

Packages/Tools cần cài đặt:

Nodejs (Node + NPM)

Git

MySQL

Cài đặt

Chạy local

Lấy code mới nhất từ github về từ nhánh mysql

```
git clone https://github.com/hwebz/pickcel-react
```

Cài đặt các packages cần thiết

```
npm install
```

Build các file PUG, SCSS, JS thành HTML, CSS, JS từ thư mục frontend/ ra thư mục /dist

```
npm run fe:prod
```

Build code server Nodejs và client React

```
npm run build
```

Build tất cả chạy lệnh (chạy lệnh này khi deploy lên server production)

```
npm run deploy
```

Build code server Nodejs và client React + chạy localhost:3000 trên máy local

```
npm run dev
```

Chạy trên server

Yêu cầu server:

Hệ điều hành: Ubuntu, CentOS, Fedora, ... (nên dùng Ubuntu)

RAM: tối thiểu 1G

SSD: 20GB

Ở đây server dùng công nghệ Nodejs (Express) để tạo RESTful API sử dụng MySQL làm database (nhánh master sử dụng PostgreSQL)

Dùng Ubuntu

Cài Nodejs:

<https://tecadmin.net/install-latest-nodejs-npm-on-ubuntu/>

Cài Nginx:

<https://www.digitalocean.com/community/tutorials/how-to-install-nginx-on-ubuntu-18-04>

cấu hình nginx: /etc/nginx/sites-available/default

```
server {
```

```
    root /var/www/html;
```

```
    index index.html index.htm index.nginx-debian.html;
```

```
    server_name your_domain.com www.your_domain.com;
```

```
    location / {
```

```

proxy_pass http://localhost:3000;
proxy_http_version 1.1;
proxy_set_header Upgrade $http_upgrade;
proxy_set_header Connection 'upgrade';
proxy_set_header Host $host;
proxy_cache_bypass $http_upgrade;
}

listen [::]:443 ssl http2 ipv6only=on; # managed by Certbot
listen 443 ssl http2; # managed by Certbot

ssl_certificate /etc/letsencrypt/live/your_domain.com/fullchain.pem; # managed by
Certbot
ssl_certificate_key /etc/letsencrypt/live/your_domain.com/privkey.pem; # managed by
Certbot
# include /etc/letsencrypt/options-ssl-nginx.conf; # managed by Certbot
ssl_ciphers
EECDH+CHACHA20:EECDH+AES128:RSA+AES128:EECDH+AES256:RSA+AES256:EEC
DH+3DES:RSA+3DES:!MD5;
ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem; # managed by Certbot
}

server {
    if ($host = www.your_domain.com) {
        return 301 https://$host$request_uri;
    } # managed by Certbot

    if ($host = your_domain.com) {
        return 301 https://$host$request_uri;
    } # managed by Certbot

    listen 80 default_server;
    listen [::]:80 default_server;

    server_name your_domain.com www.your_domain.com;
    return 404; # managed by Certbot
}

```

Cài đặt MySQL:

<https://www.digitalocean.com/community/tutorials/how-to-install-mysql-on-ubuntu-18-04>

Deploy Nodejs + React app (pm2 start server.generated.js):

<https://www.digitalocean.com/community/tutorials/how-to-set-up-a-node-js-application-for-prDUCTION-on-ubuntu-18-04>

Cài đặt HTTPS bằng LetsEncrypt:

<https://www.digitalocean.com/community/tutorials/how-to-secure-nginx-with-let-s-encrypt-on-ubuntu-18-04>

Cài đặt HTTP/2:

<https://www.digitalocean.com/community/tutorials/how-to-set-up-nginx-with-http-2-support-on-ubuntu-18-04>

Cấu hình

```
const config = {
  baseUrl: 'http://localhost:3000',
  env: process.env.NODE_ENV || 'development',
  port: process.env.PORT || 3000,
  database: {
    host: process.env.DB_HOST || process.env.DB_STAGING_HOST || 'localhost',
    user: process.env.DB_USER || process.env.DB_STAGING_USER || 'root',
    password: process.env.DB_PASSWORD || process.env.DB_STAGING_PASSWORD || 'Admin@123',
    database: process.env.DB_NAME || process.env.DB_STAGING_NAME || 'pickcel',
    port: 5432,
    ssl: false
  },
  mail: {
    host: process.env.MAIL_HOST || process.env.MAIL_STAGING_HOST || 'smtp.gmail.com',
    port: process.env.MAIL_PORT || process.env.MAIL_STAGING_PORT || 587,
    user: process.env.MAIL_USER || process.env.MAIL_STAGING_USER || 'hadm.hau@gmail.com',
    password: process.env.MAIL_PASSWORD || process.env.MAIL_STAGING_PASSWORD || 'ldxscotxawbtdnut'
  },
  oauth: {
    client_id: '984706706429-21fb1fppdf8ts60gdkapuvfblrbvk48p.apps.googleusercontent.com', // http:
    // client_id: '984706706429-m0p9jg6s0tuh4fnp8rd00tu9gmntmkce.apps.googleusercontent.com' // htt
  },
  timezone: 'Asia/Ho_Chi_Minh',
  secret: 'JokeFromProDevPickcel',
  reset_password_secret: 'JokeFromProDevPickcelResetPassword',
  // token_expiry: '30m',
  token_expiry: '10h',
  reset_password_token_expiry: '1h',
  googleUserPassword: 'vrev*&{#!@&$(%)$@sdffsfs',
  access_token_key: 'nekoTssecca',
  x_site_token_key: 'yeKsseccAx',
  fe_date_format: 'DD MMM, YYYY, hh:mm A',
  naming_date_format: 'DD MMM-YY HH:mm:ss',
  page_size: 5,
  template_temp_name: 'enosihntrofemanetaltmeton',
  default_composition_id: '00000000-0000-0000-0000-000000000000'
}

export default config
```

Cần thay đổi các cấu hình sau trong file config/config.js:

baseUrl: thay đổi theo URL được deploy project (VD: <https://vnpt.com>, <https://quangcao.com>, ...)

port: Chỉnh port cho server (hiện tại mặc định đang là 3000)

database: config MySQL server host, username, password, dbname, ... (với MySQL thì không cần quan tâm đến port: 5432 và ssl cứ giữ nguyên hoặc remove đi cũng được)

mail: config mail để send email (sau này có thể sử dụng email dạng doanh nghiệp như: support@vnpt.com, ...)

oauth: client_id cho phép đăng nhập = google (lấy mã này trong Google Console mục Credentials API)

timezone: timezone của website

secret: secret key phục vụ cho việc mã hóa password

reset_password_secret: secret key phục vụ cho việc mã hóa của password mới khi thay password

googleUserPassword: password mặc định cho những tài khoản sử dụng chức năng "đăng nhập = Google" để đăng ký tài khoản

access_token_key: key name để lưu access token khi user đăng nhập thành công và dùng access token này để gửi request đến API (nếu không có access_token hợp lệ thì sẽ bị logout ra ngoài và không thể request đến tất cả các đầu API)

x_site_token_key: tương tự như access_token_key nhưng là để lưu email của người dùng đã đăng nhập (bắt buộc để có thể request được đến API)

fe_date_format: format cho datetime hiển thị bên ngoài giao diện

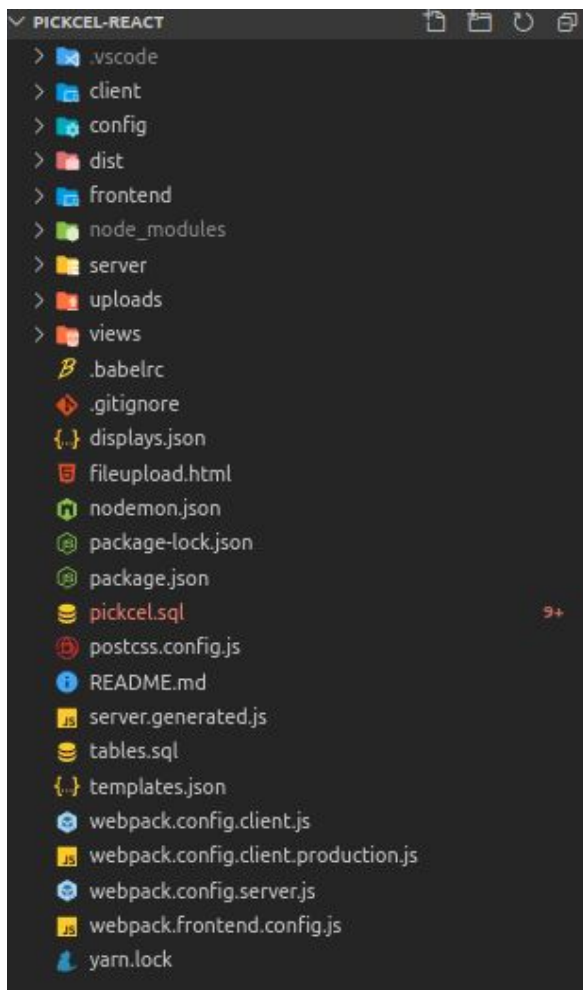
naming_date_format: format for datetime hiển thị dạng full ngày bên ngoài giao diện

page_size: số lượng items/page sử dụng cho việc phân trang

template_temp_name: tên mặc định cho template tạm thời khi sử dụng chức năng Edit Composition/Edit Template

default_composition_id: composition mặc định sử dụng khi add display mới

Cấu trúc các file / thư mục chính



client: chứa code React để viết UI cho ứng dụng Pickcel

config: chứa các file js phục vụ cho việc config ứng dụng

dist: chứa các file html, css, js đã được built sử dụng lệnh npm run fe:prod

frontend: chứa các file pug, scss, js dùng để dựng phần frontend cho site (có thể xóa đi nếu cần vì React không dùng đến nữa nhưng nên giữ lại để sau có thay đổi html, css, js gì còn build lại được)

node_modules: các packages được cài đặt sử dụng lệnh npm install

server: chứa các file js phục vụ cho việc viết RESTful API server

views: chứa 1 file preview.ejs (file template để phục vụ cho việc preview composition) và tempPreview.html (file html template phục vụ cho preview composition khi đang create hoặc edit composition)

uploads: thư mục dùng cho việc upload assets (assets sẽ được upload vào thư mục này và sau đó sẽ tự động bị xóa để không

chiếm bộ nhớ trên server vì file đã được lưu trên Google Storage CDN)

.babelrc: file config babel

.gitignore: khai báo những file/folder không cần đưa lên git repo

displays.json: dummy data dữ liệu về các displays

fileupload.html: file demo cho chức năng upload (có thể delete)

nodemon.json: file config cho server nodemon chạy trên local khi làm việc với localhost

package.json: chứa các khai báo cần thiết cho dự án nodejs (các packages cần cài đặt, các lệnh cần chạy, thông tin project, ...)

pickcel.sql: file SQL để import vào MySQL khi lần đầu cài đặt (có chứa các bảng và dữ liệu cần thiết ban đầu để chạy được website)

postcss.config.js: file để config postcss khi build SCSS ra CSS

README.md: file markdown để ghi chú về thông tin dự án

server.generated.js: là file được sinh ra khi chạy lệnh npm run build (file này là file chính dùng để chạy server RESTful API)

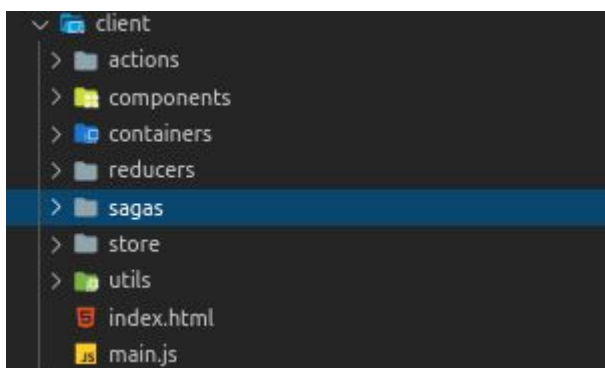
tables.sql: file SQL tạm để lưu các khai báo bảng, dữ liệu cho MySQL (chỉ cần sử dụng file pickcel.sql bên trên là được)

templates.json: Chứa các template có sẵn (có thể xóa đi vì đã lưu template vào bảng templates trong database khi import pickcel.sql rồi)

webpack.config.*.js: các file config cho webpack để build dự án (cho client, server, frontend)

yarn.lock, package-lock.json: file tự sinh bởi yarn và npm không cần quan tâm.

