## Aplikasi Kursus Mengemudi Leadrive

Dosen Pengampu:

## Ir. Paulina Heruningsih Prima Rosa, S.Si., M.Sc.



Oleh:

## Nama / NIM

1. Gerrardo Mayella Ardianta/235314003
2. Benedictus Ditus Atmarestanto/235314007
3. Valentino Banyu Ampar Paraya/235314006
4. Samuel Santoso/23531401
5. Angelo Owada Togovio/235314053

PROGRAM STUDI INFORMATIKA FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS SANATA DHARMA 2025

## Latar Belakang

Dalam era digital yang semakin berkembang, kebutuhan akan layanan yang cepat dan efisien menjadi prioritas bagi masyarakat. Salah satu layanan yang banyak diminati adalah

kursus mengemudi, yang bertujuan untuk membantu individu memperoleh keterampilan berkendara, baik untuk keperluan pribadi maupun profesional. Namun, pencarian kursus mengemudi yang sesuai dengan lokasi, jadwal, serta anggaran sering kali menjadi tantangan bagi calon peserta.

Saat ini, Banyak Client atau calon kursus yang kesulitan mendapatkan informasi seperti penjadwalan kursus, harga, materi, dan lainnya. Sebagian besar kursus mengemudi beroperasi secara konvensional, mengandalkan metode pemasaran offline, seperti brosur, rekomendasi dari mulut ke mulut, atau papan pengumuman. Akibatnya, informasi mengenai layanan kursus tersebar secara tidak terorganisir dan sulit diakses oleh calon peserta.

Berdasarkan permasalahan yang ada, terdapat beberapa alasan utama yang melatarbelakangi perlunya sistem pencarian kursus mengemudi yang lebih modern dan terintegrasi.

* 1. **Kesulitan Akses Informasi** Banyak calon peserta kesulitan menemukan informasi mengenai kursus mengemudi yang sesuai dengan kebutuhan mereka, seperti jadwal yang fleksibel, biaya yang terjangkau, atau lokasi yang dekat dengan tempat tinggal mereka.
  2. **Minimnya Transparansi dan Perbandingan** Saat ini, belum ada platform terpusat yang memungkinkan calon peserta membandingkan berbagai kursus mengemudi berdasarkan harga, kualitas instruktur, atau ulasan dari peserta sebelumnya. Hal ini menyebabkan kesulitan dalam menentukan pilihan yang tepat.
  3. **Keterbatasan Jangkauan Pasar bagi Penyedia Kursus** Banyak penyedia kursus mengemudi mengalami keterbatasan dalam menjangkau calon peserta karena masih mengandalkan metode pemasaran tradisional. Dengan adanya sistem digital, mereka dapat meningkatkan visibilitas dan daya saing di pasar.
  4. **Efisiensi dalam Proses Pendaftaran** Proses pendaftaran kursus mengemudi sering kali masih dilakukan secara manual, yang dapat memakan waktu dan tenaga baik bagi calon peserta maupun penyedia layanan. Dengan sistem berbasis web atau aplikasi,

proses ini dapat dilakukan secara lebih cepat dan praktis.

Dengan adanya sistem pencarian kursus mengemudi yang berbasis web atau aplikasi, pengguna dapat dengan mudah menemukan, membandingkan, dan mendaftar kursus yang sesuai dengan kebutuhan mereka. Selain itu, sistem ini juga dapat membantu meningkatkan transparansi dan kualitas layanan melalui ulasan serta peringkat yang diberikan oleh pengguna. Secara keseluruhan, implementasi sistem ini diharapkan dapat memberikan solusi yang lebih modern, cepat, dan efisien bagi calon peserta serta penyedia kursus mengemudi. Contact person Pak Sabar (Sabar stir mobil): 0857-0108-4000.

## Rumusan Masalah

Berdasarkan latar belakang sebelumnya, masalah-masalah yang akan dijawab dalam laporan ini dapat disajikan sebagai berikut.

* 1. Mengapa sistem pencarian kursus mengemudi perlu memudahkan pengguna dalam mendapatkan informasi harga, ulasan, dan detail lainnya?
  2. Apa keuntungan yang diberikan aplikasi dalam pencarian kursus mengemudi ini?

## Analisis Sebab Akibat

* 1. **Keterkaitan Rumusan Masalah dengan Sebab dan Akibatnya**

Sebab: Calon peserta kursus sering kesulitan mendapatkan informasi yang lengkap dan akurat mengenai kursus mengemudi, seperti harga, ulasan, dan detail lainnya. Hal ini disebabkan oleh kurangnya platform yang menyediakan data terpusat dan terverifikasi.

Akibat:

* + - Calon peserta kursus harus mencari informasi secara manual, yang memakan waktu dan tidak selalu akurat.
    - Kesulitan membandingkan harga dan layanan antar penyedia kursus.
    - Potensi mendapatkan layanan yang tidak sesuai harapan akibat kurangnya transparansi informasi.
    - Pemilik kursus kehilangan peluang mendapatkan lebih banyak klien karena kurangnya visibilitas layanan mereka.

Sebab: Pemilik kursus mengemudi sering mengalami kesulitan dalam menjangkau calon klien dan mengelola pendaftaran secara efisien.

Akibat:

* + - Proses pemasaran menjadi tidak optimal, sehingga jumlah peserta kursus yang mendaftar rendah.
    - Administrasi manual dapat menyebabkan keterlambatan dalam merespons calon klien.
    - Efisiensi operasional berkurang karena tidak ada sistem terpusat untuk mengelola jadwal tutor dan penjemputan klien.

## Tujuan Pengembangan Sistem

Berdasarkan permasalahan yang telah diidentifikasi, sistem pencarian kursus mengemudi ini dikembangkan dengan tujuan:

1. Memudahkan calon peserta kursus dalam mendapatkan informasi yang akurat dan transparan mengenai harga, ulasan, serta detail layanan kursus mengemudi.
2. Meningkatkan efisiensi pemilik kursus dalam mengelola pendaftaran, penjadwalan tutor, serta pemasaran layanan mereka.
3. Menyediakan fitur pencarian berbasis lokasi, sehingga calon peserta dapat menemukan kursus terdekat dengan layanan penjemputan yang sesuai dengan preferensi mereka.
4. Mempercepat proses komunikasi antara calon peserta kursus dan penyedia layanan melalui platform digital yang terintegrasi.

## Kendala (Constraint) yang Dihadapi

Dalam proses pengembangan sistem, beberapa kendala yang mungkin dihadapi antara

lain.

1. Keterbatasan data awal: Tidak semua penyedia kursus memiliki data yang terdigitalisasi, sehingga perlu usaha lebih untuk mengumpulkan dan mengintegrasikan informasi mereka ke dalam sistem.
2. Kepercayaan pengguna: Calon peserta kursus mungkin ragu terhadap keakuratan ulasan dan informasi dalam sistem, sehingga perlu strategi validasi data yang efektif.
3. Ketersediaan tutor dan layanan penjemputan: Sistem harus memastikan bahwa tutor tersedia sesuai dengan permintaan klien, serta mengoptimalkan proses penjadwalan agar tidak terjadi konflik waktu.
4. Biaya pengembangan dan pemeliharaan sistem: Memerlukan sumber daya untuk mengembangkan, mengelola, dan memperbarui sistem agar tetap relevan dan dapat digunakan secara luas.
5. Adopsi teknologi oleh penyedia kursus: Tidak semua penyedia kursus mengemudi terbiasa menggunakan platform digital, sehingga mungkin diperlukan edukasi atau pelatihan agar mereka dapat memanfaatkan sistem dengan maksimal.

