

ПОСТАНОВКА НА РЕАЛИЗАЦИЮ КАК ЗАВЕРШАЮЩИЙ ЭТАП ПРОЕКТИРОВАНИЯ МЕНЕДЖЕРА ЗАДАЧ

Игнатенко А.Д.

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

Научный руководитель: Алексеев В.Ф. – канд. техн. наук, доцент, доцент кафедры ПИКС

Аннотация. Изучены возможности и необходимость постановки на реализацию в проектировании менеджера задач.

Ключевые слова: реализация, разработка, менеджер задач, постановка на реализацию.

Введение. Разработка любой программы состоит из нескольких этапов, грамотная реализация которых является обязательным условием для получения хорошего результата. Четкое следование выверенным временем этапам разработки программного обеспечения становится основополагающим критерием для занимающихся созданием ПО компаний и их заказчиков, заинтересованных в получении превосходно выполняющей свои функции программы [1–6].

Самым первым этапом разработки ПО называется процедура проведения анализа выдвинутых заказчиком требований к создаваемому ПО, чтобы определить ключевые цели и задачи конечного продукта. В рамках этой стадии максимально эффективно формулируются предъявляемые к ПО требования. Результатом проведенного анализа становится формирование основного регламента, на который будет опираться исполнитель в своей работе – технического задания на разработку программного обеспечения. ТЗ должно полностью описывать поставленные перед разработчиком задачи и охарактеризовать конечную цель проекта в понимании заказчика.

Следующий ключевой этап в разработке программного обеспечения – стадия проектирования, то есть моделирования теоретической основы будущего продукта [1].

На этом этапе и происходит знакомство с понятием «постановка на реализацию», которое с первого взгляда может быть похоже на понятие «техническое задание». Однако далее в данной статье эти понятия будут разграничены, а также будет более подробно рассмотрена постановка на реализацию в разработке менеджера задач.

Основная часть. Процесс создания программного обеспечения достаточно сложен и трудоемок, так как результатом должен быть продукт, не только полностью удовлетворяющий требованиям пользователя, но и достаточно функциональный в рамках обозримого будущего. В связи с этим весь процесс проектирования программ разделен на определенные этапы, которые позволяют осуществлять всесторонний контроль и отслеживание процесса разработки программного обеспечения.

На сегодняшний день не существует универсального процесса разработки программного обеспечения – набора методик, правил и предписаний, подходящих для программного обеспечения любого вида, для любых компаний, для команд любой национальности. Каждый текущий процесс разработки, осуществляемый некоторой командой в рамках определенного проекта, имеет большое количество особенностей и индивидуальностей. Однако целесообразно перед началом проекта спланировать процесс работы, определив роли и обязанности в команде, рабочие продукты (промежуточные и финальные), порядок участия в их разработке членов команды и так далее.

Процесс создания программного обеспечения не является однородным. Тот или иной метод разработки программного обеспечения, как правило, определяет некоторую динамику развертывания тех или иных видов деятельности, то есть, определяет модель процесса (*process model*).

Техническое задание – это не то же самое, что и постановка на реализацию. Техническое задание – это результат обработки исходных требований, их уточнения и перевода на системный/технический уровень.

Постановка задачи на реализацию – это описание способа реализации исходных требований, технического задания, архитектурного решения, изложение требований к устройству спроектированного решения (на этом этапе исходные требования уже обработаны).

Постановка на реализацию выполняется в рамках проектирования перед реализацией (рисунок 1) [6].

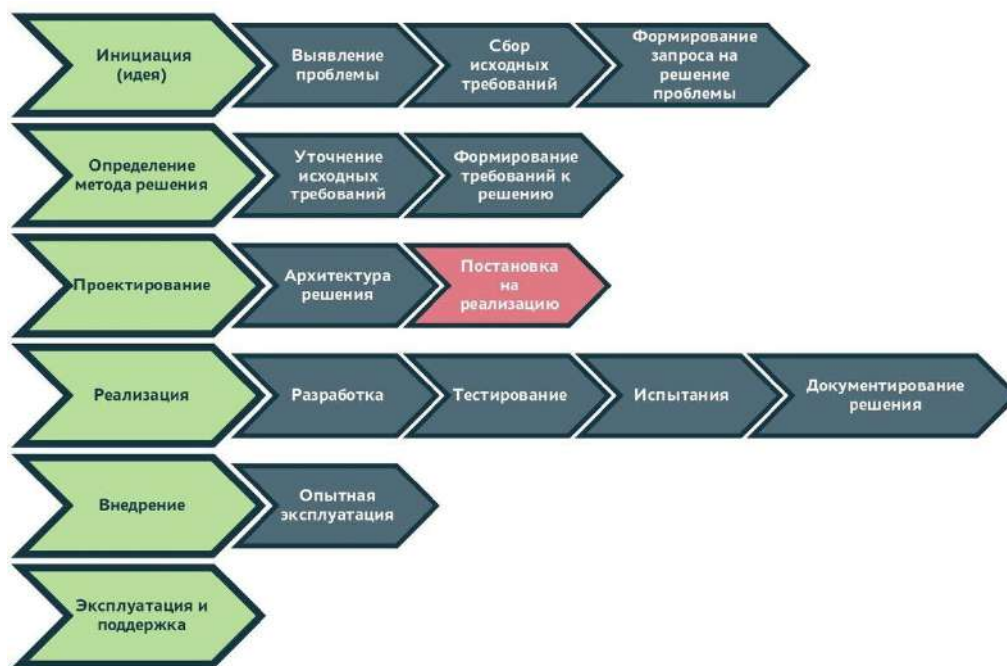


Рисунок 1 – Процесс реализации ИТ-проекта от инициации идеи до внедрения и эксплуатации готового ИТ-решения

Техническое задание разрабатывается на этапах инициации и определения метода решения бизнес-задачи. Здесь идет выявление проблемы, сбор исходных требований, формирование какого-то запроса на решение этой проблемы. А постановка на реализацию описывает, как эти функции и продукты будут реализованы. Постановка описывает конкретную функцию, модуль, что-то максимально локализованное. Объектами постановки могут быть отдельные компоненты, модули или функции – в зависимости от того, как мы декомпозировали функциональную структуру системы/решения в техническом задании.

Постановка на реализацию помогает решить следующие цели: определить границы проекта, защититься от их изменений, а также зафиксировать критерии успешного результата.

Постановка на реализацию может иметь следующую структуру:

- контекст задачи;
- ключевые источники информации;
- заинтересованные стороны;
- критерии приемки результата;
- нефункциональные требования и ограничения;
- описание решения.

Контекст задачи. Краткая информация о ситуации и проблемах, из-за которых возникла стоящая перед вами задача по автоматизации. Данная часть постановки – это, по сути, срез бизнес-требований, которые были собраны бизнес-аналитиком на предыдущем этапе.

Контекст задачи помогает:

- сформировать «мостик» между заказчиком и исполнителем,
- минимизировать риск того, что решение будет оторвано от реалий будущего использования,
- разработчикам придумать лучшее решение,
- тестировщики могут разработать максимально приближенные к реальному использованию тест-кейсы,
- команде погрузиться в предметную область и прокачать экспертизу в предметной области,
- аналитику в трассировке требований на проблемы бизнеса.

Также данный раздел может содержать: описание функций бизнес-языком, описание бизнес-процесса, бизнес-требования и перечень проблем бизнеса.

Ключевые источники информации. В этом разделе постановки речь идет о едином перечне источников информации, описание которых снижает риск использования недостоверной информации. Например, глоссарий необходим, чтобы заказчик, бизнес и команда оперировали одними и теми же согласованными терминами.

Другие примеры ключевых источников информации:

- описание внешнего или ранее реализованного *API* (например, *API* компании-перевозчика);
- *HLD (high-level design)*, описание архитектуры);
- стандарты заказчика, если речь идет про передачу листинг кода, а также когда есть свои определенные стандарты его оформления;
- ссылка на памятку по чтению постановки.

Заинтересованные стороны. Перечень людей, которые будут влиять на принятие решений и от которых зависит успех реализации.

Существуют разные роли, в которых выступают заинтересованные стороны по отношению к постановке. Например, бизнес-роли и проектные роли.

Бизнес-роль важна прежде всего на уровне сбора требований, т.к. отвечает за содержательную их часть. Бывает и так, что требования дособирают на этапе постановки, потому что иногда все то, что было наработано в техническом задании, к моменту реализации уже не совсем актуально, потому что бизнес меняется, и это нормально.

