مدل های دیتابیس سامانه

**دسترسی ها (AccsessModel) :**

در این مدل دیتابیس که فقط دریافت آن ها ممکن هست تنها 3 فیلد وجود دارد ، title,description,group که دسترسی های کاربران رو مشخص میکنه و در نصب اولیه به صورت خودکار در سامانه ساخته میشود

**سمت سازمانی (ApplicantModel):**

سمت ها سازمانی یا همان گیرنده در قسمت افزودن پرونده میباشد که جنبه دسته بندی کردن دارد و موقع سرچ کردن پرونده ها این فیلد میتواند در دسته بندی کمک کند ، در handler این مدل شامل insert,delete,get میباشد

**تنظیمات برنامه (AppSettingModel):**

این بخش شامل اطلاعات ایمیل جهت ارسال اسناد یا پرونده به کاربران خارج از سامانه میباشد همچنین میتوان با استفاده از فیلد isSupervisorActive تعیین کنید که برای سامانه ناظر فعال باشد یا خیر ، همچنین فیلد version نسخه ی فعلی برنامه را نمایش میدهد که در بخش فوتر سایت نیز نمایش داده میشود ، فیلد logo هم شامل اسم فایل لوگو سامانه میباشد که بدون نیاز به گاردی کاربران میتوانند به فایل های پ.شه config دسترسی پیدا کنند ، پس لوگویی که کاربر آپلود میکند و به صورت مستقیم وارد فیلد logo تنظیمات برنامه میشود در پوشه config وارد میشود ، دقت کنید آپلود لوگو یکroute جداگانه دارد که هم آپلود را انجم میدهد و هم ثبت آن در دیتابیس، اگر هم سامانه فاقد لوگو باشد لوگوی لیلو به صورت پیشفرض در بخش فرانت نمایش داده میشود

**بایگانی (ArchiveModel) :**

در موقع ساخت نرم افزار یک بایگانی به صورت پیشفرض ساخته میشود که قابلیت حذف ندارد ولی ویرایش میشود

Watermark این فیلد یک متن برای واتر مارک برای اسناد زیر مجموع این بایگانی میباشد

maxFileSize : حداکثر حجم مجاز برای آپلود سند در این بایگانی میباشد و به کیلوبایت میباشد

availablePattern : هر پرونده دارای یک شماره پرونده میباشد که ما پنج عدد پترن موجود داریم

پترن شماره یک : شامل یک حرف قبل از شماره میباشد که مدیر سامانه در بخش تنظیمات بایگانی وارد میکند و همچنین یک فیلد که کاربر موقع ثبت پرونده وارد میکند مثال : f-**541263**

پترن شماره دو : باز هم یک فیلد که مدیر وارد میکند و شامل یک تاریخ که توسط سامانه ساخته میشود سپس جلوی آن یک عدد که توسط کاربر موقع ثبت پرونده نوشته میشود مثال : f-**14001025741525**

پترن شماره سه : دقیقا مثل پترن شماره دو با این تفاوت که بخش سوم توسط خود سامانه ساخته میشود مثال : f-**14001025001**

پترن شماره جهار : کامل توسط سامانه تولید میشود که شبیه به پرت سه میشود خروجی ولی دیگر فیلد اول شماره را مدیر واردنمیکند

پترن شماره پنج : کامل توسط کاربر که میخواهد پرونده ایجاد کند ساخته میشود

firstStringOfPattern : همون فیلدی که مدیر اول شماره در تنظیمات بایگانی وارد میکند

illegalFormats : فرمت های ممنوعه که کاربر حق افزودن اون فرمت هارو در پروند های زیر مجموع اون بایگانی را ندارد، فرمت ها بدون نقظه وارد شود

**درخت بایگانی (ArchiveTreeModel)**

درخت بایگانی قفسه های زیر مجموع بایگانی میباشد که شامل یک روت اصلی میباشد با فیلد isMain مشخص میشود ، قفسه های دارای این isMain = true یک سری تنظیمات دارند که شامل فعال سازی روکش پرونده برای قفسه های زیر مجموع و همچنین انتخاب فرم روکش پرونده برای قفسه های زیر مجموع ، همچنین یک ارتباط با مدل خودش دارد که mainParent نام دارد ، برای تو در تو کردن درخت، همچنین مسیر این درخت به صورت String در فیلد route ذخیره میشود و همچنین اگر درخت روت اصلی باشد میتواند form مربوط به خودش را داشته باشد یعنی اگر isMain برقرار باشد میتواند form مخصوص خودش را داشته باشد پس 2 فیلد form و isFormRequired زمانی میتواند برقرارباشد که isMain = true باشد

