

Membangun Antarmuka Pengguna Menggunakan ReactJs untuk Modul Manajemen Pengguna

Satrio Krisna Murti
Jurusan Informatika, Fakultas Teknologi Industri,
Universitas Islam Indonesia
Yogyakarta
17523097@students.uii.ac.id

Ari Sujarwo
Badan Sistem Informasi,
Universitas Islam Indonesia
Yogyakarta
ari.sujarwo@uui.ac.id

Abstract—ReactJs merupakan sebuah pustaka atau library Javascript yang dibuat oleh Facebook dan bersifat opensource untuk membangun Antarmuka Pengguna. React menawarkan respons cepat untuk *user input* menggunakan metode baru dalam proses rendering website. ReactJS juga cocok digunakan untuk pembuatan modul yang fleksibel khususnya pada *frontend* karena dalam bagian *frontend* banyak pembuatan menu, tabel, dan button yang akan lebih mudah jika dibuat secara modular sehingga apa yang telah dibuat dapat digunakan kembali hanya dengan sedikit modifikasi. Dengan adanya modul pengembangan aplikasi akan menjadi lebih mudah dan efisien. Modul berasal dari kata pemrograman modular yang merupakan istilah untuk mengorganisir kode yang telah dibuat, agar setiap fungsi dapat melakukan tugas masing-masing secara spesifik dengan tujuan agar penulisan kode program mudah dipahami dan dapat digunakan kembali untuk pengembangan selanjutnya. Adapun hasil pengujian dari pengembangan Modul Manajemen Pengguna ini semua fitur dan fungsi sudah berjalan sesuai dengan yang diharapkan.

Keywords— ReactJs, Frontend, Antarmuka Pengguna, Pemrograman Modular

I. PENDAHULUAN

Teknologi informasi dan komunikasi selalu menghasilkan alat yang memudahkan segala aktivitas manusia. Perkembangan teknologi informasi dan komunikasi kini sangatlah pesat. Perkembangan tersebut dapat dirasakan pada semua aspek kehidupan. Dampak dari perkembangan tersebut semua dituntut untuk bekerja secara efektif dan efisien. Demikian juga dengan para pengembang aplikasi. Selain dituntut untuk bekerja secara efektif dan efisien mereka harus mempelajari teknologi terbaru dalam pengembangan aplikasi web. Karena dengan perkembangan teknologi perkerjaan mereka bisa menjadi lebih mudah.

Dalam Pengembangan aplikasi web sering digunakan modul atau pustaka yang dapat dimanfaatkan untuk mempermudah pengembangan aplikasi web. Penggunaan modul sangat membantu para pengembang perangkat lunak. Pengembang perangkat lunak biasa menggunakan modul untuk menghemat waktu dalam membuat aplikasi karena dapat menggunakan komponen yang sudah ada untuk digunakan kembali, tentunya dengan modifikasi sesuai kebutuhan.

Saat ini, PT. Dua Empat Tujuh akan mengembangkan modul manajemen pengguna. Modul ini akan dibuat mulai

dari awal menggunakan ReactJs untuk standarisasi library atau pustakanya karena pengembangan web sebelumnya belum menggunakan ReactJs. Nantinya modul ini akan diaplikasikan untuk proyek proyek yang sudah ada.

II. KAJIAN PUSTAKA

A. User Interface (UI)

Antarmuka merupakan salah satu bagian dari perangkat lunak yang berhubungan langsung dengan pengguna. Pengguna berinteraksi dengan sistem perangkat lunak melalui antarmuka pengguna[1].

Kemudahan saat pengguna berinteraksi dengan User Interface (UI) dari website dalam rangka memenuhi UI yang sesuai kebutuhan pengguna diperlukan pendekatan untuk mengetahui karakteristik dari pengguna, melalui penerapan metode paper prototyping, dengan teknik wireframe mampu mengidentifikasi masalah pengguna dan rekomendasi dari konsumen langsung sekaligus merekam model UX yang ada, selain itu User Experience Website Profile Dengan Five Planes (Khalid Iskandar) metode paper prototyping yang digunakan dapat menciptakan sebuah pengalaman yang menyenangkan dalam UI[2].

