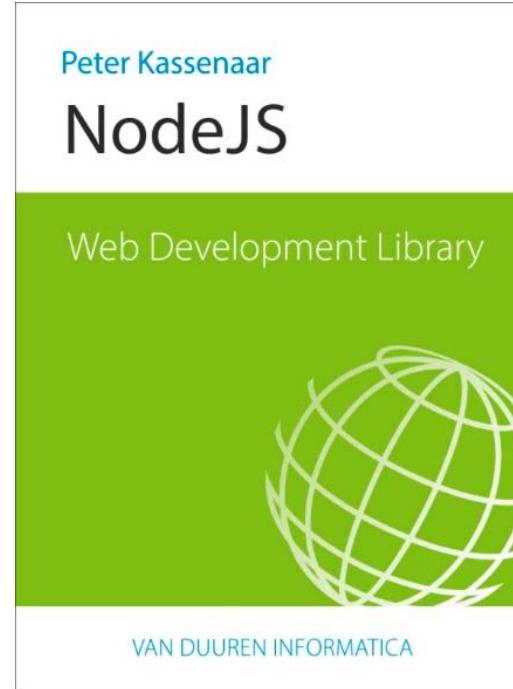


Node.js



Peter Kassenaar

Module 5 – Node.js Deployment

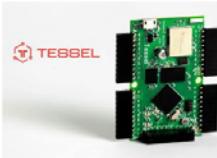


Hoofdstuk 7, p.183 en verder

Deployment - opties

- Eigen webserver
- PAAS - Platform as a Service
 - Heroku, Azure, NodeJitsu e.a.
- Internet of Things –
 - Raspberry Pi, Tessel, Arduino, e.a.

HARDWARE:



Tessel

Description:
"Tessel: hardware that speaks the language of the web."
Use your web development skills to make hardware devices with Tessel.

Price: \$99



Espruino

Description:
"What makes Espruino unique? You just plug the Espruino Board into your computer, open a Terminal application and start programming immediately without having to install special software on your PC or Mac."
Espruino uses Arduino-style commands to access the underlying hardware, while JavaScript supports Strings, Maps, Objects, and Dynamic Arrays. All on devices with as little as 8kb of RAM!"

