

# Pemrograman Web Lanjut Tugas 1

PWL-TI-20-PA-202310055

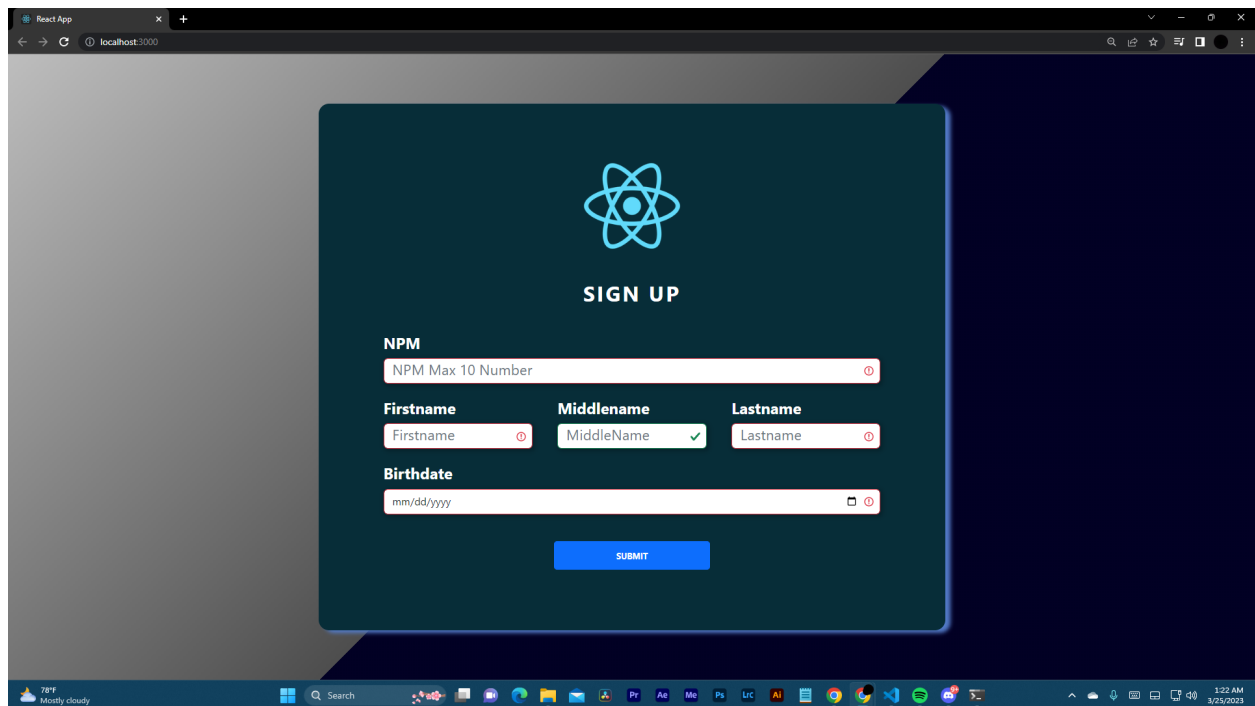
React-Bootstrap

Nama : Ryan Roshandy Sachio

NPM : 202310055

Kelas : 20-TI-PA2

## Tampilan Web Validation



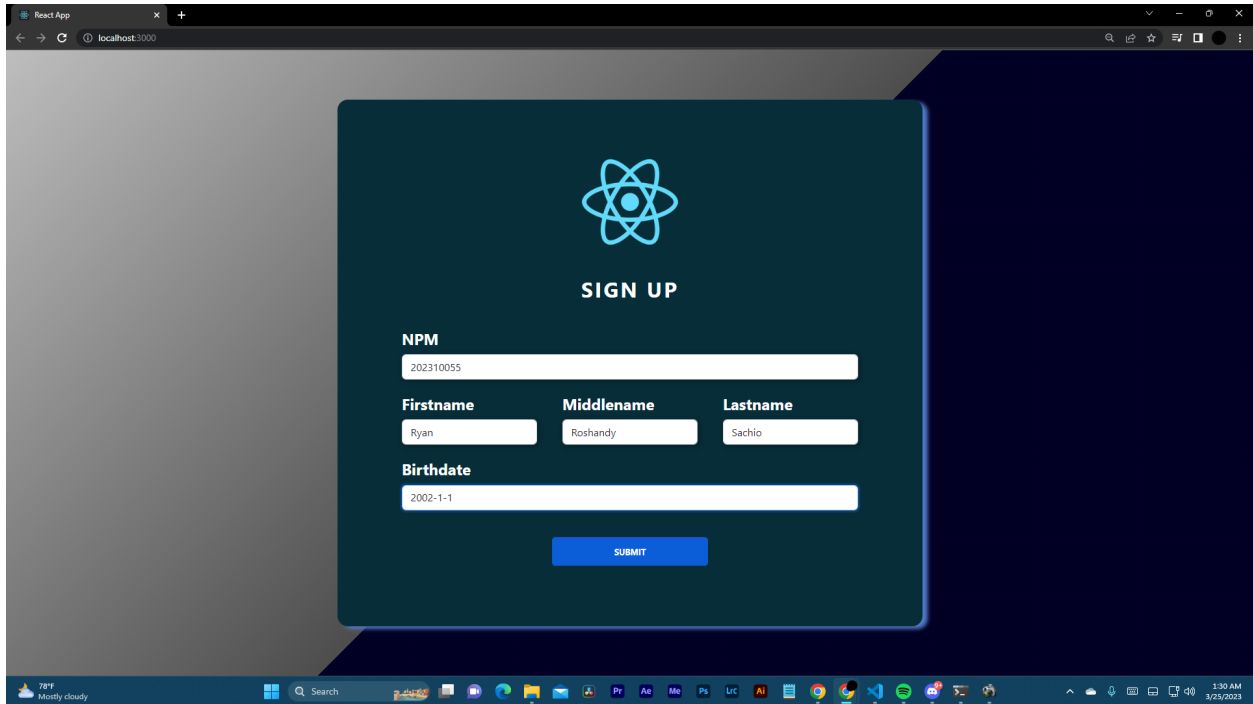
The screenshot shows a web browser window displaying a sign-up form. The form is titled "SIGN UP" and features the React logo. It includes the following fields:

- NPM**: A text input field with the placeholder "NPM Max 10 Number". It has a red error icon on the right.
- Firstname**: A text input field with the placeholder "Firstname". It has a red error icon on the right.
- Middlename**: A text input field with the placeholder "MiddleName". It has a green checkmark icon on the right, indicating it is valid.
- Lastname**: A text input field with the placeholder "Lastname". It has a red error icon on the right.
- Birthdate**: A date input field with the placeholder "mm/dd/yyyy". It has a red error icon on the right.

Below the fields is a blue "SUBMIT" button. The background of the form is dark teal with a blue gradient. The browser window shows the URL "localhost:3000" and the Windows taskbar at the bottom.

Ketika *value* "NPM", "*Firstname*", "*Lastname*" dan "*Birthdate*" kosong maka form akan memvalidasi bahwa *value* bernilai *false* dan memberikan peringatan di sebelah kanan *input box* dan tidak akan menghasilkan *output pop-up modal*.

