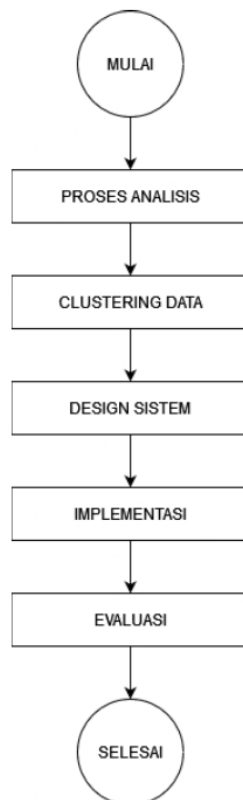


BAB III METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Tahapan penelitian pada penelitian yang penulis lakukan perlu digambarkan sebagai kerangka dan panduan proses penelitian, sehingga rangkaian proses penelitian dapat dilakukan secara searah, teratur dan sistematis. Berikut merupakan tahap penelitian yang dilakukan :



Gambar 3. 1 Tahapan Penelitian

3.2 Metode Pengembangan Perangkat Lunak

Dalam penelitian ini metode yang digunakan untuk melakukan pengembangan perangkat lunak adalah dengan menggunakan metode prototype. Proses ini juga dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat memenuhi keinginan user (pengguna) maka dari itu dalam proses ini diperlukan pengumpulan data sebagai berikut

3.2.1 Komunikasi

Komunikasi dilakukan untuk mendapatkan informasi terkait dengan penelitian yang akan dilakukan. Komunikasi dilakukan dengan cara yang tepat untuk mengumpulkan data objektif yang relevan dengan pokok pembahasan terkait penelitian ini. Dalam hal ini, peneliti menggunakan beberapa metode komunikasi sebagai berikut :

1. Observasi, metode pengumpulan data terkait penelitian ini dilakukan dengan menggunakan pengamatan secara langsung pada Program Studi Teknik Informatika IIB Darmajaya.
2. Wawancara, metode ini dilakukan dengan bertemu langsung dan melakukan proses tanya jawab atau wawancara kepada pihak terkait pada penelitian ini.
3. Studi pustaka, metode ini dilakukan dengan mencari referensi berwujud jurnal, buku dan penelitian lain yang berkenaan dengan system, serta sumber-sumber pendukung lain yang berkenaan dengan objek penelitian yang dipilih. Tujuan berasal dari studi pustaka yani untuk mendapatkan teori pendukung yang sudah sukses dijalankan pengembangan sistem untuk dijadikan referensi di dalam penelitian.

3.3 Perencanaan Secara Cepat

Perencanaan secara cepat adalah tahapan dengan melakukan analisis dan perencanaan yang tepat setelah data di dapatkan pada tahapan komunikasi

3.3.1 Analisis Kebutuhan Perangkat Lunak

Untuk menjalankan dan membangun sebuah perangkat lunak tersebut dibutuhkan sebuah perangkat lunak yang cukup agar menghasilkan sebuah perangkat lunak yang baik dan efektif untuk digunakan.

Adapun spesifikasi perangkat lunak yang dibutuhkan peneliti dalam pembuatan aplikasi ini adalah sebagai berikut :

1. Visual Studio Code
2. MongoDB
3. Robo 3T
4. Node JS
5. Postman
6. Quasar
7. Browser Chrome / Firefox
8. Sistem Operasi Windows 10

3.3.2 Analisis Kebutuhan Perangkat Keras

Analisis kebutuhan perangkat keras yang digunakan dalam pembuatan dan menjalankan layanan ini adalah sebagai berikut :

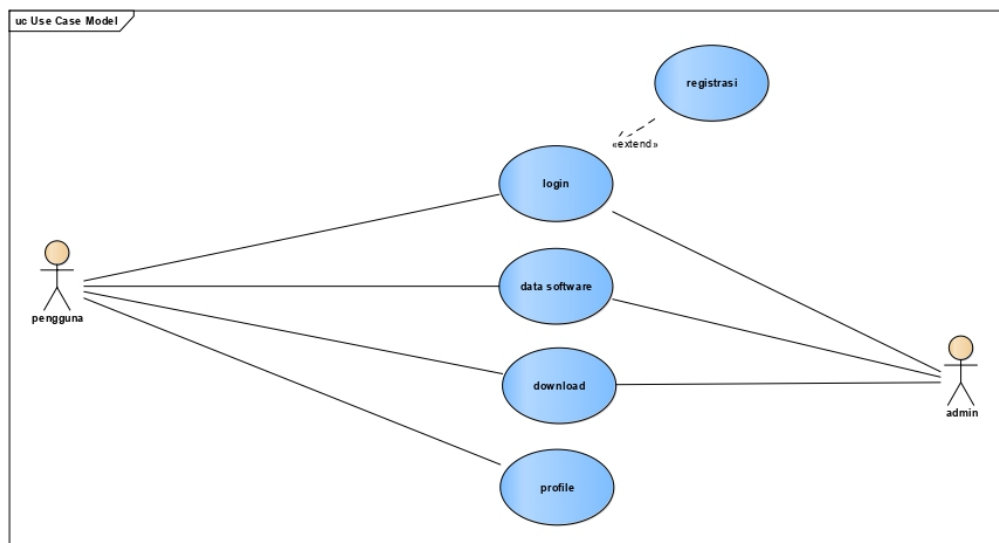
1. MSI GL63 8RD
2. RAM 8 GB
3. Harddisk 1 TB
4. Processor Intel(R) HM370 CPU @ 1.70GHz

3.4 Pemodelan Secara Cepat

Tahap ini melakukan perancangan yang dimulai dari desain perancangan UML yaitu digunakan untuk menentukan desai Use Case Diagram, Activity Diagram adalah sebagai berikut :

3.4.1 Use Case Diagram

Dalam pengembangan sistem aplikasi ini untuk pengelolaan data software di Program Studi Teknik Informatika IIB Darmajaya berbasis Web Base, telah dirancang usecase diagram yang menggambarkan fungsi-fungsi yang dimiliki oleh masing-masing rule. Dalam rancangan sistem ini terdapat dua level user yaitu admin dan pengunjung. Pada usecase diagram yang dirancang dapat melakukan beberapa interaksi antara lain: registrasi, login, melihat data software, download, mengelola profil, mengunggah data software. use case chart dapat dilihat pada Gambar berikut.