**سند (DocumentModel)**

هر سند متعلق به یک پرونده میباشد که اسناد میتوانند از هر نوعی باشند (عکس ، ویدیو ، اکسل و...) ولی جزو فرمت های ممنوعه نمیتوانند باشد(در قسمت بایگانی تعریف شده اند)

اسناد میتوانند ورژن داشته باشند و version =1 به صورت پیشفرض میباشد ، اگر isMain=false باشد یعنی سند جزو سند های فرعی میباشد (ورژن 2 به بالا) . اسناد میتوانند یک سری یادداشت داشته باشند که در لیست notes نگهداری میشوند و در زمان نمایش سند یادداشت ها قابل مشاهده میباشند

نکات مهم:

2 فیلد خیلی مهم وجود دارد یکی uniqueFileId و دیگری uniquePdfFileId

زمانی که کاربر درخواست فایل سند را ارسال میکند شما با استفاده از این 2 فیلد میتوانید به اسناد دسترسی پیدا کنید . مثلا سند شماره یک یک تصویر میباشد که در زمان ثبت هم فایل اصلی به صورت کد شده ذخیره میشود و هم به صورت pdf شما با استفاده از 2 فیلد بالا میتواند روی اسناد پوشه uploads کوئری بزنید و فایل کد شده ی آن را دیکد کنید. uniqueFileId فایل اصلی و uniquePdfFileIdفایل pdf شده میباشد

اگر سند تصویر باشد 2 فیلد imageWidth و imageHeight برای ui ذخیره میشوند و موقع ارسال اطلاعات سند در ui تصاویر بر این مبنا نمایش داده میشوند

فیلد inShelve هم برای کازیو مورد استفاده قرار میگیرد :

اسنادی که در کازیو هستند فاقد fileId هستند .

**ایمیل (EmailModel)**

ایمیل برای اشتراک گذاری پرونده یا بخشی از پرونده کاربرد دارد . کاربری که دسترسی به این بخش دارد میتواند پرونده را به کاربران خارج از سامانه ارسال نماید

این مدل شامل فهرستی از اسناد و یا کل پرونده و همچنین تاریخ انقضا میباشد

نحوه برخورد با ایمیل:

وقتی کاربر اطلاعات را جهت ارسال ایمیل را ثبت کرد ، در سامانه ابتدا یک یوزرنیم و پسورد به صورت اتوماتیک ساخته میشود که در مدل EmailsUserSchema قابل مشاهده میباشد و سپس این یوزرنیم و پسورد برای آن ایمیل ها ارسال میشوند و سپس کاربران با استفاده از آن یوزرنیم و پسورد وارد بخش ایمیل سامانه میشوند و به محض ورود میتوانند اسناد را مشاهده کنند با این قابلیت مدیر بخش ایمیل میتواند بعد از ارسال ایمیل ویرایش هایی از قبیل تاریخ انقضا و همچنین اسناد ارسال شده را ویرایش کنند

**هشدار پرونده (FileAlertModel)**

هر پرونده میتوانند هشدار هایی داشته باشد که کاربران در حین مشاهده آن ها را مشاهده کنند و از آخرین یادداشت ها احتمالی با خبر شوند ، هشدار برای تمام کاربرانی که به آن بایگانی و آن پرونده دسترسی دارد ارسال میشود (به صورت نوتیفیکیشن در بالای سامانه) و این یادداشت ها میتوانند تاریخ انقضا نیز داشته باشند

**پرونده (FileModel)**

بخش اصلی سامانه همین مدل میباشد اطلاعات اولیه که شامل :

عنوان پرونده – تاریخ آن – وضعیت پرونده – نوع پرونده – مخاطبین(اختیاری) – درخواست کننده (applicantModel) (اختیاری) و کلید واژه که اختیاری میباشد

پس از آن کاربر میتواند اسنادی را برای آن پرونده آپلود کند و یا همچنین میتواند روکش پرونده را در صورت وجود تکمیل نماید

نکته مهم:

اگر در سامانه تایید ناظر فعال باشد و نیاز باشد که ناظر بایگانی ، پرونده های بایگانی را تایید نمایش 3 فیلد isReject, isConfirm,isWaiting مهم میشوند – اگر isConfirm برابر با false باشد پرونده قابل جست و جو نیست- همچنین بجر ادمین کسی نمیتواند آن پرونده را مشاهده کند ولی اگر ناظر غیر فعال باشد در سامانه isConfirm به صورت پیشفرض true میشود

