

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт Космических и Информационных Технологий

институт

Информационные системы

кафедра

УТВЕРЖДАЮ

Заведующий кафедрой

_____ П. П. Дьячук
подпись инициалы, фамилия

« ___ » _____ 2019 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.02 – Информационные системы и технологии

место прохождения практики

Интернет платформа для оказания услуг по ремонту объектов капитального
строительства

Руководитель

подпись, дата

доцент, канд. техн. наук

Р.В. Брежнев

Выпускник

подпись, дата

Д.Д Чернов

Красноярск 2019

Продолжение титульного листа бакалаврской работы по теме «Интернет платформа для оказания услуг по ремонту объектов капитального строительства»

Нормоконтролер

подпись, дата

Ю.В. Шмагрис
инициалы, фамилия

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1 Обзор и анализ аналога.....	6
1.1 Модель AS-IS.....	6
1.2 Вывод по первой главе	11
2 Анализ требований.....	13
2.1 Модель TO-BE.....	13
2.2 Диаграмма вариантов использования (Use-Case).....	20
2.3 Диаграмма деятельности.....	21
2.3.1 Описание ключевых прецедентов.....	21
2.3.2 Прецедент К1. Вход в систему	21
2.3.3 Прецедент К2. Добавление новой заявки.....	24
2.3.4 Прецедент К3. Выбор исполнителя	27
2.4 Диаграмма компонентов и диаграмма развертывания.....	29
2.5 Вывод по второй главе	30
3 Разработка программного прототипа.....	31
3.1 Диаграмма классов.....	31
3.2 Работа с сервисом	36
3.2.1 Главная страница сайта	37
3.2.2 Регистрация и вход в систему	38
3.2.3 Добавление новой заявки (проекта).....	40
3.3 Вывод по третьей главе	42
ЗАКЛЮЧЕНИЕ	43
СПИСОК СОКРАЩЕНИЙ.....	44
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	45
ПРИЛОЖЕНИЕ А Проект технического задания	46
ПРИЛОЖЕНИЕ Б Акт об использовании	57
ПРИЛОЖЕНИЕ В Программный код капчи.....	58

ВВЕДЕНИЕ

В наше время Интернет – это одно из наиболее важных средств массовой информации и коммуникации, которое находится по своей значимости в одном ряду с прессой, телефоном и телевидением. Начиная с 1969 года, мировая сеть – Интернет стремительно развивалась, в наше время превратилась в одно из основных средств, используемых для общения, развлечений, рекламы, торговли и в источник разнообразной полезной информации для любой категории пользователей. Быстро растет количество изданий, посвященных сети Интернет, что предвещает широкое ее распространение даже в далеких от техники областях.

В Сети размещено значительное количество разнообразных специализированных сервисов, отвечающих широкому спектру тематических запросов пользователей. Однако функциональные возможности даже самых популярных сервисов не всегда в полной мере отвечают динамичному развитию информационных потребностей пользователей. Не исключением являются и сервисы, ориентированные на запросы пользователей, связанные с предоставлением услуг по ремонту объектов капитального строительства. Большинство таких сервисов представляют собой сайт-визитку или более сложный информационный портал, имеющий не только контактную информацию, но и личный кабинет, форму для заказа и т.д. Так же следует выделить сайты с доской объявлений. Чаще всего владельцы таких сайтов ведут информативную деятельность. Информация собирается и размещается на некой платформе для общего использования пользователями Сети.

В данной выпускной квалификационной работе осуществляется создание платформы для общения клиента со специалистом в сфере ремонта недвижимости. Заказчиком является общество с ограниченной ответственностью (ООО «Рассвет+»), занимающееся капитальным ремонтом общих помещений в многоквартирных домах. В связи с расширением спектра предоставляемых услуг заказчик решил создать Интернет платформу для оказания услуг по ремонту объектов капитального строительства. Платформа представляет собой доску объявлений. Посетители сайта смогут поместить свое объявление для выполнения ремонтных и/или строительных работ специалистами не только данной компании, но и сторонних организаций или частных мастеров. Любой желающий сможет разместить объявления для оказания строительных или ремонтных услуг. При этом регистрация бесплатна, но заключение сделки с заказчиком подразумевает оплату комиссии исполнителем в размере 10% от стоимости заказа. Данное решение стало оптимальным и наиболее актуальным, так как объемы работ возросли, а количество кадров осталось прежним. Более подробный функционал сайта и требования к нему будут рассмотрены в техническом задании.

Заказчик за время своего существования смог собрать хорошую клиентскую базу. Это позволит получить выгоду не только с выполненных работ самим заказчиком, но и процент со сделок сторонних организаций, исполняя роль посредника между заказчиками и исполнителями. Именно поэтому использование аналогичных сторонних ресурсов не подходит для заказчика.

Целью выпускной квалификационной работы является извлечение прибыли из имеющейся информации о потребностях клиентов посредством Интернет платформы.

Чтобы достичь цели, нам необходимо выполнить поставленные задачи:

- a) Обзор и анализ существующих решений в выбранной специфике;
- b) Выявление, анализ и спецификация требований;
- c) Проектирование информационной системы;
- d) Программное прототипирование информационной системы.

1 Обзор и анализ аналога

1.1 Модель AS-IS

В данной работе рассматриваются программные приложения и сервисы, предоставляющие услуги в сфере ремонта объектов капитального строительства. Наиболее популярными сервисами в этой области являются следующие:

- remontnik.ru — сервис, предоставляющий возможность вызова мастера по ремонту;
- youdo.ru — сервис, позволяющий оставить заявку для выполнения того или иного задания, включая так же возможность вызова мастера по ремонту;
- rydo.ru — аналогичный вышеупомянутому аналогу, широко направленный сервис для выполнения работы.

Вызов включает в себя несколько этапов. В первую очередь дается краткое название и описание (проблемы, сроки, материалы и другая полезная исполнителю информация), на следующем этапе заказчику необходимо выбрать бюджет, указать время начала работ, место работ (адрес) и указать контактные данные: логин, e-mail, номер телефона. После принятия соглашения о правилах сервиса заказ можно добавить. Далее заказчик ожидает откликов исполнителей, предложивших свои услуги, из которых он, опираясь на портфолио и отзывы, может сделать свой выбор.

На основе аналога были построены функциональные модели «AS-IS» и «TO-BE»

Модель «AS-IS» даст нам возможность провести анализ функционирования аналога и определить его недостатки. После, учитывая полученную информацию, будет построена модель «TO-BE». Далее на рисунках 1 – 7 представлена декомпозиция модели AS-IS.

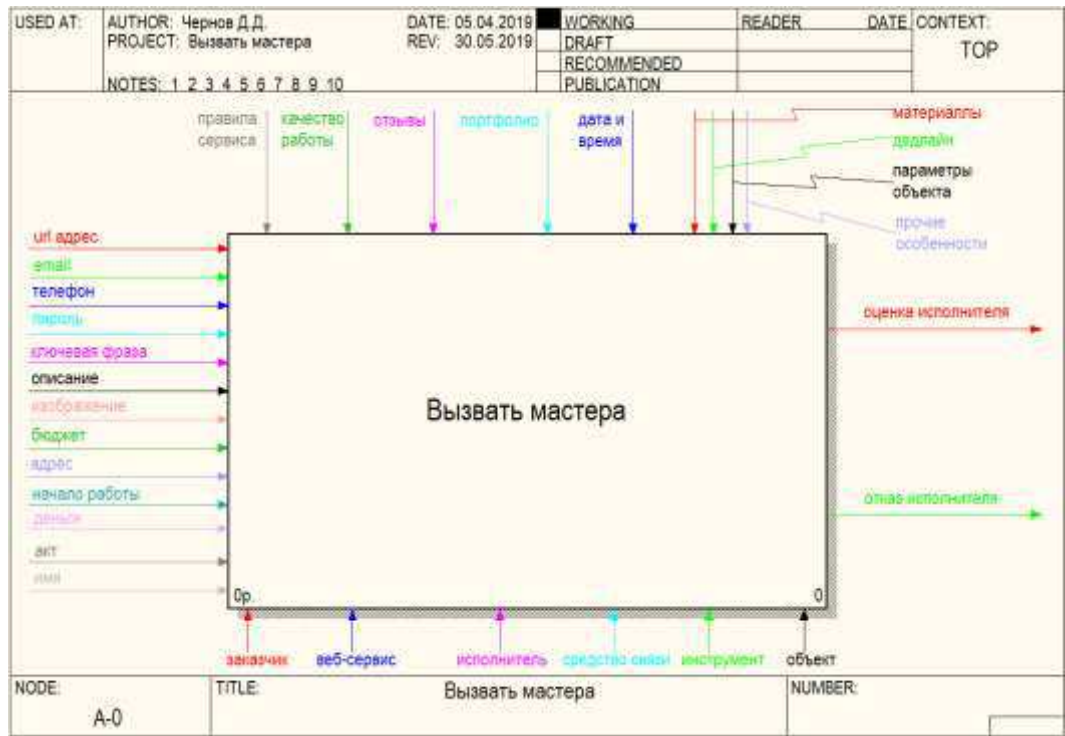


Рисунок 1 – Контекстная диаграмма «Вызвать мастера» модели AS-IS

Рисунок 2 иллюстрирует декомпозицию функционального блока «Вызвать мастера». Данная диаграмма описывает действия пользователя, начиная от входа на сайт и заканчивая оценкой выполненной работы. Более подробно каждый этап будет описан на следующих диаграммах.

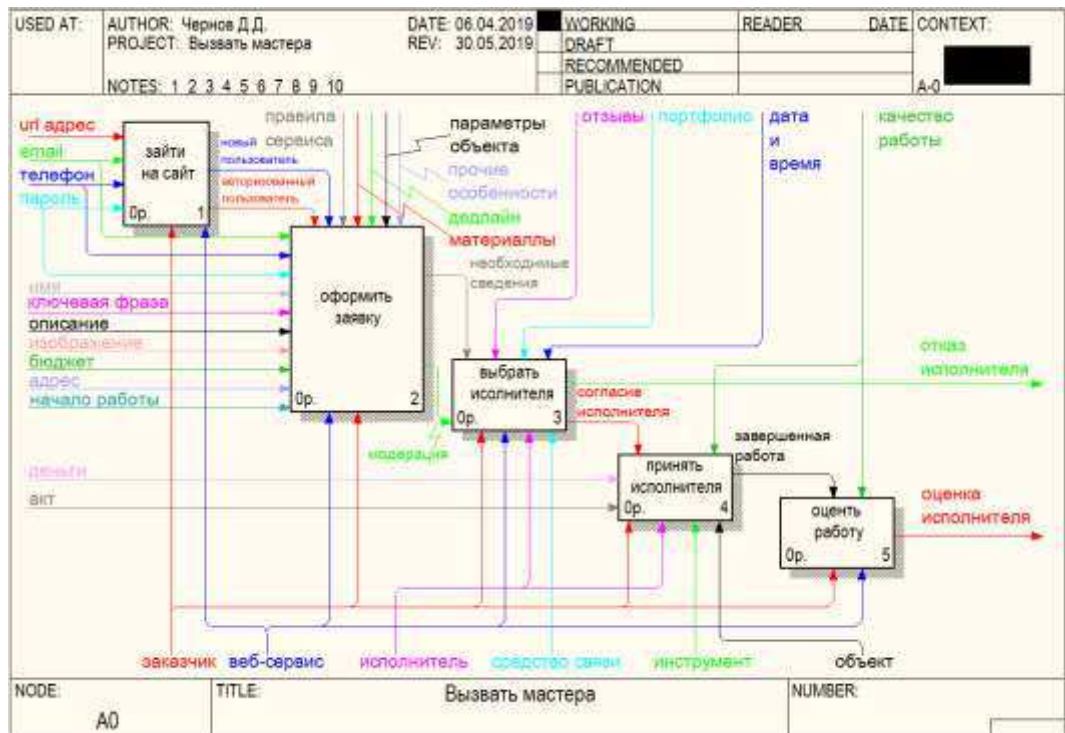


Рисунок 2 – Декомпозиция блока «Вызвать мастера»

Декомпозиция функционального блока «Зайти на сайт» представлена на рисунке 3. Диаграмма дает понимание процесса о первых действиях пользователя.

Если пользователь (далее заказчик) уже зарегистрирован в системе (далее веб-сайт), ему необходимо ввести телефон (или e-mail) и пароль.

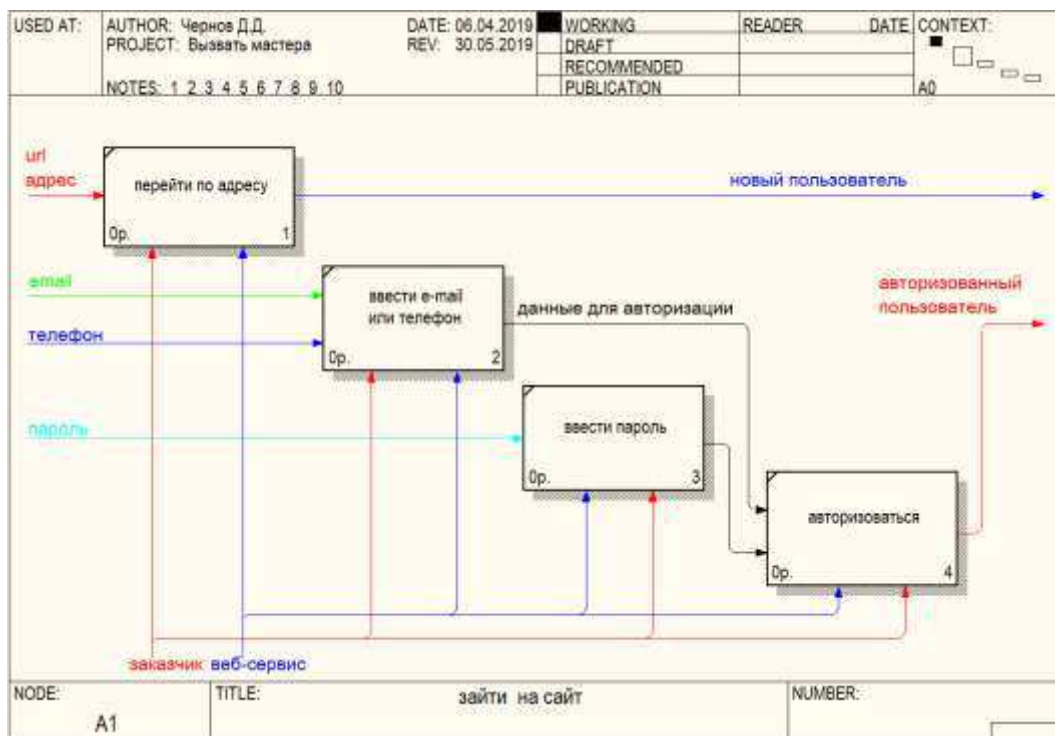


Рисунок 3 – Декомпозиция блока «зайти на сайт»

В противном случае, как видно на рисунке 3, новый пользователь сразу переходит к следующему функциональному блоку «оформить заявку», декомпозиция которого представлена на рисунке 4. На этом этапе, руководствуясь информацией статуса пользователя (новый пользователь), заказчику будет предложено пройти регистрацию, указав свои данные и тут же ввести всю нужную информацию для создания заявки. На выходе мы получим необходимые сведения, и начнется проверка корректности введенной информации (модерация).

