**Дополнительный материал**

**РАЗМЕЩЕНИЕ ВЕБ-САЙТА НА ХОСТИНГЕ**

**Системы контроля версий**

Разработчики современного программного обеспечения, как правило, используют системы контроля версий, которые упрощают совместную работу над проектом, позволяют отслеживать историю всех изменений в проекте и при необходимости возвращаться к предыдущим шагам.

Кроме того, системы контроля версий позволяют создавать несколько альтернативных версий программного продукта и параллельно работать с ними.

Самая популярная система контроля версий – **Git**, а самый известный хостинг проектов с поддержкой Git – **GitHub**.

Использование этого сервиса полезно и для индивидуальных разработчиков, поскольку позволяет сохранять историю всех изменений собственных проектов, а также обеспечивает возможность публикации проекта.

При разработке проекта на Django использовать GitHub удобно ещё и потому, что для проектов, выложенных на GitHub, существуют простые способы размещения проектов на хостинге.

**Создание аккаунта GitHub**

Для создания аккаунта следует перейти по ссылке:

[https://github.com](https://github.com/)

и нажать в правом верхнем углу кнопку «**Sign Up**». В результате откроется страница, на которой следует заполнить следующие поля:

* **Username** (имя пользователя). Имя пользователя должно состоять из английских букв и может содержать цифры.
* **Email address** (адрес электронной почты).
* **Password** (пароль). Пароль должен быть длиной не менее 8 символов, состоять из английских букв в нижнем регистре и обязательно содержать цифру или цифры.

Под надписью **Verify your account** потребуется подтвердить, что вы человек, выполнив не сложное задание.

Затем ниже следует нажать кнопку «**Create an account**». В результате откроется страница, на которой можно выбрать тариф. По умолчанию выбирается бесплатный тариф (**Free**), позволяющий создавать неограниченное число публичных и частных репозиториев проектов. Необходимо нажать кнопку «**Choose Free**». Откроется страница с анкетой пользователя. Ее заполнение можно пропустить, нажав внизу ссылку «**skip this step**».

Затем следует зайти в почтовый ящик, указанный при регистрации, и в полученном от GtiHub письме нажать на ссылку для подтверждения e-mail адреса (**Verify email address)**.

После этого можно входить в свой аккаунт на странице:

<https://github.com/login>

**Добавление компонента для работы с GitHub в Visual Studio 2017**

Чтобы размещать проекты на GitHub, используя Visual Studio 2017, потребуется установить дополнительный компонент. Для этого необходимо скачать файл по ссылке:

<https://github.com/github/VisualStudio/releases/download/v2.6.0.6187/GitHub.VisualStudio.vsix>

Запустить его на выполнение кликом по файлу в **Проводнике** и дождаться окончания установки.

После этого в Visual Studio 2017 появляется возможность подключиться к GitHub, вызвав через меню «**Вид**» панель «**Team Explorer**». Вверху панели при клике по ссылке «**Управление подключениями**» будет раскрываться панель со ссылкой «**Connect to GitHub**» («Подключиться к GitHub»). При выборе этой ссылки откроется панель, в которой надо ввести **логин** и **пароль** и нажать на кнопку «**Sign In**».

**Создание репозитория на GitHub**

После подключения к GitHub вверху панели «**Team Explorer**» добавится секция GitHub. В меню этой секции для создания репозитория на GitHub следует нажать «**Create**». Откроется панель создания репозитория.

В ней следует заполнить следующие поля:

* **Name** – имя репозитория. В это поле скопируйте имя публикуемого проекта Django. Это имя обычно имеет папка созданного для проекта решения Visual Studio 2017 и вложенная в нее папка проекта Django.
* **Local path** – локальный путь к репозиторию. Используя кнопку «**Browse**» справа следует выбрать произвольную папку для создания в ней папки пустого репозитория. Затем эта папка будет объединена с папкой вашего проекта.
* **Git ignore** – выбор списка игнорируемых при публикации типов файлов и папок. Эта опция исключает из публикации временные и служебные файлы. В этом поле обязательно следует **выбрать** из раскрывающегося списка **Python**.
* Значения остальных полей (**Description, License, Private Repository**) изменять не требуется.

После заполнения полей следует нажать кнопку «**Create**».

Затем в **Проводнике** следует найти созданную на предыдущем шаге папку пустого репозитория и, не входя в неё, командой контекстного меню, вызываемого по правой кнопке мыши, скопировать папку в буфер обмена.

После этого войти в папку решения Visual Studio 2017. При этом одноименная папка проекта Django будет отображаться в текущей папке. Правой кнопкой мыши выбрать команду «**Вставить**». После этого папки репозитория и проекта Django должны будут объединиться. Проверить это можно, зайдя в папку проекта Django и увидев там файл **.gitignore**.