Dengan mempertimbangkan sebab-akibat, tujuan, dan kendala yang dihadapi, sistem ini diharapkan dapat memberikan solusi efektif bagi calon peserta kursus maupun pemilik

kursus mengemudi, menciptakan ekosistem yang lebih efisien dan menguntungkan kedua belah pihak.

## Manfaat

Sistem pencarian kursus mengemudi ini memberikan berbagai manfaat bagi pengguna dan penyedia layanan. Berikut adalah manfaat utama yang diperoleh.

* 1. Manfaat bagi Calon Peserta Kursus
     + Kemudahan akses informasi: Pengguna dapat dengan mudah mendapatkan informasi harga, ulasan, dan detail layanan kursus mengemudi secara transparan.
     + Pencarian lebih efisien: Dengan fitur pencarian berbasis lokasi, pengguna dapat menemukan kursus terdekat yang sesuai dengan kebutuhan mereka.
     + Proses pemesanan yang praktis: Sistem memungkinkan calon peserta untuk mendaftar secara online tanpa harus datang langsung ke tempat kursus.
     + Layanan penjemputan yang memudahkan: Tutor dapat langsung menjemput peserta kursus, sehingga menghemat waktu dan tenaga.
     + Kepastian kualitas: Adanya ulasan dari pengguna lain membantu calon peserta memilih kursus yang berkualitas.
  2. Manfaat bagi Pemilik Kursus Mengemudi
     + Meningkatkan jumlah klien: Sistem ini membantu pemilik kursus menjangkau lebih banyak calon peserta melalui platform digital.
     + Pemasaran yang lebih efektif: Dengan sistem yang terintegrasi, pemilik kursus dapat mempromosikan layanan mereka secara lebih luas.
     + Manajemen yang lebih baik: Memudahkan pengelolaan jadwal tutor, pendaftaran peserta, dan layanan penjemputan.
     + Optimasi operasional: Mengurangi kesalahan dalam pencatatan data dan mempermudah komunikasi dengan calon peserta kursus.
  3. Manfaat bagi Tutor atau Instruktur Mengemudi
     + Peluang kerja yang lebih baik: Tutor dapat dengan mudah mendapatkan klien sesuai dengan jadwal yang mereka tentukan.
     + Penjadwalan yang fleksibel: Sistem memungkinkan tutor untuk mengatur jadwal kerja mereka dengan lebih efisien.
     + Transparansi dalam pembayaran: Sistem dapat mencatat setiap transaksi dan memastikan tutor mendapatkan pembayaran secara adil.
  4. Manfaat bagi Masyarakat Umum
     + Meningkatkan keselamatan berkendara: Dengan lebih banyak orang mendapatkan

akses ke kursus mengemudi yang berkualitas, diharapkan keterampilan mengemudi di masyarakat meningkat dan mengurangi angka kecelakaan.

* + - Mendukung transformasi digital: Sistem ini mendorong industri kursus mengemudi untuk lebih digital dan mengikuti perkembangan teknologi.

## Ruang Lingkup (Batasan Masalah)

* 1. **Batasan Sistem yang Dirancang**

Sistem ini dirancang untuk membantu pencarian dan manajemen kursus mengemudi dengan fitur utama sebagai berikut.

Termasuk dalam Sistem

* + - Pendaftaran pengguna (calon peserta kursus dan pemilik kursus): Pengguna dapat membuat akun dan mengelola profil mereka.
    - Pencarian kursus berbasis lokasi: Sistem menampilkan daftar kursus mengemudi berdasarkan lokasi pengguna.
    - Tampilan informasi kursus: Pengguna dapat melihat harga, ulasan, fasilitas, dan detail kursus lainnya.
    - Pemesanan kursus dan penjadwalan tutor: Calon peserta dapat memilih kursus dan menjadwalkan sesi belajar.
    - Layanan penjemputan peserta: Tutor atau pihak kursus dapat menjemput peserta sesuai jadwal yang ditentukan.
    - Sistem ulasan dan rating: Pengguna dapat memberikan ulasan dan rating terhadap kursus yang mereka ikuti.
    - Sistem pembayaran online: Pembayaran kursus dapat dilakukan melalui sistem (jika diimplementasikan).
    - Layanan klaim atau refund manual: Jika ada permasalahan terkait pembayaran, penyelesaian dilakukan langsung oleh penyedia kursus.

Tidak Termasuk dalam Sistem

* + - Pembuatan materi pembelajaran: Sistem hanya menghubungkan peserta dengan kursus, tidak menyediakan materi belajar.
    - Pelaksanaan ujian atau sertifikasi SIM: Sistem tidak berperan dalam ujian resmi mendapatkan Surat Izin Mengemudi (SIM).
    - Pengelolaan kendaraan oleh pemilik kursus: Sistem tidak mencakup manajemen kendaraan yang digunakan untuk pelatihan.

Entitas Eksternal:

* + - Calon Peserta Kursus → Mengakses sistem untuk mencari kursus, melihat informasi, mendaftar, membayar, dan memberikan ulasan.
    - Pemilik Kursus Mengemudi → Mendaftarkan layanan mereka, mengelola peserta, menyediakan tutor, dan menerima pembayaran.
    - Tutor (Instruktur) → Menerima jadwal dari sistem, menjemput peserta, dan memberikan pelatihan mengemudi.
    - Sistem Pembayaran → Memproses transaksi kursus antara peserta dan pemilik kursus.

Proses Utama dalam Sistem.

1. Pendaftaran pengguna dan pemilik kursus
2. Pencarian dan pemesanan kursus
3. Penjadwalan sesi mengemudi & tutor
4. Proses pembayaran (jika ada)
5. Layanan penjemputan peserta
6. Ulasan dan rating

Diagram Konteks untuk sistem ini.