Gambar 3. 2 Use Case Sistem

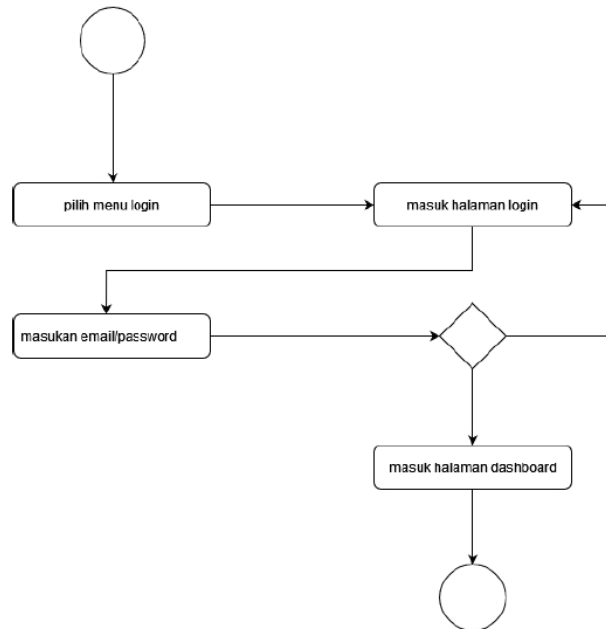
3.4.2 Activity Diagram

Activity diagram menggambarkan perkembangan aliran dan latihan serta digunakan untuk menggambarkan latihan yang dibingkai dalam suatu aktivitas.

Membuat bagan gerakan menuju awal interaksi dapat membantu memahami seluruh aktifitas.

1. Activity diagram login

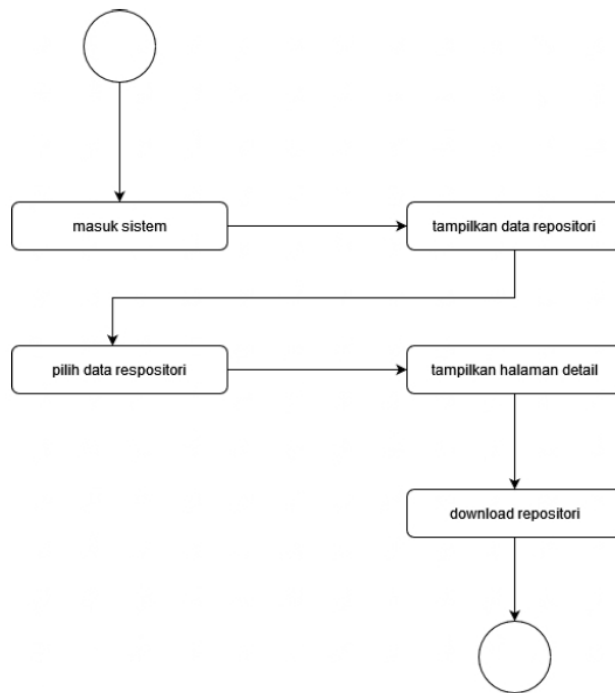
Activity ini menjelaskan bagaimana proses user masuk kedalam layanan dan memvalidasi akun yang sudah ada. Jika akun valid maka sistem akan menampilkan halaman utama pada aplikasi. Jalannya grafik tindakan login dapat di temukan pada gambar berikut.



Gambar 3. 3 Activity Login

2. Activity diagram detail data

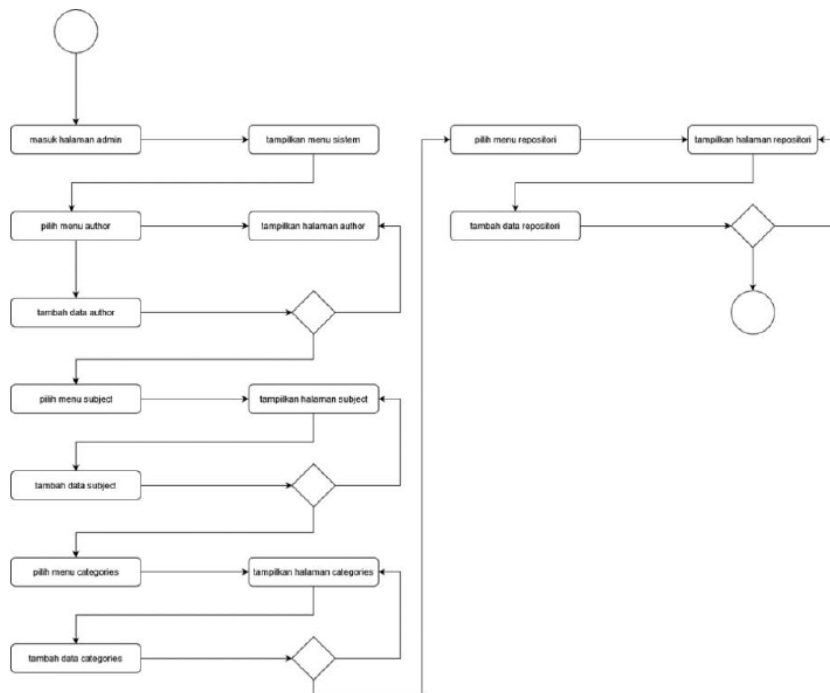
Activity ini berfungsi untuk melakukan proses download di dalam sistem dengan masuk ke halaman detail data yang selanjutnya akan dilanjutkan ke proses download dengan memilih repositori mana yang akan dilakukan download. Proses activity ini dapat dilihat pada gambar berikut.



