

Perancangan Dan Implementasi *Catering Ordering System* “Hena Catering” Menggunakan *Rapid Application Development*

Abdullah Ammar¹, M. Zulfikri², Sirajuddin Hawari³, Trevy Jonatya Novella⁴, Ahlijati Nuraminah⁵
Program Studi Ilmu Komputer
Sekolah Tinggi Ilmu Manajemen dan Ilmu Komputer ESQ
Jl. TB Simatupang, RT.3/RW.3, Cilandak Timur., Jakarta 12560
abdullah.a@students.esqbs.ac.id¹*, m.zulfikri.m@students.esqbs.ac.id²,
sirajuddin.h@students.esqbs.ac.id³, t.jonatya.n@students.esqbs.ac.id⁴,
ahlijati.nuraminah@esqbs.ac.id⁵

Abstrak. Dalam proses pemesanan catering masalah yang sering muncul adalah kesalahpahaman antara pembeli dan penjual dikarenakan masih menggunakan cara manual dalam pemesanan, sulitnya pembukuan rincian histori pesanan oleh penjual karena diharuskan menuliskan satu persatu pada buku besar. Untuk mengatasi hal ini maka, *catering ordering system* dirancang dan diimplementasikan untuk memfasilitasi pemesanan catering dengan platform berbasis website. Penelitian ini menggunakan metodologi *Rapid Application Development (RAD)* untuk menghasilkan sistem yang cepat dengan pendekatan *object oriented programming* agar sistem lebih *reusable*. Bahasa pemrograman yang digunakan dalam membangun sistem ini adalah PHP dan *database* MySQL. Hasil akhir berupa sistem yang memiliki fitur untuk pemesanan menu catering, detail menu catering dan pemilik mempunyai fitur histori pemesanan, fitur memanipulasi menu catering, mengkonfirmasi pembayaran dan pengiriman. Hasil pengujian menunjukkan bahwa pemesanan menu catering dapat dilakukan dengan mudah dan cepat melalui platform website yang dapat diakses melalui *device* apa saja.

Kata Kunci: Rapid application development, pemesanan catering, rekayasa perangkat lunak, object-oriented programming

1 Pendahuluan

Perkembangan teknologi informasi berpengaruh besar bagi kehidupan. Teknologi informasi dibutuhkan dalam semua bidang, contohnya yaitu bidang usaha makanan seperti catering. “Hena Catering” merupakan sebuah usaha catering di kota Depok yang berdiri sejak tahun 2012 yang dimiliki oleh Bapak Royani. Saat ini Hena Catering menggunakan metode manual untuk sistem pemesanannya yaitu hanya melalui telepon dan media sosial. Jika *customer* ingin melihat menu yang tersedia, maka pihak Hena Catering akan mengirimkan foto-foto menu kepada *customer*. Pencatatan histori pembelian dan pemesanan juga masih dilakukan secara manual dengan mencatat pada satu buku besar.

Berdasarkan hasil wawancara dengan pemilik Hena Catering, diperoleh informasi bahwa proses pencatatan pesanan secara manual mempunyai beberapa kelemahan sehingga informasi yang diterima oleh konsumen tidak begitu jelas dan tercatat dengan baik. Contohnya dalam penyampaian ketersediaan menu yang diinginkan konsumen pada sebuah catering, proses untuk pengantaran pemesanan, serta pembukuan yang masih manual. Proses pencatatan pesanan dilakukan oleh staf Hena Catering secara manual sehingga rawan terjadi kesalahan dalam pencatatan ke dalam buku besar sehingga dapat menyebabkan adanya ketidakseimbangan antara barang yang dipesan dan uang yang diterima. Jika hal ini berlanjut maka dapat terjadi kesalahpahaman antara pembeli maupun penjual yang dapat merugikan nama baik catering itu sendiri. Jika tidak ada inovasi dalam hal pemesanan menu catering, maka Hena Catering dikhawatirkan tidak mampu bersaing dalam mendapatkan pelanggan yang lebih banyak.

Berdasarkan latar belakang tersebut, tujuan dari penelitian ini adalah untuk membuat aplikasi pemesanan catering berbasis web agar dapat membantu dan mempermudah proses pemesanan yang dilakukan secara online. *Catering ordering system* ini difokuskan untuk salah satu catering. Diharapkan dengan adanya sistem ini Hena Catering dapat memanfaatkan kemajuan teknologi untuk meningkatkan proses bisnis, serta memberikan kemudahan bagi pembeli maupun Hena Catering sendiri dalam melakukan transaksi pemesanan catering.

