```
/**
 * @file WestSecaucus.jsx
 * @author Joey Damico
 * @date September 25, 2019
 * @summary React JSX Component Class that is for West Secaucus
Interlocking
 *
 * Extends the React Component Class and is the UI part of the West
Secaucus Interlocking,
 * this class controls all the drawings of routes, and also gives a
visual reprenstation
 * of that status of the interlocking
 */

// Import React Component
import React, { Component } from 'react';
// Import CSS style sheet
import '../../../css/Main_Line/west_secaucus.css';

// Import Images
// Switch Images
import SW_D_E from '../../../../public/images/SW_D_E.png';
import SW_D_E_Lined from '../../../../public/images/SW_D_E_Lined.png';
import SW_D_E_Occupied from '../../../../public/images/
SW_D_E_Occupied.png'
import SW_D_E_R from '../../../../public/images/SW_D_E_R.png';
import SW_D_E_R_Lined from '../../../../public/images/
SW_D_E_R_Lined.png';
import SW_D_E_R_Occupied from '../../../../public/images/
SW_D_E_R_Occupied.png';

import SW_D_W from '../../../../public/images/SW_D_W.png';
import SW_D_W_Lined from '../../../../public/images/SW_D_W_Lined.png';
import SW_D_W_Occupied from '../../../../public/images/
SW_D_W_Occupied.png'
import SW_D_W_R from '../../../../public/images/SW_D_W_R.png';
import SW_D_W_R_Lined from '../../../../public/images/
SW_D_W_R_Lined.png';
import SW_D_W_R_Occupied from '../../../../public/images/
SW_D_W_R_Occupied.png';

// Signal Images
import SIG_W from '../../../../public/images/SIG_W.png';
import SIG_W_Clear from '../../../../public/images/SIG_W_Clear.png';
import SIG_W_Stop from '../../../../public/images/SIG_W_Stop.png';
import SIG_E from '../../../../public/images/SIG_E.png';
import SIG_E_Clear from '../../../../public/images/SIG_E_Clear.png';
import SIG_E_Stop from '../../../../public/images/SIG_E_Stop.png';

// Color Constants For Drawing Routes
```

```
const Empty = '#999999';
const Lined = '#75fa4c';
const Occupied = '#eb3323';


/**
 * The React JSX Component Class for the West Secaucus Interlocking
 *
 * This class is a JSX React Component for the West Secaucus
Interlocking, this will control all the UI for the comonent,
 * and the click events that will pass reference between the backend
and the user. This also controls drawing the
 * route drawings to show if a route(s) is setup in the interlocking
or if the route is occupied
 */
class WestSecaucus extends Component {
    /**
     * State
     * @summary Object that holds the state or status information for
the component
     *
     * This object holds all the information for the interlocking that
is required to display the routes
     * correctly
     *
     * Anything that has "this.props." is passed down from the CTC
interlocking class
     */
    state = {
        // Switch Status
        sw_1: this.props.status.sw_1,
        sw_3: this.props.status.sw_3,
        // Image File for the switch - Will change depending on route
        sw_1_src: SW_D_W,
        sw_3_src: SW_D_E,
        // Image File for the signals - Will change depending on route
        sig_2w_src: SIG_W,
        sig_4w_src: SIG_W,
        sig_2e_src: SIG_E,
        sig_4e_src: SIG_E,
        // Colors for tail tracks - Will change depending on route
        tail_1_e: Empty,
        tail_1_w: Empty,
        tail_2_e: Empty,
        tail_2_w: Empty,
        // Information For Interlocking Routes
        routes: this.props.status.routes,
        occupied: this.props.status.occupied
    };
    /**
     * componentWillReceiveProps()
     * @summary Function that updates the state of the component
     *
     * The data that is being changed is passed down from the CTC
classes in the simulation backend
     *
     * @param nextProps, the new data to set the component state too
     */
    componentWillReceiveProps(nextProps) {
        this.setState({
            sw_1: nextProps.status.sw_1,
            sw_3: nextProps.status.sw_3,
            sig_2w_src: SIG_W,
            sig_4w_src: SIG_W,
            sig_2e_src: SIG_E,
            sig_4e_src: SIG_E,
            tail_1_e: Empty,
            tail_1_w: Empty,
            tail_center: Empty,
            tail_2_e: Empty,
            tail_2_w: Empty,
            routes: nextProps.status.routes,
            occupied: nextProps.status.occupied
        });
    }
    // ---- END componentWillReceiveProps() ----

    /**
     * render()
     * @summary standard React function that draws the interlocking to
the screen
     */
    render() {
        // Set the switch images based off the state of each crossover
        this.set_switch_img();
        // Draw all the current routes in the interlocking
        this.set_route_drawing();

        // Returns the HTML to draw the interlocking and it's current
state to the screen
        return (
            <div>
                {/* Tags */}
                <div className="westSecaucus_title">WEST SECAUCUS</
div>
                <div className="westSecaucus_milepost">MP 5.0</div>
                {/* East Side Tail Tracks */}
                <div className="m_westSecaucus_1_east"
style={{background: this.state.tail_1_e}}></div>
```

```
                <div className="m_westSecaucus_2_east"
style={{background: this.state.tail_2_e}}></div>
                {/* Switches */}
                <div className="westSecaucus_SW_1"
