**LAPORAN PRAKTIKUM**

**TEKNOLOGI CLOUD**

**PERTEMUAN KE – 14**

**Disusun Oleh :**

**NAMA : TARISA DWI SEPTIA**

**NIM : 205410126**

**JURUSAN : TEKNIK INFORMATIKA**

**JENJANG : S1**

**UNIVERSITAS TEKNOLOGI DIGITAL INDONSIA**

**YOGYAKARTA**

**2021**

**Deploy, Scale, and Update Your Website on Google Kubernetes Engine**

1. **Tujuan**

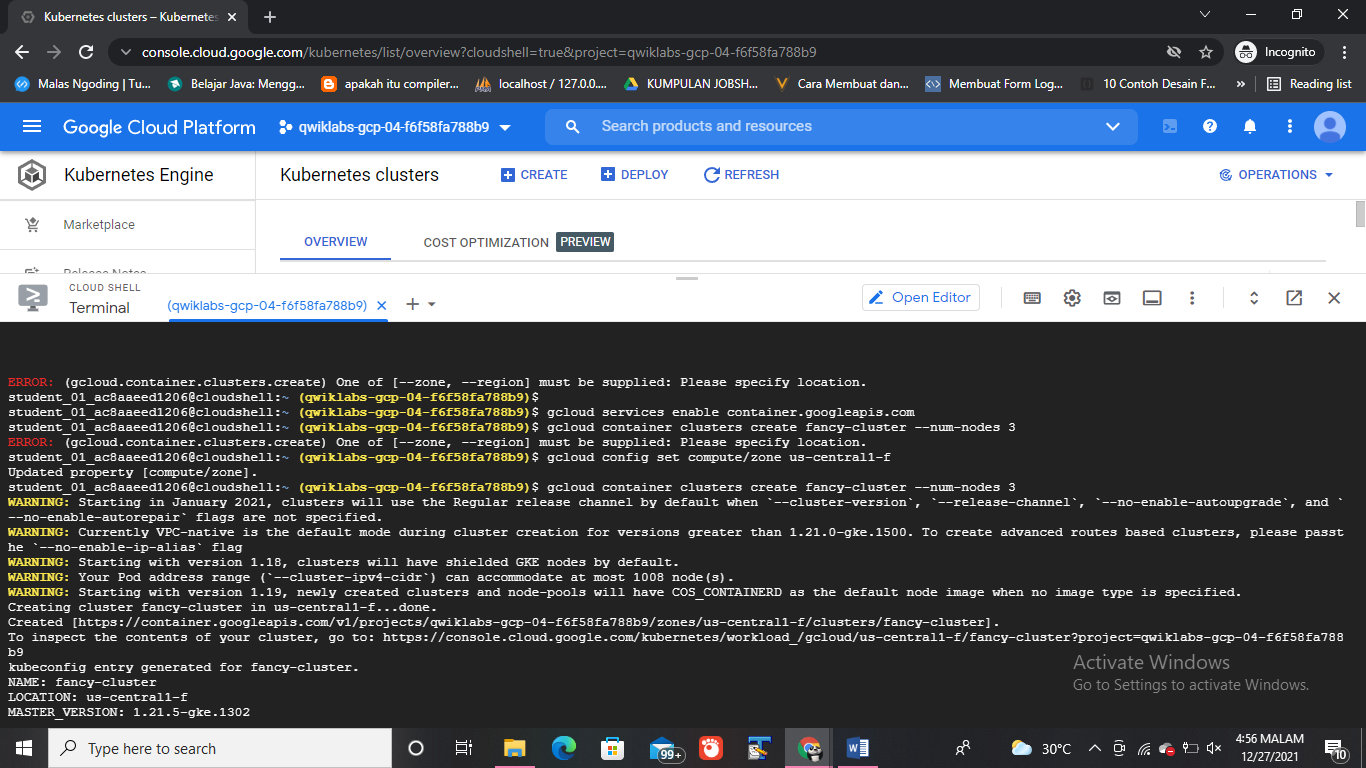
* Mahasiswa dapat membuat Google Kubernetes Engine cluster
* Mahasiswa dapat membuat docker image
* Mahasiswa dapat mendeploy docker image ke kubernetas
* Mahasiswa dapat scale an application on Kubernetes
* Mahasiswa dapat melakukan pembaruan bergulir di Kubernetas

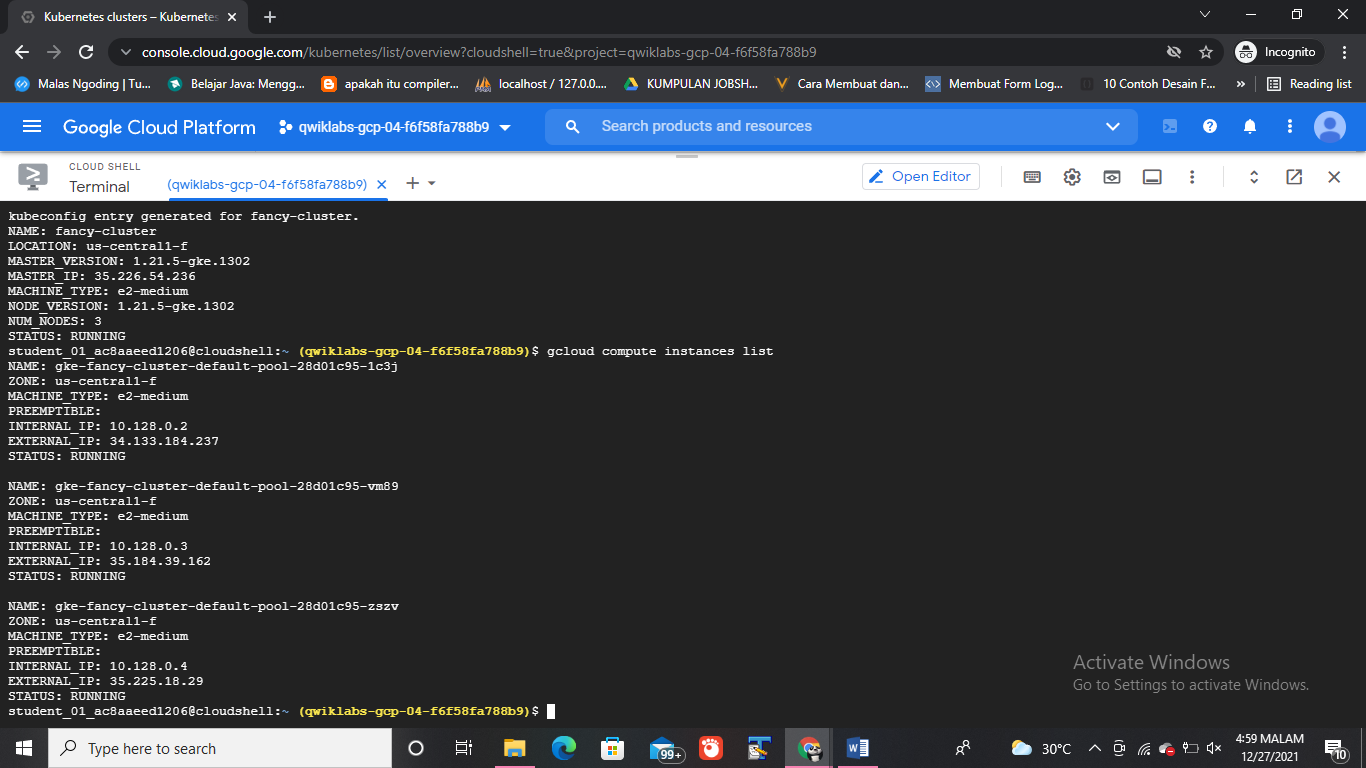
1. **Praktik**
2. Create a GKE cluster

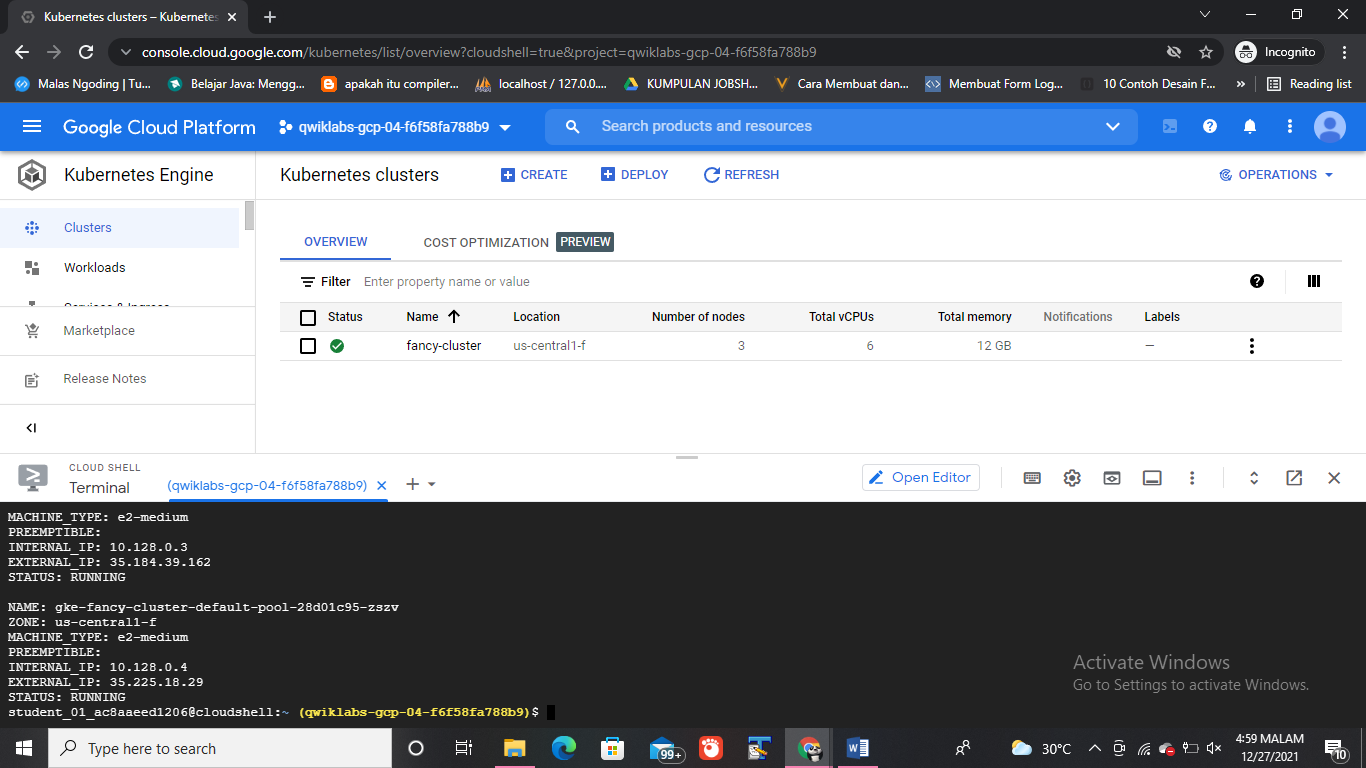
* Anda memerlukan cluster Kubernetes untuk men-deploy situs web Anda. Pertama, pastikan API yang tepat diaktifkan.

Sekarang anda sudah mempunyai akun.

