

评审文档

时间	操作者	修改内容	版本号
2022年03月04日	刘红宇	创建并完成文档	v1.0

一般性问题

- [√]是否定义了系统需要集成的其他应用和/或系统?有哪些?描述对每一个的集成程度和集成策略。
- [√]用户群是否是地理分散的?
- [√]是否定义了系统提供服务需要的计算资源?
- [√]是否定义了系统的预期寿命?
- [√]是否描述了适应用户群变化、存储数据迁移和提交方式改变的方案?
- [√]是否定义了用户群大小和他们预期的性能水平?

风格问题

- [√]软件架构是否尽可能简单(ASAP,但不是简单)?
- [√]定义高层组件时是否不超过7个, 并且做到松耦合和一致?
- [√]是否尽可能地将低层次组件集合到高层组件中 (层次性)?
- [√]是否使用标准 (化) 组件?

一致性问题

- [√]是否明确了关联到软件需求的解释和动机?
- [√]是否明确了为什么选定的架构方案能够实现软件需求?
- [√]软件架构是否证明了关键需求和属性?

完整性问题

- [√]系统架构是否完整?
- [√]是否覆盖了所有的软件需求?
- [√]通过软件架构是否可以跟踪到关键第求 (比如使用用例) ?
- [√]所有正式文档和视因化产物是否都配有足够的自然语言的解释文字?

准确性问题

- [√]组件描述是否足够准确?
- [√]是否允许独立构建组件?
- [√]是否详细描述了高层组件的接口和外部功能?
- [√]是否避免了实现细节, 没有详细描述到每个类?
- [√]是否描述了接口的详细信息?
- [√]文档中是否明确记录了组件之间的关系?
- [√]是否解释了所用视图中的图形元素的含义?

可实施问题

- [v]选定的解决方案是否可实施？
- [v]所有组件是否可实现或者被获得，然后整合到一起？
- [v]有需要的话，是否可以进一步的次分解，以便更好地把握可实施的要求？
- [v]是否考虑到了一般性的横切关注点？比如异常处理、初始化和复位、安全性、国际化和内置帮助等。
- [v]设计决定明确记载和动机？
- [v]是否限制开发人员的发挥空间。

其他问题

- [v]架构的备选解决方案是否都得到描述并评价价值？
- [v]是否考虑到了非功能性的需求？
- [v]是否使用到了一些负面指标来评价架构？
- [x]高复杂：一个组件有复杂的接口或功能。
- [x]低内聚：一个组件包含无关的功能。
- [x]高耦合：两个组件间（相互）联系过多。
- [x]高扇入：一个组件需要的其他组件过多。
- [x]高扇出：一个组件依赖于其他组件过多。
- [v]是否证明了软件架构的灵活性？
- [v]是否考虑到如何应对可能出现的需求变化？比如文档中是否描述过相关的变化场景。
- [v]是否尝试划分出团队缺乏经验的技术主题，并给予描述？