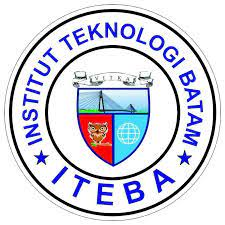
# Rekayasa Perangkat Lunak

# Peminjaman Buku di Perpustakaan



**Nama Kelompok :**

Arvy Kurnia Ramadhan (2022009)

Andika Zahrah Ramadhanti (2022029)

Armando Mendoza Putra (2022012)

Budi Prasetio (2022015)

Jeremy Andika (2022002)

Fajri Rahmat Ilahi (2022013)

Ricky Herlangga (2022018)

Ricky Arianto (2022010)

## 

## **Model Desain**

Untuk membuat model diperlukan pemahaman yang lebih baik dari entitas aktual yang akan dibangun.

Model harus mampu mewakili :

* Informasi transformasi software,
* Arsitektur dan fungsi yang memungkinkan transformasi terjadi,
* Fitur yang diinginkan pengguna, dan
* Perilaku sistem dimana transformasi terjadi.

Dalam software engineering, dapat dibuat 2 kelas model yaitu:

1. Model Persyaratan
2. Model Desain
3. **Model Persyaratan (atau analysis model)**

Merepresentasikan persyaratan konsumen dengan menggambarkan software dalam 3 domain yang berbeda:

* Domain infomrasi
* Domain fungsional
* Domain perilaku

**Elemen Model Persyaratan**

a) Elemen Scenario - based.

* Sistem ini dijelaskan dari sudut pandang pengguna menggunakan pendekatan
* berbasis scenario
* Elemen Scenario - based dari model persyaratan seringkali menjadi bagian
* pertama dari model yang dikembangkan.
* Dengan demikian, mereka berfungsi sebagai input untuk pembuatan elemen pemodelan lainnya.

b) Elemen Class - based.

* Setiap penggunaan skenario menyiratkan satu set objek yang dimanipulasi sebagai aktor yang berinteraksi dengan system.
* Obyek ini dikategorikan ke dalam kelas yaitu sekumpulan hal - hal yang memiliki atribut yang sama dan perilaku yg umum.
* Kelas perlu berkolaborasi dengansatu sama lain dan hubungan dan interaksi antara kelas

c) Elemen Perilaku.

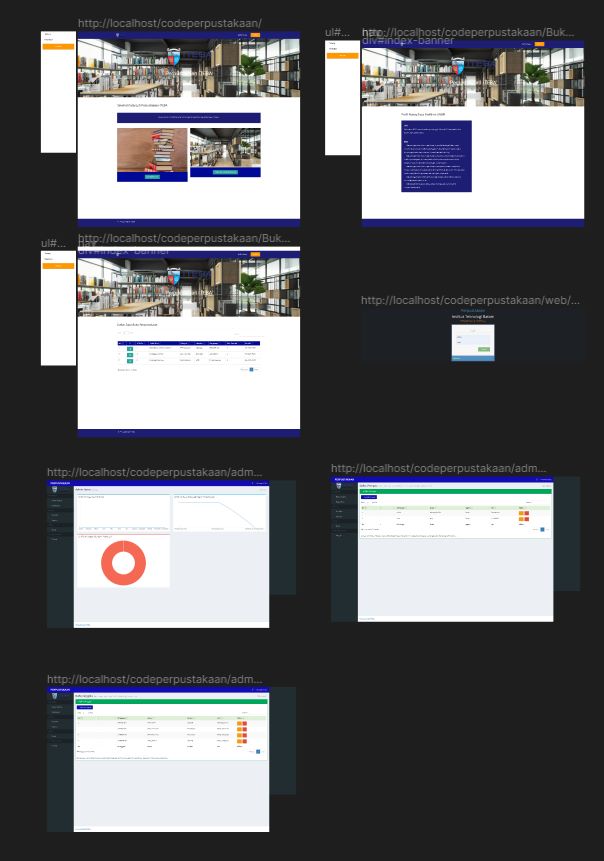
* Perilaku sistem berbasis komputer dapat dapat memiliki efek mendalam pada desain yang dipilih dan pendekatan implementasi yang diterapkan.
* Oleh karena itu, model persyaratan harus menyediakan unsur - unsur pemodelan yang menggambarkan perilaku.

d) Elemen Flow - oriented.