مکاتبات ایجاد کننده و ناظر:

وقتی کاربر پرونده را جهت بررسی به ناظر ارسال میکند ، ناظر پرونده را کامل مشاهده میکند و دسترسی کامل به آن پرونده را پیدا میکند ، اگر از دید ناظر پرونده موردی نداشت میتواند آن را تایید کند و غیر این صورت میتواند آن پرونده را رد کند و دلیل رد آن را نیز درج کند ، پس از آن کاربر ایجاد کننده میتواند دلیل رد را مطالعه کند و ایرادات را برطرف کنید سپس مجدد اقدام به ارسال آن کند و این موضوع تا زمانی که ناظر تاییدد کند ادامه خواهد داشت و پس از تایید پرونده در حالت isConfirm در خواهد آمد و قابل مشاهده برای کسانی که به آن بایگانی دسترسی لازم را دارد میشود

نکته ی مهم در مورد زمان ارسال ایجاد کننده به ناظر:

ممکن است چند ناظر برای یک بایگانی ایجاد شده باشد و کاربر میتواند تعیین کند که برای چه ناظری آن پرونده جهت تایید و بررسی ارسال شود

**فرم (FormModel)**

سامانه دارای یک بخش بنام فرم ساز میباشد که مدیریت و یا کاربری که به آن بخش دسترسی دارد میتواند فرم های بر اساس نیاز بایگانی تعریف کند که فیلد های قابل ساخت شامل input text , check box , radio button, selector , date picker , Persian date picker , text area میباشد که هر کدام میتواند شامل ویژگی اجباری بودن و placeholder باشد و همچنین label همچنین برای input نیز یک سری پترن هایی از قبیل email , mellicode , number , unique input , phone number وارد کرد

که در صورت انتخاب هر کدام ولیدیشن مربوط به آن بررسی میشود . نکته در مورد unique input :

این نوع ورودی نمیتواند تکراری باشد مثلا 2 پرونده به نام پرونده شماره یک و پرونده شماره دو که از این روکش یا همان فرم پیروی میکنند نمیتوانند فیلدی که از نوع unique میباشد تکراری وارد شود . روکش فرم برای قفسه مادر (isMain = true) و یا یک پرونده خاص قابل تنظیم میباشد یسعنی تمام پرونده های زیر مجوع درخت مادر از روکش آن پیروی میکند مگر در هنگان وارد کردن اطلاعات پرونده برای ثبت ، کاربر آن را تغییر دهد. نکته مهم: اگر کاربر روکش پرونده را بعد از ثبت پرونده پر کند و ثبت کند و بعد از آن مدیر روکش را ویرایش کند ، روکش آن پرونده (که اطلاعات ثبت شده است) بدون تغییر میماند ولی اگر کاربر اطلاعات روکش را وارد نکرده باشد با تغییر روکش توسط مدیر ، روکش آن پرونده نیز بروزمیشود

**اشخاص حقوقی . حقیقی (PersonModel , LegalPersonModel)**

فهرستی از اشخاص حقوقی و حقیقی در سامانه ثبت میشود که میتوانند جزو مخاطبین یک پرونده باشند مثلا 2 نفر جرو اشخاص حقوقی هستند و در یک پرونده دخالت داشته اند ، در زمان ثبت پرونده آن اشخاص در پرونده نیز ثبت میشوند که در سرچ و گزارش گیری پرونده ها راحت بتوان پرونده را بر اساس این اشخاص جستجو کرد.

**قرض دادن پرونده (LendModel)**

کاربر اگر به این بخش دسترسی داشته باشد میتواند پرونده را به کاربران دیگر در سامانه قرض دهد ، و آن کاربر میتواند پرونده را فقط مشاهده کند ولی هیچ عملیاتی روی آن نمیتواند انجام دهد.

