

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	5
1 АНАЛИТИЧЕСКИЙ РАЗДЕЛ	6
1.1 Статистика потерянных и найденных вещей	6
1.2 Типы существующих решений для поиска и возврата утерянных вещей	6
1.3 Анализ существующих систем для поиска и возврата утерянных вещей	8
2 СПЕЦИАЛЬНЫЙ РАЗДЕЛ	11
2.1 Требования к разрабатываемой системе	11
2.1.1 Функциональные требования	11
2.1.2 Нефункциональные требования	11
2.2 Проектирование модулей автоматизации процессов	12
2.2.1 Модуль бесконечных лент объявлений потерянных, найденных вещей	12
2.2.2 Модуль полнотекстового поиска объявлений	12
2.2.3 Модуль генерации описания объявлений	12
3 ТЕХНОЛОГИЧЕСКИЙ РАЗДЕЛ	13
3.1 Выбор технологий для реализации системы	13
3.2 Реализация модулей автоматизации процессов	13
3.2.1 Модуль бесконечных лент объявлений потерянных, найденных вещей	13
3.2.2 Модуль полнотекстового поиска объявлений	13
3.2.3 Модуль генерации описания объявлений	13
4 ЭКОНОМИЧЕСКИЙ РАЗДЕЛ	14
4.1 Планирование разработки системы	14
4.1.1 Определение трудоемкости и продолжительности работ по созданию УСПД	14
4.1.2 Построение ленточного графика проведения исследования	17
4.2 Расчет сметы затрат на разработку представленной работы . .	19
4.2.1 Основная заработная плата исполнителей	19

4.2.2	Дополнительная заработная плата исполнителей	20
4.2.3	Расчет отчислений на социальное страхование	20
4.2.4	Накладные расходы	20
4.2.5	Расчет стоимости машинного времени	21
ЗАКЛЮЧЕНИЕ		24
ПРИЛОЖЕНИЕ А		26

ВВЕДЕНИЕ

Поиск утерянных вещей является актуальной проблемой, которая возникает при различных обстоятельствах. Эта проблема может возникнуть в результате потери ключей, документов, мобильных телефонов, кошельков или других ценных или важных вещей. В связи с этим существует необходимость разработки системы, которая поможет людям вернуть утерянные вещи.

Целью данной работы является разработка системы для поиска утерянных вещей на основе анализа существующих систем и технологий, а также определение требований к системе и ее функциональности. Для достижения этой цели будут рассмотрены различные методы и технологии, которые могут быть использованы для создания такой системы.

В разделе 1 будет проведен анализ существующих систем поиска утерянных вещей и выделены их преимущества и недостатки. Во разделе 2 будут определены требования к разрабатываемой системе, рассмотрены методы и технологии, которые можно использовать для реализации системы. Раздел 3 будет посвящен описанию процесса разработки и тестирования системы. В разделе 4 будет приведен план разработки и расчет сметы затрат.

Таким образом, разработка системы для поиска утерянных вещей позволит создать удобный инструмент для поиска потерянных вещей, что приведет к уменьшению количества потерянных вещей и улучшению качества жизни людей.

1 АНАЛИТИЧЕСКИЙ РАЗДЕЛ

1.1 Статистика потерянных и найденных вещей

Для подтверждения актуальности и важности разрабатываемой системы, необходимо провести исследование рынка и определить основные проблемы и потребности пользователей. Одним из способов сбора информации является проведение опроса среди пользователей.

Одним из основных факторов, определяющих актуальность разрабатываемой системы является статистика потерянных и найденных вещей. Необходимо определить количество потерянных вещей в месяц, год и за весь период работы системы. Это поможет оценить нагрузку на систему и определить ее производительность.

Статистика, взятая с сайта столнаходок.рф [1], утверждает, что только 20 % пользователей их сайта смогли установить и вернуть вещи. Также на рисунках 1 и 2 представлена гистограмма количества созданных объявлений за 2022 и 2023 года.

