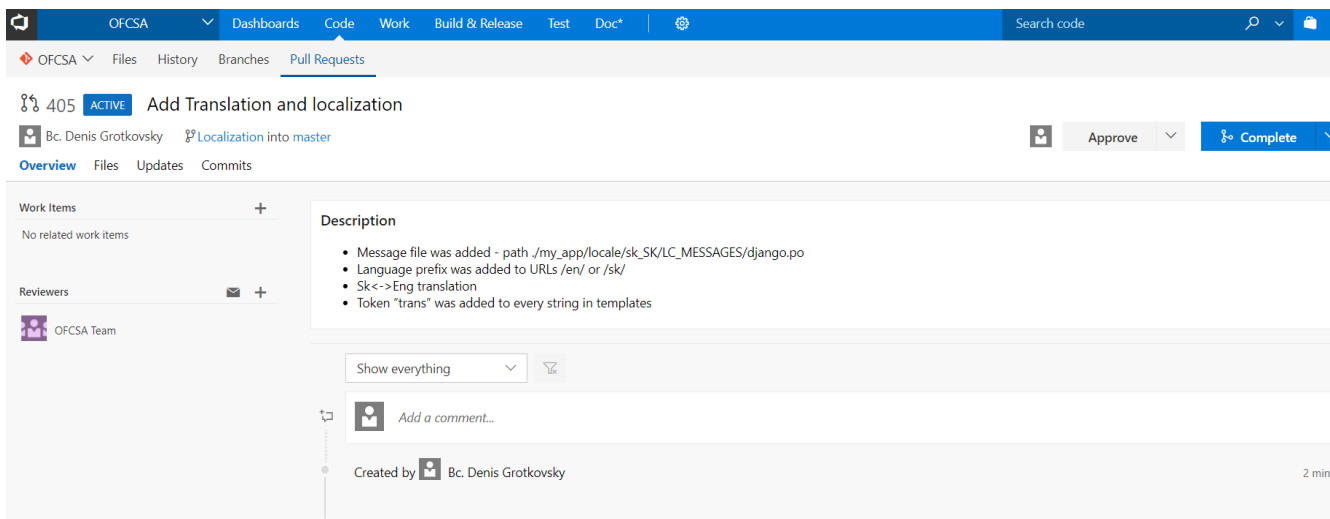


2. Tam nájdí branchu, ktorú si práve pushol a stlač "New pull request".

3. Vyplň "Title" a "Description" podľa metodiky na zdieľaný vývoj pomocou gitu a potvrd' vytvorenie pull-requestu stlačením "Create".

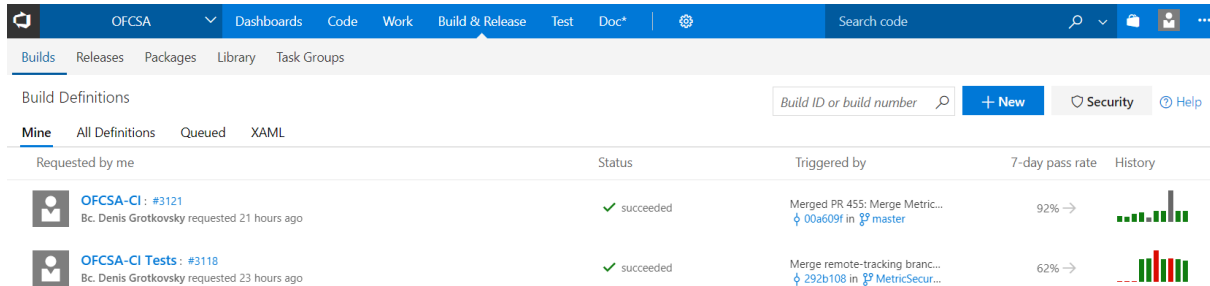


4. Po vytvorení ťa stránka presmeruje na "Overview" tvojho pull-requestu. Na tejto stránke budú môcť tvoji recenzenti zrobiť náhľad na tvoj kód a napísať komentáre. Komentáre sa budú môcť písať do "Overview", alebo ku jednotlivým riadkom kódu v možnosti "Files". Na komentáre aktívne odpovedaj.
5. Ak nastane situácia, keď recenzent nájde nejakú chybu alebo nejasnosť v







tvojom kóde a napíše ju do komentára, je potrebné túto chybu opraviť alebo odpísať na tento komentár a stlačiť tlačidlo “Resolve” alebo iné podľa situácie.

