

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
ФГБОУ ВО «Пермский государственный национальный исследовательский
университет»

Механико-математический факультет
Кафедра информационных технологий

УДК 004.9

Проектирование, разработка и документирование информационной
системы учета времени самостоятельной работы студентов в
компьютерном классе
Выпускная квалификационная работа бакалавра

Работу выполнил студент
группы ИТХ-19 4 курса
механико-математического
факультета
Васильева Игоря
Дмитриевича

«К защите допустить»
Зав. Кафедрой
информационных технологий
_____ / В. Н. Раевский/
« ____ » _____ 20__ г.

Электронная версия работы помещена в
электронную библиотеку кафедры ИТ
название файла _____
дата _____

Руководитель:
Кандидат педагогических наук, доцент
кафедры ИТ
Василюк Надежда Николаевна
« ____ » _____ 202__ г.

Пермь 2023

Аннотация

Васильев Игорь Дмитриевич

Направление «Информационные системы и технологии», группа ИТХ-19.

Проектирование, разработка и документирование информационной системы учета времени студентов для самостоятельной работы в компьютерном классе. Выпускная квалификационная работа. 2023 год.

В данной работе проанализированы аналоги существующих систем учета времени. В результате анализа составлены требования к проектированию системы, выявлены задачи на проектирование.

Работа содержит описание логики работы информационной системы в виде диаграмм, модель данных, а также возможный интерфейс пользователя.

Во введении обосновывается актуальность темы, поставленные задачи и цель, определяется предмет и объект исследования.

В первой главе дана теоретическая основа для разработки информационной системы учета времени студентов. В ней проведен обзор и анализ аналогов, а также выявлены требования к разрабатываемой системе.

Во второй главе описан процесс выбора инструментальных средств проектирования, проектирования модели информационной системы и описания используемых технологий.

В третьей главе представлен выбор инструментальных средств разработки, а также сама разработка информационной системы. Кроме того, в главе представлены основные рекомендации по совершенствованию разработанной информационной системы.

В заключении сформированы выводы по теме выпускной квалификационной работы.

Спроектированная информационная система, может быть использована для работника компьютерного класса и для студентов, желающих записаться в компьютерный класс на самостоятельную работу.

Работа содержит: стр. ил., библи. назв

Ключевые слова: учет времени, самостоятельная работа студента, компьютерный класс, информационная система, uml-диаграммы.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
Обзор литературы.....	6
Глава 1 Анализ предметной области	8
1.1 Теоретическое обоснование разработки	8
1.2 Обзор и сравнение существующих аналогов ИС	8
1.3 Требования к проектируемой системе.....	14
1.4 Вывод	14
Глава 2 Проектирование модели информационной системы учета СРС студента ..	15
2.1 Обзор и выбор методологии моделирования.....	15
2.2 Выбор CASE-средства проектирования ИС	18
2.3 Моделирование информационной системы.....	20
2.4 Вывод	27
Глава 3 Разработка прототипа информационной системы учет СРС студента	28
3.1.1 Интерфейс студента.....	28
3.1.2 Интерфейс лаборанта.....	31
3.2 Выбор средств разработки ИС	34
3.2.1 Система управления базой данных	34
3.2.2 Серверная часть	35
3.2.3 Клиентская часть	37
3.3 Создание базы данных	37
3.4 Реализация прототипа ИС.....	38
3.5 Рекомендации по совершенствованию ИС	42
3.6 Вывод	42
Заключение.....	43
Библиографический список.....	44
ПРИЛОЖЕНИЕ А	47
ПРИЛОЖЕНИЕ Б	48
ПРИЛОЖЕНИЕ В.....	49
ПРИЛОЖЕНИЕ Г	50

ВВЕДЕНИЕ

В современном мире наиболее важную роль в развитии играет информация. Основываясь на ней, человек может делать выводы и строить свое и общее будущее. Для того, чтобы не запутаться и не затеряться в многочисленных информационных потоках, человек создает и совершенствует инструменты управления информацией. И именно информационная система является главным инструментом управления сведений об окружающем мире и протекающих в нем процессах.

Информационная система (ИС) – система, которая организует хранение и манипулирование информацией о предметной области. Такое определение в широком смысле описывает ГОСТ 34.321-96 [1]. В узком смысле, информационная система – это система, предназначенная для автоматизации бизнес-процессов компании или отдельных пользователей, позволяющая обработку данных, включая удаления, изменение и добавление. Главной задачей ИС является полное удовлетворение информационных потребностей конечного пользователя.

Разработка ИС учета времени студентов для самостоятельной работы в компьютерном классе является актуальной задачей, потому что в высших учебных заведениях наблюдается тенденция к компьютеризации и переход к электронному документированию, а также необходимость к быстрому и удобному доступу к данным и их обработке. Также немаловажным фактором является доступность ресурса из любого места, где есть интернет.

Таким образом, объектом исследования является учет времени самостоятельной работы студентов.

Предметом исследования является автоматизация учета времени самостоятельной работы студентов в компьютерном классе.

Цель данной курсовой работы – проектирование, разработка и документирование ИС учета времени студентов для самостоятельной работы в компьютерном классе, для автоматизации работы сотрудника, отвечающего за компьютерный класс.

Исходя из цели работы, выделим следующие задачи, которые необходимо решить в ходе проектирования:

- 1) изучить существующую литературу по теме исследования;

- 2) изучить аналоги системы учета времени;
- 3) провести сравнительный анализ этих систем;
- 4) выявить и определить требования к разрабатываемой ИС;
- 5) провести анализ существующих инструментов для проектирования ИС;
- 6) определить набор инструментальных средств в соответствии с требованиями;
- 7) спроектировать модель ИС;
- 8) разработать прототип интерфейса ИС;
- 9) разработать прототип ИС;
- 10) сформировать рекомендации по совершенствованию ИС.

Обзор литературы

Целью данного обзора является выяснение основных проблем в управлении процесса учета самостоятельной работы студентов (СРС) в компьютерном классе. Самостоятельная работа – это важнейшая часть жизни студентов, потому что она позволяет развивать навыки творческого мышления и уникального представления о дисциплинах, которые невозможно получить без саморазвития. Это утверждение подтверждает учебный план по направлению «Информационные системы и технологии» [2], в котором выделено 5042 часа самостоятельной работы из 7996 часов планируемой трудоемкости. Это составляет около 63% всей учебной деятельности студента в университете. Так как во множестве дисциплин требуется дорогостоящее компьютерное оснащение и программное обеспечение, учащимся приходится пользоваться вычислительными возможностями университета. Вследствие чего, обучающемуся необходимо правильно распределять свои временные ресурсы. Поэтому особенно важно знать наиболее надежные и эффективные методы создания благоприятных условий для организации и учета СРС в компьютерном классе.

Данную тему частично рассматривает методист Чебаркульского профессионального техникума Т. А. Кудрявцева в своей статье «Проблемы организации самостоятельной работы студентов в условиях реализации федеральных государственных образовательных стандартов среднего профессионального образования третьего поколения» [3]. Основными трудностями при организации СРС в компьютерном классе стали:

- 1) неумение студентов планировать СРС;
- 2) слабые навыки в оформлении самостоятельной работы студентами;
- 3) недостаточное знание технологий сотрудниками и студентами;
- 4) неумение рассчитать трудоемкость заданий СРС.

Таким образом, одним из главных решений, является осуществление и обеспечение материально-технических условий, предполагающие наличие необходимого уровня оснащенности кабинетов, методические материалы для ознакомления работников с бизнес-процессом, наглядные инструкции для оформления СРС, необходимая информация для обучения студента самостоятельному планированию своего времени в компьютерном классе и

организация учета для СРС.

С точки зрения локальных актов этот вопрос рассмотрела заместитель директора по учебно-методической работе Н. В. Ананьина в статье «Организация самостоятельной работы студентов в условиях реализации ФГОС» [4], где основными документами, регламентирующие вопросы реализации учета СРС стали:

- 1) положение о самостоятельной работе студентов, определяющее сущность самостоятельной работы студентов, ее назначение, планирование, формы организации и виды контроля;
- 2) положение о журнале учета самостоятельной работы студентов, определяющее порядок оформления и ведения журналов учета СРС.

В связи с этим, можно предположить, что заполнение журналов отчета СРС, подготовка каждого студента к самостоятельной работе без использования современных информационных технологий в сфере обработки данных замедляет процесс осуществления СРС. В свою очередь, возникает проблема хранения, сортировки, доступа к данным и их своевременной передачи. Хранение бумажных носителей является дорогостоящим и трудоемким процессом, так как важно позаботиться о месте хранения, оценить риски повреждения информации и дублирования бумажных носителей. Также заполнение документов в письменном виде, влечет за собой множество ошибок и опечаток, которые могут повлечь за собой неблагоприятные последствия.

Отметим, что проектирование и документирование информационной системы учета времени студентов для самостоятельной работы в компьютерном классе является необходимостью, потому что использование информационных технологий и вычислительной техники существенно снизит имеющиеся недостатки в процессе заполнения документов, записи СРС, снизит трудоемкость и повысит достоверность данных.

Глава 1 Анализ предметной области

В этой главе описана актуальность проектируемой системы, проанализированы существующие информационные системы, выявлены и определены основные требования к ИС.

1.1 Теоретическое обоснование разработки

В настоящее время большее число учебных учреждений внедряют и используют различные ИС для автоматизации работы преподавателей и подразделений университета. Компьютеризация бизнес-процессов позволяет добиваться высокой скорости ведения документов и предотвращает большинство допускаемых ошибок при фиксации.

Как правило, для того чтобы студенту записаться на самостоятельную работу в компьютерный класс, ему каждый раз необходимо лично подходить к лаборанту или ответственному за компьютерный класс, который по разным причинам может отсутствовать на рабочем месте. Информационная система по учету СРС, может решить эту проблему путем доступа к записи из сети интернет.

В свою очередь, сотрудник компьютерного класса должен понимать, сколько студентов записалось и в какое время, за какими компьютерами, они будут работать, сколько времени потребуется для решения их задач, и с какой целью им потребовался компьютер. Хранить эту информацию на бумажном носителе неэффективно, потому что анализировать такие данные гораздо сложнее, чем в информационных системах.

Таким образом, проектирование такой ИС является актуальной проблемой, которая позволит упростить доступ студентов к записи и облегчит анализ отчетов.

1.2 Обзор и сравнение существующих аналогов ИС

На сегодняшний день в открытом доступе существует некоторое количество систем для учета времени СРС студента. Многие такие системы разрабатываются внутри учебных учреждений и являются частью системы университета. Большую часть рынка занимают коммерческие продукты, для учета клиентов в разных предметных областях. Рассмотрим системы, предназначенные для СРС студентов, а также изучим самые подходящие и универсальные продукты, которые можно приобрести и использовать в компьютерном классе.

1) «1С: Университет» [5]

Разработан фирмой «1С». Представляется обширным программным обеспечением, основной задачей которого является анализ и обработка всех данных студентов, но к тому же им можно пользоваться и при учете времени студентов.

Отличительной особенностью данной системы оказывается ее настраиваемость под определенные задачи. Именно поэтому пользователи системы зависят от первоначальной конфигурации.

Перечислим основные особенности системы:

- хранение и обработка сведений о контингенте студентов вуза (Рис. 1);
- создание расписаний для компьютерного класса;
- возможность распределения студентов по компьютерам в классе;
- учет посещения студентов;
- формирование графиков работы студентов и компьютерных классов;
- создание почасовых и суточных отчетов по посещениям студентов.

Тем не менее данная система является не лучшим вариантом для нашей предметной области. Дело в том, что она имеет множество недостатков, таких как:

- долгое внедрение и развертывание;
- трудоемкий процесс обучения персонала;
- постоянная поддержка ИТ;
- высокая стоимость владения (около 96 000 рублей за копию) [6].

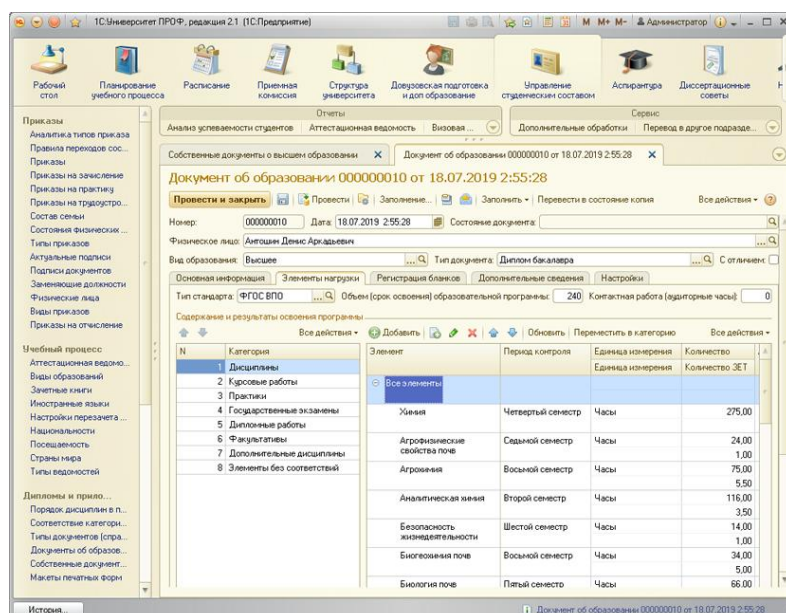


Рисунок 1 – 1С:Университет

2) Информационная система «Электронный журнал» [7]

Данная система разработана ГАПОУ МО "МЦК — Техникум имени С. П. КОРОЛЁВА" и является частью большого программного комплекса Платформа «Цифровой колледж Подмосковья».

ИС «Электронный журнал» предназначена для ведения учета и анализа учебной деятельности студентов (Рис. 2) в образовательных организациях, повышения объективности оценивания учебных результатов студентов, а также обеспечения учета посещаемости занятий студентами. ИС осуществляет автоматизированный подсчет рейтинга студентов, а также позволяет управлять настройками для изменения выводимой информации. Данные электронного журнала по окончании учебного года хранятся в ИС.

