**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра МОЭВМ**

**КУРСОВАЯ РАБОТА**

**по дисциплине «Объектно-ориентированные технологии разработки программного обеспечения»**

**Тема: Разработка программы для составления расписания**

|  |  |  |
| --- | --- | --- |
| Студент гр. 5303 |  | Допира В.Е. |
| Преподаватель |  | Спицын А.В. |

Санкт-Петербург

2020**ЗАДАНИЕ**

**НА КУРСОВУЮ РАБОТУ**

|  |  |  |
| --- | --- | --- |
| Студент Допира В.Е. | | |
| Группа 5303 | | |
| Тема практики: Разработка программы для составления расписания | | |
| Исходные данные:  Сведения о предметах, группах, кабинетах, времени проведения занятий и академических часах в учебном заведении. Хранятся в формате JSON. | | |
| Содержание пояснительной записки:  «Содержание», «Введение», «», «», «», «», «Заключение», «Список использованных источников» | | |
| Предполагаемый объем пояснительной записки:  Не менее страниц. | | |
| Дата выдачи задания: 10.02.2020 | | |
| Дата сдачи реферата: 04.06.2020 | | |
| Дата защиты реферата: 04.06.2020 | | |
| Студент |  | Допира В.Е. |
| Преподаватель |  | Спицын А.В. |

**АННОТАЦИЯ**

**SUMMARY**

**СОДЕРЖАНИЕ**

|  |  |  |
| --- | --- | --- |
|  | Введение | 5 |
| 1. | Укрупненное описание постановки задачи | 6 |
| 1.1. | Определения и термины | 6 |
| 1.2. | Постановка задачи | 8 |
| 2. | Обзор литературы | 10 |
| 2.1. | Организация виртуальной памяти в ядре | 10 |
| 2.2. | Управления памятью в Android | 12 |
| 2.2. | Linux API | 13 |
| 3. | План работы на весенний семестр | 15 |
| 4. | Участие в учебном процессе бакалавров | 16 |
|  | Заключение | 17 |
|  | Список использованных источников | 18 |

**ВВЕДЕНИЕ**

Переход Российской Федерации на путь инновационной и инвестиционной стратегии развития экономики, повышение качества услуг и разрабатываемых технологий требует существенных изменений в организации управления деятельностью хозяйствующих субъектов. Учебные заведения, а в частности и университеты, не могут быть в стороне от этих процессов и должны соответствовать потребностям развития отраслей экономики регионов. В настоящее время необходимым условием для успешного функционирования образовательных учреждений становится разработка, внедрение и сопровождение информационных систем, обеспечивающих эффективное функционирование всех внутренних процессов.

Использование информационных систем в образовательных учреждениях активно внедряется. Спектр применения информационных технологий широк и варьируется от автоматизации отдельно взятых областей до полной автоматизации деятельности учебного заведения.

Вне зависимости от области автоматизации, внедряемые информационные системы имеют конечную цель: повышение качества образования. Значительное влияние на автоматизацию учебных процессов оказывают наличие денежных средств, готовность использования предлагаемых рынком информационных услуг и программных продуктов.

Составление расписания является сложной и требующей большой ответственности от сотрудников университета, выполняющих эту обязанность. Из-за большого количества студентов составление расписания отнимает большое количество времени. Автоматизация составления расписания занятий позволит сократить время и оптимизировать работу, поэтому выбранная тема актуальна.

Целью курсовой работы является разработка приложения для составления расписания и визуализация полученного результата.

Поставлены следующие задачи:

1. Обеспечение минимальных усилий пользователя при составлении расписания.

2. Создание удобного интерфейса.

3. Обеспечение быстрого доступа к данным, хранящимся в программе.

4. Обеспечение возможности добавления, изменения и удаления данных в программе.

5. Отображение расписание в виде многодольного графа.

6. Обеспечение сериализации данных при работе с программой.

# 1. Укрупненное описание постановки задачи

## 1.1. Определения и термины

Расписание занятий - документ, определяющий педагогически целесообразную последовательность учебных занятий в образовательном учреждении на каждый день учебной недели и конкретизирующий таким образом учебный план.

