МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение  
высшего образования

**«Южно-Уральский государственный университет**

**(национальный исследовательский университет)»**

**Высшая школа электроники и компьютерных наук**

**Кафедра системного программирования**

|  |
| --- |
| ДОПУСТИТЬ К ЗАЩИТЕ  Заведующий кафедрой, д.ф.-м.н., профессор  \_\_\_\_\_\_\_\_\_\_ Л.Б. Соколинский  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2023 г. |

**Разработка веб-приложения для отслеживания**

**курса крипто-валют**

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

ЮУрГУ – 02.03.02.2023.308-268.ВКР

|  |  |
| --- | --- |
|  | Научный руководитель,  доцент кафедры СП, к.ф.-м.н., доцент \_\_\_\_\_\_\_\_\_\_ Радченко Г.И.  Автор работы, студент группы КЭ-401  \_\_\_\_\_\_\_\_\_\_ В.А. Немцев  Ученый секретарь  (нормоконтролер)  \_\_\_\_\_\_\_\_\_\_\_\_\_ И.Д. Володченко  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2023 г. |

Челябинск, 2023 г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение  
высшего образования

**«Южно-Уральский государственный университет**

**(национальный исследовательский университет)»**

**Высшая школа электроники и компьютерных наук**

**Кафедра системного программирования**

УТВЕРЖДАЮ

Зав. кафедрой СП

\_\_\_\_\_\_\_\_\_\_ Л.Б. Соколинский

07.02.2023 г.

**ЗАДАНИЕ**

**на выполнение выпускной квалификационной работы бакалавра**

студенту группы КЭ-401

Немцеву Вячеславу Александровичу,

обучающемуся по направлению

02.03.02 «Фундаментальная информатика и информационные технологии»

1. **Тема работы** (утверждена приказом ректора от \_\_.\_\_.2022 г. № \_\_\_)

Разработка веб-приложения для отслеживания курса крипто-валют.

1. **Срок сдачи студентом законченной работы:** 06.06.2023 г.
2. **Исходные данные к работе2**
3. API веб-приложения. [Электронный ресурс] URL: https://rapidapi.com/Coinranking/api/coinranking1(дата обращения 30.01.2023г.).
4. Веб-приложение forklog. [Электронный ресурс] URL: https://forklog.com/rates(дата обращения 30.01.2023).
5. Документация библиотеки состояний Redux-toolkit. [Электронный ресурс] URL: https://redux-toolkit.js.org/(дата обращения 30.01.2023).
6. Льюис Э., Основы биткойнов и блокчейнов(2018 г.).
7. Документация библиотеки языка JavaScript React [Электронный ресурс] URL: https://reactjs.org/
8. Документация библиотеки redux-saga [Электронный ресурс] URL: redux-saga.js.org/docs/api/
9. Документация CSS-in-JS библиотеки styled-components [Электронный ресурс] URL: styled-components.com
10. **Перечень подлежащих разработке вопросов**
11. Выполнить анализ предметной области.
12. Спроектировать веб-приложение.
13. Реализовать веб-приложение.
14. Произвести тестирование разработанной системы.
15. **Дата выдачи задания:** 07.02.2023 г.

**Научный руководитель,**

доцент кафедры СП, к.ф.-м.н., доцент Г.И. Радченко

**Задание принял к исполнению** В.А. Немцев

# ГЛОССАРИЙ

1. *Майнинг* – добыча блоков в сети блокчейн для обеспечения крипто-валютных платформ.
2. *Реактивность* – способ автоматически обновлять систему в зависимости от изменения потока данных.
3. *Рендеринг* – процесс получения изображения по модели с помощью компиляции программы.
4. *DOM-дерево* – это интерфейс, обрабатывающий HTML-документ как древовидную структуру в которой каждый узел представляет собой объект, содержащий часть документа.
5. *Монтирование* – вставка элементов в DOM-дерево.

