

Backend - Final Project Assignment

Problem Statement

Pak Gagah dan Bu Guguh adalah orang tua dari Gigih. Mereka menjalankan bisnis katering selama bertahun-tahun dan mencatat data-data bisnis mereka di sebuah buku kas. Sebagai seorang anak yang baik, Gigih ingin membantu orang tuanya dalam menjalankan bisnis mereka. Oleh karena itu, Gigih berencana membuat aplikasi untuk membantu pencatatan data bisnis katering orang tuanya.

Business process dari orang tua Gigih adalah sebagai berikut: setiap pagi, Pak Gagah dan Bu Guguh akan mengumumkan menu makanan dan harganya ke grup WhatsApp katering. **Customer** biasanya akan **memesan melalui grup WhatsApp** dan melakukan pembayaran melalui transfer bank. **Jika customer belum membayar pesanan sampai jam 5 sore, customer dianggap membatalkan pesanan mereka.** Setelah jam 5 sore, Pak Gagah dan Bu Guguh akan **melihat daftar pesanan yang telah dibayar** dan akan menyiapkan **menu makanan untuk dikirim sesuai dengan pesanan**. Selain itu, setiap harinya, Pak Gagah dan Bu Gigih juga akan **melihat berapa total pendapatan mereka pada hari itu**.

Gigih bekerja sebagai seorang project manager di sebuah perusahaan bernama The Eco Mews. Meski bisa membuat user story, Gigih tidak bisa membuat aplikasi yang diinginkan sendiri. Oleh karena itu, Gigih membuat spesifikasi berikut untukmu agar kamu bisa membantunya membuat **aplikasi berbasis API** berikut.

User Story

| # | User Story | Keterangan Detil |
|---|---|---|
| 1 | As an owner , I want to create a new menu item So that I can show them to my customers later | Sebuah menu makanan selalu memiliki nama dan harga . Nama menu makanan tidak bisa duplikat. Harga menu makanan tidak boleh lebih kecil dari 0.01. Menu makanan bisa memiliki deskripsi yang panjangnya tak lebih dari 150 karakter. Menu makanan harus memiliki kategori. Bonus poin jika kamu mengimplementasikan menu makanan dengan lebih dari satu kategori. |
| 2 | As an owner , I want to update an existing menu item So that I can modify info related to the menu item | |

| | | |
|---|---|---|
| 3 | As an owner , I want to show the list of all menu items So that my customers can see the list of all menu items that I sell | |
| 4 | As an owner , I want to delete an existing menu item So that I can remove a menu item that is no longer provided by my catering service | |
| 5 | As an owner , I want to add customer's order So that I can prepare their order | <p>Pada tiap pesanan, pemilik catering menyimpan data email dari customer yang melakukan pemesanan. Data email harus dalam format email yg valid (contoh: savira@generasigigih.com adalah format email yang valid, halo@gigih adalah format email yang tidak valid).</p> <p>Satu customer bisa memesan lebih dari satu menu makanan dalam satu pesanan. Misalnya, Customer A bisa memesan "Nasi Uduk" dan "Nasi Kuning".</p> <p>Untuk tiap menu makanan, customer bisa memesan lebih satu porsi. Misalnya, Customer A bisa memesan dua porsi "Nasi Uduk".</p> <p>Perlu diingat, karena owner bisa mengubah data menu makanan, harga menu makanan pada pesanan yang telah dibuat sebelumnya tidak boleh berubah ketika data menu makanan berubah. Misalnya, owner mengubah harga "Nasi Uduk" dari 15,000 menjadi 20,000. Data harga "Nasi Uduk" pada order-order yang sudah dibuat sebelum perubahan harga tidak boleh berubah.</p> <p>Pemilik catering juga perlu untuk bisa melihat total harga dari tiap pesanan.</p> |
| 6 | As an owner , I want to update a customer's order So that I can modify info related to the order | <p>Ketika pesanan pertama kali dibuat, status pesanan tersebut adalah "NEW". Jika customer telah mengkonfirmasi pembayaran di grup WhatsApp, data pesanan akan berubah statusnya menjadi</p> |

| | | |
|---|---|---|
| | | <p>"PAID". Setelah jam 5 sore, owner akan mengubah status semua pesanan yang belum dibayar menjadi "CANCELED".</p> <p>Bonus poin jika kamu dapat membuat otomasi untuk mengubah status semua pesanan yang belum dibayar pada jam 5 sore menjadi "CANCELED".</p> |
| 7 | <p>As an owner, I want to see a daily report of orders So that I can see the revenue that I have generated for that day</p> | <p>Laporan harian harus menampilkan daftar semua pesanan pada hari tersebut beserta alamat email customer dan total harga dari tiap pesanan.</p> <p>Bonus poin jika kamu bisa menyediakan cara untuk menampilkan laporan:</p> <ul style="list-style-type: none"> - Berdasarkan email customer (misal: tampilkan semua pesanan dari zanya@generasigigih.com) - Berdasarkan total harga (misal: tampilkan semua pesanan dengan total harga lebih dari 50,000) - Berdasarkan rentang waktu (misal: tampilkan semua pesanan dari tanggal 11-17 April 2022) |

The Rule of The Game

1. Kamu punya 10 hari untuk membuat implementasi project ini, batas akhir pengumpulan adalah pada tanggal 23 April jam 11.00 WIB ke link berikut: <https://bit.ly/gg-be-finalproject>,
2. Gunakan Rails untuk menyelesaikan problem statement ini,
3. Kamu bisa menggunakan skeleton Rails project di Replit berikut: <https://replit.com/@gojek22/Gigih-Family-Catering>
4. Jika kamu memiliki asumsi, tulis asumsi tersebut pada file README.md,
5. Idealnya, README.md yang kamu tulis bisa membantu instruktur untuk menjalankan aplikasi yang kamu buat dengan mudah
6. Kamu boleh menggunakan library atau gem apa saja,
7. Kamu bisa bertanya tentang problem statement pada para instruktur,
8. Project ini akan dinilai dengan menjalankan RUN pada Replit dan menguji semua API yang kamu buat. Sebaiknya tidak perlu ada langkah-langkah ekstra untuk bisa menjalankan project yang kamu submit.

Grading Criteria

1. Kamu berhasil membuat API yang bisa menyelesaikan 7 user story pada problem statement. **Bonus point** jika kamu juga membuat web UI untuk problem statement ini, tapi perlu diingat bahwa membuat web UI tidak wajib.
2. Program tidak memiliki error atau crash.
3. Program mendemonstrasikan desain Object Oriented yang baik sesuai dengan yang diajarkan pada module 1.
4. Program mendemonstrasikan desain Database yang baik sesuai dengan yang diajarkan pada module 2.
5. Program mendemonstrasikan implementasi MVC architecture pada Rails yang baik sesuai dengan yang diajarkan pada module 3.
6. Program memiliki unit test pada Model dan Controller sesuai dengan yang diajarkan pada module 4.
7. Kamu membuat README.md yang mudah dipahami. Sebagai referensi untuk README.md, lihat halaman berikut: <https://www.makeareadme.com/>.
8. Kamu mengikuti konvensi Rails untuk struktur direktori. Sebagai referensi, lihat halaman berikut: <https://www.tutorialspoint.com/ruby-on-rails/rails-directory-structure.htm>.
9. Kamu mengikuti Ruby code writing styling guide. Sebagai referensi, lihat halaman berikut: <https://ruby.style/>.
10. Kamu mengintegrasikan Replit dengan akun Github dan memiliki riwayat git commit yang granular. **Bonus poin** jika riwayat git commit mengikuti pola TDD. Sebagai referensi, lihat halaman berikut: <https://replit.com/talk/learn/Replit-Git-Tutorial/23331>.
11. Idealnya, kode yang kamu tulis harus mudah dibaca dan dipahami. Coba gunakan nama variable, fungsi, dan kelas sebaik yang kamu bisa.

KM_G2BE4084
Muhammad Afandi Aziz
afandiaziz46@gmail.com