B. User Experience (UX)

User Experience (UX) merupakan seluruh aspek yang berkaitan dengan pengalaman seorang pengguna dalam menggunakan sebuah produk, seberapa mudah cara kerjanya untuk dipahami, bagaimana perasaan ketika menggunakan produk, dan bagaimana pengguna mencapai tujuannya melalui produk. Sementara, salah satu bagian dari program yang bersentuhan dan berinteraksi langsung dengan user dinamakan User Interface (UI)[3].

Pembangunan user experience (UX) telah mendapatkan perhatian yang amat penting dalam dunia masa kini. Dimana user experience dapat mendukung keberhasilan dalam membangun suatu website, karena masih berkaitan dengan satu hal, yaitu bagaimana pengguna merasakannya[2].

C. ReactJS

Menurut Sanchit Aggarwal, ReactJS adalah pustaka JavaScript yang digunakan untuk mengembangkan komponen antarmuka pengguna (UI) yang dapat digunakan kembali. Berdasarkan dokumentasi resmi React, berikut adalah definisi React adalah perpustakaan untuk membangun antarmuka pengguna modular[4].

React pada dasarnya memungkinkan pengembangan aplikasi berbasis web yang besar dan kompleks yang dapat mengubah datanya tanpa perlu melakukan *refresh* pada halaman berikutnya. Ini digunakan sebagai View (V) di Model-View-Controller (MVC). React menggunakan Model Objek Dokumen (DOM), sehingga menawarkan kemudahan dan efektivitas dalam pengalaman pengembangan aplikasi. React sebagian besar dirender di sisi server menggunakan NodeJS, dan dukungan untuk aplikasi *mobile* menggunakan React Native[4].

Menurut Elar Saks, React merupakan salah satu pustaka JavaScript paling populer mengalahkan pustaka JavaScript lain seperti Angular dan Vue. React juga termasuk pustaka JavaScript yang paling mudah untuk dipelajari. Secara kinerja React juga mendapatkan hasil pengujian kinerja yang bagus, dari pengujian yang telah dilakukan antara tiga pustaka yaitu React, Angular, dan Vue. Peringkat pertama didapatkan oleh Vue, kemudian disusul React, dan yang terakhir Angular[5].

Berdasarkan hasil kajian pustaka didapatkan bahwa dalam pengembangan web peran UI dan UX yang baik sangatlah menentukan kenyamanan dan dapat meningkatkan pengalaman yang bagus dalam menggunakan website. Dalam UI UX tidak hanya fungsionalitasnya saja yang harus diperhatikan, namun tampilan dan performa dari website tersebut juga sangat menentukan pengalaman pengguna.

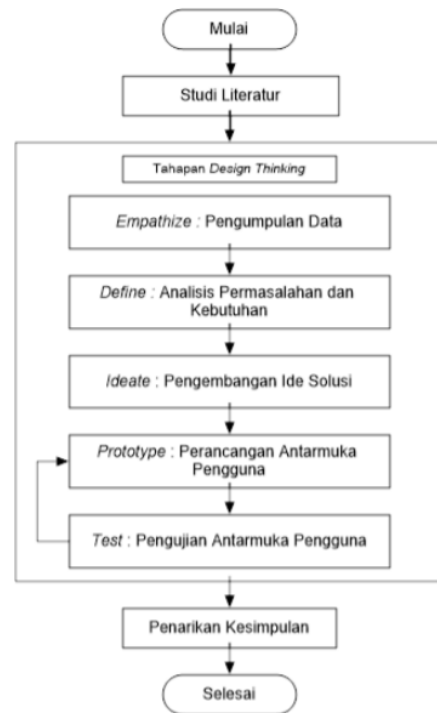
Dari sisi pengembang penggunaan ReactJs sangatlah membantu, selain mudah dipelajari, React juga mendapatkan hasil pengujian kinerja yang bagus, dan React cocok untuk digunakan dalam pembuatan website modular.

