The main project:

Index.html: <!DOCTYPE html>

<html lang="en">

    <head>

        <meta charset="UTF-8">

        <meta name="viewport" content="width=device-width, initial-scale=1.0">

        <title>Volume Modulator</title>

        <link rel="stylesheet" href="style.css">

        <script defer src="theme.js"></script>

        <script src="libs/plotly.min.js"></script>

        <script src="https://www.youtube.com/iframe\_api"></script>

    </head>

<body>

    <nav class="navbar">

        <ul class="navbar-nav">

            <li class="logo">

                <a href="#" class="nav-link">

                    <span class="link-text logo-text">Modulator</span>

                    <svg

                        aria-hidden="true"

                        focusable="false"

                        data-prefix="fad"

                        data-icon="angle-double-right"

                        role="img"

                        xmlns="http://www.w3.org/2000/svg"

                        viewBox="0 0 448 512"

                        class="svg-inline--fa fa-angle-double-right fa-w-14 fa-5x"

                    >

                        <g class="fa-group">

                            <path

                                fill="currentColor"

                                d="M224 273L88.37 409a23.78 23.78 0 0 1-33.8 0L32 386.36a23.94 23.94 0 0 1 0-33.89l96.13-96.37L32 159.73a23.94 23.94 0 0 1 0-33.89l22.44-22.79a23.78 23.78 0 0 1 33.8 0L223.88 239a23.94 23.94 0 0 1 .1 34z"

                                class="fa-secondary"

                            ></path>

                            <path

                                fill="currentColor"

                                d="M415.89 273L280.34 409a23.77 23.77 0 0 1-33.79 0L224 386.26a23.94 23.94 0 0 1 0-33.89L320.11 256l-96-96.47a23.94 23.94 0 0 1 0-33.89l22.52-22.59a23.77 23.77 0 0 1 33.79 0L416 239a24 24 0 0 1-.11 34z"

                                class="fa-primary"

                            ></path>

                        </g>

                    </svg>

                </a>

            </li>

            <li class="nav-item">

                <a href="#" class="nav-link" onclick="toggleSubmenu('graphSubmenu')">

                    <svg aria-hidden="true" focusable="false" data-prefix="fad" data-icon="graph\_select" role="img" style="enable-background:new 0 0 32 32;" viewBox="0 0 32 32" xml:space="preserve" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" class="svg-inline--fa fa-upload fa-w-16 fa-7x">

                        <style type="text/css">

                            .st0{clip-path:url(#SVGID\_2\_);}

                            .st1{clip-path:url(#SVGID\_4\_);}

                        </style>

                        <g>

                            <g>

                                <path fill="currentColor" class="fa-primary" d="M16,28c-0.5527344,0-1-0.4472656-1-1V5c0-0.5522461,0.4472656-1,1-1s1,0.4477539,1,1v22C17,27.5527344,16.5527344,28,16,28z"/>

                            </g>

                            <g>

                                <path fill="currentColor" class="fa-secondary" d="M11.6669922,26c-0.5527344,0-1-0.4472656-1-1V7c0-0.5522461,0.4472656-1,1-1s1,0.4477539,1,1v18C12.6669922,25.5527344,12.2197266,26,11.6669922,26z"/>

                            </g>

                            <g>

                                <g>

                                    <defs>

                                        <line id="SVGID\_1\_" x1="7.3333335" x2="7.3333335" y1="9" y2="23"/>

                                    </defs>

                                    <use style="overflow:visible;" xlink:href="#SVGID\_1\_"/>

                                    <clipPath id="SVGID\_2\_">

                                        <use style="overflow:visible;" xlink:href="#SVGID\_1\_"/>

                                    </clipPath>

                                </g>

                                <path fill="currentColor" class="fa-primary" d="M7.3330078,24c-0.5527344,0-1-0.4472656-1-1V9c0-0.5522461,0.4472656-1,1-1s1,0.4477539,1,1v14C8.3330078,23.5527344,7.8857422,24,7.3330078,24z"/>

                            </g>

                            <g>

                                <path fill="currentColor" class="fa-secondary" d="M3,21c-0.5527344,0-1-0.4472656-1-1v-8c0-0.5522461,0.4472656-1,1-1s1,0.4477539,1,1v8C4,20.5527344,3.5527344,21,3,21z"/>

                            </g>

                            <g>

                                <path fill="currentColor" class="fa-secondary" d="M29,21c-0.5527344,0-1-0.4472656-1-1v-8c0-0.5522461,0.4472656-1,1-1s1,0.4477539,1,1v8C30,20.5527344,29.5527344,21,29,21z"/>

                            </g>

                            <g>

                                <g>

                                    <defs>

                                        <line id="SVGID\_3\_" x1="24.666666" x2="24.666666" y1="9" y2="23"/>

                                    </defs>

                                    <use style="overflow:visible;" xlink:href="#SVGID\_3\_"/>

                                    <clipPath id="SVGID\_4\_">

                                        <use style="overflow:visible;" xlink:href="#SVGID\_3\_"/>

                                    </clipPath>

                                </g>

                                <path fill="currentColor" class="fa-primary" d="M24.6669922,24c-0.5527344,0-1-0.4472656-1-1V9c0-0.5522461,0.4472656-1,1-1s1,0.4477539,1,1v14C25.6669922,23.5527344,25.2197266,24,24.6669922,24z"/>

                            </g>

                            <g>

                                <path fill="currentColor" class="fa-secondary" d="M20.3330078,26c-0.5527344,0-1-0.4472656-1-1V7c0-0.5522461,0.4472656-1,1-1s1,0.4477539,1,1v18C21.3330078,25.5527344,20.8857422,26,20.3330078,26z"/>

                            </g>

                        </g>

                    </svg>

                    <span class="link-text">Select Graph</span>

                </a>

                <div id="graphSubmenu" class="submenu">

                    <!-- Thumbnails for graphs will be dynamically populated -->

                </div>

            </li>

            <li class="nav-item">

                <a href="#" class="nav-link" onclick="toggleSubmenu('uploadSubmenu')">

                    <svg aria-hidden="true" focusable="false" data-prefix="fad" data-icon="upload" role="img" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 29 29" style="fill-rule:evenodd;clip-rule:evenodd;stroke-linejoin:round;stroke-miterlimit:2;" class="svg-inline--fa fa-upload fa-w-16 fa-7x"><path fill="currentColor" d="M27.964,23.272c-0.116,0.424 -0.504,0.735 -0.964,0.735l-0.996,-0l-0,0.99c-0,0.552 -0.448,1 -1,1c-0.552,-0 -1,-0.448 -1,-1l-0,-0.99l-1.004,-0c-0.552,-0 -1,-0.448 -1,-1c0,-0.552 0.448,-1 1,-1l1.004,-0l-0,-1.007c-0,-0.552 0.448,-1 1,-1c0.552,0 1,0.448 1,1l-0,1.007l0.996,-0c0.462,-0 0.851,0.314 0.966,0.74l0.043,-15.552c-0,-1.769 -1.435,-3.203 -3.204,-3.203l-17.614,0c-1.77,0 -3.204,1.434 -3.204,3.203l0,17.603c0,1.768 1.434,3.202 3.204,3.202l17.775,-0.037c1.651,-0.003 2.989,-1.34 2.994,-2.991l0.004,-1.7Zm-15.967,-5.279l-0.002,-0c-1.104,-0 -2.001,0.897 -2.001,2.001c-0,1.105 0.897,2.002 2.001,2.002c1.105,0 2.002,-0.897 2.002,-2.002l0,-7.122c0,-0 6.001,-0.75 6.001,-0.75l0.003,3.867c-1.105,0 -2.002,0.897 -2.002,2.002c-0,1.105 0.897,2.002 2.002,2.002c1.104,-0 2.001,-0.897 2.001,-2.002l-0.005,-7.003c-0,-0.286 -0.123,-0.559 -0.338,-0.749c-0.215,-0.189 -0.501,-0.278 -0.786,-0.242l-8,1c-0.5,0.062 -0.876,0.488 -0.876,0.992l0,6.004Z" class="fa-primary"></path></svg>

