

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1. АНАЛИЗ ТРЕБОВАНИЙ	5
1.1. Описание системы тайм-менеджмента	5
1.2. Обзор программ-аналогов	6
1.1.1. Менеджер задач <i>Any Do</i>	6
1.1.2. Менеджер задач <i>Wunderlist</i>	7
1.1.3. Игрофикационная система <i>Epic Win</i>	7
1.3. Функциональная модель разрабатываемой системы	8
1.4. Функциональные требования	9
1.5. Выводы	11
2. ТЕХНИКО-ЭКОНОМИЧЕСКАЯ ЧАСТЬ	12
2.1. Организационная структура проекта	12
2.2. Календарный план проекта	12
2.3. Расчёт затрат на разработку продукта	15
2.3.1. Расчёт заработной платы исполнителей работ по созданию программного продукта	16
2.3.2. Расчёт отчислений на социальные нужды	17
2.3.3. Арендные платежи за производственные помещения	18
2.3.4. Амортизация используемых основных средств и нематериальных активов	19
2.3.5. Расходы на модернизацию и приобретение основных средств	20
2.3.6. Расходы на приобретение необходимого ПО	20
2.3.7. Расходы на интернет и связь	20
2.3.8. Расходы на канцелярские товары и расходные материалы	20
2.3.9. Прочие расходы	20
2.3.10. Расчёт себестоимости программного продукта	20
3. ИССЛЕДОВАТЕЛЬСКАЯ ЧАСТЬ	23
3.1. Цель и задачи исследования	23

3.2.	Описание исследования методик мотивации.....	23
3.3.	Выводы.....	30
4.	РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА.....	31
4.1.	Архитектура программного продукта	31
4.2.	Модель данных.....	32
4.3.	Низкоуровневое проектирование	34
4.4.	Проектирование интерфейса	36
4.5.	Руководство пользователя	37
5.	ЭКСПЕРЕМЕНТАЛЬНАЯ ЧАСТЬ	40
5.1.	Испытания.....	40
5.1.1.	<i>План испытаний.....</i>	40
5.1.2.	<i>Проверка выполнения функциональных требований</i>	40
5.1.3.	<i>Нагрузочное тестирование</i>	42
5.1.4.	<i>Тестирование в исключительных ситуациях</i>	43
5.2.	Вывод	43
6.	ОРГАНИЗАЦИОННАЯ ЧАСТЬ.....	44
6.1.	Постановка проблемы	44
6.2.	Излучение от устройств сотовой связи.....	45
6.2.1.	<i>Стандарты сотовой связи и их биологический смысл.....</i>	46
6.2.2.	<i>Проживание в районе активности сотового излучения.....</i>	53
6.3.	Выводы.....	54
	ЗАКЛЮЧЕНИЕ	55
	СПИСОК ЛИТЕРАТУРЫ.....	56
	ПРИЛОЖЕНИЕ	58

ВВЕДЕНИЕ

В современном мире, живущем на высоких скоростях и загруженном большим количеством информации, экономия времени, удобство и мобильность, являются главными факторами в повседневности занятого человека. Взглянув на разрозненные клочки информации, записанные частично на листах, в компьютерах и телефонах, зачастую невозможно сориентироваться и найти то, что действительно необходимо в данный момент. Одним из самых значимых типов такой информации является список запланированных дел, от которого напрямую зависит продуктивность и успех каждого отдельного человека. Именно поэтому создание менеджера задач является решением проблемы разрозненности данных.

Для этого необходимо рассмотреть предметную область и программы аналоги, поставить **цель** – разработать менеджер задач для платформы iOS с элементами игрофикации. Следовательно, для написания приложения необходимо выполнить следующие **задачи**:

- исследовать предметную область;
- сравнить программы-аналоги;
- сформировать функциональные требования к программе;
- оценить экономические показатели;
- исследовать инструменты повышения мотивации;
- спроектировать мобильное приложение;
- разработать систему;
- протестировать программный продукт и исправить ошибки.

Объектом является процесс планирования и исполнения списка дел.

Предметом является автоматизация процесса планирования и повышение мотивации при использовании системы.

1. АНАЛИЗ ТРЕБОВАНИЙ

1.1. Описание системы тайм-менеджмента

В современном проактивном обществе людям всё чаще приходится сталкиваться с перегруженностью делами и заданиями. Попытка удержать большой объём информации в голове зачастую приводит к стрессу и потере энтузиазма, как следствие снижение мотивации к выполнению поставленных целей. Именно для решения данной проблемы было придумано ведение списка дел.

Персональный информационный менеджер [15] – список дел (чаще компьютерная программа), служащий для облегчения работы с разного рода личной информацией. В простейшем случае это программа, выполняющая функции персонального органайзера. Более сложные программы имеют дополнительные функции, позволяющие вести совместное планирование и организовывать совместную работу над проектами.

Сместив фокус внимания с удержания информации в голове в пользу её анализа и разрешения поставленных задач, можно увеличить свою продуктивность в несколько раз.

Однако, наряду с простым планированием, встаёт вопрос мотивации человека не только во ведении подобного списка, что само по себе минимизирует беспокойность, но и в своевременном выполнении поставленных задач. Причём, основным принципом разработки системы с подобным подходом должен стать – уход от рутины [10], свойственной обычным органайзерам. Основное место отдаётся мотивации, желанию сделать больше и удовольствию от проделанной работы.

Так же, необходимо держать подобную систему всегда под рукой, чтобы с минимальной потерей времени внести новые данные и откорректировать старые. Для этого необходимо разместить автоматизированную систему на мобильном устройстве, которое имеет при себе практически любой современный человек.

1.2. Обзор программ-аналогов

Рассмотрим наиболее распространенные мобильные приложения, платформы iOS схожей тематики, а также разберём их сильные и слабые стороны.

1.1.1. Менеджер задач Any Do

Рассмотрим один из аналогов – Any Do [17]. Минималистичный список дел, позволяющий быстро записать новые задачи и пометить старые как выполненные. Очень прост в освоении и использовании (рис. 1.1).

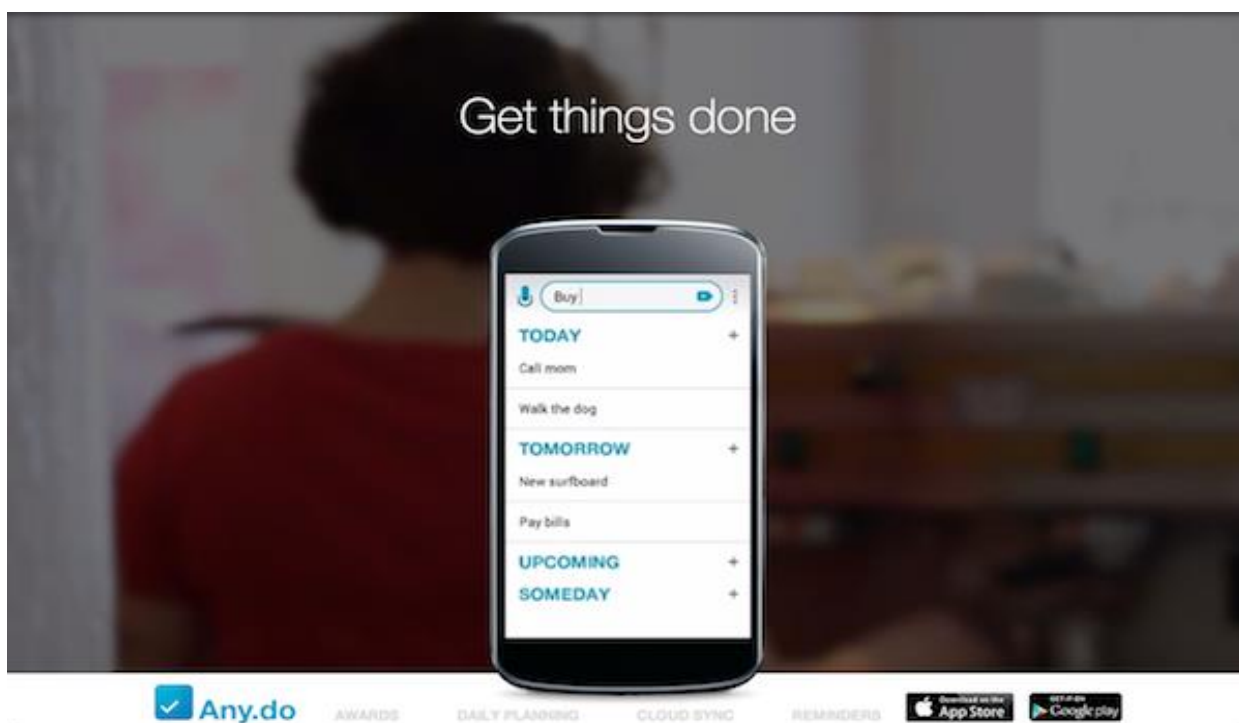


Рис. 1.1. Приложение AnyDo

К основным достоинствам можно отнести:

- экономию времени пользователя;
- минималистичный дизайн.

К недостаткам относятся:

- очень малый набор функций;
- невозможность описать задачу подробнее.

1.1.2. Менеджер задач Wunderlist

Менеджер задач с расширенным набором функций – Wunderlist [19], включает возможность сортировки и делегирования задач. Имеет премиальную подписку для расширения набора базовых функций.



Рис. 1.2. Приложение WunderList

К основным достоинствам можно отнести:

- высокий уровень функциональности;
- продуманный дизайн;
- наличие веб-сервиса для доступа к списку дел с персонального компьютера.

К недостаткам можно отнести:

- платная подписка для расширения функциональности;
- отсутствие мотивационных инструментов.

1.1.3. Игрофикационная система Epic Win

Первая из игрофикационных систем ведения списка дел для платформы iOS [18]. Главной особенностью является попытка мотивировать пользователя к продуктивности и выполнению поставленных задач.

В данном приложении используются механизмы обратной связи и углубленной статистики. Однако, приложение имеет непродуманную систему рейтингов и узконаправленный дизайн.



Рис. 1.3. Приложение Epic Win

Достоинство данной системы – мотивационный принцип подхода к использованию.

К недостаткам можно отнести:

- базовый набор функций;
- дизайн, направленный на игровую аудиторию;
- высокая стоимость системы.

1.3. Функциональная модель разрабатываемой системы

В данном разделе описаны функциональные требования разрабатываемой системы на основе технического задания. На рисунке 1.4

изображено поле системы с включениями функций [7], а так же пользователь приложения.

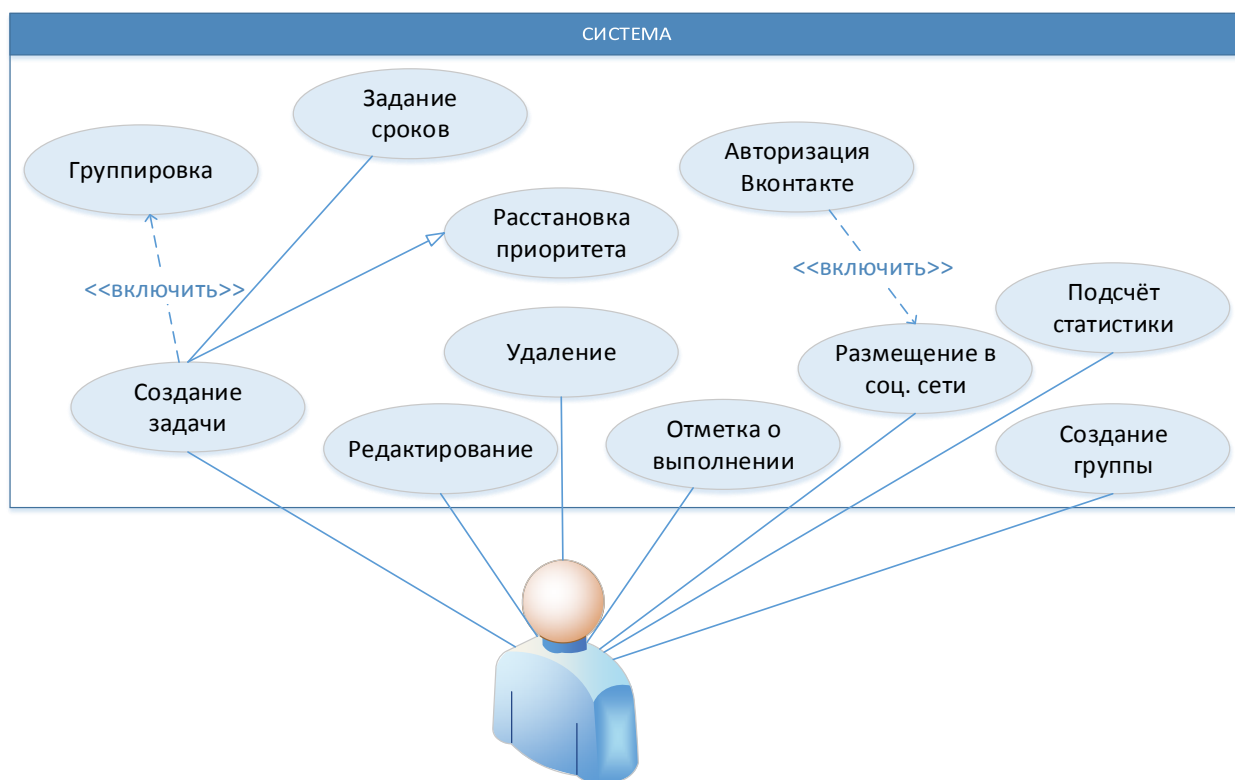


Рис. 1.4. Диаграмма вариантов использования

Отдельно необходимо описать процедуру размещения сообщения в социальной сети. Так, задав необходимую задачу, у пользователя есть возможность сделать её гласной, для этого, нажав на соответствующую кнопку, появляется окно авторизации и сообщение для отправки, после чего необходимо явно подтвердить действие пользователя.

Расстановка приоритетов осуществляется по системе «Матрица управления временем». Так, самые главные задачи займут позицию важное/срочное и будут иметь высокий приоритет к выполнению, остальные, согласно методологии, будут иметь градацию по убыванию.

1.4. Функциональные требования

Ниже описаны функциональные требования для мобильного приложения, представлены основные требования к программе, учтены потенциальные пользователи системы.

Мобильное приложение призвано расширить привычный функционал списка дел, путём добавления мотивационных методик, более глубоких принципов ведения статистики и игрофикационных инструментов в привычный набор функций.

Запустив приложение, необходимо попасть в окно приветствия. После чего, необходимо отобразить список всех категорий, при нажатии на которые отображается список с задачами данной категории. Добавление групп необходимо реализовать по нажатию символа “+”. После создания группы необходимо реализовать возможность создания в ней задач. Создание задач сопровождается заданием типа задачи и сроком выполнения. Завершение задачи должно происходить при установке переключателя в положение выполнено, после чего задача тускнеет и может быть удалена проведением пальца справа-налево.

Игрофикационные инструменты необходимо реализовать следующим образом, сперва ввести четыре типа задач:

- Важно / срочно (ВС).
- Важно / не срочно (ВН).
- Неважно / срочно (НС).
- Неважно / не срочно (НН).

На основании данных категорий будут введены оценки значимости задач пользователя. Внизу экрана необходимо реализовать блок мотивационных сообщений и статистики. При нажатии на который необходимо реализовать вывод статистики по всем типам задач. Так же в программе необходимо предусмотреть возможность рассказать о взятой задаче в социальной сети «вконтакте».

За выполнение определённых действий, в окне наград необходимо вести учёт списка достижений пользователя. Данное окно должно быть доступно для перехода из основного окна приложения.

1.5. Выводы

Рассмотрев предметную область, и, приняв во внимание все недостатки программ аналогов, была поставлена **цель** – разработать менеджер задач для платформы iOS с элементами игрофикации. Такая необходимость обусловлена изменением парадигмы восприятия проблемы списка дел в сторону вовлечения и контроля процесса. Следовательно, для написания приложения необходимо выполнить следующие **задачи**:

- исследовать предметную область;
- сравнить программы-аналоги;
- сформировать функциональные требования к программе;
- оценить экономические показатели;
- исследовать инструменты повышения мотивации;
- спроектировать мобильное приложение;
- разработать систему;
- протестировать программный продукт и исправить ошибки.

Объектом является процесс планирования и исполнения списка дел.

