### **Soal From Manual to Automation Testing**

1. Jelaskan jenis-jenis dari performance testing (minimal: 3).
2. Load Testing, load test ini tahapan pertamanya adalah menentukan target load. Load testing umumnya dipakai untuk acara musiman yang sifatnya selalu bertambah seiring berjalannya waktu, salah satu contohnya adalah sistem penjualan mudik lebaran, semakin dekat dengan acara yang ada semakin tinggi pula trafficnya.
3. Spike Testing, spike testing ini memiliki tujuan untuk mengetahui kapasitas sistem dalam mengurus traffic yang sesuai ekspektasi. Seperti namanya yaitu spike, maka load yang dikirim dilakukan secara dadakan dalam waktu yang sama, salah satu contohnya adalah sistem pengisian KRS pada kampus yang datangnya lebih sering dalam waktu yang tidak menentu dan mendadak.
4. Stress testing, stress testing ini biasanya melakukan pengujian dengan load yang di luar ekspektasi dan akhirnya membuat sistem tidak dapat mengurus traffic yang ada, tujuan dari pengujian ini untuk mengetahui kapasitas dari sistem yang berjalan.
5. Soak testing, soak testing biasanya akan melakukan pengujian dengan load yang biasa saja namun dalam jangka waktu yang lama, bertujuan agar dapat menganalisis dan mengevaluasi kebiasaan dari sistem yang kita miliki, biasanya bertujuan untuk mengetahui apakah sistem akan mengalami memory leaks dalam penggunaan yang berjangka lama.
6. Volume testing, volume testing digunakan untuk mengetahui kebiasaan dari sistem yang memiliki data besar, jadi datanya akan terus bertambah hingga jumlahnya sangat besar, contohnya adalah data dari suatu transaksi yang terus disimpan sebagai rekap transaksi.
7. Sebutkan dan jelaskan tools yang dapat digunakan untuk melakukan performance testing (minimal: 3).
8. Apache JMeter, tools ini adalah alat sumber terbuka yang banyak digunakan untuk menguji kinerja aplikasi web atau layanan web. JMeter mendukung banyak protokol dan teknologi seperti HTTP, HTTPS, FTP, SOAP, REST, JDBC dan banyak lagi. JMeter juga memiliki fitur grafis yang memudahkan pengaturan dan menjalankan tes.
9. LoadRunner, tools ini adalah alat bisnis yang dikembangkan oleh Micro Focus untuk menguji kinerja aplikasi bisnis berbasis web yang kompleks. LoadRunner mendukung berbagai teknologi seperti HTML5, Java, .NET, SAP, dan banyak lagi. LoadRunner berfitur lengkap sehingga pengguna dapat menguji aplikasi dalam berbagai skenario.
10. Gatling, tools ini adalah alat open source yang berfokus pada pengujian beban tinggi aplikasi web atau layanan web. Gatling menawarkan fitur simulasi pengguna yang sangat baik dan mendukung protokol HTTP, WebSocket, dan JMS. Gatling juga memiliki fitur real-time monitoring, sehingga memudahkan pengguna untuk mengontrol secara langsung sebuah aplikasi.
11. K6, tools ini adalah alat open source yang berfokus pada pengujian beban tinggi aplikasi web atau layanan web. K6 ditulis dalam Go dan mendukung beberapa protokol seperti HTTP, WebSocket, dan gRPC. K6 juga hadir dengan analitik yang kuat dan dasbor waktu nyata yang memudahkan pengguna untuk memantau aplikasi.
12. Apache Bench, tools ini adalah alat open source yang terintegrasi dengan Apache HTTP Server untuk menguji kinerja aplikasi web atau layanan web. Apache Bench dapat digunakan untuk mengirim permintaan HTTP bersamaan dan memantau aplikasi. Namun, Apache Bench hanya mendukung protokol HTTP dan tidak memiliki kemampuan pemantauan waktu nyata.
13. Locust, tools ini adalah alat open source yang memungkinkan pengguna untuk melakukan pengujian beban tinggi aplikasi web atau layanan web menggunakan Python. Locust mendukung protokol HTTP dan WebSocket dan memiliki simulasi pengguna yang dapat dikonfigurasi. Locust juga memiliki kemampuan pelaporan dan dasbor waktu nyata yang memudahkan pengguna memantau aplikasi.
14. Artillery, tools ini adalah alat open source yang memungkinkan pengguna untuk melakukan pengujian beban tinggi aplikasi web atau layanan web menggunakan JavaScript. Artileri mendukung protokol HTTP dan WebSocket dan memiliki simulasi pengguna yang dapat dikonfigurasi. Artileri juga memiliki kemampuan analitik dan pelaporan penuh.

