<template>

  <div class="app-container">

    <div class="filter-container">

      <el-input

        v-model="listQuery.val"

        placeholder="搜索内容"

        style="width: 200px"

        class="filter-item"

        @keyup.enter.native="handleFilter"

      />

      <el-select

        style="margin: 0 10px;top: -4px;"

        v-model="listQuery.status"

        placeholder="请选择"

        @change="getList"

      >

        <el-option

          v-for="item in typeList"

          :key="item.value"

          :label="item.label"

          :value="item.value"

        >

        </el-option>

      </el-select>

      <el-date-picker

          style="margin: 0 10px;top: -4px;"

          v-model="listQuery.date"

          type="daterange"

          align="right"

          unlink-panels

          range-separator="~"

          start-placeholder="开始日期"

          end-placeholder="结束日期"

          value-format="yyyy-MM-dd"

          @change="handleFilter"

          >

      </el-date-picker>

      <el-button

        v-waves

        class="filter-item"

        type="primary"

        icon="el-icon-search"

        @click="handleFilter"

      >

        搜索

      </el-button>

      <el-button

        v-waves

        :loading="downloadLoading"

        class="filter-item"

        type="primary"

        icon="el-icon-download"

        @click="handleDownload"

      >

        导出

      </el-button>

    </div>

    <el-table

      :key="tableKey"

      v-loading="listLoading"

      :data="tableData"

      border

      fit

      highlight-current-row

      style="width: 100%"

      @sort-change="sortChange"

    >

      <el-table-column

        :label="$t('table.id')"

        prop="id"

        sortable="custom"

        align="center"

        width="80"

        :class-name="getSortClass('id')"

      >

        <template slot-scope="{ $index }">

          <span>{{ $index + 1 + (listQuery.limit\*(listQuery.page-1)) }}</span>

        </template>

      </el-table-column>

      <el-table-column

        label=""

        align="center"

      >

        <template slot-scope="{ row }">

          <el-image

          style="width: 60px; height: 60px"

          :src="editSrc(row.image)"

          fit="cover"></el-image>

        </template>

      </el-table-column>

      <el-table-column label="报案人" align="center">

        <template slot-scope="{ row }">

          <span>{{ row.name }}</span>

        </template>

      </el-table-column>

      <el-table-column

        label="所在网格"

        align="center"

      >

        <template slot-scope="{ row }">

          <span>{{ row.reseau }}</span>

        </template>

      </el-table-column>

      <el-table-column

        label="待救人员"

        align="center"

      >

        <template slot-scope="{ row }">

          <span>{{ row.num }}人</span>

        </template>

      </el-table-column>

      <el-table-column

        label="描述"

        align="center"

      >

        <template slot-scope="{ row }">

          <el-popover

            placement="top-start"

            title=""

            width="200"

            trigger="hover"

            :content="row.describes">

            <span slot="reference" style="white-space: nowrap;

    text-overflow: ellipsis;

    overflow: hidden;

    word-break: break-all;">{{ row.describes }}</span>

          </el-popover>

        </template>

      </el-table-column>

      <el-table-column

        label="电话"

        align="center"

      >

        <template slot-scope="{ row }">

          <span>{{ row.phone }}</span>

        </template>

      </el-table-column>

      <el-table-column

        label="报案地址"

        align="center"

        prop="address"

        sortable="custom"

      >

        <template slot-scope="{ row }">

          <el-link :underline="false" type="primary" @click="lookAddress(row)">{{ row.address }}</el-link>

        </template>

      </el-table-column>

      <el-table-column

        label="报案时间"

        align="center"

        prop="add\_time"

        sortable="custom"

      >

        <template slot-scope="{ row }">

          <span>{{ row.add\_time }}</span>

        </template>

      </el-table-column>

      <el-table-column

        :label="$t('table.status')"

        class-name="status-col"

        width="100"

        prop="status"

        sortable="custom"

      >

        <template slot-scope="{ row }">

          <el-tag :type="row.status | statusFilter">

            {{ row.status\_text}}

          </el-tag>

        </template>

      </el-table-column>

      <el-table-column

        :label="$t('table.actions')"

        align="center"

        class-name="small-padding fixed-width"

      >

        <template slot-scope="{ $index }">

          <el-button

            type="text"

            icon="el-icon-view"

            @click="handleEdit($index, 'look')"

            >救援记录</el-button

          >

        </template>

      </el-table-column>

    </el-table>

    <pagination

      v-show="total > 0"