Gambar 3. 4 Activity Detail Data

3. Activity diagram kelola data repository

Activity kelola data repository menjelaskan proses untuk menambahkan data author, subject, kategori dan repositori di dalam sistem. Proses activity ini dapat dilihat pada gambar berikut.



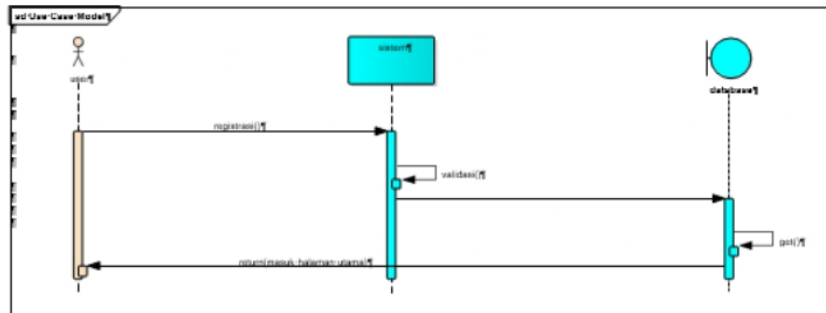
Gambar 3. 5 Activity Kelola Data

3.4.3 Sequence Diagram

Sequence diagram menggambarkan kerjasama antara setiap item dalam setiap use case dalam satu periode berturut-turut. Interaksi ini berupa pengiriman serangkaian data antara objek yang saling berinteraksi. Sequence diagram memiliki dua dimensi vertical yang menggambarkan waktu yang terkait oleh objek dan dimensi horizontal menggambarkan objek yang terkait pada masing-masing diagram.

1. Login

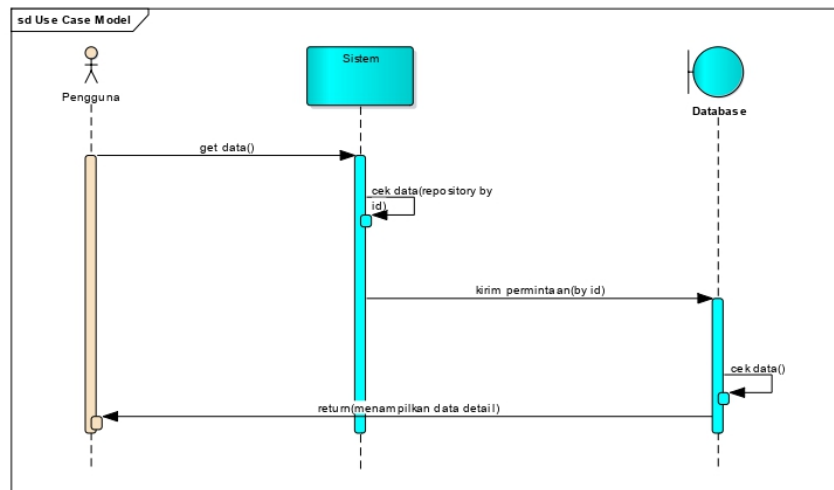
Pengguna aplikasi melakukan proses login untuk dapat masuk ke dalam sistem, dengan menggunakan data email dan password yang telah di daftarkan. User hanya admin sistem. Setelah proses login dilakukan pengguna akan di alihkan ke halaman utama dari sistem. Lihat pada gambar berikut.



Gambar 3. 6 Sequence Diagram Login

2. Halaman detail data

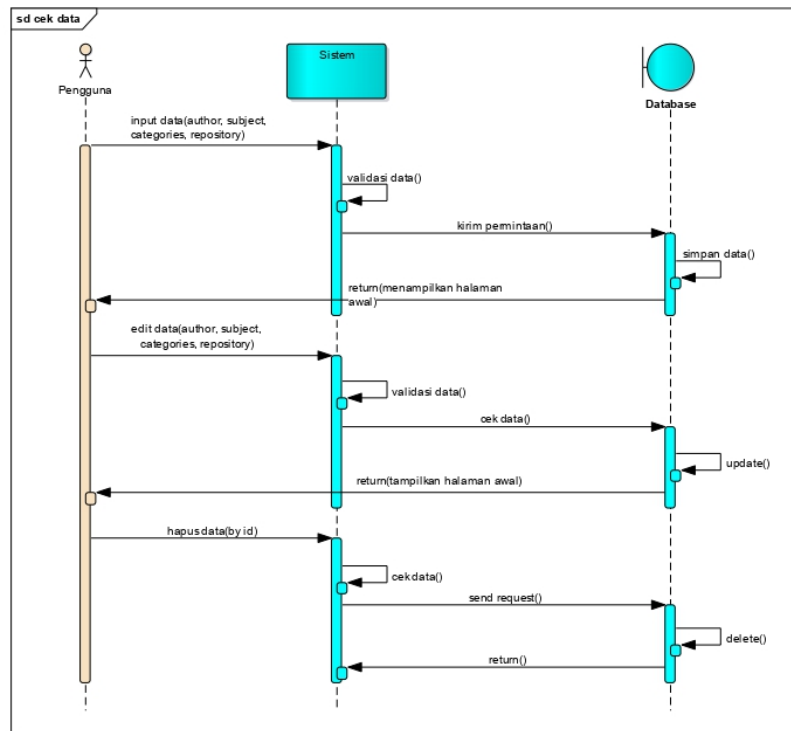
Mahasiswa dan dosen dapat melakukan pencarian data repository di halaman utama sistem. Di dalam halaman ini terdapat seluruh list data software beserta informasi yang di cari, dengan memasukan kata kunci data akan di tampilkan ke pengguna sesuai data yang ingin di cari. Dengan menekan tombol detail maka user akan di arahkan ke dalam halaman detail data sesuai dengan data yang di pilih dan user dapat melakukan download berdasarkan URL yang tertera yang dapat di lihat pada gambar berikut.