Учебное заведение — учреждение для (чего) обучения, организация образования. К ним относятся: школы, колледжи, пансионы, интернаты, гимназии, лицеи, техникумы, политехникумы, семинарии, медресе, курсы, институты, университеты.

Учебная дисциплина (предмет) - система знаний, умений и навыков, отобранных из определенной отрасли науки, техники, искусства, производственной деятельности для изучения в образовательном учреждении.

Преподаватель - работник высших, средних специальных и профессионально-технических учебных заведений, ведущий какой-либо предмет и воспитательную работу.

Факультет — отделение высшего учебного заведения, обнимающее науки, относящиеся к одной какой нибудь отрасли знаний.

Деканат — административно учебное управление факультета при декане.

Учебная группа (группа) - определенное число лиц с примерно одинаковым уровнем подготовки, изучающих одно и то же в одно и то же время под руководством одних и тех же преподавателей на протяжении одинакового для всех периода; обособленная часть контингента образовательного учреждения, являющаяся для ее членов первичным коллективом.

Учебное помещение (кабинет) - специально оборудованное помещение, для обучения студентов. Обычно номер кабинета имеет четырехзначное значение и нумеруется по типу: КЭНН, где К - номер корпуса, Э - номер этажа в определенном корпусе, Н - номер кабинета на определенных этажах и корпусах.

Этаж - уровень здания над (или под) уровнем земли.

Корпус - Одно из нескольких зданий, принадлежащих учебному заведению.

Учебная неделя - дни, по которым проходят занятия по определенным дисциплинам. Воскресенье является выходным днем.

Академический час - традиционное название учебного часа в вузах и других учреждениях профессионального образования. Академический час не равен астрономическому и устанавливается нормативными документами. Обычно равен 45 минутам.

Время начала лекций - время в которое проходят занятия по учебным дисциплинам.

Семестр - в высших учебных заведениях: полугодие учебного года; различают осеннее и весеннее.

Четность недель - способность номера недели, который является целым числом, делиться без остатка на 2. Принято считать учебные недели с начала четверти/семестра. В зависимости от четности недели расписание занятий может быть разным.

Многодольный граф - [граф](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84_(%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)), множество вершин которого можно разбить на k независимых множеств (доль).

Вершина графа (узел, точка) - элемент графа, обозначающий объект любой природы, входящий в множество объектов, описываемое графом.

Ребро графа - термин теории графов, линия, соединяющая пару смежных вершин графа.

Qt - кроссплатформенный фреймворк для разработки программного обеспечения на языке программирования C++.

С++ - компилируемый, статически типизированный язык программирования общего назначения.

# 1.2. Область применения

Одна из основных составляющих учебного процесса - расписание занятий. Составленное расписание влияет на трудовой ритм, творческую отдачу преподавателей, поэтому его можно рассматривать как фактор оптимизации использования ограниченных трудовых ресурсов: преподавательского состава. Технологию же разработки расписания следует воспринимать не только как трудоемкий технический процесс, объект механизации и автоматизации с использованием ЭВМ, но и как акцию оптимального управления. Таким образом, это проблема разработки оптимальных расписаний занятий с очевидным экономическим эффектом, поскольку интересы участников учебного процесса многообразны, задача составления расписания многокритериальная.

Представляемый программный продукт предназначен для облегчения ручного и умственного труда сотрудников при составлении учебного расписания. Использование данного продукта позволит сэкономить время и силы потраченные на выполнение этого трудоёмкого процесса.

# 1.3. Позиционирование

Постановка проблемы представлена в табл. 1.

Таблица 1 - Постановка проблемы

|  |  |
| --- | --- |
| Проблема | Повышение сложности составления расписания |
| На кого влияет | Составители расписаний в школах/Вузах |
| Как влияет | Увеличиваются временные затраты на составления расписания |
| Решение | Создать приложение, позволяющее автоматически составить расписание по заданным требованиям |

# Обзор продукта

Данное приложение предназначено для оптимизации процесса составления расписания в образовательных учреждениях.