      :page-count.sync="total"

      :page.sync="listQuery.page"

      :limit.sync="listQuery.limit"

      @pagination="getList"

    />

    <el-dialog

      title=""

      :visible.sync="editVisible"

      width="55%"

      center

      :destroy-on-close="true"

      :modal-append-to-body="false"

    >

      <el-timeline v-if="recordList.length>0">

        <el-timeline-item :timestamp="v.add\_time" placement="top" v-for="(v,i) in recordList" :key="i+'\_item'">

        <el-collapse accordion>

          <el-collapse-item :name="i">

              <template slot="title">

                <i class="header-icon el-icon-user-solid" v-if="v.type=='user'" style="margin-right: 10px;color:#F56C6C;"></i>

                <i v-else class="header-icon el-icon-s-custom" style="margin-right: 10px;color:#409EFF;"></i>

                <el-tag effect="plain" :type="v.type=='user'?'danger':''" style="margin-right: 10px;">{{ v.name }}</el-tag>

                <el-tag effect="plain" type="success">{{ v.phone }}</el-tag>

              </template>

              <div>

                <el-image

                  v-for="(vv,ii) in v.image"

                  :key="ii+'\_image'"

                  style="width: 100px; height: 100px;margin-right:10px;"

                  :src="editSrc(vv)"

                  :preview-src-list="editSrc(v.image)">

                </el-image>

              </div>

              <div>

                <el-image v-for="(vv,ii) in v.video" :key="ii+'\_v'" src="//file.paidn.cn/video\_play.png" style="width: 100px; height: 100px;margin-top: 10px;margin-right:10px;" @click="previewVideo(vv)"></el-image>

              </div>

              <div v-if="v.describes">

                <el-divider>描述</el-divider>

                {{v.describes}}

              </div>

          </el-collapse-item>

        </el-collapse>

        </el-timeline-item>

     </el-timeline>

     <el-empty v-else description="暂无救援记录"></el-empty>

    </el-dialog>

    <el-dialog

      title="查看"

      :visible.sync="dialogVisibleAddress"

      width="30%"

      :modal="false"

      center

      z-index="9999999999999"

      style="z-index: 999999999"

      :modal-append-to-body="false"

    >

      <iframe

        id="Address"

        width="100%"

        height="550"

        frameborder="0"

        :src="'https://apis.map.qq.com/tools/poimarker?marker=coord:'+mapData['lat']+','+mapData['lng']+';title:'+mapData['address']+';addr:'+mapData['address']+mapUrl"

      ></iframe>

    </el-dialog>

    <el-dialog title :visible.sync="dialogPlay" width="40%" @close="closeDialog" :modal-append-to-body="false">

         <video :src="videoUrl" controls autoplay class="video" style="max-height: 400px;" :ref="dialogVideo"

         width="100%"></video>

      </el-dialog>

  </div>

</template>

<script>

import {

  fetchList,

  fetchPv,

  createArticle,

  updateArticle,

} from "@/api/article";

import waves from "@/directive/waves"; // waves directive

import { parseTime } from "@/utils";

import Pagination from "@/components/Pagination"; // secondary package based on el-pagination

import Tinymce from "@/components/Tinymce";

import base from "../../../public/js/base64";

import { wgs84togcj02, gcj02towgs84 } from "../../../public/js/jcg02to84";

import $ from "jquery" //在需要使用的页面中

const calendarTypeOptions = [

  { key: "CN", display\_name: "China" },

  { key: "US", display\_name: "USA" },

  { key: "JP", display\_name: "Japan" },

  { key: "EU", display\_name: "Eurozone" },

];

// arr to obj, such as { CN : "China", US : "USA" }

const calendarTypeKeyValue = calendarTypeOptions.reduce((acc, cur) => {

  acc[cur.key] = cur.display\_name;

  return acc;

}, {});

export default {

  name: "ComplexTable",

  components: { Pagination, Tinymce },

  directives: { waves },

  filters: {

    statusFilter(status) {

      const statusMap = {

        2: "success",

        1: "info",

        0: "danger",

      };

      return statusMap[status];

    },

    typeFilter(type) {

      return calendarTypeKeyValue[type];

    },

  },

  data() {

    return {

      tableKey: 0,

      list: null,

      total: 0,

      listLoading: true,

      listQuery: {

        page: 1,

        limit: 20,

        type: "",

        status:-1,

        importance: undefined,

        val: undefined,

        sort: "-status",

        date:[],

      },

      mapData:{

          lat:"",

          lng:"",

          address:""

