

Федеральное государственное образовательное бюджетное учреждение
высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финансовый университет)
Колледж информатики и программирования

Выпускная квалификационная работа
на тему: «Разработка экономической инфографики для НИФИ»

Студент группы. 4ПКС-116 Зайцев Никита Валерьевич 
(№ учебной группы). (фамилия имя отчество) (подпись)

Образовательная программа
09.02.03 «Программирование в компьютерных системах»
(индекс и наименование специальности)

Форма обучения: очная

Руководитель _____ Козлобаев А.А.
(подпись) (Фамилия И.О.)

Консультант _____ Варьяш И.Ю.
(при наличии) (подпись) (Фамилия И.О.)

Председатель ПЦК _____ Пестов А.И.
(подпись) (Фамилия И.О.)

Москва – 2020

Федеральное государственное образовательное бюджетное учреждение
высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финансовый университет)

Колледж информатики и программирования

УТВЕРЖДАЮ

(председатель
ПЦК)

(подпись) (И.О.
Фамилия)

«____» 20 __ г.

ЗАДАНИЕ
на выпускную квалификационную работу

обучающемуся Зайцеву Никите Валерьевичу

(фамилия имя отчество)

1. Тема выпускной квалификационной работы
Разработка сайта "Экономическая инфографика" для НИФИ

2. Срок сдачи обучающимся законченной выпускной квалификационной
работы «____» 20 __ г.

3. Исходные данные
Экономическая инфографика

4. Перечень подлежащих разработке задач/вопросов:
4.1 Анализ требований и определение спецификаций
4.2 Выбор языка программирования
4.3 Проектирование веб-приложения
4.4 Проектирование базы данных
4.5 Разработка веб-приложения

4.6 Тестирование веб-приложения

5. Перечень графического/ иллюстративного/ практического материала:

5.1 Структурная схема программы

5.2 Схема базы данных

5.3 Презентация выпускной квалификационной работы

6. Консультант по выпускной квалификационной работе с указанием
относящихся к ним разделов работы)

Козлобаев А.А., преподаватель ВКК (проектирование веб-приложения,
тестирование веб-приложения)

Дата выдачи задания «5 » июня 2020г.

Руководитель _____

(подпись)

(И.О. Фамилия)

Задание принял к исполнению «5 » июня 2020г.



(подпись обучающегося)

Содержание

Введение.....	5
Глава 1 Исследование по теме	7
1.1 Описание предметной области	7
1.2 Среда и язык программирования.....	10
1.3 Нормативная документация.....	12
Глава 2 Разработка приложения	13
2.1 Технологии, применяемые при разработки веб-сайта.	13
2.2 Разработка схемы приложения	18
2.3 Проектирование базы данных	19
2.4 Разработка внешнего вида приложения	20
2.5 Разработка внутренних модулей приложения	28
2.6 Тестирование приложения	28
2.7 Руководство системного программиста	30
2.8 Руководство программиста	34
2.9 Руководство администратора	41
2.10 Руководство пользователя.....	49
Заключение	52
Список использованных источников	53
Приложение А	54

Введение

Рыночная экономика любого уровня – от государства до муниципального образования, хозяйствующего субъекта и отдельного домохозяйства – не может функционировать вне финансового рынка. Каждый субъект экономических отношений в той или иной степени вовлечен в финансовые отношения, использует механизмы и инструменты финансового рынка, осознанно или по необходимости вписываясь в экономические реалии современной жизни, зная или интуитивно догадываясь об используемых финансовых инструментах. Усвоение основ функционирования финансового рынка с позиций его участников, финансовых инструментов, оценки потенциала отдельных инвесторов и особенностей функционирования инфраструктуры позволит повысить уровень знаний, умений и навыков работы на рынке.

Функционирование экономики невозможно без финансового рынка. Финансовый рынок как кровеносная система рыночной экономики позволяет обеспечить бесперебойное функционирование всех ее сегментов. От того, как работает финансовый рынок, зависит состояние экономики в целом и отдельных ее составляющих. Чем мощнее финансовый рынок, чем слаженнее работают его сегменты, тем устойчивее текущее состояние экономики и успешнее ее развитие.

Целью выпускной квалификационной работы является разработка портала для отображения экономической инфографики для более наглядного просмотра прогнозов по финансовому рынку, которая позволит участникам торгов наглядно видеть прогнозируемую ситуацию на рынке.

Для достижения этой цели необходимо реализовать следующие задачи:

- Разработать общую схему и концепцию сайта
- Определиться со стеком технологий
- Разработать дизайн сайта
- Разработать необходимые модели к базам данных
- Разработать приложение;

- Реализовать основные функциональные требования к Программному Обеспечению;
- Тестирование приложения;
- Разработать руководства по использованию приложения;

Глава 1 Исследование по теме

1.1 Описание предметной области

Целью выпускной квалификационной работы является разработка портала для отображения экономической инфорграфики для более наглядного просмотра прогнозов по финансовому рынку, которая позволит участникам торгов наглядно видеть прогнозируемую ситуацию на рынке. На портале будет 6 прогнозируемых сегментов – USD/RUB, Индекс МосБиржи (МОЕХ), Индекс ОФЗ (RGBI), Золото, Нефть (Brent) и XRP/USD (Ripple/USD).

Экономическая инфографика – это рассчитываемый Научно-Исследовательским Финансовым Институтом (НИФИ) индикатор ожидаемых условий биржевой торговли на финансовом рынке России (индикатор условий финансирования, ИУФ) основан на результатах ежедневного анализа мнений профессиональных участников рынка о перспективах изменения ценовой конъюнктуры биржевой торговли. В круг компаний входит более 40 участников, представляющих российские и зарубежные банки, инвестиционные и негосударственные пенсионные фонды, консалтинговые фирмы, международные информационные и рейтинговые агентства, экономические ведомства РФ.

Дополнительный индикатор релевантности ожидаемых условий (ИРО) показывает выраженные в весах структурные соотношения между ожиданиями участников рынка и факторами формирования условий рынка, и служит для оценки вероятности следования поведения участников торгов их ожиданиям.

В основе расчетов лежит широко используемая в современной экономике методология РМІ - «диффузный индикатор», модифицированный НИФИ для финансового рынка с учетом процентных соотношений между долями в численности повышательных и понижательных оценок с учетом распределения их соотношения в данных об отсутствии изменений в период наблюдений. Показатель индикатора выше 50.0 пунктов указывает на повышательный тренд, ниже – на понижательный.

Рассчитываемые НИФИ индикаторы основных сегментов финансового рынка (валютного, фондового, ценных бумаг, а также золота и нефти) используются для уточнения кратко- и среднесрочных тенденций в конъюнктуре финансового рынка, при разработке и реализации инвестиционной политики. НИФИ оставляет за собой возможность коррекции обзора в соответствии с изменяющимися перспективами биржевой торговли. Исторические данные, связанные с базовыми (некорректированными) числами доступны для подписчиков.

Сама инфографика это проект Научно-Исследовательского Финансового Института. НИФИ был создан по инициативе Наркомата финансов 27 февраля 1937 года. В его состав вошли семь секций: бюджетная, налоговая, денежного обращения, кредита и оборотных средств, сводно-балансовая, истории советских финансов, учета выполнения финансовых планов. Структура подразделений института и организационная привязка к Наркомфину с полной определенностью демонстрировали, для решения каких задач он предназначен.

В годы Великой Отечественной войны институт не функционировал. Все годные к военной службе сотрудники ушли на фронт. В марте 1945 г. было утверждено новое «Положение о НИФИ Наркомата финансов СССР». Институт активно включился в разработку финансовых проектов восстановления хозяйства страны, разрушенного войной. Научные сотрудники принимали участие в подготовке и проведении послевоенной денежной реформы.

Начало нового этапа истории НИФИ относится к 1956 г. Совет министров СССР принял постановление о расширении его деятельности. Институт стал ведущим центром по теоретической разработке проблем финансов и обобщению передового опыта финансовых органов и банков. В этом качестве в 1960–80-е гг. НИФИ активно занимался проблемами повышения эффективности хозяйствования и социальными вопросами. Сотрудники института готовили материалы по финансовым аспектам

реформы управления народным хозяйством, проводили расчеты финансовых показателей к пятилетним планам. Годовые планы НИР утверждала коллегия министерства.

В 1991 г. НИФИ включили в структуру Министерства экономики и финансов Российской Федерации. В августе 1992 г. после разделения министерств институт вернулся в ведение Минфина. По уставу, утвержденному Минфином России в августе 1998 г., НИФИ стал государственным научным учреждением общеэкономического профиля. В эти годы институт расширил круг исследований по основным направлениям своей деятельности, участвовал в разработке новых законов в целях развития рыночной экономики и соответствующих ей финансово-кредитных отношений. Велась подготовка бюджетной реформы, включая совершенствование межбюджетных отношений, а также законопроекта о государственном финансовом контроле. Вместе с Институтом профессиональных бухгалтеров России НИФИ выступил головным учреждением по реализации программы реформирования бухгалтерского учета в соответствии с международными стандартами финансовой отчетности.

С момента образования в НИФИ работали такие видные ученые экономисты, как: В. П. Дьяченко, И. Д. Злобин, Д. П. Боголепов, К. Н. Плотников, Н. Н. Ровинский, А. В. Бачурин, Г. П. Косяченко, С. А. Ситарян, В. К. Сенчагов, Л. Е. Бабашкин, С. И. Лушин, В. Н. Семенов, Ю. М. Артемов, Г. В. Базарова, Р. Д. Винокур, А. Н. Кашаев, Г. Б. Поляк, К. Я. Чижов, С. М. Борисов, Д. Д. Бутаков, Н. В. Шеин, Ю. М. Осипов, И. Г. Русакова, О. В. Можайсков, Ю. А. Данилевский, Е. В. Коломин. Некоторые из ученых НИФИ известны и как крупные руководители органов государственной системы финансового управления.

В 2005 г. НИФИ был включен в состав Академии бюджета и казначейства Минфина России (с 2011 г. – Государственный университет Минфина России) в качестве структурного подразделения.

В мае 2012 г. Правительство Российской Федерации в целях создания оптимальных условий для развития фундаментальных и прикладных научных исследований в области экономики и финансов восстановило самостоятельность института в качестве федерального государственного бюджетного учреждения «Научно-исследовательский финансовый институт». Функции учредителя были возложены на Минфин России.

Анализ ресурсов в сети показал, что аналогичной инфографики по прогнозу сегментов биржи отсутствует.