Данный программный продукт может использоваться без привязки к определенному учебному заведению. Программный продукт будет распространяться бесплатно под лицензией MIT.

Пользователи:

* Составители расписаний
* Студенты
* Преподаватели
* Рекламодатели

# Особенности продукта

Данная программа будет предоставлять два режима отображения расписания:

* В виде таблицы;
* В виде многодольного графа.

Разрабатываемый программный продукт будет иметь следующие возможности:

1. Обеспечение оперативного доступа к данным, хранящимся в программе.

2. Обеспечение возможности изменения и дополнения данных в программе.

3. Отображение расписание в виде многодольного графа.

4. Обеспечение сериализации данных при работе с программой.

5. Кроссплатформенность: разрабатывается под Windows и Linux.

Разрабатываемый программный продукт имеет следующие ограничения:

1. Расписание составляется один раз без предоставления альтернативных вариантов.

**1.4. План разработки программного обеспечения**

Этот план будет обновляться по ходу разработки продукта. Целевые даты окончания каждого этапа показаны ниже (см. табл. 2).

Таблица 2 - Даты окончания каждого этапа

|  |  |
| --- | --- |
| ***Название этапа*** | ***Дата окончания*** |
| Начало (Inception) | 03.05.2020 |
| Уточнение (Elaboration) | 15.05.2020 |
| Построение (Construction) | 23.05.2020 |
| Внедрение (Transition) | 31.05.2020 |

Распределение ролей представлено в таблице 3.

Таблица 3 - Распределение ролей

|  |  |
| --- | --- |
| **Роль** | **Фамилия** |
| Technical writer | Допира Валерия |
| Implementer | Бочкарев Иван |
| Design Reviewer | Бочкарев Иван |
| System Analyst | Павлов Данила |
| Deployment Manager | Павлов Данила |
| Configuration Manager | Бочкарев Иван |
| Implementer | Ильянов Вячеслав |
| Designer | Павлов Данила |
| Test Analyst | Бочкарев Иван |
| Tool Specialist | Допира Валерия |
| Tester | Ильянов Вячеслав |
| Integrator | Допира Валерия |
| Design Reviewer | Ильянов Вячеслав |
| System Analyst | Ильянов Вячеслав |
| Deployment Manager | Допира Валерия |
| Configuration Manager | Павлов Данила |
| Developer of requirements | Допира Валерия |
| Designer | Допира Валерия |
| Developer of architecture | Допира Валерия |
| Implementer | Павлов Данила |
| Integrator | Ильянов Вячеслав |
| Tester | Бочкарев Иван |
| Tester | Допира Валерия |
| Developer of requirements | Павлов Данила |

Аналитики отвечают за обеспечение пригодности требований к тестированию и за ясность формулировок требований к выполняемым тестам. Разработчики должны помнить о тестировании при разработке приложений и нести ответственность за тестирование собственного кода.

Руководители должны обеспечить наличие планов тестирования и ресурсов, необходимых для формирования среды тестирования и выполнения необходимых тестов.

Испытатели - это эксперты по качеству. Они отвечают за все тестирование продукта (включая функциональное тестирование, тестирование системы и тестирование производительности) и должны лучше всех понимать, что такое качество и как его достичь.

Распределение ролей и обязанностей исполнителям представлено ниже (см. табл. 4).

