



GitHub, GitLab, BitBucket

Guides and Tutorials

NUM CodeHub Club - Team NightRunners

Удиртгал

Энэхүү гарын авлага, заавар нь сүүлийн үед тренд болж буй Git төрлийн гурван платформ болох GitHub, GitLab, BitBicket-ийг хэрхэн ашиглах заавар, үндсэн ойлголтууд, функцууд, болон зөвлөмжийг хэрэгцээт screenshot-той цуг оруулсан болно.

Зорилго

1. Git гэж юу вэ?, яагаад ашиглах ёстой вэ?
2. GitHub ашиглах заавар, гарын авлага
3. GitLab ашиглах заавар, гарын авлага
4. Bitbucket ашиглах заавар, гарын авлага



```
<Hello World/>
```

The **Hello World** project is a time-honored tradition in computer programming. It is a simple exercise that gets you started when learning something new. Let's get started with GitHub!

You'll learn how to:

- Create and use a repository
- Start and manage a new branch
- Make changes to a file and push them to GitHub as commits
- Open and merge a pull request

GitHub гэж юу вэ?

GitHub бол version control болон хамтын хамтын ажиллагаанд зориулсан code hosting платформ юм. Энэ нь таныг болон бусдыг төсөл дээр хамтран ажиллах боломжийг олгоно.

Энэхүү заавар гарын авлагаар Github-ийн үндсэн ойлголтууд болох repositories, branches, commits, and pull requests-үүдийг заах болно. Мөн та өөрийн Hello World repository-г үүсгэж хэрэгцээт алхамуудыг хийх болно.

Алхам 1. Repository үүсгэх

Repository нь ихэвчлэн цөөн тооны төслүүд (single project) дээр ажиллаж байх үед ашиглагддаг. Repository нь folders, files, images, videos гэх мэт контентүүдийг агуулж болно. Мөн README файл буюу тухайн төслийн талаар мэдээлэл агуулсан файльтай байдаг. Үүнийг Repository үүсгэх үед хамт үүсгээд өгдөг давуу талтай.

Таны Hello-World Repository шинэ санаа хадгалах эсвэл бусадтай хуваалцах, зөвлөлдөх газар байх болно.

Шинэ Repository үүсгэх нь

1. Дэлгэцийн баруун дээд булан болох аватар зурагны хажууд байрлах нэмэх буюу + товчийг дараад New Repository сонголтыг сонго.
2. Repository-гийн нэрийг hello-world гэж нэрлэ.
3. Товч тодорхойлолт бич.
4. **Initialize this repository with a README.** Энэхүү сонголтыг сонго.

PUBLIC

Owner

Repository name

hubot

 /

hello-world

Great repository names are short and memorable. Need inspiration? How about [petulant-shame](#).

Description (optional)

Just another repository

☒ Public

Anyone can see this repository. You choose who can commit.

☐ Private

You choose who can see and commit to this repository.

☒ Initialize this repository with a README

This will allow you to `git clone` the repository immediately. Skip this step if you have already run `git init` locally.