                    <span class="link-text">Upload Music</span>

                </a>

                <div id="uploadSubmenu" class="submenu">

                    <input type="file" id="fileInput" accept="audio/mp3" multiple>

                </div>

            </li>

            <li class="nav-item">

                <a href="#" class="nav-link" onclick="connectToSpotify()">

                    <svg aria-hidden="true" focusable="false" data-prefix="fab" data-icon="spotify" role="img" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 496 512" class="svg-inline--fa fa-spotify fa-w-16 fa-5x">

                        <path fill="#00DA5A" d="M248 8C111 8 0 119 0 256s111 248 248 248 248-111 248-248S385 8 248 8zm113.5 356.8c-4.1 5.9-11.5 7.8-17.6 3.7-48.7-32.1-110-39.4-183-23.1-6.8 1.5-13.3-2.7-14.8-9.4-1.5-6.7 2.7-13.3 9.4-14.8 80.3-17.7 148.5-9.1 203.7 26.1 5.9 3.9 7.8 11.5 3.7 17.5zm23.6-56.2c-5 6.9-14.5 8.8-21.5 3.8-55.8-36.7-141-47.3-207.6-27.8-8 2.3-16.4-2.3-18.7-10.3-2.3-8 2.3-16.4 10.3-18.7 76.3-21.5 170.5-9.5 232.7 31.5 6.9 5 8.7 14.5 3.8 21.5zm-5.6-63.5C313 201.1 181.3 197.3 103.5 222c-9.3 3-19.1-2.2-22.1-11.5-3-9.3 2.2-19.1 11.5-22.1 87.6-28.7 238.1-24.8 328.7 35.3 8.1 5.3 10.4 16.3 5.1 24.4-5.3 8.2-16.3 10.4-24.4 5.1z"></path>

                      </svg>

                    <span class="link-text">Connect to Spotify</span>

                </a>

            </li>

            <li class="nav-item">

                <a href="#" class="nav-link" onclick="toggleSubmenu('youtubeSubmenu')">

                    <svg aria-hidden="true" focusable="false" width="80px" height="80px" viewBox="2.5 0 95 95" xmlns="http://www.w3.org/2000/svg" class="svg-inline--fa fa-youtube fa-w-16 fa-5x">

                        <style type="text/css">

                            .st0{fill:#FFFFFF;}

                            .st1{fill:#F5BB41;}

                            .st2{fill:#2167D1;}

                            .st3{fill:#3D84F3;}

                            .st4{fill:#4CA853;}

                            .st5{fill:#398039;}

                            .st6{fill:#D74F3F;}

                            .st7{fill:#D43C89;}

                            .st8{fill:#B2005F;}

                            .st9{fill:none;stroke:#000000;stroke-width:3;stroke-linecap:round;stroke-linejoin:round;stroke-miterlimit:10;}

                            .st10{fill-rule:evenodd;clip-rule:evenodd;fill:none;stroke:#000000;stroke-width:3;stroke-linecap:round;stroke-linejoin:round;stroke-miterlimit:10;}

                            .st11{fill-rule:evenodd;clip-rule:evenodd;fill:none;stroke:#040404;stroke-width:3;stroke-linecap:round;stroke-linejoin:round;stroke-miterlimit:10;}