* Jalankan perintah berikut untuk membuat cluster GKE bernama fancy-cluster dengan 3 node:

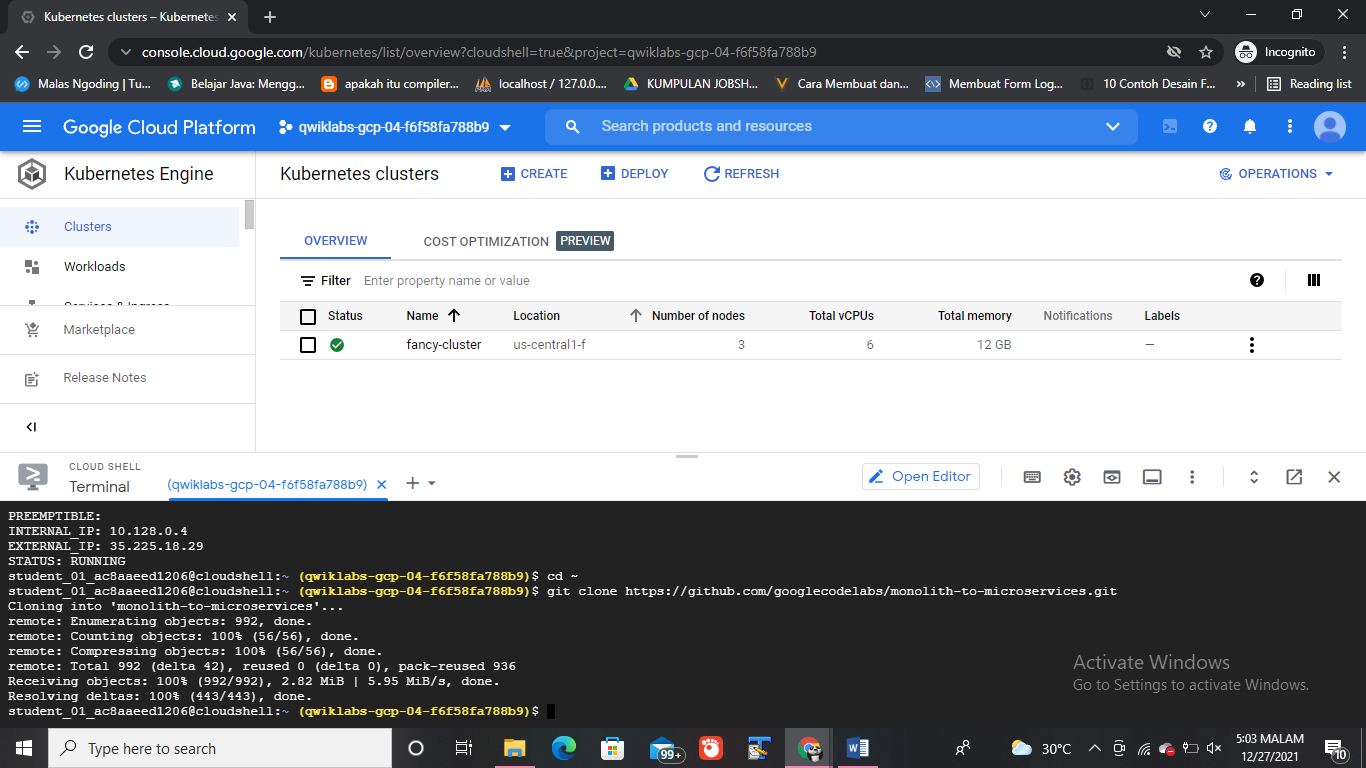
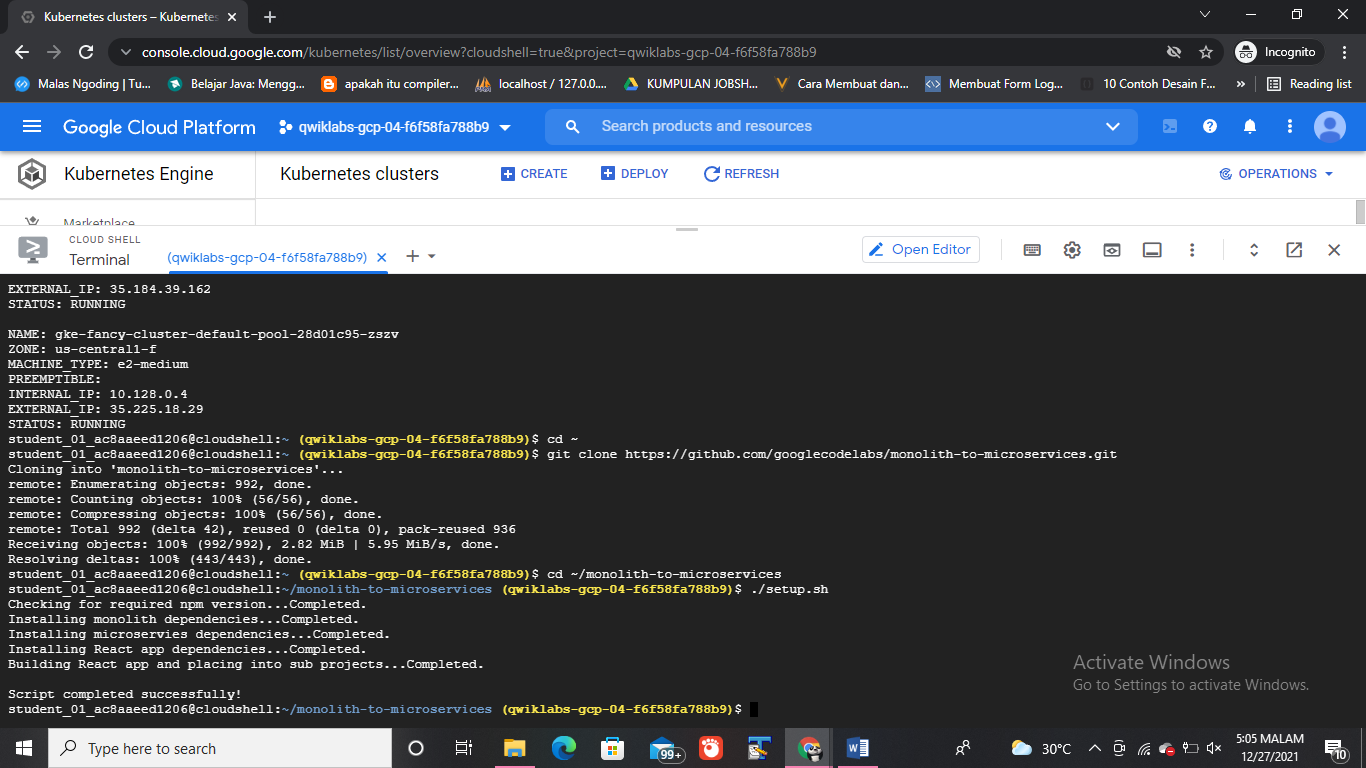
Jika Anda mendapatkan kesalahan tentang region/zona yang tidak ditentukan, silakan lihat bagian pengaturan lingkungan untuk memastikan Anda menyetel zona komputasi default.

* Sekarang jalankan perintah berikut dan lihat tiga instance VM pekerja cluster:

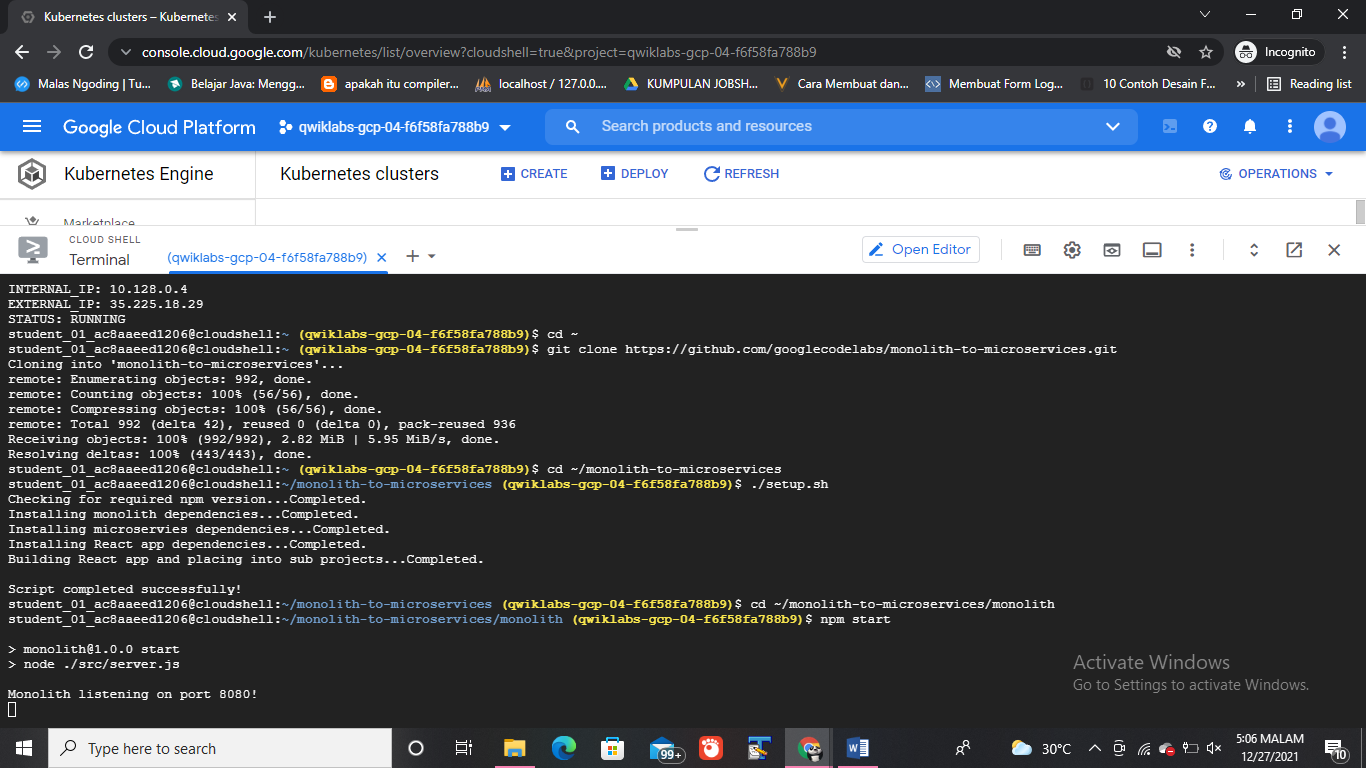
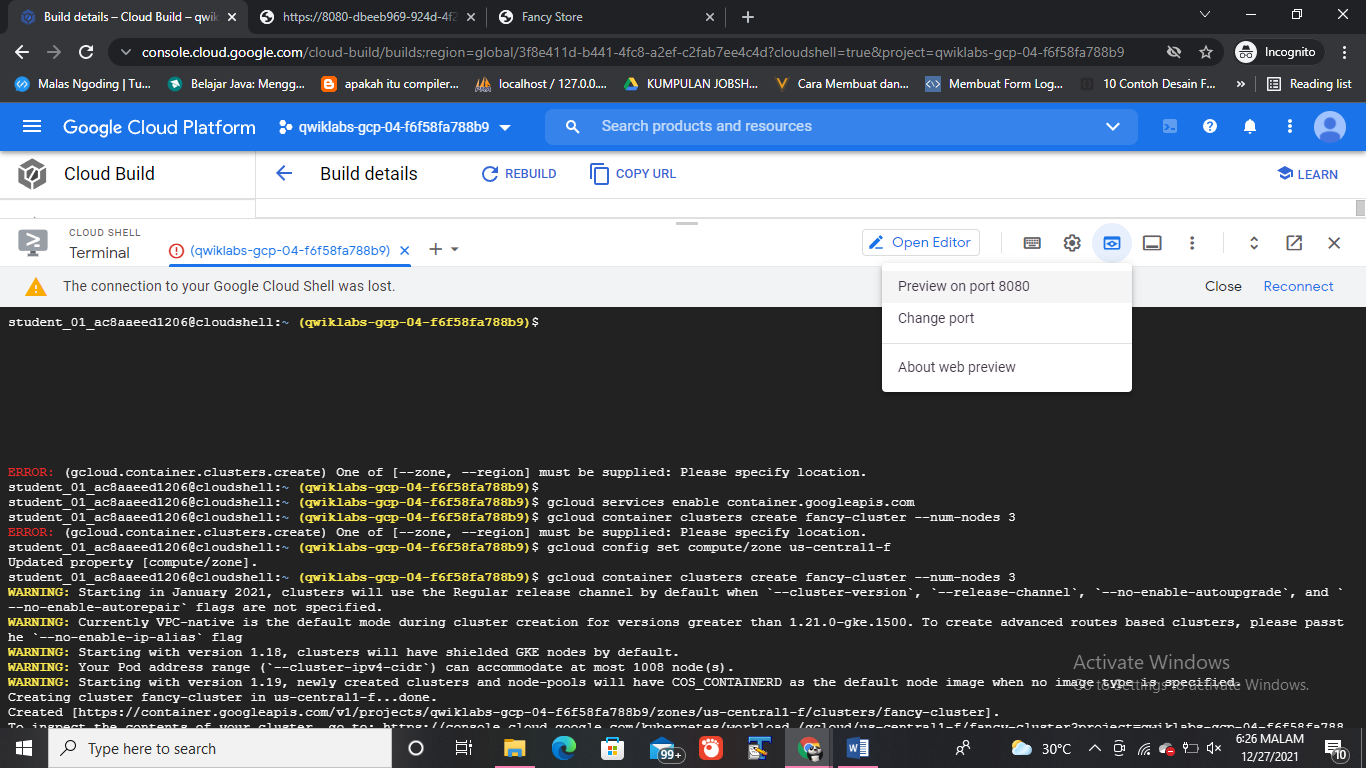
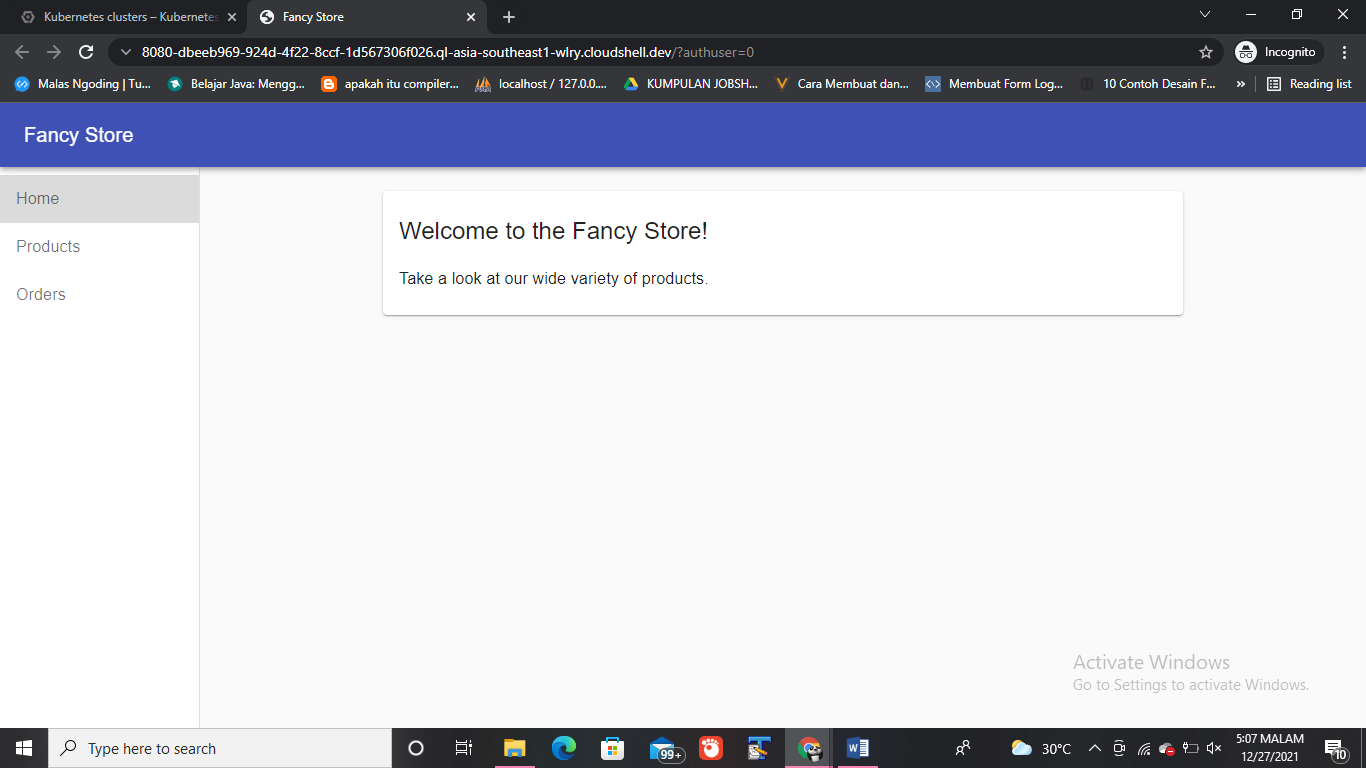
* Temukan cluster Kubernetes Anda dan informasi terkait di Google Cloud Console. Klik menu **Navigasi**, lalu gulir ke bawah ke **Kubernetes Engine** dan klik **Cluster**.

