

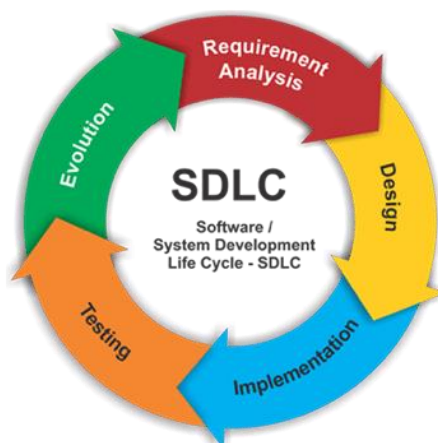
## Домашняя работа (Занятие 2)

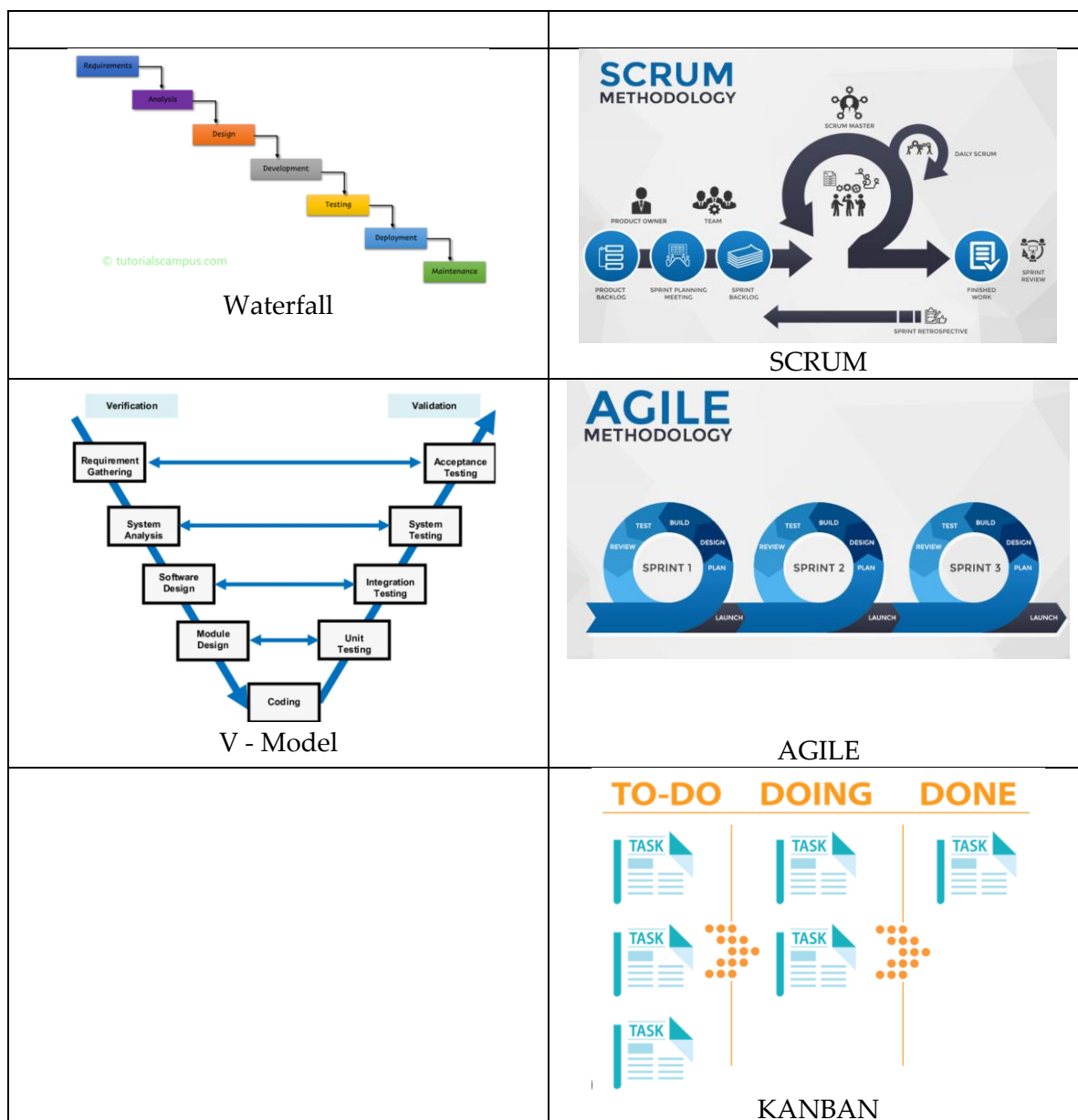
|               |            |                 |              |
|---------------|------------|-----------------|--------------|
| Имя           | Андрей Н.  | Предмет         | QA-3 Testing |
| Преподаватель | Виталий З. | Период обучения | 04-10/2023   |
| Дата          | 04.05.2023 |                 |              |