Предметом является автоматизация процесса планирования и повышение мотивации при использовании системы.

2. ТЕХНИКО-ЭКОНОМИЧЕСКАЯ ЧАСТЬ

Перед началом разработки любой программной системы необходимо выяснить, насколько целесообразна ее разработка, как с точки зрения полезности, так и с точки зрения экономической эффективности. В технико-экономической части работы рассматриваются вопросы организации работ по созданию и внедрению программной системы, а также приводится расчёт ее себестоимости.

2.1. Организационная структура проекта

Организационная структура проекта (OBS) приведена на рис. 2.1.

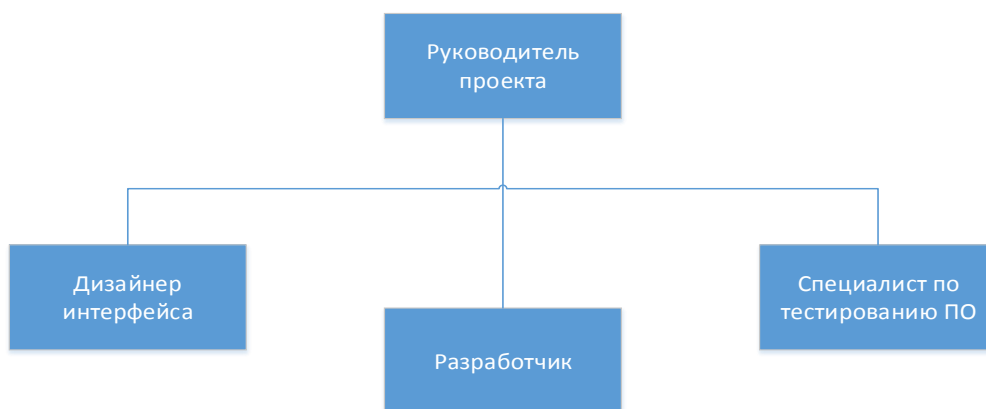


Рис. 2.1. Организационная структура проекта

2.2. Календарный план проекта

Для оценки расходов на реализацию проекта в числе прочих необходимо определить временные затраты на его реализацию [26]. Для определения временных затрат проекта необходимо разработать календарный план проекта. Реализуемый проект является типовым для Компании, исходя из этого, был сформирован состав работ проекта, определена их длительность, а также распределение ресурсов по ним. При разработке календарного плана были учтены ограничения и допущения, накладываемые на проект Заказчиком.

Для реализации проекта необходимо выполнить следующие работы:

- 1) сбор требований Заказчика к разрабатываемому ПО;
- 2) разработка и согласование технического задания;
- 3) разработка технического проекта;
- 4) разработка ПО;
- 5) проектирование интерфейса;
- 6) разработка пользовательской документации;
- 7) тестирование ПО.

Распределение человеческих ресурсов по работам проекта и степень их загрузки приведены в табл. 2.1.

Таблица 2.1

Структура общего времени на создание программного продукта

№	Этап работ	Ответственные исполнители (занятость на этапе)	Дней
1	Сбор требований Заказчика к разрабатываемому ПО	Руководитель проекта [100%] Разработки [40%]	2
2	Разработка и согласование технического задания	Руководитель проекта [100%]	5
3	Разработка технического проекта	Руководитель проекта [100%] Разработчик [40%]	10
4	Разработка ПО	Разработчик [100%]	30
5	Проектирование интерфейса	Дизайнер интерфейсов [100%]	15
5	Разработка пользовательской документации	Руководитель проекта [100%]	10

6	Тестирование ПО	Специалист по тестированию ПО [100%]	15
---	-----------------	---	----

При реализации данного проекта работы выполняются последовательно и параллельно. Диаграмма Ганта [26] приведена в таблице 2.2 и на рисунке 2.2.

Таблица 2.2

Список задач к выполнению

Название задачи	Длительность	Начало	Окончание	Названия ресурсов
Сбор требований Заказчика к разрабатываемому ПО	2 дней	Пн 03.03.14	Вт 04.03.14	Руководитель проекта[100%]; Разработчик[40%]
Разработка и согласование технического задания	5 дней	Ср 05.03.14	Вт 11.03.14	Руководитель проекта[100%]
Разработка технического проекта	10 дней	Ср 12.03.14	Вт 25.03.14	Руководитель проекта[100%]; Разработчик[40%]
Разработка ПО	30 дней	Ср 26.03.14	Вт 06.05.14	Разработчик[100%]
Проектирование интерфейса	15 дней	Ср 16.04.14	Вт 06.05.14	Дизайнер интерфейса[100%]
Разработка пользовательской документации	10 дней	Ср 07.05.14	Вт 20.05.14	Руководитель проекта[100%]
Тестирование ПО	15 дней	Ср 07.05.14	Вт 27.05.14	Специалист по тестированию ПО[100%]

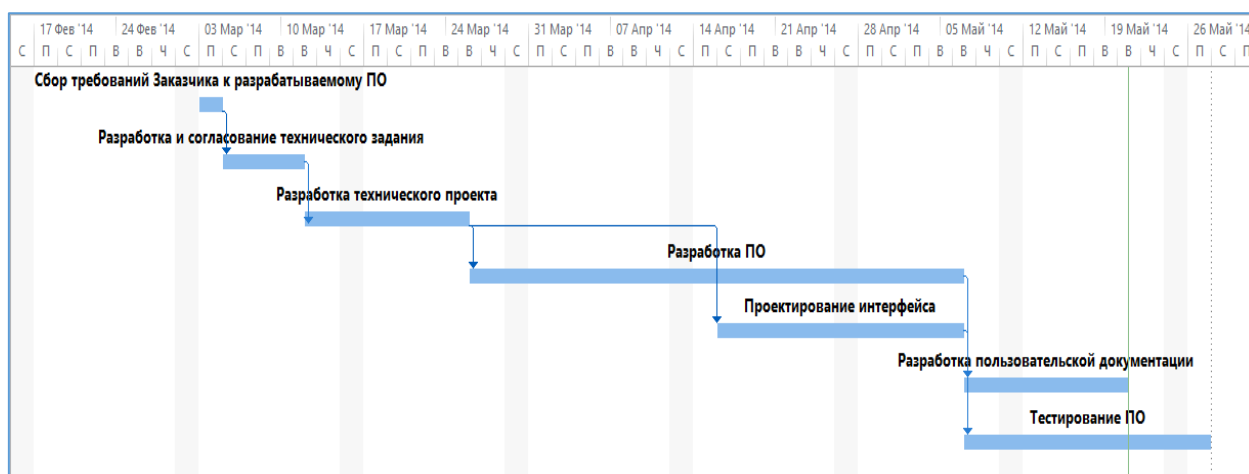


Рис. 2.2. Графическое представление Диаграммы Ганта

Исходя из длительности работ и коэффициента загрузки членов проектной команды, определим их трудозатраты при реализации проекта (табл. 2.3).

Таблица 2.3

Трудозатраты членов проектной команды

№	Исполнитель	Трудозатраты, человеко-часов
1	Руководитель проекта	131,2
2	Разработчик	278,4
3	Дизайнер интерфейса	120
6	Специалист по тестированию ПО	120

2.3. Расчёт затрат на разработку продукта

Расчет затрат на создание и внедрение ПО включает следующие составляющие с последующим их графическим представлением в виде круговой диаграммы [26]:

- заработная плата исполнителей работ по проекту – $ЗП_{осн}$;
- дополнительная заработная плата $ЗП_{дон}$;
- заработная плата обслуживающего и административного персонала;

- отчисления на социальные нужды (страховые взносы) – $H_{зн}$;
- арендные платежи за производственные помещения – $A_{нм}$;
- амортизация используемых основных средств и активов – A ;
- расходы на модернизацию и приобретение основных средств – $P_{мод}$;
- расходы на приобретение необходимого ПО – $P_{ПО}$;
- расходы на интернет, связь – $P_{тел}$;
- расходы на канцелярские товары и расходные материалы – $P_{р.м.}$;
- прочие расходы – $P_{р.р.}$.

2.3.1. Расчёт заработной платы исполнителей работ по созданию программного продукта

Основная ЗП определяется по формуле [26]:

$$ЗП_{осн} = \frac{M \cdot T}{Ч_p \cdot t_{p.д.}} \left(1 + \frac{П}{100} \right), \text{ руб.}$$

- M – месячная зарплата (руб.);
- T – общие трудозатраты (чел.-ч);
- $Ч_p$ – число рабочих дней в месяц;
- $t_{p.д.}$ – продолжительность рабочего дня в часах;
- $П$ – процент премии.

В данной работе $Ч_p = 21 \text{ день}$, $t_{p.д.} = 8 \text{ ч}$, $П = 0$.

Значение месячной заработной платы (M), суммарные трудозатраты членов, а также рассчитанная по формуле основная заработная плата проектной команды приведены в табл. 2.4.

Таблица 2.4

Основная заработная плата членов проектной команды

№	Исполнитель	Месячная заработная плата (М), руб.	Трудозатраты, человеко- часов	ЗП _{осн} , руб.
1	Руководитель проекта	50 000	131,2	41 000
2	Разработчик	30 000	278,4	52 200
3	Дизайнер интерфейса	30 000	120	22 500
6	Специалист по тестированию ПО	20 000	120	15 000

Суммарное значение основной заработной платы проектной команды на период реализации проекта составит 130 700 (руб.).

Дополнительная заработная плата берется в размере 15% от основной.

$$\text{ЗП}_{\text{доп}} = 19\,605 \text{ (руб.)}.$$

Общая заработная плата составит 150 305 (руб.).

Проект реализуется в небольшой ИТ-компании, где доля вспомогательного и административного персонала по отношению к основному персоналу не велика. Большая часть административного персонала задействована в проектной деятельности в качестве руководителей проекта. Кадровый учет, бухгалтерский и налоговый учет в Компании отдан на аутсорсинг. Затраты на аутсорсинг войдут в прочие расходы. В связи с этим прием заработной платы обслуживающего персонала равной 0 руб.

2.3.2. Расчёт отчислений на социальные нужды (страховые взносы)

Теперь можно рассчитать величину отчислений на социальные нужды (страховые взносы), которые начисляются на заработную плату и в 2014 г. для организаций, осуществляющих деятельность в области информационных технологий, составляют 14% по выплатам в пределах 565 тыс. руб. Структура отчислений на социальные нужды (страховые взносы) приведена в таблице 2.5.

Таблица 2.5

Структура отчислений на социальные нужды (страховые взносы)

Пенсионный фонд Российской Федерации	8,0%
<i>для лиц 1967 года рождения и моложе</i>	
<i>страховые взносы на страховую часть трудовой пенсии</i>	2,0%
<i>страховые взносы на накопительную часть трудовой пенсии</i>	6,0%
Фонд социального страхования Российской Федерации	2,0%
Федеральный фонд обязательного медицинского страхования	4,0%

Таким образом, $H_{зн} = 21\,042,7$ (руб.).

2.3.3. Арендные платежи за производственные (офисные) помещения

Компания, реализующая проект по разработке мобильного приложения списка дел, арендует офисные помещения в г. Брянск. Стоимость аренды составляет 6 000р. в месяц за 30 м².

Арендная плата включает в себя оплату как площади занимаемых компанией помещений, так и электроэнергии, водоснабжения, интернета, телефонии и парковочных мест на автостоянке.

На каждого члена проектной команды приходится 7,5 м² арендуемого офисного помещения. На период данного проекта члены проектной команды в других проектах не задействованы.

Исходя из изложенного выше, затраты на аренду помещений, с периода – марта 2014 по май 2014 (включительно), составят:

$$A_{nm} = 18\,000 \text{ (руб.)}.$$

2.3.4. Амортизация используемых основных средств и нематериальных активов

При реализации проекта по разработке мобильного приложения задействованы следующие основные средства:

- 1 персональных компьютеров Mac-mini стоимостью 24 890 (руб.);
- монитор BenQ 21.5” стоимостью 4 850 (руб.);
- magic Mouse – 3 190 (руб.);
- apple Wireless Keyboard – 2 990 (руб.);
- 3 ноутбука и 3 компьютерных мыши первоначальной стоимостью 15 500 (руб.) каждый.

Срок полезного использования для задействованных в проекте основных средств определен в 3 года.

Метод начисления амортизации – линейный.

Амортизационные отчисления для персонального компьютера на 1 месяц и периферии составят:

$$(24\,890 + 4\,850 + 2\,990) / 36 = 909,17 \text{ (руб.)}.$$

Амортизационные отчисления для ноутбука на 1 месяц составят:

$$15\,500 / 36 = 430,56 \text{ (руб.)}.$$

Амортизационные отчисления по ОС, относящиеся на проект составят:

$$A_{oc} = 5 \cdot 909,17 + 3 \cdot 5 \cdot 403,56 = 10\,599,25 \text{ (руб.)}.$$

В качестве ОС используется свободно распространяемое ПО MacOSX Mavericks. Среда разработки xCode так же распространяется бесплатно [21].

Амортизационные отчисления по РБП, относящиеся на проект составят:

$$A_{РБП} = 0 \text{ (руб.)}.$$

Суммарные амортизационные отчисления составят:

$$A = 10\,599,25 \text{ руб.}$$

2.3.5. Расходы на модернизацию и приобретение основных средств

При реализации проекта по мобильного приложения менеджера задач не планируется приобретение новых и модернизация существующих основных средств.

2.3.6. Расходы на приобретение необходимого ПО

При реализации проекта не планируется приобретение ПО.

2.3.7. Расходы на интернет и связь

Так как в Компании, реализующей проект не производится биллинг и тарификация телекоммуникационных услуг в разрезе сотрудников, затраты на интернет и связь войдут в прочие затраты, рассчитываемые как процент от прямых затрат.

2.3.8. Расходы на канцелярские товары и расходные материалы

Затраты на расходные материалы берутся по факту и составляют $P_{р.м.} = 2\,500$ (руб.).

2.3.9. Прочие расходы

Прочие расходы составляют 30% от суммы следующих элементов структуры затрат [26]: $З_{Посн}$, $З_{Пдоп}$, $Н_{зп}$, $А_{пм}$, $А$, $Р_{мод}$, $Р_{ПО}$, $Р_{тел}$ и $Р_{р.м.}$.

$$P_{р.р.} = 0.3(З_{Посн} + З_{Пдоп} + Н_{зп} + А_{пм} + А + Р_{мод} + Р_{ПО} + Р_{тел} + Р_{р.м.})$$

Таким образом, $Pr.p. = 60\,734,09$ (руб.).

2.3.10. Расчёт себестоимости программного продукта

В себестоимость программного продукта входят следующие элементы: $З_{Посн}$, $З_{Пдоп}$, $Н_{зп}$, $А_{пм}$, $А$, $Р_{мод}$, $Р_{ПО}$, $Р_{тел}$, $Р_{р.м.}$ и $Pr.p.$.

Сложив все элементы, можно определить себестоимость программного продукта и услуг по его внедрению: Сп.п.= 263 181,04 (руб.).

Структура себестоимости программного продукта отражена на рисунке 2.3 и представлена в таблице 2.6.

