

**ОТЧЕТ**

по производственной (преддипломной) практике

по специальности 09.02.03 «Программирование в компьютерных системах»

Выполнил студент гр. П2-16

Петренко А.С.

\_\_\_\_\_\_\_\_\_\_ (подпись)

Принял преподаватель

Гусятинер Л.Б

\_\_\_\_\_\_\_\_\_\_(подпись)

\_\_\_\_\_\_\_\_\_\_\_(оценка)

Королев, 2020

Содержание

[Введение 3](#_Toc40881611)

[Технико-экономическая характеристика 4](#_Toc40881612)

[1. Общие положения 4](#_Toc40881613)

[2. Организационная структура и управление 5](#_Toc40881614)

[3. Основные задачи 5](#_Toc40881615)

[4. Функции отдела 6](#_Toc40881616)

[5. Взаимодействие с другими подразделениями 7](#_Toc40881617)

[6. Права и обязанности работников отдела 8](#_Toc40881618)

[7. Ответственность руководителя отдела 9](#_Toc40881619)

[8. Структура отдела 10](#_Toc40881620)

[Аппаратное обеспечение отдела ПО 11](#_Toc40881621)

[Программное обеспечение отдела ПО 12](#_Toc40881622)

[Методы проектирования и разработки 13](#_Toc40881623)

[Метод декомпозиции. 13](#_Toc40881624)

[Метод итерации. 14](#_Toc40881625)

[Каскадная модель разработки ПО 15](#_Toc40881626)

[Математическая постановка задачи 17](#_Toc40881627)

[Программные решения 20](#_Toc40881628)

[1. HSE App 20](#_Toc40881629)

[2. МГИМО 22](#_Toc40881630)

[3. ВШБ МГУ 23](#_Toc40881631)

[Технологии обработки данных 24](#_Toc40881632)

[Алгоритм решения 25](#_Toc40881633)

[Выбор инструментов 26](#_Toc40881634)

[Тестовые наборы к программе 28](#_Toc40881635)

[Описание главного модуля программы 29](#_Toc40881636)

[Руководство оператора 30](#_Toc40881637)

[Техника безопасности 31](#_Toc40881638)

[Источники 33](#_Toc40881639)

[Дневник практики 36](#_Toc40881640)

# Введение

На 4 курсе обучения в ККМТ, мной была пройдена преддипломная практика. Студент получил задание на разработку кросс платформенного мобильного приложения для студентов и преподавателей Технологического университета МГОТУ и его подразделений, а так же изучить учреждение на котором проходит практику

# Технико-экономическая характеристика

## Общие положения

1.1. Отдел программного обеспечения (далее - Отдел) Государственного бюджетного образовательного учреждения высшего образования Московской области «Технологический университет» (далее – Университет) является самостоятельным структурным подразделением.

1.2. Настоящее Положение регулирует деятельность Отдела, определяет его задачи, функции, права и обязанности.

1.3. Координацию деятельности Отдела осуществляет проректор по информационным технологиям.

1.4. В своей деятельности Отдел руководствуется законодательством Российской Федерации в сфере образования, Уставом и Правилами внутреннего трудового распорядка, установленными в Университете, приказами и распоряжениями ректора, а также настоящим Положением.

1.5. Условия труда работников Отдела определяются Правилами внутреннего трудового распорядка, установленными в Университете, трудовыми договорами, заключаемыми с каждым работником, и должностными инструкциями.

1.6. Отдел входит в состав службы проректора по информационным технологиям.

1.7. Должностные обязанности работников Отдела, включая начальника Отдела, определяются должностными инструкциями, утверждаемыми ректором Университета.

1.8. Отдел создается, реорганизуется и ликвидируется приказом ректора Университета.

## Организационная структура и управление

2.1. Структура и штат сотрудников Отдела, а также их изменения утверждаются приказом ректора Университета по представлению проректора по информационным технологиям.

2.2. Руководство Отделом осуществляет начальник отдела, который непосредственно подчиняется проректору по информационным технологиям.

2.3. Работники Отдела выполняют свои обязанности согласно должностным инструкциям и трудовым договорам.