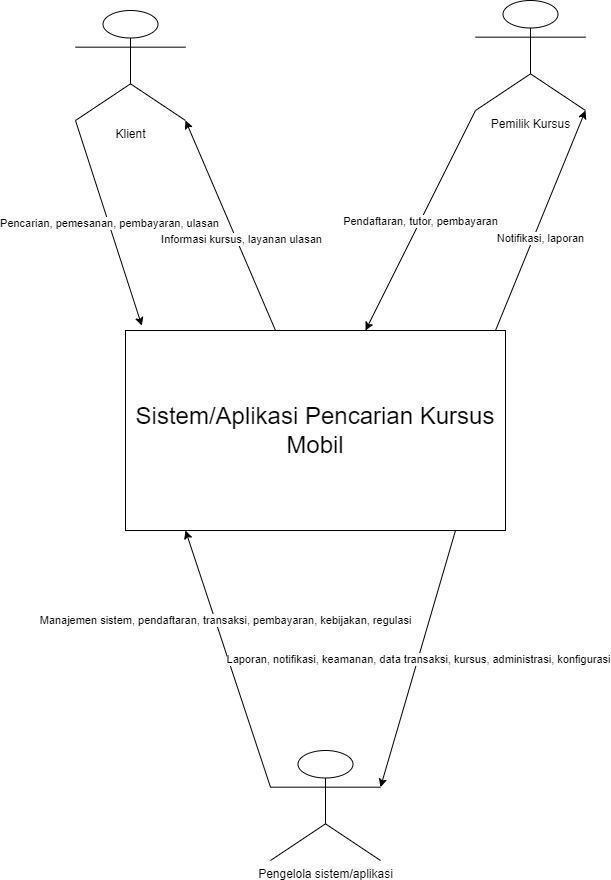


Diagram ini menunjukkan bagaimana sistem menghubungkan berbagai pihak yang terlibat dalam proses pencarian dan pelaksanaan kursus mengemudi.

Proyek sistem pencarian kursus mengemudi melibatkan beberapa pemangku kepentingan utama yang berperan dalam pengembangan dan operasional sistem. Berikut adalah stakeholder beserta perannya:

1. **Pemangku Kepentingan** (Stakeholders)
   1. Calon Peserta Kursus (End User)
      * Menggunakan sistem untuk mencari, membandingkan, dan memesan kursus mengemudi.
      * Memberikan ulasan dan rating terhadap kursus yang telah mereka ikuti
   2. Pemilik Kursus Mengemudi (Service Provider)
      * Mendaftarkan kursus mereka ke dalam sistem.
      * Mengelola jadwal kursus, tutor, dan peserta.
      * Menerima pembayaran dari peserta melalui sistem.
      * Manajemen kendaraan
      * Promosi dan pemasaran.
      * Tantangan operasional
      * Harapan terhadap sistem digital
   3. Tutor/Instruktur Mengemudi
      * Memberikan layanan pelatihan mengemudi kepada peserta.
      * Menjemput peserta sesuai dengan jadwal yang telah ditentukan dalam sistem.
   4. Pengembang Sistem (Developer/IT Team)
      * Mengembangkan dan memelihara sistem agar tetap berjalan optimal.
      * Menambahkan fitur sesuai dengan kebutuhan pengguna.
      * Mengatasi permasalahan teknis yang muncul dalam sistem.

# GAMBARAN UMUM PERANGKAT LUNAK

1. **IDENTITAS DAN TUJUAN PERANGKAT LUNAK:**

Nama Perangkat Lunak: Sistem Pencarian Kursus Mengemudi (LeaDrive) Tujuan Pembangunan:

1. Mempermudah calon peserta kursus dalam memperoleh informasi harga, ulasan, dan detail layanan kursus mengemudi.
2. Meningkatkan efisiensi pengelolaan pendaftaran dan jadwal tutor bagi pemilik kursus.
3. Memberikan platform terpusat untuk membandingkan berbagai kursus mengemudi.

# FUNGSI PRODUK PERANGKAT LUNAK

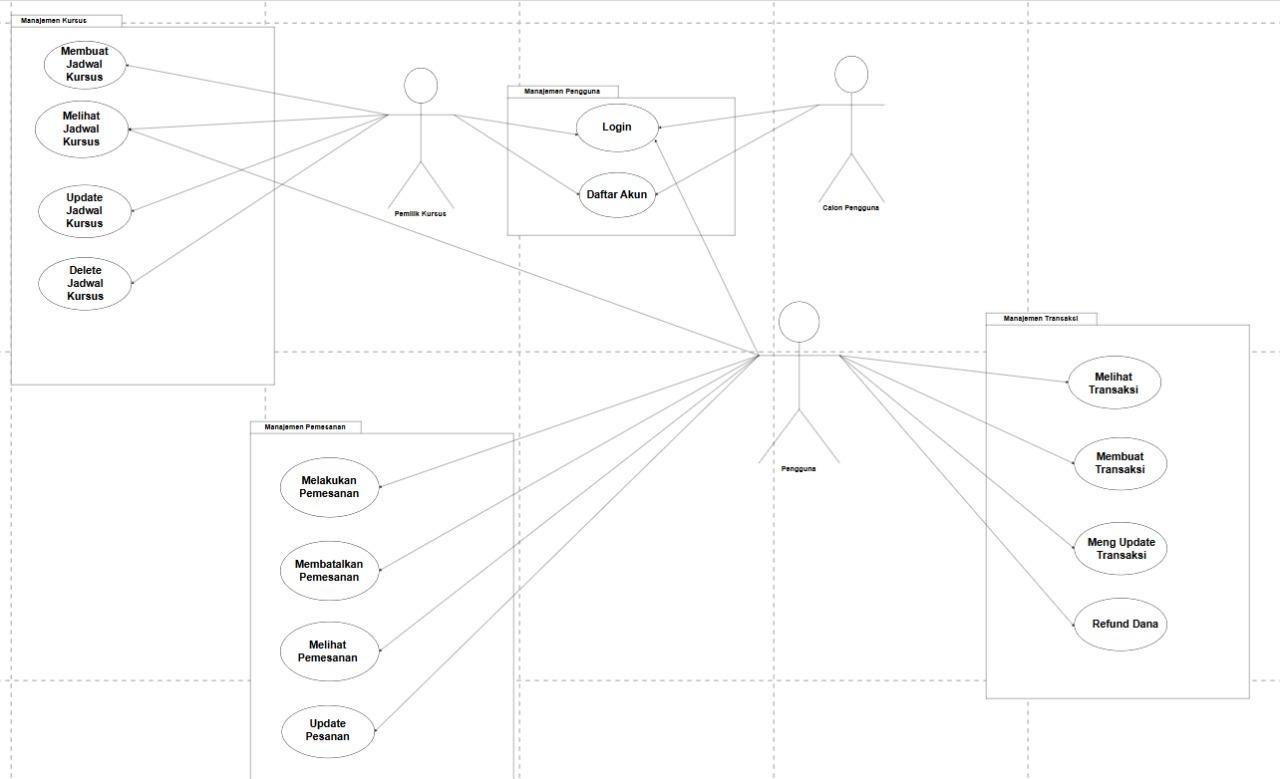
1. Pencarian kursus berbasis lokasi.
2. Penjadwalan sesi belajar dan layanan penjemputan.
3. Sistem ulasan dan rating.
4. Proses pembayaran online.

