

Usecase lihat menu

|  |  |
| --- | --- |
| **Nomor** | 001 |
| **Nama** | Lihat menu |
| **Tujuan** | Proses ini digunakan untuk melihat menu pada system yang tersedia di setiap meja, pelanggan dapat melihat-lihat informasi yang tersedia dalam system, pelanggan dapat melihat nomor meja yang sudah ditentukan dalam system. |
| **Deskripsi** | Usecase ini menggambarkan proses pelanggan melihat menu pizza yang kemudian pelanggan harus melihat nomor meja yang tersedia di system, pelanggan bebas memilih menu yang tersedia dan melihat informasi apa saja yang tersedia di system, nomor meja digunakan untuk memudahkan pelayan dan koki membuatkan pesanan jika pelanggan memesan menu. |
| **Actor** | pelanggan |
| **Prekondisi** | Pelanggan membuka system lalu melihat nomor meja dan melihat menu dan informasi yang tersedia. |
| **Scenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pelanggan memilih meja yang kosong |  |
| 1. Pelanggan membuka system yang tersedia di meja |  |
| 1. Pelanggan melihat nomor meja | 1. System memberikan nomor meja |
| 1. Pelanggan membuka menu |  |
| 1. Pelanggan melihat informasi dan menu pizza yang tersedia | 1. System menyediakan informasi dan menu pizza |
| **Scenario alternative: tidak memesan** | |
| 1. Pelanggan membuka sistem | 1. System menyediakan informasi, menu makanan, dan nomor meja |
| 1. Pelanggan tidak memesan | 1. System akan kembali ke semula |
| **Post Kondisi** | System memberi pilihan apakah ingin memesan atau tidak dan akan keluar dari system atau tidak. |

Usecase lihat nomor meja

|  |  |
| --- | --- |
| **Nomor** | 002 |
| **Nama** | Lihat nomor meja |
| **Tujuan** | Proses ini digunakan untuk melihat nomor meja yang disediakan oleh system |
| **Deskripsi** | Usecase ini menggambarkan proses pelanggan melihat nomor meja yang tersedia pada system untuk memudahkan pelayan dan koki melayani pelanggan |
| **Actor** | pelanggan |
| **Prekondisi** | Pelanggan membuka system lalu melihat nomor meja. |
| **Scenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pelanggan memilih meja yang kosong |  |
| 1. Pelanggan membuka system yang tersedia di meja |  |
| 1. Pelanggan melihat nomor meja | 1. System memberikan nomor meja |
| **Scenario alternative: tidak membuka menu** | |
| 1. Pelanggan membuka sistem | 1.System menyediakan informasi nomor meja |
| 1. Pelanggan tidak membuka menu | 2.System tetap akan menampilkan nomor meja |
| **Post Kondisi** | System memberi pilihan apakah akan keluar dari system atau tidak. |

Usecase isi biodata

|  |  |
| --- | --- |
| **Nomor** | 003 |
| **Nama** | Isi biodata |
| **Tujuan** | Proses ini digunakan untuk pelanggan mengisi biodata saat system pertama kali dibuka. |
| **Deskripsi** | Usecase ini menggambarkan proses pelanggan mengisi biodata minimal nama dan no tlp untuk data dalam system agar pelayan dan koki dapat dengan mudah meghubungi pelanggan jika suatu saat terjadi kesalahan pelayanan dan system akan menyimpan data tersebut sebagai database perusahaan dan laporan penjualan. |
| **Actor** | pelanggan |
| **Prekondisi** | Pelanggan membuka system lalu melihat nomor meja dan mengisi data dirinya walaupun pelanggan tidak memesan menu pizza. |
| **Scenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pelanggan memilih meja yang kosong |  |
| 1. Pelanggan membuka system yang tersedia di meja |  |
| 1. Pelanggan melihat nomor meja | 1.System memberikan nomor meja |
| 1. Pelanggan mengisi biodata |  |
| 1. Pelanggan melihat informasi dan menu pizza yang tersedia | 2.System menyediakan informasi dan menu pizza |
| **Scenario alternative: tidak mengisi** | |
| 1. Pelanggan membuka sistem | 1.System menyediakan informasi, menu makanan, dan nomor meja |
| 1. Pelanggan tidak mengisi data | 2.System tidak akan memberikan informasi menu dan pelayanan. |
| **Post Kondisi** | System memberi pilihan apakah akan keluar dari system atau tidak. |