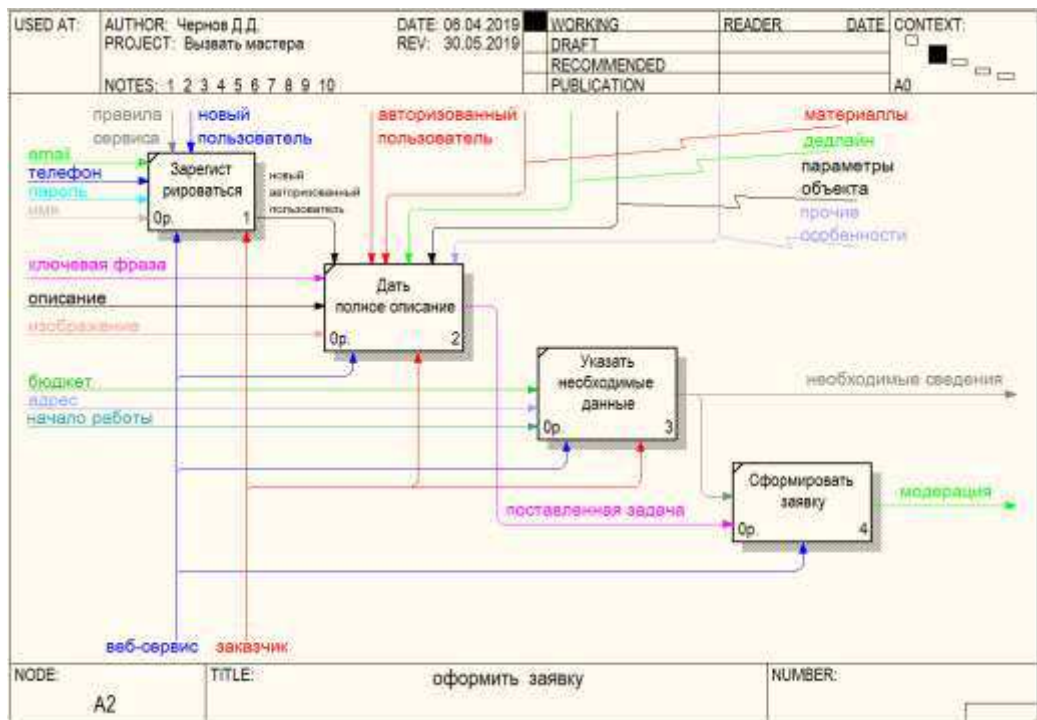


Рисунок 4 – Декомпозиция блока «оформить заявку»

Далее заказчику остается дождаться окончания модерации, и отклика(-ов) от исполнителя(-ей). Выбрать понравившегося исполнителя, опираясь на отзывы и портфолио, и связаться с ним, используя средства связи (телефон или сообщение на сайте). Данный этап представлен на рисунке 5. Выходными данными этой диаграммы будет отказ исполнителя или согласие.

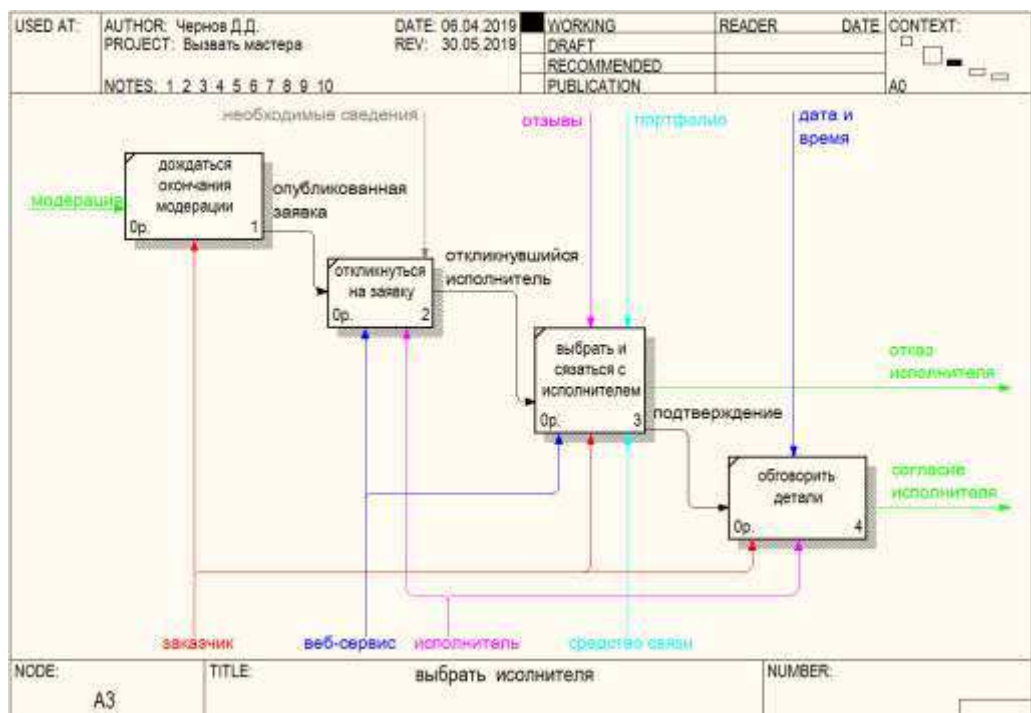


Рисунок 5 – Декомпозиция блока «выбрать исполнителя»

В случае согласия исполнителя заказчик ожидает прихода мастера (бригады), дает разрешение на начало работ. Исполнитель, используя инструменты, ведет работу над объектом. Закончив, заказчик проверяет готовую работу, после чего принимает её, руководствуясь качеством выполненной работы. Приняв работу, заказчик производит оплату, после чего работа считается завершенной, а исполнитель выдает заказчику документ подтверждающий проведение работ. Диаграмма этого этапа представлена на рисунке 6.

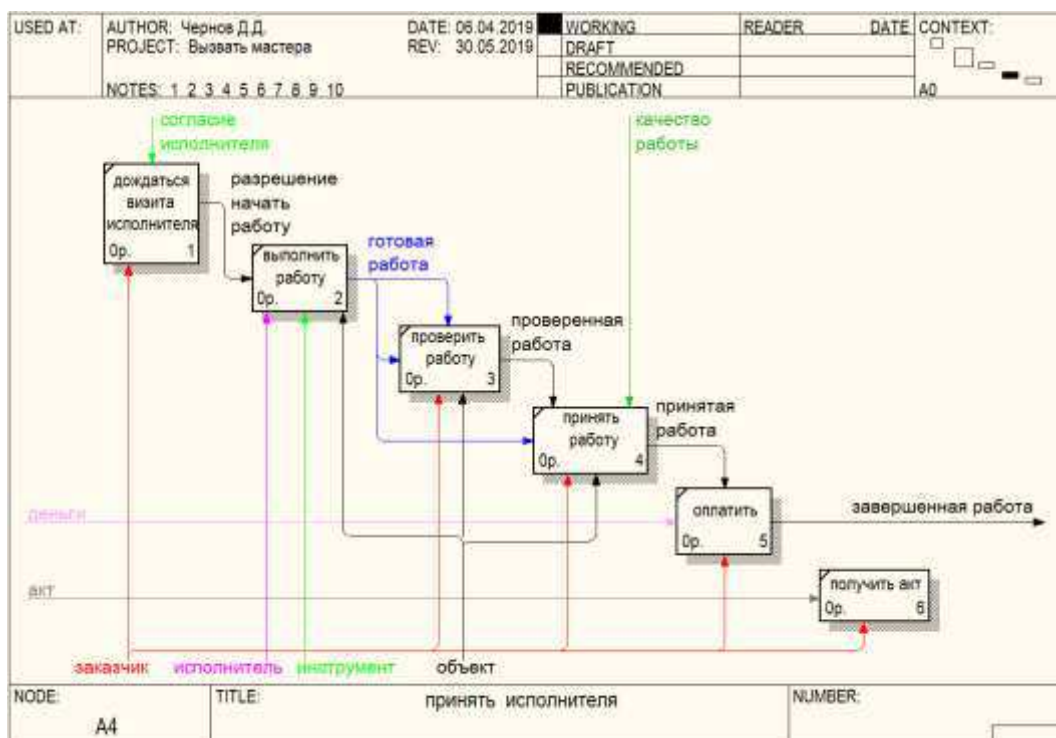


Рисунок 6 – Декомпозиция блока «принять исполнителя»

Последним этапом является написание отзыва и общая оценка исполнителю, опираясь на результат завершенной работы (полнота выполненной работы, срок окончания) и качество выполненной работы. На выходе мы получим отзыв и оценку. Декомпозиция функционального блока «оценить работу» представлена на рисунке 7.

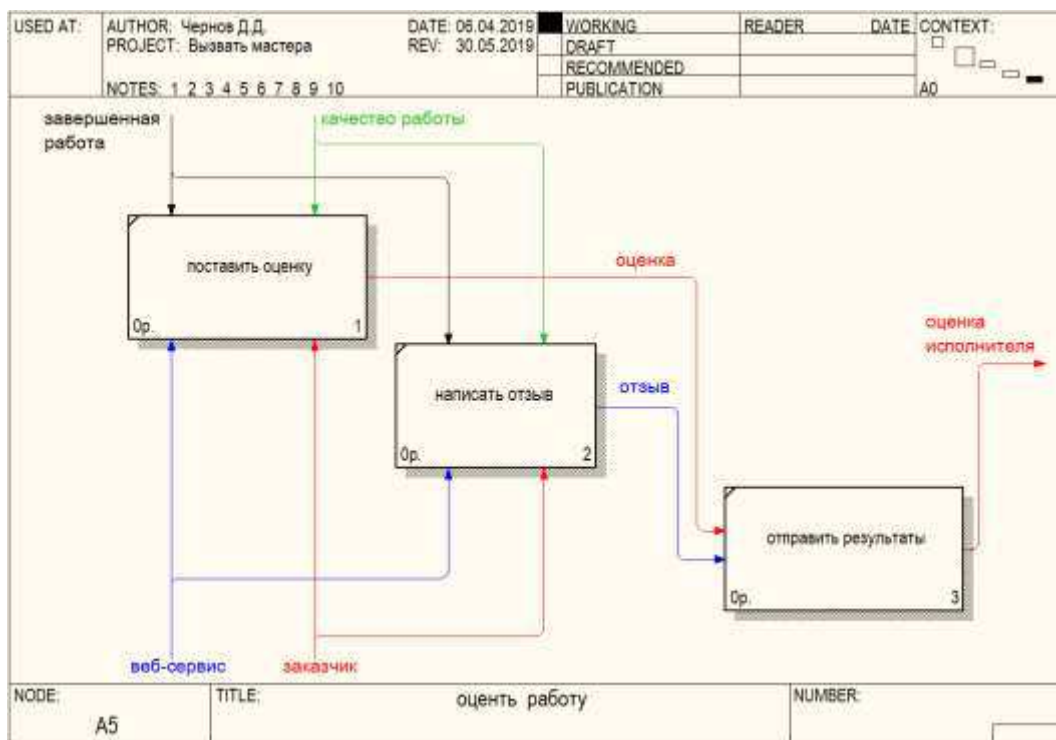


Рисунок 7 – Декомпозиция блока «оценить работу»

Построенные декомпозиционные диаграммы помогли выявить недостатки. К их числу можно отнести процесс модерации (проверки менеджером введенных данных и определение категории). Так как этот процесс занимает время, то его устранение позволит ускорить оформление заявки, а выбор категории работ предложить самому заказчику. В модели AS-IS: заказчик заводит заявку → исполнитель откликается на заявку → заказчик выбирает исполнителя. В модели TO-BE принцип работы изменен: заказчик заводит заявку → заказчик выбирает исполнителя → исполнитель откликается (положительно или отрицательно). Таким образом, заказчик будет иметь возможность рассмотреть полную базу исполнителей по категории, не ограничиваясь выбором откликнувшихся исполнителей.

1.2 Вывод по первой главе

В данной главе были проанализированы аналоги разрабатываемой Интернет платформы. По результатам анализа была построена модель бизнес-процесса «Вызвать мастера», являющегося общим для всех аналогов. В итоге данная модель помогла выявить функциональные возможности и недостатки существующей системы.

Функциональные возможности системы:

- Создание новой заявки в системе;
- Выбор исполнителя для проекта;

- Регистрация пользователей на сайте.

Недостатки системы:

- модерация администратором (так как этот процесс занимает время, то его устранение позволит ускорить оформление заявки);
- алгоритм назначения исполнителем (заказчик заводит заявку → исполнитель откликается на заявку → заказчик выбирает исполнителя).

2 Анализ требований

2.1 Модель ТО-ВЕ

Декомпозиционные диаграммы помогли построить новую модель ТО-ВЕ, приняв во внимание обнаруженные недостатки. Был изменен принцип работы и отменен процесс модерации заявки. А для того чтобы защититься от спама было решено использовать капчу (картинка с текстом) при регистрации нового пользователя.

На рисунках с 8 по 15 представлена модель ТО-ВЕ и её декомпозиционные блоки.

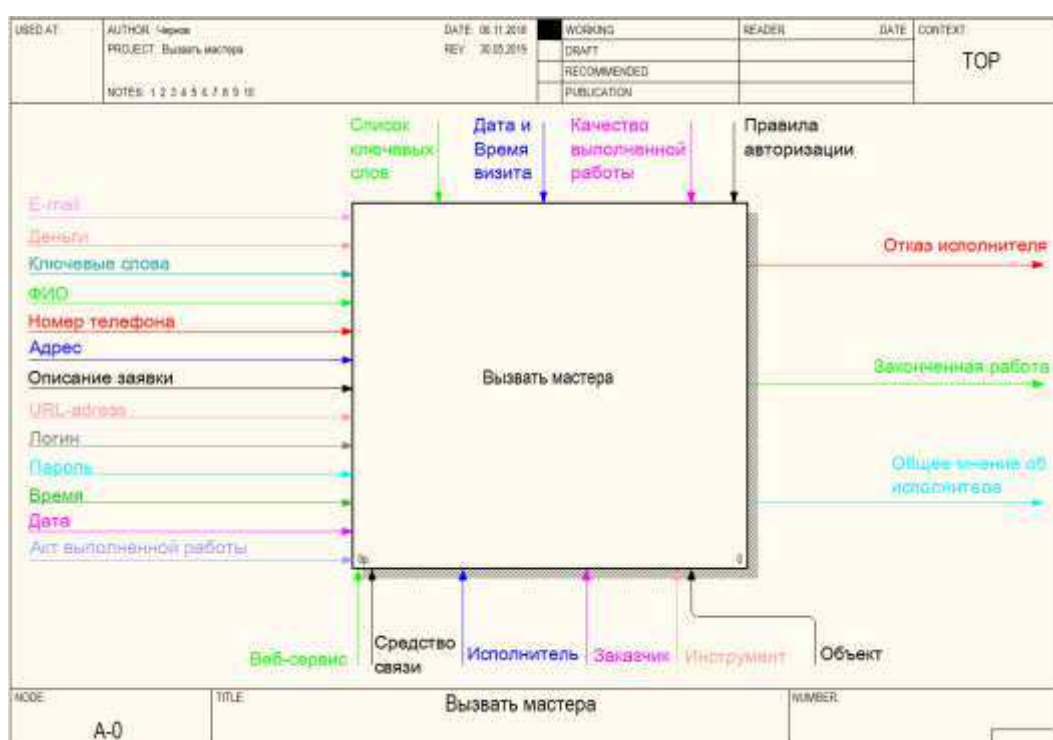


Рисунок 8 – Композитная диаграмма «Вызвать мастера» модели ТО-ВЕ

На рисунке 9 представлена декомпозиция функционального блока «Вызвать мастера». Диаграмма иллюстрирует процесс вызова мастера, начиная от входа на сайт до написания отзыва исполнителю. В качестве *управляющей информации* выступают правила авторизации, список ключевых слов, дата и время визита, качество выполненной работы. *Механизмы*, участвующие в процессе: веб-сайт, заказчик, исполнитель, инструмент, объект и средство связи. *Входные данные*, введенные пользователем, необходимые для реализации процессов: URL-адрес, E-mail, ФИО, номер телефона, логин, пароль, ключевые слова, описание заявки, адрес, время, дата, деньги, акт выполненной работы. В качестве *выходных*

данных мы получаем отказ исполнителя или завершённую работу с отзывом и оценкой.