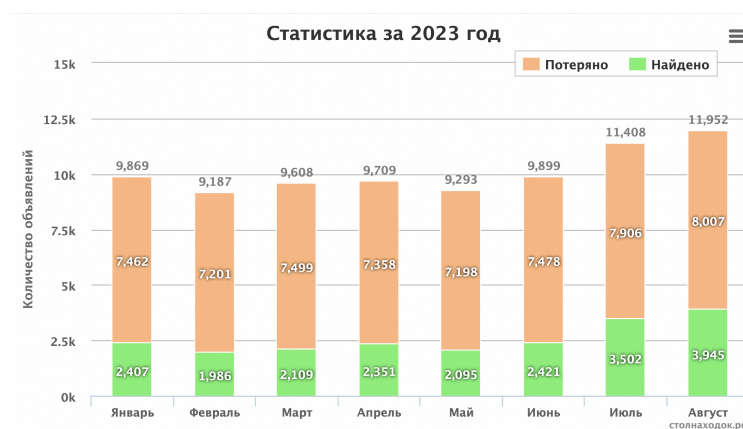


Рисунок 1 – Востребованность системы столнаходок.рф в 2023 году

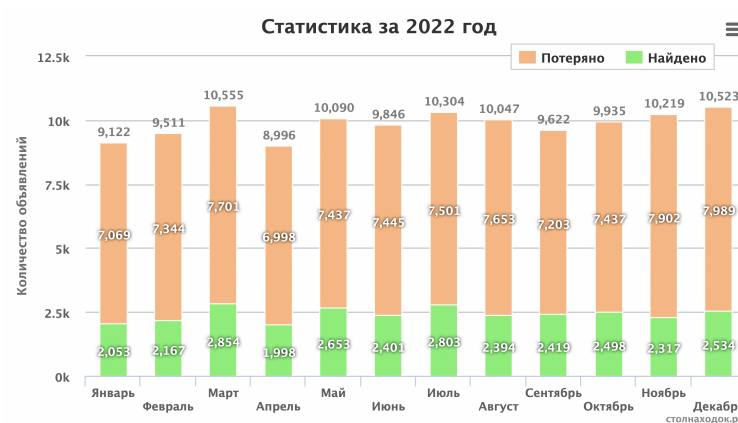


Рисунок 2 – Востребованность системы столнаходок.рф в 2022 году

1.2 Типы существующих решений для поиска и возврата утерянных вещей

Существует несколько типов существующих решений для поиска и возврата утерянных вещей. Ниже приведены некоторые из них:

- 1) Веб-сайты и приложения «Бюро находок»: Эти сервисы предоставляют платформу для регистрации утерянных вещей и поиска их владельцев. Пользователи могут создавать объявления о потерянных или найденных вещах и связываться друг с другом для возврата. Некоторые из этих сервисов предлагают возможность добавления фотографий или описания вещи для облегчения поиска.
- 2) Технология RFID (Radio Frequency Identification): RFID-метки могут быть прикреплены к ценным вещам, и их сигнал может быть отслежен с помощью специального считывателя. Это позволяет пользователям быстро определить местоположение утерянных вещей через дополнительное программное обеспечение.
- 3) GPS-трекеры: Эти устройства с GPS-модулем могут быть прикреплены к вещам, и их местоположение может быть отслежено с помощью специализированного приложения для смартфонов или веб-панели управления. Пользователи могут получать уведомления о перемещении вещи и быстро определить ее местонахождение.
- 4) Автоматизированные системы обнаружения: Некоторые организации, такие как аэропорты или железнодорожные станции, имеют системы обнаружения утерянных вещей. Эти системы используют технологии,

такие как видеонаблюдение, датчики движения или распознавание образов, чтобы отслеживать и возвращать потерянные вещи своим владельцам.

Каждый из этих типов решений имеет свои преимущества и недостатки. Некоторые из них могут быть более подходящими для конкретных ситуаций, например, GPS-трекеры могут быть полезными при поиске утерянных вещей на открытой местности, в то время как RFID-метки могут быть более подходящими для использования внутри помещений. Веб-сайты и приложения «Бюро находок» предоставляют более универсальное решение, которое может быть использовано в различных ситуациях.

1.3 Анализ существующих систем для поиска и возврата утерянных вещей

В настоящем разделе будет проведен обзор существующих сервисов и приложений, которые предлагают функциональность поиска и возврата утерянных вещей. Данный обзор позволит выявить основные преимущества и недостатки этих сервисов, а также определить потенциальные возможности для улучшения их функциональности.

«столнаходок.рф» [1] — это один из наиболее популярных веб-сервисов, предоставляющих возможность объявлять о потерянных и найденных предметах. Сервис имеет простой и интуитивно понятный интерфейс, позволяющий пользователям быстро разместить информацию о потерянных вещах и связаться с владельцами найденных предметов. Однако, отсутствие системы уведомлений и неудобное сопоставление объявлений ограничивают его функциональность.

«Find My Stuff» [2] — это мобильное приложение, разработанное для операционных систем iOS и Android. Оно предлагает функцию отслеживания утерянных предметов через GPS-модуль смартфона. Пользователи могут отмечать свои вещи на карте и получать уведомления, когда они находятся рядом с утерянным предметом. Однако, ограничение использования только наличием смартфона с GPS-модулем и низкая точность определения местоположения представляют существенные ограничения данного приложения.

«Lost Property Office» [3] — это веб-сервис, предоставляемый государственными организациями и органами правопорядка. Сервис позволяет пользователям сообщать о потерянных и найденных предметах, а также предоставляет информацию о процедуре возврата утерянных вещей. Однако, ограниченный доступ к сервису и неудобный процесс регистрации и подачи заявки являются значительными недостатками данного сервиса.