Usecase pesan menu

|  |  |  |
| --- | --- | --- |
| **Nomor** | | 004 |
| **Nama** | | Pesan menu |
| **Tujuan** | | Proses ini digunakan untuk pemesanan menu melalui aplikasi pemesanan menu yang sudah dilengkapi dengan dengan nomor meja pada tiap aplikasi |
| **Deskripsi** | | Usecase ini menggambarkan proses pemesanan menu pizza oleh pelanggan yang kemudian akan diperiksa oleh koki apakah menu yang dipilih masih tersedia atau tidak. Jika persediaan ada maka proses pemesanan akan dilakukan sesuai dengan nomor meja pelanggan kemudian sistem mengeluarkan bill total bayar sesuai dengan menu yang dipesan. Jika menu yang dipilih sudah habis maka sistem akan memberi notifikasi kepada pelanggan bahwa menu yang dipilih sudah habis. |
| **Aktor** | | Pelanggan, pelayan, koki, kasir |
| **Prekondisi** | | Pelanggan telah memesan menu melalui sistem yang sudah tersedia pada masing-masing meja pelanggan. |
| **Skenario Utama** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. Pelanggan memilih menu yang telah disediakan | |  |
| 1. Pelanggan memesan menu sesuai menu yang dipilih | |  |
| 1. Pelanggan memasukan biodata untuk data pengunjung | |  |
|  | | 1. Sistem mengirimkan daftar pesanan yang dipilih pelanggan beserta nomor meja pemesan |
| 1. Koki akan memasak menu yang dipilih pelanggan | |  |
| 1. Koki memberikan pesanan kepada pelayan | |  |
|  | | 1. Sistem mengeluarkan bill total harga sesuai pesanan yang dipesan pelanggan |
| 1. Pelayan akan melihat nomor meja sesuai dengan bill yang dikeluarkan sistem | |  |
| 1. Pelayan mengantarkan pesanan ke pelanggan dengan nomor meja yang tertera pada bill | |  |
| 1. Pelayan memberikan bill (bukti bayar) kepada pelanggan | |  |
| 1. Pelanggan membayar pesanan sesuai dengan nominal yang tertera pada bill | |  |
| 1. Pelayan menyerahkan uang pembayaran kepada kasir | |  |
|  | | 1. Sistem menghitung pemasukan keuangan |
| Skenario Alternatif: Persediaan Habis | | |
| 1. Pelanggan memilih menu yang telah disediakan | |  |
| 1. Pelanggan memesan menu sesuai menu yang dipilih | |  |
| 1. Pelanggan memasukan biodata untuk data pengunjung | |  |
|  | | 1. Sistem mengirimkan daftar pesanan yang dipilih pelanggan beserta nomor meja pemesan |
| 1. Koki memeriksa ketersediaan pesanan | |  |
| 1. Koki menolak pesanan karena persediaan pesanan habis | |  |
|  | | 1. Sistem memberi notifikasi bahwa pesanan yang dipesan pelanggan dengan nomor meja tertentu ditolak karena sudah habis |
| 1. Pelanggan menerima pesan bahwa menu yang dipilih ditolak karena persediaan habis | |  |
| **Post Kondisi** | Sistem memberi pilihan apakah akan memilih menu kembali atau tidak | |

