МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное автономное образовательное учреждение высшего образования «Самарский национальный исследовательский университет имени академика С.П. Королева»  
(Самарский университет)

Институт информатики и кибернетики

Кафедра программных систем

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

**«РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ ОБУЧЕНИЯ ПО СИСТЕМЕ ЛЕЙТНЕРА С РЕАЛИЗАЦИЕЙ АЛГОРИТМА ИНТЕРВАЛЬНОГО ПОВТОРЕНИЯ»**

по направлению подготовки 02.03.02

Фундаментальная информатика и информационные технологии

(уровень бакалавриата)

направленность (профиль) «Информационные технологии»

Обучающийся А.А. Алёнушка

(подпись, дата)

Руководитель ВКР

к.т.н., доцент О.А. Гордеева

(подпись, дата)

Нормоконтролер Е.В. Сопченко

(подпись, дата)

Самара 2025

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное   
образовательное учреждение высшего образования  
 «Самарский национальный исследовательский университет   
имени академика С.П. Королева»

Кафедра программных систем

УТВЕРЖДАЮ

Заведующий кафедрой

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С.В. Востокин

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2025 г.

**задание на выпускную квалификационную работу (ВКР)**

обучающемуся Алёнушка Александру Александровичу

группы 6401-020302D

1. Тема работы: Разработка веб-приложения для обучения по системе Лейтнера с реализацией алгоритма интервального повторения

Исходные данные: Объект автоматизации — процесс самостоятельного обучения с использованием карточек по методу Лейтнера. Задача — автоматизировать процесс обучения с применением алгоритма интервального повторения, обеспечивающего эффективное усвоение информации за счёт оптимального распределения повторений во времени. Методы решения — анализ предметной области, применение алгоритмов интервального повторения, проектирование клиент-серверного веб-приложения. Технические средства — веб-технологии: Java (Spring Boot) для серверной части, PostgreSQL для хранения данных, React для фронтенда, GraphQL для организации гибкого взаимодействия между фронтендом и бэкендом, Liquibase для управления версиями и миграциями базы данных.

2. Перечень вопросов, подлежащих разработке в ВКР:

1. Провести анализ предметной области: Основные понятия интервального повторения, система Лейтнера, флеш-карточки, модель SM-2, процессы обучения и запоминания информации.
2. Сделать обзор систем-аналогов в области существующих решений для интервального повторения и обучения с карточками.
3. Разработать проект системы с использованием методологии структурного и объектно-ориентированного проектирования
4. Разработать и реализовать информационное и программное обеспечение
5. Провести тестирование и отладку разработанного веб-приложения для обучения по системе Лейтнера с реализацией алгоритма интервального повторения

3.Дата выдачи задания: «24» апреля 2025г.

4.Срок представления на кафедру законченной ВКР: « 03 »июня 2025г.

Руководитель ВКР

к.т.н., доцент, доцент кафедры программных систем О.А. Гордеева

« 24 » 04 2025 г.

Задание принял к исполнению А.А. Алёнушка

« 24 » 04 2025 г.

РЕФЕРАТ

Пояснительная записка 101 с, 55 рисунков, 10 таблиц, 31 источник, 5 приложений. (ВСЕ ОЬЪЕМЫ УКАЗЫВАЮТСЯ С УЧЕТОМ ПРИЛОЖЕНИЙ)

Графическая часть: 34 слайда презентации PowerPoint.

ИНТЕРВАЛЬНОЕ ПОВТОРЕНИЕ, СИСТЕМА ЛЕЙТНЕРА, ОБУЧАЮЩИЕ КАРТОЧКИ, ВЕБ-ПРИЛОЖЕНИЕ, АЛГОРИТМ SM-2, GRAPHQL, SPRING BOOT, REACT, LIQUIBASE, POSTGRESQL.

Цель работы – разработать автоматизированную систему для обучения по методу Лейтнера с реализацией алгоритма интервального повторения, обеспечивающую эффективное запоминание информации с помощью обучающих карточек.

В процессе работы были разработаны алгоритмы и соответствующее программное обеспечение, позволяющее пользователю создавать, редактировать и повторять карточки в соответствии с принципами интервального повторения. Система автоматически рассчитывает оптимальные интервалы для повторения и отслеживает прогресс пользователя.

Система разработана на языке Java с использованием фреймворка Spring Boot, а также технологий GraphQL, Liquibase (для миграций базы данных) и React (для пользовательского интерфейса). Функционирует под управлением операционных систем семейства Windows и Linux. Доступ к данным осуществляется с помощью СУБД PostgreSQL.

СОДЕРЖАНИЕ

[Введение 7](#_Toc198254296)

[1 Описание и анализ предметной области 9](#_Toc198254297)

[1.1 Основные понятия и определения 9](#_Toc198254298)

[1.2 Актуальность задачи, 12](#_Toc198254299)

[1.3 Описание систем-аналогов. 14](#_Toc198254300)

[1.3.1 Anki 14](#_Toc198254301)

[1.3.2 StudyStack 15](#_Toc198254302)

[1.3.3 Конкурентный анализ систем-аналогов 17](#_Toc198254303)

[1.4 Описание автоматизируемого процесса 18](#_Toc198254304)

[1.5 Постановка задачи 18](#_Toc198254305)

[2 Проектирование системы 19](#_Toc198254306)

[2.1 Выбор и обоснование архитектуры системы 19](#_Toc198254307)

[2.2 Проект системы 19](#_Toc198254308)

[2.2.1 Построение проекта системы 19](#_Toc198254309)

[2.2.2 Выбор и обоснование средств реализации 19](#_Toc198254310)

[3 Реализация системы 20](#_Toc198254311)

[3.1 Описание интерфейса пользователя 20](#_Toc198254312)

[3.2 Диаграммы реализации 20](#_Toc198254313)

[3.3 Физическая модель данных. 20](#_Toc198254314)

[3.4 Апробация системы 20](#_Toc198254315)

[3.5 Описание проведенных исследований 20](#_Toc198254316)

[Заключение 21](#_Toc198254317)

[ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ 22](#_Toc198254318)

[Список использованных источников 23](#_Toc198254319)

[ПРИЛОЖЕНИЕ А Руководство пользователя 26](#_Toc198254320)

[А.1 Назначение системы 26](#_Toc198254321)

[А.2 Условия работы системы 26](#_Toc198254322)

[А.3 Установка системы 26](#_Toc198254323)

[А.4 Работа с системой 26](#_Toc198254324)

[ПРИЛОЖЕНИЕ Б Код программы 27](#_Toc198254325)

[ПРИЛОЖЕНИЕ В Акт внедрения системы 28](#_Toc198254326)

[ПРИЛОЖЕНИЕ Г Диплом за доклад, представленный на научную конференцию «Перспективные информационные технологии (ПИТ-2019)» 29](#_Toc198254327)

[ПРИЛОЖЕНИЕ Д Свидетельство о регистрации программы 30](#_Toc198254328)

Введение

В условиях стремительного развития цифровых технологий и увеличения объёма информации, которую необходимо усваивать, особую актуальность приобретают методы эффективного запоминания и обучения. Одним из таких методов является интервальное повторение, основанное на принципе увеличения интервалов между повторениями изучаемого материала, что способствует его долговременному запоминанию. [1] Этот подход был впервые описан немецким психологом Германом Эббингаузом в конце XIX века, который выявил закономерность забывания информации со временем и предложил способы её закрепления через повторение . [2]

Среди практических реализаций метода интервального повторения выделяется система Лейтнера, предложенная немецким научным журналистом Себастьяном Лейтнером в 1970-х годах. [3] Суть метода заключается в использовании карточек с информацией, которые распределяются по группам в зависимости от уровня усвоения, и повторяются с различной частотой: чем хуже запомнена информация, тем чаще она повторяется .

