

УДК 002.6:37.016

EDN [YWITCD](#)



Разработка цифрового образовательного ресурса «Прототипирование и моделирование»

А.А. Шалтабаев*, Б. Кизат

Жетысуский университет имени И. Жансугурова, Талдыкорган, 040000,
Казахстан

*E-mail: altai_shaltabaev@mail.ru

Аннотация. Данная статья посвящена разработке цифрового образовательного ресурса «Прототипирование и моделирование». В данной статье рассмотрены типы, этапы, а также преимущества и недостатки прототипирования. В программной инженерии методология прототипа представляет собой модель разработки программного обеспечения, в которой прототип создается, тестируется, а затем при необходимости перерабатывается до тех пор, пока не будет получен приемлемый прототип. Представлены шесть шагов прототипирования, такие как 1) сбор и анализ требований, 2) быстрый дизайн, 3) создание прототипа, 4) первоначальная пользовательская оценка, 5) уточнение прототипа, 6) внедрение продукта и поддержка. Проанализированы типы моделей прототипирования: 1) Быстроразовые прототипы; 2) Эволюционный прототип; 3) Инкрементный прототип; 4) Экстремальный прототип. К тому же в статье рассмотрены модели прототипирования, в частности, спиральная и водопадная модель, и данные модели представлены в схемах. Отмечается, что регулярные встречи необходимы для своевременного выполнения проекта и предотвращения дорогостоящих задержек в рамках прототипирования. При этом можно определить отсутствующие функции, что помогает снизить риск сбоя, поскольку прототипирование также рассматривается и как действие по снижению риска в SDLC.

Ключевые слова: модели прототипирования, быстроразборные прототипы, эволюционный прототип, инкрементный прототип, экстремальный прототип, прототипирование.

Development of a digital educational resource "Prototyping and modeling"

A. Shaltabaev*, B. Kizat

Zhetysu University named after I. Zhansugurov, Taldykorgan, 040000, Kazakhstan

*E-mail: altai_shaltabaev@mail.ru

Abstract. This clause is dedicated to the exploitation of a digital instructional resources initiative "Prototyping and Modeling". This clause converses about the types, stages, likewise as the superiority and deprivation of prototyping. In software engineering, a archetype epistemology is a software development model in which a prototype is created, tested, so reworked as necessary until an satisfactory prototype is obtained. Six steps of prototyping such as 1) qualification gathering and analysis, 2) rapid design, 3) prototype creation, 4) original user evaluation, 5) prototype refinement, 6) consequence deed and support. classification of prototyping representation 1) accelerated Prototypes 2) Evolutionary Prototype 3) Incremental Prototype 4) Extreme Prototype. In addition, the article converses about prototyping representation that is, the spiral and waterfall models, and are shown in the diagrams. Regular meetings are indispensable to keep the project on time and avoid costly delays in the prototyping approach. Missing characteristic can be identified, which lend a hand reduce the risk of failure since prototyping is also advised a risk mitigation activity in the SDLC.

Keywords: prototyping models, quick-release prototypes, evolutionary prototype, incremental prototype, extreme prototype, prototyping.

1. Введение

Модель прототипирования — это модель разработки программного обеспечения, в которой прототип создается, тестируется и перерабатывается до тех пор, пока не будет получен удовлетворительный прототип. Модель также формирует базу для создания окончательной сборки программного обеспечения. Это лучше всего проявляется в сценариях, где требования проекта не известны в деталях. Это интерактивное средство экспериментов и ошибок, которые обязательно возникают при взаимодействии между разработчиком и клиентом.

Модель прототипирования — одна из наиболее часто используемых моделей жизненного цикла разработки программного обеспечения (модели SDLC). Эта модель используется, когда клиенты предварительно не знают пунктуальных требований к проекту. В этой модели прототип окончательного продукта сначала разрабатывается, тестируется и неоднократно улучшается в соответствии с отзывами и рекомендациями клиентов, пока не будет получен безусловно приемлемый вариант прототипа, который станет фундаментом для разработки окончательного продукта [1].

