

Kurulum

- [Sistem Gereksinimleri](#)
- [Linux Kurulumu](#)
- [Windows Kurulumu](#)

Sistem Gereksinimleri

- Wiki.js, Node.js'in desteklendiği hemen hemen her sistemde çalışır. Bu, Linux, macOS, Windows'un yanı sıra Docker/Kubernetes ve Heroku konteyner çözümlerinde çalışacağı anlamına gelir. Wiki.js zaman zaman yeni güncellemeleri, dilleri, temaları vb. otomatik olarak kontrol eder.
- **CPU:** Wiki.js tek bir CPU çekirdeğinde mükemmel şekilde çalışır. Ancak arkaplan çalışanlarından tam olarak yararlanmak için 2 veya daha fazla çekirdek önerilir.
- **RAM:** Linux sistemlerinin Wiki.js'yi çalıştırabilmesi için en az 1 GB RAM'e sahip olması gerekir. Windows ve macOS sistemleri genellikle biraz daha fazla RAM gerektirir. İşlemin kendisi genellikle yaklaşık 70 MB RAM'de yer alırken, bazı olaylar(sayfa oluşturma, indeksleme vb.) RAM kullanımında kısa süreli artışlara neden olur.
- **Depolama Alanı:** Depolama gereksinimleri gireceğiniz içeriğe göre belirlenir. Neredeyse tamamen metinden oluşan Wiki'lerin birkaç megabaytı aşması pek mümkün değildir. Ancak resim, video veya diğer dosyaları yüklediğiniz anda depolama gereksinimlerinizi buna göre planlamalısınız. En az 1 GB depolama alanı önerilir.
- Wiki.js, özel bir alt alan adı ve alan adı gerektirir. Wiki.js'i bir alt klasörle eşleyemezsiniz.
- En iyi kullanım için PostgreSQL'in 9.5 ve sonraki sürümleri önerilir. Bunun dışında MySQL 8.0 ve sonraki sürümleri, MariaDB 10.2.7 ve sonraki sürümleri, MS SQL Server 2012 ve sonraki sürümleri, SQLite 3.9 ve sonraki sürümleri önerilir. Bu veri tabanı motorlarından birini zaten kurmuş olmanız beklenmektedir. Wiki.js, boş bir veri tabanına bağlanmak için benzersiz bir kullanıcı ve şifreye ihtiyaç duyar.
- Node.js'in 12.0 sürümünden sonraki sürümleri için tek sayıyla biten sürümler desteklenmez (13.X, 15.X, 17.X).
- **Docker için:** Node.js, Docker görüntüsüne otomatik olarak dahil edilir. Ek olarak Node.js kurmanıza gerek yoktur.
- Wiki.js herhangi bir gerçek web sunucusuna (nginx veya Apache gibi) ihtiyaç duymaz. Ancak gelişmiş Ağ/DNS yapılandırmasına ihtiyacınız varsa Wiki.js'in önüne **“reverseproxy”** yerleştirmeniz gerekebilir.
- Google Chrome, Mozilla Firefox, Apple Safari gibi popüler tarayıcılar tarafından desteklenir. Tüm tarayıcılar varsayılan olarak arka planda otomatik olarak güncellenir.

Linux Kurulumu

Kullanılan sürüm Ubuntu 22.04 LTS sürümüdür.

Öncelikle sistemimize Docker kuruyoruz.

```
root@admin: /home/administrator
root@admin:/home/administrator# sudo apt -qqy -o Dpkg::Options::='--force-confdef' -o Dpkg::Options::='--force-confold' install ca-certificates curl gnupg lsb-release
lsb-release is already the newest version (11.1.0ubuntu4).
lsb-release set to manually installed.
ca-certificates is already the newest version (20230311ubuntu0.22.04.1).
ca-certificates set to manually installed.
gnupg is already the newest version (2.2.27-3ubuntu2.1).
gnupg set to manually installed.
The following NEW packages will be installed:
  curl
0 upgraded, 1 newly installed, 0 to remove and 2 not upgraded.
Need to get 194 kB of archives.
After this operation, 454 kB of additional disk space will be used.
Selecting previously unselected package curl.
(Reading database ... 198920 files and directories currently installed.)
Preparing to unpack .../curl_7.81.0-1ubuntu1.13_amd64.deb ...
Unpacking curl (7.81.0-1ubuntu1.13) ...
Setting up curl (7.81.0-1ubuntu1.13) ...
Processing triggers for man-db (2.10.2-1) ...
```

