

(三) Component diagram

1. Component diagram

接下来，进一步放大和分解每个容器，以确定主要的结构构建块及其相互作用。组件图显示了容器是如何由多个“组件”组成的，每个组件是什么，它们的职责以及技术/实现细节。

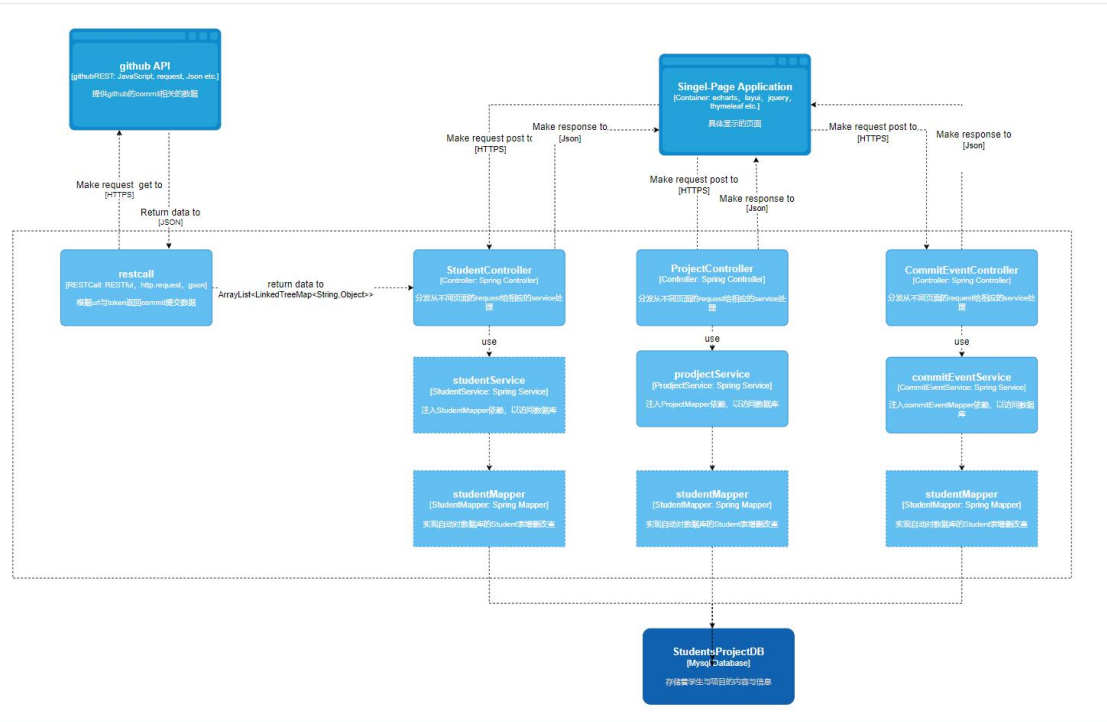
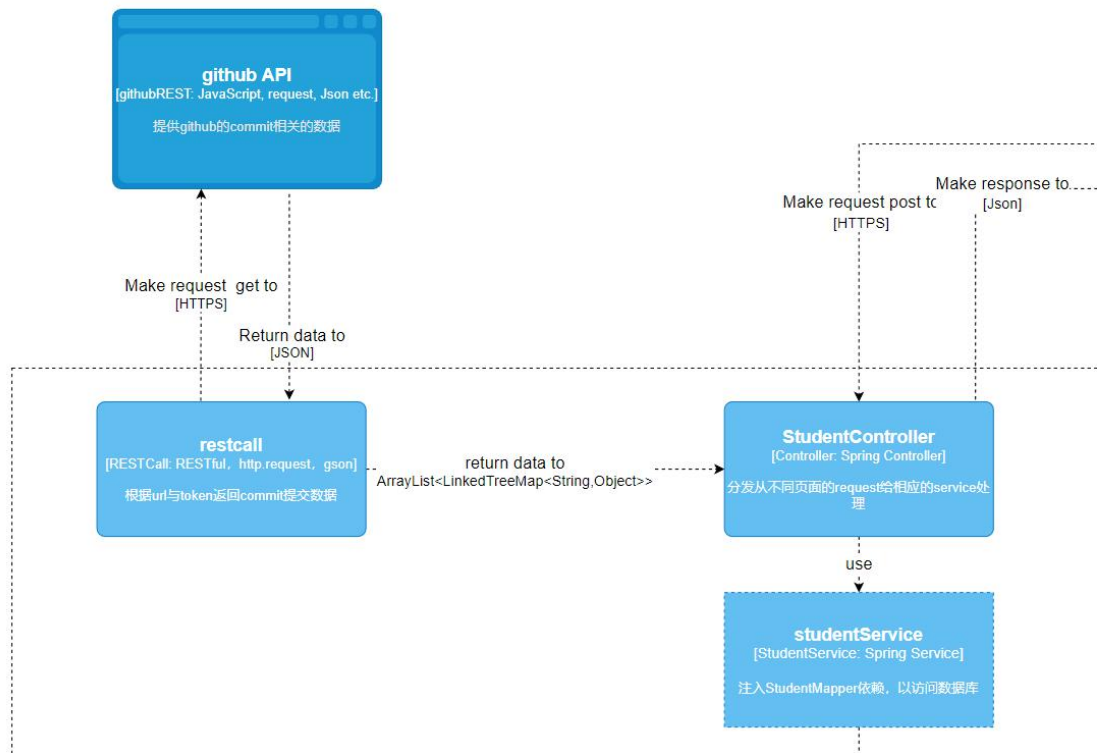