Anda baru saja membuat cluster Kubernetes pertama Anda!

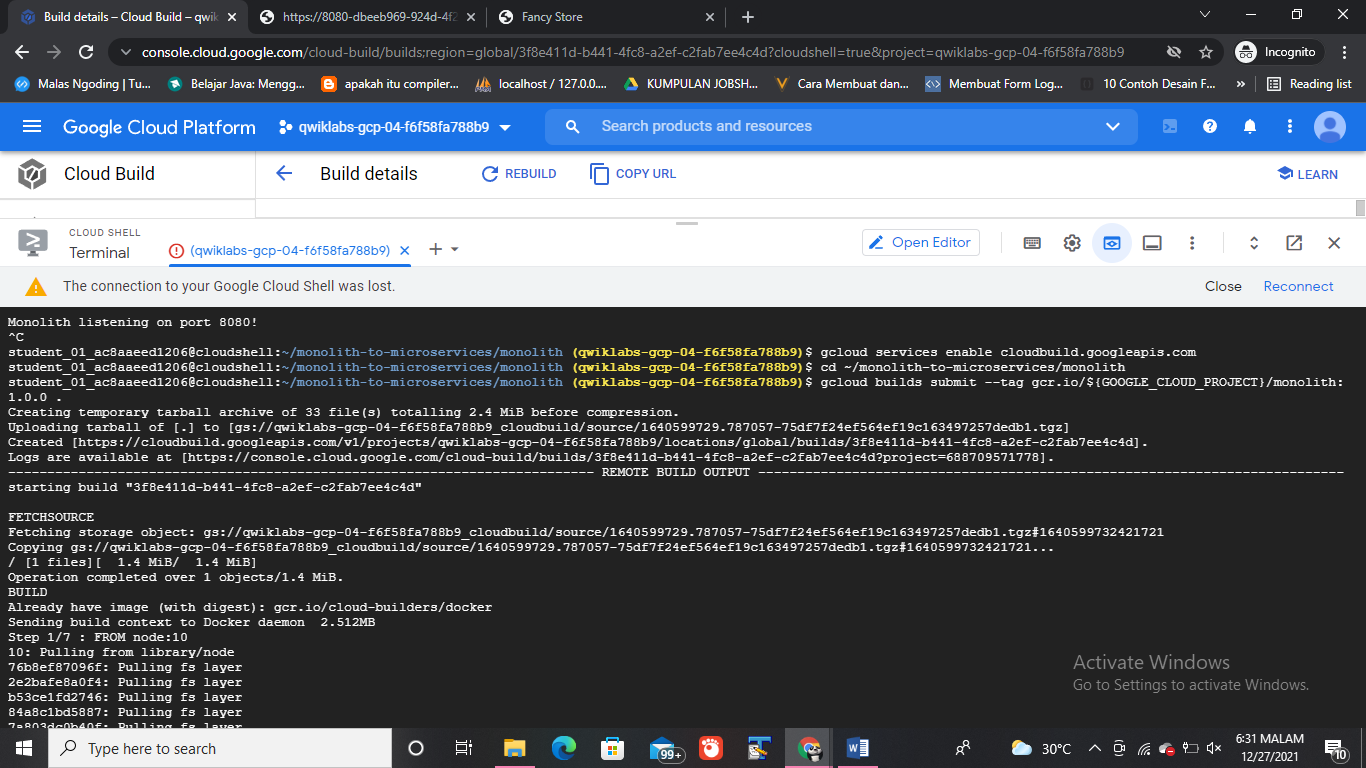
1. Clone source repository

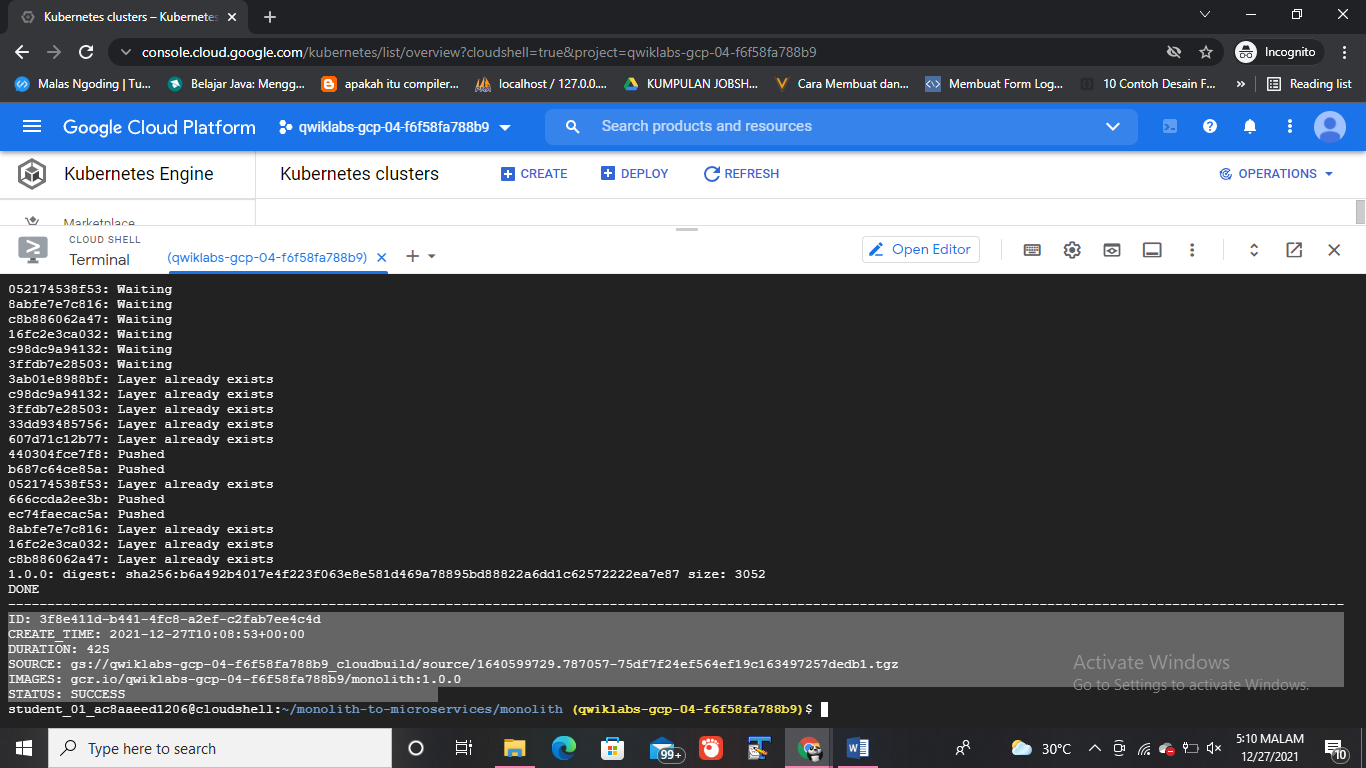
* Jalankan perintah berikut untuk mengkloning git repo ke instance Cloud Shell Anda:
* Ubah ke direktori yang sesuai. Anda juga akan menginstal dependensi NodeJS sehingga Anda dapat menguji aplikasi Anda sebelum menerapkan:

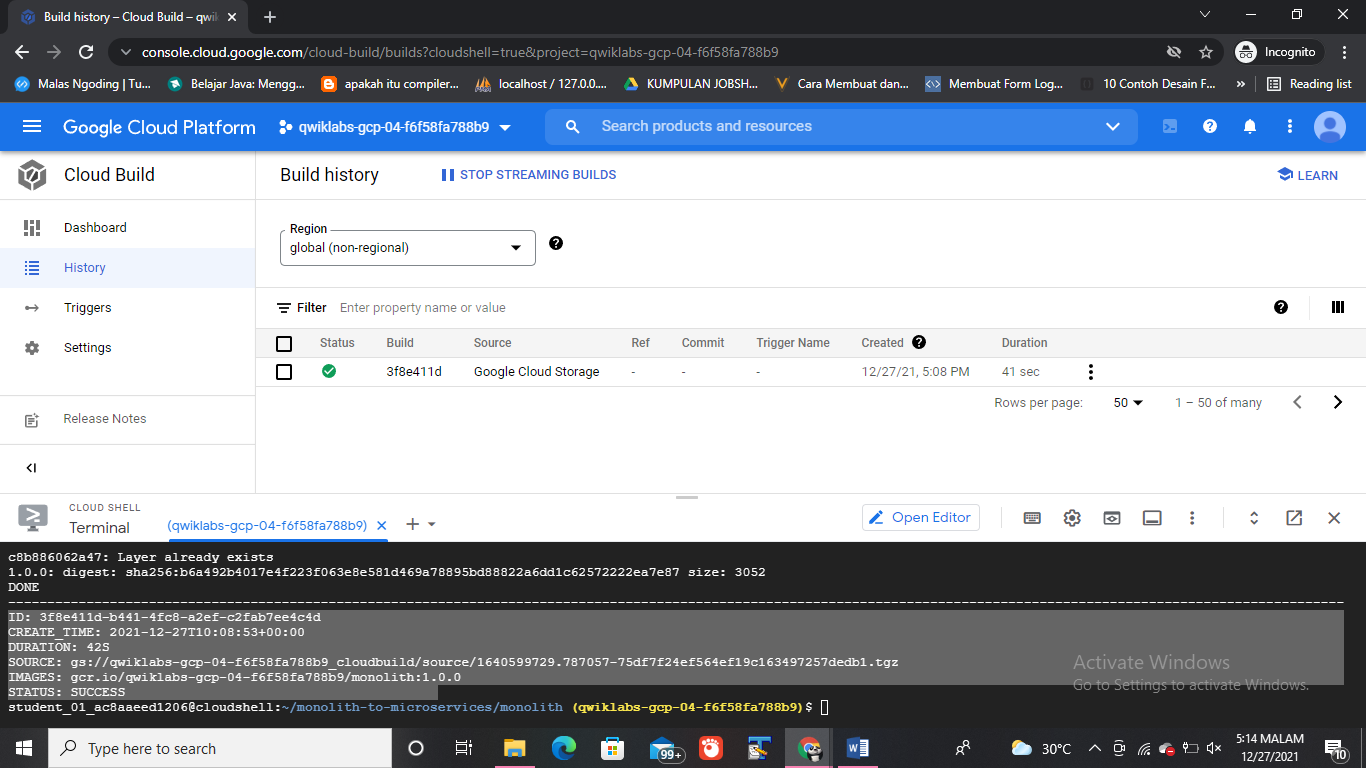
Tunggu beberapa menit hingga script selesai running

* Ubah ke direktori yang sesuai dan uji aplikasi dengan menjalankan perintah berikut untuk memulai server web:
* Kamu dapat melihat preview website dengan mengklik **Preview on port 8080**:
* Ini akan membuka jendela baru di mana Anda dapat melihat Toko Mewah kami beraksi!

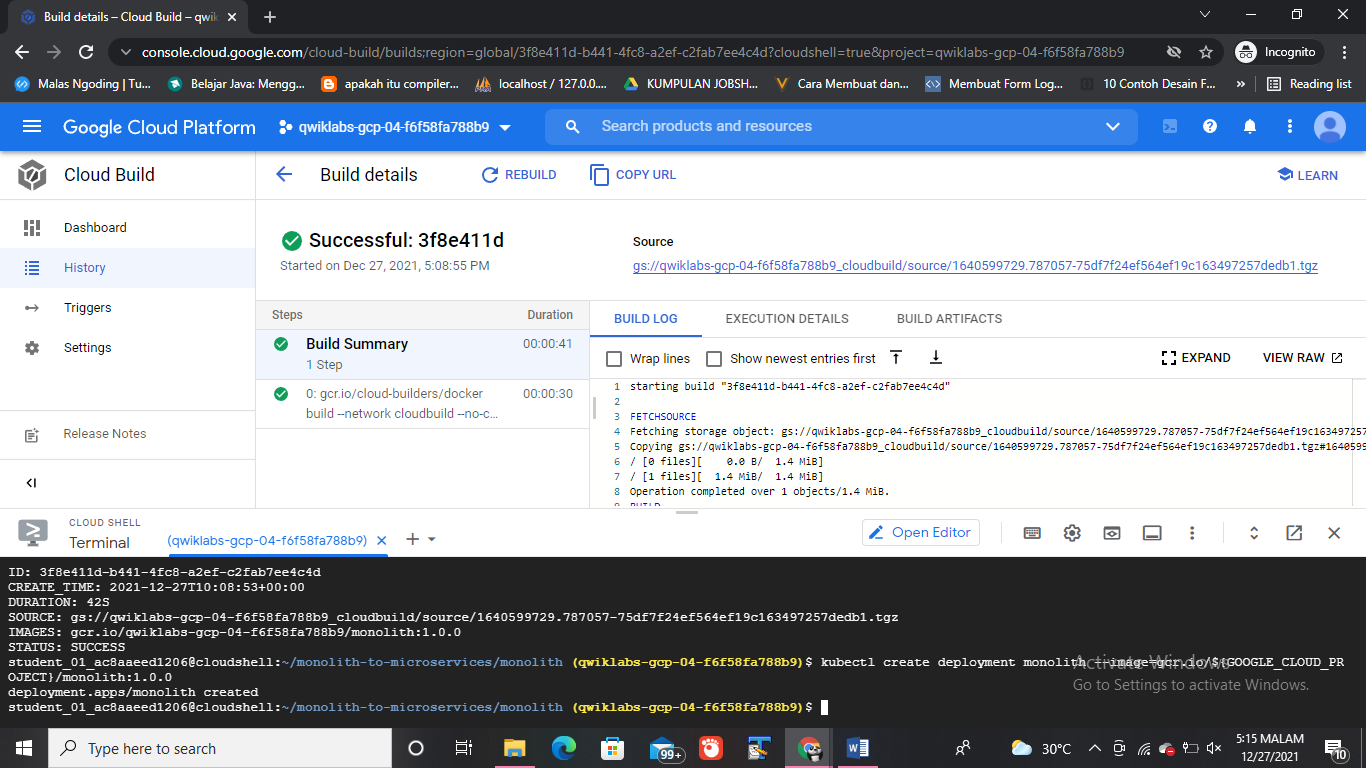
1. Create Docker container with Cloud Build

* Pertama, untuk memastikan Anda mengaktifkan Cloud Build API, jalankan perintah berikut:

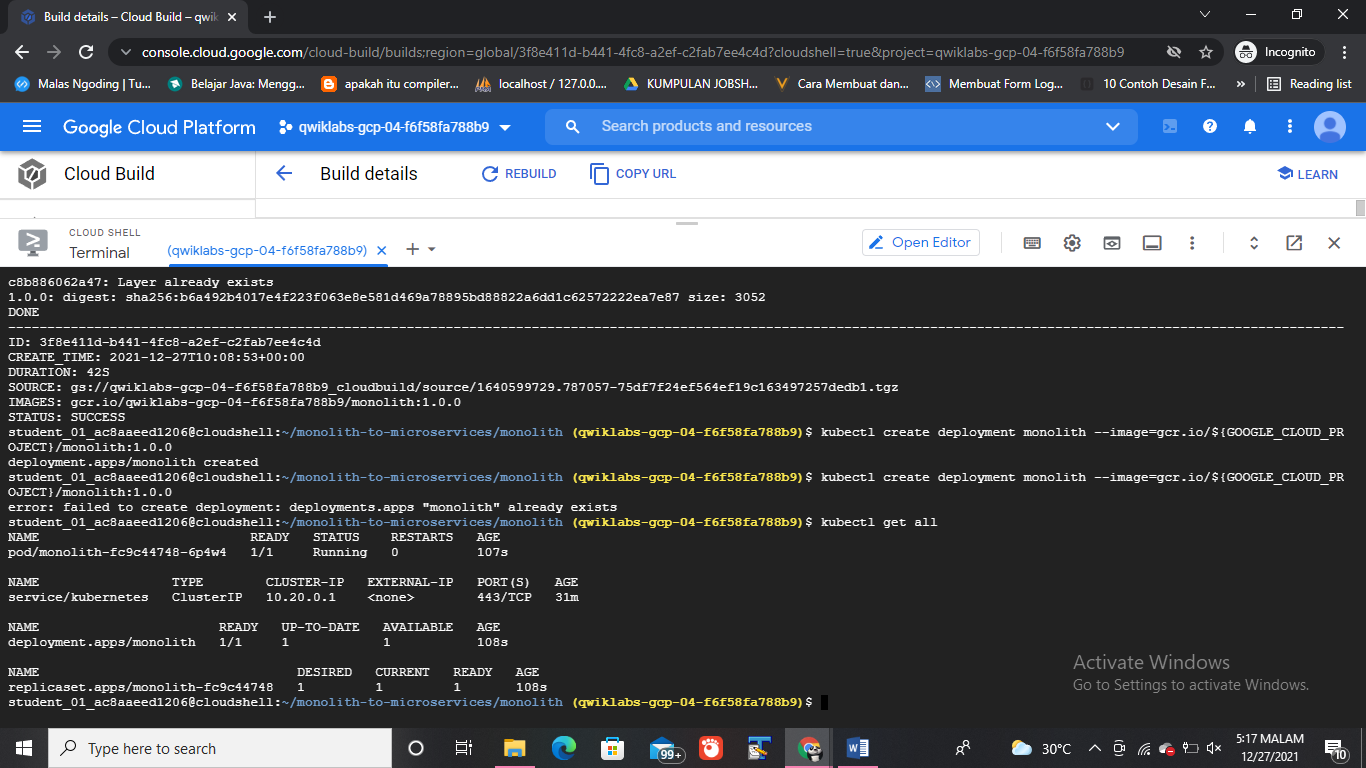
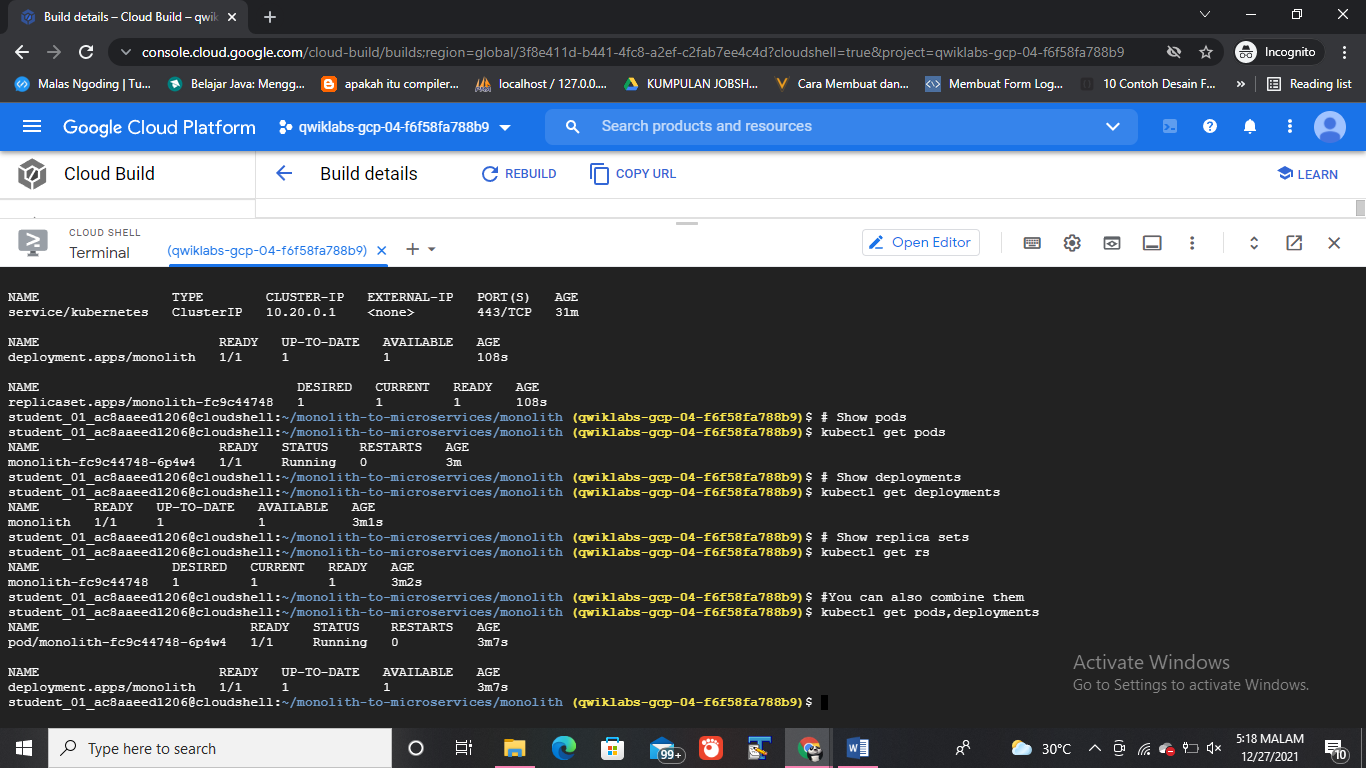
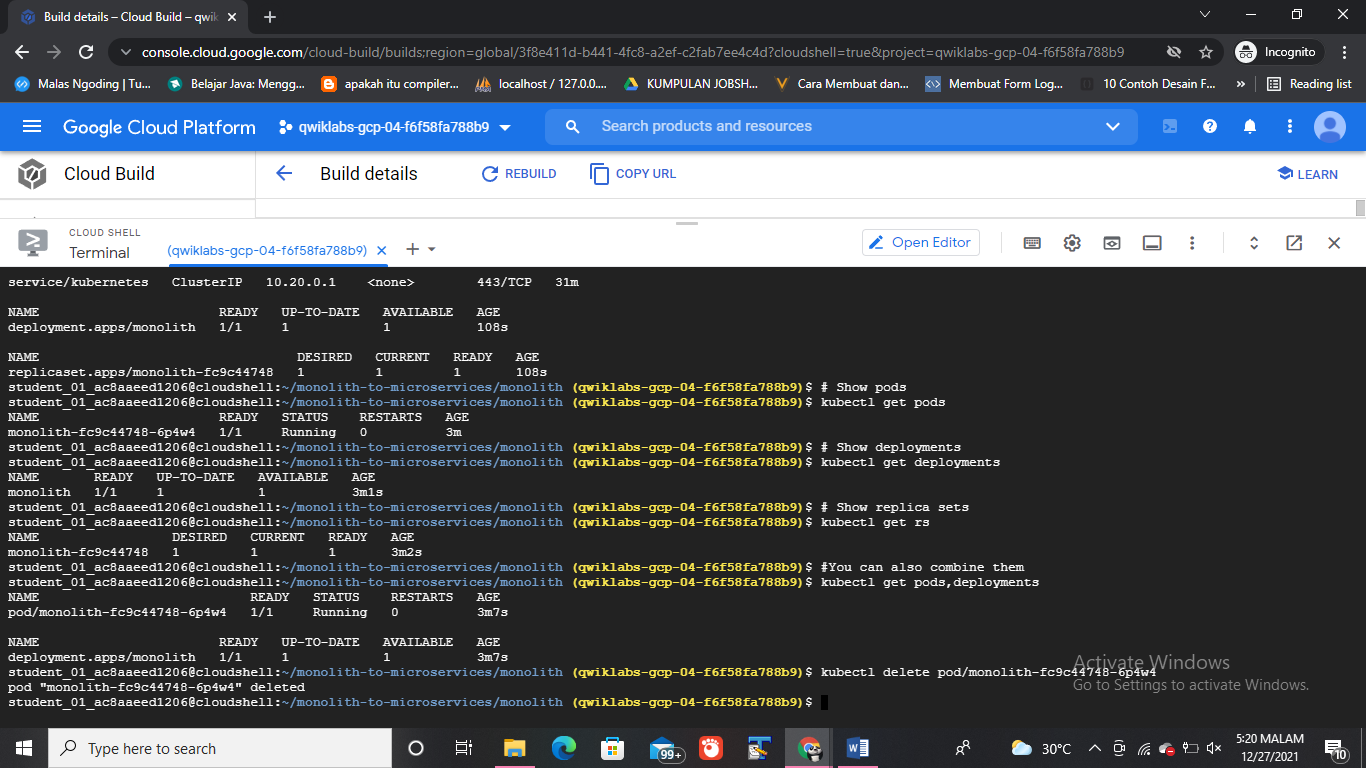
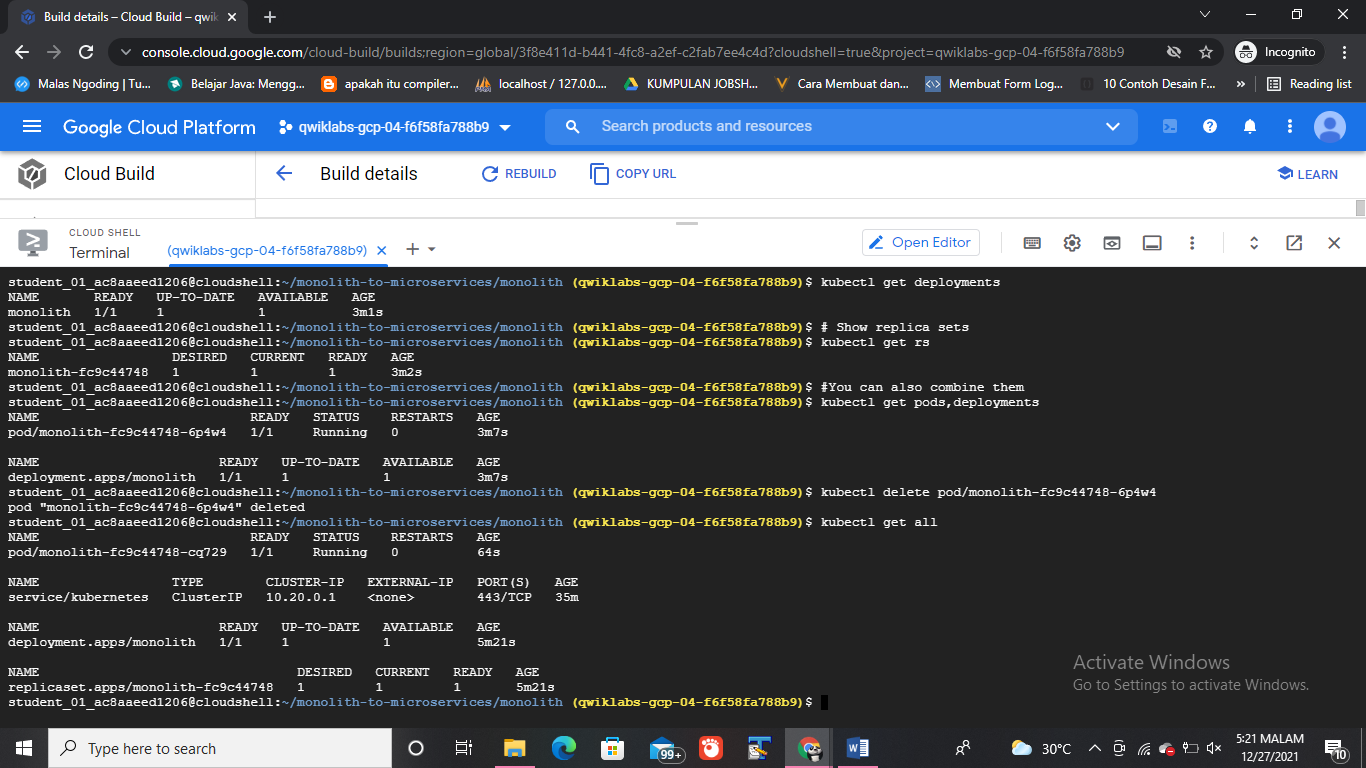
Output :

* Untuk melihat riwayat build Anda atau melihat prosesnya secara real time dengan mengeklik menu Navigasi dan menggulir ke bawah ke bagian Alat, lalu klik Cloud Build > History. Di sini Anda dapat melihat daftar semua bangunan Anda sebelumnya.

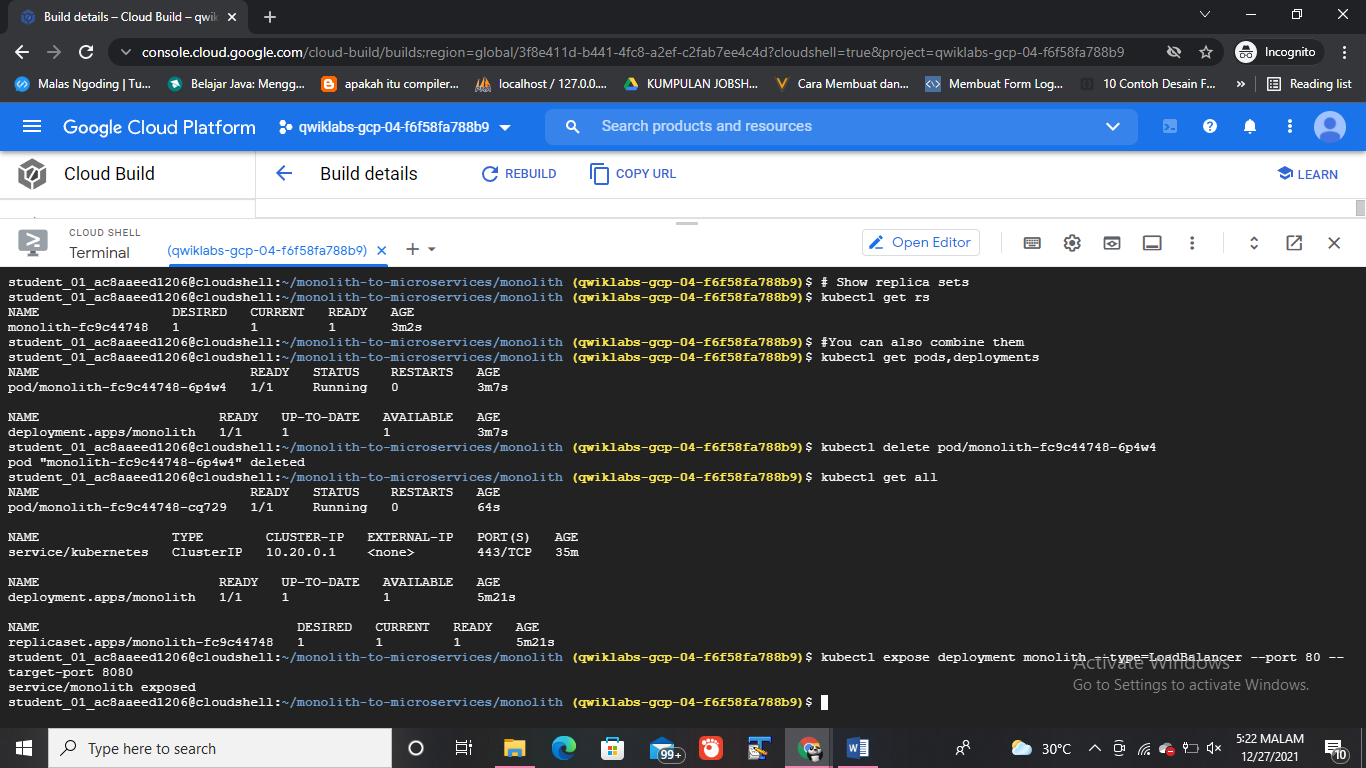
1. Deploy container to GKE

* Jalankan perintah berikut untuk menerapkan aplikasi Anda:

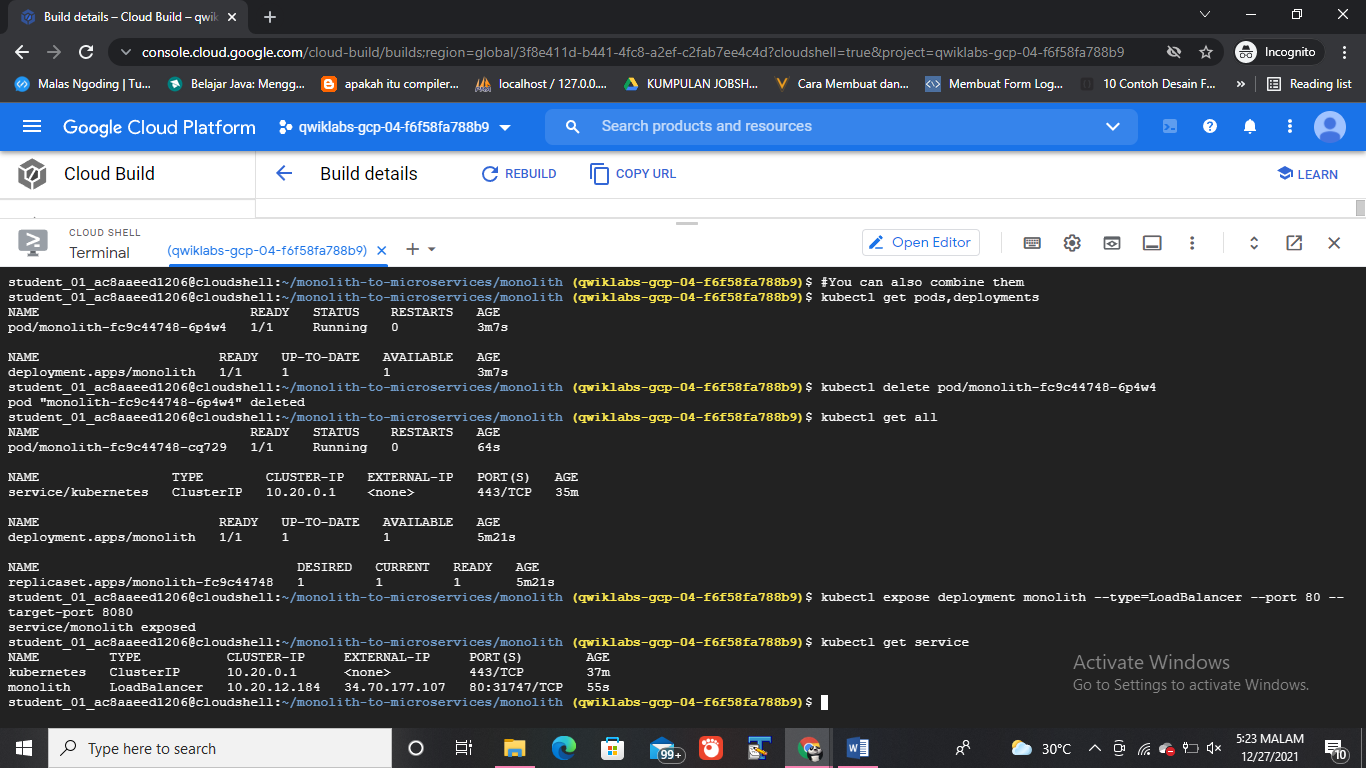
1. Verify Deployment

* Pastikan Deployment berhasil dibuat:
* Anda juga dapat menjalankan perintah ke penerapan Anda secara terpisah:
* Salin nama pod dari perintah sebelumnya, lalu gunakan saat Anda menjalankan perintah berikut untuk menghapusnya:
* Jika Anda cukup cepat, Anda dapat menjalankan get all lagi, dan Anda akan melihat dua pod: satu berhenti dan yang lainnya membuat atau menjalankan:

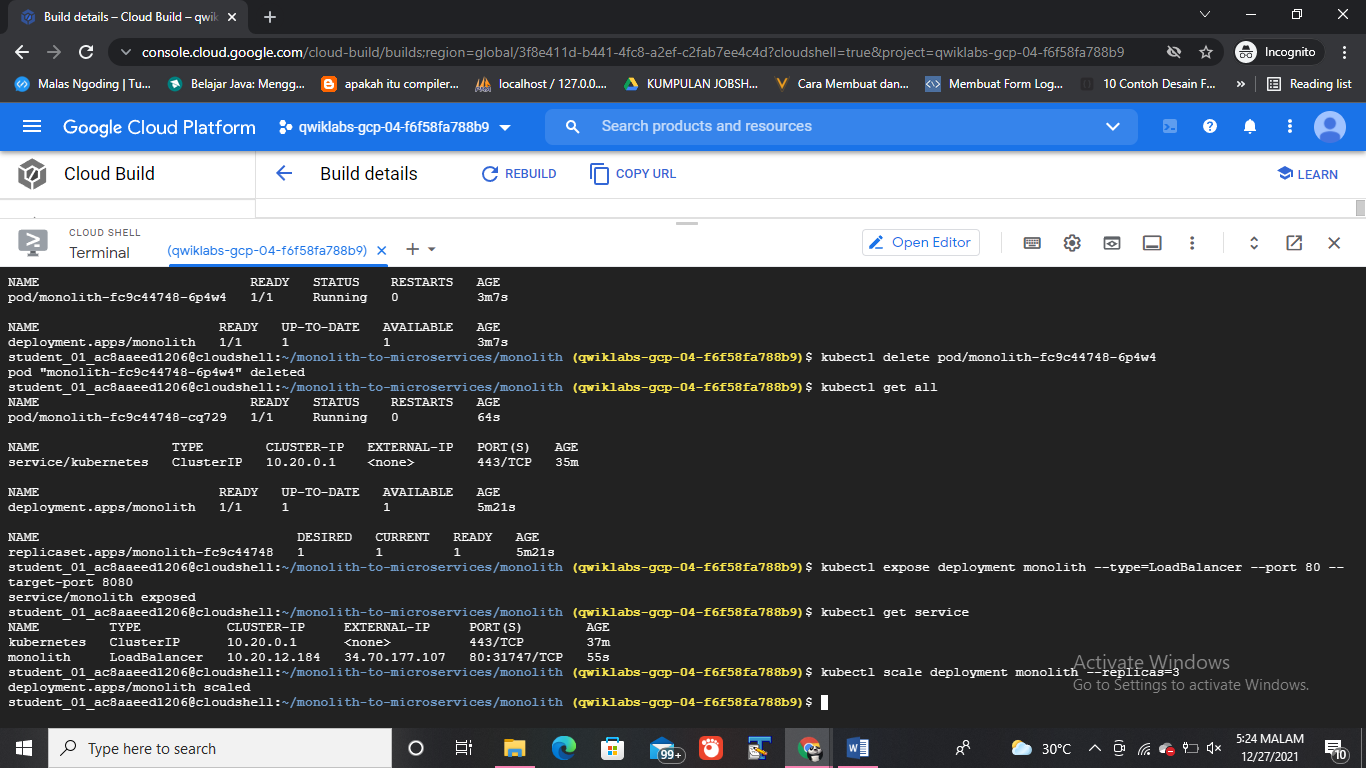
1. Expose GKE Deployment

* Jalankan perintah berikut untuk mengekspos situs web Anda ke Internet:

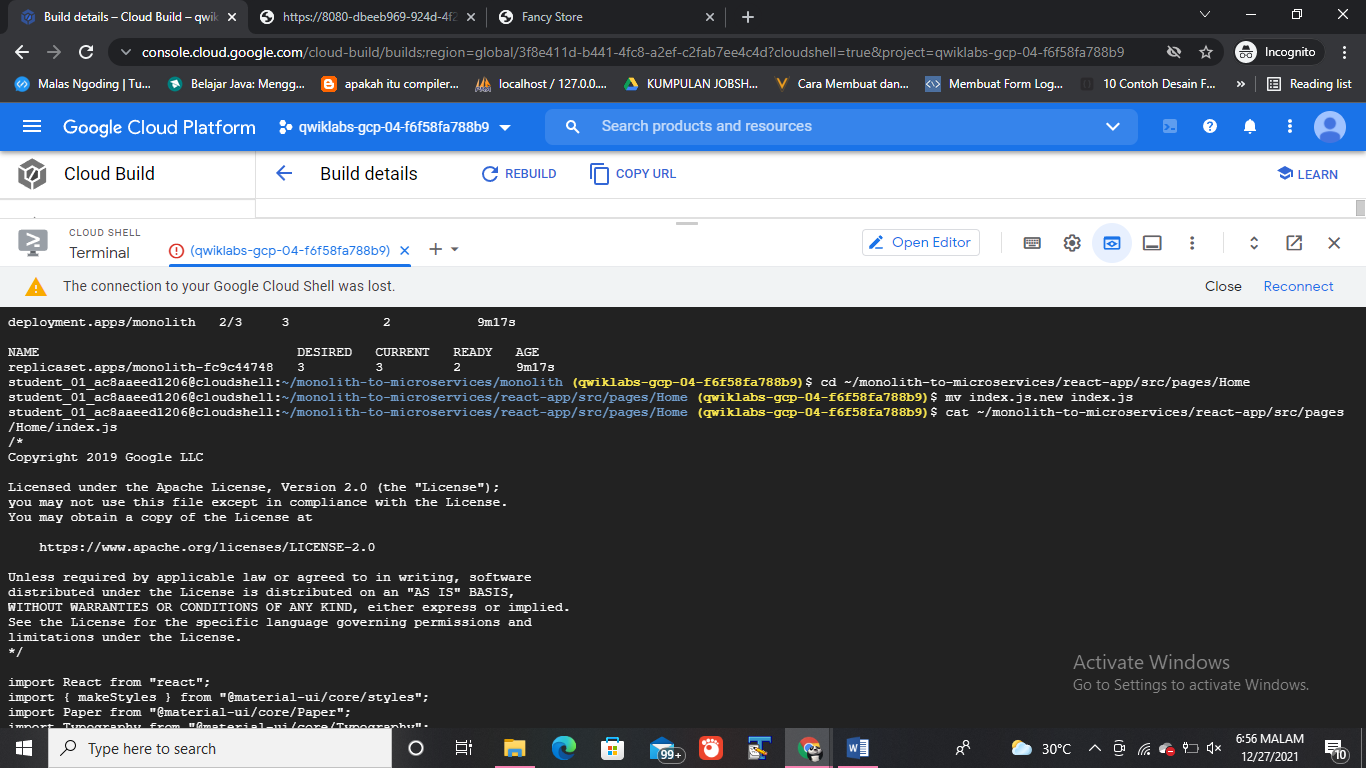
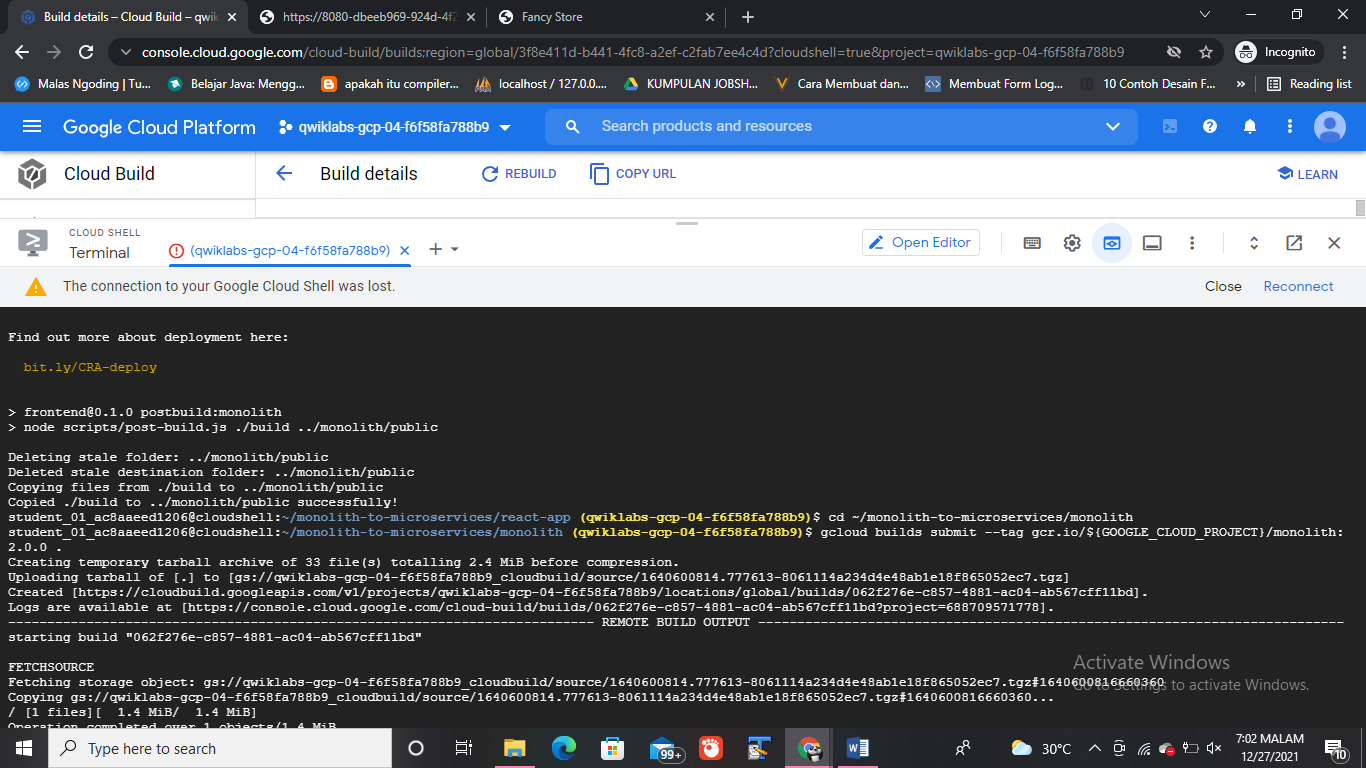
1. Accessing the service