Docker'ı yüklemek için bağımlılıkları yüklüyoruz:

```
sudo apt -qqy -o Dpkg::Options::='--force-confdef' -o Dpkg::Options::='--force-confold' install ca-certificates curl gnupg lsb-release
```

```
root@admin:/home/administrator# sudo mkdir -p /etc/apt/keyrings
root@admin:/home/administrator# curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
root@admin:/home/administrator# echo "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

Docker paket kayıt defterini kaydediyoruz:

```
sudo mkdir -p /etc/apt/keyrings
```

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
```

```
echo "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg]https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

```
root@admin:/home/administrator# sudo apt -qqy update
2 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@admin:/home/administrator# apt list --upgradable
Listing... Done
wikijs/wikijs-updates 1.72.4-0ubuntu0.22.04.1 amd64 [upgradable from: 1.72.2-0ubuntu2]
libwikijs/wikijs-updates 1.72.4-0ubuntu0.22.04.1 amd64 [upgradable from: 1.72.2-0ubuntu2]
root@admin:/home/administrator# sudo apt -qqy -o Dpkg::Options::='--force-confdef' -o Dpkg::Options::='--force-confold' install docker-ce docker-ce-cli containerd.io docker-compose-plugin
The following additional packages will be installed:
  docker-buildx-plugin docker-ce-rootless-extras git git-man liberror-perl libslirp0 pigz slirp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin git git-man liberror-perl libslirp0 pigz slirp4netns
0 upgraded, 12 newly installed, 0 to remove and 2 not upgraded.
Need to get 118 MB of archives.
After this operation, 430 MB of additional disk space will be used.
Selecting previously unselected package pigz.
(Reading database ... 198927 files and directories currently installed.)
Preparing to unpack .../00-pigz_2.6-1_amd64.deb ...
Unpacking pigz (2.6-1) ...
Selecting previously unselected package containerd.io.
Preparing to unpack .../01-containerd.io_1.6.24-1_amd64.deb ...
Unpacking containerd.io (1.6.24-1) ...
Selecting previously unselected package docker-buildx-plugin.
```

Paket güncellemelerini yeniliyoruz ve Docker'ı yüklüyoruz:

```
sudo apt -qqy update
```

```
sudo apt -qqy -o Dpkg::Options::='--force-confdef' -o Dpkg::Options::='--force-confold' install docker-ce docker-ce-cli containerd.io docker-compose-plugin
```

Şimdi konteynerleri kurmak için gerekli hazırlıklara geçiyoruz.

```
root@admin:/home/administrator# mkdir -p /etc/wiki
root@admin:/home/administrator# openssl rand -base64 32 > /etc/wiki/.db-secret
root@admin:/home/administrator# docker network create wkinet
root@admin:/home/administrator# docker volume create pgdata
pgdata
```

Wiki.js için kurulum dizini oluşturduk:

```
mkdir -p /etc/wiki
```

Dijital kimlik bilgisi oluşturduk:

```
openssl rand -base64 32 > /etc/wiki/.db-secret
```

Docker iç ağı oluşturduk:

docker network create wikinet

PostgreSQL için veri birimi oluşturduk:

docker volume create pgdata

Şimdi gerekli konteynerların kurulumuna geçiyoruz.

