

Процессы, обеспечивающие поддержание жизненного цикла
программного продукта

ЮНИВУЗ-3. Управление НИОКР

(в том числе устранение неисправностей, выявленных в ходе эксплуатации,
совершенствование и развитие программного комплекса, а также информация
о персонале, необходимом для осуществления такой поддержки)

на 12 листах



Содержание

Назначение документа	3
Понятия и определения	3
Некритическая ошибка	3
Критическая ошибка	3
Доработка	3
Расчет сложности и трудоемкости доработки	3
Совершенствование на основе доработки	3
Сопровождение продукта	4
Исправление ошибок	4
Доработка	6
Развитие продукта	6
Требования к специалистам для развертывания, поддержки и развития программного продукта ЮНИВУЗ-3	7
Роли и задачи	7
Требования к исполнителям ролей:	8
Разработчик UI	9
Администратор	9
Системный аналитик	9
Технический писатель	10
Приложение 1	11
Приложение 2	12



Назначение документа

Настоящий документ содержит описание процессов, обеспечивающих поддержание жизненного цикла программного продукта, а также процедуры сопровождения и развития программного продукта «ЮНИВУЗ-3. Управление НИОКР». Документ определяет правила взаимодействия с заказчиком при условии заключения соглашений о технической поддержке и без них; а также требования к совершенствованию программного продукта.

Фактический адрес размещения:

- инфраструктуры разработки: 141407, Российская Федерация, г. Москва, МКАД, 74й километр, владение 3.

- разработчиков и службы поддержки: 424003, Российская Федерация, Республика Марий Эл, город Йошкар-Ола, ул. Машиностроителей, д. 59, помещ. 1.

Понятия и определения

Некритическая ошибка

– ошибка функционирования программного кода продукта или интерфейсной части, которая не влияет на доступ к функциям, их выполнение и результат, при условии, что данная ошибка не вызвана действием (бездействием) пользователя, а результат выполнения функции может быть получен альтернативной функцией.

Критическая ошибка

– ошибка функционирования программного кода продукта или интерфейсной части, являющаяся причиной невозможности доступа пользователей к функциям программного продукта, или их невыполнение / выполнение с ошибкой, которые не вызваны неправильными действиями (бездействием) пользователя и не имеют альтернативного пути выполнения данной функции.

Доработка

– изменение и/или добавление функций в программный продукт и его функциональные контуры в соответствии с техническим заданием (включая все необходимые исходные данные), которое предоставляет Заказчик.

Расчет сложности и трудоемкости доработки

– расчет трудоемкости осуществляется Исполнителем на основании предоставленного технического задания (включая все необходимые исходные данные). Для расчета используется накопленный опыт в разработке, а также такие отраслевые подходы к расчету трудоемкости, как метод функциональных точек и COCOMO II.

Совершенствование на основе доработки

– доработка, выполненная для конкретного заказчика, может быть включена в функционал программного продукта в качестве совершенствования и



перспективной доработки. Условия такой доработки и ее распространения согласовываются с Заказчиком в процессе формирования соглашения о технической поддержке.

Сопровождение продукта

Сопровождение продукта осуществляется на основании договора на техническую поддержку, заключенного между Заказчиком (покупателем) и Исполнителем (ООО «ЮНИСИСТЕМС») и включает в себя:

- исправление ошибок, выявленных Заказчиком (критических и некритических),
- консультации специалистов Заказчика по вопросам внедрения и сопровождения продукта,
- доработки, по согласованию технического задания.

В рамках технической поддержки программного продукта «ЮНИВУЗ-3» оказываются следующие услуги:

- помощь в установке программного обеспечения;
- помощь в настройке и администрировании программного обеспечения;
- помощь в установке обновлений программного обеспечения;
- помощь в поиске и устранении проблем в случае некорректной установки обновления программного обеспечения;
- пояснение функционала программного обеспечения, помощь в эксплуатации;
- предоставление актуальной документации по установке/настройке/работе программного обеспечения.

Для коммуникаций по технической поддержке Заказчик и Исполнитель использует сайт технической поддержки <https://support.uni-systems.ru/>.

