

Deployment

Gold - Chapter 6 - Topic 5

Selamat datang di **Chapter 6 Topic 5**
online course **Back End Java** dari
Binar Academy!

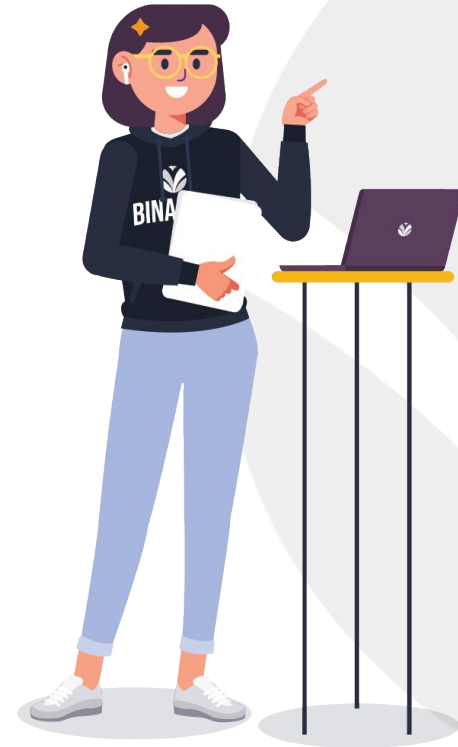


Hai kamu, kita ketemu lagi ☐

Pada topik sebelumnya, kita udah belajar tentang Spring Java Logging.

Lanjut di topik lima ini, kita bakal mengelaborasi tentang **Deployment**. Mulai dari konsep sampai persiapan deployment menggunakan Railway.app. Pokoknya semuanya kita pelajarin bareng-bareng.

Yuk, langsung aja kita kepoin~



Dari sesi ini, kita bakal bahas hal-hal berikut:

- Konsep Deployment
- Persiapan Deployment - menggunakan Railway.app



Seperti yang kita sebutin kemarin, materi deployment ini bakal jadi materi yang serius.

Walaupun serius, kamu nggak perlu khawatir karena kita coba otak atik materi ini mulai dari perkenalannya dulu.

Sebelum bahas deployment, gimana kalau kita mengawali obrolan kita dengan **Persiapan Deployment?**

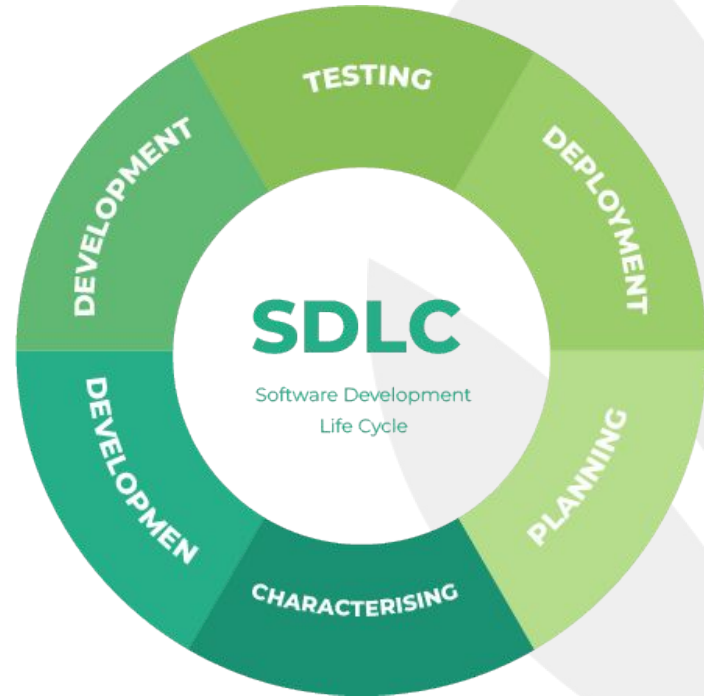


“Deployment itu apa sih?”