Таблица 4 – Роли и обязанности

|  |  |  |
| --- | --- | --- |
| ***ФИО*** | ***Роль*** | ***Обязанность*** |
| Павлов Данила,  Ильянов Вячеслав | Аналитик | Устанавливает требования к разрабатываемой системе путем определения необходимой функциональности |
| Допира Валерия,  Павлов Данила | Разработчик требований | Формирует детальные требования к разрабатываемой системе |
| Допира Валерия,  Павлов Данила | Проектировщик, Дизайнер | Проектирование компонентов системы с учётом требований, заданных проектом |
| Допира Валерия | Разработчик архитектуры | Управление разработкой программной архитектуры системы, что включает продвижение и поддержку ключевых технических решений, задающих рамки реализации проекта |
| Бочкарев Иван, Вячеслав Ильянов,  Павлов Данила | Реализатор | Разработка программных компонентов и проверка их функциональности после интеграции в подсистемы |
| Бочкарев Иван, Ильянов Вячеслав | Рецензент дизайна | Защита качества дизайна, проверка удобства пользователя |
| Ильянов Вячеслав,  Допира Валерия | Интегратор | Специализация на объединении компонентных подсистем в единое целое и обеспечении совместной работы этих подсистем |
| Допира Валерия,  Бочкарев Иван,  Ильянов Вячеслав | Разработчик тестов | Выполнение тестирования продукта и описание исхода тестирования.  Определение общей стратегии тестирования и контроль успешности ее реализации |
| Бочкарев Иван | Аналитик тестирования | Защита качества тестирования, планирование ресурсов и управление ими, а также решение проблем, препятствующих процессу тестирования |
| Допира Валерия | Технический писатель | Составление отчетов, описание разработанного продукта |
| Павлов Данила,  Допира Валерия | Менеджер по развертыванию | Организация одновременного централизованного развертывания продукта на компьютерах, повышение рентабельности обслуживания клиентов и повышение прибыли предприятия |
| Бочкарев Иван,  Павлов Данила | Менеджер конфигураций | Управление наборами рабочих продуктов и их версиями |
| Допира Валерия | Руководитель проекта | Планирование затрат ресурсов, распределение ресурсов, выделение приоритетов, направление усилий коллектива |
| Спицын Александр Валентинович | Проверяющий | Оценивание планирования проекта и ценности рабочих продуктов на границах важнейших этапов жизненного цикла проекта |

Разработка системы будет производиться с использованием поэтапного подхода. Фазы и относительная временная шкала показаны ниже (см. табл. 5)

Таблица 5 – Фазы и относительная временная шкала

|  |  |  |
| --- | --- | --- |
| ***Фаза*** | ***Дата начала*** | ***Дата окончания*** |
| Начало | 1 неделя | 6 неделя |
| Уточнение | 7 неделя | 10 неделя |
| Построение | 11 неделя | 12 неделя |
| Внедрение | 13 неделя | 14 неделя |

Рабочие продукты получаемые после завершения определенного этапа представлены ниже (см. табл. 6)

Таблица 6 – Рабочие продукты каждого этапа

|  |  |  |
| --- | --- | --- |
| ***Фаза*** | ***Описание*** | ***Рабочие продукты*** |
| Начало | Определение содержания проекта, разработка основных сценариев | План разработки ПО  Видение  Глоссарий  Модель вариантов использования  Инфраструктура разработки |
| Уточнение | Создание стабильной формальной архитектуры для выполнения разработки ПО | План разработки ПО  Видение  Глоссарий  Модель вариантов использования  Инфраструктура разработки  Процесс разработки  Диаграмма вариантов использования, Диаграмма классов, Диаграмма пакетов, Диаграмма компонентов, Диаграмма последовательностей, Диаграмма состояний  Тестовый набор |
| Построение | Завершение разработки системы в соответствии с базовой архитектурой | План итерации  Разработка системы  Тестирование системы  Материалы поддержки пользователей  План развертывания |
| Внедрение | Обеспечение готовности программного обеспечения к представлению пользователям | План итерации  Демонстрация работы  системы  Компоновка продукта  Материалы поддержки пользователей  Элементы реализации |

Для иллюстрации плана, графика работ и занятости членов команды, работающих над проектом, удобно использовать диаграмму Ганта. Диаграмма Ганта со списком рабочих продуктов и исполнителями представлена ниже (см. рис. 1).