2. Постановка задачи

В этой модели процесса система частично вводится в тестовую эксплуатацию до или во время фазы анализа, что дает клиентам возможность увидеть продукт на ранних этапах жизненного цикла. Процесс начинается с опроса покупателей и разработки неполной бумажной модели высокого уровня. Этот документ используется для создания начального прототипа, поддерживающего только базовую работоспособность по желанию заказчика. Как только клиент обозначает проблемы, прототип дорабатывается для их устранения. Процесс длится до тех пор, пока пользователь не подтвердит успешность прототипа и не найдет рабочую модель удовлетворительной.

Рассмотрим четыре типа моделей прототипирования: быстроразборный прототип, эволюционный прототип, инкрементный прототип, экстремальный прототип.

2.1. Быстроразборный прототип

Быстроразборный прототип базируется на предварительном требовании. Он быстро разрабатывается, чтобы показать, как требование будет выглядеть визуально. Отзывы клиентов ориентируют записывать изменения в требование, и прототип организовывается снова, пока требование не станет базовым. В этом способе

подготовленный прототип будет отброшен и не будет частью абсолютно общепринятого прототипа. Этот метод общеплезен для изучения мыслей и получения моментальной обратной связи по условиям клиентов [2].

2.2. Эволюционное прототипирование

Здесь подготовленный прототип постепенно улучшается на основе отзывов клиентов, пока не будет абсолютно принят. Это поможет сэкономить время, а также усилия. Это связано с тем, что разработка прототипа с нуля для каждого взаимодействия в процессе разработки иногда может быть очень утомительной. Эта модель полезна для проекта, в котором используется новая технология, которая еще недостаточно изучена. Он также используется для сложных проектов, где каждый функционал должен быть проконтролирован не один раз. Это полезно, когда требование нестабильно или не совсем понятно на начальном этапе.

2.3. Инкрементное прототипирование

При поэтапном прототипировании конечный продукт разбивается на несколько небольших прототипов и они разрабатываются индивидуально. В конце процесса, все созданные прототипы объединяются в единый продукт. Этот метод полезен для сокращения времени обратной связи между пользователем и командой разработчиков приложения.

2.4. Экстремальное прототипирование

Экстремальный метод прототипирования в основном используется для веб-разработки. Он состоит из трех последовательных этапов:

- Базовый прототип со всеми существующими страницами представлен в формате HTML.
- Имитация обработки данных с использованием слоя сервисов-прототипов.
- Сервисы реализованы и интегрированы в окончательный прототип.

Сформулируем важные плюсы/преимущества использования моделей прототипирования:

- Пользователи активно участвуют в разработке. Таким образом, ошибки могут быть выявлены на начальном шаге жизненного цикла разработки программного обеспечения.
- Можно выявить отсутствующую функциональность, что помогает уменьшить риск сбоя, поскольку прототипирование вдобавок рассматривается как активность по уменьшению риска.

- Помогает членам команды продуктивно общаться.
- Удовлетворенность заказчика гарантируется, потому что пользователь может осуществить апробацию продукта на очень ранней стадии.
- Вероятность отказа программного обеспечения будет существенно снижена.
- Быстрая обратная связь с пользователями поможет найти наилучшие рекомендации для совершенствования разрабатываемого программного обеспечения.
- Позволяет пользователю определить, соответствует ли программный код спецификации программного обеспечения.
- Это помогает найти отсутствующую работоспособность в системе.
- Описывает трудоемкие или трудные функции.
- Поощряет новинки и гибкий дизайн.
- Это элементарная модель, следовательно, ее легко понять.
- Нет необходимости в специализированных профессионалах для создания концепции модели.
- Прототип служит фундаментом для получения спецификации системы.
- Прототип помогает лучше понять требования клиента.
- Прототипы можно менять и даже отбрасывать неудовлетворительные.
- Прототип вдобавок служит основанием для эксплуатационных спецификаций.
- Прототипы могут рекомендовать раннюю подготовку будущих пользователей для работы с программной системой [3].