ОГЛАВЛЕНИЕ

[ГЛОССАРИЙ 3](#_Toc70412912)

[ВВЕДЕНИЕ 5](#_Toc70412913)

[1. СТРУКТУРНО-ИЕРАРХИЧЕСКАЯ ДИДАКТИЧЕСКАЯ МОДЕЛЬ 6](#_Toc70412914)

[1.1. Описание предметной области 6](#_Toc70412915)

[1.2. Сравнительный анализ аналогов 7](#_Toc70412916)

[2. АНАЛИЗ МОДЕЛЕЙ ПРЕДСТАВЛЕНИЯ ДЕРЕВЬЕВ 8](#_Toc70412917)

[2.1. Модель «родитель-потомок» 8](#_Toc70412918)

[2.2. Модель «потомок-потомок» 10](#_Toc70412919)

[ЗАКЛЮЧЕНИЕ 11](#_Toc70412920)

[ЛИТЕРАТУРА 13](#_Toc70412921)

[ПРИЛОЖЕНИЯ 14](#_Toc70412922)

[Приложение А. Спецификация вариантов использования 14](#_Toc70412923)

[Приложение Б. Cпецификация вариантов использования 17](#_Toc70412924)

# ВВЕДЕНИЕ

**Актуальность**

В современном мире существует множество видов хранения денежных активов – финитные валюты, недвижимость, фондовый рынок, драгоценные металлы и пр. Одним из таких видов хранения является крипто-валюта. Это достаточно популярный способ хранения денежных средств в современном мире, все благодаря его надежности. Однако, курсы отдельных крипто-валют постоянно меняются, что вводит неопытного пользователя в ступор.

Популярность крипто-валют как вида хранения денежных активов побудила собой создание множества различных крипто-валют. В связи с этим обычному пользователю, недостаточно погруженному в блокчейн технологии, однако желающему прикоснуться к миру блокчейн, становится очень сложно отслеживать котировки крипто-валют, разбираться в их видах, а также начать зарабатывать на крипто-биржах.

**Постановка задачи**

Целью выпускной квалификационной работы является разработка веб-приложения для отслеживания курсов крипто-валют. Для достижения поставленной цели необходимо решить следующие задачи:

1. Выполнить анализ предметной области;
2. Спроектировать веб-приложение;
3. Реализовать веб-приложение;
4. Произвести тестирование разработанной системы.

**Структура и содержание работы**

Работа состоит из введения, четырех глав, заключения и списка литературы. Объем работы составляет 50 страниц, объем списка литературы – 21 источник.

В первой главе описывается …

Вторая глава посвящена …

В третьей главе …

В приложении А содержится …

Глоссарий является необязательной частью текста работы.

# 1. АНАЛИЗ ЛИТЕРАТУРЫ И СМЕЖНЫХ ПРОЕКТОВ

1.1. Понятие блокчейн

Блокчейн – это неизменяемая, децентрализованная база данных, доступ к которой имеют все узлы компьютерной сети. Он хранит в себе данные в цифровом формате, при этом каждый последующий узел в сети имеет хэш предыдущего. Блокчейн гарантирует безопасность и точность в записи и хранении данных, без третьих лиц. Ключевое отличие блокчейн от стандартных баз данных заключается в определенном подходе в их структурировании.

Блокчейн собирает информацию в группы, так называемые «блоки», хранящие множества данных. «Блоки» имеют определенные возможности для их хранения, когда они переполняются, они закрываются и ссылаются на предыдущий заполненный узел (блок), вся последующая информация записывается по аналогичному принципу.

Блокчейн работает по принципу добычи цифровой информации для записи и распространения, но не для изменения. Все записи в блокчейн нельзя изменить, удалить или иным образом уничтожить. В следствие этого блокчейн принято называть DLT (с англ. Технология распределенного реестра).

В следствие высокой степени безопасности баз данных блокчейн, у множества пользователей возникла потребность хранить свои денежные сбережения в блокчейн. В следствие этого, блокчейн имеет ключевую роль в развитии крипто-валютных систем, например, таких как Bitcoin, Ethereum и других.

