МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ

ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»

**институт информационных технологий и технологического образования**

**кафедра информационных технологий и электронного обучения**

Основная профессиональная образовательная программа

Направление подготовки 09.03.01 Информатика и вычислительная техника

Направленность (профиль) «Технологии разработки программного обеспечения»

форма обучения – очная

**Курсовая работа**

по дисциплине «Управление программными проектами»

Управление разработкой проекта конфигурации для "1С:Предприятие" для автоматизации деятельности организации

Обучающегося 3 курса

Собинина Е.Я.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель:

к.ф-м.н, доцент

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Жуков Н. Н.

«\_\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Санкт-Петербург

2022

ОГЛАВЛЕНИЕ

[1 Теоретическая часть 5](#_Toc123273437)

[1.1 Управление программными проектами 5](#_Toc123273438)

[1.2 Методологии управление программными проектами 6](#_Toc123273439)

[1.3 Waterfall 7](#_Toc123273440)

[1.4 Scrum 8](#_Toc123273441)

[1.5 Канбан 9](#_Toc123273442)

[2 Практическая часть 11](#_Toc123273443)

[2.1 Диаграмма Ганта 11](#_Toc123273444)

[2.2 Нотации 12](#_Toc123273445)

[2.3 Разработка системы учета 12](#_Toc123273446)

[2.4 Разработка дизайна 12](#_Toc123273447)

[2.5 Разработка документации и составление базы знаний 12](#_Toc123273448)

[2.5.1 Документация 12](#_Toc123273449)

[2.6.2 Разработка базы знаний 13](#_Toc123273450)

[2.6 Тестирование 14](#_Toc123273451)

[2.6.1 Ручное тестирование 14](#_Toc123273452)

[2.6.2 Автоматическое тестирование 14](#_Toc123273453)

[Приложение 1 16](#_Toc123273454)

[Приложение 2 17](#_Toc123273455)

[Приложение 3 18](#_Toc123273456)

# ВВЕДЕНИЕ

Данная курсовая работа посвящена управлению программным проектом по разработке конфигурации для «1С:Предприятие». Важность управления проектами заключается в том, что оно отслеживает сроки, обязанности, роли и задачи, чтобы ничего не упустить. Хороший руководитель проекта может держать команду хорошо организованной и идти по правильному пути, что повышает производительность и эффективность. Они помогают команде сосредоточиться на долгосрочных целях и работать вместе для достижения общих целей. Понимание принципов управления программным проектом важно для любой организации.

Хотя изучение методов и методологий управления бизнес-проектами может показаться сложной и пугающей задачей, это важный навык для современных руководителей проектами, которым приходится управлять сложными проектами с высоким риском. Кроме того, изучение различных методологий управления проектами поможет разработать более эффективные стратегии и процессы управления для различных типов бизнес-проектов.

Объектом исследования данной работы является управление программным проектами по разработке конфигурации для «1С:Предприятие».

Предметом исследования является методологии разработки управления программными проектами.

Цель исследования – анализ процесса управления разработкой программных проектов.

Существует несколько подходов, которые можно использовать для организации и управления проектом по настройке работы системы 1С. Нашей командой было выбрана методология Scram и систему управления программными проектами Trello.

Структура курсовой работы:

* в первой главе рассмотрен теоретический материал: общие положения о управлении бизнес-проектами и методологиях, а также их сравнительная характеристика
* во второй главе рассмотрена реализация задач курсовой работы: …

# 1 Теоретическая часть

## 1.1 Управление программными проектами

На сегодняшний день управление проектами является неотъемлемой чертой в успешности проекта. Важность управления проектами заключается в обеспечении того, чтобы проект был завершен к определенному сроку эффективным и минимально затратным способом. Управление проектами имеет важное значение, поскольку оно позволяет компаниям заранее выявлять потенциальные риски или проблемы и принимать меры для их смягчения. Это также помогает поддерживать участие всех членов команды в процессе, так что все работают для достижения одной цели.