Однако, следует выделить и ряд важных минусов/недостатков моделей прототипирования:

- Прототипирование является медленным и трудоемким процессом.
- Стоимость разработки прототипа — это пустая трата денег, так как прототип в конечном итоге утилизируется.
- Прототипирование может стимулировать чрезмерные запросы на разные дополнительные изменения.
- Иногда клиенты могут не проявлять желание участвовать в интеракционном цикле в течение более длительного периода времени.
- Может быть слишком много вариаций в требованиях к программному обеспечению, когда заказчик каждый раз снова оценивает прототип.

- Плохое документирование, так как требования клиентов меняются.
- Разработчикам программного обеспечения очень трудно учесть все изменения, требуемые клиентами.
- Увидев ранний вариант прототипа, заказчик может подумать, что реальный продукт ему скоро предоставят в готовом виде.
- Клиент может потерять интерес к конечному продукту, если его не устроит первоначальный прототип.
- Разработчики, которые хотят быстро создавать прототипы, могут в конечном итоге создать нестандартные решения для разработки ПО [4].

3. Этапы модели прототипирования: модели водопада и спирали

Этапы модели прототипирования (рисунок 1):

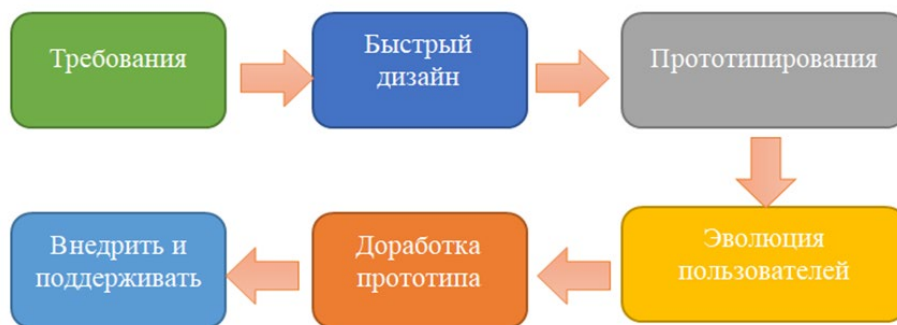


Рисунок 1. Этапы модели прототипирования.

Модель прототипирования имеет следующие шесть этапов SDLC:

- Шаг 1: Сбор и анализ требований.

Модель прототипирования начинается с анализа требований. На этом этапе детально определяются требования к системе. В ходе этого процесса пользователи системы опрашиваются, чтобы узнать, чего они ожидают от системы.

- Шаг 2: Быстрый дизайн.

Второй этап – предварительный проект или экспресс-дизайн. На этом этапе создается простой проект системы. Однако, это не законченная конструкция. Она дает пользователю краткое представление о системе. Быстрый дизайн помогает в разработке прототипа.

- Шаг 3: Создание прототипа.

На этом этапе фактический прототип разрабатывается на основе информации, полученной в результате быстрого проектирования. Это небольшая рабочая модель необходимой системы.

- Шаг 4: Первоначальная пользовательская оценка.

На этом этапе предлагаемая система представляется клиенту для первоначальной оценки. Он помогает выяснить сильные и слабые стороны работающей модели. Замечания и предложения собираются от заказчика и передаются разработчику.

- Шаг 5: Уточнение прототипа.

Если пользователя не устраивает текущий прототип, то необходимо доработать прототип в соответствии с отзывами и предложениями пользователя. Эта фаза не будет завершена до тех пор, пока не будут выполнены все требования, указанные пользователем. Как только пользователь удовлетворен разработанным прототипом, то на основе утвержденного окончательного прототипа разрабатывается окончательная система.

- Шаг 6: Внедрение продукта и поддержка.

