

СОВЕРШЕНСТВОВАНИЕ УПРАВЛЕНИЯ РАЗРАБОТКОЙ ИС РЕШЕНИЙ

Чаусова Полина Михайловна

студентка

Санкт-Петербургского государственного экономического университета,

г. Санкт-Петербург, Россия

Исследованы методы управления разработкой для ИТ-решений и представлен автором вариант для разработки на платформе ИС. Рассмотрены основные методологии управления разработкой программного обеспечения. Определены отличия между различными методологиями. Выделена наиболее подходящая методология и описан вариант решения по совершенствованию управления ИС решений. Перечислены преимущества подобранного инструментария.

Ключевые слова: ИТ-решение, методы управления, программное обеспечение, ИС, Agile, DevOps, Waterfall.

Программирование ИС-решений достаточно востребованный вид деятельности в России.

Для начала углубимся, с помощью какой методологии можно достичь высокого уровня разработки качественного продукта.

Методология разработки программного обеспечения — это процесс или серия процессов, используемых при разработке программного обеспечения. Понятие довольно широкое и включает такие этапы, как этап проектирования и этап разработки. Выбор правильной методологии разработки для проекта в значительной степени зависит от размера вашей группы, целей и других различных переменных.

Методологии разработки ИС-решений не отличаются от других ИТ-решений помимо ограниченного списка программ, которые умеют работать с транскрипцией кода ИС.

Для анализа рассмотрим наиболее распространенные методологии, которые применяются при разработке ПО.

Waterfall - «Водопад».

Самая распространенная методология до появления гибких методологий это Waterfall - «Водопад». Что касается улучшения программирования, то Waterfall - самое традиционное и последовательное решение. Даже при том, что ее обычно рассматривают как «старую школу» или устаревшую стратегию, полезно понять историю и структуру Водопада, чтобы больше понимать адаптированность более современных подходов. Впервые, сделанный в 1970 году, Waterfall был одним из самых заметных методов в течение очень долгого времени в результате подхода, основанного на устройстве каскада, который требует сформированную структуры и всю документацию по разработке заранее. В этой методике отсутствует адаптивность, а это означает, что то, что сформировал заказчик и аналитик в техническом задании, должно быть четко выполнено. Обычно одна фаза должна быть выполнена до того, как начинается следующая это помогает с назначениями ответственного. Более того, поскольку весь объем работ понимается заранее, прогресс программирования можно оценить без особых усилий.

Agile.

Agile — это вовсе не методология, а набор принципов разработки программного обеспечения. Принципы, изложенные в Agile-манифесте, выделяют четыре основных убеждения:

- Люди и взаимодействия по процессам и инструментам;
- Рабочая программа над исчерпывающей документацией;
- Сотрудничество с клиентом в рамках переговоров по контракту;
- Реагировать на изменения в соответствии с планом.

Быть гибким — это скорее философия и набор ценностей, а не процесс, который вы можете напрямую применить к проекту. Итак, Agile не методология или процесс. Даже если вы придерживаетесь принципов, вам все равно нужно определить процессы для реализации проектов. Когда люди говорят о Agile методологии управления проектами, имеется в виду что она, как правило, имеет гибкий, итеративный дизайн и сборку процесса. Гибкие проекты характеризуются рядом задач, которые задуманы, выполнены и адаптированы в соответствии с требованиями, а не заранее запланированным процессом. Гибкость помогает командам реагировать на непредсказуемость посредством дополнительных итеративных рабочих процессов.

Agile отличается от других методов управления проектами, которые обычно предполагают, что вещи, влияющие на проект, предсказуемы, и поэтому он подчеркивает адаптивность к изменяющимся ситуациям, адекватное и постоянное общение между командой проекта и между клиентом. Гибкие методологии отлично подходят для использования в динамических средах, где есть потенциал для изменения или развития требований, таких как разработка программного обеспечения или игр.

Как набор принципов, Agile, как правило, употребляется как общий термин, используемый для обозначения Agile, включая Scrum, eXtreme Programming (XP), Kanban и Scrumban.

Между компаниями и клиентами довольно гибкое понимание о методологии Agile.

Agile в значительной степени хвалят за то, что он «не Водопад», и его часто неправильно понимают как «делать больше с меньшими затратами, быстрее и дешевле, чем когда-либо прежде».

Клиенты, как правило, любят идею Agile из-за ее очевидной гибкости в управлении проектом, предоставляя им больше возможностей постоянно менять свое мнение на протяжении всего проекта. Заказчики часто думают, что это означает, что они получают больше работы за меньшие деньги или, что им никогда не придется принимать окончательное решение о чем-либо, потому что они могут изменить свое мнение вплоть до последней минуты. По правде говоря, Agile может быть лучше, но Agile не бывает дешевым или легким.

Подвох заключается в том, что такой уровень гибкости дорог. Да, вы можете изменить свое решение, но оно пожирает время, а время стоит денег. Другая проблема заключается в том, что для того, чтобы быть успешными и по-настоящему гибкими, клиенты должны быть доступны и иметь возможность принимать решения (что редко встречается в иерархических и ориентированных на правление организациях). Они должны обеспечивать постоянную обратную связь и расстановку приоритетов на лету, чтобы поддерживать работу проекта, что часто очень сложно.

В качестве методологии я бы применила DevOps, которая имеет множество последователей в современной разработке и применяется при создании множества программных решений.

Что такое DevOps?

Нет четкого определения данной методологии, каждый подразумевает что-то свое.