图 Component diagram

如上图所示，组件图大体上可以划分为后端、前端两个部分。

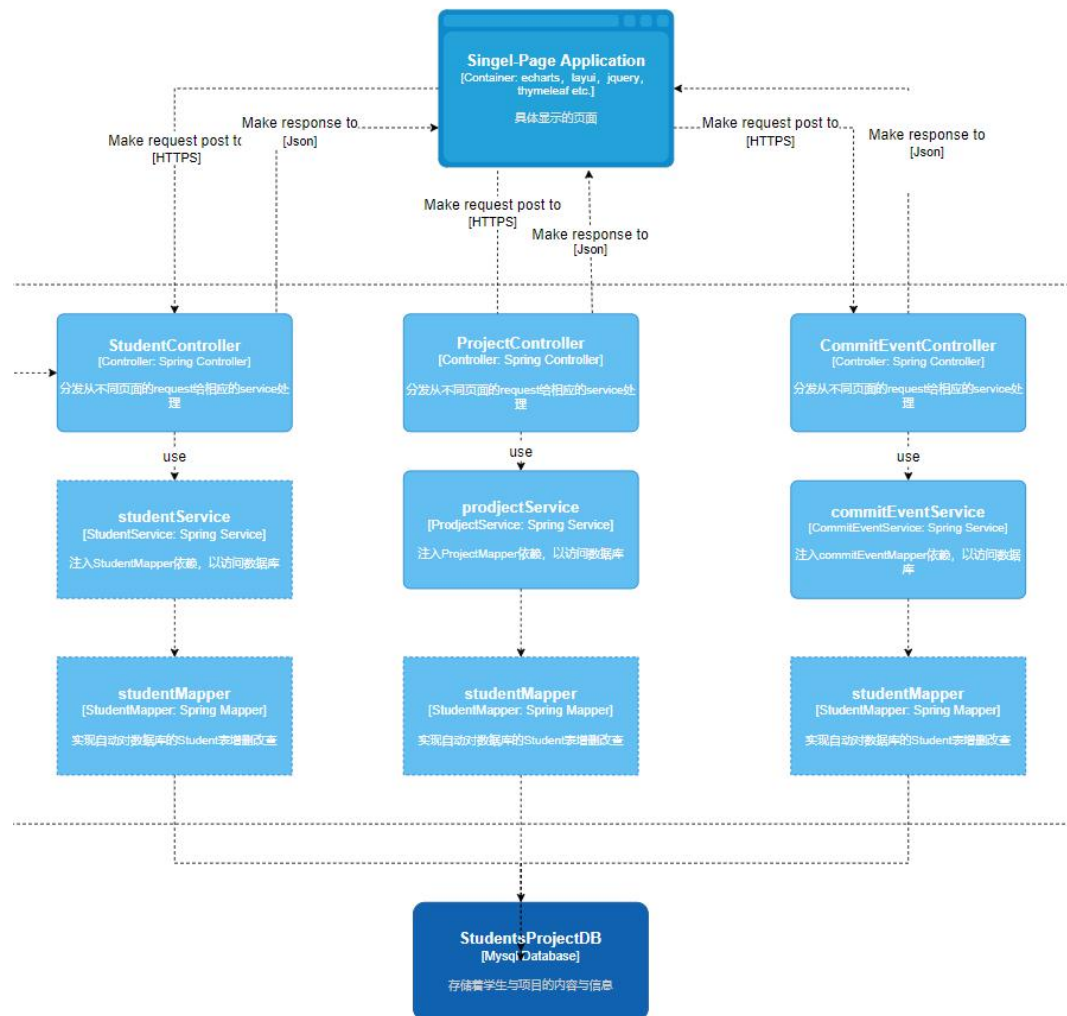
2. Component diagram 详细介绍

(1) 后端



后端主要凭借 Github token，由 `makeRESTCall()` 函数，向 Github API 发出数据请求。Github 将请求的数据返回给 `RESTCall`，并将数据从 `StudentController` 返回 Spring boot 容器。`StudentController` 通过使用方法 `StudentService`，将数据写回数据库，并保存。