Для достижения этой цели необходимо:

- Разработать общую схему и концепцию сайта;
- Определиться с конкретным стеком технологий;
- Разработать дизайн сайта;
- Разработать необходимые модели к базам данных;
- Написать контроллеры, которые будут выводить страницы и обрабатывать выводимую информацию;
- реализовать основные функциональные требования к ПО;
- протестировать готовый программный продукт;
- разработать руководство по использованию ПО.

Задачи, которые сайт должен будет выполнять:

- Вывод графиков прогноза по сегментам
- Возможность отправки уведомлений об обновлении прогнозов

1.2 Среда и язык программирования

Разработка портала выполнена на языке Python, версии 3.7. Python – это высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. В то же время стандартная библиотека включает большой объём полезных функций.

Python поддерживает структурное, объектно-ориентированное, функциональное, императивное и аспектно-ориентированное

программирование. Основные архитектурные черты — динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений, высокоуровневые структуры данных. Поддерживается разбиение программ на модули, которые, в свою очередь, могут объединяться в пакеты.

Эталонной реализацией Python является интерпретатор CPython, поддерживающий большинство активно используемых платформ. Он распространяется под свободной лицензией Python Software Foundation License, позволяющей использовать его без ограничений в любых приложениях, включая проприетарные. Есть реализация интерпретатора для JVM с возможностью компиляции, CLR, LLVM, другие независимые реализации. Проект PyPy использует JIT-компиляцию, которая значительно увеличивает скорость выполнения Python-программ.

Python — активно развивающийся язык программирования, новые версии с добавлением/изменением языковых свойств выходят примерно раз в два с половиной года. Язык не подвергался официальной стандартизации, роль стандарта де-факто выполняет CPython, разрабатываемый под контролем автора языка. В настоящий момент Python занимает третье место в рейтинге TIOBE с показателем 8,5 %. Аналитики отмечают, что это самый высокий балл Python за все время его присутствия в рейтинге.

Выбранная среда разработки — JetBrains PyCharm Professional. Это интегрированная среда разработки для языка программирования Python. Предоставляет средства для анализа кода, графический отладчик, инструмент для запуска юнит-тестов и поддерживает веб-разработку на Django. PyCharm разработана компанией JetBrains на основе IntelliJ IDEA. Это кроссплатформенная среда разработки, которая совместима с Windows, MacOS, Linux. PyCharm Community Edition (находится под лицензией JetBrains Private Policy 2.1, а PyCharm Professional является проприетарным ПО).

Возможности данной среды разработки:

- Статический анализ кода, подсветка синтаксиса и ошибок;
- Навигация по проекту и исходному коду: отображение файловой структуры проекта, быстрый переход между файлами, классами, методами и использованием методов;
- Рефакторинг: переименование, извлечение метода, введение переменной, введение константы, подъём и спуск метода и т.д.;
- Инструменты для веб-разработки с использованием фреймворка Django;
- Встроенный отладчик для Python;
- Встроенные инструменты для юнит-тестирования;
- Разработка с использованием Google App Engine;
- Поддержка систем контроля версий: общий пользовательский интерфейс для Mercurial, Git, Subversion, Perforce и CVS с поддержкой списков изменений и слияния;

1.3 Нормативная документация

- ГОСТ Р ИСО/МЭК 15910-2002 — Процесс создания документации пользователя программного средства
- ГОСТ 19781—90 Единая система программной документации.
- ГОСТ Р ИСО 9241-151-2014 Эргономика взаимодействия человек-система. Часть 151 Руководство по проектированию пользовательских интерфейсов сети Интернет
- ГОСТ 34.602-89 «Техническое задание на создание автоматизированной системы»
- Федеральный закон "О персональных данных" № 152-ФЗ
- Доктрина информационной безопасности Российской Федерации

Глава 2 Разработка приложения

2.1 Технологии, применяемые при разработки веб-сайта.

Как уже писалось выше, основным языком разработки был выбран Python 3. Это высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. В то же время стандартная библиотека включает большой объём полезных функций.

Python поддерживает структурное, объектно-ориентированное, функциональное, императивное и аспектно-ориентированное программирование. Основные архитектурные черты — динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений, высокоуровневые структуры данных. Поддерживается разбиение программ на модули, которые, в свою очередь, могут объединяться в пакеты.

Основным фреймворком для разработки сайта будет Django. Это web-фреймворк для языка Python 3. Это свободный фреймворк для веб-приложений на языке Python, использующий шаблон проектирования MVC. Проект поддерживается организацией Django Software Foundation.

Сайт на Django строится из одного или нескольких приложений, которые рекомендуется делать отчуждаемыми и подключаемыми. Это одно из существенных архитектурных отличий этого фреймворка от некоторых других (например, Ruby on Rails). Один из основных принципов фреймворка — DRY (англ. Don't repeat yourself)

Также, в отличие от других фреймворков, обработчики URL в Django конфигурируются явно при помощи регулярных выражений.

Для работы с базой данных Django использует собственный ORM, в котором модель данных описывается классами Python, и по ней генерируется схема базы данных.

Для хранения данных была выбрана следующая система управления базами данных — MySQL. Это система, предназначенная для хранения и

обработки информации. Комплекс таблиц, взаимосвязанных между собой, для доступа к которым применяется система управления базами данных (СУБД) MySQL. По сути, MySQL – это специальная программа с открытым кодом, которая используется на сервере SQL. Данная программа не способна обрабатывать большое количество информации, однако она идеальна для небольших и крупных веб-ресурсов. Как и любой продукт, MySQL имеет сильные и слабые стороны.

Преимущества MySQL:

- простоту в работе (установка не требует наличия специальных навыков, для работы может потребоваться дополнительное приложение GUI, которое делает работу с СУБД еще проще);
- богатый функционал;
- безопасность (включает большое количество функций для обеспечения безопасности, причем они поддерживаются по умолчанию);
- масштабируемость (может работать с большими объемами информации);
высокая скорость (увеличить производительность удалось путем упрощения некоторых стандартов).

Недостатки MySQL:

- наличие ограничений функционала (имеет большинство возможностей SQL, но не все, а иногда они требуются для работы в особо «капризных» приложениях);
- платную поддержку даже для бесплатной версии.

Исходя из этих достоинств и недостатков было принято решение использовать именно MySQL.

Как говорилось в 2.1.2., для основы верстки сайта был выбран фреймворк Bootstrap четвертой версии. Bootstrap - свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML- и CSS-шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript-расширения.

К его преимуществам можно отнести:

- Сокращения времени, необходимого для создания макета сайта. Фреймворк предлагает достаточное количество шаблонов и готовых решений;
- Кросс-браузерность и адаптивность. Сайты, созданные с помощью Bootstrap, идентично отображаются на разных устройствах и в современных браузерах;
- Простота использования. Для работы с фреймворком потребуются элементарные навыки верстки;
- Легкость в освоении. Достаточное количество обучающих видеоматериалов поможет вам быстро и без особых трудностей освоить Bootstrap.

Недостатков у Bootstrap, по большому счету всего два. Первый – кода обычно в библиотеке написано больше, чем если бы вы написали при разработке с нуля. Потому что, когда вы делаете самостоятельно, вы реализуете только необходимый функционал и все. В Bootstrap же есть все на все случаи жизни. Даже то, что вам может не пригодиться. Но опять же, эта проблема очень легко решается тем, что вы можете сами выбирать, какие компоненты фреймворка загрузить в CSS-файл (файл, в котором хранятся стили). Например, вы вообще можете скачать только сетку, а все остальное делать самостоятельно. Второй недостаток – шаблонный дизайн. Да, действительно, очень часто на разных сайтах можно увидеть одинаковые кнопки. Но и эта проблема легко решается, потому что она будет существовать только в том случае, если вы будете использовать только готовые компоненты фреймворка и ничего никогда не кастомизировать под себя.

Сами страницы сверстаны, используя язык разметки HTML, версии 5. HTML5 – это язык для структурирования и представления содержимого всемирной паутины. Это пятая версия HTML. Хотя стандарт был завершён (рекомендованная версия к использованию) только в 2014 году, уже с 2013 года браузерами оперативно осуществлялась поддержка, а разработчиками —

использование рабочего стандарта. Цель разработки HTML5 — улучшение уровня поддержки мультимедиа-технологий с одновременным сохранением обратной совместимости, удобочитаемости кода для человека и простоты анализа для парсеров.

HTML 5 предоставляет множество новых элементов, благодаря которым структурирование веб-документов будет существенно упрощено. Множество страниц, основанных на HTML 4, содержали в себе типичные структурные фрагменты, таких как заголовок, колонки текста, нижний колонтитул и т.д. На сегодняшний день обычным решением считается определение этих блоков с помощью контейнеров `div` с указанием для каждого из них имени соответствующего класса или уникального идентификатора.

Для описания стилей используется язык CSS. Это формальный язык описания внешнего вида документа, написанного с использованием языка разметки.

Преимущественно используется как средство описания, оформления внешнего вида веб-страниц, написанных с помощью языков разметки HTML и XHTML, но может также применяться к любым XML-документам, например, к SVG или XUL. CSS используется создателями веб-страниц для задания цветов, шрифтов, стилей, расположения отдельных блоков и других аспектов представления внешнего вида этих веб-страниц. Основной целью разработки CSS являлось отделение описания логической структуры веб-страницы (которое производится с помощью HTML или других языков разметки) от описания внешнего вида этой веб-страницы (которое теперь производится с помощью формального языка CSS). Такое разделение может увеличить доступность документа, предоставить большую гибкость и возможность управления его представлением, а также уменьшить сложность и повторяемость в структурном содержимом.

Кроме того, CSS позволяет представлять один и тот же документ в различных стилях или методах вывода, таких как экранное представление, печатное представление, чтение голосом (специальным голосовым браузером

или программой чтения с экрана), или при выводе устройствами, использующими шрифт Брайл.

Для визуализации графиков используется библиотека ChartJS на языке JavaScript.

JavaScript это — мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. Является реализацией стандарта ECMAScript (стандарт ECMA-262).

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам.

Основные архитектурные черты: динамическая типизация, слабая типизация, автоматическое управление памятью, прототипное программирование, функции как объекты первого класса.

На JavaScript оказали влияние многие языки, при разработке была цель сделать язык похожим на Java. Языком JavaScript не владеет какая-либо компания или организация, что отличает его от ряда языков программирования, используемых в веб-разработке.

JavaScript является объектно-ориентированным языком, но используемое в языке прототипирование обуславливает отличия в работе с объектами по сравнению с традиционными класс-ориентированными языками. Кроме того, JavaScript имеет ряд свойств, присущих функциональным языкам — функции как объекты первого класса, объекты как списки, карринг, анонимные функции, замыкания — что придаёт языку дополнительную гибкость.

ChartJS — это популярный инструмент, который предназначен для создания графиков и диаграмм. В данной серии уроков будут раскрыты все аспекты работы с этой библиотекой. Вы сможете создавать адаптивные диаграммы любой сложности на основе HTML5 Canvas.

Данная библиотека позволяет без особого труда создавать графики и диаграммы любого типа, а также выстраивать данные на диапазоне времени и логарифмической шкале. Также в неё встроены средства работы с анимацией, что позволит эффективно видоизменять графики в зависимости от новых данных, а также экспериментировать с цветом

2.2 Разработка схемы приложения

Прежде чем начать разработку сайта, необходимо проработать его будущую архитектуру. После обсуждения технического задания, была выверена карта сайта, которая представлена на Рисунке 1:

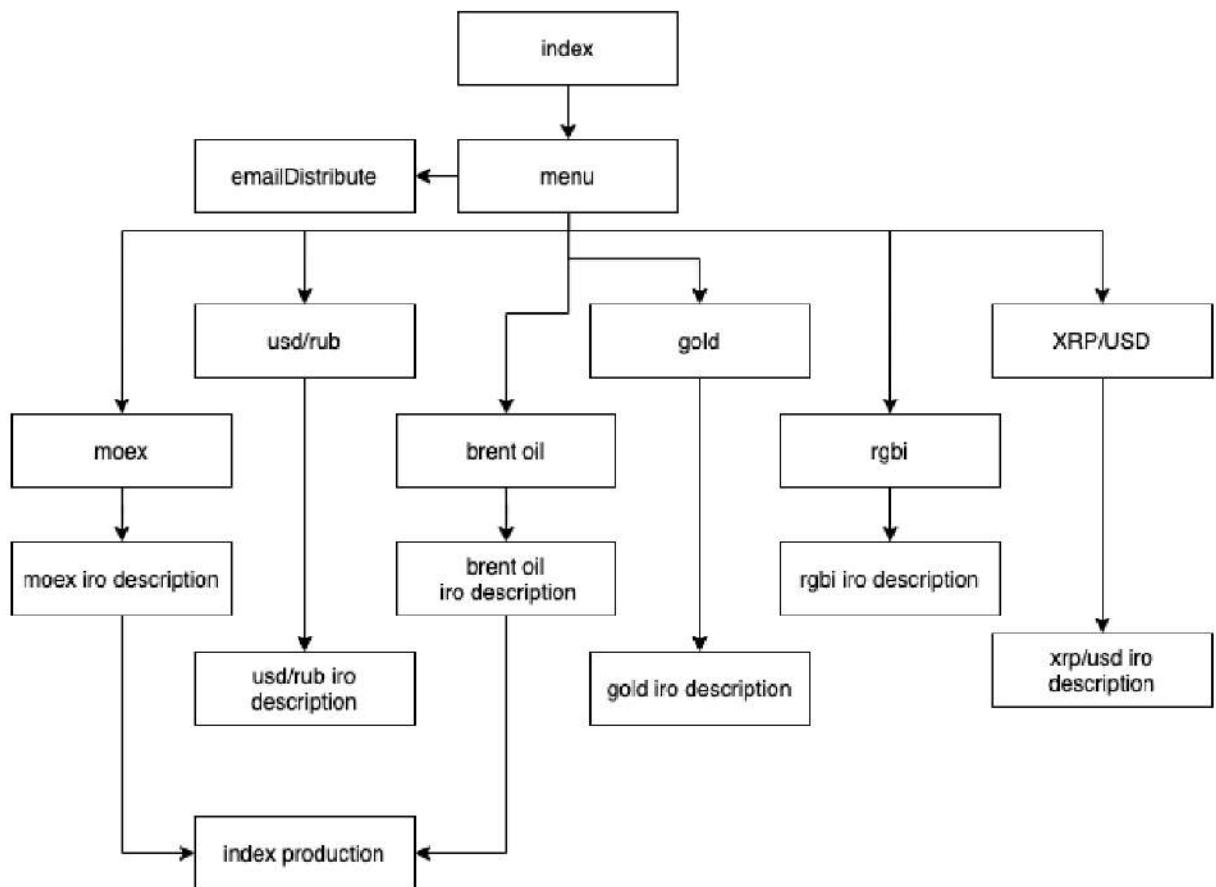


Рисунок 1 – Карта сайта

Далее нужно проработать архитектуру веб-сервера. Основным фреймворком для сайта был выбран Django. В качестве системы управления базами данных была выбрана MySQL. На Рисунке 2 показана архитектура web-сервера:

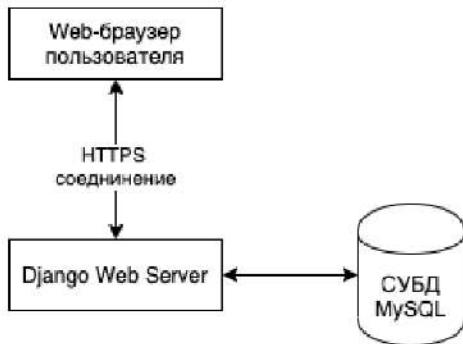


Рисунок 2 – Архитектура

2.3 Проектирование базы данных

Модели в Django описывают структуру используемых данных. Используемые в программе данные хранятся в базах данных, и с помощью моделей как раз осуществляется взаимодействие с базой данных.

По умолчанию Django в качестве базы данных использует SQLite. Она очень проста в использовании и не требует запущенного сервера. Все файлы базы данных могут легко переноситься с одного компьютера на другой. Однако при необходимости мы можем использовать в Django большинство распространенных СУБД. Поскольку MySQL будет более подходящим решением, был выбран именно он.

По сути, MySQL – это специальная программа с открытым кодом, которая используется на сервере SQL. Данная программа не способна обрабатывать большое количество информации, однако она идеальна для небольших и крупных веб-ресурсов. Как и любой продукт, MySQL имеет сильные и слабые стороны.

Преимущества MySQL:

- простоту в работе (установка не требует наличия специальных навыков, для работы может потребоваться дополнительное приложение GUI, которое делает работу с СУБД еще проще);
- богатый функционал;
- безопасность (включает большое количество функций для обеспечения безопасности, причем они поддерживаются по умолчанию);

- масштабируемость (может работать с большими объемами информации);
высокая скорость (увеличить производительность удалось путем упрощения некоторых стандартов).

Недостатки MySQL:

- наличие ограничений функционала (имеет большинство возможностей SQL, но не все, а иногда они требуются для работы в особо «капризных» приложениях);
- платную поддержку даже для бесплатной версии.

Исходя из этих достоинств и недостатков было принято решение использовать именно MySQL.

2.4 Разработка внешнего вида приложения

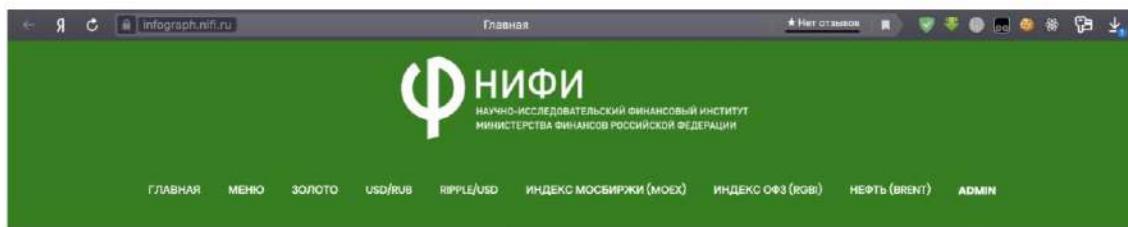
Во время продумывания сайта было решено, что интерфейс должен быть:

- понятен;
- интуитивен;
- прост;
- минималистичен;

Для более удобной разработки, за основу был выбран фреймворк Bootstrap четвертой версии для разметки страницы. В самом Django реализован механизм HTML-template, который позволяет подставлять шаблоны на страницу. Он во многом похож на Jinja2-template, точнее, даже построен на ней.

На главной странице представлена информация об данном сайте, информация об самой организации НИФИ, а также в конце есть контакты руководителя. Сам сайт проприетарный, и свободного доступа к его данным нет.

Главная страница представлена на Рисунке 3.



Обзор условий биржевой торговли на финансовом рынке Российской Федерации: Май 2020

Рассчитываемый НИФИ индикатор ожидаемых условий биржевой торговли на финансовом рынке России (индикатор условий финансирования, ИУФ) основан на результатах ежедневного анализа мнений профессиональных участников рынка о перспективах изменения ценовой конъюнктуры биржевой торговли. В круг компаний входит более 40 участников, представляющих российские и зарубежные банки, инвестиционные и негосударственные пенсионные фонды, консалтинговые фирмы, международные информационные и рейтинговые агентства, экономические ведомства РФ.

Дополнительный индикатор релевантности ожидаемых условий (ИРО) показывает выраженные в весах структурные соотношения между ожиданиями участников рынка и факторами формирования условий рынка, и служит для оценки вероятности следования поведения участников торгов их ожиданиям.

В основе расчетов лежит широко используемая в современной экономике методология PMI – «диффузный индикатор», модифицированный НИФИ для финансового рынка с учетом процентных соотношений между долями в численности повышательных и понижательных оценок с учетом распределения их соотношения в данных об отсутствии изменений в период наблюдений. Показатель индикатора свыше 50,0 пунктов указывает на повышательный тренд, ниже – на понижательный.

Рассчитываемые НИФИ индикаторы основных сегментов финансового рынка (валютного, фондового, золотого, нефтяного и т.д.) используются для отслеживания и определения тенденций в

Рисунок 3 – Главная страница сайта

Для доступа к меню сайта, а также сегментам необходимо авторизоваться. Сама авторизация не находится в свободном доступе, аккаунт можно создать через панель администратора. Это было сделано для того, чтобы закрыть часть уязвимостей, которые могут быть при регистрации со стороны пользователя.

На Рисунке 4 представлено основное меню навигации сайта:

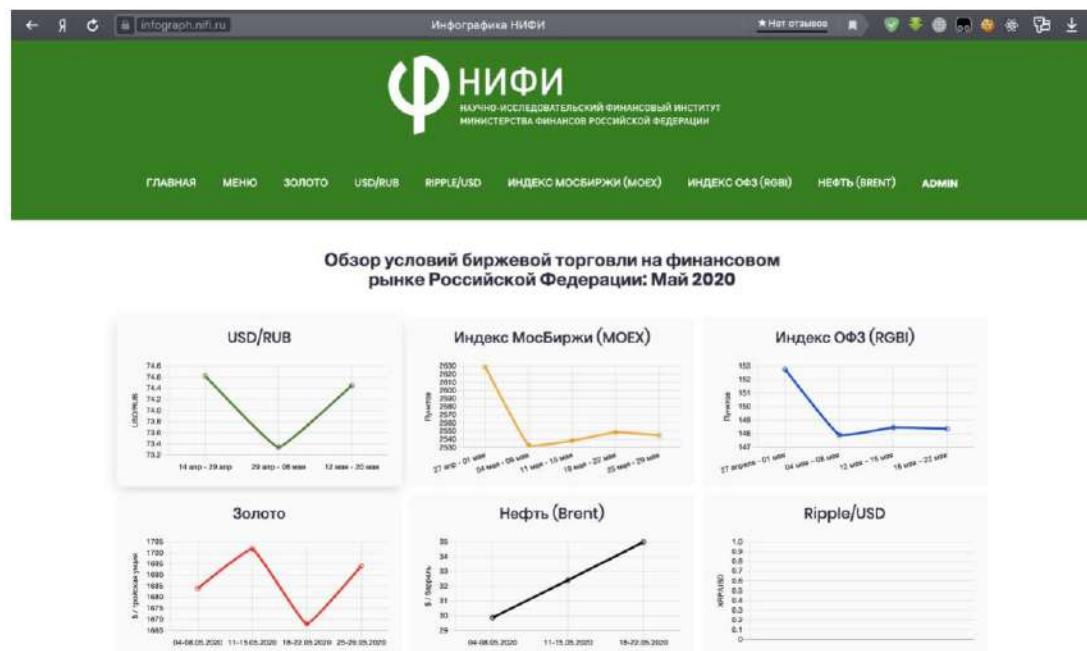


Рисунок 4 – Меню навигации сайта

Само меню представляет из себя шесть плиток, сегментов. Для большей наглядности было решено разместить внутри плиток графики, что бы можно было сразу оценить прогноз на сегмент. При нажатии на плитку происходит переход к самому разделу выбранного сегмента. На Рисунке 5 представлен раздел сегмента по золоту:

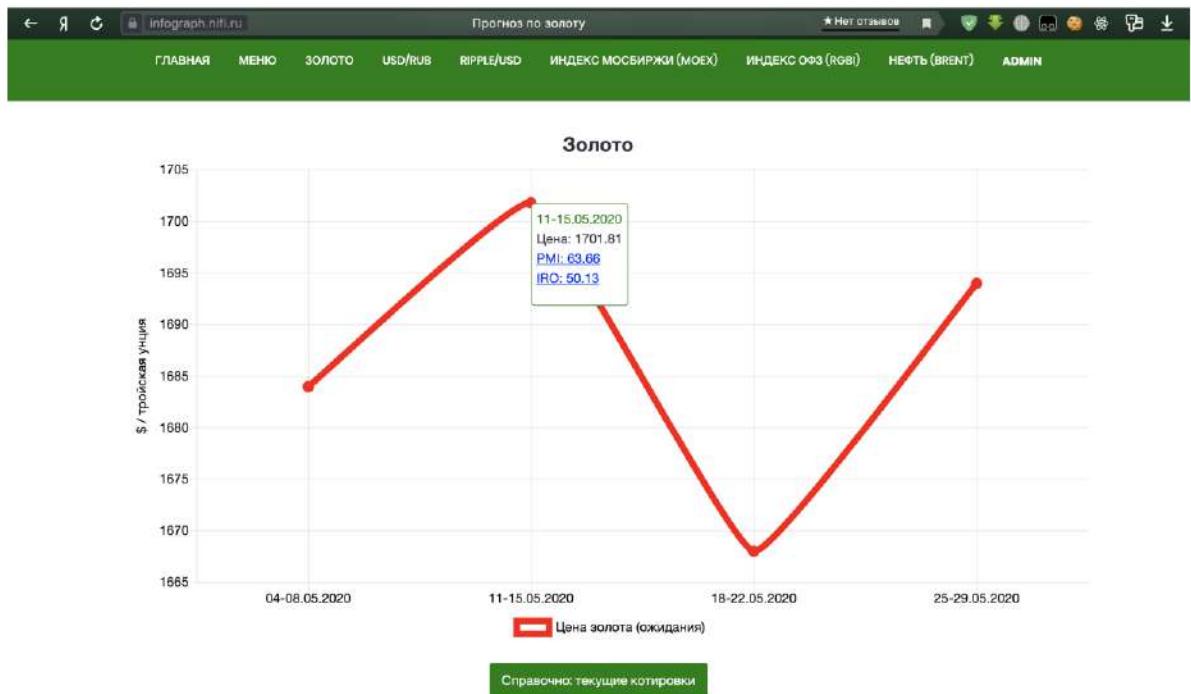


Рисунок 5 – Раздел прогноза по Золоту

Здесь уже более подробно можно рассмотреть график. К тому же добавляется всплывающая подсказка при наведении на точку, в которой есть ссылки на PMI и IRO. При нажатии на PMI, внизу под графиком всплывает его описание, при нажатии на IRO произойдет перенаправление на страницу с финансовыми факторами данного прогноза. Эта страница представлена на Рисунке 6. На данной странице представлено описание IRO на период, в котором содержится информация об релевантности ожидания, а также таблица факторов на период. Сами факторы в таблице кликабельны, и переходят уже на сторонние ресурсы, на основе которых и делался прогноз. Внизу есть 2 кнопки: «Назад» и «Текущие котировки». Аналогичные страницы находятся и в других сегментах.

Описание IRO на период:

Индикатор релевантности ожиданий рынка составил 50,13 л, что указывает на возможность сохранения в целом позитивных оценок профессиональными участниками российского финансового рынка факторов изменения мировой цены золота. Анализ участниками рынка влияния факторов на рынок золота свидетельствует о превалировании повышательных факторов – (0,63), что связывается участниками рынка с ожиданием замедления роста мировой экономики в условиях пандемии коронавируса, сохранением спроса на золото как надежного актива в условиях нестабильности в мировой экономике, продолжением политики денежно-кредитного смягчения центральными банками мира, ослаблением курса доллара. Вес понижательных факторов составил 0,37 л, что связывается рядом участников рынка со снижением спроса на золото, в условиях ослабления карантинных мер странами мира, в виду сокращения числа зараженных коронавирусом, укреплением доллара США.

Факторы на период:		
Факторы	Веса факторов изменения условий рынка золота	
	Повышение	Понижение
Мировая экономика	0,21	0,07
Мировой финансовый рынок	0,25	0,24
Монетарная политика	0,1	0,0
Геополитика	0,07	0,06
Веса факторов	0,63	0,37

[Назад](#) [Текущие котировки](#)

Рисунок 6 – Страница факторов на период

Все данные разделены и у каждого сегмента свои факторы прогноза, таблицы этих факторов прогноза могут различаться.

Теперь можно рассмотреть панель администратора, главная ее часть отображена на Рисунке 7:

Администрирование сайта | Административный сайт Django

Последние действия

- test
 - Индекс производства добьчи нефти и газа
- test1
 - Индекс производства добьчи нефти и газа
- Февраль 2019
 - Индекс производства добьчи нефти и газа
- Март 2019
 - Индекс производства добьчи нефти и газа
- test
 - Индекс производства обр. пром.
- test1
 - Индекс производства обр. пром.
- Май 2020
 - Факторы золота
- Апрель 2020
 - Индекс производства добьчи нефти и газа
- Май 2020
 - Индекс производства обр. пром.
- Апрель 2020
 - Индекс производства обр. пром.

MAIN

Индекс производства добьчи нефти и газа	Добавить Изменить
Индекс производства обр. пром.	Добавить Изменить
Ссылки на индексе промышленности	Добавить Изменить
Текущий месяц на страницах	Добавить Изменить
Точки на Ripple	Добавить Изменить
Точки на золото	Добавить Изменить
Точки на индексе ОФЗ	Добавить Изменить
Точки на индексе мосбиржи	Добавить Изменить
Точки на нефти	Добавить Изменить
Точки на рубле	Добавить Изменить
Факторы Ripple	Добавить Изменить
Факторы золота	Добавить Изменить
Факторы индекса Мосбиржи	Добавить Изменить
Факторы индекса ОФЗ	Добавить Изменить
Факторы нефти	Добавить Изменить
Факторы рубля	Добавить Изменить

ПОЛЬЗОВАТЕЛИ И ГРУППЫ

Группы	Добавить Изменить
Пользователи	Добавить Изменить

Рисунок 7 – Главная страница панели администратора

В блоке «Main» идет работа с самим наполнением сайта. Каждая строка отвечает за добавление данных в какой-либо сегмент. На Рисунке 8 отображен раздел «Точки на Золоте»:

The screenshot shows a Django admin page titled 'Выберите Точки на золоте для изменения | Административный сайт Django'. At the top right, there are links for 'Нет отзывов', 'открыть сайт / изменить пароль / выйти'. Below the title, it says 'ДОБРО ПОЖАЛОВАТЬ, ADMIN'.

The main area is titled 'Выберите Точки на золоте для изменения'. It includes a 'Действия:' dropdown and a 'Выполнить' button. A message 'Выбрано 0 объектов из 4' is displayed. A list of five items is shown, each with a checkbox:

- ТОЧКИ НА ЗОЛОТЕ
- 25-29.05.2020
- 18-22.05.2020
- 11-15.05.2020
- 04-08.05.2020

At the bottom left, it says '4 Точки на золоте'. On the right, there is a blue button labeled 'Добавить Точки на золоте' with a plus sign.

Рисунок 8 – Раздел Точек на золоте

В данном разделе отображены текущие точки на графике, на Рисунке 9 можно их увидеть:

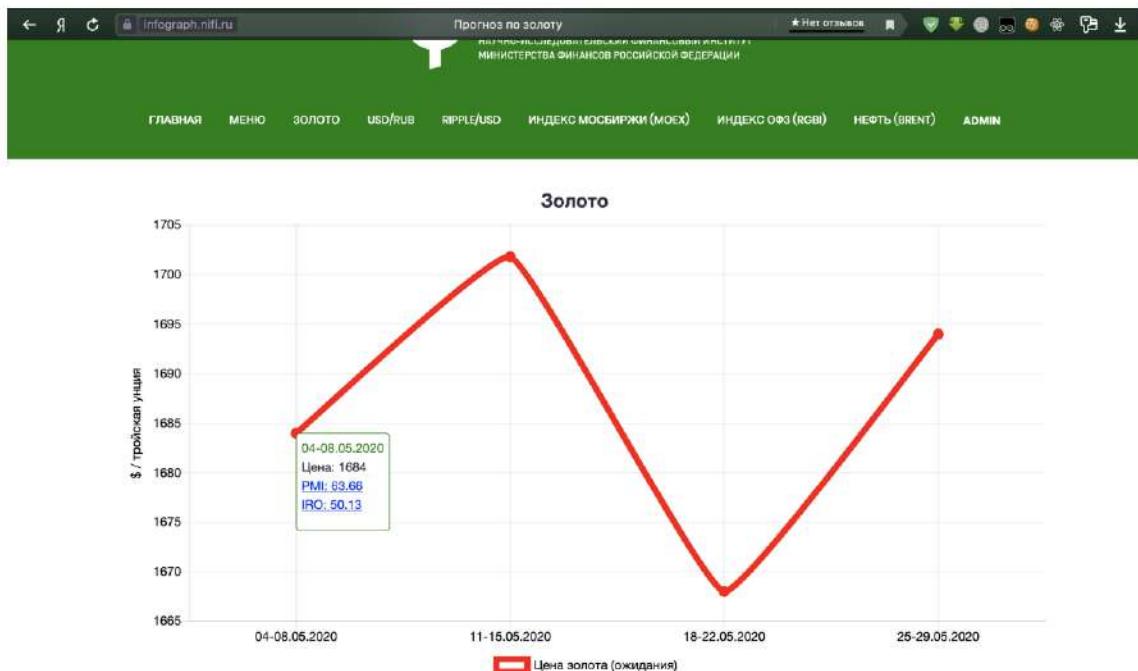
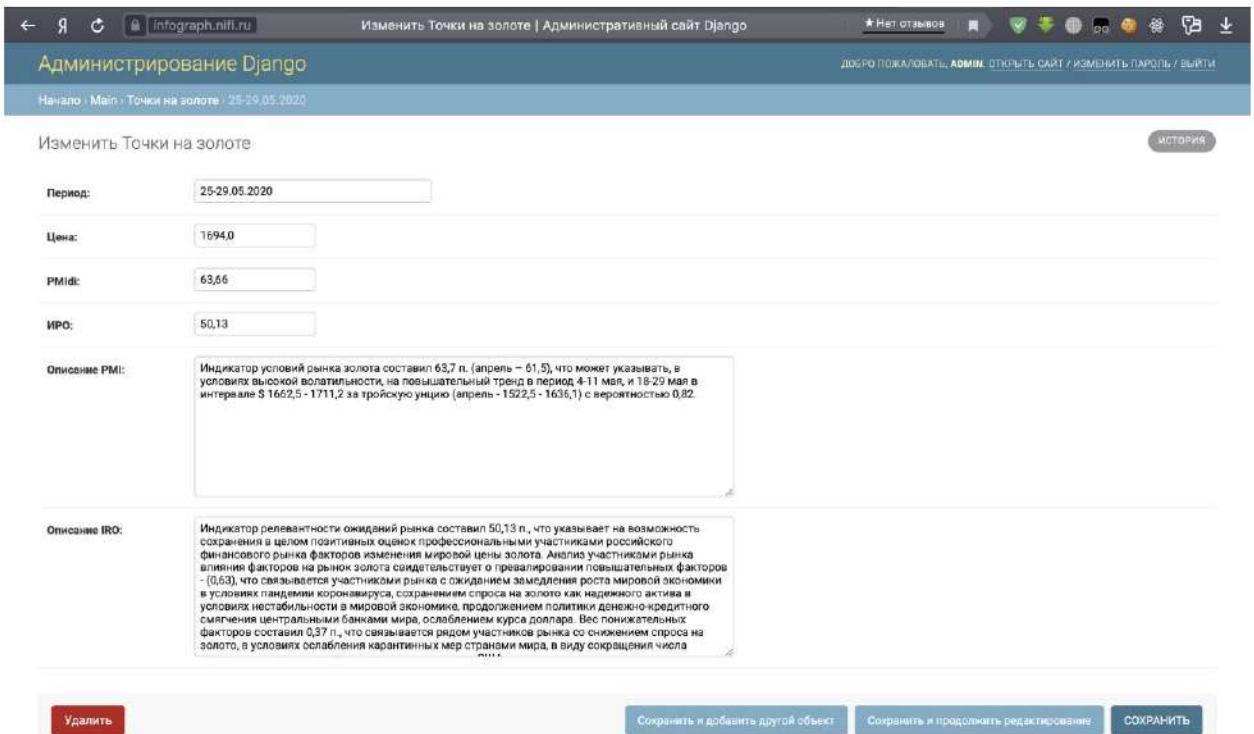


Рисунок 9 – График золота.

Для более удобной ориентировки между точками в админ панели, они выводятся как периоды прогноза. На Рисунке 10 изображена страница конкретной точки в админ-панели:



Изменить Точки на золоте

Период: 25-29.05.2020

Цена: 1694.0

PMid: 63,66

IPO: 50,13

Описание PMI: Индикатор условий рынка золота составил 63,7 л. (апрель – 61,5), что может указывать, в условиях высокой волатильности, на повышательный тренд в период 4-11 мая, и 18-29 мая в интервале \$ 1662,5 - 1711,2 за тройскую унцию (апрель - 1522,5 - 1636,7) с вероятностью 0,82.

Описание IPO: Индикатор релевантности ожиданий рынка составил 50,13 л., что указывает на возможность сохранения в целом позитивных оценок профессиональными участниками российского финансового рынка факторов изменения мировой цены золота. Анализ участников рынка влияния факторов на рынок золота свидетельствует о преувеличении повышательных факторов - (0,63), что связывается участниками рынка с ожиданием замедления роста мировой экономики в условиях пандемии коронавируса, сохранением спроса на золото как надежного актива и условиях нестабильности в мировой экономике, продолжением политики денежно-кредитного смягчения центральными банками мира, ослаблением курса доллара. Всё понижательных факторов составил 0,37 л., что связывается рядом участников рынка со снижением спроса на золото, в условиях ослабления карантинных мер странами мира, в виду сокращения числа

Удалить Сохранить и добавить другой объект Сохранить и продолжить редактирование СОХРАНИТЬ

Рисунок 10 – Конкретная точка золота

Здесь вносятся основные данные об точки, после сохраняются. На Рисунке 11 можно увидеть факторы золота:



Выберите Факторы золота для изменения

Действие: Выполнить Выбрано 0 объектов из 1

ФАКТОРЫ ЗОЛОТА
 Май 2020

1 Факторы золота

Добавить факторы золота +

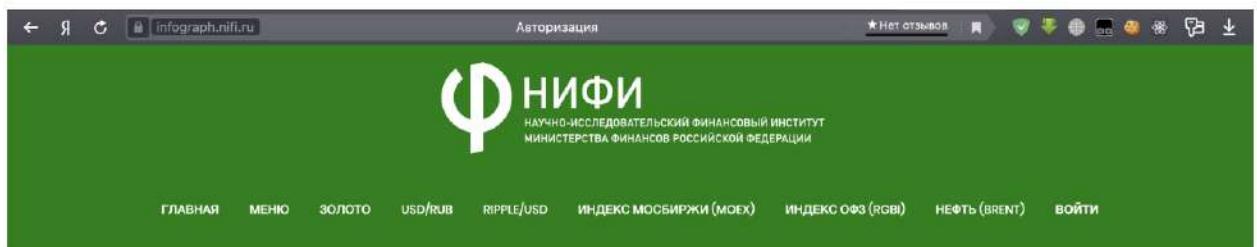
Рисунок 11 – Факторы золота

На Рисунке 12 и 13 изображено само тело фактора:

Рисунок 12 – Тело объекта «Факторы золота»

Рисунок 13 - Тело объекта «Факторы золота»

На Рисунке 14 изображено окно авторизации:



КОНТАКТЫ

Приемная:
Телефон: +7 495 899-74-14
Эл почта:
mail@nifi.ru

Рисунок 14 – Окно авторизации

На Рисунке 15 изображена форма рассылки оповещений по электронной почте:

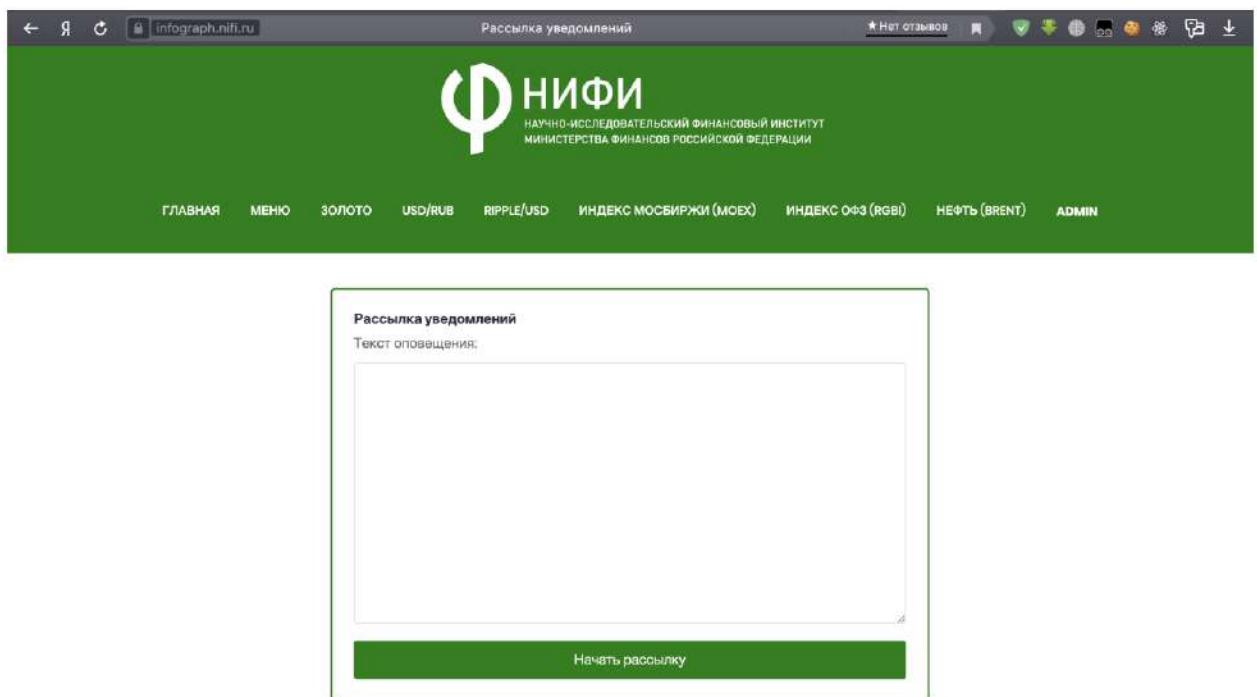


Рисунок 15 - Форма рассылки оповещений по электронной почте:

2.5 Разработка внутренних модулей приложения

Центральным моментом любого веб-приложения является обработка запроса, который отправляет пользователь. В Django за обработку запроса отвечают представления или `views`. По сути представления представляют функции обработки, которые принимают данные запроса в виде объекта `request` и генерируют некоторый результат, который затем отправляется пользователю.