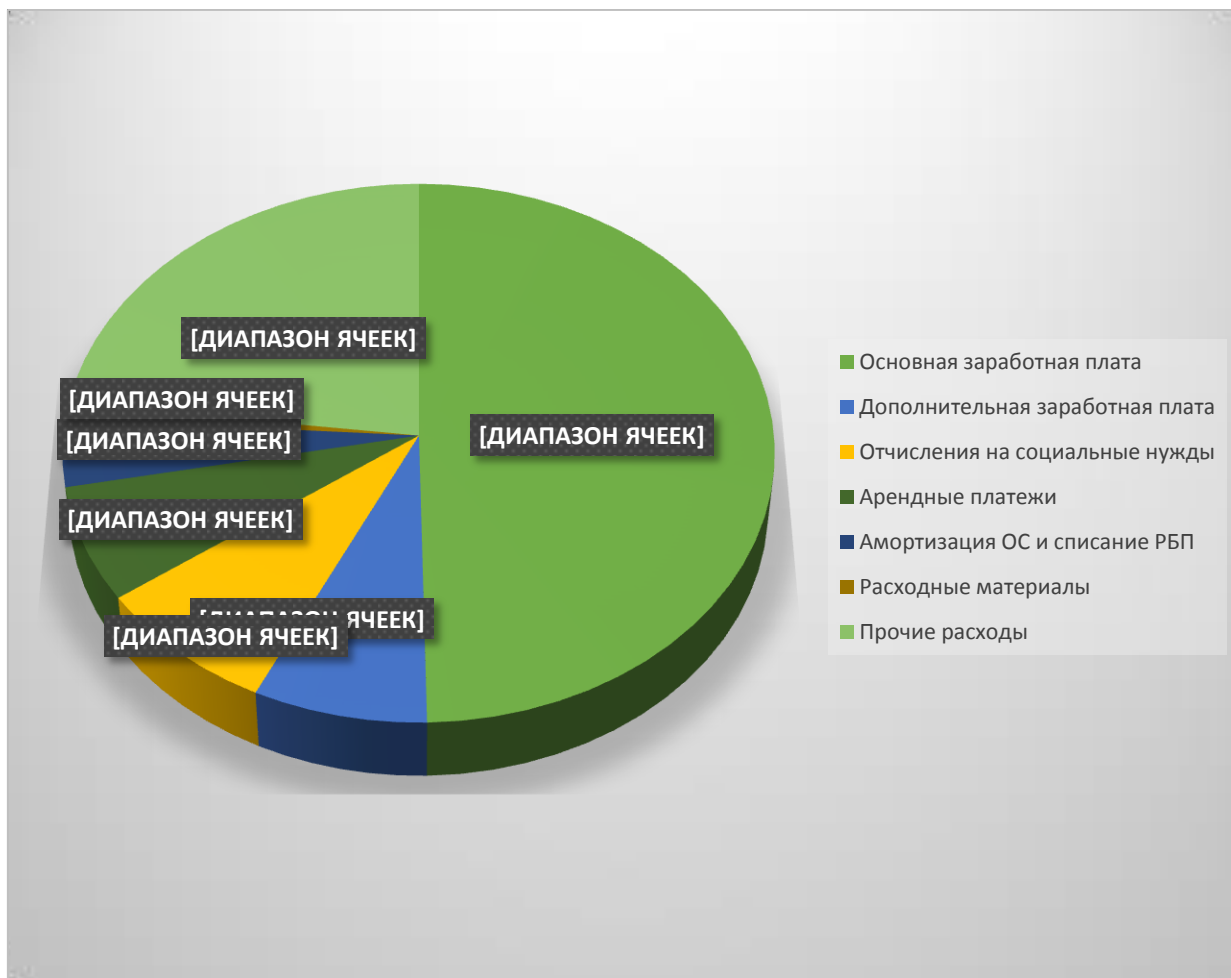


Рис. 2.3. Структура себестоимости программного продукта

Таблица 2.6

Структура себестоимости программного продукта

№	Элементы себестоимости	Сумма (руб.)	% в общ. сумме себестоимости
1	Основная заработная плата исполнителя	130 700	49,66
2	Дополнительная заработная плата исполнителя	19 605	7,45
3	Отчисления на социальные нужды (страховые взносы)	21 042,70	8

4	Арендные платежи за производственные (офисные) помещения	18 000	6,84
5	Амортизация используемых основных средств и нематериальных активов	10 599,25	4,03
6	Расходы на модернизацию и приобретение основных средств	—	—
7	Расходы на приобретение необходимого ПО	—	—
8	Расходы на интернет, связь	—	—
9	Расходы на канцелярские товары и расходные материалы	2 500	0,95
10	Прочие расходы	60 734,09	23,08
Итого:		263 181,04	100

3. ИССЛЕДОВАТЕЛЬСКАЯ ЧАСТЬ

3.1. Цель и задачи исследования

Целью данного исследования является анализ инструментов повышения мотивации пользователя на взаимодействие с разрабатываемой системой, увеличение его реальной продуктивности и привнесение в программный продукт работающих методологий тайм-менеджмента.

Задачи данного исследования:

- исследовать инструменты мотивации;
- исследовать инструменты тайм-менеджмента;
- выбрать более гибкий и результативный метод ведения списка дел и вовлечения пользователя в систему;
- адаптировать выбранные инструменты к разрабатываемому программному продукту.

3.2. Описание исследования методик мотивации

Проблема мотивации в современном обществе стоит особенно остро, как на уровне создания продуктов в IT-сфере, так и на уровне их потребления конечным пользователем. Чем больше представителей нового поколения приходят на рынок информационных технологий, тем более неординарным должен быть подход по их удержанию и вовлечению на всех уровнях потребления. Так называемое «поколение миллениума» [10] уже не поддаётся стандартным приёмам управления фокусом внимания, всё больше отвлекающих факторов конкурентной среды и представителей вне вашей плоскости конкуренции, перехватывают внимание на себя. Так же с развитием современных развлекательных средств, всё больше представителей и более взрослой целевой аудитории отказываются «играть» по старым правилам. Стоит добавить сюда природное желание минимизировать нагрузку или отказаться от неё вовсе, и вот проблема уже

выходит за рамки управляемости. В таких условиях главной задачей становится необходимость добиться не просто поведенческой лояльности в любом направлении в рамках сферы ИТ, но и вывести её на новый уровень потребительской активности.

Инструменты менеджмента доступны представителям управляющих должностей уже довольно давно, однако, чем дольше они используются тем более низкоэффективными становятся, в конце концов компания приходит в положение застоя, после которого экстренно вводится ориентация на инновации. Однако, чувство удовлетворённости остаётся за границей планирования. Особенно остро этот вопрос стоит в отношении искушённых представителей «молодого ИТ» [10]. Люди, привыкшие к одобрению своих действий, подтверждению эффективности, не видят своей значимости без инструментов вроде грамотно выстроенной обратной связи и возможности заявить о своих успехах. Если копнуть глубже, на этом чувстве уже давно играют многие ведущие представители социальных сфер, ставя коммуникацию и признание заслуг «во главу угла». «Классическому менеджменту» для выхода за пределы своего поля влияния не хватает сравнительно свежей интерпретации управления вовлечением – игрофикации.

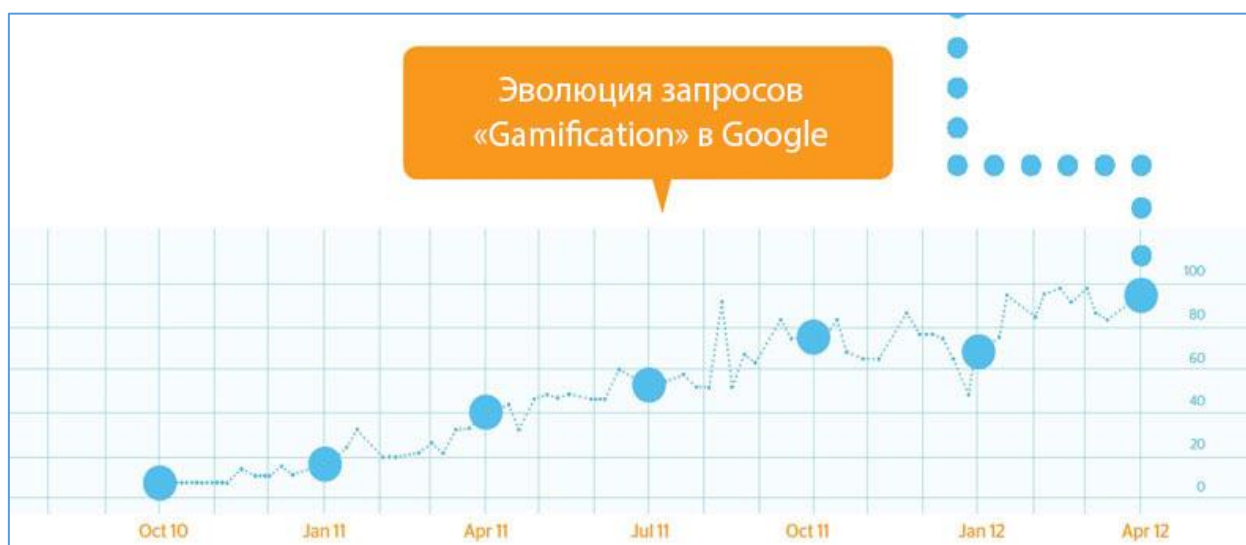


Рис. 3.1. График увеличения запросов по слову «Игрофикация»

Игрофикация включает в себя те самые недостающие «частички пазла», способствующие полному изменению парадигмы взглядов на решение проблемы. Ориентация на клиента, способствует увеличению продуктивности и вовлечения [13]. Грамотно выстроив систему, можно с минимальными изменениями удержать внимание участника на долгое время. Более того, создание ощущения личной победы – лучший мотиватор из всех, которые доступны для использования. Игрофикация – это использование игровых элементов в неигровых задачах [1]. Говоря иначе – привнесение в обычную рутину удовлетворения от выполняемых действий. Сделать это можно при помощи как простых элементов, так и более глубоких (игрофикация 2.0), влекущих за собой возможный пересмотр всей концепции проекта.

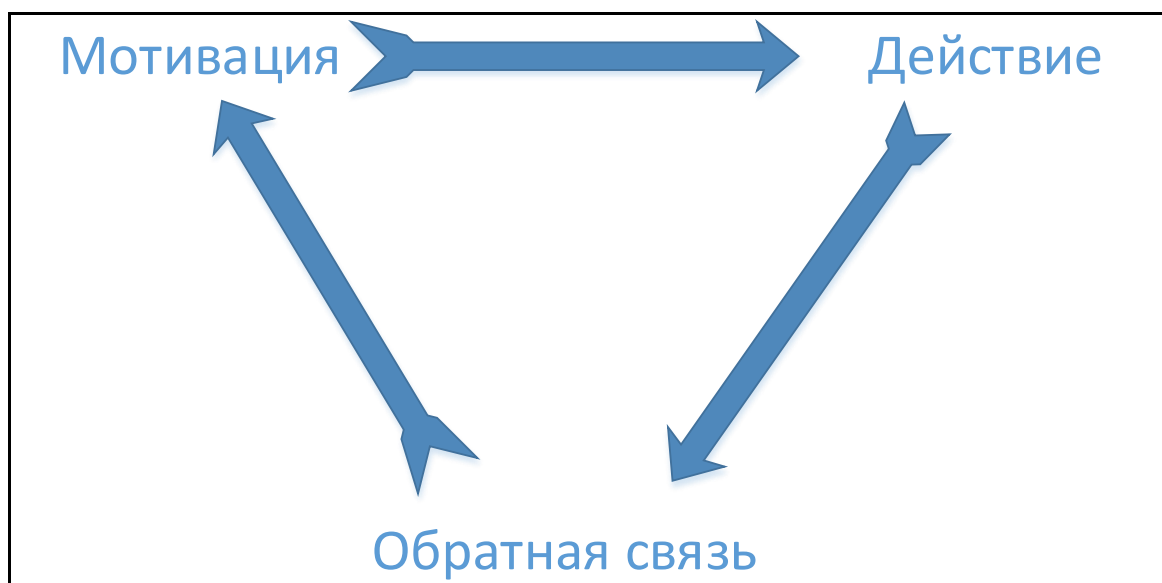


Рис. 3.2. Обратная связь в «Кольце вовлечения»

Постановка проблемы позволила по-иному взглянуть на всю концепцию подхода к разработке ПО, было решено применить методику в системе тайм-менеджмента, на основании списка задач (To-Do List). Анализ предметной области выявил несоответствие конечной цели похожих систем с реальным положением дел. Плоскость интересов разработчика списка дел,

лежит в удовлетворении пользователем потребности в хранении и систематизации заметок, и задач [15]. Однако, упускается один из основополагающих фактором успеха подобных проектов – мотивация к выполнению. В проектировании программ подобной направленности, необходимо кардинально сменить парадигму взглядов в сторону вовлечения представителя целевой аудитории к выполнению поставленных целей, а не просто их фиксации. Именно для решения подобной, более усложнённой, в сравнении с оригинальным списком дел, задачи, и предстояло справиться при помощи игрофикационных методик и инструментов.

За основу проектирования берутся самые функциональные подходы вовлечения:

- **Обратная связь** [10] – один из самых важных элементов игрофикации, именно он позволяет отслеживать в реальном времени эффект от деятельности и, чем быстрее будет получен отзыв системы, тем больший фокус внимания гарантирован проекту. Реализация данного подхода осуществляется при помощи индикаторов активности. С одной стороны – это статистический маячок для клиента, с другой – весьма явный показатель выгоды от активных действий.



Рис. 3.3. Знаки достижений

- **Поощрение** [10] – своеобразная благодарность за действия внутри системы. Для клиента программного обеспечения это могут быть очки, бейджи, звания, нематериальная выгода в виде особого статуса и так далее.

В данном проекте реализована развёрнутая система достижений, за определённый набор действий внутри системы. В последствии подобный механизм может быть легко дополнен и переработан на любой стадии улучшения, что позволит концепции постоянно удерживать интерес клиента.

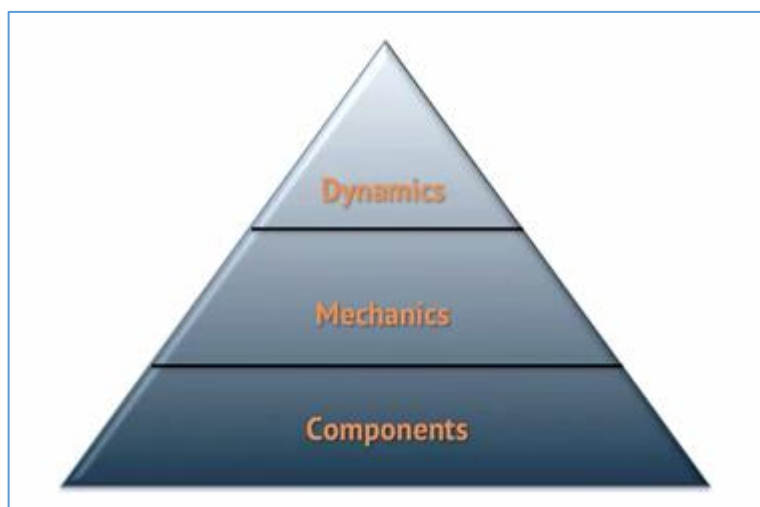


Рис. 3.4. Пирамида объектов игрофикации

- **Соревновательный элемент** [10] – выстроить систему обратной связи и поощрений бывает не всегда достаточно, для таких ситуаций можно сыграть на азарте клиента. Создание условия, при котором соревнование будет не только выгодным, но и желанным, немаловажный фактор успеха системы. Добиться подобного уровня лояльности к проекту можно используя элемент на базе известных социальных систем или специализированных интернет ресурсов [13].

- Ну и самое главное в этой системе – **элемент веселья**. Именно он отличает рутину от вовлечения [1]. Без эмоциональной разгрузки, механизмы мотивации – всего лишь средства манипуляции и не будут иметь долгосрочного эффекта. Стоит заметить, что веселье не всегда стоит понимать «в лоб», это скорее нестандартный подход к стандартным

проблемам. Для примера рассмотрим дополнение к MSOffice под названием RibbonHero [11].

Эта система позволяет добавить в привычное использование офисным пакетом, элементы игрофикации. За выполнение действий над документах, пользователь получает очки и достижения, а элемент веселья построен вокруг истории скрепки, путешествующей во времени. Такой необычный подход позволяет разгрузить человека, отвлечь от рутинного набора текста, в то же время – в систему встроен принцип поощрения улучшению знаний «офисного пакета», что вместе с приятным оформлением, является очень мощным инструментом мотивации и удержания фокуса внимания. Внимание к мелочам и эффект неожиданности способствуют заинтересованности пользователя системы. В программном продукте списка дел такой подход основан на поощрительных сообщениях и скрытых достижениях, получение которых не очевидно, неожиданно и приятно для клиента.



Рис. 3.5. Игрофикационное решение Microsoft Labs

В приложение тайм-менеджмента на основе игрофикационных подходов, были использованы элементы обратной связи с пользователем, на примере статистики выполнения задач за сутки, чтобы дать человеку наглядно понять, как близок он к «идеальному дню».

Все задачи поделены по известной системе, описанной в книге Стивена Р. Кови «7 навыков высокоэффективных людей» [12] – «Матрица управления временем», что позволило распределить “веса” задач по значимости и выстроить грамотную градацию оценок их выполнения (табл. 3.1).

