

Penjelasan

SISTEM ANTRIAN KETIKA PENGUNJUNG WEB/APLIKASI MEMBLUDAK DALAM WAKTU YANG BERSAMAAN.

Penjelasan Flowchart dan ERD TIX ID

Dimulai user membuka aplikasi, setelah terbuka user auto diarahkan ke homepage/ landingpage TIX ID, lalu user memilih film dan memilih jadwal film, lalu menekan beli tiket. Ketika beli tiket user terdeteksi belum mempunyai akun maka akan langsung diarahkan ke halaman login/ register. Ketika User sudah melakukan login maka akan diarahkan langsung pada pemilihan tempat duduk. Ketika user sudah mengisi/ melakukan registrasi maka akan diarahkan ke pemilihan tempat duduk.

Pada halaman pemilihan tempat duduk user akan bebas memilih dimana ia akan duduk misal kursi A1, A2, ketika user sudah menekan kursi A1, A2 maka otomatis kursi akan terlock sementara katakannlah dibatasi hanya 2 menit. Jika user tidak kunjung melakukan checkout selama lebih dari 2 menit maka halaman pemilihan kursi akan otomatis ke refresh/ reload secara otomatis.

Namun ketika user melakukan checkout dibawah batas 2 menit maka akan mengarah halaman pembayaran. Pada halaman pemilihan pembayaran user diberi batas waktu selama 7 menit untuk memilih metode pembayaran. Jika User sudah melakukan pilihan pembayaran dan sudah melakukan pembayaran maka otomatis kursi akan ter-booking pada pemesan. Namun jika user tidak melanjutkan pembayaran lebih dari 7 menit maka akan dianggap cancel/ booking kursi gagal, yang dimana akan membuka lock kursi yang sementara. Sehingga kursi yang sebelumnya diboeking sementara maka akan terbuka kembali.

Ide untuk membuat sistem antrian ketika pengunjung web/aplikasi membludak dalam waktu yang bersamaan.

Jadi jika terjadi banyak request maka solusi yang paling tepat dan realistis menurut saya adalah

Opsi Pertama dengan membatasi user pada halaman yang user buka, dengan memasang rate limiter, misal katakanlah 500 user per menit, maka akan mengurangi resiko lonjakan user yang signifikan.

Opsi Kedua dengan membatasi session user, semisal pada system pembooking-an user sedang memilih produk yang “terbatas ketersediaannya”, maka halaman product akan otomatis ke refresh selama 5 menit sekali, sehingga terjadi peng-update an data. Dan jika user melakukan checkout namun user tidak segera checkout maka akan otomatis gagal untuk mem-booking.

Opsi Terakhir adalah dengan menggunakan API Gateway. Sederhananya pada system yang menggunakan API maka Ketika suatu website di *hit* jutaan request dari user, fungsi API ini adalah melakukan Load Balancing ke pada API's yang lain. Dengan kata lain Ketika di *hit* jutaan request maka akan terbagi dan diarahkan ke bagian API – API yang lain. Sehingga jutaan request yang terjadi ter-distribusi dengan merata.