Пользователями системы, могут быть как студенты, так и преподаватели, вход осуществляется через логин и пароль.

Профессиональная образовательная организация	МЦК-Техникум им. С.П. Королева (demo@ac204.ru)
Курс	Технология каменной кладки
Группа	группа_02
Проведено занятий	7
Дата	13/09/2017

Преподаватель
 Все

Форма контроля
 Все

Период обучения
 Весь период обучения

Добавить занятие

☒ Показать тематический план
☐ Показать итоговые оценки по курсу
☐ Показать рейтинг студентов по классно-урочным занятиям

№	Студент	Сентябрь 2017						Дата	Тема	Форма контроля	Комментарии	Преподаватель
		1	4	5	7	8						
		СР	ЭУМК	СР	РЕФ	УО	ЛР					
1	Мечая Валерия	4	2	5	5	5	5	01/09/2017	Классификация зданий	Самостоятельная работа		Пахомов Владимир
2	Николаева Любовь	4	2	4	5	5	4	04/09/2017	Тестирование "Технология каменной кладки"	ЭУМК		Пахомов Владимир
3	Петров Василий	5	2	4	5	5	4	05/09/2017	Система перевязки швов каменной кладки	Самостоятельная работа		Пахомов Владимир
4	Соловьев Герман	4		3	5	5	5	05/09/2017	Типы зданий	Реферат		Пахомов Владимир
5	Суханова Анастасия	5		5	5	5	5	07/09/2017	Кладка по цепной системе перевязки швов	Устный ответ		Пахомов Владимир
6	Тимофеев Владимир	5	2	4	4	5	5	07/09/2017	Облегченная кладка	Лабораторная работа		Пахомов Владимир
7	Черных Николай	4	2	4	4	5	5	08/09/2017	Разряды каменщиков	Контрольная работа		Пахомов Владимир

Рисунок 2 – «Цифровой колледж Подмосковья»

Основной функционал ИС:

- обеспечение контроля за посещаемостью студентов;
- систематизация и отображение информации о ходе и результатах

образовательного процесса студентов;

- просмотр итоговых результатов посещения студентов за определенный период обучения;
- осуществление гибкой системы настроек структуры учебного года;
- опциональное размещение комментариев и замечаний;
- просмотр и редактирование данных в электронном журнале в соответствии с уровнем доступа.

Отличительной чертой этого программного продукта, является то, что он доступен только для колледжей Подмосковья в системе «Цифровой колледж Подмосковья», и никак не может быть интегрирован в другие системы.

3) Yclients [8]

Данный продукт разработала компания – ООО "УАЙКЛАЕНТС". Эта система предназначена для учета клиентов различных предприятий и всевозможных сфер услуг, где без сомнения, важнейшую роль играют посетители. Yclients, помогает не только распланировать весь день для потребителей, но и вести весь учет в электронном виде.

Пользователям системы могут быть сотрудники, так и потребители, потому что в текущем программном продукте, можно настроить онлайн запись через интернет.

Основной функционал:

- записать человека на определенный день и время (Рис. 3);
- учет количества клиентов и даты их записи (Рис. 4);
- онлайн запись клиентов;
- ведение статистики по дням, месяцам, кварталам и годам;
- сохранение информации о клиенте в базу данных.

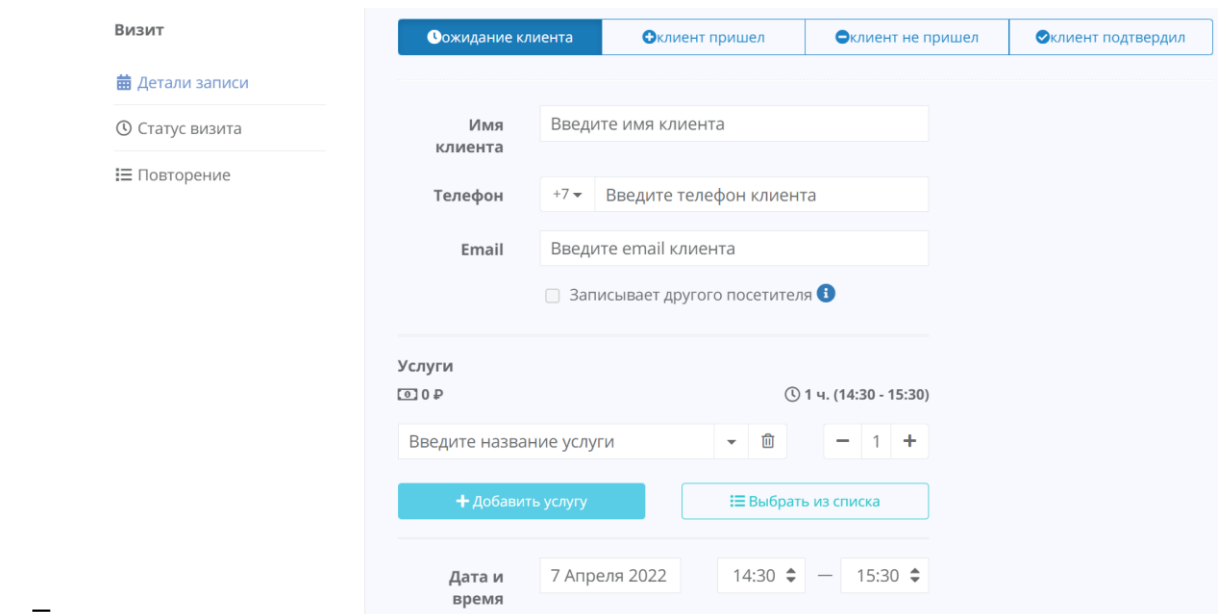


Рисунок 3 – Запись

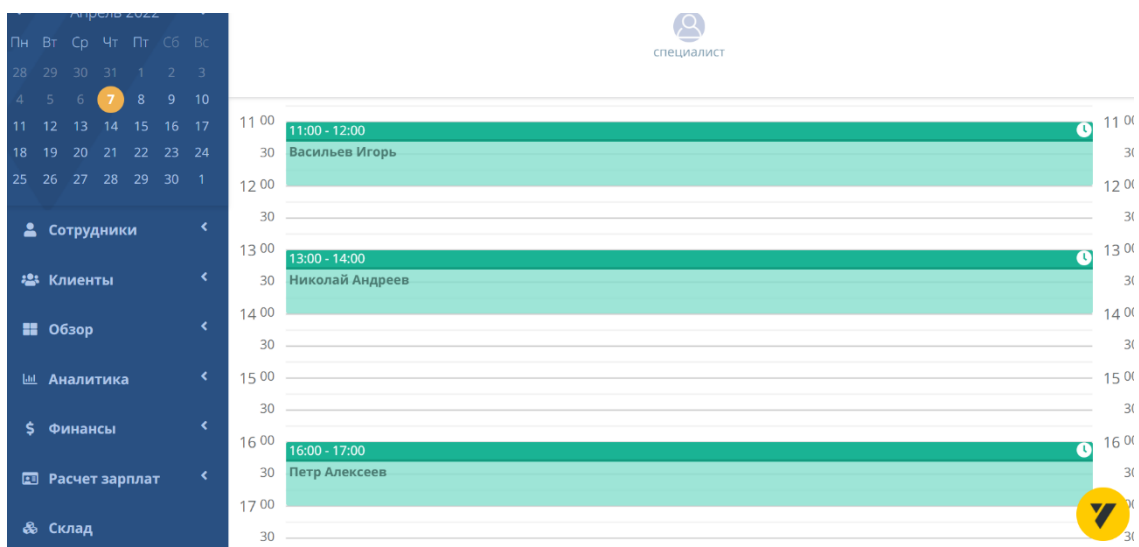


Рисунок 4 – Расписание

Стоимость системы рассчитывается по подписке. Цены варьируются от 714 руб. до 1250 руб. за месяц использования.

4) Универсальная система учета [9]

Данная система разработана компанией «USU Software», является разработкой программного обеспечения для автоматизации бизнес-процессов. Представляет собой программу, в которой администратор может оперативно принимать заявки от клиентов и вести запись в электронном формате.

Основные функциональные возможности:

- формирование единой базы клиентов со всеми необходимыми контактными данными (Рис. 5);
- обозначение как выполненной, так и планируемой работы по любому

клиенту;

- напоминая о любых важных делах;
- планирование резервного копирования;
- импорт/экспорт информации.

Стоимость программы составляет 60 американских долларов.

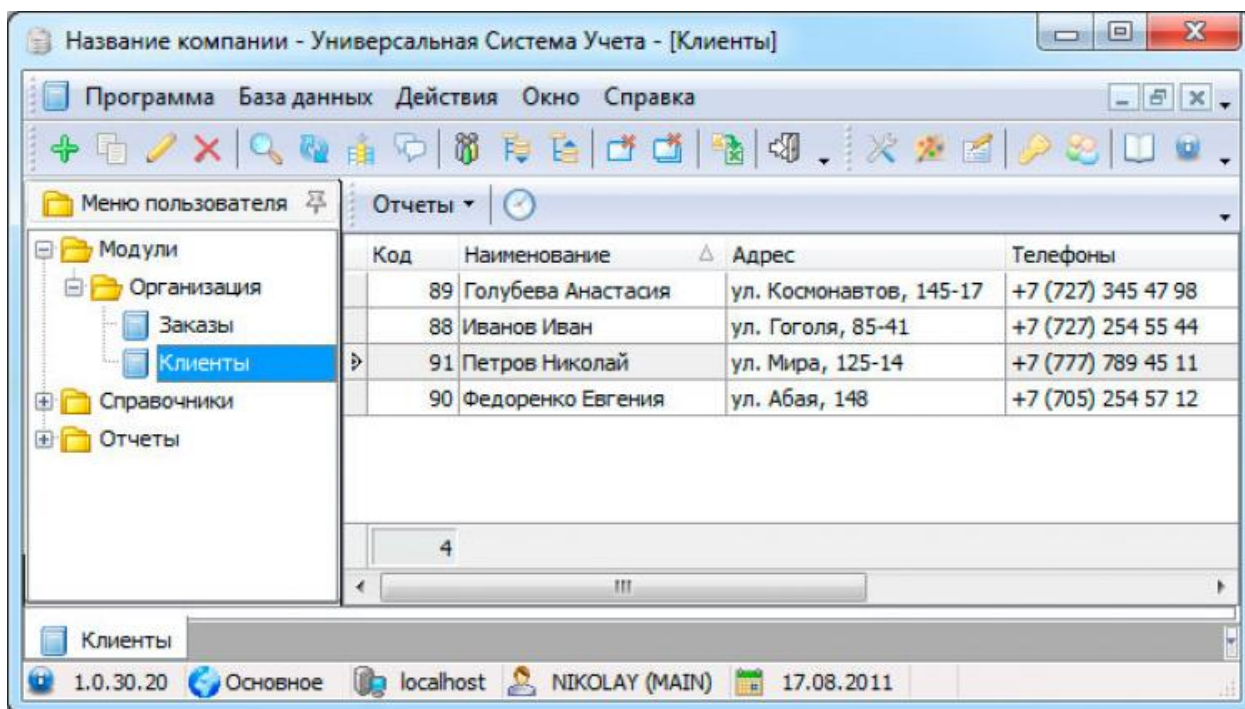


Рисунок 5 – Универсальная система учета

Таким образом, рассмотренные выше системы, являются многофункциональными и имеют как преимущества, так и ряд недостатков. Например, одна из рассмотренных систем, не предполагает использование в учебных учреждениях, не являющихся колледжами Подмосквья, к тому же эта система рассчитана на форму и структуру организации только для этих образовательных организаций.

Вместе с тем, решение от «1С», хоть и является универсальным, но тем не менее внедрение такой огромной системы является ресурсозатратным процессом, и не является правильным действием в данной предметной области.

Кроме того, многие из представленных оболочек, не ориентированы на учет времени самостоятельной работы студента, что является критическим в выборе программного продукта. Особенно важной чертой продукта должна быть запись студента через интернет, что присутствует только в 2-х рассмотренных системах.

Анализ существующих аналогов демонстрирует, что разработка своей ИС

по учету времени СРС в компьютерном классе предпочтительнее, чем покупка аналогичного продукта у стороннего разработчика. При этом проектирование, документирование и разработка программного продукта позволит организовать именно те функциональные требования, которые необходимы работникам компьютерного класса и студентам, а также убрать избыток функционала других программ.

1.3 Требования к проектируемой системе

На основе анализа существующих системы выбраны следующие функциональные требования, которые должны осуществляться разрабатываемой ИС:

- учет времени работы студента в компьютерном классе;
- учет времени бронирования компьютера студентом;
- контроль свободных компьютеров в классе;
- хранение данных о компьютерах, студентах и времени их работы;
- предоставление отчетов по запросу руководителя подразделения;
- удобное представление информации работнику образовательного учреждения;
- дистанционный доступ к записи для студентов;
- отметка о посещении.

1.4 Вывод

В первой главе исследована актуальность проектируемой системы, проанализированы существующие аналоги информационных систем, выявлены и определены основные требования к ИС.

Глава 2 Проектирование модели информационной системы учета СРС студента

В данной главе описаны методологии моделирования и выбрана подходящая, средства моделирования для данной методологии для текущей темы работы.

2.1 Обзор и выбор методологии моделирования

В настоящее время, проектируются информационные системы с крайне высокой степенью сложности, это зависит от количества функционала, который предоставляет продукт. Довольно часто, функции и процедуры, классы и методы, могут видоизменяться, объединяться, выделяться в независимые модули, а также модифицироваться. В целях повышения организации проектирования ИС наиболее оптимальным способом и улучшения качества приложения, прежде всего нужно смоделировать основные процессы. Для этого используются различные методологии моделирования.