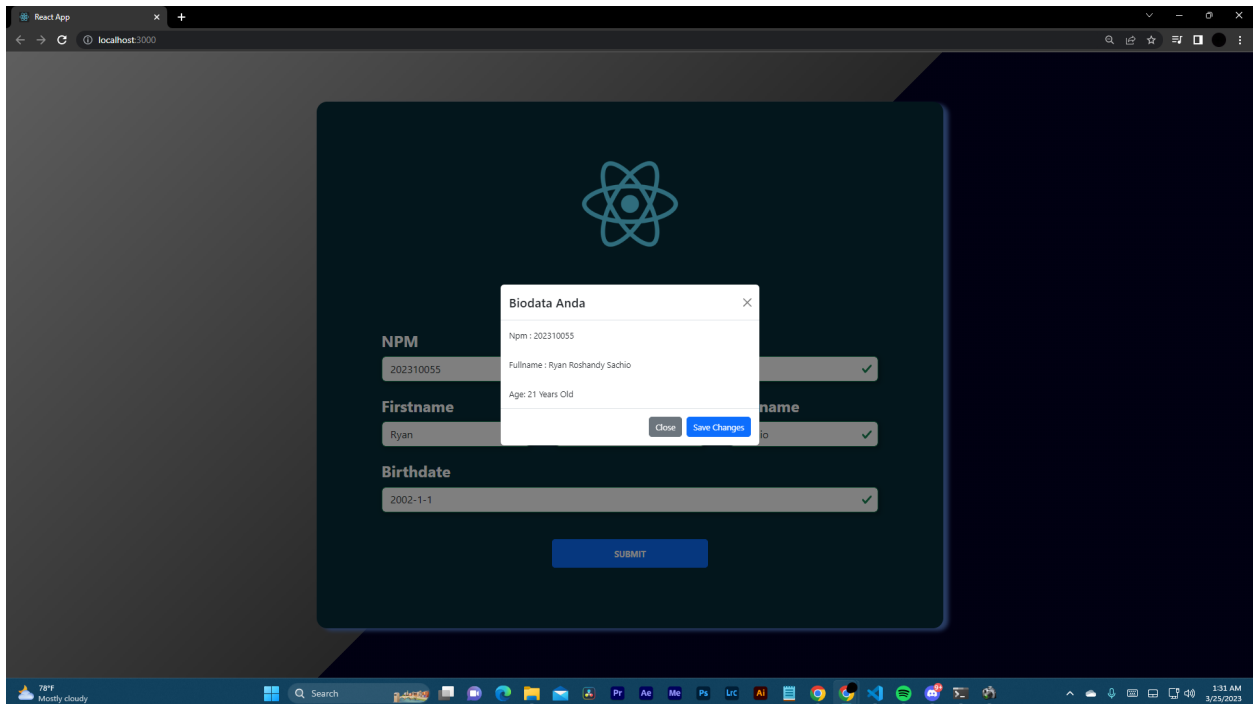
## Tampilan hasil web



A screenshot of a web browser displaying a "SIGN UP" form. The form is centered on a dark blue background with a React logo at the top. The form fields are as follows:

- NPM**: A single-line text input containing "202310055".
- Firstname**: A single-line text input containing "Ryan".
- Middlename**: A single-line text input containing "Roshandy".
- Lastname**: A single-line text input containing "Sachio".
- Birthdate**: A single-line text input containing "2002-1-1".

Below the fields is a blue "SUBMIT" button. The browser's address bar shows "localhost:3000". The Windows taskbar at the bottom shows the time as 1:30 AM on 3/25/2023.



A screenshot of the same web application, but now a modal window titled "Biodata Anda" is open over the form. The modal displays the following information:

- Npm**: 202310055
- Fullname**: Ryan Roshandy Sachio
- Age**: 21 Years Old

The modal has a "Close" button and a "Save Changes" button. The background form fields are visible but slightly dimmed. The "SUBMIT" button is still at the bottom of the form. The browser and taskbar details are the same as in the first screenshot.

## Code

```
import { useState } from 'react';
// import 'bootstrap/dist/css/bootstrap.min.css';
import { Form, Container, Button, Col, Row, Modal } from
'react-bootstrap';
import '../App.css';

function ValidateFo () {
  const date = new Date();
  const currentYear = date.getFullYear();

  const initialValues = { Firstname: "", Middlename: "", Lastname: "",
email: "", password: "", Birthdate: "" };
  const [ NPM, setNPM ] = useState(0);
  const [ formValues, setFormValues ] = useState(initialValues);
  // const [ formErrors, setFormErrors ] = useState("");
  // const [ isSubmit, setIsSubmit ] = useState(false);
  const [ show, setShow ] = useState(false);
  const [ validated, setValidated ] = useState(false);

  const handleChange = (e) => {
    const { name, value } = e.target;
    setFormValues({ ...formValues, [name]: value });
  };

  const handleSubmit = (e) => {
    const form = e.currentTarget;
    if (form.checkValidity() === false) {
      e.preventDefault();
      e.stopPropagation();
    } else {
      e.preventDefault();
      setShow(true);
    }
    setValidated(true);
  };

  const handleClose = () => setShow(false);
  // const handleShow = () => setShow(true);
```

```

const Age = currentYear - parseInt(formValues.Birthdate.slice(0, 4));

return (
  <div className='Content-header'>
    <div className='Con'>
      <Container fluid>
        <Form noValidate validated={validated} onSubmit={handleSubmit}>
          <div style={{textAlign:"center"}}>
            <img style={{alignItems:'center'}}
src={process.env.PUBLIC_URL + "/logo192.png"} width="192" alt="logo" />

            <h1 id='TitleForm' className='Head mt-5'>Sign Up</h1>
          </div>
          <Form.Group className='mb-3' controlId='formGridNpm'>
            <div className='Field-1'>
              <Form.Label>NPM</Form.Label>
              <Form.Control
                size='lg'
                name="NPM"
                placeholder="NPM Max 10 Number"
                maxLength={10}
                // value={formValues.npm}
                onChange={(e) => setNPM(e.target.value)}
                onKeyDown={(event) => {
                  if (!/[0-9]/.test(event.key)) {
                    event.preventDefault();
                  }
                }}
                required
              />
            </div>
            <Row>
              <Col>
                <div className='Field-2'>
                  <Form.Label>Firstname</Form.Label>
                  <Form.Control
                    size='lg'
                    className='Input-control'
                    type='text'
                    name='Firstname'

```

```

        placeholder='Firstname'
        value={formValues.Firstname}
        onChange={handleChange}
        required
      />
    </div>
  </Col>
  <Col>
    <div className='Field-2'>
      <Form.Label>Middlename</Form.Label>
      <Form.Control
        size='lg'
        className='Input-control'
        type='text'
        name='Middlename'
        placeholder='MiddleName'
        value={formValues.Middlename}
        onChange={handleChange} />
    </div>
  </Col>
  <Col>
    <div className='Field-2'>
      <Form.Label>Lastname</Form.Label>
      <Form.Control
        size='lg'
        className='Input-control'
        type='text'
        name='Lastname'
        placeholder='Lastname'
        value={formValues.Lastname}
        onChange={handleChange}
        required
      />
    </div>
  </Col>
</Row>
<div className='Field-3'>
  <Form.Label>Birthdate</Form.Label>
  <Form.Control
    size='lg'

```

```

        className='Input-control datepicker'
        placeholder='YYYY-MM-DD'
        name='Birthdate'
        value={formValues.Birthdate}
        onChange={handleChange}
        required
      />
    </div>
    <div className='Submit'>
      <Button className='Submitbut'
type='submit'><strong>SUBMIT</strong></Button>
    </div>
  </Form.Group>
</Form>
</Container>
</div>
<Modal show={show} onHide={handleClose} centered>
  <Modal.Header closeButton>
    <Modal.Title>Biodata Anda</Modal.Title>
  </Modal.Header>
  <Modal.Body>Npm : {NPM} </Modal.Body>
    <Modal.Body>Fullname : {formValues.Firstname}
{formValues.Middlename} {formValues.Lastname} </Modal.Body>
    <Modal.Body>Age: {Age} Years Old</Modal.Body>
  <Modal.Footer>
    <Button variant="secondary" onClick={handleClose}>
      Close
    </Button>
    <Button variant="primary" onClick={handleClose}>
      Save Changes
    </Button>
  </Modal.Footer>
</Modal>
</div>
);
}

export default ValidateFo;

```

=====

=====

=====

=====

=====

=====

=====

```
import { useState } from 'react';
import 'bootstrap/dist/css/bootstrap.min.css';
import '../App.css';
import { Form, Container, Button, Col, Row, Modal } from
'react-bootstrap';

function ValidateFo () {
  const date = new Date();
  const currentYear = date.getFullYear();

  const initialValues = { Firstname: "", Middlename: "", Lastname: "",
email: "", password: "", Birthdate: "" };
  const [ NPM, setNPM ] = useState(0);
  const [ formValues, setFormValues ] = useState(initialValues);
  // const [ formErrors, setFormErrors ] = useState("");
  // const [ isSubmit, setIsSubmit ] = useState(false);
  const [ show, setShow ] = useState(false);
  const [ validated, setValidated ] = useState(false);

  const handleChange = (e) => {
    const { name, value } = e.target;
    setFormValues({ ...formValues, [name]: value });
  };

  const handleSubmit = (e) => {
    const form = e.currentTarget;
    if (form.checkValidity() === false) {
      e.preventDefault();
      e.stopPropagation();
    } else {
      e.preventDefault();
      setShow(true);
    }
    setValidated(true);
  };

  const handleClose = () => setShow(false);
  // const handleShow = () => setShow(true);

  const Age = currentYear - parseInt(formValues.Birthdate.slice(0, 4));
```



```

return (
  <div>
    <div className='Content-header'>
      <div className='Con'>
        <Container>
          <Form noValidate validated={validated}
onSubmit={handleSubmit}>
            <div style={{textAlign:"center"}}>
              <img style={{alignItems:'center'}}
src={process.env.PUBLIC_URL + "/logo192.png"} width="192" alt="logo" />

              <h1 id='TitleForm' className='Head mt-5'>Sign Up</h1>
            </div>
            <Form.Group className='mb-3' controlId='formGridNpm'>
              <div className='Field-1'>
                <Form.Label>NPM</Form.Label>
                <Form.Control
size='lg'
name="NPM"
placeholder="NPM Max 10 Number"
maxLength={10}
onChange={ (e) => setNPM(e.target.value)}
onKeyPress={ (event) => {
                  if (!/[0-9]/.test(event.key)) {
                    event.preventDefault();
                  }
                }}
required
              />
            </div>
            <Row>
              <Col>
                <div className='Field-2'>
                  <Form.Label>Firstname</Form.Label>
                  <Form.Control
size='lg'
className='Input-control'
type='text'
name='Firstname'

```

```
        placeholder='Firstname'
        value={formValues.Firstname}
        onChange={handleChange}
        required
      />
    </div>
  </Col>
  <Col>
    <div className='Field-2'>
      <Form.Label>Middlename</Form.Label>
      <Form.Control
        size='lg'
        className='Input-control'
        type='text'
        name='Middlename'
        placeholder='MiddleName'
        value={formValues.Middlename}
        onChange={handleChange} />
    </div>
  </Col>
  <Col>
    <div className='Field-2'>
      <Form.Label>Lastname</Form.Label>
      <Form.Control
        size='lg'
        className='Input-control'
        type='text'
        name='Lastname'
        placeholder='Lastname'
        value={formValues.Lastname}
        onChange={handleChange}
        required
      />
    </div>
  </Col>
</Row>
<div className='Field-3'>
  <Form.Label>Birthdate</Form.Label>
  <Form.Control
    size='lg'
```

```

        className='Input-control datepicker'
        placeholder='YYYY-MM-DD'
        name='Birthdate'
        value={formValues.Birthdate}
        onChange={handleChange}
        required
      />
    </div>
    <div className='Submit'>
      <Button className='Submitbut'
type='submit'><strong>SUBMIT</strong></Button>
    </div>
  </Form.Group>
</Form>
</Container>
</div>
</div>

    <Modal show={show} onHide={handleClose}
aria-labelledby="contained-modal-title-vcenter" centered>
  <Modal.Header closeButton>
    <Modal.Title>Biodata Anda</Modal.Title>
  </Modal.Header>
  <Modal.Body className='show-grid'>
    <Container>

      <Row>
        <Col>NPM</Col>
        <Col>:</Col>
        <Col>{NPM}</Col>
      </Row>

      <Row>
        <Col>Fullname</Col>
        <Col>:</Col>
        <Col>{formValues.Firstname} {formValues.Middlename}
{formValues.Lastname} </Col>
      </Row>

      <Row>
        <Col>Age</Col>

```

```
        <Col>:</Col>
        <Col>{Age}th</Col>
    </Row>
</Container>
</Modal.Body>

<Modal.Footer>
    <Button variant="secondary" onClick={handleClose}>
        Close
    </Button>
    <Button variant="primary" onClick={handleClose}>
        Save Changes
    </Button>
</Modal.Footer>
</Modal>
</div>
);
}

export default ValidateFo;
```