### **Soal Behavior Driven Development (BDD)**

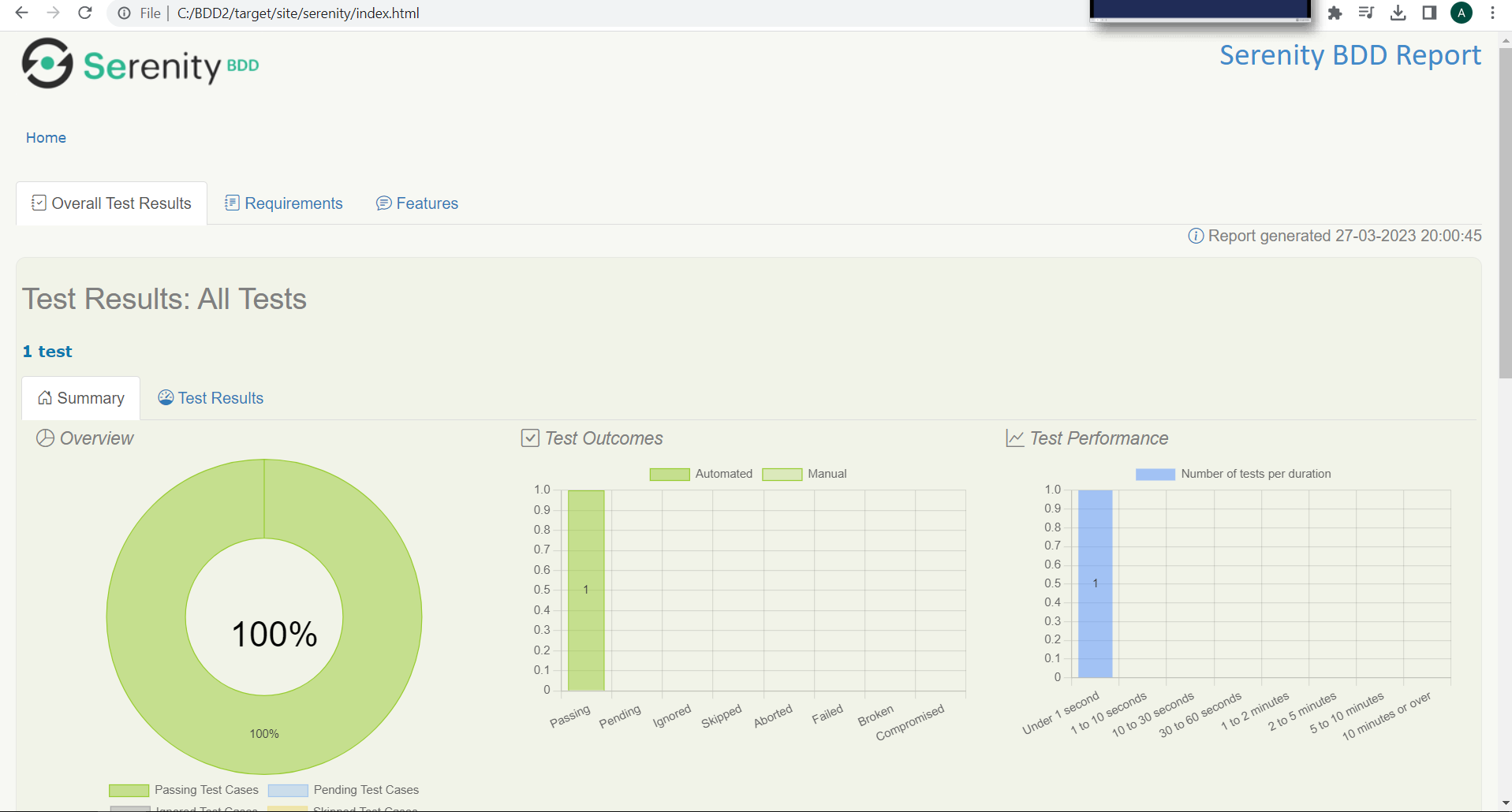
