# 软件工程综合实践个人总结报告

## 第一部分 关于团队

我在本次软件工程综合实践中担任了组长。在召集组员过程中，先是以我们寝室4名成员为基础，加入了4名在上一次谭老师软件工程课程组队时的老成员，并有1名新成员加入到我们的团队。虽大多是曾经合作过的同学，但上一次软工组队我是组员，而本次我是第一次担任组长来负责整体软件工程项目的安排工作，包括所有4台前后端服务器的持续集成自动部署的配置、前端Vue和后端Gin、Flask的环境构建和对组员的教学及Debug、所有技术栈和框架的选择（Vue、Gin、Flask、MySQL、Google Cloud Platform）等。

我们团队的实力还是比较强的，团队里累计获得过4个冯如杯科技竞赛一等奖，2个冯如杯科技竞赛二等奖，和冯如杯科技竞赛三等奖若干及冯如杯创意大赛特等奖2个，每一位成员的开发能力和开发效率都是相当高的。和其他某些组不同，我们不需要肝，便能快速、顺利、高质量地完成在站会时所布置的任务，不需要加班，也没有过加班。我们能做到不加班便把几乎所有的需求都做到相当好的程度的原因除了我们能力较强外，也离不开我在以往软件工程实践中运用到的一些高效率的管理方法和管理工具——持续集成，持续自动部署，我们的前后端累计部署了超过500次。前端成员无需安装后端的运行环境，只需要访问后端部署至服务器上的Swagger API（通过代码注释自动生成的文档）即可非常方便地查看最新的接口，并能够在该Swagger页面上一键运行该接口查看到接口的返回字段详情。Google Cloud Platform上我们部署了前端Vue、后端Gin（主要后端业务）以及后端Flask（爬虫相关业务），在Heroku平台上部署了Flask疫情预测服务，另有阿里云上的MySQL数据库服务和腾讯云服务器用来做爬虫和数据导入工作。另外，我们组的学习能力非常强，有几个成员快速地从只会后端短短一天之内便学会了前端，成为了全栈选手（比如我，以前只做过后端，这次在我旁边组员的带领下很快学会了怎么样用CSS的Flex进行页面布局，使用UI组件库组件以及接口对接），从数据获取到后端API到前端页面与API对接能一条龙地完成，大幅降低了对接过程的时间成本，也提升了效率。

当然，团队的管理不只是技术层面拥有了好的工具，这一次软件工程实践我也学会了一些团队管理方法，以Scrum敏捷为指导，通过每日站会来对工作进行回顾总结和反思，并展望当日所要完成的任务。通过划分用户故事和看板，填写工时，编写测试计划和验收标准等方法来监控团队工作效率。通过区分出PO、Scrum Master等人员，对团队进行管理，而不只是人人都是Developer，埋头只写代码……从讲座中敏捷教练的口中，我也得知这一套团队协作的管理流程也正是许多大厂中所真实实践运用的相对简化版，确实非常有效，我受益匪浅。

在平时工作中，我们团队的氛围一直保持着相当好。每一位成员遇到了任何困难都可快速地通过寻求其他组员的帮助获得快速有效的解答。在我们下班后的时间里，我们组织了团体建设团建活动，据不完全地记录，我们在木屋烧烤，头一号，麦当劳，肯德基，必胜客等多个场所和餐厅，进行了下班后的团建活动，组员们积攒了一天的或好或坏的情绪都会一起释放掉，这保证了第二天的良好状态。

## 第二部分 关于产品

我对我们最终呈现的产品是相当满意的，虽然在答辩时失误了，许多数据分析的图表没有展示出来，也失误了（并非是数据有问题，我选择的浙江省在那段时间内的新增病例就是0，中国疫情控制得那么好，我没反应过来说错了），但我们的UI经过重构变得简洁美观大方，功能方面也非常地全面，除了AI政策其他都做了，且完成度都很高。尤其是我们的行程轨迹在Echarts中的气泡效果来显示经停站，有的组互评的时候评到了我们给他们留下了深刻的印象，跑过来问我们是怎么做的，向我们学习。具体的功能在此不再详述，在用户手册中和演示视频中会有完整的说明。以下，我将具体说说我所负责的几个模块，我在完成这些部分时的心得体会。

一、订阅模块。这个是我一天边学前端边完成的第一个部分。从这个模块的实践中我学会了如何进行接口的对接，GET和POST方法的后端请求应该如何获取，console.log可用来调试，需要通过var \_this=this来避免一些JS的底层问题，v-for、v-show等Vue方法的使用。我是第一次做前端，这次实践能学到初入门前端是相当大的进步了。由于订阅部分的API接口也是我用Gin写的，这一部分我相当于是全栈了，全权由我来负责，前后端的联调我一个人就好了，效率非常高。在与核心用户即林老师和助教的交流过程中，我了解到订阅要显示出变化，比如苹果iOS中有新消息提示时在图标的一角会显示消息的数量，因此我们在最后几天特别做了订阅消息提示的功能，对于用户所订阅的城市有新增病例的变化时，便会把消息推送给用户，提升了用户体验。