Таким образом, блокчейн – это неизменяемая и надежная база данных, благодаря которой, пользователи могут хранить свои сбережения и средства.

1.2. Понятие крипто-валюты. Виды крипто-валют. Наиболее популярные крипто-валюты и их ключевые особенности.

Крипто-валюта – особенный вид денежных средств в цифровой форме, использующий шифрование для защиты транзакций. Выпуск крипто-валюты происходит децентрализовано. Все участники сети равноправны и могут в любой момент переводить или получать крипто-валюту.

Крипто-валюты делятся на различные категории. В силу того, что Bitcoin – одна из самых первых валют, ее принято выделять в отдельную категорию, остальные делятся на несколько видов:

1. Альткоины;
2. Токены;
3. Стейблкоины;
4. NFT;
5. DeFi.

Альткоин (или альтернативный коин) – это любая крипто-валюта со своим блокчейн (за исключением Bitcoin). В связи с тем, что у Bitcoin открытый исходный код, разработчики альткойнов могут ускорять транзакции, оптимизировать процесс майнинга, создавать различные автоматизированные контракты, формировать базу для работы с крипто-приложениями.

Токен – цифровой актив, не имеющий своего собственного блокчейн. Вместо майнинга токены выпускаются в полной эмиссии. Зачастую такие активы выпускают различные компании с целью привлечения средств на развитие своих продуктов. Инвесторы, в свою очередь получают гарантии того, что компания выполнит перед ними свои обязательства.

Стейблкоины ­ цифровые деньги, цена которых привязана к материальным активам (обычная валюта, золото и пр.). Курс Bitcoin и альткоинов достаточно нестабилен и меняется каждый день, в то время как стейблкоины предельно стабильны, и несмотря на то, что их котировки могут меняться, происходит это гораздо реже и без резких скачков.

NFT – невзаимозаменяемые токены. Зачастую их используют для переноса блокчейн прав на владение уникальными активами. Например, произведения искусства, внутриигровые предметы в онлайн-играх и даже нотариально заверенные документы.

DeFi – децентрализованные финансовые сервисы. Стоит отметить, что это не отдельные крипто-валюты, а платформы для объединения различных видов цифровых активов и их функций.

Самыми популярными крипто-валютами являются:

1. Bitcoin (BTC);
2. Ethereum (ETH);
3. Binance Coin (BNB).

Таким образом, крипто-валюта – это один из способов хранить денежные активы, а также иметь возможность заработать на скачках и падениях их цен.

# 1.3. Биржи крипто-валют.

Крипто-валютная биржа – это площадка для осуществления торговли и обмена одно цифровой валюты на другую либо на валюты определенной страны (доллары, рубли и пр.). Биржа, наряду с майнингом, один из способов получения крипто-валюты.

Благодаря высокой волатильности, крипто-валюты используют как спекулятивный инструмент, это позволяет извлечь из нее максимальную прибыль.

Биржи делятся на два типа:

1. Площадки, на которых предусмотрена возможность обменивать крипто-валюту на фиатные валюты.
2. Площадки, на которых можно обменивать только крипто-валюту на крипто-валюту.

На текущий момент существует множество бирж со своими достоинствами и недостатками. Самыми популярными являются:

1. Poloniex;
2. Exmo.

Poloniex – американская крипто-валютная биржа, включающая в свой функционал помимо обменника крипто-валют детальную статистику.

Exmo – универсальный сервис, занимающийся торговлей самых популярных коинов:

* BitCoin;
* ETH;
* DogeCoin.

Основным преимуществом этой биржи является возможность обменивать валюту на рубли, доллары, евро и др.

Таким образом, крипто-валютная биржа, это место, где пользователь может обменять или купить крипто-активы за финитную или крипто-валюту.