Deployment merupakan tahap lanjutan setelah tahap development dan testing.

Seorang Product Owner bakal melakukan list fitur apa aja yang mau di deliver ke production dan jadwal untuk naik ke production.

Tim Development juga menentukan versi dari release yang bakal dilakukan.



Biasanya nih ya, di Project Management Waterfall, banyak approval yang harus dilalui supaya bisa naik ke production.

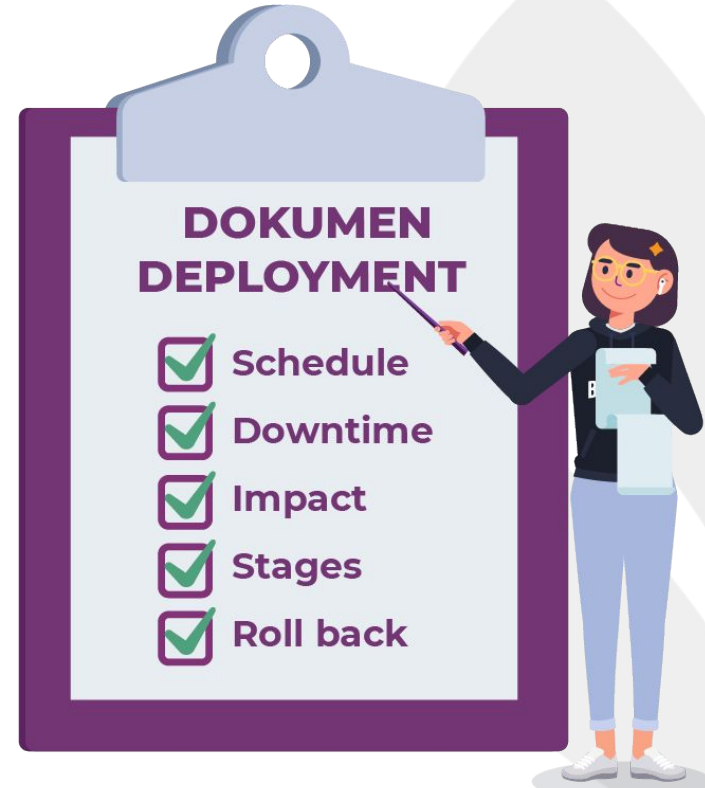
Approval tersebut disertai dengan **dokumen deployment** dan **dokumen test result**.



“Emang dokumen buat deployment itu apa aja?”

Dokumen deployment tuh isinya kayak gini, sob:

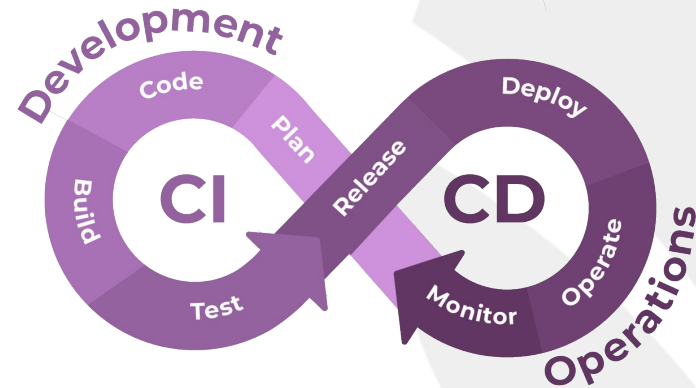
- Jadwal deployment.
- Downtime dari deployment.
- Impact dari deployment.
- Tahapan melakukan deployment.
- Tahapan melakukan roll back kalau terjadi incident saat melakukan deploy.



Nah, untuk bisa bantu deployment supaya prosesnya jadi lebih cepat, kita bisa pakai CI/CD.

Dengan **CI/CD** kita bisa **gampang banget** buat melakukan **deployment** dan melakukan rollback versi sebelumnya kalau ada insiden khusus setelah melakukan deploy.

