一、库、框架的选型（~/package.json）

1.构建、打包工具Webpack 4.x  
sass-loader/babel-loader/file-loader等loaders，web-webpack-plugin/mini-css-extract-plugin等plugins，建议模块化的代码编写、组织方式，对打包后的代码做些压缩、分割、按需加载等优化，webpack本身的优化特性可以活用，社区、github上有大量插件可选。  
2.业务使用到的第三方库、插件等  
axios/lodash,vue全家桶,可视化库选择了echarts,页面组件选用的是ElementUI

二、业务流程（前端）  
class\_monitor接收来自filesys的forward违规元数据信息后存储给web端处理使用。AI peeker recver接收来自AI peeker的自动识别forward数据，将数据处理后发送到消息队列给AI Analyzer进行人工智能处理，人工智能将识别出来的风险数据发送给class\_monitor进行违规处理。

1.单页应用程序，请查看routes.js的路由配置

2.风控质检地图，后台使用websockets推送数据给前端，前端展示中国地图背景，根据数据维度进行不同的气泡展示样式，违规的课程会以动画的形式反馈给质检员，质检员可以后续点击进行详细处理

3.质检的标记，根据质检的违规类型分别对其进行标记，由于大量是由AI进行此项工作，仍需要人工对其准确性进行校正

4.质检数据统计：包括课程违规统计、AI标记正确率统计等等方便分析优化的统计维度，大多以常见的统计图表的外观形式进行展现的，交互则是一些查询、超链接等等

三、项目重构、优化（前端）

1.项目的成长离不开维护、优化等等持续性迭代，以下几方面是项目的薄弱点：

1.1 代码风格规范化,社区中比较成熟的有自动化linter工具

1.2 自动化测试,也有很多自动化测试框架如jest/mocha等

1.3 模块化、面向对象，es6推荐的import方式，强类型、预编译语言（TypeScript、flow），对于长周期、大体量、易维护、多人合作的项目是目前的最佳实践了