В связи с этим крайне важно выбрать подходящую методологию для проектирования для того, чтобы в достаточной степени описать протекающие процессы внутри системы. В конечном итоге именно от этого зависит, достаточно ли наглядной, удобной и понятной окажется модель.

Рассмотрим наиболее популярные методологии проектирования [10]:

1) IDEF (Integrated Definition for Function Modeling) [11]

IDEF – это целый набор аналитических средств, применяемых не только в управлении бизнесом, но и во многих других сферах. С помощью диаграмм IDEF можно эффективно отображать и анализировать модели деятельности широкого спектра сложных систем в различных разрезах. При этом широта и глубина обследования процессов в системе определяется самим разработчиком, что позволяет не перегружать создаваемую модель излишними данными.

В методологии IDEF есть множество стандартов, но IDEF0 – наиболее часто используемая методология функционального моделирования.

IDEF0 (Рис. 6) имеет несколько преимуществ:

- наглядное представление процесса, имеющее входы/выходы, ресурсы и управление;

- декомпозиция, с помощью которой можно разбить сложный процесс, на составляющие ее функции, при этом уровень детализации определяется разработчиком;
- глоссарий является набором из соответствующих определений, которые характеризуют объект и описывают сущность элемента IDEF0.



Рисунок 6 – Пример диаграммы IDEF0

Таким образом, при проведении сложных проектов обследования предприятий, разработка моделей в стандарте IDEF0 позволяет наглядно и эффективно отобразить весь механизм деятельности проекта в нужном разрезе.

2) BPMN (Business Process Management Notation) [12]

BPMN (Рис. 7) – это язык моделирования бизнес-процессов, который является промежуточным звеном между формализацией и воплощением бизнес-процесса. Значимым является то, что моделирование BPMN позволяет провести автоматизацию бизнес-процессов в соответствии с созданной схемой.

Достоинства BPMN:

- простота трансляции диаграмм в исполняемые модели с помощью языка формального описания бизнес-процессов;
- доступность и понятность диаграмм для большинства участников бизнес-процессов, зачастую не требующих дополнительного разъяснения;
- импорт готовых схем бизнес-процессов в BPM-системы.

Иначе говоря, BPMN-диаграммы рассчитаны на всех участников бизнес-процессов, с целью получения доступной картины задействованных этапов.

Отличительной особенностью является то, что схемы, созданные в среде моделирования BPMN, можно преобразовать в исполняемый файл в BPM-системах.

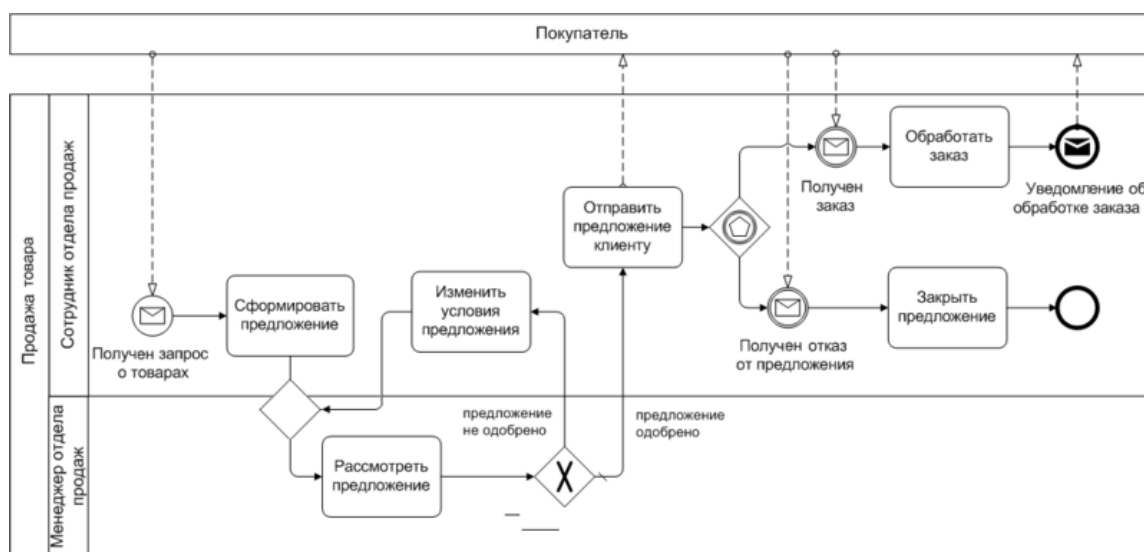


Рисунок 7 – Пример диаграммы BPMN

3) UML (Unified Modeling Language) [13]

UML – унифицированный язык моделирования, предназначенный для визуализации, специфицирования, конструирования и документирования процессов, создаваемых в ходе разработки. Методология UML – это определенное множество уникальных диаграмм, которые используются для демонстрации ИС с разных сторон.

В первую очередь UML-диаграммы – это средство визуализации. Благодаря простым правилам моделирования и интуитивно понятным элементам диаграмм, такие схемы могут понимать все заинтересованные лица. Также схемы могут использоваться как средство общения между людьми. Вместе с тем, можно изобразить информационную систему с разных сторон в разных диаграммах, которые можно показать заказчику и обсудить с ним.

К тому же UML позволяет строить модели программных систем. По этим моделям потом может производиться генерация каркасного кода проектируемых приложений. Более того, возможен процесс «реверс-инжиниринга», создание UML-модели из существующего кода приложения.

Кроме того, UML-модели сами по себе уже являются документами. Причем любой элемент на любой диаграмме может быть снабжен пояснительной запиской, для более глубокого понимания процесса, описанного в диаграмме.

Таким образом, UML можно использовать для рисования диаграмм, которые можно использовать для коммуникаций внутри команды и в ходе взаимодействия с заказчиком, т. е. он может служить средством обмена информацией. Кроме этого, UML является отличным средством спецификации систем, причем спецификации в процессе разработки. Немаловажным достоинством является то, что разработанные архитектурные решения, задокументированные с помощью UML, могут быть использованы повторно.

В результате обзора наиболее популярных методологий проектирования, для проектирования и документирования информационной системы учета времени самостоятельной работы студентов в компьютерном классе больше всего подойдет UML, поскольку совокупность методов, которая содержит данная методология, позволит описать ИС наиболее полно, в виду того, что UML является набором множества диаграмм, с помощью которых можно представить систему с разных точек зрения.

В том числе, созданные диаграммы, позволяют решить проблему документирования системной архитектуры и всех ее деталей. Схемы, разработанные в UML, будут служить документами, которые объясняют, как устроена система, даже для людей, не относящихся к этой предметной области.

2.2 Выбор CASE-средства проектирования ИС

Для эффективного проектирования с методологией UML, нам потребуется соответствующее CASE-средство.

Под термином CASE-средства [14] понимаются программные средства, поддерживающие процессы создания и сопровождения ИС, включая анализ и формулировку требований, проектирование прикладного ПО (приложений) и баз данных, генерацию кода, тестирование, документирование, обеспечение качества, конфигурационное управление и управление проектом, а также другие процессы. CASE-средства вместе с системным ПО и техническими средствами образуют полную среду разработки ИС.

В настоящее время создано множество CASE-средств, каждое из которых ориентировано на те или иные процессы жизненного цикла разработки ИС.

Для проектирования и документирования ИС учета времени СРС в компьютерном классе, выделим основные критерии для выбора программного

продукта: поддержка UML, наличие бесплатной версии, поддержка операционной системы Windows, простой и интуитивно понятный интерфейс.

Исследуем несколько CASE-средств наиболее подходящих под критерии для проектирования нашей ИС:

1) StarUML [15]

StarUML – это сложное средство моделирования программного обеспечения, предназначенное для поддержки быстрого и лаконичного моделирования. StarUML поддерживает несколько платформ: MacOS, Windows и Linux. StarUML строго придерживается спецификации UML, разработанной OMG для моделирования программ. StarUML использует концептуальный подход, который применим к любым методологиям/процессам. Легко создаются не только модели под средства разработки для конкретных платформ типа .NET или J2EE, но также и для других основных структур программных моделей. StarUML также поддерживает обратный инжиниринг и может создавать диаграмму классов из существующего кода.

Среда разработки StarUML на хорошем уровне настраивается в соответствии с требованиями пользователя и имеет высокую степень расширяемости, особенно в области своих функциональных возможностей.

Распространяется бесплатно без ограничения по времени, для оценки работоспособности программы, но при сохранении имеет водяные знаки «UNREGISTERED».

2) Violet UML Editor [16]

Violet UML Editor – редактор UML диаграмм, предназначенный для разработчиков, студентов, учителей и авторов, которым необходимо быстро создавать простые UML диаграммы. Данный редактор имеет следующие преимущества:

- простой и понятный интерфейс;
- поддержка основных видов диаграмм UML;
- распространяется бесплатно;
- кроссплатформенность.

Тем не менее, Violet UML Editor имеет ряд существенных недостатков, таких как недостаточно хорошая проверка семантики, т.е. присутствует возможность нарисовать пользователю противоречивые диаграммы, нет импорта/экспорта XMI

документов, нет генерации кода из диаграммы, нет обратного инжиниринга.

3) Diagram Designer [17]

Diagram Designer – бесплатное решение для создания блок-схем, диаграмм и слайд-шоу. Данная программа распространяется с открытым кодом и любой человек, может модифицировать код, что определенно является достоинством.

Основные функции приложения:

- построение простых диаграмм классов;
- проверка орфографии;
- настраиваемая палитра шаблонов;
- «карманный» калькулятор для решений уравнений;
- импорт/экспорт изображений.

Однако, данная система, не лучший вариант для проектирования информационной системы, так как она поддерживает только диаграмму классов, и не поддерживает другие виды UML-диаграмм. Также отсутствует генерация кода по диаграмме, что является большим недостатком.

4) ArgoUML [18]

ArgoUML — средство UML моделирования. ArgoUML является открытым программным обеспечением и распространяется под лицензией EPL. ArgoUML полностью написан на Java и для работы ему подходит любая операционная система с установленной Java 2 JRE или JDK версии 1.4 или выше. Поддерживает спецификации UML 1.4, генерацию исходного кода, Обратный инжиниринг из исходного кода.

Но с моей точки зрения, ArgoUML имеет не интуитивно понятный интерфейс, в котором тяжело разобраться новым пользователям системы.

Исходя из всех вышеперечисленных CASE-средств, учитывая критерии отбора, наиболее подходящим для моделирования ИС является StarUML, так как имеет дружелюбный и понятный интерфейс, поддерживает Windows, имеет подробную документацию и может использоваться бесплатно.

2.3 Моделирование информационной системы

При моделировании ИС будем опираться на задачи, определенные ранее. Доступ к ИС для лаборанта и для студентов через интернет-сайт.

Для достижения целей визуального проектирования с использованием

нотации UML вначале строится диаграмма вариантов использования. Диаграмма вариантов использования помогает исследовать и документировать различные случаи использования системы и ее функциональные возможности.

Диаграмма вариантов использования – это исходное концептуальное представление или концептуальная модель системы в процессе ее проектирования и разработки [19]. Можем выделить главные задачи диаграммы:

- определить общие границы и контекст моделируемой предметной области на начальных этапах проектирования системы;
- сформулировать общие требования к функциональному поведению проектируемой системы;
- разработать исходную концептуальную модель системы для ее последующей детализации в форме логических и физических моделей;
- подготовить исходную документацию для взаимодействия разработчиков системы с ее заказчиками и пользователями.

Таким образом, логика работы информационной системы представлена на следующих рисунках (Рис. 8, Рис. 9).

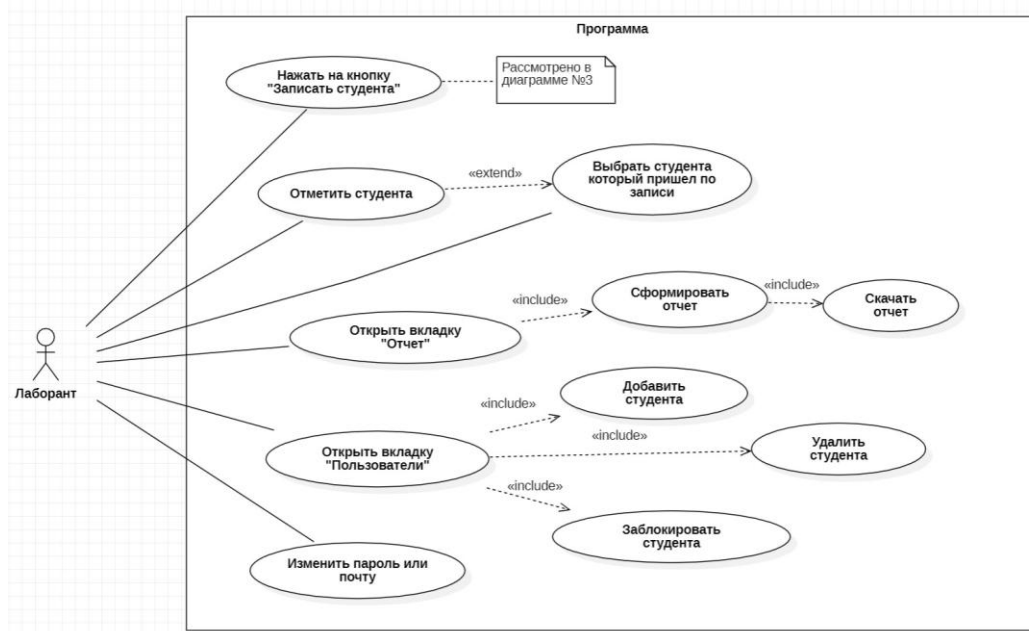


Рисунок 8 – Диаграмма прецедентов «Лаборант» №1

В процессе изучения диаграммы прецедентов «Лаборант» №1, изображенной выше (Рис. 8), можно выделить 5 основных действий, которые может совершить лаборант при работе с ИС:

- записать студента;

- отметить присутствие студента, т. е. изменить статус посещения;
- открыть отчет о студентах, и при надобности скачать;
- изучить пользователей-студентов, а также добавить, заблокировать и удалить. Заблокированный пользователь, больше не сможет входить в систему.
- изменить email или пароль.

