**RENCANA PEMBELAJARAN SEMESTER**

**MATA KULIAH ANALISIS SISTEM DAN DESAIN**

Nama : Damaiyanti Sihombing

NIM : 12030123120074

Kelas : D

| **Minggu** | **Materi Pembelajaran** | **Aktivitas dan Tugas** | **Aplikasi yang Digunakan** |
| --- | --- | --- | --- |
| 1 | **Pendahuluan Analisis Sistem dan Desain**  - Pengertian dan ruang lingkup | - Diskusi kelas tentang pengertian analisis sistem dan desain.  - Tugas membaca artikel tentang pentingnya analisis sistem dalam dunia IT. | Google Classroom untuk diskusi online |
| 2 | **Proses Pengembangan Sistem**  - Siklus hidup sistem (SDLC) | - Penjelasan SDLC: Waterfall, Agile, dll.  - Tugas individu: Membuat ringkasan tahapan SDLC. | Google Docs untuk penulisan ringkasan |
| 3 | **Analisis Kebutuhan Sistem**  - Teknik pengumpulan data: wawancara, kuesioner, observasi | - Latihan: Simulasi wawancara untuk pengumpulan kebutuhan.  - Tugas kelompok: Membuat laporan analisis kebutuhan dari hasil simulasi. | Microsoft Word untuk laporan |
| 4 | **Modeling Sistem**  - Diagram Alir Data (DFD) dan Entity Relationship Diagram (ERD) | - Workshop: Pembuatan DFD dan ERD untuk sistem perpustakaan sederhana.  - Tugas: Membuat DFD dan ERD untuk studi kasus yang diberikan. | Lucidchart atau Microsoft Visio |
| 5 | **Analisis dan Desain Berorientasi Objek**  - Pengantar OOP dan UML (Use Case Diagram, Class Diagram) | - Penjelasan tentang OOP dan UML.  - Tugas: Membuat Use Case Diagram dan Class Diagram untuk aplikasi pemesanan tiket online. | StarUML atau draw.io |
| 6 | **Desain Interface dan User Experience (UI/UX)**  - Prinsip UI/UX dalam sistem informasi | - Workshop: Mendesain interface aplikasi mobile sederhana.  - Tugas: Membuat mockup desain interface untuk aplikasi e-commerce. | Figma atau Adobe XD |
| 7 | **Desain Database**  - Normalisasi dan denormalisasi | - Penjelasan tentang normalisasi dan denormalisasi database.  - Tugas: Merancang database normalisasi untuk sistem manajemen gudang. | MySQL Workbench atau PostgreSQL |
| 8 | **UTS**  - Evaluasi tengah semester | - Ujian tertulis dan studi kasus analisis sistem | - |
| 9 | **Pengujian Sistem**  - Jenis-jenis pengujian (unit testing, integration testing, system testing) | - Penjelasan tentang metode pengujian.  - Tugas: Membuat rencana pengujian sistem untuk aplikasi yang telah dirancang. | Selenium untuk pengujian otomatis atau JUnit |
| 10 | **Implementasi dan Pemeliharaan Sistem**  - Tahap implementasi dan maintenance | - Diskusi kasus: Implementasi dan pemeliharaan sistem di perusahaan.  - Tugas: Membuat rencana implementasi dan maintenance untuk sistem yang telah dirancang. | Microsoft Project atau Asana untuk perencanaan |
| 11 | **Rekayasa Perangkat Lunak**  - Pengantar Rekayasa Perangkat Lunak dan metodologi agile | - Penjelasan tentang rekayasa perangkat lunak dan metodologi agile.  - Tugas: Merancang sprint planning untuk pengembangan perangkat lunak sederhana. | Jira atau Trello |
| 12 | **Manajemen Proyek Sistem Informasi**  - Aspek manajemen dalam pengembangan sistem | - Diskusi: Tantangan dalam manajemen proyek TI.  - Tugas: Membuat timeline pengembangan proyek dengan Gantt Chart. | Microsoft Project atau GanttProject |
| 13 | **Prototype dan Rapid Application Development (RAD)**  - Teknik pengembangan cepat dengan RAD | - Workshop: Membuat prototype aplikasi sederhana dengan RAD.  - Tugas: Membuat prototype aplikasi inventaris barang menggunakan RAD. | Axure RP atau Balsamiq |
| 14 | **UAS**  - Evaluasi akhir semester | - Presentasi tugas akhir: prototype aplikasi | - |

**Detail Tugas Prototype yang Unik dan Sederhana:**

* **Kasus:** **Pengembangan sistem inventaris barang untuk toko retail.**
* Berikut adalah penjelasan lebih rinci tentang tujuan dan langkah-langkah penyelesaian prototype agar lebih mudah dipahami:

**1. Analisis Kebutuhan**

**Tujuan**: Untuk memahami apa yang dibutuhkan oleh sistem yang akan dibangun, berdasarkan permintaan pengguna atau pemilik bisnis.

**Langkah Penyelesaian**:

**Pengumpulan Informasi**: Melakukan wawancara atau survei dengan pemilik toko retail untuk memahami kebutuhan mereka, seperti bagaimana mereka mengelola inventaris saat ini dan fitur apa yang mereka inginkan di sistem baru.