Penelitian mengenai perancangan sistem pemesanan catering telah dilakukan oleh [1] yaitu pembuatan aplikasi pemesanan catering pada CV Maharani menggunakan metode RAD. Aplikasi dibuat menggunakan pemodelan *Data Flow Diagram*. Kesimpulan dari penelitian tersebut adalah aplikasi yang dihasilkan akan memudahkan masyarakat untuk melakukan pemesanan catering secara online. Penelitian lainnya yang juga mengangkat topik pengembangan aplikasi pemesanan menu secara online dilakukan oleh [2]. Pengembangan aplikasi pemesanan menu *online* dilatarbelakangi oleh wisatawan asing di Korea yang menolak untuk membeli makanan karena tidak mengetahui komposisi menu dan harga yang ditawarkan. Sistem yang dikembangkan berupa aplikasi mobile yang diharapkan mampu meningkatkan penjualan menu dan meningkatkan wisata kuliner Korea.

2 Tinjauan Pustaka

Pada bagian ini akan dipaparkan beberapa tinjauan teori yang digunakan sebagai landasan dalam pengembangan sistem.

2.1 *Software Engineering*

Software Engineering (SE) adalah disiplin pengembangan perangkat lunak termasuk pembuatan, pemeliharaan, manajemen organisasi pengembangan perangkat lunak dan manajemen kualitas [3]. Menurut *IEEE Computer Society* yang dijelaskan dalam [3] rekayasa perangkat lunak sebagai penerapan suatu pendekatan yang sistematis, disiplin dan terkuantifikasi atas pengembangan, penggunaan dan pemeliharaan perangkat lunak, serta studi atas pendekatan-pendekatan ini, yaitu penerapan pendekatan *engineering* atas perangkat lunak.

2.2 *Rapid Application Development (RAD)*

Metode pengembangan perangkat lunak yang digunakan adalah Metode RAD (*Rapid Application Development*). *Rapid application development* merupakan kumpulan metodologi yang muncul sebagai tanggapan terhadap kelemahan pengembangan *waterfall* dan variasinya. RAD menggabungkan teknik dan *tools* untuk mempercepat tahap analisis, desain, dan implementasi untuk mengembangkan sistem dengan cepat sehingga dapat di-*deliver* kepada pengguna untuk mendapatkan evaluasi dan umpan balik [4]. Dalam prosesnya metode RAD ini tidak perlu mengeluarkan biaya yang besar, baik ketika persyaratan pengguna tidak terlalu jelas, juga baik untuk sistem yang kompleks sehingga metode tersebut dipilih untuk pembangunan sistem.

Metode *Rapid Application Development* yang bersifat *Iterative Incremental Development* di mana dilakukan pemecahan keseluruhan proyek menjadi serangkaian versi yang dikembangkan secara berurutan. Tahapan *planning* dan analisis dilakukan pada awal proyek. Kemudian sistem dipecah ke dalam beberapa versi. Pada masing-masing versi sistem dilakukan tahapan analisis, desain dan implementasi. Pengembangan sistem secara cepat menghasilkan sistem versi awal yang bisa langsung dicoba oleh pengguna. Pengguna dapat memberikan evaluasi dan masukan untuk pengembangan ke sistem versi berikutnya.

3 Metodologi Penelitian

Metodologi penelitian dilakukan sesuai dengan tahapan *Rapid Application Development* sebagai metode pengembangan sistem. Tahapan metode pengembangan sistem menggunakan *Rapid Application Development* yang dipaparkan oleh Kendal dalam [4] dijelaskan sebagai berikut:

Tahapan metode RAD terdiri dari 3(tiga) tahap yang terstruktur dan saling bergantung di setiap tahap, yaitu :

1. *Requirements planning*

- a. Melakukan pemecahan masalah bisnis dan menentukan apa saja yang dapat dilakukan oleh sistem.
- b. Identifikasi kebutuhan informasi dan tujuan dari sistem yang dibangun melalui pertemuan antara tim pengembang yang direpresentasikan oleh analis dengan pengguna sistem.

