

Министерство науки и высшего образования Российской Федерации

Томский государственный университет
систем управления и радиоэлектроники

А. А. Сидоров
М. В. Владимиров

ВНЕДРЕНИЕ И СОПРОВОЖДЕНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ

Методические указания к лабораторным работам
и организации самостоятельной работы для студентов направления
«Бизнес-информатика»
(уровень магистратуры)

Томск
2022

УДК 004
ББК 32.972
С34

Рецензент:

Салмина Н.Ю., декан факультета систем управления
Томского государственного университета
систем управления и радиоэлектроники, канд. техн. наук, доцент

Сидоров, Анатолий Анатольевич

С34 Внедрение и сопровождение информационных систем: методические указания к лабораторным работам и организации самостоятельной работы для студентов направления «Бизнес-информатика» (уровень магистратуры) / А. А. Сидоров, М. В. Владимиров. – Томск : Томск. гос. ун-т систем упр. и радиоэлектроники, 2022. – 24 с.

Методические указания по дисциплине «Внедрение и сопровождение информационных систем» содержат необходимые разъяснения по форме организации и содержанию лабораторных работ и самостоятельной деятельности студентов. Выполнение данных указаний будет способствовать успешному освоению дисциплины, в том числе выработке навыков по решению практических задач итогового тестирования, внедрения, сопровождения и ликвидации информационных систем, а также всех процессов, связанных со взаимодействием с заказчиком и его представителями.

Для студентов высших учебных заведений, обучающихся по направлению направления «Бизнес-информатика» (уровень магистратуры). Методические указания могут быть использованы и для направлений подготовки укрупненной группы направлений и специальностей «Информатика и вычислительная техника».

Одобрено на заседании кафедры АОИ, протокол № 1 от 20.01.2022

УДК 004
ББК 32.972

© Сидоров А. А., Владимиров М. В., 2022
© Томск. гос. ун-т систем упр.
и радиоэлектроники, 2022

Оглавление

ВВЕДЕНИЕ	4
1 МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ПРОВЕДЕНИЮ ЛАБОРАТОРНЫХ РАБОТ	5
1.1 Порядок организации лабораторных работ	5
1.2 Лабораторная работа «Выбор предметной области с потенциалом автоматизации»	5
1.3 Лабораторная работа «План внедрения информационной системы»	7
1.4 Лабораторная работа «Написание руководства пользователя»	9
1.5 Лабораторная работа «План обучения сотрудников заказчика»	11
1.6 Лабораторная работа «Анализ качества внедрения»	12
1.7 Лабораторная работа «Continuous Integration & Continuous Delivery»	14
1.8 Лабораторная работа «Основы настройки CI/CD»	16
1.9 Лабораторная работа «Ценообразование при разработке и сопровождении информационных систем»	18
2 МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ПРОВЕДЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ	21
2.1 Общие положения	21
2.2 Проработка лекционного материала	21
2.3 Изучение тем (вопросов) теоретической части дисциплины, вынесенных для самостоятельной подготовки	21
2.4 Подготовка к лабораторным работам	22
2.5 Подготовка к промежуточной аттестации	22
ЗАКЛЮЧЕНИЕ	23
ЛИТЕРАТУРА	24

ВВЕДЕНИЕ

Целью изучения дисциплины «Внедрение и сопровождение информационных систем» является получение обучающимися практических навыков в организации взаимодействия с клиентом на этапе передачи ему готового программного продукта, а также на всех последующих этапах жизненного цикла.

Задачи дисциплины:

- изучение участников процесса передачи готового программного продукта после его разработки;
- изучение принципов проведения опытной эксплуатации программного обеспечения;
- исследование процесса организации поддержки программного продукта;
- изучение принципов ценообразования, технической поддержки и прочих услуг, сопряженных с сопровождением программного продукта;
- изучение основ организации процесса непрерывной доставки, развертки и автоматического тестирования для программных продуктов.

По окончании изучения дисциплины студент должен:

- знать особенности протекания завершающих этапов жизненного цикла разработки программного продукта; принципы ценообразования для услуг сопровождения программного продукта; принципы организации процесса непрерывной доставки и развертывания обновлений; основы идентификации заинтересованных сторон и всех участников в том или ином процессе, связанном с поздними этапами жизненного цикла разработки программного обеспечения.

- уметь анализировать требования заказчика к разработке и сопровождению системы; просчитывать цену, удовлетворяющую бюджету заказчика и потребностям исполнителя; анализировать и выбирать инструменты для организации процесса непрерывной доставки; находить лиц, принимающих решения по тем или иным вопросам, связанным с программным продуктом; находить всех участников того или иного процесса, связанного с внедрением и сопровождением, коммуницировать с ними.

Освоение дисциплины «Внедрение и сопровождение информационных систем» необходимо для качественного оказания услуг заказной разработки для организаций. Данная дисциплина будет полезна представителям таких IT-профессий как разработчик программного обеспечения, менеджер команды, руководитель проекта и пр.

1 МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ПРОВЕДЕНИЮ ЛАБОРАТОРНЫХ РАБОТ

1.1 Порядок организации лабораторных работ

Лабораторные работы направлены на формирование у обучающихся умений и навыков по использованию теоретических основ при организации процессов, связанных с внедрением и последующим сопровождением программного продукта вплоть до завершения его жизненного цикла. Обучающиеся в процессе подготовки к занятиям должны самостоятельно изучить рекомендованную преподавателем научно-техническую литературу по теме занятия. В процессе проведения занятия обучающемуся выдается индивидуальное задание по описанию определенных процессов, связанных с внедрением и сопровождением программных продуктов. По итогу выполненной работы студент формирует отчет, отражающий все детали организации того или иного процесса.

В конце занятия преподавателем выставляется оценка за проделанную работу.

1.2 Лабораторная работа «Выбор предметной области с потенциалом автоматизации»

Цель работы

Формирование компетенций по поиску перспективной для автоматизации ниши в действующих бизнесах.

Форма проведения

Выполнение индивидуального задания.