Gambar 3. 7 Sequence Diagram Detail Data

3. Kelola data repository

Pihak admin dapat mengelola data repository dengan menghapus, menambah serta mengubah data author, seubjek, kategori dan reository sesuai dengan informasi valid terkait data. Lihat pada gambar berikut.



Gambar 3. 8 Sequence Diagram Kelola Data

3.5 Tampilan Antarmuka (User Interfaces)

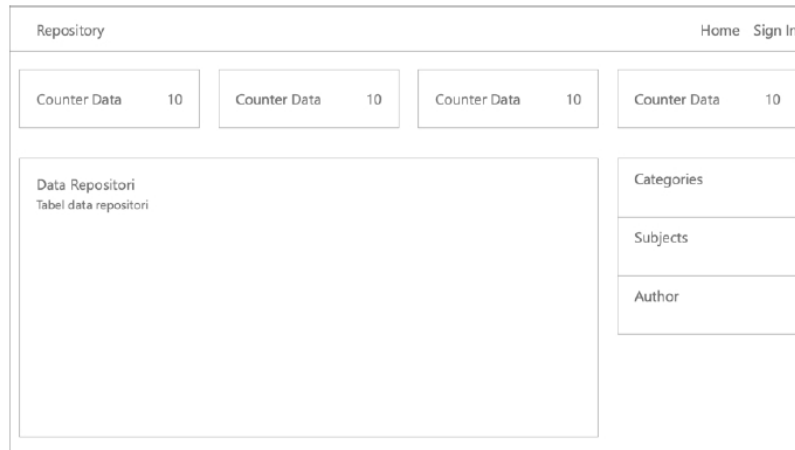
Rancangan inteface pada sistem ini adalah sebagai berikut :

3.5.1 Rancangan Sistem

Rancangan interface yang akan di tampilkan di sisi admin terdiri dari beberapa bagian diantaranya sebagai berikut :

1. Halaman Utama Sistem

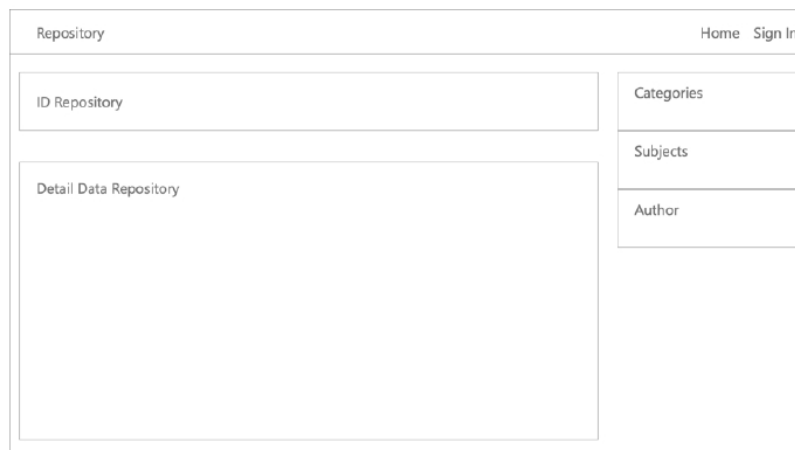
Halaman utama sistem digunakan oleh pengguna saat pertama kali masuk ke dalam sistem, pengguna akan di tampilkan seluruh data repository yang terdaftar di dalam sistem.



Gambar 3. 9 Halaman Utama Sistem

2. Halaman Detail Data

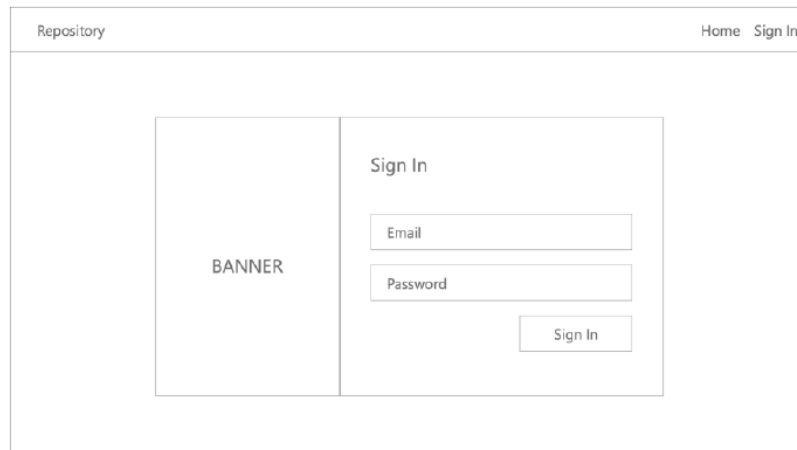
Halaman detail data dapat di akses dengan menekan tombol yang terdapat di dalam table data repository, pada halaman detail data pengguna dapat melakukan download file dengan mengkases URL tempat dimana file di upload.



Gambar 3. 10 Halaman Detail Data

3. Halaman login

Halaman login digunakan oleh pengguna yaitu untuk masuk ke dalam sistem dan mengakses seluruh fitur yang dibangun di dalam sistem.