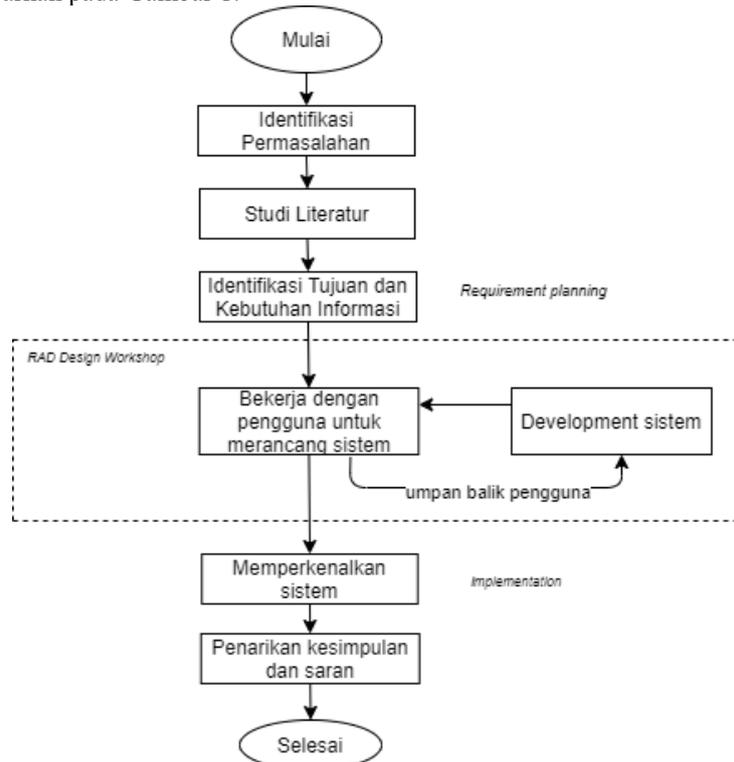
2. *Design Workshop*

- a. Analis dan pemrogram bekerja sama membangun dan menunjukkan representasi visual desain dan pola kerja kepada pengguna.

- b. Pengembang membangun *prototype* awal sistem dan mendemonstrasikan *prototype* tersebut kepada pengguna, kemudian pengguna memberikan evaluasi terhadap *prototype* sistem.
- c. Tim pengembang melakukan penyempurnaan *prototype* berdasarkan masukan dari pengguna. Hasil perbaikan sistem adalah versi kedua dari sistem. Versi kedua ini kemudian didemonstrasikan kembali kepada pengguna.

3. Implementation

- a. Analis bekerja dengan para pengguna secara intens selama fase *design workshop* dan merancang aspek-aspek bisnis dan nonteknis lainnya. Setelah semua aspek disetujui dan sistem telah selesai dibangun, kemudian dilakukan pengujian sistem. Setelah pengujian selesai tanpa ada tambahan cacat lainnya, sistem kemudian diperkenalkan kepada pengguna. Tahapan penelitian dilakukan sesuai dengan alur yang digambarkan pada Gambar 1.



Gambar. 1. Alur penelitian.

Penjelasan alur penelitian pada Gambar 1 sebagai berikut:

1. Identifikasi Permasalahan. Bertujuan untuk mendefinisikan latar belakang permasalahan yang mendasari penelitian permasalahan utama penelitian.
2. Studi Literatur. Pelaksanaan tinjauan pustaka dilakukan untuk mencari teori, penelitian, serta metodologi yang relevan dengan pertanyaan penelitian yang telah didefinisikan pada tahapan sebelumnya yang besumber dari buku teks, makalah dan jurnal internasional sehingga menghasilkan keluaran berupa tinjauan pustaka yang mendasari penelitian.
3. Tahap *requirement planning*. Bertujuan untuk mengetahui apa saja yang menjadi kebutuhan sistem, mengetahui dengan cara mengidentifikasi kebutuhan informasi dan masalah yang dihadapi, batasan sistem, kendala dan juga alternatif pemecahan masalah.
4. Tahap *Design Workshop*. Bertujuan untuk mengidentifikasi solusi alternatif dan memilih solusi yang terbaik. Pada fase ini dilakukan perancangan arsitektur sistem, skema *database*, dan *prototype* sistem.
5. Tahap *Implementation*. Bertujuan untuk mengimplementasikan sistem dalam bentuk yang dimengerti oleh mesin dengan wujud berbentuk program atau unit program. Termasuk melakukan pengujian sistem
6. Penyusunan Kesimpulan dan Saran. Langkah terakhir adalah pembuatan kesimpulan dan saran berdasarkan hasil dan analisis dari tahapan.

4 Hasil dan Pembahasan

Pada bab ini akan dipaparkan hasil implementasi sistem sesuai dengan tahapan penelitian. Setiap tahapan penelitian menghasilkan output berupa artefak sistem.

4.1 Fase *Requirement Planning*

Sebelum *requirement* dapat dianalisis, dimodelkan, atau ditentukan, maka harus melalui aktivitas komunikasi terhadap seluruh *stakeholder*. Dalam tahap *communication* hal yang dilakukan menemui *client* dan dilakukan proses *interview* terkait Hena Catering agar seluruh *requirement* dapat tercatat dengan baik.

Berdasarkan hasil wawancara dengan pemilik Hena Catering, ada dua kebutuhan yang dibutuhkan oleh sistem, yaitu kebutuhan fungsional dan kebutuhan non-fungsional.