                            .st12{fill-rule:evenodd;clip-rule:evenodd;}

                            .st13{fill-rule:evenodd;clip-rule:evenodd;fill:#040404;}

                            .st14{fill:url(#SVGID\_1\_);}

                            .st15{fill:url(#SVGID\_2\_);}

                            .st16{fill:url(#SVGID\_3\_);}

                            .st17{fill:url(#SVGID\_4\_);}

                            .st18{fill:url(#SVGID\_5\_);}

                            .st19{fill:url(#SVGID\_6\_);}

                            .st20{fill:url(#SVGID\_7\_);}

                            .st21{fill:url(#SVGID\_8\_);}

                            .st22{fill:url(#SVGID\_9\_);}

                            .st23{fill:url(#SVGID\_10\_);}

                            .st24{fill:url(#SVGID\_11\_);}

                            .st25{fill:url(#SVGID\_12\_);}

                            .st26{fill:url(#SVGID\_13\_);}

                            .st27{fill:url(#SVGID\_14\_);}

                            .st28{fill:url(#SVGID\_15\_);}

                            .st29{fill:url(#SVGID\_16\_);}

                            .st30{fill:url(#SVGID\_17\_);}

                            .st31{fill:url(#SVGID\_18\_);}

                            .st32{fill:url(#SVGID\_19\_);}

                            .st33{fill:url(#SVGID\_20\_);}

                            .st34{fill:url(#SVGID\_21\_);}

                            .st35{fill:url(#SVGID\_22\_);}

                            .st36{fill:url(#SVGID\_23\_);}

                            .st37{fill:url(#SVGID\_24\_);}

                            .st38{fill:url(#SVGID\_25\_);}

                            .st39{fill:url(#SVGID\_26\_);}

                            .st40{fill:url(#SVGID\_27\_);}

                            .st41{fill:url(#SVGID\_28\_);}

                            .st42{fill:url(#SVGID\_29\_);}

                            .st43{fill:url(#SVGID\_30\_);}

                            .st44{fill:url(#SVGID\_31\_);}

                            .st45{fill:url(#SVGID\_32\_);}

                            .st46{fill:url(#SVGID\_33\_);}

                            .st47{fill:url(#SVGID\_34\_);}

                            .st48{fill:url(#SVGID\_35\_);}

                            .st49{fill:url(#SVGID\_36\_);}

                            .st50{fill:url(#SVGID\_37\_);}

                            .st51{fill:url(#SVGID\_38\_);}

                            .st52{fill:url(#SVGID\_39\_);}

                            .st53{fill:url(#SVGID\_40\_);}

                            .st54{fill:url(#SVGID\_41\_);}

                            .st55{fill:url(#SVGID\_42\_);}

                            .st56{fill:url(#SVGID\_43\_);}

                            .st57{fill:url(#SVGID\_44\_);}

                            .st58{fill:url(#SVGID\_45\_);}

                            .st59{fill:#040404;}

                            .st60{fill:url(#SVGID\_46\_);}

                            .st61{fill:url(#SVGID\_47\_);}

                            .st62{fill:url(#SVGID\_48\_);}

                            .st63{fill:url(#SVGID\_49\_);}

                            .st64{fill:url(#SVGID\_50\_);}

                            .st65{fill:url(#SVGID\_51\_);}

                            .st66{fill:url(#SVGID\_52\_);}

                            .st67{fill:url(#SVGID\_53\_);}

                            .st68{fill:url(#SVGID\_54\_);}

                            .st69{fill:url(#SVGID\_55\_);}

                            .st70{fill:url(#SVGID\_56\_);}

                            .st71{fill:url(#SVGID\_57\_);}

                            .st72{fill:url(#SVGID\_58\_);}

                            .st73{fill:url(#SVGID\_59\_);}

                            .st74{fill:url(#SVGID\_60\_);}

                            .st75{fill:url(#SVGID\_61\_);}

                            .st76{fill:url(#SVGID\_62\_);}

                            .st77{fill:none;stroke:#000000;stroke-width:3;stroke-miterlimit:10;}

                            .st78{fill:none;stroke:#FFFFFF;stroke-miterlimit:10;}

                            .st79{fill:#4BC9FF;}

                            .st80{fill:#5500DD;}

                            .st81{fill:#FF3A00;}

                            .st82{fill:#E6162D;}

                            .st83{fill:#F1F1F1;}

                            .st84{fill:#FF9933;}

                            .st85{fill:#B92B27;}

                            .st86{fill:#00ACED;}

                            .st87{fill:#BD2125;}

                            .st88{fill:#1877F2;}

                            .st89{fill:#6665D2;}

                            .st90{fill:#CE3056;}

                            .st91{fill:#5BB381;}

                            .st92{fill:#61C3EC;}

                            .st93{fill:#E4B34B;}

                            .st94{fill:#181EF2;}

                            .st95{fill:#FF0000;}

                            .st96{fill:#FE466C;}

                            .st97{fill:#FA4778;}

                            .st98{fill:#FF7700;}

                            .st99{fill-rule:evenodd;clip-rule:evenodd;fill:#1F6BF6;}

                            .st100{fill:#520094;}

                            .st101{fill:#4477E8;}

                            .st102{fill:#3D1D1C;}

                            .st103{fill:#FFE812;}

                            .st104{fill:#344356;}

                            .st105{fill:#00CC76;}

                            .st106{fill-rule:evenodd;clip-rule:evenodd;fill:#345E90;}

                            .st107{fill:#1F65D8;}

                            .st108{fill:#EB3587;}

                            .st109{fill-rule:evenodd;clip-rule:evenodd;fill:#603A88;}

                            .st110{fill:#E3CE99;}

                            .st111{fill:#783AF9;}

                            .st112{fill:#FF515E;}

                            .st113{fill:#FF4906;}

                            .st114{fill:#503227;}

                            .st115{fill:#4C7BD9;}

                            .st116{fill:#69C9D0;}

                            .st117{fill:#1B92D1;}

                            .st118{fill:#EB4F4A;}

                            .st119{fill:#513728;}

                            .st120{fill:#FF6600;}

                            .st121{fill-rule:evenodd;clip-rule:evenodd;fill:#B61438;}

                            .st122{fill:#FFFC00;}

                            .st123{fill:#141414;}

                            .st124{fill:#94D137;}

                            .st125{fill-rule:evenodd;clip-rule:evenodd;fill:#F1F1F1;}

                            .st126{fill-rule:evenodd;clip-rule:evenodd;fill:#66E066;}

                            .st127{fill:#2D8CFF;}

                            .st128{fill:#F1A300;}

                            .st129{fill:#4BA2F2;}

                            .st130{fill:#1A5099;}

                            .st131{fill:#EE6060;}

                            .st132{fill-rule:evenodd;clip-rule:evenodd;fill:#F48120;}

                            .st133{fill:#222222;}

                            .st134{fill:url(#SVGID\_63\_);}

                            .st135{fill:#0077B5;}

                            .st136{fill:#FFCC00;}

                            .st137{fill:#EB3352;}

                            .st138{fill:#F9D265;}

                            .st139{fill:#F5B955;}

                            .st140{fill:#DD2A7B;}

                            .st141{fill:#66E066;}

                            .st142{fill:#EB4E00;}

                            .st143{fill:#FFC794;}

                            .st144{fill:#B5332A;}

                            .st145{fill:#4E85EB;}

                            .st146{fill:#58A45C;}

                            .st147{fill:#F2BC42;}

                            .st148{fill:#D85040;}

                            .st149{fill:#464EB8;}

                            .st150{fill:#7B83EB;}

                        </style>

                        <g id="Layer\_1"/>

                        <g id="Layer\_2">

                            <g>

                                <g>

                                    <g>

                                        <path class="st95" d="M50,2.5C23.766,2.5,2.5,23.823,2.5,50.126c2.502,63.175,92.507,63.157,95-0.001 C97.5,23.823,76.233,2.5,50,2.5z M50,77.399c-15.036,0-27.27-12.233-27.27-27.27c0.74-18.662,14.654-27.134,27.269-27.134 c0.001,0,0.001,0,0.002,0c12.616,0.001,26.531,8.473,27.267,27.073C77.27,65.167,65.036,77.399,50,77.399z"/>

                                        <path class="st95" d="M50.002,26.103c-15.946-0.001-23.704,12.486-24.165,24.088C25.838,63.453,36.677,74.292,50,74.292 S74.162,63.453,74.162,50.13C73.705,38.591,65.948,26.105,50.002,26.103z"/>

                                    </g>

                                    <path class="st83" d="M41.055,52.528c-0.001,2.575,0.001,7.867,0,10.46c0,0,21.802-13.417,21.802-13.417L41.055,37.272V52.528z"/>

                                </g>

                            </g>

                        </g>

                    </svg>

                    <span class="link-text">Play YouTube Playlist</span>

                </a>

                <div id="youtubeSubmenu" class="submenu">

                    <input type="text" id="youtubeLinkInput" placeholder = "Enter YouTube Music/Video/Playlist Link">

                    <button onclick ="addYoutubeLink()">Add Link</button>

            </li>

            <li class="nav-item" id="themeButton">

                <a href="#" class="nav-link">

                    <svg class="theme-icon" id="lightIcon" aria-hidden="true" focusable="false" data-prefix="fad" data-icon="moon-stars" role="img" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 512 512" class="svg-inline--fa fa-moon-stars fa-w-16 fa-7x"><path fill="currentColor" d="M320 32L304 0l-16 32-32 16 32 16 16 32 16-32 32-16zm138.7 149.3L432 128l-26.7 53.3L352 208l53.3 26.7L432 288l26.7-53.3L512 208z" class="fa-secondary"></path><path fill="currentColor" d="M332.2 426.4c8.1-1.6 13.9 8 8.6 14.5a191.18 191.18 0 0 1-149 71.1C85.8 512 0 426 0 320c0-120 108.7-210.6 227-188.8 8.2 1.6 10.1 12.6 2.8 16.7a150.3 150.3 0 0 0-76.1 130.8c0 94 85.4 165.4 178.5 147.7z" class="fa-primary"></path></svg>

                    <svg class="theme-icon" id="solarIcon" aria-hidden="true" focusable="false" data-prefix="fad" data-icon="sun" role="img" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 512 512" class="svg-inline--fa fa-sun fa-w-16 fa-7x"><path fill="currentColor" d="M502.42 240.5l-94.7-47.3 33.5-100.4c4.5-13.6-8.4-26.5-21.9-21.9l-100.4 33.5-47.41-94.8a17.31 17.31 0 0 0-31 0l-47.3 94.7L92.7 70.8c-13.6-4.5-26.5 8.4-21.9 21.9l33.5 100.4-94.7 47.4a17.31 17.31 0 0 0 0 31l94.7 47.3-33.5 100.5c-4.5 13.6 8.4 26.5 21.9 21.9l100.41-33.5 47.3 94.7a17.31 17.31 0 0 0 31 0l47.31-94.7 100.4 33.5c13.6 4.5 26.5-8.4 21.9-21.9l-33.5-100.4 94.7-47.3a17.33 17.33 0 0 0 .2-31.1zm-155.9 106c-49.91 49.9-131.11 49.9-181 0a128.13 128.13 0 0 1 0-181c49.9-49.9 131.1-49.9 181 0a128.13 128.13 0 0 1 0 181z" class="fa-secondary"></path><path fill="currentColor" d="M352 256a96 96 0 1 1-96-96 96.15 96.15 0 0 1 96 96z" class="fa-primary"></path></svg>

                    <svg class="theme-icon" id="darkIcon" aria-hidden="true" focusable="false" data-prefix="fad" data-icon="sunglasses" role="img" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 576 512" class="svg-inline--fa fa-sunglasses fa-w-18 fa-7x"><path fill="currentColor" d="M574.09 280.38L528.75 98.66a87.94 87.94 0 0 0-113.19-62.14l-15.25 5.08a16 16 0 0 0-10.12 20.25L395.25 77a16 16 0 0 0 20.22 10.13l13.19-4.39c10.87-3.63 23-3.57 33.15 1.73a39.59 39.59 0 0 1 20.38 25.81l38.47 153.83a276.7 276.7 0 0 0-81.22-12.47c-34.75 0-74 7-114.85 26.75h-73.18c-40.85-19.75-80.07-26.75-114.85-26.75a276.75 276.75 0 0 0-81.22 12.45l38.47-153.8a39.61 39.61 0 0 1 20.38-25.82c10.15-5.29 22.28-5.34 33.15-1.73l13.16 4.39A16 16 0 0 0 180.75 77l5.06-15.19a16 16 0 0 0-10.12-20.21l-15.25-5.08A87.95 87.95 0 0 0 47.25 98.65L1.91 280.38A75.35 75.35 0 0 0 0 295.86v70.25C0 429 51.59 480 115.19 480h37.12c60.28 0 110.38-45.94 114.88-105.37l2.93-38.63h35.76l2.93 38.63c4.5 59.43 54.6 105.37 114.88 105.37h37.12C524.41 480 576 429 576 366.13v-70.25a62.67 62.67 0 0 0-1.91-15.5zM203.38 369.8c-2 25.9-24.41 46.2-51.07 46.2h-37.12C87 416 64 393.63 64 366.11v-37.55a217.35 217.35 0 0 1 72.59-12.9 196.51 196.51 0 0 1 69.91 12.9zM512 366.13c0 27.5-23 49.87-51.19 49.87h-37.12c-26.69 0-49.1-20.3-51.07-46.2l-3.12-41.24a196.55 196.55 0 0 1 69.94-12.9A217.41 217.41 0 0 1 512 328.58z" class="fa-secondary"></path><path fill="currentColor" d="M64.19 367.9c0-.61-.19-1.18-.19-1.8 0 27.53 23 49.9 51.19 49.9h37.12c26.66 0 49.1-20.3 51.07-46.2l3.12-41.24c-14-5.29-28.31-8.38-42.78-10.42zm404-50l-95.83 47.91.3 4c2 25.9 24.38 46.2 51.07 46.2h37.12C489 416 512 393.63 512 366.13v-37.55a227.76 227.76 0 0 0-43.85-10.66z" class="fa-primary"></path></svg>

                    <span class="link-text">Theme</span>

                </a>

            </li>

        </ul>

    </nav>

    <main id="mainContent">

        <div id="plotArea" class="flex-container">

            <div id="plotContainer" class="flex-item plot">

                <div id="controlOverlay" class="flex-container">

                    <div class="controls">

                        <button id="loopBtn">Loop</button>

                    </div>

                </div>

                <div id="plot"></div>

                <div id="controlOverlay" class="flex-container">

                    <div class="controls">

                        <button id="playPauseBtn">Play</button>

                    </div>

                </div>

            </div>

        </div>

        <div id="graphDescriptionContainer" class="flex-item description">

            <div id="graphDescription"></div>

        </div>

        <div id="pageDescription" class="flex-item description">

            <h2>About This Website</h2>

            <p>This website is a volume modulator...</p>

            <a href="https://github.com/username/repo">GitHub Repository</a>

            <a href="license.md">License</a>

        </div>

        <div id="companyInfo" class="flex-item description">

            <p>Company Information and Logo Animation</p>

            <div id="logoAnimation"></div>

            <div id="version">Version: 0.8.1</div>

        </div>

        <audio id="audio" preload="auto"></audio>

        <div id="player" style="display: none;"></div>

    </main>

    <script src="script.js"></script>

</body>

</html>

.

Style.css: :root {

    font-size: 16px;

    font-family: 'Open Sans', sans-serif;

    --text-primary: #b6b6b6;

    --text-secondary: #ececec;

    --bg-primary: #23232e;

    --bg-secondary: #141418;

    --transition-speed: 600ms;

    --grey-light: #ececec;

    --grey-dark: #b6b6b6;

    --grey-darker: #232323;

}

body {

    color: var(--text-primary);

    background-color: var(--bg-secondary);

    margin: 0;

    padding: 0;

    overflow-x: hidden;

}

body::-webkit-scrollbar {

    width: 0.25rem;

}

body::-webkit-scrollbar-track {

    background: #1e1e24;

}

body::-webkit-scrollbar-thumb {

    background: #6649b8;

}

main {

    margin-left: 5rem;

    padding: 1rem;

    display: flex;

    flex-direction: column;

    justify-content: flex-start;

    align-items: center;

    min-height: 100vh;

    position: relative;

}

.navbar {

    position: fixed;

    background-color: var(--bg-primary);

    overflow: hidden;

}

.navbar-nav {

    list-style: none;

    padding: 0;

    margin: 0;

    display: flex;

    flex-direction: column;

    align-items: center;

    height: 100%;

}

.nav-item {

    width: 100%;

}

.nav-item:last-child {

    margin-top: auto;

}

.nav-link {

    display: flex;

    align-items: center;

    height: 5rem;

    color: var(--text-primary);

    text-decoration: none;

    filter: grayscale(100%) opacity(0.7);

    transition: var(--transition-speed);

}

.nav-link:hover {

    filter: grayscale(0%) opacity(1);

    background: var(--bg-secondary);

    color: var(--text-secondary);

}

.link-text {

    display: none;

    margin-left: 1rem;

}

.nav-link svg {

    width: 2rem;

    min-width: 2rem;

    margin: 0 1.5rem;

}

.nav-link .svg-inline--fa path {

    transition: fill var(--transition-speed);

}

.fa-primary {

    color: #ff7eee;

}

.fa-secondary {

    color: #df49a6;

}

.fa-primary,

.fa-secondary {

    transition: var(--transition-speed);

}

.logo {

    font-weight: bold;

    text-transform: uppercase;

    margin-bottom: 1rem;

    text-align: center;

    color: var(--text-secondary);

    background: var(--bg-secondary);

    font-size: 1.5rem;

    letter-spacing: 0.3ch;

    width: 100%;

    display: flex;

    justify-content: center;

    align-items: center;

    transition: var(--transition-speed);

}

.logo svg {

    transform: rotate(0deg);

    transition: var(--transition-speed);

}

.logo-text {

    display: inline;

    position: absolute;

    left: -999px;

    transition: var(--transition-speed);

}

.navbar.active {

    width: 16rem;

}

.navbar.active .link-text {

    display: inline;

}

.navbar.active .logo svg {

    margin-left: 15rem;

    transform: rotate(180deg);

}

.navbar.active .logo-text {

    left: 15px;

}

.submenu {

    display: none;

    flex-direction: column;

    padding: 10px;

    background-color: var(--bg-secondary);

    overflow-y: auto;

    max-height: 200px;

}

.submenu::-webkit-scrollbar {

    width: 0.5rem;

}

.submenu::-webkit-scrollbar-track {

    background: #1e1e24;

}

.submenu::-webkit-scrollbar-thumb {

    background: #6649b8;

}

.submenu input {

    margin-top: 10px;

}

.thumbnail {

    display: flex;

    flex-direction: column;

    align-items: center;

    margin-bottom: 10px;

}

.thumbnail img {

    width: 50px;

    height: 50px;

}

.thumbnail-label {

    margin-top: 5px;

    color: var(--text-primary);

}

#graphSubmenu {

    max-height: 300px;

    overflow-y: auto;

}

@media only screen and (max-width: 600px) {

    .navbar {

        bottom: 0;

        width: 100vw;

        height: 5rem;

    }

    .logo {

        display: none;

    }

    .navbar-nav {

        flex-direction: row;

    }

    .nav-link {

        justify-content: center;

    }

    main {

        margin: 0;

    }

}

@media only screen and (min-width: 600px) {

    .navbar {

        top: 0;

        width: 5rem;

        height: 100vh;

    }

    .navbar.active {

        width: 16rem;

    }

}

.solar {

    --text-primary: #b6b6b6;

    --text-secondary: #ececec;

    --bg-primary: #23232e;

    --bg-secondary: #141418;

}

.light {

    --text-primary: #1f1f1f;

    --text-secondary: #000000;

    --bg-primary: #ffffff;

    --bg-secondary: #e4e4e4;

}

.dark {

    --text-primary: #576e75;

    --text-secondary: #35535c;

    --bg-primary: #fdf6e3;

    --bg-secondary: #f5e5b8;

}

.theme-icon {

    display: none;

}

.dark #darkIcon {

    display: block;

}

.light #lightIcon {

    display: block;

}

.solar #solarIcon {

    display: block;

}

.flex-container {

    display: flex;

    justify-content: center;

    align-items: center;

    margin-top: 10px;

    width: 100%;

}

.flex-item {

    margin: 10px;

    padding: 10px;

    background-color: var(--bg-secondary);

    border-radius: 5px;

    width: 90%;

    max-width: 800px;

}

.plot {

    display: flex;

    justify-content: center;

    align-items: center;

    position: relative;

    width: 100%;

    height: 60vmin;

}

#plot \* {

    cursor: default;

}

.controls {

    display: flex;

    flex-direction: column;

    gap: 10px;

    pointer-events: auto;

    transition: transform 0.6s;

}

button {

    padding: 10px;

    background-color: #444;

    color: white;

    border: none;

    cursor: pointer;

    font-size: 1em;

    border-radius: 5px;

    transition: background-color 0.3s, transform 0.3s;

}

button:hover {

    background-color: #555;

    transform: scale(1.05);

}

#pageDescription {

    color: white;

    font-size: 1em;

    text-align: center;

}

#version {

    position: absolute;

    bottom: 10px;

    right: 10px;

    color: rgb(102, 102, 102);

    font-size: 0.8em;

    opacity: 0.8;

    z-index: 1;

}

#player {

    display: none;

}

#controlOverlay {

    position: flex;

    top: 60vmin;

    width: 100%;

    display: flex;

    justify-content: center;

    pointer-events: none;

}

#graphDescriptionContainer {

    color: white;

    text-align: center;

}

#companyInfo {

    color: white;

    font-size: 1em;

    text-align: center;

}

#logoAnimation {

    margin-top: 10px;

    height: 100px;

    width: 100px;

    background-color: #444;

    border-radius: 50%;

    animation: spin 4s linear infinite;

}

@keyframes spin {

    0% { transform: rotate(0deg); }

    100% { transform: rotate(360deg); }

}

.