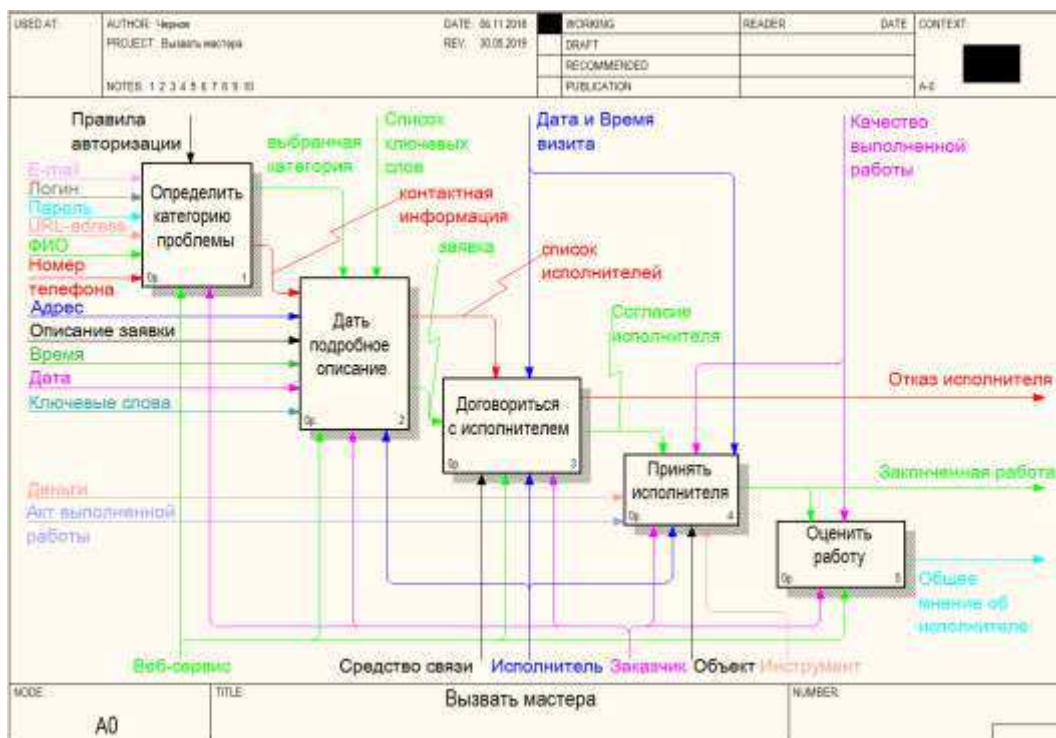


Рисунок 9 – Декомпозиция блока «Вызвать мастера»

На рисунке 10 представлена декомпозиция функционального блока «Определить категорию». Диаграмма иллюстрирует процесс выбора категории пользователем, предварительно пройдя процесс идентификации. В качестве *управляющей информации* выступают правила авторизации. *Механизмы*, участвующие в процессе: веб-сайт и заказчик. *Входные данные*, введенные пользователем, необходимые для регистрации или авторизации: URL-адрес, E-mail, ФИО, номер телефона, логин, пароль. В качестве *выходных данных* мы получаем контактную информацию о заказнике и выбранную категорию.

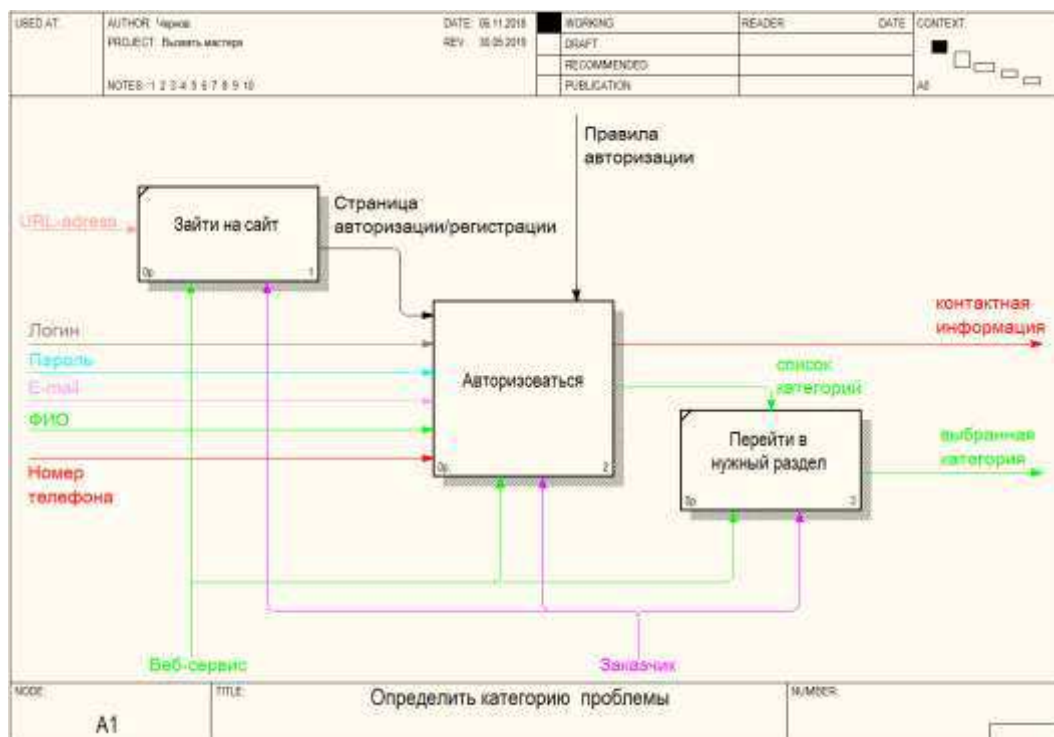


Рисунок 10 – Декомпозиция блока «Определить категорию»

На рисунке 11 представлена декомпозиция функционального блока «Авторизоваться». Диаграмма иллюстрирует процесс авторизации/регистрации пользователя на сайте. При этом регистрация происходит с защитой от ботов (посредством капчи). В качестве *управляющей информации* выступают правила авторизации. *Механизмы*, участвующие в процессе: веб-сайт и заказчик. *Входные данные*, введенные пользователем, необходимые для регистрации или авторизации: URL-адрес, E-mail, ФИО, номер телефона, логин, пароль и страница авторизации/регистрации. В качестве *выходных данных* мы получаем контактную информацию и список категорий.

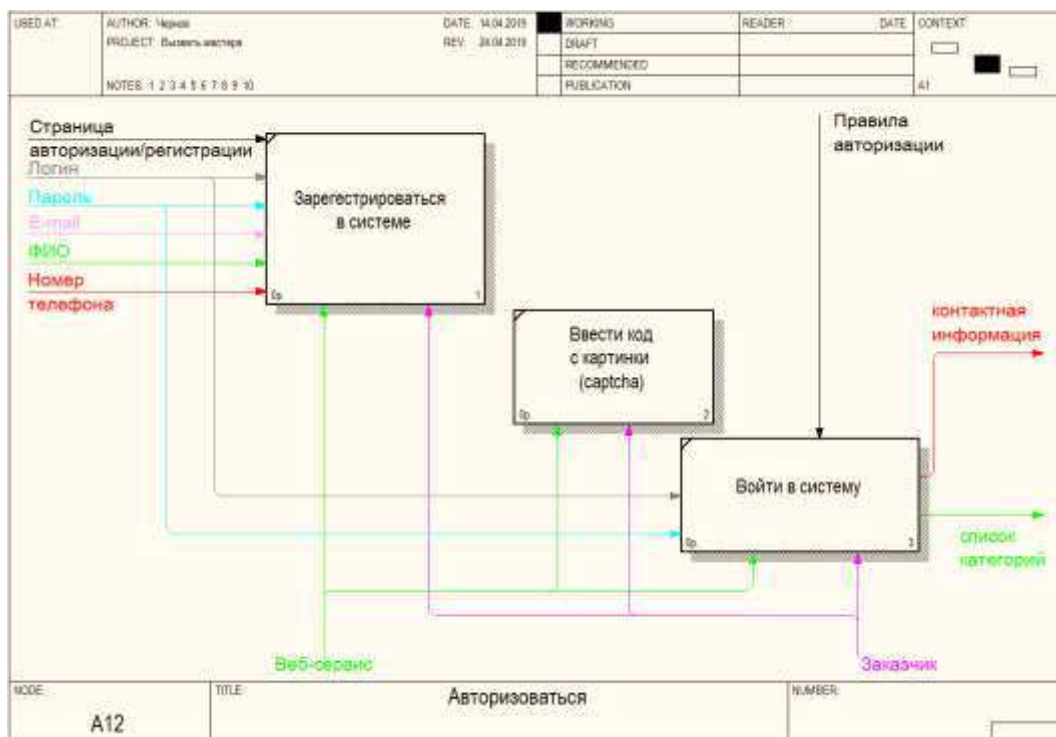


Рисунок 11 – Декомпозиция блока «Авторизоваться»

На рисунке 12 представлена декомпозиция функционального блока «Дать подробное описание». Диаграмма иллюстрирует процесс сбора данных для формирования заявки. В качестве *управляющей информации* выступает список ключевых слов и выбранная категория. *Механизмы*, участвующие в процессе: веб-сайт, заказчик и исполнитель. *Входные данные*, введенные пользователем, необходимые для подробного описания заявки: ключевые слова, описание заявки, адрес, время, дата. На выходе из каждого функционального блока мы получаем данные для создания заявки: набор ключевых слов, описанная проблема, личные данные и желаемое дата/время. В качестве *выходных данных* мы получаем два параметра: список исполнителей и заявка.

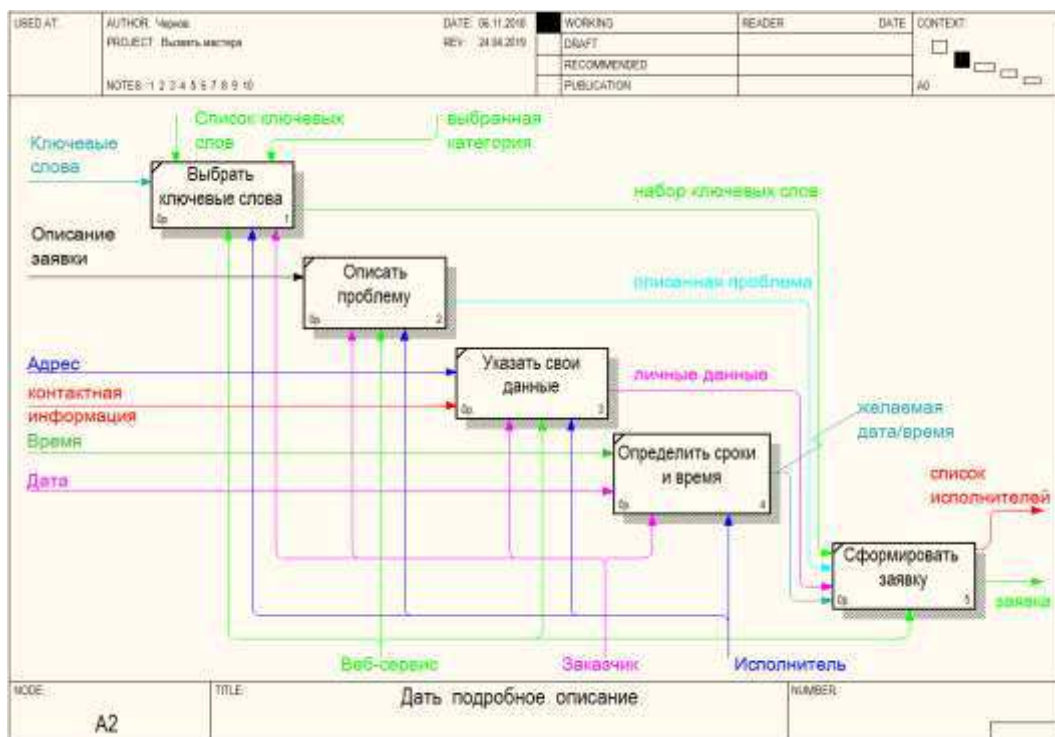


Рисунок 12 – Декомпозиция блока «Дать подробное описание»

На рисунке 13 представлена декомпозиция функционального блока «Договориться с исполнителем». Диаграмма иллюстрирует процесс сбора данных для формирования заявки. В качестве *управляющей информации* выступает список исполнителей и «Дата и время визита». *Механизмы*, участвующие в процессе: веб-сайт, заказчик, исполнитель и средство связи (телефон, ЛС онлайн). *Входные данные* представляют собой сформированную заявку. В качестве *выходных данных* мы получаем отказ или согласие исполнителя.

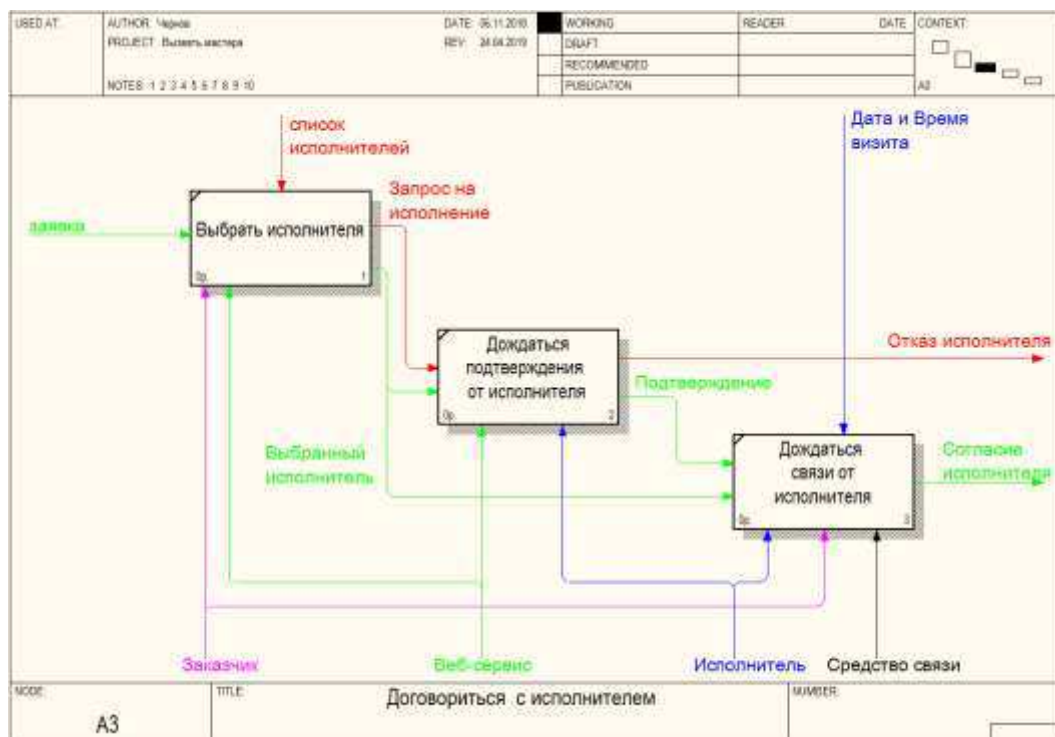


Рисунок 13 – Декомпозиция блока «Договориться с исполнителем»

На рисунке 14 представлена декомпозиция функционального блока «Принять исполнителя». Диаграмма иллюстрирует процессы, начиная от «Дождаться визита исполнителя» до «Получить акт о проделанной работе». В качестве *управляющей информации* выступает согласие исполнителя, дата и время визита, качество выполненной работы. *Механизмы*, участвующие в процессе: заказчик, исполнитель инструмент и объект. *Входные данные* являются деньги, которыми рассчитывается заказчик и акт, который он получает после оплаты. В качестве *выходных данных* мы получаем законченную работу.