Kebutuhan Fungsional

1. Untuk melakukan pemesanan menu catering, pengguna harus terdaftar di dalam sistem.
2. Pemilik catering dapat menambahkan, mengubah dan menghapus menu catering sesuai kebutuhannya
3. Pemilik catering ingin dapat mengubah informasi *profile*-nya supaya tetap terkini
4. Pembeli juga dapat mengubah data dirinya, menambahkan alamat pengiriman menu yang dikehendaknya
5. Pemilik catering dapat melihat siapa yang memesan menu catering dan apa saja menu yang dipesan
6. Pengunjung website catering dapat melihat daftar menu catering yang ditawarkan dan memfilter menu berdasarkan kategori dan keyword nama menu
7. Pemilik catering dapat mengkonfirmasi (menyetujui/menolak) pemesanan pembeli
8. Pembeli dapat melihat seluruh histori pemesanan sebelumnya

Kebutuhan Non-Fungsional

1. *Availability*. Sistem harus terus menerus beroperasi selama 7 hari perminggu, 24 jam per hari tanpa gagal.
2. *Ergonomy*. Sistem bisa digunakan dengan mudah oleh semua kalangan pengguna.
3. *Portability*. Sistem bisa diakses oleh semua versi browser dan semua perangkat gawai.
4. *Security*. Semua data yang disimpan tidak bisa diakses selain oleh administrator database.

Kebutuhan sistem diusukan dalam model *Use Case Diagram*. *Use Case Diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem dengan merepresentasikan sebuah interaksi antara aktor dengan sistem [5]. *Use Case Diagram* sistem pemesanan catering dapat dilihat pada Gambar 2.



Gambar. 2. Use Case Diagram sistem pemesanan Hena Catering

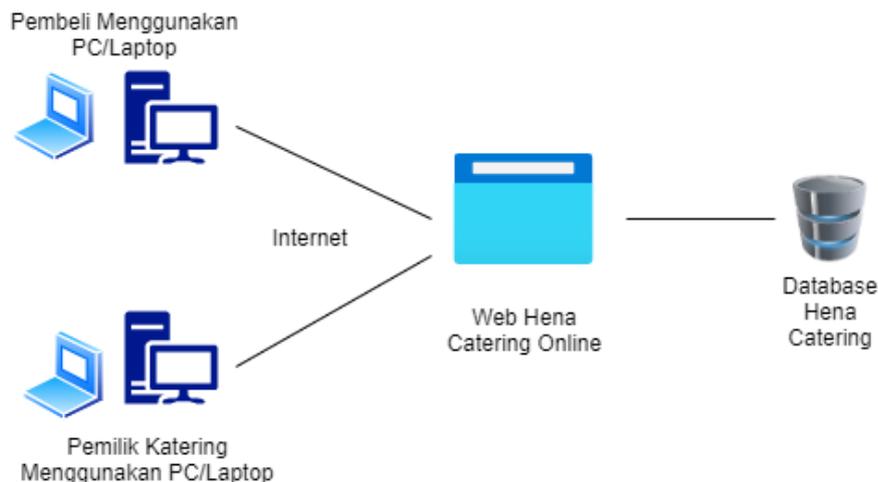
Pada Gambar 2 Use Case Digram Hena Catering terdapat tiga aktor yaitu *Guest* (pengunjung), Pembeli dan Penjual. Dari kebutuhan fungsional yang dianalisis, menghasilkan 20 (dua puluh) Use Case yang akan dikembangkan. Guest/pengunjung dapat melihat menu catering yang tersedia dan menambahkannya ke keranjang. Namin untuk melakukan pemesanan, pengunjung harus melakukan registrasi sebagai pembeli ke dalam sistem dan login menggunakan *account* yang telah diregistrasikan. Pembeli dapat memilih alamat pengiriman catering. Setiap pesanan catering yang masuk akan dikonfirmasi oleh penjual. Pembayaran dilakukan melalui transfer ke rekening penjual. Setelah melakukan pembayaran, penjual akan mengkonfirmasi pembayaran dan melakukan pengiriman catering yang dipesan. Pembeli juga dapat melihat riwayat pemesanan sebelumnya.

4.2 Fase Design Workshop

Dalam fase design dilakukan perancangan sistem berdasarkan hasil analisis kebutuhan yang telah dilakukan pada fase sebelumnya. Perancangan yang dilakukan berdasarkan [3] antara lain perancangan arsitektur sistem, perancangan skema basis data, dan perancangan *user interface* sistem.

Perancangan Arsitektur Sistem

Arsitektur sistem menggunakan *web architecture* di mana kode program akan ditempatkan di *web server*. *Database server* juga ditempatkan di *server* yang sama dengan *web server* untuk memudahkan *maintenance* sistem. Pengguna dapat mengakses sistem melalui jaringan internet dengan menggunakan browser di personal komputer atau laptop. Arsitektur sistem dapat dilihat pada Gambar 3.