**Dokumentasi Kebutuhan**: Menuliskan semua informasi yang dikumpulkan ke dalam sebuah dokumen yang menjelaskan kebutuhan sistem secara rinci. Ini termasuk kebutuhan fungsional (apa yang harus dilakukan sistem) dan non-fungsional (seperti kecepatan, keamanan, dan kemudahan penggunaan).

**2. Pembuatan Diagram Alir Data (DFD) dan Entity Relationship Diagram (ERD)**

**Tujuan**: Untuk memvisualisasikan bagaimana data mengalir dalam sistem dan bagaimana entitas-entitas di dalam sistem saling berhubungan.

**Langkah Penyelesaian**:

**Diagram Alir Data (DFD)**:

Buat **DFD Level 0** untuk menunjukkan interaksi antara sistem dan entitas luar seperti pengguna.

Lanjutkan dengan **DFD Level 1** untuk menjabarkan proses-proses internal sistem, seperti bagaimana data dari pengguna diproses dalam sistem.

**Entity Relationship Diagram (ERD)**:

Identifikasi entitas utama (seperti Barang, Transaksi, dan Pengguna) dan hubungan di antaranya.

Gambarkan hubungan tersebut dengan ERD, misalnya, hubungan antara entitas Barang dan Transaksi.

**3. Perancangan UI/UX dengan Mockup**

**Tujuan**: Untuk menciptakan antarmuka pengguna yang intuitif dan menarik, memastikan sistem mudah digunakan oleh pengguna akhir.

**Langkah Penyelesaian**:

**Sketsa Awal**:

Gambarkan sketsa kasar dari layar utama sistem, termasuk bagaimana pengguna akan menavigasi antara berbagai fitur (seperti menambah barang, melihat laporan stok).

**Pembuatan Mockup**:

Gunakan alat desain seperti Figma untuk membuat mockup visual dari antarmuka pengguna, memperhatikan aspek-aspek seperti tata letak, warna, dan keterbacaan.

**Iterasi Desain**:

Dapatkan umpan balik dari pemangku kepentingan (misalnya, pemilik toko) dan lakukan penyesuaian pada desain berdasarkan saran mereka.

**4. Desain Database dengan Normalisasi**

**Tujuan**: Untuk memastikan bahwa data disimpan secara efisien dan terstruktur dalam database, mengurangi redundansi data.

**Langkah Penyelesaian**:

**Identifikasi Tabel-tabel Database**:

Berdasarkan ERD, buat tabel untuk setiap entitas, seperti tabel Barang, Transaksi, dan Pengguna.

**Proses Normalisasi**:

Lakukan normalisasi tabel untuk memastikan tidak ada data yang terduplikasi secara tidak perlu. Misalnya, pastikan data barang dan transaksi dipisahkan sehingga setiap barang hanya perlu dimasukkan sekali dalam database.

**Implementasi Skema**:

Gunakan alat seperti MySQL Workbench untuk mendesain skema database dan menguji struktur tersebut dengan data sampel.

**5. Pembuatan Prototype dengan Rapid Application Development (RAD)**

**Tujuan**: Untuk menghasilkan prototype sistem yang cepat dan fungsional, memungkinkan pengguna untuk menguji sistem sebelum pengembangan penuh.

**Langkah Penyelesaian**:

**Pembagian Fitur ke dalam Sprint**:

Pisahkan pengembangan menjadi beberapa sprint kecil, fokus pada satu atau dua fitur utama per sprint, seperti fitur penambahan barang dalam satu sprint, dan fitur laporan stok pada sprint berikutnya.

**Pengembangan Prototype**:

Gunakan alat seperti Axure RP untuk membuat prototype yang interaktif. Prototype ini akan meniru bagaimana sistem sebenarnya akan bekerja, meskipun belum ada logika backend yang lengkap.

**Pengujian dan Iterasi**:

Dapatkan umpan balik dari pengguna yang menguji prototype, lalu lakukan perbaikan dan penyesuaian berdasarkan umpan balik tersebut.

**6. Pengujian Prototype**

**Tujuan**: Untuk memastikan bahwa prototype berfungsi sesuai dengan yang diharapkan dan bebas dari kesalahan yang signifikan.

**Langkah Penyelesaian**:

**Unit Testing**:

Uji setiap fitur atau modul secara individual untuk memastikan mereka bekerja dengan benar. Misalnya, uji fitur penambahan barang untuk memastikan barang ditambahkan ke database dengan benar.

**Integration Testing**:

Uji bagaimana modul-modul yang berbeda bekerja bersama. Contohnya, pastikan ketika barang ditambahkan, itu langsung muncul dalam laporan stok.

**Dokumentasi Hasil Pengujian**:

Buat laporan yang mendokumentasikan hasil pengujian, termasuk masalah yang ditemukan dan perbaikan yang dilakukan.

Dengan mengikuti langkah-langkah ini, mahasiswa akan memperoleh pemahaman praktis dan holistik tentang bagaimana mengembangkan sebuah sistem informasi mulai dari tahap analisis hingga pembuatan prototype. Prototype yang dihasilkan juga dapat menjadi alat yang berguna untuk mendapatkan umpan balik awal dari pengguna dan memastikan bahwa sistem akhir akan memenuhi kebutuhan mereka

* **Aplikasi yang Digunakan:** Axure RP atau Balsamiq untuk prototyping, MySQL Workbench untuk desain database, Figma untuk desain UI/UX.

Berikut adalah penjelasan lebih rinci tentang tujuan dan langkah-langkah penyelesaian prototype agar lebih mudah dipahami: