

فایل داکيومنت برای یک وبسایت ساخته شد

## React

نام و نام خانوادگی: سید علی رفاضی

شماره دانشجویی: ۹۸۰۱۲۹۵۱۰

درس: طراحی نرم افزار

ساعت تشکیل کلاس: پنجشنبه ساعت ۱۷:۳۰ تا ۲۰

توجه شود این سایت در دامنه زیر بالا آمده است :

rafaziportfolio.ir

ادرس گیت هاب :

<https://github.com/alirfz/portfolio-react.git>

همچنین اگر به مشکل timeout برخورد کرده اید در آخرین راه فایل زیر را دانلود کنید  
که شامل node\_modules نیز میباشد :

<https://s29.picofile.com/d/8465246492/da1818a8-3510-4bbf-a560-5e49d6973503/portfolio.rar>

۳	..... Useability of the Project
۵	..... Build فولدر
۷	..... node_modules فولدر
۹	..... node_modules چند بلاگین مهم استفاده شده
۱۴	..... Public فولدر
۱۷	..... Src فولدر
۱۹	..... component فولدر در فولدر src
۱۹	..... LineGradient.jsx فایل
۲۰	..... SocialMediaIcons.jsx فایل
۲۱	..... scenes فولدر در فولدر src
۲۲	..... landing.jsx فایل
۲۲	..... MySkills.jsx فایل
۲۳	..... Projects.jsx فایل
۲۴	..... Feedbacks.jsx فایل
۲۵	..... About.jsx فایل
۲۶	..... Experience.jsx فایل
۲۷	..... Contact.jsx فایل
۲۸	..... Footer.jsx فایل
۲۹	..... App.js فایل

## Useability of the Project

این وب سایت Portfolio با استفاده از React، یک کتابخانه جاوا اسکریپت برای ساخت رابط های کاربری ساخته شده است. این وب سایت برای نشان دادن مهارت ها و تجربه من به عنوان یک توسعه دهنده وب طراحی شده است.

## Technologies Used

وب سایت با استفاده از فناوری های زیر ساخته شده است:

React

JavaScript

Tailwind CSS

Framer Motion(for animation)

React Hook Form

CSS

Html

## Architecture and Infrastructure

وب سایت به عنوان یک برنامه کاربردی تک صفحه ای (SPA) طراحی شده است. این به این معنی است که در اولین بازدید کاربر از وب سایت، کل برنامه در مرورگر بارگذاری می شود. این باعث می شود برنامه بسیار سریع و پاسخگو باشد.

این وبسایت بر روی هاست بالا آمده است که ادرس آن به شرح زیر است:

Rafaziportfolio.ir

# Deployment on the Web

The website can be deployed on the web using the following steps:

1. Create a new GitHub repository for the project.
2. Clone the repository to your local machine.
3. Install the dependencies using `npm install`.
4. Run the project using `npm start`.
5. Then run `npm run build`
6. Upload build folder into your host (also you can upload into vercel )

# فایل‌بندی های پروژه

## فولدر Build

فایل ساخت در React فایل‌ای است که با اجرای دستور `npm run build` ایجاد می‌شود. این دستور پروژه React شما را می‌گیرد و یک بیلد آماده تولید از آن ایجاد می‌کند. فایل ساخت مجموعه‌ای از فایل‌های ثابت است که می‌تواند توسط یک وب سرور ارائه شود.

فایل ساخت شامل جاوا اسکریپت، CSS و فایل‌های تصویری شماست. همچنین حاوی یک فایل HTML است که به وب سرور می‌گوید که چگونه برنامه شما را رندر کند. فایل بیلد معمولاً بسیار کوچکتر از کد منبع شما است زیرا کوچک شده و gzip شده است.

فایل ساخت مهم است زیرا به شما امکان می‌دهد برنامه React خود را در یک محیط تولید مستقر کنید. هنگامی که برنامه خود را مستقر می‌کنید، باید فایل ساخت را از یک وب سرور ارائه دهید.

فایل ساخت نیز برای عملکرد مهم است. کوچک سازی و gzipping فایل ساخت می‌تواند زمان بارگذاری برنامه شما را به میزان قابل توجهی بهبود بخشد.

هنگام کار با فایل بیلد در React باید به چند نکته توجه داشته باشید:

فایل ساخت بخشی از کد منبع شما نیست. این یک فایل جداگانه است که با اجرای دستور `npm run build` ایجاد می‌شود.

فایل ساخت معمولاً بسیار کوچکتر از کد منبع شما است. این به این دلیل است که کوچک شده و gzip شده است.

فایل ساخت برای استقرار برنامه React شما در محیط تولید مهم است.

فایل ساخت می تواند عملکرد برنامه شما را بهبود بخشد.

در اینجا برخی از مزایای استفاده از فایل بیلد در React آورده شده است:

اندازه فایل کوچکتر: فایل ساخت معمولاً بسیار کوچکتر از کد منبع شما است. این به این دلیل است که کوچک شده و gzip شده است. اندازه فایل های کوچکتر می تواند زمان بارگذاری برنامه شما را بهبود بخشد.

عملکرد بهبود یافته: کوچک سازی و gzipping فایل ساخت می تواند به طور قابل توجهی عملکرد برنامه شما را بهبود بخشد.

استقرار آسانتر: فایل بیلد یک فایل جداگانه است که به راحتی می توان آن را در محیط تولید مستقر کرد.

امنیت بهبود یافته: کوچک سازی فایل ساخت می تواند امنیت برنامه شما را با سخت تر کردن مهندسی معکوس کد شما برای مهاجمان بهبود بخشد.

در اینجا برخی از اشکالات استفاده از فایل بیلد در React آورده شده است:

اشکال زدایی ممکن است دشوار باشد: فایل بیلد یک فایل کوچک شده و gzip است که می تواند اشکال زدایی را دشوار کند.

سفارشی کردن آن دشوار است: فایل ساخت یک فایل ثابت است، به این معنی که سفارشی کردن آن دشوار است.

می تواند کند باشد: تولید فایل ساخت، به خصوص برای برنامه های بزرگ، ممکن است کمی طول بکشد.

## فولدر node\_modules

پوشه node\_modules یک دایرکتوری است که هنگام نصب کتابخانه ها یا بسته های شخص ثالث در پروژه React خود ایجاد می شود. این شامل تمام کدهای کامپایل شده و دارایی های کتابخانه هایی است که شما نصب کرده اید.

پوشه node\_modules معمولاً در ریشه پروژه React شما قرار دارد. این یک پوشه مخفی است، بنابراین برای دیدن آن باید فایل های مخفی را در اکسپلورر فایل خود نشان دهید.

پوشه node\_modules مهم است زیرا به شما امکان می دهد از کتابخانه های شخص ثالث در پروژه React خود استفاده کنید بدون اینکه نیازی به گنجاندن کد منبع آن کتابخانه ها در پروژه خود داشته باشید. این می تواند فضای زیادی را برای شما ذخیره کند و مدیریت پروژه شما را آسان تر کند.

با این حال، پوشه node\_modules نیز می تواند منبع مشکلات باشد.

اگر تعداد زیادی کتابخانه شخص ثالث نصب کرده اید، پوشه node\_modules می تواند بسیار بزرگ شود. این می تواند روند توسعه شما را کند کند و ردیابی مشکلات در کد شما را دشوار کند.

چند کار وجود دارد که می توانید برای مدیریت اندازه پوشه node\_modules انجام دهید:

فقط کتابخانه های شخص ثالثی را که واقعاً نیاز دارید نصب کنید.

از یک مدیر بسته مانند npm یا yarn برای مدیریت وابستگی های خود استفاده کنید. این مدیران بسته به شما کمک می کنند تا کتابخانه های شخص ثالثی را که نصب کرده اید پیگیری کنید و در صورت لزوم آنها را به روز کنید.

از ابزاری مانند npm prune برای حذف وابستگی های استفاده نشده از پوشه node\_modules استفاده کنید.

پوشه node\_modules بخشی ضروری از توسعه React است. با این حال، مدیریت اندازه پوشه node\_modules برای جلوگیری از مشکلات عملکرد و ردیابی مشکلات در کد شما بسیار مهم است.

در اینجا برخی از مزایای استفاده از پوشه node\_modules در React آورده شده است:

راحت: به شما این امکان را می دهد که از کتابخانه های شخص ثالث در پروژه React خود استفاده کنید بدون اینکه نیازی به گنجاندن کد منبع آن کتابخانه ها در پروژه خود داشته باشید. این می تواند فضای زیادی را برای شما ذخیره کند و مدیریت پروژه شما را آسان تر کند.

Portable: پوشه node\_modules یک دایرکتوری استاندارد است که توسط تمام پروژه های React استفاده می شود. این باعث می شود پروژه خود را با دیگران به اشتراک بگذارید یا آن را در یک محیط تولید مستقر کنید.

مقیاس پذیر: پوشه node\_modules را می توان برای گنجاندن هر اندازه پروژه بزرگ کرد. این باعث می شود که برای پروژه های بزرگ و پیچیده React گزینه خوبی باشد.



## چند بلاگین مهم استفاده شده node\_modules

React و react-dom کتابخانه های اصلی React هستند. آن ها بلوک های اساسی برای ایجاد اجزای React را فراهم می کنند.

Redux یک کتابخانه مدیریت حالت است که به شما کمک می کند تا وضعیت برنامه React خود را مدیریت کنید. Redux یک انتخاب محبوب برای مدیریت وضعیت در برنامه های بزرگ و پیچیده React است. این یک کتابخانه مستند و به خوبی نگهداری شده است و جامعه بزرگی از توسعه دهندگان وجود دارد که از آن استفاده می کنند و در آن مشارکت دارند.

React Router یک کتابخانه مسیریابی است که به شما کمک می کند بین صفحات مختلف برنامه React خود حرکت کنید. React Router یک کتابخانه ساده و با کاربری آسان است که ایجاد طرح های مسیریابی پیچیده را آسان می کند. همچنین به خوبی مستند شده و به خوبی نگهداری می شود.

Webpack ابزاری برای ساخت است که به شما کمک می کند تا برنامه React خود را در یک فایل واحد قرار دهید که می تواند در یک محیط تولید مستقر شود. Webpack یک ابزار ساخت قدرتمند و همه کاره است که می تواند برای بسته بندی برنامه های React و همچنین انواع دیگر برنامه های جاوا اسکریپت استفاده شود. به خوبی مستند شده و به خوبی نگهداری می شود، و جامعه بزرگی از توسعه دهندگان وجود دارند که از آن استفاده می کنند و به آن کمک می کنند.

Babel ترانسپایلری است که به شما کمک می کند کد React خود را به جاوا اسکریپت تبدیل کنید که می تواند در مرورگرهای مدرن اجرا شود. Babel یک ابزار ضروری برای توسعه برنامه های React است زیرا به شما امکان می دهد از ویژگی های جاوا اسکریپت مدرن استفاده کنید که هنوز توسط همه مرورگرها پشتیبانی نمی شوند. Babel به خوبی مستند شده و به خوبی نگهداری می شود و جامعه بزرگی از توسعه دهندگان وجود دارند که از آن استفاده می کنند و به آن کمک می کنند.

Jest یک چارچوب آزمایشی است که به شما کمک می کند تا برنامه React خود را آزمایش کنید. Jest یک انتخاب محبوب برای آزمایش برنامه های React است زیرا استفاده از آن آسان است و دارای طیف گسترده ای از ویژگی ها است. Jest به خوبی مستند شده و به خوبی نگهداری می شود و جامعه بزرگی از توسعه دهندگان وجود دارند که از آن استفاده می کنند و به آن کمک می کنند.

eslint و prettier خط‌هایی هستند که به شما کمک می‌کنند کدی تمیز و یکدست بنویسید. ESLint خطی است که به شما کمک می کند تا خطاها و مشکلات احتمالی را در کد خود پیدا کنید. Prettier یک فرمت‌کننده کد است که به شما کمک می‌کند کد خود را به سبکی ثابت قالب‌بندی کنید. هر دو ESLint و Prettier ابزارهای مهمی برای نوشتن کد React با کیفیت هستند.

Framer Motion یک کتابخانه React است که ایجاد انیمیشن و تعامل در برنامه های React شما را آسان می کند. این یک کتابخانه اعلامی است، به این معنی که می توانید انیمیشن یا تعامل مورد نظر را از نظر وضعیت و ویژگی های آن توصیف کنید. Framer Motion طیف گسترده ای از انیمیشن ها و تعاملات اولیه، از جمله انتقال، فنر و ژست

ها را ارائه می دهد. همچنین تعدادی توابع کاربردی را ارائه می دهد که ایجاد انیمیشن ها و تعاملات پیچیده را آسان می کند.

React Hook Form کتابخانه ای است که مدیریت حالت فرم را در برنامه های React شما آسان می کند. از قلاب های React برای ارائه روشی برای مدیریت حالت فرم استفاده می کند. React Hook Form تعدادی ویژگی را ارائه می دهد که ایجاد و مدیریت فرم ها را آسان می کند، از جمله:

مدیریت حالت: React Hook Form راهی برای مدیریت وضعیت فیلدهای فرم ارائه می دهد.

اعتبار سنجی: React Hook Form راهی برای اعتبارسنجی فیلدهای فرم شما ارائه می دهد.

ارسال: React Hook Form راهی برای ارسال اطلاعات فرم شما فراهم می کند.

React Anchor Link Smooth Scroll کتابخانه ای است که اضافه کردن اسکرول صاف به پیوندهای لنگر در برنامه های React شما را آسان می کند. از روش scrollTo برای اسکرول هموار به هدف پیوند لنگر استفاده می کند. React Anchor Link Smooth Scroll تعدادی ویژگی را ارائه می دهد که اضافه کردن اسکرول صاف به پیوندهای لنگر را آسان می کند، از جمله:

سرعت اسکرول: می توانید سرعت پیمایش را کنترل کنید.

Easing: می توانید آسان شدن اسکرول را کنترل کنید.

تماس‌های برگشتی: می‌توانید تماس‌هایی را ثبت کنید که در صورت شروع، پایان یا شکست پیمایش فراخوانی می‌شوند.

react-vertical-timeline-component یک کتابخانه سبک و با کاربری آسان است که تعدادی ویژگی را فراهم می‌کند که ایجاد خطوط زمانی عمودی را آسان می‌کند، از جمله:

آیتم‌های جدول زمانی: می‌توانید آیتم‌های جدول زمانی را با متن، تصاویر و محتوای دیگر ایجاد کنید.

رویدادهای خط زمانی: می‌توانید رویدادها را به موارد جدول زمانی اضافه کنید تا تاریخ‌ها یا نقاط عطف مهم را ردیابی کنید.

سبک‌های تایم لاین: می‌توانید سبک‌های خط زمانی خود را مطابق با طراحی برنامه‌تان سفارشی کنید.

تعاملات جدول زمانی: می‌توانید تعاملاتی را به جدول زمانی خود اضافه کنید، مانند امکان کلیک بر روی موارد جدول زمانی برای مشاهده اطلاعات بیشتر.

react-vertical-timeline-component یک انتخاب عالی برای ایجاد خطوط زمانی عمودی در برنامه‌های React شما است. استفاده از آن آسان است و تعدادی ویژگی را فراهم می‌کند که ایجاد جدول زمانی زیبا و تعاملی را آسان می‌کند.

در اینجا برخی از مزایای استفاده از React-vertical-Timeline-Component آورده شده است:

استفاده از آن آسان است: کتابخانه به خوبی مستند شده است و تعدادی مثال ارائه می‌دهد که شروع به کار را آسان می‌کند.

این سبک وزن است: حجم کتابخانه تنها چند کیلوبایت است، بنابراین وزن قابل توجهی به برنامه شما اضافه نخواهد کرد.

انعطاف پذیر است: کتابخانه تعدادی گزینه سفارشی سازی را فراهم می کند، بنابراین می توانید جدول زمانی را با نیازهای خاص خود تنظیم کنید.

## فولدر Public

Public file در React یک پوشه خاص است که حاوی فایل های ثابت مانند تصاویر، فایل های CSS و فایل های جاوا اسکریپت است. این فایل ها توسط Webpack پردازش نمی شوند، بنابراین با بقیه برنامه شما همراه نیستند. در عوض، زمانی که برنامه شما ساخته می شود، مستقیماً در فهرست خروجی کپی می شوند.

چند دلیل وجود دارد که ممکن است بخواهید از Public file در React استفاده کنید. اول، به شما این امکان را می دهد که فایل های استاتیک خود را جدا از کد برنامه خود نگه دارید. این می تواند خواندن و نگهداری کد شما را آسان تر کند و همچنین می تواند برنامه شما را سریع تر کند، زیرا Webpack نیازی به پردازش این فایل ها ندارد.

دوم، Public file به شما امکان می دهد فایل های استاتیک خود را روی سروری متفاوت از کد برنامه خود میزبانی کنید. اگر می خواهید از یک CDN برای ارائه تصاویر یا فایل های CSS خود استفاده کنید، یا اگر می خواهید برنامه خود را روی سروری میزبانی کنید که فضای ذخیره سازی کافی برای فایل های استاتیک شما ندارد، می تواند مفید باشد.

برای استفاده از Public file در React، کافی است یک پوشه به نام public در ریشه پروژه خود ایجاد کنید. سپس، می توانید فایل های استاتیک خود را به این پوشه اضافه کنید. هنگامی که برنامه خود را می سازید، این فایل ها در دایرکتوری خروجی کپی می شوند.

می توانید با استفاده از ویژگی `src` به فایل های استاتیک در پوشه عمومی از اجزای `React` خود ارجاع دهید. به عنوان مثال، کد زیر یک تصویر را از پوشه عمومی بارگذاری می کند:

```

```

اگر می خواهید به فایلی از پوشه عمومی در قالب `HTML` خود ارجاع دهید، می توانید از متغیر محیطی `PUBLIC_URL` استفاده کنید. این متغیر توسط `Create React App` در طول فرآیند ساخت تنظیم می شود و حاوی مسیر مطلق به پوشه عمومی است. به عنوان مثال، الگوی `HTML` زیر یک تصویر را از پوشه عمومی بارگذاری می کند:

```
<img src={process.env.PUBLIC_URL + '/images/my-image.png'} />
```

متغیر `process.env.PUBLIC_URL` یک متغیر ویژه است که حاوی اطلاعات خاص محیط است. در این مورد، حاوی مسیر مطلق به پوشه عمومی است. عملگر `+` برای الحاق متغیر `PUBLIC_URL` با مسیر فایل تصویر استفاده می شود.

پوشه عمومی ابزار قدرتمندی است که می تواند به شما کمک کند برنامه های `React` خود را منظم و کارآمد نگه دارید. با استفاده از پوشه عمومی، می توانید فایل های استاتیک خود را جدا از کد برنامه خود نگه دارید و می توانید فایل های استاتیک خود را در سروری متفاوت از کد

برنامه خود میزبانی کنید. این می تواند خواندن، نگهداری و استقرار  
برنامه های شما را آسان تر کند.



## فولدر Src

پوشه src در React قلب برنامه شماست. این شامل همه جاوا اسکریپت، CSS و فایل های تصویری است که برنامه شما را تشکیل می دهند. پوشه src توسط Webpack پردازش می شود، که همه فایل های شما را با هم در یک فایل جاوا اسکریپت که می تواند توسط مرورگر بارگیری شود، جمع می کند.

چند نکته در مورد پوشه src در React وجود دارد:

فقط فایل های داخل پوشه src می توانند به اجزای React شما وارد شوند.

پوشه src بخشی از سیستم ماژول است. این بدان معناست که می توانید فایل ها را از پوشه های دیگر به اجزای React خود وارد کنید.

پوشه src در حین ساخت و توسعه برنامه شما استفاده می شود.

در اینجا برخی از مهم ترین فایل هایی که در پوشه src پیدا خواهید کرد، آورده شده است:

**index.js:** این نقطه ورود برای برنامه شما است. این اولین فایلی است که توسط مرورگر بارگذاری می شود و مسئول بوت استرپ برنامه شماست.

**App.js:** این جزء اصلی برنامه شما است. این مؤلفه ای است که در سطح بالای برنامه شما ارائه می شود و مسئولیت مدیریت وضعیت و منطق برنامه شما را بر عهده دارد.

Index.css: این فایل شامل سبک های CSS برای برنامه شما می باشد.

images: این پوشه حاوی تمام فایل های تصویری برنامه شما است.

علاوه بر این فایل ها، می توانید فایل های دیگری مانند فایل های آزمایشی، مستندات و اجزای سفارشی را نیز در پوشه src بیابید.

پوشه src مهمترین پوشه در برنامه React شما است. اینجا جایی است که همه کدها و دارایی هایی را که برنامه شما را تشکیل می دهند ذخیره می کنید. با مرتب نگه داشتن کد خود در پوشه src، می توانید خواندن، نگهداری و استقرار برنامه خود را آسان تر کنید.

## بیکربندی فولدر component در فولدر src

### فایل LinearGradient.jsx

این کد یک جزء React است که یک خط گرادیان افقی را ارائه می دهد. کامپوننت یک پایه تکی به نام عرض می گیرد که عرض خط را مشخص می کند. مقدار پیش فرض عرض w-full است، به این معنی که خط تمام عرض ظرف خود را پر می کند.

کامپوننت از کلاس Name prop برای اعمال کلاس bg-gradient-rainblue CSS به خط استفاده می کند. این کلاس از شیب خطی استفاده می کند ویژگی CSS برای ایجاد یک گرادیان که از آبی در بالا به آبی روشن در پایین تغییر می کند.

این کامپوننت با استفاده از کلمه کلیدی export ارسال می شود. این بدان معنی است که کامپوننت را می توان به اجزای دیگر وارد کرد و برای ارائه یک خط گرادیان افقی استفاده کرد.



تصویر بالا نتیجه استفاده از این کامپوننت رت به ما نشان می دهد.

## فایل SocialMediaIcons.jsx

این کد یک کامپوننت React است که ردیفی از نمادهای رسانه های اجتماعی را ارائه می دهد. این کامپوننت یک عنصر div را با کلاس های CSS زیر ارائه می کند:

flex: این کلاس عنصر div را به یک ظرف انعطاف پذیر تبدیل می کند.

justify-center: این کلاس محتویات عنصر div را به صورت افقی متمرکز می کند.

md:justify-start: این کلاس تراز عنصر div را برای شروع در صفحه های متوسط و بزرگتر تغییر می دهد.

my-10: این کلاس 10 پیکسل حاشیه به بالا و پایین عنصر div اضافه می کند.

gap-7: این کلاس 7 پیکسل فاصله بین فرزندان عنصر div اضافه می کند.

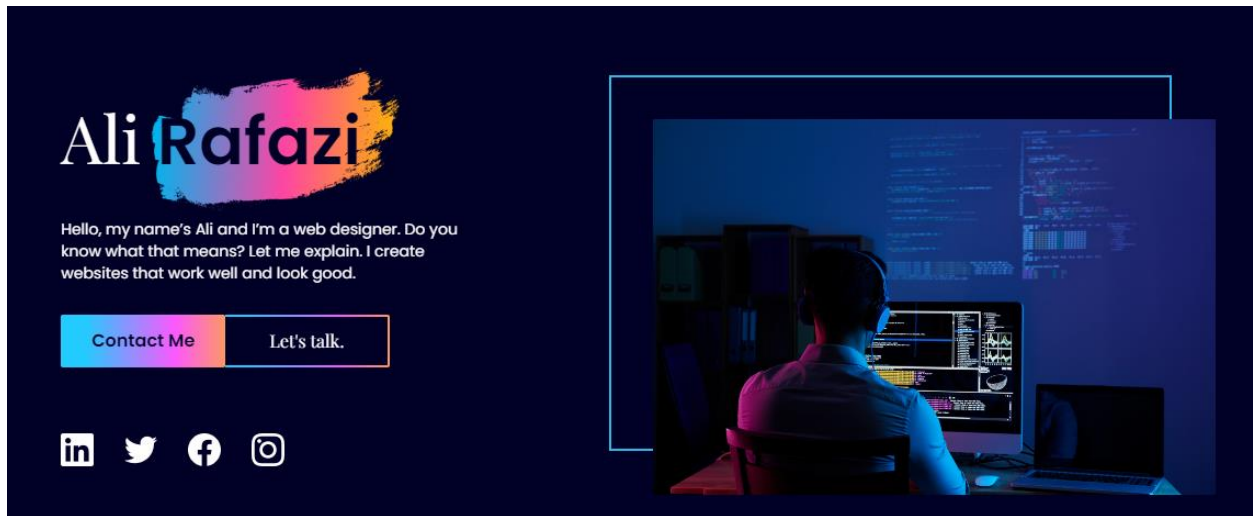


تصویر بالا نتیجه استفاده از این کامپوننت رت به ما نشان می دهد.

بیکربندی فولدر scenes در فولدر src

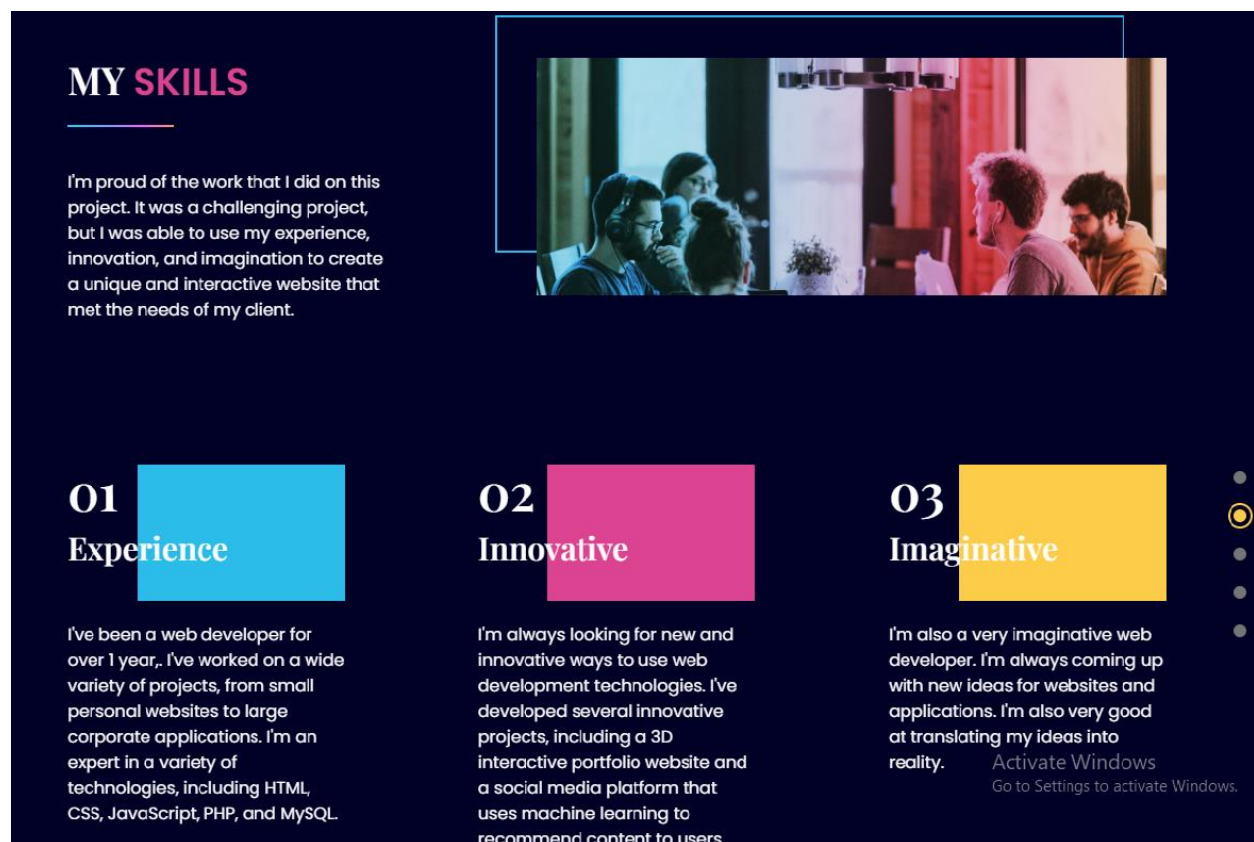
فایل landing.jsx

این کامپوننت شامل تغییرات در قسمت اول سایت میباشد مانند تصویر زیر :



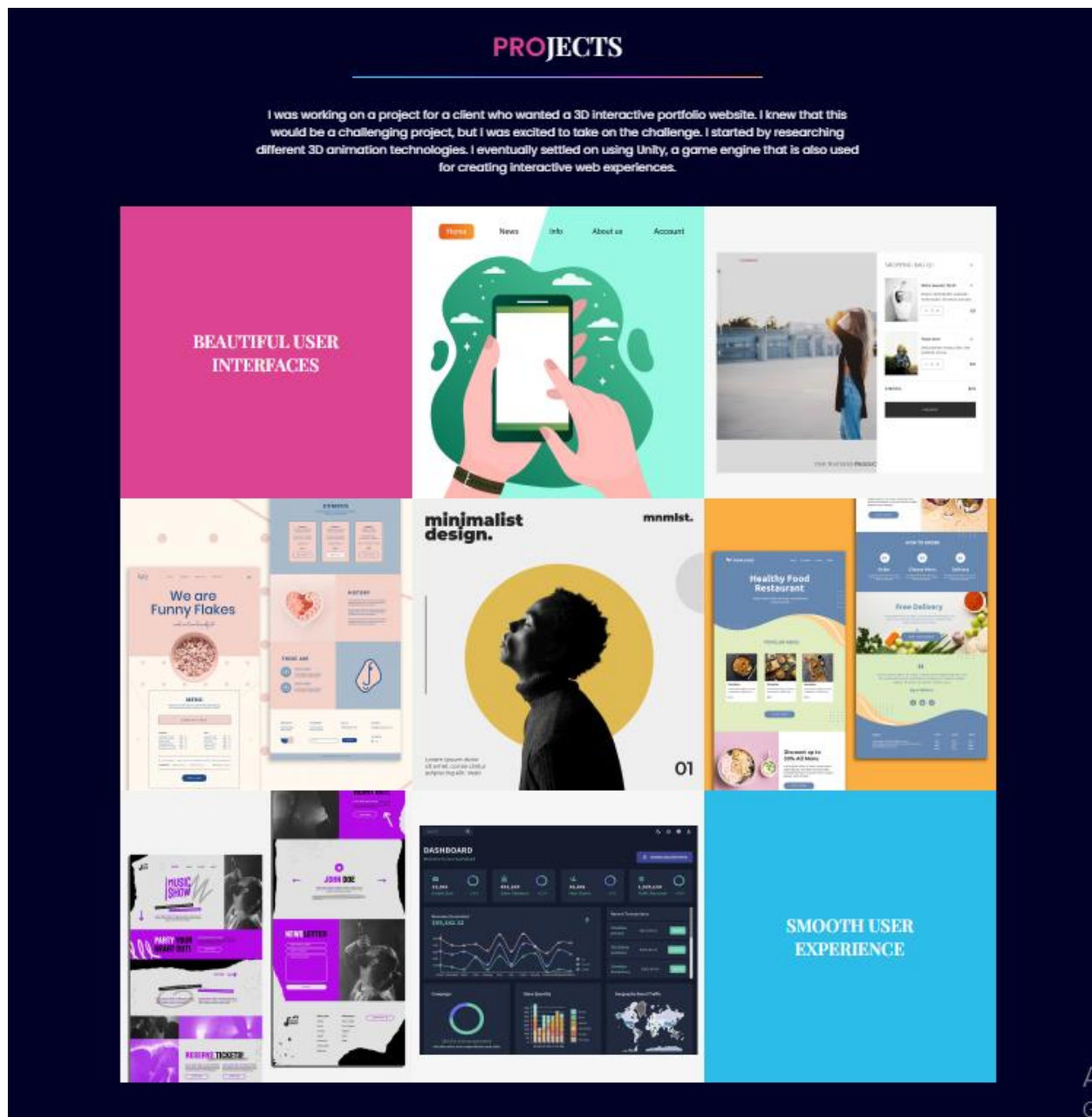
## فایل MySkills.jsx

این کامپوننت شامل تغییرات در قسمت دوم سایت میباشد مانند تصویر زیر:



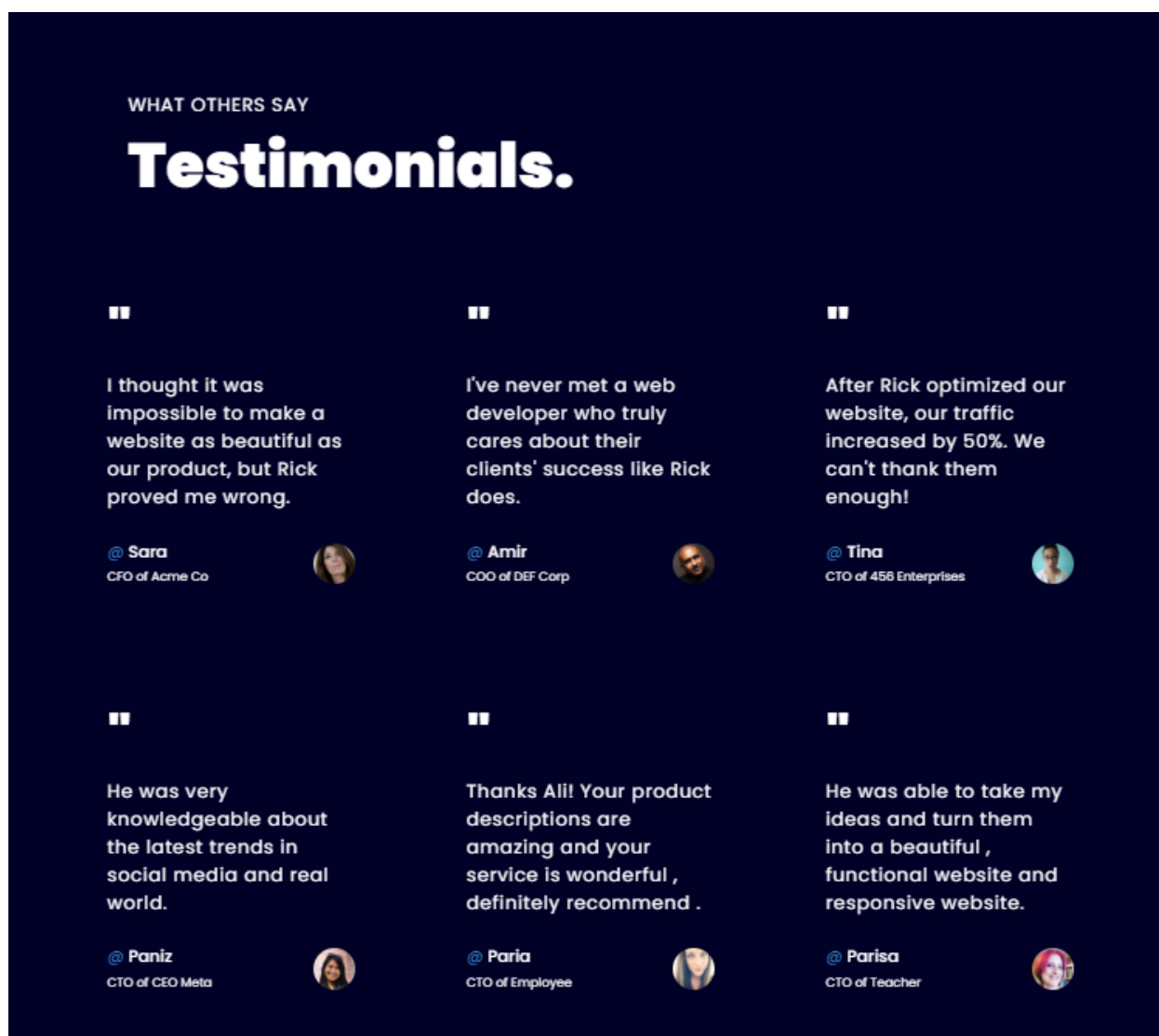
## فایل Projects.jsx

این کامپوننت شامل تغییرات در قسمت سوم سایت میباشد مانند تصویر زیر



## فایل Feedbacks.jsx

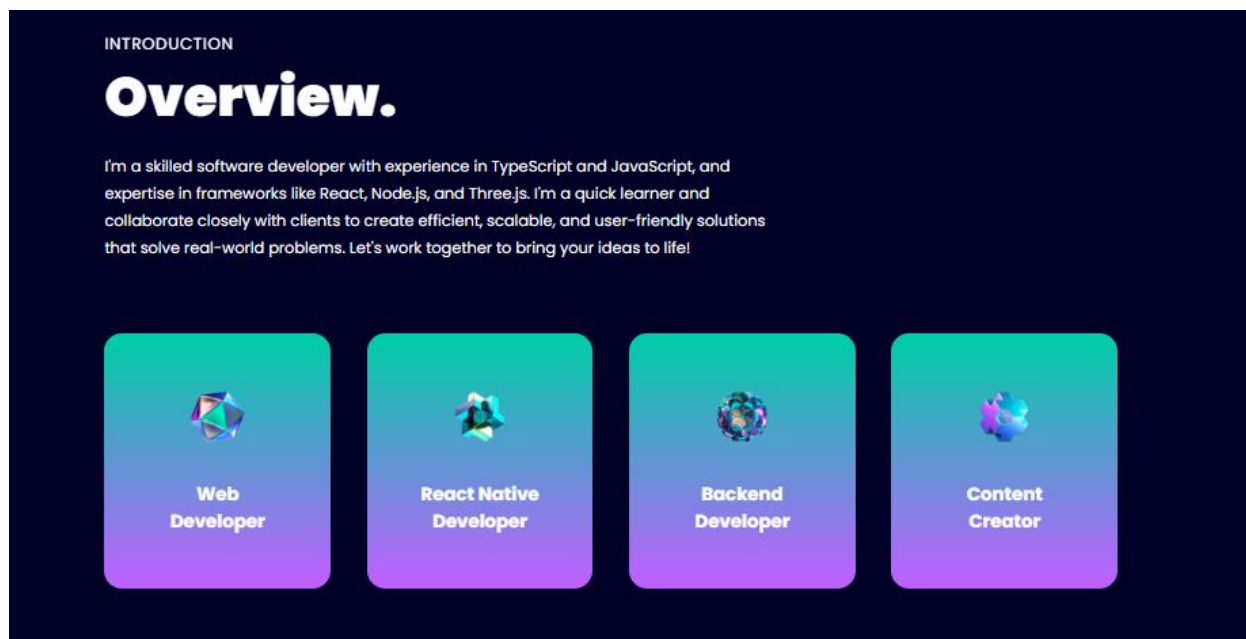
این کامپوننت شامل تغییرات در قسمت چهارم سایت میباشد مانند تصویر زیر ( توجا شود در این قسمت میتوانستیم از کامپوننت Testimonials.jsx نیز استفاده کنیم ولی از کامپوننت دیگری که به یک api برای تصاویر متصل است )