На основании проведенного обзора можно сделать вывод, что существующие веб-сервисы и приложения для поиска и возврата утерянных вещей имеют некоторые преимущества, но также недостатки, которые ограничивают их функциональность и удобство использования. Веб-сервис Бюро находок будет разработан с учетом этих недостатков и предлагать более удобное взаимодействие между пользователями и сервисом.

Ниже приведена сравнительная таблица 1 основных характеристик и функций приведенных выше аналогов:

Таблица 1 – Сравнительная таблица аналогов

Сервис / Приложение	Интерфейс и удобство использования	Оповещения	Точность определения местоположения	Регистрация и подача заявки	Доступность
столна-ходок.рф	Простой и интуитивно понятный интерфейс	Отсутствуют	Неопределено	Простой процесс регистрации	Широкий доступ
Find My Stuff	Простой и интуитивно понятный интерфейс	Оповещения через уведомления	Низкая точность	Простой процесс регистрации	Доступен только на смартфонах с GPS
Lost Property Office	Неудобный процесс регистрации и подачи заявки	Отсутствуют	Неопределено	Неудобный процесс регистрации и подачи заявки	Ограниченный доступ

Вывод по разделу

В аналитическом разделе моего исследования проведен подробный обзор различных существующих веб-сервисов и приложений, которые предназначены для поиска и возврата утерянных вещей. Мы изучили и проанализировали их функциональность, особенности, преимущества и недостатки.

Веб-сервисы и приложения «Бюро находок» представляют собой одно из самых популярных и широко используемых решений в данной области. Они предоставляют платформу, на которой пользователи могут зарегистрировать утерянные вещи и связаться с их владельцами. Это позволяет упростить процесс поиска и возврата утерянных вещей, обеспечивая удобный и интуитивно понятный интерфейс для пользователей.

2 СПЕЦИАЛЬНЫЙ РАЗДЕЛ

2.1 Требования к разрабатываемой системе

TODO

2.1.1 Функциональные требования

ER-диаграмма представлена на рис. 3.