(2) 前端

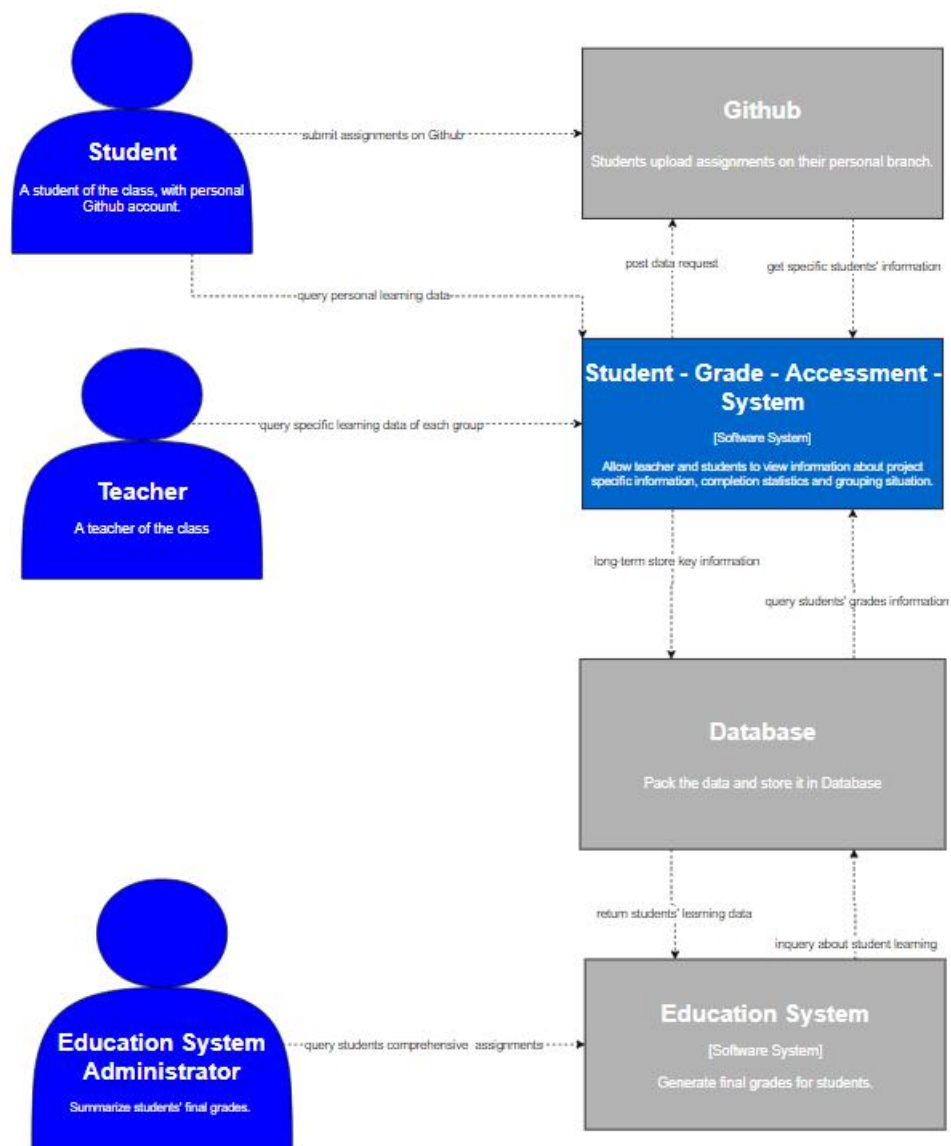


前端主要是 3 个实体，分别为 Student、Project、commitEvent，3 个实体配套会产生相应的 bean、service、controller、mapper。

用户从网页 html 中发出请求，使用 ajax 和 url 等方式将请求发送给 controller 控制器，控制器经简单处理后，调用 service 和 serviceImp 将处理进行具体处理。由于 service 中会注入 mapper，这样，就同时实现了对数据库和 xml 文件的调用和数据访问。