Программный проект – это комплект взаимосвязанных операций, осуществляемых группой специалистов проекта, имеющих целью создание уникального программного продукта или услуги за требуемый временной период при заданном бюджете и используемых при выполнении проекта ресурсах.

Управление программным проектом является работой по организации управления на всех этапах жизненного цикла программного продукта, за счёт составления плана, создания, мониторинга и контроля работ по проектированию, при которых реализуется достижения проектной цели при наличии компромиссных решений между объёмом работ, ресурсными и временными затратами, уровнем качества и допустимым риском. Цели программного проекта необходимо назначать в форме требуемого итогового результата, который должен быть достигнут специалистами проекта, если реализация успешна.

Благодаря этому менеджеры проекта могут контролировать следующие пармаетры:

* сроки;
* объём (количество полезных функций);
* объём работы;
* полезность (ценность для Потребителя);
* техническая и технологическая сложность;
* внешнее качество (с точки зрения Потребителя);
* внутреннее качество (с точки зрения специалиста, инженера, конструктора);
* риски;
* психологическое состояние команды.

В конечном счете, управление проектами необходимо для любого успешного бизнеса, поскольку оно позволяет повысить эффективность и прибыльность. Хотя поначалу это может показаться большой работой, преимущества управления проектами в долгосрочной перспективе окупают затраченные ресурсы.

## 1.2 Методологии управление программными проектами

Одна из самых важных вещей при управлении программным проектом — это выбор методологии, которая лучше всего подходит для команды и проекта. Это может быть Scrum, Kanban или другой подход. Важно уделить время обдумыванию того, какая методология лучше всего подойдет для конкретного проекта, поскольку необходимо учитывать множество факторов. Например, размер, сложность и объем проекта, уровень опыта команды и культура организации.

В конечном счете выбор правильной методологии для проекта может стать неотъемлемой частью его успеха, поэтому важно принять обоснованное решение. Для этого были изучены наиболее популярные методологии и их проведена их сравнительная характеристика.

Каждая методология имеет собственный подход к систематизации процессов, процедур, передовых практик и шаблонов документов, необходимых для успешного управления проектами. Если внимательно к ним присмотреться, то мы обнаружим во всех методологиях очень много похожего, тем не менее, каждая методика имеет свои достоинства и недостатки.

Рассмотрим наиболее популярные методологии, чтоб выбрать подходящую для реализуемого проекта.

## 1.3 Waterfall

Каскадная модель управления, также известная как «водопад», тоже довольно популярна. «Водопад» — это настоящая методология с очень чёткими правилами. Каскадная методология, также известная как цикл разработки программного обеспечения (ЦРПО) представляет собой линейный процесс, в котором работа ниспадает каскадом (как водопад) и организована в последовательном порядке.

При использовании этого подхода все рабочие задачи связываются друг с другом зависимостями. Это означает, что для того, чтобы начать работу над задачей, должна быть выполнена предшествующая ей задача. Благодаря этому работа идёт по плану, а также обеспечивается чёткий обмен информацией в течение всего процесса.

Хотя некоторые современные организации считают данный подход устаревшим, эта методология отлично подходит для создания предсказуемого и хорошо продуманного плана проекта.

Поскольку каскадная методология управления является очень подробной, она хорошо подходит для работы над крупными проектами с множеством заинтересованных сторон. Эта модель обеспечивает наличие чёткой информации о необходимых действиях в течение всего проекта и зависимостей, позволяющих отследить работу, которую следует выполнить для достижения целей.

Основное преимущество методологии Waterfall заключается в том, что она обеспечивает четко определенный и предсказуемый процесс управления проектами. Это также очень логичный и методичный подход, позволяющий тщательно планировать и документировать.

Однако одним из основных недостатков методологии Waterfall является то, что она может быть довольно жесткой и не допускает гибкости или изменения требований. Это может привести к задержкам и увеличению стоимости. Также важно отметить, что методология Waterfall не всегда является лучшим подходом для каждого проекта — перед выбором конкретной методологии необходимо учитывать множество факторов.