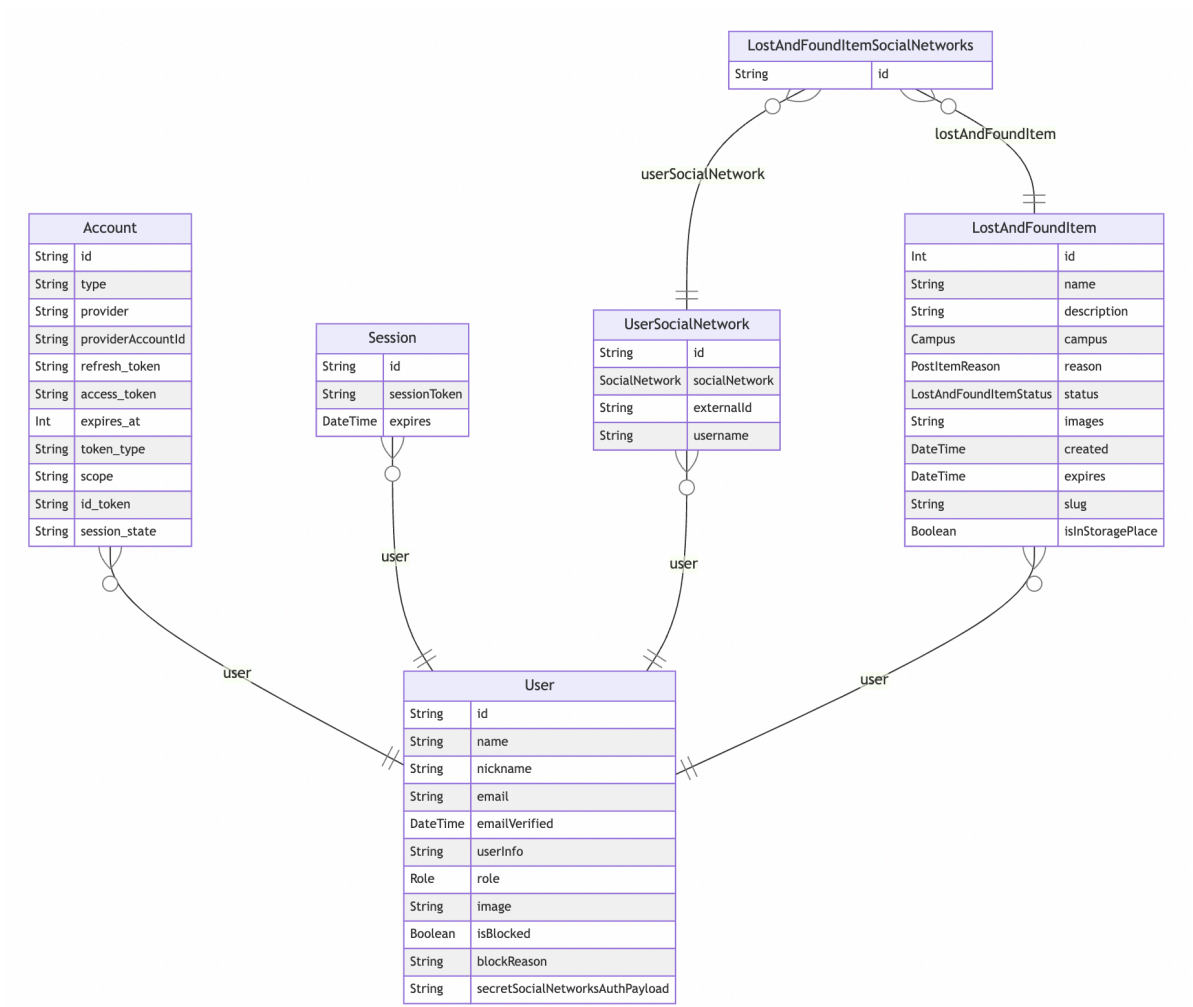


Рисунок 3 – ER-диаграмма системы

2.1.2 Нефункциональные требования

TODO

2.2 Проектирование модулей автоматизации процессов

TODO

2.2.1 Модуль бесконечных лент объявлений потерянных, найденных вещей

TODO

2.2.2 Модуль полнотекстового поиска объявлений

TODO

2.2.3 Модуль генерации описания объявлений

TODO

Вывод по разделу

TODO

3 ТЕХНОЛОГИЧЕСКИЙ РАЗДЕЛ

3.1 Выбор технологий для реализации системы

TODO

3.2 Реализация модулей автоматизации процессов

TODO

3.2.1 Модуль бесконечных лент объявлений потерянных, найденных вещей

TODO

3.2.2 Модуль полнотекстового поиска объявлений

TODO

3.2.3 Модуль генерации описания объявлений

TODO

Вывод по разделу

TODO

4 ЭКОНОМИЧЕСКИЙ РАЗДЕЛ

4.1 Планирование разработки системы

Основой данной выпускной квалификационной работы является «Разработка системы для поиска и возврата утерянных вещей». В данном разделе определяются уровень сложности и затраты на создание программного и аппаратного обеспечения, а также оценивается экономическая выгода, которую можно получить от использования разрабатываемого программного обеспечения.

4.1.1 Определение трудоемкости и продолжительности работ по созданию УСПД

В этом разделе определяются уровень сложности и затраты на создание программного и аппаратного обеспечения, а также проводится оценка экономической выгоды от использования разрабатываемого ПО. Процесс разработки включает анализ предметной области, имитационный анализ, создание, настройку и тестирование системы.

- Техническое задание. Регламентированы следующие этапы исследования: составление технического задания, включающего формулировку задач, подбор литературы, сбор исходных данных, определение системных требований, а также определение этапов, фаз и сроков разработки программного обеспечения.
- Эскизный проект. Этот этап включает в себя использование программных средств для анализа схожих тем, разработки общих программных структур и структур по подсистемам, создания прототипов и документации.
- Технический проект. Этот этап включает в себя определение требований к программному обеспечению и выбор инструментов и использование программных средств.
- Рабочий проект. Этап включает компоновку и дизайн, программирование, тестирование и отладку ПО, проектирование плат, а также координацию и утверждение работоспособности всей системы.

— Внедрение. Подразумевает под собой использование на реальной инфраструктуре; анализ полученных данных в результате исследований, благодаря которым можно будет скорректировать техническую документацию.

Трудоемкость работ по созданию программного обеспечения носит вероятностный характер, поскольку определяется суммой сложности этапов и видов работ, оцениваемых экспертами в ручные дни, и зависит от многих факторов, которые трудно учесть.

Трудоемкость каждого вида работ определяется по формуле (1).

$$t_i = \frac{3 \cdot t_{min} + 2 \cdot t_{max}}{5}, \quad (1)$$

где t_{min} — минимально возможная трудоемкость выполнения отдельного вида работ;

t_{max} — максимально возможная трудоемкость выполнения отдельного вида работ.

Различные виды работ имеют свою продолжительность в календарных днях (T_i), определяясь по формуле (2), в днях:

$$T_i = \frac{t_i}{\text{Ч}_i} \cdot K_{\text{вых}}, \quad (2)$$

где t_i — трудоемкость работ, человеко-дней;

Ч_i — численность исполнителей, человек;

$K_{\text{вых}}$ — коэффициент, учитывающий выходные и праздничные дни, находится по формуле (3):

$$K_{\text{вых}} = \frac{K_{\text{кал}}}{K_{\text{раб}}}, \quad (3)$$

где $K_{\text{кал}}$ — календарные дни;

$K_{\text{раб}}$ — рабочие дни;

$K_{\text{вых}}$ — 1,48.