Add .gitignore: **None**

Add a license: **None**

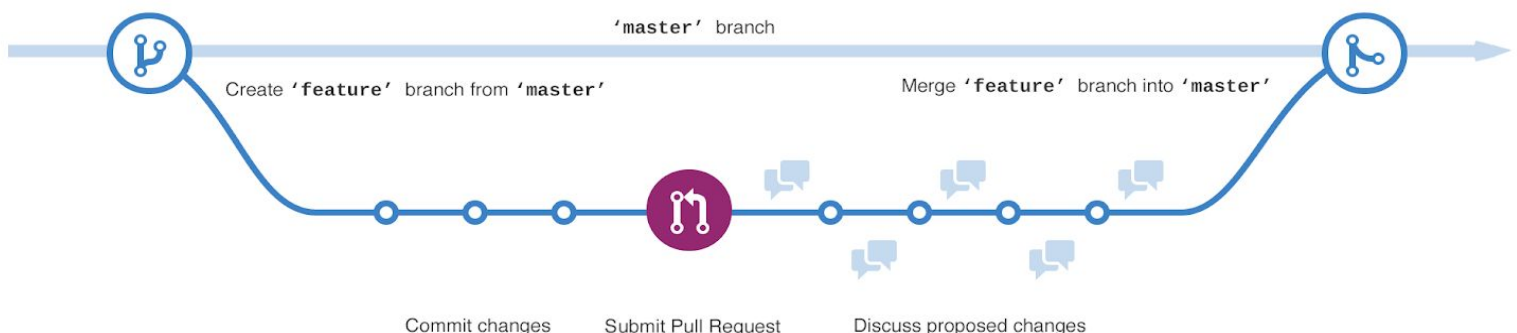
Create repository

Алхам 2. Branch үүсгэх

Branching хийх нь repository дээр нэгэн зэрэг олон version үүсгэх ажиллах боломжийг олгодог. Repository нь цаанаасаа ганцхан бранчтай байдаг буюу ганц мастер бранчтай байдаг. Бранчийг мастер бранчруу commit хийхээс өмнө аливаа нэгэн туршилт болон засваруудыг хийхэд ашигладаг.

Шинээр бранч үүсгэхэд мастер бранчийн хуулбарыг үүсгэдэг. Хэрэв өөр нэгэн хүн таныг бранч дээр тань ажиллаж байх үед мастер бранчид өөрчлөлт хийвэл дараах шинэчлэлийг хийж болно.

Дараах диаграмыг харна уу.



Жишээ нь:

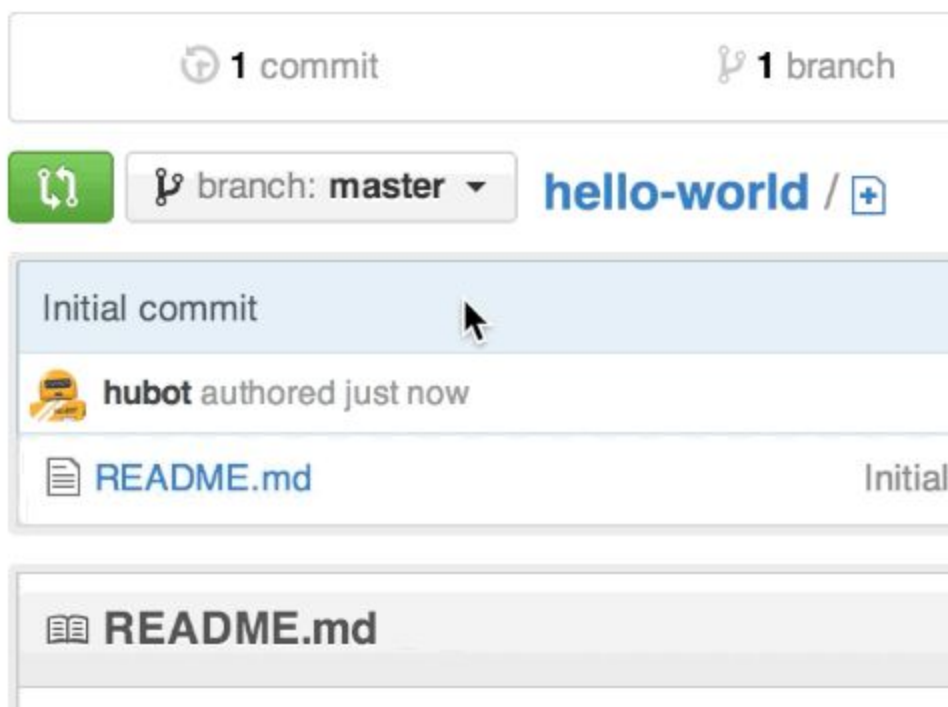
- story.txt
- story-joe-edit.txt
- Story-joe-edit-reviewed.txt

Github-д хөгжүүлэгчид бранчийг bug fix буюу алдаагаа засах, нэмэлт боломжуудыг нэмж өгөх үүднээс мастер бранчаас өөр бранч дээр ажилладаг. Өөрчлөлтүүд бэлэн болоход тэдгээрийг мастер бранч руу нэгтгэн оруулдаг.