Таблица 3.1

Матрица управления временем

	Срочно	Не срочно
Важно	<i>Неотложные задачи</i>	<i>Планирование</i>
Неважно	<i>Задачи меньшей важности</i>	<i>Задачи низшего приоритета</i>

На уровне поощрений разработана система бейджей за выполнение различного рода условий постановки целей и их выполнения. Помимо этого, выполнение задачи сопровождается мотивационным сообщением для явного акцентирования заслуги пользователя. Так же одним из самых важным принципов постановки цели является её гласность [3]. Она позволяет удерживать человека от отказа выполнения. Этот аспект так же был учтён, и в систему введен механизм, с помощью которого можно поделиться своими начинаниями с друзьями через социальные сети, подобное действие так же являются одним из условий получения бейджа внутри системы.

По данным Gartner Group, к 2015 году 70% крупнейших мировых компаний будут применять в своих решениях игровой элемент [10]. На Российском рынке это весьма новая тенденция, однако, она развивается столь

стремительно, что, если не взять её на рассмотрение сейчас, можно безнадежно отстать от конкурентов в будущем. Подобный подход прекрасно ложится в основу внутренних бизнес процессов, равно, как и во внешних проектных решениях, для максимального привлечения клиентов, удержания внимания, увеличения продуктивности.

3.3. Выводы

В результате исследования методов мотивации пользователя, а так же методологии управление временем, было решено использовать игрофикационный подход к вовлечению внутри системы на примере следующих инструментов:

- поощрительных сообщений;
- системы достижений;
- развёрнутой статистики и обратной связи;
- социализации.

Так же было решено использовать гибкую и в тоже время эффективную систему распределения важности задач – матрицу управления временем, для более грамотного управления «весом» задачи.

На примере поставленных задач – организации мотивационных подходов к ведению списка дел, применение методологии игрофикационных механизмов, способствует решению проблем, обозначенных в вводной части, увеличению клиентской удовлетворённости программным продуктом, а так же повышению поведенческой лояльности к системе.

4. РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА

4.1. Архитектура программного продукта

Архитектурное проектирование – фаза «принятия решения». Цель этой фазы – определить совокупность компонент программного изделия и их интерфейсы, чтобы дать каркас для последующей разработки программного изделия [4]. Архитектурный проект должен охватывать все требования, сформулированные на предыдущей фазе – системный анализ.

Разрабатываемое приложения носит «монолитный» характер, что значит – даёт весь набор необходимых инструментов и функций для использования системы без непосредственного подключения к сети Интернет. Единственная функция, недоступная с таким типом использования – размещение сообщений в социальных сетях.

На рисунке 4.1 приведена архитектурная схема приложения. На ней отображены все модули, который включает текущий проект, а также те элементы управления, которые используются в этом модуле.

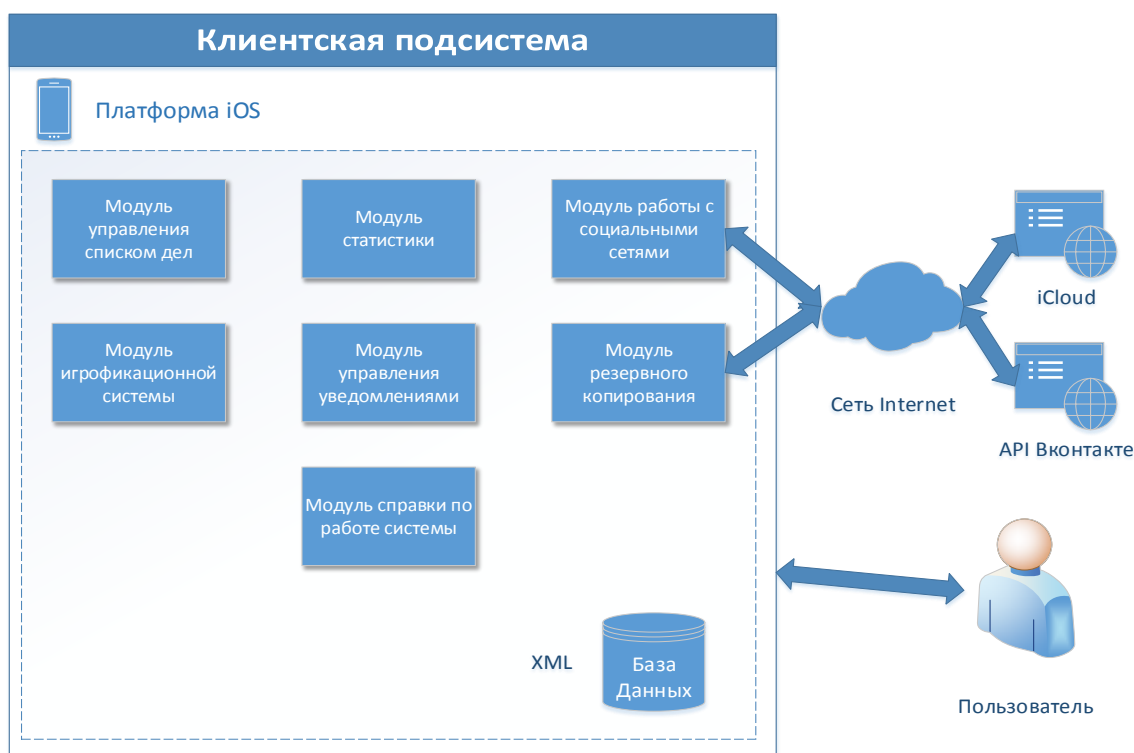


Рис. 4.1. Модули программного продукта

Далее рассмотрим все входящие в каждый модуль функции, описанные в виде набора «Модуль-Функция» в таблице 4.1.

Таблица 4.1

Модули мобильного приложения

Модуль	Функции
Модуль управления списком дел	Добавление, редактирование, удаление задач и групп
Модуль статистики	Подсчёт выполненных и предстоящих задач
Модуль игрофикационной системы	Поощрительно-мотивационная подсистема
Модуль работы с социальными сетями	Авторизация и размещение сообщений в социальной сети
Модуль управления уведомлениями	Задание сигналов, всплывающих окон и стикеров

Для разработки системы была выбрана объектно-ориентированная среда разработки от фирмы Apple – xCode. Являясь предпочтительной для платформодержателя, она предоставляет высокое качество и удобные инструменты программирования [16]. Среда содержит модуль тестирования, что облегчает прохождение порога одобрения программного обеспечения для внутреннего магазина Apple. К тому же, среда распространяется совершенно бесплатно [21].

4.2. Модель данных

При разработке проектного решения, встал вопрос о необходимости хранения определённого набора данных для корректного ведения учета действий внутри приложения и формирования статистических отчётов. В конечный список попали следующие данные, описывающие различные типы информации внутри разрабатываемой системы:

- данные о задачах;
- данные о группах;
- статистические данные;

- учет задач с пометкой важности;
- список достижений;
- сообщения от системы.

На основании рассмотренных требований к хранению информации, была спроектирована база данных [23]. Выделены объекты и установлены их атрибуты. База данных приведена в третью нормальную форму.

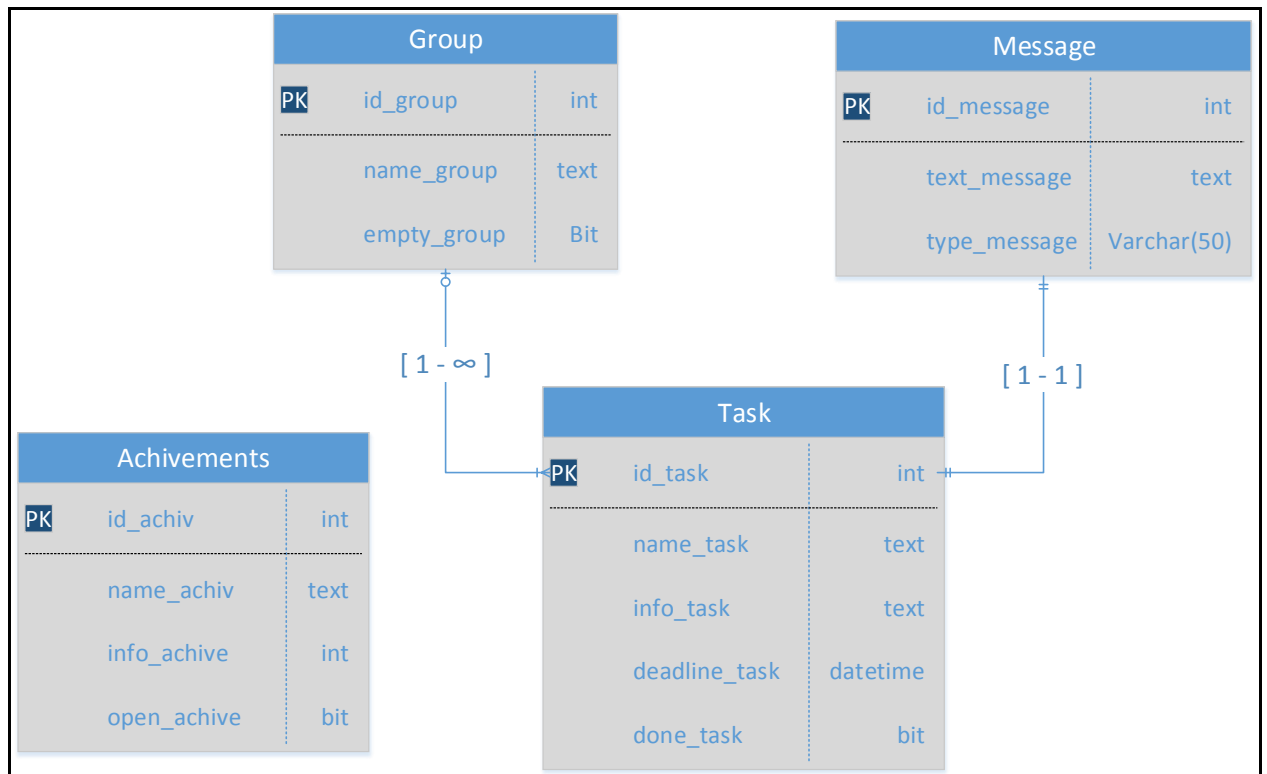


Рис. 4.2. Модель базы данных

База данных мобильного приложения «ToDoPlay» состоит из 5 таблиц, три из которых связаны между собой.

Таблица «Group» служит для хранения основной информации о группе задач и флаг о том, пуста ли она в данный момент.

Таблица «Task» служит для хранения идентификатора задачи, её названия, описания. Так же в данной таблице указаны сроки выполнения и флаг активности.

Таблица «Achivements» хранит в себе информацию полученных достижений, их названиях и описании для получения.

Таблица «Messanges» – список сообщений разного рода, от хвалебных, до системных. Содержит текст сообщения и его тип, для разграничения в зависимости от надобности появления.

4.3. Низкоуровневое проектирование

В данном разделе рассматривается проектирование программного продукта на основании диаграммы классов UML [14] (рис. 4.3).

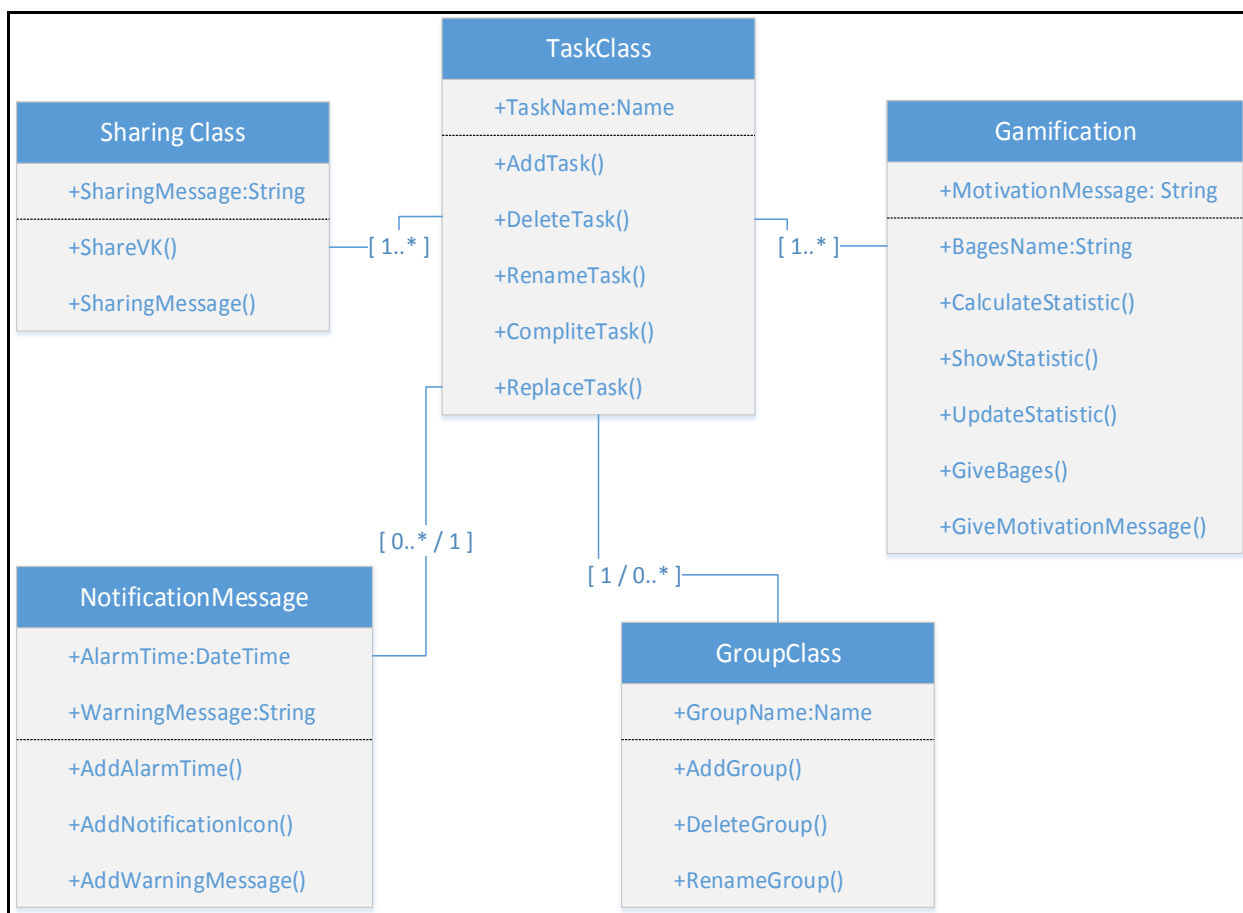


Рис. 4.3. Диаграмма классов

В диаграмме представлены классы, реализованные в конечной системе.

Ниже описаны основные из них:

- **Task Class** – класс задач, включающий функции добавления, удаления, переименования и перемещения задач. Класс связан со всеми остальными классами.

- Group Class – класс групп, служит для добавления, удаления и переименования групп. Связан с классом задач и резервного копирования.
- Gamification Class – класс игровых механик, служит для подсчёта статистики, выдачи наград и мотивационных сообщений. Так же включает в себя информацию о достижениях. Связан с классом задач и резервным копированием.
- Notification Class – класс выдачи сообщений и сигналов для своевременного информирования пользователя о сроках выполнения.
- Sharing Class – служит для передачи сообщений в социальные сети, по выбору пользователя.

Помимо диаграммы классов, в проекте представлена диаграмма состояний – StateChart [23] (рис. 4.4.).

