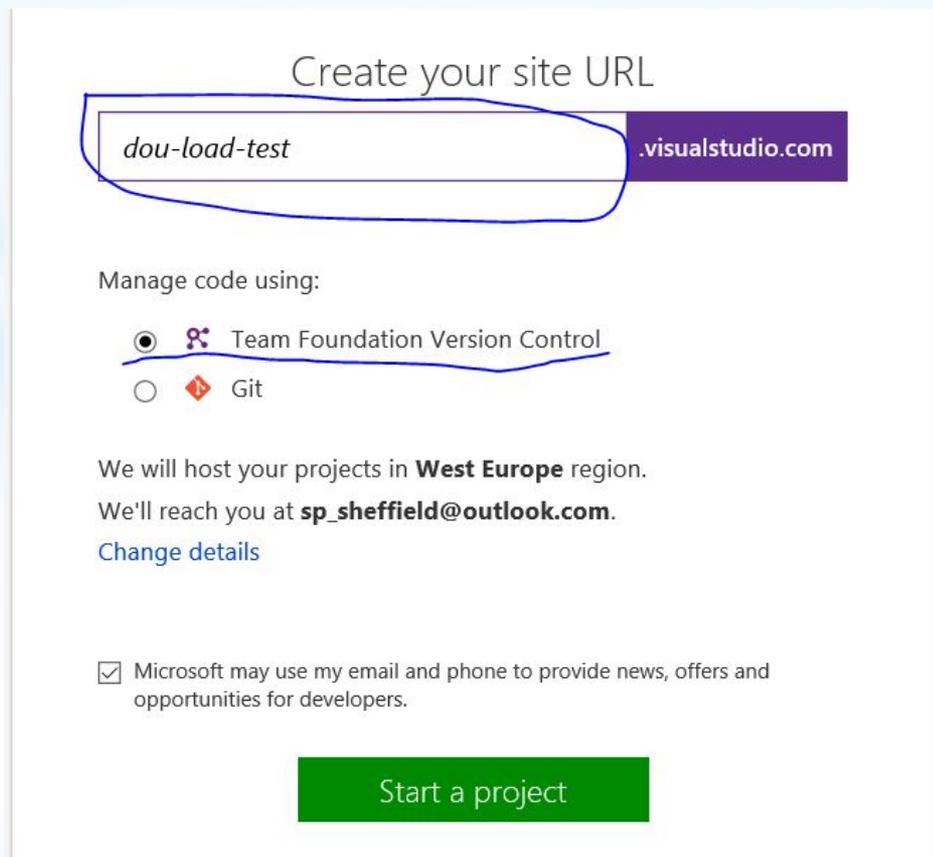


Load testing with Visual Studio Enterprise 2015

1. скачать и установить себе Visual Studio Enterprise 2015
<https://cloud.mail.ru/public/LrZG/9vb7QoNjr>
2. Создать аккаунт Visual Studio Team Services
<https://www.visualstudio.com/ru/team-services/>

После того, как вы создали бесплатный аккаунт в Visual Studio Team Services, вам необходимо создать новый проект, указав название и систему контроля версий, которую вы будете использовать (для этой демонстрации не принципиально, какую выбирать):



Create your site URL

dou-load-test .visualstudio.com

Manage code using:

 Team Foundation Version Control

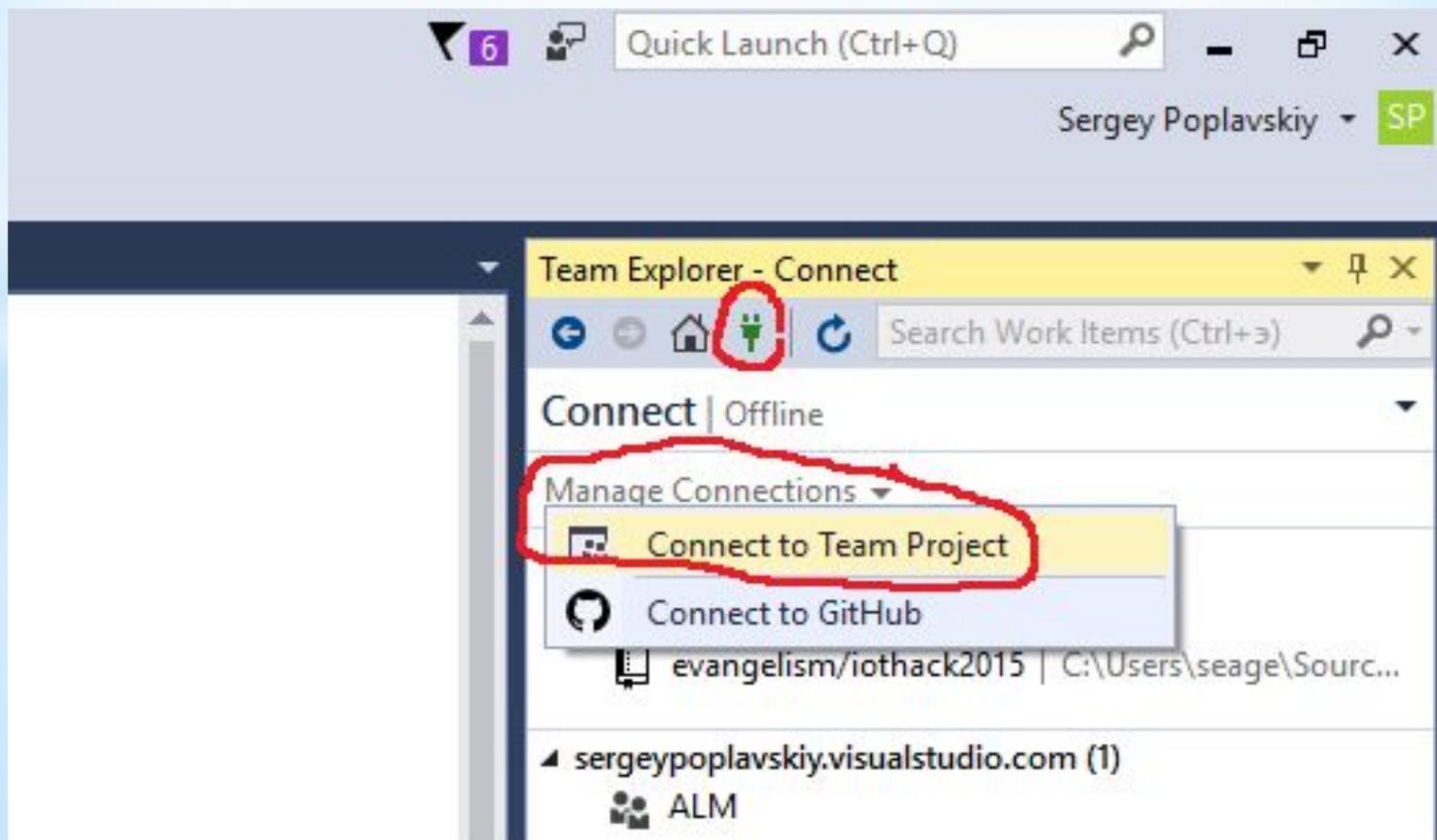
 Git

We will host your projects in **West Europe** region.
We'll reach you at **sp_sheffield@outlook.com**.
[Change details](#)

Microsoft may use my email and phone to provide news, offers and opportunities for developers.

