

实验总结

张敏濠 | 2020080308 | 土木 03

首先，最初的困难来自于设计数据库。起初，设计某些表格似乎很容易或直接，但考虑到在学年期间做这个项目的时间有限，才真正考虑实施。后来在应用系统的开发开始的时候，开始出现一些问题，这就需要我重新设计甚至删除一些数据库的表。比如，我原本打算创建一个关系表来存储玩家的统计信息。但是，我开始发现这种实现需要大量的数据以统计的方式呈现，变得很麻烦。因此，我决定从我原来的计划中删除球员统计表。

其次，我意识到这个实验最大的困难实际上是使用 **PowerBuilder** 软件本身。**PowerBuilder** 乍一看似乎简单易学，因为它是一个简单的编程语言的拖放式图形用户界面软件，并带有很多功能，例如使用起来非常方便的 **DataWindow** 和 **SQLQuery** 功能。但是，当涉及到某些语言语法时，**powerbuilder** 使用起来非常混乱，当您无法进行一些简单的更改或调整时，它会变得非常令人沮丧。例如，在生成基于字符串的 **ID** 时，我试图想出算法来随机生成它。但是，存在一些问题，我甚至无法将字符串转换为 **int** 数据类型，这会妨碍 **ID** 生成算法的设计。此外，您不能直接使用 **IF**、**ELSE IF** 语句这一事实也是 **PowerBuilder** 的一个问题。还需要注意的是，某些窗口不能同时运行。也许，你可以考虑使用响应式窗口来防止相互干扰和影响。

我还尝试使用 **PowerBuilder** 中的全局值功能，但相反，我发现将值存储在窗口的特定位置更容易（例如，将 **ID** 作为占位符存储在不同的页面中）。在其他情况下，我还尝试创建一个 **GLOBAL_VAR** 表来远程存储一些全局变量值，例如 **deadline** 和 **deadline_status** 来确定是否启用了申请按钮。

尽管我遇到了所有困难，但探索 **SQL** 和数据库是一次非常好的体验。在这门课程之前，我对数据库的经验为零，我通常只处理普通的编程，例如使用游戏引擎的游戏开发以及不同的编程语言，例如 **C/C++**、**Java**、**Python**。学完这门课后，我可以开始学习在其他领域实现数据库知识，例如移动应用程序开发、游戏开发、图形编程等。