

Программное обеспечение «Moscow digital school — платформа развития компетенций для цифровой экономики» (ПО)

Описание жизненного цикла, поддержки и обслуживания программного обеспечения

Аннотация

В настоящем документе содержится описание процессов, обеспечивающих поддержание жизненного цикла ПО «Moscow digital school - платформа развития компетенций для цифровой экономики», в том числе устранение неисправностей, выявленных в ходе эксплуатации программного обеспечения, совершенствование программного обеспечения, а также информацию о персонале, необходимом для обеспечения такой поддержки.

1. Термины и определения

Аудит	Независимая оценка программных продуктов и процессов, проводимая уполномоченным лицом с целью оценить их соответствие требованиям
Базовая линия	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
Валидация	Подтверждение (на основе представления объективных свидетельств) того, что требования, предназначенные для конкретного использования или применения, выполнены
Верификация	Подтверждение (на основе представления объективных свидетельств) того, что заданные требования полностью выполнены
Версия	Идентифицированный экземпляр составной части или программного продукта
Жизненный цикл	Развитие системы, продукта, услуги, проекта или других изготовленных человеком объектов, начиная со стадии разработки концепции и заканчивая прекращением применения
Мониторинг	Текущий контроль состояния деятельности поставщика и результатов этой деятельности, проводимый приобретающей или третьей стороной
Отказоустойчивость	Свойство системы сохранять свою работоспособность после отказа одной или нескольких её составных частей.
Пользователь	Лицо или группа лиц, извлекающих пользу из системы в процессе ее применения
Правообладатель	Лицо или организация, имеющие право, долю, требование или интерес в системе или в обладании ее характеристиками, удовлетворяющими ее потребности и ожидания
Программная составная часть	Исходный код, объектный код, контрольный код, контрольные данные или совокупность этих составных частей
Программный продукт	Совокупность компьютерных программ, процедур и, возможно, связанных с ними документации и данных
Проект	Попытка действий с определенными начальными и конечными сроками, предпринимаемая для создания продукта или услуги в соответствии с заданными ресурсами и требованиями

2. Общие сведения

Назначение ПО

ПО «Moscow digital school - платформа развития компетенций для цифровой экономики» – это цифровой B2C-сервис, позволяющий проходить обучение онлайн. Представляет из себя систему из личных кабинетов для педагога, преподавателя, куратора и администратора. Сервис пользуется спросом у юристов из разнообразных сфер: от банковского сектора до игровой индустрии. Система решает комплекс задач по автоматизации процесса обучения:

- организация образовательного процесса
- онлайн-оплата доступа к курсу
- автоматический расчет прогресса и проверка знаний

Результатом обучения является сертификат, который автоматически отправляется на почту пользователя, согласно бизнес-процессу.

Сведения о правообладателе программного обеспечения:

ООО "Диджитал Скиллс"

127055, г. Москва, ул. Новосуцёвская, д. 19Б, пом. I комн 19 этаж 5

Телефон: +7 (495) 128-49-50

e-mail: info@mosdigitals.ru

3. Процессы жизненного цикла

Жизненный цикл программных средств, входящих в состав программного обеспечения

ПО «Moscow digital school - платформа развития компетенций для цифровой экономики» обеспечивается в соответствии с требованиями ГОСТ Р ИСО/МЭК 12207-2010.

3.1. Стадии жизненного цикла

3.1.1. Стадия подготовки

а) Идея

Данный этап включает в себя сбор и анализ требований, разработку возможных решений и оценку их реализуемости, предварительный расчет затрат на реализацию решений и оценку их полезности с учетом целей правообладателя, оценку рисков.

Любая идея фиксируется в Story (или в другом типе задачи в зависимости от флоу работы на проекте).

В результате действий принимается решение о продолжении выполнения работ на стадии разработки или об отказе от дальнейшей работы.

б) Определение заинтересованных сторон (стейкхолдеров)

Данный этап включает в себя определение заинтересованных в проекте сторон (стейкхолдеров), которые могут повлиять на проект; выявление потенциальных сложностей, которые могут прервать проект или снизить успешность проекта; выделение ключевых лиц, которые должны быть информированы о ходе проекта, определение группы лиц, которые должны быть вовлечены на каждом этапе проекта, оценку средств, правил и принципов коммуникации на протяжении всего проекта и планирование действий для снижения негативного влияния стейкхолдеров на ход проекта, а также анализ бизнестребований, которые содержат указания на проблемы и потребности.