Далее предоставляется перечень разновидностей и стадий рабочей деятельности по разработке ПО, экспертные оценки, а также рассчитываемые переменные их трудоемкости, а также продолжительность каждого вида работ, рассчитанные по формулам (1) и (2), представлены в таблице 2.

Таблица 2 – Расчет трудоемкости и продолжительности работ по созданию ПО и аппаратных средств календарных дней

№ работы	Стадии разработки	Трудоемкость, чел. дни			Количество работников, чел.	Продолжительность работ, календарные дни
		t_{min}	t_{max}	t_i		
		t_{min}	t_{max}	t_i	$Ч_i$	T_i
Техническое задание						
1	- постановка задачи	3	6	4.2	1	6.2
2	- подбор литературы	2	3	2.4	1	3.6
3	- сбор исходных данных	4	5	4.4	1	6.5
4	- определение требований к системе	3	4	3.4	1	5.0
5	- определение стадий, этапов и сроков разработки ПО	2	3	2.4	1	3.6
Эскизный проект						
6	- анализ программных средств схожей тематики	7	8	7.4	1	11.0
7	- разработка общей структуры ПО	4	8	5.6	1	8.3
8	- разработка структуры программы и подсистем	5	8	6.2	1	9.2
9	- создание прототипа	4	6	4.8	1	7.1
10	- документирование	2	3	2.4	1	3.6
11	- определение требований к ПО	3	4	3.4	1	5.0
12	- выбор инструментальных средств	3	4	3.4	1	5.0

Продолжение таблицы 2.

№ работы	Стадии разработки	Трудоемкость, чел. дни			Количество работников, чел.	Продолжительность работ, календарные дни
Рабочий проект						
13	- программирование	20	25	22	1	32.6
14	- тестирование и отладка ПО	7	8	7.4	1	11.0
15	- разработка программной документации	4	6	4.8	1	7.1
16	- согласование и утверждение работоспособности системы	2	3	2.4	1	3.6
Внедрение						
17	- опытная эксплуатация	6	8	6.8	1	10.1
18	- анализ данных, полученных в результате эксплуатации	2	4	2.8	1	4.1
19	- корректировка технической документации по результатам испытаний	1	2	1.4	1	2.1
	Общая трудоемкость разработки			98		144

Следовательно, общая продолжительность проведения работ $T_i = 144$.

4.1.2 Построение ленточного графика проведения исследования

В качестве инструмента планирования используется ленточный график (4).