Usecase memasak

|  |  |
| --- | --- |
| **Nomor** | 005 |
| **Nama** | Memasak |
| **Tujuan** | Proses ini digunakan untuk koki setelah koki mendapat informasi pemesanan menu pizza dari pelanggan dan koki akan langsung memasak. |
| **Deskripsi** | Usecase ini menggambarkan proses koki memasak setelah koki mendapat informasi pemesanan dari system yang dilakukan oleh pelanggan sesuai nomor meja. |
| **Actor** | Pelanggan dan koki |
| **Prekondisi** | Pelanggan memilih menu |
| **Scenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pelanggan memilih meja yang kosong |  |
| 1. Pelanggan membuka system yang tersedia di meja |  |
| 1. Pelanggan melihat nomor meja | 1.System memberikan nomor meja |
| 1. Pelanggan membuka menu |  |
| 1. Pelanggan melihat informasi dan menu pizza yang tersedia | 2.System menyediakan informasi dan menu pizza |
| 1. Pelanggan menentukan pesanan menu | 3. system menyampaikan informasi ke koki |
| **Scenario alternative: tidak memesan** | |
| 1. Pelanggan membuka sistem | 1.System menyediakan informasi, menu makanan, dan nomor meja |
| 1. Pelanggan tidak memesan | 2.System akan kembali ke semula |
| **Post Kondisi** | System memberi pilihan apakah ingin memesan atau tidak dan akan keluar dari system atau tidak. |

usecase batal pesan

|  |  |
| --- | --- |
| **Nomor** | 006 |
| **Nama** | Batal pesan |
| **Tujuan** | Proses ini digunakan untuk pelanggan membatalkan pemesanan kepada koki. |
| **Deskripsi** | Usecase ini menggambarkan proses pelanggan membatalkan pemesanan dan system akan langsung menyampaikan kepada koki. |
| **Actor** | Pelanggan dan koki |
| **Prekondisi** | Pelanggan membuka system lalu melihat nomor meja dan melihat menu dan informasi yang tersedia dan pelanggan membatalkan pemesanan |
| **Scenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pelanggan memilih meja yang kosong |  |
| 1. Pelanggan membuka system yang tersedia di meja |  |
| 1. Pelanggan melihat nomor meja | 1.System memberikan nomor meja |
| 1. Pelanggan membuka menu |  |
| 1. Pelanggan melihat informasi dan menu pizza yang tersedia | 2.System menyediakan informasi dan menu pizza |
| 1. Pelanggan membatalkan pesanan | 3. System menyampaikan ke koki |
| **Scenario alternative: pesanan sudah dibuat** | |
| 1. Pelanggan membuka sistem | 1.System menyediakan informasi, menu makanan, dan nomor meja |
| 2. Pelanggan membatalkan | 1. System menyampaikan ke koki |
| 3. Koki sudah memasak | 1. System memberi tahu pelanggan |
| **Post Kondisi** | System memberi pilihan apakah ingin memesan atau tidak dan akan keluar dari system atau tidak. |

Usecase tolak pesanan

|  |  |
| --- | --- |
| **Nomor** | 007 |
| **Nama** | Tolak pesanan |
| **Tujuan** | Proses ini digunakan untuk koki menolak pesanan karena menu yang di pesan pelanggan tidak tersedia. |
| **Deskripsi** | Usecase ini menggambarkan proses pelanggan melihat menu pizza kemudian pelanggan memesan pizza, lalu system menyampaikan ke koki dan koki melihat persediaan habis maka koki menolak permintaan pelanggan melalui system dan system menyampaikan kepada pelanggan. |
| **Actor** | Pelanggan dan koki |
| **Prekondisi** | Pelanggan membuka system lalu melihat nomor meja dan melihat menu dan informasi yang tersedia lalu pelanggan memesan kemudian koki menolak |
| **Scenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pelanggan memilih meja yang kosong |  |
| 1. Pelanggan membuka system yang tersedia di meja |  |
| 1. Pelanggan melihat nomor meja | 1.System memberikan nomor meja |
| 1. Pelanggan membuka menu |  |
| 1. Pelanggan melihat informasi dan menu pizza yang tersedia | 2.System menyediakan informasi dan menu pizza |
| 1. Pelanggan memesan menu pizza | 3.System menyampaikan ke koki |
|  | 4.Koki menolak pesanan karena menu tidak tersedia |
| 1. Pelanggan menerima informasi penolakan dan memilih menu lain atau membatalkan | 5.Sistem memberi pilihan menu lainnya |
| **Scenario alternative: tidak memesan** | |
| 1.Pelanggan membuka sistem | 1.System menyediakan informasi, menu makanan, dan nomor meja |
| 2.Pelanggan tidak memesan | 2.System akan kembali ke semula |
| **Post Kondisi** | System memberi pilihan apakah ingin memesan atau tidak dan akan keluar dari system atau tidak. |