Обычно это взаимодействие между людьми из команды разработчиков и людьми из команды обслуживания ПО. Также DevOps – это относиться ко своей инфраструктуре, как к коду, так как это один из ресурсов, который нам нужен для работы нашего ПО. Помимо прочего DevOps подразумевает использование автоматизации.

DevOps это небольшие развертывания версий новых релизов. Эта методология чаще всего подразумевает использование канбан доски. Для успешного внедрения новой практики подразумевается пройти через 3 этапа:

- Люди - должны понимать, что такое применение новой культуры DevOps.
- Процесс -формализация этапов в процессе, со всеми взаимодействиями разных команд и настроенными автоматизированными тестами, и выпуском в продуктивную базу.
- Продукт – инструменты, которые помогают автоматизировать все описанные процессы.

Большая проблема при внедрении новой методологии это неприятие новшеств со стороны сотрудников и негативное отношение к обучению работы с новыми продуктами и процессами. Многие продукты, помогающие автоматизировать тесты или осуществлять мониторинг, часто уже присутствуют на проекте до принятия новой методологии. Так происходит и в самых крупных и передовых ИТ-компаниях. Чтобы избежать такого негативного отношения, следует описать всем сотрудникам, что многие решения по автоматизации уже присутствуют в существующем процессе разработки. Например, канбан доска, для мониторинга тенденции выполнения задач в релизе, или автоматические нагрузочные тесты, которые помогают выявить, насколько программный продукт готов к взаимодействию с сотнями пользователей и т.д.

При разработке 1С решений происходит уникальная ситуация, когда большинство программных решений просто не умеет работать с языком 1С, так как он имеет русскую транскрипцию (английскую тоже, но этот вариант менее распространенный) и небольшую широту распространения по сравнению с другими языками (HTML, Java, C+ и т.д.).

Да и для такой разработки 1С на рынке существуют решения.

Team Foundation Server– программный комплекс, который позволяет автоматизировать жизненный цикл разработки ПО.

Для этого решений компании разработали плагины для интеграции Team Foundation Server и Visual Studio Team Services со своими продуктами, такие как: GitHub; Jenkins; JetBrains; Slack; Octopus Deploy; Chef; Xamarin и т.д.

При разработке Git умеет проверять на наиболее частые баги при выгрузке в репозитории. Его расширение можно установить и настроить автоматическую выгрузку в хранилище прямо из приложения Git, в котором, например, привык работать разработчик.

В качестве инструментов, по этой схеме предложен следующий вариант решений:

OneScript или 1Script

Инструмент автоматизации для специалистов по 1С. OneScript - скриптовый движок. Имеет программу – интерпретатор, которая запускается из командной строки и получает на вход имя файла скрипта для выполнения.

Преимущества:

- Возможность писать программы на языке 1С. Нет необходимости изучать другие скриптовые языки
- Не надо устанавливать технологическую платформу 1С:Предприятие. Выполнение скриптов осуществляется с помощью интерпретатора
- Автоматизация рутинных процедур. Запуск скриптов с готовыми командами.

Git Распределенная система управления версиями

Git поддерживает быстрое разделение и слияние версий, включает инструменты для визуализации и навигации по нелинейной истории разработки. Git предоставляет каждому разработчику локальную копию всей

истории разработки, изменения копируются из одного репозитория в другой.

1С:Автоматизированная проверка конфигураций

Программный продукт для автоматизированной проверки конфигураций

Проверка конфигураций 1С на соответствие стандартам и иным требованиям технического характера.

АПК существенно расширяет платформенную проверку конфигурации и выполняет статический анализ технического качества конфигураций в автоматическом режиме, не требуя их запуска

Team Foundation Server

Продукт корпорации Microsoft

Представляет собой комплексное решение, объединяющее в себе систему управления версиями, сбор данных, построение отчётов, отслеживание статусов и изменений по проекту и предназначенное для совместной работы над проектами по разработке программного обеспечения

xUnitFor1С

Набор инструментов для выполнения тестирования для 1С

Простой и мощный фреймворк для тестирования в 1С. Позволяет тестировать в разных режимах обычное приложение, тонкий и толстый клиент управляемого приложения. Поддерживаются любые платформы 1С - от 8.2.17 до 8.3.5 и выше. Любые наборы тестов могут прогоняться в полностью автоматическом режиме. Автозапуск используется в различных build-серверах в системах Continuous Integration.

Таким образом, можно для любого решения найти удобный способ для совершенствования процесса разработки, используя DevOps и наиболее подходящий инструментарий.

Список литературы:

1. Михайлов, С. Е. 1С программирование как дважды два / С. Е. Михайлов. – СПб.: Тритон, 2013. – 173 с.
2. Радченко, М. Г., Хрусталева, Е. Ю. 1С: Предприятие 8.2. практическое пособие разработчика. Примеры и типовые приемы / М. Г. Радченко, Е. Ю. Хрусталева. – СПб.: 1С-Паблишинг, 2011. – 874 с.
3. Ажеронок, В. А., Габец, А. П., Радченко, М. Г. Профессиональная разработка в системе 1С: Предприятие 8 / В. А. Ажеронок, А. П. Габец, М. Г. Радченко. – СПб.: 1С-Паблишинг, 2013. – 704 с.
4. DevOps Glossary[Электронный ресурс],- Электрон. текстовые дан. – Режим доступа: <https://www.sumologic.com/glossary/devops/>, свободный.