Pembahasan tentang CI/CD bakal kita kupas nanti di topic selanjutnya, ya!



Ada beberapa hal yang harus dilakukan sebelum melakukan deployment~

Sebelum melakukan deploy terhadap project dari challenge yang udah dibuat, kita harus cek apakah aplikasi udah berjalan dengan baik atau belum. Caranya gini:

1. Pastikan aplikasi bisa berjalan saat di-run dan nggak ada error. Pastikan juga semua fitur bisa berfungsi dengan baik.



2. Database yang dipakai saat ini masih local, konfigurasi seperti URL, username, dan password nantinya bakal diubah karena database yang dipakai bukan pakai local server lagi.
3. Pastikan juga udah pakai log dengan baik buat memudahkan debugging proses pas aplikasi udah di-deploy.

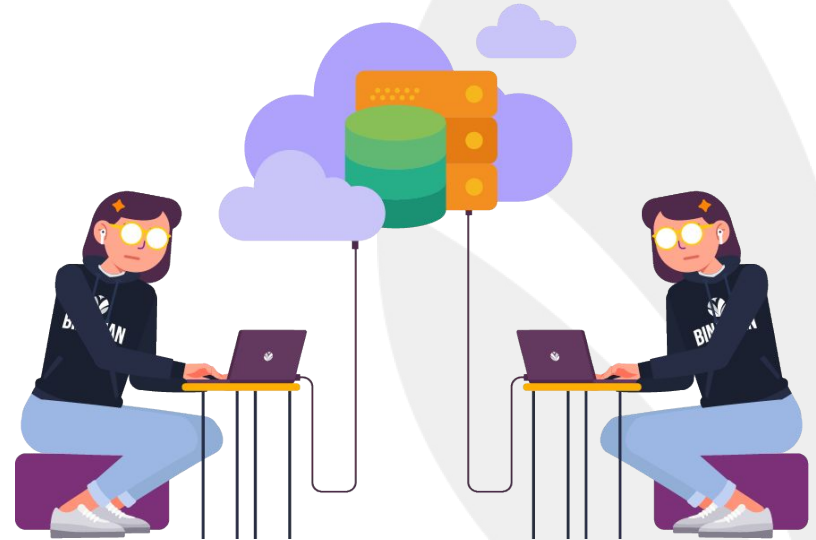
Dalam melakukan deploy, kita nggak pakai server sendiri, melainkan pakai **cloud platform**.



Pasti kamu udah nggak asing sama istilah cloud, kan?

Cloud merupakan sekumpulan server dan pusat data yang tersebar di seluruh dunia yang di dalamnya bisa menyimpan data kita.

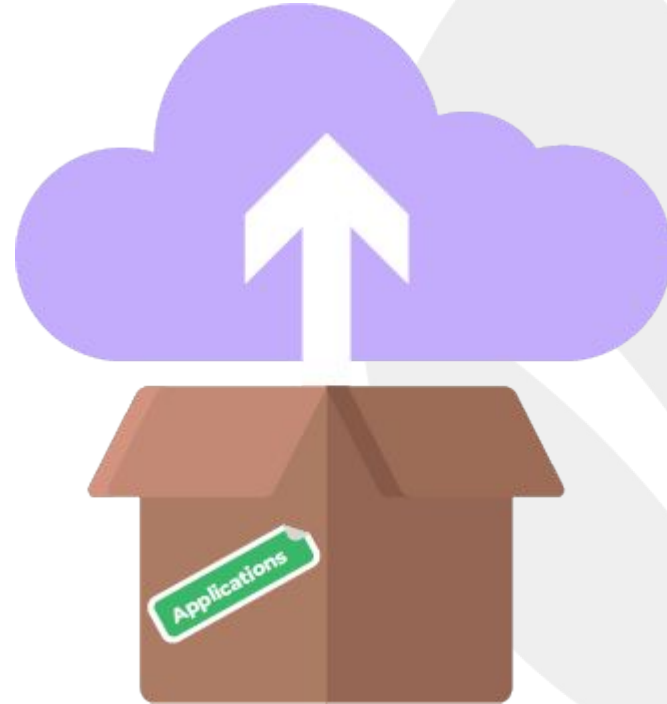
Cloud platform menyediakan sebuah platform untuk mempublikasikan sebuah aplikasi sebagai pengganti dari server on premise.