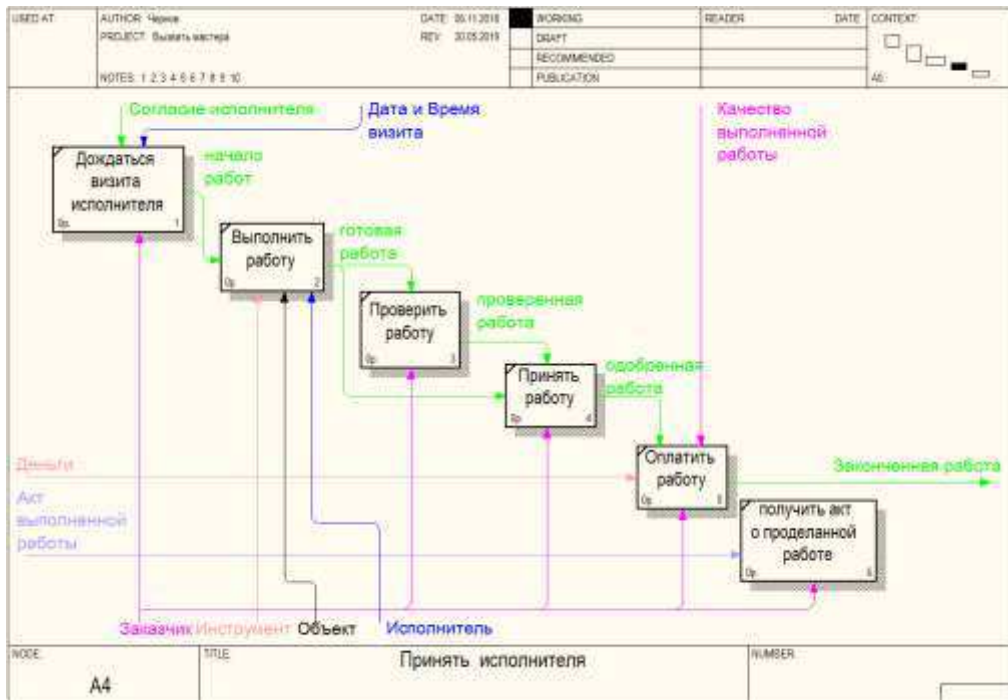


Рисунок 14 – Декомпозиция блока «Принять исполнителя»

На рисунке 15 представлена декомпозиция функционального блока «Оценить работу». Диаграмма иллюстрирует процессы написания отзыва и выставления оценки исполнителю. В качестве *управляющей информации* выступает законченная работа и качество выполненной работы. *Механизмы*, участвующие в процессе: заказчик и веб-сайт. *Входные данные* у нас отсутствуют. В качестве *выходных данных* мы получаем оценку и отзыв.

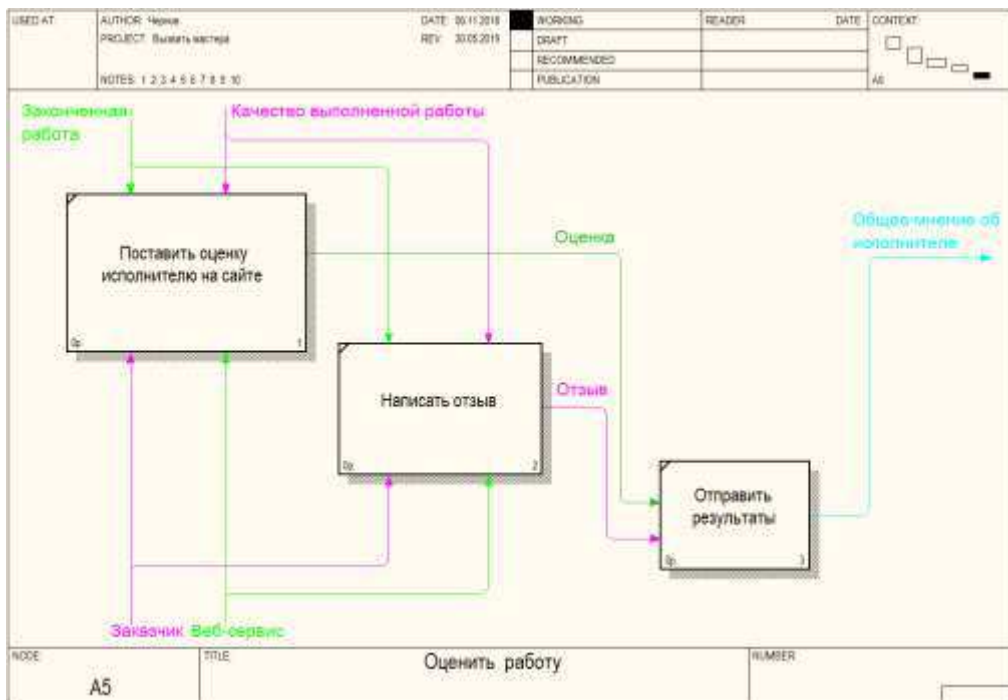


Рисунок 15 – Декомпозиция блока «Оценить работу»

Функциональная модель «ТО-ВЕ» составлена с учетом следующих недостатков аналога: отменили процесс модерации, взамен предложили заказчику выбрать категорию и изменили принцип назначения исполнителя.

Далее на основе этой модели будет разработана платформа для общения клиента со специалистом в сфере ремонта недвижимости.

2.2 Диаграмма вариантов использования (Use-Case)

На рисунке 16 представлена диаграмма вариантов использования.

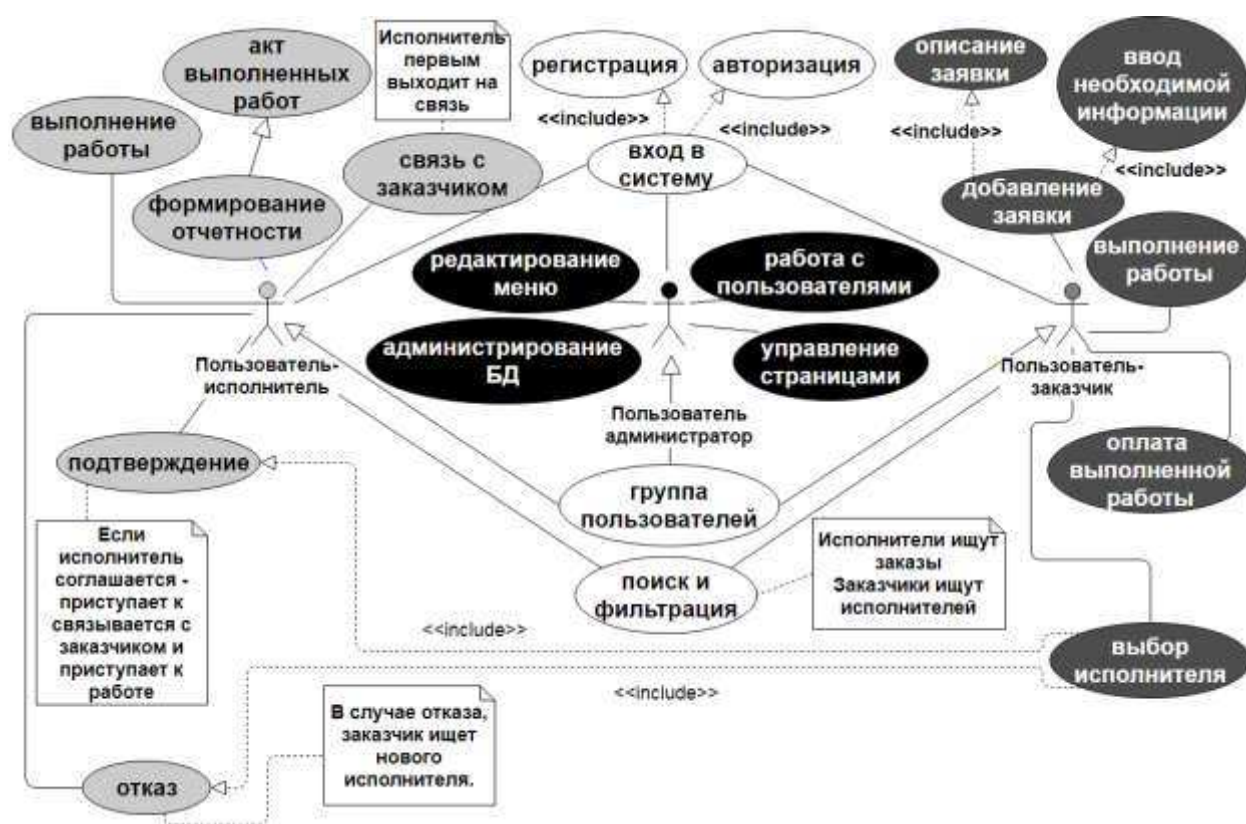


Рисунок 16 – Диаграмма вариантов использования

При построении диаграммы в первую очередь были созданы 3 актора: Пользователь-исполнитель, Пользователь-Заказчик и Пользователь-администратор. Затем были смоделированы все прецеденты (сценарии поведения), а после настроены внутренние связи между прецедентами и связи между прецедентами и акторами.

2.3 Диаграмма деятельности

2.3.1 Описание ключевых прецедентов.

По методологии RUP описание прецедента должно содержать следующие пункты:

- краткое описание прецедента;
- описание основного потока;
- описание альтернативных потоков;
- специальные требования;
- предусловия;
- постусловия;
- точки расширения.

Для дальнейшего рассмотрения были выбраны три прецедента:

- К1. Вход в систему;
- К2. Добавление заявки;
- К3. Выбор исполнителя.

2.3.2 Прецедент К1. Вход в систему

Краткое описание прецедента

Пользователям системы необходимо пройти разовый процесс регистрации для последующих авторизаций. Данный прецедент присваивает пользователю идентификатор и определяет его группу. *Действующие лица этого прецедента* – Пользователь.

Описание основного потока

Начинается прецедент с того что пользователь выбирает желаемое действие – войти в систему, нажав кнопку «войти», находящуюся в правом верхнем углу в любом разделе сайта. Либо зарегистрироваться как новый пользователь, нажав на кнопку «регистрация», находящуюся справа от кнопки «войти».

Базовый поток – Вход в систему.

1. Пользователь нажимает «регистрация», так как не имеет учетной записи.
2. Система спрашивает, как его зарегистрировать.
3. Пользователь выбирает группу. При любом выборе система выдаст одинаковые страницы регистрации, но в дальнейшем карточки пользователей различных групп будут отличаться.
4. Пользователь решил зарегистрироваться как заказчик.

5. Система вывела форму регистрации на экран.
6. Пользователь последовательно вводит:
 - a. имя;
 - b. фамилию;
 - c. e-mail;
 - d. логин;
 - e. пароль;
 - f. подтверждение пароля;
 - g. пройти проверку (капча).
7. Пользователь отправляет данные на сервер.
8. Система определила, что ответ на картинку и оба пароля верны.
9. Система создает нового пользователя.
10. Система выводит на экран форму авторизации.
11. Пользователь вводит последовательно:
 - a. логин (или e-mail);
 - b. пароль.
12. Пользователь нажимает кнопку «отправить».
13. Система определяет, что введенная пара логин/пароль верна.
14. Система авторизует пользователя.

Описание альтернативных потоков

Пользователь при регистрации ввел неверно подтверждение пароля либо ответ на картинку (пункты: 1-7 совпадают с базовым потоком).

8. Система определила, что ответ на картинку или один из паролей не верны.
9. Система отправляет пользователю ранее заполненную форму, за исключением пунктов б.f и б.g из базового потока.
10. Пользователь последовательно вводит:
 - a. подтверждение пароля;
 - b. пройти проверку (капча).
11. Пользователь отправляет данные на сервер.

Цикл продолжается до тех пор, пока пользователь сам не прекратит попытки либо до положительного ответа системы (базовый поток).

Следующий альтернативный поток.

Пользователь делает выбор авторизации в системе.

1. Пользователь нажимает «войти», так как ранее уже зарегистрировался.
2. Система выводит форма авторизации на экран.
3. Пользователь вводит последовательно:
 - a. логин (или e-mail);
 - b. пароль.
4. Пользователь нажимает кнопку «отправить».
5. Система определяет, что введенная пара логин/пароль верна.

6. Система авторизует пользователя.

Следующий альтернативный поток.

Пользователь при вводе пары логин/пароль допускает ошибки менее трех попыток.

1. Пользователь нажимает «войти», так как ранее уже зарегистрировался.
2. Система выводит форма авторизации на экран.
3. Пользователь вводит последовательно:
 - a. логин (или e-mail);
 - b. пароль.
4. Пользователь нажимает кнопку «отправить».
5. Система определяет, что введённая пара логин/пароль неверна.
6. Система вычисляет, что количество попыток меньше либо равно трём.
7. Пользователь повторно вводит логин/пароль.

Цикл продолжается пока пользователь не введет верно пароль или не превысит количество допустимых попыток.

Следующий альтернативный поток.

1. Пользователь нажимает «войти», так как ранее уже зарегистрировался.
2. Система выводит форма авторизации на экран.
3. Пользователь вводит последовательно:
 - a. логин (или e-mail);
 - b. пароль.
4. Пользователь нажимает кнопку «отправить».
5. Система определяет, что введённая пара логин/пароль неверна.
6. Система вычисляет, что количество попыток больше трёх.
7. Система блокирует попытки входа с этого IP адреса на некоторое время.
8. Система отправляет сообщение на почту пользователю.

Специальные требования

Количество попыток для входа не должно превышать трех в сутки, с одного IP адреса. Пароль должен содержать буквы и цифры латинского алфавита длиной не менее 6 символов.

Постусловия

При успешном выполнении прецедента пользователь будет допущен к пользованию системой с расширенными возможностями в рамках политики его группы пользователей. При неуспешном – пользователь сможет пользоваться системой как гость, не имея возможности добавлять заявки (только просмотр списки проектов и исполнителей).

Диаграмма деятельности для прецедента К1 «Вход в систему» представлен на рисунке 17.

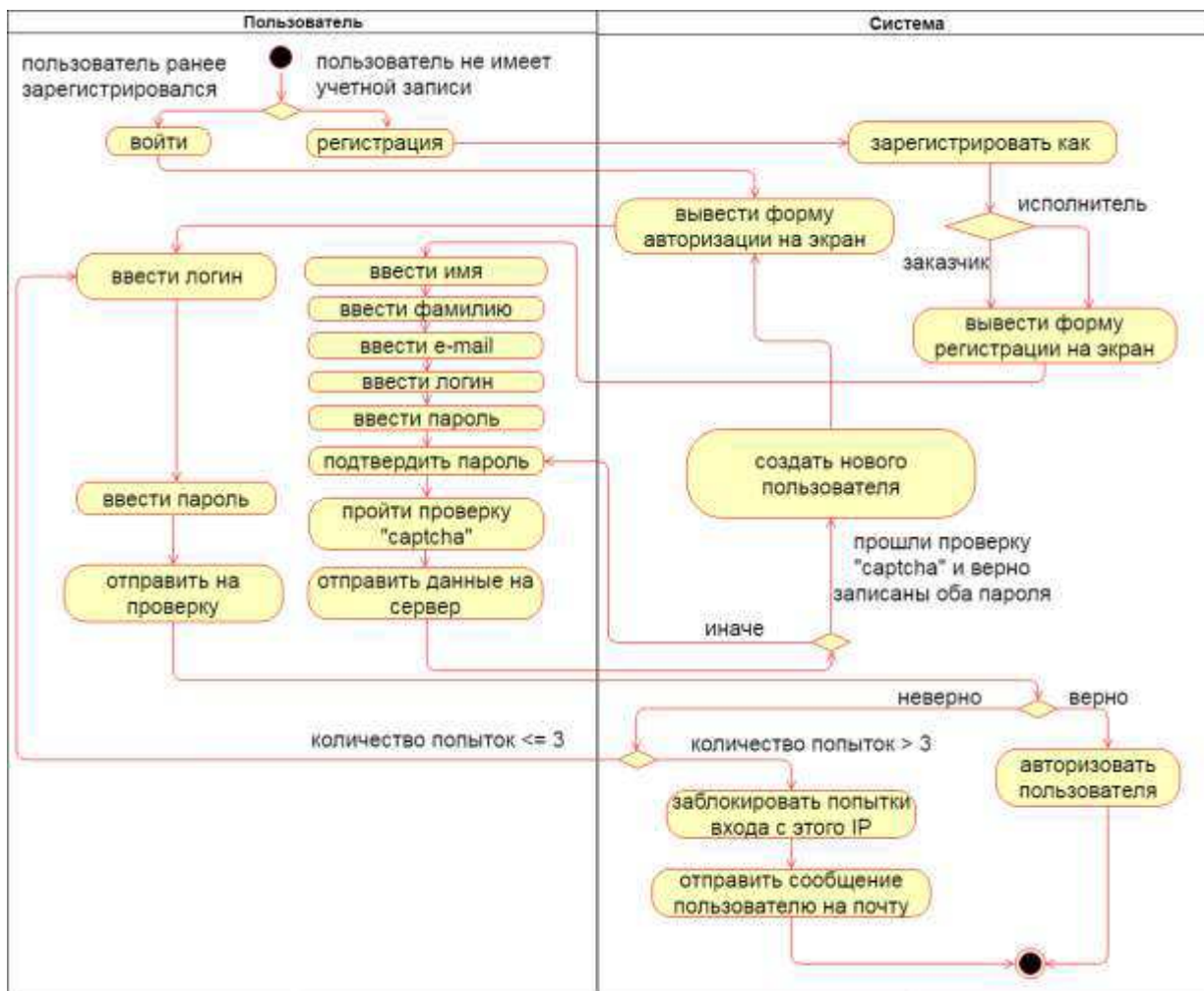


Рисунок 17 – Диаграмма деятельности прецедента К1 «Вход в систему»

2.3.3 Прецедент К2. Добавление новой заявки

Краткое описание

Заказчик создает новую заявку, описывая коротко суть работы и указывая контактную информацию. Действующие лица этого прецедента – Пользователь-заказчик.

Поток событий

Прецедент начинается, когда Пользователь-заказчик выбирает деятельность «опубликовать проект» из любого места на веб-сайте.

Базовый поток – Добавление новой заявки

1. Заказчик кликает в шапке сайта на раздел «Заказчики» → «Опубликовать проект».
2. Система определяет, что к ней обращается авторизованный пользователь.
3. Перенаправляет пользователя на страницу с формой заявки.
4. Система нашла в карточке товара контактную информацию.
5. Заказчик вводит последовательную необходимую информацию:
 - a. Описание заявки;
 - b. Сроки;
 - c. Ключевые слова;
 - d. Адрес.
6. Пользователь отправляет заявку на обработку.
7. Система добавляет полученную информацию в базу данных.
8. После чего система размещает заявку на рабочем пространстве сайта – на доске объявлений.

Альтернативные потоки

Добавление новой заявки невозможно, не будучи авторизованным. Если Пользователь-заказчик не авторизовался, то система отправит его на страницу регистрации/авторизации (предыдущий прецедент).

1. Заказчик кликая в шапке сайта на «Заказчики» → «Опубликовать проект»
2. Система определяет, что к ней обращается не авторизованный пользователь и перенаправляет его на страницу регистрации/авторизации.

Следующий альтернативный поток.

Если система не обнаружит контактной информации в карточке пользователя, то система оставит поля с контактами пустыми, предоставив пользователю в ручную ввести эти данные.

1. Заказчик кликая в шапке сайта на «Заказчики» → «Опубликовать проект»
2. Система определяет, что к ней обращается авторизованный пользователь.
3. Перенаправляет пользователя на страницу с формой заявки.
4. Система нашла в карточке товара контактную информацию.
5. Заказчик вводит последовательную необходимую информацию:
 - a. Описание заявки;
 - b. Сроки;
 - c. Ключевые слова;
 - d. Адрес;
 - e. Ввести контактную информацию.
6. Пользователь отправляет заявку на обработку.
7. Система добавляет полученную информацию в базу данных.

8. После чего система размещает заявку на рабочем пространстве сайте – на доске объявлений.

Специальные требования

Время планирования одного заказа не должно превышать 15 минут.

Предусловия

Авторизация

Перед тем как начинается этот прецедент, пользователь должен быть авторизован в системе как Пользователь-заказчик.

Постусловия

При успешном окончании прецедента Пользователь-заказчик увидит свой опубликованный проект (заявку) в соответствующем разделе сайта. При неуспешном пользователь перенаправляется на страницу авторизации.

Точки расширения

Если при выполнении п. 2 выясняется, что пользователь не авторизован, Система переходит к выполнению расширяющего прецедента «Вход в систему».

Диаграмма деятельности для прецедента К2 «Добавление заявки», представлен на рисунке 18.



Рисунок 18 – Диаграмма деятельности прецедента К2 «Добавление заявки»

2.3.4 Прецедент КЗ. Выбор исполнителя

Краткое описание

Заказчик выбирает исполнителя для своего проекта, либо Исполнитель выбирает для себя проект. *Действующие лица этого прецедента* – Пользователь-заказчик и Пользователь-исполнитель.

Поток событий

Прецедент начинается с момента нажатия одним из пользователей кнопки для поиска исполнителя, или проекта. В базовом случае пользователь-заказчик ищет исполнителя.

Базовый поток – Выбор исполнителя

1. Заказчик нажимает на кнопку «Исполнители» в шапке сайта.
2. Система выводит список исполнителей.
3. Заказчик выбирает исполнителя.
4. Заказчик выбирает из своих проектов желаемый.
5. Заказчик отправляет приглашение исполнителю.
6. Система отправляет уведомление исполнителю.
7. Исполнитель подтверждает приглашение:
 - a. делает ставку;
 - b. указывает сроки;
 - c. пишет комментарий;
 - d. отправляет свое предложение.
8. Система добавляет исполнителя к проекту.
9. Заказчик открывает список исполнителей проекта.
10. Заказчик выбирает понравившегося исполнителя.
11. Заказчик нажимает «Принять заявку».
12. Система отправляет запрос на подтверждение назначения исполнителем.
13. Заказчик подтверждает запрос.

Альтернативные потоки

Заказчик на этапе подтверждения передумал назначать данного исполнителя и хочет назначить другого. При запросе на подтверждение назначения заказчик отказывается от исполнителя.

(пункты 1-12 совпадают с базовым потоком)

13. Заказчик не подтверждает запрос.
14. Заказчик переходит к выбору нового исполнителя из списка исполнителей проекта.
15. Заказчик нажимает «Принять заявку»
16. Система отправляет запрос на подтверждение назначения исполнителем.

Цикл повторяется до тех пор, пока заказчик не подтвердит запрос назначения исполнителя проекта.

Следующий альтернативный поток.

Исполнитель отказывается от приглашения заказчиком.

1. Заказчик нажимает на кнопку «Исполнители» в шапке сайта.
2. Система выводит список исполнителей.
3. Заказчик выбирает исполнителя.
4. Заказчик выбирает из своих проектов желаемый.
5. Заказчик отправляет приглашение исполнителю.
6. Система отправляет уведомление исполнителю.
7. Исполнитель отказывается.

Следующий альтернативный поток.

Исполнитель ищет проект для выполнения.

1. Исполнитель нажимает кнопку «Проект», которая расположена в шапке сайта.
2. Система выводит список проектов.
3. Исполнитель выбирает проект.
 - a. делает ставку;
 - b. указывает сроки;
 - c. пишет комментарий;
 - d. отправляет свое предложение.
4. Система добавляет исполнителя к проекту.
5. Заказчик открывает список исполнителей проекта.
6. Заказчик выбирает понравившегося исполнителя.
7. Заказчик нажимает «Принять заявку».
8. Система отправляет запрос на подтверждение назначения исполнителем.
9. Заказчик подтверждает запрос.

Предусловия

Авторизация

Перед тем как начинается этот прецедент действующие лица в момент своих действий должны быть авторизованы в системе.

Постусловия

При успешном окончании прецедента исполнитель назначается на проект заказчика. При неуспешном - проект остается без исполнителя.

Диаграмма деятельности для прецедента КЗ «Выбор исполнителя», представлен на рисунке 19.

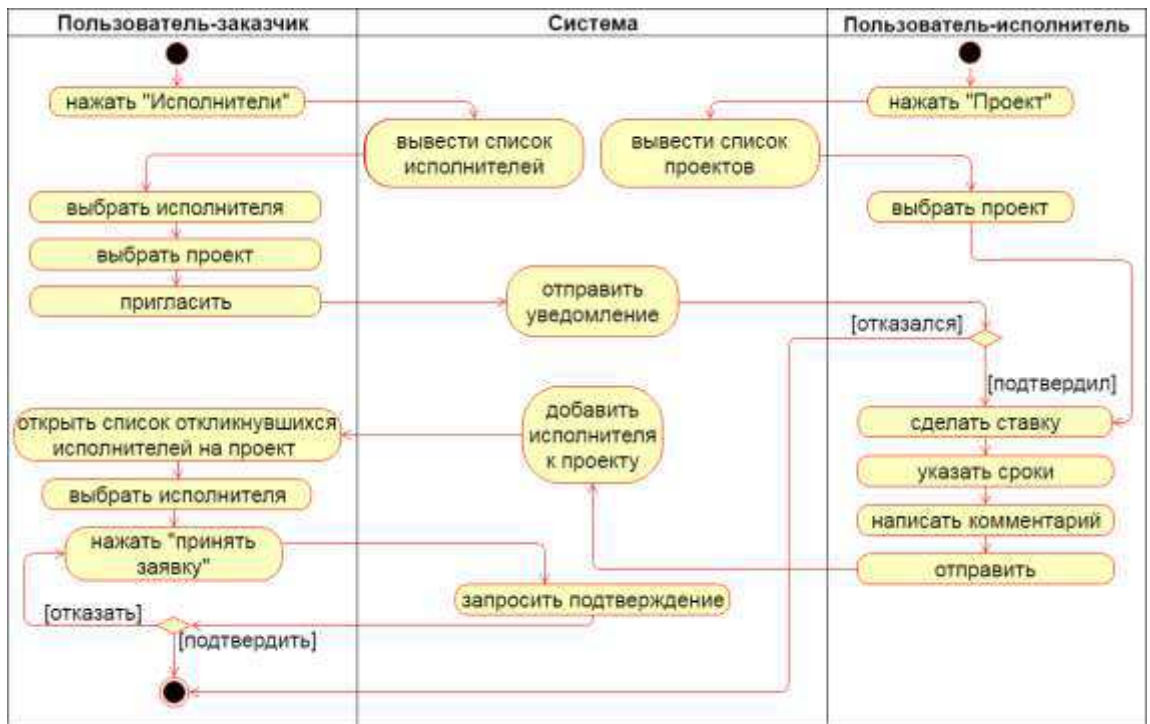


Рисунок 19 – Диаграмма деятельности прецедента КЗ «Выбор исполнителя»

2.4 Диаграмма компонентов и диаграмма развертывания

Диаграмма компонентов – диаграмма, которая отображает компоненты системы и связи между ними.

Для построения диаграммы компонентов разобьем функционал системы по компонентам:

- DBProvider – компонент, реализующий взаимодействие с БД системы.
- CRequest – компонент, реализующий работу с заявками.
- CControlSite – компонент управления сайтом.
- CUser – компонент для программного представления пользователя, обеспечивает разделение ролей.
- GUI – компонент, реализующий графический интерфейс.

Диаграмма развертывания – диаграмма, которая отображает способ взаимодействия компонентов с аппаратными средствами в физической системе, а также соединение аппаратных средств между собой. Основным элементом системы является узел. Узел – представление любого вычислительного ресурса.

Для создания диаграммы развертывания, необходимо определить, какие аппаратные средства должна использовать создаваемая система. Т.к.

наша система работает с базой данных, к которой осуществляется многопользовательский доступ, необходим сервер для хранения базы данных и выполнения запросов пользователей. В нашей системе реализовано три АРМа (пользователь-заказчик, пользователь-исполнитель, пользователь-администратор), соответственно, для полного функционирования системы необходимо, чтобы у каждого были аппаратные средства для обращения к системе. На рисунке 20 представлены диаграммы компонентов и развертывания.

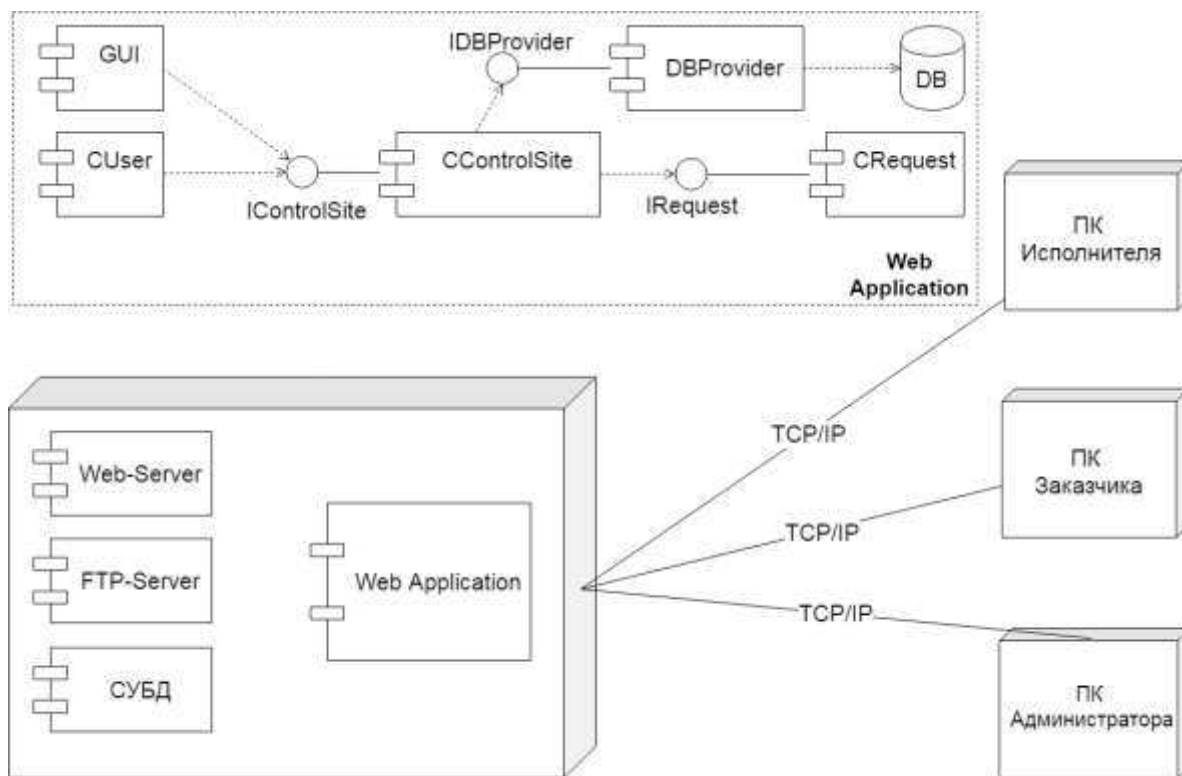


Рисунок 20 – Диаграмма компонентов и развертывания

2.5 Вывод по второй главе

Во второй главе проведен анализ требований к системе. С учетом выявленных недостатков системы из предыдущей главы построена новая модель бизнес-процесса «Вызвать мастера» и составлен проект технического задания (Приложение А).

Выполнено проектирование Интернет платформы посредством UML-диаграмм – построены диаграмма вариантов использования, диаграммы деятельности ключевых прецедентов, диаграммы компонентов и развертывания, а так же построена модель взаимодействия компонентов окружения сервиса.

3 Разработка программного прототипа

3.1 Диаграмма классов

Для построения диаграммы классов определим классы, которые будут использованы в нашей системе:

1. CRight – определяет право пользователя на работу с типа документов.
2. CUser – пользователь с его набором прав и специфичных атрибутов.
3. CRequest – конкретная заявка со своим набором параметров.
4. CCollectRequest – совокупность заявок. Класс хранения всех заявок, ранжирование по категориям.
5. CCategoryRequest – определяет категорию заявки.
6. CKeyWords – ключевые слова относящиеся к определенной категории.
7. CAttainments – навыки необходимые для выполнения заявки.
8. CCity – класс определяющий набор городов.