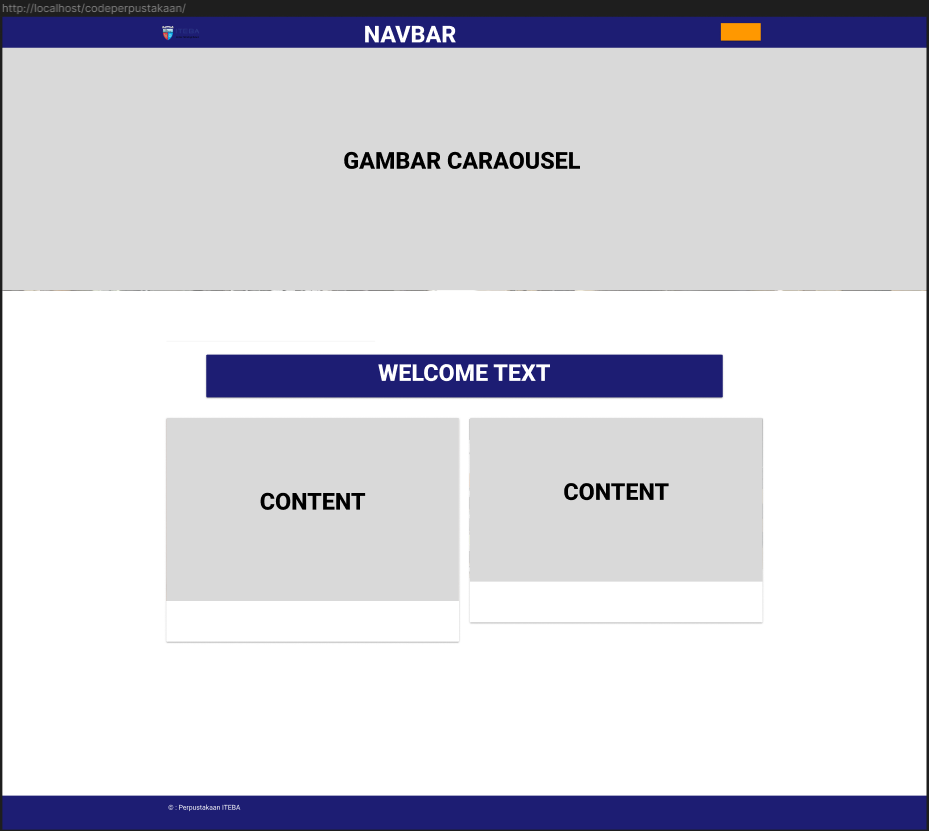
* Informasi berubah karena mengalir melalui sebuah sistem berbasis komputer.
* Sistem ini menerima masukan dalam berbagai bentuk, menggunakan fungsi untuk mengubahnya, dan menghasilkan output dalam berbagai bentuk.

## **Mockup**

Berikut adalah tampilan dari mockup kami, yang menunjukkan bagaimana desain aplikasi atau website akan terlihat dan berfungsi setelah selesai dikembangkan.



## Mockup ini menampilkan elemen-elemen yang akan ada di halaman utama, seperti navigasi, header, dan konten utama, serta interaksi yang akan terjadi antar elemen tersebut. Hal ini memungkinkan kami untuk mengevaluasi desain sebelum pengembangan sebenarnya dimulai, sehingga kami dapat membuat perubahan yang diperlukan sebelum waktu dan biaya yang lebih besar dikeluarkan. Dengan melihat hasil mockup ini, kami berharap Anda dapat membayangkan bagaimana aplikasi atau website akan terlihat dan berfungsi setelah selesai dikembangkan.



## **Deskripsi**

Desain perangkat lunak ini menawarkan antarmuka yang intuitif dan mudah digunakan, dengan navigasi yang jelas dan tombol yang jelas ditandai. Ini membuat penggunaan perangkat lunak ini menjadi sangat mudah dan efisien. Pengguna dapat dengan mudah menemukan fitur yang mereka cari dan menjalankan tugas dengan cepat.

* **Pengujian**

1. **Usability Testing**

|  |  |  |
| --- | --- | --- |
| Uji Navigasi | Tombol menu, dan tautan ke berbagai page | [✓] |
| Uji Konten | Konten dapat dibaca tanpa kesalahan ejaan | [✓] |

1. **Compability Testing**

|  |  |  |
| --- | --- | --- |
| Uji kompatibilitas Browser | menguji apakah aplikasi web Anda ditampilkan dengan benar di seluruh browser, JavaScript, AJAX | [✓] |
| Render element web | Dapat menampilkan element web seperti tombol,teks, dan berfungsi dengan baik dengan berbagai OS seperti Linus, Mac dan browser Firefox, Chrome. | [✓] |

1. **Interface Testing**

|  |  |  |
| --- | --- | --- |
| Server Database | kueri yang dikirim ke database memberikan hasil yang diharapkan. | [✓] |