Форма отчетности

На проверку должен быть предоставлен отчет с описанием выбранной предметной области, выбранного представителя предметной области (компании, занимающейся оказанием услуг или реализацией товаров в определенной нише), а также указанием деятельности, имеющей экономически выгодный потенциал для автоматизации. Кроме того, отчет должен содержать анализ автоматизируемой деятельности с приведением числовых значений показателей, обосновывающих экономическую эффективность данного решения, и краткое техническое задание для предлагаемого программного продукта, автоматизирующего выбранное направление деятельности, содержащее функциональные требования к проекту.

Теоретические основы

При поиске сферы для автоматизации посредством создания программного продукта в первую очередь необходимо определиться с нишей. Основным критерием выбора предметной области является понимание оной. Т.е. необходимо понимание или сторонняя экспертиза для получения данных обо всех бизнес-процессах, которые подлежат автоматизации.

Поиск сфер для автоматизации может основываться на различных критериях:

- наличие компетенций (обладание обучающимся знаниями или навыками, позволяющими сделать программный продукт, максимально удовлетворяющий всем требованиям заказчика);
- по аналогии (наличие программного продукта, например, в другой стране и отсутствие такового в месте потенциальной локализации);
- новостные поводы (появление технологий, позволяющих решить проблему, которую раньше было невозможно решить в связи с технологическими ограничениями).

После выбора предметной области необходимо провести расчеты, показывающие экономическую эффективность автоматизации той деятельности, которая была выбрана. Для анализа экономической эффективности необходимо рассчитать количество финансовых ресурсов, которое тратится на решение проблемы на данный момент, или экономическую выгоду,

которую способно принести данное решение. Сделать это можно получив данные у действующих специалистов, занимающихся данной деятельностью, или рассчитав самостоятельно, используя открытые источники данных.

Следующий этап – создание краткого технического задания, описывающего функциональные требования, с последующим расчетом ориентировочной стоимости разработки данного продукта.

После расчета стоимости разработки необходимо рассчитать стоимость данного программного продукта для потенциального заказчика. На данном этапе уже можно рассчитать экономику проекта. Если проект заказной, то достаточно предоставить информацию о сроках и стоимости разработки заказчику. Если проект не является заказным, то для получения информации о целесообразности разработки данной системы необходимо рассчитать срок окупаемости, стоимость сопровождения и потенциальный рынок. На основе данных расчетов можно сделать вывод о необходимости реализации данной системы.

Показатели и конкретные числовые значения для каждой системы определяются индивидуально в зависимости от бизнес-процессов, присущих выбранной сфере.

Порядок выполнения лабораторной работы

Необходимо провести анализ предметных областей. Например, Вы обнаружили, что магазин с товарами для автомобилей заказал для себя интернет-магазин. Из этого можно сделать вывод, что подобным магазинам тоже может быть интересно создание интернет-магазина для себя.

В данном случае, интернет-магазин – это не автоматизация какой-либо деятельности, а создание дополнительного канала реализации товара, способного увеличить выручку конкретного заказчика.

С целью расчета экономических основ проекта необходимо составить краткое техническое задание с описанием функциональных требований. Это может быть реализовано в виде таблицы (таблица 1).

Таблица 1 – Функциональные требования к интернет-магазину

Функционал	Время, час
Авторизация пользователей	8
Регистрация пользователей	8
Восстановление пароля	6
Смена пароля	8
Смена данных пользователя	8
Управление адресами пользователя	16
Каталог товаров	24
Работа с корзиной	24
Работа с избранными товарами	16
Оформление заказа	40
Администрирование товаров	40
Администрирование акций	40
Администрирование методов доставки	32
Ведение справочников	40
Управление настройками системы	16
Интеграция внешних сервисов	80
Тестирование	40
Правки от заказчика	40

Итого 486 часов = 61 рабочий день = 3 месяца: срок разработки простого интернет-магазина. Цена для такого интернет магазина при цене часа работы компании 1000 рублей в час составит 486 тысяч рублей.

Для данного проекта существуют дополнительные траты на дизайнера, которые составят от 50 000 рублей, на специалиста по поисковой оптимизации, которые также составят в районе 50 000 рублей.

В рамках данного примера нет необходимости считать сроки окупаемости, т.к. данный продукт подразумевает коммерческое предложение заказчику как продукт, а не как услуга на долгосрочный период.

1.3 Лабораторная работа «План внедрения информационной системы»

Цель работы

Формирование и закрепление навыков по типовым сценариям действий всех участников при внедрении информационных систем.

Форма проведения

Выполнение индивидуального задания.

Форма отчетности

На проверку должен быть предоставлен отчет с описанием плана внедрения и сопровождения данной системы с описанием ролей всех участников процесса внедрения, а также всех этапов внедрения с указанием лиц, чье участие на данном этапе требуется.

Данный план должен подробно описывать каждый этап: какие работы будут вестись на определенном этапе, кто должен быть привлечен со стороны заказчика, кто должен быть привлечен со стороны исполнителя, в какие сроки должен быть реализован тот или иной этап, какие критерии завершения того или иного этапа.

Теоретические основы

При написании плана внедрения информационной системы необходимо сформировать понимание того, какие должны быть реализованы процессы, кто должен принимать участие в том или ином процессе, а также определить критерии завершения этапов.

Внедрение информационной системы начинается с внутреннего итогового тестирования, задачами которого являются следующие виды активности:

- проверка работоспособности системы;
- проверка соответствия внешнего вида и утвержденных макетов;
- проверка соответствия требованиям технического задания.

Данный этап реализуется силами проектной команды на стороне исполнителя. Этап считается выполненным при полном соответствии системы проектной документации.

Следующий этап – проведение приемочных испытаний. На данном этапе помимо представителей со стороны исполнителя (руководитель проектной команды и ведущий разработчик (опционально)) присутствуют представители заказчика. Состав представителей заказчика может отличаться от проекта к проекту и зависит от пожеланий заказчика и необходимости в демонстрации системы для потенциальных пользователей. Данный этап всегда сопровождается написанием методики приемочных испытаний и считается завершенным при подписании акта об успешном проведении приемочных испытаний со стороны заказчика.

Третий этап заключается в развертывании системы на стороне заказчика. Для реализации данного этапа существует четыре способа:

- развертывание заказчиком. Данный метод позволяет не тратить лишнее время исполнителю и возлагает всю ответственность на персонал компании-заказчика. Является наиболее предпочтительным для крупных компаний с большим количеством технических специалистов, т.к. позволяет экономить существенное количество денег.