На рисунке 21 представлена диаграмма классов данной системы.

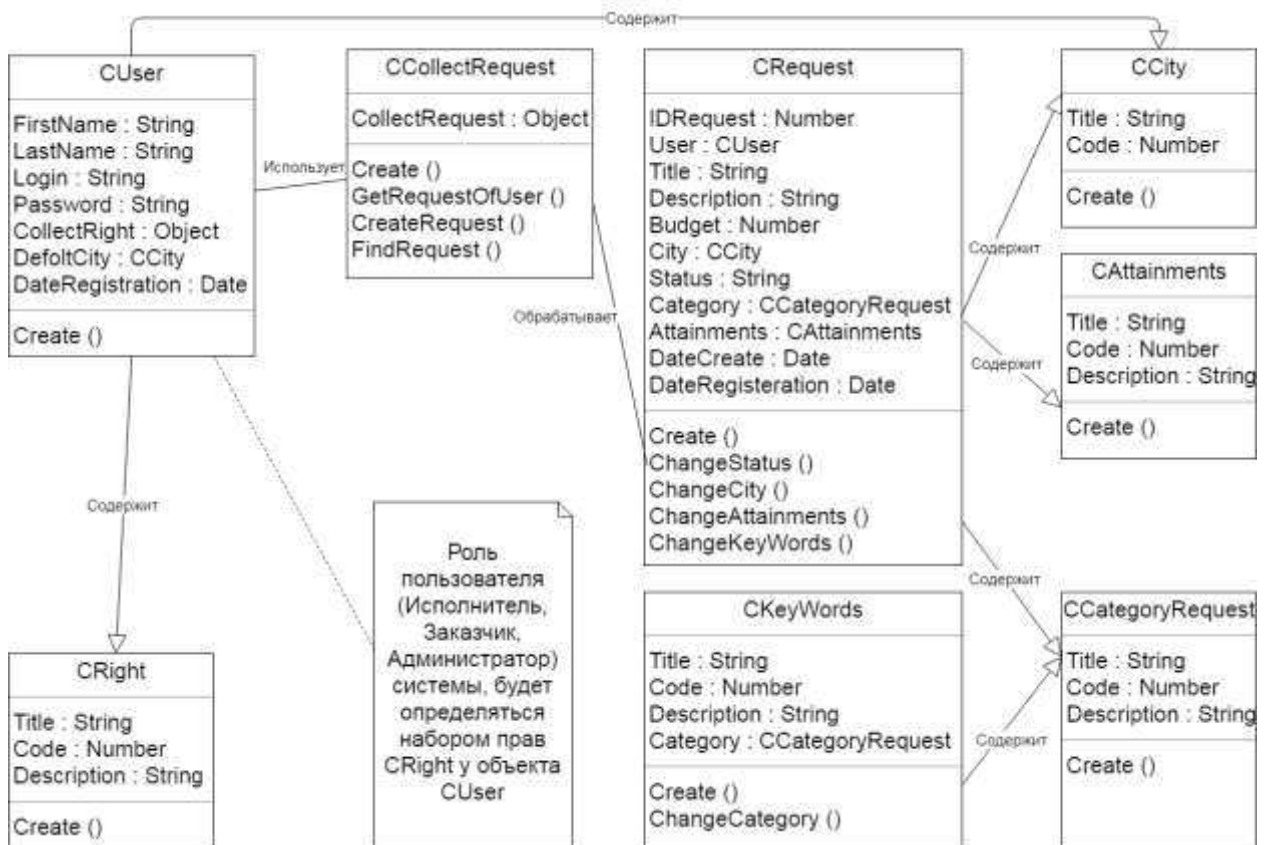


Рисунок 21 – Диаграмма классов

Рассмотрим более детально каждый из классов – опишем атрибуты и определим кванторы видимости (указаны в круглых скобках перед соответствующими атрибутами и операциями).

CRight

Графическое изображение класса **CRight** представлено на рисунке 22.

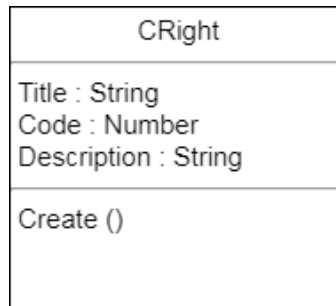


Рисунок 22 – Атрибуты и операции класса **CRight**

- (Public) Title – название права;
- (Public) Code – код права;
- (Public) Description – описание права;
- (Public) Create () – создание нового права.

CRequest

Графическое изображение класса **CRequest** представлено на рисунке 23.

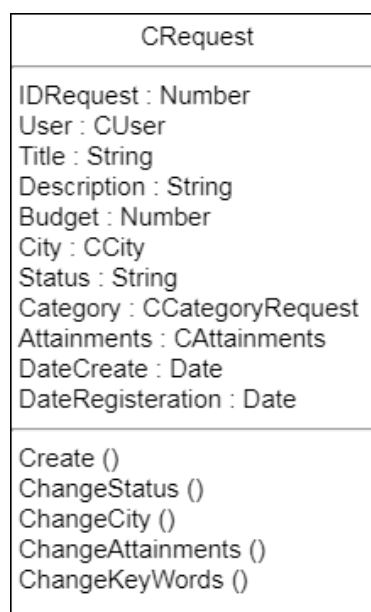


Рисунок 23 – Атрибуты и операции класса **CRequest**

- (Private) IDRequest – идентификационный номер заявки;
- (Private) User – владелец/создатель заявки;
- (Public) Title – название заявки;
- (Public) Description – описание заявки;
- (Private) Budget – бюджет, установленный на выполнение заявки;
- (Public) City – город, в котором должна быть выполнена заявка;
- (Private) Status – статус заявки;
- (Private) Category – категория заявки;
- (Private) Attainments – навыки, необходимые для выполнения заявки;
- (Private) DateCreate – дата начала работы;
- (Private) DateRegistration – дата создания заявки
- (Public) Create () – создание новой заявки;
- (Public) ChangeStatus () – изменение статуса заявки;
- (Public) ChangeCity () – изменение города;
- (Public) ChangeAttainments () – изменение требуемых навыков;
- (Public) ChangeKeyWords () – изменение ключевых слов.

CKeyWords

Графическое изображение класса CKeyWords представлено на рисунке 24.

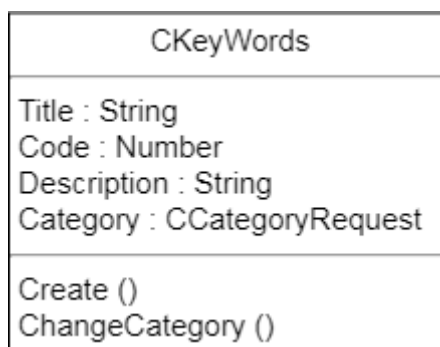


Рисунок 24 – Атрибуты и операции класса CKeyWords

- (Public) Title – ключевое слово;
- (Public) Code – код слова;
- (Public) Description – примечание;
- (Public) Category – категория, к которой относится ключевое слово;
- (Private) Create () – создание нового ключевого слова;

- (Private) ChangeCategory () – изменение категории ключевого слова.

CCollectRequest

Графическое изображение класса CCollectRequest представлено на рисунке 25.

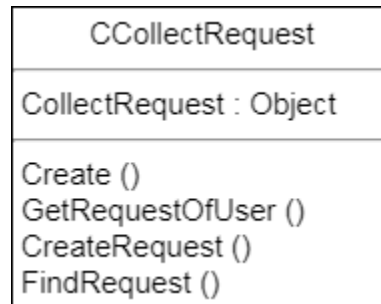


Рисунок 25 – Атрибуты и операции класса CCollectRequest

- (Private) CollectRequest – коллекция (набор) заявок;
- (Public) Create () – создание нового права;
- (Public) GetRequestOfUser () – получение заявок конкретного пользователя;
- (Public) CreateRequest () – создание новой заявки;
- (Public) FindRequest () – поиск заявки;

CCity

Графическое изображение класса CCity представлено на рисунке 26.

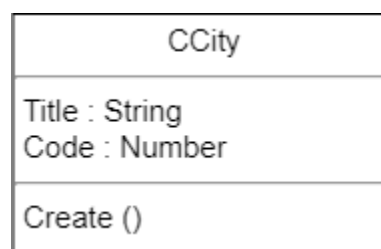


Рисунок 26 – Атрибуты и операции класса CCity

- (Public) Title – название населенного пункта;
- (Public) Code – код населенного пункта;
- (Private) Create () – создание нового населенного пункта.

CCategoryRequest

Графическое изображение класса CCategoryRequest представлено на рисунке 27.

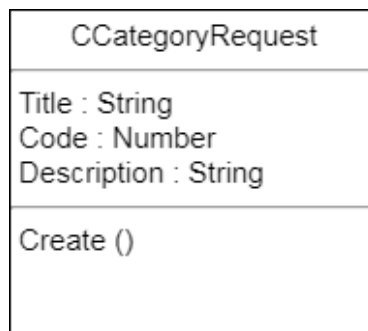


Рисунок 27 – Атрибуты и операции класса CCategoryRequest

- (Public) Title – название категории;
- (Public) Code – код категории;
- (Public) Description – описание категории;
- (Private) Create () – создание новой категории.

CAttainments

Графическое изображение класса CAttainments представлено на рисунке 28.

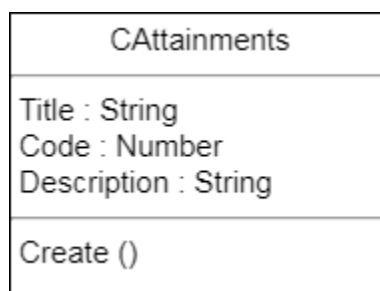


Рисунок 28 – Атрибуты и операции класса CAttainments

- (Public) Title – название навыка;
- (Public) Code – код навыка;
- (Public) Description – описание навыка;
- (Private) Create () – создание нового навыка.

CUser

Графическое изображение класса CUser представлено на рисунке 29.

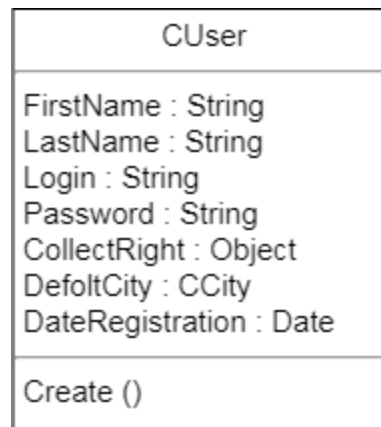


Рисунок 29 – Атрибуты и операции класса CUser

- (Public) FirstName – имя ;
- (Public) LastName – фамилия;
- (Public) Login – логин;
- (Public) Password – пароль;
- (Public) CollectRight – набор прав пользователя;
- (Public) DefoltCity – город по умолчанию;
- (Public) DateRegistration – дата регистрации пользователя;
- (Public) Create () – создание нового пользователя;

3.2 Работа с сервисом

Согласно проекту технического задания (Приложение А), для разработки Интернет платформы применяются такие инструменты, как HTML, CSS, JavaScript, PHP.

HTML — язык гипертекстовой разметки, с помощью которого создаются веб-страницы. Данный язык интерпретируется браузером. Он выступает как средство логической разметки страницы. В HTML используются такие инструменты, которые позволяют создать структуру документа и хранение информации о нем.

CSS — язык, который отвечает за визуальное представление документов.

Java Script — динамический язык программирования, применяемый к HTML-документу. Обеспечивает интерактивность на веб-страницах.

PHP — скриптовый язык общего назначения, применяемый для разработки веб-сайтов. Для защиты от спама применяется reCAPTCHA, которая основана на тесте Тьюринга. Позволяет оградить от автоматических алгоритмов и программ.

Далее на рисунках 30-37 представлена разработанная Интернет платформа для оказания услуг по ремонту объектов капитального строительства.

3.2.1 Главная страница сайта

На рисунке 30 представлена главная страница сайта доступная по адресу: <http://willdon.ru>.

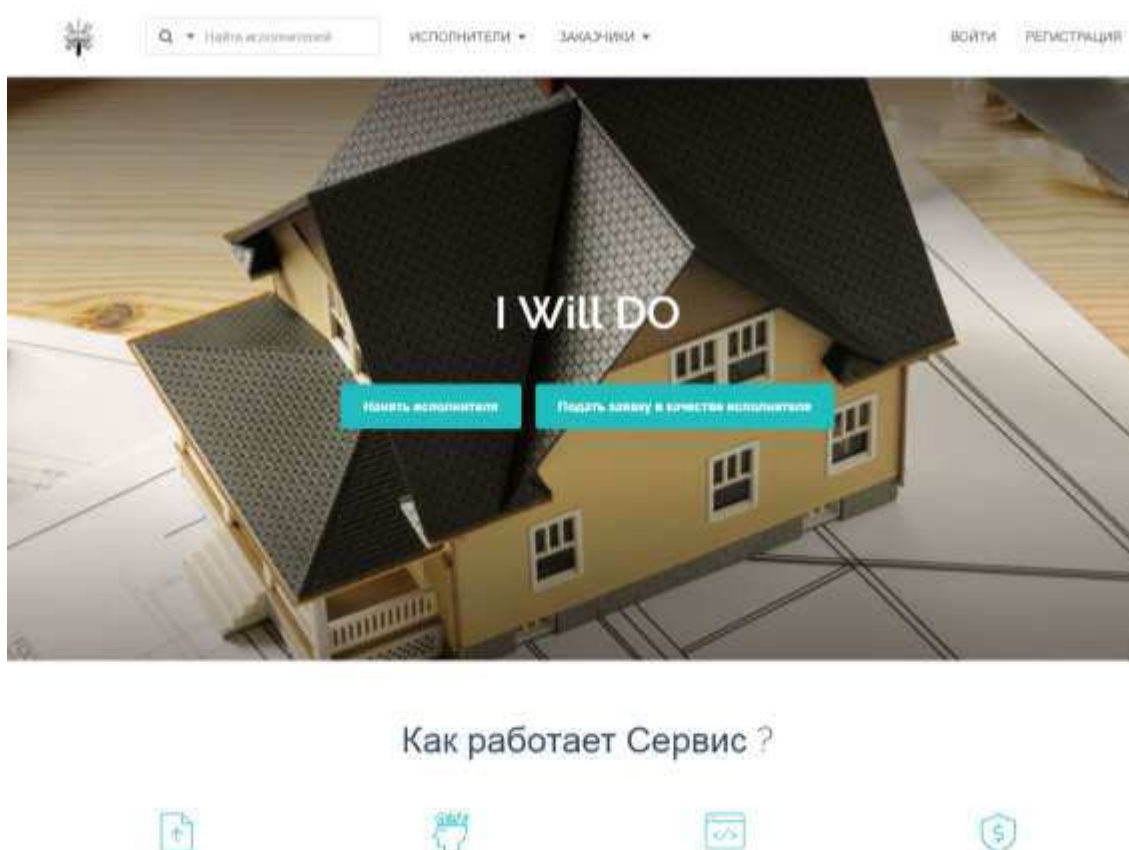


Рисунок 30 – Главная страница сайта

В шапке сайта справа от логотипа поле для поиска исполнителей и проектов. Еще правее функциональные возможности, разделенные по группам пользователей: Исполнители -> найти проект, создать профиль; Заказчики -> опубликовать проект, найти исполнителей. В правом верхнем углу расположены кнопки входа в систему «Регистрация» и «Войти». В центре баннера под названием сайта, имеются две функциональные кнопки «Нанять исполнителя», «Подать заявку в качестве исполнителя» (необходимо быть авторизованным).