Start a project

После этого нам необходимо привязать созданный аккаунт к Visual Studio. Для этого в Team Explorer в Visual Studio нам нужно ввести параметры соединения к нашему проекту:



Connect to Team Foundation Server



Select a Team Foundation Server:

[Empty dropdown menu]

Servers...

Team Project Collections:

Select a Team Foundation Server to view Team

Team Projects:

[Empty dropdown menu]

Add/Remove Team Foundation Server



Team Foundation Server list:

Name	URL
------	-----

Add...

Remove

Close

Add Team Foundation Server



Name or URL of Team Foundation Server:

https://dou-load-test.visualstudio.com

Connection Details

Path: tfs

Port number: 8080

Protocol: HTTP HTTPS

Preview: https://dou-load-test.visualstudio.com/

OK

Cancel

Также необходимо будет добавить нового пользователя в список аккаунтов Visual Studio и ввести пароль:

Microsoft account

Sign in

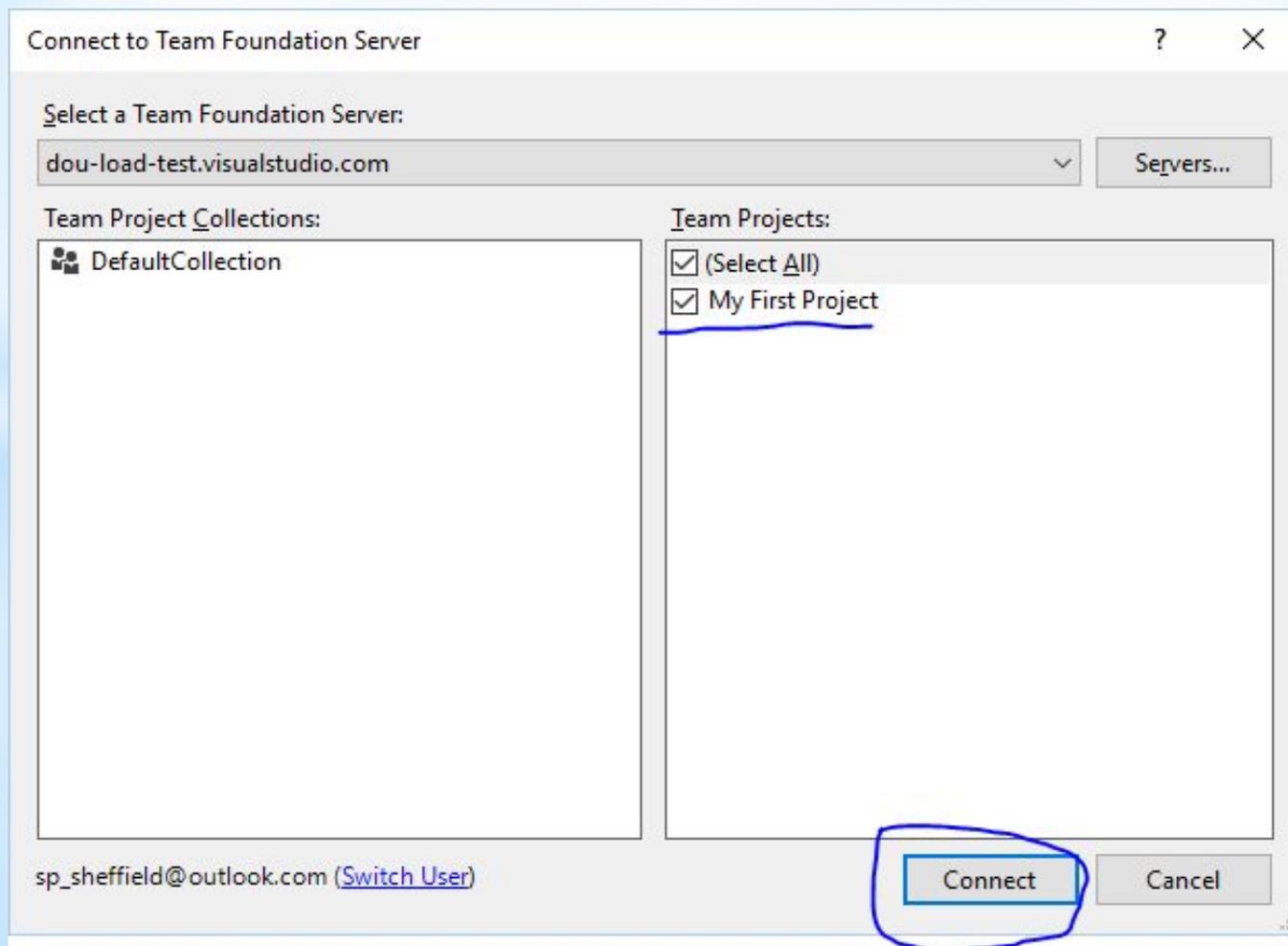
Microsoft account

Password

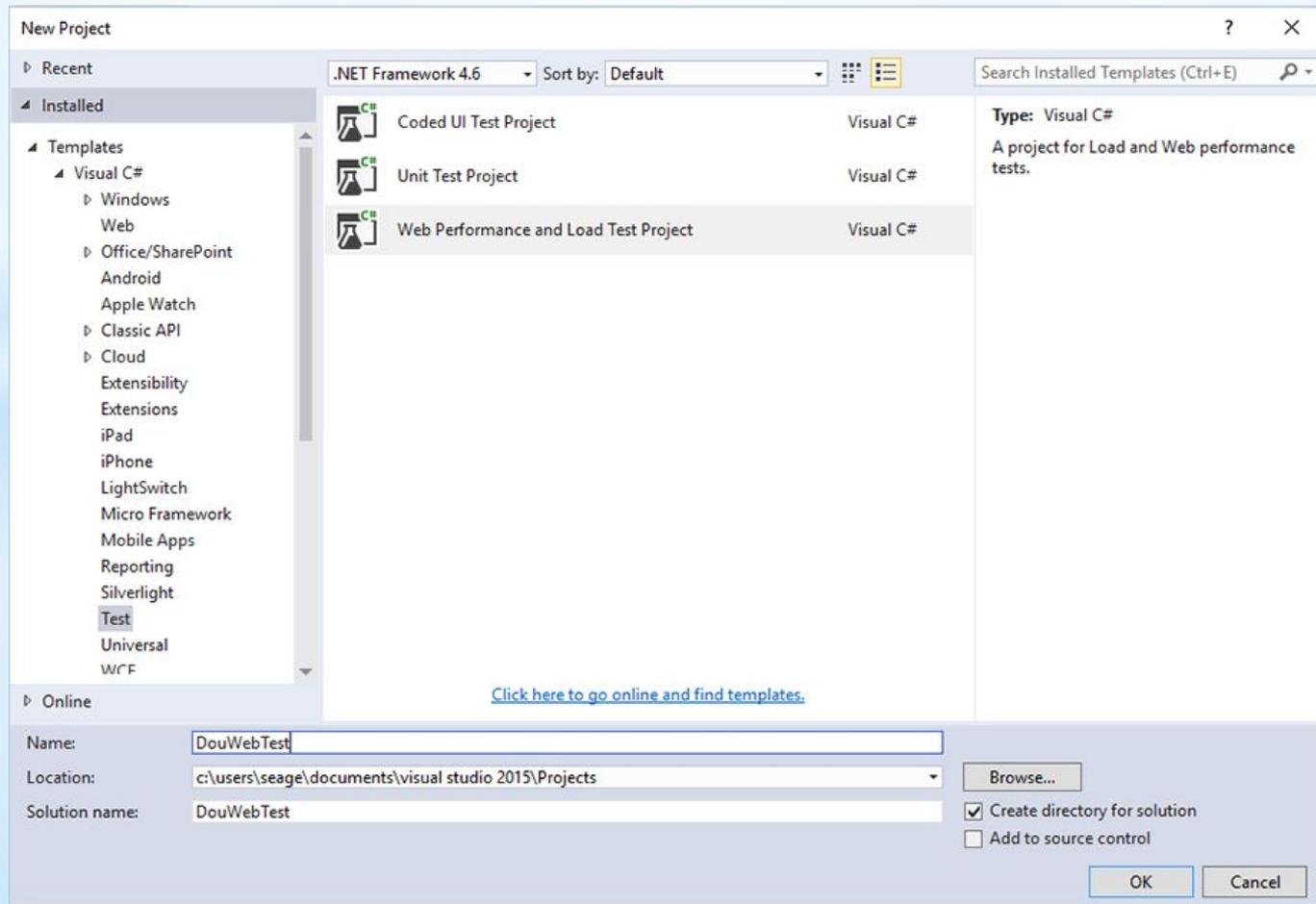
[Sign in](#)

Don't have a Microsoft account? [Sign up now](#)

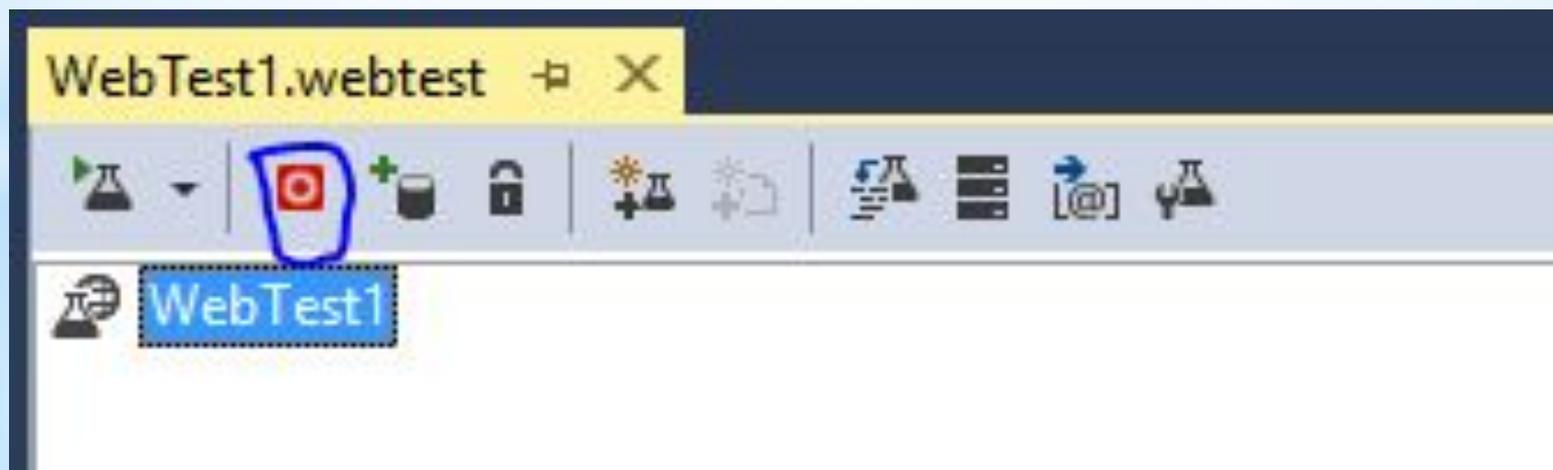
После соединения в окне «Connect to Team Foundation Server» отобразится список проектов. У нас он один, его и выбираем:



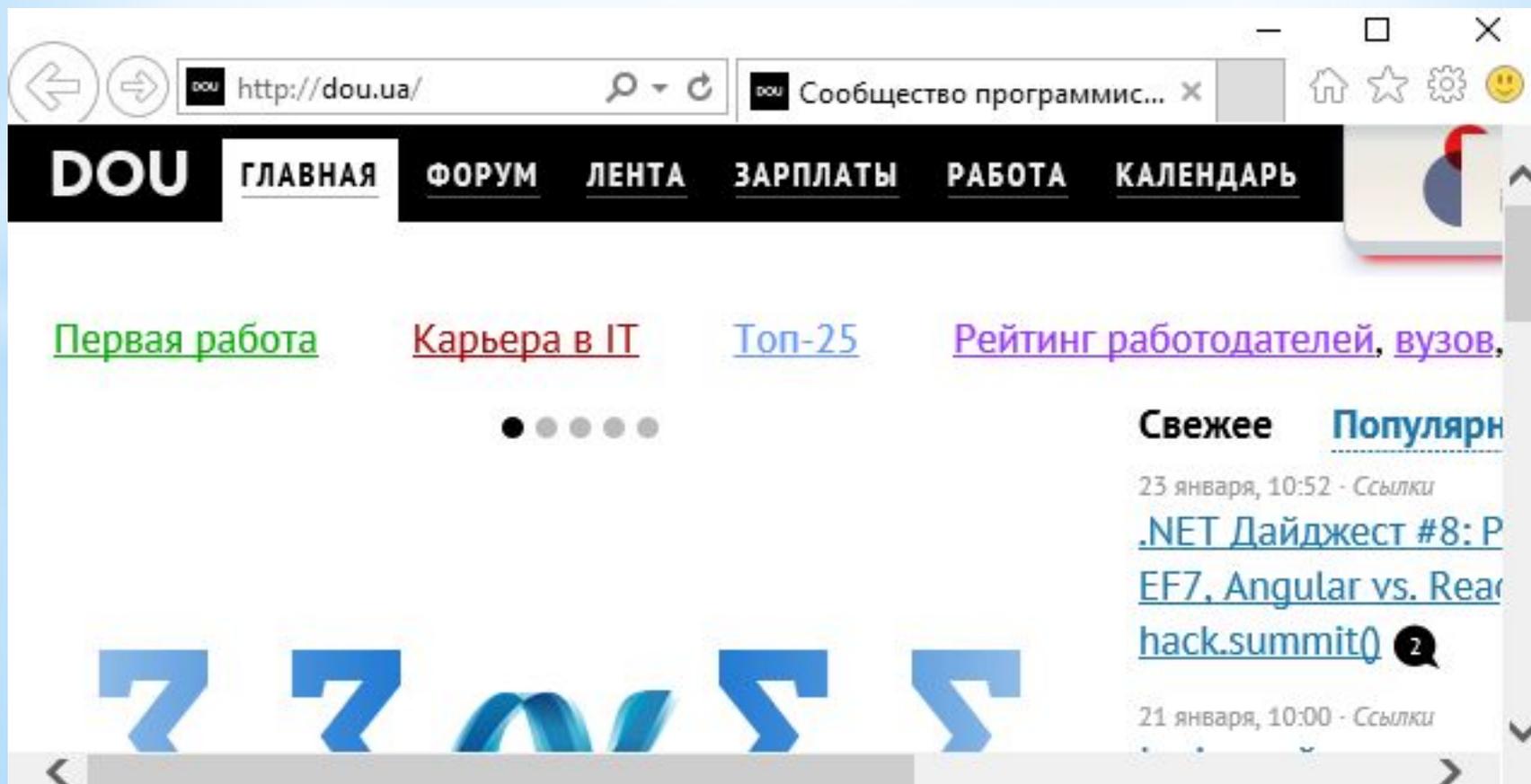
Далее создаем новый проект в Visual Studio. В разделе «Test» выбираем «Web performance and Load Test Project», задаем новое имя проекту:



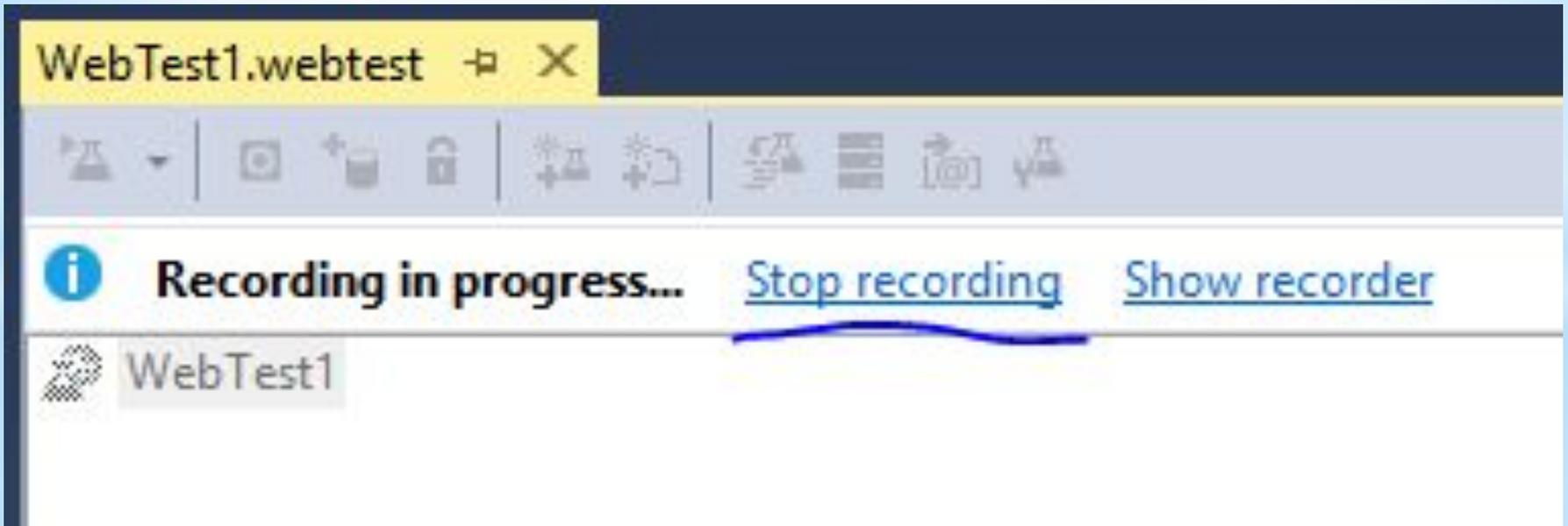
После нажатия на кнопку «ОК» у нас появится новый проект веб теста. Собственно, сейчас наша задача записать этот веб тест. Для этого в нашем веб тесте жмем кнопку «Add recording»:



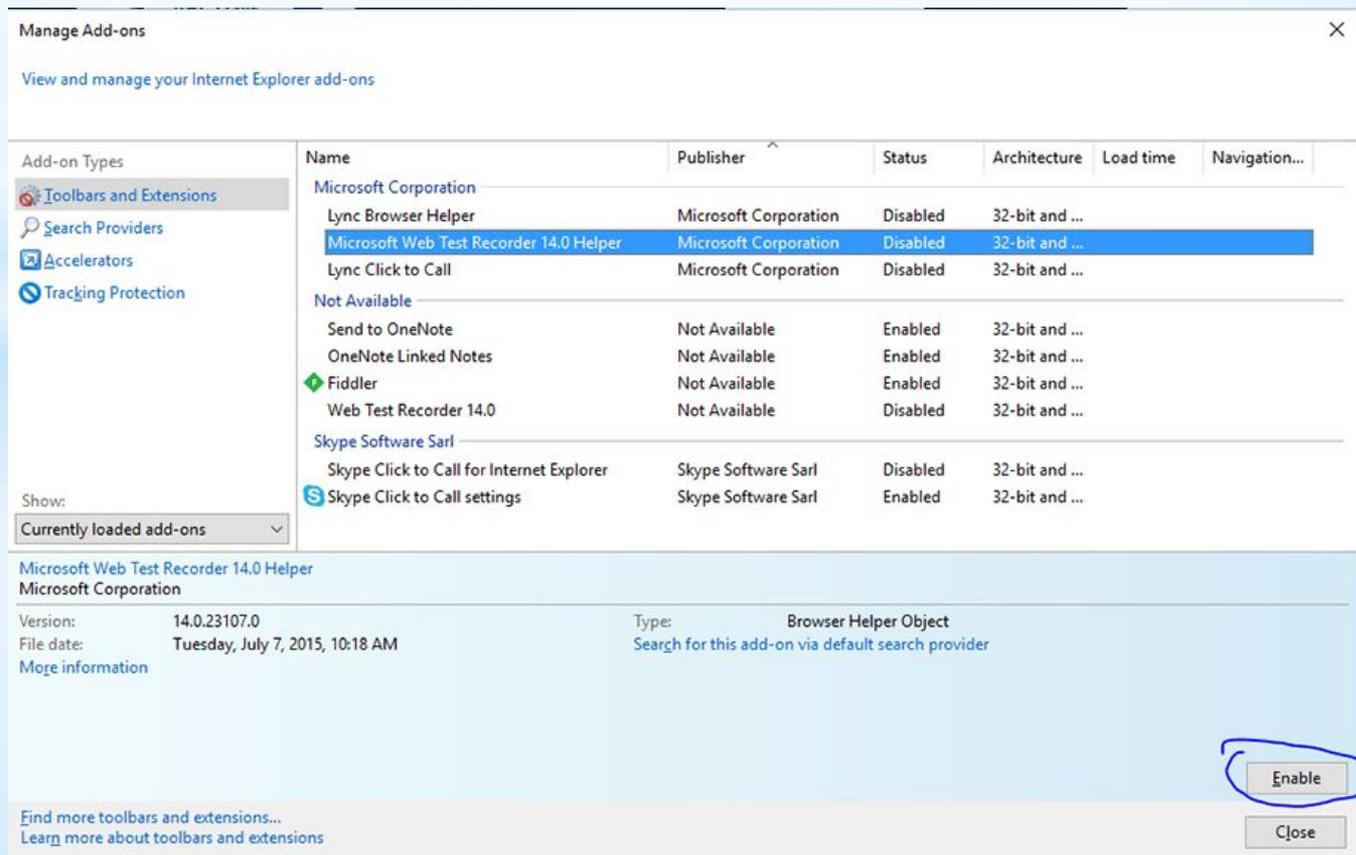
Откроется веб браузер. Все, что вы будете в нем делать, будет записано. Я просто зашел на главную страничку dou.ua, собственно ее и будем тестировать