Шаблоны (`template`) отвечают за формирование внешнего вида приложения. Они предоставляют специальный синтаксис, который позволяет внедрять данные в код HTML.

Нередко шаблоны должны иметь одинаковую базовую структуру, одни и те же блоки, при этом определять для отдельных блоков различное содержимое. Это позволяет сформировать единообразный стиль сайта, когда веб-страницы имеют одни и те же структурные элементы - меню, хедер, футер, сайдбары и так далее.

В этом случае мы можем определять все шаблоны по отдельности. Однако если возникнет необходимость изменить какой-то блок, например, добавить в общее меню еще один пункт, тогда придется менять все шаблоны, коих может быть довольно много. И в этом случае оптимальнее повторно использовать один базовый шаблон, который определяет все основные блоки. По умолчанию представления размещаются в приложении в файле `views.py`.

2.6 Тестирование приложения

Тестирование, как заключительный этап разработки веб-сайта, играет жизненно важную роль в процессе создания высококачественного программного обеспечения. Для более удобного проведения тестов, они были разбиты на несколько шагов:

1. Тестирование документации

На данном этапе тестирования, производится сверка реальной функциональности сайта и требуемой функциональностью, исходя из технического задания. Создается план тестирования.

2. Функциональное тестирование

Оно направлено на то, чтобы каждая функция веб-сайта работала в соответствии с требованиями спецификации. Тестирование функциональности веб-сайта показывает «Что делает система».

3. Тестирование удобства использования

Оно предназначено для оценки веб-страницы с точки зрения конечного пользователя. Это помогает определить соответствие продукта ожиданиям пользователей, выявляет проблемные места в интерфейсе.

Однако стоит учесть, что в целом тестирование проводилось скорее модульно, то есть при добавлении нового функционала сразу же проводилось его тестирование. Это позволило упростить выявление ошибок, так как если бы тестирование проводилось в самом конце разработки сайта, то могли бы быть упущены некоторые нюансы, которые в итоге могли бы оказаться критично.

И так, соответственно после описания шагов можно перейти к самому тестированию. На первом шаге был создан общий план тестирования:

- Проверка пользовательского варианта использования
- Проверка административного варианта использования

Пользовательский вариант использования изображен на Рисунке 16

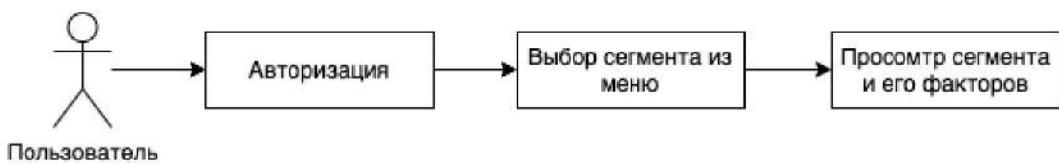


Рисунок 16 – Пользовательский вариант использования

На данном этапе было выявлено меньше всего ошибок. В середине разработки наблюдались проблемы с выдачей данных для графиков из-за проблем с primary key, которая возникала после удаления старых точек. Данная проблема была решена выдачей отдельных индивидуальных значений каждой точке.

Административный вариант использования изображен на Рисунке 17:

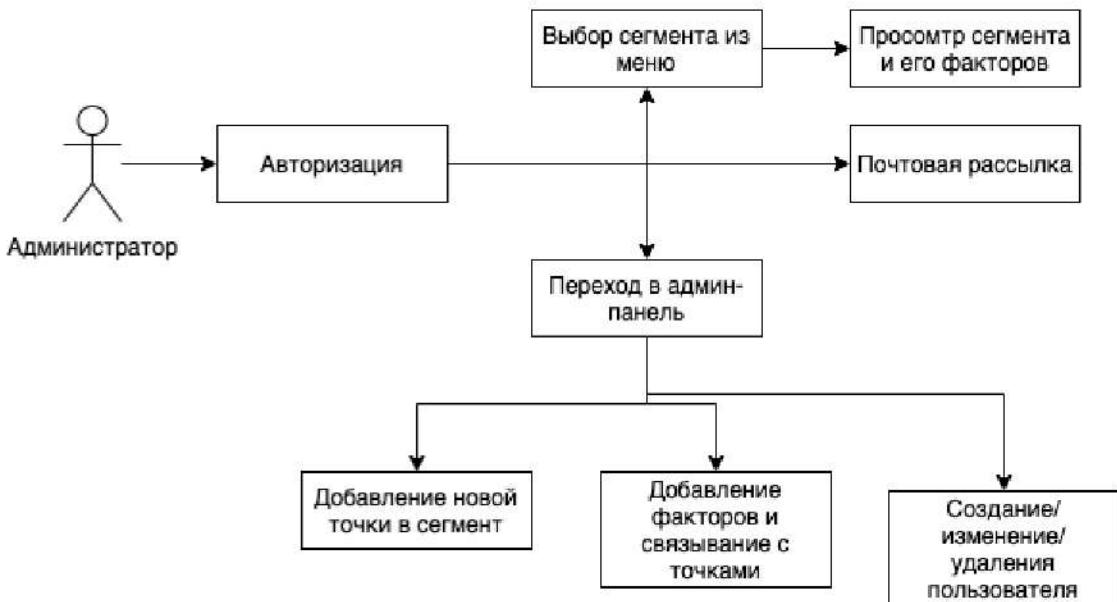


Рисунок 17 - Административный вариант использования

В этом случае ошибок было больше. Наблюдались сбои в обработке новых точек, новых факторов. Однако в итоге все критические и некритические ошибки были исправлены.

2.7 Руководство системного программиста

Для успешного функционирования инфографики необходимо иметь сервер под управлением операционной системы Linux, который имеет следующие минимальные характеристики:

- 1-Core CPU
- 2Gb RAM
- 8Gb ROM
- Широкополосный интернет

Перед запуском сервера необходимо удостовериться, что на нем установлено необходимое программное обеспечение. Рекомендуемая операционная система – Ubuntu Server, версии не ниже 16.04 (рекомендуется 18.04 из-за обновленных репозиториев). На сервере необходимо установить MySQL, желательно не ниже версии 8. Данный продукт работает и с 5-ой версией MySQL, однако перед выполнением миграции в созданную базу будет необходимо прописать ее кодировку в MySQL.

Итак, вы установили MySQL, и сконфигурировали пользователя. Теперь необходимо перейти в папку с проектом, далее в папку nifi_infograph, и там открыть файл settings.py, как показано на Рисунках 18, 19:

```
[root@v31-varjas ~]# ls
root@v31-varjas-3:~# ls
nifi-dump.sql  nifi_infograph  parser  TEMP
root@v31-varjas-3:~# cd nifi_infograph/
root@v31-varjas-3:~/nifi_infograph# ls
main  manage.py  nifi_infograph __pycache__ README.md venv
root@v31-varjas-3:~/nifi_infograph# cd nifi_infograph/
root@v31-varjas-3:~/nifi_infograph/nifi_infograph# ls
__init__.py  __pycache__  settings.py  urls.py  wsgi.py
root@v31-varjas-3:~/nifi_infograph/nifi_infograph# cat settings.py
```

Рисунок 18 – Открытие файла settings.py

```
GNU nano 2.9.3                               settings.py

"""
Django settings for nifi_infograph project.

Generated by 'django-admin startproject' using Django 2.2.3.

For more information on this file, see
https://docs.djangoproject.com/en/2.2/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/2.2/ref/settings/
"""

import os

# Build paths inside the project like this: os.path.join(BASE_DIR, ...)
BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))

# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/2.2/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = [REDACTED]

# SECURITY WARNING: don't run with debug turned on in production!

ALLOWED_HOSTS = [REDACTED]

# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
]

[REDACTED]
[REDACTED] Read 128 lines [REDACTED]
[REDACTED] Get Help  [REDACTED] Write Out  [REDACTED] Where Is  [REDACTED] Cut Text  [REDACTED] Justify  [REDACTED] Cur Pos  [REDACTED] Undo
[REDACTED] Exit   [REDACTED] Read File [REDACTED] Replace [REDACTED] Uncut Text [REDACTED] To Linter [REDACTED] Go To Line [REDACTED] Redo
[REDACTED] Mark Text [REDACTED] To Bracket [REDACTED] Copy Text [REDACTED] WhereIs Next [REDACTED]
```

Рисунок 19 – Сам файл settings.py

В файле settings.py находим блок с логином и паролем от MySQL, и вставляем установленный вами логин и пароль от пользователя MySQL, а так же вводим название database, как показано на Рисунке 20:

```
GNU nano 2.9.3                               settings.py

# Database
# https://docs.djangoproject.com/en/2.2/ref/settings/#databases

DATABASES = [
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'HOST': '127.0.0.1',
        'NAME': '[REDACTED]',
        'USER': '[REDACTED]',
        'PASSWORD': '[REDACTED]',
    },
]
```

Рисунок 20 – Меняем настройки MySQL

Далее идем в MySQL и создаем там базу данных с тем же названием, который мы указали в файле настроек сайта, как показано на Рисунке 21:

```
[sudo] password for varjas:  
root@v31-varjas-3:~# mysql -u root -p  
Enter password:  
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: YES)  
root@v31-varjas-3:~# mysql -u root -p  
Enter password:  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 4703  
Server version: 8.0.18 MySQL Community Server - GPL  
  
Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
mysql> ■
```

Рисунок 21 – Вход в MySQL

Далее в нем создаем базу данных, сохраняем ее, и выходим. Теперь возвращаемся в корневую папку с проектом, необходимо активировать виртуальное окружение Python 3.

На Рисунке 22 показан процесс активации виртуального окружения:

```
[sudo] password for varjas:  
root@v31-varjas-3:~# mysql -u root -p  
Enter password:  
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: YES)  
root@v31-varjas-3:~# mysql -u root -p  
Enter password:  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 4703  
Server version: 8.0.18 MySQL Community Server - GPL  
  
Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql> exirt  
->  
-> ;  
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'exirt' at line 1  
mysql> exit  
Bye  
root@v31-varjas-3:~# cd /home/v  
varjas/ vromanov/  
root@v31-varjas-3:~# cd /home/v  
varjas/ vromanov/  
root@v31-varjas-3:~# cd /home/varjas/  
.config/ .gnupg/ nifi_infograph/ parcer/ TEMP/  
root@v31-varjas-3:~# cd /home/varjas/  
.config/ .gnupg/ nifi_infograph/ parcer/ TEMP/  
root@v31-varjas-3:~# cd /home/varjas/  
.config/ .gnupg/ nifi_infograph/ parcer/ TEMP/  
root@v31-varjas-3:~# cd /home/varjas/nifi_infograph/  
root@v31-varjas-3:~/nifi_infograph# ls  
main manage.py nifi_infograph __pycache__ README.md venv  
root@v31-varjas-3:~/nifi_infograph# source venv/bin/activate  
(venv) root@v31-varjas-3:~/nifi_infograph# ■
```

Рисунок 22 – Активация виртуального окружения

Далее необходимо провести миграции в базу данных, как это показано на Рисунке 23:

```

Your MySQL connection id is 4703
Server version: 8.0.18 MySQL Community Server - GPL

Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> exit
->
->;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'exit' at line 1
mysql> exit
Bye
root@v31-varjas-3:~# cd /home/v
varjas/ vromanov/
root@v31-varjas-3:~# cd /home/v
varjas/ vromanov/
root@v31-varjas-3:~# cd /home/varjas/
.config/ .gnupg/ nifi_infograph/ parcer/ TEMP/
root@v31-varjas-3:~# cd /home/varjas/
.config/ .gnupg/ nifi_infograph/ parcer/ TEMP/
root@v31-varjas-3:~# cd /home/varjas/
.config/ .gnupg/ nifi_infograph/ parcer/ TEMP/
root@v31-varjas-3:~# cd /home/varjas/nifi_infograph/
main manage.py nifi_infograph __pycache__ README.md venv
root@v31-varjas-3:~/nifi_infograph# source venv/bin/activate
(venv) root@v31-varjas-3:~/nifi_infograph# python3 manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, main, sessions
Running migrations:
  No migrations to apply.
  Your models have changes that are not yet reflected in a migration, and so won't be applied.
  Run 'manage.py makemigrations' to make new migrations, and then re-run 'manage.py migrate' to apply them.
(venv) root@v31-varjas-3:~/nifi_infograph#
```

Рисунок 23 – Миграции в базу данных

В вашем случае вы увидите список проведенных миграций (их 17 штук). В случае чистой установки, необходимо заранее создать профиль администратора, команной, которая показана на Рисунке 24:

```

->
->;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'exit' at line 1
mysql> exit
Bye
root@v31-varjas-3:~# cd /home/v
varjas/ vromanov/
root@v31-varjas-3:~# cd /home/v
varjas/ vromanov/
root@v31-varjas-3:~# cd /home/varjas/
.config/ .gnupg/ nifi_infograph/ parcer/ TEMP/
root@v31-varjas-3:~# cd /home/varjas/
.config/ .gnupg/ nifi_infograph/ parcer/ TEMP/
root@v31-varjas-3:~# cd /home/varjas/
.config/ .gnupg/ nifi_infograph/ parcer/ TEMP/
root@v31-varjas-3:~# cd /home/varjas/nifi_infograph/
root@v31-varjas-3:~/nifi_infograph# ls
main manage.py nifi_infograph __pycache__ README.md venv
root@v31-varjas-3:~/nifi_infograph# source venv/bin/activate
(venv) root@v31-varjas-3:~/nifi_infograph# python3 manage.py createsuperuser
```

Рисунок 24 – Команда создания суперпользователя (администратора)

После всех операций, осталось только одно – запустить сайт. Запустить его необходимо на порту 80, как это показано на Рисунке 25:

```

root@v31-varjas-3:~/nifi_infograph# source venv/bin/activate
(venv) root@v31-varjas-3:~/nifi_infograph# python3 manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, main, sessions
Running migrations:
  No migrations to apply.
    Your models have changes that are not yet reflected in a migration, and so won't be applied.
    Run 'manage.py makemigrations' to make new migrations, and then re-run 'manage.py migrate' to apply them.
(venv) root@v31-varjas-3:~/nifi_infograph# python3 manage.py runserver 0.0.0.0:80
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
May 18, 2020 - 04:46:11
Django version 3.0, using settings 'nifi_infograph.settings'
Starting development server at http://0.0.0.0:80/
Quit the server with CONTROL-C.
Error: That port is already in use.
(venv) root@v31-varjas-3:~/nifi_infograph# 

```

Рисунок 25 – Запуск сервера

В данном случае сервер уже запущен, поэтому выводится ошибка о том, что порт занят. В вашем же случае, если вылезает такая ошибка, необходимо отключить Apache или Ngnix. После всех проведенных процедур сайт запущен, его процесс необходимо отправить в бэкграунд командной disown %1.

2.8 Руководство программиста

Данный проект написан на языке Python 3 с использованием фреймворка Django. Данный проект имеет стандартную структуру: типичную для Django-проекта. Она показана на Рисунке 26:

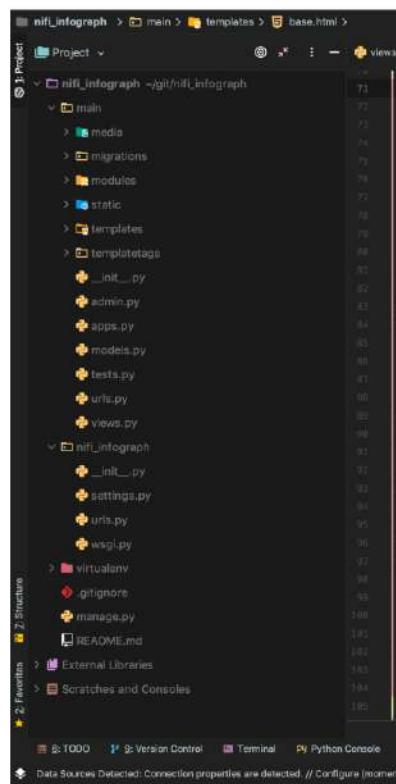
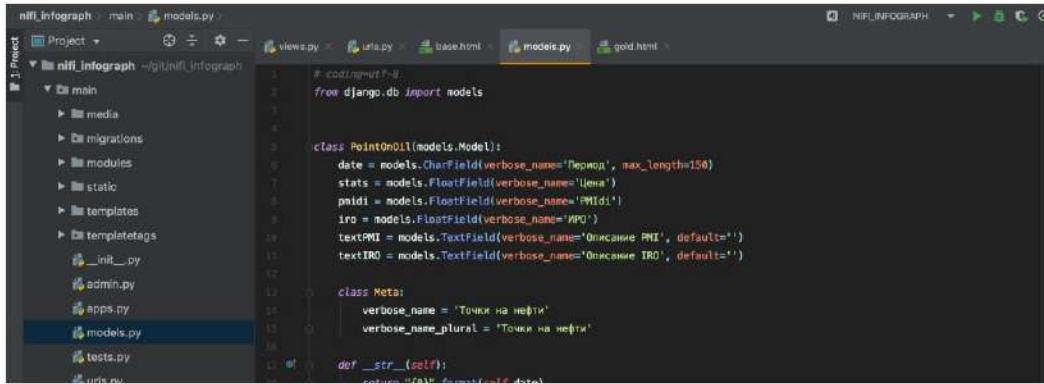