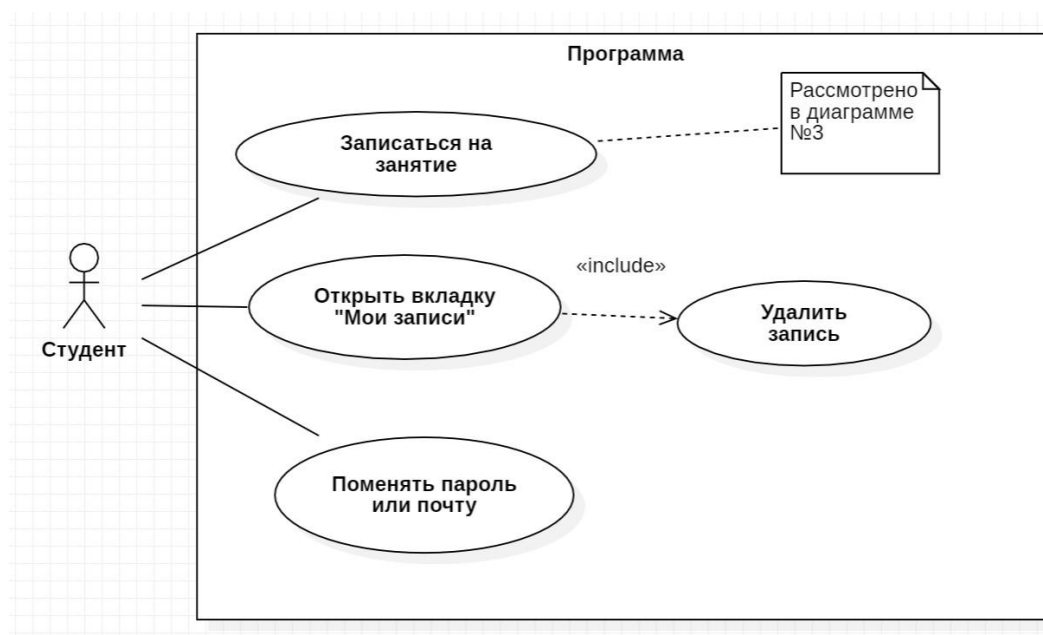


Рисунок 9 – Диаграмма прецедентов «Студент» №2

Рассмотрим диаграмму прецедентов «Студент» №2 (Рис. 9). На ней можно выделить 3 действия:

- записаться на занятие в компьютерный класс;
- посмотреть свои записи, и удалить при надобности уже существующую запись;
- поменять email или почту.

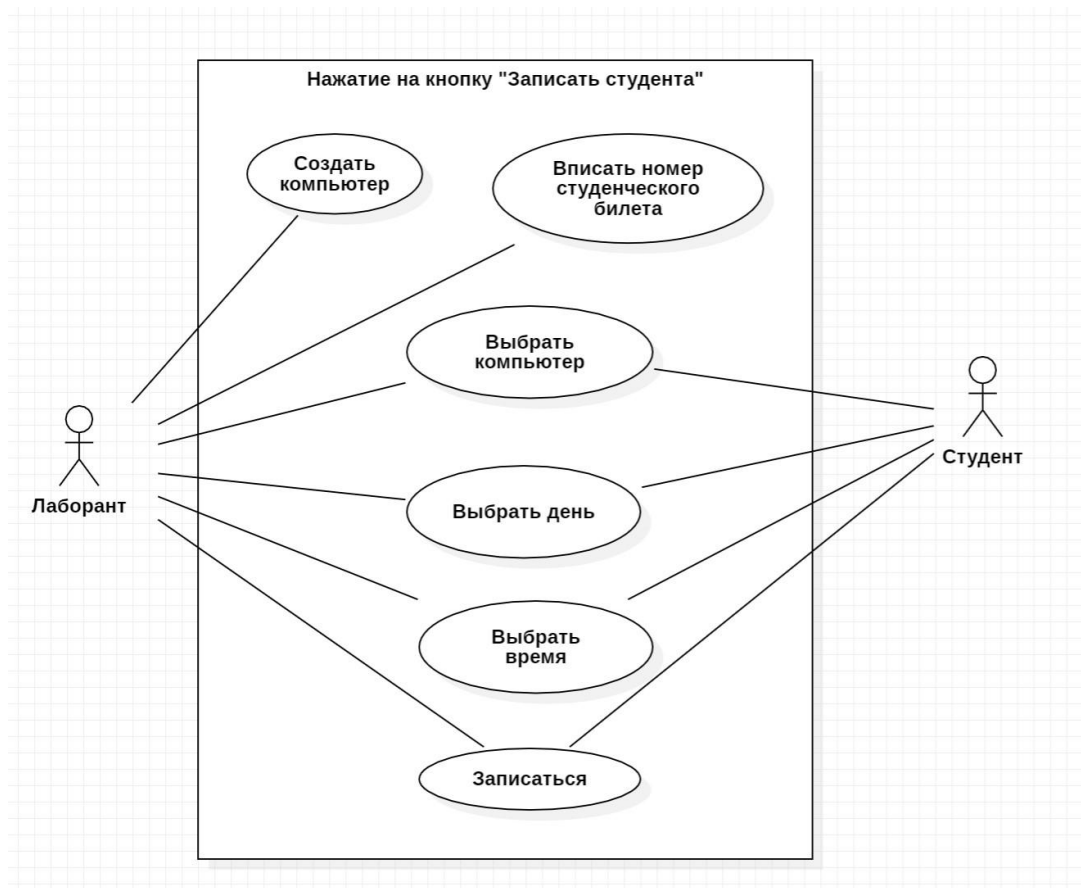


Рисунок 10– Диаграмма прецедентов «Записать студента» №3

Рассмотрим диаграмму прецедентов №3 (Рис. 10). На ней изображены два актера: студент и лаборант. Лаборант, помимо записи в компьютерный класс, может создать компьютер для последующей записи студентов. ИС подразумевает, что для записи студента на свободный компьютер для лаборанта, необходимо проделать следующие операции, в том порядке, который указан далее:

- 1) выбрать компьютер, соответствующий требованиям и пожеланиям студента;
- 2) выбрать пришедшего студента, по его номеру студенческого билета, а если студента нет в системе, указать информацию о студенте, а именно: имя, фамилию, номер студенческого билета, пароль сгенерируется автоматически;
- 3) выбрать интересующий день, для работы в компьютерном классе;
- 4) выбрать свободное время;
- 5) сохранить запись.

В свою очередь, для записи в компьютерный класс, будучи студентом, не нужно указывать студенческий номер, так как при авторизации, система точно

понимает, какой пользователь работает, и, создавая запись, автоматически подставляет нужного студента. Таким образом, запись для студента точно такая же, как у лаборанта, не включая ввод студенческого номера.

Далее рассмотрим диаграмму классов. Диаграмма классов — диаграмма языка UML, на которой представлена совокупность декларативных или статических элементов модели, таких как классы, с атрибутами и операциями, а также связывающие их отношения [19]. Диаграмма классов предназначена для представления статической структуры модели системы в терминологии классов объектно-ориентированного программирования.

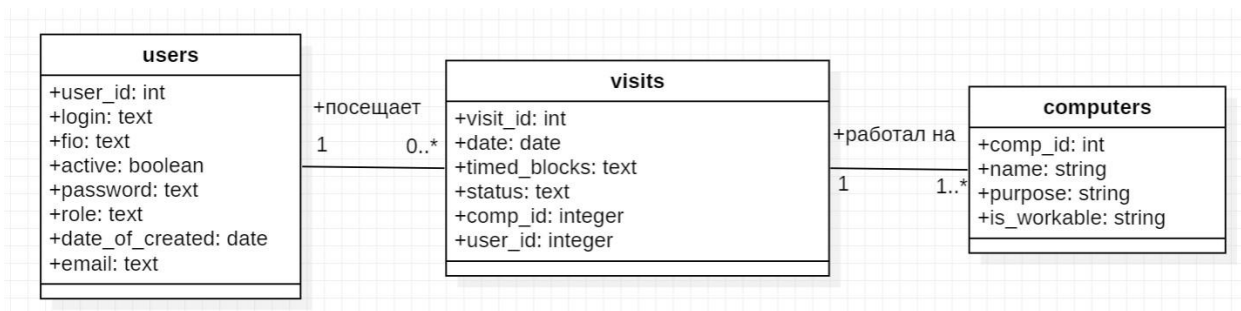


Рисунок 11 – Диаграмма классов

Следовательно, на диаграмме классов (Рис. 11), представлены 3 класса:

- users - «пользователи»;
- computers - «компьютеры»;
- visits - «посещения компьютерного класса».

Сущность «Студенты» нужна для хранения информации о пользователях, и поможет организовать авторизацию и аутентификацию в проектируемой системе. Данный класс имеет следующие атрибуты:

- user_id(int) – является первичным ключом, именно по этому атрибуту можно определить запись в таблице;
- login(string) – для студента, является студенческим номером, по которому можно определить студента, для лаборанта, любой набор символов, не пересекающийся с другими пользователями;
- fio(string) – ФИО пользователя, для упрощенного понимания, чей этот аккаунт;
- active(boolean) – состояние активности аккаунта, если имеет значение true, значит пользователь может работать системой, иначе пользователь заблокирован и не имеет право авторизоваться на сайте, заблокировать

пользователя может лаборант;

- password(string) – этот атрибут необходим для авторизации студента через сеть, хранение будет осуществляться в виде захешированной строки, для того, чтобы злоумышленники не смогли понять, какой пароль у пользователя;
- role(string) – в разрабатываемой системе может принимать значения «ROLE_STUDENT» и «ROLE_WORKER», для разграничения информации, представляемой в системе, а также для доступа к разным веб-страницам;
- date_of_created(date) – дата создания пользователя;
- email(string) – электронная почта для связи с пользователем, также для студента почта будет использоваться для отправки пароля.

Сущность «Компьютеры» необходима для хранения информации о компьютерах в классе и имеет 3 атрибута:

- comp_id(int) – является первичным ключом, именно по этому атрибуту можно определить запись в таблице;
- name(string) – наименование компьютера;
- purpose(string) – назначение компьютера (подходит ли этот компьютер для задач студента)
- is_workable(bool) – отвечает за работоспособность компьютера.

И главная сущность «Посещения компьютерного класса», с которой ИС будет работать чаще всего. Она требуется, потому что именно в ней мы будем хранить историю посещений, записи студентов на самостоятельную работу. И благодаря этой таблице в информационной системе, мы сможем определять свободные компьютеры. Данная таблица имеет 6 атрибутов:

- visit_id(int) – является первичным ключом, именно по этому атрибуту можно определить запись в таблице;
- date(date) – дата посещения студента;
- timed_blocks(string) – интервал блоков времени, на которое записался студент, рабочее время компьютерного класса с 9-00 до 18-00, студент может записываться с промежутком в 15 минут, и если он запишется с 9-00 до 9-30, то в базу данных, это время запишется, как «0-2», таким

образом это поле может принимать интервал от 0 до 35, так как в расписании с 9-00 до 18-00 – 36 пятнадцатиминутных блоков;

- status(string) – состояние, в котором находится эта запись:
 - 1) («sign»)(Записался(лась)) – студент записался, но еще не пришел и время не просрочено;
 - 2) «attended»(Посетил(а)) – студент пришел в компьютерный класс;
 - 3) «absent»(Не посетил(а)) – студент не пришел в заявленное время;
 - 4) «undefined»(Не установлено) – вспомогательное состояние.
- comp_id(int) – идентификатор компьютера для определения за каким устройством будет работать или уже работал студент;
- user_id(int) – идентификатор студента, чтобы понять какой студент записался на занятие или уже его отработал.

Для полного представления процесса записи студента на СРС через информационную систему, рассмотрим диаграммы последовательностей с точки зрения лаборанта (Приложение А) и студента (Приложение Б).

Рассматривая диаграмму последовательностей с точки зрения лаборанта, можно заметить, что если студента нет в ИС, лаборант может внести его в базу данных, тем самым добавив нового пользователя, тогда этого студента можно будет записать на занятие в компьютерный класс, также при создании, пароль сгенерируется автоматически системой, и отправят на email студенту. Также, при выборе компьютера, дня и времени, пользователю, не будет предложено неактуальных для записи данных, тем самым ускоряя процесс обработки информации, и не предоставляя ненужную информацию. Даже, если студента не устраивает предложенный день или время, лаборант может вернуться на шаг назад, и выбрать другой компьютер.

Обратив внимание, на запись с точки зрения студента, пользователю не нужно регистрировать себя и вписывать свой логин каждый раз, чтобы записаться на занятие. Вся необходимая информация о студенте хранится в базе данных, и при необходимости система сама может определить, кто с ней сейчас работает. Поэтому запись сразу ИС начинается с определения необходимого компьютера. Как только студент выберет компьютер, дату и время, пользователя перенаправит на его существующие записи, для того, чтобы он был в курсе, когда и на какой

компьютер ему нужно приходить в ближайшее время. Также на этой вкладке можно удалить запись, если студент передумает идти на занятие.

Решая задачу записи на самостоятельную работу в компьютерный класс, может возникнуть множество ситуаций, когда студента не будут устраивать компьютер, день или время, поэтому в диаграмме присутствует ряд возвратов к тем вопросам, которые решались ранее, для того чтобы наиболее полно решить вопрос записи в удобное время.

Также рассмотрим некоторые тонкости при функции ИС «Отметить студента» (Рис. 12), когда студент приходит в назначенное время в компьютерный класс, и лаборант должен указать, что студент пришел в назначенное время. Для этого лаборант отмечает нужную запись, тем самым меняя статус данного посещения на «Посетил(а)» или «Не посетил(а)».

