

**Информационная система
«Агрегированные расписания»
Описание жизненного цикла
программного обеспечения**

Оглавление

1	Процессы жизненного цикла программного обеспечения	4
1.1	Общие сведения	4
1.2	Процессы внедрения программных средств	4
1.2.1	Основной процесс внедрения	4
1.2.2	Процесс анализа требований к программным средствам	4
1.2.3	Процессы проектирования программных средств	5
1.2.4	Процесс конструирования программных средств	5
1.2.5	Процесс комплексирования программных средств	5
1.2.6	Процесс квалификационного тестирования программных средств	6
1.3	Процессы поддержки программных средств	6
1.3.1	Процесс управления документацией программных средств	6
1.3.2	Процесс управления конфигурацией программных средств	7
1.3.3	Процесс обеспечения гарантии качества программных средств	7
1.3.4	Процесс верификации программных средств	7
1.3.5	Процесс валидации программных средств	8
1.3.6	Процесс ревизии программных средств	8
1.3.7	Процесс аудита программных средств	8
1.3.8	Процесс решения проблем в программных средствах	9
2	Схема реализации функционала	10
2.1	Проработка и постановка задачи аналитиками	10
2.2	Проработка и реализация задач командами разработки	11
3	Устранение неисправностей программного обеспечения	13
4	Совершенствование программного обеспечения	14

Аннотация

Данный документ содержит:

- описание процессов, обеспечивающих поддержание жизненного цикла программного обеспечения (ПО);
- устранение неисправностей, выявленных в ходе эксплуатации программного обеспечения;
- совершенствование программного обеспечения (ПО);
- информацию о персонале, необходимом для обеспечения такой поддержки.

Данный документ описывает жизненный цикл информационной системы «Агрегированные расписания».

Перечень сокращений

Сокращение	Описание
ПО	Программное обеспечение
Базовая линия (baseline)	Спецификация или продукт, которые были официально рассмотрены и согласованы с тем, чтобы впоследствии служить основой для дальнейшего развития, и которые могут быть изменены только посредством официальных и контролируемых процедур изменения
Жизненный цикл (life cycle)	Развитие системы, продукта, услуги, проекта или других изготовленных человеком объектов, начиная со стадии разработки концепции и заканчивая прекращением применения
Квалификационное тестирование (qualification testing)	Тестирование, проводимое разработчиком и санкционированное приобретающей стороной (при необходимости), с целью демонстрации того, что программный продукт удовлетворяет спецификациям и готов для применения в заданном окружении или интеграции с системой, для которой он предназначен
Комплексирование (integration)	Объединение системных элементов (включая составные части технических и программных средств, ручные операции и другие системы, при необходимости) для производства полной системы, которая будет удовлетворять системному проекту и ожиданиям заказчика, выраженным в системных требованиях
Конструирование (constraction)	Создание исполняемых программных блоков, которые должным образом отражают проектирование программных средств
Система	Информационная система «Агрегированные расписания»
СТП	Служба технической поддержки
ТЗ	Техническое задание

1 Процессы жизненного цикла программного обеспечения

1.1 Общие сведения

Основные процессы жизненного цикла программных средств, входящих в состав информационной системы «Агрегированные расписания», описаны в данном разделе в соответствии с ГОСТ Р ИСО/МЭК 12207-2010.

1.2 Процессы внедрения программных средств

1.2.1 Основной процесс внедрения

В результате успешного осуществления основного процесса внедрения (реализации) программных средств:

- определяется стратегия внедрения;
- определяются ограничения по технологии реализации проекта;
- изготавливается программная составная часть;
- программная составная часть упаковывается и хранится в соответствии с соглашением о её поставке.

1.2.2 Процесс анализа требований к программным средствам

В результате успешного осуществления процесса анализа требований к программным средствам:

- определяются требования к программным элементам системы и их интерфейсам;
- требования к программным средствам анализируются на корректность и тестируемость;
- осознается воздействие требований к программным средствам на среду функционирования;
- устанавливается совместимость и прослеживаемость между требованиями к программным средствам и требованиями к системе;
- определяются приоритеты реализации требований к программным средствам;
- требования к программным средствам принимаются и обновляются по мере необходимости;
- оцениваются изменения в требованиях к программным средствам по стоимости, графикам работ и техническим воздействиям;
- требования к программным средствам воплощаются в виде базовых линий и доводятся до сведения заинтересованных сторон.

1.2.3 Процессы проектирования программных средств

В результате успешной реализации процесса проектирования архитектуры программных средств:

- разрабатывается проект архитектуры программных средств и устанавливается базовая линия, описывающая программные составные части, которые будут реализовывать требования к программным средствам;
- определяются внутренние и внешние интерфейсы каждой программной составной части;
- устанавливаются согласованность и прослеживаемость между требованиями к программным средствам и программному проекту.

В результате успешного осуществления процесса детального проектирования программных средств:

- разрабатывается детальный проект каждого программного компонента, описывающий создаваемые программные модули;
- определяются внешние интерфейсы каждого программного модуля;
- устанавливается совместимость и прослеживаемость между детальным проектированием, требованиями и проектированием архитектуры.

1.2.4 Процесс конструирования программных средств

В результате успешного осуществления процесса конструирования программных средств:

- определяются критерии верификации для всех программных блоков относительно требований;
- изготавливаются программные блоки, определенные проектом;
- устанавливается совместимость и прослеживаемость между программными блоками, требованиями и проектом;
- завершается верификация программных блоков относительно требований и проекта.

1.2.5 Процесс комплексирования программных средств

В результате успешного осуществления процесса комплексирования программных средств:

- разрабатывается стратегия комплексирования для программных блоков, согласованная с программным проектом и расположенными по приоритетам требованиями к программным средствам;

- разрабатываются критерии верификации для программных составных частей, которые гарантируют соответствие с требованиями к программным средствам, связанными с этими составными частями;

- программные составные части верифицируются с использованием определенных критериев;

- программные составные части, определенные стратегией комплексирования, изготавливаются;

- регистрируются результаты комплексного тестирования;

- устанавливаются согласованность и прослеживаемость между программным проектом и программными составными частями;

- разрабатывается и применяется стратегия регрессии для повторной верификации программных составных частей при возникновении изменений в программных блоках (в том числе в соответствующих требованиях, проекте и кодах).

1.2.6 Процесс квалификационного тестирования программных средств

В результате успешного осуществления процесса квалификационного тестирования программных средств:

- определяются критерии для комплектованных программных средств с целью демонстрации соответствия с требованиями к программным средствам;

- комплектованные программные средства верифицируются с использованием определенных критериев;

- записываются результаты тестирования;

- разрабатывается и применяется стратегия регрессии для повторного тестирования комплектованного программного средства при проведении изменений в программных составных частях.

1.3 Процессы поддержки программных средств

1.3.1 Процесс управления документацией программных средств

В результате успешного осуществления процесса управления документацией программных средств:

- разрабатывается стратегия идентификации документации, которая реализуется в течение жизненного цикла программного продукта или услуги;

- определяются стандарты, которые применяются при разработке программной документации;

- определяется документация, которая производится процессом или проектом;

- указываются, рассматриваются и утверждаются содержание и цели всей документации;
- документация разрабатывается и делается доступной в соответствии с определенными стандартами;
- документация сопровождается в соответствии с определенными критериями.

1.3.2 Процесс управления конфигурацией программных средств

В результате успешного осуществления процесса управления конфигурацией программных средств:

- разрабатывается стратегия управления конфигурацией программных средств;
- составные части, порождаемые процессом или проектом, идентифицируются, определяются и вводятся в базовую линию;
- контролируются модификации и выпуски этих составных частей;
- обеспечивается доступность модификаций и выпусков для заинтересованных сторон;
- регистрируется и сообщается статус составных частей и модификаций;
- гарантируются завершенность и согласованность составных частей;
- контролируются хранение, обработка и поставка составных частей.

1.3.3 Процесс обеспечения гарантии качества программных средств

В результате успешного осуществления процесса гарантии качества программных средств:

- разрабатывается стратегия обеспечения гарантии качества;
- создается и поддерживается свидетельство гарантии качества;
- идентифицируются и регистрируются проблемы и (или) несоответствия с требованиями;
- верифицируется соблюдение продукцией, процессами и действиями соответствующих стандартов, процедур и требований.

1.3.4 Процесс верификации программных средств

В результате успешного осуществления процесса верификации программных средств:

- разрабатывается и осуществляется стратегия верификации;
- определяются критерии верификации всех необходимых программных рабочих продуктов;
- выполняются требуемые действия по верификации;
- определяются и регистрируются дефекты;

– результаты верификации становятся доступными заказчику и другим заинтересованным сторонам.

1.3.5 Процесс валидации программных средств

В результате успешного осуществления процесса валидации программных средств:

- разрабатывается и реализуется стратегия валидации;
- определяются критерии валидации для всей требуемой рабочей продукции;
- выполняются требуемые действия по валидации;
- идентифицируются и регистрируются проблемы;
- обеспечиваются свидетельства того, что созданные рабочие программные продукты пригодны для применения по назначению;
- результаты действий по валидации делаются доступными заказчику и другим заинтересованным сторонам.

1.3.6 Процесс ревизии программных средств

В результате успешного осуществления процесса ревизии программных средств:

- выполняются технические ревизии и ревизии менеджмента на основе потребностей проекта;
- оцениваются состояние и результаты действий процесса посредством ревизии деятельности;
- объявляются результаты ревизии всем участвующим сторонам;
- отслеживаются для закрытия позиции, по которым необходимо предпринимать активные действия, выявленные в результате ревизии;
- идентифицируются и регистрируются риски и проблемы.

1.3.7 Процесс аудита программных средств

В результате успешного осуществления процесса аудита программных средств:

- разрабатывается и осуществляется стратегия аудита;
- согласно стратегии аудита, определяется соответствие отобранных рабочих программных продуктов и (или) услуг или процессов, соответствующих требованиям, планам и соглашениям;
- аудиты проводятся соответствующими независимыми сторонами;
- проблемы, выявленные в процессе аудита, идентифицируются, доводятся до сведения ответственных за корректирующие действия и затем решаются.

1.3.8 Процесс решения проблем в программных средствах

В результате успешной реализации процесса решения проблем в программных средствах:

- разрабатывается стратегия менеджмента проблем;
- проблемы регистрируются, идентифицируются и классифицируются;
- проблемы анализируются и оцениваются для определения приемлемого решения (решений);
- выполняется решение проблем;
- проблемы отслеживаются вплоть до их закрытия;
- известно текущее состояние всех зафиксированных проблем.

2 Схема реализации функционала

Процесс создания, разработки задач, который в последствии дает информацию о трудозатратах производственного персонала показан на рисунке 1.