Этап включает в себя также согласование проекта с заинтересованными сторонами (стейкхолдерами).

с) Дизайн

На данной этапе дизайнер проектируют дизайн на основе подготовленного набора артефактов.

д) Согласование системных требований

На данном этапе анализируются особенности планируемого применения разрабатываемой системы для задания системных требований. Системные требования расставляются по приоритетам, утверждаются и фиксируются в базовой линии.

Спецификация системных требований должна описывать:

- функции и возможности системы;
- требования деловой сферы, организационные и пользовательские требования;
- требования по безопасности, защищенности, эргономике, интерфейсам, рабочим операциям и сопровождению;

- проектные ограничения и квалификационные требования. Спецификация системных требований должна быть документирована.

е) Формирование задач и установка приоритета

На данном этапе по результатам разработки и согласования дизайна, системных и функциональных требований осуществляется постановка задач группам разработчиков.

3.1.2. Стадия разработки

В ходе выполнения стадии проектируется архитектура программного обеспечения и его элементов. При реализации программного обеспечения производятся, комплексированы, испытываются и оцениваются технические и программные средства и интерфейсы, определяются требования к средствам производства, обучения и поддержки, разрабатывается документация.

Стадия завершается готовностью программного продукта (прототипа) к предварительным испытаниям.

Результатом стадии разработки является программный продукт (или прототип), готовый к предварительным испытаниям, вместе с технической документацией, а также требования, решения, оценки и прочие аналитические данные, предназначенные для использования на последующих стадиях.

3.1.3. Стадия ввода в действие

Стадия ввода в действие заключается в изготовлении, сборке, комплексировании и проведении испытаний программного обеспечения, разработанного на предыдущем этапе. Стадия может включать в себя процессы улучшения и реконфигурации программного продукта.

Стадия включает в себя следующие этапы тестирования:

- **Предварительные испытания.** В ходе предварительных испытаний проверяется работоспособность и соответствие техническому заданию, устраняются выявленные неисправности и недостатки. После этого программный продукт передается в опытную эксплуатацию.
- **Опытная эксплуатация.** В ходе опытной эксплуатации выполняется работа с программным продуктом по назначению, собираются статистические данные о характеристиках и результатах функционирования. При необходимости осуществляется доработка программного продукта.

Результатом стадии является ввод программного продукта в эксплуатацию.

3.1.4. Стадия внедрения

Началом стадии внедрения служит установка и передача программного обеспечения для применения по назначению.

Стадия включает в себя процессы, относящиеся к использованию программного обеспечения в целях, отвечающих назначению. Совместно с использованием программного обеспечения осуществляется контроль характеристик, а также сбор и анализ данных об отклонениях, недостатках и отказах. В результате анализа таких данных могут быть инициированы:

- стадия сопровождения применения с целью технического обслуживания или незначительной модификации;
- стадии разработки и производства с целью значительной модификации;
- стадия изъятия и списания, в случае принятия решения о завершении эксплуатации.

3.1.5. Стадия сопровождения внедрения

Стадия сопровождения внедрения состоит в обеспечении техническим обслуживанием и сопровождением, материально-техническим снабжением и другими видами поддержки функционирования и использования программного обеспечения.

Стадия может включать в себя контроль характеристик, а также сбор и анализ данных об отклонениях, недостатках и отказах, выявленных в системе поддержки.

3.1.6. Стадия изъятия и списания

Стадия изъятия и списания обеспечивает ликвидацию программного обеспечения и связанных с ним эксплуатационных и поддерживающих служб.

Причиной перевода в данную стадию может служить замещение новой системой, невозможность восстановления износ, катастрофический отказ, неэффективность дальнейшего применения и поддержки.