# 1.4. Приложения для отслеживания курсов крипто-валют.

Для того, чтобы сделать правильный выбор крипто-валюты при инвестировании, а также узнать тенденции в мире блокчейн, пользователи используют веб-приложения с трекерами крипто-валют.

**Forklog**

Веб-приложение содержит в себе полный набор информации о крипто-валютах, биржах и блокчейн в целом. На главной странице приложения расположен список статей о блокчейн. Интерфейс главной страницы представлен на рисунке 1.



Рисунок 1 – Главная страница приложения «Forklog»

Помимо этого, forklog имеет страницы с котировками коинов, подробной информации о них, страницу со прогнозами тенденций в мире блокчейн. Внутри страницы нет возможности отсортировать коины по наибольшему изменению, а также узнать важные детали коина такие как история появления, или к какому типу он относится. Также, у forklog нет особенных возможностей, которые могли бы его выделить на фоне конкурентов – например, сравнение двух коинов на графиках. Однако, forklog предлагает пользователю детальную информацию на гибких и кастомизируемых графиках, дает возможность найти желаемый коин при помощи поля ввода. Интерфейс страницы котировок крипто-валют представлен на рисунке 2.



Рисунок 2 – Интерфейс страницы «курсы криптовалют» на «forklog»

**BitInfoCharts**

Веб-приложение содержит в себе детальную статистику по крипто-валютам, включающую в себя:

* Капитализацию на бирже;
* Цену на текущий момент в американском долларе;
* Количество блоков, полученных майнингом;
* Награду за блок, добытый майнингом;
* Интересную информацию о коине (дата первого добытого блока, количество звезд на GitHub, последний коммит в GitHub репозиторий).

Так же, как и Forklog, BitInfoCharts в интерфейсе имеет поиск и сортировку коинов по различным полям. В приложении есть отдельная страница со списком всех крипто-валют, на которой отображена базовая информация о них, дополнительно при нажатии на любой из коинов, приложение перенаправит пользователя на отдельную ссылку с более подробной информацией о крипто-валюте. Интерфейс страницы о BitCoin представлен на рисунке 3.



Рисунок 3 – Интерфейс страницы о BitCoin в веб-приложении «BitInfoCharts»

На странице с детальной информацией о конкретной крипто-валюте отображается график изменения цены валюты за последний месяц, цена указана в долларах США. Помимо этого, отображается количество крипто-валюты в целом, капитализация крипто-валюты. Также, отображается информация о средней цене транзакций, количестве блоков в блокчейн, объем блокчейн и гистограмма комиссии крипто-валюты за 24 часа. Гистограмма BitCoin показана на рисунке 4.



Рисунок 4 – Гистограмма комиссии для BitCoin

Основное преимущество «BitInfoCharts» – это детальная статистика и подробная информация по крипто-валютам. В то время как основной недостаток – устаревший дизайн.

Таким образом, основная цель приложений для отслеживания курсов крипто-валют – отображение статистики с наглядными графиками и тенденциями роста или спада конкретного коина.

# 1.5. Платформы для разработки веб-приложений

В современном мире веб-разработки существует 3 различных вида подхода к разработке приложений – SPA, MPA, PWA.

SPA (single page application) – одностраничное приложение, загружающееся на одну HTML-страницу, переход между страницами в таких приложениях реализован средствами языка JavaScript. Такой подход позволяет упростить разработку, а также увеличить скорость работы приложения. Однако, главным минусом у такого подхода является полное отсутствие SEO оптимизации, так как рендеринг страницы происходит на клиенте, в силу этого, поисковые роботы Google (и другие) видят страницу приложения пустой. Однако, в случаях, где контент статичен и важен для SEO, этот недостаток SPA, может закрыть серверный рендеринг, позволяющий писать собственный API для рендеринга на сервере.