## 1.4 Scrum

Методология Scrum предусматривает использование коротких «спринтов», из которых формируется цикл проекта. Эти промежутки длятся от одной до двух недель и рассчитаны на команды в составе не более 10 человек. Это основное отличие от каскадной методологии, где отдельные задачи связываются друг с другом зависимостями.

У Scrum много уникальных особенностей, одной из которых является наличие мастера Scrum или, другими словами, руководителя проекта, который проводит ежедневные Scrum-совещания, демонстрации, спринты и ретроспективы после окончания спринтов. Все эти встречи нужны для общения ключевых участников проекта и своевременного выполнения задач.

Несмотря на то, что технически Scrum является самостоятельной методологией управления проектами, её часто ассоциируют с системой Agile. Связано это с тем, что два эти подхода объединены общими принципами, в том числе принципом важности совместной работы и тем, что личность ценится выше процессов.

Командам, применяющим подход Agile, также следует прибегнуть к методологии Scrum, или, по крайней мере, попробовать её в действии. Так как спринты проводятся для небольших команд, этот подход работает как для небольших, так и для крупных коллективов.

Основное преимущество методологии Scrum заключается в том, что она обеспечивает очень гибкий и динамичный подход к управлению проектами. Это позволяет быстро изменять требования, графики поставок и даже объем проекта.

У методологии Scrum также есть несколько недостатков. Во-первых, это может быть проблемой для поддержания подотчетности членов команды. Это связано с тем, что для каждого члена команды нет установленных ролей или обязанностей. Кроме того, соблюдение сроков может быть затруднено, поскольку нет установленного графика выполнения работы.

## 1.5 Канбан

В методологии Канбан невыполненные задачи в рамках проекта представляются с помощью визуальных элементов, а именно досок. Этот подход используется Agile-командами для эффективной визуализации процессов и хода выполнения проектов, а также снижения вероятности возникновения задержек. Чаще всего для этого используется программное обеспечение, в котором можно легко перетаскивать доски внутри проектов, хотя это и не обязательное требование.

Поскольку, в отличие от других, этот метод не имеет строго определённого процесса, команды используют его по-разному. Здесь нужно понимать, что в Канбан основное внимание уделяется наиболее важным задачам проекта, структура же остаётся довольно простой.

Канбан-доски могут использовать коллективы любых размеров, а особенно этот вариант хорош для удалённых команд. Связано это с тем, что визуальные возможности Канбан-досок позволяют сотрудникам оставаться в курсе происходящего, где бы они ни были.

Основное преимущество методологии Канбан заключается в том, что она обеспечивает очень наглядный и прозрачный подход к управлению проектами. Это позволяет легко увидеть, где в настоящее время выполняется работа и какие этапы она должна пройти.

Главный недостаток методологии Канбан заключается в том, что трудно отслеживать прогресс в долгосрочных проектах. Также может быть сложно определить сроки и результаты, если требования четко не определены.

Рассмотрев различные методологии в процессе обсуждения было решено придерживаться методологии Scrum в процессе разработки программного продукта.

# 2 Практическая часть

## 2.1 Диаграмма Ганта

Диаграмма Ганта - тип диаграмм, который используется для демонстрации плана работ. Используется в планировании проектов.

Диаграмма Ганта состоит из горизонтальных полос вдоль оси времени, концы диаграммы - начало и конец работы. Одна полоса - одна задача в составе проекта. Вертикальная ось диаграммы - перечень всех задач. Кроме того, на такой диаграмме возможно указать совокупные задачи, проценты завершения, указатели последовательности и зависимости работ.

В рамках нашего учебного проекта диаграмма Ганта использовалась для лучшего понимания сроков выполнения задач. Сама диаграмма была составлена членами нашей команды средствами MS Excel.

