总结报告

“网络学堂客户端”项目的基本功能已经实现，经过真实应用场景下的测试，能够满足日常使用网络学堂相关需求，有效简化了操作。对比于选题报告，完成了大部分目标，一些小功能由于时间不足，决定暂不实现，对本软件的正常使用没有影响。本项目已在GitHub上开源，欢迎有兴趣参与本项目者共同开发。

以下将从收获、挑战、不足和展望四方面对本项目进行阶段性的总结。

1. 收获：
   * 对一个较大的项目进行多人开发、维护，积累了团队开发的经验，一定程度上提升了Java实际开发的能力。
   * 对Java的GUI、Web请求、信息序列化等进行一定尝试，获得了比较满意的效果。
   * 在开发过程中，阅读了一定量的开源项目的文档与源代码，提升了理解Java代码、使用开源项目的能力。
2. 挑战：
   * 本项目依赖的一些开源项目文档比较简略，部分用法未在文档中体现，通过阅读源代码和尝试解决了这个问题。
   * 本项目需要用多种方式将原本的网页信息转化为Java对象，再转换过程中，由于没有相关API约定，遇到了一些边缘情况，在测试报告中也有所体现，通过进一步判断解决了此问题。
   * 数据更新需要与UI同步。通过数据绑定、监听解决了此问题。

其他的挑战与解决已在设计文档中说明。

1. 不足：
   * 本项目暂未支持2018版网络学堂的相关课程。虽然这与选题报告中的期望相符，且在实际应用中也覆盖了大多数课程，但仍然是一个遗憾。
   * 如果同时在浏览器与本客户端中使用网络学堂，会导致浏览器中做出的变化在本客户端中更新不及时。这是由于实现轮询（polling）有一定的技术挑战，同时我们担心增加客户端的资源消耗，故在这一个阶段性版本中没有进行实现。因此，我们也不建议同时在浏览器与本客户端中使用网络学堂，如果确实认为有某些必要操作本项目没有提供，本项目提供了意见反馈渠道，可以日后进一步改进。
   * 本客户端启动速度较慢，但启动后的响应速度正常。启动速度较慢经过资料查询可能和JavaFX的设计有关，在启动时进行JVM的warmup之后，可以保证使用过程中的流畅性。同时也可能与本项目的UI结构有关，在之后可以考虑在启动时只渲染第一屏必要的组件，优化启动速度。
2. 展望：
   * 尝试Android应用制作，争取最大化复用现有代码，改善手机上使用网络学堂的体验。
   * 尝试系统提醒以外的提醒方式，例如邮件提醒、微信公众号提醒（可能是传达率最高的方式）。但目前主要考虑保证用户隐私，一切由本客户端发起的网络通信仅指向网络学堂。之后需要考虑保护用户信息的同时增设相关提醒服务，进一步优化用户体验。
   * 补充对2018版网络学堂的支持。
   * 尝试进行合理频率的后台数据同步，以实现新公告公布时的提醒。
   * 改善对windows以外平台的支持。
   * 探索windows的原生API，尝试为消息提示设置回调函数（用desktop包提供的API无法实现）。