2.4. В период временного (болезнь, отпуск, командировка) отсутствия начальника Отдела его обязанности могут быть возложены на заместителя начальника отдела, который несет ответственность за надлежащее их исполнение.

2.5. Прием на работу и освобождение от занимаемых должностей осуществляется приказом ректора Университета в соответствии с трудовым законодательством Российской Федерации, по представлению проректора по информационным технологиям.

2.6. Полномочия, права, обязанности, ответственность, требования к квалификации сотрудников Отдела приведены в соответствующих должностных инструкциях сотрудников.

## Основные задачи

Основными задачами Отдела являются:

* 1. Организация освоения и применения новых программных и технических средств, информационных технологий в образовании, накопление и систематизация общего и тематического прикладного программного обеспечения.
  2. Консультирование пользователей Университета по вопросам приобретения и внедрения в образовательный процесс современного программного обеспечения.
  3. Осуществление деятельности, направленной на формирование и развитие корпоративной информационной среды Университета, а также ее интеграцию в российское и международное информационное пространство.
  4. Техническое сопровождение и развитие информационной инфраструктуры Университета, в том числе:
     + участие в распределении вычислительной техники и оргтехники по структурным подразделениям Университета;
     + участие в определении потребности Университета в новой вычислительной технике и оргтехнике;
     + участие в определении потребностей в новом прикладном программном обеспечении.

## Функции отдела

В соответствии с возложенными задачами Отдел осуществляет следующие функции:

4.1. Участвует в обеспечении информационно-технического сопровождения учебного процесса.

4.2. Участвует в освоении и внедрении в образовательный процесс, а также в структурных подразделениях Университета новых программных средств и информационных технологий.

4.3. Участвует в организации и проведении учебно-методических, научно-практических конференций, методических семинаров.

4.4. Осуществляет информационную поддержку структурных подразделений Университета по вопросам применения информационных и коммуникационных технологий.

4.5. Участвует в осуществлении технического обслуживания, администрировании компьютерных классов и мультимедийных аудиторий Университета.

4.6. Проводит консультации пользователей информационно-вычислительной системы Университета по вопросам использования компонентов системного программного обеспечения.

4.7. Организует доступ и обеспечивает использование информационных ресурсов Университета и внешних источников.

4.8. Осуществляет сетевое администрирование пользователей информационно-вычислительной системы Университета, в том числе распределение прав доступа пользователей к вычислительным и информационным ресурсам.

4.9. Осуществляет эксплуатацию, в том числе инсталляцию и настройку сетевого программного обеспечения для обеспечения согласованной работы компонентов информационно-вычислительной сети.

4.10. Осуществляет мелкий ремонт и техническое обслуживание вычислительной техники и оргтехники.

4.11. Участвует в оптимальном распределении вычислительной техники и оргтехники по структурным подразделениям в соответствии с решаемыми задачами.

## Взаимодействие с другими подразделениями

Отдел:

* 1. ***Получает***: достоверную первичную документацию и информацию от всех структурных подразделений Университета в установленные сроки.
  2. ***Предоставляет***: материалы руководителям структурных подразделений в соответствии с заявленными задачами, консультации по вопросам:
  + правил эксплуатации периферийных устройств;
  + эксплуатации персональных компьютеров, находящихся в структурных подразделениях Университета;
  + использования программного обеспечения, установленного на персональных компьютерах в учебных лабораториях;
  + по применению и правилам эксплуатации технических средств.

Реализует взаимодействие и сотрудничество с организациями, работающими в сфере информационных технологий.

## Права и обязанности работников отдела

* 1. Работники имеют права
     + повышать квалификацию с установленной периодичностью;
     + запрашивать от всех структурных подразделений и служб Университета достоверную исходную информацию и документацию;
     + осуществлять руководство структурными подразделениями и службами по использованию сетевых ресурсов, осуществлять контроль над соблюдением правил эксплуатации периферийных устройств;
     + вносить предложения по обновлению технической базы Отдела и учебных классов проректору по информационным технологиям.
  2. Обязанности работников
* соблюдать Правила внутреннего трудового распорядка Университета, правила по охране труда и противопожарной безопасности;
* строго следовать нормам профессиональной этики;
* обеспечивать необходимый профессиональный уровень своей деятельности, качественно выполнять должностные обязанности, указанные в их должностных инструкциях.