3.2. Технические процессы

3.2.1. Основной процесс внедрения

В результате успешного осуществления основного процесса внедрения (в ГОСТ Р ИСО/МЭК 12207-2010 используется термин «реализация») программных средств:

- определяется стратегия внедрения программного обеспечения;
- определяются ограничения по технологии реализации программного обеспечения;
- изготавливается программная составная часть программного обеспечения;
- программная составная часть упаковывается и хранится в соответствии с соглашением о ее поставке и лицензионной политикой правообладателя.

3.2.2. Процесс анализа требований к программным средствам

В результате успешного осуществления процесса анализа требований к программным средствам:

- определяются требования к программным элементам программного обеспечения и их интерфейсам;
- требования к программным средствам анализируются на корректность и тестируемость;
- осмысливается воздействие требований к программным средствам на среду функционирования;
- устанавливается совместимость и прослеживаемость между требованиями к программным средствам и требованиями к программному обеспечению;
- 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.

3.2.3. Процессы проектирования программных средств

Раздел включает процесс проектирования архитектуры программных средств и процесс детального 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 программным проектом.

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 и устанавливается совместимость и прослеживаемость между детальным проектированием, требованиями и проектированием архитектуры.

3.2.4. 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.

3.2.5. 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).

3.2.6. 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.

3.3. Процессы поддержки программных средств

3.3.1. Процесс управления документацией программных средств

В результате успешного осуществления процесса менеджмента документации программных средств:

- разрабатывается стратегия идентификации документации, которая реализуется в течение жизненного цикла программного обеспечения 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.

3.3.2. 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.

На основании плана осуществляется управление конфигурацией, которое включает в себя:

- идентификацию и регистрацию заявок на изменения;
- анализ и оценка изменений;
- принятие или отклонение заявок;
- реализацию, верификацию и выпуск модифицированной составной части;
- проверочные испытания, на основании которых можно проследить каждую модификацию, ее причины и полномочия на проведение изменений;
- управление и аудит всего доступа к контролируемым программным составным частям, связанным с выполнением критических функций по безопасности или защите.

3.3.3. 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;
- отказоустойчивость программных средства обеспечивается разворачиванием нескольких экземпляров каждого микросервиса программного обеспечения на виртуальных машинах контура. Управление разворачиванием и поддержкой экземпляров микросервисов в рабочем состоянии осуществляется стандартными средствами кластера;
- отказоустойчивость базы данных обеспечивается кластерным решением с классическим методом перехода на резерв в ручном режиме. Сервер базы данных имеет три машины: master, slave и backup.

3.3.4. 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 заинтересованным сторонам.

3.3.5. 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 созданное рабочее программное обеспечение пригодно для применения по назначению;
- результаты действий по валидации делаются доступными заинтересованным сторонам;
- проверка работоспособности балансируемых серверов балансировщиком нагрузки. Балансировщик осуществляет проверку работоспособности (мониторинг) серверов кластера;
- проверка работоспособности микросервисов. Проверка готовности проверяет, готов ли контейнер обрабатывать запросы. Ошибка проверки готовности означает, что контейнер не должен получать трафик от прокси, даже если он работает. Проверка живучести проверяет, работает ли контейнер. Балансировщик осуществляет проверку работоспособности (мониторинг) контейнера. При проверке на живучесть и проверку на работоспособность микросервиса, в случае, когда http-ответ возвращается не с кодом 200, сервер будет непрерывно стараться перезапустить экземпляр микросервиса.

План валидации включает в себя следующие проверки:

- тестирование в условиях повышенной нагрузки, граничных значений параметров и необычных входов;
- тестирование программного обеспечения на его способность изолировать и минимизировать влияние ошибок; то есть осуществлять плавную деградацию после отказов, обращение за помощью в условиях повышенной нагрузки, граничных значений параметров и необычных входов;
- тестирование того, что основные пользователи могут успешно решать намеченные задачи, используя данный программный продукт;
- тестирование программного продукта на соответствие своему назначению;
- тестирование программного продукта в выбранных областях заданной среды применения по назначению.

3.3.6. 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. Проблемы, выявленные при проведении ревизии, регистрируются и передаются в процесс решения проблем в программных средствах.

3.3.7. 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 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.

По каждому аудиту устанавливается:

- повестка дня;
- состав проверяемых программных продуктов и результатов деятельности;
- область распространения и процедуры аудита;
- исходные и итоговые критерии проведения аудита.