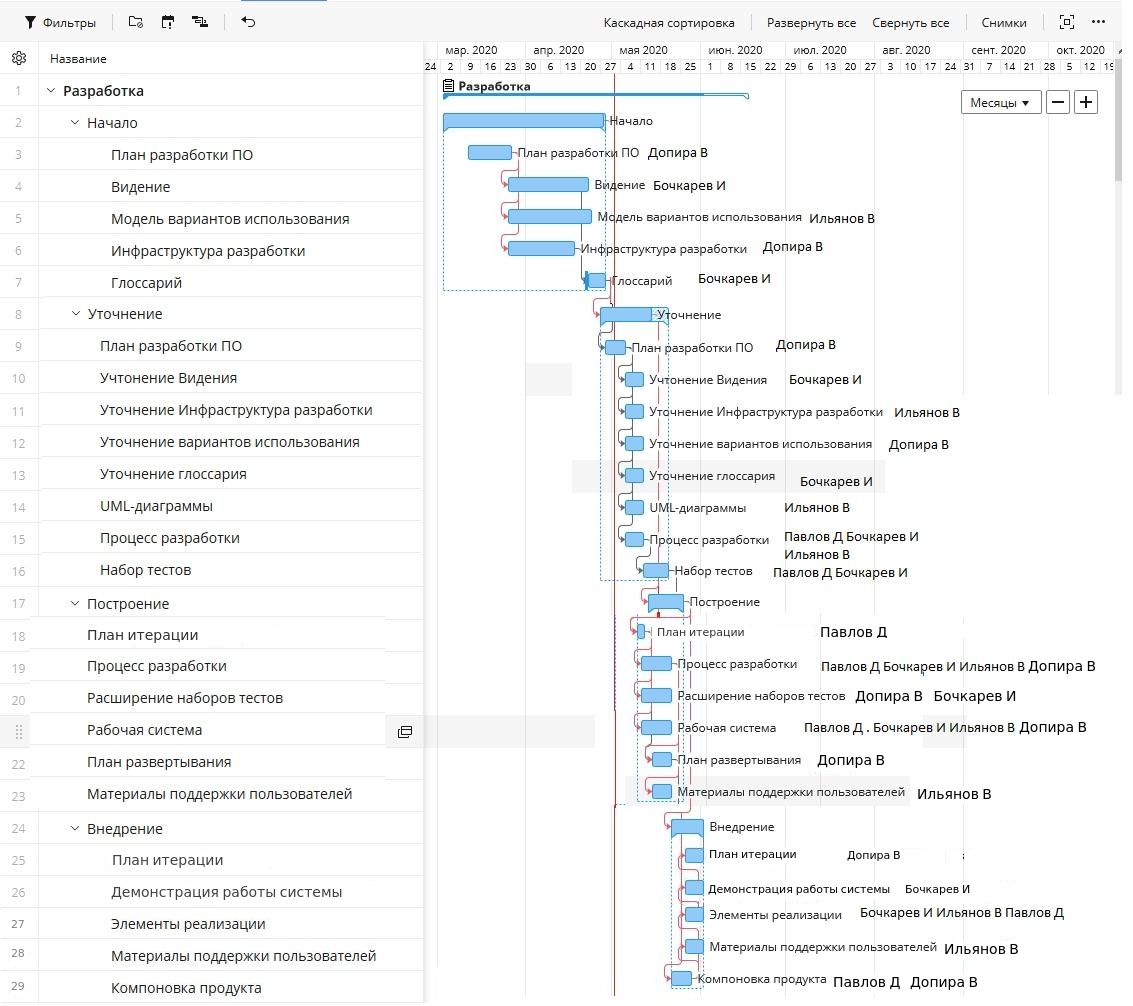


Рисунок 1 - Диаграмма Ганта

График сдачи каждой задачи проекта представлен ниже (см. табл. 7).