MPA (multi page application) – многостраничное приложение, один из первых подходов в разработке приложений, при нем страница полностью обновляется при малейшем изменении состояний на клиенте. Главный плюс такого подхода, в отличие от SPA – отличная SEO оптимизация, т.к. рендер приложения происходит на сервере. Минусов у такого подхода много. Первый из них – сложность разработки. Для разработки MPA требуется использовать фреймворки, заточенные на фронтенд и бекенд одновременно, например, Flask или Ruby on rails, что усложняет разработку и делает ее менее поддерживаемой и вариативной. Второй большой минус MPA – плохая скорость работы, в силу того, что перерисовка страницы происходит каждый раз при изменении состояний в приложении.

PWA (progressive web applications) – прогрессивное веб приложение, позволяющее запускать себя как в браузере, так и в качестве десктопного или мобильного приложения. Главная особенность и преимущество PWA – кроссплатформенность, приложение, написанное под персональный компьютер может быть использовано в качестве мобильного или веб-приложения. Однако, минусом такого подхода является поддержка некоторых возможностей таких приложениях в некоторых браузерах (Safari, Edge).

На данный момент, веб-приложения пишутся на различных фреймворках и библиотеках языка JavaScript, в силу их удобства и скорости разработки. Они не требуют работы с DOM-деревом, вместо этого в них есть специальные методы и функции для работы с событиями, референсами и куки. Самые популярные JavaScript-фрейморки на данный момент:

* React (для серверного рендеринга React приложений используют Next);
* Vue (для серверного рендеринга Vue приложений используют Nuxt);
* Svelte (для серверного рендеринга Svelte приложений используют SvelteKit).

Также, важнейшим трендом веб-разработки является типизация. Для типизации в веб-приложениях используют TypeScript. Типизация очень полезна для командной разработки, а также в случаях, где нужно работать с большим количеством запросов и валидацией.

Помимо этого, в современных веб-приложениях используют библиотеки состояний для хранения данных и устранения проблемы сильной связанности (prop-drilling) между множеством компонентов. Самые популярные библиотеки для работы с состояниями:

* Redux (Redux Toolkit);
* Vuex (библиотека состояний для Vue);
* MobX;
* Zustand.

Самая часто используемая из них – Redux, из-за хорошего отладчика в виде Redux DevTools. Однако в сообществе разработчиков есть и люди, которые скептически относятся к Redux из-за сильной нагроможденности в редьюсерах. Проблема была решена с выпуском Redux Toolkit, однако в некоторых проектах до сих пор используется старая версия Redux.

Таким образом, современная веб разработка имеет множество различных способов и подходов к реализации веб приложений. Каждый из них имеет право на существование, выбор инструмента и подхода к разработке зависит от конкретной задачи разработчика.

# 2. АНАЛИЗ МОДЕЛЕЙ ПРЕДСТАВЛЕНИЯ ДЕРЕВЬЕВ

2.1. Модель «родитель-потомок»

Модель, известная под названием «родитель-потомок», является наиболее простой в реализации. В соответствии с этой моделью, каждая запись таблицы будет соответствовать узлу дерева и хранить его уникальный идентификатор и ссылку на родительский узел. Если узел является корнем, то ссылка на родителя в его записи будет пустой (рисунок 1). Пример дерева и его представление согласно данной модели изображены на рисунке 1.

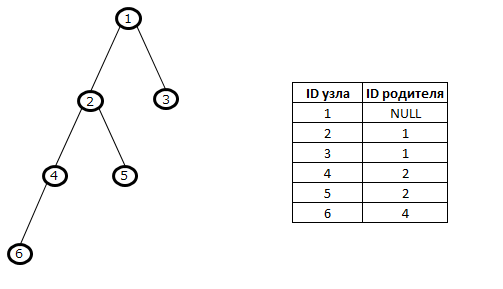


Рисунок 1 – Пример дерева и его представление согласно модели   
«родитель-потомок»

Пусть алгоритм решает алгоритмическую задачу, вычисляемую по формуле (1):

|  |  |  |
| --- | --- | --- |
|  |  | (1) |

где – множество выходных значений, причем ;

является константой или вычислимой функцией параметров .