Script.js: document.addEventListener('DOMContentLoaded', function() {

    const plotElement = document.getElementById('plot');

    const audioElement = document.getElementById('audio');

    const playPauseBtn = document.getElementById('playPauseBtn');

    const loopBtn = document.getElementById('loopBtn');

    const fileInput = document.getElementById('fileInput');

    const graphDescriptionContainer = document.getElementById('graphDescriptionContainer');

    const graphDescriptionElement = document.getElementById('graphDescription');

    const pageDescription = document.getElementById('pageDescription');

    const navbar = document.querySelector('.navbar');

    const graphSubmenu = document.getElementById('graphSubmenu');

    const logoLink = document.querySelector('.logo a');

    let selectedGraph = 'heart';

    let isMuted = false;

    let isPlaying = false;

    let isLooping = true;

    let a = 10;

    let musicFiles = [];

    let loopInterval;

    const loopSpeed = 100;

    /\* YouTube shenanigans \*/

    let player;

    let youtubeLinks = [];

    let currentVideoIndex = 0;

    /\* Graph shenanigans \*/

    const graphConfigs = {

        heart: {

            script: 'graphs/heart.js',

            description: 'graphs/description-heart.txt',

            thumbnail: 'assets/thumbnails/heart.png',

            sampleMusic: 'audio/heart.mp3'

        },

        graph2: {

            script: 'graphs/graph2.js',

            description: 'graphs/description-graph2.txt',

            thumbnail: 'assets/thumbnails/graph2.png',

            sampleMusic: 'audio/sample.mp3'

        }

    };

    function loadGraphScript(scriptUrl, callback) {

        const script = document.createElement('script');

        script.src = scriptUrl;

        script.onload = callback;

        document.head.appendChild(script);

    }

    function loadGraphDescription(descriptionUrl) {

        fetch(descriptionUrl)

            .then(response => response.text())

            .then(text => {

                graphDescriptionElement.innerText = text;

            });

    }

    function plotCurrentGraph(angleModifier = 0) {

        loadGraphScript(graphConfigs[selectedGraph].script, function() {

            plotGraph(angleModifier, plotElement, isMuted, a);

        });

        loadGraphDescription(graphConfigs[selectedGraph].description);

    }

    function selectGraph(graph) {

        selectedGraph = graph;

        plotCurrentGraph();

        setSampleMusic();

        closeSubmenus();

    }

    function setSampleMusic() {

        const graph = graphConfigs[selectedGraph];

        if (graph.sampleMusic) {

            audioElement.src = graph.sampleMusic;

        }

    }

    function toggleMute() {

        isMuted = !isMuted;

        audioElement.muted = isMuted;

        plotCurrentGraph();

        if (isMuted) {

            showMuteOverlay();

        } else {

            removeMuteOverlay();

        }

    }

    function showMuteOverlay() {

        const overlay = document.createElement('div');

        overlay.className = 'mute-overlay';

        plotElement.appendChild(overlay);

    }

    function removeMuteOverlay() {

        const overlay = plotElement.querySelector('.mute-overlay');

        if (overlay) {

            plotElement.removeChild(overlay);

        }

    }

    function uploadMusic(event) {

        musicFiles = Array.from(event.target.files);

        if (musicFiles.length > 0) {

            setMusicSource(0);

        }

    }

    function setMusicSource(index) {

        if (index >= 0 && index < musicFiles.length) {

            const file = musicFiles[index];

            const reader = new FileReader();

            reader.onload = function(e) {

                audioElement.src = e.target.result;

            };

            reader.readAsDataURL(file);

        }

    }

    function connectToSpotify() {

        alert("Connect to Spotify feature coming soon!");

    }

    function startLoop() {

        let angle = 0;

        clearInterval(loopInterval);

        loopInterval = setInterval(() => {

            angle += Math.PI / 100;

            if (angle > Math.PI) angle = -Math.PI;

            plotCurrentGraph(angle);

        }, loopSpeed);

    }

    function stopLoop() {

        clearInterval(loopInterval);

        plotCurrentGraph();

    }

    function toggleLoop() {

        isLooping = !isLooping;

        if (isLooping) {

            loopBtn.classList.add('active');

            startLoop();

        } else {

            loopBtn.classList.remove('active');

            stopLoop();

        }

    }

    function closeSubmenus() {

        const submenus = document.querySelectorAll('.submenu');

        submenus.forEach(submenu => {

            submenu.style.display = 'none';

        });

    }

    plotElement.addEventListener('click', toggleMute);

    loopBtn.addEventListener('click', toggleLoop);

    playPauseBtn.addEventListener('click', function() {

        isPlaying = !isPlaying;

        if (isPlaying) {

            audioElement.play();

            playPauseBtn.textContent = 'Pause';

            if (isLooping) {

                startLoop();

            }

        } else {

            audioElement.pause();

            playPauseBtn.textContent = 'Play';

            stopLoop();

        }

    });

    function handleStart(event) {

        const initialA = a;

        const initialMouseX = event.clientX || event.touches[0].clientX;

        const initialMouseY = event.clientY || event.touches[0].clientY;

        function onMove(event) {

            const clientX = event.clientX || event.touches[0].clientX;

            const clientY = event.clientY || event.touches[0].clientY;

            const deltaX = clientX - initialMouseX;

            const deltaY = clientY - initialMouseY;

            a = Math.min(30, Math.max(0, initialA + deltaX / 10 - deltaY / 10));

            audioElement.volume = a / 30;

            plotCurrentGraph();

            if (a === 0 && !isMuted) {

                toggleMute();

            } else if (a > 0 && isMuted) {

                toggleMute();

            }

        }

        function onEnd() {

            document.removeEventListener('mousemove', onMove);

            document.removeEventListener('mouseup', onEnd);

            document.removeEventListener('touchmove', onMove);

            document.removeEventListener('touchend', onEnd);

        }

        document.addEventListener('mousemove', onMove);

        document.addEventListener('mouseup', onEnd);

        document.addEventListener('touchmove', onMove);

        document.addEventListener('touchend', onEnd);

    }

    plotElement.addEventListener('mousedown', handleStart);

    plotElement.addEventListener('touchstart', handleStart);

    audioElement.muted = false;

    audioElement.volume = a / 30;

    for (const [graphName, config] of Object.entries(graphConfigs)) {

        const thumbnailDiv = document.createElement('div');

        thumbnailDiv.className = 'thumbnail';

        thumbnailDiv.onclick = () => selectGraph(graphName);

        const img = document.createElement('img');

        img.src = config.thumbnail;

        img.alt = `${graphName} Graph`;

        const label = document.createElement('div');

        label.className = 'thumbnail-label';

        label.textContent = graphName.charAt(0).toUpperCase() + graphName.slice(1);

        thumbnailDiv.appendChild(img);

        thumbnailDiv.appendChild(label);

        graphSubmenu.appendChild(thumbnailDiv);

    }

    fileInput.addEventListener('change', uploadMusic);

    plotCurrentGraph();

    setSampleMusic();

    logoLink.addEventListener('click', function() {

        if (navbar.classList.contains('active')) {

            navbar.classList.remove('active');

            closeSubmenus();

        } else {

            navbar.classList.add('active');

        }

    });

    document.addEventListener('click', function(event) {

        const isClickInside = navbar.contains(event.target) || logoLink.contains(event.target);

        if (!isClickInside) {

            navbar.classList.remove('active');

            closeSubmenus();