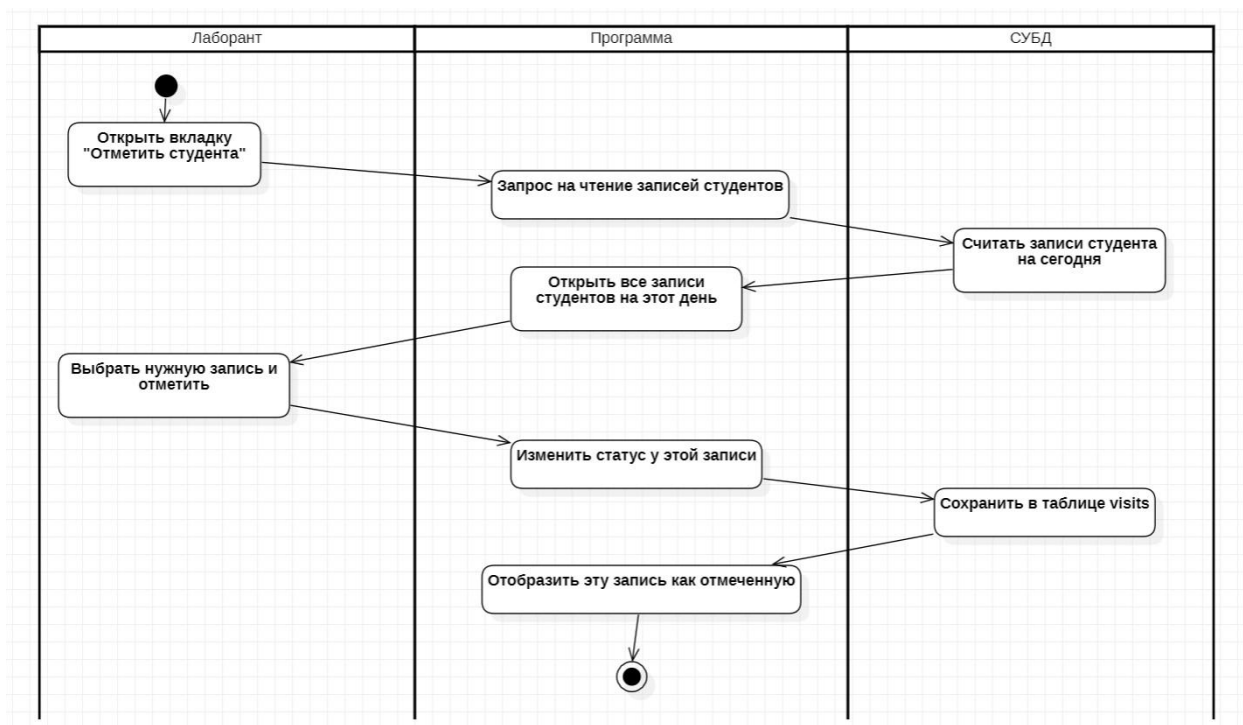


Рисунок 12 – Диаграмма последовательностей «Отметить студента»

2.4 Вывод

Итак, мы спроектировали поведение ИС в различных ситуациях, с помощью средств объектно-ориентированного моделирования UML, посмотрели на задачу с разных точек зрения, со стороны студента и со стороны лаборанта, задокументировали в виде диаграмм и, таким образом, они будут служить как пояснительная записка для людей, незнакомых с данной системой.

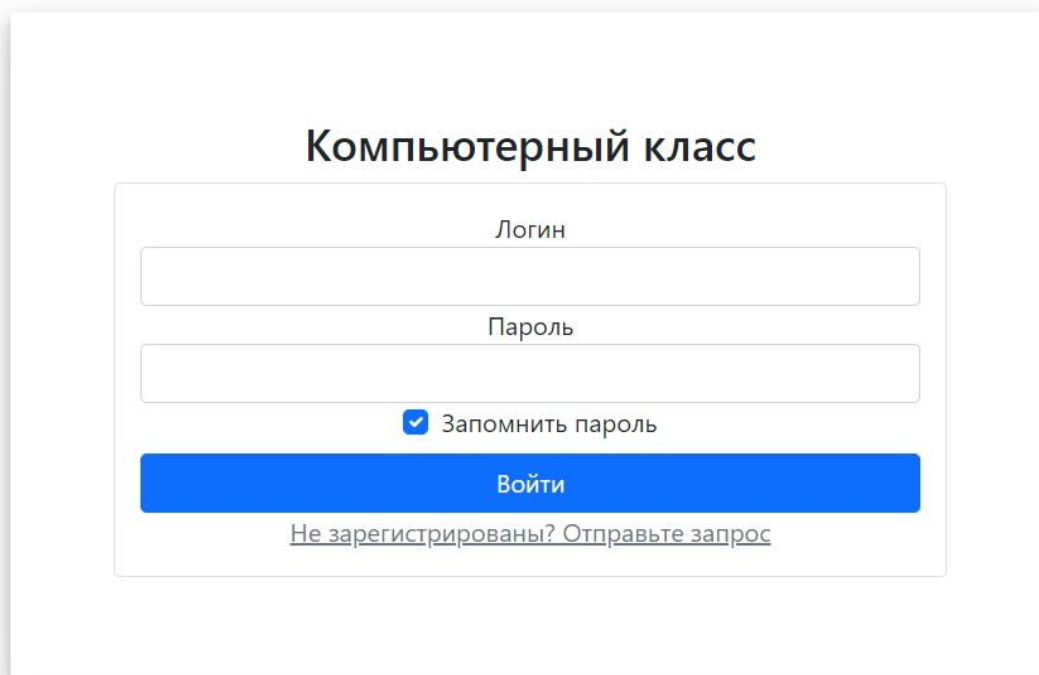
Глава 3 Разработка прототипа информационной системы учет СРС студента

В данной главе описан этап прототипирования интерфейса, выбор инструментальных средств, разработка прототипа информационной системы, а также представлены основные рекомендации по совершенствованию разработанной информационной системы

3.1.1 Интерфейс студента

Первым делом, рассмотрим интерфейс для удаленной записи студента через сайт, так как именно таким способом чаще всего будут взаимодействовать пользователи с ИС.

При открытии любого каталога сайта, студент первым увидит следующее окно (Рис. 13)



The image shows a web form titled "Компьютерный класс" (Computer Class). It contains a login section with two input fields: "Логин" (Login) and "Пароль" (Password). Below the password field is a checkbox labeled "Запомнить пароль" (Remember password) which is checked. A blue button labeled "Войти" (Login) is positioned below the checkbox. At the bottom of the form, there is a link: "Не зарегистрированы? Отправьте запрос" (Not registered? Send request).

Рисунок 13 – Авторизация

Если студент уже записывался ранее, ему необходимо ввести номер своего студенческого билета и пароль, тогда он сможет нажать кнопку «Войти» и перейти к непосредственно записи на СРС в компьютерный класс. Но в то же время, если студент впервые заходит на этот сайт, то тогда ему следует отправить запрос лаборанту, подтвердив свою личность, назвав студенческий номер билета, ФИО и

дату поступления в университет, а также почту для регистрации. Таким же образом можно подойти к лаборанту лично, чтобы он зарегистрировал студента в информационной системе для дистанционной записи через сайт. С другой стороны, если у ВУЗа есть возможность массово проинформировать студентов, то можно рассмотреть возможность генерации данных авторизации всех студентов в БД и сообщить студентам их пароли в других информационных системах для дальнейшего использования.

После авторизации студенту открывается окно с выбором подходящего компьютера для его конкретной задачи (Рис. 14).

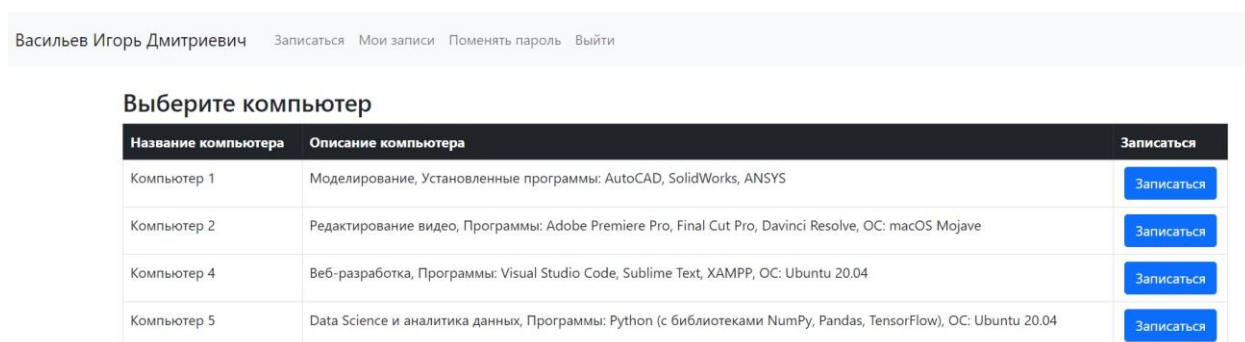


Рисунок 14 – Выбор ПК (Студент)

На данном рисунке (Рис. 14) видно, что в навигации, присутствует его ФИО, кроме этого присутствуют ссылки на другие вкладки сайта, а также система сразу же предлагает на выбор компьютеры, указывая необходимую информацию о них. В первом столбце указывается название компьютера, а во втором необходимое описание, для того чтобы студент смог подобрать компьютер в связи со своей задачей, а в третьем столбце кнопка «записаться» привязанная к каждому компьютеру. Также стоит отметить, что в данной таблице не будут отображаться компьютеры, которые не работоспособны.

В то же время присутствует возможность поменять электронную почту или пароль (Рис. 15). Этот пароль должен соответствовать требованиям безопасности [20], такими как:

- минимальная длина пароля 8 символов;
- не более 128 символов;
- как минимум одна заглавная и одна строчная буква;
- только латинские или кириллические буквы;
- как минимум одна цифра;

- только арабские цифры;
- без пробелов;

Профиль пользователя

Электронная почта

vasilevid@gmail.com

Изменить пароль

Новый пароль

Подтвердите новый пароль

Подтвердите пароль

Текущий пароль

Сохранить изменения

Рисунок 15 – Изменение почты или пароля

После выбора необходимого компьютера (Рис. 14), перед студентом отображается новая информация о выборе дня и времени посещения (Рис. 16).

Васильев Игорь Дмитриевич Записаться Мои записи Поменять пароль Выйти

Май 2023						
Назад			Вперед			
Пн	Вт	Ср	Чт	Пт	Сб	Вс
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Время			
9:00	9:15	9:30	9:45
10:00	10:15	10:30	10:45
11:00	11:15	11:30	11:45
12:00	12:15	12:30	12:45
13:00	13:15	13:30	13:45
14:00	14:15	14:30	14:45
15:00	15:15	15:30	15:45
16:00	16:15	16:30	16:45
17:00	17:15	17:30	17:45

Время записи

Записаться

Рисунок 16 – Запись

На представленном рисунке (Рис. 16), предложено студенту выбрать подходящий день и время. При выборе дня, меняются и часы записи (становятся активными или наоборот деактивируются), также стоит уточнить, что нельзя выбрать день, в котором нет свободной записи, в свою очередь на календаре слева

отображается текущий день (темно-синяя ячейка), свободные дни (зеленые ячейки), и недоступные дни (светло-серые). Вместе с этим, справа отображаются часы записи: светло-серым выделены часы, на которые записаться не представляется возможным, также у них отключена возможность выбрать, а все записи, где часы доступны, выделить можно. Выделение времени происходит по блокам, каждый блок – это 15 минут, т.е. 10-15 в ячейке справа, это время с 10-15 до 10-30. Сначала пользователь выделяется первое время, например 10-15, а потом следующее время, например 11-15, и таким образом пользователь записывается на время с 10-15 до 11-30. Таким образом, происходит запись на соответствующий день и время. В то же время, пользователи дублируется время над кнопкой «Записаться», чтобы пользователь не ошибся с записанным временем. Также всегда можно вернуться к выбору компьютера, если студента не устраивает расписание на этом компьютере.

После выбора даты и времени студенту предложено посмотреть его существующие записи (Рис. 17), а также предоставляется возможность отменить запись.

Запланированные посещения

Дата и Время	Компьютер	Описание	Отменить запись
с 15-00 до 17-00 22 мая, 2023	Компьютер 1	Моделирование, Установленные программы: AutoCAD, SolidWorks, ANSYS	Отменить запись
с 10-00 до 11-15 22 мая, 2023	Компьютер 2	Редактирование видео, Программы: Adobe Premiere Pro, Final Cut Pro, Davinci Resolve, ОС: macOS Mojave	Отменить запись
с 14-00 до 14-15 22 мая, 2023	Компьютер 5	Data Science и аналитика данных, Программы: Python (с библиотеками NumPy, Pandas, TensorFlow), ОС: Ubuntu 20.04	Отменить запись

Рисунок 17 – Запланированные посещения студента

3.1.2 Интерфейс лаборанта

Теперь рассмотрим интерфейс лаборанта. После авторизации, лаборанта перенаправит на его, на страницу «Отметить студента», так как пользователь с ролью «ROLE_WORKER», чаще всего будет взаимодействовать с системой именно с этой страницей (Рис. 18). На этой странице изображены 2 таблицы, в которых отмечено 2 группы студентов: Отмеченные и не отмеченные. Работник может

отменить пришел студент или пропустил занятие, в таблице не отмеченных, таким образом он попадет в таблицу отмеченных. Тем же методом, можно вернуть студента обратно в таблице неотмеченных.

