**1**

Интеграционный пример теста: наша задача проверить взаимосвязь всех компонентов согласно требованиям. Мы получили письмо во вкладку "входящие" и удалили его, переходим во вкладку "удаленные" и проверяем наличие письма. Если письма не оказалось, то стоит проверить оставшиеся вкладки.

Пример смоук-теста: наша задача проверить компонент на наличие простых ошибок. К примеру, попытка авторизации своей почты, попытка отправки письма, получение письма.

Пример ре-теста: наша задача проверить, что ошибка, выявленная ранее, была исправлена. К примеру, была проблема авторизации, при вводе корректного логина и пароля система выдавала ошибку, проверяем на наличие этой ошибки.

Пример нефункционального теста: к примеру тест на совместимость, нам нужно узнать устанавливается ли наше приложение, обновляется и удаляется ли оно, так же можно проверить на др ОС.

**2**

Необходимость регрессионного тестирования

Добавлен раздел “Спам”. Как самостоятельный компонент “Спам” не требует рег. теста, но рег. тест необходим если мы рассматриваем систему в целом или же связь остальных компонентов со “Спам”.

Раздел “Удаленные” переименован в раздел “Корзина”. Если переименование не влияет на функциональность компонента, то регресс не нужен. Если переименование влияет на его имя в коде, то регресс необходим, т.к нужно убедиться в правильности связи между компонентами.

На «Странице входа» устранен ранее обнаруженный дефект. Регресс не нужен, т.к вместо него нужно проводить повторное тестирование.

**3**

Необходимо провести смоук-тест, чтобы убедиться в правильности работы основных функций приложения.

**4**

Регресс тест поможет определить дефект, т.к новый компонент повлиял на работу кнопки.