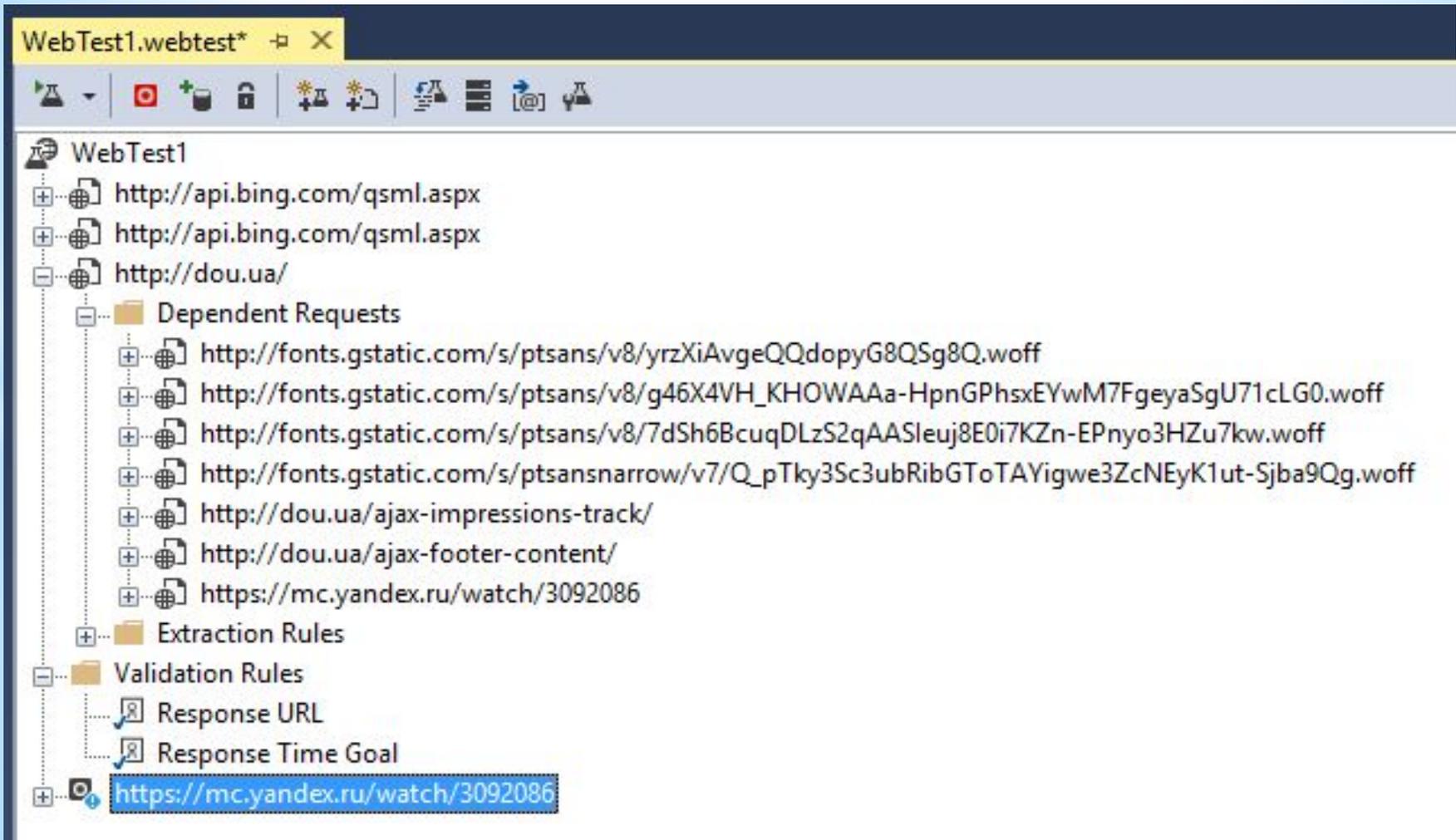
Возвращаемся в Visual Studio и жмем Stop recording:



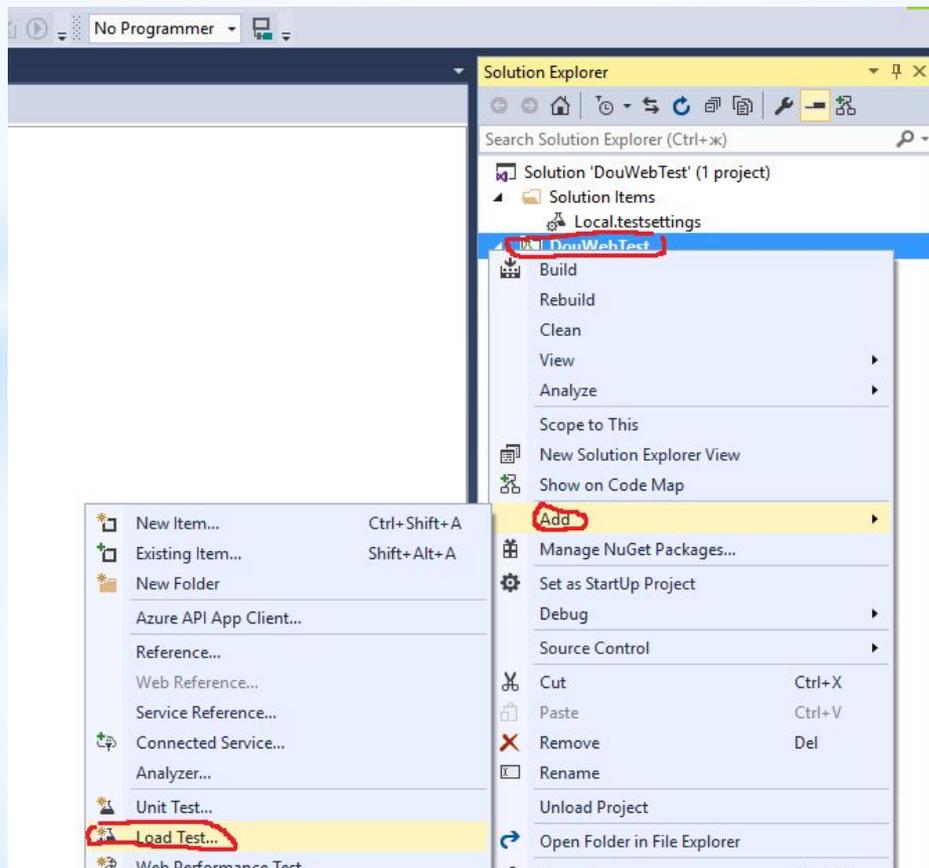
Если после нажатия «Stop recording» ничего не поменялось (в веб тесте не появился записанный url с набором атрибутов), нужно включить аддон в веб браузере «Microsoft Web Test Recorder Helper»:



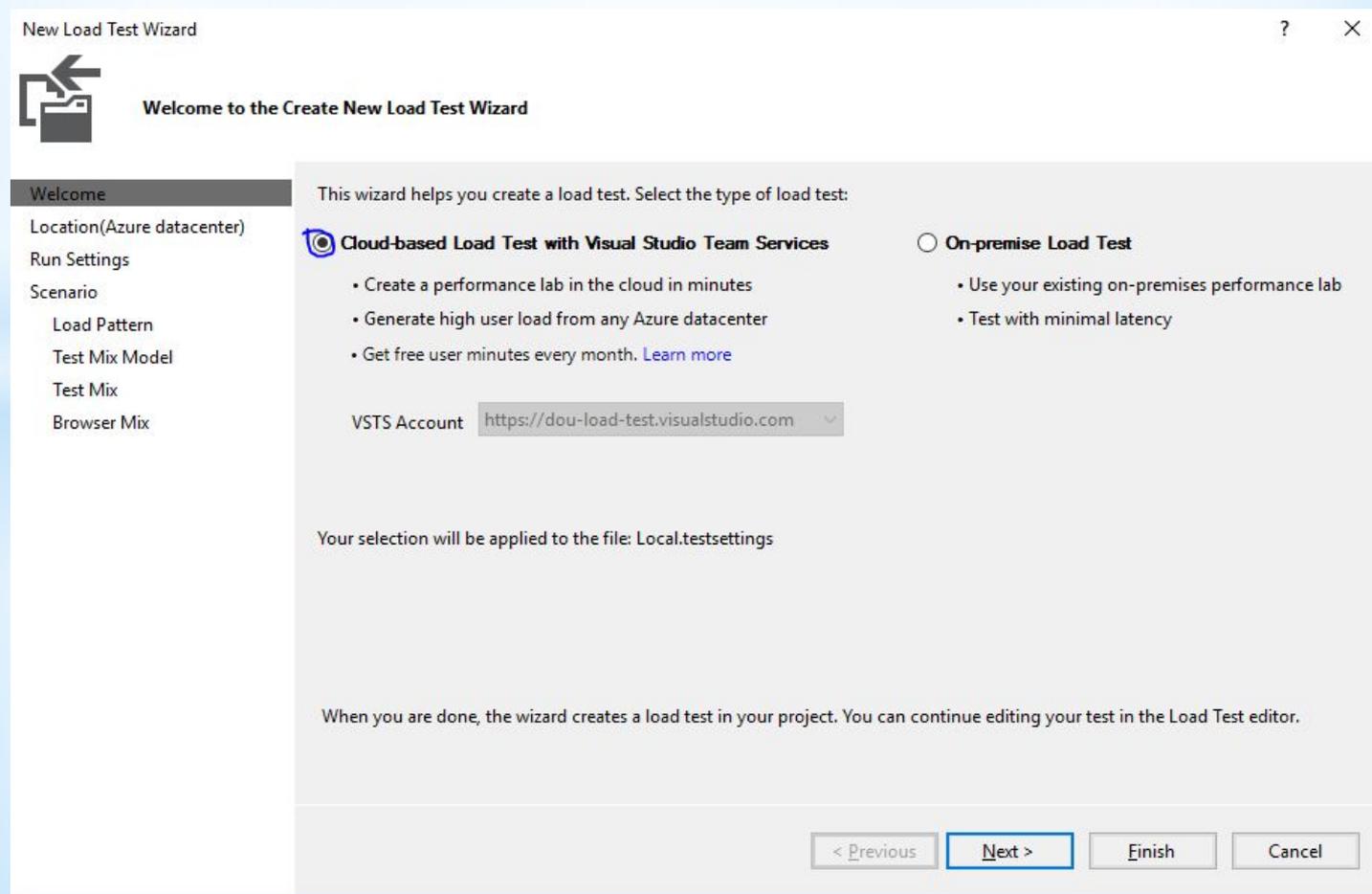
В итоге картина должна быть примерно такая:



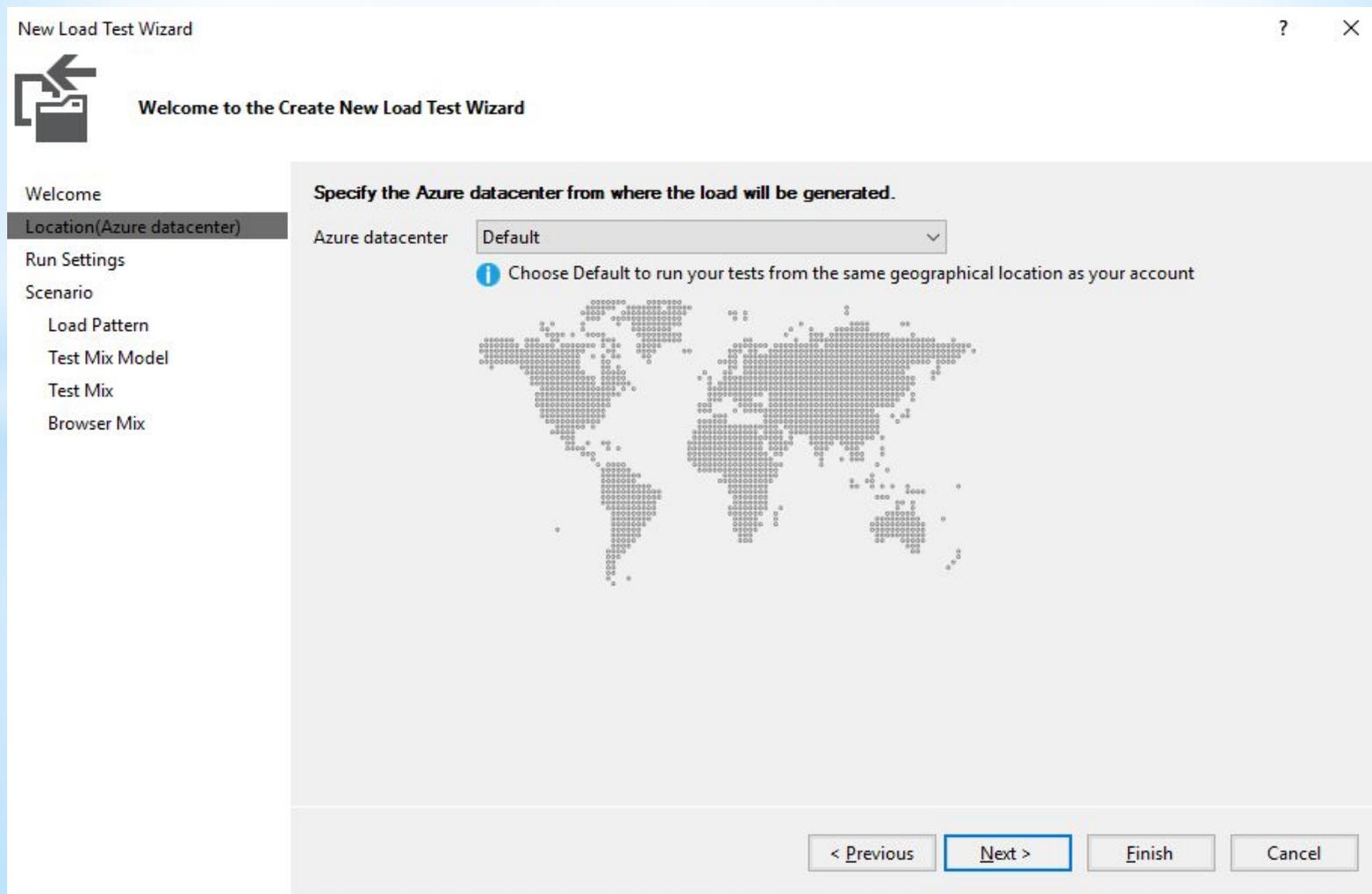
Все, веб тест записан, теперь его нужно запустить из под нужного количества виртуальных пользователей. Для этого в наш проект Visual Studio добавляем Load Test. Клик по проекту нашего веб теста в *Solution Explorer* -> *Add* -> *Load Test*:



У нас откроется wizard, который нужно пройти. На первом шаге выбираем «Cloud-based Load Test with Visual Studio Team Services»:



Далее выбираем локацию дата центра, откуда будет генерироваться нагрузка:



Далее устанавливаем длительность теста:

New Load Test Wizard ? X

 **Review and edit run settings for a load test**

Welcome
Location(Azure datacenter)
Run Settings
Scenario
Load Pattern
Test Mix Model
Test Mix
Browser Mix

⚠ Test iterations are not supported on Cloud-based Load Tests.

Specify the length of the load test by:

Load test duration

Warm-up duration (hh mm ss):

Run duration (hh mm ss):

Test iterations

Test iterations:

Details

Sampling rate: seconds

Description:

Save Log on Test Failure:

Validation level:

После доходим до «Load Pattern». Выбираем «Step Load». Смысл тут в том, что есть стабильная нагрузка. Это значит, что нагрузка будет генерировать от заданного количества пользователей на протяжении теста.

Мы же выбираем более интересный сценарий, в рамках которого количество одновременных пользователей будет постепенно возрастать по заданному шаблону:



Edit load pattern settings for a load test scenario

Welcome

Location(Azure datacenter)

Run Settings

Scenario

Load Pattern

Test Mix Model

Test Mix

Browser Mix

Select a load pattern for your simulated load:

Constant Load:

User Count: users

Step load:

Start user count: users

Step duration: seconds

Step user count: users/step

Maximum user count: users

< Previous

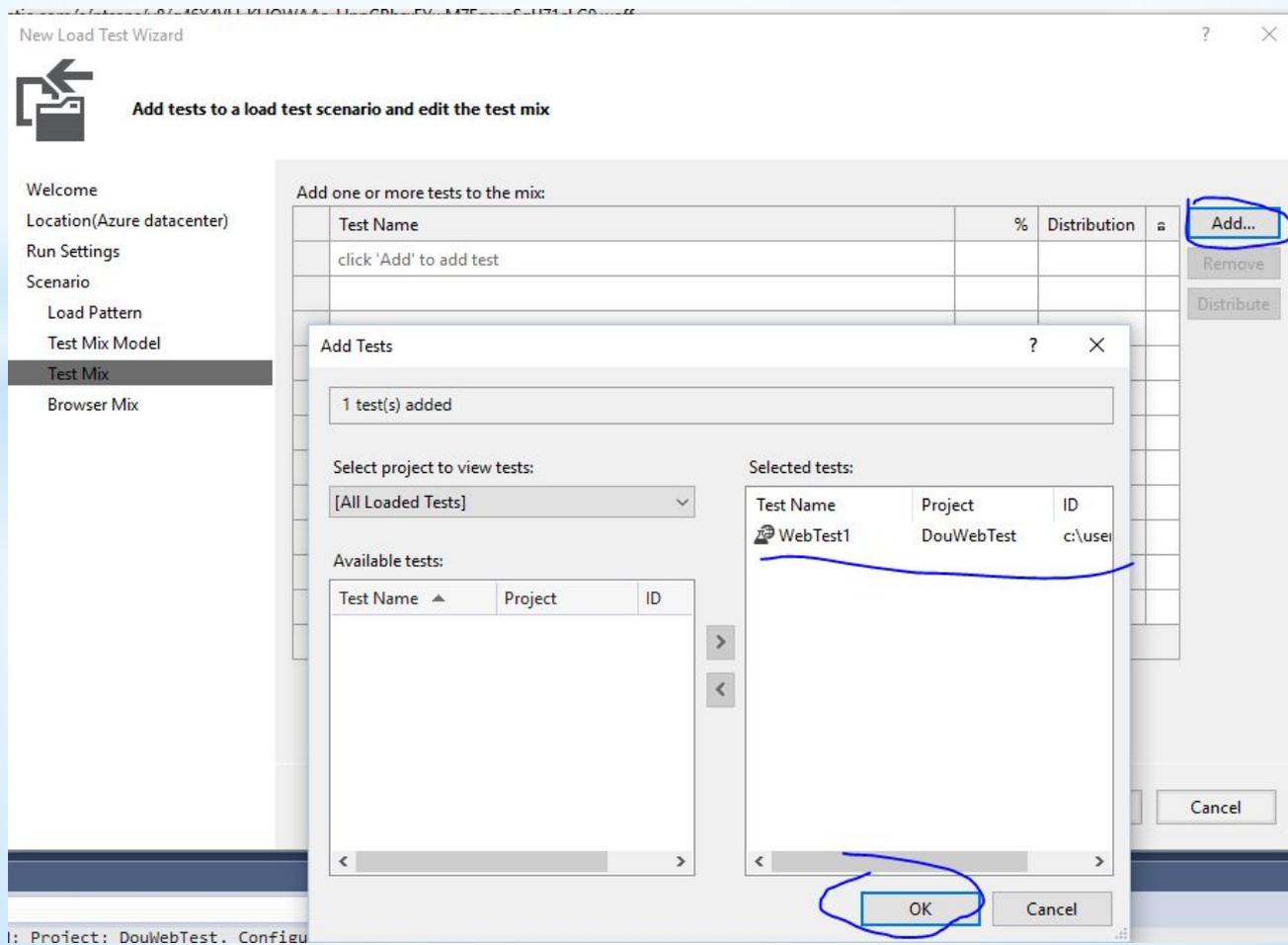
Next >

Finish

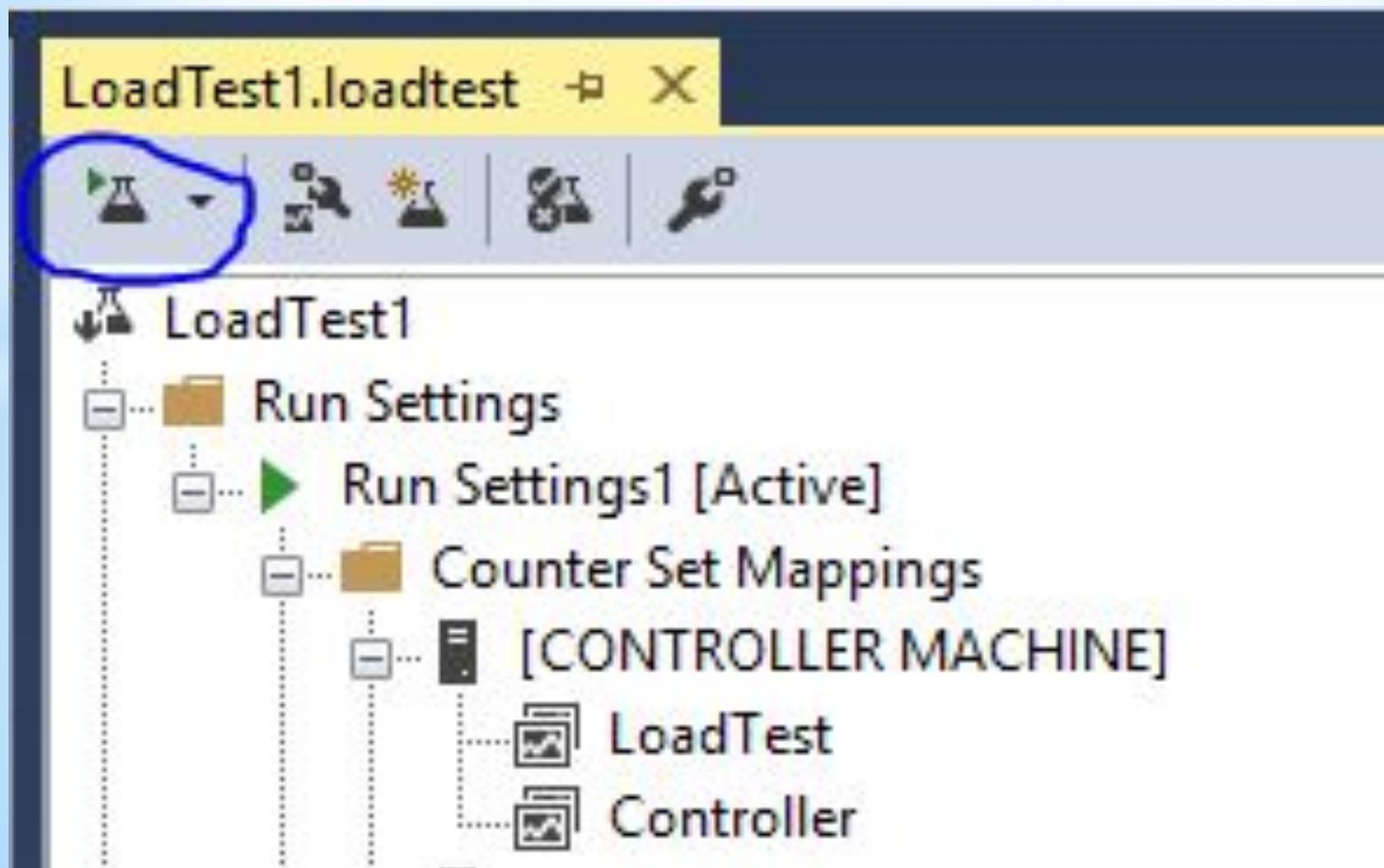
Cancel

Как вы догадались, мы нагрузим DOU тысячей одновременных пользователей.

Далее в Test Mix добавляем наш веб тест:



После этого можем сконфигурировать микс браузеров, после чего жмем Finish. Запускаем нагрузочный тест:



Пройдет какое-то время, и мы начнем видеть графики зависимости времени отклика от количества одновременных пользователей:



По окончании теста можно загрузить в Visual Studio детальный отчет и посмотреть, что происходит. В данном конкретном случае увидел, что почти сразу начали валиться ошибки. Вот статистика по всем ошибкам теста:

Errors			
Type	Subtype	Count	Last Message
Total		4,005	
Extraction Rule Error	ExtractHiddenFields	2,000	No hidden fields were found in the response.
Http Error	503 - ServiceUnavailable	2,000	503 - ServiceUnavailable
Exception	LoadTestErrorLimitExceededExce...	4	More than 1000 errors of type 'ExtractionRuleError' and sub t
Exception	WebException	1	The request was aborted: The request was canceled.

Прелесть в том, что мы можем проанализировать каждую запись отдельно:

The screenshot shows a web browser window with a failed request. The browser title is "Failed Internet Explorer 9.0 LAN". The address bar shows "http://dou.ua/". The status bar indicates "503 Service Temporarily Unavailable". The request details are as follows:

Request	Status	Total Time	Request Time	Request Bytes	Response Bytes
http://dou.ua/	503 Service Temporarily Un	-	0.408 sec	0	1,417

Below the table, there are tabs for "Web Browser", "Request", "Response", "Context", and "Details". The "Request" tab is selected, showing the error message "Ошибка 503". Below the error message, there are links for "Свежее на форуме", "Ленту", and "Вакансии". At the bottom, there is a copyright notice: "© 2005—2016 ДООУ support@dou.ua".

Подозреваю, что это сработала защита веб сервера, так как с одного IP повалило куча запросов. При этом во время проведения теста я периодически рефрешил страничку у себя в браузере, все было ОК (помним, что нагрузка генерилась из облака).

Детальные отчеты вы можете выгрузить в Excel и проанализировать там. А если у вас есть предыдущие измерения, то Visual Studio умеет анализировать, насколько изменилась производительность с прошлого раза.