## Ответственность руководителя отдела

Руководитель отдела несет ответственность за:

* Ненадлежащее исполнение или неисполнение своих обязанностей, предусмотренных настоящей должностной инструкцией в пределах, определенных действующим трудовым законодательством Российской Федерации.
* Правонарушения, совершенные в процессе осуществления своей деятельности, – в пределах, определенных действующим административным, уголовным и гражданским законодательством Российской Федерации.
* Причинение материального ущерба — в пределах, определенных. действующим трудовым и гражданским законодательством Российской Федерации.

## Структура отдела



Рис. 1 Структура отдела

# Аппаратное обеспечение отдела ПО

Аппаратное обеспечение отдела ПО состоит из:

* принтер PIXMA TR4540;
* маршрутизатор ASUS DSL-AC68U;
* 5 персональных компьютеров;
* сервер Dell PowerEdge T30.

Для всех компьютеров в отделе предусмотрена следующая конфигурация:

Характеристика ПК:

Процессор: Intel Core i3-4160

Материнская плата: GIGABYTE GA-B250

Видеокарта: Radeon RX 570

Оперативная память: 4Gb DDR4

Жесткий диск: 480 Gb

БП: 500W

Комплектующие:

Монитор: NEC PA311D

Клавиатура: Ritmix RKB-155

Мышь: Genius DX-120

# Программное обеспечение отдела ПО

Набор программного обеспечения отдела ПО приведён ниже:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *№* | *Наименование* | *Количество* | *Оборудование* | *Стоимость* |
| 1 | MS Windows 10 Pro | 5 | ПК | 85000р. |
| 2 | MS Office 365 | 5 | ПК | 1250р/год |
| 3 | TeamViewer | 5 | ПК | 7299р/месяц |
| 4 | Adobe Acrobat DC | 5 | ПК | 4380р/месяц |
| 5 | Google Chrome | 5 | ПК | Бесплатно |
| 6 | 7zip | 5 | ПК | Бесплатно |
| 7 | DrWeb Enterprise | 5 | ПК/Сервер | 13500р/год |
| 8 | MS Windows Server 2019 | 1 | Сервер | 71000р |
| 8 | Visual Studio 2019 Community | 3 | ПК | Бесплатно |
| 8 | Filezilla | 3 | ПК | Бесплатно |

Вебсайты продуктов:

|  |  |  |
| --- | --- | --- |
| *№* | *Наименование* | *Сайт* |
| 1 | MS Windows 10 Pro | <https://www.microsoft.com/> |
| 2 | MS Office 365 | <https://www.microsoft.com/> |
| 3 | TeamViewer | <https://www.teamviewer.com/> |
| 4 | Adobe Acrobat DC | <https://www.adobe.com/> |
| 5 | Google Chrome | <https://www.google.com/chrome/> |
| 6 | 7zip | <https://www.7-zip.org/> |
| 7 | DrWeb | <https://www.drweb.ru/> |
| 8 | MS Windows Server 2019 | <https://www.microsoft.com/> |
| 8 | Visual Studio 2019 Community | <https://visualstudio.microsoft.com/ru/vs/> |
| 8 | Filezilla | <https://filezilla-project.org/> |

# Методы проектирования и разработки

## Метод декомпозиции.

Данный метод подразумевает, что любой объект будет рассматриваться как сложную систему, состоящую из нескольких подсистем, которые в свою очередь также могут быть поделены на отдельные части.

Конкретно в данном проекте декомпозиция будет использоваться для разработки отдельных частей одной системы. Это влечёт собой следующие преимущества:

* Уменьшается время, занимаемое на разработку:

Это позволяет разбить исходный код программы на отдельные модули, которые могут разрабатываться несколькими разработчиками параллельно, причём реализация одного модуля не будет зависеть от реализации другого модуля. Так же готовые модули одной программы можно использовать при разработке другой.