Cấu trúc các file trong client/



actions: chứa các file khai báo actions cho Redux

components: chứa các components của React

containers: chứa các file khai báo mapStateToProps và mapDispatchToProps của Redux store đến Props của component tương ứng để gọi trích xuất dữ liệu trong component

reducers: chứa các file khai báo reducers khi state thay đổi tùy theo side effects của

Redux store

sagas: chứa các file khai báo Redux-sagas để quản lý side-effects khi làm việc với RESTful API từ phía server

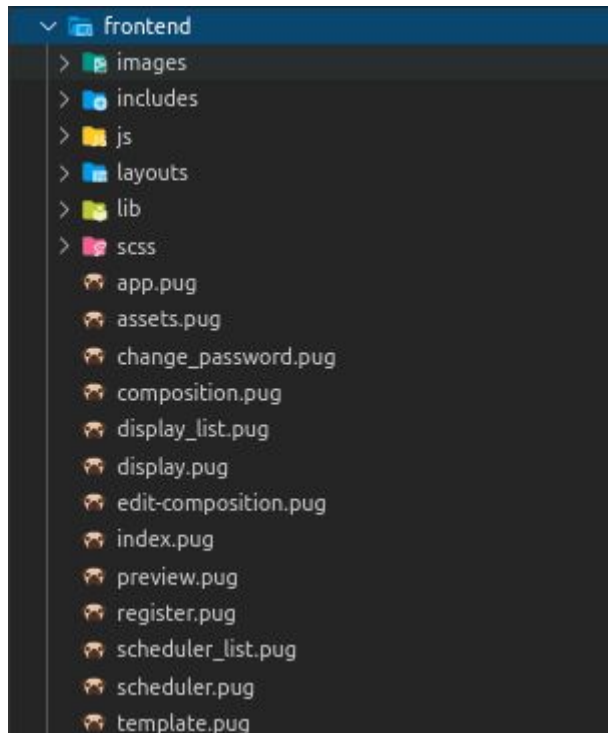
store: chứa file khai báo store cho Redux

utils: chứa các file gồm các hàm tiện ích phục vụ trong các components

index.html: file chính để chạy ứng dụng React trên web

main.js: file chính để khai báo thư viện React và các thành phần liên quan như: Redux, Router, ...

Cấu trúc các file trong frontend/



images: chứa các file ảnh JPG, PNG, JPEG, ...

includes: chứa các file pug như các modals, header, footer phục vụ cho việc reusable (include cho các trang cần)

js: chứa file javascript cần cho các trang front-end (VD: datepicker, sortable, zone, calendar, ...)

layouts: chứa layouts cho các trang có cùng chung layout

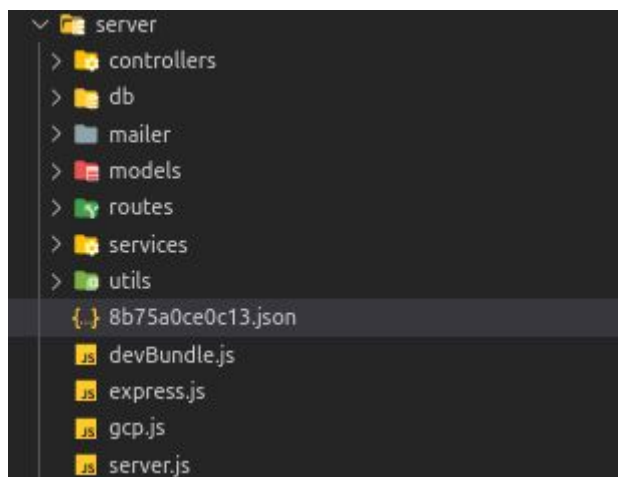
scss: thư mục chứa file SCSS và sẽ build thành CSS sau này

***.pug:** là các trang HTML được viết dưới dạng pug, sau sẽ được built thành HTML

Sử dụng lệnh sau để build PUG, SCSS, ... ra thư mục dist/

`npm run fe:prod`

Cấu trúc các file trong server/



controllers: chứa các file controllers

db: chứa các file khai báo các câu lệnh SQL để thao tác với MySQL từ Nodejs

mailer: chứa file để gửi mail = nodejs

routes: chứa các file routes của RESTful API

services: chứa các file có những xử lý chính với database để lấy ra/thêm/sửa/xóa dữ liệu với MySQL

utils: chứa các file gồm có các hàm tiện ích dùng cho các file trong server/

8b75a0ce0c13.json: là file tự sinh ra bởi Google Cloud Storage CDN (lưu trữ ảnh, video được upload).

Cần đăng ký Google Storage CDN và tải về key mới và update file gcp.js

devBundle.js: xử lý build React code

express.js: chứa khai báo express server

gcp.js: file config để sử dụng Google Cloud Storage CDN

server.js: file chính để khởi tạo server