**ORXE-Input**

### Document Properties

| Item | Details |
| --- | --- |
| Document | Technical Specification Document |
| Application | ORXE-Input |
| Application Version | 0.0.3 |
| Author | Pranati Ghorpade |
| Creation Date | 20/09/2020 |
| Last Updated | 21/09/2020 |
| Status | Completed |

Contents

[Document Properties 1](#_Toc51609767)

[1. Component Overview 3](#_Toc51609768)

[1.1 Objective 3](#_Toc51609770)

[1.2 Component Diagram 4](#_Toc51609771)

[1.3 Use Cases 5](#_Toc51609772)

[1.3 Risk 5](#_Toc51609773)

[1.4 Out of Scope 5](#_Toc51609774)

[2. Technical Specifications 5](#_Toc51609775)

[2.1 NPM details 7](#_Toc51609776)

[2.2 NPM repo 8](#_Toc51609777)

[3.Unit Testing 8](#_Toc51609779)

[4. Assumptions 9](#_Toc51609780)

[5. Concerns and Issues …………….. 9](#_Toc51609781)

1. Component Overview

This component provides overview of orxe-input component. An input field is used to accept data from a user in the form of a set of characters. Based on the type of input field and property like error, active, focus it will show user message and corresponding style will be applied to input box.

1.1 Objective

1. To build the reusable component.
2. Allowing the user to enter a set of characters, including alphabets, numbers, and special characters; based on the context of the use case.

1.2 Component Diagram

|  |  |
| --- | --- |
| Input Field – Default | Input Field – Mandatory |
|  |  |
| Input Field – Focus | Input Field – Active |
|  |  |
| Input Field - Filled | Input Field - Error |
|  |  |

1.3 Use Cases

Following are the different scenarios where an input field is used:

1. Search Screen: The input field is used to allow a user to search for information.
2. Select Screen: The input field is used to select the required information which is relevant for the user.
3. Date Screen: The input field is used to pick the date listed in the calendar.
4. Price Range: The input field is used to display the price range selected by the user.
5. Shell Screen: The input field is used to add and edit the profile information of a user.
6. Cart Screen: The input field is used to enter and review travelers' information and payment details to complete a booking.

1.3 Risk

NA

1.4 Out of Scope

Input validations are not implemented.

2. Technical Specifications

Following are the technical specification for the orxe-input component:

Properties:

1. **Class**: Classes are dynamically applied to the input field depending on the state of input.

Ex: if input state is error then class will be “input-class error”.

**Default Input: <orxe-input label="Username" type="string" value="" placeholder="Placeholder"> </orxe-input>**

1. **Active Input (isActive)**: This will need to set true if input field is active, else false. And class will be “input-class active”.

**<orxe-input label="Username" type="string" value="" isActive="true" placeholder="Placeholder"> </orxe-input>**

1. **Error Input (isError)**: This will need to set true if input field is in error state, else false. And class will be “input-class error”.

**<orxe-input label="Username" type="string" value="" required="true" isError="true" placeholder="Placeholder"> </orxe-input>**

1. **Filled Input (isFilled)**: This will need to set true if input field is in filled state, else false. And class will be “input-class filled”.

**<orxe-input label="Username" type="string" value="Tavisca" isFilled="true" placeholder="Placeholder "> </orxe-input>**

1. **On Focus Input (isFocus)**: This will need to set true if input field is in focus state, else false. And class will be “input-class focus”.

**<orxe-input label="Username" type="string" value="" isFocus="true" placeholder="Placeholder"></orxe-input>**

1. **Disabled Input (disabled)**: If this attribute is set to true then input field will be in disable state else it will be in enable state.

**<orxe-input label="Username" type="string" value="" disabled="true" placeholder="Placeholder"> </orxe-input>**

1. **Mandatory Input (required)**: If this attribute is set to true then input field will be mandatory.

**<orxe-input label="Username\*" type="string" value="" placeholder="Placeholder">**

**</orxe-input>**

1. **readonly**: If this attribute is set to true then input field will be in readonly state.

**<orxe-input label="Username" type="string" value="Tavisca" readonly="true" placeholder="Placeholder"> </orxe-input>**

1. **type**: User can set the type of input field as string, number etc.
2. **min**: In case of number type this property will be applicable to input. And user can select minimum this value.
3. **max**: In case of number type this property will be applicable to input. And user can select maximum this value.

|  |  |
| --- | --- |
| **Properties** | **Value** |
| Border Color | #6A7078 |
| Border Radius | 4px |
| Border Thickness | 1px |
| Component Height | 56px |
| Error Font Weight | Semi bold (600) |
| Error Text Color | #D9222A |
| Error Text Size | 1.2 rem |
| Font Family | Proxima Nova, sans-serif |
| Font Size | 1.4 rem |
| Font Weight | regular |
| Icon Color | #6A7078 |
| Input Active Border Color | #242C38 |
| Input Active, Error, Focus Border Width | 2px |
| Input Error Border Color | #D9222A |
| Input Field Height | 48px |
| Label Font Color | #505660 |
| Letter Spacing | 0 |
| Line Height | 1.5 |
| Max Width | 100% of the column |
| Min Width | 136px |
| Placeholder Font Size | 1.6 rem |
| Placeholder Text Color | #969DA4 |

2.1 NPM details

Following dev dependencies are used:

"@orxe-devkit/axe": "^1.0.1",

"@orxe-devkit/jest-config": "^1.0.2",

"@orxe-devkit/lint": "1.0.0",

"@orxe-devkit/webpack-config": "1.0.2"

"@orxe-components/icon": "^1.0.14",

"@orxe-components/icons": "^1.0.9",

Following are the steps performed to build and execute input component:

1. npm install -g @orxe3/cli
2. orxe new component-workspace my-first-project
3. cd my-first-project
4. orxe generate c input
5. orxe serve

Following are the steps to reuse the orxe-input component in another component is as follow:

1. npm install
2. npm install @my-first-project/input@0.0.3
3. Add the orxe input as specified in technical specification point.
4. orxe serve

2.2 NPM repo

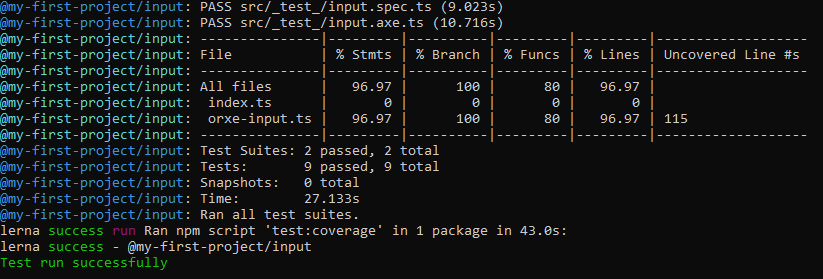
Following is the NPM repo link:

<https://packages.common.cnxloyalty.com/service/rest/repository/browse/npm-stage-hosted/>

3.Unit Testing

Test case are as follow:

1. It should exist
2. check if class of input is active
3. check if class of input is filled
4. check if class of input is error
5. check if class of input is focus
6. should support all WCAG Accessibility Rules. when default toolbar is rendered
7. test case for disabled input
8. test case for required input



4. Assumptions

Input component is developed using the technical specifications given by the Tavisca on

https://tavisca.atlassian.net/wiki/spaces/ORXE3/pages/773164902/Input+Fields+0.3.

5. Concerns and Issues   
 Close icon “Ic-close” was used in Active input field state. But as the nexus repository of @orxe-components/icon and @orxe-components/icons is different from this component repository so the code is commented for the icon and icon will be rendered.