Предположим, что , , – подмножества множества , удовлетворяющие следующим условиям:

1. ;
2. ;
3. одно или два из множеств могут быть пустыми.

Рассмотрим множество Q-термов , удовлетворяющее условиям:

1. – безусловный Q-терм, ;
2. – условный Q-терм, ;
3. – условный бесконечный Q-терм, .

Если в тексте формула одна, ее можно не нумеровать.

Алгоритм консенсуса tendermint основан на задаче о византийских генералах и представлен в листинге 1 [36].

Особенностью данного алгоритма является то, что решения о включении блока в цепочку принимаются специальными участниками, которых называют валидаторами (листинг 1).

Это создает некоторую централизацию, которая представлена на рисунке 14 [41].

Рис. 8. Архитектура системы ECoD

CREATE TABLE IF NOT EXISTS `gpnode` (

`idNod` int(11) NOT NULL AUTO\_INCREMENT,

`idCour` int(11) NOT NULL,

`idMod` int(11) DEFAULT NULL,

`leftKey` int(11) NOT NULL,

`rightKey` int(11) NOT NULL,

`level` int(11) NOT NULL,

PRIMARY KEY (`idNod`),

KEY `idCour` (`idCour`),

KEY `idMod` (`idMod`)

) ENGINE=InnoDB AUTO\_INCREMENT=385 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

Рисунок 14 – Код, создающий таблицу gpnode в СУБД MySQL

Листинг 1 – Функция извлечения признаков

def get\_features(wav):

y\_mel = librosa.feature.melspectrogram(wav, sr=16000)

spectr = librosa.power\_to\_db(y\_mel).astype(np.float32)

mfcc = librosa.feature.mfcc(S=spectr, sr=16000).astype(np.float32)

x\_mfcc = np.swapaxes(mfcc, 0, 1)[None]

mean = (np.mean(x\_mfcc[0], axis=0) + 1e-8)

x\_mfcc[0] -= mean

x\_mfcc = pad\_sequences(x\_mfcc, 430,

dtype='float32', padding='post',

truncating='post', value=0.0)

return x\_mfcc

Как видно из листинга, функция возвращает значение x\_mfcc.

2.2. Модель «потомок-потомок»

Содержание второго параграфа второй главы.

В тексте работы можно использовать многоуровневые списки:

1. первым способом создания многоуровневого списка является:

* использование в верхнем уровне нумерации с круглой скобкой;
* использование в вложенном уровне маркированного списка;
* пункты этого вида многоуровневого списка могут быть написаны в несколько строк;

1. при этом пункты всех уровней начинаются со строчной буквы:

* все пункты, кроме последнего, заканчиваются точкой с запятой;
* последний пункт/подпункт заканчивается точкой.

Вторым способом организации многоуровневого списка является следующий.

* + - 1. Преамбула (предложение, непосредственно предществующее списку) должна представлять собой законченное предложение.
  1. Пункты и подпункты начинаются с заглавной буквы.
  2. Отступ до первой буквы первого уровня списка – 1,25 см.
     + 1. Пункты этого вида многоуровневого списка могут содержать несколько предложений. Все предложения заказчиваются точкой.
  3. Отступ до первой буквы второго уровня списка – 2 см.
  4. В пункте должно быть не менее двух подпунктов.

Каждый параграф главы должен заканчиваться кратким выводом.

Каждая глава должна заканчиваться кратким выводом.

# ЗАКЛЮЧЕНИЕ

В рамках данной работы был разработан веб-сервис, который по аудиозаписи речи позволяет определить психологический подтип человека (этик, логик) в классификации К.Г. Юнга. При этом были решены следующие задачи.

Произведен обзор литературы и существующих приложений по предметной области.

* + - 1. Подготовлена обучающая и тестовая выборки.
      2. Выполнено проектирование архитектуры нейронной сети для решения задачи классификации аудиозаписей.
      3. Проведено обучение и тестирование спроектированной нейронной сети.
      4. Разработан и протестирован веб-сервис для распознавания психологического подтипа человека в классификации К.Г. Юнга по аудиозаписи речи.