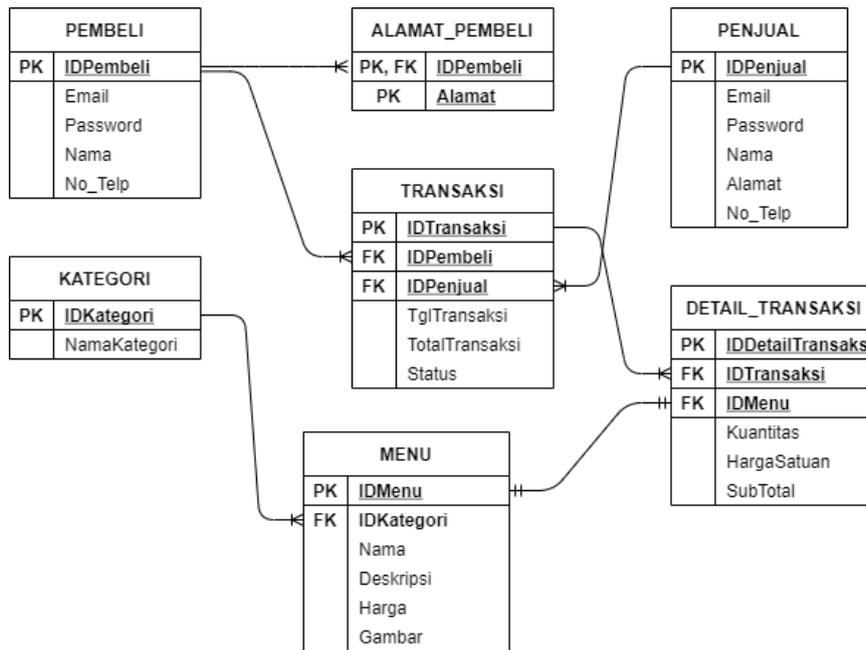


Gambar. 3. Arsitektur sistem

Gambar 4 merupakan arsitektur sistem di mana pengguna yaitu pembeli dan pemilik catering dapat mengakses web Hena Catering online melalui jaringan internet dengan menggunakan *browser* di personal komputer atau laptop.

Perancangan skema basis data

Perancangan skema basis data menggunakan teknik perancangan database yaitu *conceptual design*, *logical design* dan *physical design* [6]. Pada tahap perancangan konseptual, digambarkan model konseptual sesuai dengan kebutuhan sistem menggunakan *Entity Relationship Modelling*. Pada tahapan perancangan *logical*, ER Diagram diterjemahkan menjadi *relational data model*. Selanjutnya pada tahap *physical design*, dibuat skema database spesifik sesuai dengan DBMS yang dipilih yaitu MySQL. Rancangan basis data dapat dilihat pada Gambar 4.

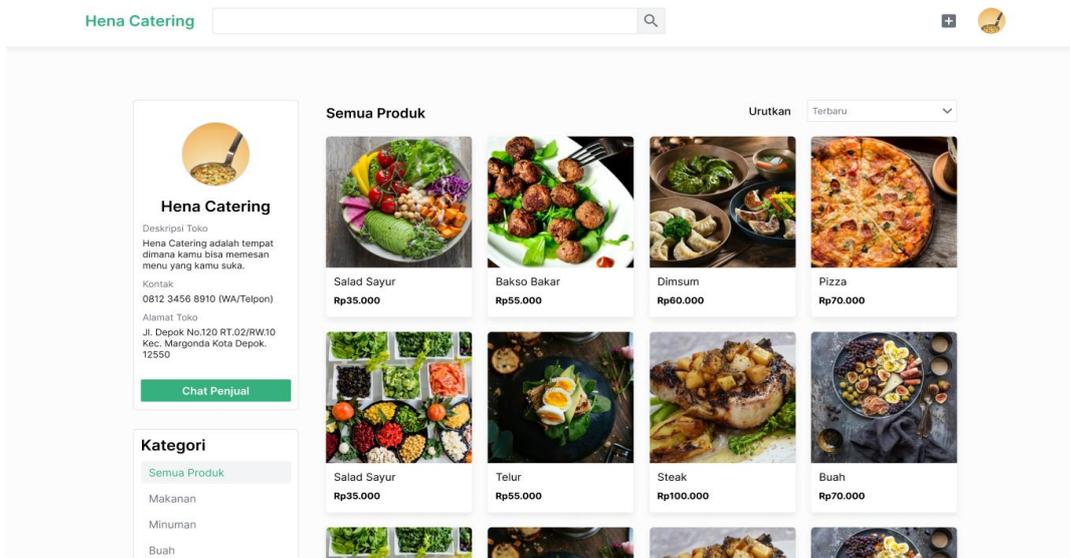


Gambar. 4. Rancangan basis data

Pada Gambar 5 dapat dilihat hasil perancangan basis database *relational* di mana terdapat tujuh tabel untuk menyimpan data. Tabel pembeli menyimpan data-data pembeli, sedangkan tabel alamat pembeli menyimpan data alamat pembeli di mana setiap pembeli dapat memiliki lebih dari satu alamat. Tabel menu menyimpan data-data menu. Setiap menu memiliki kategori tersendiri. Tabel transaksi menyimpan data transaksi pemesanan catering. Setiap transaksi pemesanan dapat terdiri atas banyak menu yang disimpan di tabel Detail_Transaksi. Tabel Penjual digunakan untuk menyimpan data profil penjual.