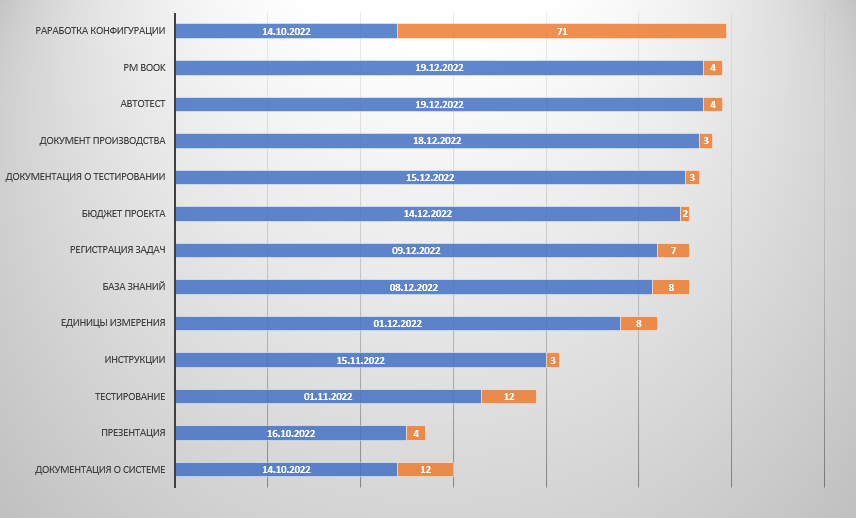


Рисунок – реализация диаграммы Ганта средствами MS Excel

## 2.2 Нотации

Для успешного функционирования нашей системы участниками команды были составлены некоторые нотации для лучшего понимания архитектуры нашей системы.

Всего было составлено 3 нотации, а именно «Открытие магазина» (Приложение 1), «Процесс закупки» (Приложение 2), и «Продажа товаров» (Приложение 3). Для создания использовалось специальное средство Aris.

## 2.3 Разработка системы учета

…

## 2.4 Разработка дизайна

Разработка дизайна велась методом Brainstorm, в ходе которого все участники команды выдвинули свои предложения, из которых были взяты наилучшие варианты.

Для лучшего понимания цветового оформления, была взята палитра цветов. Итоговое решение создано согласно принципу сочетания цветов «Монохромия». Кроме того цвета удачно сочетаются с тематикой магазина, для которого мы создаем конфигурацию.

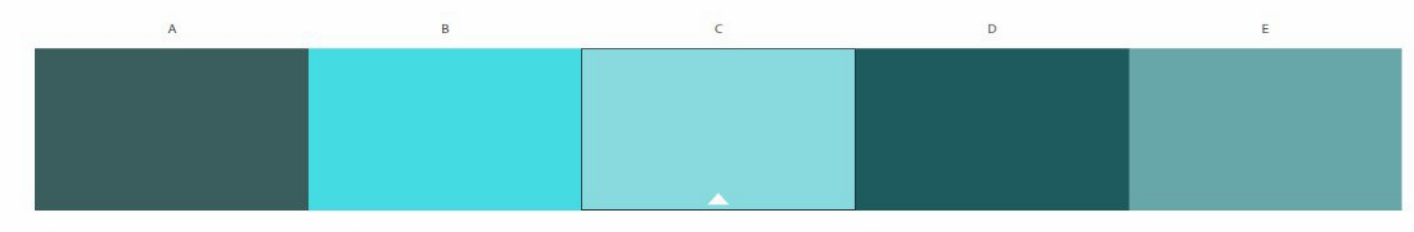


Рисунок – цветовое решение

Так же было принято решение изменить стандартные иконки конфигурации, чтобы интерфейс стал более интуитивно-понятным.



Рисунок – иконки

## 2.5 Разработка документации и составление базы знаний

### 2.5.1 Документация

Документация проекта – одна из самых важных вещей при реализации проекта. После создания документации экономится огромное количество времени, которое нужно потратить для новых участников группы при разработке системы управления.

Написание документации фактически производилось двумя людьми: одним разработчиком и одним писателем технических текстов. Такой подход был необходим для ускорения написания документации и скорейшего ввода ее в эксплуатацию.