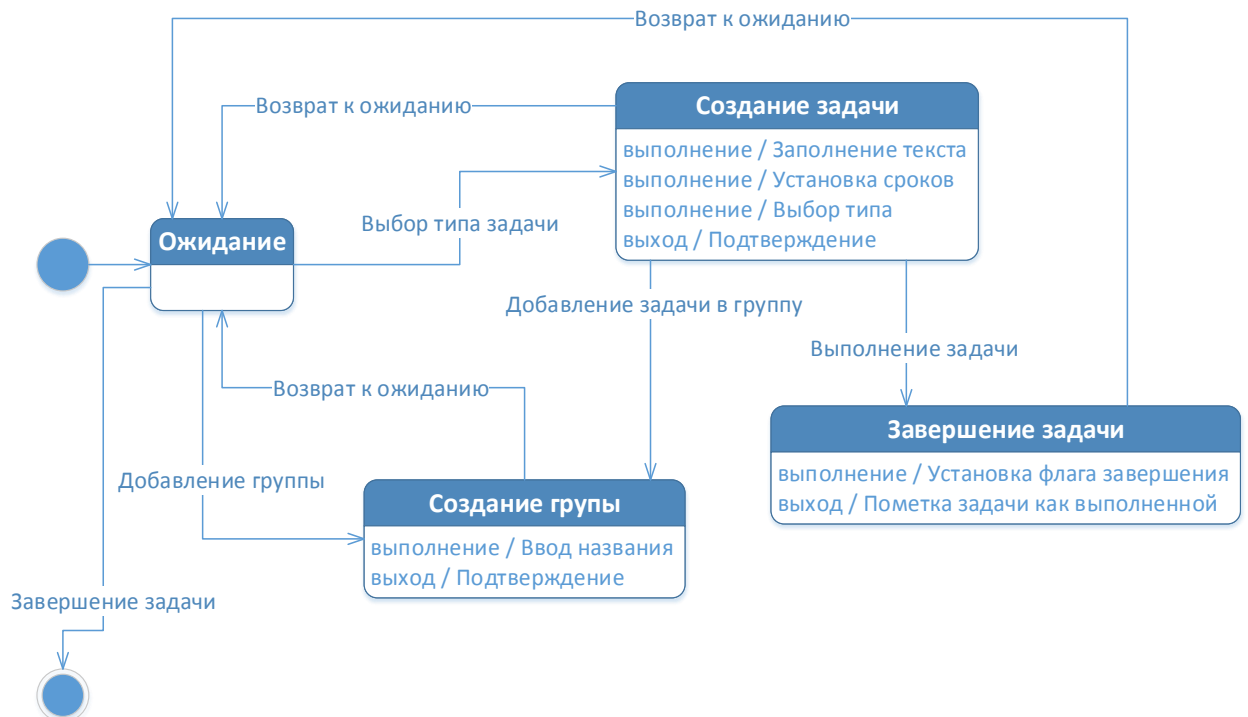


Рис. 4.4. Диаграмма состояний

А так же диаграмма последовательности [14], изображающая последовательность связей между классами внутри разрабатываемой системы (рис. 4.5).

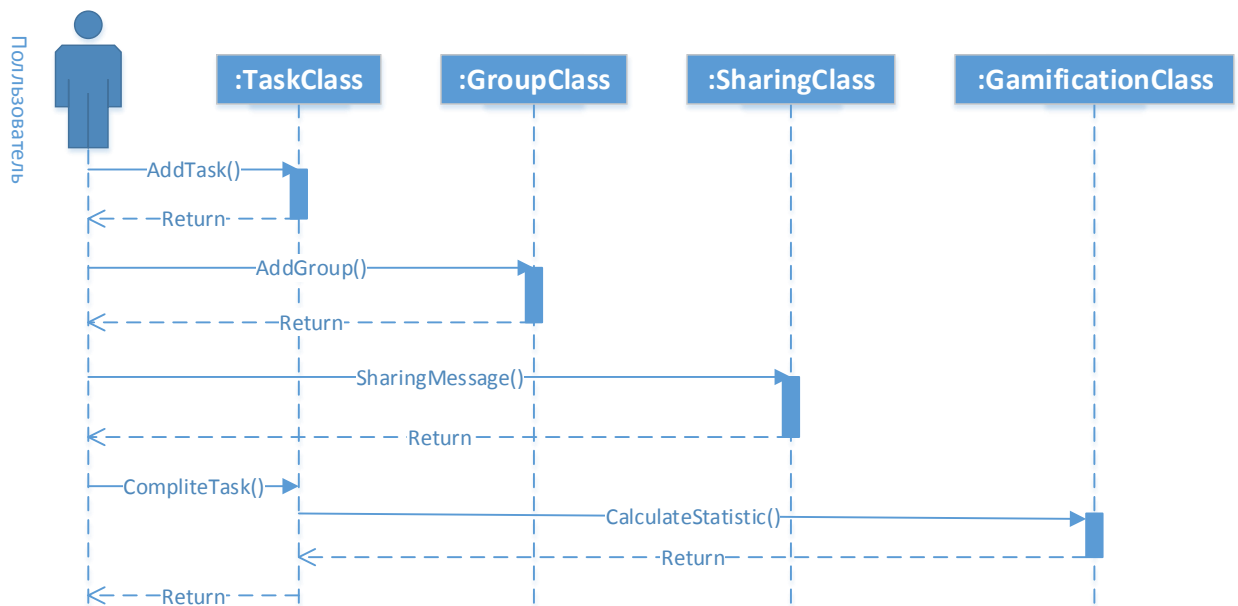


Рис. 4.5. Диаграмма последовательности

4.4. Проектирование интерфейса

В данном разделе были рассмотрены функциональные требования программного продукта и на основании них – спроектирован прототип пользовательского интерфейса [2]. На рисунке 4.6 представлен пример рабочей области главного окна.

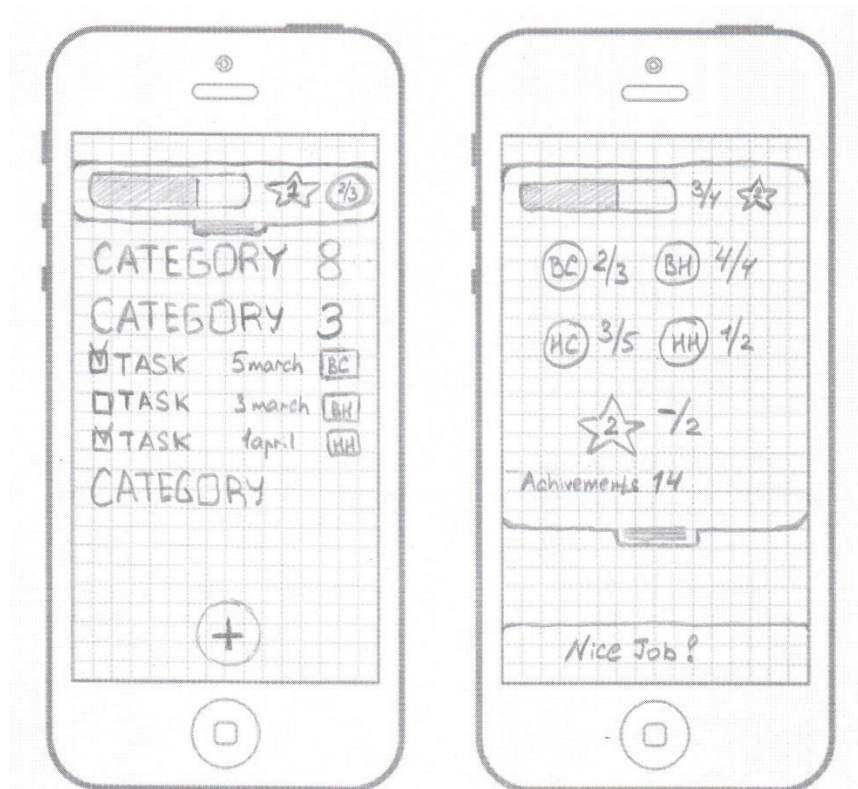


Рис. 4.6. Эскиз главного окна рабочей области

Основной акцент, по результатам исследования, сделан в сторону игровых механик и элементов. Так в рабочей области можно увидеть область выдачи мотивационных сообщений. Интерфейс выполнен в минималистичном стиле, чтобы позволить пользователю за короткий промежуток времени получить желаемый результат.

Добавление групп осуществляется по нажатию значка “+” в верхней части экрана. Кроме того, задачи можно добавлять в уже созданную группу (рис. 4.7). Так же любую задачу можно разместить в сети «ВКонтакте».

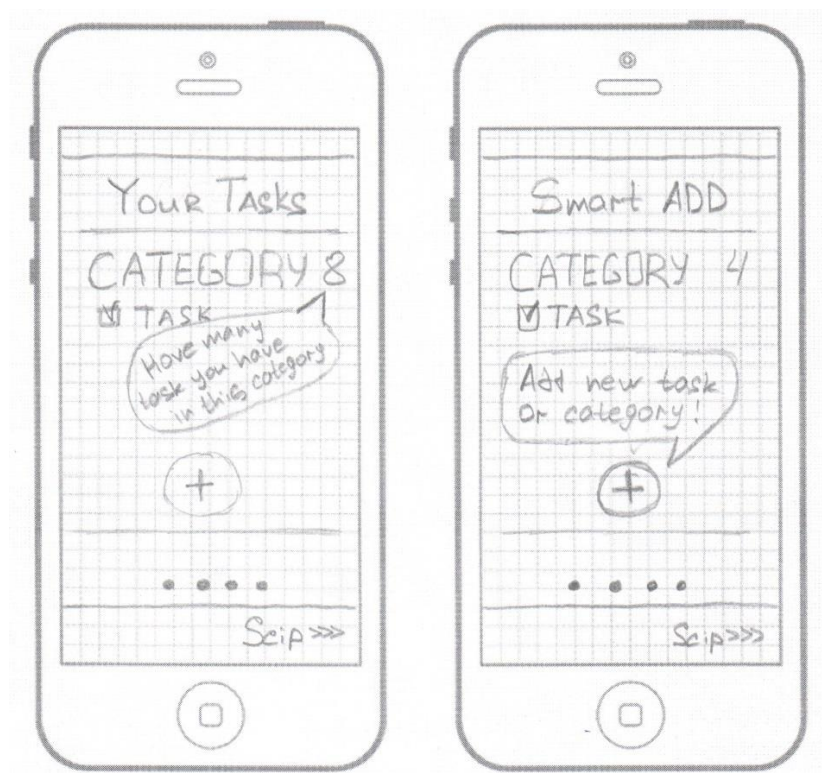


Рис. 4.7. Эскиз добавления задачи

4.5. Руководство пользователя

Запуск приложения осуществляется с иконки на рабочем столе, после которого появляется главное окно рабочей области.

Для того, чтобы добавить новую группу, необходимо нажать на пиктограмму “+”, после чего будет предложено выбрать название.

В самой группе реализована возможность добавления задач. Для этого необходимо ввести название и текст задачи, после чего выбрать её тип. Так же, по желанию, можно задать сроки выполнения (рис. 4.8).

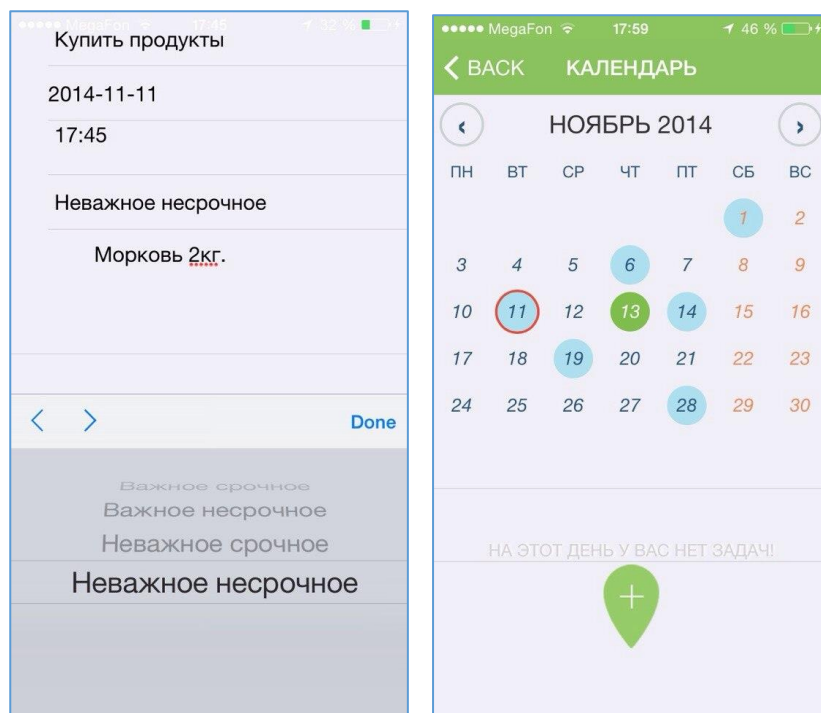


Рис. 4.8. Добавление задачи

Удаление выполненной или ненужной задачи происходит при помощи движения пальца справа налево по задаче. После чего появляется пиктограмма удаления, нажав на которую задача будет изъята из списка (рис. 4.9). Так же, для перемещения задачи, необходимо нажать «Изменить» в левом углу экрана. После чего, схватив задачу пальцем, перенести её на нужную позицию.

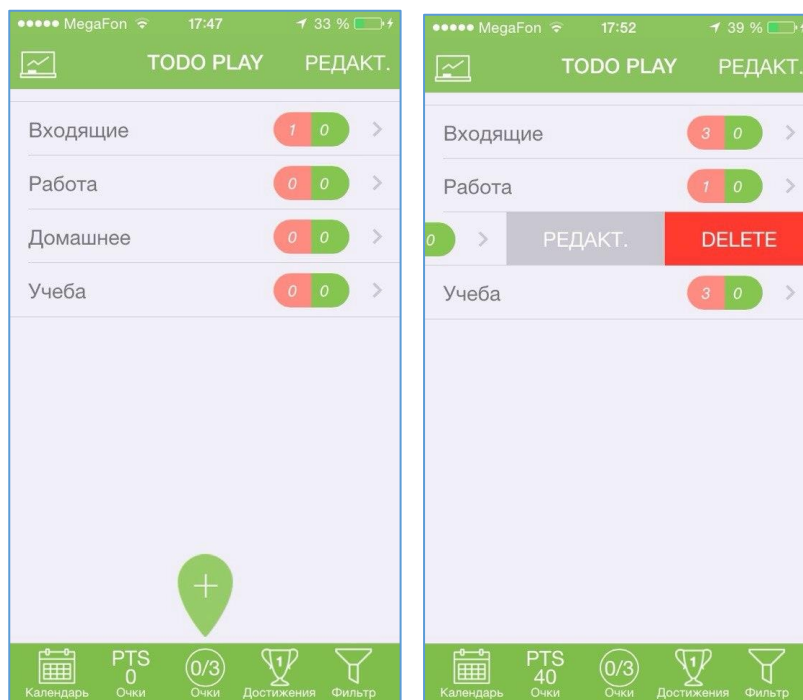
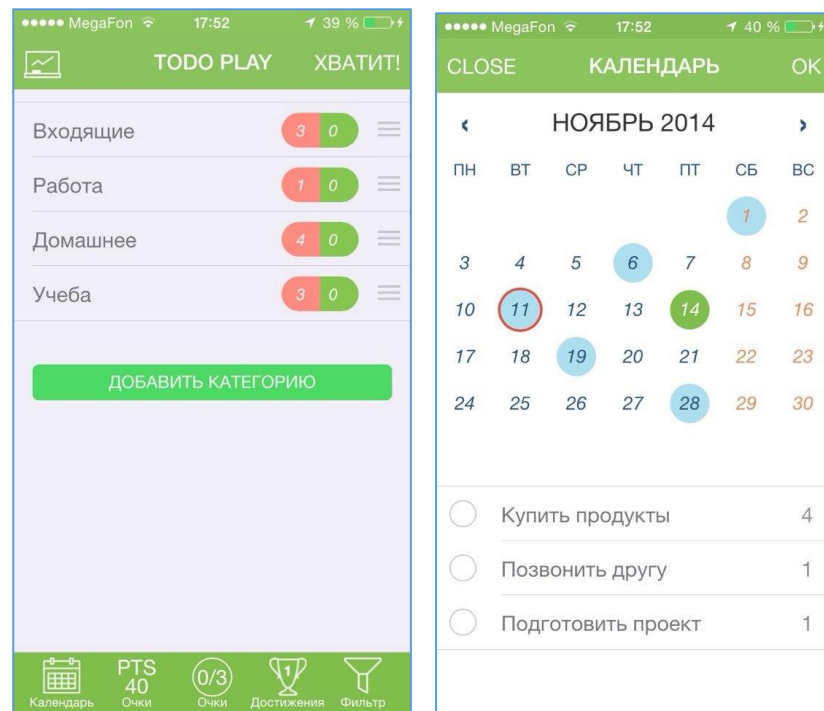


Рис. 4.9. Удаление задачи

Отдельное окно занимает подробная статистика пользователя по выполненным задачам, а так же списку достижений в системе.