6. Tiež je potrebné skontrolovať či ti zbehol build. To nájdeš v “Build & Release” vo svojich buildoch, kde klikneš na build svojho pull-requestu. Build musí byť úspešný aby sa dal pull-request schváliť.



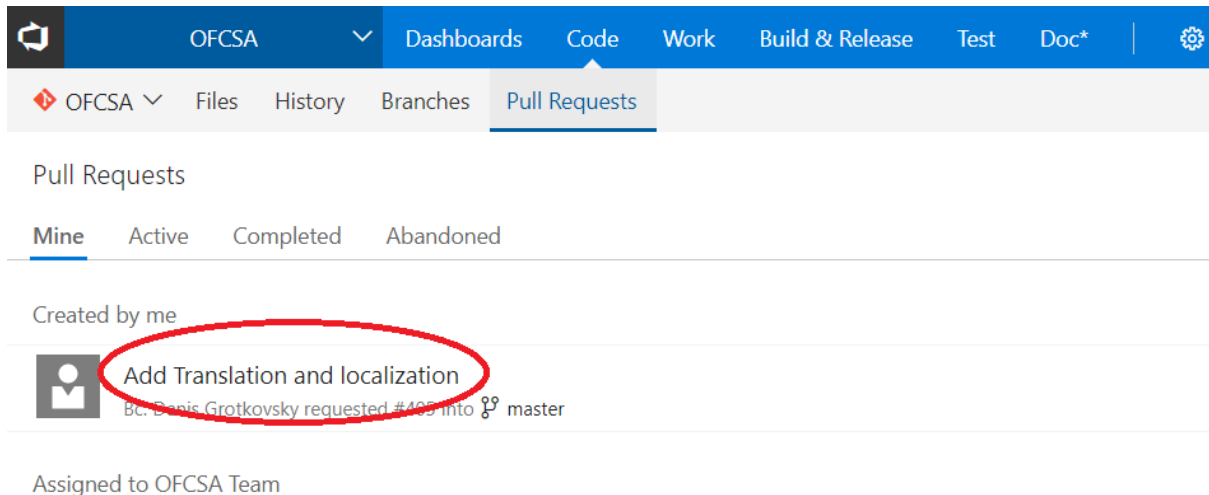
The screenshot shows the Azure DevOps interface for the 'Build & Release' section. The top navigation bar includes 'Dashboards', 'Code', 'Work', 'Build & Release', 'Test', and 'Doc'. Below the navigation, there are tabs for 'Builds', 'Releases', 'Packages', 'Library', and 'Task Groups'. The main content area is titled 'Build Definitions' and includes a search bar for 'Build ID or build number', a '+ New' button, and links for 'Security' and 'Help'. There are three tabs: 'Mine', 'All Definitions', 'Queued', and 'XAML'. The 'Mine' tab is active, showing a table of build definitions. The table has columns for 'Requested by me', 'Status', 'Triggered by', '7-day pass rate', and 'History'. Two build definitions are listed, both with a status of 'succeeded'.

Requested by me	Status	Triggered by	7-day pass rate	History
 OFCSA-CI : #3121 Bc. Denis Grotkovsky requested 21 hours ago	✓ succeeded	Merged PR 455: Merge Metric... 00a609f in master	92% →	
 OFCSA-CI Tests : #3118 Bc. Denis Grotkovsky requested 23 hours ago	✓ succeeded	Merge remote-tracking branc... 292b108 in MetricSecur...	62% →	

7. Po tom ako ti tvoj kód schváli dostatok recenzentov a si s kódom spokojný, môžeš kód mergnúť pomocou tlačidla “Complete”.

