

Helpointa on, jos ihan ensiksi lisätään projektihakemistoon *.gitignore* jolla seuraava sisältö

```
node_modules
```

Tehdään projektihakemistosta git-repositorio:

- Ensin luodaan Githubin puolella samanniminen repositorio kuin oman koneen projektihakemisto on.
- Ei rastia ruutuun "initialize with a readme"
- Klikkaa "Create Repository"

Jos Git Init-komentoa EI OLE annettu oman koneen projektihakemistoon (mitä ei todennäköisesti ole tehty uusien projektien kohdalla):

- Scrollaa alas kohtaan "create a new repository on the command line" ja anna siinä lukevat komennot Git Bashiin, joka on avattu oman koneen oikeassa kansiossa, siis nämä:

```
echo "# Testirepo" >> README.md
git init
git add README.md
git add .
git commit -m "first commit"
git remote add origin git@github.com:osxosxos/Testirepo.git
git push -u origin master
```

Jos Git Init-komento ON annettu jo kertaalleen komentoriviltä ja projektikansiota on käytetty paikallisena repositoriona, päästään tällä:

- Scrollaa alas suoraan kohtaan "Push Existing Repository From The Command Line" ja anna siinä lukevat komennot Git Bashiin, joka on avattu oman koneen oikeassa kansiossa, siis nämä:

```
git remote add origin git@github.com:osxosxos/Testirepo.git
git push -u origin master
```

Se mitä tulee kohdan osxosxos/ jälkeen vaihtuu repon nimen mukaan, joten varminta on vain suoraan Copypastea nämä rivit Githubista, menee varmasti oikein. Ja jos mitään ei liiku, anna komennot git add . ja git commit -m "committi". (-m "committi"-päättettä ei saa unohtaa tai joudut vaikeuksiin!)

Tässä kohtaa tulee selväksi, onko SSH-avain olemassa (tosin jos sitä ei ole erikseen otettu pois, Githubiin on tehty Löllö-PC:lle oma avain. Se on nimeltään "varsinainen".) Avainta itseään ei pitäisi tarvita antaa tässä mihinkään, mutta git push -u origin master-komennon jälkeen, kehoitteen kysyessä "enter passphrase for key", vastaa KissanKusi. Älä säikähdä sitä, että mitään ei näy, kun kirjoitat, se kuuluu asiaan.

Luodaan heroku-sovellus komennolla *heroku create*. Näin:

Eli siis mennään Windowsin omaan komentokehoteeseen, Eli Git Bashiin, samassa sijainnissa missä äsken käytettiin Git Bashia. Siellä ensin komento:

```
heroku login
```

Sitten komento (edelleen Windowsin komentorivillä, koska Heroku-salasanantaminen ei toimi Git Bashissa):

```
heroku create --ssh-git
```

(huom!! ssh-sanan edessä KAKSI viivaa.)

Jos se kysyy Heroku-salasanan uudelleen, anna se. Ruudulla pitäisi pienen ruksuttamisen jälkeen lukea:

```
Creating app... done, ? glacial-ocean-11955
```

```
https://glacial-ocean-11955.herokuapp.com/ | git@heroku.com:glacial-ocean-11955.git
```

jossa glacial-ocean-11955 on Herokun antama hieno nimi joka vaihtelee sovelluksesta toiseen. Windowsin omassa komentokehoteessa ei nyt tehdä muuta mutta ÄLÄ SULJE SITÄ.

Vaihda Git Bashiin ja anna komento:

```
git remote add heroku git@heroku.com:glacial-ocean-11955.git
```

jossa sanan Heroku jälkeen tuleva osuus on kirjoitettu (koska windowsin copy-paste kusee) Windowsin komentorivin puolella pullahtaneista tiedoista, käytännössä viimeisen rivin keskellä olevan pystyviivan oikealla puolella oleva asia.

Saattaa käydä niinkin että Git tietää tämän jo jos kaikki tähän asti on mennyt täydellisen ihanasti, eli Git sanoo "remote already exists".

Sitten, jos ei virheilmoituksia tule (toimiessaan äskeinen komento ei tulosta välttämättä yhtään mitään, mutta komennolla `git remote -v` pitäisi tulla jotain mikä näyttää tuolta äskeiseltä), saapi vihdoin antaa Git Bashissa komennon

```
git push heroku master
```

Tästä seuraa paljon tulostusta näytölle, ja luultavasti joudut antamaan passfrasen KissanKusi.

Yksi viimeisistä riveistä on "verifying deploy... done."