- развертывание исполнителем. Данный метод является самым дорогим, т.к. заказчик обязан оплатить все затраченное время компании, что, как правило, составляет внушительную

сумму. Преимуществом данного подхода является качество развертывания, т.к. занимаются этим процессом специалисты, досконально знающие проект.

- развертывание с привлечением руководителя проекта со стороны исполнителя. Данный метод является промежуточным в плане финансовой нагрузки на заказчика по сравнению с предыдущими. Преимущество привлечения руководителя проекта заключается в понимании проекта и в понимании того, каких специалистов необходимо привлекать. Но основной объем работ ложится на плечи сотрудников компании заказчика.

- развертывание с привлечением технического специалиста со стороны заказчика. Данный метод так же является промежуточным в плане финансовой нагрузки, как и предыдущий. Достоинство привлечения технического специалиста состоит в понимании нюансов развертывания системы.

Независимо от выбранного способа развертывания системы, исполнитель обязан подготовить инструкцию по развертыванию для технических специалистов компании заказчика. Необходимо помнить, что ко всему прочему не удастся избежать привлечения специалистов со стороны заказчика к процессу развертывания, т.к. в крупных компаниях необходимо предоставить доступ к внутреннему контуру компании, серверам и прочим аппаратно-программным инструментам, оказывающим влияние на работоспособность системы.

Четвертый этап заключается в обучении персонала. На данном этапе исполнитель предоставляет заказчику эксплуатационную документацию и при необходимости проводит обучение всех потенциальных пользователей системы. Отдельные процессы в рамках этого этапа могут быть необязательными и реализуются по согласованию с заказчиком.

Пятый этап – это введение системы в опытную эксплуатацию. Опытная эксплуатация требуется для того, чтобы отследить особенности работы с системой реальных пользователей и ошибки в боевых условиях без риска потери данных. Особенностью данного этапа является то, что помимо новой системы пользователям в работе приходится использовать еще и старую систему. Делается это для того, чтобы не допустить потери данных в случае критических ошибок. Как правило, на данный этап отводится порядка трех месяцев.

Шестой этап – ввод в эксплуатацию. На данном этапе заказчик уже окончательно отказывается от старого метода решения тех задач, для которого была создана новая система. После проведения опытной эксплуатации подписывается акт о сдаче работ и окончательный расчет по проекту.

На данном этапе внедрение информационной системы заканчивается. Далее может быть подписан отдельный договор на техническое обслуживание. Также специалистов компании-исполнителя могут привлекать на этап ликвидации программного продукта по окончании срока его службы. Помимо описанных выше этапов могут существовать и другие. Их наличие зависит от требований заказчика и особенностей разрабатываемого проекта.

Порядок выполнения лабораторной работы

При выполнении лабораторной работы «Поиск предметной области с потенциалом автоматизации» была выбрана предметная область. Для данной предметной области необходимо описать все этапы и всю сопутствующую информацию к ним.

Для внедрения, например, интернет-магазина в организацию, торгующую одеждой, будут существовать следующие этапы:

- внутреннее тестирование;
- приемочные испытания с заказчиком;
- написание документации для персонала;
- написание документации для системных администраторов;
- подготовка программного продукта к передаче;
- развертывание программного продукта на мощностях заказчика;
- опытная эксплуатация;
- эксплуатация.

Анализ для каждого этапа должен содержать работы, ориентировочный срок реализации, критерий успешного завершения этапа, участников со стороны заказчика и участников со стороны исполнителя.

Например, для этапа «Приемочные испытания с заказчиком» может присутствовать следующее описание:

Этап «Приемочные испытания с заказчиком».

Работы:

- создание документации по проведению приемочных испытаний;
- организация встречи для проведения приемочных испытаний;
- проведение приемочных испытаний;
- подписание акта об успешном проведении приемочных испытаний.

Срок плановой реализации этапа: 5 рабочих дней.

Критерий завершения этапа: подписание акта об успешном проведении приемочных испытаний.

Участники со стороны заказчика: заказчик, аналитик.

Участники со стороны исполнителя: менеджер проекта, технический специалист

1.4 Лабораторная работа «Написание руководства пользователя»

Цель работы

Формирование компетенций по определению роли потенциальных пользователей и описанию руководства пользователя для них.

Форма проведения

Выполнение индивидуального задания.

Форма отчетности

На проверку должен быть предоставлен отчет с описанием выбранной системы, ролей потенциальных пользователей и алгоритмов использования системы для каждой из ролей.

Теоретические основы

При внедрении программного продукта в организацию необходимо предоставить документацию, позволяющую пользователю понять, как взаимодействовать с новой системой. Данная документация называется «руководство пользователя» и служит для обучения и помощи во взаимодействии с программным продуктом.

Качество внедрения напрямую зависит от пользовательского опыта и количества вопросов / проблем, возникших у персонала, работающего с системой. Следовательно, качество документации существенно влияет на данный показатель.

При создании руководства пользователя в первую очередь необходимо определиться, кто из представителей компании заказчика будет использовать разработанную систему и какой функционал нужен. Для этого в первую очередь расписываются роли, которые должны быть в системе. Как правило, эта информация присутствует в техническом задании, т.к. разделение функционала на части, к которым имеют доступ определенные роли, происходит на этапе проектирования системы и/или написания технического задания.

После определения ролей необходимо определить функционал для каждой из них. Под функционалом имеются в виду алгоритмы использования или User Story. User Story – это достаточно молодая практика описания комплексных тестов системы, позволяющая реализовать цикл множественных взаимодействий с системой для достижения макроцели.

Например, чтобы в интернет-магазине появился новый товар, сначала необходимо зайти в административную панель, в меню кликнуть на ссылку на раздел «Товары». Далее

необходимо нажать кнопку «Добавить товар» и заполнить открывшуюся форму. После нажатия кнопки «Сохранить» в списке товаров в административной панели должен появиться новый товар.

Данные макроцели часто коррелируют с бизнес-задачами организации и также часто бывают описаны в техническом задании.

Описание подобной User Story должно содержать в себе описание бизнес-задачи, алгоритм для выполнения и ожидаемый результат. Также качественная документация содержит скриншоты системы с указанием мест, куда необходимо нажать для достижения того или иного результата.

Для качественного описания необходимо по мере написания документации проверять работоспособность всех описанных алгоритмов.

Порядок выполнения лабораторной работы

При выполнении лабораторной работы «Поиск предметной области с потенциалом автоматизации» была выбрана предметная область и описана предлагаемая для реализации система. Для данной системы необходимо описать все роли, бизнес-задачи, которые они будут выполнять в рамках данной системы и ожидаемый результат.

Для выбранного в качестве примера интернет-магазина будут существовать следующие роли: роль покупателя, роль менеджера, роль администратора, роль руководства.

Для роли пользователя доступны все возможности, не представленные в административной панели. Роль менеджера включает в себя все возможности роли пользователя и возможность управлять наполнением сайта через административную панель. Роль руководства включает в себя весь функционал роли менеджера и дополнительно имеет доступ к разделу с формированием отчетов о деятельности интернет-магазина. Роль администратора имеет неограниченный доступ к функционалу (в т.ч. к настройкам проекта).

Тогда описание бизнес-задачи будет выглядеть следующим образом:

Изменение реквизитов расчетного счета для принятия платежей

Роль: администратор.

Алгоритм:

1. Зайдите на сайт <название сайта>.
2. Войдите в свой аккаунт.
 1. В шапке сайта нажмите кнопку «Войти».
 2. В открывшейся форме введите адрес электронной почты и пароль.
 3. Нажмите кнопку «Войти».
3. В шапке сайта нажмите появившуюся кнопку «Административная панель».
4. В меню выберите пункт «Настройки».
5. Нажмите на кнопку «Настройки счета».
6. Введите необходимые данные о расчетном счете в открывшуюся форму.
7. Нажмите кнопку «Сохранить».

Ожидаемый результат: после нажатия на кнопку «Сохранить» должно отобразиться сообщение об успешном изменении данных и при повторном открытии данной страницы должна отображаться введенная информация.

В рамках лабораторной работы не требуется прикреплять скриншоты работы системы.

1.5 Лабораторная работа «План обучения сотрудников заказчика»

Цель работы

Формирование компетенций по планированию взаимодействия с сотрудниками заказчика с целью обучения работе с системой.

Форма проведения

Выполнение индивидуального задания.

Форма отчетности

На проверку должен быть предоставлен отчет с описанием выбранной системы, описанием потенциальных пользователей системы и алгоритма обучения каждой из групп пользователей.

Теоретические основы

При внедрении программного продукта в организацию необходимо предоставить документацию, позволяющую пользователю понять, как взаимодействовать с новой системой. Не для каждой системы достаточно только документации. В редких случаях сложность системы настолько высока, что требуется провести обучение персонала для работы с ней.

В таких случаях от качества обучения напрямую зависит качество выполнения пользователями своих обязанностей, а это в свою очередь влияет на качество работы системы.

При составлении плана обучения персонала заказчика необходимо, в первую очередь, понять, какие группы людей будут работать с системой. Как правило, помимо пользователей (которых тоже можно разделить на группы в случае сложных систем) требуется обучить технических специалистов (системных администраторов и программистов). Набор групп пользователей определяется исходя из задач проекта и должен быть описан в техническом задании.

Далее следует определить, какую информацию нужно дать для каждой из групп и в каком формате. Могут быть использованы разные подходы. Это может быть личное присутствие специалиста из компании Исполнителя на предприятии и очное обучение. Также может быть записан специальный курс в виде видео-роликов, описывающий решение тех или иных задач и проблем. В редких случаях по итогу прохождения курса пользователю выдается сертификат, подтверждающий его компетентность для работы с системой. Последний пример активно используется компанией Vitrix в своей практике.

Следующий этап – это подготовка плана обучения. На данном этапе специалист, отвечающий за проведение обучения, должен описать, какую информацию, в каком объеме необходимо предоставить той или иной группе. Содержание обучения также индивидуально для каждого программного продукта. После создания плана необходимо подготовить материал для проведения обучения. Это может быть печатный материал, веб-ресурс или презентация для очных встреч.

Завершающим этапом является согласование с представителем заказчика времени и места проведения обучения.

Порядок выполнения лабораторной работы

При выполнении лабораторной работы «Поиск предметной области с потенциалом автоматизации» была выбрана предметная область и описана предлагаемая для реализации система. Для данной системы необходимо описать все роли, бизнес-задачи, которые они будут выполнять в рамках данной системы и ожидаемый результат.

Для выбранного в качестве примера интернет-магазина будут существовать следующие группы пользователей: покупатели, менеджеры, системные администраторы, руководство, разработчики.

В связи с тем, что интернет-магазин не является сложной системой, а интерфейс публичной части прост и интуитивно понятен, пользователей обучать взаимодействию с программным продуктом нет необходимости. Необходимо обучить работе с административной панелью все остальные обозначенные группы. Менеджерам для обучения работе с системой

достаточно руководства пользователя, потому что обучение этой группы также не имеет необходимости. Существует необходимость обучения системных администраторов и разработчиков принципам развертывания и обновления проекта, т.к. все ситуации описать в документации невозможно. Также существует необходимость обучить руководство использованию административной панели и формированию отчетов.

В качестве формата обучения были выбраны личные встречи, что позволит самостоятельно продемонстрировать процесс взаимодействия с информационной системой и оперативно обработать возражения и ответить на вопросы.

Завершающий этап данной лабораторной работы – написание плана обучения, который может выглядеть, как пример в таблице 2.

Таблица 2 – План обучения сотрудников

Группа пользователей	Состав обучения	Дата, время и место
Руководство	<ul style="list-style-type: none"> - формирование отчетов: поиск нужного раздела и выбор подходящего набора данных и формата файла для генерации отчета; - формирование графиков: поиск нужного раздела и выбор подходящего набора данных; - поиск детализации: при возникновении конфликтных ситуаций поиск информации в логах работы менеджеров 	13.04.2021 13:00 ул. Ленина 345, каб. 16
Системные администраторы	<ul style="list-style-type: none"> - проверка работоспособности систем: проверка работоспособности через интерфейс/консоль; - реакция на сбои: алгоритм действий при сбое в одном из контейнеров; - перезагрузка системы; - создание резервных копий базы данных; - проверка взаимодействия с интегрированными системами: эквайринг, 1С, почта для домена 	13.04.2021 16:00 ул. Ленина 345, каб. 18
Разработчики	<ul style="list-style-type: none"> - развертывание системы; - настройка системы; - выгрузка фикстур; - структура и архитектура системы 	13.04.2021 10:00 ул. Ленина 345, каб. 18

Учитывая формат проведения обучения создание какого-либо дополнительного обучающего материала в данном случае не предусматривается.

1.6 Лабораторная работа «Анализ качества внедрения»

Цель работы

Формирование навыков по анализу качества внедрения информационных систем.

Форма проведения

Выполнение индивидуального задания.

Форма отчетности

На проверку должен быть предоставлен отчет с описанием алгоритма действий и план мероприятий по анализу качества внедрения и эффективности работы информационной системы с характеристикой четких критериев для определения степени качества выполненных процессов внедрения.

В отчете по данной лабораторной работе должны быть подробно описаны различные аспекты внедрения информационной системы, а также критерии, по которым можно считать данный аспект успешно реализованным.

Теоретические основы

При проведении внедрения информационной системы необходимо понимать, какие аспекты внедрения критичны для заказчика и его деятельности. Чтобы заказчик остался доволен, необходимо удостовериться в качестве предоставленных услуг. Для этого проводится контроль качества внедрения информационной системы.

Набор характеристик, необходимых для отслеживания при внедрении, может меняться от проекта к проекту и зависит от особенностей программного продукта и требований заказчика. Как правило, особенно важные моменты описаны в техническом задании в разделе «нефункциональные требования».

Чаще всего используются следующие стандартные характеристики:

- скорость восстановления после сбоя;
- скорость ответа на запросы пользователя;
- безопасность хранения данных;
- поисковая оптимизация (для сайтов и web-приложений);
- трудозатраты на доставку и развертывание обновлений;
- сложности при работе с системой у пользователей и т.п.

Порядок выполнения лабораторной работы

При выполнении лабораторной работы «План внедрения информационной системы» был описан план внедрения определенного проекта. Для данного программного продукта необходимо описать все значительные критерии, которые необходимо отслеживать при развертывании продукта, а также критерии успешного прохождения теста, сроки и методы проверки.

Например, для внедрения интернет-магазина для организации на их мощности могут быть критерии оценки качества выполнения данного процесса, описанные в таблице 3.

Таблица 3 – Аспекты внедрения информационной системы

Аспект	Описание	Критерий качества	Действия для проверки
Простота развертывания	Отсутствие проблем и сложностей при развертывании на стороне заказчика	Отсутствие вопросов у системных администраторов при развертывании системы с помощью предоставленной документации	Ожидание завершения процесса развертывания (1 день)
Проверка поисковой оптимизации	Проверка качества поисковой оптимизации для поисковых систем Google, Yandex и Yahoo	Сайт находится на одной из первых 5 рекламных позиций в любой из описанных поисковых систем по запросу «...»	Поиск в описанных поисковых системах запроса «...» (6 месяцев)

Проверка скорости работы сайта	Исследование скорости ответа сервера на ключевые запросы, среди которых получение каталога товаров, ...	Ответ сервера по описанным запросам не превышает 300 мс	Использование специальных сервисов для проверки скорости ответа сайта на разных скоростях интернет – соединения (1 день)
Отсутствие сложности у администрации магазина при использовании административной панели	Отсутствие сложности у администрации магазина при использовании административной панели	Наличие лишь незначительных вопросов, не требующих правок от разработчиков	2 месяца опытной эксплуатации

Также необходимо расписать план мероприятий с указанием точной последовательности действий и сроков их реализации.

Для выбранного интернет-магазина действия для проверки будут описаны как в таблице 4.

Таблица 4 – Последовательность действий для тестирования

№ пп	Действие	Срок реализации, час
1	Развертывание системы	8
2	Проверка скорости работы системы	4
3	Проверка удобства взаимодействия с административной панелью. Запрос статистики по обращениям от пользователей системы	3
4	Проверка качества поисковой оптимизации	8

Данная таблица позволит ответственному лицу внести эти задачи себе с рабочий график и проверить качество оказанных заказчику услуг.

1.7 Лабораторная работа «Continuous Integration & Continuous Delivery»

Цель работы

Формирование навыков по использованию инструментов организации непрерывной доставки обновлений для информационных систем.

Форма проведения

Выполнение индивидуального задания.

Форма отчетности

На проверку должен быть предоставлен отчет с описанием задач, необходимых для автоматизации процесса тестирования и доставки программного продукта заказчику. Также необходимо предоставить анализ выбора тех или иных инструментов и решений для данной работы.

В отчете для данной лабораторной работы должно присутствовать обоснование необходимости в организации процесса непрерывной доставки для данного продукта, описание найденных решений для организации процесса CI/CD, их сравнение по значимым для проекта критериям, а также обоснование выбора системы.

Теоретические основы

Непрерывная интеграция (Continuous Integration, CI) и непрерывная поставка (Continuous Delivery, CD) представляют собой культуру, набор принципов и практик, которые

позволяют разработчикам чаще и надежнее разворачивать изменения программного обеспечения.

CI/CD – это одна из DevOps-практик. Она также относится к agile-практикам: автоматизация разворачивания позволяет разработчикам сосредоточиться на реализации бизнес-требований, на качестве кода и безопасности.

Непрерывная интеграция (CI, Continuous Integration) – практика разработки программного обеспечения, которая заключается в постоянном слиянии рабочих копий в общую основную ветвь разработки (до нескольких раз в день) и выполнении частых автоматизированных сборок проекта для скорейшего выявления потенциальных дефектов и решения интеграционных проблем. В обычном проекте, где над разными частями системы разработчики трудятся независимо, стадия интеграции является заключительной.

Непрерывная доставка (CD, Continuous delivery) – это подход к разработке программного обеспечения, при котором программное обеспечение производится короткими итерациями, гарантируя, что программное обеспечение является стабильным и может быть передано в эксплуатацию в любое время, а передача его происходит вручную. Целью является сборка, тестирование и релиз программного обеспечения с большей скоростью и частотой. Подход позволяет уменьшить стоимость, время и риски внесения изменений путем более частых мелких обновлений в продакшн-приложение.