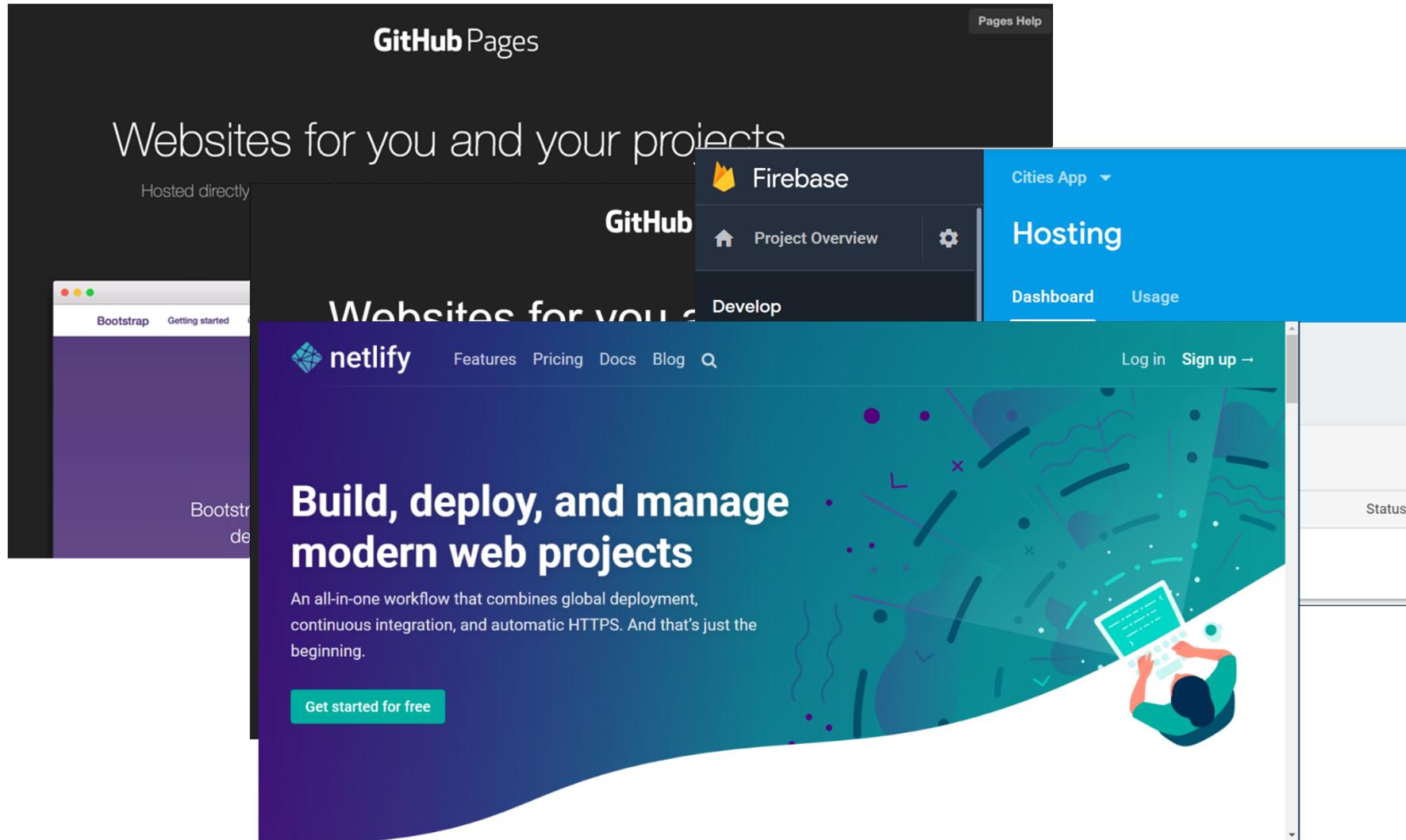
Price: £19

Additional

- Tessel supports packages from [npm](#)
- Additional modules will include: BLuetooth, RFID, GPS, Temp & Humidity
- iOS and Android app
- 180MHz ARM Cortex-M3 LPC1830
- 32mb SDRAM
- 32mb Flash
- TI CC3000 WiFi Radio
- 16-pin GPIO bank for prototyping
- Open source code, open source hardware
- [Github page](#)

- Pads to allow HC-05 Bluetooth modules to be added
- TM32 32-bit 72MHz ARM Cortex M3 CPU
- 256KB of Flash memory, 48KB of RAM
- 44 GPIO Pins
- WiFi Support with the TI CC3000
- Open source code, open source hardware
- [Code Examples](#)