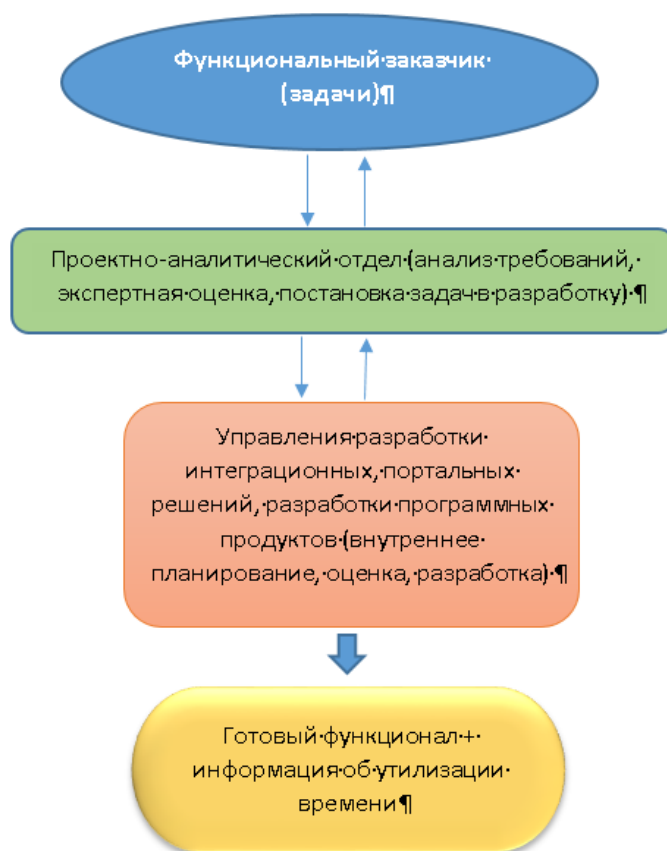


Рисунок 1 – Схема реализации функционала

2.1 Проработка и постановка задачи аналитиками

Руководитель проектно-аналитического управления (ПАУ) получает задачи от функционального заказчика (ФЗ) и назначает для каждой задачи аналитика на проработку. Аналитик проводит первичный анализ требований, на предмет того, что необходимо для выполнения задачи. Далее происходит первый этап планирования функционала для дальнейшей аллокации по командам разработки с участием руководителей управлений и Директора по разработке программного обеспечения. При необходимости, если данных недостаточно, аналитик ПАУ уточняет требования у ФЗ. После уточнения всех требований у ФЗ (если этот шаг присутствовал) аналитик обращается в соответствующие команды разработки для получения экспертной оценки по трудозатратам на разработку задачи и определению подхода к разработке. Далее аналитик ПАУ согласовывает концептуальное решение задачи совместно с ФЗ.

В случае несогласования решения со стороны ФЗ, аналитик ПАУ повторно обращается в соответствующие команды разработки для переоценки задачи.

После согласования принятого решения по реализации задачи с ФЗ, аналитик ПАУ производит оценку данной задачи:

- проработка необходимых изменений в существующих программных продуктах;
- декомпозиция задачи, структурирование материала до уровня ТЗ;
- постановка в трекерах;
- оценка на написание сопутствующей технической документации (при необходимости).

На аналитику закладывается 30 - 50% от итоговой оценки задачи на разработку и зависит от сложности задачи.

После валидации данной итоговой оценки задачи руководителем ПАУ, данная оценка согласуется с Техническим Директором, и, в случае подтверждения, направляется в сторону ФЗ для согласования. Задача уходит в разработку.

2.2 Проработка и реализация задач командами разработки

После проработки задач аналитиками, задачи распределяются в соответствующие управления разработки Директором по разработке программного обеспечения.

Перед началом реализации, все задачи проходят внутреннее планирование в командах разработки, где происходит оценка задачи разработчиками и тестировщиками, её декомпозиция при необходимости.

Руководитель соответствующего структурного подразделения назначает исполнителей на задачи: разработчиков и тестировщиков.

Задачи, после постановки, попадают в бэклог проекта команды разработки. Там задачи приоритизируются совместно с менеджером продукта.

Разработка происходит отрезками по четыре недели. Каждый из отрезков планируется по трудозатратам и приоритетности задач. После окончания проходит тестирование и формируется сборка для установки на продуктив.

Каждые четыре недели задачи проходят планирование в релиз. Предварительно они изучаются командой, проектируются и составляется план реализации. Все задачи оцениваются и берутся в работу, исходя из приоритетов и ёмкости команды на четыре недели.

После планирования и начала итерации, задачи распределяются на разработчиков. Для выполнения работы по задаче разработчик формирует отдельную фиче-ветку в репозитории от последней стабильной версии. В рамках фиче-ветки проходит разработка нового функционала.