Современные исследования подтверждают высокую эффективность интервального повторения в обучении. Так, в работе Tabibian и др. представлена вычислительная модель оптимизации алгоритмов интервального повторения, адаптирующихся к индивидуальным особенностям обучающегося, что позволяет значительно повысить эффективность запоминания. [4]

Несмотря на наличие программных решений, реализующих методы интервального повторения (например, Anki, SuperMemo), существует потребность в разработке адаптивных и ориентированных на пользователя систем, учитывающих современные требования к интерфейсу.

Цель данной работы — разработка веб-приложения, реализующего метод интервального повторения на основе системы Лейтнера. Приложение должно обеспечивать эффективное управление процессом обучения, предоставлять инструменты для создания и редактирования карточек.

В качестве методической основы выбран подход, сочетающий принципы активного воспроизведения и интервального повторения, что соответствует современным тенденциям в области педагогики. Разработка будет осуществляться с использованием современных веб-технологий: Java (Spring Boot) для серверной части, GraphQL для организации взаимодействия между клиентом и сервером, React для клиентской части, PostgreSQL в качестве системы управления базами данных, а также Liquibase для управления версиями базы данных.

Таким образом, предлагаемая система будет представлять собой современное, адаптивное и эффективное средство для самостоятельного обучения, основанное на проверенных научных методах и современных технологических решениях.

1. Описание и анализ предметной области
   1. Основные понятия и определения

Флэш-карточка — это эффективное средство обучения, представляющее собой двустороннюю карточку, на одной стороне которой записан вопрос, термин или фрагмент текста, а на другой — ответ, определение или пояснение. Такой формат заточен под активное воспроизведение задуманных сведений и самопроверку. [5]

Cloze, или тест с пропусками, — это упражнение, в котором из исходного текста удаляются отдельные слова или фразы, а обучающийся восстанавливает их по контексту. Данный метод развивает навыки понимания текста и активного воспоминания, требуя от пользователя анализа синтаксиса и семантики фрагмента. [6]

На рисунке XX показан пример флэш-карточки с cloze-пропусками.

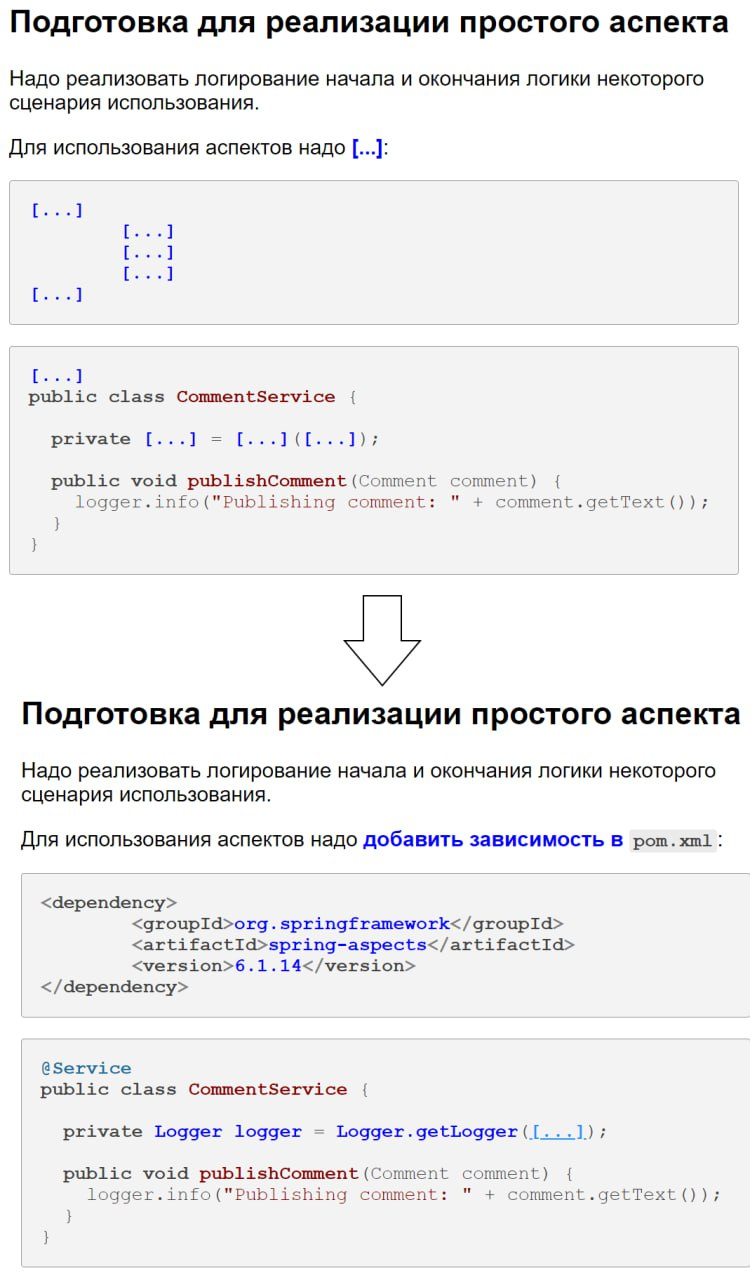


Рисунок XX – Пример флэш-карточки с cloze-пропусками

Классическая кривая забывания Эббингауза — график, иллюстрирующий закономерности утраты информации с течением времени после первоначального запоминания. Сразу после изучения материал удерживается практически полностью, но затем забывание происходит наиболее интенсивно, а со временем скорость утраты знаний замедляется и стремится к асимптоте.

На рисунке XX приведено изображение этой кривой.