**Рис. 4.10. Статистика**

В системе предусмотрена возможность поделиться задачей в социальной сети, для этого необходимо, выбрав соответствующую задачу и нажать на пиктограмму. После чего, заполнив поля сообщения, она будет размещена.

5. ЭКСПЕРЕМЕНТАЛЬНАЯ ЧАСТЬ

5.1. Испытания

Для проверки соответствия программного продукта, необходимо провести ряд тестирований. Для этого все заявленные требования к системе систематизируются и сверяются с реальным положением с использованием тестирующей системы или группы тестировщиков [22].

5.1.1. План испытаний

Для того, чтобы провести полное и всесторонне тестирования соответствия программного продукта поставленным целям, составлен план тестирования:

- Проверка системы на добавление группы.
- Проверка системы на добавление задачи для группы.
- Тестирование системы статистики и обратной связи на предмет скорости реакции и ошибок внутренней математики.
- Тестирование системы выдачи достижений за определённую последовательность действий.
- Проверка работы с социальной сетью «вконтакте» и подключения к внешнему API.

5.1.2. Проверка выполнения функциональных требований

Необходимо сверить программный продукт с заявленными требованиями из части технического задания.

Для проверки работоспособности функций создания задачи и групп, были созданы несколько тестовых задачи разной типизации и сроков окончания действия, а так же групп – пустых и с вложениями (рис. 5.1).

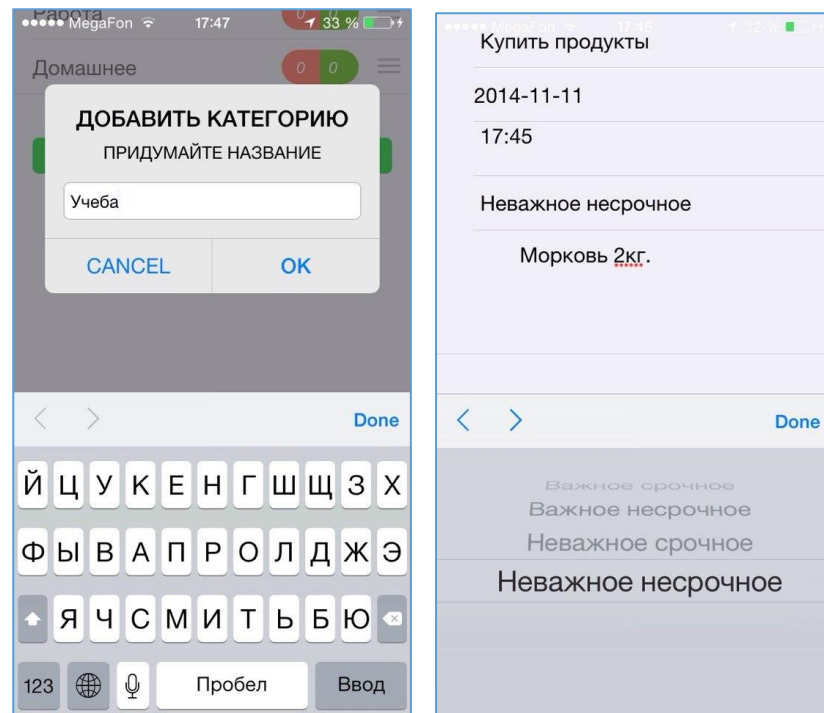


Рис. 5.1. Тестирование функции добавления задачи и группы

Модуль статистики так же подвергся тестированию, при проверке работоспособности задач и групп (рис. 5.2).

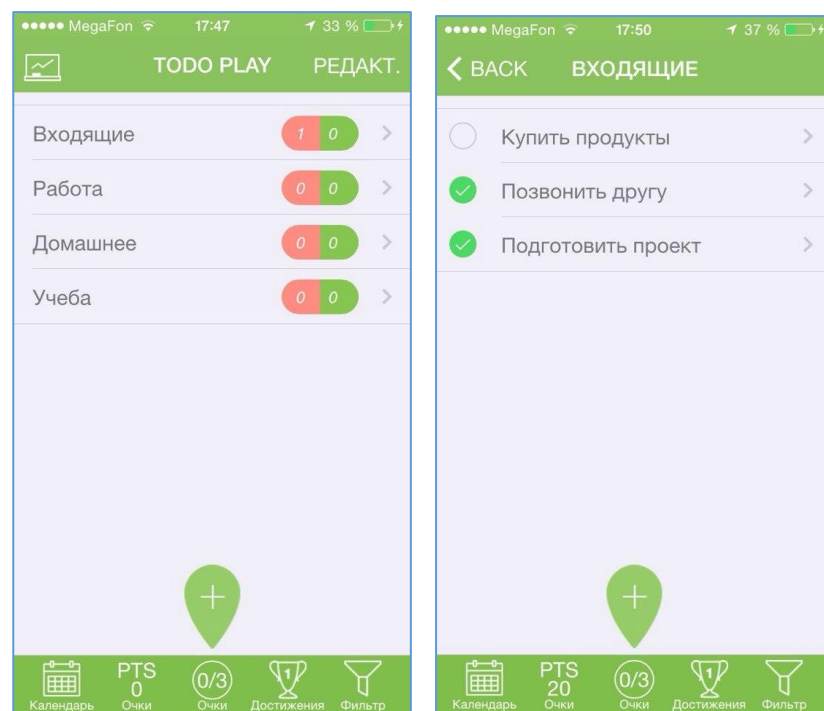


Рис. 5.2. Тестирование модуля выдачи статистики

При внесении изменений, выполнении и удалении задач, статистика сразу вносила изменения в систему и обрабатывала отправку обратной связи,

который, в свою очередь, взаимодействовал со службой выдачи сообщений пользователю, корректность которой так же подтверждена опытным путём.

Последним этапом тестирования стала проверка модулей социализации, для тестирования была выбрана система «вконтакте». Размещение 5 задач разного типа и содержания подтвердили работоспособность модуля.

5.1.3. Нагрузочное тестирование

Программный продукт был протестирован по средствам внутренних инструментов среды xCode [8] (рис. 5.3).

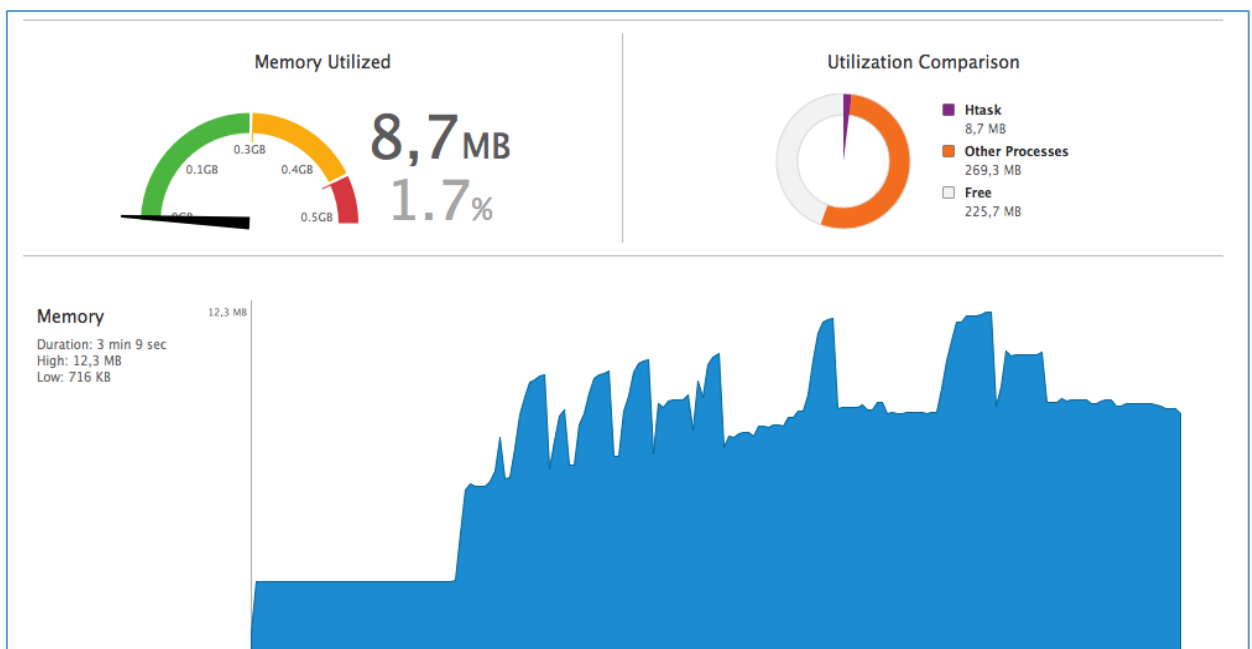


Рис. 5.3. Нагрузочное тестирование xCode

Тестирование проходило при помощи нагрузки на разные аппаратные и программные модули системы.

Тестирование показало, что программный продукт использует малую часть от доступных ресурсов устройства, не перегружает аппаратные модули и быстро реагирует на любые действия пользователя.

5.1.4. Тестирование в исключительных ситуациях

В данном разделе система тестируется на возникновение исключительных ситуаций:

- Закрытие приложения во время создания задачи – если задача была заполнена не полностью, процесс внесения задачи обнуляется.
- При поступлении входящего вызова, приложение переходит в фоновый режим, после которого можно продолжить работу с созданием и редактированием задач в системе.
- Входящее сообщение или нотификация от другого приложения не влияют на работу программы.
- Сбой работы интернет соединения при попытке поделиться задачей в социальных сетях – приводит к выдаче сообщения о невозможности доступа к сети.
- Выключение устройства во время работы приложения – при повторном включении, приложение свёрнуто в тее и возможно продолжение работы.

5.2. Вывод

На основании результатов тестирования, можно сделать вывод, что система полностью соответствует требованиям технического задания, стабильна и корректно реагирует на исключительные ситуация, своевременно оповещая пользователя.

6. ОРГАНИЗАЦИОННАЯ ЧАСТЬ

6.1. Постановка проблемы

Мобильные телефоны, иногда называемые сотовыми, быстро становятся неотъемлемой частью современных средств электросвязи. В некоторых районах мира они являются наиболее надежными или единственными имеющимися в наличии телефонами. В других местах телефоны очень популярны, поскольку позволяют людям поддерживать непрерывную связь, и при этом не нарушается свобода передвижения – качество, оцениваемое по достоинству, как на работе, так и на отдыхе.

Бурная мобильная «телефонизация», захлестнувшая человечество каких-то десять лет назад, продолжается, и по сей день. Неудивительно, что людей стал беспокоить вопрос о влиянии электромагнитных волн, базовых станций и других технических тонкостей на здоровье человека. Эта проблема исследуется уже давно, еще со второй половины XX века, но она затрагивала довольно узкий круг лиц, преимущественно персонал вещательных и специализированных радиостанций. Первые разговоры о вреде сотовой телефонии в России начинались уже в начале 2000г. С тех пор телефоны очень далеко продвинулись в своем развитии, и это дало новые поводы для споров об их вреде. При этом, разумеется, споров стало больше. И если раньше говорилось в основном о неблагоприятном воздействии излучения телефонного передатчика на органы слуха и кору головного мозга, то сейчас говорится и о влиянии на половую систему, на сердце, на внимание и способность к концентрации, на познавательную функцию и даже на сон [24].

В настоящее время ни одна лаборатория или медицинский центр не смогли точно ответить на этот животрепещущий вопрос. Виной тому сложность анализа статистических данных, которыми располагают ученые. Сотовый телефон – устройство электрическое. Общение с другими абонентами, отправка сообщений, мобильные игры, передача данных и

простой в режиме ожидания в большей или меньшей степени сопровождаются генерацией электромагнитных волн. На человека в современном мире воздействуют такое большое количество вредоносных излучений и веществ, что выделить роль сотового телефона в заболеваниях очень сложно. И действительно, разговоры на эту тему не прекращаются с того момента, когда мобильные телефоны только начали входить в нашу жизнь [24].

Всем хорошо известны микроволновые печи, в которых электромагнитное излучение расшатывает мельчайшие составляющие – атомы, и продукты разогреваются. При этом специальная посуда остается холодной, что указывает на избирательность излучения. Точно известно, что электромагнитное излучение частотой выше 1 МГц разогревает наши ткани. Человеческие клетки болезненно относятся к этому процессу. Однако количественно оценить его пока никто не может. Производители сотовых телефонов планомерно поднимают частоты мобильных телефонов. Трубки начинают работать в частотных диапазонах 1800 МГц и 1900 МГц. В этом сантиметровом диапазоне волны становятся непредсказуемыми. Разумеется, зависимость возникает от мощности облучения, вида тканей, времени и частоты.

Как вы понимаете, речь идет о вреде излучателей, то есть, проще говоря, антенн, используемых в телефонах. Ни какие другие составляющие вреда нанести не способны, это касается как материалов используемых в производстве (стекло, пластик и т.д.), так и отдельных элементов.

Исходя из изложенного выше текста, целью работы явилось познакомиться с проблемой влияния мобильных телефонов на представителей разных возрастных.

6.2. Излучение от устройств сотовой связи

SAR [24] – (Specific Absorbtion Rate) – это единица измерения, энергия электромагнитного поля, показывающая максимальную удельную

мощность, поглощаемую человеческим телом при обычном разговоре по сотовому телефону. Максимальный безопасный уровень – 2,0. Подавляющее большинство современных телефонов имеют SAR от 0,5 до 1,0 Единицей измерения SAR является ватт на килограмм (Вт/кг). В Северной Америке и странах Азиатско – Тихоокеанского региона величина SAR определяется на 1 грамм ткани. Максимальное значение SAR составляет 1,6 Вт/кг.

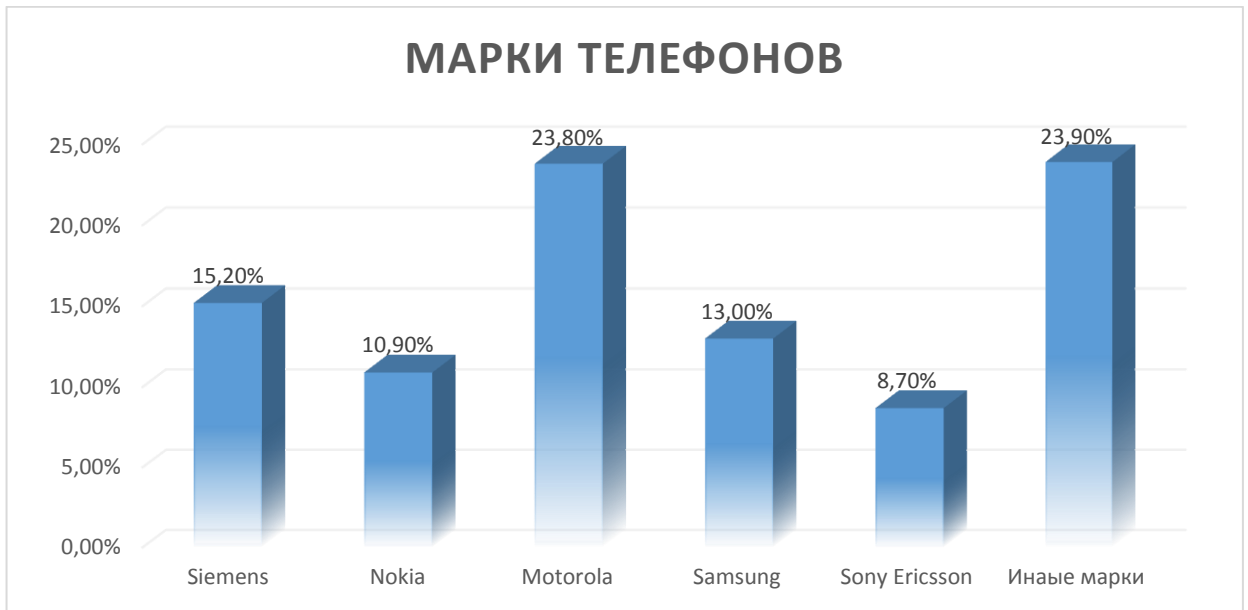


Рис. 6.1. Марки телефонов

Страны с наибольшим количеством мобильных телефонов на душу населения:

- Тайвань;
- Люксембург;
- Гонконг;
- Италия;
- Исландия.