Usecase memberi pesanan

|  |  |
| --- | --- |
| **Nomor** | 008 |
| **Nama** | Memberi pesanan |
| **Tujuan** | Proses ini digunakan untuk koki memberikan pesanan menu yang telah dibuat kepada pelayan untuk diantar kepada pelanggan sesuai nomor meja |
| **Deskripsi** | Usecase ini menggambarkan proses koki memberikan pesanan yang telah di masak kepada pelayan untuk diantarkan kepada pelanggan. |
| **Actor** | Pelanggan, koki , dan pelayan |
| **Prekondisi** | Pelanggan menerima pesanan dari pelayan |
| **Scenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pelanggan memilih menu |  |
| 1. Pelanggan memesan menu |  |
| 1. Koki menerima pesanan |  |
| 1. Koki memasak |  |
| 1. Pelayan menerima masakan dari koki |  |
| 1. Pelayan mengantar menu pizza ke pelangga |  |
|  | 1.sistem memberi konfirmasi pesana telah diantar |
| **Scenario alternative: tidak memesan** | |
| 1.Pelanggan membuka sistem | 1.System menyediakan informasi, menu makanan, dan nomor meja |
| 2.Pelanggan tidak memesan | 2.System akan kembali ke semula |
| **Post Kondisi** | System memberi pilihan apakah ingin memesan atau tidak dan akan keluar dari system atau tidak. |

Usecase antar pesanan

|  |  |
| --- | --- |
| **Nomor** | 009 |
| **Nama** | Antar pesanan |
| **Tujuan** | Proses ini digunakan untuk pelayan mengantar pesanan ke pelanggan |
| **Deskripsi** | Usecase ini menggambarkan proses pelayan mengantar pesanan ke pelanggan setelah pelayan menerima masakan dari koki |
| **Actor** | Pelanggan, koki, pelayan |
| **Prekondisi** | Pelanggan menerima pesanan dari pelayan |
| **Scenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1.Pelanggan memesan | 1. system menyampaikan ke koki sesuai nomor meja |
| 2.Pelanggan menunggu pesanan |  |
| 3.koki memasak |  |
| 4.koki memberikan masakan ke pelayan |  |
| 5.pelayan mengantar ke pelanggan | 2. system mengkonfirmasi bahwa pesanan telah diantar |
| 6.pelayan melihat nomor meja |  |
| **Scenario alternative: tidak memesan** | |
| 1.Pelanggan membuka sistem | 1.System menyediakan informasi, menu makanan, dan nomor meja |
| 2.Pelanggan tidak memesan | 2.System akan kembali ke semula |
| **Post Kondisi** | System memberi pilihan apakah ingin memesan atau tidak dan akan keluar dari system atau tidak. |

Usecase hitung total harga

|  |  |
| --- | --- |
| **Nomor** | 010 |
| **Nama** | Hitung total harga |
| **Tujuan** | Proses ini digunakan untuk pelayan menghitung total harga jika pelanggan ingin membayar ke pelayan dan kasir menerima total pesanan dan menghitung total harga jika pelanggan ingin membayar ke kasir |
| **Deskripsi** | Usecase ini menggambarkan proses pelanggan membayar, pelayan akan menghitung semua total biaya, pelanggan juga dapat membayar ke kasir karena kasir juga mendapat konfirmasi total pesanan dari system, pelanggan bebas membayar ke kasir ataupun ke pelayan jika pelanggan membayar ke pelayan maka pelayan harus konfirmasi ke kasir untuk memberi bukti pembayaran kepada pelanggan. |
| **Actor** | Pelanggan, pelayan, kasir |
| **Prekondisi** | Pelanggan membuka system lalu melihat total harga pesanan yang telah di pesan dan membayar ke pelayan atau ke kasir. |
| **Scenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pelanggan menerima pesanan |  |
| 1. Pelanggan telah selesai |  |
| 1. Pelanggan membayar | 1.System memberi informasi total harga |
| 1. Pelanggan membayar ke kasir |  |
| 1. Pelanggan membayar ke pelayan jika tidak ke kasir | 2.System mengkonfirmasi bahwa pelanggan telah membayar |
| **Scenario alternative: tidak memesan** | |
| 1.Pelanggan membuka sistem | 1.System menyediakan informasi, menu makanan, dan nomor meja |
| 2.Pelanggan tidak memesan | 2.System akan kembali ke semula |
| **Post Kondisi** | System memberi pilihan apakah ingin memesan atau tidak dan akan keluar dari system atau tidak. |

