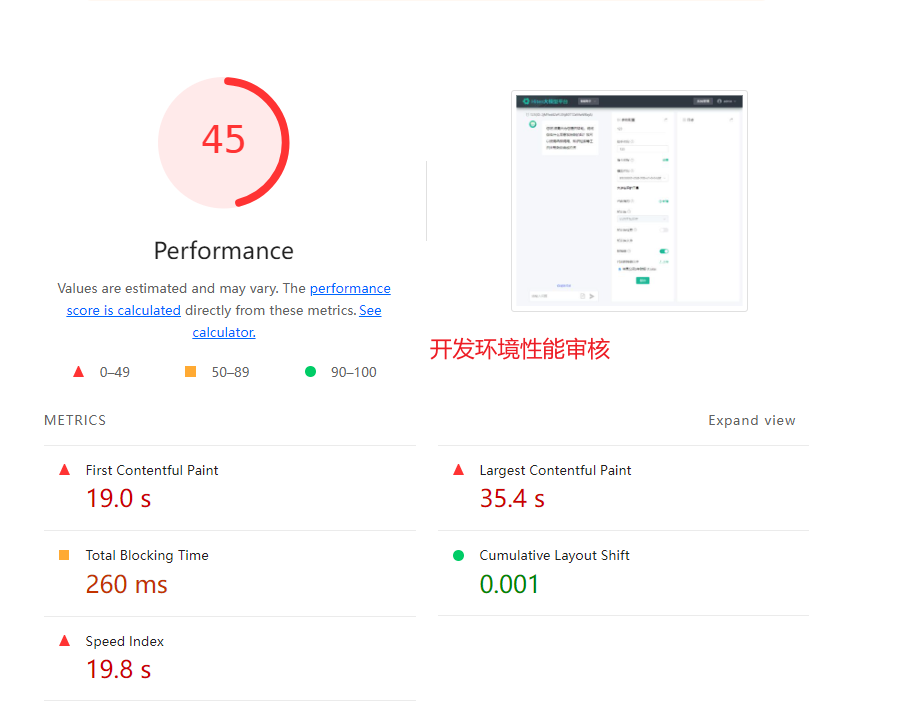
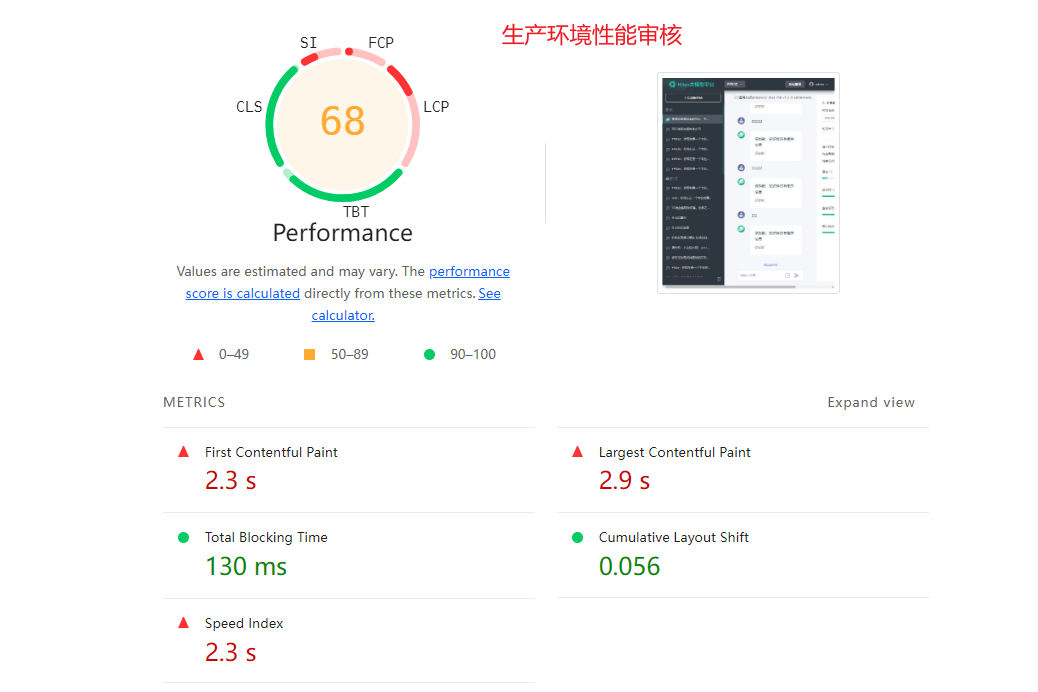
开发环境性能审核



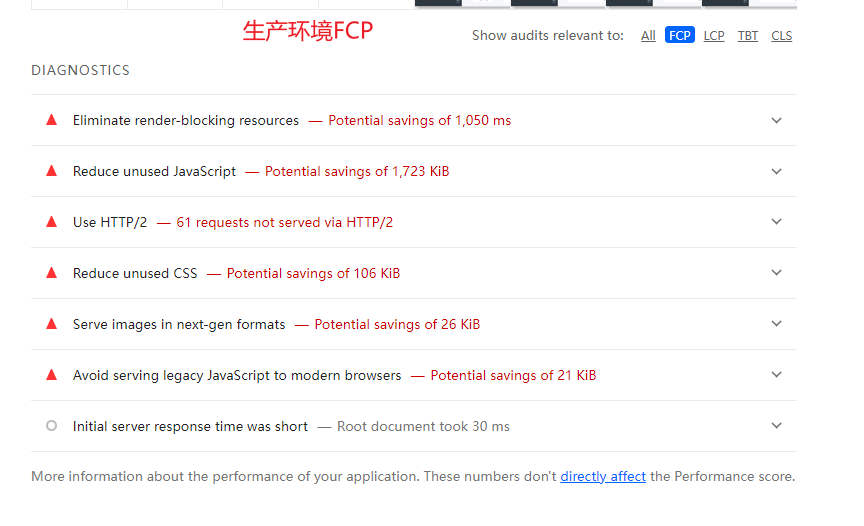
生产环境



我们实际上可以看到本地与生产环境的差别其实还是比较大的，**尤其是FCP、LCP与SI，他们相差了有将近10倍**

下面我们单独去看这三个指标

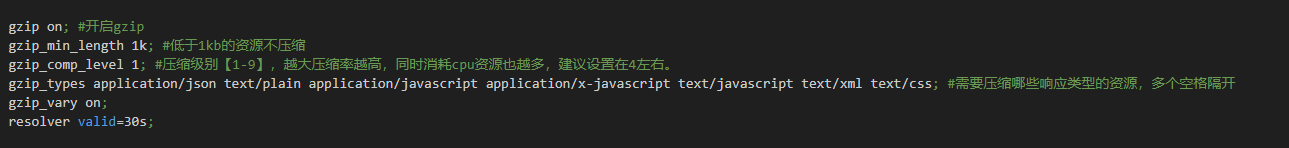




我们可以看到实际上有些指标下降幅度是比较大的，包括JS、文本、CSS压缩,移除未使用的CSS与JS,这些如果是以我的角度来看的话，是使用webPack去解决的,在optimization模块里面使用css-minimizer-webpack-plugin这个插件实现对css代码的高度压缩，使用babel插件实现对js代码的压缩