# IDENTIFIKASI PENGGUNA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Pengguna | Tanggung Jawab | Hak Akses | Tingkat Pendidikan | Tingkat Keterampilan | Pengalaman | Jenis Pelatihan |
| Calon Peserta Kursus | Mencari dan melihat tempat kursus | Akses penuh informasi | SMP SMA  Universitas | Dasar | Tidak perlu | - |
| Peserta Kursus | Mencari dan mendaftar kursus | Akses penuh informasi dan pembayaran | SMP SMA  Universitas | Dasar | Tidak perlu | - |
| Pemilik Kursus | Mengelola data kursus dan peserta | Admin penuh | SMA/Universitas | Menengah | Pengalaman mengelola kursus | Pelatihan manajemen platform |
| Tutor/Instruktur | Memberikan layanan pelatihan | Akses jadwal dan data peserta | SMA | Tinggi | Minimal 2 tahun mengajar | Pelatihan penggunaan sistem |

1. **MODEL FUNGSI PERANGKAT LUNAK**

**Diagram Use Case**



* 1. **Narasi Use Case**

**LeadDrive (Sistem Pencarian Kursus Mobil)**

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Login | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-LG-001 |
| **Prioritas:** | High |
| **Sumber:** | Requirement-LD-LG-001 |

|  |  |
| --- | --- |
| **Aktor**  **Utama Bisnis:** | User |
| **Akor Partisipasi Lainnya:** | Pemilik Kursus |
| **Pemangku Kepentingan Lainnya:** | **-** |
| **Deskripsi:** | Use Case ini mendeskripsikan kejadian ketika user dan pemilik kursus akan melakukan login, ketika ingin mengakses aplikasi |

|  |  |  |
| --- | --- | --- |
| **Kondisi Awal:** | User dan Pemilik kursus berada pada halaman login | |
| **Pemicu:** | User dan pemilik kursus ingin melakukan login ke aplikasi | |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |
|  | Step 1: Pengguna  membuka halaman login pada aplikasi | Step 2: Sistem kemudian merespon dengan menampilkan form untuk login |

|  |  |  |
| --- | --- | --- |
|  | Step 3: User dan pemilik kursus memasukan data pada mengisi form yang disediakan | Step 4: Sistem melakukan verifikasi data yang diberikan |
|  | Step 5: Sistem akan menampilkan pesan “Login Success” mengizinkan pengguna untuk mengakses aplikasi dan menampilkan halaman utama |
| **Skenario Alternatif:** | Alternatif 5 : Bila input data tidak valid akan ada peringatan untuk user dan Pemilik kursus dan diminta mengisi dan menginput ulang data | |

|  |  |
| --- | --- |
| **Kesimpulan:** | Use case berhasil apabila sistem telah memberitahukan kepada member bahwa telah berhasil login ke sistem |
| **Kondisi Akhir:** | Kondisi Akhir (Jika Login Berhasil): User telah teridentifikasi Kondisi Akhir (Jika Login Gagal): User tidak teridentifikasi |

|  |  |
| --- | --- |
| **Aturan Bisnis:** | User hanya dapat melakukan login dengan mengisikan username dan password yang valid dan hanya teridentifikasi sebagai pelanggan |
| **Kendala Implementasi dan Spesifikasi:** | **-** |
| **Tanggapan:** | **-** |
| **Masalah Terbuka:** | **-** |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Daftar Akun | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-DA-002 |
| **Prioritas:** | High |
| **Sumber:** | Requirement-LD-DA-002 |

|  |  |
| --- | --- |
| **Aktor UtamaBisnis:** | Calon User |
| **Akor Partisipasi Lainnya:** | Pemilik Kursus |
| **Pemangku Kepentingan Lainnya:** | - |
| **Deskripsi:** | Use Case ini menggambarkan proses untuk mendaftar sebagai anggota/user dalam sebuah sistem atau aplikasi. |
| **Kondisi Awal:** | Pelanggan berada di halaman registrasi. |

|  |  |  |
| --- | --- | --- |
| **Pemicu:** | Pendaftar akan membuat akun agar bisa menggunakan aplikasi. | |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |
|  | Step 1: Pendaftar membuka halaman registrasi pada aplikasi Lead-Drive | Step 2: Sistem menampilkan halaman registrasi yang berisi form dengan informasi yang harus diisi oleh customer |

|  |  |  |
| --- | --- | --- |
|  | Step 3: Pendaftar mengisi form dengan informasi yang diperlukan. | Step 4: Sistem memvalidasi informasi yang diisi oleh pendaftar |
|  | Step 5: Sistem menyimpan data pendaftar dalam database dan mengirimkan email konfirmasi ke alamat email yang diberikan oleh pendaftar dan menampilkan pesan “Daftar Akun Berhasil” |
| **Skenario Alternatif:** | Alternatif 5: Sistem menampilkan pesan error jika ada informasi yang tidak valid atau jika pengguna tidak mengisi semua informasi yang diperlukan | |

|  |  |
| --- | --- |
| **Kesimpulan:** | Pendaftar akan menerima konfirmasi dari system mengenai akun yang telah di registrasi |
| **Kondisi Akhir:** | Kondisi Akhir (Jika Login Berhasil): Sistem memberikan akses ke pelanggan dan pelanggan dapat mengakses fitur-fitur yang tersedia  Kondisi Akhir (Jika Login Gagal): Sistem tidak memberikan akses ke pelanggan dan pelanggan tidak dapat mengakses fitur-fitur yang tersedia |

|  |  |
| --- | --- |
| **Aturan Bisnis:** | Pendaftar harus memberikan informasi pribadi yang valid sesuai dengan kebijakan privasi aplikasi. Ini memastikan bahwa data yang masuk akurat dan berguna |
| **Kendala Implementasi dan Spesifikasi:** | - |
| **Tanggapan:** | - |
| **Masalah Terbuka:** | - |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Membuat Jadwal Kursus | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-MJ-L003 |
| **Prioritas:** | High |
| **Sumber:** | Requirement-LD-MJ-L003 |
| **Aktor**  **Utama Bisnis:** | Pemilik Kursus | |

|  |  |
| --- | --- |
| **Akor Partisipasi Lainnya:** | - |
| **Pemangku Kepentingan Lainnya:** | - |
| **Deskripsi:** | Use case ini memungkinkan pemilik kursus untuk membuat jadwal kursus sesuai dengan kebutuhan mereka |
| **Kondisi Awal:** | Pemilik kursus berada di halaman membuat jadwal |

|  |  |  |
| --- | --- | --- |
| **Pemicu:** | Pemilik kursus ingin membuat jadwal | |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |
|  | Step 1: Pemilik kursus mengklik “Tambah Jadwal” | Step 2: Aplikasi menampilkan layar berisi fitur untuk pemilik kursus mengetikkan jadwal yang akan dimasukkan |

|  |  |  |
| --- | --- | --- |
|  | Step 3: Pemilik kursus memasukkan jadwal dan mengklik “Tambah Jadwal” | Step 4: Sistem menyimpan data jadwal ke database dan menampilkan pesan “Jadwal Berhasil Ditambahkan”. |
| **Skenario Alternatif:** | Alternatif 3: Saat pemilik kursus ingin melengkapi form pembuatan jadwal, jika tidak lengkap akan menampilkan pesan, “Form Belum Berhasil Disimpan” karena ada  beberapa yang harus dilengkapi | |
| **Kesimpulan:** | Pemilik kursus dapat memilih waktu jadwal kursus yang ia inginkan untuk disediakan ke para pelanggan kursus | |
| **Kondisi Akhir:** | Kondisi akhir yaitu pemilik kursus sudah membuat jadwal kursus | |