Документация писалась средствами текстового процессора MS Word. Документация обладает достаточной технической информацией для запуска нашей системы на персональном компьютере или сервере, а также для разработки и поддерживания итогового продукта. Как пример были взяты некоторые шаблоны из интернета, для лучшего понимания структуры необходимого файла.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок – разработка документации

### 2.6.2 Разработка базы знаний

База знаний по большей части писалась для оператора (пользователя) информационной системой. Для реализации был выбран онлайн сервис Notion, в основном благодаря его богатому функционалу, и возможностью опубликовать документ в виде веб-страницы. Данная база знаний написана членами команды, а именно одним разработчиком и одним писателем технических текстов.

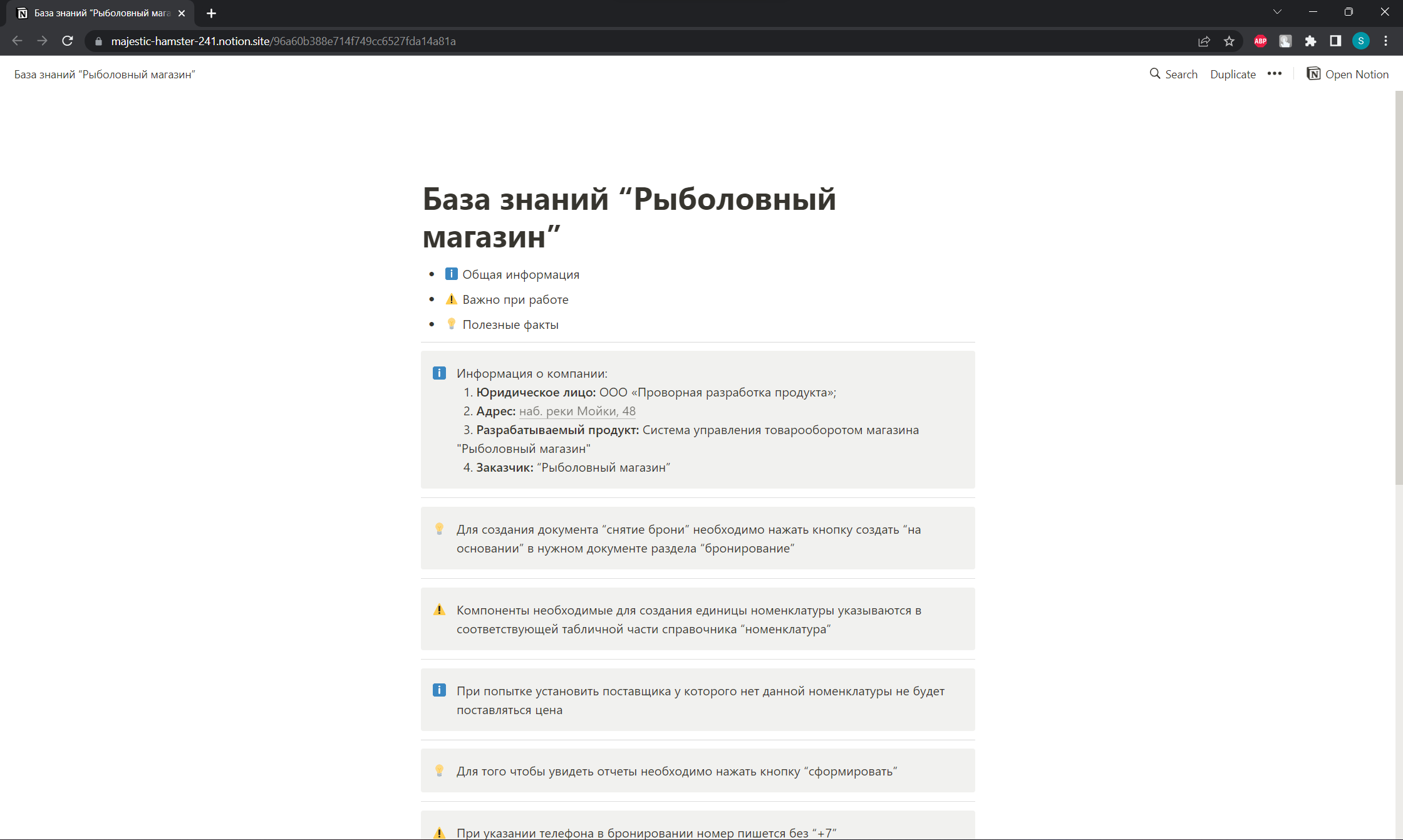


Рисунок – база знаний

## 2.6 Тестирование

Тестирование системы было проведено двумя разными способами: ручное тестирование и автоматическое тестирование. Оба способа хороши, однако у всех способов есть свои достоинства и недостатки.

### 2.6.1 Ручное тестирование

Так ручное тестирование было довольно эффективно на ранних стадиях проекта, так как приложение имело крайне ограниченный функционал. Такой способ тестирования был крайне быстр, для этого способа не нужны никакие специальные знания и в крайнем случае всегда можно делегировать эту задачу любому участнику проекта. Однако со временем функционал программы разросся до такой степени, что ручное тестирование становится очень долгим и трудоемким процессом.

Итогом ручного тестирования всегда был файл-отчет, в котором перечислялись все пройденные и не пройдённые тесты, а также в конце писался общий вывод с перечислением всех ошибок.