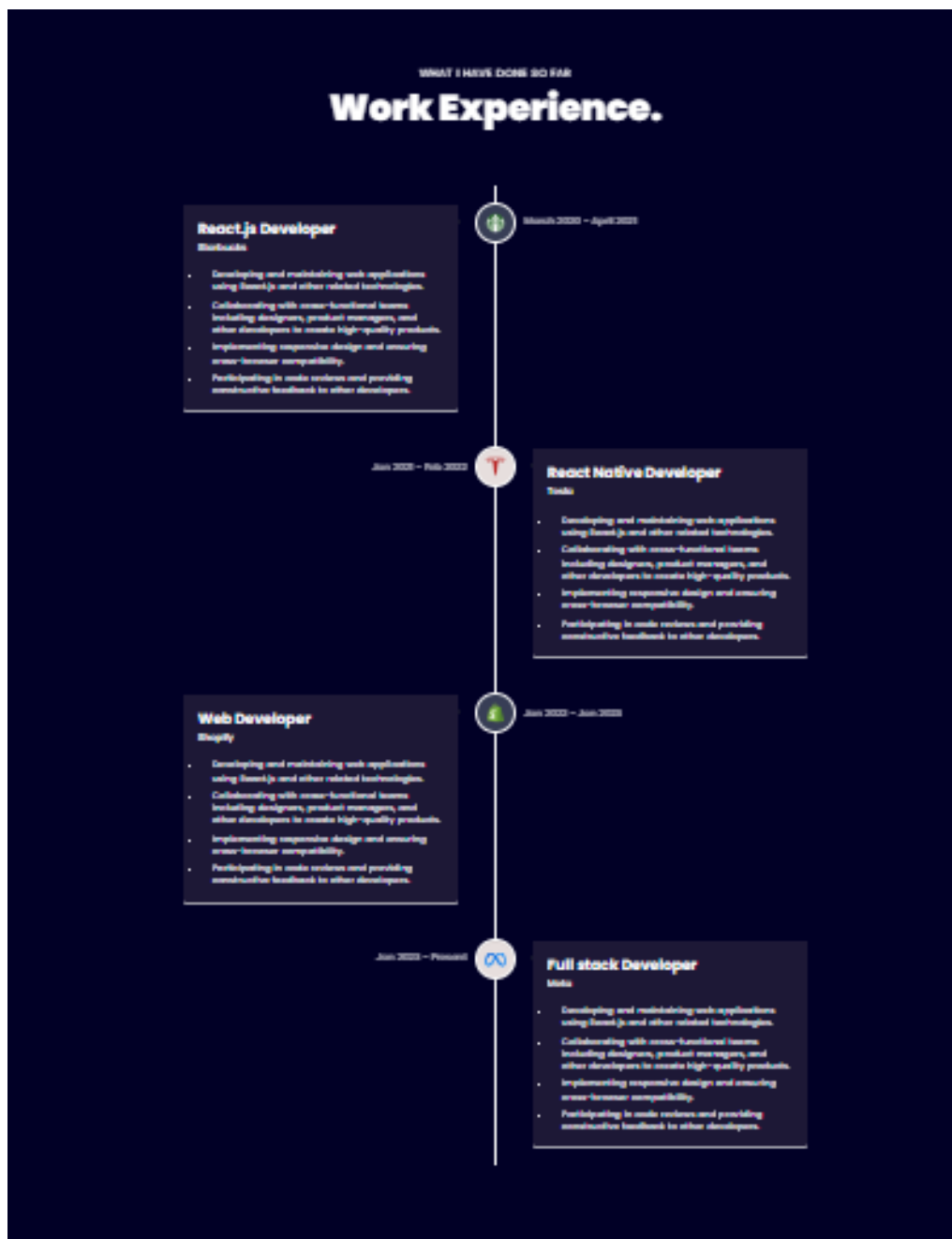
## فایل About.jsx

این کامپوننت شامل تغییرات در قسمت بنجم سایت میباشد مانند تصویر زیر



## فایل Experience.jsx


این کامپوننت شامل تغییرات در قسمت ششم سایت میباشد مانند تصویر زیر



## فایل Contact.jsx

این کامپوننت شامل تغییرات در قسمت هفتم سایت می باشد مانند تصویر زیر

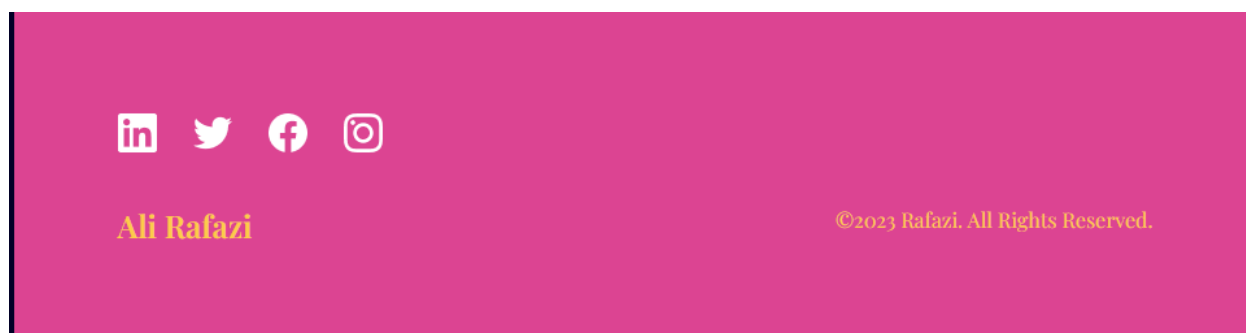
### CONTACT ME TO GET STARTED



SEND ME A MESSAGE

## فایل Footer.jsx

این کامپوننت شامل تغییرات در قسمت آخر سایت میباشد مانند تصویر زیر



## فایل اصلی App.js

این کد جزء اصلی برنامه React است. این مؤلفه یک تغییر حالت تاریک، یک نوار پیمایش، یک صفحه فرود، یک صفحه مهارت ها، یک صفحه پروژه ها، یک صفحه بازخوردها، یک صفحه درباره، یک صفحه تجربه، یک صفحه تماس و یک پاورقی را ارائه می دهد.

این مؤلفه از قلاب `useState` برای ذخیره صفحه فعلی و از قلاب `useMediaQuery` برای رندر کردن عناصر مختلف بسته به عرض پنجره مرورگر به صورت مشروط استفاده می کند. این مؤلفه همچنین از کتابخانه حرکتی برای متحرک سازی ظاهر عناصر صفحه استفاده می کند.

در اینجا یک تفکیک کد است:

عبارات `import` برای بارگذاری ماژول هایی که جزء به آن نیاز دارد استفاده می شود.

قلاب `useState` برای ذخیره صفحه فعلی استفاده می شود.

قلاب `useMediaQuery` برای ارائه شرطی عناصر مختلف بسته به عرض پنجره مرورگر استفاده می شود.

کتابخانه حرکتی برای متحرک سازی ظاهر عناصر صفحه استفاده می شود.

جزء `Navbar` یک نوار ناوبری را ارائه می دهد.

مؤلفه `Landing` یک صفحه فرود را ارائه می دهد.

مؤلفه MySkills یک صفحه مهارت را ارائه می دهد.  
جزء Projects یک صفحه پروژه را ارائه می دهد.  
مؤلفه Feedbacks یک صفحه بازخورد ارائه می کند.  
جزء About یک صفحه درباره را ارائه می دهد.  
مؤلفه Experience یک صفحه تجربه را ارائه می کند.  
مؤلفه Conntact یک صفحه تماس را ارائه می دهد.  
مؤلفه Footer یک پاورقی را ارائه می دهد.