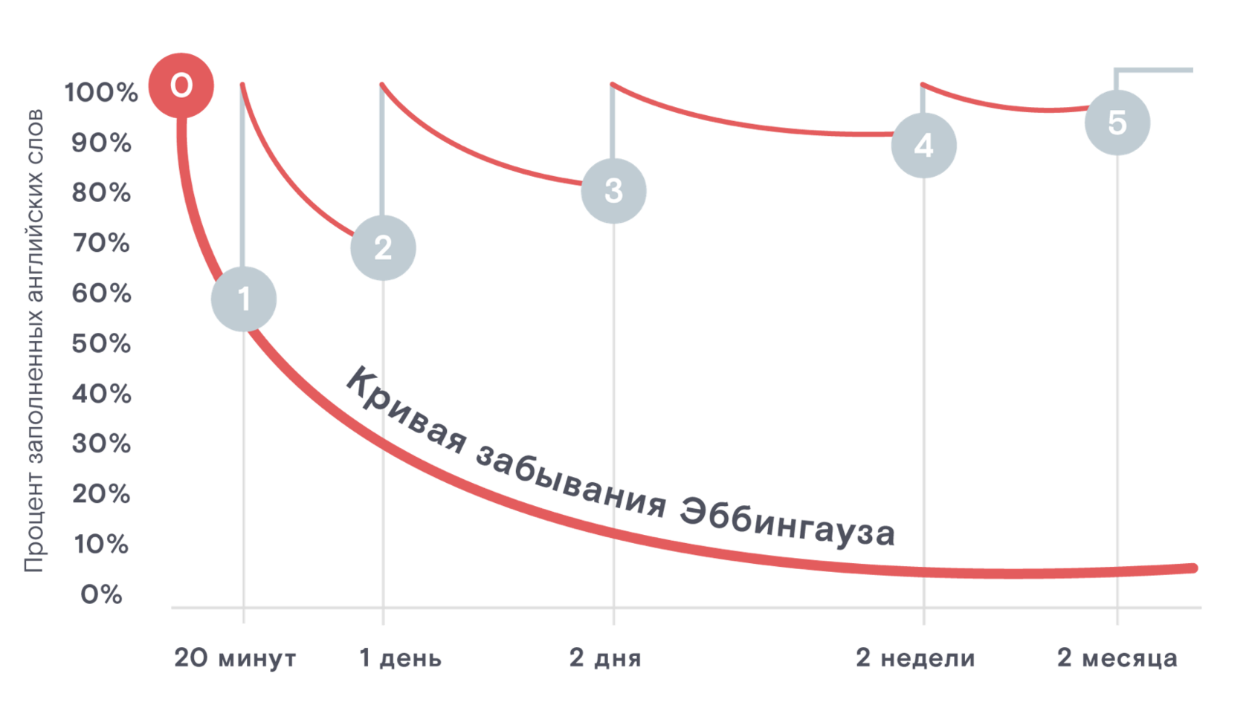


Рисунок XX – Кривая Эббингауза

Множество методов интервального повторения опираются на кривую Эббингауза, назначая повторения в такие промежутки, когда забывание ещё не стало критическим, а частичное ослабление памяти при последующем повторении обеспечивает более прочное закрепление материала.

Система Лейтнера — один из первых практических алгоритмов интервального повторения, предложенный Себастьяном Лейтнером в 1970-х годах. Суть метода состоит в разделении набора карточек на несколько «ящиков», каждый из которых соответствует своему интервалу повторения (например, 1 день, 2 дня, 4 дня и т. д.). При успешном воспроизведении карточка перемещается в следующий ящик с увеличенным интервалом; если ответ был неверным, карточка возвращается в предыдущий ящик для более частых повторений.

На рисунке XX показана схема работы системы Лейтнера.

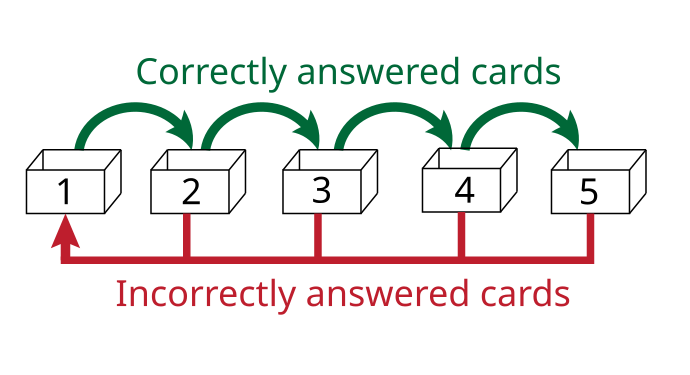


Рисунок XX – Схема работы системы Лейтнера

Алгоритм SM-2 лежит в основе оригинальной программы SuperMemo и до сих пор остаётся открытым стандартом. Он включает следующие шаги:

1. разделить материал на отдельные элементы и каждому присвоить исходное значение коэффициента легкости EF = 2.5;
2. первые два интервала повторения задаются жёстко:
   * I(1) = 1 день
   * I(2) = 6 дней
3. для n-го повторения при n > 2 рассчитывать интервал  
   ;
4. сразу после ответа оценивать качество реакции q по шкале 0–5 (5 — идеальный ответ, 0 — полное забывание);
5. пересчитывать EF по формуле , если EF' < 1.3, присвоить EF' = 1.3;
6. если q < 3, считать, что элемент не выучен, и начать повторение заново, без изменения EF;
7. в тот же день повторить все элементы с q < 4 до достижения оценки ≥ 4.

Таким образом SM-2 адаптируется к индивидуальным результатам пользователя, автоматически увеличивая интервалы для хорошо запоминаемых карточек и чаще возвращая «трудные».

* 1. Актуальность задачи,

Актуальность задачи, области применения результатов решения, существующие методики решения, кем и когда получены, особенности выбранной задачи или решения по сравнению с существующими. Выводы по актуальности задачи и необходимости ее решения.

В условиях постоянного увеличения объёмов информации, необходимой для профессиональной и образовательной деятельности, актуальность методов эффективного обучения и запоминания существенно возрастает. Одним из наиболее эффективных подходов является интервальное повторение, реализуемое в системе Лейтнера, которая основана на увеличении интервалов между повторениями изучаемого материала. Это способствует его надёжному закреплению в долговременной памяти.

Разрабатываемая система предназначена не только для студентов и учащихся, но также может эффективно использоваться при подготовке к профессиональным и квалификационным экзаменам. Среди таких сфер:

1. курсы повышения квалификации и профессиональной переподготовки сотрудников;
2. подготовка сотрудников предприятий и организаций к сдаче квалификационных экзаменов, например, при повышении профессиональной категории инженеров и технических специалистов;
3. подготовка сотрудников органов внутренних дел к прохождению профессиональных аттестаций и экзаменов, требующих запоминания значительного объёма нормативной и правовой информации;
4. обучение и подготовка кандидатов на получение водительских удостоверений, где требуется запоминать правила дорожного движения и другую необходимую информацию.

Помимо непосредственного запоминания информации, система также решает важную задачу управления временем пользователей (тайм-менеджмент), позволяя оптимально распределить процесс обучения во времени и минимизировать затраты на повторение материала.

Существуют уже внедрённые методики интервального повторения, такие как SuperMemo, Anki и Duolingo, которые успешно используются в образовательных целях и для изучения языков. SuperMemo, разработанная Петром Возняком в 1982 году, предлагает алгоритмы интервального повторения, адаптирующиеся под индивидуальные особенности обучающихся. Приложение Anki, созданное на основе алгоритма SM-2, предоставляет открытый и доступный всем пользователям инструмент с возможностью гибкой настройки интервалов повторения. Duolingo внедряет модель интервального повторения, сочетающую машинное обучение с психолингвистическими подходами, ориентированную преимущественно на изучение языков.