6.2.1. Стандарты сотовой связи и их биологический смысл

В определенной мере интенсивность электромагнитного излучения зависит от модели сотового телефона. Однако, принципиально больше интенсивность ЭМИ зависит не от модели телефона, а от стандарта сотовой связи. Иными словами – от способа использования радиочастот, который

практикует ваш сотовый оператор. Этих стандартов в мире около двух десятков, но в большинстве стран и в России используются в основном 4 из них. Различные стандарты требуют различных уровней мощности передатчика и приемника и, соответственно, два одинаковых радиотелефона у разных компаний – операторов сотовой связи испускают на своих пользователей электромагнитные излучения. Как видно из таблицы, самые вредные (и уже устаревшие) – аналоговые стандарты. Наиболее безопасной, самый экономичный для аккумуляторов стандарт сотовой связи – это CDMA [24].

Таблица 6.1

Стандарты сотовой связи

<i>Стандарт сотовой связи</i>	<i>Отечественные компании сотовой связи</i>	<i>ЭМИ</i>
NMT -450 (Nordic Mobile Telephone), аналоговый	Сотел	3 Вт
GSM (Global System for Mobile Communications)	НСС, МТС, Билайн, Мегафон	2Вт
AMPS (Advanced Mobile Phone System – Усовершенствованная система мобильной связи), аналоговый	ПССР. В настоящее время практически не используется в густонаселенных районах из-за малого количества активных абонентов в пределах одной соты	0,6 Вт
CDMA (Code Division Multiple Access), цифровой.	Ведется проектирование центров коммуникации сотовой связи для ЗАО «Скайлинк»	0,2 Вт

Системы наиболее подверженные вредному влиянию мобильных телефонов:

- центральная нервная система;
- иммунная система;

- эндокринная система;
- половая система.

Центральная нервная система.

1. Повышается риск развития опухоли головного мозга.
2. Нарушение нормального режима сна, нарушение смены фаз дня.
3. Влияние на биоэлектрическую активность мозга.
4. У автомобилистов появляется эффект «невнимательной слепоты».
5. У водителей транспорта сетчатка глаза не реагирует даже на смену цвета светофора и на дорожные знаки.
6. После 15 секунды разговора наступает угнетение альфа волн мозговой деятельности, проявляется ослаблением волевых функций, памяти, внимания, концентрации.
7. При использовании телефона в течение нескольких лет наблюдается нарушение сна, угнетение иммунитета, анемия.
8. При длительном разговоре, наблюдается увеличение температуры уха, барабанной перепонки, прилегающих тканей и прилегающего участка мозга.
9. Использование мобильного телефона в течение 10 лет увеличивает риск развития опухоли головного мозга.

Иммунная система.

1. Мобильные телефоны провоцируют астму и экзему.
2. Сбой в иммунной системе способствует уязвимости для болезней.

Эндокринная система.

1. Разрушение клеток крови – эритроцитов (анемия).
2. Увеличение кровяного давления.
3. Увеличение выработки адреналина, регистрируются небольшие колебания уровня гормонов.

Половая система.

1. Основной вред – для мужчин. Наблюдается угнетение сперматогенеза, угнетение рождаемости девочек, повышение числа врожденных пороков и уродств.
2. Изменение ДНК, что может спровоцировать рождение детей с врожденными пороками.

Психологические спецэффекты.

Определенная часть населения имеет психологические нарушения и подвержена возникновению разнообразных страхов (фобий) и пристрастий (маний).

Мобильный телефон вызывает зависимость – такой вывод делают психологи, изучая владельцев сотовых телефонов. Психологическая зависимость от аппарата может проявляться в нескольких видах.

- Беспокойство по поводу отсутствия телефона. Если у вас нет заветной трубки, то возникает ощущение одиночества, депрессивное настроение, человек испытывает сильное беспокойство, в некоторых случаях способное перерасти в панику. При этом фиксируются гипертонические явления – скачки артериального давления. Если близкий человек не взял с собой сотовый телефон – повод для беспокойства.

- SMS- мания. Человек пишет и отправляет десятки SMS в день и не может остановиться, даже если ему проще позвонить.

- Информационная мания. Владелец телефона постоянно ждет звонка или SMS, достает мобильник, проверяет, не пропустил ли он важную информацию, читает электронную почту и SMS- сообщения. В итоге человек не может запомнить, какая именно информация для него важна, и обращается к своему мобильному телефону снова и снова.

- Зависимость от самого прибора (Новая модель, аксессуары). Владелец телефона стремится купить новую модель, какие-либо аксессуары, выделяющие его сотовый из массы обычных – средних телефонов. В этой ситуации человек ощущает повышение социального

статуса и готов тратить все деньги на свой сотовый, даже в ущерб другим важным вещам.

- Люди испытывают сильное волнение, раздражительность, бессонницу, необъяснимые страхи. И все это из-за панического опасения, что телефон прослушивается, а содержание разговоров могут узнать начальник на работе или другие люди.

- Мобильный фантом. Психологи сравнивают подобное поведение с синдромом ампутированной конечности, когда у человека появляется ложное ощущение боли в ампутированной ноге.

- Звуковые галлюцинации или «Мания звонка». Порой людям кажется, что они слышат звонок мобильного телефона, хотя в действительности аппарат не звонил и был вовсе выключен. Такие случаи хотя бы раз происходили со всеми обладателями сотовый, однако у некоторых людей несуществующие звонки становятся хронической фобией.

- Повышение риска ДТП. Хотя многие аспекты влияния мобильных телефонов на здоровье остаются пока не выясненными, эксперты сходятся в одном: разговоры по телефону за рулем опасны. Анализ 700 дорожно-транспортных происшествий в Великобритании показал, что разговоры по телефону повышают риск попадания в аварию в 4 раза.

Мобильный телефон особенно опасен для беременных женщин.

Беременность – важный и ответственный период в жизни женщины и её не рожденного еще ребенка. Многочисленные наблюдения говорят о том, что электромагнитное излучение снижает иммунитет, влияет на эндокринную систему. А самое неприятное – «прогулки по электромагнитным полям во время беременности могут привести к самопроизвольным абортam, преждевременным родам, появлению врожденных пороков развития у детей [24].

При разговоре беременной женщины по сотовому телефону практически все её тело подвергается воздействию электромагнитного поля (ЭМП), включая развивающийся плод. Чувствительность эмбриона к повреждающим факторам значительно выше, чем чувствительность материнского организма. Внутриутробное повреждение плода ЭМП может произойти на любом этапе его развития от момента оплодотворения, однако периодами максимальной чувствительности к ЭМП являются ранние стадии развития зародыша – так считают в Федеральном научном центре гигиены им. Ф.Ф. Эрисмана.

В научных исследованиях биологического действия ЭМП установлено, что на эмбриональное развитие потомства влияет даже малая интенсивность ЭМП. Так, потомство облученных животных менее жизнеспособно, наблюдаются аномалии развития, уродства, отставание в весе, нарушение функции высших отделов центральной нервной системы (замедленная выработка и снижение способности к сохранению оборонительных и двигательных-пищевых условных рефлексов), смещение темпов постнатального развития (после рождения).

Центральная нервная система

1. Повышается риск развития опухоли головного мозга.
2. Нарушение нормального режима сна, нарушение смены фаз дня.
3. Влияние на биоэлектрическую активность мозга.
4. После 15 секунды разговора наступает угнетение альфа волн мозговой деятельности, проявляется ослаблением волевых функций, памяти, внимания, концентрации.
5. При длительном разговоре, наблюдается увеличение температуры уха, барабанной перепонки, прилегающих тканей и прилегающего участка мозга.

Иммунная система

1. Мобильные телефоны провоцируют астму и экзему.
2. Сбой в иммунной системе способствует уязвимости для болезней.

Эндокринная система

1. Разрушение клеток крови – эритроцитов (анемия).
2. Увеличение кровяного давления.
3. Увеличение выработки адреналина, регистрируются небольшие колебания уровня гормонов.

Симптомы, наиболее часто возникающие у гиперчувствительных людей:

1. Усталость, напряжение, нарушение сна (нервная система).
2. Головная боль, головокружение, утомляемость, жар.
3. Ломота и боль в мышцах (тело).
4. Жжение в глазах.
5. Различные, менее общие симптомы, которые затрагивают уши, нос, горло, а также расстройство желудка.
6. Гиперчувствительность характеризуется широким спектром возникающих неспецифических симптомов.

Экзотические случаи пагубного влияния мобильных телефонов

Летом 2008 года тема поражения молнией поднималась в Великобритании: в поле зрения попал случай, когда молния ударила в телефон 15-летней школьницы как раз в тот момент, когда она принимала звонок. Заключение это удалось по характеру травм. Как утверждают врачи, разряд прошел через ухо и привел к разрыву перепонки и поражению головного мозга. В результате чего девочка потеряла слух, память и оказалась парализованной. По опубликованным данным, ежегодно только в Великобритании молнией ударяет около 40 человек, десятая часть из которых погибает. Вероятность попадания молнии в работающий телефон в несколько раз выше вероятности попадания в человека. Это удалось установить в Китае, в 2004 году, когда один из туристов получил удар молнии во время экскурсии около Великой китайской стены. По мнению китайских врачей и сотрудников Пекинского университета, мобильный телефон, изучающий электромагнитные волны, способствуют прохождению

электричества. В случае обычного попадания молнии в человека заряд рассеивается по всему телу, благодаря чему, многие из пострадавших выживают. В случае с мобильным телефоном ударно приходится точно в цель, так что вероятность летального исхода многократно возрастает. Подобные случаи, имели место быть не только в Великобритании и Китае, но ещё и во Франции, в Южной Корее, а также Канаде и на южноамериканском континенте.

Увлечение СМС может привести к проблемам со здоровьем

Итальянские медики предупреждают, что слишком сильное увлечение текстовыми сообщениями с мобильного телефона, может закончиться **тендинитом**, то есть воспалением сухожилий пальцев. Особенно высок риск, заболеть им у подростков. Как сообщается, в городе Савона на севере Италии была госпитализирована с тендинитом 13-летняя девочка. Юная любительница СМС набирала не меньше 100 сообщений в день. Врачи назначили ей противовоспалительные средства, но главное в лечении было дать отдых пальцам.

По данным недавнего исследования в Италии 37% детей страдают «телефонной зависимостью», в то время как в России 87%. Помимо тендинита неумеренное увлечение мобильным телефоном может привести к раздражительности и перепадам настроения.

Кроме того, японский доктор Хаджим Кимата полагает, что микроволны, испускаемые мобильными телефонами, могут возбудить антигены, которые вызывают аллергические реакции.

6.2.2. Проживание в районе активности сотового излучения

В Германии была проведена большая работа по выяснению зависимости между онкологическими заболеваниями и местожительством. В результате выяснили, что у людей, которых живут рядом с телефонными станциями, повышается риск заболеть раком. Длительное воздействие магнитных полей проявляется ухудшением памяти, снижением половой

активности, понижением иммунитета и повышенной утомляемостью. Жители таких районов чаще заболевают инфекционными заболеваниями, они дольше протекают и тяжело поддаются лечению

Согласно докладу ученых Кореи 10,9% из 1034 опрошенных в возрасте от 20 лет и старше заявили, что они чувствовали дискомфорт из-за использования сотового телефона. В частности, 67 человек жаловались на возникновение кратковременной глухоты, а 59 респондентов страдали от головных болей, 46 человек периодически донимают внезапные приступы усталости, 29 – периодически не могут заставить себя сконцентрироваться. Кроме того, 93% опрошенных считают, что электромагнитные излучения (или ЭДС – электродвижущая сила) сотовых телефонов вредны и влияют на наше здоровье. И лишь 0,5% корейцев сказали, что, по их мнению, мобильники безвредны. Чтобы оградить себя от возможного вреда здоровью, 23,6% респондентов предпочитают чаще пользоваться фиксированной телефонной связью, нежели чем мобильной. А 20,8% не любят долго держать телефон в районе головы, а потому чаще переписываются сообщениями и реже разговаривают.

6.3. Выводы

На основании данных, описанных в работе, были сделаны выводы о вреде воздействия сотовой связи на здоровье и самочувствие людей, использующих мобильные телефоны. Следовательно, рекомендуется соблюдать все меры предосторожности при использовании мобильного телефона, а так же минимизировать использование экранированных чехлов.

ЗАКЛЮЧЕНИЕ

В результате реализации проектного решения, были выполнены все поставленные цели, результат полностью соответствует заявленным ожиданиям. Мотивационные инструменты, в виде игрофикационных методик, в данном программном продукте подтвердили выводы, сделанные по результатам исследования, и показывают высокую результативность на всех уровнях вовлечения и мотивации конечного пользователя.

Результаты, полученные в ходе выполнения дипломной работы докладывались на 4-ой Международной конференции в области ИТ и 69-ой научной студенческой конференции БГТУ, и опубликованы в сборниках конференций [5, 6].

Применение данного программного продукта позволяет увеличить реальную продуктивность конечного пользователя по средствам его вовлечения в процесс, а так же, используя гибкую методологию тайм-менеджмента, грамотно распределять важность поставленных задач.

Направлением дальнейшего развития проекта:

1. Приложение является коммерческой разработкой и будет размещено во внутренний цифровой магазин фирмы Apple, для всестороннего распространения за рубежом.
2. Запланировано улучшение и оптимизация уже существующих подходов в функционированию программы. Внесение изменений, на основании первых отзывов клиентов, а так же периодическое обновление внутренней структуры игрофикационного модуля, для длительного поддержания пользовательской лояльности к системе.
3. Проектирование и разработка Android-версии приложения, после анализа необходимых улучшений iOS-версии.
4. Разработка веб-интерфейса проекта для доступа через персональный компьютер и смежные устройства.

СПИСОК ЛИТЕРАТУРЫ

1. Gamification Coursera. Режим доступа:
<https://class.coursera.org/gamification>.
2. iOS Human Interface Guidelines – проектирование интерфейсов Apple.
Режим доступа: <https://developer.apple.com/library/ios/MobileHIG>.
3. Аллен, Д. Как привести дела в порядок: искусство продуктивности без стресса / Д. Аллен. – Изд-во Манн, Иванов и Фербер, 2014. – 368 с.
4. Аллен, Э. Типичные ошибки проектирования / Э. Аллен. – Изд-во Питер, 2003. – 224 с.
5. Варламов, Д. Сборник 4-ой международной конференции в области инноваций в IT / Д. Варламов. – Изд-во Юго-Зап. Гос. Ун-т. Курск, 2014. – Т.1.С. – 183-187.
6. Варламов, Д. Сборник 69ой студенческой научной конференции Брянского Государственного Технического Университета / Д. Варламов. – Изд-во БГТУ, 2014. В печати.
7. Гамма, Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Р. Хелм, Р. Джонсон, Д. Влиссидес. – Изд-во Питер, 2007. – 366 с.
8. Гленфорд, М. Искусство тестирования программ / Т. Баджетт, К. Сандлер. – Изд-во Диалектика, Вильямс, 2012. – 272 с.
9. Грабер, М. SQL / М. Грабер. – Изд-во Лори, 2007. – 672 с.
10. Зикерман, Г. Геймификация в бизнесе. Как пробиться сквозь шум и завладеть вниманием сотрудников и клиентов / Д. Линдер. – Изд-во Манн, Иванов и Фербер, 2014. – 272 с.
11. Игрофикационная система Microsoft Labs. Режим доступа:
<http://www.ribbonhero.com/>
12. Кови, С. Семь навыков высокоэффективных людей / С. Кови. – Изд-во Альпина Паблишер, 2013. – 374 с.