Pokyny pre recenzenta:

1. Po tom, ako ti bol pridelený task na code review, chod do Code -> Pull requests a nájdí názov pull requestu, ktorý ti bol pridelený.



Pull Requests

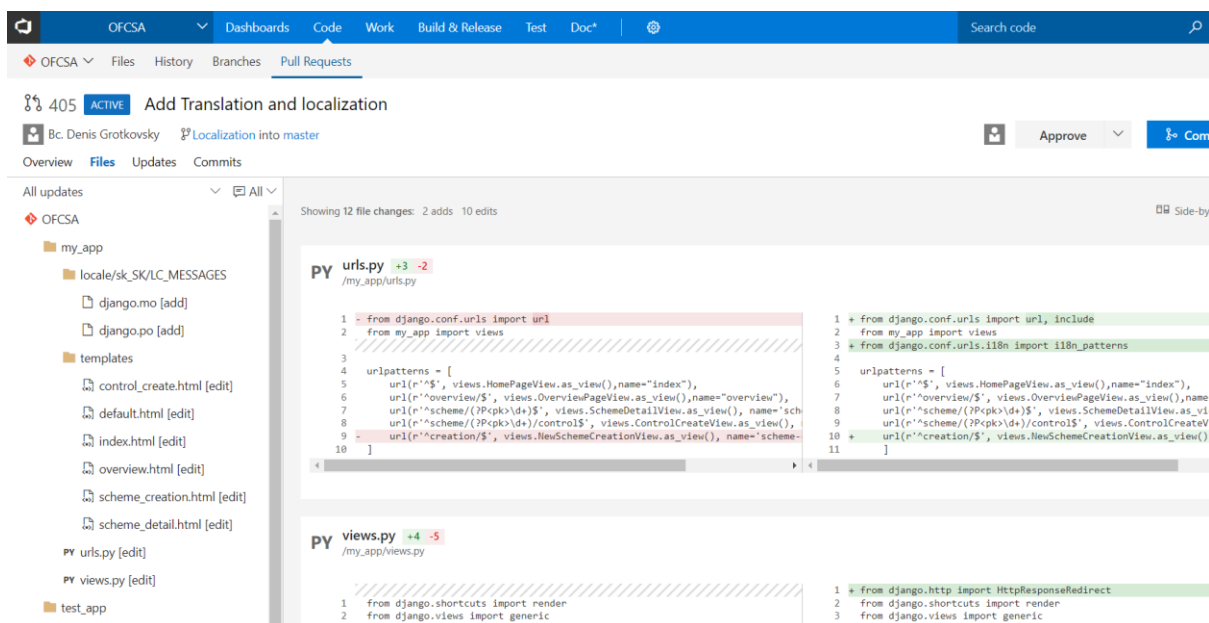
Mine Active Completed Abandoned

Created by me

Add Translation and localization
Bc. Denis Grotkovsky requested #405 into master

Assigned to OFCSA Team

2. V možnosti "Files" môžeš vidieť kód autora.



405 ACTIVE Add Translation and localization

Bc. Denis Grotkovsky Localization into master

Overview Files Updates Commits

All updates

OFCSA

my_app

- locale/sk_SK/LC_MESSAGES
 - django.mo [add]
 - django.po [add]
- templates
 - control_create.html [edit]
 - default.html [edit]
 - index.html [edit]
 - overview.html [edit]
 - schema_creation.html [edit]
 - schema_detail.html [edit]
- urls.py [edit]
- views.py [edit]
- test_app