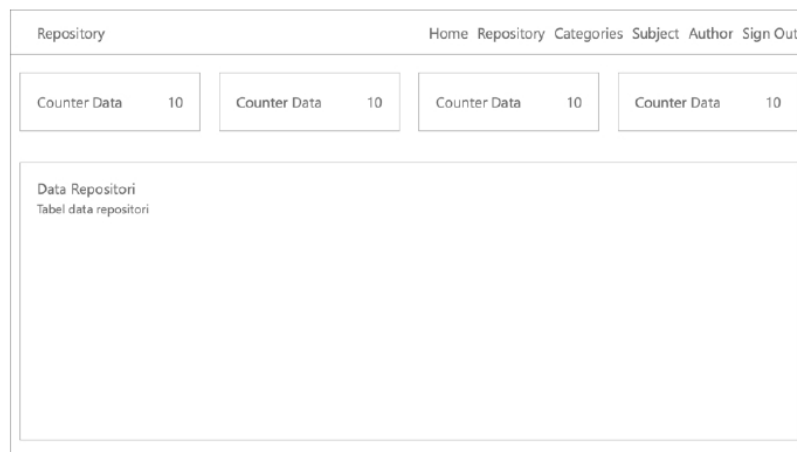


The screenshot shows a web page for a login interface. At the top left, the word "Repository" is displayed. At the top right, there are links for "Home" and "Sign In". The main content area is split into two columns. The left column contains a box labeled "BANNER". The right column is titled "Sign In" and contains three elements: an input field for "Email", an input field for "Password", and a "Sign In" button.

Gambar 3. 11 Halaman Login

4. Halaman Utama Admin

Halaman admin berisikan data repository yang telah di inputkan admin..

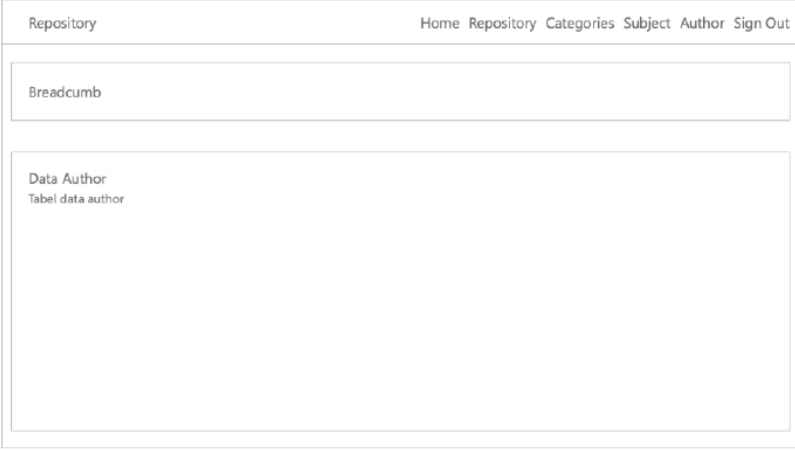


The screenshot shows an admin dashboard. The header includes "Repository" on the left and navigation links "Home", "Repository", "Categories", "Subject", "Author", and "Sign Out" on the right. Below the header, there are four identical boxes, each labeled "Counter Data" and showing the number "10". Below these boxes is a large section titled "Data Repositori" with the subtitle "Tabel data repositori".

Gambar 3. 12 Halaman Utama Admin

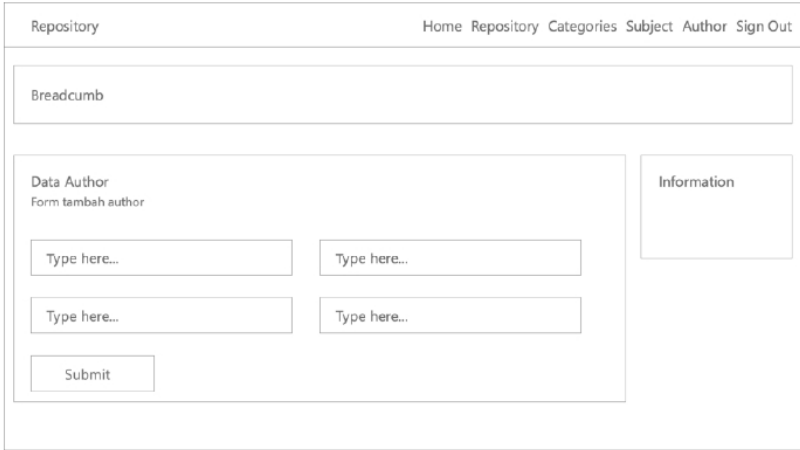
5. Halaman Data Author

Halaman author terdiri dari dua halaman yaitu detail data dan form tambah data, data author digunakan sebagai identifikasi kepemilikan data repository.



The screenshot shows a web interface for 'Data Author'. At the top, there is a navigation bar with 'Repository' on the left and 'Home Repository Categories Subject Author Sign Out' on the right. Below the navigation bar is a breadcrumb section labeled 'Breadcumb'. The main content area is titled 'Data Author' and contains the text 'Tabel data author', indicating a table that is currently empty or not rendered.

Gambar 3. 13 Halaman Data Author

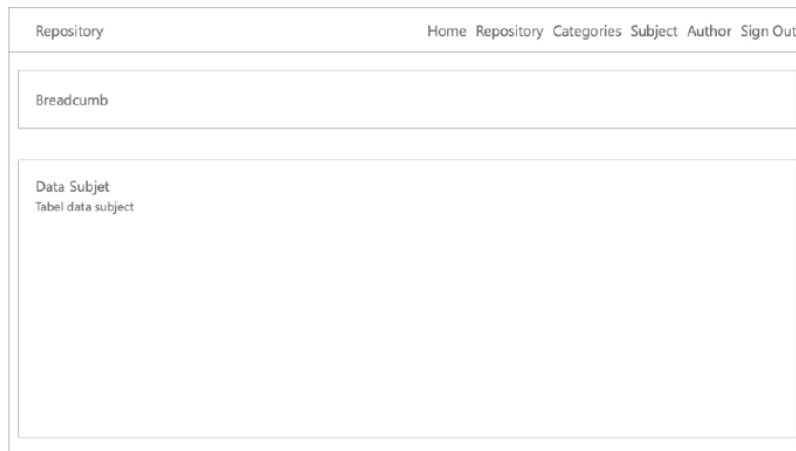


The screenshot shows the 'Form tambah author' page. It features a navigation bar at the top with 'Repository' and 'Home Repository Categories Subject Author Sign Out'. A breadcrumb section is present. The main content area is titled 'Data Author' and 'Form tambah author'. It contains four text input fields arranged in a 2x2 grid, each with the placeholder text 'Type here...'. Below the input fields is a 'Submit' button. To the right of the form is an 'Information' box.