Введите инструкции для теста. Например, укажите, что учащиеся должны написать короткое описание к ответу на каждый приведенный вопрос.

1. Выучить фазы процесса разработки программного обеспечения и отобразить их схематически в любой удобной Вам форме (например: в Word).

### Software development life cycle (SDLC)- phases





2. Найти самый интересный, по Вашему мнению, баг из любого ресурса интернет. Описать на какой фазе разработки ПО, Вы считаете, возник этот баг.

#### ИСТОРИЯ 1.

Несколько лет назад в eBay столкнулись с рядом ошибок управления версиями во время перехода на технологию Java на их серверах.

Некоторые из проблем с версиями были совершенно очевидны. Например, вы не можете скомпилировать код для IBM JVM 1.5, а затем запустить байт-код для Microsoft JVM 1.4.

Позже были похожие проблемы, когда проект переходил на JVM 1.6. Код, скомпилированный для IBM JVM 1.5, примерно в 99,9% случаев был совместим с JVM 1.6, но оставшихся 0,1% было достаточно. Необходимо тщательно следить за тем, чтобы весь скомпилированный код был скомпилирован для соответствующих целей JVM. Большинство разработчиков Java также знают об этих потенциальных проблемах.

Позже eBaу столкнулись с серьезной проблемой управления версиями: разные сборки IBM JVM 1.6 были несовместимы друг с другом. Одна и та же основная и дополнительная версия JVM, один и тот же производитель, просто разные сборки. Это потребовало значительного количества исследования, чтобы выяснить, и вызывало значительные периодические отключения в течение периода около 4 недель, потому что они просто не могли определить причину проблемы. В конце концов они пригласили представителя IBM, чтобы тот помог нам диагностировать проблему. Никто даже отдаленно не ожидал, что разные сборки JVM при прочих равных условиях сами по себе могут быть причиной такой непонятной проблемы.

Мораль этой истории: всегда компилируйте ту же версию Java JVM и JRE, с которой собираетесь работать.

К счастью, у eBaу было достаточно избыточности, так что ни одно из отключений сервера за этот 4-недельный период не повлияло на клиентов.

## ИСТОРИЯ 2.

Несколько лет назад работая в системе электронной почты кампуса, позвонил председатель статистического отдела.

«У нас проблема с отправкой электронной почты из отдела».

"В чем проблема?" спросил ИТ специалист.

«Мы не можем отправлять почту дальше, чем на 500 миль», — пояснил председатель.

"Что?"

«Мы не можем посылать почту дальше, чем за 500 миль отсюда», — повторил он. — На самом деле, немного больше. Скажем, 520 миль. Но не дальше.

«Эм... Электронная почта вообще-то так не работает», — сказала администратор, пытаясь сдержать панику в голосе. При разговоре с заведующим кафедрой паники не бывает, даже такого относительно бедного отдела, как статистика. «Почему вы думаете, что не можете отправлять почту дальше, чем на 500 миль?»

«Это не то, что я \*думаю\*», — раздраженно ответил председатель. «Видите ли, когда мы впервые заметили это несколько дней назад...

"Вы ждали несколько ДНЕЙ?" — перебил его работник, дрожащим голосом. "И вы не могли отправить письмо все это время?"

«Мы могли бы отправить электронное письмо. Только не больше, чем...»

— Да, 500 миль, — понял. Но почему Вы не позвонили раньше?

«Ну, до сих пор мы не собрали достаточно данных, чтобы быть уверенными в том, что происходит».

Верно. Это председатель \*статистики\*. — Во всяком случае, я попросил одного из геостатистиков изучить это...

"Геостатистики..."

-- Да, и она подготовила карту, показывающую, что радиус, в пределах которого мы можем отправлять электронную почту, составляет немногим более 500 миль. В этом радиусе есть несколько пунктов назначения, до которых мы либо не можем добраться, либо добираемся время от времени, но мы никогда не сможем отправлять электронные письма дальше этого радиуса».

— Понятно, — сказал администратор и обхватил голову руками. «Когда это началось? Несколько дней назад, вы сказали, но изменилось ли что-нибудь в ваших системах в то время?»

«Ну, пришел консультант, пропатчил наш сервер и перезагрузил его. Но я ему позвонил, и он сказал, что не трогал почтовую систему».

«Хорошо, дай-ка я посмотрю, и перезвоню», — сказал он, едва веря, что подыгрываю. Это был не День смеха.

Администратор зашел на сервер их отдела и отправил несколько тестовых писем. Это было в исследовательском треугольнике Северной Каролины, и тестовое письмо на мой собственный счет было доставлено без заминок. То же самое для одного, посланного в Ричмонд, Атланту и Вашингтон. Еще в Принстон (400 миль) работал.

Но потом попытался отправить электронное письмо в Мемфис (600 миль). Это не удалось. Бостон, провал. Детройт, провал. Достал свою адресную книгу и начал пытаться сузить круг. Нью-Йорк (420 миль) сработал, а Провиденс (580 миль) провалился.

Потом администратор начал задаваться вопросом, не потерял ли он рассудок. Он попытался отправить электронное письмо другу, который жил в Северной Каролине, но чей интернет-провайдер находился в Сиэтле. К счастью, это не удалось. Если бы проблема была связана с географией получателя, а не с его почтовым сервером, он думаю, расплакался бы.

Убедившись, что — невероятно — проблема, о которой сообщалось, была правдой и повторялась, взглянул на файл **sendmail.cf**. Выглядело вполне нормально. На самом деле, это выглядело знакомо.

Он сравнил его с **sendmail.cf** в домашнем каталоге. Файл не был изменен — это был написанный им **sendmail.cf**. И я был совершенно уверен, что не включил опцию «FAIL\_MAIL\_OVER\_500\_MILES». В растерянности он подключился к SMTP-порту по телнету. Сервер радостно ответил баннером SunOS **sendmail**.

Минутку... баннер SunOS для отправки почты? В то время Sun все еще поставляла Sendmail 5 со своей операционной системой, хотя Sendmail 8 был довольно зрелым. Будучи хорошим системным администратором, он стандартизировал Sendmail 8. Администратора, он написал **sendmail.cf**, в котором использовалось хорошее длинное возмoжность самодокументирования и имена переменных, доступные в Sendmail 8, а не загадочные коды знаков препинания, которые использовались в Sendmail 5.

Кусочки сразу же встали на свои места. Когда консультант «пропатчил сервер», он, по-видимому, обновил версию SunOS и при этом \*понижил\* версию Sendmail. Обновление помогло оставить **sendmail.cf** в покое, хотя теперь это была неправильная версия.

Так получилось, что Sendmail 5 — по крайней мере, версия, выпущенная Sun с некоторыми изменениями, — могла работать с **sendmail.cf** Sendmail 8, так как большинство правил на тот момент оставались неизменными. Но новые длинные параметры конфигурации — те, которые он считал ненужными и пропустил. И бинарный файл **sendmail** не имел скомпилированных значений по умолчанию для большинства из них, поэтому, не найдя подходящих настроек в файле **sendmail.cf**, они были установлены на ноль.

Одной из настроек, которая была установлена на ноль, было время ожидания для подключения к удаленному SMTP-серверу. Некоторые эксперименты установили, что на этой конкретной машине с ее типичной нагрузкой нулевой тайм-аут прервал бы вызов подключения чуть более чем через три миллисекунды.

Странной особенностью нашей кампусной сети в то время было то, что она была на 100% коммутируемой. Исходящий пакет не вызывает задержки маршрутизатора до тех пор, пока не попадет в точку присутствия и не достигнет маршрутизатора на дальней стороне. Таким образом, время подключения к малонагруженному удаленному хосту в соседней сети фактически будет в значительной степени зависеть от скорости света на расстоянии до

пункта назначения, а не из-за случайных задержек маршрутизатора.

При наборе в своей оболочке:

\$ единиц

1311 единиц, 63 префикса

У вас есть: 3 миллисветовых секунды

Вы хотите: миль

\* 558.84719

/ 0,0017893979

«500 миль или чуть больше».

Источник - Страница на [berkeley.edu](http://berkeley.edu)