Как только окончательная система разработана на основе окончательного прототипа, она тщательно тестируется и внедряется в производство. Система проходит плановое техническое обслуживание, чтобы свести к минимуму время простоя и предотвратить масштабные сбои [5].

3.1. Прототипирование: модели водопада и спирали

Прототип — это пример реализации системы. Это обеспечивает ограниченные и основные функциональные возможности предлагаемой системы (движение по кругу). Прототип построен на более ранних версиях (рисунок 2).

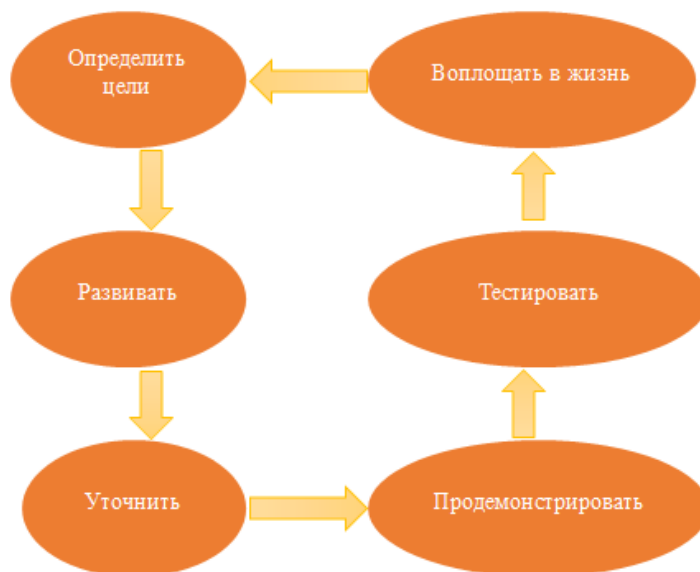


Рисунок 2. Прототипирование.

Водопадная модель — это традиционная модель жизненного цикла, в которой фазы жизненного цикла выполняются последовательно: (подобно водопаду). Отличие водопадной модели заключается в том, что каждый этап полностью завершается до того, как вы начнете следующий. Сложнее изменить то, что не было обнаружено на этапе вашей концепции [6] (см. рисунок 3).



Рисунок 3. Водопадная модель.

Спиральная модель — это генератор моделей процессов, управляемых рисками, для программных проектов. Основываясь на уникальных шаблонах рисков конкретного проекта, спиральная модель помогает команде принять элементы одной или нескольких

моделей процессов, таких как инкрементное, каскадное или эволюционное прототипирование (рисунок 4).



Рисунок 4. Спиральная модель.

4. Выводы

В заключении отметим, что в данной статье обоснована важность прототипирования и моделирования, даны аспекты, в которых заключается их преимущество, а так же обозначены их недостатки. Пошагово рассмотрены этапы и процедуры прототипирования, что проиллюстрировано схемами на рисунках.

Список литературы

1. Гайсина, С.В. Робототехника. 3D-моделирование и прототипирование в дополнительном образовании / С.В. Гайсина. – М.: Каро, 2017. – 445 с.
2. Гибсон, Я. Технологии аддитивного производства Трехмерная печать, быстрое прототипирование и прямое цифровое производство / Я. Гибсон, Д. Розен, Б. Стакер. – Москва: СИНТЕГ, 2016. – 970 с.
3. Bennett, M.J., Modeling Radar Countermeasure Systems / M.J. Bennett // Defense Computing, JulyAugust. – 1989.
4. Аббасов, И.Б. Компьютерное моделирование в промышленном дизайне / И.Б. Аббасов. – М.: ДМК, 2013. – 92 с.
5. Акаев, А.А. Моделирование и прогнозирование глобального, регионального и национального развития / А.А. Акаев, А.В. Коротаев, Г.Г. Малинецкий. – М.: КД Либроком, 2012. – 488 с

6. Boehm, B.W. A Spiral Model of Software Development and Enhancement / B.W. Boehm // IEEE Computer. – 1988. – P. 61.