Lots of free hosting sites available

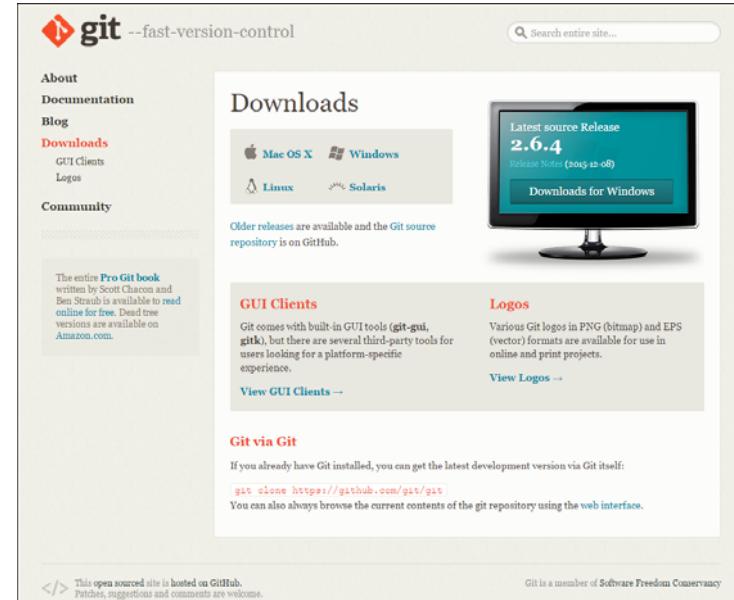


PaaS – Platform as a Service

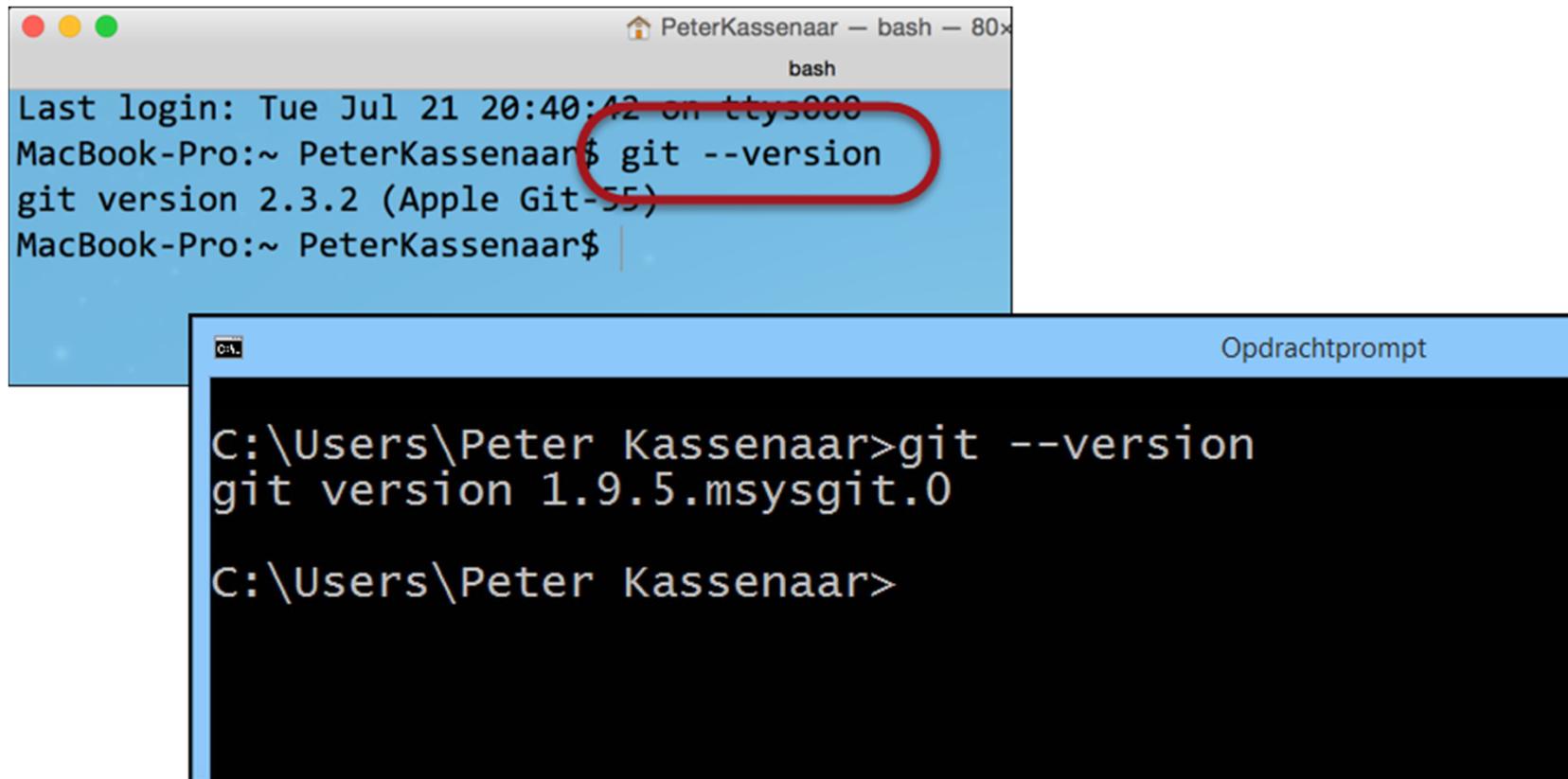
- Gespecialiseerde aanbieder hardware en software laten verzorgen
- Gratis instap-abonnement, maandelijkse fee bij add-ons
- Opties:
 - www.heroku.com
 - azure.microsoft.com
 - <https://modulus.io/>
 - Firebase, Github Pages, Netlify....
 - Google Cloud Services, Amazon Web Services

App voorbereiden

- Dynamische poort maken (niet meer: localhost:3000)
- ApiUrl met URL van de applicatie (voor hosting .json-data)
- Git installeren
 - Altijd: werken met lokale git-repository
 - Github niet verplicht, mag meestal wel, voor CI.



Lokale git-repo



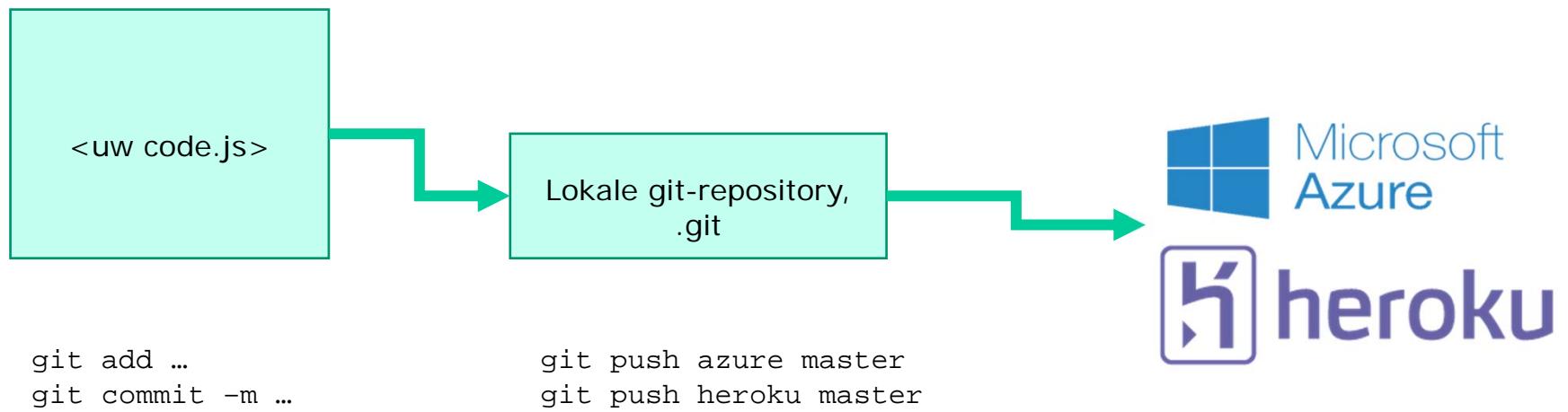
The image shows two separate terminal windows. The top window is a macOS terminal titled 'PeterKassenaar – bash – 80x' with a red circle highlighting the status bar. It displays the command 'git --version' and its output: 'git version 2.3.2 (Apple Git-55)'. The bottom window is a Windows Command Prompt titled 'Opdrachtprompt' with a blue border. It also displays the command 'git --version' and its output: 'git version 1.9.5.msysgit.0'. Both outputs show the command being run from the user's home directory.

```
Last login: Tue Jul 21 20:40:42 on ttys000
MacBook-Pro:~ PeterKassenaar$ git --version
git version 2.3.2 (Apple Git-55)
MacBook-Pro:~ PeterKassenaar$ |
```