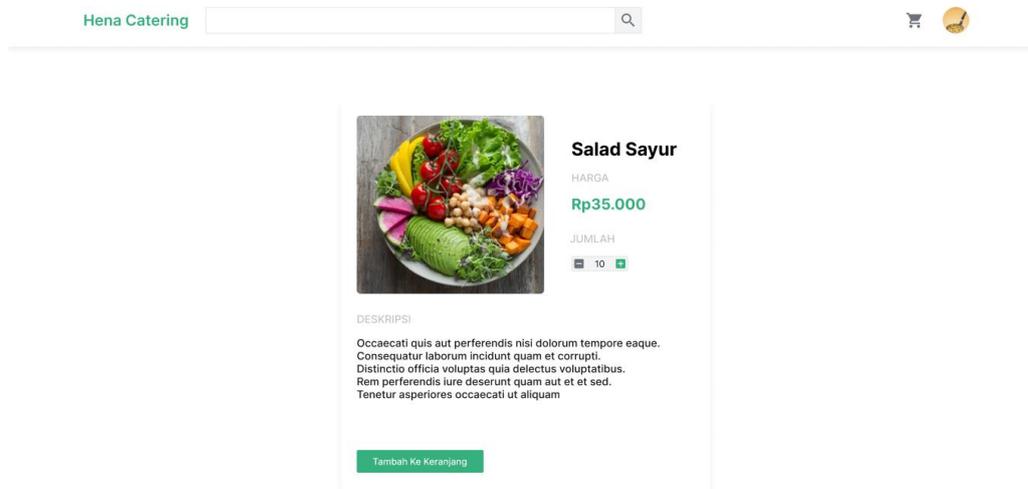
Perancangan User Interface

Perancangan *user interface* sistem dilakukan dengan menterjemahkan kebutuhan fungsional dan non fungsional sistem ke dalam desain tampilan antar muka sistem yang mudah digunakan (*easy to use*) dan mudah dipelajari (*easy to learn*) [7]. Prinsip desain yang memprioritaskan pada *usability* sistem di mana user dapat memanfaatkan seluruh fitur yang ada dengan mudah. Beberapa hasil rancangan user interface ditampilkan pada Gambar 5 dan Gambar 6.



Gambar. 5. Rancangan user interface halaman dashboard penjual.

Gambar 5 adalah rancangan user interface untuk halaman dashboard penjual. Pada halaman ini ditampilkan semua produk yang dijual oleh Hena Catering. Selain itu juga terdapat informasi profil Hena Catering pada sidebar sebelah kiri.



Gambar. 6. Rancangan *user interface* halaman detail menu

Gambar 6 merupakan rancangan user interface halaman detail menu catering yang dapat dipesan. Pada halaman ini ditampilkan nama produk, deskripsi, harga, jumlah kuantitas yang dapat dipesan, serta sebuah button untuk menambahkan menu ke keranjang pemesanan.

4.3 Fase Implementation

Ada beberapa hal yang dilakukan dalam fase implementation yaitu *preparation*, *coding*, *testing* dan *deployment* yang dijelaskan sebagai berikut