این قرض دادن دارای تاریخ انقضا میباشد تا کاربر در یک بازه زمانی مشخص بتواند پرونده را مشاهده کند

**قفسه های کازیو (LibraryShelfModel)**

همه کاربران دارای یک بخش به نام کازیو میباشند که میتوانند اسنادی را وارد سامانه کنند که فاقد هر گونه پرونده ای مباشند به عنوان مثال ما یک سند داریم که هنوز پرونده ای برای آن ثبت نشده و یا در سیستم فقط میخواهیم بارگذاری کنیم بعد از آن کاربر میتواند آن را در صورتی که به پرونده ای دسترسی داشته باشد وارد آن بخش کند . پس کازیو اسنادی هست که فاقد پرونده میباشند . حال مدلی تحت عنوان قفسه داریم که کاربر میتواند در کازیو پوشه هایی درست کند و اسناد را در پوشه بریزد . این پوشه میتواند مجموعه ای از اسناد باشد که به صورت embedded در دیتابیس ذخیره میشود

**تاریخچه تغییرات (LogModel)**

تاریخچه تغییرات سامانه در این بخش ذخیره میشود . این تاریخچه میتواند فقط مخصوص یک پرونده خاص باشد در این صورت fileId پر میشود

**نقش های و کاربران (RoleModel , UserModel)**

هر نقش شامل فهرستی از دسترسی های میباشد که در فایل ضمیمه دسترسی ها قرار گذاشته شده است. و این نقش دارای عنوان میباشد . سپس مدیر در هنگام افزودن کاربر به سامانه مشخص میکند که چه کاربرانی چه نقشی برای چه بایگانی دارد . پس کاربر دارای فهرستی از نقش ها و بایگانی ها دارد. مثلا کاربر یک دارای دسترسی مدیر کل بایگانی 2 را دارد ولی دسترسی ناظر را برای بایگانی شماره 3 دارد.

ساختار و معماری پروژه:

این پروژه دارای معماری MVC میباشد و مدل و کنترلر و routes دارد که در قسمت routes شما میتوانید end-point ها و route های هر بخش را مشاهده کنید که تفکیک شده و منظم قابل مشاهده هستند و هر route دارای کامنت میباشد که ورودی و خروجی و دسترسی را مشخص کرده است

در بخش authUtilities ابزار های اهراز هویت قرار دارد که Auth.js دارای 2 تابع میباشد firstGuard برای چک کردن توکن میباشد و roleGuard چک میکند که کاربر دسترسی به آن بخش از بایگانی را دارد یا خیر

پوشه config

این پوشه دارای asset های عمومی میباشد که برای عموم کاربران باز هستند مانند تصویر پروفایل و لوگو سامانه

Dbconfig

برای کانکت شدن به دیتابیس استفاده میشود

Handler

شامل تمامیه منطق های پروژه میباشد و از دیزاین پترن CQRS استفاده شده یعنی command و query برای سهولت کار جدا شده

سیستم لاگ گیری :

از سیستم لاگ گیری Winston برای این پروژه استفاده شده که تمامیه خطا های سامانه در دیتابیس توسط این سیستم ذخیره میشود

Uploads

تمامیه اسناد پرونده ها به صورت کد گذاری شده در این قسمت ذخیره میشوند

فهرست وابستگی ها:

Server:

"@pdftron/pdfnet-node": "^9.2.0",

"adm-zip": "^0.5.9",

"archiver": "^5.3.0",

"archiver-zip-encryptable": "^1.0.10",

"bcrypt": "5.0.1",

"cors": "^2.8.5",

"dotenv": "^12.0.3",

"express": "^4.17.1",

"express-async-errors": "^3.1.1",

"file-convert": "^1.1.3",

"generate-password": "^1.7.0",

"generate-unique-id": "^2.0.1",

"glob": "^7.2.0",

"imagemagick": "^0.1.3",

"jalali-moment": "^3.3.10",

"javascript-obfuscator": "^4.0.0",

"joi": "^17.4.0",

"joi-objectid": "^3.0.1",

"joi-password": "^3.0.1",

"joi-password-complexity": "^5.1.0",

"jsbarcode": "^3.11.5",

"jsonwebtoken": "^8.5.1",

"jszip": "^3.7.1",

"libreoffice-convert": "^1.3.5",

"lodash": "^4.17.21",

"mime": "^3.0.0",

"mongoose": "^5.12.14",

"mongoose-autopopulate": "^0.16.0",

"multer": "^1.4.2",

"mv": "^2.1.1",

"node-file-encrypt": "^0.0.6",

"node-schedule": "^2.1.0",

"node-tesseract-ocr": "^2.2.1",

"nodemailer": "^6.7.2",

"nodemailer-smtp-transport": "^2.7.4",

"pdf-lib": "^1.17.1",

"pdfkit": "^0.13.0",

"process": "^0.11.10",

"purejswatermark": "^0.0.7",

"qrcode": "^1.5.0",

"read-excel-file": "^5.3.4",

"request-image-size": "^2.2.0",

"socket.io": "^4.4.1",

"unzipper": "^0.10.11",

"uuid": "^3.4.0",

"winston": "^3.3.3",

"winston-mongodb": "^5.0.5",

"xlsx": "^0.18.5",

"xoauth2": "^1.2.0",.