Шинэ Branch үүсгэх нь

1. Hello-world repository-руугаа орно уу.
2. **Branch: master** гэсэн dropdown-ийг дарна уу.
3. Бранчийн нэрийг бичнэ үү. **Readme-edits**
4. **Create branch** товч эсвэл enter товчийг дарна уу.

Just another repository — Edit



Тэгэхээр одоо мастер бранч болон readme-edits гэсэн бранчуудтай боллоо. Тэд адилхан харагдаж байгаа хэдий ч тийм биш байдаг.

Алхам 3. Commit үүсгэж оруулах

Readme-edits бранч буюу мастер бранчийнхаа хуулбар бранч дээр засвар хийе.

Commit гэдэг saved changes буюу хадгалагдсан өөрчөлтийг нэрлэдэг. Commit болгон Commit message-тай байна. Энэ нь яагаад тухайн өөрчлөлтийг хийх болсон товч тодорхойлолт болох юм. Commit message нь өөрчлөлтүүдийн түүхийг агуулдаг учир бусад хөгжүүлэгчид ямар өөрчлөлт оруулсаныг болон яагаад гэдгийг ойлгох болно.

Commit үүсгэж оруулах нь

1. Readme.md файл дээр дарна уу.

2. Баруун дээд буланд байгаа харандааны дүрстэй төвчийг дарна уу.
3. Editor дээр өөрийнхөө тухай бичнэ үү.
4. Commit message буюу өөрчөлтөө тодорхойлж бичнэ үү.
5. **Commit changes** товчийг дарна уу.

The screenshot shows the GitHub web interface for the 'hubot / hello-world' repository. The repository has 1 watch, 0 stars, and 0 forks. The 'Code' tab is selected, and the 'README.md' file is open in the editor. The editor shows the following content:

```
1 # hello-world
2
3 Hi Humans!
4
5 Hubot here, I like Node.js and Coffeescript (that's what I'm made of!).
6 I've had tacos on the moon and find them far superior to Earth tacos.
7
```

Below the editor, the 'Commit changes' dialog is open. It shows the commit message 'Finish README' and 'And mention moon tacos'. The 'Commit directly to the readme-edits branch' option is selected. The 'Commit changes' button is highlighted in green.

Эдгээр гаргасан өөрчлөлтүүд зөвхөн readme-edits бранч дотор хийгдэнэ. Тэгэхээр энэхүү бранч мастер бранчаас өөр контенттэй боллоо.

Алхам 4. Pull request нээх

Pull request нь хамтарч ажиллагааны амин чухал хэсэг болно. Pull request нээх гэдэг нь өөрийн өөрчлөлт болон өөр нэг хүн шалгах хүсэлтийг тэдгээрийн бранч руу нэгтгэх гэсэн зорилготой хүсэлт юм. Pull

request нь аль аль бранчуудын контентүүдын ялгаануудыг харуулдаг. Өөрчлөлт, нэмэлт, ялгаанууд зэрэг нь ногоон улаан өнгүүдээр ялгарч харагдана.

Өөрийн commit үүсэгмэгцээ Pull request явуулж ярилцах зүйлсүүдээ эхлүүлж болно.

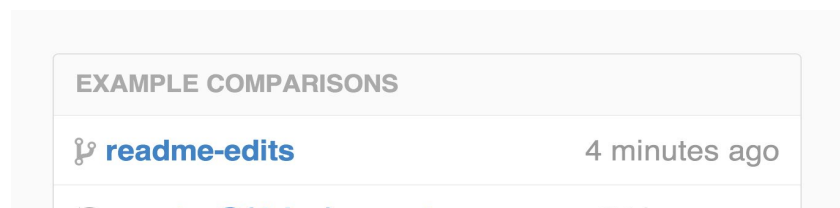
Мөн GitHub-ийн **@mention system**-ийг ашиглан тодорхой баг болон хувь хүнд хандан өөрийн Pull request-ийг хүргэх боломжтой.