После этого шага надо вернуться в **Team Explorer** и в секции «**Локальные репозитории Git**» нажать ссылку «**Добавить**». Появится поле ввода пути к репозиторию. Справа от него по кнопке «**…**» можно вызвать панель выбора папки. В ней следует выбрать папку Django проекта, вложенную в папку решения Visual Studio 2017. Внутри выбираемой папки должны, в частности, находиться папки **app** и **env**. Когда репозиторий добавится в список, по его имени следует кликнуть двойным кликом. В результате откроется панель управления репозиторием. Следует зайти в раздел «**Изменения**». Там выполняется подготовка к публикации изменений в проекте. Вверху обязательно надо ввести какое-то название для обновления, например **First commit**. После ввода имени станет активна кнопка «**Зафиксировать всё**», которую следует нажать. После этого надо перейти в раздел «**Синхронизация**». Там внизу будет секция «**Исходящие фиксации**», в которой надо нажать ссылку «**Отправить**». Выполнение этой операции занимает некоторое время.

После отправки данных можно зайти через веб-браузер в свой аккаунт на GitHub и убедиться, что репозиторий существует и в нём находятся файлы проекта.

После выполнения каких-либо изменений в проекте, аналогичным образом, используя разделы «**Изменения**» и «**Синхронизация**», можно обновлять репозиторий на GitHub.

**Создание аккаунта для веб-проекта Django на хостинге**

Для размещения проекта на хостинге рекомендуется использовать бесплатный хостинг:

<http://www.pythonanywhere.com/>

Страница регистрации пользователя:

<https://www.pythonanywhere.com/registration/register/beginner/>

После заполнения полей и отправки формы, следует подтвердить e-mail, кликнув по ссылке в полученном письме. Необходимо учитывать, что имя, указанное при регистрации, будет именем вашего сайта, поэтому и выбирать имя необходимо с учетом этого обстоятельства.

Страница входа в панель управления хостингом:

<https://www.pythonanywhere.com/login/>

После успешной регистрации и входа необходимо **создать токен доступа**. Для этого в правом верхнем углу надо нажать ссылку **Account**, на открывшейся странице выбрать вкладку **API Token** и там нажать кнопку «**Create a new API token**».

**Публикация сайта на хостинге**

Вернитесь на главную страницу **PythonAnywhere**, кликнув по логотипу. В левом нижнем углу нажмите кнопку **$Bash**. Откроется командное консольное окно.

Далее следует ввести команду, которая устанавливает программу для развертывания проектов Django на хостинге:

*pip3.6 install --user pythonanywhere*

Учтите, что при копировании через буфер эти команды не выполняются, их надо вводить вручную с клавиатуры!

Затем следует запустить утилиту для загрузки на хостинг репозитория с GitHub.

*pa\_autoconfigure\_django.py https://github.com/имя\_пользователя\_GitHub/имя\_репозитория.git*

Например:

*pa\_autoconfigure\_django.py https://github.com/ivanpetrov98/DjangoWebProject1.git*

Затем следует ввести команду создания суперпользователя, который будет администрировать сайт:

*python manage.py createsuperuser*

Потребуется ввести имя, e-mail, пароль, подтверждение пароля - ввести те же данные суперпользователя (администратора), которые вы вводили при создании суперпользователя для своего веб-проекта в среде Visual Studio.

После выполнения изложенных выше шагов ваш сайт будет открываться по адресу:

[http://имя\_сайта.pythonanywhere.com](http://xn--_-8sba3ahp0bk6i.pythonanywhere.com/)

Панель управления хостингом позволяет редактировать файлы проекта на хостинге и добавлять файлы. Эти операции доступны в разделе **Files**. В левой колонке там отображаются папки, в правой – файлы. При клике по файлу он загружается в текстовый редактор. Кнопка «**Upload a file**» позволяет загрузить локальные файлы на сервер.

При разворачивании Django проекта на хостинге в состав его файлов не включается файл базы данных **db.sqlite3**. Если Вы хотите опубликовать сайт, включая данные из локальной базы, можно загрузить локальный файл **db.sqlite3** в папку **имя\_сайта.pythonanywhere.com** на хостинге.

Кроме того, вы можете загружать в папку **/static/app/content** графические и другие файлы, на которые вы ссылаетесь в макетах страниц.

Для того, чтобы заменить ранее опубликованный проект сайта новой версией, можно использовать описанную выше команду **pa\_autoconfigure\_django.py**, добавив в конец строки ключ --**nuke** .

**Задание для самостоятельной работы**

**Цель самостоятельного задания**: Получение навыков размещения разработанного веб-сайта на хостинге.

1. Создать аккаунт GitHub.
2. Добавить компонент для работы с GitHub в Visual Studio 2017.
3. Создать на GitHub репозиторий и закачать файлы созданного веб-проекта.
4. Создать аккаунт для веб-проекта Django на хостинге.
5. Опубликовать разработанный сайт на хостинге.