onClick={this.props.throw_sw_3}><img src={this.state.sw_3_src}/></div>
                <div className="m_westSecaucus_bridge"
style={{background: this.state.tail_center}}></div>
                <div className="westSecaucus_SW_3"
onClick={this.props.throw_sw_1}><img src={this.state.sw_1_src}/></div>
                {/* West Side Tail Tracks */}
                <div className="m_westSecaucus_1_west"
style={{background: this.state.tail_1_w}}></div>
                <div className="m_westSecaucus_2_west"
style={{background: this.state.tail_2_w}}></div>
                {/* Signals */}
                <div className="westSecaucus_sig_2e"
onClick={this.props.click_sig_2e}><img src={this.state.sig_2e_src}/></
div>
                <div className="westSecaucus_sig_4e"
onClick={this.props.click_sig_4e}><img src={this.state.sig_4e_src}/></
div>
                <div className="westSecaucus_sig_2w"
onClick={this.props.click_sig_2w}><img src={this.state.sig_2w_src}/></
div>
                <div className="westSecaucus_sig_4w"
onClick={this.props.click_sig_4w}><img src={this.state.sig_4w_src}/></
div>
            </div>
        );
    }
    // ---- END render() ----

    /**
     * @summary Sets the drawing for the route through the
interlocking
     *
     * Function takes what routes are currently set in the
Interlocking class and displays that route in the UI, the drawing
     * will change depending on if the interlocking is occupied or not
     */
    set_route_drawing = () => {
        // Loop through all the Routes
        for (let i = 0; i < this.state.routes.length; i++) {
            if (this.state.routes[i] === "W_1_1__|
__1_mill_westSecaucus" || this.state.routes[i] === "E_1_1__|
__2_westSecaucus_laurel") {
                // The Route Is Occupied
                if (this.state.occupied) {
                    // Set Tail Tracks Color
                    this.state.tail_1_e = Occupied;
                    this.state.tail_center = Occupied;
                    this.state.tail_1_w = Occupied;

                    // Switches
                    this.state.sw_1_src = SW_D_W_Occupied;
                    this.state.sw_3_src = SW_D_E_Occupied;

                    // Signals
                    this.state.sig_2w_src = SIG_W_Stop;
                    this.state.sig_2e_src = SIG_E_Stop;
                    this.state.sig_4w_src = SIG_W_Stop;
                    this.state.sig_4e_src = SIG_E_Stop;
                }
                // The Route Is NOT Occupied
                else {
                    // Set Tail Track Colors
                    this.state.tail_1_e = Lined;
                    this.state.tail_center = Lined;
                    this.state.tail_1_w = Lined;

                    // Switches
                    this.state.sw_1_src = SW_D_W_Lined;
                    this.state.sw_3_src = SW_D_E_Lined;

                    // Signals
                    // West Bound Signals
                    if (this.state.routes[i] === "W_1_1__|
__1_mill_westSecaucus") {
                        this.state.sig_2w_src = SIG_W_Clear;
                        this.state.sig_2e_src = SIG_E_Stop;
                        this.state.sig_4w_src = SIG_W_Stop;
                        this.state.sig_4e_src = SIG_E_Stop;
                    }
                    // East Bound Signals
                    else {
                        this.state.sig_2w_src = SIG_W_Stop;
                        this.state.sig_2e_src = SIG_E_Clear;
                        this.state.sig_4w_src = SIG_W_Stop;
                        this.state.sig_4e_src = SIG_E_Stop;
                    }
                }
            }
            else if (this.state.routes[i] === "W_1_2__|
__2_mill_westSecaucus" || this.state.routes[i] === "E_2_1__|
__2_westSecaucus_laurel") {
                // The Route Is Occupied
                if (this.state.occupied) {
                    // Set Tail Track Colors
                    this.state.tail_1_e = Occupied;
                    this.state.tail_center = Occupied;
                    this.state.tail_2_w = Occupied;

                    // Switches
                    this.state.sw_1_src = SW_D_W_R_Occupied;
                    this.state.sw_3_src = SW_D_E_Occupied;

                    // Signals
                    this.state.sig_2w_src = SIG_W_Stop;
                    this.state.sig_2e_src = SIG_E_Stop;
                    this.state.sig_4w_src = SIG_W_Stop;
                    this.state.sig_4e_src = SIG_E_Stop;
                }
                else {
                    // Set Tail Track Colors
                    this.state.tail_1_e = Lined;
                    this.state.tail_center = Lined;
                    this.state.tail_2_w = Lined;

                    // Switches
                    this.state.sw_1_src = SW_D_W_R_Lined;
                    this.state.sw_3_src = SW_D_E_Lined;

                    // Signals
                    // West Bound Signals
                    if (this.state.routes[i] === "W_1_2__|
__2_mill_westSecaucus") {
                        this.state.sig_2w_src = SIG_W_Clear;
                        this.state.sig_2e_src = SIG_E_Stop;
                        this.state.sig_4w_src = SIG_W_Stop;
                        this.state.sig_4e_src = SIG_E_Stop;
                    }
                    // East Bound Signals