После завершения, разработчик пишет краткое описание в задаче того, как была выполнена задача, и на что следует обратить внимание тестировщику. Затем переводит задачу на кодревью.

В рамках кодревью другой разработчик из команды или старший разработчик проводит проверку согласно принятым нормам написания кода в команде, логику и правильность алгоритма, проверяет наличие тестов на написанный функционал. По итогу проверки задача может быть отправлена на доработку.

После завершения кодревью задача переходит в тестирование. Тестировщик проводит все необходимые проверки согласно конкретной задаче. Пишет сценарии тестирования и регистрирует факты проверки. Если задача не проходит тестирование она может быть отправлена на доработку.

После успешного прохождения тестирования код задачи из фиче-ветки вносится в ветку релиза для формирования релиза.

После того как все задачи, запланированные на итерацию перенесены в ветку релиза, начинается создание сборки под релиз. Запускаются автотесты, и по успешному их прохождению формируется сборка. Сборка разворачивается на тестовом полигоне для прохождения регрессионного тестирования.

Согласно регресс плану тестирования команда тестирования выполняет полную проверку решения перед его установкой на предпродуктовый стенд.

Если регрессионное тестирование выявило проблемы или ошибки, разработчик выполняет внесение необходимых изменений в код и сборка формируется заново.

По итогу регрессионного тестирования руководитель тестирования пишет заключение, и принимается решение по установке релиза на предпродуктовый стенд.

На предпродуктовом стенде проходит короткое смоук (smoke) тестирование, и, если все работает корректно, стенд переключается на продуктивный полигон.

3 Устранение неисправностей программного обеспечения

Перечень этапов процесса устранения неисправностей программного обеспечения (ПО) приведено в п. 1.3.8 «Процесс решения проблем в программных средствах».

Штатный порядок работы ПО определяется эксплуатационной документацией, предоставляемой производителем ПО. Поддерживаемый ПО набор функций определяется требованиями ТЗ, утвержденного Заказчиком.

В случае обнаружения ошибок в работе ПО, которые являются нарушением требований ТЗ или противоречат порядку работы ПО, описанному в документации, администратор ПО должен направить заявку в службу технической поддержки (СТП) организации, проводившей работы по внедрению ПО. СТП организации, внедрившей ПО, проверяет, при необходимости уточняет полученную заявку и пытается выполнить ее, используя собственные ресурсы и знания.

В случае, если силами СТП организации, внедрившей ПО, выполнить заявку не удастся, то указанная организация обращается за помощью к производителю ПО. Производитель ПО проверяет наличие ошибки и рекомендаций по ее устранению в базе знаний технической поддержки (в соответствии с договорённостями между производителем и заказчиком).

После устранения неисправности разработчики ПО выпускают обновление к текущей версии ПО или включают исправление в следующую версию ПО. Информация о наличии обновления или новой версии ПО доводится до партнёров производителя ПО. В случае наличия у Заказчика контракта или договора на поддержку ПО, Заказчик имеет право на получение обновления ПО.

4 Совершенствование программного обеспечения

Работа по совершенствованию ПО включает в себя два основных направления:

- повышение качества и надежности ПО;
- актуализация перечня функций, поддерживаемых ПО.

В ходе постоянно проводимой работы по совершенствованию ПО используются хорошо зарекомендовавшие себя методы повышения качества и надежности ПО:

- совершенствование процесса разработки ПО – повышение качества ПО за счёт использования современных методик и инструментов разработки;
- совершенствование процесса тестирования ПО – обеспечение необходимой полноты покрытия.

Актуализация перечня функций, поддерживаемых ПО, включает в себя:

- добавление новых и изменение существующих функций в соответствии со стратегией развития ПО;
- добавление новых и изменение существующих функций по предложениям Заказчиков и партнёров производителя ПО;
- исключение устаревших функций.