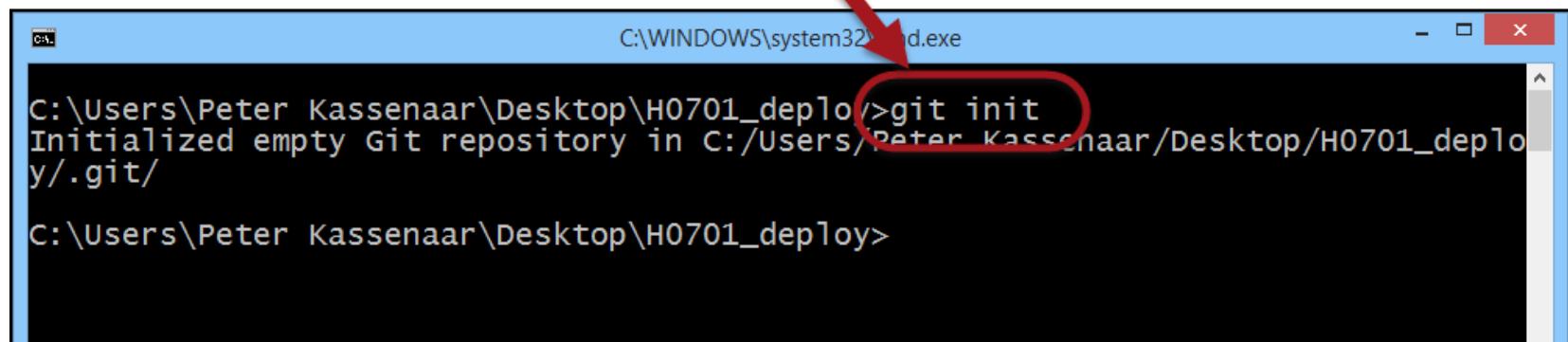
```
C:\Users\Peter Kassenaar>git --version
git version 1.9.5.msysgit.0

C:\Users\Peter Kassenaar>
```

Git workflow

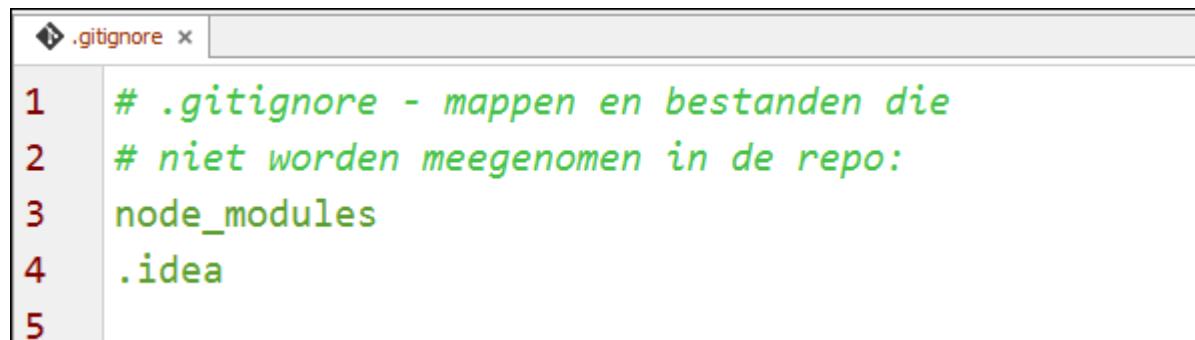


```
git init
```



```
C:\Windows\system32\cmd.exe
C:\Users\Peter Kassenaar\Desktop\H0701_deploy>git init
Initialized empty Git repository in C:/Users/Peter Kassenaar/Desktop/H0701_deploy/.git/
C:\Users\Peter Kassenaar\Desktop\H0701_deploy>
```

```
.gitignore -file
```



```
❶ # .gitignore - mappen en bestanden die
❷ # niet worden meegenomen in de repo:
❸ node_modules
❹ .idea
❺
```

Bestanden inchecken

Git status

Git add .