        }

    });

    document.querySelectorAll('.nav-link').forEach(item => {

        item.addEventListener('click', function() {

            navbar.classList.add('active');

        });

    });

    window.addEventListener('resize', function() {

        const plotContainerRect = document.getElementById('plotContainer').getBoundingClientRect();

        const controlOverlay = document.getElementById('controlOverlay');

        controlOverlay.style.top = `${plotContainerRect.bottom}px`;

    });

    window.dispatchEvent(new Event('resize'));

    document.getElementById('youtubeLinkInput').addEventListener('keypress', function(event) {

        if (event.key === 'Enter') {

            addYouTubeLink();

        }

    });

});

function toggleSubmenu(id) {

    const submenu = document.getElementById(id);

    submenu.style.display = submenu.style.display === 'none' ? 'flex' : 'none';

}

function closeSubmenus() {

    const submenus = document.querySelectorAll('.submenu');

    submenus.forEach(submenu => {

        submenu.style.display = 'none';

    });

}

function connectToSpotify() {

    alert("Connect to Spotify feature coming soon!");

}

// YouTube API code

function onYouTubeIframeAPIReady() {

    console.log('YouTube IFrame API is ready');

}

function addYouTubeLink() {

    const linkInput = document.getElementById('youtubeLinkInput').value;

    const videoId = parseYouTubeLink(linkInput);

    if (videoId) {

        youtubeLinks.push(videoId);

        if (!player) {

            loadYouTubePlayer(videoId);

        }

    } else {

        alert('Invalid YouTube link');

    }

}

function parseYouTubeLink(link) {

    const regex = /(?:https?:\/\/)?(?:www\.)?(?:youtube\.com\/(?:[^\/\n\s]+\/\S+\/|(?:v|e(?:mbed)?)\/|\S\*?[?&]v=)|youtu\.be\/)([a-zA-Z0-9\_-]{11})/;

    const match = link.match(regex);

    return match ? match[1] : null;

}

function loadYouTubePlayer(videoId) {

    player = new YT.Player('player', {

        height: '360',

        width: '640',

        videoId: videoId,

        playerVars: {

            'autoplay': 1,

            'controls': 0

        },

        events: {

            'onReady': onPlayerReady,

            'onStateChange': onPlayerStateChange

        }

    });

}

function onPlayerReady(event) {

    event.target.setVolume(a \* 3.33);

    event.target.playVideo();

}

function onPlayerStateChange(event) {

    if (event.data === YT.PlayerState.ENDED) {

        currentVideoIndex = (currentVideoIndex + 1) % youtubeLinks.length;

        player.loadVideoById(youtubeLinks[currentVideoIndex]);

    }

}

function addToPlaylist(videoId) {

    fetch(`https://www.googleapis.com/youtube/v3/videos?part=snippet&id=${videoId}&key=YOUR\_API\_KEY`)

        .then(response => response.json())

        .then(data => {

            const videoData = data.items[0];

            const thumbnail = videoData.snippet.thumbnails.default.url;

            const title = videoData.snippet.title;

            const playlistItem = document.createElement('div');

            playlistItem.className = 'playlist-item';

            playlistItem.innerHTML = `

                <img src="${thumbnail}" alt="${title}" class="playlist-thumbnail">

                <div class="playlist-info">

                    <div class="playlist-title">${title}</div>

                    <div class="playlist-status" id="status-${videoId}">Paused</div>

                </div>

                <div class="playlist-controls">

                    <button onclick="playVideo('${videoId}')">Play</button>

                    <button onclick="removeFromPlaylist('${videoId}')">Remove</button>

                </div>

            `;

            playlistElement.appendChild(playlistItem);

        });

}

window.playVideo = function(videoId) {

    player.loadVideoById(videoId);

    updatePlaylistStatus(videoId, 'Playing');

};

window.removeFromPlaylist = function(videoId) {

    youtubeLinks = youtubeLinks.filter(id => id !== videoId);

    document.getElementById(`item-${videoId}`).remove();

};

function updatePlaylistStatus(videoId, status) {

    const statusElement = document.getElementById(`status-${videoId}`);

    if (statusElement) {

        statusElement.textContent = status;

        statusElement.style.color = status === 'Playing' ? 'green' : 'yellow';

    }

}

.

Theme.js: const themeMap = {

    dark: "light",

    light: "solar",

    solar: "dark"

};

const theme = localStorage.getItem('theme')

    || (tmp = Object.keys(themeMap)[0],

        localStorage.setItem('theme', tmp),

        tmp);

const bodyClass = document.body.classList;

bodyClass.add(theme);

function toggleTheme() {

    const current = localStorage.getItem('theme');

    const next = themeMap[current];

    bodyClass.replace(current, next);

    localStorage.setItem('theme', next);

}

document.getElementById('themeButton').onclick = toggleTheme;

.

Heart.js: function plotGraph(angleModifier, plotElement, isMuted, a) {

    const e = Math.E;

    const π = Math.PI;

    function calculateY(x, isPositive) {

        const angle = (isPositive ? π : -π) \* x + angleModifier;

        const y = Math.pow(Math.abs(x), 2 / 3) + (e / 3) \* Math.sqrt(π - Math.pow(x, 2)) \* Math.sin(a \* angle);

        return x === 0 ? 0 : y;

    }

    const xValues = [];

    const yValuesPositive = [];

    for (let x = -2; x <= 2; x += 0.01) {

        xValues.push(x);

        yValuesPositive.push(calculateY(x, true));

    }

    Plotly.newPlot(plotElement, [

        {

            x: xValues,

            y: yValuesPositive,

            type: 'scatter',

            mode: 'lines',

            line: { color: isMuted ? 'grey' : 'rgba(255, 99, 71, 0.6)', width: isMuted ? 1 : Math.max(1, 5 - (a / 6)) },

            hoverinfo: 'skip'

        }

    ], {

        xaxis: { range: [-2.3, 2.3], visible: false, scaleanchor: 'y', scaleratio: 1 },

        yaxis: { range: [-1.5, 2.5], visible: false },

        plot\_bgcolor: 'transparent',

        paper\_bgcolor: 'transparent',

        showlegend: false,

        margin: { l: 0, r: 0, t: 0, b: 0 },

        hovermode: false

    }, { displayModeBar: false });

}

.

Graph2.js: function plotGraph(angleModifier, plotElement, isMuted, a) {

    const e = Math.E;

    const π = Math.PI;

    function calculateY(x, isPositive) {

        const y = Math.cbrt(Math.pow(x, 2)) + (π / 4) \* Math.cos(π \* x) + (1 / 5) \* Math.exp(-Math.pow(x, 2));

        return x === 0 ? 0 : y;

    }

    const xValues = [];

    const yValuesPositive = [];

    for (let x = -2; x <= 2; x += 0.01) {

        xValues.push(x);

        yValuesPositive.push(calculateY(x, true));

    }

    Plotly.newPlot(plotElement, [

        {

            x: xValues,

            y: yValuesPositive,

            type: 'scatter',

            mode: 'lines',

            line: { color: isMuted ? 'grey' : 'rgba(255, 99, 71, 0.6)', width: isMuted ? 1 : Math.max(1, 5 - (a / 6)) },

            hoverinfo: 'skip'

        }

    ], {

        xaxis: { range: [-2.3, 2.3], visible: false, scaleanchor: 'y', scaleratio: 1 },

        yaxis: { range: [-1.5, 2.5], visible: false },

        plot\_bgcolor: 'transparent',

        paper\_bgcolor: 'transparent',

        showlegend: false,

        margin: { l: 0, r: 0, t: 0, b: 0 },

        hovermode: false

    }, { displayModeBar: false });

}

.