```
root@admin:/home/administrator# docker create --name=db -e POSTGRES_DB=wiki -e POSTGRES_USER=wiki -e POSTGRES_PASSWORD_FILE=/etc/wiki/.db-secret -v /etc/wiki/.db-secret:/etc/wiki/.db-secret:ro -v pgdata:/var/lib/postgresql/data --restart=unless-stopped -h db --network=wikinet postgres:15
Unable to find image 'postgres:15' locally
15: Pulling from library/postgres
a883e7c4d830: Pull complete
12dd3b657e91: Pull complete
147adba297eb: Pull complete
56ee7c8dd616: Pull complete
f9aac30f429: Pull complete
7ad57b8030eb: Pull complete
786562b3be85: Pull complete
84efcd3a2a4: Pull complete
2c35234636c9: Pull complete
6e62fa63f18: Pull complete
f1a157d7d7b0: Pull complete
6b153abf3882: Pull complete
7877e54346e0: Pull complete

Status: Downloaded newer image for postgres:15

root@admin:/home/administrator# docker create --name=wiki -e DB_TYPE=postgres -e DB_HOST=db -e DB_PORT=5432 -e DB_PASS_FILE=/etc/wiki/.db-secret -v /etc/wiki/.db-secret:/etc/wiki/.db-secret:ro -e DB_USER=wiki -e DB_NAME=wiki -e UPGRADE_COMPANION=1 --restart=unless-stopped -h wiki --network=wikinet -p 80:3000 -p 443:3443 ghcr.io/requarks/wiki:2
Unable to find image 'ghcr.io/requarks/wiki:2' locally
2: Pulling from requarks/wiki
51e352748f53: Pull complete
2629b68d4311: Pull complete
ddb7cc70f260: Pull complete
18afe6373474: Pull complete
88ec59e3b80e: Pull complete
4f4fb708ef54: Pull complete
710fc81a2a82: Pull complete
4e75fc391abd: Pull complete
d11f75e9538b: Pull complete
375954564088: Pull complete
66015734afb2: Pull complete
6c63ee65d20b: Pull complete
2442912e7d55: Pull complete

Status: Downloaded newer image for ghcr.io/requarks/wiki:2

root@admin:/home/administrator# docker create --name=wiki-update-companion -v /var/run/docker.sock:/var/run/docker.sock:ro --restart=unless-stopped -h wiki-update-companion --network=wikinet ghcr.io/requarks/wiki-update-companion:latest
```

```
docker create --name=db -e POSTGRES_DB=wiki -e POSTGRES_USER=wiki -e POSTGRES_PASSWORD_FILE=/etc/wiki/.db-secret -v /etc/wiki/.db-secret:/etc/wiki/.db-secret:ro -v pgdata:/var/lib/postgresql/data --restart=unless-stopped -h db --network=wikinet postgres:15
```

```
docker create --name=wiki -e DB_TYPE=postgres -e DB_HOST=db -e DB_PORT=5432 -e DB_PASS_FILE=/etc/wiki/.db-secret -v /etc/wiki/.db-secret:/etc/wiki/.db-secret:ro -e DB_USER=wiki -e DB_NAME=wiki -e UPGRADE_COMPANION=1 --restart=unless-stopped -h wiki --network=wikinet -p 80:3000 -p 443:3443 ghcr.io/requarks/wiki:2
```

```
docker create --name=wiki-update-companion -v /var/run/docker.sock:/var/run/docker.sock:ro --restart=unless-stopped -h wiki-update-companion --network=wikinet ghcr.io/requarks/wiki-update-companion:latest
```

Şimdi de firewall kuruyoruz ve aktif ediyoruz.

```
root@admin: /home/administrator
root@admin:/home/administrator# sudo ufw allow ssh
Rules updated
Rules updated (v6)
root@admin:/home/administrator# sudo ufw allow http
Rules updated
Rules updated (v6)
root@admin:/home/administrator# sudo ufw allow https
Rules updated
Rules updated (v6)
root@admin:/home/administrator# sudo ufw --force enable
Firewall is active and enabled on system startup
```

```
sudo ufw allow ssh

sudo ufw allow http

sudo ufw allow https

sudo ufw --force enable
```

Konteynerleri başlatıyoruz.