Git commit -m "message text"

```
C:\Users\Peter Kassenaar\Desktop\git-demo>git status
On branch master

Initial commit

Untracked files:
  (use "git add <file>..." to include in what will be committed)

    .bowerrc
    .gitignore
    data/
    package.json
    public/
    router/
    server.js

nothing added to commit but untracked files present (use "git add" to track)

C:\Users\Peter Kassenaar\Desktop\git-demo>
```

Na het toevoegen:

```
create mode 100644 public/js/lib/jquery/src/var/support.js
create mode 100644 public/js/lib/jquery/src/var/toString.js
create mode 100644 public/js/lib/jquery/src/wrap.js
create mode 100644 public/login.html
create mode 100644 router/index.js
create mode 100644 server.js

C:\Users\Peter Kassenaar\Desktop\git-demo>git status
On branch master
nothing to commit, working directory clean

C:\Users\Peter Kassenaar\Desktop\git-demo>git log
commit eac17fa3d81f0bad249960546362dac6ac2a5cd0
Author: Peter Kassenaar <info@kassenaar.com>
Date:   Mon Dec 14 14:08:09 2015 +0100

    initial commit (14-dec-2015)

C:\Users\Peter Kassenaar\Desktop\git-demo>
```

Deployment naar Azure? :

<https://docs.microsoft.com/en-us/azure/app-service/app-service-web-get-started-nodejs>

You can follow the steps below using a Mac, Windows, or Linux machine. Once the prerequisites are installed, it takes about five minutes to complete the steps.

Filter

App Service Documentation

> Overview

✓ Quickstarts

 Create .NET app

Create Node.js app

 Create PHP app

 Create Java app

 Create Python app

 Create static HTML site

> Tutorials

> Samples

> Concepts

> How-To guides

> Reference

> Resources

0:10 / 6:33

Prerequisites

To complete this quickstart:

- [Install Git](#)
- [Install Node.js and NPM](#)

If you don't have an Azure subscription, create a [free account](#) before you begin.

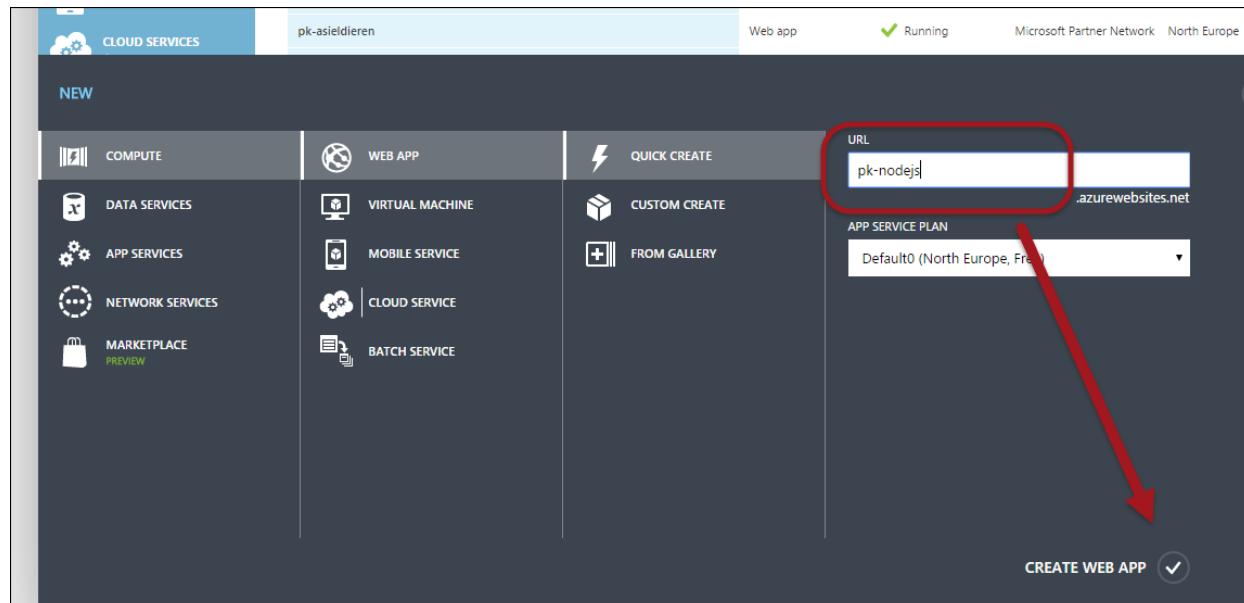
Download PDF

Is this page helpful?