3.2.2 Регистрация и вход в систему

Первый шаг регистрации – нажатие на кнопку «Регистрация». Система перенаправляет на страницу регистрации, где необходимо выбрать – зарегистрироваться как заказчик или как исполнитель. Страница регистрации представлена на рисунке 31.

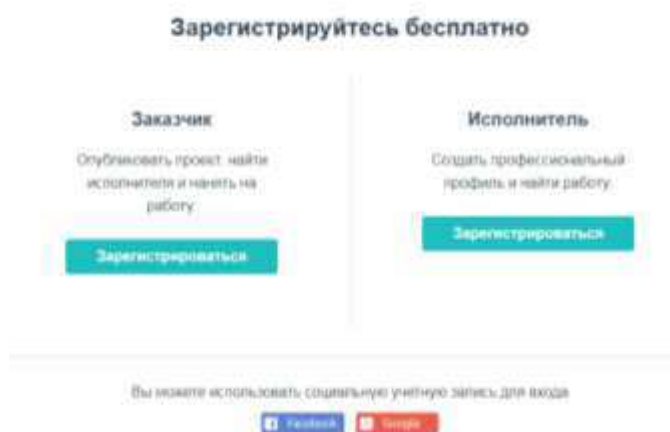


Рисунок 31 – Страница регистрации, выбор группы

На втором шаге пользователь вводит необходимые данные (все поля являются обязательными к заполнению): имя, фамилия, e-mail, логин, пароль и подтверждение пароля. А так же проходит проверку капчи (в приложении В представлен код капчи). По завершению ввода данных, нажатие кнопки «Зарегистрировать» завершает процесс регистрации, и система выводит уведомление об успешной или неуспешной регистрации. На рисунке 32 представлена форма регистрации пользователя (группа пользователя на поля и внешний вид пользователя не влияет).

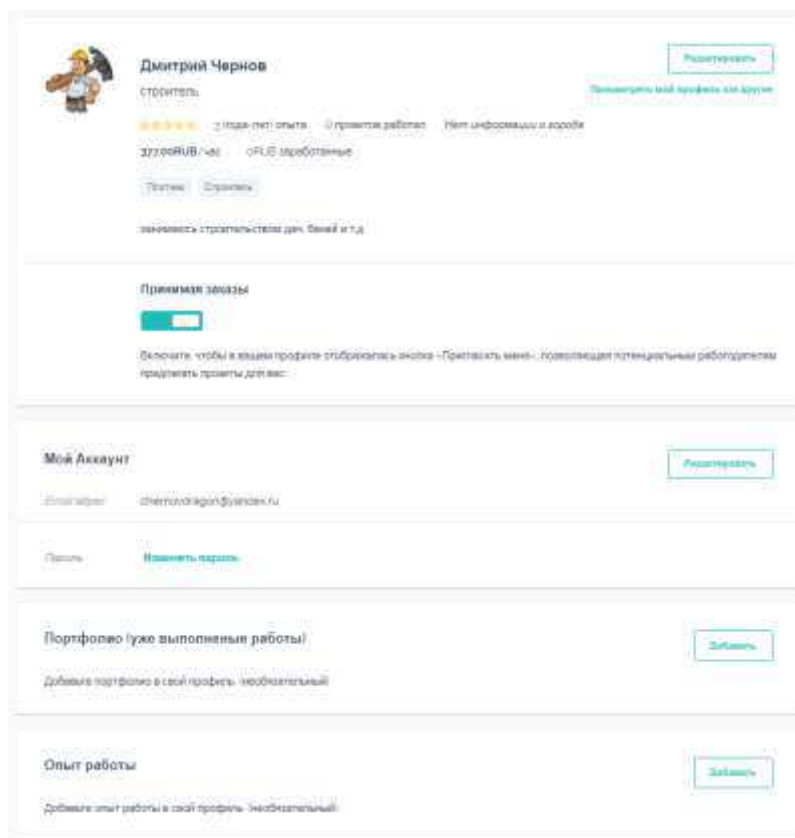


Рисунок 34 – Личный кабинет исполнителя

3.2.3 Добавление новой заявки (проекта)

Будучи авторизованным пользователем, создать новую заявку можно путем нажатия на кнопку в шапке сайта «Проект» -> «Опубликовать проект». На рисунке 35 представлено выпадающее окно с указанными кнопками.

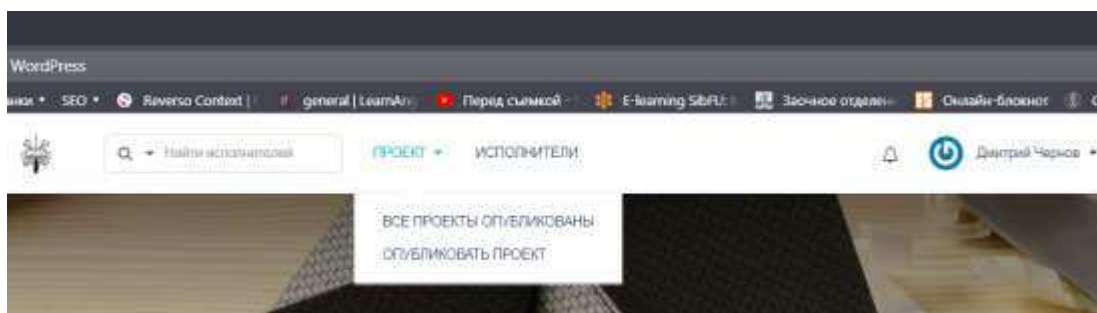


Рисунок 35 – Первый шаг к созданию заявки

Далее система открывает форму с полями для получения сведений о заявке. На рисунке 36 представлена данная форма.

Детали вашего проекта

В каких категориях работает ваш проект?

Ремонт квартиры ✕

Название вашего проекта

Ремонт детской комнаты

Опишите, что вам нужно сделать

B I U ☰ ☷ 🔗 ↺

Необходимо в детской комнате покрасить стены. Выложить ламинат. Натянуть потолки.

P

Вложения (опционально)

Загрузить файлы

(Загрузить максимум 5 файлов с расширениями, включая png, jpg, pdf, xls и doc)

Какие навыки вам нужны?

Штукатур ✕ Маляр ✕

Ваш бюджет проекта

RUB 25000

Расположение (необязательно)

Красноярск ▼

Этапы (Вы можете создать этап 5)

Добавить этап

Добавить проект

Рисунок 36 – Форма нового проекта

После нажатия кнопки «Добавить проект». Система запишет данные и выведет информацию по заявке на экран, как показано на рисунке 37.

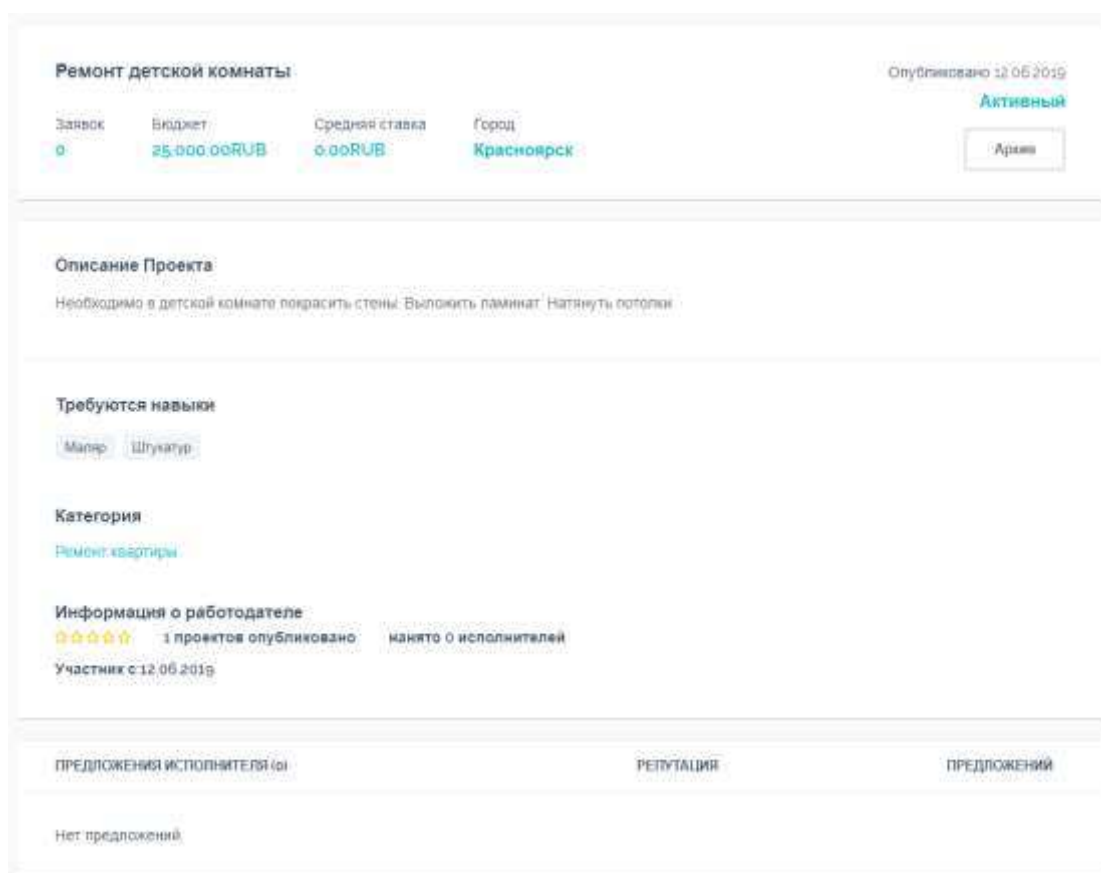


Рисунок 37 – Информация о проекте

3.3 Вывод по третьей главе

В последней главе разработана функционирующая Интернет платформа, которая дает пользователю возможность создать заявку на ремонт и/или постройку объектов капитального строительства. А так же зарегистрироваться как исполнитель и выбирать проекты для выполнения. Для разработки использовались HTML, CSS, JavaScript и PHP.

Разработанный проект соответствует техническому заданию (Приложение А)

ЗАКЛЮЧЕНИЕ

В ходе выпускной квалификационной работы были проанализированы аналоги разрабатываемой Интернет платформы. Создана модель «AS-IS» бизнес-процесса «Вызвать мастера», являющаяся общей для всех аналогов. С учетом выявленных недостатков сформирована модель «TO-BE» того же процесса. Учитывая пожелания заказчика, был составлен проект технического задания (Приложение А).

Выполнено проектирование Интернет платформы посредством UML-диаграмм – построены диаграмма вариантов использования, диаграммы деятельности ключевых прецедентов, диаграммы компонентов и развертывания, а так же построена модель взаимодействия компонентов окружения сервиса.

На основе проектирования была разработана функционирующая Интернет платформа с использованием HTML, CSS, PHP, JavaScript. С помощью Интернет платформы пользователи имеют возможность создавать заявки на ремонт или постройку объектов капитального строительства. А так же зарегистрироваться в системе как исполнитель и выбирать себе существующие заказы.

Разработанная Интернет платформа размещена на хостинге провайдера Beget. Явки и пароли от аккаунта переданы представителю ООО «Рассвет+», о чем свидетельствует составленный акт об использовании сервиса в структуре работы организации (Приложение Б).

СПИСОК СОКРАЩЕНИЙ

CSS — Cascading Style Sheets

FTP — File Transport Protocol

HTML — Hyper Text Markup Language

PHP — Hypertext Preprocessor

TCP — Transmission Control Protocol

IP — Internet Protocol

UML — Unified Modeling Language

СУБД — Система Управления Базой Данных

АС — Автоматизированная Система

АИС — Автоматизированная Информационная система

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Боггс, М. UML и Rational Rose / М. Боггс. – Москва : РГГУ, 2016. - 438 с.
2. Леоненков, А. В. Самоучитель UML / А. В. Леоненков .— 2-е изд., перераб. и доп.— Санкт-Петербург : БХВ-Петербург, 2004
3. Буч, Г. UML. Руководство пользователя / Г. Буч, Д. Рамбо, А. Джекобсон. – Москва : ДМК Пресс; Издание 2-е, стер., 2014. - 432 с.
4. Нильсен, П. Язык SQL: учебный курс MCAD/ MCSE? MCDBA: пер. с англ. / П. Нильсен. – 2-е изд., испр. – Москва : Издательство-торговый дом «Русская редакция», 2003. – 512с.
5. Бабич, А. В. UML. Первое знакомство. Пособие для подготовки к сдаче теста UMO-100 (OMG Certified UML Professional Fundamental) (+ CD-ROM) / А.В. Бабич. – Москва : Бином. Лаборатория знаний, 2008. - 176 с.
6. Марка, Д. Методология структурного анализа и проектирования SADT / Д. Марка, К. МакГоуэн. – Москва : МетаТехнология, 1993. – 240 с.
7. Суэринг, С. PHP и MySQL. Библия программиста, 2-е издание = PHP 6 and MySQL 6 Bible / С. Суэринг, Т. Конверс, П. Джойс – Москва : «Диалектика», 2010. – 912с.
8. Ульман, Л. MySQL. / Л. Ульман. Пер. С англ. А.А. Слинкина. – Санкт-Петербург : Питер, 2004. – 352 с.
9. Леоненков, Александр Самоучитель UML 2 / Александр Леоненков. – Москва : БХВ-Петербург, 2007. - 431 с.
10. Ивлев, В. А. ABIS. Информационные системы на основе действий / В.А. Ивлев, Т.В. Попова. – Москва : 1С-Паблишинг, 2005. - 248 с.
11. Приемы объектно-ориентированного проектирования: Паттерны проектирования / Э. Гамма и др. – Москва : Addison Wesley Longman, Inc., 2003. - 368 с.
12. Флаулер, М. UML. Основы. Краткое руководство по унифицированию языку моделей / М. Фаулер, К. Скотт. – Санкт-Петербург : Символ Плюс, 2004. – 192 с.
13. Кузнецов PHP. Практика создания Web-сайтов / Кузнецов, М.В. и – Москва : БВХ-Петербург, 2016. – 894 с.
14. Фаронов, В. А. Программирование на языке высокого уровня : учебное пособие / В. А. Фаронов. – Издательство: Питер, 2006. – 604 с.
15. Прохоренок, Н. А. HTML, javascript, php и MySQL. Джентельменский набор веб-мастера : учебник / Н. А. Прохоренок, В. А. Дронов. – Москва : Профиздат, 2014. – 207 с.
16. СТАНДАРТ ОРГАНИЗАЦИИ Система менеджмента качества Общие требования к построению, изложению и оформлению, документов учебной деятельности СТО 4.2–07–2014 – Красноярск : ИПК СФУ, 2014. – 60с.

ПРИЛОЖЕНИЕ А

Проект технического задания

УТВЕРЖДЕН

ООО «Рассвет+»

ИНН 2462064228, КПП 246201001

ОГРН 1182468066065

Юр. адрес: 660031 г. Красноярск,

ул. Глинки, д. 46, стр. 11

Факт. адрес: 660031 г. Красноярск,

ул. Глинки, д. 46 стр. 11

Т. 8 (923) 292-33-37

Генеральный директор

ООО «Рассвет+»

_____ А.Н. Лунев

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

(проект)

На 13 листах

**На разработку автоматизированной информационной
системы «Ремонтник»**

АИС «Рем»

Продолжение приложения А

Проект технического задания

А.1 ОБЩИЕ СВЕДЕНИЯ

А.1.1 Полное наименование системы и её условное обозначение

Полное наименование: Автоматизированная система обработки пользовательских запросов сферы ремонта. «АС Ремонтник».

Условное обозначение: АС Ремонтник

А.1.2 Наименование предприятий (объединений) разработчика и заказчика (пользователя) системы и их реквизиты

Заказчиком системы является ООО «Рассвет+»:

Юр. адрес: 660031 г. Красноярск, ул. Глинки, д. 46, стр. 11;

Факт. адрес: 660031 г. Красноярск, ул. Глинки, д. 46 стр. 11.