Исправление ошибок

Критические и некритические ошибки, выявленные в работе программного продукта, фиксируются исполнителем в системе управления проектами по обращению Заказчика. Сведения об ошибке должны содержать следующие параметры:

- Тип: Ошибка,
- Название: Краткое название ошибки,
- Описание:
 - ФИО ответственного сотрудника Заказчика - автора ошибки и координаты связи с ним.



- Ссылка из адресной строки браузера на web-страницу с ошибкой и/или идентификатор ошибки;
- Логин пользователя, сообщившего об ошибке (при необходимости).
- Подробное описание ошибки с указанием последовательности действий пользователя, приведших к ошибке, с предоставлением скриншотов.
- Указание на результат воспроизведения ошибки на тестовом сервере.
- Приоритет: низкий, обычный, высокий или критично.

Сроки исправления ошибок определяет специалист Исполнителя только после проведения диагностики и классификации ошибки, с учетом требований. Отсчет трудозатрат по устранению ошибки начинается с момента назначения специалистом Заказчика в системе управления проектами вновь поступившей задачи на устранение ошибки на конкретного исполнителя, или с момента запроса у Заказчика дополнительных данных по такой ошибке.

Ошибки с высоким приоритетом и со статусом «критично» исправляются в первую очередь по следующему алгоритму:

1. Воспроизведение ошибки на тестовом сервере, анализ возможных причин возникновения ошибки и утверждение статуса «ошибка» (выявленная заказчиком ошибка может не соответствовать критериям критической и некритической ошибки и может быть отнесена к задачам типа «доработка»).
2. Оценка трудозатрат на устранение ошибки и определение срока исправления;
3. Выработка решения по устранению ошибки, и, при необходимости, разработка постановки задачи на устранение ошибки.
4. Согласование с Заказчиком работ по устранению ошибки.
5. Реализация решения по устранению ошибки.
6. Внеочередное обновление модуля программного продукта, в котором была обнаружена ошибка (с учетом возможного попадания в тестовую версию иных доработок/исправлений с тем же приоритетом).

Ошибки с низким и обычным приоритетом исправляются в соответствии с планом доработок, утвержденным в техническом задании (Приложение 1), по следующему алгоритму:

1. воспроизведение ошибки на тестовом сервере, анализ возможных причин возникновения ошибки и утверждение статуса «ошибка» (выявленная заказчиком ошибка может не соответствовать критериям критической и некритической ошибки и может быть отнесена к задачам типа «доработка»).
2. оценка трудозатрат на устранение ошибки и определение срока исправления;
3. Выработка решения по устранению ошибки, и, при необходимости, разработка постановки задачи на устранение ошибки.
4. Согласование с Заказчиком работ по устранению ошибки.



5. Включение доработки по устранению ошибки в план доработок Исполнителя.
6. Реализация решения по устранению ошибки в соответствии с планом.
7. Плановое обновление модуля программного продукта, в котором была обнаружена ошибка (с учетом возможного попадание в тестовую версию иных доработок/исправлений с тем же приоритетом).
8. Автор ошибки обязан в срок до одного рабочего дня протестировать исправленную ошибку, и сообщить результаты проверки. По результатам ответственный Исполнитель переводит задачу в статус «Отклонена», а если ошибка не проявилась, в статус «Готово».

Доработка

Доработки продукта, предназначенные для адаптации программного продукта под бизнес-процессы и требования заказчика, выполняются по согласованию работ и формированию Исполнителем технического задания [Приложение 1], которое составляется на основании договора о технической поддержке. Согласование работ осуществляется путем передачи посредством e-mail, телефон от Заказчика Исполнителю перечня работ к исполнению, Исполнитель оценивает трудозатраты и передает техническое задание Заказчику (Приложение 1). После утверждения технического задания Заказчиком, Исполнитель приступает к исполнению работ.

Условия и требования к Исполнителю:

- Время реакции на заявку Заказчика не должно превышать 8-ти рабочих часов.
- Суммарное количество часов работы Исполнителя в месяц не должно превышать 60 часов.
- Минимальный объем работ по заявке составляет один час.

Условия и требования к Заказчику:

- Заказчик обязуется по требованию Исполнителя предоставить необходимую информацию для выполнения работы.
- Заказчик своевременно оплачивает работу Исполнителя в размере и сроки, предусмотренные в договоре о технической поддержке.
- Заказчик должен предоставить Исполнителю контактное лицо для уточнения технических вопросов в том числе ФИО, e-mail, телефон.