### 2.6.2 Автоматическое тестирование

Автоматическое тестирование занимает намного больше времени с точки зрения погружения. Так тестировщику необходимо подробно изучить функционал программы, для автоматического тестирования и разобраться с синтаксисом написания тестов. Однако, в результате это дает нам возможность довольно быстро тестировать все функции нашей конфигурации.

Автоматическое тестирование проводилось средствами vanessa-automation. Для проведения автоматического тестирования необходимо установить данную программу, и открыть ее средствами 1С: Предприятие. Данная программа, а точнее клиент тестирования запускает нашу конфигурацию как пользователь и выполняет скрипт, написанный тестировщиком.

Несмотря на ограничение по количеству подключений на учебной версии 1С: Предприятие, провести автоматическое тестирование возможно, для этого необходимо создать тестировочную базу и тестируемую базу. Из первой мы запускаем vanessa-automation, а вторая является базой, которую необходимо протестировать (Приложение 4).

В самой программе мы можем писать самые разные тесты, реализуемые с помощью скриптов. Всего есть два варианта создания тестов: запись сценария и написание скрипта.

Запись сценария, это метод, который, по сути, не отличается от ручного тестирования. Необходимо нажать на соответствующую кнопку, после чего откроется тестируемая информационная система. Vanessa-automation будет записывать каждый шаг, который происходит в тестируемой системе. По завершению записи сформируется скрипт, который можно будет запустить позже.

Использовать даже такой метод можно, так как в конце концов у нас все равно остается автоматический тест, которым можно будет пользоваться, чтобы убедиться, что старый функционал все еще корректно работает.

