**合并项目总结 – LIVE CHAT cs补充**

#### 需求

1. 希望项目快结束前约2周就可以拿到下一个项目的需求，这样项目的交替会比较顺利。
2. 应该有一份Live Chat总的需求文档, 每个项目自己编制单独的需求文档，但是项目结束以后一定要更新总的需求文档。

#### **设计**

1. 总的数据库设计文档很久没有更新了, 用这样的文档进行培训的效果要打折扣, Conversion, Bot , Merge 这些项目新加的表都没有体现在总的设计文档里面, 应该在项目结束后安排人去更新总的数据设计文档。

#### **开发**

1. 依赖注入、 Asp.Net Core、 Redis 一类的技术在新的项目里面可以建议采用。技术更新有点慢，很多新技术可以提高开发效率。
2. 提交测试 - 需要由项目经理发邮件，发给RandD，需要包含提测模块，Release.txt，修改了什么Bug。

--> 提交冒烟测试的时候才需要，普通的提交没必要。

--> bug 修复和验证的流程如下:

1. 开发人员修复bug后提交代码。
2. PM 通过Jenkins 打发布包, 发布到开发自测环境，开发人员验证一下发布包是否有问题。
3. 测试把发布包发布到测试环境, 在Teams 里面通知已发布成功，发布了哪些模块。
4. 开发人员在测试环境验一下bug, 如果确认没有问题就把bug 的状态改成fixed
5. 开发人员通知测试bug 已修复，测试人员只验证状态为fixed 的bug。
6. 公共模块(Framework) 现在是一个大问题。
7. 没文档没说明，有些功能在具体的项目里面又再实现了一次。
8. 大家都可以改里面的代码，A项目可能会给B 项目带来潜在的问题。

#### **测试**

1. 项目结束以后要出一份测试报告，但是没有确定测试报告的模版是什么。
2. Bug 要分级，建议的分级定义，现在大家的bug分级相对随意

1 级 引起主流程走不通、界面直接报错，报黄页

2 级 导致异常流程走不通, 功能实现不完整或和需求有出入, 不影响主流程

3 级 文字拼写错误, 错误提示不对，样式问题，有效性、唯一性、边界条件校验有问题

4 级 建议性问题，可改可不改

1. 项目发布控制
2. 冒烟以后，直接按测试用例进行测试，测试未结束以前**不允许**打发布包。测试完成以后进入bug验证阶段。
3. 每打一次bug修复包视为一轮bug验证，每天最多打2个bug修复包，前面一轮的bug验证完成后才可以进行下一轮bug验证。每一轮测试尽量把bug验证的轮次控制在3次以内。
4. 测试用例一定要有优先级， 1 级的就是冒烟测试用例， 1-2 级的就是回归测试用例。
5. 测试普遍觉得没有对应的需求文档是很难理解测试用例的，测试用例本身不像需求文档那样可以结合图片、文字、流程对需求进行描述。
6. 自动化测试用例目前都没有和项目一起同步更新，这样很难运用到具体的项目里来。

#### **计划、发布**

1. 发布时间还是第二轮单元测试结束以后再定比较好，RC 或IT 测试结束以后再定会让其它部门过于被动。
2. 既然运维组要处理线上的bug， 建议项目结束以后项目组和运维组一起运维一段时间，期间项目组要给运维组进行必要的培训。

#### **其它**

1. 公司培训很多，但是针对性不强，以前还做过Python 的培训，像Chat Server , Agent Console 、压力测试做这些都没有组织过公共的培训，虽然大家可能没有代码权限，但是了解一下对自己解决问题还是有帮助的。
2. 长沙这边缺乏文化氛围，公司里面还挂着上一家公司的一些东西，给人感觉不好。