二、防疫小知识版块。这部分我写了接口，找了数据源（爬虫是另外一位同学写的），我再做了前端的展示。本来这一块的展示是通过占地方比较大的卡片来呈现的，我们把我们的界面给核心用户林老师看，林老师用OA系统为例，让我醍醐灌顶，原来实用性是第一位的。界面的样式不一定要特别好看，特别炫酷，特别多样化，特别炫技，但一定要实用，页面中的信息量要大，关键信息要一目了然。于是，我再和PO交流，将卡片改成了类似于OA的列表，使用到Vuetify的表格组件，后面的效果确实也很好看，且更实用了。

三、疫苗数据分析。这一部分我找了数据，建了数据库表，定义了我在前端想要的数据字段格式，然后把后端API交给了另外一位同学去写，我并行化地做了前端数据的展示。在这一部分，我第一次学会使用ECharts来进行图表的绘制，虽然是相对简单的饼图来刻画世界疫苗接种的情况，我在引入ECharts组件开始时遇到了些问题，要非常感谢组员的帮助，通过查询ECharts的文档，对显示控件进行了调整，最后呈现出了不错的数据可视化效果。另外，国内疫苗接种情况和接种率的展示我是通过表格来显示的数据，也很直观。

四、预测分析。由于我们在冯如杯科技竞赛一等奖项目中做了比较多的机器学习的训练。我在最后2天做了该部分的工作。将各国的所有的疫情数据基于SEIR模型，并综合使用了pandas、numpy和scipy等工具对曲线进行了迭代拟合与优化。

五、后端Gin支持。我配置了Swagger，以通过写注释的方式来自动生成API文档，提升前后端对接的效率并大幅降低了接口对接时的成本。我们不再需要一直去维护一个在线的石墨文档，写代码的时候修改下注释即可，这点非常方便，应该是只有我们组在使用。众所周知，Go作为一个静态编译型语言，并发性好，我们也有信心我们的平台，在高并发场景下，后端远胜于许多用Django等Python Web框架的小组。我们组里会Gin框架的只有我一个人，但我有意识地培养另外一位同学去使用Gin框架，虽然他一开始写接口很慢，而后面他有什么问题来问我，我给他解答，到最后他也会立了Go中的哈希map、结构体继承等方式来提升代码和接口质量，他写接口的速度也越来越快了，能帮我分担后端的任务了。

六、服务器支持。本次我们组最终交付平台的所有5台服务器都是我去支持的。在Google Cloud Platform上，前端Vue服务部署在App Engine上，后端Gin服务部署在App Engine上，后端Flask服务部署在Cloud Run上。疫情预测服务部署在Heroku上。数据库是阿里云MySQL 1核1G服务（经过我们的测试，在当前并发条件下完全足够了，该服务器的最高CPU负载一直没有超过40%）。我通过设置好Trigger以及配置Docker，完成了全服务的持续集成、自动部署。

## 第三部分 关于过程

作为组长，我和Scrum Master共同制定了团队规则：

1. 遇到问题一定要立刻和PO或组长沟通。
2. 站会不能迟到，全员参与。每个人汇报昨天已经完成的工作，由PO安排当日需完成的工作。
3. 每一次代码提交commit需经过测试，以保证自动部署的顺利运行。
4. 完成每一个用户故事时，必须拖动看板填写实际工时等。
5. 听从PO和Scrum Master的指挥，代码风格统一，尽量保证完成任务。
6. 不鼓励加班，白天工作一定要高效，晚上保证充足睡眠迎接第二天的工作。

在团队规则的基础上，我们在软件过程实践中逐步学习和体会到了Scrum敏捷开发的各种思想。通过每日站会来对工作进行回顾总结和反思，并展望当日所要完成的任务。通过划分用户故事和看板，填写工时，编写测试计划和验收标准等方法来监控团队工作效率。通过区分出PO、Scrum Master等人员，对团队进行管理，而不只是人人都是Developer，埋头只写代码……从讲座中敏捷教练的口中，我也得知这一套团队协作的管理流程也正是许多大厂中所真实实践运用的相对简化版，确实非常有效，我受益匪浅。

其中，最让我们所有人都有了教训和体会最深的是要面向用户，关注用户真正想要什么，而不是埋头只干自己的技术。正因为我们一开始没有和用户多多沟通交流，了解核心用户如林老师的需求，我们在第一次迭代展示时的UI界面做得华而不实，林老师不喜欢，我们后面推导重来，全部重构。在第二次迭代时，因为第一次迭代做出来的东西不符合老师想要的，基本无法复用，面对要来不及干的局面，我们组的刘坤昊同学做出了关键决策——都到前端来，组里从1人前端到8人前端，这使得大部分同学都能全栈去完成某一项功能，大大减少了数据以及前后端交接过程中带来的沟通障碍，效率提升了非常多，最终的功能也是得到了林老师的认可，是可用、实用且好用的。这也是Scrum中的另一点，要持续为用户提供最大化的价值，保证交付产品的可用好用。

## 第四部分 关于课程的建议

这门课我受益匪浅，感谢老师和助教这两周里基本上都陪在我们身边一直为我们答疑解惑。一个建议是，老师和助教都实在是太辛苦了，老师和助教其实不用一直在机房待着，可以更多地以线上（如腾讯会议）的形式展开，这样也能和我们快捷地进行交流反馈。