|  |  |
| --- | --- |
|  |  |
| **Aturan Bisnis:** | Aplikasi harus memastikan bahwa hanya pemilik kursus yang dapat membuat jadwal kursusnya |

|  |  |
| --- | --- |
| **Kendala Implementasi dan Spesifikasi:** | - |
| **Tanggapan:** | - |
| **Masalah Terbuka:** | - |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Melihat Jadwal | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-LJ-004 |
| **Prioritas:** | Medium |
| **Sumber:** | Requirement-LD-LJ-004 |
| **Aktor**  **Utama Bisnis:** | User | |
| **Akor Partisipasi Lainnya:** | Pemilik Kursus | |

|  |  |
| --- | --- |
| **Pemangku Kepentingan Lainnya:** | - |
| **Deskripsi:** | Pada use case ini user dan pemilik kursus dapat melihat Jadwal yang tersedia |
| **Kondisi Awal:** | User dan pemilik kursus berada di dasboard awal |
| **Pemicu:** | User dan pemilik kursus memilih melihat jadwal yang tersedia |

|  |  |  |
| --- | --- | --- |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |
|  | Step 1:  User dan pemilik kursus memilih “Melihat  Jadwal” | Step 2:  Sistem menampilkan jadwal yang tersedia |

|  |  |
| --- | --- |
| **Skenario Alternatif:** | - |
| **Kesimpulan:** | Use case ini berfungsi untuk melihat jadwal yang tersedia |
| **Kondisi Akhir:** | Setelah melihat jadwal user dan prmilik bisa melihat jadwal yang tersedia |

|  |  |
| --- | --- |
| **Aturan Bisnis:** | Sebelum melihat jadwal user dan pemilik kursus harus login terlebih dahulu |
| **Kendala Implementasi dan Spesifikasi:** | - |
| **Tanggapan:** | - |
| **Masalah Terbuka:** | - |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Update Jadwal Kursus | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-UJ-005 |
| **Prioritas:** | Medium |
| **Sumber:** | Requirement-LD-UJ-005 |
| **Aktor**  **Utama Bisnis:** | Pemilik Kursus | |
| **Akor Partisipasi Lainnya:** | - | |
| **Pemangku Kepentingan Lainnya:** | - | |
| **Deskripsi:** | Pada use case ini pemilik kursus dapat memodifikasi / mengubah jadwal awal | |
| **Kondisi Awal:** | Pemilik kursus sudah membuat jadwal kursus | |

|  |  |
| --- | --- |
| **Pemicu:** | Klik menu “Update Jadwal Kursus” |

|  |  |  |
| --- | --- | --- |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |
|  | Step 1: | Step 2: |
| Pemilik kursus mengklik “Update Jadwal Kursus” | Menampilkan form jadwal yang telah dibuat sebelumnya |
| Step 3:  Pemilik kursus mengedit  form jadwal dan klik | Step 4:  Menyimpan data jadwal yang terbaru ke database dan menampilkan pesan “Jadwal berhasil diedit” |
| “Update Perubahan” |  |
| **Skenario Alternatif:** | - | |

|  |  |
| --- | --- |
| **Kesimpulan:** | Use case ini berfungsi untuk mengupdate jadwal kursus oleh pemilik |
| **Kondisi Akhir:** | Setelah melakukan update jadwal, pemilik kursus akan mendapatkan perubahan jadwal terbaru |

|  |  |
| --- | --- |
| **Aturan Bisnis:** | Sebelum melakukan pemesanan pemilik kursus harus melakukan login terlebih dahulu dan punya jadwal awal |
| **Kendala Implementasi dan Spesifikasi:** | - |
| **Tanggapan:** | - |
| **Masalah Terbuka:** |  |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Delete Jadwal Kursus | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-DJ-006 |
| **Prioritas:** | Medium |

|  |  |  |
| --- | --- | --- |
| **Sumber:** | Requirement-LD-DJ-L006 |  |

|  |  |  |
| --- | --- | --- |
| **Aktor**  **Utama Bisnis:** | Pemilik Kursus | |
| **Akor Partisipasi Lainnya:** | - | |
| **Pemangku Kepentingan Lainnya:** | - | |
| **Deskripsi:** | Use case ini berfungsi untuk melakukan hapus jadwal yang sudah dibuat | |
| **Kondisi Awal:** | Pemilik kursus sudah login untuk menghapus jadwal | |
| **Pemicu:** | Pemilik kursus menghapus jadwal | |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |

|  |  |  |
| --- | --- | --- |
|  | Step 1 :  Pemilik kursus mengklik tombol “Hapus Jadwal” | Step 2:  Menampilkan form jadwal yang telah dibuat sebelumnya |

|  |  |  |
| --- | --- | --- |
|  | Step 3:  Pemilik kursus harus memilih jadwal untuk di hapus | Step 4:  Sistem menghapus jadwal dari database dan menampilkan pesan “Jadwal Berhasil Dihapus” |
| **Skenario Alternatif:** | Alternatif 3: Jika pemilik belum memilih jadwal yang dihapus, maka akan menampilkan pesan “Silahkan Pilih Jadwal yang Ingin dihapus Dahulu” | |
| **Kesimpulan:** | Use case ini berfungsi untuk menghapus jadwal | |
| **Kondisi Akhir:** | Jadwal kursus yang dipilih telah berhasil dihapus dari sistem dan tidak lagi ditampilkan di daftar jadwal pengguna | |

|  |  |
| --- | --- |
| **Aturan Bisnis:** | Sebelum menghapus jadwal aktor harus memilih jadwal terlebih dahulu |
| **Kendala Implementasi dan Spesifikasi:** | - |
| **Tanggapan:** | - |

|  |  |
| --- | --- |
| **Masalah Terbuka:** | - |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Melakukan Pemesanan | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-MP-007 |
| **Prioritas:** | High |
| **Sumber:** | Requirement-LD-MP-L007 |
| **Aktor**  **Utama Bisnis:** | User | |

|  |  |
| --- | --- |
| **Akor Partisipasi Lainnya:** | - |
| **Pemangku Kepentingan Lainnya:** | - |
| **Deskripsi:** | Pada use case ini user dapat melakukan pemesanan booking jadwal latihan |
| **Kondisi Awal:** | User telah berhasil login dan berada di halaman daftar kursus yang tersedia untuk dipesan |
| **Pemicu:** | User menekan tombol “Pesan Kursus” pada salah satu kursus yang tersedia |

|  |  |  |
| --- | --- | --- |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |
|  | Step 1:  User mengklik tombol “Pesan Kursus” | Step 2:  Sistem menampilkan form pemesanan kursus, termasuk detail kursus dan jadwal kursus |

|  |  |  |
| --- | --- | --- |
|  | Step 3:  User mengisi form pemesanan  Step 5:  User mengklik tombol  konfirmasi pemesanan | Step 4:  Sistem memverifikasi data yang diinput  Step 6:  Sistem menyimpan data pemesanan ke database dan menampilkan pesan “Pemesanan Berhasil Dilakukan” |
| **Skenario Alternatif:** | Alternatif 3: Ketika user tidak melengkapi form pemesanan, maka sistem menampilkan pesan “Data yang Anda Masukkan Belum Lengkap” | |
| **Kesimpulan:** | Use case ini berfungsi untuk membuat pesanan user | |
| **Kondisi Akhir:** | User memiliki pesanan baru (jadwal yang dipesan) | |