Server on premise perlu pengadaan infrastruktur dan pengelolaan sendiri sehingga jadi sangat mahal.

Dengan adanya cloud platform ini, ibaratnya kita bisa menumpang untuk melakukan publikasi di aplikasi yang udah dibikin.

Canggih banget, khann?!



Saat ini, udah ada beberapa penyedia cloud platform terkenal, lho!

- Amazon Web Services (AWS) punya Amazon.
- Google Cloud platform (GCP) punya Google.
- Openshift Cloud Platform (OCP) punya Red Hat.
- Railway.app

Tapi, kita cuma bakal pakai salah satu cloud platform yang sesuai sama kebutuhan kita, yaitu **Railway.app**.



Kalau tadi belajar cara mempersiapkan deployment, sekarang kita masuk ke yang lebih detail nih, yaitu **Persiapan Deployment dengan Railway.app**.

Hmm kira-kira bakal kayak gimana, yaa?

Kita cari tahu, yuk!



Yuk kenalan sama railway.app ☐

Pernah denger website railway.app? Sebagaimana udah kita mention sebelumnya, doi adalah salah satu perusahaan penyedia server.

Ngomong-ngomong, Railway.app ini terinspirasi dari Heroku, lho. Iya, Heroku yang berbasis Docker dengan plugin untuk berbagai penyimpanan data.



Nggak seperti kebanyakan hosting provider yang lain, tools Railway.app mendorong semacam injection service yang memungkinkan kode dijalankan secara lokal untuk menggunakan layanan yang dihosting di cloud.

Selain itu, Railway.app dibuat dengan tujuan membuat development tools lebih mudah diakses oleh developer.



Fitur yang ada di Railway.app

Yepp, belum afdol kayaknya kalau kita kenalan sama server ini tapi nggak ngebahas fitur yang ada di dalamnya. Ada apa aja sih?

- **Autoscaling**

Berdasarkan load, Railway.app akan secara otomatis meng-scale aplikasi untuk memenuhi permintaan user.



- **Metrics**

Railway.app menawarkan berbagai real time metric untuk men-debug issue secara instan.

Platform ini akan mengirimkan query untuk negative trend dan menunjukkan kepada developer apa yang salah, sehingga developer bisa melakukan troubleshoot issue pada saat diperlukan.



- **Aplikasi berjalan tanpa batas**

Dibandingkan dengan Heroku, di mana aplikasi akan shut down saat tidak menerima input apapun selama lebih dari 30 menit, deployment di Railway.app berjalan tanpa batas.

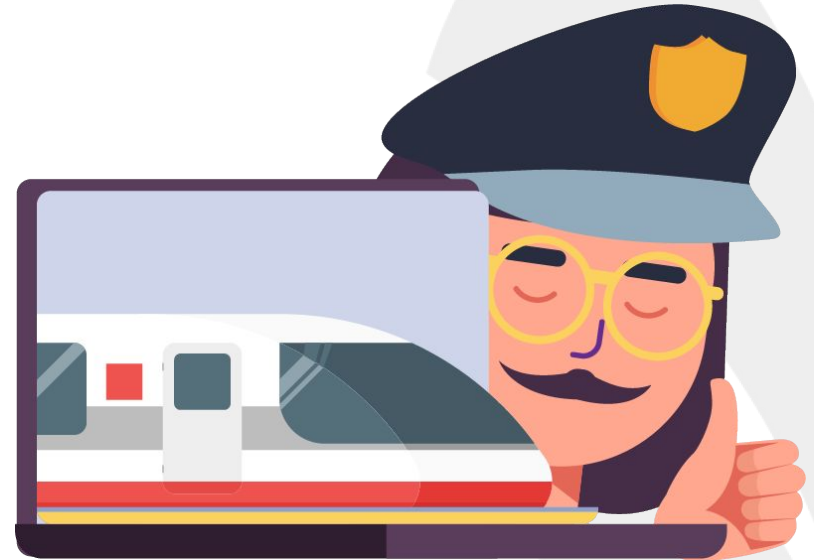
Selain itu, kamu juga bisa menjalankan aplikasi menggunakan Procfile atau deploy file Docker.