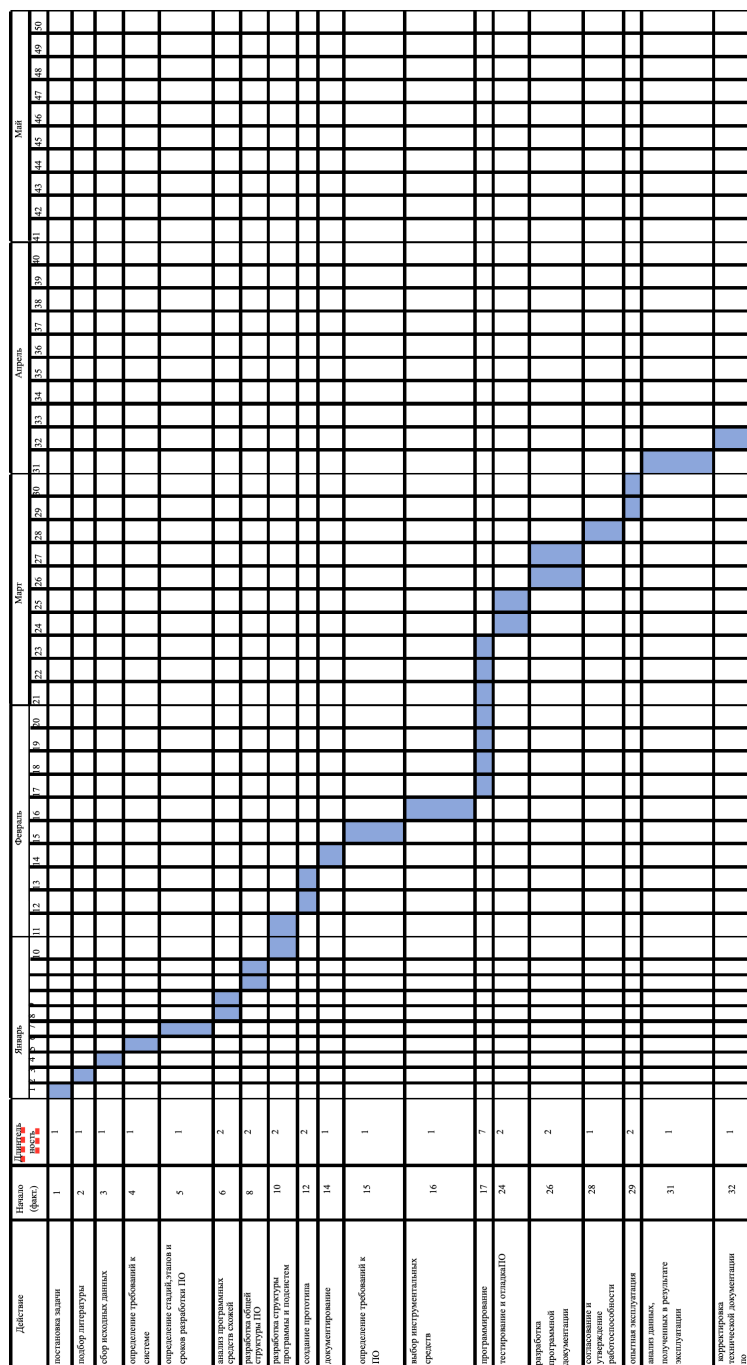


Рисунок 4 – Ленточный график

Подобный график позволяет наглядно представить логическую последовательность и взаимосвязь отдельных работ. График представляет собой таблицу с перечислением названий стадий разработки, видов работ, длительность выполнения работ. Данный график построен по данным таблицы 2. В этом графике временная единица выполнения работ оценивается в 3 дня.

4.2 Расчет сметы затрат на разработку представленной работы

Сметная стоимость проектирования и внедрения программы включает в себя затраты, определяемые по формуле (4):

$$C_{\text{пр}} = C_{\text{осн}} + C_{\text{доп}} + C_{\text{соц}} + C_{\text{м}} + C_{\text{маш.вр}} + C_{\text{н}}, \quad (4)$$

где $C_{\text{пр}}$ — стоимость разработки ПО;

$C_{\text{осн}}$ — основная заработная плата исполнителей;

$C_{\text{доп}}$ — дополнительная заработная плата исполнителей, учитывающая потери времени на отпуска и болезни (принимается в среднем 10% от основной заработной платы);

$C_{\text{соц}}$ — отчисления в фонд социального страхования – 30% от основной и дополнительной заработной платы;

$C_{\text{м}}$ — затраты на используемые материалы;

$C_{\text{маш.вр}}$ — стоимость машинного времени;

$C_{\text{н}}$ — накладные расходы включают затраты на управление, уборку, ремонт, электроэнергию, отопление и др. (принимаются в размере 60% от основной и дополнительной заработной платы).

4.2.1 Основная заработная плата исполнителей

На статью «Заработная плата» относят заработную плату научных, инженерно-технических и других работников, непосредственно участвующих в разработке ПО. Расчет ведется по формуле (5):

$$Z_{\text{исп}} = Z_{\text{ср}} \cdot T, \quad (5)$$

где $Z_{\text{исп}}$ — заработная плата исполнителей (руб.);

$Z_{\text{ср}}$ — средняя тарифная ставка работника организации разработчика ПО (руб./чел./дни);

T — трудоемкость разработки ПО (чел.дни).

$Z_{\text{ср}}$ определяется по формуле (6):

$$Z_{\text{ср}} = \frac{C}{\Phi_{\text{мес}}}, \quad (6)$$

где C — зарплата труда на текущий момент времени (руб./мес.);

$\Phi_{\text{мес}}$ — месячный фонд рабочего времени исполнителя (дни).

Затраты на статью «Заработной платы» приведены в таблице 3.

Таблица 3 – Затраты на заработную плату

№	Исполнитель	Оклад, руб./мес.	Оклад, руб./мес.	Трудоемкость, чел.дни	Сумма, руб.
1	Инженер-программист	70000	3500	98	343000
Общая основная заработная плата исполнителей, $C_{\text{осн}}$				98	343000

4.2.2 Дополнительная заработная плата исполнителей

Дополнительная заработная плата на период разработки ПО рассчитывается относительно основной и составляет 10% от её величины, рассчитывается по формуле 7.

$$C_{\text{доп}} = C_{\text{осн}} \cdot 0,1 = 34300 \text{ (руб.)}. \quad (7)$$

4.2.3 Расчет отчислений на социальное страхование

Отчисления на социальное страхование рассчитываются относительно выплаченной заработной платы по формуле (8). Составляют 30

$$\begin{aligned} C_{\text{соц}} &= (C_{\text{доп}} + C_{\text{осн}}) \cdot 0,3 \\ C_{\text{соц}} &= (34300 + 343000) \cdot 0,3 = 113190 \text{ (руб.)} \end{aligned} \quad (8)$$

На эту статью относят все затраты на магнитные носители данных, бумагу, для печатных устройств, канцтовары и др. Затраты по ним определяются по экспертным оценкам. Расчет расходов на материалы приведен в таблице 4.