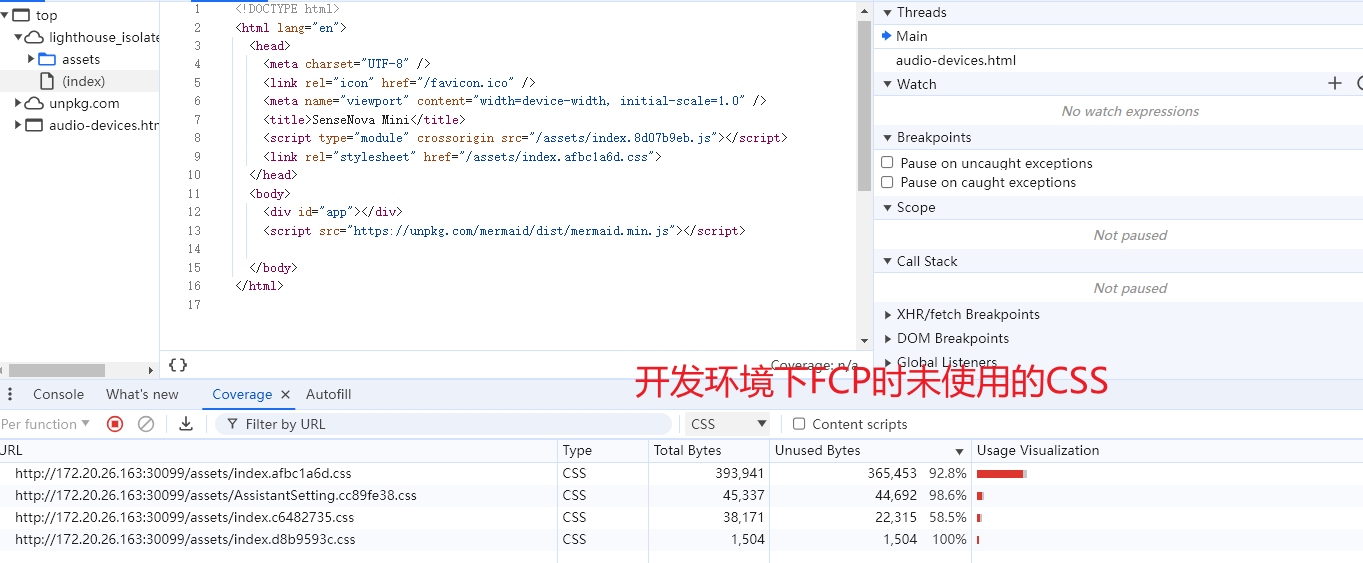
**至于对文本的压缩**，

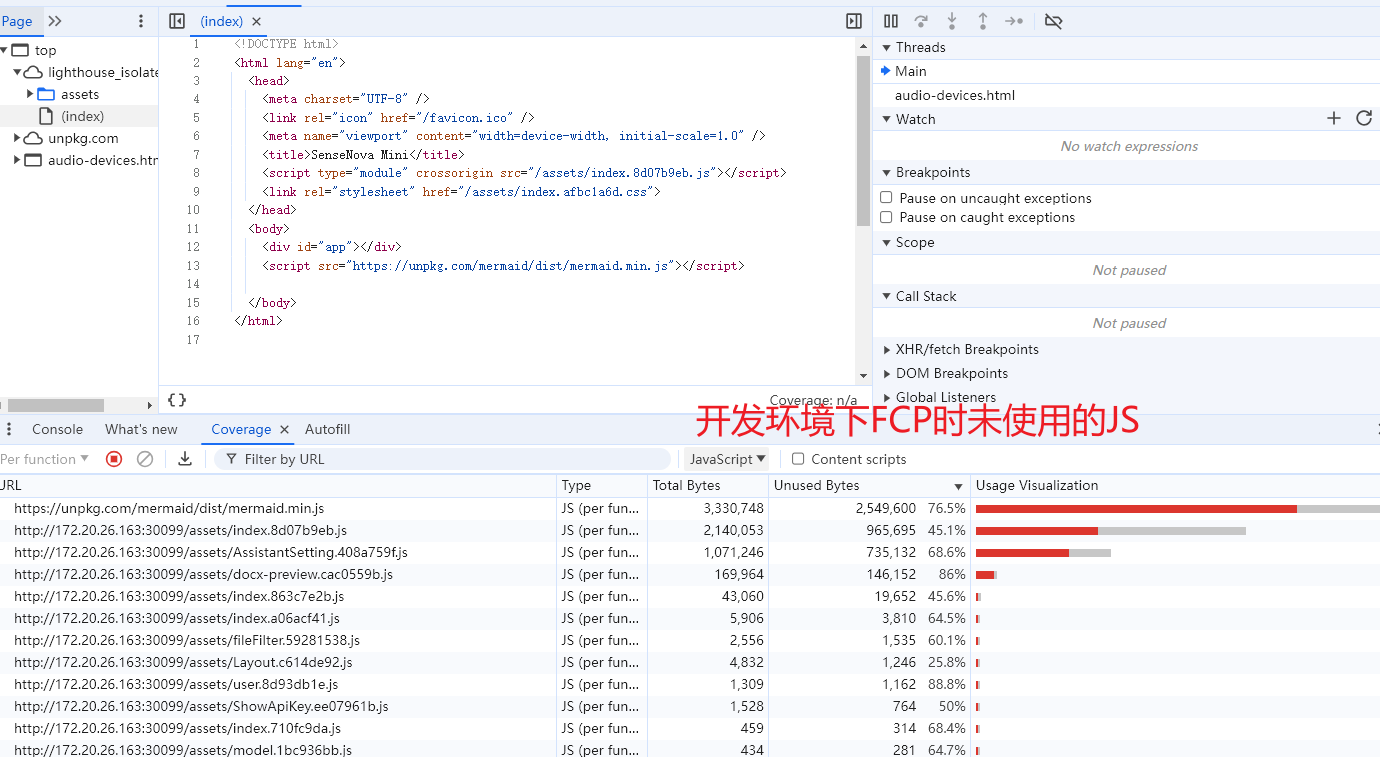
我们需要在服务器上去配置启动文本压缩，当前的项目代码是在nginx里面进行的配置，这个是作用与服务器的响应阶段，



也可以在代码的构建阶段去进行对文本的压缩，使用webpack提供的插件，这两者的作用时间和场景是不一样的，所以我们可以同时配置，但是我在我们的项目下没有看到webpack对项目代码打包时的构建，不知道是不是在构建镜像的时候去处理的呢？

**然后就是移除未使用的css与js**





这些感觉是比较少的，或许我们可以去手动处理，如果没有使用到的数量很多，同时也比较大，那么再去手动处理就比较麻烦了，所以有没有一些插件去自动检测这些未使用的资源文件（设置阈值），来减少FCP时间

然后其他的就是对图片的一些处理的了，可以使用懒加载，也可以使用Avif与webP格式的图片