Oleh karena itu dalam pembuatan Modul Manajemen Pengguna ini digunakan ReactJs sebagai pustaka JavaScript sehingga diharapkan Modul yang dibuat nantinya akan mempunyai UI yang tidak hanya berfungsi dengan baik, namun juga mempunyai tampilan menarik dan responsif.

III. METODOLOGI

Dalam pembuatan Modul Website Manajemen Pengguna dikerjakan oleh satu tim yang beranggotakan tiga orang yang masing masing memiliki tugas pokok yang berbeda. Makalah ini membahas bagian front-end modul manajemen pengguna. Dalam melakukan perancangan Modul Management Pengguna ini, digunakan metode "*Design thinking*". Metode *design thinking* ini merupakan cara berpikir komprehensif yang konsentrasinya adalah untuk menemukan solusi, dimulai dari empati atas kebutuhan tertentu yang berpusat pada manusia (*human centered*) menuju perbaikan berkelanjutan demi memenuhi kebutuhan penggunanya. Semula ada tiga tahapan pokok dalam metode desain thinking, yaitu *inspiration*, *ideation*, *implementation*. *Inspiration* yaitu kebutuhan atau masalah yang memotivasi pencarian suatu solusi atau inovasi, *ideation* yaitu proses menghasilkan gagasan, pengembangan, dan pengujian gagasan, yang terakhir *implementation* yaitu finalisasi pemeranan ke pengguna. Dalam perkembangannya, ketiga tahapan dalam metode *design thinking* tersebut berkembang menjadi 5 tahapan yang pada dasarnya tidak berbeda jauh namun terdapat penekanan pada bagian tertentu sehingga menghasilkan prosedur yang lebih terperinci[6].

Berikut adalah gambar mengenai alur dan tahapan *design thinking*, dimulai dengan melakukan studi literatur untuk referensi sekaligus pendukung dalam melaksanakan pengembangan Modul Manajemen Pengguna.



Gambar 1. Alur Metodologi Design Thinking

Berikut detail tahapan *Design Thinking* yang dilakukan:

A. Pengumpulan Data

Pada tahap ini pengumpulan data dilakukan dengan wawancara langsung. Narasumber yang di wawancarai adalah pemimpin dalam proyek ini yaitu Bapak Septian.

B. Analisis Kebutuhan

Pada tahap ini mempelajari hal-hal yang berkaitan dan yang dibutuhkan dalam pengembangan Modul Manajemen Pengguna dipelajari. Mulai dari desain yang akan dibuat hingga fitur fitur yang akan dibuat. Hal ini dilakukan agar para pengembang memahami kebutuhan dan juga batasan dalam mengembangkan Modul ini.

C. Desain Antarmuka pengguna

Pada tahap ini dirancang desain antarmuka pengguna sesuai arahan dan desain yang telah diberikan kepada tim pengembang. Dalam merancang desain ini dilakukan pembagian layout, penyusunan menu, hingga isi konten.

D. Implementasi Desain ke dalam Kode Program

Pada tahap ini penulisan kode program dilakukan menggunakan bahasa pemrograman Java, menggunakan ReactJS sebagai pustaka untuk membangun antarmuka pengguna dan dipadukan dengan *template* bootstrap Admin LTE.

E. *Pengujian*

Pada Tahap ini dilakukan pengujian dengan beberapa test case untuk semua komponen, hal ini bertujuan untuk mengetahui apakah semua komponen telah sesuai dengan fungsi dan tampilannya sesuai dengan yang direncanakan.

IV. HASIL DAN PEMBAHASAN

A. *Pengumpulan Data*

Tahap analisis dilakukan dengan wawancara kepada Bapak Septian selaku pemimpin pada proyek ini. Hasil yang didapat kami diminta membuat modul aplikasi web Manajemen Pengguna yang dibangun menggunakan ReactJs.