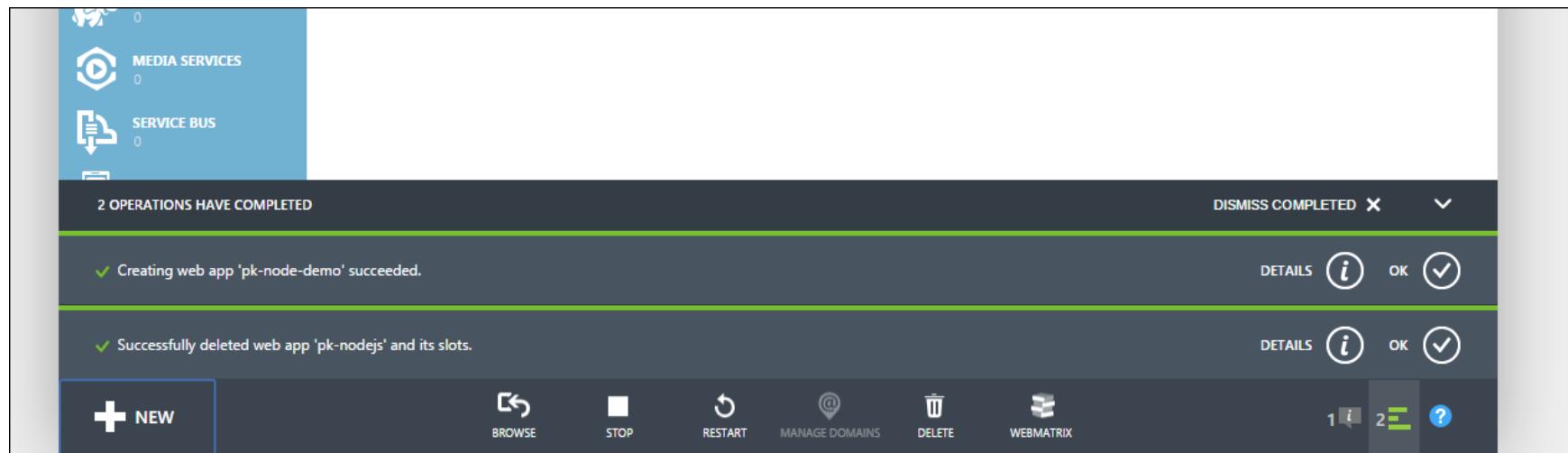
YES NO

(oud) Deployment naar Azure

- Account maken (free trial beschikbaar)
- Web app maken
- Publiceren vanuit Git.



Applicatie is gemaakt



Setup deployment control

The screenshot shows the Azure portal interface for a web app named 'pk-nodejs'. At the top, there's a timeline chart showing deployment slots: 'pk-webapi' (blue) and 'pk-nodejs' (orange). The timeline ranges from 10:00 to 11:00, with a single deployment slot shown at 10:55.

web endpoint status [PREVIEW](#)

You have not configured a web endpoint for monitoring. Configure one to get started.

[CONFIGURE WEB ENDPOINT MONITORING](#)

autoscale status

With a Standard web app, you can configure autoscale and spend only as much as you need for your service.

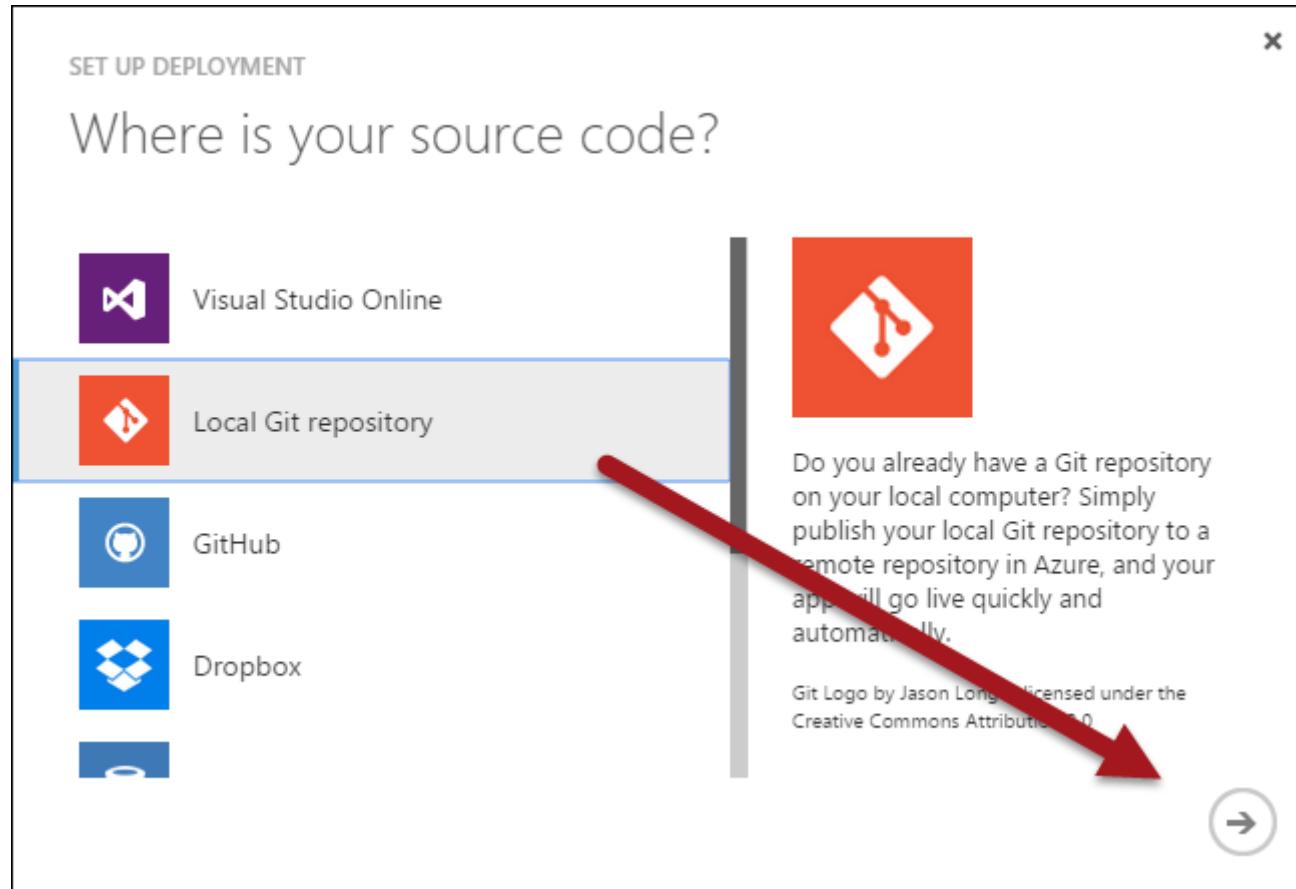
[CONFIGURE AUTOSCALE](#)