1	Записать студента	Отметить студента	Отчет	Пользователи	Поменять пароль	Выйти
---	-------------------	-------------------	-------	--------------	-----------------	-------

Отметить студента за 22-05-2023						
Логин	ФИО	Время	Компьютер	Статус	Пришел	Пропустил
MM20122	Васильев Игорь Дмитриевич	с 13-00 до 15-00	Компьютер 1	Записался(лась)	Пришел	Пропустил
QQ	Черемных Александра Николаевна	с 15-00 до 17-00	Компьютер 1	Записался(лась)	Пришел	Пропустил

Студенты отмеченные						
Логин	ФИО	Время	Компьютер	Статус	Вернуть к неопределенным	
MM20122	Васильев Игорь Дмитриевич	с 15-00 до 15-30	Компьютер 5	Присутствовал(а)		Вернуть
QQ	Черемных Александра Николаевна	с 10-00 до 11-15	Компьютер 2	Присутствовал(а)		Вернуть
QQ	Черемных Александра Николаевна	с 14-00 до 14-15	Компьютер 5	Пропустил(а)		Вернуть

Рисунок 18 – Отметить студента

Запись студента на конкретный компьютер, день и время, со стороны лаборанта, представлена на Рис. 19 и на Рис. 20. При выборе компьютера, у работника университета есть возможность создать новый компьютер, а также удалить уже существующий, также у него есть возможность смотреть является ли компьютер рабочим или нет. Во время выбора даты и времени, лаборант обязан выбрать студента, с помощью поля ввода логина пользователя, также внизу календаря продублируется ФИО студента, чтобы понять, что это действительно тот студент, которого записывает лаборант.

1	Записать студента	Отметить студента	Отчет	Пользователи	Поменять пароль	Выйти
---	-------------------	-------------------	-------	--------------	-----------------	-------

Создать компьютер						
Название компьютера:						
Описание компьютера:						
<input type="checkbox"/> Компьютер рабочий						
Добавить компьютер						
Выберите компьютер						
Название компьютера	Описание компьютера	Записаться	Рабочий	Удалить		
Компьютер 1	Моделирование, Установленные программы: AutoCAD, SolidWorks, ANSYS	Записаться	<input type="checkbox"/>	Удалить		
Компьютер 2	Редактирование видео, Программы: Adobe Premiere Pro, Final Cut Pro, Davinci Resolve, ОС: macOS Mojave	Записаться	<input type="checkbox"/>	Удалить		
Компьютер 4	Веб-разработка, Программы: Visual Studio Code, Sublime Text, XAMPP, ОС: Ubuntu 20.04	Записаться	<input type="checkbox"/>	Удалить		
Компьютер 5	Data Science и аналитика данных, Программы: Python (с библиотеками NumPy, Pandas, TensorFlow), ОС: Ubuntu 20.04	Записаться	<input type="checkbox"/>	Удалить		
Компьютер 3	Аудио-производство, Программы: Ableton Live, Pro Tools, FL Studio, ОС: macOS Catalina	Записаться	<input type="checkbox"/>	Удалить		

Рисунок 19 – Выбор ПК (Лаборант)

1
Записать студента
Отметить студента
Отчет
Пользователи
Поменять пароль
Выйти

Логин пользователя:

Введите логин

Указать

Отменить

Май 2023

Назад

Вперед

Пн	Вт	Ср	Чт	Пт	Сб	Вс
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Время

9:00	9:15	9:30	9:45
10:00	10:15	10:30	10:45
11:00	11:15	11:30	11:45
12:00	12:15	12:30	12:45
13:00	13:15	13:30	13:45
14:00	14:15	14:30	14:45
15:00	15:15	15:30	15:45
16:00	16:15	16:30	16:45
17:00	17:15	17:30	17:45

Время записи

Записаться

Рисунок 20– Отметить студента (Лаборант)

Работник университета имеет возможность просмотра отчета о студентах на странице, которая представлена на Рис. 21. Эта страница позволяет задавать несколько настроек для фильтрации отчета:

- по логину: Работник может указать конкретного человека, по которому нужно произвести фильтрацию отчета. Это позволяет отобразить информацию только для выбранного студента.
- по компьютеру: Работник может выбрать определенный компьютер для фильтрации отчета. Таким образом, отчет будет содержать информацию только о студентах, которые использовали выбранный компьютер.
- по дате "от" и "до": Работник может задать диапазон дат для фильтрации отчета. Отчет будет содержать информацию только о студентах, у которых дата активности находится в указанном диапазоне.

Эти настройки фильтрации позволяют работнику университета получать более точные и специфические данные из отчета о студентах, в зависимости от требуемых критериев и условий. Также сформированный отчет можно скачать в формате электронных таблиц (.xlsx). Пример фильтрации и сгенерированного отчета продемонстрирован в приложении В.

Логин пользователя:
 Имя компьютера:
 Дата с:
 Дата по:

[Скачать \(xlsx\)](#)
[Сформировать отчет](#)

Активность	Логин	ФИО	Почта	Время и Дата	Компьютер	Статус
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 13-00 до 15-00 22 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 15-00 до 16-45 30 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 09-00 до 13-00 30 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 13-00 до 15-00 30 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 16-45 до 18-00 30 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	QQ	Черемных Александра Николаевна	chered@gmail.com	с 15-00 до 17-00 22 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 15-00 до 15-30 22 мая, 2023	id=27, Компьютер 5	Присутствовал(а)
<input checked="" type="checkbox"/>	QQ	Черемных Александра Николаевна	chered@gmail.com	с 10-00 до 11-15 22 мая, 2023	id=24, Компьютер 2	Присутствовал(а)
<input checked="" type="checkbox"/>	QQ	Черемных Александра Николаевна	chered@gmail.com	с 14-00 до 14-15 22 мая, 2023	id=27, Компьютер 5	Пропустил(а)

Рисунок 21– Отчет

Страница с изменением электронной почты и пароля у лаборанта идентична странице студента. Кроме того, в панели навигации присутствует кнопка "Выйти". При нажатии на эту кнопку, пользователь выходит из аккаунта и перенаправляется на страницу авторизации, где ему требуется ввести учетные данные для входа в систему снова.

3.2 Выбор средств разработки ИС

Необходимо начинать выбор с СУБД потому, что она является фундаментом всей информационной системы, и определение ее характеристик, функций и возможностей помогает определить последующие шаги при выборе других компонентов и инструментов для разработки ИС.

3.2.1 Система управления базой данных

Существует множество СУБД, которые можно использовать для создания информационной системы, однако, для данной работы лучше всего подойдет PostgreSQL [21]. Это бесплатная и открытая реляционная СУБД, которая обладает высоким уровнем надежности и безопасности благодаря использованию транзакций, механизмов резервного копирования и аутентификации пользователей [22]. PostgreSQL также обладает широким набором инструментов и расширений, что позволяет настроить базу данных под любые требования проекта. Благодаря многопоточной обработке эта СУБД может распараллеливать выполнение запросов и ускорять работу с базой данных. Более того, она обладает хорошей масштабируемостью и производительностью, что позволяет ей обрабатывать

большой объем данных и обеспечивать высокую скорость выполнения запросов. Таким образом, PostgreSQL является идеальным выбором для создания информационной системы об учете студентов для записи в компьютерный класс.

3.2.2 Серверная часть

В информационной системе для учета работы студентов в компьютерном классе, наиболее приоритетным составляющими является безопасность, производительность, надежность и количество разработанных библиотек. Для сочетания всех свойств системы, были выбраны следующие технологии: язык программирования Java, а также фреймворк Spring Boot, библиотека Spring JPA, библиотека Spring Security, обработчик шаблонов FreeMarker и другие библиотеки. Важно отметить, что все эти инструменты являются бесплатными и могут быть использованы в коммерческих проектах без необходимости платить за лицензии.

Java [23] является одним из наиболее широко используемых языков программирования в мире и предлагает множество преимуществ для разработки приложений. В частности, этот язык обеспечивает высокую производительность, надежность и безопасность, что является важным для систем, используемых множеством пользователей. Кроме того, Java имеет масштабируемую архитектуру, что позволяет легко расширять и изменять функциональность приложения. Также в Java имеет масштабируемую архитектуру, что позволяет добавлять новые компоненты с минимальными затратами на изменение основного кода приложения.

Spring Boot [24], в свою очередь, является фреймворком, который был разработан для упрощения процесса создания веб-приложений на Java. Он предоставляет широкий набор инструментов и библиотек, которые сокращают время и усилия, необходимые для конфигурации приложения, и позволяют быстро разрабатывать функциональность.

Spring JPA [25] - это библиотека Spring Framework, которая предоставляет удобный способ работать с базами данных в Java приложениях. Она предоставляет высокоуровневый API для управления сущностями. Использование Spring JPA позволяет значительно ускорить и упростить процесс разработки и снизить количество ошибок, связанных с работой с базами данных.

Spring Security [26]- это фреймворк безопасности, предназначенный для обеспечения защиты веб-приложений, разработанных на платформе Spring. Spring

Security предоставляет множество видов защиты для веб-приложений, включая:

- аутентификацию: Проверка подлинности пользователей для доступа к защищенным ресурсам, используя различные виды аутентификации, такие как форма входа, аутентификация на основе токена и др;
- авторизацию: Управление правами доступа к защищенным ресурсам для различных категорий пользователей, определяя права доступа на основе ролей, принадлежности к группам и других критериев;
- защиту от атак типа CSRF: Предотвращение атак, связанных с межсайтовой подделкой запросов, путем генерации случайных токенов и проверки их валидности при каждом запросе;
- защиту от атак типа XSS: Предотвращение атак, связанных с внедрением вредоносного кода на стороне клиента, путем применения различных механизмов фильтрации и экранирования данных;
- защиту паролей: Хеширование паролей пользователей и использование солей для защиты от атак типа подбора пароля;
- межсерверную защиту: Предоставление механизмов безопасной коммуникации между серверами, таких как SSL и HTTPS;
- логирование и мониторинг: Предоставление инструментов для записи и анализа событий безопасности, а также мониторинга работы приложения в целом.

FreeMarker [27]- это бесплатный обработчик шаблонов, который позволяет разработчикам создавать динамические HTML страницы и другие виды текстовых документов в Java приложениях. Он интегрируется с фреймворком Spring и позволяет разделять логику и представление, улучшая читаемость и поддерживаемость кода. Использование FreeMarker в фреймворке Spring обеспечивает высокую производительность и масштабируемость, так как он может кэшировать результаты обработки шаблонов и быстро отображать динамические страницы для пользователей. Это делает FreeMarker одним из лучших выборов для создания пользовательского интерфейса в веб-приложениях на базе фреймворка Spring Boot.

Также, был использован git [28] для удобства отслеживания версионности приложения, а библиотека Lombok [29] для упрощения написания кода и ускорения

разработки.

В итоге, благодаря объединению этих технологий, конечное приложение будет удовлетворять потребностям пользователя, а также соответствовать нормам безопасности и надежности, а также повысить читаемость и поддерживаемость кода, что позволит дальнейшее развитие проекта.

3.2.3 Клиентская часть

Для реализации клиентской части веб-приложения, было решено использовать технологии HTML, CSS и JavaScript.

Для ускорения процесса разработки клиентской части веб-приложения был использован Bootstrap. Bootstrap [30] — это популярный фреймворк, который позволяет быстро создавать адаптивные и красивые пользовательские интерфейсы. Фреймворк содержит готовые CSS-стили, JavaScript-компоненты и шаблоны, которые можно использовать для быстрого создания интерфейса приложения. Преимуществами Bootstrap являются удобство в использовании, высокая скорость разработки и возможность создания адаптивного интерфейса для различных устройств.

3.3 Создание базы данных

Созданная база данных в PostgreSQL выглядит следующим образом (Рис.22). Связи между таблицами, представлены таким образом:

- Таблица «users» связана с таблицей «visit» по полю user_id.
- Таблица «computer» связана с таблицей «visit» по полю comp_id.

При попытке удаления информации из таблиц «users» и «computer», если есть записи с этим идентификатором в таблице, visits, запись не удалится и выдаст ошибку, и удаление будет неуспешным. Это поможет поддерживать целостность данных и избегать неконсистентности информации в базе данных.

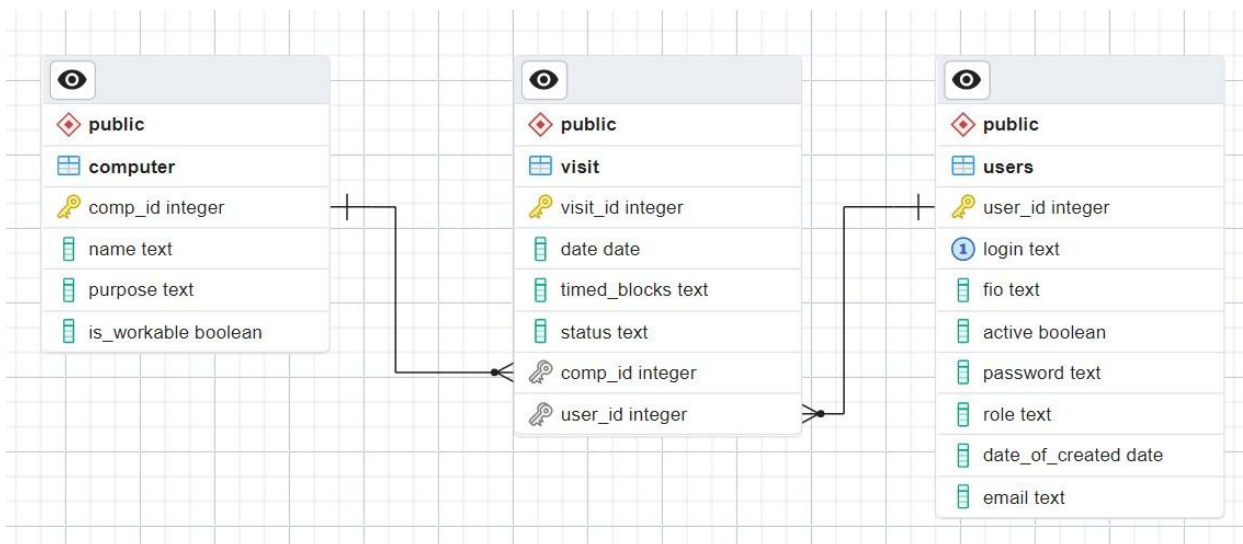


Рисунок 22 – База данных студентов

3.4 Реализация прототипа ИС

Разработанный прототип имеет множество классов, которые изображены на диаграмме классов в приложении Ж, большинство из них представлены классами, которые реализуют функциональность, а именно:

- Controller отвечает за управление потоком приложения. Он принимает запросы пользователей, обрабатывает их и затем отвечает соответствующим образом клиенту. Для получения или изменения данных он взаимодействует с уровнем Service.
- Service – это бизнес-логика приложения обрабатывается сервисным уровнем. Данные, полученные из слоя DAO, обрабатываются и затем передаются обратно контроллеру. Для того чтобы получить или изменить данные, он также взаимодействует с уровнем DAO.
- DAO – это взаимодействие с базой. Он возвращает данные в класс Service после получения их из базы данных. При необходимости он также сохраняет данные в базе данных.
- DTO – это объект из базы данных. С помощью него производится обмен информации между слоями, а также осуществляется чтение и запись в базу данных.