Apa aja sih kelebihan dari Railway.app?

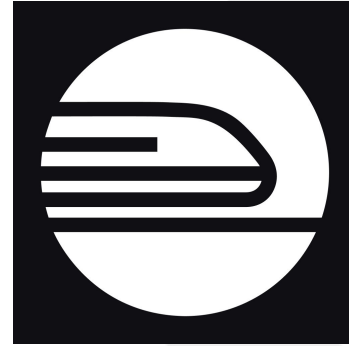
Ada banyak keunggulan dari Railway ini, yaitu:

- Kamu bisa memilih database kamu. Mulai dari PostgreSQL, MySQL atau Redis
- Ada banyak template yang tersedia. Kamu bisa mem-build aplikasimu dari awal alias from scratch atau pilih salah satu template yang ada.



- Lastly, kamu bisa pake railway.app buat deploy backend, database maupun frontend. Dimana untuk menggunakannya cukup register akun kamu.

Gimana nih, keunggulan Railway.app udah bisa di approve sama kamu belum?



Backend
Deploy

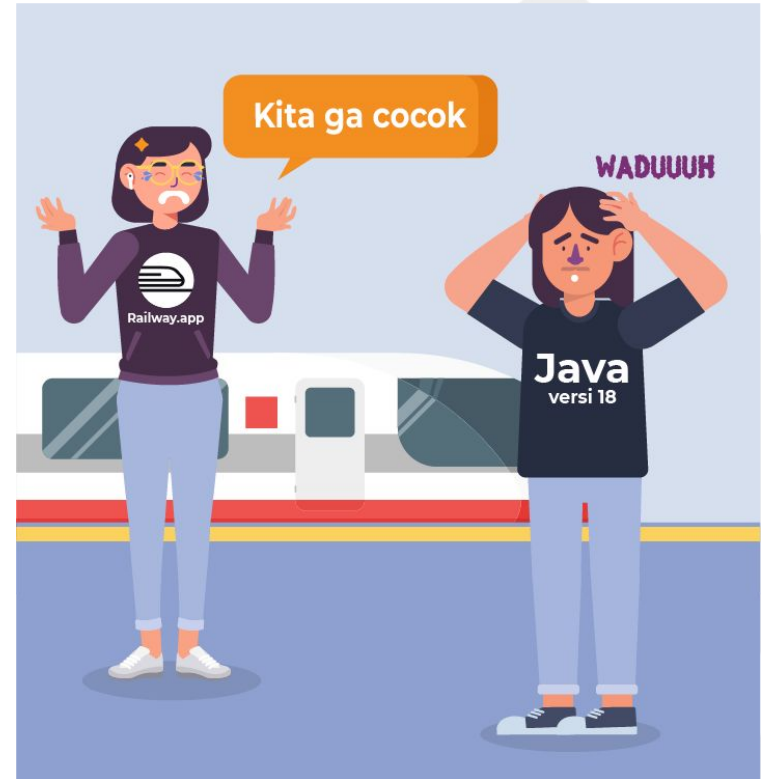
Database
Deploy

Frontend
Deploy

Ada kelebihan ada juga kekurangan si **Railway.app** ☹️

Di balik keunggulannya, ternyata Railway.app ini memiliki kekurangan khususnya untuk penggunaan di Java.