Предлагаемое решение отличается от существующих аналогов тем, что ориентировано на универсальность применения и нацелено на поддержку различных типов обучающих материалов. Карточки в приложении создаются с помощью удобного и простого синтаксиса Markdown, в основном используя формат cloze-тестов для активного воспоминания скрытых фрагментов текста. В отличие от других систем с более сложным интерфейсом и неудобным процессом создания карточек, данное веб-приложение обеспечивает адаптивность, интуитивную простоту и лёгкость интеграции в различные образовательные и профессиональные контексты.

Таким образом, разработка предлагаемого веб-приложения является актуальной и востребованной задачей, так как решает реальные проблемы эффективного запоминания информации, способствует оптимизации учебного процесса и может успешно применяться в различных профессиональных и образовательных средах.

* 1. Описание систем-аналогов.

Для анализа были выбраны наиболее популярные системы-аналоги, реализующие подход интервального повторения или схожие функциональные возможности. В этом разделе приведены краткие характеристики каждой системы, их достоинства и недостатки, а также сравнительный анализ.

* + 1. Anki

Anki – популярная система для интервального повторения, широко используемая для изучения языков, терминов, медицинской информации и других образовательных материалов, требующих эффективного запоминания. Приложение построено на основе алгоритма SM-2, предложенного Петром Возняком, и предназначено для длительного сохранения знаний путем планомерного увеличения интервалов между повторениями. В Anki предусмотрены различные режимы повторения карточек, статистика прогресса, а также расширенные возможности настройки параметров интервального повторения и внешнего вида карточек. Несмотря на широкие функциональные возможности, пользователи часто отмечают необходимость наличия технических навыков для создания и редактирования карточек, что может стать препятствием для начинающих. [X]

На рисунке XX приведена экранная форма создания карточки в системе Anki.

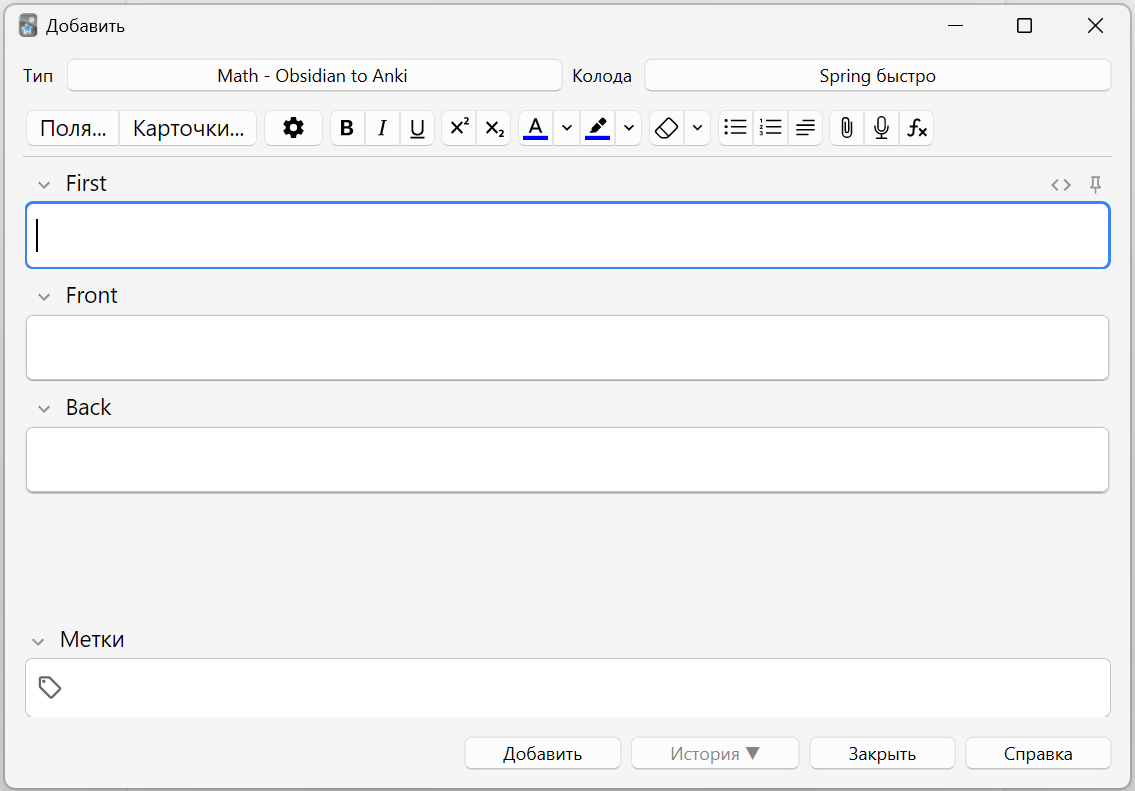


Рисунок XX – Экранная форма создания карточки в системе Anki

На рисунке XX приведен режим повторения карточек в Anki.

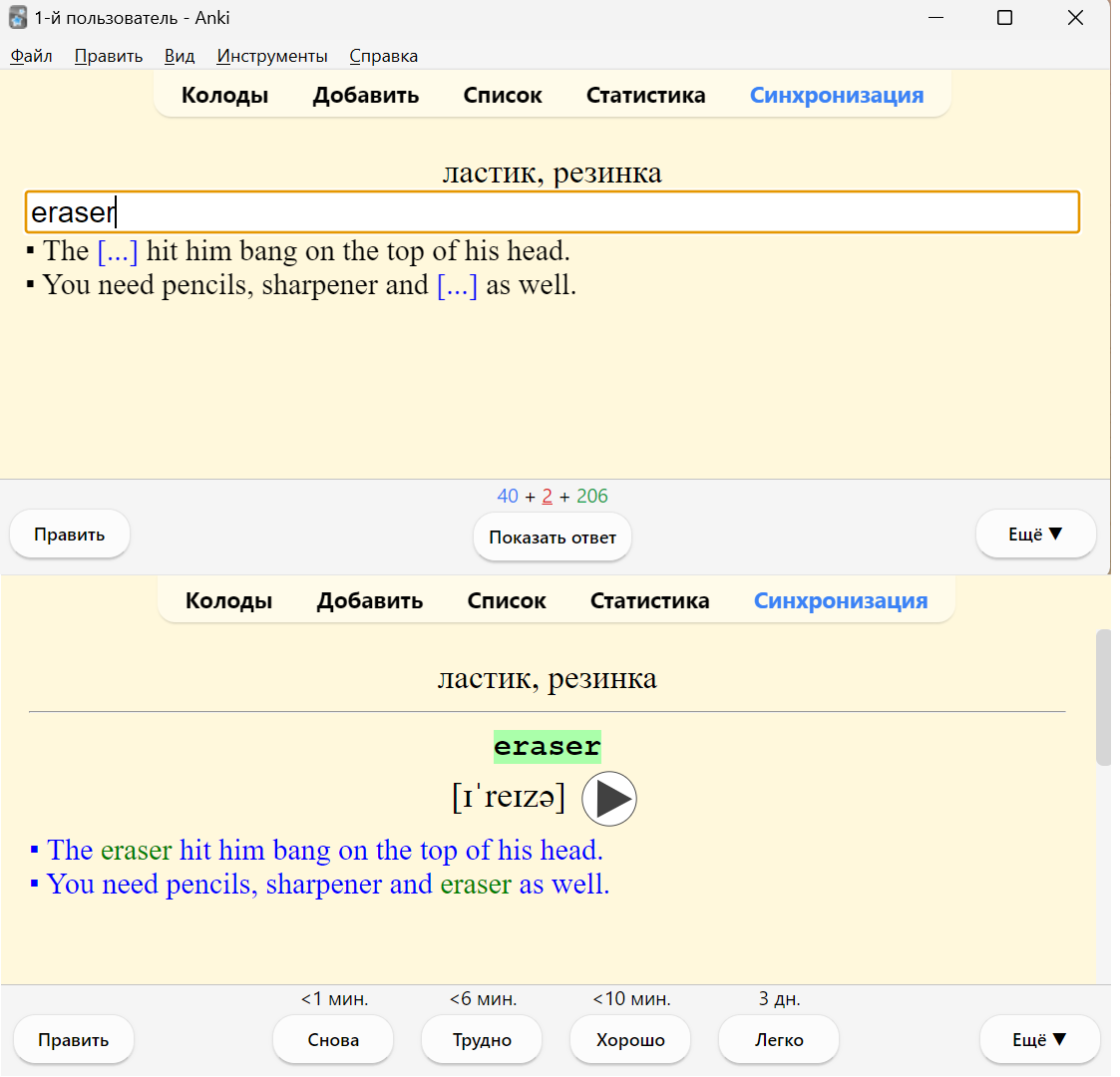


Рисунок XX – Режим повторения карточек в Anki

К достоинствам системы Anki относятся:

* гибкие и адаптивные алгоритмы интервального повторения;
* открытый исходный код;
* поддержка множества платформ (Windows, Linux, MacOS, Android, iOS).

К недостаткам системы относятся:

* сложный процесс создания карточек с использованием шаблонов на HTML, CSS и JavaScript;
* неудобный интерфейс для новых пользователей.
  + 1. StudyStack

StudyStack – онлайн-приложение, позволяющее создавать и использовать обучающие карточки, а также различные образовательные игры и активности на их основе. StudyStack ориентирован на активное повторение информации с помощью различных игровых механик, однако не поддерживает алгоритмы интервального повторения.

На рисунке XX приведена экранная форма создания карточки в StudyStack.

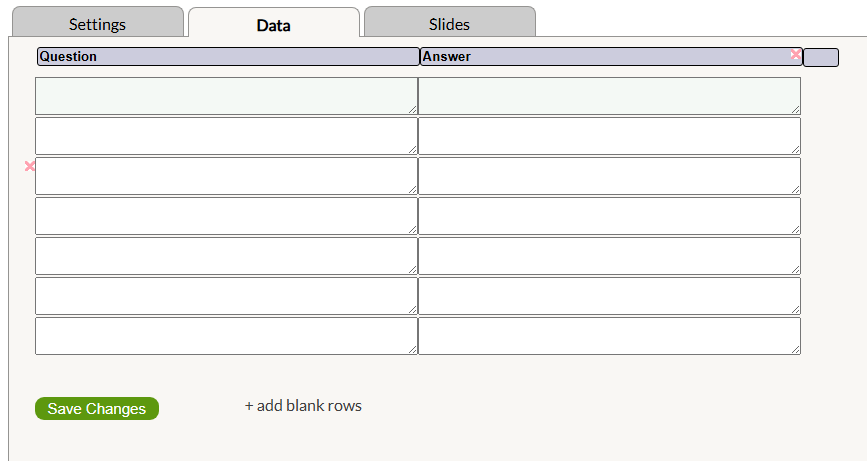


Рисунок XX – Экран создания карточки в StudyStack

На рисунке XX приведен режим повторения карточек в StudyStack.

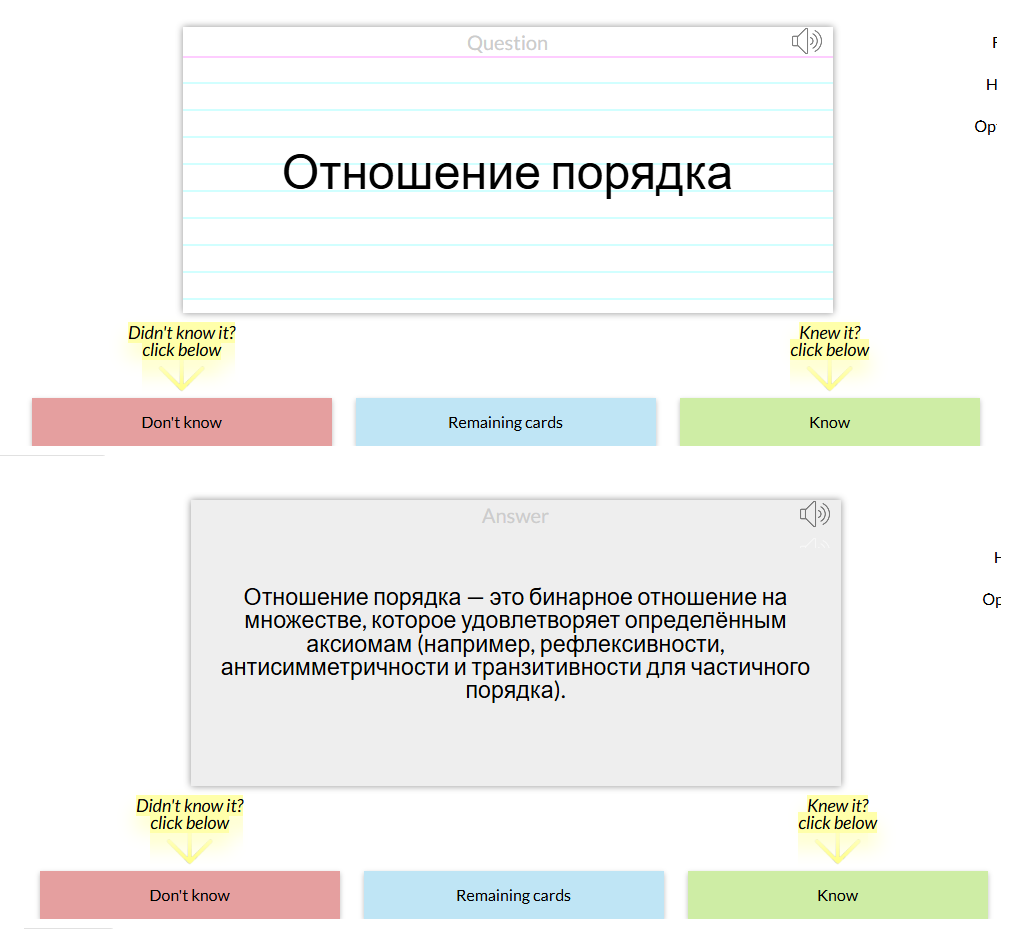


Рисунок XX – Режим повторения карточек в StudyStack

К достоинствам системы StudyStack относятся:

* простота и доступность создания карточек;
* разнообразие игровых форм для повторения материала.

К недостаткам системы относятся:

* отсутствие алгоритмов интервального повторения;
* маленький размер карточек и неудобное их отображение при повторении;
* ограниченные возможности персонализации и настройки карточек.
  + 1. Конкурентный анализ систем-аналогов

Для наглядного сравнения выбранных систем-аналогов представлена таблица 1.

Таблица 1 – Сравнительная характеристика систем-аналогов

| Показатель | Anki | StudyStack | Разрабатываемое приложение |
| --- | --- | --- | --- |
| Интервальное повторение | + | – | + |
| Удобство создания карточек | – | ± | + |
| Поддержка Markdown-синтаксиса | – | – | + |
| Использование cloze-тестов | ± | – | + |
| Простота интерфейса | – | ± | + |
| Наличие игрового режима | – | + | – |
| Возможность тонкой настройки карточек | + | – | - |

* 1. Описание автоматизируемого процесса

Разработка веб-приложения для обучения по системе Лейтнера с реализацией алгоритма интервального повторения включает в себя несколько ключевых этапов. Каждый из них направлен на создание эффективной и удобной системы для пользователей, стремящихся к долговременному запоминанию информации.

* + 1. Анализ предметной области и постановка задачи

На начальном этапе проводится исследование существующих методов интервального повторения, включая систему Лейтнера и алгоритм SM-2. Изучаются потребности целевой аудитории, такие как студенты, профессионалы, готовящиеся к квалификационным экзаменам, и сотрудники предприятий. Целью является выявление требований к функциональности системы, обеспечивающей эффективное запоминание и удобство использования.

* + 1. Проектирование архитектуры системы

Система разрабатывается как клиент-серверное веб-приложение с использованием следующих технологий:

* Backend: Java с использованием фреймворка Spring Boot для обработки бизнес-логики и взаимодействия с базой данных;
* Frontend: React для создания интерактивного пользовательского интерфейса;
* API: GraphQL для эффективного обмена данными между клиентом и сервером;
* база данных: PostgreSQL для хранения информации о пользователях, карточках и расписании повторений;
* управление миграциями: Liquibase для контроля версий схемы базы данных.

Архитектура обеспечивает масштабируемость, модульность и возможность дальнейшего расширения функциональности.

* + 1. Реализация алгоритма интервального повторения

В основе системы лежит алгоритм SM-2, адаптированный для автоматического расчета интервалов повторения карточек. Пользователь оценивает свою уверенность в ответе на карточку, и на основе этой оценки система определяет оптимальное время следующего повторения. Это позволяет индивидуализировать процесс обучения и повысить эффективность запоминания.

* + 1. Разработка интерфейса создания и управления карточками

Особое внимание уделяется удобству создания карточек. Используется синтаксис Markdown, что позволяет пользователям легко форматировать текст и создавать cloze-тесты для активного воспоминания. Интерфейс предусматривает предпросмотр карточек и простое редактирование, что снижает порог входа для новых пользователей.

* + 1. Тестирование и отладка системы

Проводится комплексное тестирование всех компонентов системы:

* модульное тестирование: проверка отдельных функций и методов;
* интеграционное тестирование: оценка взаимодействия между различными модулями системы;
* тестирование пользовательского интерфейса: обеспечение удобства и интуитивной понятности интерфейса.

На основе результатов тестирования вносятся необходимые корректировки для обеспечения стабильной и надежной работы приложения.

* + 1. Внедрение

После завершения разработки и тестирования система разворачивается на сервере и становится доступной для пользователей.

Таким образом, автоматизируемый процесс охватывает полный цикл разработки веб-приложения, начиная от анализа требований и заканчивая внедрением системы. Использование современных технологий и адаптация проверенных алгоритмов интервального повторения обеспечивают высокую эффективность и удобство обучения для пользователей.

* 1. Постановка задачи

Цель работы: во время выпускной квалификационной работы необходимо разработать веб-приложение для автоматизации процесса обучения по системе Лейтнера, включающее создание, редактирование и повторение карточек с реализацией алгоритма интервального повторения. Приложение должно обеспечивать автоматический расчёт оптимальных интервалов повторений, отслеживать прогресс по колодам и поддерживать эффективное усвоение информации посредством удобного и простого интерфейса.

Задачи:

* изучить основные понятия предметной области: интервальное повторение, система Лейтнера, флеш-карточки, модель SM-2, процессы обучения и запоминания информации;
* выполнить обзор систем-аналогов в области существующих решений для интервального повторения и обучения с карточками;
* разработать проект автоматизированной системы с использованием методологии структурного и объектно-ориентированного проектирования;
* разработать информационное и программное обеспечение системы, произвести его тестирование и отладку.

Разрабатываемая автоматизированная система должна выполнять следующие функции.

Функции клиентской части:

* удобное создание и редактирование обучающих карточек с использованием markdown-синтаксиса и cloze-тестов;
* создание и удаление колод для обучающих карточек;
* отображение списка карточек и возможность их повторения;
* просмотр текущего прогресса по колодам;

Функции серверной части:

* расчёт оптимальных интервалов повторений на основе алгоритма интервального повторения SM-2;
* хранение информации о пользователях, карточках и расписании повторений;
* обеспечение взаимодействия между клиентской и серверной частями с использованием технологии GraphQL;
* обработка и сохранение данных, переданных с клиентской части;
* управление миграциями базы данных с использованием Liquibase для обеспечения целостности и актуальности схемы данных.

1. Проектирование системы
   1. Выбор и обоснование архитектуры системы

Описание архитектуры разрабатываемого приложения, ее обоснование.

* 1. Проект системы
     1. Построение проекта системы

Описание проекта системы (в соответствии с выбранной концепцией проектирования).

Если описывается структурная схема системы, нужно описать все подсистемы.

Если используется UML, то включаются как минимум следующие диаграммы: вариантов использования, классов (логического уровня) системы, деятельности для всей системы, последовательностей для важных вариантов использования.

Описывается проект как программного, так и информационного обеспечения (структура БД, логическая модель данных).

Каждый этап проектирования лучше оформлять в виде отдельного подпункта.

* + 1. Выбор и обоснование средств реализации

Описание применяемых средств реализации (системы, среды, платформы, языки программирования, библиотеки, СУБД, и т.д.). Можно каждое отдельным подпунктом.

1. Реализация системы
   1. Описание интерфейса пользователя

Разработанное программное приложение имеет клиент-серверную архитектуру. Экранные формы серверной части представлены на рисунках N-M.

Описать экранные формы разработанного приложения, возможные исключительные ситуации и реакцию на них приложения. Можно каждую форму в виде отдельного подпункта. (Форма авторизации, главная форма, форма работы со справочниками, и т.д.)

* 1. Диаграммы реализации

Представляются здесь, если они есть.

* 1. Физическая модель данных.

Если есть.

* 1. Апробация системы

При необходимости. Примеры использования, демонстрация работоспособности системы.