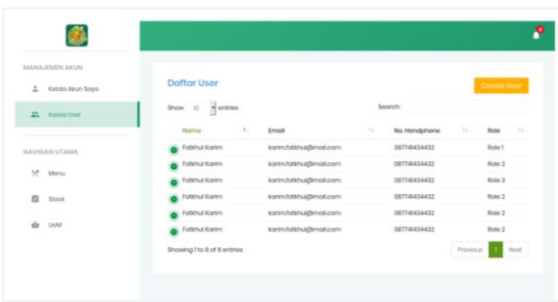
B. *Analisis Kebutuhan*

Pada desain yang diminta terdapat beberapa point yang akan dibuat yaitu:

1. Header dan Side Menu Bar
2. Konten Tabel untuk Menu Kelola *User*
3. Fitur Pendukung untuk Tabel (*Sorting, Paging, Entry Limit Show, dan Searching*)
4. Membuat tampilan *form* untuk menambah *user*
5. Membuat validasi pada form tambah *user*
6. Membuat tampilan untuk fitur edit dan delete

C. *Desain Antarmuka pengguna*

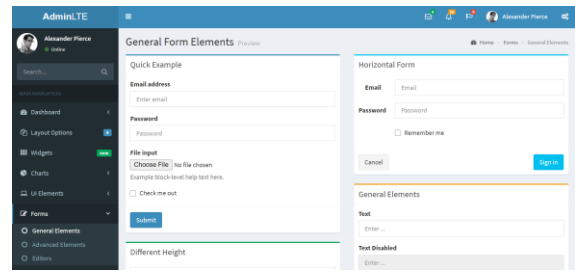
Berikut adalah desain halaman manajemen pengguna yang akan disempurnakan dan dibuat kode programnya.



Gambar 2. Desain Awal

D. *Implementasi Desain ke dalam Kode Program*

Setelah tahap perancangan selesai, tahapan berikutnya adalah penulisan kode program. Penulisan tersebut menggunakan bahasa Java menggunakan library atau pustaka dari ReactJS. Untuk mempermudah pembuatan antarmuka pengguna tersebut, digunakan juga *template* bootstrap dari Admin LTE.



Gambar 3. *Template* Bootstrap Admin LTE

Dengan *template* bootstrap seperti Admin LTE ini dapat meningkatkan efektivitas dan lebih efisien waktu. Selain itu penggunaan *template* Admin LTE ini juga berguna untuk menjaga konsistensi dalam pembuatan bagian-bagian dari antar pengguna itu sendiri.

Penulisan kode dilakukan menggunakan bahasa pemrograman Java menggunakan aplikasi Visual Studio Code menggunakan ReactJS sebagai pustakanya. Dengan React kita juga dapat menyisipkan kode HTML ke dalam Javascript. Hal ini disebut dengan JSX. JSX adalah singkatan dari JavaScript XML. Oleh karena itu kita dapat menuliskan kode HTML ke dalam Javascript menggunakan ekstensi JSX ini. Hal ini membuat kita lebih mudah karena pada saat kita membuat sebuah program React, pada dasarnya component-component yang dibuat di React akan mengembalikan data berupa jsx code, sehingga kita dapat menggunakannya berkali kali.

Berikut adalah kode program untuk bagian *Header, Side Menu bar dan Table Content* yang telah dibuat.

```

import React, { Component } from 'react'
import './index.css'

export default class Header extends Component {
  //description method untuk menampilkan header pada halaman dashboard
  render() {
    return (
      <div className="main-header">
        <img alt="index.html" className="logo"/>
        <span href="#login" className="login-link"/>
        <span href="#" className="dropdown-toggle"/>
        </div>
      </div>
    )
  }
}
    
```