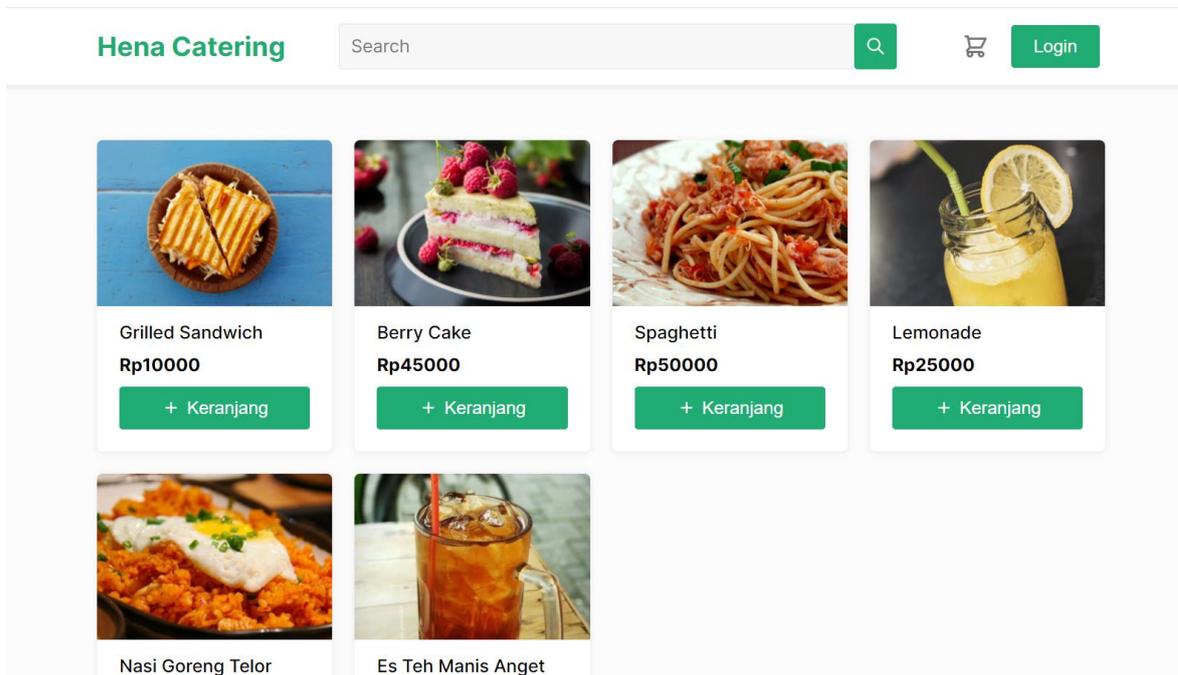
1. Preparation

- a. Pemilihan bahasa pemrograman dan tools. Bahasa pemrograman yang digunakan dalam membangun sistem ini adalah PHP. Tools yang digunakan Visual Studio Code sebagai code editor, Composer untuk Dependencies Manager. Version Control System menggunakan Git dan Github sebagai repository untuk mempermudah kolaborasi.
- b. Membuat Struktur Proyek. Struktur proyek yang baik memudahkan pengembang untuk mengelola proyeknya. Struktur dalam proyek ini adalah sebagai berikut
 - *Class*. Digunakan untuk menyimpan class
 - *Mail*. Digunakan untuk menyimpan template email, template email digunakan untuk mengirimkan notifikasi dengan struktur email yang sama kepada pengguna.
 - *Pages*. Digunakan untuk menyimpan halaman-halaman pengguna seperti halaman register, login, dashboard penjual, dan dashboard pembeli.
 - *Uploads*. Digunakan untuk menyimpan file-file yang diupload oleh pengguna seperti foto profile dan foto menu.
 - *Vendor*. Vendor adalah folder yang otomatis terbentuk ketika menggunakan library yang diinstall menggunakan Composer.

2. Coding

Setelah struktur proyek terbentuk, maka mulai dilakukan development dengan membuat code. Code dibuat membuat pendekatan *object oriented programming*. *Object Oriented Programming* adalah suatu metode pemrograman yang berorientasi kepada objek yang bertujuan untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada di *real-world* [8]. Pembuatan *code* dilakukan secara bertahap sesuai dengan prioritas kebutuhan sistem. Setelah *code* fitur selesai dibuat maka perubahan tersebut akan ditambahkan ke dalam Github menggunakan perintah-perintah yang ada di Git seperti *git*

add, *git commit*, dan *git push* [9]. Hasil implementasi sistem berupa aplikasi web yang dapat dijalankan dari *browser* apa saja dengan konsep *responsive system* dapat dilihat pada Gambar 7.



Gambar. 7. Hasil implementasi sistem pemesanan *catering*

Gambar 7 menunjukkan hasil implementasi sistem berupa aplikasi web responsif . Halaman ini merupakan halaman utama sistem yang akan diakses oleh pengguna pada saat memasukkan *url* sistem melalui *browser*.

3. Testing

Untuk memastikan bahwa sistem yang dibangun sesuai dengan spesifikasi kebutuhan user, maka dilakukan pengujian sistem dengan beberapa tingkatan yaitu *unit testing*, *integration testing* dan *system testing* [10] . Teknik pengujian yang diterapkan adalah *black box testing* dengan berfokus pada *functional testing*. Hasil pengujian *functional* sistem dapat dilihat pada Tabel 1.

Tabel 1. Hasil pengujian *functional system* menggunakan metode *black box testing*

Use Case	Test Case dieksekusi	Test Case "Passed"	Test Case pending	Prioritas
UC001 – Register	100%	100%	0	High
UC002 – Login	100%	100%	0	High
UC003 – Melihat Menu Katering	100%	100%	0	High
UC004 – Filter Menu Berdasarkan Kategori	100%	50%	0	Low
UC005 – Filter Menu Berdasarkan Nama	100%	100%	0	Medium
UC006 – Add To cart	100%	100%	0	High
UC007 – Checkout	100%	100%	0	High
UC008 – Memilih Alamat	50%	50%	1	Low
UC009 – Memasukan Alamat	0%	0%	1	Low
UC0010 – Melakukan Pembayaran	100%	100%	0	High
UC0011 – Mengkonfirmasi Pembayaran	100%	100%	0	High
UC0012 – Mengkonfirmasi Pengiriman	50%	50%	1	Low
UC0013 – Menyelesaikan Pembelian	50%	50%	1	Low
UC0014 – Melihat invoice pembelian	100%	100%	0	High