Таблица 4 – Расчёт расходов на материалы

№	Материалы	Количество, штуки	Стоимость, рубли
1	Бумага писчая, листов	1000	800
2	Картридж для принтера, шт.	1	1000
3	Другие канцтовары	-	1000
Общая стоимость материалов, C			2800

4.2.4 Накладные расходы

На статью «Накладные расходы» относят расходы, связанные с управлением и организацией работ. Накладные расходы рассчитываются относительно основной заработной платы. Величина накладных расходов принимается равной 60 % от основной зарплаты исполнителей. Формула расчета (9):

$$C_{\text{н}} = C_{\text{осн}} \cdot K, \quad (9)$$

где $C_{\text{н}}$ — накладные расходы;

$C_{\text{осн}}$ — основная заработная плата исполнителей;

K — коэффициент учета накладных расходов.

$$C_{\text{н}} = 343000 \cdot 0,6 = 205800 \text{ (руб.)}$$

4.2.5 Расчет стоимости машинного времени

Затраты на машинное время, необходимое для разработки ПО, расходы на приобретение и подготовку материалов научно-технической информации, расходы на использование средствами связи. Расчет затрат на машинное время осуществляется по формуле (10):

$$C_{\text{маш.вр}} = K_{\text{маш.вр}} \cdot Z_{\text{маш.вр}} \quad (10)$$

где $_{\text{маш.вр}}$ — тарифная стоимость одного часа машинного времени ($_{\text{маш.вр}} = 60 \text{ (руб./час)}$);

$Z_{\text{маш.вр}}$ — машинное время, используемое на проведение работ.

Необходимое количество машинного времени для реализации проекта

по разработке программы рассчитывается по формуле (11):

$$З_{\text{маш.вр}} = t_i \cdot T_{\text{см}} \cdot T_{\text{ср.маш}}, \quad (11)$$

где t_i — трудоемкость работ, чел.дней;

$T_{\text{см}}$ — продолжительность рабочей смены (При пятидневной рабочей неделе $T_{\text{см}} = 8\text{ч.}$);

$T_{\text{ср.маш}}$ — средний коэффициент использования машинного времени ($T_{\text{ср.маш}} = 0,7$).

Из этого следует:

$$З_{\text{маш.вр}} = 98 \cdot 8 \cdot 0,7 = 548,8 \text{ (ч.)}$$

Стоимость машинного времени составит:

$$C_{\text{маш.вр}} = 60 \cdot 548,8 = 32928 \text{ (руб.)}$$

Результаты расчета затрат на проектирование программного обеспечения сведены в таблице 5.

Таблица 5 – Расчёт расходов на материалы

№	Наименование статей	Обозначение	Сумма, руб.	В % к итогу
1	Основная заработная плата	$C_{\text{осн.}}$	343000	46,86
2	Дополнительная заработная плата	$C_{\text{доп.}}$	34300	4,69
3	Отчисления на социальные нужды	$C_{\text{соц.}}$	113190	15,46
4	Материалы	$C_{\text{м.}}$	2800	0,38
5	Стоимость машинного времени	$C_{\text{маш.вр.}}$	32928	4,5
6	Накладные расходы	$C_{\text{н.}}$	205800	28,11
Итого:		$C_{\text{осн.}}$	732018	100

Следовательно, себестоимость разработки составляет 732018 руб.

Данная программа может быть реализована на рынке. При расчетном количестве реализованных программ ($n = 5$) оптовая цена программы ($C_{\text{опт.}}$)

может быть рассчитана по формуле (12):

$$\Pi_{\text{опт}} = \frac{C_{\text{пр}}}{n} + \Pi \quad (12)$$

где $C_{\text{пр}}$ — себестоимость разработки программы;

Π — прибыль, определяется по формуле (13):

$$\Pi_i = Y_p \cdot \frac{C_{\text{пр}}}{n} \cdot 100 \quad (13)$$

где Y_p — средний уровень рентабельность ($Y_p = 20\%$).

Таким образом, оптовая цена программы составит:

$$\Pi_{\text{опт}} = \frac{732018}{5} + \left(\frac{732018}{5} \cdot 0,2 \right) = 146403,6 + 29280,72 = 175684,32 \text{ (р.)}$$

Отпускная цена реализации программы потребителям ($\Pi_{\text{опт}}$), рассчитывается по формуле:

$$\Pi_{\text{опт}} = \Pi_{\text{опт}} + \text{НДС}$$

где НДС — налог на добавленную стоимость, рассчитывается в соответствии с действующей ставкой этого налога — 20% от оптовой цены программы.

$$\Pi_{\text{опт}} = 175684,32 + (175684,32 \cdot 0,2) = 210821,18 \text{ (руб.)}$$

Следовательно, отпускная цена программы составит 210821,18 руб., в том числе НДС — 35136,86 руб.