В настоящий момент производится интеграция разработанного веб-приложения в систему ECoD. В будущем планируется продолжать разработку и улучшение веб-приложения, в частности реализовать поиск учебных модулей для сопоставления с узлами граф-плана, работу с образовательными компетенциями и др.

В рамках работы были опубликованы следующие статьи.

1. Golodov V.A., Mittseva A.A. Weld Segmentation and Defect Detection in Radiographic Images of Pipe Welds. // 2019 International Russian Automation Conference (RusAutoCon), 2019. – 145–156 pp.
2. Митцева А.А., Голодов В.А. Программный комплекс для автоматического распознавания дефектов сварных швов по снимкам, полученным с установки ренгенотелевизионного контроля. // Молодой исследователь, 2019. – С. 378–382.

Также в рамках данной работы были выступления на следующих мероприятиях.

Выставка и конференция научно-технических и творческих работ студентов ЮУрГУ, 24–26 апреля 2019, диплом II степени.

Международная конференция Seymartec digital. Автоматизация металлургического производства – 2019.

# ЛИТЕРАТУРА

1. Богданов Д.В. Оптимальный способ хранения и обработки древовидных структур в базах данных. // Программные продукты и системы, 2009. – № 1. – С. 140–142.
2. ГОСТ 33245-2015 (ISO/IEC TR 29163-1:2009) Информационные технологии (ИТ). Эталонная модель распределенного объекта контента (SCORM®) 2004 3-я редакция. Часть 1. Обзор. Версия 1.1. [Электронный ресурс] URL: http://docs.cntd.ru/document/1200127254 (дата обращения: 10.05.2020 г.).
3. Жигальская Н.С. Модель вариантов использования универсальной среды электронного обучения UniCST. // Инновационные технологии обучения: проблемы и перспективы: Материалы всерос. науч.-метод. конф. (29-30 марта 2008 г., Липецк) – Липецк: Изд-во ЛГПУ, 2008. – С. 204–207.
4. Маликов А.В. Ориентированные графы в реляционных базах данных. // Доклады ТУСУР, 2008. – № 2(18). – Ч. 2. – С. 100–104.
5. Desktop Browser Market Share Worldwide | StatCounter Global Stats. [Электронный ресурс] URL: http://gs.statcounter.com/browser-market-share/desktop/worldwide (дата обращения: 24.05.2020 г.).
6. Dracula Graph Library. [Электронный ресурс] URL: https://www.graphdracula.net (дата обращения: 16.03.2020 г.).
7. Ivanova O.N., Silkina N.S. Competence-Oriented Model of Representation of Educational Content. // Proceedings of the 40th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO'2017, Opatija, Croatia, May 22–26, 2017. IEEE, 2017. – 791–794 pp.
8. Юнг К.Г. Психологические типы / под ред. Зеленского В. – СПб: Азбука, 2001. – 736 с.
9. Маликов А.В. Ориентированные графы в реляционных базах данных. // Доклады ТУСУР, 2008. – № 2(18). – Ч. 2. – С. 100–104.
10. Desktop Browser Market Share Worldwide | StatCounter Global Stats. [Электронный ресурс] URL: http://gs.statcounter.com/browser-market-share/desktop/worldwide (дата обращения: 24.05.2020 г.).
11. Dracula Graph Library. [Электронный ресурс] URL: https://www.graphdracula.net (дата обращения: 16.03.2020 г.).
12. Ivanova O.N., Silkina N.S. Competence-Oriented Model of Representation of Educational Content. // Proceedings of the 40th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO'2017, Opatija, Croatia, May 22–26, 2017. IEEE, 2017. – 791–794 pp.

# литература

# ПРИЛОЖЕНИЯ

Приложение А. Спецификация вариантов использования

Спецификация вариантов использования (ВИ) системы приведена в таблицах 1–4.