Showing 12 file changes: 2 adds 10 edits

```
py urls.py +3 -2
/my_app/urls.py

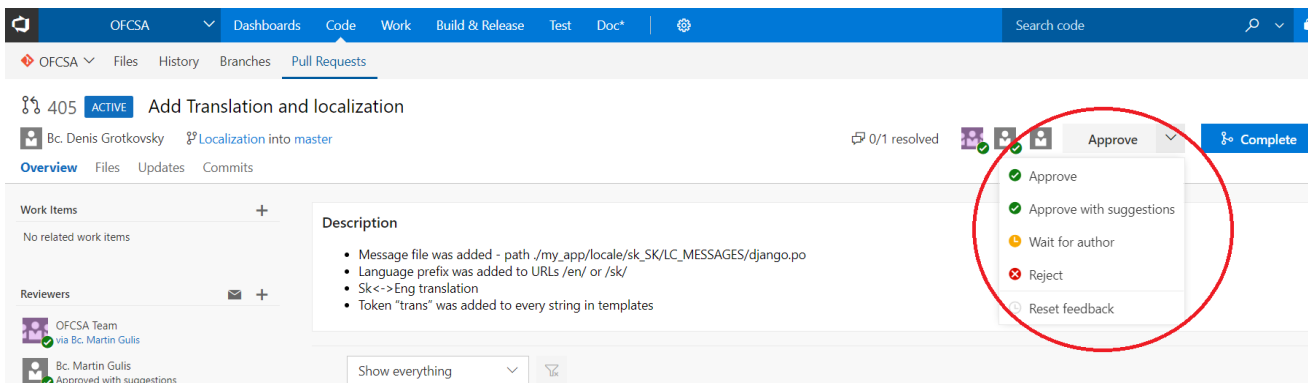
1 - from django.conf.urls import url
2 from my_app import views
3 + from django.conf.urls.i18n import i18n_patterns
4
5 urlpatterns = [
6     url(r'^$', views.HomePageView.as_view(), name="index"),
7     url(r'^overview/$', views.OverviewPageView.as_view(), name="overview"),
8     url(r'^scheme/(?P<pk>[d+])$', views.SchemeDetailView.as_view(), name="scheme-
9     url(r'^scheme/(?P<pk>[d+])/control$', views.ControlCreateView.as_view(),
10 ]
11 ]

py views.py +4 -5
/my_app/views.py

1 + from django.http import HttpResponseRedirect
2 from django.shortcuts import render
3 from django.views import generic
4 from django.views.generic import
```

3. Po tom ako review-neš kód (pozri čo je potrebné kontrolovať), môžeš ho:

- a. **Approve** - schváliť pull-request môžeš len vtedy ak si prešiel všetky body v časti “Čo je potrebné kontrolovať” a všetko je v poriadku.
 - b. **Reject** - zamietni pull-request len vtedy ak obsahuje veľké chyby (zlá funkcionálnosť, zlé testy a pod), ktoré sa nedajú ľahko opraviť (dopísať chybný docstring a pod)
4. Na malé chyby sa snaž upozorniť a napísať ich do komentárov alebo ku riadku kódu. Ak sa ti čokoľvek nezdá, tiež to napíš do komentárov alebo ku



riadku kódu.

Čo je potrebné kontrolovať:

1. Statická analýza -
 - a. Vymazal autor niečo čo nemal?
 - b. Je kód v súlade s metodikou na písanie kódu?
 - c. Sú docstringy všade, kde majú byť?
2. Dynamická analýza -
 - a. Kód si stiahneme a spustíme
 - b. Je jeho funkcionálnosť správna?
 - c. Snažíme sa jeho funkcionálnosť otestovať používateľskými úlohami.
3. Testy
 - a. Zbehli všetky testy? (pozri bod 6 pri pokynoch pre autora)
 - b. Testuje sa to čo sa má testovať?

Ako písať komentáre a ako sa správať pri code review nájdeš tu:

<https://github.com/thoughtbot/guides/tree/master/code-review>

Doplňkové info:

<https://help.github.com/articles/reviewing-changes-in-pull-requests/>

<https://help.github.com/articles/about-pull-requests/>