The music player project:

Index.html: <!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <!-- font awesome cdn link -->

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.1/css/all.min.css"

        integrity="sha512-DTOQO9RWCH3ppGqcWaEA1BIZOC6xxalwEsw9c2QQeAIftl+Vegovlnee1c9QX4TctnWMn13TZye+giMm8e2LwA=="

        crossorigin="anonymous" referrerpolicy="no-referrer" />

    <title>Music Player</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <div class="player-container">

        <div class="player">

            <div class="details">

                <div class="track-art">

                    <svg width="100%" height="100%" viewBox="0 0 24 24" fill="none" xmlns="http://www.w3.org/2000/svg">

                        <path d="M12 0C5.37258 0 0 5.37258 0 12C0 18.6274 5.37258 24 12 24C18.6274 24 24 18.6274 24 12C24 5.37258 18.6274 0 12 0ZM12 22C6.47715 22 2 17.5228 2 12C2 6.47715 6.47715 2 12 2C17.5228 2 22 6.47715 22 12C22 17.5228 17.5228 22 12 22Z" fill="#ffffff"/>

                        <path d="M12 5.5C10.067 5.5 8.5 7.067 8.5 9C8.5 10.933 10.067 12.5 12 12.5C13.933 12.5 15.5 10.933 15.5 9C15.5 7.067 13.933 5.5 12 5.5ZM12 11.5C10.8954 11.5 10 10.6046 10 9.5C10 8.39543 10.8954 7.5 12 7.5C13.1046 7.5 14 8.39543 14 9.5C14 10.6046 13.1046 11.5 12 11.5Z" fill="#ffffff"/>

                        <path d="M12 14C8.13401 14 5 17.134 5 21H7C7 18.2386 9.23858 16 12 16C14.7614 16 17 18.2386 17 21H19C19 17.134 15.866 14 12 14Z" fill="#ffffff"/>

                    </svg>

                </div>

                <div class="track-info">

                    <div class="track-name">Track Name</div>

                    <div class="track-artist">Track Artist</div>

                </div>

            </div>

            <div class="player-controls">

                <div class="buttons">

                    <div class="prev-track" onclick="prevTrack()">

                        <i class="fa fa-step-backward"></i>

                    </div>

                    <div class="playpause-track" onclick="playpauseTrack()">

                        <i class="fa fa-play-circle"></i>

                    </div>

                    <div class="next-track" onclick="nextTrack()">

                        <i class="fa fa-step-forward"></i>

                    </div>

                    <div class="random-track" onclick="randomTrack()">

                        <i class="fas fa-random" title="random"></i>

                    </div>

                    <div class="repeat-track" onclick="repeatTrack()">

                        <i class="fa fa-repeat" title="repeat"></i>

                    </div>

                </div>

                <div class="slider-container">

                    <input type="range" min="1" max="100" value="0" class="seek-slider" onchange="seekTo()">

                </div>

                <div class="slider-container">

                    <input type="range" min="1" max="100" value="99" class="volume-slider" onchange="setVolume()">

                </div>

                <div class="info">

                    <div class="current-time">00:00</div>

                    <div class="volume-percentage">99%</div>

                </div>

            </div>

        </div>

        <div class="separator"></div>

        <div class="playlist">

            <div class="track-list"></div>

        </div>

    </div>

    <script src="main.js"></script>

</body>

</html>

.

Style.css: /\* Root Variables for Customization \*/

:root {

    --primary-bg: #0a0057;

    --secondary-bg: #3f00ee;

    --primary-text: #fff;

    --secondary-text: gold;

    --control-bg: rgba(0, 0, 0, 0.7);

    --highlight-color: #83a9ff;

    --active-color: #ff9f1a;

    --hover-color: #3f51b5;

    --alt-bg-color: #2a2a2a;

    --border-radius: 20px;

    --box-shadow: 0 0 20px rgba(0, 0, 0, 0.5);

    --track-art-size: 40vw; /\* Use percentage for responsiveness \*/

    --track-art-max-size: 150px; /\* Max size for larger screens \*/

    --track-art-radius: 50%;

    --track-height: 70px; /\* Track height customization \*/

}

/\* Universal Reset \*/

\* {

    margin: 0;

    padding: 0;

    box-sizing: border-box;

}

/\* Body Styling \*/

body {

    min-height: 100vh;

    background: var(--primary-bg);

    background: linear-gradient(to bottom right, var(--primary-bg), var(--secondary-bg));

    font-family: sans-serif;

    color: var(--primary-text);

    display: flex;

    justify-content: center;

    align-items: center;

}

/\* Player Container Styling \*/

.player-container {

    width: 100%;

    max-width: 400px;

    background-color: var(--control-bg);

    border-radius: var(--border-radius);

    overflow: hidden;

    box-shadow: var(--box-shadow);

    display: flex;

    flex-direction: column;

}

/\* Player Styling \*/

.player {

    padding: 20px;

}

/\* Track Details Styling \*/

.details {

    display: flex;

    flex-direction: column;

    align-items: center;

    text-align: center;

}

.track-art {

    width: var(--track-art-size);

    height: var(--track-art-size);

    max-width: var(--track-art-max-size);

    max-height: var(--track-art-max-size);

    background-color: #333;

    background-size: cover;

    background-position: center;

    border-radius: var(--track-art-radius);

    margin-bottom: 15px;

}

.track-info {

    flex: 1;

}

.track-name {

    font-size: 1.2em;

    color: var(--secondary-text);

    margin-bottom: 5px;

}

.track-artist {

    font-size: 1em;

    color: var(--primary-text);

}

/\* Player Controls Styling \*/

.player-controls {

    margin-top: 15px;

}

.buttons {

    display: flex;

    justify-content: space-around;

    align-items: center;

    margin-bottom: 15px;

}

.buttons div {

    color: var(--primary-text);

    cursor: pointer;

    font-size: 1.5em;

}

.slider-container {

    display: flex;

    justify-content: center;

    align-items: center;

    margin-bottom: 10px;

}

.seek-slider {

    width: 80%;

    height: 5px;

    background: var(--highlight-color);

    border-radius: 5px;

    cursor: pointer;

}

.volume-slider {

    width: 30%;

    height: 5px;

    background: var(--highlight-color);

    border-radius: 5px;

    cursor: pointer;

    margin-top: 5px;

}

.seek-slider::-webkit-slider-thumb, .volume-slider::-webkit-slider-thumb {

    width: 10px;

    height: 10px;

    background: var(--highlight-color);

    border: none;

    border-radius: 50%;

    cursor: pointer;

}

.info {

    display: flex;

    justify-content: space-between;

    font-size: 0.8em;

    color: #fff;

}

/\* Separator Styling \*/

.separator {

    width: 95%;

    height: 2px;

    background: linear-gradient(to right, transparent, #aaa, transparent);

    margin: 10px auto;

}

/\* Playlist Styling \*/

.playlist {

    padding: 20px;

    flex: 1;

    overflow-y: auto;

}

.track-list {

    max-height: 200px; /\* Ensure the tracklist has a scroll \*/

    overflow-y: auto;

}

.track-list .track {

    display: flex;

    align-items: center;

    padding: 10px 0;

    cursor: pointer;

    height: var(--track-height);

    background-color: var(--control-bg);

}

.track-list .track:nth-child(even) {

    background-color: var(--alt-bg-color);

}

.track-list .track.active {

    background-color: var(--active-color);

}

.track-list .track:hover {

    background-color: var(--hover-color);

}

.track-list .track img {

    width: 40px;

    height: 40px;

    margin-right: 15px;

}

.track-list .track .track-info {

    flex: 1;

    display: flex;

    flex-direction: column;

    justify-content: center;

}

.track-list .track .track-name {

    font-size: 0.9em;

    color: var(--primary-text);

    margin-bottom: 5px;

}

.track-list .track .track-artist {

    font-size: 0.8em;

    color: var(--secondary-text);

}

/\* Custom scrollbar for track-list \*/

.track-list::-webkit-scrollbar {

    width: 8px;

}

.track-list::-webkit-scrollbar-thumb {

    background: var(--highlight-color);

    border-radius: 10px;

}

/\* Media Queries for Responsiveness \*/

@media (max-width: 768px) {

    .player-container {

        max-width: 90vw;

    }

    .track-art {

        width: 50%;

        height: 50%;

    }

    .buttons div {

        font-size: 1.2em;

    }

    .slider-container {

        flex-direction: column;

    }

    .seek-slider {

        width: 100%;

    }

    .volume-slider {

        width: 50%;

    }

    .info {

        flex-direction: column;

    }

}

@media (max-width: 480px) {

    .track-name, .track-artist {

        font-size: 1em;