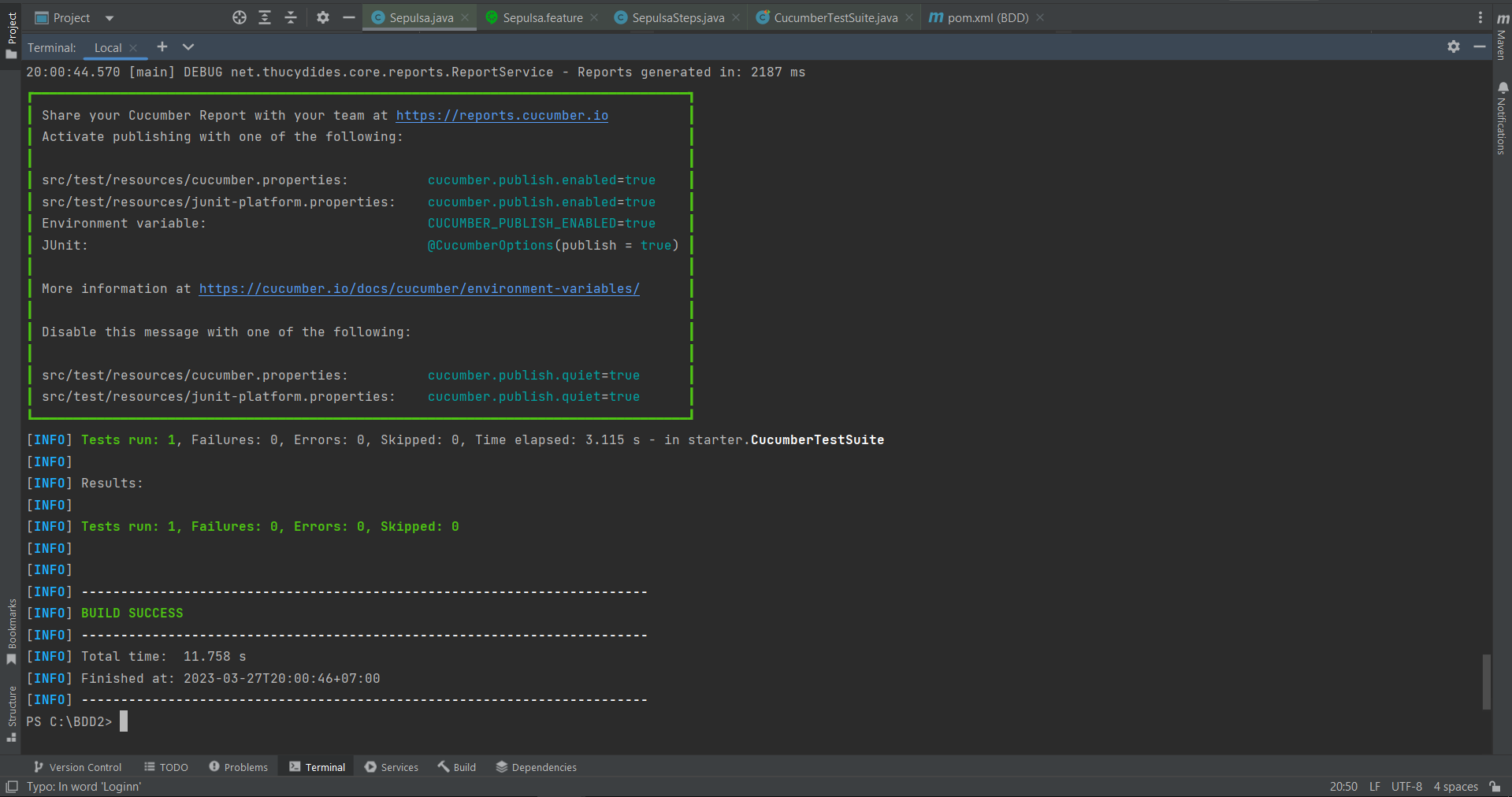
### **Soal Prioritas 1 (80)**