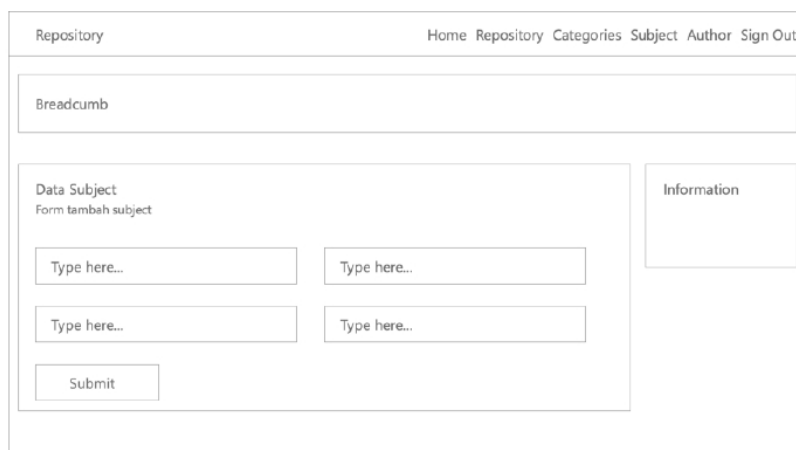
Gambar 3. 14 Halaman Tambah Data Author

6. Halaman Data Subject

Halaman subject terdiri dari dua halaman yaitu detail data dan form tambah data, data subject digunakan sebagai identifikasi matakuliah dari data repository.



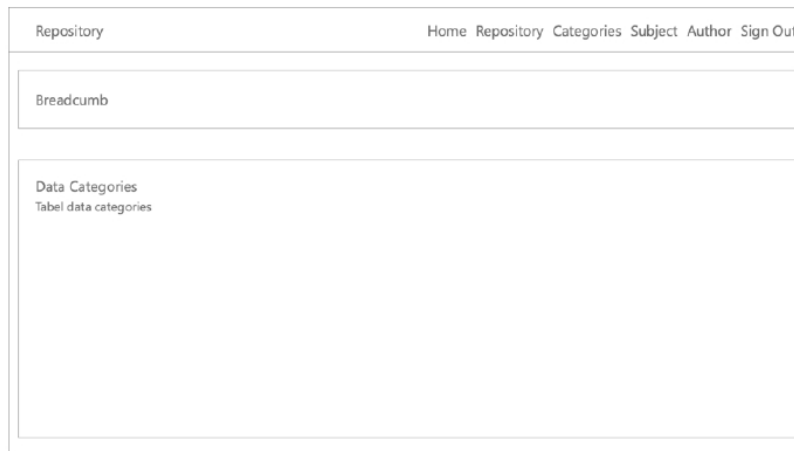
Gambar 3. 15 Halaman Data Subject



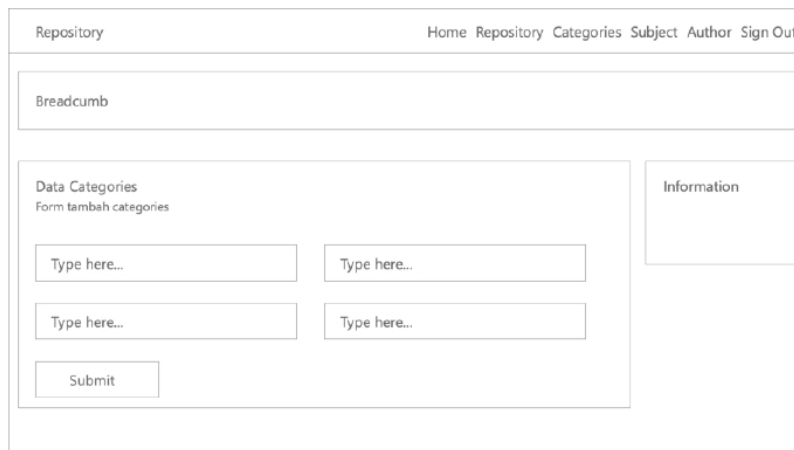
Gambar 3. 16 Halaman Tambah Data Subject

7. Halaman Data Categories

Halaman categories terdiri dari dua halaman yaitu detail data dan form tambah data, data categories digunakan sebagai identifikasi kategori dari file sesuai dengan jenis yang di samakan.