* Повышается надёжность программы:

В случае возникновения ошибки, разработчикам не приходится тратить большое количество времени на поиск места возникшей ошибки, она будет локализована в том модуле, в котором она возникла. По этой же причине отладка программы, которая состоит из отдельных модулей, становится менее трудоёмкой.

В данном проекте этот метод будет применяться при разработке программы. Программа будет иметь графический интерфейс, который будет реагировать на действия пользователя посредством вызова выполнения определённых команд. Например, если пользователь нажмёт на кнопку вывода списка своих одногруппников, то произойдёт вызов метода, который будет получать список одногруппников для текущего пользователя. После чего список будет выведен в интерфейс программы в удобном для восприятия виде.

## Метод итерации.

Данная методика решает проблемы неточностей и противоречий исходных данных в процессе проектирования.

В случае долгой разработки приложения, некоторые из модулей могут некорректно работать с другими модулями, а также могут появляться дополнительные условия работы ранее проектируемых модулей.

Методика решения данных проблем состоит в выполнении итерационных процедур. Первый цикл итерации решения поставленной задачи состоит в взятии в расчёт ограниченное число исходных данных и учитываемых факторов. С последующими циклами итераций мы возвращаемся в начало поставленной задачи и повторяем её решение, но уже с уточнёнными значениями исходных данных и т.д. до тех пор, пока поставленная задача не будет выполнена.

Данный метод разработки будет применяться в процессе реструктуризации исходного кода программы (далее рефакторинг). После реализации нового функционала программы, проводится исследование структуры исходного кода на дублирование функций, временные решения, а также делается тестирование программы и поиск возникших ошибок. После чего принимается решение, стоит ли проводить рефакторинг или нет. В процессе рефакторинга методы могут быть переписаны заново, с учётом нового функционала. Это позволяет:

* Упростить дальнейшую разработку и сопровождение программы: по возможности новые методы должны использовать общие функции и классы программы.
* Упросить тестирование и отладку программы: в случае возникновения ошибки в работе программы легче понять, где она возникла, и одновременно с этим она будет локализована в том программном модуле, в котором она возникла.

## Каскадная модель разработки ПО

Каскадная модель (или водопадная) — модель процесса разработки программного обеспечения, в которой процесс разработки выглядит как поток, последовательно проходящий фазы анализа требований, проектирования, реализации, тестирования, интеграции и поддержки.

В исходной каскадной модели следующие фазы идут в таком порядке:

1. Определение требований
2. Проектирование
3. Конструирование (также «реализация» либо «кодирование»)
4. Воплощение
5. Тестирование и отладка (также «***верификация***»)
6. Развёртывание программного продукта
7. Поддержка

В проекте этот метод разработки будет использоваться «как есть». Сначала будут определены требования к программе и задачи, которые будут решаться данным проектом. Затем будет спроектирована структура будущего программного продукта. После согласования структуры, начнётся реализация функционала, которая постепенно перейдёт в воплощение программы. Дальше будет производиться тестирование и отладка для выявления ошибок и недочётов как в программной логике, так и в визуальном интерфейсе. Потом последует развёртывание программы на устройства конечных пользователей, Предположительно в дальнейшем программа будет расширяться, и будет добавляться новый функционал.

# Математическая постановка задачи

Для защиты при обмене данными между нашим приложением (Client) и сервером, мы использовали алгоритм шифрования RSA

**Алгоритм RSA**

RSA (аббревиатура от фамилий Rivest, Shamir и Adleman) — криптографический алгоритм с открытым ключом, основывающийся на вычислительной сложности задачи факторизации больших целых чисел. Алгоритм RSA стал первым алгоритмом, пригодным и для шифрования, и для цифровой подписи.

В данном алгоритме имеется открытый ключ и закрытый ключ. Работа алгоритма происходит следующим образом:

Осуществляется генерация ключей: выбираются два достаточно больших случайных простых числа (желательно разрядностью 100-200 единиц или больше). Для большей безопасности ключи должны иметь равную длину.

p = 3557

q = 2579

Затем вычисляется произведение N = p \* q.