[AUTOSCALE OPERATION LOGS](#)

quick glance

- [Visit the new portal](#) [PREVIEW](#)
- [View Applicable Applications and services](#)
- [View connection strings](#)
- [Download the publish profile](#)
- [Reset your deployment credentials](#)
- [Reset your publish profile credentials](#)
- [Set up deployment from source control](#)
- [Add a new deployment slot](#)

Local Git repository



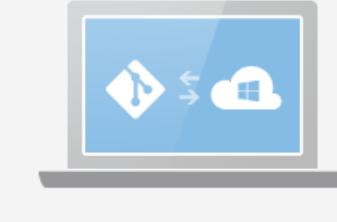
Remote host toevoegen aan local Git repository

pk-node-demo

DASHBOARD DEPLOYMENTS MONITOR WEBJOBS CONFIGURE SCALE LINKED RESOURCES BACKUPS

Not Modified: maandag 14 december 2015 14:15

Your Git repository is ready
Azure will build and deploy your web app on your next push.



YOUR REPOSITORY

GIT URL <https://PeterKassenaar@pk-node-demo.scm.azurewebsites.net>  

Push my local files to Windows Azure

```
Git remote add azure https://<....>.git
```

Publiceren naar Azure

```
git push azure master
```

```
remote:    |    +- depd@1.0.1
remote:    |    +- destroy@1.0.3
remote:    |    +- mime@1.3.4
remote:    +-+ serve-static@1.10.0
remote:    +-+ utils-merge@1.0.0
remote:    +-+ vary@1.0.1
remote:
remote: Finished successfully.
remote: Deployment successful.
To https://PeterKassenaar@pk-node-demo.scm.azurewebsites.net:443/pk-node-demo.git
 * [new branch]      master -> master
C:\Users\Peter Kassenaar\Desktop\git-demo>
```

The screenshot shows a deployment history entry for a Node.js application. It includes the deployment URL, a progress indicator, deployment details, and a timestamp.

GIT URL: <https://PeterKassenaar@pk-node-demo.scm.azurewebsites.net:443/pk-node-demo.git>

DEPLOYING: maandag 14 december 2015 14:20

Running deployment command...

ID: eac17fa3d8 AUTHOR: Peter Kassenaar DEPLOYED BY: PeterKassenaar

Not Modified: maandag 14 december 2015 14:20

Gereed – update ? Opnieuw via Git

The screenshot shows a web browser window displaying a table of books. The URL in the address bar is highlighted with a red box. The table has columns for ID, Titel, auteur, and ISBN. A blue button at the bottom right of the table area says 'Boek toevoegen'. Below the table, there is a link 'Startpagina'.