Gambar 4. Code untuk Header

Pada pembuatan *header* dan *side menu bar*, penggunaan *template* bootstrap dari Admin LTE sangat membantu, antara lain untuk mengatur layout, pemilihan logo atau *icon*, dan membuat tampilan menjadi responsif.

```

project-manajuser-master > project-dashboard > src > assets > # menus.js
1 import React, { Component } from 'react'
2
3 export default class Menu extends Component {
4   /**
5    * @description method untuk menampilkan menu pada halaman dashboard
6    */
7   render() {
8     return (
9       <div>
10        <aside className="main-sidebar">
11          /* sidebar style can be found in sidebar.less */
12          <section className="sidebar">
13            /* sidebar menu : style can be found in sidebar.less */
14            <ul className="sidebar-menu" data-widget="tree">
15              <li></li>
16              <li className="header"><b>MANAJEMEN ARLIN</b></li>
17              <li></li>
18              <li><a href="pages/akunsaya.html">
19                <i className="fa fa-user"> <span>Kelola Akun Saya</span>
20              </li>
21              <li></li>
22              <li><a href="pages/kelolaper/content.html">
23                <i className="fa fa-users"> <span>Kelola User</span>
24              </li>
25              <li></li>
26              <li className="header"><b>NAVIGASI UTAMA</b></li>
27              <li></li>
28              <li><a href="pages/menu.html">
29                <i className="fa fa-cutlery"> <span>Menu</span>
30              </li>
31              <li></li>
32              <li></li>
33              <li><a href="pages/stock.html">
34                <i className="fa fa-check-square"> <span>Stock</span>
35              </li>
36              <li></li>
37            </ul>

```

Gambar 5. Code Side Menu Bar

```

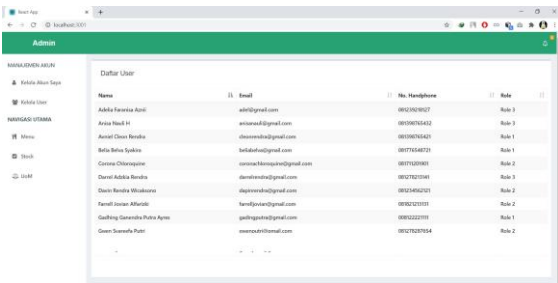
<div className="content-wrapper">
  /* main content */
  <section className="content">
    <div className="row">
      <div className="col-xs-12">
        /* box */
        <div className="box">
          <div className="box-header">
            <section className="content header">
              <h3>Daftar User</h3>
              <ol>
                <li><span>Create user</span> <button type="button" className="btn btn-warning" data-toggle="modal" data-target="#modal-default">
                  Create user
                </button>
              </ol>
            </div>
          </div>
          <div>
            <table id="example1" className="table table-bordered table-striped">
              <thead>
                <tr>
                  <th>Nama</th>
                  <th>Email</th>
                  <th>No. Handphone</th>
                  <th>Role</th>
                </tr>
              </thead>
              <tbody>
                <tr>
                  <td colspan="4">{this.loadfilldata()}
                </td>
              </tbody>
            </table>

```

Gambar 6. Code Konten Tabel Daftar User

Dalam pembuatan Konten Tabel menggunakan DataTable. Hal ini bertujuan agar nantinya kita menjadi lebih mudah untuk mengatur dan membuat fitur pendukung tabel. Dalam DataTable terdapat banyak sekali fitur yang kita bisa gunakan misalnya fitur *sorting* dan *searching*. Oleh karena itu DataTable sangat membantu dalam pembuatan konten tabel.

Berikut adalah tampilan antarmuka pengguna yang sudah jadi, Halaman ini memuat konten Daftar User dalam tabel.



Gambar 7. Halaman Daftar User

Pada halaman tersebut telah dibuat *header*, *side menu bar* dan konten daftar *user* yang berbentuk tabel sesuai dengan desain awal yang diberikan.

Setelah pembuatan konten tabel dilanjutkan dengan pembuatan fitur pendukung seperti *Sorting*, *Paging*, *Entry Limit Show*, dan *Searching*. Fitur pendukung tabel tersebut dibuat dengan fitur bawaan DataTable yang dimodifikasi agar sesuai dengan apa yang diperlukan.

Berikut adalah tampilan kode pengambilan Script untuk menggunakan fitur bawaan DataTable.

```

<!-- DataTables -->
<script src="PUBLIC_URL/bower_components/datatables.net/js/jquery.dataTables.min.js"></script>
<script src="PUBLIC_URL/bower_components/datatables.net-bs/js/dataTables.bootstrap.min.js"></script>