N = 3557 \* 2579

N = 9173503

После рассчитывается значение функции Эйлера по формуле:

φ(n) = (p-1)\*(q-1)

φ(n) = (3557-1)\*(2579-1) = 9167365

Далее выбирается открытый ключ(открытая экспонента) e (1 < e < φ(n)), взаимно простое со значением функции Эйлера

e = 3

Следом с помощью расширенного алгоритма Евклида вычисляется закрытый ключ шифрования D(секретная экспонента), удовлетворяющий условию:

**e \*** D≡1mod **(**φ(n))

D = 6111579

Заметим, что D и N также взаимно простые числа.

Числа E и N – это открытые ключи, а число D – закрытый.

Два простых числа p и q больше не нужны. Они могут быть отброшены, но не должны быть раскрыты.

При шифровании сообщение M сначала разбивается на цифровые блоки, размерами меньше N (для двоичных данных выбирается самая большая степень числа 2, меньшая N). Зашифрованное сообщение С будет состоять из блоков Ci такой же самой длины.

Предположим текст для шифрования M = 111111

Формула шифрования выглядит так:

Ci = E(Mi) = Mie mod(N)

C = 1111113 mod(9173503) = 4051753

При расшифровке сообщения для каждого зашифрованного блока Сi вычисляется по следующей формуле:

Mi = D(Ci) = Cid mod(N)

M = 40517536111579 mod (9173503) = 111111

# Программные решения

## HSE App

HSE App – мобильное приложение для студентов и преподавателей НИУ ВШЭ, которое позволяет быстро узнать расписание, найти свободную аудиторию в любом здании Вышки, следить за дедлайнами вместе с одногруппниками. Сайт приложения в Google play и App store: <https://play.google.com/store/apps/details?id=com.chichkanov.hseapp>

<https://apps.apple.com/ru/app/hse-app/id1286305049>

Функционал, представленный на сайте:

• Расписание из РУЗ НИУ ВШЭ: узнавайте время, преподавателя, аудиторию, сведения о потоке, дате добавления и изменения информации о паре.

• Продвинутый поиск: ищете расписание студентов, преподавателей, аудиторий и групп.

• Создавайте дедлайны и делитесь ими с одногруппниками, чтобы никогда не пропускать самое важное.

• Фильтр свободных аудиторий: выбирайте день, здание и интервал времени, в котором вам необходима свободная аудитория.

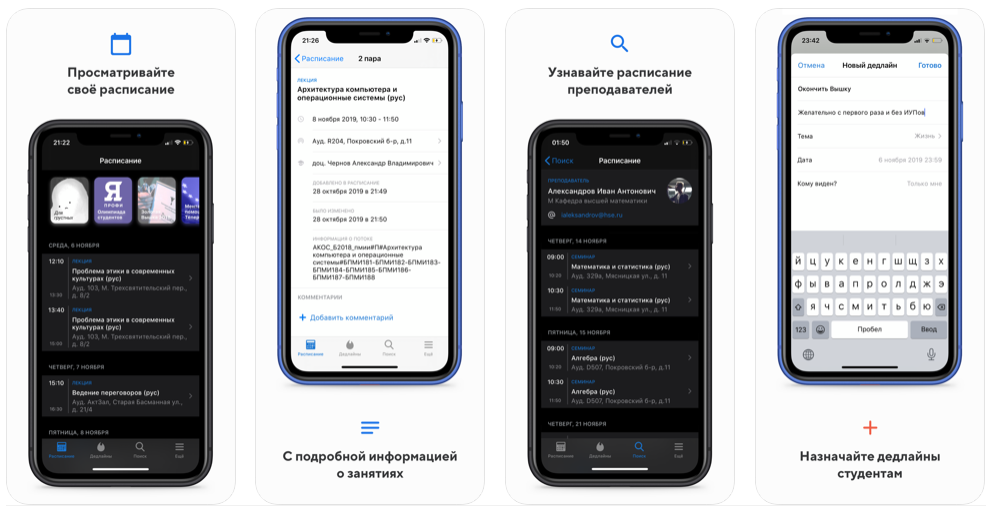


Рис. 2 Интерфейс программы HSE App

## МГИМО

«МГИМО» – мобильное приложение для студентов и преподавателей МГИМО (Московский Государственный Институт Международных Отношений). Обладает по большей части схожим функционалом, но в добавление к этому позволяет узнать информацию о ректорате этого института. Сайт приложения в Google play и App store:

<https://play.google.com/store/apps/details?id=com.genisoft.launcher55193>

<https://apps.apple.com/ru/app/%D0%BC%D0%B3%D0%B8%D0%BC%D0%BE/id1209989995>

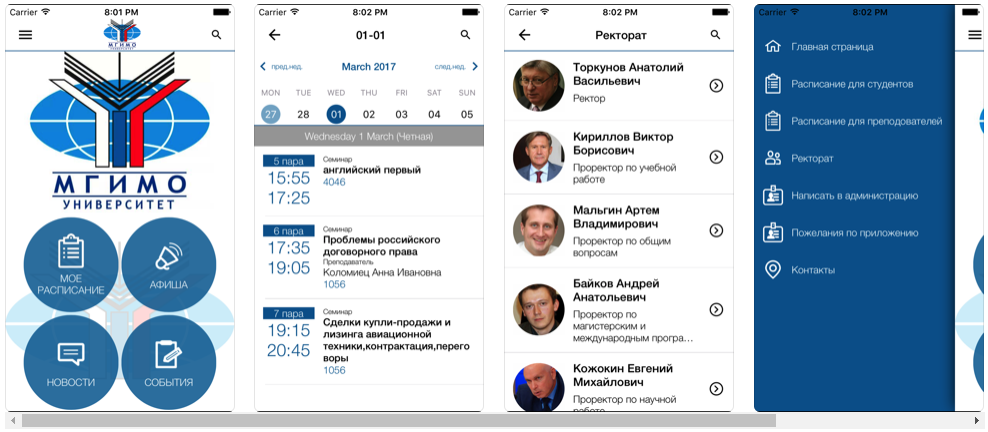


Рис. 3 Интерфейс программы МГИМО

## ВШБ МГУ

ВШБ МГУ - Приложение для студентов Высшей школы бизнеса. Приложение представляет из себя инструмент для более быстрого и оперативного оповещения студентов. Изменения в расписание. Оповещение о предстоящих мероприятиях. Сайт приложения в Google play и App store:

<https://play.google.com/store/apps/details?id=link.yappix.app.vshb>

<https://apps.apple.com/ru/app/%D0%B2%D1%88%D0%B1-%D0%BC%D0%B3%D1%83/id1179073910>

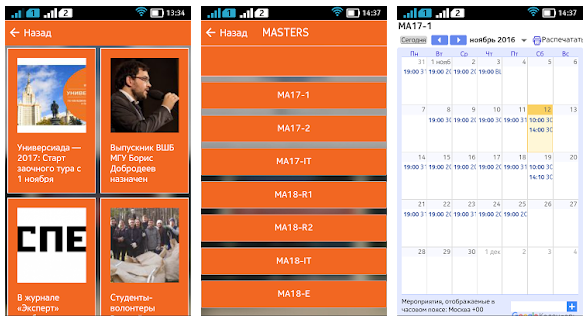


Рис. 4 Интерфейс программы ВШБ МГУ

# Технологии обработки данных

Используя http запросы мы получаем данные в формате Json строки.

Перед выводом в графический интерфейс полученный из запроса Json конвертируется в экземпляр требуемого класса.

Примеры приходящих Json:

После удачной аутентификации в подтверждение сервер отправляет информация о пользователе вида:  
{

"id":"20100",

"pin":"26627",

"fio":"Свистов Антон Олегович",

"admin\_auth":"0",

"editor\_auth":"0",

"user\_type":"2",

"first\_auth":"0",

"email":"[svistov.anton@mail.ru](mailto:svistov.anton@mail.ru)",

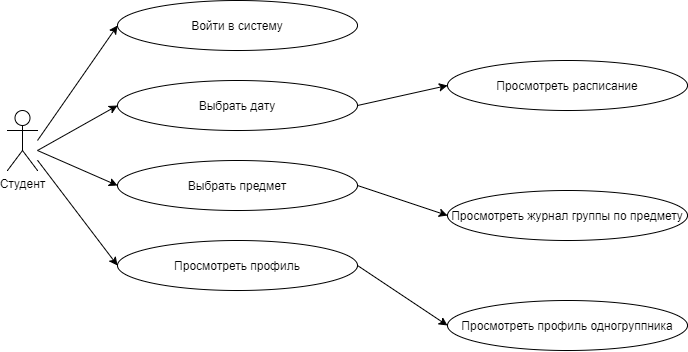
"allowed":"1",

"avatar":"Mfp8xR1F4sGv.jpg"