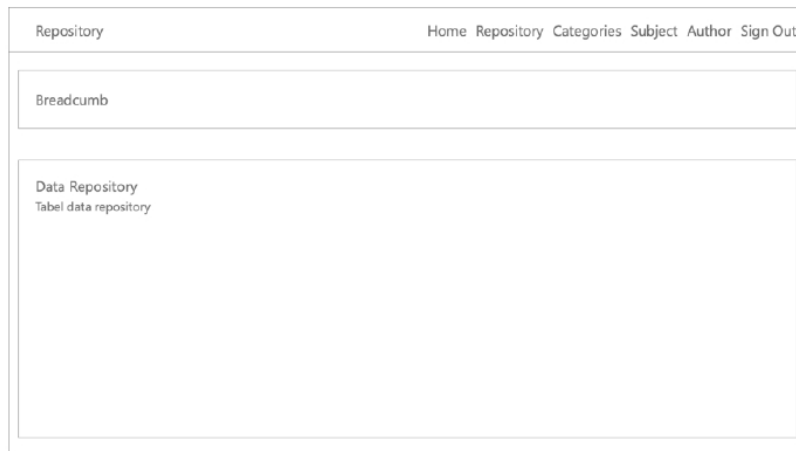
Gambar 3. 17 Halaman Data Categories



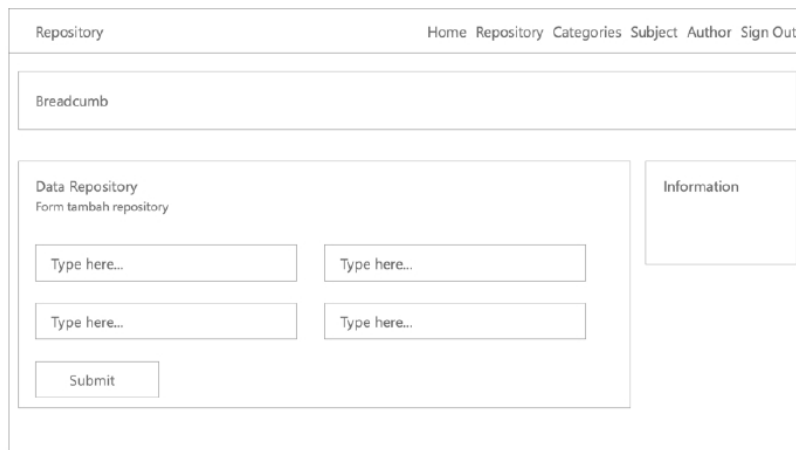
Gambar 3. 18 Halaman Tambah Data Categories

8. Halaman Data Repository

Halaman repository terdiri dari dua halaman yaitu detail data dan form tambah data, data repository adalah gabungan dari data author, subject, kategori dan tambahan data URL dari file yang telah di upload digunakan sebagai identifikasi kepemilikan data repository.



Gambar 3. 19 Halaman Data Repository



Gambar 3. 20 Halaman Tambah Data Repository

3.6 Implementasi Algoritma Dalam Kode Program

Pada penerapannya sistem akan di bangun menggunakan layer frontend dan backend, frontend bertindak sebagai display atau visualisasi dari sistem yang akan digunakan, sedangkan backend sendiri bertugas untuk berkomunikasi antara layer depan dengan database atau biasa disebut juga sebagai API's. implementasi selection sort sendiri dapat digunakan pada kedua layer tersebut, berikut contoh source code pada masing-masing layer :

1. Layer depan

Source code data repository

```
selectionShort: function (data) {
  let n = data.length;

  for(let i = 0; i < n; i++) {
    // Finding the smallest number in the subarray
    let min = i;
    for(let j = i+1; j < n; j++){
      if(data[j] < data[min]) {
        min=j;
      }
    }
    if (min != i) {
      // Swapping the elements
      let tmp = data[i];
      data[i] = data[min];
      data[min] = tmp;
    }
  }

  return data;
},
```

Gambar 3. 21 Source code data repository

2. Layer belakang

Source code controller data repository

```
const getAll = async (req, res) => {
  try {
    const data = await service.getAll(req.body);
    response = { ...requestResponse.success, data };
  } catch (error) {
    logger.error(error);
    response = { ...requestResponse.server_error };
  }
  res.json(response);
}

const getById = async (req, res) => {
  try {
    const data = await service.getById({ REPOSITORY_CODE: req.params.code });
    response = { ...requestResponse.success, data };
  } catch (error) {
    logger.error(error);
    response = { ...requestResponse.server_error };
  }
  res.json(response);
}
```

Gambar 3. 22 Source Code Controller

Source code service data repository

```
const getAll = async (condition) => {
  return model.aggregate([ { $match: { $and: [ condition ] }
  }, { $lookup: { from: "author", localField: "AUTHOR", foreignField: "GUID", as: "AUTHOR_DATA" }
  }, { $unwind: { path: '$AUTHOR_DATA', preserveNullAndEmptyArrays: true }
  }, { $lookup: { from: "subjects", localField: "SUBJECT", foreignField: "GUID", as: "SUBJECT_DATA" }
  }, { $unwind: { path: '$SUBJECT_DATA', preserveNullAndEmptyArrays: true }
  }, { $lookup: { from: "categories", localField: "CATEGORIES", foreignField: "GUID", as: "CATEGORIES_DATA" }
  }, { $unwind: { path: '$CATEGORIES_DATA', preserveNullAndEmptyArrays: true }
  }, { $sort: { SUBJECT_DATA: -1 } }
  ]
});
```

Gambar 3. 23 Source Code Service 1

```
const getById = async (condition) => {
  return model.aggregate([ {
    $match: {
      $and: [
        condition
      ]
    }
  }, {
    $lookup: {
      from: "author",
      localField: "AUTHOR",
      foreignField: "GUID",
      as: "AUTHOR_DATA"
    }
  }, {
    $unwind: {
      path: '$AUTHOR_DATA',
      preserveNullAndEmptyArrays: true
    }
  }, {
    $lookup: {
      from: "subjects",
      localField: "SUBJECT",
      foreignField: "GUID",
      as: "SUBJECT_DATA"
    }
  }, {
    $unwind: {
      path: '$SUBJECT_DATA',
      preserveNullAndEmptyArrays: true
    }
  }, {
    $lookup: {
      from: "categories",
      localField: "CATEGORIES",
      foreignField: "GUID",
      as: "CATEGORIES_DATA"
    }
  }, {
    $unwind: {
      path: '$CATEGORIES_DATA',
      preserveNullAndEmptyArrays: true
    }
  }
  ]
});
```

Gambar 3. 24 Source Code Service 2

3.7 Implementasi Algoritma Selection Sort

Selection sort merupakan metode yang digunakan untuk melakukan pengurutan nilai secara ekstrim seperti nilai terkecil atau terbesar dari data yang disajikan dengan mencari nilai data terkecil yang dimulai dari index 0 hingga di posisi N-1.