Проблемы, выявленные при проведении аудитов, передаются процессу решения проблем в программных средствах.

3.3.8. 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;
- реализация требований к режиму сопровождения устанавливается следующим образом: для промышленного контура: 24 часа 7 дней в неделю; для контуров функционального тестирования: 8 часов 5 дней в неделю.

3.4. Сопровождение и техническая поддержка

Способ оказания технической поддержки: по электронной почте СПП (mail@cron.design).

3.5. Информация, необходимая для установки и эксплуатации ПО

3.5.1. Минимальная требуемая квалификация для установки и активации ПО:

- базовые навыки работы на персональном компьютере с современными офисными приложениями и операционной системой (клавиатура, мышь, управление окнами и приложениями, файловая система);
- базовые навыки использования интернет-браузера (установка подключений, доступ к веб-сайтам, навигация, формы и другие типовые интерактивные элементы веб-интерфейса).

3.5.2. Минимальная требуемая квалификация для использования ПО:

- базовые навыки работы на персональном компьютере с современными офисными приложениями и операционной системой (клавиатура, мышь, управление окнами и приложениями, файловая система).

3.6. Персонал, задействованный в жизненном цикле ПО

Разработка:

- **Руководитель группы разработки (1 сотрудник)** – принимает участие во всех стадиях жизненного цикла в процессах планирования и контроля исполнения проекта, утверждения архитектурных решений, менеджмента и стратегии проекта в соответствии с должностной инструкцией. Руководитель группы разработки обладает следующей квалификацией: высшее профильное образование; опыт в области разработки программного обеспечения более 5-ти лет; опыт руководства более 2-х лет.
- **Руководитель продукта (1 сотрудник)** – принимает участие в качестве руководителя команды продукта во всех стадиях жизненного цикла в процессах анализа требований, разработки архитектурных решений, разработки планов и программ, реализации программного продукта, выпуска версий, тестирования, применения и сопровождения программных средств. Руководитель продукта обладает следующей квалификацией: высшее образование; стаж работы более 5-ти лет.
- **Разработчик (1 сотрудник)** – принимает участие во всех стадиях жизненного цикла в процессах реализации программного продукта, выпуска версий, тестирования и сопровождения программных средств. Разработчик обладает следующей квалификацией: высшее профильное образование; опыт разработки программного обеспечения более 1-го года.
- **Ведущий дизайнер (1 сотрудник)** – принимает участие в разработке дизайна проекта и стратегии проекта в соответствии с должностной инструкцией. Ведущий

дизайнер обладает следующей квалификацией: высшее профильное образование; опыт в области дизайна программного обеспечения более 3-х лет.

Техническая поддержка

- **Ведущий аналитик (1 сотрудник)** – принимает участие во всех стадиях жизненного цикла в процессах разработки архитектурных решений, реализации программного продукта, выпуска версий, тестирования и сопровождения программных средств. Ведущий аналитик обладает следующей квалификацией: высшее профильное образование; опыт более 2-х лет.
- **Администратор проектов (1 сотрудник)** – принимает участие во всех стадиях жизненного цикла в процессах планирования и контроля исполнения проекта, утверждения архитектурных решений, менеджмента и стратегии проекта в соответствии с должностной инструкцией. Администратор проектов обладает следующей квалификацией: высшее профильное образование; опыт в области разработки программного обеспечения более 5-ти лет.
- **Разработчик (1 сотрудник)** – принимает участие во всех стадиях жизненного цикла в процессах реализации программного продукта, выпуска версий, тестирования и сопровождения программных средств. Разработчик обладает следующей квалификацией: высшее профильное образование; опыт разработки программного обеспечения более 1-го года.

4. Контактная информация

Фактический адресе размещения инфраструктуры разработки, фактический адрес размещения разработчиков, фактический адрес размещения службы поддержки:

644024, Омск, Учебная, 76, офис 308

Данные о возможных средствах коммуникации со службой поддержки:

mail@cron.design

telegram: @crondesign

Данные о режиме работы службы поддержки

С понедельника по пятницу, с 9:00-15:00 по Московскому времени