                    else {
                        this.state.sig_2w_src = SIG_W_Stop;
                        this.state.sig_2e_src = SIG_E_Stop;
                        this.state.sig_4w_src = SIG_W_Stop;
                        this.state.sig_4e_src = SIG_E_Clear;
                    }
                }
            }
            else if (this.state.routes[i] === "W_2_1__|
__1_mill_westSecaucus" || this.state.routes[i] === "E_1_2__|
__4_westSecaucus_laurel") {
                // The Route Is Occupied
                if (this.state.occupied) {
                    // Set Tail Track Colors
                    this.state.tail_2_e = Occupied;
                    this.state.tail_center = Occupied;
                    this.state.tail_1_w = Occupied;
                    // Switches
                    this.state.sw_1_src = SW_D_W_Occupied;
                    this.state.sw_3_src = SW_D_E_R_Occupied;

                    // Signals
                    this.state.sig_2w_src = SIG_W_Stop;
                    this.state.sig_2e_src = SIG_E_Stop;
                    this.state.sig_4w_src = SIG_W_Stop;
                    this.state.sig_4e_src = SIG_E_Stop;
                }
                // The Route Is NOT Occupied
                else {
                    // Set Tail Track Colors
                    this.state.tail_2_e = Lined;
                    this.state.tail_center = Lined;
                    this.state.tail_1_w = Lined;

                    // Switches
                    this.state.sw_1_src = SW_D_W_Lined;
                    this.state.sw_3_src = SW_D_E_R_Lined;

                    // Signals
                    // West Bound Signals
                    if (this.state.routes[i] === "W_2_1__|
__1_mill_westSecaucus") {
                        this.state.sig_2w_src = SIG_W_Stop;
                        this.state.sig_2e_src = SIG_E_Stop;
                        this.state.sig_4w_src = SIG_W_Clear;
                        this.state.sig_4e_src = SIG_E_Stop;
                    }
                    // East Bound Signals
                    else {
                        this.state.sig_2w_src = SIG_W_Stop;
                        this.state.sig_2e_src = SIG_E_Clear;
                        this.state.sig_4w_src = SIG_W_Stop;
                        this.state.sig_4e_src = SIG_E_Stop;
                    }
                }
            }
            else if (this.state.routes[i] === "W_2_2__|
__2_mill_westSecaucus" || this.state.routes[i] === "E_2_2__|
__4_westSecaucus_laurel") {
                // The Route Is Occupied
                if (this.state.occupied) {
                    // Set Tail Track Colors
                    this.state.tail_2_e = Occupied;
                    this.state.tail_center = Occupied;
                    this.state.tail_2_w = Occupied;

                    // Switches
                    this.state.sw_1_src = SW_D_W_R_Occupied;
                    this.state.sw_3_src = SW_D_E_R_Occupied;

                    // Signals
                    this.state.sig_2w_src = SIG_W_Stop;
                    this.state.sig_2e_src = SIG_E_Stop;
                    this.state.sig_4w_src = SIG_W_Stop;
                    this.state.sig_4e_src = SIG_E_Stop;
                }
                // The Route Is NOT Occupied
                else {
                    // Set Tail Track Colors
                    this.state.tail_2_e = Lined;
                    this.state.tail_center = Lined;
                    this.state.tail_2_w = Lined;

                    // Switches
                    this.state.sw_1_src = SW_D_W_R_Lined;
                    this.state.sw_3_src = SW_D_E_R_Lined;

                    // Signals
                    // West Bound Signals
                    if (this.state.routes[i] === "W_2_2__|
__2_mill_westSecaucus") {
                        this.state.sig_2w_src = SIG_W_Stop;
                        this.state.sig_2e_src = SIG_E_Stop;
                        this.state.sig_4w_src = SIG_W_Clear;
                        this.state.sig_4e_src = SIG_E_Stop;
                    }
                    // East Bound Signals
                    else {
                        this.state.sig_2w_src = SIG_W_Stop;
                        this.state.sig_2e_src = SIG_E_Stop;
                        this.state.sig_4w_src = SIG_W_Stop;
                        this.state.sig_4e_src = SIG_E_Clear;
                    }
                }
            }
        }
    }
    // ---- END set_route_drawings() ----

    /**
     * set_switch_img()
     * @summary Changes image sources for the switches, depending on
switch status
     *
     * This function uses the data passed in through status from the
CTC classes and
     * shows if the switches are reversed or not on the screen, by
```

```
changing the image
     * source files, to the correct .png file respectivly
     */
    set_switch_img = () => {
        if (this.state.sw_1) {
            this.state.sw_1_src = SW_D_W_R;
        }
        else {
            this.state.sw_1_src = SW_D_W;
        }

        if (this.state.sw_3) {
            this.state.sw_3_src = SW_D_E_R;
        }
        else {
            this.state.sw_3_src = SW_D_E;
        }
    }
    // ---- END set_switch_image() ----
}

// Export the interlocking to be drawn on the screen
export default WestSecaucus;
```