```
/** @jsx jsx */
import { useLayoutEffect, useRef } from "react";
import { css, jsx, keyframes, Global } from "@emotion/core";
import { Link } from "@reach/router";
import { FontAwesomeIcon } from "@fortawesome/react-fontawesome";
import "animate.css/animate.css";

const Notification = ({ unread }) => {
    const iconElm = useRef(null);
    const firstUpdate = useRef(true);

    useLayoutEffect(() => {
        if (firstUpdate.current) {
            firstUpdate.current = false;
            return;
        }
        iconElm.current.firstChild.classList.add("bellshake");
        setTimeout(() => {
            iconElm.current.firstChild.classList.remove("bellshake");
        }, 830);
    });

    const styleLink = css`
        color: black;
        text-decoration: none;
        position: relative;
    `;
    const styleIcon = css`
        font-size: 1.5em;
    `;
    const styleBadge = css`
        background-color: red;
        color: white;
        font-size: 0.625em;
        padding: 0.05em 0.35em;
        border-radius: 1em;
        border: 0.0625em solid white;
        display: inline-block;
        position: absolute;
        top: -1em;
        right: -0.625em;
    `;
    const shake = keyframes`
        10%, 90% { transform: translate(-1px); }
        20%, 80% { transform: translate(2px); }
        30%, 50%, 70% { transform: translate(-4px); }
        40%, 60% { transform: translate(4px); }
    `;
    const bellshake = css`
        .bellshake {
            animation: ${shake} 0.82s cubic-bezier(0.36, 0.07, 0.19, 0.97)
both;
            transform: translate(0);
        }
    `;

    return (
        <Link ref={iconElm} /*onChange={shakeOnChange}*/ css={styleLink}
to="/">
            <Global styles={bellshake} />
            <FontAwesomeIcon css={styleIcon} icon={["far", "bell"]} />
            {unread && unread !== "0" ? (
                <span css={styleBadge}>{unread}</span>
            ) : null}
        </Link>
    );
};

export default Notification;

```