* GKE menetapkan alamat IP eksternal ke resource Layanan, bukan Deployment. Jika Anda ingin mengetahui IP eksternal yang disediakan GKE untuk aplikasi Anda, Anda dapat memeriksa Service dengan perintah kubectl get service:

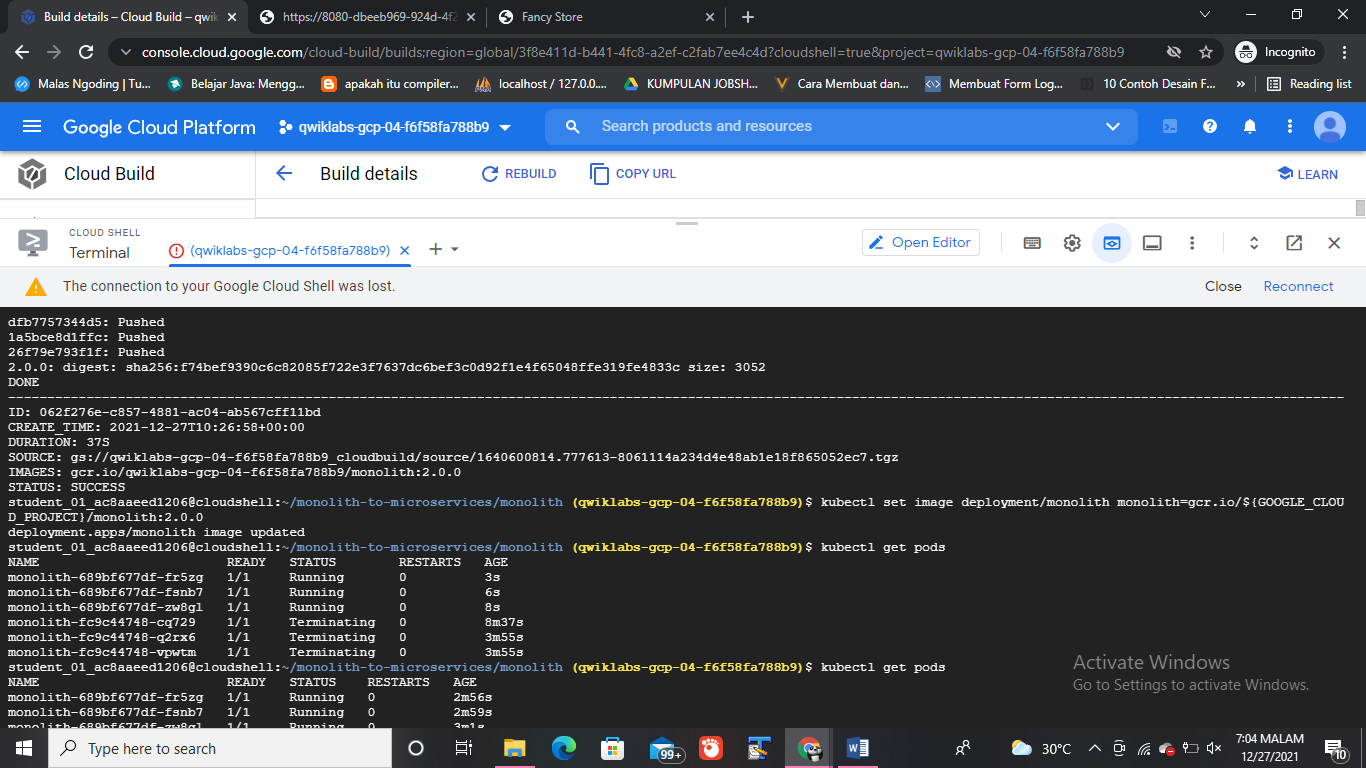
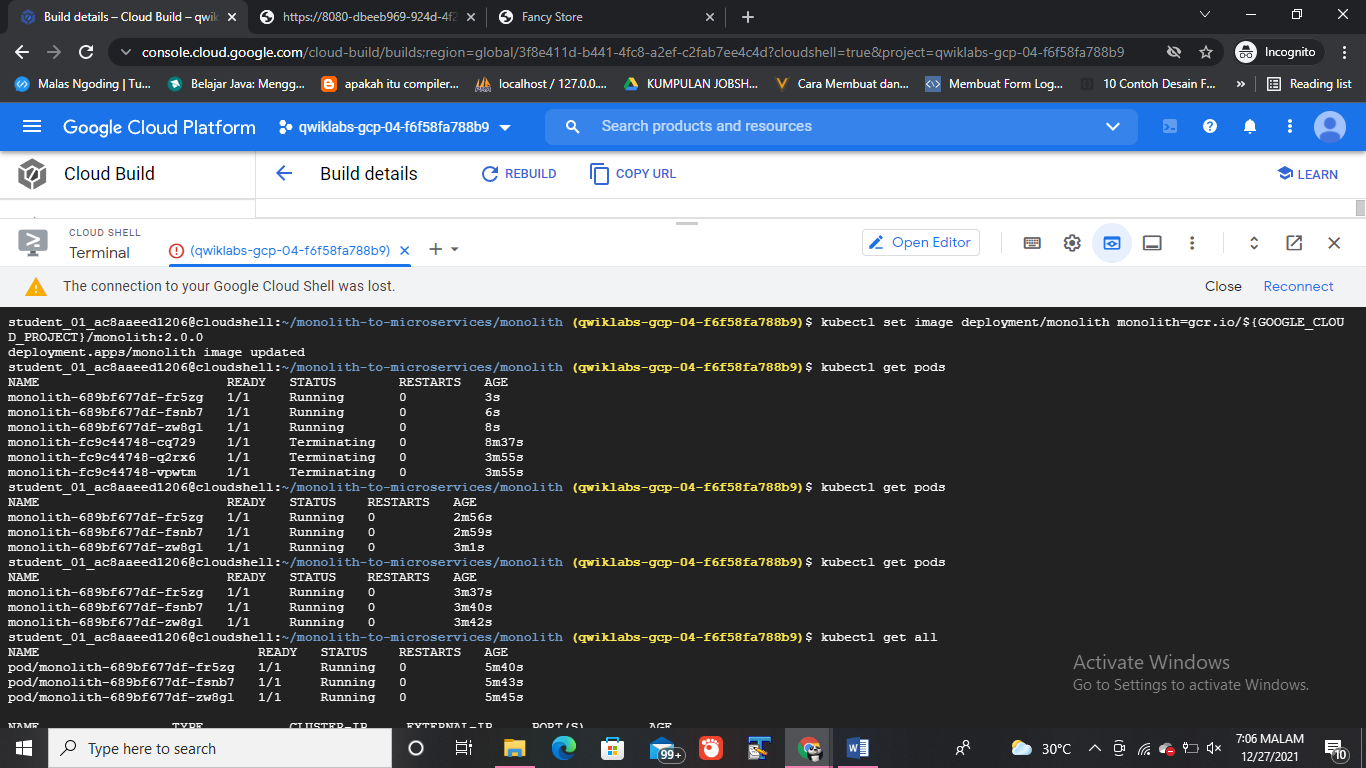
1. Scale GKE deployment

* Di Cloud Shell, jalankan perintah berikut untuk menskalakan penerapan Anda hingga 3 replika:
* Pastikan Deployment berhasil diskalakan:

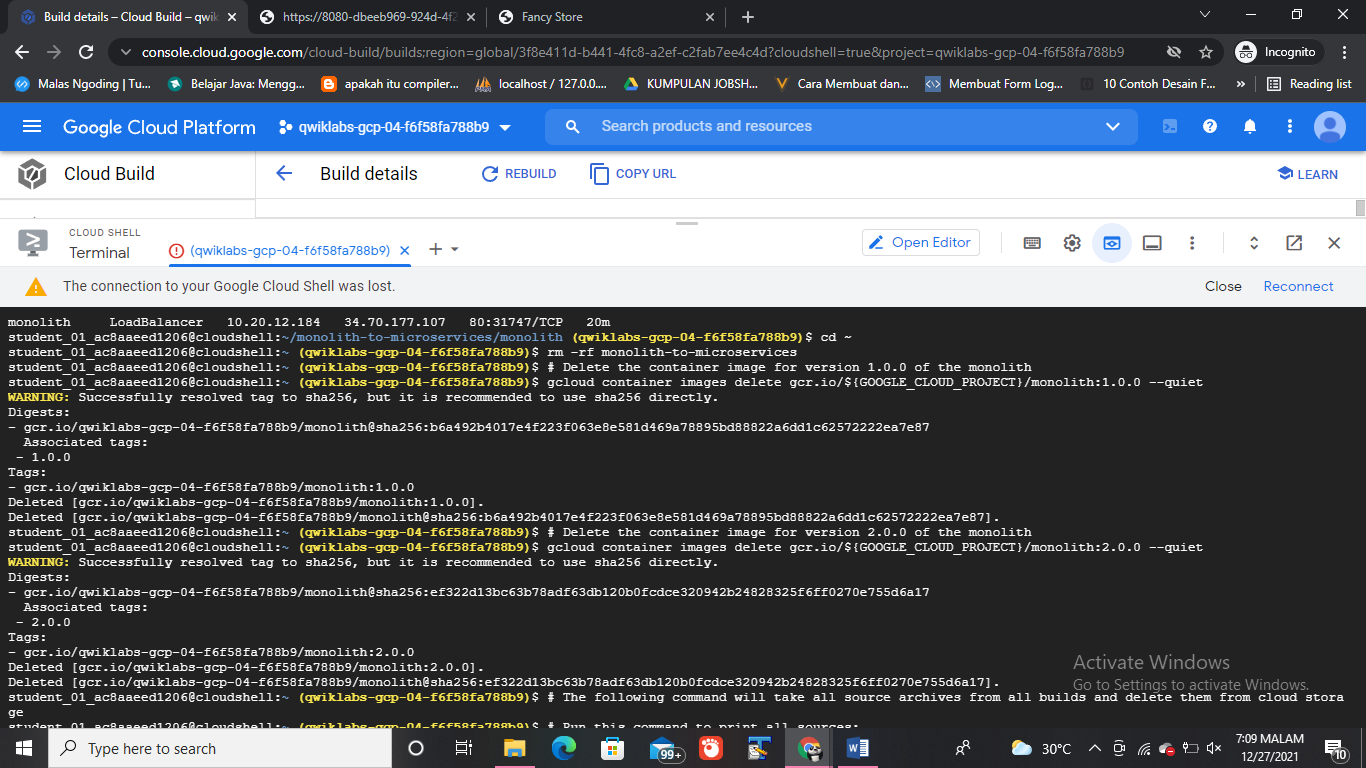
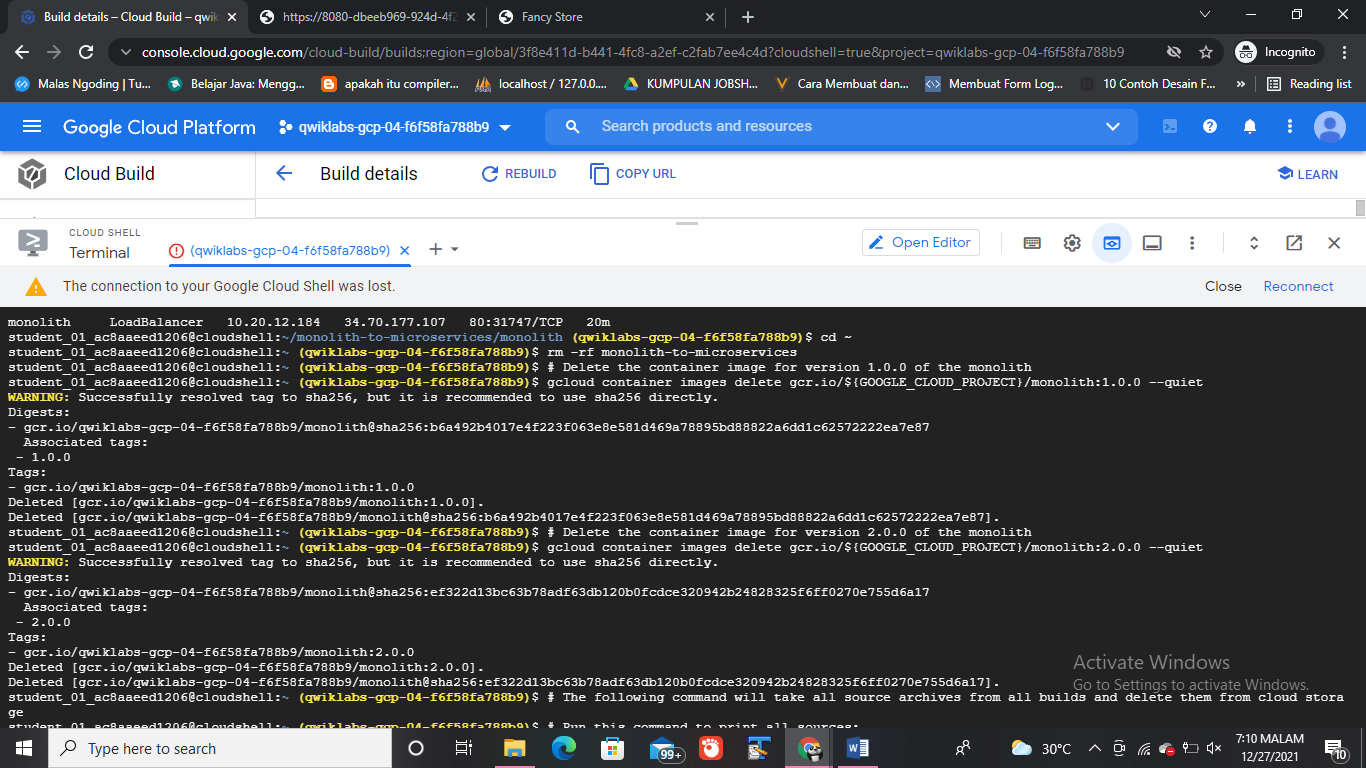
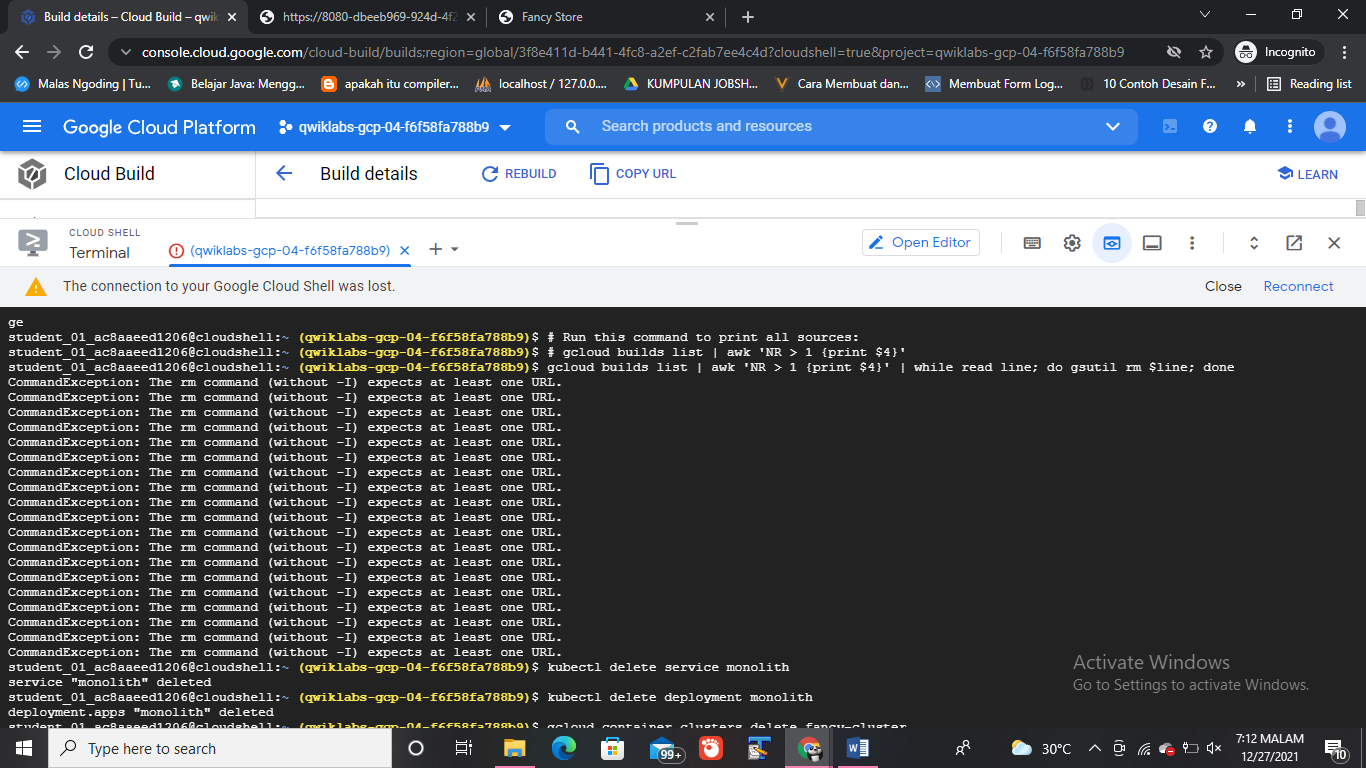
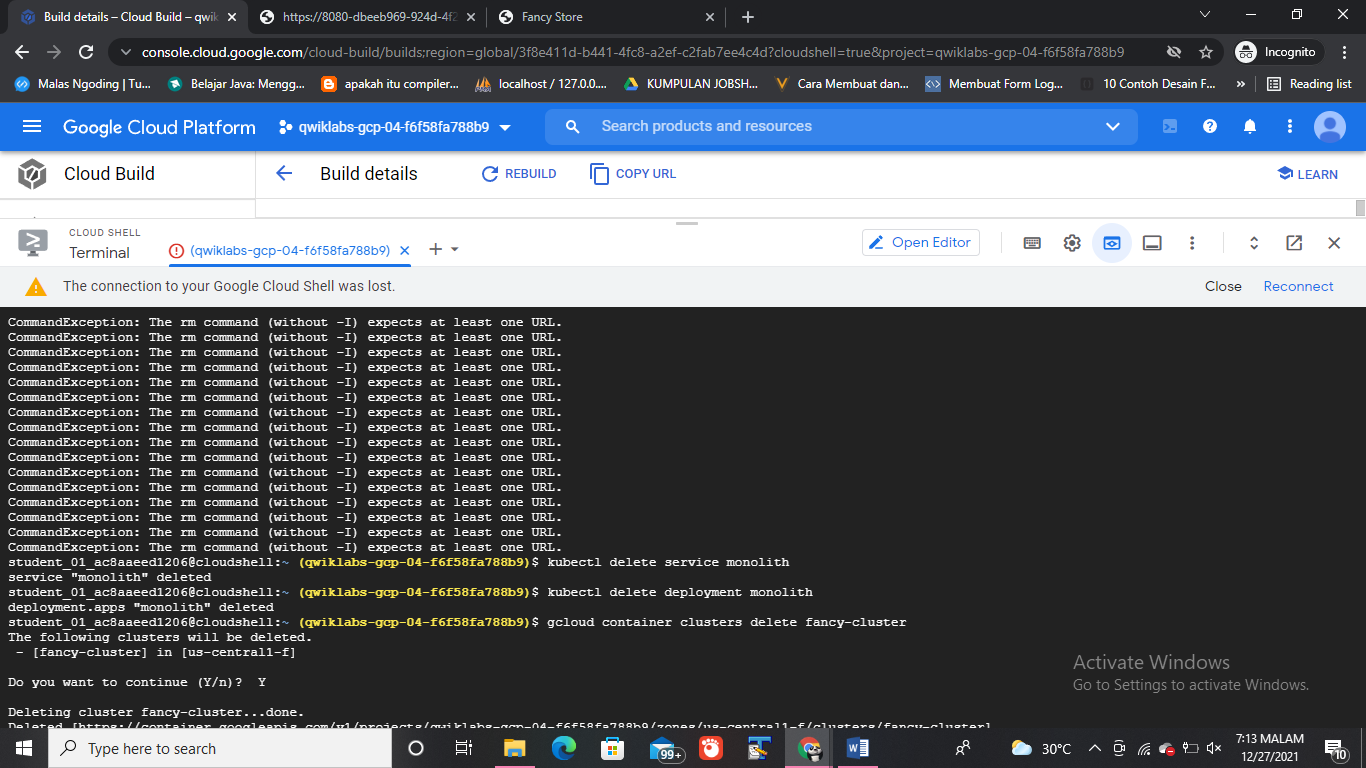
1. Make changes to the website

* Jalankan perintah berikut salin file yang diperbarui ke nama file yang benar:
* Jalankan perintah berikut untuk membangun aplikasi React dan salin ke direktori publik monolit:
* Jalankan perintah berikut untuk memicu pembangunan cloud baru dengan versi gambar yang diperbarui 2.0.0:

1. Update website with zero downtime

* Beri tahu Kubernetes bahwa Anda ingin memperbarui image untuk penerapan Anda ke versi baru dengan perintah berikut:
* Anda dapat memvalidasi pembaruan penerapan dengan menjalankan perintah berikut:

1. Cleanup

* Delete git repository:
* Hapus gambar Google Container Registry:
* Hapus artefak Google Cloud Build dari Google Cloud Storage:
* Delete GKE Service and Delete GKE Cluster:

1. **Kesimpulan**

Setelah melakukan praktik diatas, dapat disimpulkan bahwa mahasiswa dapat mendeploy docker image ke kubernetas, scale an application on Kubernetes dan dapat melakukan melakukan pembaruan bergulir di Kubernetas

1. 