* 1. Описание проведенных исследований

Если есть исследования. Планы экспериментов, входные данные, параметры, полученные результаты, их интерпретация, обобщения и выводы по экспериментам.

Заключение

В процессе выполнения выпускной работы была разработана автоматизированная система …, позволяющая ….

В первом разделе были приведены основные понятия и определения предметной области подробно, приведены характеристики систем-аналогов, на основании этого была сформулирована постановка задачи и основные требования к системе.

Во втором разделе была разработана структура системы, разработан проект системы на языке, логическая модель данных, а также был выбран комплекс программных средств.

В третьем разделе описан интерфейс пользователя, физическая модель данных системы, приведены результаты проведенных исследований или примеры использования системы …

Разработанная автором система используется там-то и там-то. Имеется акт о внедрении разработанной системы в эксплуатацию (Приложение В).

Разработанная автором система используется там-то и там-то. Имеется акт о внедрении разработанной системы в эксплуатацию (Приложение В).

Результаты работы, приведенные в пояснительной записке, были представлены на XXVI Международной конференции «Очень крутая конференция» ОКК – 2024 (г. Бобруйск, май 2024 г.), докладывались на LXXIII Молодежной научной конференции, посвященная 35-летию со дня первого полёта МТКС «Энергия-Буран» (г. Самара, апрель 2024 г.) (Приложение Г), были представлены на международную научно-техническую конференцию «Еще одна очень крутая конференция» ЕООКК-2024 (г. Самара, июнь 2023 г.).

Имеется свидетельство о регистрации программы для ЭВМ в Федеральной службе по интеллектуальной собственности (Приложение Д).

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

|  |  |  |
| --- | --- | --- |
|  |  | При необходимости! Сначала на английском, потом на русском |
| ADO | − | ActiveX data objects (объекты данных ActiveX); |
| AJAX | − | asynchronous JavaScript and XML (асинхронный JavaScript и XML); |
| ANSI | − | american national standards institute (американский национальный институт стандартов); |
| АИС | − | автоматизированная информационная система; |
| БД | − | база данных; |
| ИКТ | − | информационно-коммуникационные технологии; |
| ИОС | − | интерактивная обучающая система; |
| МКС | − | мультисервисная корпоративная сеть; |
| СУБД | − | система управления базами данных; |
| СУФ | − | система управления файлами; |
| СЭДО | − | система электронного дистанционного обучения; |
| ЭВМ | − | электронная вычислительная машина; |
| ЭЖ | − | электронный журнал. |

Список использованных источников

1. Leitner System: The Most Effective Way to Revise [Электронный ресурс]. URL: https://www.lecturio.com/blog/revamp-your-revision-with-the-leitner-system (дата обращения: 14.05.2025).

2. Метод интервального повторения иностранных слов [Электронный ресурс]. URL: https://sportzania.ru/about/publikatsii/metod-intervalnogo-povtoreniya-inostrannykh-slov (дата обращения: 14.05.2025).

3. Интервальные повторения [Электронный ресурс] // Википедия: свободная энциклопедия. URL: https://ru.wikipedia.org/wiki/Интервальные\_повторения (дата обращения: 14.05.2025).

4. Spaced repetition [Электронный ресурс] // Wikipedia: the free encyclopedia. URL: https://en.wikipedia.org/wiki/Spaced\_repetition (дата обращения: 14.05.2025).

1 ЭТО ПРИМЕРЫ!!! Покровская О.М. Совершенствование комплекса гигиенических мероприятий у пациентов с ортопедическими конструкциями на имплантатах. : дис. ... канд. мед. наук: 14.00.21: защищена 02.09.2008/ Покровская Ольга Михайловна. М., 2008. 115 с.: ил.

2 Богданов И.М., Зеленко Л.С., Филатова Н.В. Автоматизированная система расчета показателей основных стоматологических заболеваний у детского населения // Математика. Компьютер. Образование: труды XXIV межд. конф., 28 января – 2 февраля 2019 г., г. Пущино. URL: http://www.mce.su/rus/presentations/p333102/ (дата обращения: 02.02.2019).

3 Справочник по формулам зубов [Электронный ресурс]. URL: http://www.dental-revue.ru/index.php?page=15&artId=7 (дата обращения: 30.09.2018).

4 Современные системы оценки и регистрации кариеса зубов / Пастбин М.Ю., Горбатова М.А. и [др.]. Архангельск: Северный государственный медицинский университет, 2013. 7 с.

5 Методическое пособие по теме «Индексная оценка кариеса зубов и заболеваний пародонта» / Молоков В.Д., Доржиева З.В. и [др.]. Иркутск: Иркутский государственный медицинский университет, 2008. 23 с.

6 Официальный сайт Dental Software [Электронный ресурс]. URL: https://www.dentrix.com/ (дата обращения: 16.10.2018).

7 Официальный сайт Open Dental Software [Электронный ресурс]. URL: https://www.opendental.com (дата обращения: 18.10.2018).

8 Официальный сайт CS SoftDent [Электронный ресурс]. URL: htt/ps://www.carestreamdental.com/en-us/products/practice-management-software/cs-softdent/ (дата обращения: 20.10.2018).

9 Большой Российский энциклопедический словарь. М.: БРЭ, 2003. 341 с.

10 Буч, Г. Язык UML. Руководство пользователя [Текст] /Г. Буч, Д. Рамбо, А. Якобсон. 2-е изд.: Пер. с англ. Мухина Н. М.: ДМК Пресс, 2006. 496 с.: ил.

11 Диаграмма вариантов использования [Электронный ресурс] URL: https://www.intuit.ru/studies/courses/32/32/lecture/1004 (дата обращения: 15.11.2018).

12 Спецификация требований [Электронный ресурс] URL: https://www.intuit.ru/studies/courses/32/32/lecture/1006 (дата обращения: 9.10.2018).

13 Диаграмма классов [Электронный ресурс] // Студопедия: [сайт]. URL: https://studopedia.info/10-59449.html (дата обращения: 30.11.2018).

14 Основные понятия баз данных [Электронный ресурс]. URL: http://inf.susu.ac.ru/Klinachev/lc\_sga\_26.htm (дата обращения: 23.10.2018).

15 RFC 2898-2000. PKCS #5: Password-Based Cryptography Specification. Version 2.0 = Стандарт формирования ключа на основе пароля [Электронный ресурс]. URL: https://tools.ietf.org/html/rfc2898 (дата обращения: 15.11.2018).

16 ГОСТ 28806-90 Качество программных средств. Термины и определения. М., 1990. 12 с. (Издательство стандартов).

17 Официальный сайт SQLite [Электронный ресурс]. URL: https://www.sqlite.org/index.html (дата обращения: 01.07.2018).

18 ГОСТ 28397-89. Языки программирования. Термины и определения М., 1989. 8 с. (Издательство стандартов).

19 ISO/IEC 9899-2011. Programming languages C = Язык программирования C. Международный стандарт. США, Нью-Йорк: Американский национальный институт стандартов ANSI, 2012. 702 с.