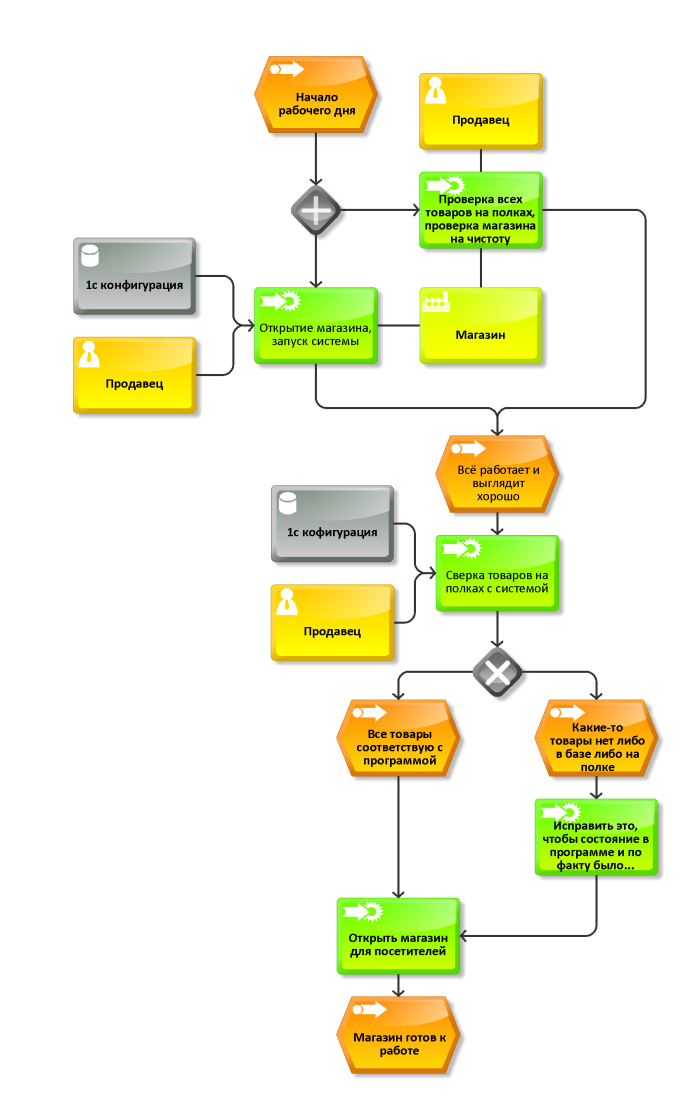
Изображение выглядит как текст

Автоматически созданное описание

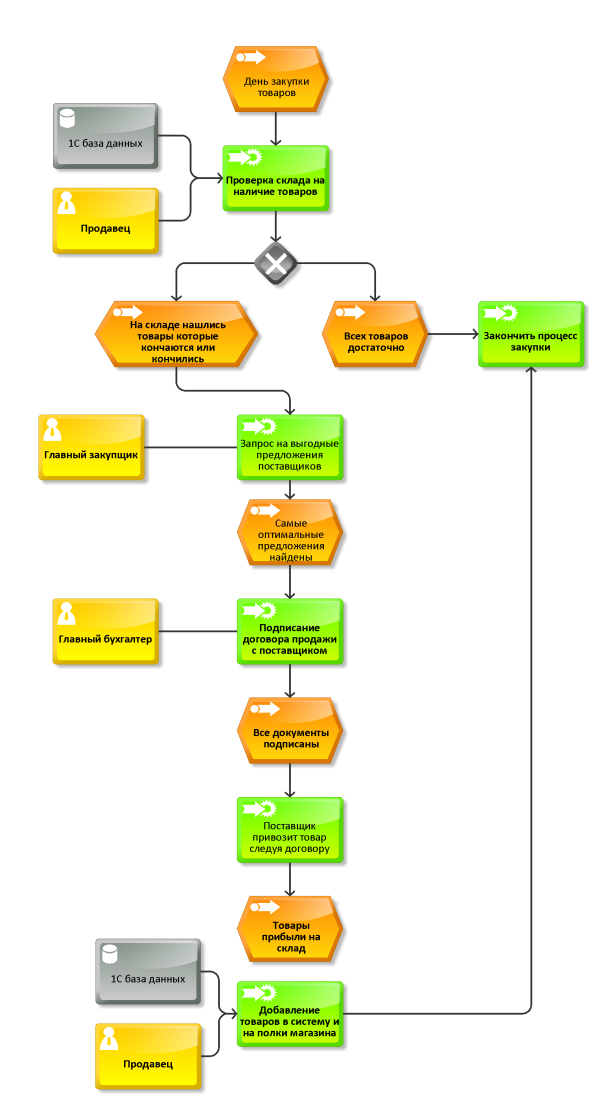
Рисунок – запись сценария

В случае написания кода, при условии хорошего уровня знания языка, можно значительно ускорить процесс написания тестов. Кроме того, становится возможным написание комплексных тестов для всей системы.

