

ОСНОВНЫЕ ЭТАПЫ И ВАЖНЫЕ МОМЕНТЫ ЖИЗНЕННОГО ЦИКЛА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Беляев Антон Игоревич

студент, кафедра телекоммуникационных систем Уфимского государственного технического авиационного университета, РФ, г. Уфа

Хардин Павел Евгеньевич

студент, кафедра электронной инженерии Уфимского государственного технического авиационного университета, РФ, г. Уфа

Юсупов Артур Разимович

студент, кафедра электромеханики Уфимского государственного технического авиационного университета, РФ, г. Уфа

Жизненный цикл программного обеспечения – это применение стандартных практик бизнеса к разработке программного обеспечения. Обычно жизненный цикл разделяют на шесть или восемь этапов [1]:

- Изначальное планирование. На этом этапе происходит предварительный подсчёт необходимых ресурсов, составление списка основных задач и сроков их выполнения. Также собираются мнения от потенциальных покупателей (или пользователей), экспертов в области применения программного обеспечения, менеджмента отдела продаж. Целью планирования является чёткое определение общей цели и конкретного функционала программного средства. Также планирование устанавливает границы, чтобы программное средство при разработке не “обрастало” изначально не установленным функционалом.
- Сбор требований. Является частью планирования и выполняется параллельно. Целью сбора требований является получение набора функционала, который должно иметь разрабатываемое программное средство.
- Дизайн и прототипирование. На этом этапе происходит моделирование того, как программное обеспечение будет работать. Этот этап включает в себя некоторые подэтапы:
 - Определение архитектуры - определяется язык программирования, применяемые практики разработки, общее устройство программного средства.
 - Дизайн пользовательского интерфейса - определяются способы взаимодействия пользователя с программным средством.
 - Определение платформы - определяется платформа, на которой будет использоваться программное средство: Android, Apple, Windows, Linux и т.д.
 - Прототипирование коммуникации - происходит определение методов связи программного средства с другими. Например, использование центрального сервера для обработки данных или применение иной архитектуры коммуникации.
 - Прототипирование системы безопасности - определяются меры безопасности, которые необходимо применить к программному средству: шифрование данных и трафика, защита учётных записей паролем и шифрование этих записей.
- Разработка - этап непосредственного создания программного обеспечения. Небольшой проект может быть создан одним разработчиком, в то время как крупный может разрабатываться несколькими командами параллельно.
- Написание документации - важный этап разработки. Могут создаваться несколько

видов документации: техническая документация для упрощения добавления нового функционала в будущем и поддержки программного средства, а также документация для пользователя, в которой описаны способы взаимодействия с пользовательским интерфейсом. Программное обеспечение без документации сложно в разработке, доработке, поддержке и использовании, поэтому документация является неотъемлемой частью разработки.

- Тестирование - происходит проверка работы функционала программного средства. Данная проверка может осуществляться как автоматически (с использованием средств автоматического тестирования) так и вручную. Тестирование является ключевым этапом разработки любого проекта любого масштаба.
- Развёртывание (доставка) - разработанное программное средство становится доступно его конечным пользователям. В случае программного обеспечения как продукта, результатом этого этапа может быть загрузка в интернет-магазины или размещение ссылки на скачивание. В случае программного обеспечения как услуги, развёртыванием является установка приложения на сервер, обрабатывающий запросы конкретных пользователей. В некоторых случаях развёртывание может быть комбинацией развёртывания программного обеспечения как продукта и как услуги.
- Поддержка - осуществляется сбор обратной связи от пользователей, отчётов об ошибках. В случае программного обеспечения как услуги, на этом этапе также осуществляется слежение за работой сервера, обеспечение безотказности. Также на этом этапе производится сбор различных метрик об использовании программного средства пользователями [2]. Некоторые из этих этапов разработки программного обеспечения могут быть применяться или не применяться в зависимости от специфики конкретного проекта.

Список литературы:

1. Erik Fogg. What Can and Should be Automated in Software Development? [Электронный ресурс]. URL: <https://prodperfect.com/blog/test-development/what-can-and-should-be-automated-in-software-development/> (дата обращения: 11.12.2021).
2. Goran Jevtic. What is SDLC? Phases of Software Development & Models [Электронный ресурс]. URL: <https://phoenixnap.com/blog/software-development-life-cycle> (дата обращения: 11.12.2021).