```
docker start db

docker start wiki

docker start wiki-update-companion
```

Tarayıcınızdaki URL kısmına sunucunuzun IP'sini veya alan adını girin.

Wiki.js Setup

You are about to install Wiki.js 2.5.300.

ADMINISTRATOR ACCOUNT

Administrator Email

The email address of the administrator account.

Password

At least 8 characters long. 0 / 255

Confirm Password

Verify your password again. 0 / 255

SITE URL

Site URL

https://wiki.yourdomain.com

Full URL to your wiki, without the trailing slash (e.g. https://wiki.example.com). This should be the public facing URL, not the internal one if using a reverse-proxy.

TELEMETRY

☒ Allow Telemetry

Help Wiki.js developers improve this app with anonymized telemetry. [Learn more](#)

✓ INSTALL

Gerekli bilgileri kendinize göre doldurun.

Güçlü bir şifre koymayı unutmayın!

Telemetry'i açabilirsiniz. Telemetry, Wiki.js'in geliştiricilerine feedback vermek için kullanılır. Bilgilerinizi girdikten sonra "Install" deyip devam edin.

Wiki.js Setup

You are about to install Wiki.js 2.5.300.

ADMINISTRATOR ACCOUNT

Administrator Email

The email address of the administrator account.

Password

At least 8 characters long. 14 / 255

Confirm Password

Verify your password again. 14 / 255

SITE URL

Site URL

Full URL to your wiki, without the trailing slash (e.g. https://wiki.example.com). This should be the public facing URL, not the internal one if using a reverse-proxy.

TELEMETRY

☒ Allow Telemetry

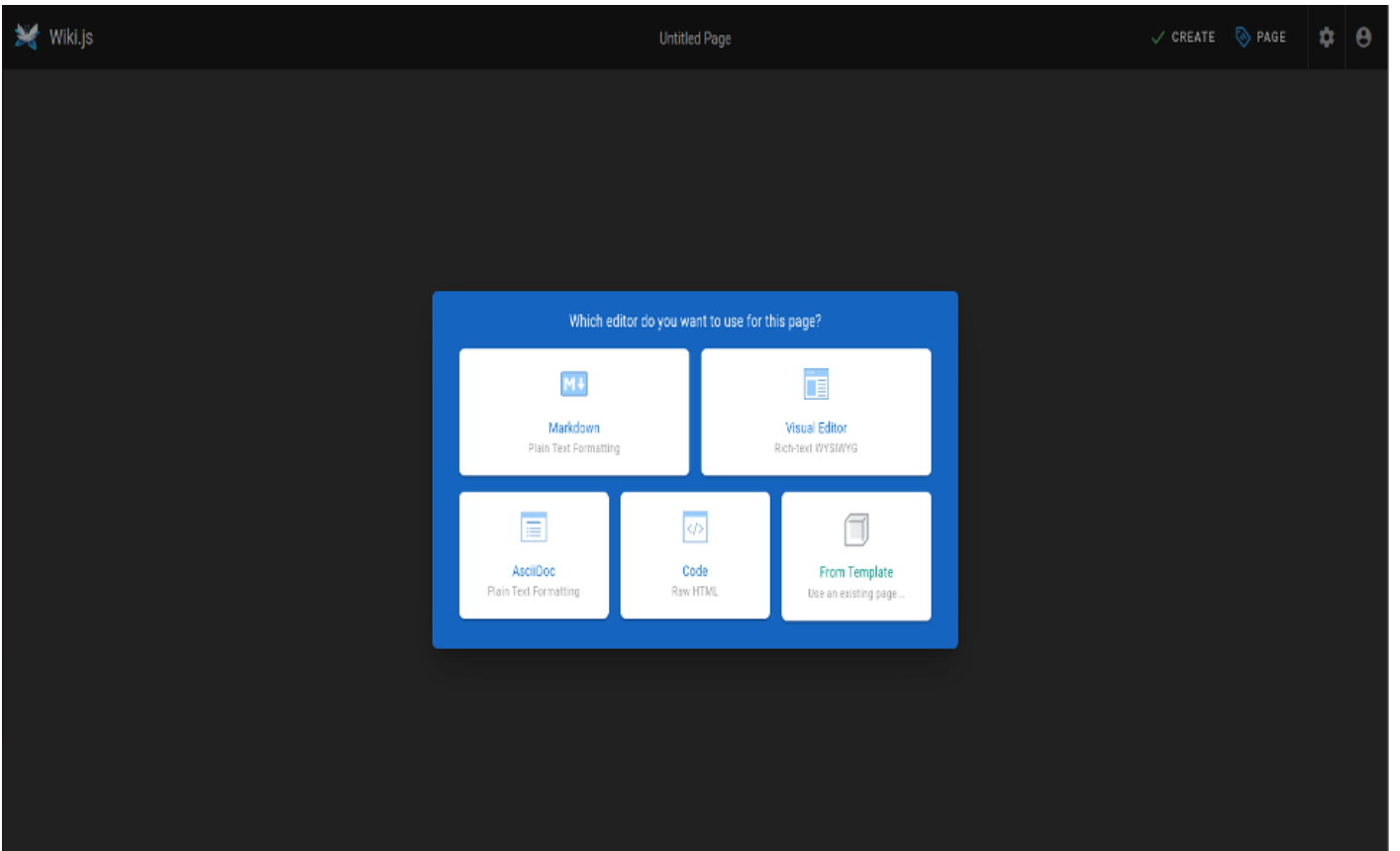
Help Wiki.js developers improve this app with anonymized telemetry. [Learn more](#)

✓ INSTALL

Devam ettikten sonra karşınıza çıkan sayfaya bilgilerinizi girip giriş yapın.



“Create Home Page” seçeneğine tıklayıp sayfamız için dilediğimiz editörü seçebiliriz.



Windows Kurulumu

Kullanılan sürüm Windows 11 Home sürümüdür.

Öncelikle arama kısmına “powershell” yazıp yönetici olarak çalıştır deyippowershell’i açıyoruz.

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Windows\system32> Invoke-WebRequest -Uri "https://github.com/Requarks/wiki/releases/latest/download/wiki-js-windows.tar.gz"
```