1. Lakukan testing pada aplikasi<https://www.linkedin.com/> dengan langkah-langkah sebagai berikut:
   1. Buatlah scenario test sebanyak mungkin menggunakan format BDD pada fitur-fitur halaman utama<https://www.linkedin.com/> .
   2. Buatlah project baru menggunakan intelliJ.
   3. Buatlah file feature Cucumber
   4. Buatlah class runner dan step
   5. Buatlah class steps yang berisi method-method yang dijalankan dalam scenario test.
   6. Jalankan class runner.
   7. Tampilkan report HTML dari Cucumber.

### 

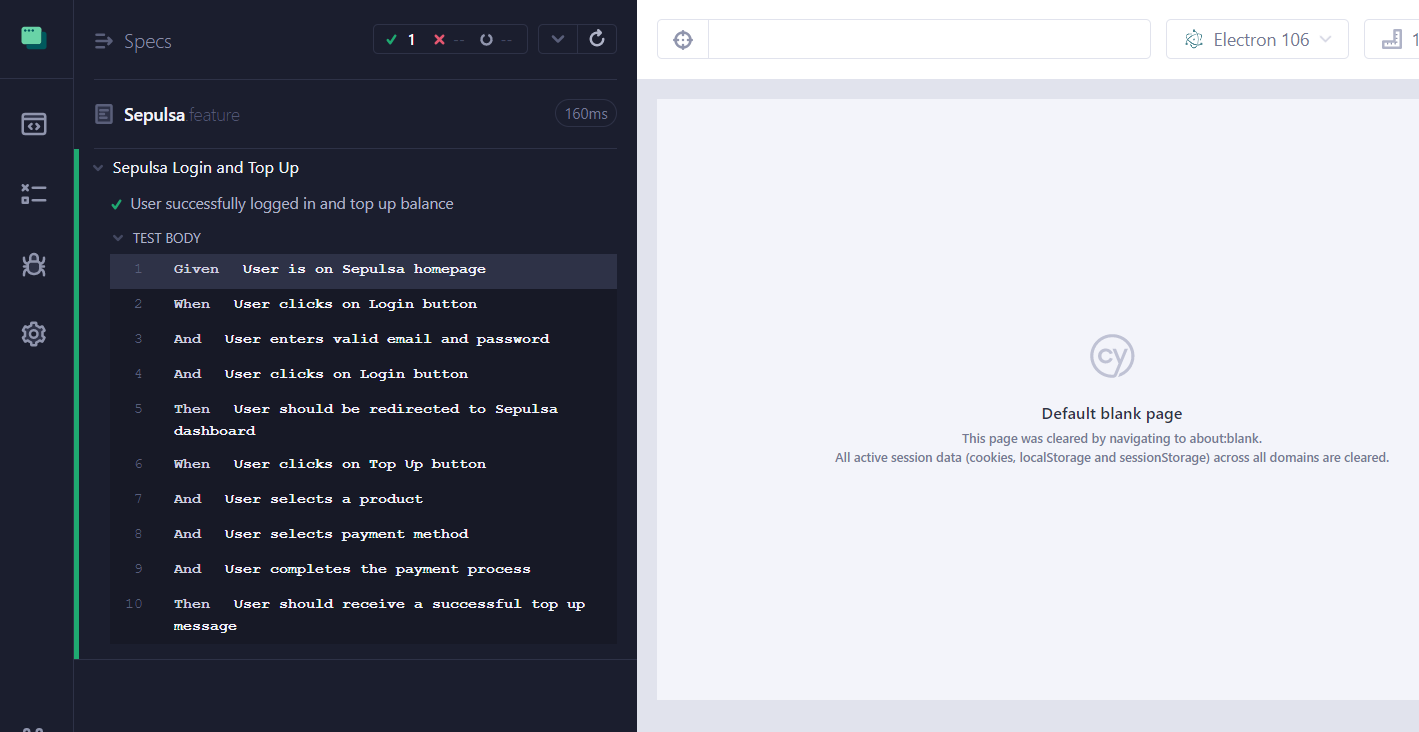
### **Soal Prioritas 2 (20)**

1. Lakukan testing pada aplikasi<https://www.sepulsa.com/> dengan langkah-langkah sebagai berikut:
   1. Buatlah scenario test menggunakan format BDD pada fitur login, pilih produk dan pilih metode pembayaran.
   2. Buatlah project baru menggunakan intelliJ.
   3. Buatlah file feature Cucumber
   4. Buatlah class runner dan step
   5. Buatlah class steps yang berisi method-method yang dijalankan dalam scenario test.
   6. Jalankan class runner.
   7. Tampilkan report HTML dari Cucumber.



### **Soal Eksplorasi (20)**

1. Lakukan testing pada aplikasi<https://www.sepulsa.com/> dengan langkah-langkah sebagai berikut:
   1. Menggunakan **cypress** sebagai tools untuk melakukan testing. Referensi penggunaan BDD dengan cypress dapat dilihat [disini](https://www.toolsqa.com/cypress/bdd-automation-framework-in-cypress/).
   2. Buatlah scenario test menggunakan format BDD pada fitur login, pilih produk dan pilih metode pembayaran.
   3. Tampilkan hasil testing dengan menggunakan cypress.



REFERENSI :

* PPT pada web alta.id
* <https://sysctl.id/jenis-performance-test/>
* <https://hariatylumbantobing.medium.com/performance-testing-pengertian-fokus-jenis-dan-contoh-7f1aaebbff8f>