```

Gambar 8. Script untuk fitur DataTable

Berikut adalah code pembuatan fungsi untuk menampilkan fitur pendukung tabel.

```

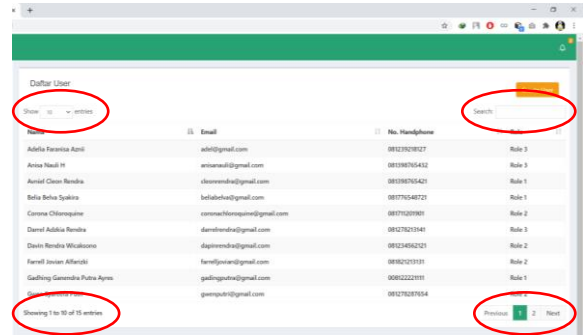
project-manajuser-master > project-dashboard > public > js > # content.js
1 $(function () {
2   $('#example1').DataTable()
3   $('#example2').DataTable({
4     'paging'      : true,
5     'lengthChange': false,
6     'searching'   : false,
7     'ordering'    : true,
8     'info'        : true,
9     'autoWidth'   : false
10  })
11 })

```

Gambar 9. Script untuk menampilkan fitur DataTable

Dengan fitur bawaan DataTable tersebut pembuatan *Sorting*, *Paging*, *Entry Limit Show*, dan *Searching* menjadi lebih mudah tanpa harus membuat satu persatu fiturnya.

Berikut adalah tampilan halaman daftar pengguna setelah tabel daftar *user* ditambah dengan fitur *Sorting*, *Paging*, *Entry Limit Show*, dan *Searching*.



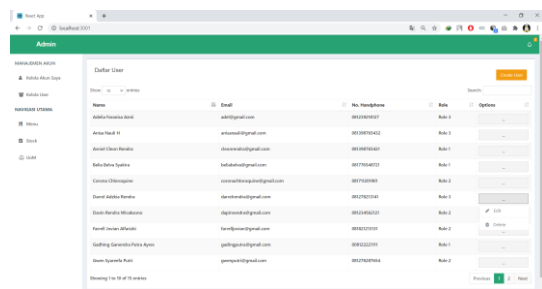
Gambar 10. Fitur Pendukung tabel

Fungsi *Sorting* adalah untuk mengurutkan data pada tabel. Fungsi *Paging* adalah untuk membuat tabel ditampilkan dalam beberapa halaman agar lebih mudah dibaca. Fungsi *Entry Limit Show* adalah untuk membatasi berapa jumlah baris yang ditampilkan pada setiap halaman tabel. Fungsi *Searching* adalah untuk mencari data dalam tabel menggunakan kata kunci yang diinginkan, sehingga mempercepat dalam pencarian data.

Setelah pembuatan fitur pendukung tabel, tahap selanjutnya adalah pembuatan tombol edit dan delete.

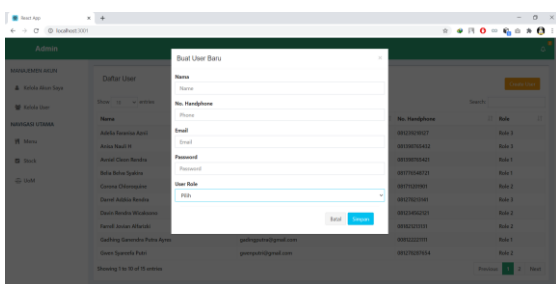
Kedua tombol ini adalah bentuk aksi yang dapat dilakukan pada setiap baris pada tabel. Fungsi edit adalah untuk mengubah data yang ada pada tabel. Saat tombol edit ditekan maka akan muncul Modal yang berisikan form data yang akan diubah dan selanjutnya akan ada tombol batal yang berfungsi untuk membatalkan pengeditan dan tombol simpan untuk menyimpan perubahan data yang sudah dilakukan. Untuk fungsi *delete* merupakan fungsi untuk menghapus data data-data yang akan dihapus pada tabel ketika tombol *delete* ditekan.

Berikut adalah tampilan untuk tombol *Edit* dan *Delete*.



Gambar 11. Tombol Edit dan Delete

Untuk menambahkan data pengguna baru dibutuhkan form untuk memasukkan data dari pengguna baru tersebut. Oleh karena itu dibuatlah tombol *Create User* yang berfungsi untuk menambahkan pengguna baru. Ketika tombol ini ditekan maka akan muncul modal berisi form pendaftaran pengguna baru. Dalam modal ini juga terdapat dua tombol lagi yaitu batal dan simpan. Tombol batal berfungsi untuk membatalkan penambahan pengguna, sedangkan tombol simpan berfungsi untuk menyimpan data pengguna sebagai pengguna baru yang akan langsung ditampilkan pada tabel.