}

Позже эти данные используются для отображения профиля залогиненного пользователя.

# Алгоритм решения

На Рис. 5 изображена Use Case диаграмма для программного продукта, который будет разработан в ходе выполнения дипломной работы: Рис 5. Use Case диаграмма будущей программы

# Выбор инструментов

Выбор того или иного инструмента должен чем-то обуславливаться. Поэтому, что бы понять, как выбрать инструмент, необходимо выделить ключевые требования, по которым инструмент и будет подбираться.

Требования к продукту и его разработке:

* Кроссплатформенность. Приложение должно запускаться как на Android, так и на IOS.
* Время на разработку, отладку и тестирование. 3-4 месяца.
* Система контроля версий. Над проектом работает 3 программиста, поэтому система контроля версий необходима.

Язык программирования и технология для разработки мобильных приложений.

Был выбран язык программирования C# и технология разработки кроссплатформенных мобильных приложений Xamarin.Froms. Язык был определён сразу потому, что он хорошо знаком всем программистам в команде и потому, что платформа Xamarin.Forms работает только на C#.

Xamarin.Forms позволяет писать один код, который будет работать на всех выбранных платформах. Исключение составляют лишь платформозависимый код (работа с файлами, системные настройки и т.д.) и комплексный интерфейс. За счёт небольшого количества платформозависимого кода время на разработку сокращается.

Среда разработки

Где C# там и Visual Studio. Эта IDE, а именно Visual Studio Community 2019 является бесплатной в использовании и хорошо знакома всем программистам в команде. В этой среде очень удобно работать с различными библиотеками/фреймворками. Установить тот же Xamarin.Forms со всеми необходимыми инструментами для разработки и отладки можно прямо из среды.

Система контроля версий

Выбор системы контроля версий тоже оказался не таким сложным. Их в принципе не сильно много. На вскидку можно привести git, bitbucket, svn. Выбор пал на git (github, не gitlab) потому что он, опять же, хорошо знаком всем программистам в команде, у всех есть учётные записи и опыт работы с github.

# Тестовые наборы к программе

Тестовый набор №1.

Логин: «kekekekeke»

Пароль: «parol»

Ожидаемый результат: неудачная авторизация через программу и через сайт университета. Отличие результата будет расценено как ошибка.

Тестовый набор №2.

Логин: логин разработчика приложения

Пароль: пароль разработчика приложения

Ожидаемый результат: Успешный вход в учётную запись через программу и через сайт. Отличие результата будет расценено как ошибка.

Примечание: под логином и паролем разработчика подразумевается действующий логин и пароль действующей учётной записи разработчика.

Тестовый сценарий работы: На устройстве отключают связь с сетью Интернет, затем запускают тестируемую программу.

Ожидаемый результат: Вывод сообщения об отсутствующем подключении к сети Интернет, с предложением повторения попытки подключения.

Тестовый сценарий работы: На устройстве отключают связь с сетью Интернет, затем запускают тестируемую программу. После вывода соответствующего сообщения, связь с сетью Интернет включают и повторяют попытку.

Ожидаемый результат: Появление на экране устройства формы ввода и пароля пользователя. Если до проверки текущего сценария работы вход в учётную запись через программу был успешен, допускается автоматический вход в учётную запись пользователя.

# Описание главного модуля программы

На Рис. 6 показана структура главного модуля будущей программы и взаимосвязь экранных форм программы между собой.