It's unfortunate to mention kalau Railway nggak kompatibel dengan Java versi 18.



Berhubung si Railway ini nggak kompatibel dengan Java versi 18, artinya kamu masih bisa gunakan Railway di versi Java di bawahnya.

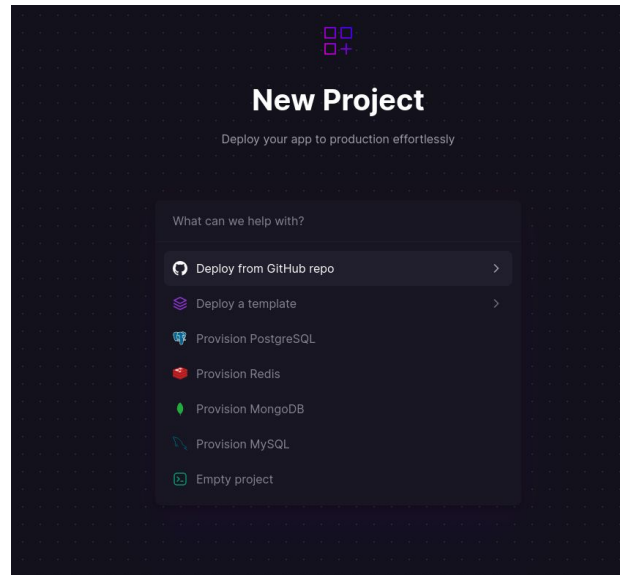
Iya, **Railway masih bisa kita pakai kalau Java yang kamu gunakan adalah Java versi 16.**

So, pastikan kamu menggunakan versi yang ini ya sebelum melakukan deployment.



Deployment On Railway

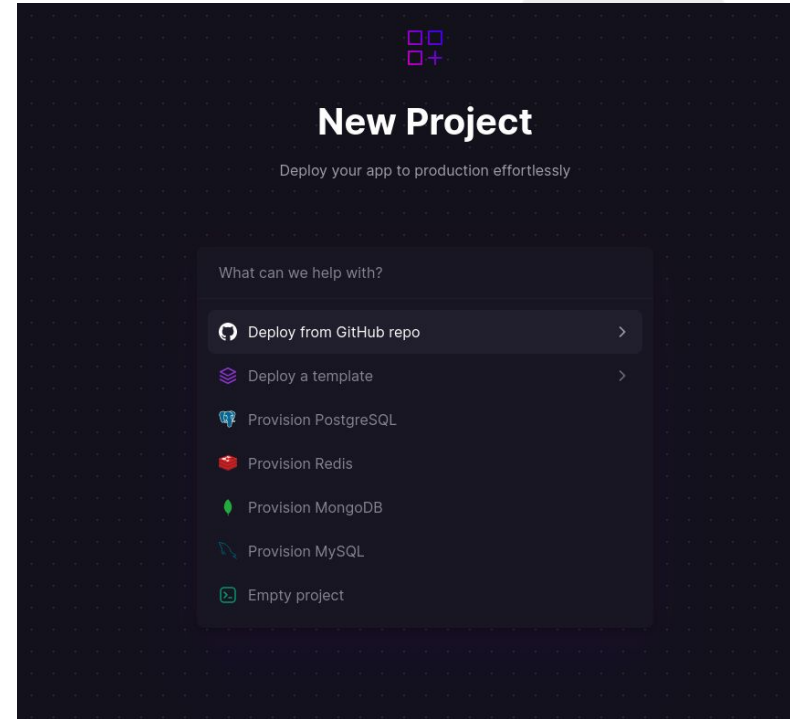
Kita lanjut, ya. Setelah memahami fitur-fitur Railway.app kita masuk ke deployment. Di sini kalo kamu udah register, kamu tinggal masuk ke dashboard. Lalu, kamu bisa bikin aplikasi yang sudah terkoneksi dengan akun Github kamu.