    }

    .buttons div {

        font-size: 1em;

    }

    .slider-container {

        flex-direction: column;

    }

    .seek-slider {

        width: 100%;

    }

    .volume-slider {

        width: 60%;

    }

    .info {

        flex-direction: column;

        align-items: center;

    }

}

.

main.js: // Audio Player and Track Elements

const audioPlayer = document.createElement('audio');

const trackArt = document.querySelector('.track-art');

const trackName = document.querySelector('.track-name');

const trackArtist = document.querySelector('.track-artist');

const playPauseBtn = document.querySelector('.playpause-track');

const nextBtn = document.querySelector('.next-track');

const prevBtn = document.querySelector('.prev-track');

const randomBtn = document.querySelector('.random-track');

const repeatBtn = document.querySelector('.repeat-track');

const seekSlider = document.querySelector('.seek-slider');

const volumeSlider = document.querySelector('.volume-slider');

const currTime = document.querySelector('.current-time');

const volumePercentage = document.querySelector('.volume-percentage');

const trackList = document.querySelector('.track-list');

let trackIndex = 0;

let isPlaying = false;

let isRandom = false;

let updateTimer;

// Default Values

const defaultArt = `<svg width="100%" height="100%" viewBox="0 0 24 24" fill="none" xmlns="http://www.w3.org/2000/svg">

    <path d="M12 0C5.37258 0 0 5.37258 0 12C0 18.6274 5.37258 24 12 24C18.6274 24 24 18.6274 24 12C24 5.37258 18.6274 0 12 0ZM12 22C6.47715 22 2 17.5228 2 12C2 6.47715 6.47715 2 12 2C17.5228 2 22 6.47715 22 12C22 17.5228 17.5228 22 12 22Z" fill="#ffffff"/>

    <path d="M12 5.5C10.067 5.5 8.5 7.067 8.5 9C8.5 10.933 10.067 12.5 12 12.5C13.933 12.5 15.5 10.933 15.5 9C15.5 7.067 13.933 5.5 12 5.5ZM12 11.5C10.8954 11.5 10 10.6046 10 9.5C10 8.39543 10.8954 7.5 12 7.5C13.1046 7.5 14 8.39543 14 9.5C14 10.6046 13.1046 11.5 12 11.5Z" fill="#ffffff"/>

    <path d="M12 14C8.13401 14 5 17.134 5 21H7C7 18.2386 9.23858 16 12 16C14.7614 16 17 18.2386 17 21H19C19 17.134 15.866 14 12 14Z" fill="#ffffff"/>

</svg>`;

const defaultTrack = {

    img: defaultArt,

    name: "Unknown Track",

    artist: "Unknown Artist",

    music: "",

};

// Music List (Edit to customize tracks)

const musicList = [

    {

        img: "assets/images/OrkhanZeynalli.jpg",

        name: "Iller Sonra",

        artist: "Orkhan Zeynalli",

        music: "assets/music/İller Sonra.mp3",

    },

    {

        img: "assets/images/ceza.jpg",

        name: "Neyim Var Ki",

        artist: "CEZA",

        music: "assets/music/Neyim Var Ki.mp3",

    },

    {

        img: "assets/images/Gazapizm.jpg",

        name: "Unutulacak Dunler",

        artist: "Gazapizm",

        music: "assets/music/Unutulacak Dunler.mp3",

    },

    {

        img: "assets/images/AID.jpg",

        name: "Boyuk Umidler",

        artist: "Orkhan Zeynalli",

        music: "assets/music/Boyuk Umidler.mp3",

    },

];

// Load Track Function

function loadTrack(trackIndex) {

    clearInterval(updateTimer);

    reset();

    let track = musicList[trackIndex] || defaultTrack;

    audioPlayer.src = track.music || "";

    if (track.music) {

        audioPlayer.load();

    }

    if (track.img) {

        trackArt.style.backgroundImage = "url(" + track.img + ")";

    } else {

        trackArt.innerHTML = defaultArt;

    }

    trackName.textContent = track.name || defaultTrack.name;

    trackArtist.textContent = track.artist || defaultTrack.artist;

    updateTimer = setInterval(setUpdate, 1000);

    audioPlayer.addEventListener("ended", nextTrack);

    updatePlaylistActiveState();

}

// Update Playlist Active State

function updatePlaylistActiveState() {

    const tracks = document.querySelectorAll('.track');

    tracks.forEach(track => {

        track.classList.remove('active');

    });

    tracks[trackIndex].classList.add('active');

}

// Reset Track Function

function reset() {

    currTime.textContent = "00:00";

    volumePercentage.textContent = "99%";

    seekSlider.value = 0;

}

// Random Track Functions

function randomTrack() {

    isRandom ? pauseRandom() : playRandom();

}

function playRandom() {

    isRandom = true;

    randomBtn.classList.add("active");

}

function pauseRandom() {

    isRandom = false;

    randomBtn.classList.remove("active");

}

// Repeat Track Function

function repeatTrack() {

    loadTrack(trackIndex);

    playTrack();

}

// Play/Pause Track Function

function playpauseTrack() {

    isPlaying ? pauseTrack() : playTrack();

}

function playTrack() {

    audioPlayer.play();

    isPlaying = true;

    playPauseBtn.innerHTML = '<i class="fa fa-pause-circle"></i>';

}

function pauseTrack() {

    audioPlayer.pause();

    isPlaying = false;

    playPauseBtn.innerHTML = '<i class="fa fa-play-circle"></i>';

}

// Next/Previous Track Functions

function nextTrack() {

    if (trackIndex < musicList.length - 1 && !isRandom) {

        trackIndex += 1;

    } else if (trackIndex < musicList.length - 1 && isRandom) {

        let randomIndex = Math.floor(Math.random() \* musicList.length);

        trackIndex = randomIndex;

    } else {

        trackIndex = 0;

    }

    loadTrack(trackIndex);

    playTrack();

}

function prevTrack() {

    if (trackIndex > 0) {

        trackIndex -= 1;

    } else {

        trackIndex = musicList.length - 1;

    }

    loadTrack(trackIndex);

    playTrack();

}

// Seek and Volume Functions

function seekTo() {

    let seekTo = audioPlayer.duration \* (seekSlider.value / 100);

    audioPlayer.currentTime = seekTo;

}

function setVolume() {

    audioPlayer.volume = volumeSlider.value / 100;

    volumePercentage.textContent = Math.round(audioPlayer.volume \* 100) + "%";

}

// Update Time and Duration

function setUpdate() {

    let seekPosition = 0;

    if (!isNaN(audioPlayer.duration)) {

        seekPosition = audioPlayer.currentTime \* (100 / audioPlayer.duration);

        seekSlider.value = seekPosition;

        let currentMinutes = Math.floor(audioPlayer.currentTime / 60);

        let currentSeconds = Math.floor(audioPlayer.currentTime - currentMinutes \* 60);

        let durationMinutes = Math.floor(audioPlayer.duration / 60);

        let durationSeconds = Math.floor(audioPlayer.duration - durationMinutes \* 60);

        if (currentSeconds < 10) { currentSeconds = "0" + currentSeconds; }

        if (durationSeconds < 10) { durationSeconds = "0" + durationSeconds; }

        if (currentMinutes < 10) { currentMinutes = "0" + currentMinutes; }

        if (durationMinutes < 10) { durationMinutes = "0" + durationMinutes; }

        currTime.textContent = currentMinutes + ":" + currentSeconds;

    }

}

// Render Playlist

function renderPlaylist() {

    musicList.forEach((track, index) => {

        let trackElement = `

            <div class="track" data-index="${index}">

                <img src="${track.img || ''}" alt="play icon">

                <div class="track-info">

                    <div class="track-name">${track.name || defaultTrack.name}</div>

                    <div class="track-artist">${track.artist || defaultTrack.artist}</div>

                </div>

            </div>

        `;

        trackList.innerHTML += trackElement;

    });

    const tracks = document.querySelectorAll('.track');

    tracks.forEach(track => {

        track.onclick = () => {

            trackIndex = parseInt(track.getAttribute('data-index'));

            loadTrack(trackIndex);

            playTrack();

        }

    });

}

// Initialize Player

renderPlaylist();

loadTrack(trackIndex);

.