Развитие продукта

Выпуск новых версий продукта осуществляется посредством выпуска новых релизов и модулей. Для контроля версий программного продукта и его частей приняты следующие обозначения:

1. Обозначение версии всего программного комплекса (зависит от набора включенных функциональных контуров).



2. Внутренняя нумерация версий контуров, из модулей. Внутренняя нумерация версий программного осуществляется по формату A.B.C[r], где:

- A – главный номер версии (major version number).
- B – вспомогательный номер версии (minor version number).
- C – номер сборки, номер логической итерации по работе над функционалом версии A.B (build number).
- [r] – условное обозначение релиза.

Нумерация версий модулей в программном продукте и его функциональных контурах синхронизируется с элементом верхнего уровня (например, модуль с контуром, в который он входит) по главному номеру версии, но имеет свою индивидуальную нумерацию по вспомогательному номеру версии.

Выпуск новой версии программного продукта сопровождается обновлением документации по эксплуатации, а также создается документ Сведений о функциональных доработках (Приложение 2).

Доработки программного продукта, которые по мнению Исполнителя (в рамках задач по сопровождению продукта) улучшают общий функционал продукта (совершенствование на основе доработки) и могут быть использованы для расширения и улучшения всей программной системы так и ее функциональных контуров, могут быть включены в перспективный план развития и доработки в следующих версиях.

Задачи, требования и предложения фиксируются в системе управления проектами в разделе управления требованиями со следующими параметрами:

- Тип: бизнес-логика, общесистемное требование, нефункциональное требование.
- Название контура и/или его модуля, для которого предлагается доработка.
- Описание предназначения, целей и задач доработки.
- Описание бизнес-процесса, входной и выходной информации, отчетных форм.
- Предложение по бизнес-логике исполнения нового функционала в программной системе.

На основе задач, после анализа ответственным работником Исполнителя требования детализируются и включаются в документ Сведений о функциональных доработках (Приложение 2).

Требования к специалистам для развертывания, поддержки и развития программного продукта ЮНИВУЗ-3

Роли и задачи



1. Системный аналитик – анализ бизнес-процессов их сравнение в бизнес-логикой, заложенной в продукт; разработка технических заданий на доработку, анализ заявок пользователей продукта, ведение и документирование логической модели.
2. Разработчик - реализация заявок по техническим заданиям, реализация автоматических тестов.
3. Администратор - обеспечение и контроль процесса обновления тестовых и рабочих серверов; создание сайтов подразделений, настройка прав доступа к сайтам/коллекциям сайтов/контурам управления/модулям корпоративного портала и развернутого на его платформе программного обеспечения; контроль за работоспособностью портала и его частей.
4. Специалист сопровождения - обработка поступающих запросов, контроль полноты заявки, перенаправление заявок, консультация по заявкам.
5. Тестировщик - проверка реализации, воспроизведение ошибок из заявок пользователей, проведение тестирования.
6. Технический писатель - разработка и поддержка в актуальном состоянии документации, инструкций для пользователей, описание версий.

Требования к исполнителям ролей:

Разработчик C#

1. Опыт работы с .NET Core и технологиями:
 - C#,
 - Умение писать Unit тесты;
 - Знать, что такое SOLID и уметь применять это на практике
2. Опыт разработки Web-приложений и знания:
 - HTML,
 - CSS,
 - JavaScript + jQuery,
 - Angular
3. Базовые знания для работы с базой данных:
 - PostgreSQL,
 - Entity Framework Core (Code First only)
 - Опыт в разработке структуры БД,
 - Опыт написания и отладки хранимых процедур.
4. Дополнительно требуется:
 - RabbitMQ
 - Docker и Kubernetes



- Работа с системами контроля версий,
- Навыки по оптимизации кода,
- Опыт написания и использования Unit тестов,
- Сервисы для командной разработки Devprom, Gitlab.

Разработчик UI

1. Опыт разработки Web-приложений и знания:
 - HTML 5,
 - CSS,
 - JavaScript + jQuery,
 - Знание принципов адаптивного дизайна с учетом особенностей браузеров (Internet Explorer, Edge, Chrome, Firefox, Opera, Яндекс),
 - AJAX,
 - Принципы оптимизации Web-приложений.
2. Базовые знания .NET Core и C#
3. Дополнительно требуется:
 - Работа с системами контроля версий,
 - Навыки по оптимизации кода,
 - Опыт написания и использования Unit тестов,
 - Сервисы для командной разработки Devprom, Gitlab.