Ilustrasi dalam proses implementasi selection sort adalah sebagai berikut :

Kondisi awal array

| | | | | | |
|-------|---|---|---|---|---|
| data | 5 | 7 | 3 | 9 | 2 |
| index | 0 | 1 | 2 | 3 | 4 |

Perulangan pertama, selection sort dalam pencarian nilai terkecil

| | | | | | | |
|-------|---|---|---|---|---|---------|
| index | 0 | 1 | 2 | 3 | 4 | |
| data | 5 | 7 | 3 | 9 | 2 | Pos = 0 |
| | 5 | 7 | 3 | 9 | 2 | Pos = 0 |
| | 5 | 7 | 3 | 9 | 2 | Pos = 2 |
| | 5 | 7 | 3 | 9 | 2 | Pos = 2 |
| | 5 | 7 | 3 | 9 | 2 | Pos = 4 |

3. Pengurutan dilakukan secara menaik (dari kecil ke besar) maka perlu mencari nilai terkecil terlebih dahulu.
4. Pada perulangan pertama, variabel pos dimulai dari indeks 0 dengan nilai elemen yaitu 5.
5. Kemudian mencari nilai yang lebih kecil dari pada 5.
6. Selanjutnya dilakukan penukaran posisi berdasarkan nilai terkecil yang berhasil ditemukan.

| | | | | | | |
|-------|---|---|---|---|---|---------|
| index | 0 | 1 | 2 | 3 | 4 | |
| data | 5 | 7 | 3 | 9 | 2 | Pos = 4 |
| | 2 | 7 | 3 | 9 | 5 | Pos = 2 |
| | 2 | 3 | 7 | 9 | 5 | Pos = 4 |
| | 2 | 3 | 5 | 9 | 7 | Pos = 4 |
| | 2 | 3 | 5 | 7 | 9 | hasil |

Dalam proses pengurutannya, jika terdapat N data dan data terkoleksi dari urutan 0 sampai dengan N-1, maka berikut merupakan pengurutan metode selection sort :

1. Cari data terkecil dalam interval $J = 0$ sampai $J = N-1$.
2. Jika posisi ditemukan data terkecil, tukarkan data diposisi pos dengan data di posisi I jika K.
3. Ulangi langkah 1 dan 2 dengan $J = J+I$ sampai dengan $J = N-1$, dan seterusnya sampai $J = N$.

Jika terdapat data dalam bentuk array sebagai berikut :

Data array : {1,7,2,3,5,-4,11,10}

Maka cara pengurutannya adalah sebagai berikut :

| | | | | | | | |
|----|---|---|---|---|----|----|----|
| 4 | 7 | 2 | 3 | 5 | -4 | 11 | 10 |
| 4 | 7 | 2 | 3 | 5 | -4 | 11 | 10 |
| -4 | 7 | 2 | 3 | 5 | 4 | 11 | 10 |
| -4 | 2 | 7 | 3 | 5 | 4 | 11 | 10 |
| -4 | 2 | 3 | 7 | 5 | 4 | 11 | 10 |
| -4 | 2 | 3 | 4 | 5 | 7 | 11 | 10 |
| -4 | 2 | 3 | 4 | 5 | 7 | 11 | 10 |
| -4 | 2 | 3 | 4 | 5 | 7 | 10 | 11 |

3.8 Rancangan Database

Dalam penerapan layanan ini, pembuatan database dibangun dengan tatanan data sebagai berikut :

Tabel 3. 1 Collection Users

| No | Nama Field | Type |
|----|------------|----------|
| 1 | ObjectId | ObjectId |
| 2 | GUID | String |

| | | |
|----|----------|--------|
| 3 | FULLNAME | String |
| 4 | USERNAME | String |
| 5 | PASSWORD | String |
| 6 | EMAIL | String |
| 7 | PHONE | String |
| 8 | ROLE | Number |
| 9 | Create | Date |
| 10 | Update | Date |

Tabel 3. 2 Collection Subject

| No | Nama Field | Type |
|----|---------------|----------|
| 1 | ObjectId | ObjectId |
| 2 | GUID | String |
| 3 | SUBJECT_CODE | String |
| 4 | SUBJECT | String |
| 5 | STUDI_PROGRAM | String |
| 6 | Create | Date |
| 7 | Update | Date |

Tabel 3. 3 Collection Author

| No | Nama Field | Type |
|----|-------------|----------|
| 1 | ObjectId | ObjectId |
| 2 | GUID | String |
| 3 | AUTHOR_CODE | String |
| 4 | AUTHOR | String |
| 5 | GENDER | String |
| 6 | PHONE | String |
| 7 | Create | Date |

| | | |
|---|--------|------|
| 8 | Update | Date |
|---|--------|------|

Tabel 3. 4 Collection Categories

| No | Nama Field | Type |
|----|-----------------|----------|
| 1 | ObjectId | ObjectId |
| 2 | GUID | String |
| 3 | CATEGORIES_CODE | String |
| 4 | TITLE | String |
| 5 | DESC | String |
| 6 | Create | Date |
| 7 | Update | Date |

Tabel 3. 5 Collection Repository

| No | Nama Field | Type |
|----|-----------------|----------|
| 1 | ObjectId | ObjectId |
| 2 | GUID | String |
| 3 | REPOSITORY_CODE | String |
| 4 | CATEGORIES | String |
| 5 | SUBJECT | String |
| 6 | AUTHOR | String |
| 7 | SEMESTER | String |
| 8 | URL | String |
| 9 | DESC | String |
| 10 | Create | Date |
| 11 | Update | Date |