UC0015 – Melihat seluruh histori pesanan	100%	100%	0	High
UC0016 – Melihat seluruh histori pembelian	50%	50%	1	Medium
UC0017 – Memanipulasi Menu Katering	100%	100%	0	High
UC0018 – Memanipulasi Kategori Menu	100%	0%	0	Low
UC0019 – Memanipulasi Informasi pribadi	100%	60%	0	Low
UC0020 – Logout	100%	100%	0	High

Tabel 1 menunjukkan hasil *test report* untuk 20 *use case* yang telah dikembangkan. Nilai rata-rata *test case* yang 'passed' adalah 76%.

4. Deployment

Pada tahapan *deployment* terdapat beberapa aktivitas yaitu *delivery*, *support*, dan *feedback*. *Delivery* dilakukan kepada *client* dengan menyerahkan *software* yang sudah dikembangkan dari seluruh tahapan yang ada. *Support* diberikan selama satu minggu untuk memastikan bahwa sistem dapat berjalan dengan baik sesuai spesifikasi. Dalam tahap *delivery* juga diserahkan *user manual system* dan panduan instalasi sistem.

4 Kesimpulan dan Saran

Dengan menggunakan metode pengembangan perangkat lunak *Rapid Application Development* sehingga menghasilkan suatu sistem *catering ordering system* yang dapat berjalan sesuai *requirements* dari pengguna yaitu Bapak Royani pemilik Hena Catering. Sistem ini telah diimplementasikan dan digunakan oleh pihak Hena Catering. Berdasarkan hasil evaluasi penggunaan awal, sistem pemesanan katering online ini membantu Hena Catering dalam melakukan pencatatan pemesanan katering secara otomatis. Calon pembeli dapat melihat langsung menu-menu yang tersedia dengan mengakses website Hena Catering, sehingga proses pemesanan menu katering menjadi lebih mudah. Untuk pengembangan lebih lanjut dari *catering ordering system* guna memberikan manfaat kepada pengguna, ada beberapa hal yang dapat dilakukan yaitu memperbaiki fitur-fitur yang belum berjalan dengan sempurna, lebih detail dan spesifik menjalankan setiap proses yang dilakukan agar hasil yang didapatkan lebih maksimal.

Referensi

- [1] G. A. Paruwansyah, "Aplikasi Pemesanan Catering Berbasis Website Menggunakan Metode Rapid Application Development (RAD) Pada C.V Maharani," no. 8, pp. 1–7, 2018.
- [2] A. M. Torres, "Electronic Menu and Ordering Application System: A Strategic Tool for Customer Satisfaction and Profit Enhancement," *Int. J. u- e- Serv. Sci. Technol.*, vol. 9, no. 4, pp. 401–410, 2016, doi: 10.14257/ijunesst.2016.9.4.39.
- [3] R. S. Pressman, *Software Engineering A Practitioner's Approach 9th Edition*. McGraw-Hill Education, 2019.
- [4] R. Kaban and R. J. Nasution, "Penerapan Metode Rapid Application Development (RAD) dalam Perancangan Sistem Pemesanan Menu menggunakan Quick Response (QR) Code," *MEANS (Media Inf. Anal. dan Sist.*, vol. 5, no. 2, pp. 144–152, 2020.
- [5] Q. uz Zaman, A. Nadeem, and M. A. Sindhu, "Formalizing the use case model: A model-based approach," *PLoS One*, vol. 15, no. 4, pp. 1–29, 2020, doi: 10.1371/journal.pone.0231534.
- [6] T. Connolly and C. Begg, *Database System, A Practical Approach to Design, Implementation and Management Sixth Edition*. Pearson Education Limited, 2019.
- [7] H. Joo, "A study on understanding of UI and UX, and understanding of design according to user interface change," *Int. J. Appl. Eng. Res.*, vol. 12, no. 20, pp. 9931–9935, 2017.
- [8] A. Urdhwaresh, "Object-Oriented Programming and its Concepts," *Int. J. Innov. Sci. Res.*, vol. 26, no. 1, pp. 1–6, 2016, [Online]. Available: <http://www.ijisr.issr-journals.org/>.
- [9] I. Azimi, P. Studi, R. Perangkat, L. Aplikasi, and U. Telkom, "Pengaruh Penggunaan Version Control System Terhadap Proses Belajar Pemrograman Mahasiswa," vol. 5, no. 2, 2019.
- [10] S. Jat and P. Sharma, "Analysis of Different Software Testing Techniques," *Int. J. Sci. Res. Res. Pap. Comput. Sci. Eng.*, vol. 5, no. 2, pp. 77–80, 2017.