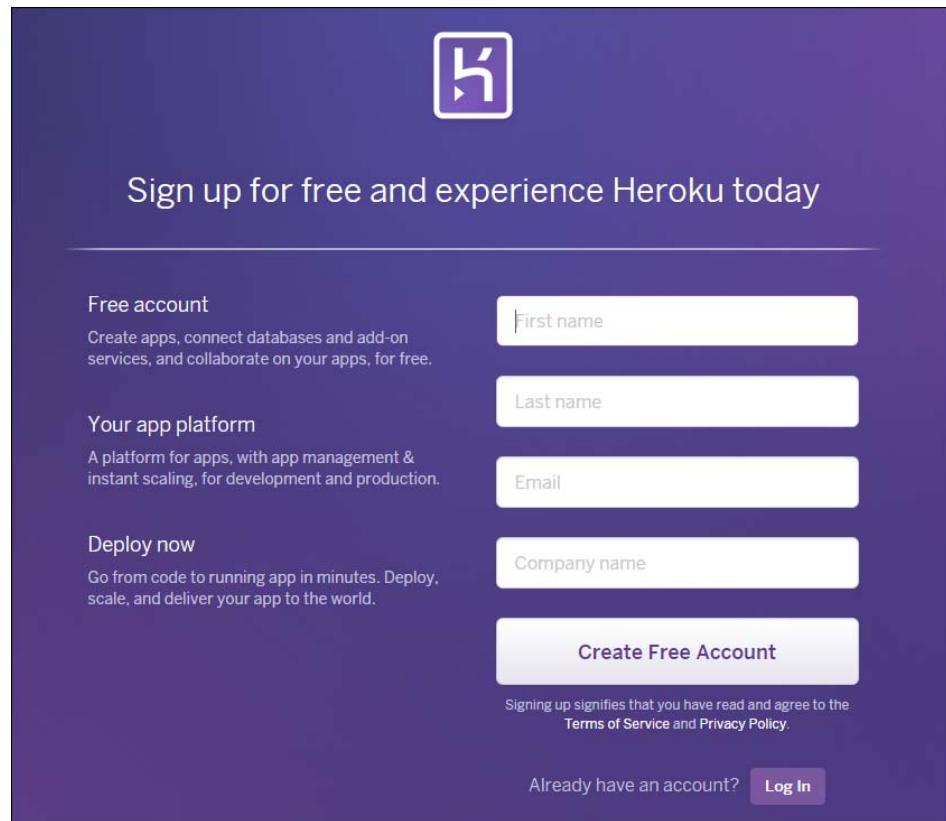
| ID | Titel | auteur | ISBN |
|----|---|-----------------|--------------|
| 1 | Web Development Library - AngularJS | Peter Kassenaar | 978905940787 |
| 2 | Focus op Fotografie: Portretfotografie | Pieter Dhaeze | 978905940812 |
| 3 | App Development Library - Apps maken voor Android | Michiel de Rond | 978905940839 |
| 4 | Ontdek de Windows-tablet | Bob van Duuren | 978905940618 |

Boek toevoegen

Startpagina

Publiceren bij Heroku

Proces grotendeels identiek
Download en installeer
de Heroku CLI
(voorheen: Heroku toolbelt)
Serie npm-hulpjes



<https://devcenter.heroku.com/start>

Heroku Dev Center [Get Started](#) [Documentation](#) [Changelog](#) More

Search Dev Center [Log in or Sign up](#)

Getting Started on Heroku

Step-by-step guides for deploying your first app and mastering the basics of Heroku

| | | | |
|---|--|--|--|
|  Node.js |  Ruby |  Java |  PHP |
|  Python |  Go |  Scala |  Clojure |

Volg de stappen in de Getting Started Guide voor NodeJS

Heroku Dev Center [⚡ Get Started](#) [Documentation](#) [Changelog](#) More

Search Dev Center  

Getting Started on Heroku with Node.js

Introduction

Set up

① The Heroku CLI requires Git, the popular version control system. If you don't already have Git installed, complete the following before proceeding:

- [Git installation](#)
- [First-time Git setup](#)

In this step you'll install the Heroku Command Line Interface (CLI). You use the CLI to manage and scale your applications, provision add-ons, view your application logs, and run your application locally.

Download and run the installer for your platform:

 macOS [Download the installer](#)
Also available via Homebrew:


 Windows
Download the appropriate installer for your Windows installation:
[64-bit installer](#) [Feedback](#)

Getting Started on Heroku with Node.js

Introduction

Set up

Prepare the app

Deploy the app

View logs

Define a Procfile

Scale the app

Declare app dependencies

Run the app locally

Push local changes

Provision add-ons

Start a console

Define config vars

Deploy the app

In this step you will deploy the app to Heroku.

Create an app on Heroku, which prepares Heroku to receive your source code.

```
$ heroku create
Creating sharp-rain-871... done, stack is heroku-18
http://sharp-rain-871.herokuapp.com/ | https://git.heroku.com/sharp-rain-871.git
Git remote heroku added
```

When you create an app, a git remote (called `heroku`) is also created and associated with your local git repository.

Heroku generates a random name (in this case `sharp-rain-871`) for your app, or you can pass a parameter to specify your own app name.

Now deploy your code:

```
$ git push heroku master
Counting objects: 488, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (367/367), done.
```

Volg de stappen:

- Applicatie maken
- Deployment
- Logs (=ter info)
- Procfile (=ter info)
- Update & push app

Bestaande app deployen naar Heroku?

The screenshot shows a web browser displaying the Heroku Dev Center. The page title is "Preparing a Codebase for Heroku Deployment". The left sidebar has a "CATEGORIES" section with links like "Heroku Architecture", "Deployment" (which is selected), "Deploying with Git", "Deploying with Docker", "Deployment Integrations", "Continuous Delivery", "Language Support", "Databases & Data Management", "Monitoring & Metrics", and "App Performance". The main content area shows the article's title, last update date (12 September 2018), and a "Table of Contents" with five items: 1. Track your codebase in a Git repository, 2. Add a Heroku Git remote, 3. Add a Procfile, 4. Listen on the correct port, and 5. Use a database or object storage instead of writing to your local filesystem.

<https://devcenter.heroku.com/articles/preparing-a-codebase-for-heroku-deployment>

Checkpoint

- Let bij deployment op juiste aanpassingen voor dynamische omgevingen
- Gebruik lokale Git-repository
- Publiceer vanuit commandline (of via visuele tool)

Oefeningen...