Gambar 12. Tampilan Form untuk membuat User baru

Dalam pengisian form pendaftaran dibutuhkan validasi untuk setiap isian yang dimasukkan. Setiap isian harus sesuai dengan aturan agar terhindar dari kesalahan data yang dimasukkan ataupun ada form yang tidak diisi.

Dalam Form tersebut Validasi dibuat untuk semua isian form, berikut adalah contoh validasinya,

1. Nama hanya boleh menggunakan Huruf dan Sepasi (Letter only)

2. No Handphone hanya boleh berupa angka dan termasuk No. Telp Indonesia.
3. Email harus valid
4. Password harus memuat minimal satu huruf kecil, satu huruf besar, satu angka dan satu simbol. Dan panjangnya minimal 8 karakter dan maksimal 15 karakter.

Berikut adalah tampilan code untuk validasi input data pada form pendaftaran.

```

//Name
if(!fields["caption"]){
  formIsValid = false;
  errors["caption"] = "Cannot be empty";
}

if(typeof fields["caption"] != "undefined"){
  if(!fields["caption"].match(/^[a-zA-Z\s]+$/)){
    formIsValid = false;
    errors["caption"] = "Only letters";
  }
}

//Email
if(!fields["caption"]){
  formIsValid = false;
  errors["caption"] = "Cannot be empty";
}

if(typeof fields["caption"] != "undefined"){
  let lastAtPos = fields["caption"].lastIndexOf('@');
  let lastDotPos = fields["caption"].lastIndexOf('.');

  if ((lastAtPos < lastDotPos && lastAtPos > 0 && fields["caption"].indexOf('@') == -1 && lastDotPos > 2 && fields["caption"].length > 15)) {
    formIsValid = false;
    errors["caption"] = "Email is not valid";
  }
}

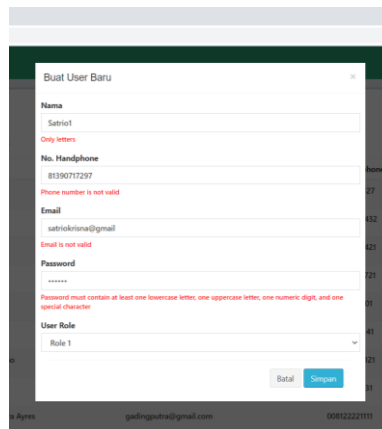
//Password
if(!fields["password"]){
  formIsValid = false;
  errors["password"] = "Cannot be empty";
}

if(typeof fields["password"] != "undefined"){
  if(!fields["password"].match(/^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!@#$%^&*~`{|}~\s]).{8,15}$/)){

```

Gambar 13. Code untuk validasi input data

Validasi ini juga digunakan pada saat fungsi edit dijalankan agar data yang diedit tetap divalidasi kebenarannya. Lalu apabila data yang dimasukkan tidak sesuai ataupun kosong maka akan menampilkan pesan sesuai dengan kesalahannya seperti gambar dibawah.



Gambar 14. Tampilan validasi dalam pengisian form

Setelah form diisi dan tombol submit ditekan maka akan dilakukan validasi terhadap masukan dari pengguna. Apabila terdapat 1 atau lebih form yang tidak diisi atau salah dalam format pengisiannya maka sistem akan menampilkan pesan error berwarna merah sesuai dengan kesalahan yang ada.

E. Pengujian

Pengujian dilakukan bersama dengan 2 orang anggota tim yang lain. Fokus pengujian adalah mengetahui apakah

semua komponen telah sesuai dengan tampilan dan fungsinya sesuai dengan yang diinginkan. Mengutip dari jurnal Anggota tim yang mengerjakan proyek ini, yang bernama Anissa Nauli, dengan jurnal berjudul “Pengujian dengan Unit Testing dan Test case pada Proyek Pengembangan Modul Manajemen Pengguna” didapatkan hasil bahwa semua fitur atau fungsionalitas dari modul yang telah dibuat sudah sesuai dengan dengan spesifikasi yang diberikan oleh perusahaan kepada tim pengembang.