Администратор

1. Знание принципов работы локальных сетей/интернета (уровни модели OSI, сетевые протоколы, маршрутизация и т.д.).
2. Опыт администрирования Linux и Linux-подобных систем.
3. Автоматическое резервное копирование.
4. Знание и навыки DevOps

Системный аналитик

1. Обязательно знание методологии UML, нотации IDEF0/DFD (например, в системе CA Erwin Process Modeler), BPMN (например, в системе Bizagi).
2. Знание принципов проектирования структуры баз данных (нотация IDEF1X, SQL)
3. Понимание процесса управления жизненным циклом разработки программного продукта, гибких методологий разработки (Agile, Scrum, Kanban).



4. Знание требований к документированию ПО (управление требованиями, разработка технического задания). Опыт составления технических заданий и постановок задач для реализации функциональных требований. Умение использовать ГОСТы при составлении проектной документации.
5. Навыки прототипирования (например, Axure, Figma)
6. Опыт анализа бизнес-процессов организации.
7. Навыки подготовки презентационных материалов.

Технический писатель

1. Представление о системной и программной инженерии
2. Знание международных стандартов жизненного цикла программ и их документирования (ISO, IEEE, ГОСТ, ЕСПД, ЕСТД)
3. Понимание основ современных ИТ, в частности стека технологий C#, понимание основных принципов работы реляционных БД, клиент-серверная архитектура ПО
4. Методологии и языки моделирования (IDEF, ER, UML) на уровне, который позволяет читать и понимать модели
5. Владение терминологией и стилем письма, удовлетворяющим стилю технической документации.



Приложение 1

Шаблон приложения 1 к договору № _____ от __.__.20__г.
Форма технического задания в рамках работ по договору.

Техническое задание № __ от __.__.____г.
на обслуживание и поддержку сайта
в рамках Договора № _____ от __.__.20__

Объект технической поддержки

Техническое задание описывает перечень, стоимость и сроки выполнения работ по техническому обслуживанию и поддержке программного продукта «ЮНИВУЗ-3».

Взаимодействие сторон

Взаимодействие сторон в рамках согласования и исполнения работ осуществляется ответственными исполнителями:

Со стороны Заказчика - ФИО посредством e-mail, телефон, и т.д.

Со стороны Исполнителя - ФИО посредством e-mail, телефон, и т.д.

Перечень и сроки исполнения работ

№ п/п	Наименование работы	Трудоемкость, ч	Стоимость, руб.
1.			
2.			
..			
	ИТОГО		

Общий срок выполнения работ: ДД.ММ.ГГГГ – ДД.ММ.ГГГГ

Общая стоимость работ составляет _____ рублей _____ копеек (прописью).

Подписи Сторон:

ЗАКАЗЧИК
Должность
И.О. Фамилия

“__” _____ 20__ года
МП

ИСПОЛНИТЕЛЬ
Директор ООО «ЮНИСИСТЕМС»

А.С. Царегородцев

“__” _____ 20__ года
МП



Приложение 2

Шаблон-пример документа «Сведения об изменениях функциональных требований»

Сведения об изменениях функциональных требований

К _____ контура _____
 <модулю/разделу/сайту и т.п. + название> <название контура или крупного функционального блока>
 за _____
 <период>

	Функция	Воздействие	Изменение	Дата занесения	Отметка о выполнении
1	Удаление индивидуальных сроков сессии	Новый функционал	В форме редактирования сроков сессии добавить кнопку «Удалить индивидуальный срок сессии»	18.12.20__	Готово
2	Ввод баллов и оценок	Новый функционал	Добавить поля ввода баллов студента по дисциплине: • Балл за текущую успеваемость – число, 1 знак после запятой, ввод вручную; • Балл за экзамен – только для типа промежуточного контроля «БРК» или «Экзамен», число, 1 знак после запятой, ввод вручную; • Всего баллов - Балл за текущую успеваемость + Балл за экзамен – рассчитывается автоматически.	18.12.20__	На тестовом сервере