При разработке во фреймворке Spring boot используется паттерн проектирования Model-View-Controller (MVC) [31], и поставляемые классы соответствуют этому паттерну. Каждый слой взаимодействует только с выше- или нижележащим слоем, что делает приложение более модульным и более простым в

обслуживании.

При рассмотрении логики информационной системы, отдельное внимание уделим организации защиты доступа к веб-сайту. Благодаря настройке () класса SecurityConfig, каждого пользователя, заходящего неавторизованным на любой каталог сайта, автоматически перенаправляет на страницу авторизации, где его просят ввести логин и пароль. После того, как пользователь аутентифицировался, сервер генерирует csrf-токен (<https://www.baeldung.com/spring-security-csrf>), и теперь при отправке post-запросов от клиента к серверу, вместе с данными, передается csrf-токен, который используется для проверки того, что аутентифицированный пользователь действительно является лицом, выполняющим запросы к приложению. Поскольку, токен хранится в сессии пользователя и изменяется каждый раз при повторном создании сессии, вредоносное приложение не может получить к нему доступ.

После прохождения формы авторизации, пользователю становятся доступны только те веб-страницы, которые относятся к его роли. Например, пользователю с ролью “ROLE_STUDENT” доступны следующие веб-документы, не считая формы login.html:

- computers.html (выбор компьютера)
- shedule.html (выбор даты и времени)
- visits.html (просмотр своих записей)
- profile.html (изменение почты/пароля)

В свою очередь, в базе данных хранятся логин и пароль, захешированный функцией Bcrypt [32]. Bcrypt - это популярная криптографическая функция хэширования, которая часто используется для безопасного хранения паролей в базах данных. Это распространенный вариант хеширования паролей во многих приложениях, поскольку он не требует больших вычислительных затрат и устойчив к атакам методом перебора.

Теперь рассмотрим, процесс записи студента в компьютерный класс, с точки зрения работы сервера. При входе на сайт, после авторизации, его встречает таблица с выбором компьютеров, эта таблица сформирована путем чтения всех компьютеров, у которых поле «is_workable» принимает значение «True», так как студентам ненужно видеть неработающие компьютеры. При выборе компьютера

студент нажимает кнопку около нужного устройства, и переходит на следующий документ, в котором изображен календарь и время для записи. В это же время на сервере, считываются все записи, начиная с текущей даты, и с помощью метода `getShedulesByCompIdOnDate`, создается `HashTable`, в котором хранятся значения дата и занятые дни, если этот день полностью занят (нет свободного времени), то тогда напротив даты стоит значение «out». После этого, этот `HashTable` передается в шаблон `FreeMarker` и обрабатывается двумя скриптами `showCalendar` и `showTimeTable`, для отображения информации полученной из базы данных, пользователю. Следующим шагом, пользователю следует выбрать подходящую дату и время, но есть ограничения, которые не позволят выбрать неподходящую дату, а именно недоступны следующие варианты:

1. даты в календаре, которые находятся раньше сегодняшней,
2. суббота и воскресенье,
3. даты в которых все время уже выбрано другими студентами,
4. время, которое превосходит 4 часа за одну запись,
5. время, в интервале которого, уже есть другая запись студента

Во время выбора времени, студенту также дублируется время записи на странице. После решения даты и времени из доступных, пользователь нажимает на кнопку «Записаться», и информация о выбранных времени и дате отправляются `post`-запросом на сервер и сохраняются в базе данных.

Для работника компьютерного класса, если он хочет записать студента в компьютерный класс, на странице с выбором даты и времени добавляется сверху форма с вводом логина пользователя, а в остальном порядок действий тот же.

На странице с собственными записями студента, есть таблица записей, где он может отменить запись, отправляя `post`-запрос с идентификатором записи серверу.

Обратим внимание на работу работника компьютерного класса и его взаимодействие системой с точки зрения работы сервера. После входа в систему, перед пользователем открывается две таблицы с отмеченными и не отмеченными посещениями студентов. В первой таблице перед каждой записью располагаются кнопки «Пришел» и «Пропустил», во второй соответственно кнопка «Вернуть», путем нажатия пользователь генерирует запрос серверу в следующем виде

«/check/{status}/{visit_id}», где {status} – это статус, на который нужно поменять запись, а {visit_id} – идентификатор записи. Далее этот запрос отправляется записей в функцию checkUser соответственно, где изменяется статус записи.

Следующим шагом рассмотрим формирование отчета для работника. Путем выбора логина студента, имени компьютера, даты с и даты по, пользователь может сформировать отчет обо всех записях, по заданным критериям, если пользователь предпочел не вводить данных, тогда это поле не учитывается при выборе. Таким образом SQL-запрос к базе данных выглядит следующим образом (Лист. 1).

```
SELECT a.* FROM shedule a
LEFT JOIN User u ON a.user_id = u.id
LEFT JOIN Computer c ON a.comp_id = c.id
WHERE 1=1
AND (u.login LIKE :login OR :login IS NULL OR :login = '')
AND (c.compId = :compId OR :compId IS NULL OR :compId = 0)
AND (a.date >= :dateFrom OR DATE(:dateFrom) IS NULL)
AND (a.date <= :dateTo OR DATE(:dateTo) IS NULL)
```

Листинг 1 – Запрос к базе данных

В дополнение к этому, пользователь может скачать отчет в формате электронных таблиц (.xlsx), путем отправки post-запроса. На сервере создается объект Workbook из библиотеки «org.apache.poi» в функции createXslxFile, далее из объекта Shedule преобразуется к странице электронной таблицы и сохраняется в Workbook. Вторым шагом, этот объект сериализуется в битовый формат и отправляется запросом пользователю с «Content-Type: application/octet-stream» и «Content-Disposition: attachment», что позволяет браузеру автоматически начать скачивание файла пользователю. Сформированная таблица в точности выглядит, как таблица из веб-документа.

Работник компьютерного класса может управлять пользователями. На странице users.html, может редактировать, удалять и добавлять новых пользователей. При регистрации пользователя, пароль отправляется на почту, указанную при регистрации и сохраняется в базе данных, в зашифрованном виде с помощью BCrypt, чем обеспечивает безопасное хранение паролей.

3.5 Рекомендации по совершенствованию ИС

Эффективное управление записями учащихся имеет решающее значение для образовательных учреждений, чтобы обеспечить бесперебойное функционирование административных процессов и предоставить точную и достоверную информацию. В условиях компьютерного класса, где записи учащихся хранятся и доступ к ним осуществляется в цифровом виде, становится необходимым постоянное совершенствование информационной системы для удовлетворения меняющихся потребностей образовательных учреждений.

Первая рекомендация предлагает включить в информационную систему функцию поиска нерабочих дней. Интегрировав надежный источник, например, «isdayoff.ru» [33], система сможет предоставлять точную информацию о нерабочих днях, праздниках и выходных. Эта функция предотвратит ввод ошибочных данных студентами в дни, когда компьютерный класс закрыт, тем самым повысив, убрав недостоверность студенческих записей в нерабочие дни.

Вторая рекомендация посвящена использованию запросов по месяцам с помощью jQuery Ajax [34]. Это усовершенствование позволит системе эффективнее получать записи студентов за определенные периоды времени. При выборе клиентское приложение будет получать ограниченную информацию, об одном месяце, а не обо всех записях

Третья рекомендация подчеркивает необходимость внедрения системы самостоятельной регистрации пользователей. После регистрации электронное письмо с основной информации о себе будет отправлено назначенному сотруднику компьютерного класса, который проверит регистрационные данные и подтвердит личность пользователя. Эта мера облегчит внесение новых пользователей в систему.

3.6 Вывод

В главе 3 был спроектирован интерфейс, включая подробное описание всех используемых элементов управления. Каждый инструмент был выбран на основе конкретных причин и преимуществ, которые он предлагает. Кроме того, в главе подчеркивалась важность безопасности информационной системы и приведены способы защиты информационной системы от злоумышленников. Наконец, разработан прототип ИС и определены рекомендации по ее совершенствованию.

Заключение

В результате проделанной работы была спроектирована, разработана и задокументирована информационная система учета времени самостоятельной работы студентов в компьютерном классе. Данная информационная система может быть использована, как для студентов ВУЗа, так и для учеников школ, для записи на самостоятельную работу в компьютерном классе.

В ходе выпускной квалификационной работы были реализованы следующие задачи:

- 1) изучена существующая литература по теме исследования;
- 2) изучены аналоги системы учета времени;
- 3) проведен сравнительный анализ этих систем;
- 4) выявлены и определены требования к разрабатываемой ИС;
- 5) проведен анализ существующих инструментов для проектирования ИС;
- 6) определен набор инструментальных средств в соответствии с требованиями;
- 7) спроектирована модель ИС;
- 8) разработан прототип интерфейса ИС,
- 9) разработан прототип ИС;
- 10) сформированы рекомендации по совершенствованию ИС.

Подводя итоги, можно сказать, что все поставленные задачи были выполнены. Цель выпускной квалификационной работы, а именно проектирование, разработка и документирование ИС учета времени студентов для самостоятельной работы в компьютерном классе была достигнута. В будущем планируется продолжить развитие данного проекта с целью его внедрения в реальную образовательную среду.

Библиографический список

- [1] ГОСТ 34.321-96 Информационные технологии. Система стандартов по базам данных. Эталонная модель управления данными. URL: <https://docs.cntd.ru/document/1200017662> (дата обращения: 22.05.2022)
- [2] Учебный план. Направление 09.03.02. Информационные системы и технологии направленность (профиль) Безопасность информационных систем степень бакалавр [Электронный ресурс]. – 2020. – URL: http://www.psu.ru/files/docs/obrazovanie/bachelors/2020/up/09_03_02_bis_up.pdf (дата обращения 03.03.2022)
- [3] Кудрявцева Т.А., Пуртова Т.И., Соколова М.Г. Проблемы организации самостоятельной работы студентов в условиях реализации федеральных государственных образовательных стандартов среднего профессионального образования третьего поколения // Инновационное развитие профессионального образования. – 2015. – N 2. – С. 39-42.
- [4] Ананьина Н. В. Организация самостоятельной работы студентов в условиях реализации ФГОС // Образование. Карьера. Общество. – 2013. – N 4. – С. 51-55.
- [5] 1С:Университет. Возможности продукта [Электронный ресурс]. – 2022. – URL: <https://solutions.1c.ru/catalog/university/features> (дата обращения 03.03.2022)
- [6] 1С:Университет. Приобретение продукта [Электронный ресурс]. – 2022. – URL: <https://solutions.1c.ru/catalog/university/buy> (дата обращения 03.03.2022)
- [7] Платформа «Цифровой колледж Подмосковья». Руководство пользователя [Электронный ресурс]. – 2022. – URL: https://e-learning.tsppk-mo.ru/seo/help/module_2_2_4.php?st=MA%3D%3D&sct=MA%3D%3D&mw=NTg3&ms=AwAAAAAABAI%3D (дата обращения 05.03.2022)
- [8] Yclients. Описание продукта [Электронный ресурс]. – 2022. – URL: <https://www.yclients.com/ru> (дата обращения 05.03.2022)
- [9] Универсальная система учета [Электронный ресурс]. – 2022. – URL: usu.kz (дата обращения 05.03.2022)
- [10] Методологии моделирования бизнес-процессов [Электронный ресурс].

– 2022. – URL: <https://hsbi.hse.ru/articles/metodologii-modelirovaniya-biznes-protsessov-osnovnye-metody-i-metodiki/> (дата обращения 05.03.2022)

[11] Основные методологии обследования организаций. Стандарт IDEF0. [Электронный ресурс]. – 2022. – URL: <https://www.cfin.ru/vernikov/idef/idef0.shtml> (дата обращения 05.03.2022)

[12] Нотация BPMN 2.0: ключевые элементы и описание [Электронный ресурс]. – 2021. – URL: <https://www.comindware.com/ru/blog-нотация-bpmn-2-0-элементы-и-описание/> (дата обращения 05.03.2022)

[13] Бабич А. Введение в UML [Электронный ресурс]. – 2022. – URL: <https://intuit.ru/studies/courses/1007/229/info> (дата обращения 10.03.2022)

[14] Вендров. А.М. CASE-технологии. Современные методы и средства проектирования информационных систем [Электронный ресурс]. – 2022. – URL: <http://citforum.ru/database/case/introduction.shtml> (дата обращения 10.03.2022)

[15] StarUML documentation [Электронный ресурс]. – 2022. – URL: <https://docs.staruml.io/> (дата обращения 10.03.2022)

[16] Classic Violet Review [Электронный ресурс]. – URL: <https://horstmann.com/violet/> (дата обращения 10.03.2022)

[17] Diagram Designer [Электронный ресурс]. – 2020. – URL: <https://logicnet.dk/DiagramDesigner/> (дата обращения 10.03.2022)

[18] Проектирование UML (ArgoUML) [Электронный ресурс]. – 2022. – URL: <https://studwood.net/1707669/informatika/proektirovanie> (дата обращения 10.03.2022)

[19] Леоненков А. Нотация и семантика языка UML [Электронный ресурс]. – 2022. – URL: <https://intuit.ru/studies/courses/32/32/info> (дата обращения 10.03.2022)

[20] Требования к паролю // MyKaspersky URL: <https://support.kaspersky.com/KPC/1.0/ru-RU/183862.htm> (дата обращения: 22.05.2023).