Наличие в постановке реестра заинтересованных сторон позволяет команде разработки на этапе выполнения задачи понимать, с какой из заинтересованных сторон нужно коммуницировать по какой конкретной части решения.

Критерии приемки результатов. Критерии, выполнение которых может говорить о том, что решение реализовано (definition of done постановки).

В широком смысле критериями приемки могут выступать:

- перечень исходных требований, реализуемых решением;
- сформированный перечень функций решения и результатов их выполнения;
- *user story*, и т.д.

Критерии приемки помогают: дать понимание границ реализации, обеспечить полноту реализации, выступают чек-листом приемки (аналитиком при авторском надзоре и заказчиком).

Нефункциональные требования и ограничения. Это требования к видам обеспечения и ограничений реализации. Нefункциональные требования нужны, чтобы их учитывать при принятии решения в рамках постановки. Ошибки при их проработке в последствии могут привести к очень серьезным проблемам и полной неработоспособности решения.

К основным требованиям относятся:

- производительность (использование в нескольких филиалах),
- доступность (24/7),
- масштабируемость (количество сотрудников),

– требования к видам обеспечения - документация и юридические аспекты,
– переходные требования - требования, позволяющие без потерь перейти к новой версии автоматизации.

Описание решения. Это основная часть постановки, описывающая способ и границы реализации.

Описание решения может содержать: сценарии использования (*UseCase*), информационные модели, статусные модели, алгоритмы, макеты интерфейсов (UX/UI), компонентные схемы и описание интеграций.

Диаграммы для *UseCase* ограничены, т.е. фиксируют конкретный набор функциональности, который, в свою очередь, помогает [2]:

- аналитику ничего не забыть,
- разработчику реализовать данный сценарий,
- заказчику увидеть сценарий его будущего взаимодействия с системой,
- тестировщикам создать тест кейсы.

Заключение. Таким образом, постановка на реализацию является важным этапом проектирования менеджеров задач, так как позволяет предотвратить неожиданные результаты в процессе реализации, а также зафиксировать границы решения для сохранения их от изменений.

Список литературы

1. *Этапы разработки программного обеспечения [Электронный ресурс]. – Режим доступа: <https://ru.intechcore.com/stages-software-development/>. – Дата доступа: 12.04.2022.*
2. *Постановка задачи на реализацию [Электронный ресурс]. – Режим доступа: <https://systems.education/implementation-spec?>. – Дата доступа: 12.04.2022.*
3. *Алексеев, В. Ф. Информационная поддержка управления инновационной деятельностью предприятия / В. Ф. Алексеев, Д. В. Лихачевский, В. В. Хорошко // BIG DATA and Advanced Analytics = BIG DATA и анализ высокого уровня: сб. материалов VI Международ. науч.-практ. конф., Минск, 20-21 мая 2020 года: в 3 ч. Ч. 3 / редкол.: В. А. Бозуш [и др.]. – Минск : Бестпринт, 2020. – С. 412–417.*
4. *Алексеев, В. Ф. Разработка онлайн платформы оценки и финансирования инновационных проектов / В. Ф. Алексеев, Д. В. Лихачевский, Г. А. Пискун // BIG DATA Advanced Analytics: collection of materials of the fourth international scientific and practical conference, Minsk, Belarus, May 3 – 4, 2018 / editorial board: M. Batura [etc.]. – Minsk, BSUIR, 2018. – P. 398 – 404.*
5. *Алексеев, В. Ф. Программное средство для оценки финансового состояния предприятия / В. Ф. Алексеев, Г. А. Пискун, Д. В. Лихачевский // BIG DATA Advanced Analytics: collection of materials of the fourth international scientific and practical conference, Minsk, Belarus, May 3 – 4, 2018 / editorial board: M. Batura [etc.]. – Minsk, BSUIR, 2018. – P. 308 – 312.*
6. *Школа системного анализа // Постановка задачи в ИТ-проектах [Электронный ресурс]. – 2022. – Режим доступа: <https://systems.education/implementation-spec> – Дата доступа: 10.04.2022.*

UDC 004.4

SETTING UP FOR IMPLEMENTATION AS THE FINAL STAGE OF DESIGNING A TASK MANAGER

Ignatenko A.D.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Alexeev V.F. – PhD, assistant professor, associate professor of the department of ICSD

Annotation. The possibilities and necessity of setting up for implementation in the design of the task manager are studied.

Keywords. implementation, development, task manager, setting for implementation.