При проведении анализа на необходимость внедрения процессов CI/CD в первую очередь необходимо определиться с частотой выпуска обновлений и размер команды. Как правило, когда над проектом работает большая команда программистов, часто необходимо объединять большое количество кода. По этой причине данная практика способна существенно сэкономить время и дать возможность тестирующему просматривать код позадачно.

Также данный подход позволит сэкономить много ресурсов компании при поддержке проекта.

Чаще всего процессы CI/CD используются при реализации сложных микросервисных приложений, где вручную сложно управлять множеством независимых систем.

Даже если выбранный проект построен на основе монолитной архитектуры, то экономия времени, которое разработчики могли потратить на доставку и разворачивание, окажется существенной.

Инструкции для pipeline могут содержать в себе команды сборки, установки зависимостей, смены переменных окружения, проведения автоматизированных тестов и пр.

На данный момент существуют разные решения для организации данных процессов. Среди них есть инструменты для настройки Pipeline на каждом сервисе для хранения репозитория (Github, GitLab, Bitbucket и др.). Есть бесплатные программные средства, такие как Jenkins. Есть платные системы, такие как TeamCity от компании JetBrains.

Выбор того или иного инструмента может зависеть от особенностей проекта, бюджета, опыта работы и прочих факторов.

Порядок выполнения лабораторной работы

При выполнении лабораторной работы «Поиск предметной области с потенциалом автоматизации» был выбран и описан проект. Для данного проекта необходимо составить список критериев, на основе которых будет выбрана подходящая система для организации CI/CD.

Например, для интернет-магазина будут важны следующие критерии:

- удобство системы и качество документации, т.к. от этого зависит скорость настройки процессов автоматической интеграции и доставки;
- возможность поддержки нескольких репозиторий, т.к. front-end часть магазина написана в виде отдельного SPA-приложения, а back-end часть реализована в виде микросервисов;
- бюджет, т.к. заказчик выделяет достаточно денег на реализацию;
- скорость доставки и разворачивания (при работе пяти разработчиков и 1 тестирующего).

Далее необходимо определиться с теми системами, среди которых необходимо провести выбор.

Рассмотрим бесплатную систему Jenkins, платную систему TeamCity и Pipeline от Bitbucket, т.к. все репозитории проектов находятся на Bitbucket.

В процессе выполнения лабораторной работы необходимо дать краткое описание всем системам. Для доступа к платной системе TeamCity возможно использовать бесплатную тестовую версию продукта или получить студенческую версию системы бесплатно на весь срок обучения в университете.

Исходя из описанных критериев и особенностей систем можно сделать вывод, что TeamCity является предпочтительным для использования с данным интернет магазином, т.к. это серьезный инструмент с хорошей документацией, позволяющий использовать разные системы для хранения репозитория. Система TeamCity уже настроена для работы с другими проектами в рамках данной организации, что значит, что не потребуется времени на развертывание и освоение данной системы.

1.8 Лабораторная работа «Основы настройки CI/CD»

Цель занятия

Изучение основы организации непрерывной доставки обновлений для информационной системы.

Форма проведения

Выполнение индивидуального задания.

Форма отчетности

На проверку должен быть предоставлен отчет с описанием Pipeline, необходимого для организации автоматизированных процессов тестирования и доставки программного продукта на сервер.

Теоретические основы

Непрерывная интеграция (Continuous Integration, CI) и непрерывная поставка (Continuous Delivery, CD) представляют собой культуру, набор принципов и практик, которые позволяют разработчикам чаще и надежнее развертывать изменения программного обеспечения.

При проведении анализа на необходимость внедрения процессов CI/CD в первую очередь необходимо определиться с частотой выпуска обновлений и размер команды. Как правило, когда над проектом работает большая команда программистов, часто необходимо объединять большое количество кода. По этой причине данная практика способна существенно сэкономить время и дать возможность тестировщику просматривать код позадачно.

Также данный подход позволит сэкономить много ресурсов компании при поддержке проекта.

Чаще всего процессы CI/CD используются при реализации сложных микросервисных приложений, где вручную сложно управлять множеством независимых систем.

Даже если выбранный проект построен на основе монолитной архитектуры, то экономия времени, которое разработчики могли потратить на доставку и развертывание, окажется существенной.

Процесс организации CI/CD сводится к написанию определенных инструкций, которые будут запускаться при отправке коммитов в репозитории. Данные инструкции называются Pipeline. Если для проекта описан данный pipeline, то система, хранящая репозиторий, при получении данных будет всякий раз запускать данные инструкции. В случае появления ошибок при выполнении данных предписаний система оповестит ответственного специалиста. В случае успешного проведения всех команд проект окажется развернут на указанной площадке. Как правило, для крупных проектов существуют несколько этапов публикации:

- develop - площадка для разрабатываемой версии, где все тестируется;

- prestage - площадка для тестирования сборки, готовой к отправке в production;
- stage - сервер, имитирующий боевую среду для проекта с целью проверить работоспособность в условиях обычного использования;
- production - сервер, на котором находится релизная версия продукта с целью ее использования настоящими пользователями.

Инструкции для pipeline могут содержать в себе команды сборки, установки зависимостей, смены переменных окружения, проведения автоматизированных тестов и пр..

На данный момент существуют разные решения для организации данных процессов. Среди них есть инструменты для настройки Pipeline на каждом сервисе для хранения репозитория (Github, GitLab, Bitbucket и др.). Есть бесплатные программные средства, такие как Jenkins. Есть платные системы, такие как TeamCity от компании JetBrains.

После выбора системы, на основе которой будет реализован процесс CI/CD необходимо определиться с последовательностью команд для автоматического выполнения. В общем случае процесс CI/CD делится на 3 части: настройка окружения, выполнение сборки проекта и проведение тестирования (при наличии автоматических тестов).

Настройка окружения чаще всего происходит с использованием технологии Docker для создания окружения на основе операционной системы Ubuntu. На данный образ устанавливаются зависимости. Например, если проект разрабатывался с применением библиотеки ReactJS, то будут установлены пакеты NodeJS и NPM. Пакеты устанавливаются исключительно из потребностей проекта в минимальном достаточном количестве.

Далее происходит сборка проекта. Типичные команды для развертывания проекта: установка зависимостей, установка конфигураций, создание базы данных, проведение миграций для базы данных, установка фикстур, сборка и архиваций ассетов, пр. Набор выполняемых команд также зависит от потребностей проекта.