Таблица 7 – График проекта

|  |  |  |  |
| --- | --- | --- | --- |
| ***Фаза*** | ***Задача*** | ***Окончание работы*** | ***Исполнители*** |
| Начало | План разработки ПО | 26.03.2020 | Допира Валерия |
| Видение | 22.04.2020 | Бочкарев Иван |
| Модель вариантов использования | 22.04.2020 | Павлов Данила |
| Инфраструктура разработки | 22.04.2020 | Бочкарев Иван |
| Глоссарий | 27.04.2020 | Ильянов Вячеслав |
| Уточнение | План разработки ПО | 30.04.2020 | Допира Валерия |
| Видение | 04.05.2020 | Бочкарев Иван |
| Модель вариантов использования | 04.05.2020 | Павлов Данила |
| Инфраструктура разработки | 04.05.2020 | Бочкарев Иван |
| Глоссарий | 04.05.2020 | Ильянов Вячеслав |
| UML диаграммы | 04.05.2020 | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| Набор тестов | 14.05.2020 | Допира Валерия, Бочкарев Иван |
| Построение | План итерации | 22.05.2020 | Павлов Данила |
| Разработка системы | 03.06.2020 | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| UML диаграммы | 15.05.2020 | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| Тестирование системы | 22.05.2020 | Допира Валерия, Бочкарев Иван |
| Материалы поддержки пользователей | 22.05.2020 | Ильянов Вячеслав |
| План развёртывания | 22.05.2020 | Допира Валерия |
| Внедрение | План итерации | 31.05.2020 | Допира Валерия |
| Демонстрация работы  системы | 31.05.2020 | Бочкарев Иван |
| Элементы реализации | 31.05.2020 | Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| Компоновка продукта | 31.05.2020 | Павлов Данила,  Допира Валерия |
| Материалы поддержки пользователей | 29.05.2020 | Ильянов Вячеслав |

Нагрузка исполнителей

Количество дней выделенных на каждый ресурс представлено ниже (см. табл. 8).

Таблица 8 – Загрузка ресурсов по дням

|  |  |
| --- | --- |
| ***Ресурсы*** | ***Количество дней*** |
| План разработки ПО | 14 |
| Видение | 35 |
| Модель вариантов использования | 35 |
| Инфраструктура разработки | 12 |
| Глоссарий | 6 |
| UML диаграммы | 6 |
| Набор тестов | 3 |
| Материалы поддержки пользователей | 7 |
| Разработка продукта | 25 |

Количество часов в неделю, которые исполнители тратят на проект описано ниже (см. табл. 9, 10, 11, 12). Нагрузка распределена равномерно на исполнителей на протяжении всего проекта. Роли распределены в соответствии с планом разработки программного обеспечения. Исполнитель может распределять время в течение недели, как ему удобно: работая каждый день по несколько часов или полноценный рабочий день. Синхронизация работы проходит раз в неделю или чаще, если это необходимо.

* Допира Валерия

Таблица 9 – График исполнителя Валерии

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Фаза*** | ***Задача*** | ***Окончание работы*** | ***Затраченное***  ***время*** | ***Исполнители*** |
| Начало | План разработки ПО | 26.03.2020 | 2 дня | Допира Валерия |
| Уточнение | План разработки ПО | 30.04.2020 | 2 дня | Допира Валерия |
| UML диаграммы | 04.05.2020 | 7 дней | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| Набор тестов | 14.05.2020 | 3 дня | Допира Валерия, Бочкарев Иван |
| Построение | Разработка системы | 03.06.2020 | 7 дней | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| UML диаграммы | 15.05.2020 | 7 дней | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| Тестирование системы | 22.05.2020 | 2 дня | Допира Валерия, Бочкарев Иван |
| План развёртывания | 22.05.2020 | 1 день | Допира Валерия |
| Внедрение | План итерации | 31.05.2020 | 1 день | Допира Валерия |
| Компоновка продукта | 31.05.2020 | 3 дня | Павлов Данила,  Допира Валерия |

* Павлов Данила

Таблица 10 – График исполнителя Данила

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Фаза*** | ***Задача*** | ***Окончание работы*** | ***Затраченное***  ***время*** | ***Исполнители*** |
| Начало | Модель вариантов использования | 22.04.2020 | 2 дня | Павлов Данила |
| Уточнение | Модель вариантов использования | 04.05.2020 | 2 дня | Павлов Данила |
| UML диаграммы | 04.05.2020 | 7 дней | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| Построение | План итерации | 22.05.2020 | 1 день | Павлов Данила |
| Разработка системы | 03.06.2020 | 7 дней | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| UML диаграммы | 15.05.2020 | 7 дней | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| Внедрение | Элементы реализации | 31.05.2020 | 2 дня | Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| Компоновка продукта | 31.05.2020 | 3 дня | Павлов Данила,  Допира Валерия |