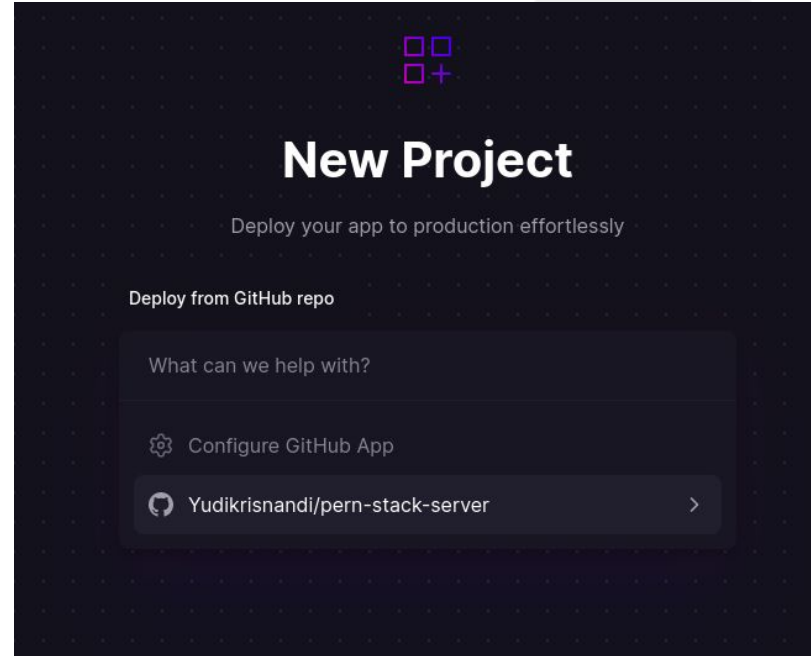
Begini Cara Deploy di Railway

Misalnya kamu deploy melalui Github, kamu tinggal ikuti langkah-langkah dibawah ini aja:

1. Pilih icon **“Deploy from GitHub repo”**

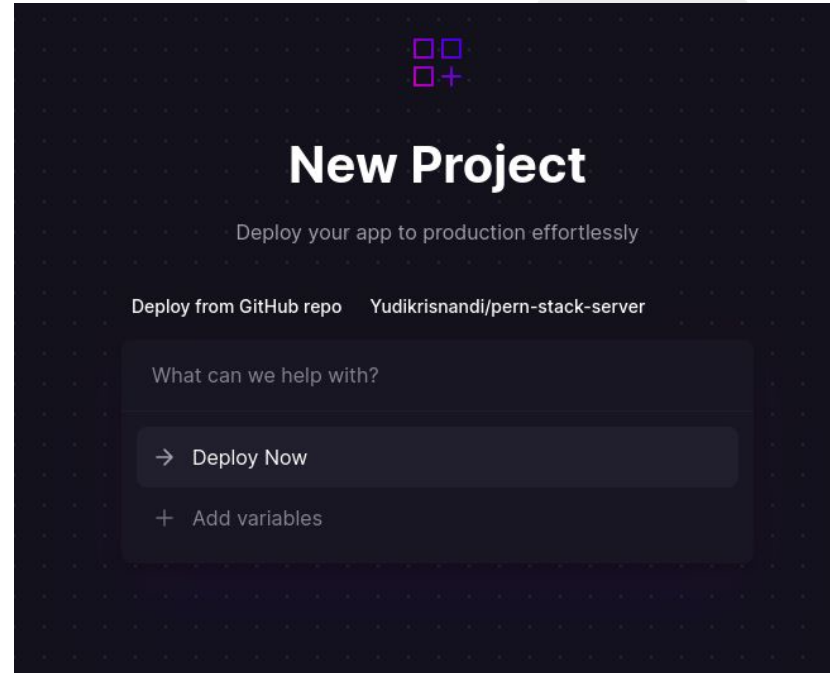


2. Lalu **pilih repo** yang ingin kamu deploy dari GitHub



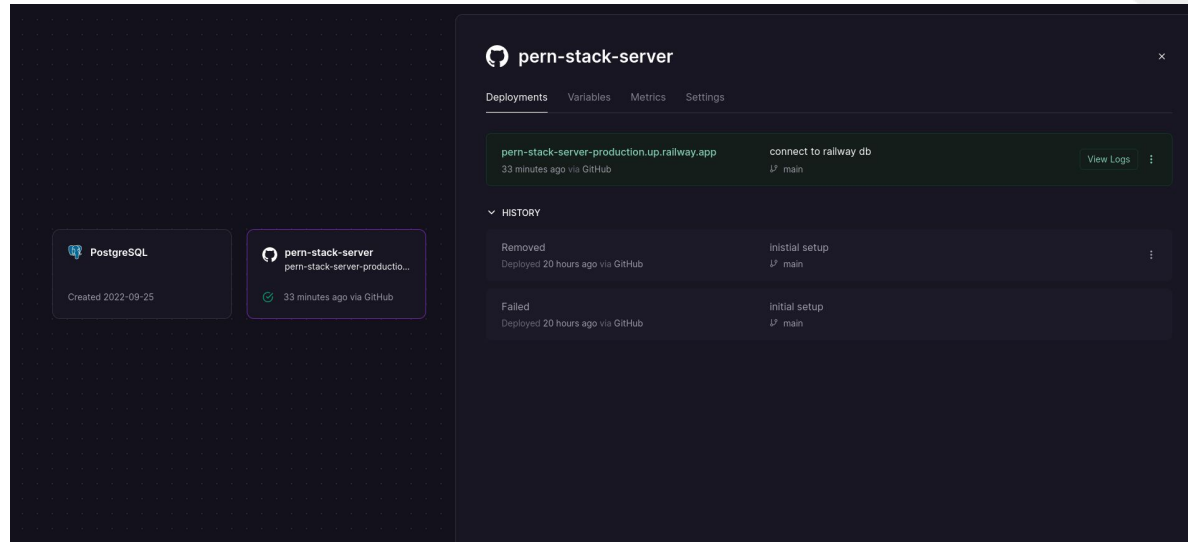
3. Selanjutnya **pilih Deploy**.

Terus udah deh jengjeng!! aplikasimu udah ter-deploy di Railway.app!



Railway dashboard

Terakhir, setelah melakukan deployment tadi maka di dashboard kamu akan muncul list aplikasi yang telah di-deploy sebelumnya. Untuk managing dan monitoring aplikasi tersebut, bisa kamu lakukan dari dashboard ini.



Eits.. tapi ada catetan kecil nih buat kamu.

Selain fokus sama deployment, kita perlu memperhatikan aplikasi-aplikasi lain yang akan menjadi dependensi dari aplikasi kamu, contohnya database.

Nah, untuk kebutuhan database, Railway.app juga sudah menyediakan banyak database seperti yang sudah dibahas sebelumnya. Untuk setup hal tersebut kamu bisa ikuti [tutorial ini](#).



Kita sudah bisa **deployment dengan Railway.app** nih!
Mantap ☐☐

Walau termasuk cloud platform yang baru, tapi
Railway.app ini merupakan opsi yang menjanjikan untuk
kita-kita yang masih dalam tahap belajar melakukan
deployment.

Coba sini share sama Sabrina, gimana pengalaman kamu
melakukan deployment dengan Railway.app? Apakah
prosesnya berjalan lancar?



Nah, selesai sudah pembahasan kita di topik ini ☐

Selanjutnya, kita bakal siap-siap untuk belajar **CI/CD** alias langkah langkah praktis yang akan banyak membantu kamu.

Penasaran? Yuk lanjut di Topik 6~