ЗАКЛЮЧЕНИЕ

TODO

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Бюро находок // столнаходок.рф: информационно-поисковый портал РФ URL: <http://nahodok.ru/> (дата обращения: 01.09.2023).
2. Find my stuff // fstuff.com: мобильное приложение URL: <https://www.fstuff.com/> (дата обращения: 01.09.2023).
3. Lost Property Office // parliament.uk: веб приложение URL: <https://www.parliament.uk/visiting/access/facilities/lost-property/> (дата обращения: 01.09.2023).

ПРИЛОЖЕНИЕ А

Схема базы данных

```
1      model Account {
2          id          String    @id @default(cuid())
3          userId      String
4          type        String
5          provider    String
6          providerAccountId String
7          refresh_token String?
8          access_token String?
9          expires_at  Int?
10         token_type   String?
11         scope        String?
12         id_token     String?
13         session_state String?
14         user         User      @relation(fields: [userId],
15                                     references: [id], onDelete: Cascade)
16
17         @@unique([provider, providerAccountId])
18     }
19
20     model Session {
21         id          String    @id @default(cuid())
22         sessionToken String    @unique
23         userId      String
24         expires     DateTime
25         user        User      @relation(fields: [userId], references:
26                                     [id], onDelete: Cascade)
27
28         @@index([userId], type: Hash)
29     }
30
31     model User {
32         id          String          @id @default(cuid())
33         name        String?
34         nickname    String          @unique
35         socialNetworks UserSocialNetwork[]
36         email       String?         @unique
37         emailVerified DateTime?
38         userInfo    String?         @db.VarChar(280)
39         role        Role            @default(USER)
40         image       String?
41         isBlocked   Boolean         @default(false)
42         blockReason String?
43         accounts    Account[]
```



```

42         sessions          Session []
43         lostAndFoundItems  LostAndFoundItem []
44
45         @@index([ id ], type: Hash)
46         @@index([ nickname ], type: Hash)
47     }
48
49     model VerificationToken {
50         identifier String
51         token      String @unique
52         expires    DateTime
53
54         @@unique([ identifier , token ])
55     }
56
57     model UserSocialNetwork {
58         id                                     String
59                                           @id @default(cuid())
60         socialNetwork                        SocialNetwork
61         link                                String
62         userId                              String
63         user                                User
64                                           @relation(fields: [ userId ],
65                                           references: [ id ], onDelete: Cascade)
66         lostAndFoundItemSocialNetworks LostAndFoundItemSocialNetworks
67         []
68
69         @@unique([ userId , socialNetwork ])
70         @@index([ socialNetwork , userId ])
71     }
72
73     enum Role {
74         USER
75         MODERATOR
76         ADMIN
77     }
78
79     model LostAndFoundItem {
80         id                                     String @id @default(
81                                           cuid())
82         name                                String @db.VarChar
83                                           (100)
84         description                        String @default("")
85                                           @db.VarChar(512)
86         campus                            Campus
87         reason                            PostItemReason

```

```

81         status          LostAndFoundItemStatus          @default(
            ACTIVE)
82         images          String []
83         userId          String
84         user            User                            @relation(
            fields: [userId], references: [id], onDelete: Cascade)
85         socialNetworks  LostAndFoundItemSocialNetworks []
86         created         DateTime                        @default(now()
            )
87         expires         DateTime                        @default(
            dbgenerated("NOW() + interval '1 week'"))
88
89         @@index([id], type: Hash)
90     }
91
92     enum LostAndFoundItemStatus {
93         ACTIVE
94         EXPIRED
95         BLOCKED
96     }
97
98     model LostAndFoundItemSocialNetworks {
99         id                String                        @id @default(cuid())
100        lostAndFoundItemId String
101        lostAndFoundItem   LostAndFoundItem @relation(fields: [
            lostAndFoundItemId], references: [id], onDelete: Cascade)
102        userSocialNetworkId String
103        userSocialNetwork  UserSocialNetwork @relation(fields: [
            userSocialNetworkId], references: [id], onDelete: Cascade)
104
105        @@unique([lostAndFoundItemId, userSocialNetworkId])
106    }
107
108     enum PostItemReason {
109         LOST
110         FOUND
111     }
112
113     enum Campus {
114         V78
115         S20
116         V86
117         MP1
118         SG22
119         SHP23
120         U7

```

```
121     }
122
123     enum SocialNetwork {
124         TELEGRAM
125         VK
126     }
```