Wiki.js son sürümünü indiriyoruz.

```
Invoke-WebRequest -Uri
"https://github.com/Requarks/wiki/releases/latest/download/wiki-js-
windows.tar.gz" -OutFile "wiki-js.tar.gz"
```

```
PS C:\Windows\system32> New-Item -Path "C:\\" -Name "wiki" -ItemType "directory"

Directory: C:\

Mode                LastWriteTime         Length Name
----                -
d-----          1.10.2023   04:08         wiki

PS C:\Windows\system32> tar xzf wiki-js.tar.gz -C "C:\wiki"
PS C:\Windows\system32> cd C:\wiki
```

İndirdiğimiz paketi seçtiğimiz son varış noktasına çıkarıyoruz ve paketin bulunduğu path’e gidiyoruz:

```
New-Item -Path "C:\\" -Name "wiki" -ItemType "directory"

tar xzf wiki-js.tar.gz -C "C:\wiki"

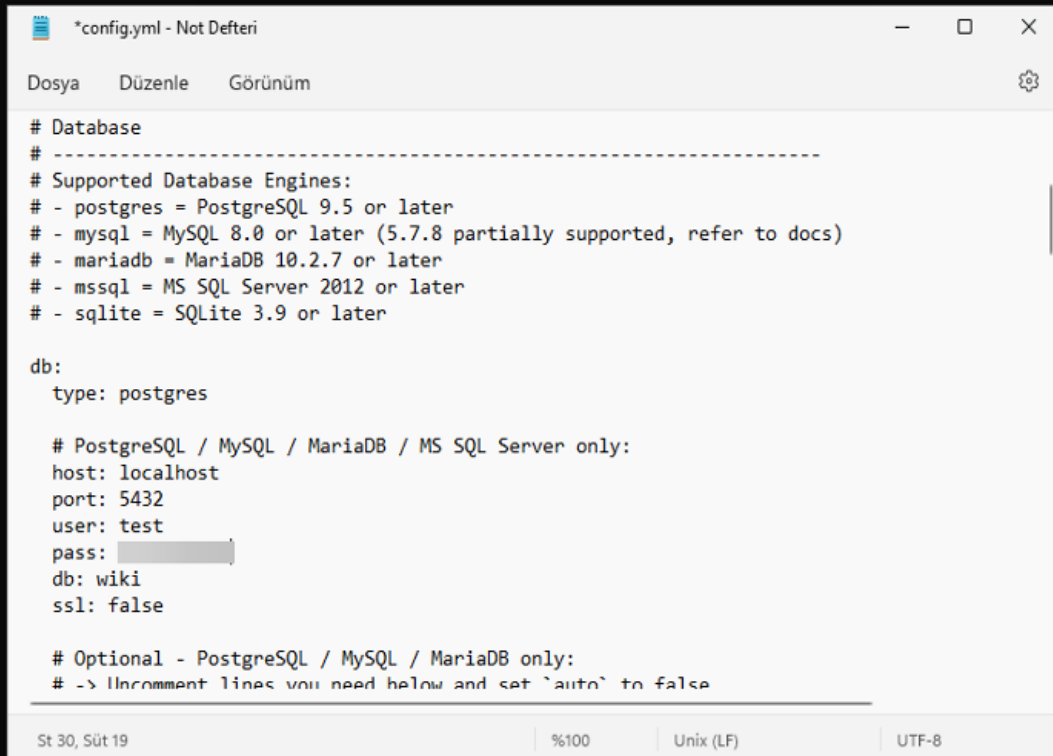
cd C:\wiki
```

```
Administrator: Windows PowerShell
PS C:\wiki> Rename-Item -Path config.sample.yml -NewName config.yml
```

Örnek yapılandırma dosyasını config.yml olarak yeniden adlandırıyoruz:

```
Rename-Item -Path config.sample.yml -NewName config.yml
```

```
PS C:\wiki> notepad .\config.yml
PS C:\wiki>
```



```
*config.yml - Not Defteri
Dosya  Düzenle  Görünüm

# Database
# -----
# Supported Database Engines:
# - postgres = PostgreSQL 9.5 or later
# - mysql = MySQL 8.0 or later (5.7.8 partially supported, refer to docs)
# - mariadb = MariaDB 10.2.7 or later
# - mssql = MS SQL Server 2012 or later
# - sqlite = SQLite 3.9 or later

db:
  type: postgres

# PostgreSQL / MySQL / MariaDB / MS SQL Server only:
host: localhost
port: 5432
user: test
pass: 
db: wiki
ssl: false

# Optional - PostgreSQL / MySQL / MariaDB only:
# -> Uncomment lines you need below and set `auto` to false

St 30, Süt 19    %100    Unix (LF)    UTF-8
```

Favori metin düzenleyicinizi (örn. Not Defteri) kullanarak yapılandırma dosyasını düzenleyin. PostgreSQL, MySQL, MariaDB, MS SQL Server, SQLite veri tabanlarından hangisini kullanmak istiyorsanız onu belirtin ve bağlantı noktası ayarlarınızı girin. Ben PostgreSQL kullandım siz farklı bir veri tabanı kullanmak istiyorsanız <https://docs.requarks.io/install/config> sayfasını ziyaret edip Database conf ayarlarına göre veri girebilirsiniz:

```
notepad .\config.yml
```



Welcome to your wiki!

Let's get started and create the home page.

+ CREATE HOME PAGE

ADMINISTRATION

Tarayıcınızdaki kurulum sayfası açılıncaya kadar bekleyin ve kurulumu tamamlamak için kurulum sihirbazını tamamlayın:

node server

Windows bilgisayarınızda “node.js” kurulu olmalıdır ve veri tabanınızile uyumlu bir biçimde çalışmalıdır!