Tabel 1. STATISTIK DATA PENGUJIAN

Fitur	Jumlah <i>Test Case</i>	Persentase Lulus Uji
<i>Create User</i>	5	100 %
<i>Edit User</i>	5	100 %
<i>Delete User</i>	2	100 %
<i>Search</i>	4	75%
<i>Paging</i>	1	100 %
<i>Entry limit</i>	1	100 %

Tabel 1 merupakan statistik data hasil pengujian dengan *test case* untuk semua fitur atau fungsionalitas dari modul Manajemen Pengguna[7].

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan Pengembangan modul manajemen pengguna menggunakan ReactJs di atas. Dapat disimpulkan bahwa:

1. Antarmuka pengguna (UI) yang baik harus memenuhi beberapa syarat agar pengalaman pengguna (UX) dalam menggunakan aplikasi menjadi maksimal, antara lain *interface* yang jelas, ringkas, familiar, responsif, konsisten, estetik, efisien, dan penanganan kepada kesalahan.
2. ReactJs dapat digunakan untuk membuat modul

aplikasi web yang mudah digunakan kembali setiap komponen komponennya dengan sedikit modifikasi.

3. Banyak fitur pada ReactJs yang memudahkan untuk pengembangan aplikasi web. Misalnya ekstensi JSX yang memungkinkan kita untuk menuliskan kode HTML ke dalam Javascript.
4. Penggunaan *template* bootstrap Admin LTE dapat meningkatkan efektivitas dan lebih efisien waktu dalam pengembangan aplikasi web. Selain itu penggunaan *template* Admin LTE ini juga berguna untuk menjaga konsistensi dalam pembuatan bagian-bagian dari antar pengguna itu sendiri.

B. Saran

Adapun saran guna memperbaiki atau menyempurnakan pengembangan selanjutnya adalah meningkatkan desain dengantampilan yang lebih baru dan animasi agar tampilan menarik lagi. Serta masih ada pengujian yang masih belum lulus, yaitu bagian *Test case* pada fitur *search*.

VI. REFERENSI

- [1] I. Febrianta Surya Nugraha *et al.*, “Perancangan Antarmuka Sistem Pakar Penyakit Padi Berbasis Web Design for Expert System Interface of Web-Based Rice Disease,” vol. 7, no. 2, p. 143, 2017.
- [2] L. Hardiansyah and K. Iskandar, “Perancangan User Experience Website Profil Dengan Metode The Five Planes (Studi kasus: BP3K Kecamatan Mundu),” *Jurnal Ilmiah INTECH (Information Technology Journal) of UMUS*, vol. 01, no. 01, pp. 11–21, 2019.
- [3] A. Ar Razi *et al.*, “Penerapan Metode Design Thinking pada Model Perancangan Ui/Ux Aplikasi Penanganan Laporan Kehilangan dan Temuan Barang Tercecer,” *Jurnal Desain Komunikasi Visual*, vol. 03, no. 02, 2018, [Online]. Available: <http://bit.do/demandia>
- [4] S. Aggarwal, “Modern Web-Development using ReactJS,” *International Journal of Recent Research Aspects*, vol. 5, pp. 133–137, 2018.
- [5] E. Sask, “JavaScript frameworks: Angular vs React vs Vue,” 2019.
- [6] A. Mursyidah, I. Aknuranda, and H. Muslimah Az-Zahra, “Perancangan Antarmuka Pengguna Sistem Informasi Prosedur Pelayanan Umum Menggunakan Metode Design Thinking (Studi Kasus: Fakultas Ilmu Komputer Universitas Brawijaya),” vol. 3, no. 4, pp. 3931–3938, 2019, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [7] A. N. Hasibuan and T. Dirgahayu, “Pengujian dengan Unit Testing dan Test case pada Proyek Pengembangan Modul Manajemen Pengguna,” 2020.