20 ISO/IEC 14882-2014. Programming languages C++ = Язык программирования C++. Международный стандарт. Швейцария, Женева: Международная электротехническая комиссия IEC, 2014. 1375 с.

21 Основные принципы C++ [Электронный ресурс]. URL: https://en.cppreference.com/w/cpp/language/basic\_concepts (дата обращения: 02.07.2018).

22 ГОСТ 15971-90 Системы обработки информации. Термины и определения. М.: Издательство стандартов. 1991. 14 с.

23 Операционная система Windows 7 [Электронный ресурс] URL: https://ru.wikipedia.org/wiki/Windows\_7 (дата обращения: 10.11.2018).

24 Знакомство с ОС Windows 7 [Электронный ресурс] URL: https://www.lessons-tva.info/edu/inf-win/win-1-2-1-1.html (дата обращения: 10.11.2018).

27 Пользовательский интерфейс [Электронный ресурс] URL: https://spravochnick.ru/informatika/arhitektura\_personalnogo\_kompyutera/polzovatelskiy\_interfeys/ (дата обращения: 15.11.2018).

28 Диаграммы реализации [Электронный ресурс] URL: http://www.maksakov-sa.ru/ModelUML/DiagrReal/index.html (дата обращения: 15.11.2018).

29 Диаграмма развёртывания [Электронный ресурс] URL: https://www.intuit.ru/studies/courses/32/32/lecture/1024 (дата обращения: 15.11.2018).

30 Физическая модель базы данных [Электронный ресурс] URL: https://studopedia.ru/6\_1898\_fizicheskaya-model-bazi-dannih.html (дата обращения: 25.10.2018).

31 Богданов И.М. Разработка автоматизированной системы расчета показателей основных стоматологических заболеваний у детского населения/ И.М. Богданов, Л.С. Зеленко, Н.В. Филатова // Перспективные информационные технологии (ПИТ-2019): сб. науч. тр. межд. научно-техн. конф.; [под ред. С.А. Прохорова]. Самара: Изд-во СНЦ РАН, 2019 (принято в печать).

ПРИЛОЖЕНИЕ А  
Руководство пользователя

А.1 Назначение системы

Данная система предназначена для ….

А.2 Условия работы системы

Для корректной работы системы необходимо наличие соответствующих программных и аппаратных средств.

1. Требования к техническому обеспечению:

* ЭВМ типа IBM PC;
* процессор типа x86 или x64 тактовой частоты 1400 МГц и выше;
* клавиатура или иное устройство ввода;
* мышь или иное манипулирующее ввода;

1. Требования к программному обеспечению:

* операционная система Windows 7 и выше.

А.3 Установка системы

Система поставляется в виде zip-архива. Данный файл необходимо распаковать в любую директорию на жестком диске. Запускаемым файлом системы является файл ….

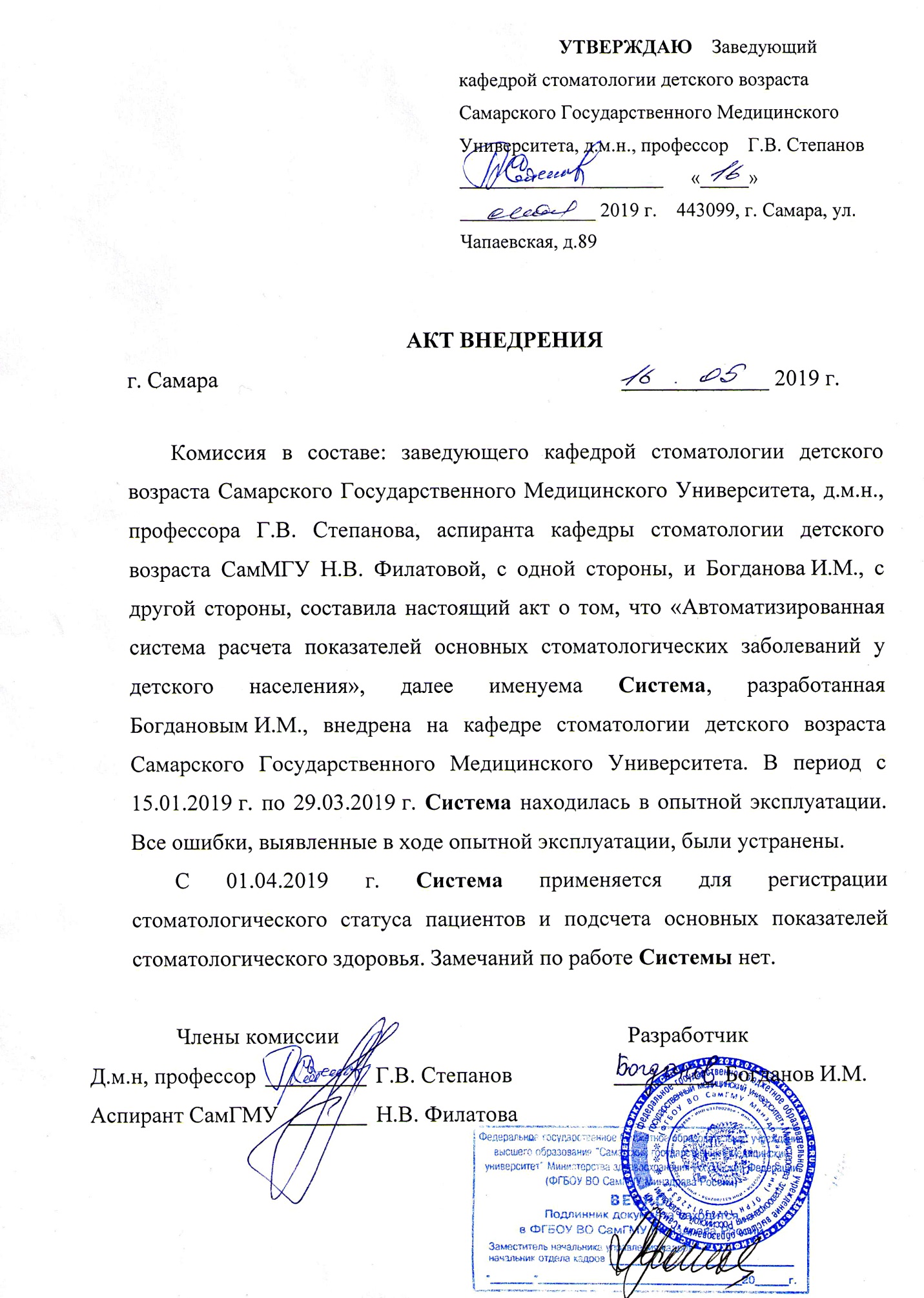
А.4 Работа с системой

…

Рисунок А.1 – Пример работы программы

ПРИЛОЖЕНИЕ Б  
Код программы

ПРИЛОЖЕНИЕ В  
Акт внедрения системы



ПРИЛОЖЕНИЕ Г  
Диплом за доклад, представленный на научную конференцию «Перспективные информационные технологии (ПИТ-2019)»



ПРИЛОЖЕНИЕ Д  
Свидетельство о регистрации программы