|  |  |
| --- | --- |
| **Aturan Bisnis:** | Sebelum memesan userr harus login terlebih dahulu |

|  |  |
| --- | --- |
| **Kendala Implementasi dan Spesifikasi:** | - |
| **Tanggapan:** | - |
| **Masalah Terbuka:** | - |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Membatalkan Pesanan | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-CP-008 |
| **Prioritas:** | High |
| **Sumber:** | Requirement-LD-CP-008 |

|  |  |
| --- | --- |
| **Aktor**  **Utama Bisnis:** | User |
| **Akor Partisipasi Lainnya:** | - |

|  |  |  |
| --- | --- | --- |
| **Pemangku Kepentingan Lainnya:** | - | |
| **Deskripsi:** | Use case ini berfungsi untuk membatalkan pesanan | |
| **Kondisi Awal:** | User sudah punya pesanan | |
| **Pemicu:** | User membatalkan pesanan | |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |
|  | Step 1 :  User memilih menu pesanan | Step 2:  Sistem menampilkan form pemesanan |

|  |  |  |
| --- | --- | --- |
|  | Step 3:  User memilih kursus | Step 4:  Sistem menampilkan detail kursus |
|  | Step 5:  User memilih “Hapus Pemesanan” | Step 6:  Sistem menghapus pemesanan dari database dan menampilkan pesan “Pesanan Berhasil Dihapus” |
| **Skenario Alternatif:** | Alternatif 3: Ketika user tidak memilih kursus yang ingin dihapus, sistem menampilkan “Pilih Kursus yang Ingin Dihapus” | |
| **Kesimpulan:** | Use case ini berfungsi untuk membatalkan pemesanan | |
| **Kondisi Akhir:** | Pesanan berhasil dibatalkan dari sistem dan tidak lagi melihat pemesanan | |

|  |  |
| --- | --- |
| **Aturan Bisnis:** | Sebelum membatalkan pesanan user harus memilih melakukan pemesanan terlebih dahulu |
| **Kendala Implementasi dan Spesifikasi:** | - |
| **Tanggapan:** | - |

|  |  |
| --- | --- |
| **Masalah Terbuka:** | - |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Melihat Pemesanan | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-SP-009 |
| **Prioritas:** | Medium |
| **Sumber:** | Requirement-LD-SP-009 |
|  | User | |

|  |  |
| --- | --- |
| **Aktor**  **Utama Bisnis:** |  |
| **Akor Partisipasi Lainnya:** | - |
| **Pemangku Kepentingan Lainnya:** | - |
| **Deskripsi:** | Pada use case ini user dapat melihat pemesanan |
| **Kondisi Awal:** | User telah melakukan pemesanan |
| **Pemicu:** | User menekan tombol lihat pemesanan |

|  |  |  |
| --- | --- | --- |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |
|  | Step 1:  User mengklik tombol “Melihat pemesanan” | Step 2:  Sistem menampilkan form pemesanan yang tersedia |

|  |  |  |
| --- | --- | --- |
|  | Step 3:  User klik tombol “Kembali” | Step 4:  Sistem mengarahkan ke dasboard utama |
| **Skenario Alternatif:** | - | |
| **Kesimpulan:** | Use case ini berfungsi untuk melihat pemesanan | |
| **Kondisi Akhir:** | Setelah melihat pemesanan user akan kembali ke halaman utama | |

|  |  |
| --- | --- |
| **Aturan Bisnis:** | Sebelum melihat pemesanan user harus melakukan pemesanan |

|  |  |
| --- | --- |
| **Kendala Implementasi dan Spesifikasi:** | - |
| **Tanggapan:** | - |
| **Masalah Terbuka:** |  |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Update Pemesanan | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-UPM-010 |
| **Prioritas:** | Medium |
| **Sumber:** | Requirement-LD-SP-010 |

|  |  |
| --- | --- |
| **Aktor**  **Utama Bisnis:** | User |
| **Akor Partisipasi Lainnya:** | - |

|  |  |  |
| --- | --- | --- |
| **Pemangku Kepentingan Lainnya:** | - | |
| **Deskripsi:** | Use case ini berfungsi untuk mengupdate pemesanan | |
| **Kondisi Awal:** | User sudah melakukan pemesanan untuk mengupdate pemesanan | |
| **Pemicu:** | User mengupdate pemesanan | |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |
|  | Step 1 :  User memilih “Edit Pemesanan” | Step 2:  Sistem menampilkan daftar pemesanan |

|  |  |  |
| --- | --- | --- |
|  | Step 3:  User memilih pesanan yang ingin di update | Step 4:  Sistem menampilkan form pemesanan |
|  | Step 5:  User mengubah informasi pemesanan dan klik tombol “Update Pesanan” | Step 6:  Sistem menyimpan dan merubah pesana, serta menampilkan “Pesanan telah Diupdate” |
| **Skenario Alternatif:** | Alternatif 5: Ketika user ingin melengkapi dan memperbaharui pesanan secara tidak lengkap, maka sistem akan menampilkan “Data Belum Berhasil Diupdate Karena Belum Lengkap” | |
| **Kesimpulan:** | Use case ini berfungsi untuk mengupdate pemesanan | |
| **Kondisi Akhir:** | Pesanan yang dipilih telah berhasil di update dari sistem | |

|  |  |
| --- | --- |
| **Aturan Bisnis:** | Sebelum mengupdate pesanan user harus melakukan pemesanan terlebih dahulu |
| **Kendala Implementasi dan Spesifikasi:** | - |
| **Tanggapan:** | - |

|  |  |
| --- | --- |
| **Masalah Terbuka:** | - |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Melihat Transaksi | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-LT-011 |
| **Prioritas:** | Medium |
| **Sumber:** | Requirement-LD-LT-011 |
| **Aktor**  **Utama Bisnis:** | User | |

|  |  |
| --- | --- |
| **Akor Partisipasi Lainnya:** | Pemilik Kursus |
| **Pemangku Kepentingan Lainnya:** | - |
| **Deskripsi:** | Pada use case ini user dan pemilik dapat melihat transaksi |
| **Kondisi Awal:** | User telah membuat transaksi |
| **Pemicu:** | User dan pemilik kursus memilih melihat transaksi |

|  |  |  |
| --- | --- | --- |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |
|  | Step 1:  User memilih “Melihat Transaksi” | Step 2:  Sistem menampilkan form melihat transaksi |

|  |  |
| --- | --- |
| **Skenario Alternatif:** | - |
| **Kesimpulan:** | Use case ini berfungsi untuk melihat transaksi |
| **Kondisi Akhir:** | Setelah melihat transaksi user kembali ke halaman utama |

|  |  |
| --- | --- |
| **Aturan Bisnis:** | Sebelum melihat transaksi user harus membuat transaksi |
| **Kendala Implementasi dan Spesifikasi:** | - |
| **Tanggapan:** | - |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Membuat Transaksi | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-MMT-012 |
| **Prioritas:** | High |
| **Sumber:** | Requirement-LD-MMT-012 |