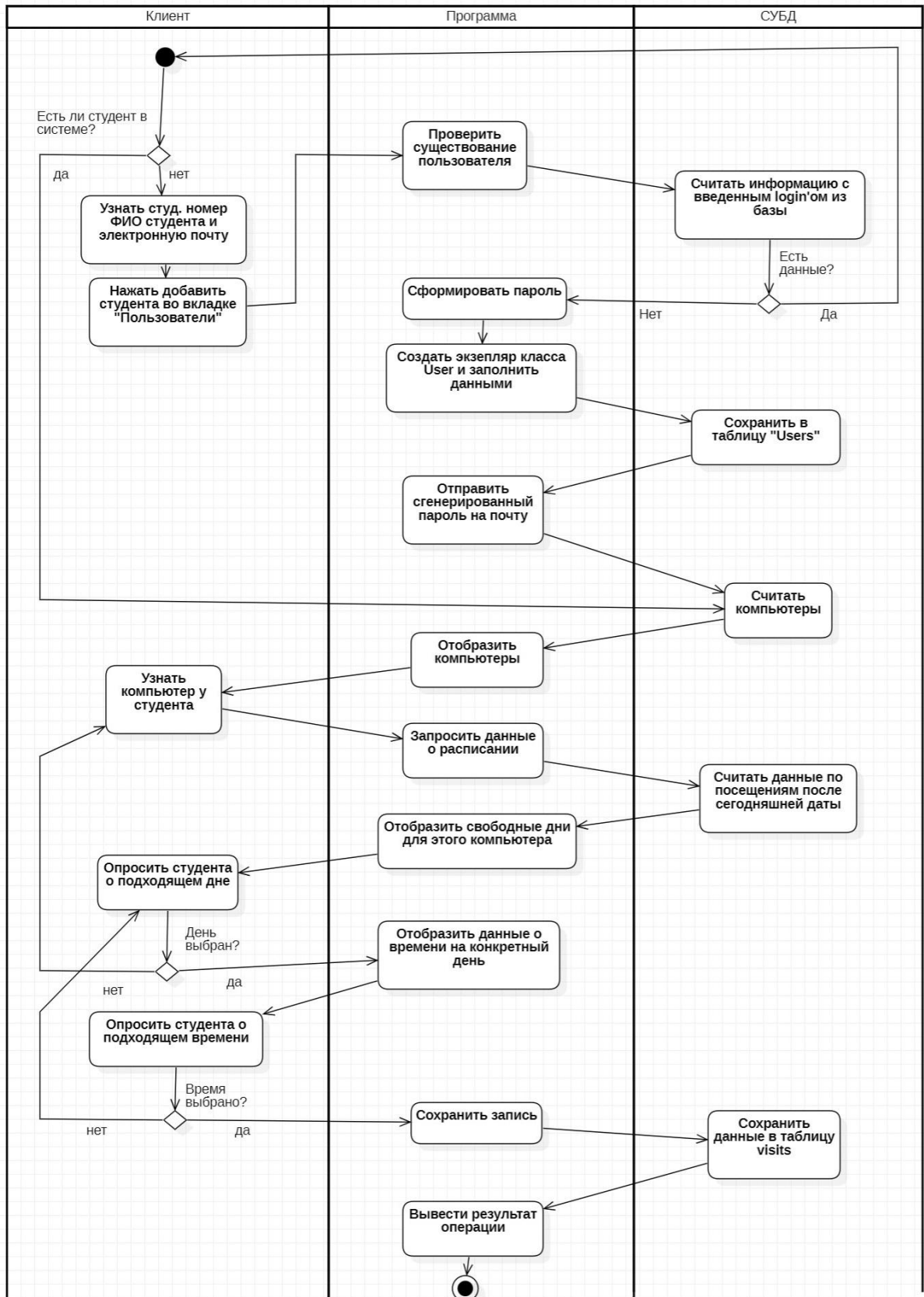
[21] Documentation // PostgreSQL URL: <https://www.postgresql.org/docs/> (дата обращения: 22.05.2023).

[22] Чем PostgreSQL лучше других SQL баз данных с открытым исходным кодом. // Хабр URL: <https://habr.com/ru/articles/282764/> (дата обращения: 22.05.2023).

- [23] Java Documentation // Oracle URL: <https://docs.oracle.com/en/java/> (дата обращения: 22.05.2023).
- [24] Spring Boot Reference Documentation // Spring URL: <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/> (дата обращения: 22.05.2023).
- [25] Spring Data JPA - Reference Documentation // Spring URL: <https://docs.spring.io/spring-data/jpa/docs/current/reference/html/> (дата обращения: 22.05.2023).
- [26] Spring Security // Spring URL: <https://docs.spring.io/spring-security/reference/> (дата обращения: 22.05.2023).
- [27] Apache FreeMarker Manual // FreeMarker URL: <https://freemarker.apache.org/docs/> (дата обращения: 22.05.2023).
- [28] git // git URL: <https://git-scm.com/> (дата обращения: 22.05.2023).
- [29] Project Lombok // Project Lombok URL: <https://projectlombok.org/> (дата обращения: 22.05.2023).
- [30] Bootstrap // Bootstrap URL: <https://getbootstrap.com/> (дата обращения: 22.05.2023).
- [31] Model View Controller (MVC) Design Pattern in Java // Java Guides URL: <https://www.javaguides.net/2019/08/model-view-controller-mvc-design-in-java.html> (дата обращения: 22.05.2023).
- [32] Bcrypt Step by Step // medium URL: <https://medium.com/bootdotdev/bcrypt-step-by-step-99dd76a3b16a> (дата обращения: 22.05.2023).
- [33] API производственного календаря // isDayOff() URL: <https://www.isdayoff.ru/> (дата обращения: 22.05.2023).
- [34] jQuery.ajax() // jQuery URL: <https://api.jquery.com/jquery.ajax/> (дата обращения: 22.05.2023).

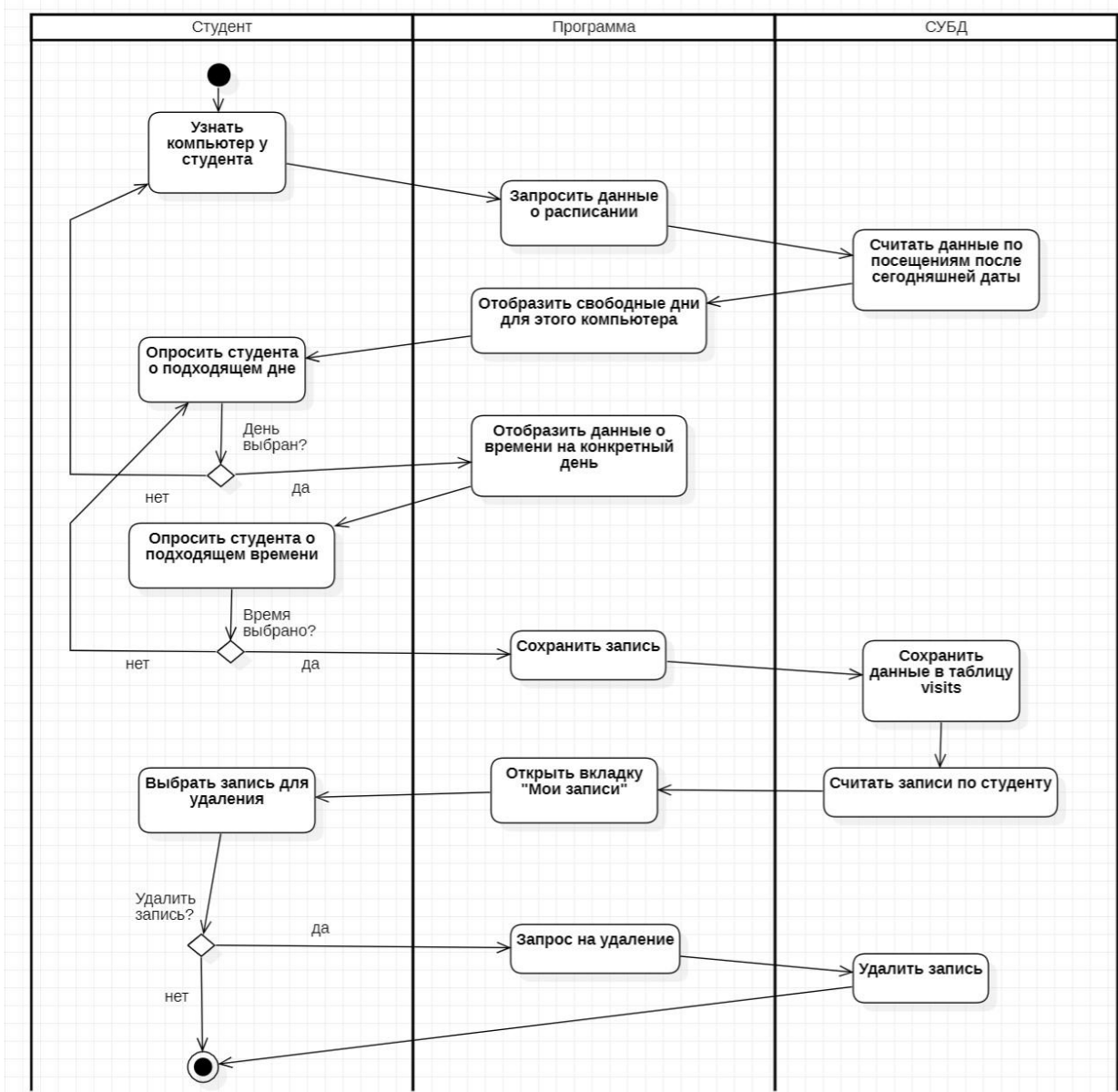
ПРИЛОЖЕНИЕ А

Диаграмма последовательности «Записать студента». Лаборант



ПРИЛОЖЕНИЕ Б

Диаграмма последовательности «Записаться». Студент.



ПРИЛОЖЕНИЕ В

Отчет. Сортировка по логину.

Логин пользователя: Имя компьютера: Дата с: Дата по:

[Скачать \(xlsx\)](#) [Сформировать отчет](#)

Активность	Логин	ФИО	Почта	Время и Дата	Компьютер	Статус
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 13-00 до 15-00 22 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 15-00 до 16-45 30 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 09-00 до 13-00 30 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 13-00 до 15-00 30 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 16-45 до 18-00 30 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 15-00 до 15-30 22 мая, 2023	id=27, Компьютер 5	Присутствовал(а)

Отчет. Сортировка по компьютеру.

Логин пользователя: Имя компьютера: Дата с: Дата по:

[Скачать \(xlsx\)](#) [Сформировать отчет](#)

Активность	Логин	ФИО	Почта	Время и Дата	Компьютер	Статус
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 13-00 до 15-00 22 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 15-00 до 16-45 30 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 09-00 до 13-00 30 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 13-00 до 15-00 30 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 16-45 до 18-00 30 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	QQ	Черемных Александра Николаевна	chered@gmail.com	с 15-00 до 17-00 22 мая, 2023	id=23, Компьютер 1	Записался(лась)

Отчет. Сортировка по дате.

Логин пользователя: Имя компьютера: Дата с: Дата по:

[Скачать \(xlsx\)](#) [Сформировать отчет](#)

Активность	Логин	ФИО	Почта	Время и Дата	Компьютер	Статус
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 13-00 до 15-00 22 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	QQ	Черемных Александра Николаевна	chered@gmail.com	с 15-00 до 17-00 22 мая, 2023	id=23, Компьютер 1	Записался(лась)
<input checked="" type="checkbox"/>	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 15-00 до 15-30 22 мая, 2023	id=27, Компьютер 5	Присутствовал(а)
<input checked="" type="checkbox"/>	QQ	Черемных Александра Николаевна	chered@gmail.com	с 10-00 до 11-15 22 мая, 2023	id=24, Компьютер 2	Присутствовал(а)
<input checked="" type="checkbox"/>	QQ	Черемных Александра Николаевна	chered@gmail.com	с 14-00 до 14-15 22 мая, 2023	id=27, Компьютер 5	Пропустил(а)

Отчет. Формирование xlsx.

	A	B	C	D	E	F	G	H	I	
1	Активность	Логин	ФИО	Почта	Время	Дата	Id компьютера	Имя компьютера	Статус	
2	ИСТИНА	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 13-00 до 15-00	2023-05-22 00:00:00.0		23 Компьютер 1	sign	
3	ИСТИНА	QQ	Черемных Александра Николаевна	chered@gmail.com	с 15-00 до 17-00	2023-05-22 00:00:00.0		23 Компьютер 1	sign	
4	ИСТИНА	MM20122	Васильев Игорь Дмитриевич	vasilevid@gmail.com	с 15-00 до 15-30	2023-05-22 00:00:00.0		27 Компьютер 5	attended	
5	ИСТИНА	QQ	Черемных Александра Николаевна	chered@gmail.com	с 10-00 до 11-15	2023-05-22 00:00:00.0		24 Компьютер 2	attended	
6	ИСТИНА	QQ	Черемных Александра Николаевна	chered@gmail.com	с 14-00 до 14-15	2023-05-22 00:00:00.0		27 Компьютер 5	absent	
7										
8										

ПРИЛОЖЕНИЕ Г

Диаграмма классов приложения