      },

      editData: {

        token: "",

        title: "",

        images: "",

        open\_time: "",

        phone: "",

        stars: 1,

        type: "",

        cate: "",

        label: "",

        lat: "",

        lng: "",

        address: "",

        details: "",

      },

      editVisible: false,

      dialogPlay:false,

      roleList: [],

      tableData: [],

      url: "",

      details: "",

      downloadLoading: false,

      dialogVisibleAddress: false,

      addressZ: true,

      images: [],

      mapUrl:

        "&type=0&key=4ZIBZ-5IPC2-URKUQ-CL63R-QLHD6-JOBQU&referer=小程序",

      typeList: [{label:"全部",value:-1},{label:"待救援",value:0},{label:"救援中",value:1},{label:"已完成",value:2}],

      cateList: [],

      labelList: [],

      recordList:[],

      videoUrl:"",

      dialogVideo:"",

    };

  },

  created() {

    this.url = this.URL;

    this.getList();

  },

  methods: {

    editSrc(value) {

      var that = this;

      if (!value) {

        return;

      }

      if (typeof value == "string") {

        if (

          ("\_\_" + value).indexOf("http://") > 0 ||

          ("\_\_" + value).indexOf("https://") > 0

        ) {

          return value;

        }else if(("\_\_" + value).indexOf("image/") > 0){

          return that.URL + "upload/" + value;

        } else {

           var url = that.URL + "upload/image/" + value;

          if(!that.verUrl(url)){

            return that.URL + "upload/" + value;

          }else{

            return url;

          }

        }

      } else {

        var array = [];

        value.forEach(function (v, i) {

          array.push(that.editSrc(v));

        });

        return array;

      }

    },

    verUrl(url){

      var status = false;

      $.ajax({

          url:url,

          async:false,

          dataType : "jsonp",//数据类型为jsonp

          jsonp : "jsonpCallback",//服务端用于接收callback调用的function名的参数

          statusCode: {

              200: function() {

                status = true;

              }

          },

      });

      return status;

    },

    previewVideo(url){

      this.dialogPlay = true;

      this.videoUrl = this.editSrc(url);

    },

    closeDialog() {

        this.videoUrl = ""; //清空数据 关闭视频播放

    },

    lookAddress(row){

        this.mapData = row;

        this.dialogVisibleAddress=true;

    },

    editLatLng(lat, lng, type) {

      if (type == "84") {

        return gcj02towgs84(lng, lat);

      } else {

        return wgs84togcj02(lng, lat);

      }

    },

    getList() {

      var that = this;

      that.listLoading = true;

      this.postData("rescue/list", this.listQuery)

        .then((res) => {

          that.tableData = res.data.list;

          that.total = res.data.pageCount;

          setTimeout(() => {

            that.listLoading = false;

          }, 0.8\*1000);

        })

        .catch(() => {

          that.listLoading = false;

        });

    },

    getRecord(token){

      console.log(token);

      var that = this;

      this.postData("rescue/record", {token:token})

        .then((res) => {

          console.log(res);

          that.recordList = res.data;

        })

    },

    // 编辑操作

    handleEdit(index, type) {

      var that = this;

      that.details = "";

      if (type == "edit") {

        if (index >= 0) {

          this.editData.token = this.tableData[index].token;

          this.postData("circum/info", {

            token: this.tableData[index].token,

          }).then((res) => {

            if (res.code == 0) {

              that.editData = res.data;

              that.images = res.data.images;

              that.getHtml(res.data.details).then((res2) => {

                console.log(res2);

                if (res2.code == 0) {

                  that.details = base.Base64.decode(res2.html);

                }

              });

            }

          });

        } else {

          this.editData = {

            token: "",

            title: "",

            images: "",

            open\_time: "",

            phone: "",

            stars: 1,

            type: "",

            cate: "",

            label: "",

            lat: "",

            lng: "",

            address: "",

            details: this.$md5(new Date().getTime() + "\_competition") + ".log",

          };

          this.details = "";

        }

        this.editVisible = true;

      } else if (type == "status") {

        this.edit(

          this.tableData[index].token,

          this.tableData[index].status == 1 ? "stop" : "start"

        );

      } else if (type == "del") {

        this.$confirm("确定要删除吗？", "提示", {

          type: "warning",

        })

          .then(() => {

            this.edit(this.tableData[index].token, "del");

          })

          .catch(() => {});