Usecase bayar

|  |  |
| --- | --- |
| **Nomor** | 011 |
| **Nama** | Bayar |
| **Tujuan** | Proses ini digunakan untuk pelayan menghitung total harga jika pelanggan ingin membayar ke pelayan dan kasir menerima total pesanan dan menghitung total harga jika pelanggan ingin membayar ke kasir |
| **Deskripsi** | Usecase ini menggambarkan proses pelanggan membayar, pelayan akan menghitung semua total biaya, pelanggan juga dapat membayar ke kasir karena kasir juga mendapat konfirmasi total pesanan dari system, pelanggan bebas membayar ke kasir ataupun ke pelayan jika pelanggan membayar ke pelayan maka pelayan harus konfirmasi ke kasir untuk memberi bukti pembayaran kepada pelanggan. |
| **Actor** | Pelanggan, pelayan, kasir |
| **Prekondisi** | Pelanggan menerima total harga pesanan. |
| **Scenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pelanggan menerima pesanan |  |
| 1. Pelanggan telah selesai |  |
| 1. Pelanggan membayar | 1.System memberi informasi total harga |
| 1. Pelanggan membayar ke kasir |  |
| 1. Pelanggan membayar ke pelayan jika tidak ke kasir | 2.System mengkonfirmasi bahwa pelanggan telah membayar |
| **Scenario alternative: tidak memesan** | |
| 1.Pelanggan membuka sistem | 1.System menyediakan informasi, menu makanan, dan nomor meja |
| 2.Pelanggan tidak memesan | 2.System akan kembali ke semula |
| **Post Kondisi** | System memberi pilihan apakah ingin memesan atau tidak dan akan keluar dari system atau tidak. |

Usecase memberi bukti bayar

|  |  |
| --- | --- |
| **Nomor** | 012 |
| **Nama** | Memberi bukti bayar |
| **Tujuan** | Proses ini digunakan untuk pelanggan mendapat bukti bayar dari kasir atau dari pelayan. |
| **Deskripsi** | Usecase ini menggambarkan proses pelanggan membayar, pelayan akan menghitung semua total biaya, pelanggan juga dapat membayar ke kasir karena kasir juga mendapat konfirmasi total pesanan dari system, pelanggan bebas membayar ke kasir ataupun ke pelayan jika pelanggan membayar ke pelayan maka pelayan harus konfirmasi ke kasir untuk memberi bukti pembayaran kepada pelanggan. |
| **Actor** | Pelanggan, pelayan, kasir |
| **Prekondisi** | Pelanggan membayar ke kasir atau pelayan dan pelanggan mendapatkan bukti pembayaran. |
| **Scenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pelanggan menerima pesanan |  |
| 1. Pelanggan telah selesai |  |
| 1. Pelanggan membayar | 1.System memberi informasi total harga |
| 1. Pelanggan membayar ke kasir |  |
| 1. Pelanggan membayar ke pelayan jika tidak ke kasir | 2.System mengkonfirmasi bahwa pelanggan telah membayar |
| 1. Pelanggan mendapatkan bukti pembayaran | 3.System mengkonfirmasi jika pelanggan telah membayar dan kembali ke menu awal. |
| **Scenario alternative: tidak memesan** | |
| 1.Pelanggan membuka sistem | 1.System menyediakan informasi, menu makanan, dan nomor meja |
| 2.Pelanggan tidak memesan | 2.System akan kembali ke semula |
| **Post Kondisi** | System memberi pilihan apakah ingin memesan atau tidak dan akan keluar dari system atau tidak. |