Третий этап организации процесса CI/CD заключается в запуске автоматических тестов. Как правило, это запуск одной команды.

Завершающим этапом при успешном выполнении всех предыдущих этапов является отправка и развертывание проекта на сервере заказчика.

Порядок выполнения лабораторной работы

При выполнении лабораторной работы «Поиск предметной области с потенциалом автоматизации» был выбран и описан проект. Для данного проекта необходимо составить список критериев, на основе которых будет выбрана подходящая система для организации CI/CD.

Например, для интернет-магазина была выбрана система Bitbucket для реализации процессов CI/CD, т.к. репозиторий находится именно на данном сервисе.

По результатам изучения документации к данному продукту в первую очередь необходимо создать файл «bitbucket-pipelines.yml», содержащий инструкции для тестирования и развертывания проекта.

Пример листинга файла с описанием команд:

```
image: medtrainermx/php:5.6 #Выбор образа, на котором развернется проект
pipelines: # начало работы с Pipeline
  branches: # указание веток системы контроля версий
    '**': # команды для любых веток
      - step: # обозначение шага для развертывания
          name: "Build" # имя шага
          caches: # сообщаем об использовании composer
            • composer
          script: # указываем список скриптов
            - cp symfony/app/config/local.yml.dist symfony/app/config/local.yml
            - cp symfony/tests/config.local.php.dist symfony/tests/config.local.php
            - cd symfony/tests
```

```

- composer install artifacts:
- build/**
- symfony/vendor/**
- symfony/app/config/local.yml
- symfony/app/cache/**
- symfony/tests/vendor/**
- symfony/tests/config.local.php
- step: # следующий шаг для чистки кэша
  name: "Coding Standards"
  caches:
    - composer
  script:
    - cd symfony
    - mkdir app/cache
    - ./bin/phpcs --cache=app/cache/.phpcs-cache
- step: # шаг для запуска тестов
  name: "Unit Tests"
  services:
    - mysql caches:
    - composer script:
    - cd symfony
    - ./bin/phpunit --testsuite=unit --debug
definitions: # указываем, что системе необходимо учитывать при реализации
services: # обозначаем смежные сервисы для корректной работы
mysql: # имя сервиса
  image: mysql:5.7 # образ для запуска сервиса
  environment: # переменные окружения для сервиса
    MYSQL_DATABASE: 'test'
    MYSQL_ROOT_PASSWORD: 'password'

```

Как итог, в данном файле указаны три шага для любых веток СКВ: сборка, чистка кэша и запуск автоматических тестов. При выполнении лабораторной работы необходимо описать Pipeline для выбранного в лабораторной работе «Выбор предметной области с потенциалом автоматизации» проекта.

1.9 Лабораторная работа «Ценообразование при разработке и сопровождении информационных систем»

Цель работы

Формирование навыков по основам ценообразования в рамках разработки и сопровождения информационных систем.

Форма проведения

Выполнение индивидуального задания.

Форма отчетности

На проверку должен быть предоставлен отчет с описанием алгоритма действий и критериев для оценивания основной и дополнительных работ, а также итоговую сумму за полный цикл взаимодействия с клиентом с подробным описанием.

В отчете к данной лабораторной работе требуется подробно описать критерии, значимые для данной системы, заказчика и исполнителя, а также показание расчетов и обоснование расчетов с указанием итоговой стоимости оказания той или иной услуги. В рамках данной лабораторной работы необходимо просчитать стоимость разработки данной информационной системы с учетом внедрения, а также стоимость технического обслуживания с указанием перечня услуг. В случае необходимости оказания еще каких-либо услуг для реализации и внедрения выбранной информационной системы их также необходимо описать.

Теоретические основы

Расчет цены для разрабатываемого проекта напрямую зависит от множества факторов:

- сложность системы;
- используемые технологии;
- квалификация персонала;
- модель взаимодействия с клиентом;
- цена часа работы каждого из сотрудников;
- амортизация трат на организацию деятельности команды и пр.

Набор факторов для расчета стоимости может меняться от проекта к проекту и зависит от особенностей продукта и требований заказчика.

При оценке сроков и стоимости всегда стоит закладывать время на форс-мажоры, правки от заказчика и длительные интеграции с внешними сервисами (например, эквайринг для организации банки одобряют в течение недели).

При расчете стоимости технического сопровождения могут дополнительно учитываться следующие критерии:

- тип информационной системы;
- новизна информационной системы;
- наличие и квалификация персонала;
- длительность использования информационной системы;
- характеристики аппаратной части;
- качество кода;
- качество документации.

В общем случае стоимость может быть определена по следующей формуле:

$$s = (whc * hp) * wc + m + d + n,$$

где s – стоимость проекта, whc – количество часов, затраченных сотрудником, hp – цена часа работы сотрудника, wc – кол-во сотрудников, m – маржа компании, d – дополнительные траты (например, софт для разработки, аренда тестовых серверов и пр.), n – налоги.

Это универсальная формула, которая может меняться в зависимости от способа расчета.

Некоторые компании рассчитывают час работы компании, а после при оценке проекта просто умножают временные затраты на проект на цену часа работы компании. Но данная практика подходит не всем и является гораздо более сложной для обоснования.

Порядок выполнения лабораторной работы

При выполнении лабораторной работы «Поиск предметной области с потенциалом автоматизации» был выбран и описан проект. Для данного проекта необходимо провести анализ стоимости разработки и технической поддержки.

В первую очередь необходимо рассчитать кол-во затраченного времени для того или иного сотрудника. Для этого необходимо декомпозировать макро-задачи на микро-задачи и определить ориентировочное время выполнения той или иной задачи. Пример данной декомпозиции можно увидеть в таблице 1. Стоимость разработки данного магазина составила 486 000 рублей. Далее расчеты будут проводиться с округлением до тысяч для простоты расчетов. В данном случае это цена для выполнения всех работ и привлечения стороннего дизайнера. Здесь отсутствует выгода для владельца компании-исполнителя, налоги и амортизирующие траты. В случае наличия клиентского менеджера проекта к стоимости проекта добавится еще 48 000 рублей (10% за ведение клиента). Тогда стоимость составит 534 600 рублей. Если над проектом работают 2 специалиста, то необходимо 2 лицензии на программное обеспечение для разработки на 3 месяца (расчет сроков производился в описании лабораторной работы «Поиск предметной области с потенциалом автоматизации»), что составит 60 долларов или 4 500 рублей. На данный момент результирующая сумма составляет 540 000 рублей. Выгода, которую хочет себе за данный проект владелец компании-исполнителя, составляет 10%. Тогда стоимость проекта с учетом маржинальности составит уже 594 000 рублей. Также необходимо

учесть налоги. Если наша организация является ИП на УСН 6%, то итоговая цена проекта составит $594\,600 * 1,06 \approx 630\,000$ рублей.