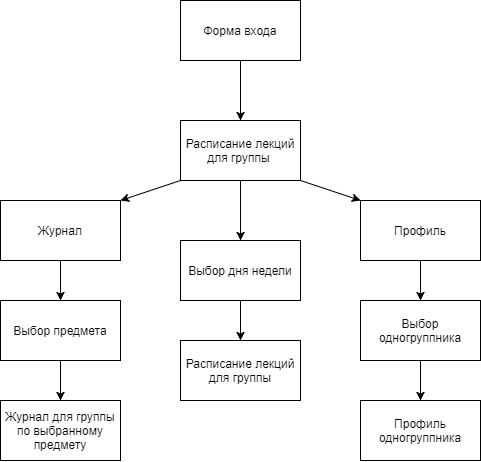


Рис. 6. Структура главного модуля программы.

# Руководство оператора

Руководство оператора приведено в файле «Руководство оператора.docx» в папке «п15. Руководство оператора».

# Техника безопасности

Техника безопасности приведена в файле «Техника безопасности.docx» в папке «п15. Техника безопасности».

# Источники

* Антамошкин, О. А. Программная инженерия. Теория и практика [Электронный ресурс] : учебник / О. А. Антамошкин. -

Красноярск: Сиб. Федер. ун-т, 2012. - 247 с. - ISBN 978-5-7638-2511-4. - Текст : электронный. -

URL: <https://znanium.com/catalog/document?pid=492527>

* Введение в программную инженерию : Учебник / В.А. Антипов, А.А. Бубнов, А.Н. Пылькин, В.К. Столчнев. —

Москва : КУРС: ИНФРА-М, 2019. — 336 с. - ISBN 978-5-16-103172-8. - Текст : электронный. -

URL: <https://znanium.com/catalog/document?pid=1035160>

* Гагарина, Л. Г. Разработка и эксплуатация автоматизированных информационных систем : учеб. пособие / Л.Г. Гагарина. —

Москва : ИД «ФОРУМ» : ИНФРА-М, 2019. — 384 с. — (Среднее профессиональное образование). -

ISBN 978-5-16-106202-9. - Текст : электронный. -

URL: <https://znanium.com/catalog/document?pid=1003025>

* Гагарина, Л. Г. Технология разработки программного обеспечения : учеб. пособие /

Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Сидорова-Виснадул ; под ред. Л.Г. Гагариной. —

Москва : ИД «ФОРУМ» : ИНФРА-М, 2019. — 400 с. — (Высшее образование: Бакалавриат). -

ISBN 978-5-16-104071-3. - Текст : электронный. -

URL: <https://znanium.com/catalog/document?pid=1011120>

* Управление разработкой программного обеспечения

Альберт Нургалиев

URL: <https://stepik.org/course/56013/>

* Проектирование информационных систем

Владислав Тарасенко

URL: <https://stepik.org/course/63510>

# Дневник практики

|  |  |  |
| --- | --- | --- |
| **Дата** | **Содержание работ** | **Отметка о выполнении** |
| 27.04 | Изучение и обсуждение официальных документов |  |
| 28.04 | Проведение инструктажа по технике безопасности |  |
| 28.04 | Выдача заданий на преддипломную практику |  |
| 30.04 | Технико-экономическая характеристика объекта практики |  |
| 12.05 | Характеристика аппаратного обеспечения предприятия |  |
| 12.05 | Характеристика программного обеспечения предприятия |  |
| 13.05 | Методы проектирования и разработки |  |
| 14.05 | Математическая постановка задачи |  |
| 14.05 | Программные решения |  |
| 15.05 | Существующие технологии обработки данных |  |
| 18.05 | Консультация по описанию алгоритмов и решения |  |
| 18.05 | Консультация по основанию выбора инструментов, средств и языков |  |
| 19.05 | Консультация по описанию тестовых сценарию и тестовых наборов |  |
| 19.05 | Консультация по разработке и описанию структуры главного модуля |  |
| 20.05 | Консультация по разработке руководства оператора |  |
| 21.05 | Консультация по разделу техники безопасности |  |
| 21.05 | Консультация по общему оформлению записки |  |
| 22.05 | Консультация по созданию презентаций |  |
| 22.05 | Сдача отчётов по практике |  |
| 23.05 | Защита отчётов по практике |  |