Краткое описание: Данная система разрабатывается для обеспечения пользователей удобным инструментом для поиска специалистов (фирм), предоставляющих услуги по ремонту.

А.1.3 Перечень документов, на основании которых создается система

Основанием для разработки АС Ремонтник является:

- договор на создание ПО № 10-01/2019 от 10.01.2019. Договор утвержден генеральным директором ООО «Рассвет+» Луневым Алексеем Николаевичем, именуемым в дальнейшем Заказчиком.
- Клиентская база данных в виде файла klientBD.xls;
- Список организаций партнеров для первоначального наполнения, файл spisok_organ.txt.

А.1.4 Плановые сроки начала и окончания работы по созданию системы

Плановый срок начала работ по созданию автоматизированной системы обработки пользовательских запросов сферы ремонта. «АС Ремонтник» - 15 января 2019г.

Плановый срок окончания работ по созданию автоматизированной системы обработки пользовательских запросов сферы ремонта. «АС Ремонтник» - 17 июня 2019 г.

Продолжение приложения А

Проект технического задания

А.1.5 Порядок оформления и предъявления заказчику результатов работ по созданию системы

Система передается в виде передачи доступа к веб-сайту размещенного на хостинге оговоренном и прописанном в договоре № 10-01/2019 от 10.01.2019 г. между Исполнителем и Заказчиком. Вся отчетность о проделанной работе должна быть задокументирована и предоставлена в распечатанном и электронном виде Заказчику. В том числе инструкции по работе с сайтом в целом и отдельном компоненте. Приемка осуществляется ведущими лицами обеих сторон. Порядок представления системы:

- 1) презентация разработанной АС, её функционал и работоспособность;
- 2) презентация непосредственно сайта (клиентская часть) и его работоспособность;
- 3) презентация администраторской части;
- 4) описание и представление разработанных компонентов;
- 5) краткий обзор доступа к сервисам хостинга;
- б) учетные записи:
 - а. к файловому менеджеру хостинга (FTP);
 - б. к SSH клиенту;
 - с. к клиенту базы данных.

А.2 НАЗНАЧЕНИЕ И ЦЕЛИ СОЗДАНИЯ СИСТЕМЫ

А.2.1 Назначение системы

АИС «Ремонтник» предназначена для предоставления единого доступа пользователей Сети к базе, объединяющей в себе множество фирм и частных специалистов в области ремонта. В АИС предусмотрено исполнение следующих основных бизнес процессов:

- регистрация пользователя в системе;
- авторизация пользователя в системе;
- размещение объявления в соответствующем разделе;
- поиск и фильтрация исполнителей по указанным критериям;
- добавление исполнителя (специалиста/компании).

И вспомогательных бизнес-процессов:

- редактор меню;
- управление страницами;
- работа с пользователями;

Продолжение приложения А

Проект технического задания

- управление базой данных.

АИС «Ремонтник» предполагается использовать в ООО «Рассвет+».

А.2.2 Цели создания системы

Основными целями создания АИС «Ремонтник» являются:

- обеспечение пользователей сайта достоверной и актуальной информацией для поиска необходимого специалиста;
- унифицировать процессы предоставления и потребления услуг по ремонту в едином интерфейсе;
- извлечение прибыли из имеющейся информации о потребностях клиентов посредством Интернет платформы;

Для реализации поставленных целей система должна решать следующие задачи:

- размещение услуг посредством интерфейса;
- осуществление поиска и заказа услуг посредством интерфейса;
- формирование клиентской базы данных;
- регистрация новых пользователей посредством интерфейса.

Продолжение приложения А

Проект технического задания

А.3 ХАРАКТЕРИСТИКА ОБЪЕКТА АВТОМАТИЗАЦИИ

А.3.1 Краткие сведения об объекте автоматизации

Объектами автоматизации являются процессы по взаимодействию пользователей с системой. Данные процессы включают в себя:

- регистрация пользователя-потребителя в системе;
- авторизация пользователя в системе;
- добавление заявки на доску с объявлениями;
- поиск и фильтрация исполнителей по указанным критериям;
- регистрация пользователя-исполнителя (специалиста/компании).
- редактор меню;
- управление страницами;
- работа с пользователями;
- администрирование базы данных;

Данные процессы осуществляются следующими лицами:

- пользователь-потребитель – клиент, осуществляющий поиск специалиста;
- пользователь-исполнитель – клиент, предоставляющий услуги ремонта;
- администратор – пользователь системы, администрирующий базу данных.

Продолжение приложения А

Проект технического задания

А.4 ТРЕБОВАНИЯ К СИСТЕМЕ

А.4.1 Требования к структуре системы

Название: Компонент «Регистрация пользователя в системе».

Название: Компонент «Авторизация пользователя в системе».

Название: Компонент «Добавление заявки».

Название: Компонент «Поиск и фильтрация».

Название: Компонент «Редактирование меню».

Название: Компонент «Управление страницами».

Название: Компонент «Работа с пользователями».

Название: Компонент «Администрирование базы данных».

А.4.2 Требования к функциям и задачам

***Название:* Компонент «Регистрация пользователя в системе»**

Краткое описание: При первом входе в систему пользователь проходит процесс регистрации – присваивания идентификатора пользователя для дальнейшей его работы в системе. При регистрации пользователь указывает только необходимую информацию, которая система выводит в форме регистрации: Имя, Фамилию, E-mail, логин, пароль (дважды для подтверждения). Вся дополнительная информация вводится из личного кабинета. Для защиты от автоматической регистрации ботов предусмотрена система защиты с помощью простой капчи. После регистрации пользователь сможет оставлять заявки в рабочей области и вносить данные в БД. Не зарегистрированные пользователи имеют возможность только просмотра базы исполнителей.

Функциональные возможности: создание нового пользователя, присвоение ID, сбор персональных данных о пользователе.

***Название:* Компонент «Авторизация пользователя в системе»**

Краткое описание: Процесс, следующий за процессом регистрации, позволяющий пользователю, вновь и вновь заходить в систему под своим идентификатором и проводить ранее перечисленные действия.

Функциональные возможности: защита от спама, идентификация пользователя.

***Название:* Компонент «Добавление заявки»**

Продолжение приложения А

Проект технического задания

Краткое описание: Процесс, позволяющий пользователю-потребителю оставить заявку в соответствующем разделе сайта на исполнение тех или иных услуг. Пользователь-исполнитель может отправить запрос на исполнение, на который должен получить подтверждение от пользователя-потребителя.

Функциональные возможности: создание новой заявки, присвоение задачи к исполнителю, сбор необходимой информации для выполнения задачи.

Название: Компонент «Поиск и фильтрация»

Краткое описание: По желанию пользователь-заказчик может осуществить поиск исполнителей по всей доступной базе, а так же отфильтровать их по различным критериям (рейтинг, месторасположение, услуги и верификация). А пользователь-исполнитель в свою очередь может искать заказы.

Функциональные возможности: поиск исполнителей, поиск заказов.

Название: Компонент «Редактирование меню».

Краткое описание: Данный компонент предназначен для возможности изменения, добавления и удаления пунктов меню на сайте, а также для изменения общего вида главного меню. Управление производится пользователем-администратором из отдельного раздела сайта – панели управления.

Функциональные возможности: управление главным меню.

Название: Компонент «Управление страницами».

Краткое описание: Данный компонент позволяет вышеупомянутой группе пользователей (пользователь-администратор) модернизировать внешний вид страниц сайта, добавлять/удалять и перемещать блоки по рабочему пространству. Управление доступно так же из отдельного раздела сайта – панели управления.

Функциональные возможности: редактирование страниц сайта.

Название: Компонент «Работа с пользователями».

Краткое описание: Процесс управляющий всеми пользователями системы. Пользователь-администратор из панели управления имеет возможность добавить описание к пользователю, изменить его персональные

Продолжение приложения А

Проект технического задания

данные (логин, пароль, E-mail, телефон и т.д.), добавить нового пользователя и полностью удалить существующего.

Функциональные возможности: управление пользователями.

Название: Компонент «Администрирование базы данных».

Краткое описание: Данный компонент позволяет пользователю-администратору производить изменения над основным контентом сайта – страницей с заявками (включая создание и удаление заявок). Управление так же производится из панели управления сайтом.

Функциональные возможности: управление заявками.

А.4.3 Требования к видам обеспечения

А.4.3.1 Требования к математическому обеспечению.

В АС Ремонтник следует использовать алгоритм шифрования **MD5** — это 128 битный алгоритм хеширования, который был разработан в 1991 году проф. Рональдом Л. Ривестом. Алгоритм хеширования md5 предназначен для создания своего рода «отпечатков» или дайджестов произвольной длины.

А.4.3.2 Требования к лингвистическому обеспечению

В АС «Ремонтник» использовать русский язык интерфейса для взаимодействий с пользователем. В качестве высокоуровневого языка программирования выбран JavaScript и PHP, а в качестве языка разметки – HTML5 и CSS3.

А.4.3.3 Требования к программному обеспечению

Программное обеспечение для разработки:

Среда разработки для АС Ремонтник: IntelliJ IDEA – интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java, JavaScript, Python, разработанная компанией JetBrains.

Программное обеспечение для хостинга:

- Веб-сервер Apache 2;
- Сервер БД MySQL 5;
- Интерпретатор PHP версии не ниже 5;
- FTP-сервер для доступа к директории веб-приложения на хостинге.

Продолжение приложения А

Проект технического задания

А.4.3.3 Требования к аппаратному обеспечению

Аппаратное обеспечение системы должно максимально и наиболее эффективным образом использовать существующие технические средства.

В состав комплекса должны входить устройства пользователей:

- ПК;
- Телефон;
- Планшет.

Минимальные требования к техническим характеристикам аппаратного обеспечения:

- Процессор: 2 x Intel Xeon 3 ГГц;
- Объем оперативной памяти: 512 Мб;
- Сетевой адаптер: 1000 Мбит;
- Дисковая квота: не менее 2000 Мб;
- Почтовая квота: не менее 5000 Мб;
- Минимальное количество сайтов: 2 шт;
- Разрешенная нагрузка: 65 СР ;
- Нагрузка на MySQL: 2500 СР.

Минимальные требования к техническим характеристикам устройств пользователя.

- Процессор – не имеет значения;
- Объем оперативной памяти – не имеет значения;
- Дисковая подсистема – не имеет значения;
- Сетевой адаптер – 100 Мбит.

А.4.3.4 Нефункциональные требования

а. Требования к GUI (внешнему интерфейсу пользователя):

Требования к юзабилити:

1. Страницы сайта загружаются не дольше, чем 5 секунд.
2. Логотип хорошо заметен, размещен сверху, на него можно кликнуть и он является ссылкой на главную страницу.
3. URL-адрес сайта — осмысленный, его легко запомнить.
4. Поиск — в верхней части сайта, его легко найти.
5. Используется поиск по сайту от Google или Яндекс.
6. Самая важная информация размещена на верхнем экране, ее можно прочитать без вертикальной прокрутки.

Продолжение приложения А

Проект технического задания

7. При разрешении 1024*768 и больше — горизонтальная полоса прокрутки отсутствует. При меньших разрешениях — горизонтальная прокрутка допускается.
8. В текстах междустрочный интервал 1,5 размера шрифта.
9. Расстояние между абзацами больше, чем междустрочный интервал
10. В текстах все ссылки подчеркнуты, отображаются другим цветом. Стандарт — текст черный, а ссылки в нем синие.
11. При наведении на ссылку курсор мышки принимает форму руки.
12. У ссылок, которые пользователь уже посещал, меняется цвет.
13. Страницы сайта нормально отображаются, если пользователь меняет в браузере размер шрифта, нажимая Ctrl+ или Ctrl-.
14. В навигационном меню есть ссылки на все основные категории страниц сайта или даже на самые важные страницы.
15. На важные страницы и разделы ведет не одна, а несколько разных ссылок. Потому что разные пользователи предпочитают разные пути навигации.
16. На страницах сайта изображения не просто украшают, а по своему содержанию дополняют тексты.
17. На сайте размещены документальные подтверждения экспертной компетенции сотрудников (дипломы, публикации, награды и т. д.).
18. Дизайн сайта подчеркивает его солидность и надежность, но также и современность.
19. На сайте нет всплывающих окон, тем более с раздражающей рекламой.
20. Нет признаков финансовых пирамид, сетевого маркетинга, интим-услуг и т. п.
21. Видно, что сайт регулярно обновляется, публикуются свежие материалы.
22. На всех страницах сайта используются одни и те же шрифты и стили оформления.

в. Атрибуты качества (FURPS):

1. **Functionality** — Функциональные требования: свойства, возможности, безопасность. Являются основными по этим требованиям строятся диаграммы вариантов использования (Use case diagram).
2. **Usability** — Требования к удобству использования (UX): человеческий фактор, эстетика, последовательность, документация.

Окончание приложения А

Проект технического задания

3. **Reliability** — Требования к надежности: частота возможных сбоев, отказоустойчивость, восстанавливаемость, предсказуемость устойчивости.

4. **Performance** — Требования к производительности: время отклика, использование ресурсов, эффективность, мощность, масштабируемость.

5. **Supportability** — Требования к поддержке: возможность поддержки, ремонтпригодность, гибкость, модифицируемость, модульность, расширяемость, возможность локализации.

ПРИЛОЖЕНИЕ Б

Акт об использовании

ООО «Рассвет»
ИНН 2462064228, КПП 246201001
ОГРН 1182468066065
Юр. адрес: 660031 г. Красноярск,
ул. Глинки д. 46, стр. 11
Факт. адрес: 660031 г. Красноярск,
ул. Глинки д. 46 стр. 11
Р/с 40702810423260002111
ФИЛИАЛ «НОВОСИБИРСКИЙ»
АО «АЛЬФА-БАНК»
К/с 3010181060000000774
БИК 045004774
Т. 8 (923) 292-33-37

АКТ ОБ ИСПОЛЬЗОВАНИИ

результатов проектирования в рамках бакалаврской работы

«17» июня 2019 г.

г. Красноярск

ООО «Рассвет» в лице генерального директора Лулева Алексея Николаевича, осуществила приемо-сдаточные испытания интернет-платформы для оказания услуг по ремонту объектов капитального строительства.

Интернет-платформа разработана студентом гр. ВКИ 14-13Б Черновым Дмитрием Дмитриевичем под руководством доцента кафедры «Системы искусственного интеллекта» ИКИТ СФУ Брежнева Руслана Владимировича в рамках выполнения бакалаврской работы.

В настоящее время интернет-платформа внедрена в опытную эксплуатацию. Его использование позволяет клиентам оставлять заявки на ремонт и строительство, а так же регистрироваться в качестве исполнителей и выполнять существующие заказы.

Генеральный директор
ООО «Рассвет»



А.Н. Лулев

Рисунок Б.1 – Акт об использовании

ПРИЛОЖЕНИЕ В

Программный код капчи

```
</div>
<div data-bbox="908 166 941 382" data-label="Text" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">
  meta-stickey="6d4k6yuu44443931v004vdukiL7f-Sd0r0h6">
  ; height: 78px;

```

```


<div data-bbox="845 166 891 642" data-label="Text" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">
  <div data-bbox="798 166 812 206" data-label="Text" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">
    lang="ru">
  
```

Рисунок В.1 – Код капчи

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
ИНСТИТУТ КОСМИЧЕСКИХ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ
Кафедра «Информационных систем»

УТВЕРЖДАЮ

Заведующий кафедрой ИС


подпись

П.П. Дьячук
инициалы, фамилия

«21» 06 2019 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.02 – Информационные системы и технологии

Интернет платформа для оказания услуг по ремонту объектов капитального
строительства

Руководитель

 21.06.19
подпись, дата

доцент, канд. техн. наук

Р.В. Брежнев

Выпускник

 21.06.19
подпись, дата

Д.Д. Чернов

Нормоконтролер

 - 21.06.19
подпись, дата

Ю.В. Шмагрис

Красноярск 2019