فهرست وابستگی های Web:

"@agjs/react-right-click-menu": "^0.0.8",

"@babel/core": "^7.17.5",

"@babel/preset-env": "^7.16.11",

"@babel/runtime": "^7.17.2",

"@ckeditor/ckeditor5-basic-styles": "^31.1.0",

"@ckeditor/ckeditor5-build-classic": "^31.1.0",

"@ckeditor/ckeditor5-essentials": "^31.1.0",

"@ckeditor/ckeditor5-paragraph": "^31.1.0",

"@ckeditor/ckeditor5-react": "^3.0.3",

"@emotion/react": "^11.7.1",

"@emotion/styled": "^11.6.0",

"@mui/material": "^5.4.0",

"@pdftron/pdfjs-express-viewer": "^8.4.0",

"@testing-library/jest-dom": "^5.11.4",

"@testing-library/react": "^11.1.0",

"@testing-library/user-event": "^12.1.10",

"@toast-ui/react-image-editor": "^3.15.2",

"axios": "^0.21.1",

"bootstrap-autocomplete": "^2.3.7",

"bulma": "^0.9.3",

"chart.js": "^3.7.1",

"dateformat": "^4.5.1",

"draft-js": "^0.11.7",

"echarts": "^5.3.0",

"echarts-for-react": "^3.0.2",

"file-saver": "^2.0.5",

"filerobot-image-editor": "^3.12.17",

"generate-password": "^1.7.0",

"imrc-datetime-picker": "^2.1.3",

"jalali-moment": "^3.3.10",

"jquery": "^3.6.0",

"js-file-download": "^0.4.12",

"jsonwebtoken": "^8.5.1",

"jwt-decode": "^3.1.2",

"list-react-files": "^0.2.0",

"lodash": "^4.17.21",

"material-design-icons": "^3.0.1",

"moment": "^2.29.2",

"moment-jalaali": "^0.9.2",

"moment-timezone": "^0.5.33",

"pdf-lib": "^1.17.1",

"pdf-viewer-reactjs": "^2.2.3",

"pdfjs-dist": "2.13.216",

"prop-types": "^15.8.1",

"react": "^17.0.2",

"react-app-polyfill": "^2.0.0",

"react-auto-translate": "^0.0.6",

"react-autocomplete": "^1.8.1",

"react-autocomplete-hint": "^1.3.0",

"react-avatar-editor": "^12.0.0",

"react-canvas-draw": "1.1.1",

"react-chartjs-2": "^4.0.1",

"react-contextmenu": "^2.14.0",

"react-cookie": "^4.1.1",

"react-datepicker": "^4.6.0",

"react-dev-utils": "^11.0.3",

"react-doc-viewer": "^0.1.5",

"react-dom": "^17.0.2",

"react-draft-wysiwyg": "^1.14.7",

"react-file-drop": "^3.1.3",

"react-file-previewer": "^0.6.3",

"react-fullscreen-loading": "^0.0.4",

"react-image-crop": "^9.1.1",

"react-pdf": "^5.7.1",

"react-pdf-watermark": "^1.0.6",

"react-persian-datepicker": "^3.0.2",

"react-player": "^2.9.0",

"react-popper-tooltip": "^4.3.1",

"react-preloaders": "^3.0.3",

"react-quill": "^1.3.5",

"react-redux": "^7.2.4",

"react-redux-loading": "^4.6.1",

"react-redux-loading-bar": "^5.0.0",

"react-refresh": "^0.8.3",

"react-responsive-pagination": "^1.4.1",

"react-ripples": "^2.2.1",

"react-router": "^5.2.0",

"react-router-dom": "^5.2.0",

"react-scripts": "3.4.1",

"react-select": "^5.2.2",

"react-simple-image-slider": "^2.3.0",

"react-slideshow-image": "^3.5.0",

"react-sticky": "^6.0.3",

"react-toastify": "^7.0.3",

"react-tooltip": "^4.2.21",

"react-translator-component": "^2.0.0",

"reactjs-pdf-reader": "^1.0.12",

"redux": "^4.1.0",

"redux-thunk": "^2.3.0",

"resolve": "1.18.1",

"resolve-url-loader": "^3.1.2",

"sass-loader": "^10.0.5",

"semver": "7.3.2",

"simple-react-validator": "^1.6.0",

"uuid": "^8.3.2"