* Бочкарев Иван

Таблица 11 – График исполнителя Ивана

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Фаза*** | ***Задача*** | ***Окончание работы*** | ***Затраченное***  ***время*** | ***Исполнители*** |
| Начало | Видение | 22.04.2020 |  | Бочкарев Иван |
| Инфраструктура разработки | 22.04.2020 |  | Бочкарев Иван |
| Уточнение | Видение | 04.05.2020 |  | Бочкарев Иван |
| Инфраструктура разработки | 04.05.2020 |  | Бочкарев Иван |
| UML диаграммы | 04.05.2020 |  | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| Набор тестов | 14.05.2020 |  | Допира Валерия, Бочкарев Иван |
| Построение | Разработка системы | 03.06.2020 |  | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| UML диаграммы | 15.05.2020 |  | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| Тестирование системы | 22.05.2020 |  | Допира Валерия, Бочкарев Иван |
| Внедрение | Демонстрация работы  системы | 31.05.2020 |  | Бочкарев Иван |
| Элементы реализации | 31.05.2020 |  | Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |

* Ильянов Вячеслав

Таблица 12 – График исполнителя Вячеслава

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Фаза*** | ***Задача*** | ***Окончание работы*** | ***Затраченное***  ***время*** | ***Исполнители*** |
| Начало | Глоссарий | 27.04.2020 |  | Ильянов Вячеслав |
| Уточнение | Глоссарий | 04.05.2020 |  | Ильянов Вячеслав |
| UML диаграммы | 04.05.2020 |  | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| Построение | Разработка системы | 03.06.2020 |  | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| UML диаграммы | 15.05.2020 |  | Допира Валерия, Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| Материалы поддержки пользователей | 22.05.2020 |  | Ильянов Вячеслав |
| Внедрение | Элементы реализации | 31.05.2020 |  | Бочкарев Иван, Павлов Данила, Ильянов Вячеслав |
| Материалы поддержки пользователей | 29.05.2020 |  | Ильянов Вячеслав |

## 1.2. Постановка задачи

# 2. Обзор литературы

# 4. Участие в учебном процессе бакалавров

**ЗАКЛЮЧЕНИЕ**

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Ren X. J. et al. An analysis of performance evolution of Linux's core operations //Proceedings of the 27th ACM Symposium on Operating Systems Principles. – ACM, 2019. – С. 554-569.

2. Android platform. URL: <https://developer.android.com/guide/platform> (дата обращения: 15.12.2019)

3. Linux kernel // Memory. URL:​ <https://en.wikibooks.org/wiki/The_Linux_Kernel/Memory> (дата обращения: 10.11.2019)

4. Cgroup Documentation. URL:

<https://www.kernel.org/doc/Documentation/cgroup-v1/cgroups.txt> (дата обращения: 10.12.2019)

5. Yaghmour K. Embedded Android: Porting, Extending, and Customizing. – " O'Reilly Media, Inc.", 2013.

6. Gorman M. Understanding the Linux virtual memory manager. – Upper Saddle River : Prentice Hall, 2004. – С. 82.

7. Open Source and Linux Lab // Linux memory management summary. URL: http://wiki.osll.ru/doku.php/etc:users:jcmvbkbc:linux-mm​ ​ (дата обращения: 23.11.2019)

8. Vimal K., Trivedi A. A memory management scheme for enhancing performance of applications on Android //2015 IEEE Recent Advances in Intelligent Computational Systems (RAICS). – IEEE, 2015. – С. 162-166.

9. Hahn S. S. et al. Fasttrack: foreground app-aware I/O management for improving user experience of android smartphones //2018 {USENIX} Annual Technical Conference ({USENIX}{ATC} 18). – 2018. – С. 15-28.

10. Baik K., Huh J. Balanced memory management for smartphones based on adaptive background app management //The 18th IEEE International Symposium on Consumer Electronics (ISCE 2014). – IEEE, 2014. – С. 1-2.