Для расчета стоимости технической поддержки учитываются иные факторы. При стоимости часа работы программиста в 1 000 рублей, допустим, был подписан договор на обслуживание интернет-магазина в течение 20 часов каждый месяц без учета правок ошибок. В связи с тем, что документацию писала та же компания-исполнитель при разработке, то она по определению является качественной. Также, в связи с тем, что код писала та же компания-исполнитель при разработке, то он по определению является качественным. Исходя из этого, увеличения стоимости часа за обслуживание данного интернет магазина не будет производиться. За каждый месяц необходимо оплачивать разработчику софт для разработки по цене 10 долларов или 750 рублей в месяц. Маржинальность договора составит 10% от суммы. Организация работает как ИП на УСН 6%, соответственно налоги составляют 6%.

Итоговый расчет стоимости технического сопровождения: (ежемесячные часы * стоимость часа работы сотрудника + дополнительные траты) * маржинальность * налоги = $(20 * 1000 + 750) * 1.1 * 1.06 \approx 25\,000$ рублей.

2 МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ПРОВЕДЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

2.1 Общие положения

Целями самостоятельной работы является систематизация, расширение и закрепление теоретических знаний, приобретение навыков научно-исследовательской и производственно-технологической деятельности.

Самостоятельная работа по дисциплине «Внедрение и сопровождение информационных систем» включает следующие виды активности обучающегося:

- проработка лекционного материала;
- изучение тем (вопросов) теоретической части дисциплины, вынесенных для самостоятельной подготовки;
- подготовка к лабораторным работам;
- подготовка к промежуточной аттестации.

2.2 Проработка лекционного материала

Для проработки лекционного материала обучающимся рекомендуется воспользоваться конспектом, сопоставить записи конспекта с соответствующими разделами учебных материалов.

Целесообразно ознакомиться с информацией, представленной в файлах, содержащих презентации лекций, предоставляемых преподавателем.

Рекомендуется сформулировать вопросы преподавателю и задать их либо посредством электронной образовательной среды вуза, либо перед началом следующей лекции.

2.3 Изучение тем (вопросов) теоретической части дисциплины, вынесенных для самостоятельной подготовки

Перечень вопросов, подлежащих изучению:

1. Методы поиска экономически-выгодных ниш. Необходимо знать:
 - 1.1. Customer Development – метод поиска ниш через изучение потребностей потенциальных пользователей.
 - 1.2. Lean Canvas – методология описания ниши для организации бизнеса.
 - 1.3. Критерии оценки экономической привлекательности IT-проектов.
2. Гибкие модели разработки. Необходимо знать:
 - 2.1. Agile.
 - 2.2. Kanban
 - 2.3. Scrum
3. Основы написания технического задания. Необходимо знать:
 - 3.1. ГОСТ 19.106
4. Жизненные циклы программных продуктов. Необходимо знать:
 - 4.1. Каскадная модель ЖЦ.
 - 4.2. Итеративная модель ЖЦ.
 - 4.3. Водопадная модель ЖЦ.
 - 4.4. Спиральная модель ЖЦ.
5. Автоматизированное тестирование программного обеспечения.
6. DevOps. Необходимо знать:
 - 6.1. Цели и задачи практики DevOps.
 - 6.2. Принципы DevOps.

6.3. Методы и средства реализации DevOps.

6.4. Когда применяется DevOps.

Методические рекомендации по изучению

Рекомендуется выбрать несколько систем и провести полный анализ всех этапов жизненного цикла с расчетом их стоимости, определением заинтересованных лиц и критериев качества выполнения. Вопросы рекомендуется задать преподавателю.

2.4 Подготовка к лабораторным работам

Подготовка к лабораторным работам включает повторение теоретического материала, необходимого для их выполнения, полученного обучающимися в рамках лекционной части курса и ознакомления с рекомендуемыми источниками.

2.5 Подготовка к промежуточной аттестации

Подготовка к промежуточной аттестации осуществляется по вопросам, приведенным в рабочей программе дисциплины.

ЗАКЛЮЧЕНИЕ

Выполнение методических указаний к лабораторным работам и организации самостоятельной работы по дисциплине «Внедрение и сопровождение информационных систем» способствует успешному освоению дисциплины, в том числе выработке навыков по решению практических задач, свойственных этапам итогового тестирования, внедрения, сопровождения и ликвидации программного продукта.

В целом дисциплина направлена на ознакомление студентов с особенностями организации процессов внедрения и обновления программного продукта, проведения приемочных испытаний и опытной эксплуатации, взаимодействия как с представителями заказчика, так и с представителями команды разработки. Дисциплина способствует развитию навыков проведения диагностики и оценки качества реализации тех или иных процессов в рамках описанных этапов, а также оценки последствий реализации действий, направленных на развитие информационных систем, что является важным для принятия эффективных управленческих решений.

ЛИТЕРАТУРА

1. Ехлаков, Ю. П. Организация бизнеса на рынке программных продуктов: Учебник [Электронный ресурс] / Ю. П. Ехлаков –Томск: ТУСУР, 2012. – 314 с. – Режим доступа: <https://edu.tusur.ru/publications/970>
2. Сенченко, П. В. Методы контроля и оценки качества программного обеспечения: Методические указания к лабораторным работам и организации самостоятельной работы [Электронный ресурс] / П. В. Сенченко. – Томск: ТУСУР, 2018. – 38 с. – Режим доступа: <https://edu.tusur.ru/publications/7780>
3. ГОСТ Р ИСО/МЭК 12207-2010. Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств: Документ. – 188 с. – Режим доступа: <http://docs.cntd.ru/document/gost-r-iso-mek-12207-2010>