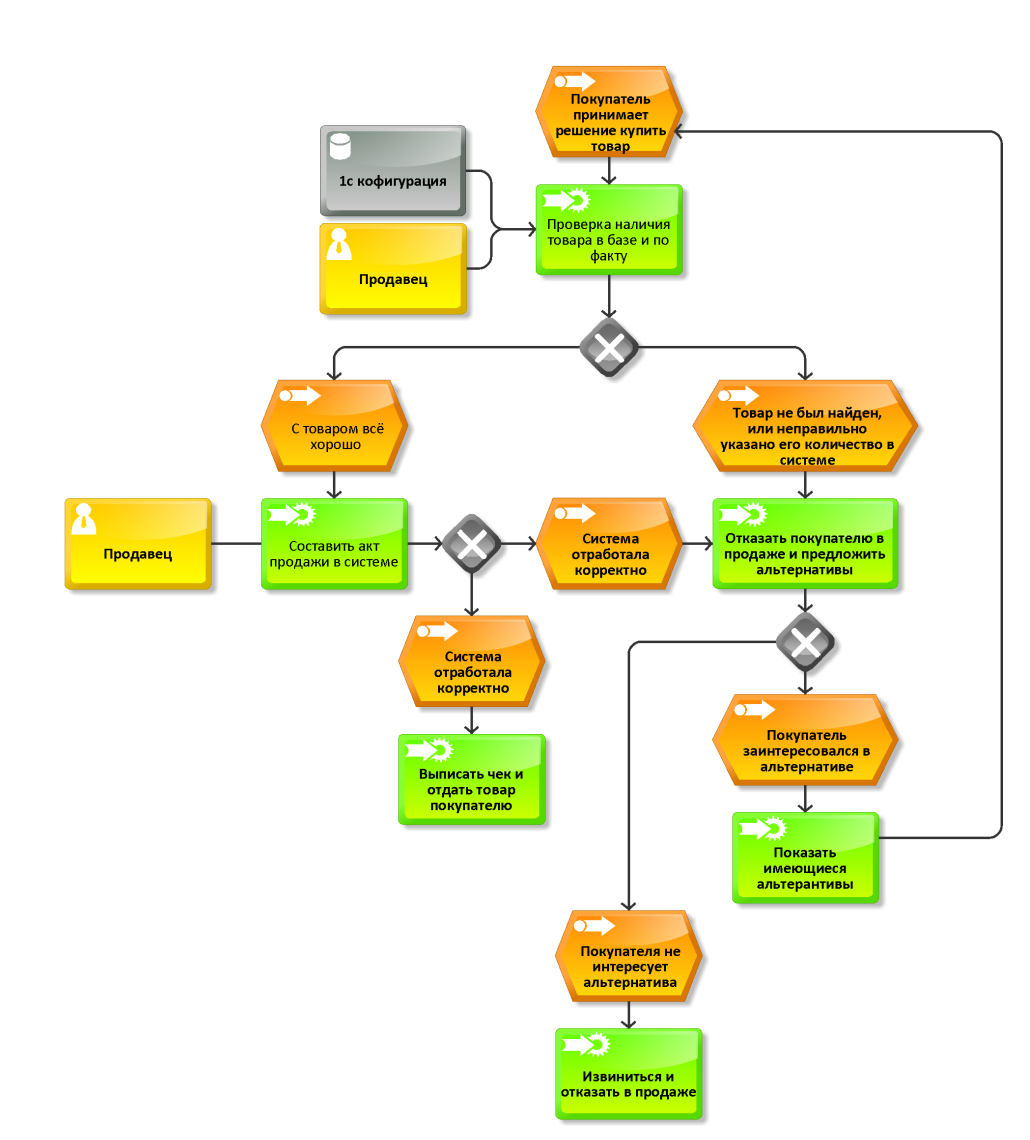
# Приложение 1



# Приложение 2



# Приложение 3



# Приложение 4

Изображение выглядит как текст

Автоматически созданное описание

# Приложение 5