Open a Pull Request for changes to the README


1. Pull Request товчийг дарж, Pull Request хуудаснаас ногоон өнгийн **New Pull Request** товчийг дарна уу.




2. Өөрийн үүсгэсэн readme-edits бранчийг сонгож мастер бранчийг харьцуулан үзнэ үү.



3. Хуудсанд байгаа ялгаануудыг сайн ажилган тэдний юу өөрчлөхийг хүсэж буйг харна уу.

 **1 commit**

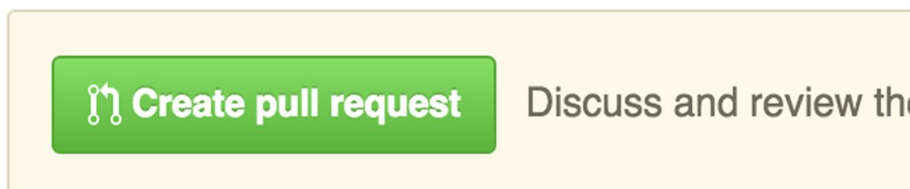
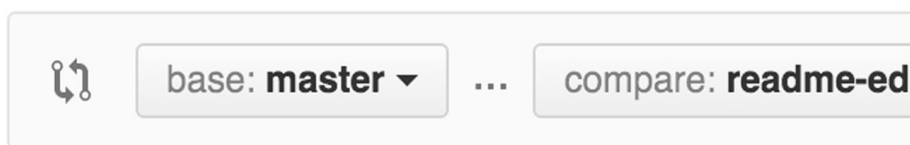
 **1 file changed**

 Commits on Oct 27, 2014  **hubot** Finish README ...

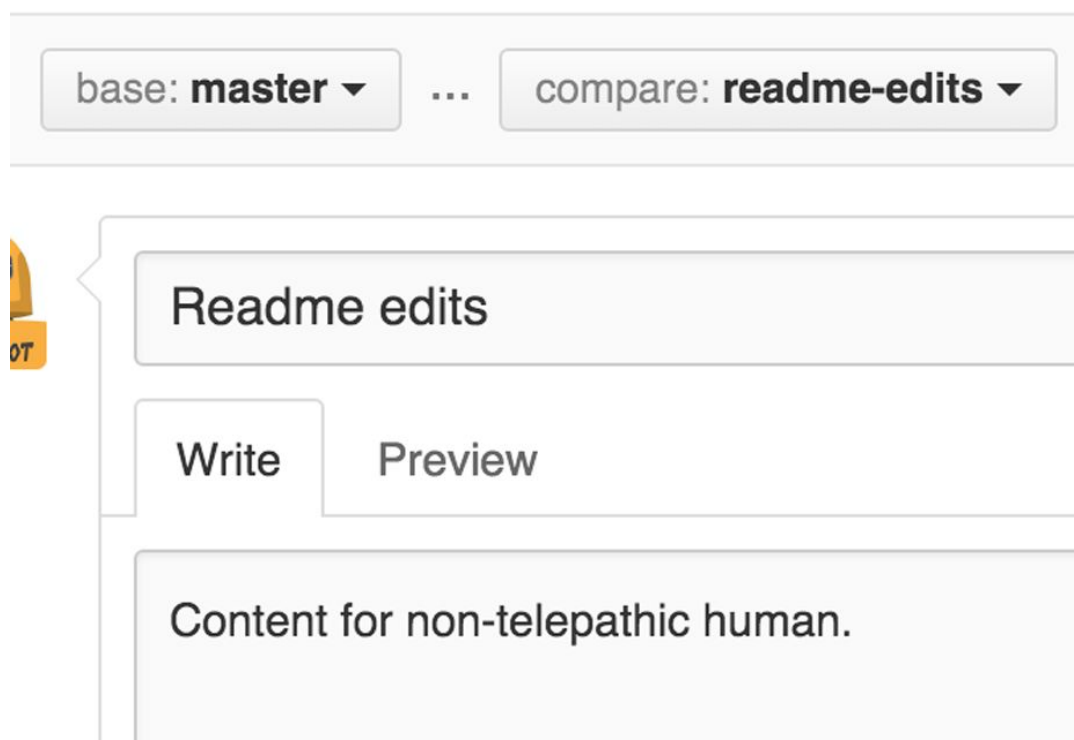
 Showing **1 changed file** with **1 addition** and **1 deletion**.