      }else if(type=="look"){

        this.getRecord(this.tableData[index].token);

        this.editVisible = true;

      }

    },

    edit(token, type) {

      this.postData("circum/operate", { type: type, list: [token] }).then(

        (res) => {

          if (res.code == 0) {

            this.$message.success(res.msg);

            this.getList();

          } else {

            this.$message.error(res.msg);

          }

        }

      );

    },

    // 保存编辑

    saveEdit() {

      var that = this;

      var images = "",

        cate = "",

        label = "";

      var datas = JSON.parse(JSON.stringify(this.editData));

      that.images.forEach((v) => {

        if (!images) {

          images = v;

        } else {

          images += "," + v;

        }

      });

      datas.images = images;

      this.editData.cate.forEach((v) => {

        if (!cate) {

          cate = v;

        } else {

          cate += "," + v;

        }

      });

      datas.cate = cate;

      this.editData.label.forEach((v) => {

        if (!label) {

          label = v;

        } else {

          label += "," + v;

        }

      });

      datas.label = label;

      datas.type = this.listQuery.type;

      this.pushHtml(

        this.editData.details,

        base.Base64.encode(this.details)

      ).then((res) => {

        if (res.code == 0) {

          that.$message.success(res.message);

          that.postData("circum/edit", datas).then((res) => {

            if (res.code == 0) {

              that.$message.success(res.msg);

              that.editVisible = false;

              that.getList();

              that.editData = {

                token: "",

                title: "",

                images: "",

                open\_time: "",

                phone: "",

                stars: 1,

                type: "",

                cate: "",

                label: "",

                lat: "",

                lng: "",

                address: "",

                details: "",

              };

              that.details = "";

            } else {

              that.$message.error(res.msg);

            }

          });

        } else {

          that.$message.error(res.message);

        }

      });

    },

    addressFun() {

      var that = this;

      that.dialogVisibleAddress = true;

      if (that.addressZ) {

        setTimeout(function () {

          var iframe = document.getElementById("Address").contentWindow;

          window.addEventListener(

            "message",

            function (e) {

              if (e.data.poiaddress) {

                that.editData["address"] = e.data.poiaddress;

                var location = e.data.latlng;

                if (location) {

                  var locat = that.editLatLng(

                    location["lat"],

                    location["lng"],

                    "84"

                  );

                  that.editData["lng"] = locat[0];

                  that.editData["lat"] = locat[1];

                  that.$parent.$message({

                    message: "选择地址：" + e.data.poiaddress,

                    type: "success",

                  });

                }

              }

            },

            false

          );

        }, 500);

        that.addressZ = false;

      }

    },

    handleFilter() {

      this.listQuery.page = 1;

      this.getList();

    },

    sortChange(data) {

      const { prop, order } = data;

      if (prop) {

        this.sortByID(order, prop);

      }

    },

    sortByID(order, prop) {

      if (order === "ascending") {

        this.listQuery.sort = "+" + prop;

      } else {

        this.listQuery.sort = "-" + prop;

      }

      this.handleFilter();

    },

    resetTemp() {

      this.temp = {

        id: undefined,

        importance: 1,

        remark: "",

        timestamp: new Date(),

        title: "",

        status: "published",

        type: "",

      };

    },

    handleDownload() {

      this.downloadLoading = true;

      import("@/vendor/Export2Excel").then((excel) => {

        const tHeader = ["所属网格","待救人员","待救人数","电话","状态","发起地址","参与救援人数", "发起时间"];

        const filterVal = ["reseau","name","num","phone","status\_text","address","count","add\_time"];

        const data = this.formatJson(filterVal);

        excel.export\_json\_to\_excel({

          header: tHeader,

          data,

          filename: "table-list",

        });

        this.downloadLoading = false;

      });

    },

    formatJson(filterVal) {

      return this.tableData.map((v) =>

        filterVal.map((j) => {

          if (j === "timestamp") {

            return parseTime(v[j]);

          } else {

            return v[j];

          }

        })

      );

    },

    getSortClass: function (key) {

      const sort = this.listQuery.sort;

      return sort === `+${key}` ? "ascending" : "descending";

    },

    handleAvatarSuccess(data) {

      if (data.file) {

        this.images.push(data.file);

        this.$parent.$message.success("上传成功");

      } else {

        this.$parent.$message({

          message: data.msg,

          type: "error",

        });

      }

    },

    /\*\*

     \* 上传前类型判断

     \*/

    beforeAvatarUpload(file) {

      return true;

    },

  },

};

</script>