|  |  |
| --- | --- |
| **Aktor**  **Utama Bisnis:** | User |
| **Akor Partisipasi Lainnya:** | - |
| **Pemangku Kepentingan Lainnya:** | - |
| **Deskripsi:** | Use case ini berfungsi untuk membuat transaksi |
| **Kondisi Awal:** | User sudah login untuk melakukan transaksi |

|  |  |  |
| --- | --- | --- |
| **Pemicu:** | User memilih pilih pesanan umtuk membuat transaksi | |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |
|  | Step 1 :  User mengklik “Pilih Pesanan Anda” | Step 2:  Menampilkan metode pembayaran yang tersedia |

|  |  |  |
| --- | --- | --- |
|  | Step 3:  User memilih metode pembayaran |  |
|  | Step 4:  User memasukkan detail pembayaran |  |
|  | Step 5:  User menekan tombol bayar | Step 6:  Sistem memproses transaksi selama 24 jam. Jika pembayaran tidak dilakukan maka pesanan akan dibatalkan.  Step 7:  Selanjutnya mengkonfirmasi pembayaran, menyimpan riwayat transaksi ke database, dan menampilkan pesan “Transaksi Anda Berhasil” |

|  |  |
| --- | --- |
| **Skenario Alternatif:** | - |
| **Kesimpulan:** | Use case ini berfungsi untuk membuat transaksi |
| **Kondisi Akhir:** | Transaksi berhasil disimpan |
| **Aturan Bisnis:** | Sebelum memilih tombol bayar, user memasukkan detail pembayaran |
| **Kendala Implementasi dan Spesifikasi:** | - |
| **Tanggapan:** | - |

|  |  |
| --- | --- |
| **Masalah Terbuka:** | - |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Update Transaksi | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-UT-013 |
| **Prioritas:** | Medium |
| **Sumber:** | Requirement-LD-UT-013 |
| **Aktor**  **Utama Bisnis:** | User | |
| **Akor Partisipasi Lainnya:** | - | |
| **Pemangku Kepentingan Lainnya:** | - | |
| **Deskripsi:** | Pada use case ini user dapat melakukan update transaksi | |
| **Kondisi Awal:** | User telah membuat transaksi dan berada di halaman melihat transaksi | |
| **Pemicu:** | User menekan tombol pesan update transaksi | |

|  |  |  |
| --- | --- | --- |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |
|  | Step 1: | Step 2: |
| User mengklik tombol “Update Transaksi” | Sistem menampilkan form transaksi |
| Step 3:  User mengupdate  transaksi dan | Step 4:  Menyimpan transaksi yang terbaru ke database dan menampilkan pesan “transaksi berhasil diupdate” |
| mengkonfirmasi transaksi |  |
| **Skenario Alternatif:** | - | |
| **Kesimpulan**: | Use case ini berfungsi untuk melakukan update transaksi | |

|  |  |
| --- | --- |
| **Kondisi Akhir:** | Setelah melakukan update transaksi user akan diarahkan ke tampilan melihat transaksi |

|  |  |
| --- | --- |
| **Aturan Bisnis:** | Sebelum melakukan update user harus membuat transaksi |
| **Kendala Implementasi dan Spesifikasi:** | - |
| **Tanggapan:** | - |
| **Masalah Terbuka:** |  |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Refund Dana | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-RD-014 |
| **Prioritas:** | Medium |
| **Sumber:** | Requirement-LD-RD-014 |

|  |  |  |
| --- | --- | --- |
| **Aktor Utama Bisnis:** | User | |
| **Aktor Partisipasi Lainnya:** | - | |
| **Pemangku Kepentingan Lainnya:** | - | |
| **Deskripsi:** | Use case ini berguna untuk melakukan pengembalian dana kepada pelanggan, dari kursus yang sudah dipesan namun dibatalkan | |
| **Kondisi Awal:** | User sudah memesan kursus dan sudah melakukan transaksi | |
| **Pemicu:** | User melakukan refund | |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |

|  |  |  |
| --- | --- | --- |
|  | Step 1 :  User memilih menu pemesanan | Step 2:  Sistem menampilkan daftar pemesanan yang sudah dibayar |

|  |  |  |
| --- | --- | --- |
|  | Step 3:  User memilih tab pesanan yang ada dan ingin direfund | Step 4:  Sistem menampilkan detail pemesanan yang dipilih |
|  | Step 5:  User menekan tombol “Refund” | Step 6:  Sistem menampilkan pesan “Pengajuan Refund Berhasil! Dana Kembali Sebanyak 50%” |
| **Skenario Alternatif:** | - | |

|  |  |
| --- | --- |
| **Kesimpulan:** | Use case ini berfungsi untuk membatalkan pesanan serta merefund dana pelanggan yang sudah dibuat dan dibayar |
| **Kondisi Akhir:** | Pesanan yang direfund dihapus dari daftar pemesanan |
| **Aturan Bisnis:** | Refund dana dan pembatalan ini memiliki tenggat waktu, yaitu 24 jam. ketika pemesanan sudah dibuat dan dibayar lewat dari 24 jam maka pesanan tidak bisa direfund. Refund dana hanya bisa 50 % dari total dana yang sudah dibayarkan pelanggan kursus. |
| **Kendala Implemen tasi dan Spesifikasi**  **:** | - |
| **Tanggapan:** | - |

|  |  |
| --- | --- |
| **Masalah Terbuka:** | - |

|  |  |  |
| --- | --- | --- |
| **Nama Use-Case:** | Tutor | Use-Case Type "System Requirements" |
| **ID Use-Case:** | LD-TT-001 |
| **Prioritas:** | High |
| **Sumber:** | Requirement-LD-TT-0015 |

|  |  |
| --- | --- |
| **Aktor**  **Utama Bisnis:** | Tutor |
| **Akor Partisipasi Lainnya:** | - |
| **Pemangku Kepentingan Lainnya:** | Pemilik Kursus, Pengguna |
| **Deskripsi:** | Use case ini menggambarkan interaksi Tutor dalam mengelola program pembelajaran pada aplikasi, seperti membuat program, memperbarui detail program, dan melihat data transaksi yang berkaitan dengan program tersebut. |

|  |  |  |
| --- | --- | --- |
| **Kondisi Awal:** | Tutor telah login ke dalam aplikasi dan berada pada dashboard akun tutor. | |
| **Pemicu:** | Tutor ingin mengelola konten atau transaksi dari program kursus yang dibuat/diajar. | |
| **Skenario Utama:** | Aksi Aktor | Reaksi Sistem |
|  | Step 1: Tutor membuka halaman login pada aplikasi | Step 2: Sistem kemudian merespon dengan menampilkan form untuk login |

|  |  |  |
| --- | --- | --- |
|  | Step 3: Tutor memasukan data pada mengisi form yang disediakan | Step 4: Sistem melakukan verifikasi data yang diberikan |
| Step 5: Tutor mengakses pesanan yang aktif | Step 6**:** menampilkan maps lokasi siswa/pengguna |
| **Skenario Alternatif:** | Alternatif 5 : Bila input data tidak valid akan ada peringatan untuk tutor dan diminta mengisi dan menginput ulang data | |

|  |  |
| --- | --- |
| **Kesimpulan:** | Use case dianggap berhasil apabila tutor dapat melihat, memperbarui, dan mengelola program serta transaksi terkait dengan sukses tanpa kesalahan. |
| **Kondisi Akhir:** | Data program atau pesanan yang aktif berhasil diperbarui dan tersimpan. |

|  |  |
| --- | --- |
| **Aturan Bisnis:** | User hanya dapat melakukan login dengan mengisikan username dan password yang valid dan hanya teridentifikasi sebagai pelanggan |
| **Kendala Implementasi dan Spesifikasi:** | **-** |
| **Tanggapan:** | **-** |
| **Masalah Terbuka:** | **-** |