13. Марафон вебинаров – BGS. Режим доступа:
<https://www.facebook.com/groups/223578241180204>.
14. Марсинчук, Р. UML для простых смертных. Руководство / Э. Нейбург. – Изд-во Лори, 2008. – 304 с.
15. Персональный информационный менеджер. Режим доступа:
<http://dic.academic.ru/dic.nsf/1091018>.
16. Портал разработчиков Apple. Режим доступа:
<https://developer.apple.com>.
17. Программа-аналог AnyDo. Режим доступа: <https://www.any.do>.
18. Программа-аналог EpicWin. Режим доступа:
<https://itunes.apple.com/ru/app/epicwin/id372927221?mt=8>.
19. Программа-аналог WunderList. Режим доступа:
<https://www.wunderlist.com/ru>.
20. Руководство пользователя – советы по составлению. Режим доступа:
<http://analyst.by/articles/rukovodstvo-polzovatelya-sovetyi-dlya-sostavleniya>.
21. Среда разработки xCode. Режим доступа:
<https://developer.apple.com/xcode>.
22. Тамре, Л. Введение в тестирование программного обеспечения / Л. Тамре. – Изд-во Вильямс, 2003. – 368 с.
23. Фаулер, М. UML. Основы. Краткое руководство по стандартному языку объектного моделирования / М. Фаулер. – Изд-во Символ-Плюс, 2011. – 192 с.
24. Характеристика влияния сотового излучения на организм. Режим доступа: <http://www.medicinform.net/human/humanis/human119.htm>.
25. Харрингтон, Дж. Проектирование реляционных баз данных / Дж. Харрингтон. – Изд-во Лори, 2006. – 230 с.
26. Элдер, А. Экономика предприятия / А. Элдер. – Изд-во Высшая школа, 2011. – 464 с.

ПРИЛОЖЕНИЕ 1

Сведения об авторе, руководителе и вузе

1. Наименование темы конкурсной работы:

“Мобильное приложение «Менеджер задач» с элементами игрофикации”.

2. Данные об авторе:

- 2.1. Варламов Дмитрий Олегович
- 2.2. Информатика и вычислительная техника, бакалавр
- 2.3. 2014
- 2.4. Ул. 2-я Аллея, дом 7а кв. 9. Индекс 241020
- 2.5. 8(920)-8-468-468
- 2.6. Dmitriy1708@bk.ru

3. Данные о научном руководителе:

- 3.1. Лагереv Дмитрий Григорьевич
- 3.2. Брянский Государственный Технический Университет
- 3.3. Преподаватель
- 3.4. Кандидат технических наук, доцент
- 3.5. .
- 3.6. .
- 3.7. .

4. Данные о вузе:

- 4.1. Брянский Государственный Технический Университет
- 4.2. Факультет информационных технологий
- 4.3. Информатика и вычислительная техника
- 4.4. .
- 4.5. Подвесовский Александр Георгиевич

*Листинг 1. Добавление элемента***HTask_addItem.h**

```
// Htasks
//
// Copyright (c) 2014 DVarlamov. All rights reserved.
//
#import <UIKit/UIKit.h>
@class Con_HTask_listTableViewController;
@interface Con_HTask_addItem : UITableViewController <UITextFieldDelegate>
-(IBAction)cancelBtn:(id)sender;
-(IBAction)doneBtn:(id)sender;
@property (nonatomic, strong) IBOutlet UITextField * nameField;
@property (weak, nonatomic) IBOutlet UIDatePicker *deadlineDatePicker;
@property (weak, nonatomic) IBOutlet UISegmentedControl *modeTask;

@property (nonatomic, strong) Con_HTask_listTableViewController *taskListViewController;
@end
```

HTask_addItem.m

```
// Htasks
//
// Copyright (c) 2014 DVarlamov. All rights reserved.
//
#import "Con_HTask_addItem.h"
#import "Con_HTask_listTableViewController.h"
#import "Htask.h"
@interface Con_HTask_addItem ()

@end

@implementation Con_HTask_addItem
@synthesize nameField = _nameField,taskListViewController = _taskListViewController;

- (id)initWithStyle:(UITableViewStyle)style
{
    self = [super initWithStyle:style];
    if (self) {
        // Custom initialization
    }
    return self;
}

- (void)viewDidLoad
{
    [super viewDidLoad];
    [self.nameField setDelegate:self];
    [self.nameField setReturnKeyType:UIReturnKeyDone];
    [self.nameField addTarget:self
                        action:@selector(textFieldFinished:)
                        forControlEvents:UIControlEventEditingDidEndOnExit];
    // Uncomment the following line to preserve selection between presentations.
    // self.clearsSelectionOnViewWillAppear = NO;

    // Uncomment the following line to display an Edit button in the navigation bar for this view controller.
    // self.navigationItem.rightBarButtonItem = self.editButtonItem;
}

- (void)didReceiveMemoryWarning
```

```

{
    [super didReceiveMemoryWarning];
    // Dispose of any resources that can be recreated.
}
#pragma actions
- (void)cancelBtn:(id)sender{
    [self dismissViewControllerAnimated:YES completion:nil];
}
- (void)doneBtn:(id)sender{
    [self.taskListViewController addTaskName:self.nameField.text deadline:[self.deadlineDatePicker date]
    type:[self.modeTask titleForSegmentAtIndex:self.modeTask.selectedSegmentIndex]];
    [self.tableView reloadData];
    [self dismissViewControllerAnimated:YES completion:nil];
    [self.taskListViewController.tableView reloadData];
}
- (void)textFieldFinished:(id)sender
{
    [sender resignFirstResponder];
}
@end

```

Листинг 2. Изменение элемента

HTask_editTask.h

```

// Htasks
//
// Copyright (c) 2014 DVarlamov. All rights reserved.
//

#import <UIKit/UIKit.h>
@class Con_HTask_listTableViewController;
@class Htask;
@interface Con_HTask_editTask : UITableViewController
@property (nonatomic,strong) IBOutlet UITextField *nameField;
@property (nonatomic,strong) IBOutlet UISwitch *doneSwitch;
@property (nonatomic,strong) Htask *task;
@property (nonatomic, strong) Con_HTask_listTableViewController *taskListViewController;
@property (nonatomic, assign) NSInteger row;
-(IBAction)changeData:(id)sender;

@end

```

HTask_editTask.m

```

// Htasks
//
// Copyright (c) 2014 DVarlamov. All rights reserved.
//

#import "Con_HTask_editTask.h"
#import "Con_HTask_listTableViewController.h"
#import "Htask.h"
#import "AppDelegate.h"
@interface Con_HTask_editTask ()
@property (nonatomic, readonly) NSManagedObjectContext *managedObjectContext;
@end

@implementation Con_HTask_editTask
@synthesize nameField = _nameField, doneSwitch = _doneSwitch, task = _task;

```

```

@synthesize taskListViewController = _taskListViewController;
@synthesize row = _row;

- (void)viewDidLoad
{
    [super viewDidLoad];
    // id val = self.task.done;
    self.nameField.text = self.task.name;
    [self.doneSwitch setOn:[self.task.done boolValue]];
}
- (NSManagedObjectContext *)managedObjectContext{
    return [(AppDelegate *) [[UIApplication sharedApplication] delegate] managedObjectContext];
}
- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
    // Dispose of any resources that can be recreated.
}
#pragma action
- (void)changeData:(id)sender {
    if (self.doneSwitch.isOn && ![self.task.done boolValue]) {
        if ([self.task.type isEqualToString:@"Легкий"]) {
            [self.taskListViewController showMessage:@"Молодец, так держать!"];
        }
        else if ([self.task.type isEqualToString:@"Нормальный"]){
            [self.taskListViewController showMessage:@"Знаешь, это было круто!"];
        }
        else if ([self.task.type isEqualToString:@"Сложный"]){
            [self.taskListViewController showMessage:@"Я с самого начала верил в тебя!"];
        }
        else if ([self.task.type isEqualToString:@"Невероятный"]){
            [self.taskListViewController showMessage:@"Я вижу у нас новый супергерой!"];
        }
    }
    self.task.name = self.nameField.text;
    self.task.done = [NSNumber numberWithBool:self.doneSwitch.isOn];
}
@end

```

Листинг 3. ViewController

HTask_listTableViewController.h

```

// Htasks
//
// Copyright (c) 2014 DVarlamov. All rights reserved.
//

#import <UIKit/UIKit.h>
@interface Con_HTask_listTableViewController : UITableViewController

@property (nonatomic, strong) NSArray *lists;
@property (nonatomic, readonly) NSManagedObjectContext * managedObjectContext;
@property (nonatomic, strong) NSMutableArray *tasks;
@property (weak, nonatomic) IBOutlet UILabel *allTasks;
@property (weak, nonatomic) IBOutlet UILabel *completeTasks;
@property (weak, nonatomic) IBOutlet UILabel *message;

```

```

-(IBAction)editingBtn:(id)sender;
-(void)addTaskName:(NSString *)name deadline:(NSDate *)date type:(NSString *)type;
-(void)showMessage:(NSString *)str;
@end

```

HTask_listTableViewController.m

```

// Htasks
//
// Created by Denis on 18.05.14.
// Copyright (c) 2014 DVarlamov. All rights reserved.
//
#import "AppDelegate.h"
#import "Con_HTask_listTableViewController.h"
#import "Htask.h"
#import "Con_HTask_addItem.h"
#import "Con_HTask_editTask.h"

@interface Con_HTask_listTableViewController ()
//@property (nonatomic, strong) NSArray *lists;
//@property (nonatomic, readonly) NSManagedObjectContext * managedObjectContext;
//-(void)addTaskName:(NSString *)name deadline:(NSDate *)date;
@end

@implementation Con_HTask_listTableViewController
@synthesize tasks = _tasks, lists = _lists;
- (void)viewDidLoad
{
    [super viewDidLoad];
    NSFetchRequest *fetchRequest = [NSFetchRequest fetchRequestWithEntityName:@"Htask"];
    fetchRequest.sortDescriptors = [NSArray arrayWithObject:[NSSortDescriptor
sortDescriptorWithKey:@"created" ascending:YES]];
    self.lists = [self.managedObjectContext executeFetchRequest:fetchRequest error:nil];
    [self.tableView reloadData];
    [self updateStatistic];
}
- (NSManagedObjectContext *)managedObjectContext {
    return [(AppDelegate *) [[UIApplication sharedApplication] delegate] managedObjectContext];
}
- (void)viewWillAppear:(BOOL)animated {
    [super viewWillAppear:animated];
    [self.tableView reloadData];
    [self updateStatistic];
}
- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
    // Dispose of any resources that can be recreated.
}

#pragma mark - Table view data source

- (NSInteger)numberOfSectionsInTableView:(UITableView *)tableView
{
    return 1;
}

- (NSInteger)tableView:(UITableView *)tableView numberOfRowsInSection:(NSInteger)section
{
    return self.lists.count;
}

```

```

- (UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    static NSString *doneIdentifier = @"done_task";
    static NSString *notDoneIdentifier = @"not_done_task";

    Htask *model = [self.lists objectAtIndex:indexPath.row];
    NSString *CellIdentifier = [model.done boolValue]? doneIdentifier: notDoneIdentifier;
    UITableViewCell *Cell = [tableView dequeueReusableCellWithIdentifier:CellIdentifier];
    if (Cell == nil) {
        Cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault reuseIdentifier:CellIdentifier];
    }
    NSDateFormatter *formatter = [[NSDateFormatter alloc] init];
    [formatter setDateFormat:@"%yyyy/MM/dd"];

    Cell.textLabel.text = model.name;
    Cell.detailTextLabel.text = [NSString stringWithFormat:@"Завершить до: %@ Сложность: %@", [formatter
stringFromDate:model.deadline], model.type];
    // Configure the cell...
    [self updateStatistic];
    return Cell;
}

- (void)prepareForSegue:(UIStoryboardSegue *)segue sender:(id)sender {
    if ([segue.identifier isEqualToString:@"addTask"]) {
        UINavigationController *nav = segue.destinationViewController;
        Con_HTask_addItem *addTask = [nav.viewControllers objectAtIndex:0];
        addTask.taskListViewController = self;
    }
    else if ([segue.identifier isEqualToString:@"editTask"] || [segue.identifier isEqualToString:@"editDoneTask"]){
        Con_HTask_editTask *editTask = segue.destinationViewController;
        editTask.task = [self.lists objectAtIndex:self.tableView.indexPathForSelectedRow.row];
        editTask.taskListViewController = self;
        editTask.row = self.tableView.indexPathForSelectedRow.row;
    }
    [self updateStatistic];
}
/*
// Override to support conditional editing of the table view.
- (BOOL)tableView:(UITableView *)tableView canEditRowAtIndexPath:(NSIndexPath *)indexPath
{
    // Return NO if you do not want the specified item to be editable.
    return YES;
}
*/

// Override to support editing the table view.
- (void)tableView:(UITableView *)tableView commitEditingStyle:(UITableViewCellEditingStyle)editingStyle
forRowAtIndexPath:(NSIndexPath *)indexPath
{
    if (editingStyle == UITableViewCellEditingStyleDelete) {
        // Delete the row from the data source
        Htask *task = [self.lists objectAtIndex:indexPath.row];
        BOOL fl = [task.done boolValue];
        if (!fl) {
            [self showMessage:@"Действительно, прощай ведь удалил!"];
        }
        else {
            [self showMessage:@"Ты мне всю статистику испотил :("];
        }
    }
}

```

```

        [self.managedObjectContext deleteObject:[self.lists objectAtIndex:indexPath.row]];
        [self.managedObjectContext save:nil];
        NSFetchedRequest *fetchRequest = [NSFetchedRequest fetchRequestWithEntityName:@"Htask"];
        fetchRequest.sortDescriptors = [NSArray arrayWithObject:[NSSortDescriptor
sortDescriptorWithKey:@"created" ascending:YES]];
        self.lists = [self.managedObjectContext executeFetchRequest:fetchRequest error:nil];

        [tableView deleteRowsAtIndexPaths:@[indexPath] withRowAnimation:UITableViewRowAnimationFade];
    } else if (editingStyle == UITableViewCellEditingStyleInsert) {
        // Create a new instance of the appropriate class, insert it into the array, and add a new row to the table view
    }
    [self updateStatistic];
}

// Override to support rearranging the table view.
- (void)tableView:(UITableView *)tableView moveRowAtIndexPath:(NSIndexPath *)fromIndexPath
toIndexPath:(NSIndexPath *)toIndexPath
{
    Htask *tsk = [self.tasks objectAtIndex:fromIndexPath.row];
    [self.tasks removeObjectAtIndex:fromIndexPath.row];
    [self.tasks insertObject:tsk atIndex:toIndexPath.row];
    [self updateStatistic];
}

// Override to support conditional rearranging of the table view.
- (BOOL)tableView:(UITableView *)tableView canMoveRowAtIndexPath:(NSIndexPath *)indexPath
{
    // Return NO if you do not want the item to be re-orderable.
    return YES;
}

/*
#pragma mark - Navigation

// In a storyboard-based application, you will often want to do a little preparation before navigation
- (void)prepareForSegue:(UIStoryboardSegue *)segue sender:(id)sender
{
    // Get the new view controller using [segue destinationViewController].
    // Pass the selected object to the new view controller.
}
*/
#pragma event
- (void)editingBtn:(id)sender {
    self.editing = !self.editing;
    [self updateStatistic];
}
-(void)addTaskName:(NSString *)name deadline:(NSDate *)date type:(NSString *)type {
    Htask *newTask = [NSEntityDescription
insertNewObjectForEntityForName:@"Htask" inManagedObjectContext:self.managedObjectContext];
    newTask.created = [NSDate date];
    newTask.name = name;
    newTask.deadline = date;
    newTask.done = [NSNumber numberWithInt:NO];
    newTask.type = type;
    [self.managedObjectContext save:nil];
    self.lists = [self.lists arrayByAddingObject:newTask];
    [self.tableView reloadData];
    [self updateStatistic];
}

```

```

[self showMessage:@@"Ммм, новая задачка :");
}
-(void)updateStatistic{
    self.allTasks.text = [NSString stringWithFormat:@"%d",self.lists.count];
    self.comleteTasks.text = [NSString stringWithFormat:@"%d",0];
    NSInteger cnt = 0;
    Htask *model;
    for (NSInteger i = 0; i< self.lists.count; i++) {
        model = [self.lists objectAtIndex:i];
        BOOL fl = [model.done boolValue];
        if (fl) {
            cnt++;
        }
    }
    self.comleteTasks.text = [NSString stringWithFormat:@"%d",cnt];
}
}
-(void)showMessage:(NSString *)str{
    self.message.text =str;
    self.message.hidden = NO;
    [NSTimer scheduledTimerWithTimeInterval:2.5f target:self selector:@selector(hideMessage) userInfo:nil
    repeats:NO];
}
-(void)hideMessage{
    self.message.hidden = YES;
}
@end

```