

样式化代码审查平台

在构建现代 Web 应用程序时，样式不仅仅是让东西看起来好看——它关乎创建直观、响应式和引人入胜的用户体验。我最近探索了一个基于 Vue.js 的代码审查平台的 Stylus 样式表，其 CSS 架构是技术宝库，值得深入研究。让我们深入了解这个应用如何使用 Stylus 来打造其精致的 UI，从布局结构到悬停效果，同时保持代码的可维护性和可扩展性。

为什么选择 Stylus? 快速入门

Stylus 是一种 CSS 预处理器，它去除了传统 CSS 的冗长（无需大括号或分号），并添加了强大的功能，如变量、混合和嵌套。提供的代码从 `variables.styl` 导入变量，从 `base.styl` 导入基础样式表，为一致和可重用的样式奠定基础。例如，主色 `#1CB2EF` 可能在 `variables.styl` 中定义，并在按钮和背景中重复使用。

结构布局：部分和容器

应用的主页被分为不同的部分——`.slide`、`.feature`、`.reviewer`、`.example` 和 `.contact`，每个部分都有自己的样式策略。以下是 `.slide`（英雄）部分的样式：

```
.slide
  height 800px
  position relative
  color #fff
  width 100%
  overflow hidden

  .bg
    background url("../img/home/hero.jpg") no-repeat
    background-size cover
    background-position-y 40%
    position 200% 200%
    width 100%
    height 100%
    padding-top 280px
```

关键技术：

- **全屏英雄**：`height 800px` 和 `width 100%` 创建了一个大胆的全宽横幅。`overflow hidden` 确保没有内容溢出。
- **背景图像**：`.bg` 类使用 `background-size cover` 将英雄图像按比例缩放，而 `background-position-y 40%` 微调其垂直对齐以获得视觉效果。
- **嵌套**：Stylus 的嵌套将相关样式分组，提高了可读性，相比平面 CSS 更好。

响应式网格：Flexbox 和 clearfix

.feature 部分展示了一个三列布局：

```
.feature
  height 450px
  padding 125px 0
  background white
  .list
    width 1160px
    margin 0 auto
    display flex
    flex-direction row
    li
      height 200px
      padding-left 50px
      flex-grow 1
      &:first-child
        padding-left 0
    .short
      width 235px
      height 200px
      margin 0 auto
```

亮点：

- **Flexbox**：display flex 和 flex-direction row 将列表项水平对齐，而 flex-grow 1 确保它们均匀扩展以填充容器。
- **居中**：width 1160px 与 margin 0 auto 结合，居中内容，这是固定宽度布局的经典技术。
- **伪类魔法**：&:first-child 选择器从第一个项目中删除填充，防止不规则间距。

.example 部分进一步使用了审查卡片的网格，使用 clearfix() 混合：

```
.example
  .list
    clearfix()
  .row
    clearfix()
    li:first-child
      margin-left 0
  li
```

```
height 354px
margin-left 48px
pull-left()
margin-bottom 48px
```

- **Clearfix**: 这个混合（可能在 `base.styl` 中定义）处理浮动清除，确保在旧浏览器或自定义布局中行正确堆叠。
- **浮动网格**: `pull-left()`（另一个实用程序混合）将项目浮动到左侧，`margin-left 48px` 添加间距。这种方法与 Flexbox 相结合，以实现更广泛的兼容性。

交互样式：悬停效果和过渡

`.example` 中的审查卡片在平滑的悬停交互中闪耀：

```
li
  .info
    position relative
    height 354px
    width 100%
    color white
    box-shadow 0 4px 4px 1px rgba(135,135,135,.1)
    overflow hidden
    cursor pointer
    &:hover
      img
        transform scale(1.2,1.2)
        -webkit-filter brightness(0.6)
      .title
        -webkit-transform translate(0, -20px)
        opacity 1.0
      .tips
        -webkit-transform translate(0, -10px)
        opacity 0.8
  img
    height 100%
    -webkit-filter brightness(0.4)
    transition all 0.35s ease 0s
```

解析：

- **悬停效果：**悬停时，图像放大（`transform scale(1.2,1.2)`）并变亮（`-webkit-filter brightness(0.6)`），而文本元素向上移动并调整不透明度。
- **过渡：**`transition all 0.35s ease 0s` 确保所有属性的平滑动画，持续时间为 350 毫秒，带有缓动曲线。
- **分层：**`.text` 的 `position absolute` 将其定位在图像上方，`z-index 2` 确保可见性。

`.author` 按钮也有反应：

```
.author
  position absolute
  background black
  margin-left 30px
  margin-top 30px
  height 30px
  padding-left 20px
  padding-right 20px
  transition all 0.35s ease 0s
  &:hover
    background #1cb2ef
```

简单的颜色从黑色变为品牌颜色 #1CB2EF 的悬停效果，增加了愉快的触感。

视觉抛光：阴影、按钮和图标

阴影增强了深度，如 `.info` 的 `box-shadow 0 4px 4px 1px rgba(135,135,135,.1)`。按钮，如 `.contact` 中的按钮，被精心设计：

```
.contact
  .rightbtn
    .more
      width 127px
      height 50px
      color #1CB2EF
      background white
      border-radius 3px
      border 1px solid #00A3E6
      -webkit-box-shadow 0px 1px 0px rgba(255,255,255,0.15) inset, 0px 1px 2px rgba(0,0,0,0.15)
```

- **内阴影：**微妙的内阴影（`inset`）与外阴影结合，创建了按下按钮的效果。
- **一致性：**边框颜色 #00A3E6 与品牌调色板相匹配。

图标，如 `.icon_crown`，使用背景图像：

```
.icon_crown
  background url("../img/icon/crown@2x.png") no-repeat
  background-size contain
  width 49px
  height 52px
```

@2x 后缀表示适配 Retina 显示器的资源，`background-size contain` 确保正确缩放。

最佳实践和总结

这个 Stylus 实现为任何 CSS 项目提供了教训：1. **使用预处理器**：Stylus 的嵌套和混合（例如 `clearfix()`）简化了复杂布局。2. **平衡布局**：结合 Flexbox 用于现代浏览器，并使用浮动作为后备，以实现健壮性。3. **增强用户体验**：平滑的过渡和悬停效果使 UI 看起来更生动。4. **保持可维护性**：利用变量和导入在大型代码库中保持一致。

无论你是为代码审查平台还是个人作品集设计样式，这些技术都可以提升你的 CSS 水平。下次编写样式表时，考虑嵌套、过渡和一些阴影如何改变你的设计！