* 1. **Kebutuhan Non Fungsional**

Kebutuhan non-fungsional yang harus dilakukan sistem/aplikasi kursus mobil ini yaitu:

## Keamanan (Security)

* + Otentikasi dan otorisasi pengguna (Login/Role: Pemilik Kursus, Pelanggan, Calon Pengguna).
  + Enkripsi data penting seperti password dan transaksi.
  + Validasi input agar tidak terjadi SQL Injection atau XSS (jika berbasis web).
  + Pembatasan akses (fitur tertentu hanya bisa diakses oleh aktor tertentu).

## Ketersediaan (Availability)

* + Aplikasi harus selalu tersedia 24/7 untuk pengguna melakukan transaksi atau melihat jadwal.
  + Fitur pemulihan sistem jika terjadi error (contoh: retry transaksi, simpan data lokal sementara jika offline).

## Kinerja (Performance)

* + Respon aplikasi harus cepat (< 3 detik untuk load halaman penting seperti jadwal kursus).
  + Optimisasi database agar pencarian dan pemesanan tidak lambat, terutama jika data pengguna/jadwal banyak.

## Kapasitas & Skalabilitas

* + Sistem harus bisa menangani banyak pengguna (misalnya 1000 user aktif secara bersamaan).
  + Skalabel untuk penambahan kursus, jadwal, transaksi baru, tanpa merombak sistem dari awal.

## Kemudahan Penggunaan (Usability)

* + Interface pengguna harus mudah dipahami dan digunakan bahkan oleh pengguna non-teknis.
  + Navigasi jelas, tombol dan label deskriptif, serta adanya pesan kesalahan dan konfirmasi yang mudah dimengerti.

## Portabilitas

* + Aplikasi bisa digunakan di berbagai perangkat: desktop, tablet, atau smartphone.
  + Bisa berbasis web atau jika versi mobile, bisa dibuat versi Android/iOS.

## Reliabilitas (Keandalan)

* + Sistem tidak boleh crash atau gagal saat pemrosesan penting seperti transaksi atau pemesanan.
  + Wajib ada sistem logging/error log.

## Maintainability (Kemudahan Pemeliharaan)

* + Struktur kode harus modular, mudah diubah atau ditambahkan fitur.
  + Dokumentasi internal yang baik untuk developer baru.

## Backup dan Recovery

* + Data transaksi dan jadwal kursus harus dibackup secara berkala.
  + Harus ada fitur recovery data jika terjadi kerusakan.

# K. DAFTAR PUSTAKA

Modul 3 RPL-Inisiasi Proyek

Modul 4 RPL-Inception Phase Workflow Analisis Kebutuhan Modul 5 RPL-Elaboration Phase#1

# L. LAMPIRAN

1. Daftar Pertanyaan Wawancara Bapak Sunaryo (pemilik Kursus Sabar Stir Mobil)
   * Bisa ceritakan sedikit tentang kursus mengemudi ini?

**Jawaban:** Tempat kursus menghadapi masalah dalam hal penjadwalan dan keterbatasan jumlah mobil, serta memberikan layanan antar-jemput dengan biaya sesuai jarak.

* + Berapa jumlah instruktur yang tersedia?

**Jawaban:** Instruktur tersedia satu yaitu bapak Sunaryo sendiri

* + Apa saja jenis paket kursus yang ditawarkan?

**Jawaban:** Ada, terdapat 3 paket yaitu paket reguler/biasa, paket kilat (latihan setiap hari), dan paket kilat khusus (2 hari latihan selesai)

* + Bagaimana proses pendaftaran kursus saat ini? Apakah dilakukan secara manual atau sudah ada sistem digital?

**Jawaban:** Pendaftaran bisa menghubungi langsung nomor WhatsApp ke Admin kursus, atau ke kantor langsung.

* + Bagaimana sistem penjadwalan latihan mengemudi? Apakah fleksibel atau ada jadwal tetap?

**Jawaban:** Ada kendala penjadwalan karena jumlah mobil yang terbatas, sehingga sering terjadi bentrokan jadwal antar siswa.

* + Berapa banyak kendaraan yang tersedia untuk kursus?

**Jawaban:** Jumlah kendaraan yang terbatas (ada satu) menjadi salah satu kendala dalam manajemen penjadwalan kursus.

* + Apakah ada sistem pemantauan kondisi kendaraan?

**Jawaban:** Ada, dirawat rutin kondisi kendaraannya.

* + Apakah kursus ini membantu siswa dalam pembuatan SIM?

**Jawaban:** Tidak.

* + Apakah ada sertifikat yang diberikan setelah menyelesaikan kursus?

**Jawaban:** Ada sertifikat yang diberikan, tetapi tidak bisa membantu dalam pembuatan SIM.

* + Apakah kursus ini sudah menggunakan aplikasi atau sistem digital dalam operasionalnya?

**Jawaban:** Kursus ini masih mengandalkan banner yang ditempel di mobil sebagai bahan promosinya, menggunakan media sosial (Instagram), mempromosikan di Google, dan mempromosikan dari mulut ke mulut.

* + Apa saja tantangan terbesar dalam mengelola kursus mengemudi?

**Jawaban:** Tantangan terbesar adalah dalam hal penjadwalan karena jumlah kendaraan yang terbatas dan layanan antar-jemput yang memerlukan pengaturan biaya sesuai jarak.

* + Masalah apa yang sering dialami siswa selama proses belajar?

**Jawaban:** Siswa masih takut mencoba (biasanya orang dewasa), siswa yang suka mengebut dan sangat pede (siswa remaja), dan pernah menabrak, jumlah kerugian jika ringan maka ditanggung oleh pemilik kursus dan jika kerugian besar maka ditanggung bersama.

* + Jika ada sistem aplikasi, bagaimana harapan Anda terhadap aplikasi tersebut?

**Jawaban:** Harapan terhadap sistem digital adalah agar siswa bisa melihat jadwal kursus, memilih sesuai tingkat keahlian, dan memberikan ulasan instruktur. Selain itu, sistem ini akan menyarankan kursus yang ada di zona terdekat, sehingga fokusnya lebih pada kenyamanan dan lokasi daripada harga.

1. Jadwal Kerja Kelompok:

26 Feb 2025, rapat awal dan brainstorming ide semua anggota.

1. Dokumentasi:

