React Material UI

* React Material UI is a popular open-source library of React components.
* It follows Google's Material Design guidelines.
* It provides a set of reusable UI components such as buttons, forms, modals, and more that can be easily integrated into your React application.
* Material UI makes it easy to create beautiful, responsive, and accessible user interfaces.
* It comes with a rich set of features such as theming, custom styling, server-side rendering, and more.
* Material UI also supports React Hooks and Type Script, which can help you write better, more maintainable code.

If you are building a React application and want to create a modern and consistent user interface, React Material UI can be a great choice.

**Advantages:**

1. Consistent and Modern Design.
2. Reusable Components.
3. Rapid development.
4. Beautiful by default.
5. Customizability and Flexible.
6. Cross-team Collaboration.
7. Accessibility – Designed to use by disabilities people as well, such as those who use screen readers, keyboard navigation, or other assistive technologies
8. Most Trusted and Active Community.

**MUI Core** contains foundational React UI component libraries for shipping new features faster.

* [**Material UI**](https://mui.com/material-ui/getting-started/overview/)is a comprehensive library of components that features our implementation of Google's [Material Design](https://m2.material.io/design/introduction/) system.
* [**Joy UI**](https://mui.com/joy-ui/getting-started/overview/) is a beautifully designed library of React UI components.
* [**MUI Base**](https://mui.com/base/getting-started/overview/) is our library of "unstyled" components and low-level hooks. With Base, you gain complete control over your app's CSS and accessibility features.
* [**MUI System**](https://mui.com/system/getting-started/overview/) is a collection of CSS utilities to help you rapidly lay out custom designs.

**Installation:**

* npm install @mui/material @emotion/react @emotion/styled
* npm install @mui/joy @emotion/react @emotion/styled
* npm install @mui/base
* npm install @mui/system @emotion/react @emotion/styled

**With Styles Components:**

* npm install @mui/material @mui/styled-engine-sc styled-components

**Example:**

import \* as React from 'react';

import Button from '@mui/material/Button';

function App() {

return <Button variant="contained">VKC</Button>;

}

Ex:

import { Container, Paper, TextField, Box, styled, Button } from "@mui/material";

const StyledBox = styled(Box)(()=>({

    padding: "10px",

    margin: "10px"

}))

const MatRegistration = () => {

    return (

        <Container maxWidth="md">

            <Paper elevation={10}>

                <StyledBox>

                    <TextField label="User Name" fullWidth margin="normal" required focused></TextField>

                    <TextField type="password" fullWidth label="Pasword" margin="normal" required></TextField>

                    <Box>

                        <Button variant="contained" fullWidth>Sign In</Button>

                    </Box>

                </StyledBox>

            </Paper>

        </Container>

    )

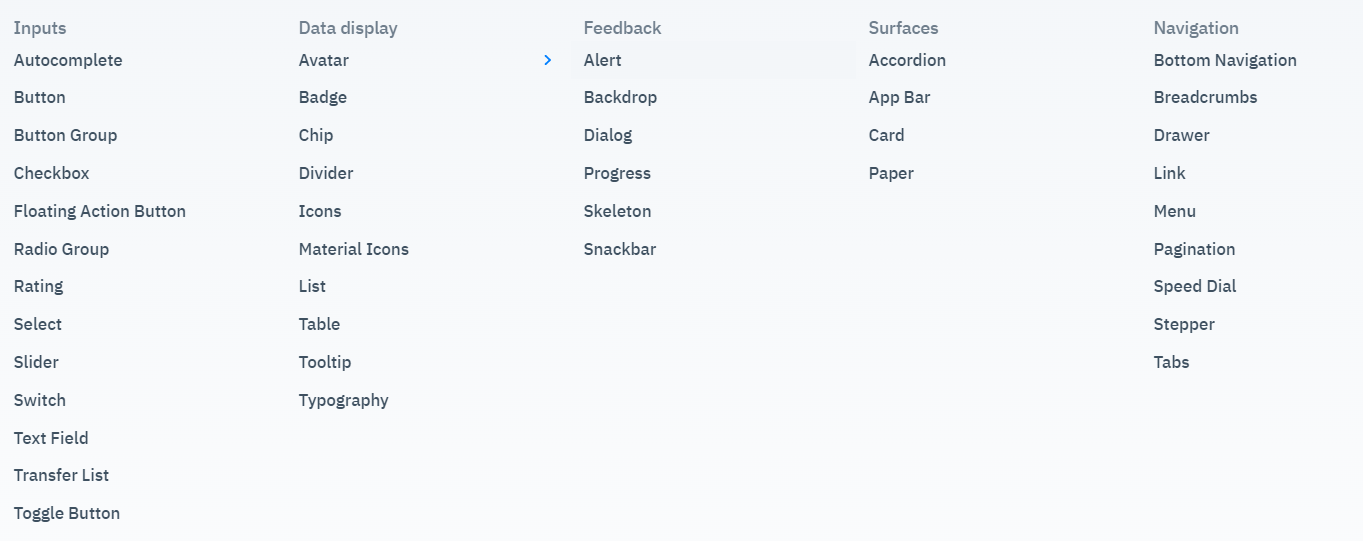
}

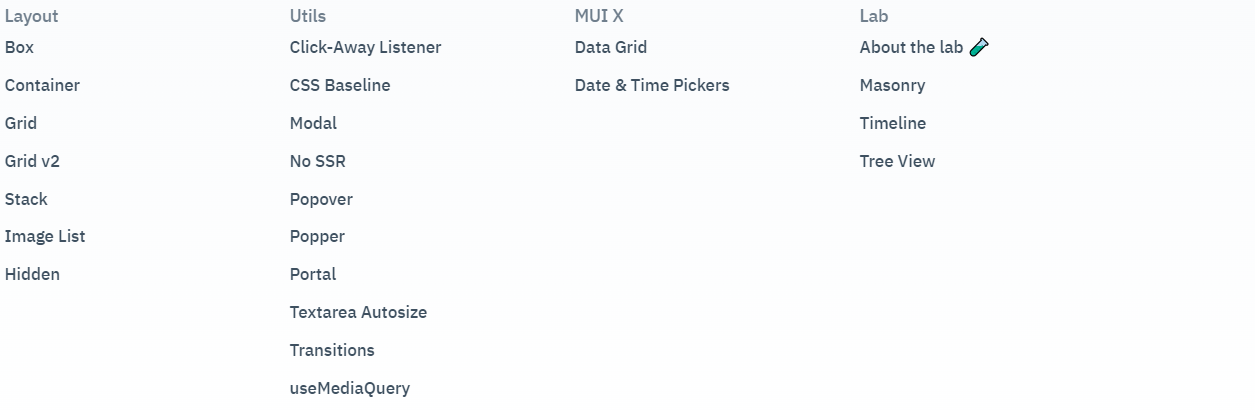
export default MatRegistration;

**Components:** Material UI provides the each feature as a component.

Types:

* 1. Inputs
  2. Data Display
  3. Feedback
  4. Surfaces
  5. Navigation
  6. Layout
  7. Utils
  8. MUI X
  9. Lab





sx -> System properties are available via sx. The sx props allows to specify the css props.

**Box:** A division element by default.

* Change element by **component** prop.

Ex:

<Box component="span">

     <Button variant="contained" fullWidth>Sign In</Button>

 </Box>

<Box component="button" variant="contained" color="red">Box Sign In</Box>

<Box component="button" sx={{ border: '1px dashed grey' }}>Save</Box>

Box supports all the system properties.

<https://mui.com/system/properties/>

**Container:** Container centers our content horizontally.

Fluid -> specify the maxWidth prop as per reolution.

maxWidth -> sm, md, lg, xl, false -> disable maxWidth.

Ex: <Container maxWidth="sm">

Fixed -> Auto accommodate the size as per resolution.

maxWidth matches the minWidth.

Ex: <Container fixed>

Defaut container is a **div,** by specifying component it will convert.

Ex: <Container fixed component="label">

**AppBar:** Add top banner. It’s used for branding, titles, navigations and actions.

* Always at top fixed position.

**Toolbar**

**ButtonGroup**

**Button**