Таблица 1 – Спецификация ВИ «Смотреть граф-план»

|  |
| --- |
| Прецедент: Смотреть граф-план |
| ID: 1 |
| Краткое описание:  Визуализация граф-плана курса. |
| Главные актеры:  Пользователь |
| Второстепенные актеры: Нет |
| Предусловия: отсутствуют |
| Основной поток:  1. Прецедент начинается, когда пользователь открывает вкладку работы с граф-планом. 2. Приложение формирует визуализацию граф-плана на основе данных, полученных из базы данных. |
| Постусловия: 1. Визуализация граф-плана выводится в рабочую область программы. |
| Альтернативные потоки: I. Ошибка при обращении к базе данных.  1. Приложение выводит сообщение об ошибке.  II. В базе данных отсутствуют данные о граф-планах.  1. Приложение выводит сообщение об отсутствии граф-планов. |

Продолжение приложения А

Таблица 2 – Спецификация ВИ «Добавить узел»

|  |
| --- |
| Прецедент: Добавить узел |
| ID: 2 |
| Краткое описание:  Добавление нового узла курса. |
| Главные актеры:  Пользователь |
| Второстепенные актеры: Нет |
| Предусловия:  Выбран целевой узел граф-плана |
| Основной поток.   * + 1. Прецедент начинается, когда пользователь нажимает кнопку «Добавить дочерний узел».     2. Приложение изменяет структуру граф-плана курса, добавляя к выбранному узлу дочерний узел.     3. Если в меню приложения был выбран модуль, то ссылка на этот модуль записывается в добавленный узел. |
| Постусловия.  1. В граф-план курса в базе данных был добавлен новый узел. |
| Альтернативные потоки.  I. Не выбрана цель для действия.  1. Приложение выводит сообщение об ошибке. |

Таблица 3 – Спецификация ВИ «Переопределить модуль»

|  |
| --- |
| Прецедент: Переопределить модуль |
| ID: 3 |
| Краткое описание:  Изменение ссылки на учебный модуль в узле курса. |
| Главные актеры:  Пользователь |
| Второстепенные актеры: Нет |
| Предусловия: 1. Выбраны узел и номер модуля |
| Основной поток:  1. Прецедент начинается, когда пользователь нажимает на кнопку «Изменить номер модуля». 2. Приложение добавляет ссылку на модуль, выбранный в меню приложения, к текущему узлу |
| Альтернативные потоки: I. Не выбрана цель для действия.  1. Приложение выводит сообщение об ошибке. |

Окончание приложения А

Таблица 4 – Спецификация ВИ «Удалить узел»

|  |
| --- |
| Прецедент: Удалить узел |
| ID: 4 |
| Краткое описание:  Удаление узла курса. |
| Главные актеры:  Пользователь |
| Второстепенные актеры: Нет |
| Предусловия: 1. Выбран целевой узел граф-плана |
| Основной поток:  1. Прецедент начинается, когда пользователь нажимает на кнопку «Удалить узел». 2. Приложение удаляет из граф-плана курса выбранный узел и все дочерние по отношению к нему узлы. |
| Постусловия: 1. Выбранный узел и все его дочерние узлы удалены из граф-плана курса в базе данных. |
| Альтернативные потоки: I. Ошибка при обращении к базе данных.  1. Приложение выводит сообщение об ошибке.  II. Не выбрана цель для действия.  1. Приложение выводит сообщение об ошибке. |

# Приложение Б. Скриншоты приложения

Скриншоты разработанного приложения приведены на рисунках 1–10.

Нумерация таблиц и рисунков в приложении начинается с единицы и является сквозной по всем приложениям.

Если приложение одно, то оно не нумеруется. Если приложений несколько, то в содержание выносится отдельная позиция «ПРИЛОЖЕНИЯ» (заголовок первого уровня), приложения нумеруются буквами А, Б, В, Г и т.д. (заголовки второго уровня).

В тексте в приложениях применяются те же правила оформления, что и к основному тексту ВКР.