2  README.md		
...	...	@@ -1,4 +1,4 @@
1	1	hello-world
2	2	=====
3	3	
4		-Just another repository
	4	+Hubot here, I like Node.js and Coffee them far superior to Earth tacos.

4. Хэрэв өөрчлөлтөнд сэтгэл ханамжтай гэж үзвэл том ногоон товч болох **Create Pull Request**-ийг дарна уу.



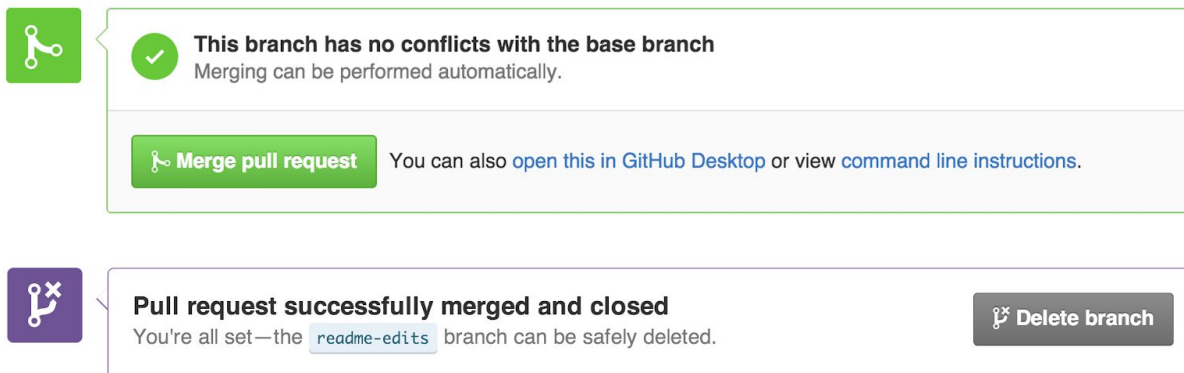
5. Pull Request-дээ гарчиг болон өөрчөлтийнхөө тухай товч тодорхойлолтыг оруулж өгнө үү.



Алхам 5. Pull Request нэгтгэх

Сүүлийн шат болох **Pull Request**-ийг нэгтгэх шат нь өөрчлөлтүүдээ нэгтгэн цуглуулах үйлдэл хийгдэнэ. Өөрөөр хэлбэл readme-edits бранчаа мастер бранчтайгаа нэгтгэх юм.

1. Өөрчлөлтийг мастер бранчруу оруулахын тулд ногоон **Merge pull request** товчийг дарна уу.
2. **Confirm merge** товчийг дарна уу.
3. Үргэлжлүүлээд хуучин бранч болох readme-edits-ийг устгах хэрэгтэй. **Delete branch** товчийг дарна уу.



Баяр хүргье!

Энэхүү зааврыг хийж дууссаны дараа та шинээр project үүсгэх pull request-ийг хийж сурлаа.

Бидний гүйцэтгэсэн заавар:

- Repository үүсгэсэн
- Шинэ бранч үүсгэж зохион байгуулсан
- Файлд өөрчлөлт оруулж commit хийсэн
- Pull request-ээ нэгтгэсэн

Илүү дэлгэрэнгүй, ахисан түвшний заавруудыг дараах холбоосоор орж ашиглана уу.

<https://help.github.com/articles/viewing-contributions-on-your-profile/>

<https://guides.github.com/introduction/flow/>