(一) System Landscape diagram

1. System Landscape diagram

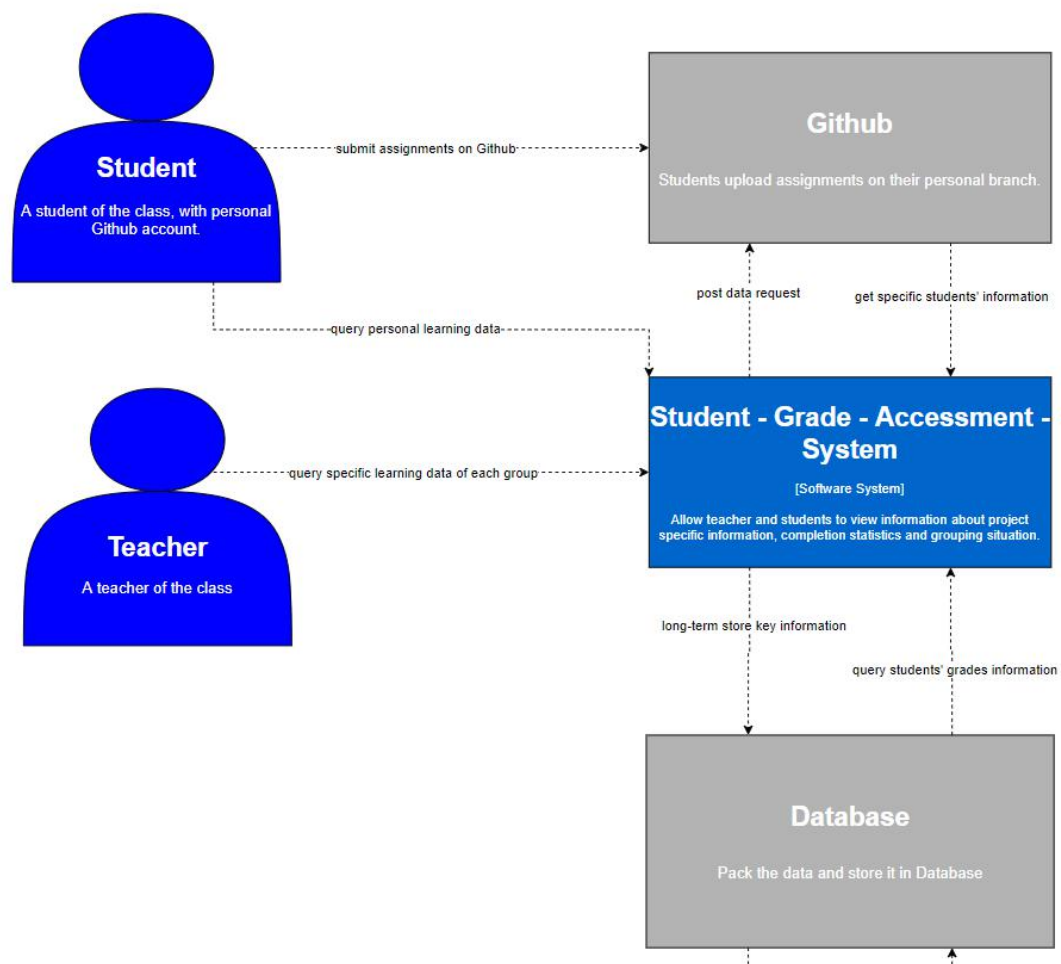


C4 模型提供了单个软件系统的静态视图，但在现实世界中，软件系统永远不会孤立存在。出于这个原因，特别是如果您负责一组软件系统，了解所有这些软件系统如何在企业范围内组合在一起通常很有用。为此，只需添加另一个位于 C4 图表“顶部”的图表，以从 IT 角度显示系统格局。与系统上下文图一样，该图可以显示组织边界、内部/外部用户和内部/外部系统。

从本质上讲，这是企业级软件系统的高级地图，每个感兴趣的软件系统都有C4 下钻。从实践的角度来看，系统景观图实际上只是一个系统上下文图，没有特别关注特定的软件系统。

2. System Landscape diagram 详细介绍

(1) 学生成绩管理系统与学生、老师、数据库的交互



(2) 学生成绩评价系统还可以与教务系统相关联，通过学生成绩评价系统，教务系统管理员可以查询学生学习的详细信息。管理员通过对学生学习情况综合测

评，最终生成学生的综合测评成绩。