Рисунок 26 – Справа видна стандартная структура приложения

Для начала перейдем к MVC. Модели располагаются по пути /main/models.py, как показано на Рисунке 27:



```
# coding=utf-8
from django.db import models

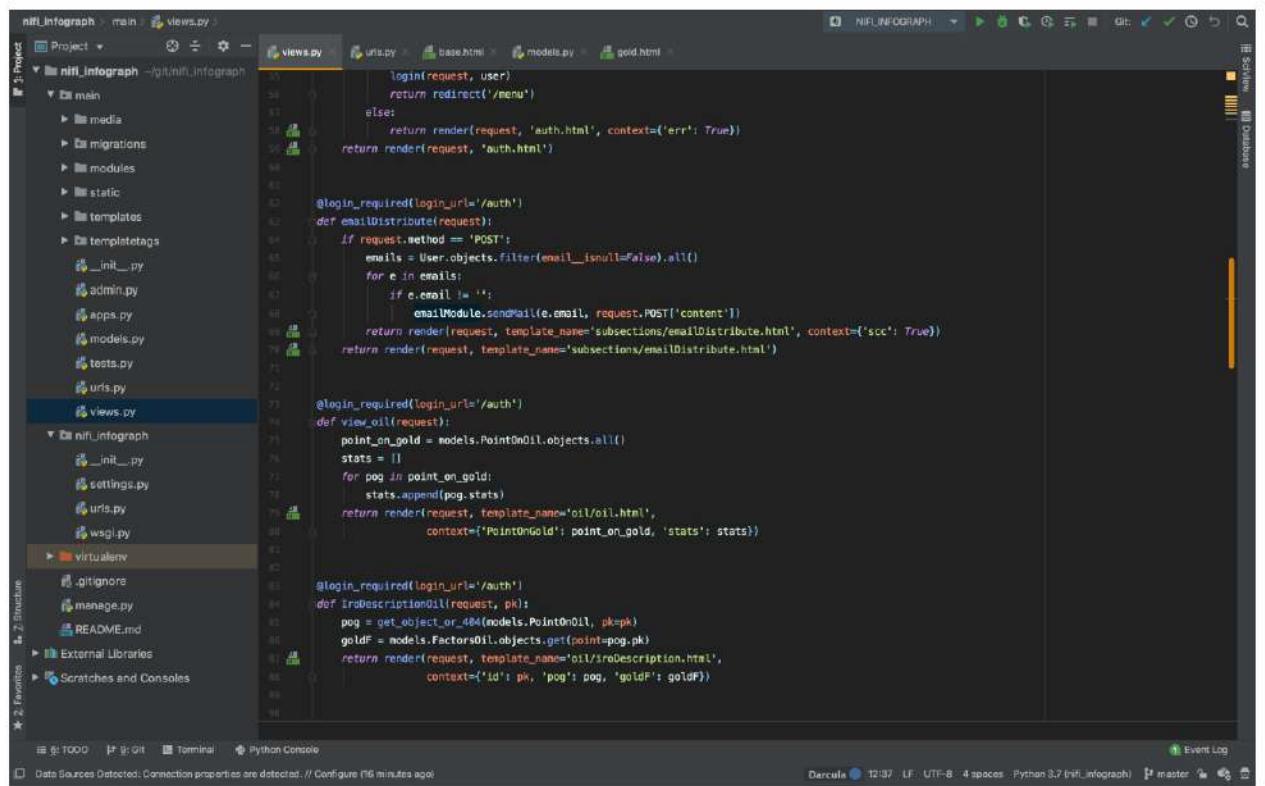
class PointOnOil(models.Model):
    date = models.CharField(verbose_name='Период', max_length=150)
    stats = models.FloatField(verbose_name='Цена')
    pmidi = models.FloatField(verbose_name='PMidi')
    iro = models.FloatField(verbose_name='IRO')
    textPMI = models.TextField(verbose_name='Описание PMI', default='')
    textIRO = models.TextField(verbose_name='Описание IRO', default='')

    class Meta:
        verbose_name = 'Точки на нефти'
        verbose_name_plural = 'Точки на нефти'

    def __str__(self):
        return f'{self.date} - {self.stats}
```

Рисунок 27 – Файл models.py

Здесь хранятся все необходимые модели в базах данных. Далее перейдем к вьюшкам. Они располагаются в этой же директории, в файле views.py. На Рисунке 28 можно это увидеть:



```
def login(request, user):
    return redirect('/menu')
else:
    return render(request, 'auth.html', context={'err': True})

@login_required(login_url='/auth')
def emailDistribute(request):
    if request.method == 'POST':
        emails = User.objects.filter(email__isnull=False).all()
        for e in emails:
            if e.email != '':
                emailModule.sendEmail(e.email, request.POST['content'])
    return render(request, template_name='subsections/emailDistribute.html', context={'scc': True})
return render(request, template_name='subsections/emailDistribute.html')

@login_required(login_url='/auth')
def view_oil(request):
    point_on_gold = models.PointOnOil.objects.all()
    stats = []
    for pog in point_on_gold:
        stats.append(pog.stats)
    return render(request, template_name='oil/oil.html',
                 context={'PointOnGold': point_on_gold, 'stats': stats})

@login_required(login_url='/auth')
def iroDescriptionOil(request, pk):
    pog = get_object_or_404(models.PointOnOil, pk=pk)
    goldF = models.FactorsOil.objects.get(point=pog.pk)
    return render(request, template_name='oil/iroDescription.html',
                 context={'id': pk, 'pog': pog, 'goldF': goldF})
```

Рисунок 28 – файл views.py

В этом файле располагаются функции, отвечающие за вывод страницы. Здесь происходит прием данных от БД, их обработка, и вывод на страницу к пользователю. Последнее, что нужно отметить это контроллер. В данном

фреймворке контроллер — это файл urls.py в этой же директории. Он представлен на Рисунке 29:

The screenshot shows the PyCharm IDE interface with the file `urls.py` open in the center editor window. The code defines a Django URL pattern for the `main` application, mapping various URLs to specific views like `index`, `menu`, `gold`, `auth`, etc. The left sidebar shows the project structure with files like `views.py`, `migrations`, `models.py`, and `tests.py`. The bottom status bar indicates the Python version is 3.7 and the current file is `urls.py`.

```
from django.urls import path
from main import views
from django.conf.urls import url

app_name = 'main'
urlpatterns = [
    path('', views.index, name='index'),
    path('menu', views.menu, name='menu'),
    path('gold', views.view_gold, name='gold'),
    path('auth/', views.auth, name='auth'),
    path('logout/', views.log_out, name='log_out'),
    url(r'^gold/iroDescription/(?P<pk>\d+)/$', views.IroDescriptionGold, name='iroDescriptionGold'),
    url(r'^gold/reviewGold/$', views.reviewGold, name='reviewGold'),
    path('rouble', views.view_rouble, name='rouble'),
    url(r'^rouble/iroDescription/(?P<pk>\d+)/$', views.IroDescriptionRouble, name='iroDescriptionRouble'),
    path('ripple', views.view_ripple, name='ripple'),
    url(r'^ripple/iroDescription/(?P<pk>\d+)/$', views.IroDescriptionRipple, name='iroDescriptionRipple'),
    path('indexMosB', views.view_indexMosB, name='indexMosB'),
    url(r'^indexMosB/iroDescription/(?P<pk>\d+)/$', views.IroDescriptionIndexMosB, name='iroDescriptionIndexMosB'),
    path('indexOFZ', views.view_indexOFZ, name='indexOFZ'),
    url(r'^indexOFZ/iroDescription/(?P<pk>\d+)/$', views.IroDescriptionIndexOFZ, name='iroDescriptionIndexOFZ'),
    path('oil', views.view_oil, name='oil'),
    url(r'^oil/iroDescription/(?P<pk>\d+)/$', views.IroDescriptionOil, name='iroDescriptionOil'),
    url('subsections/indexProduction', views.indexProduction, name='indexProduction'),
    url('subsections/emailDistribute', views.emailDistribute, name='emailDistribute'),
    # url('subsections/review', views.review, name='review'),
    # path('test', views.test, name='test'),
]
```

Рисунок 29 – файл urls.py

В контроллере прописываются пути страниц сайта, функция, которая отвечает за обработку этого пути (вьюшка) и имя данного пути.

В папке static находятся вся статика, точнее js, css, и другие файлы. В папке templates находятся шаблоны страниц, как показано на Рисунке 30:

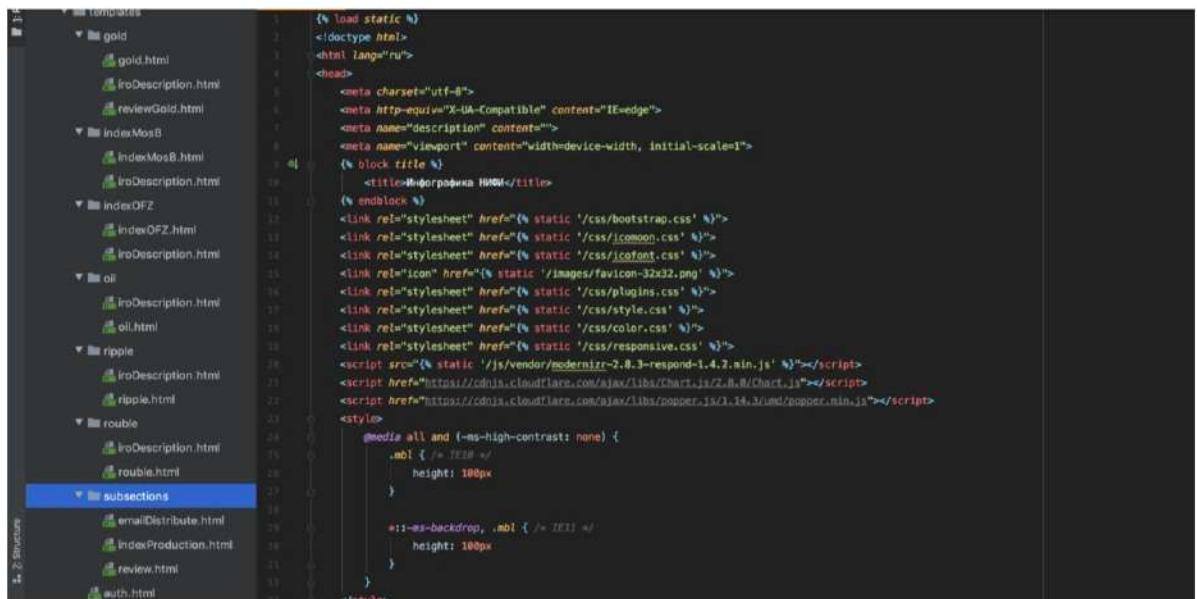


Рисунок 30 – Папка с шаблонами страниц

Нужно отметить, что основные шаблоны лежат в корневой папке, а шаблоны конкретных разделов – в папках по этим разделам. Шаблоны страниц подразделов – в папке subsections.

Теперь перейдем к разделу администратора. Что бы зайти в админку, необходимо что бы был инициализирован superuser. Затем переходим по такому адресу <https://<ваш домен>/admin>, и увидим окно авторизации, как на Рисунке 31:

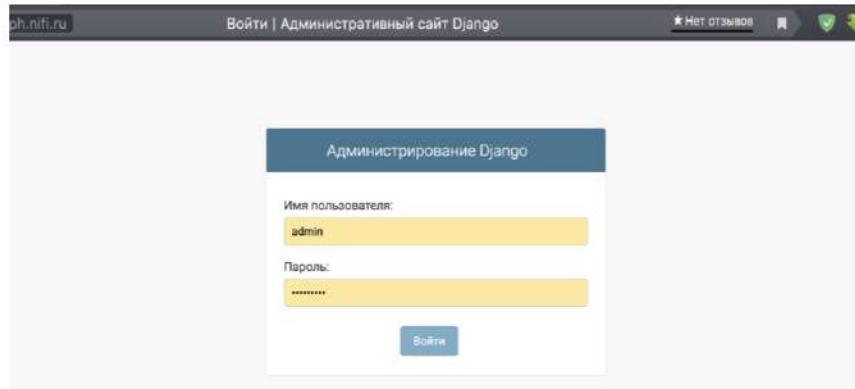


Рисунок 31 – Окно авторизации в админке

Вводим логин и пароль, который был указан при инициализации superuser, входим, и оказываемся на главной странице администраторской панели. Она показана на Рисунке 32:

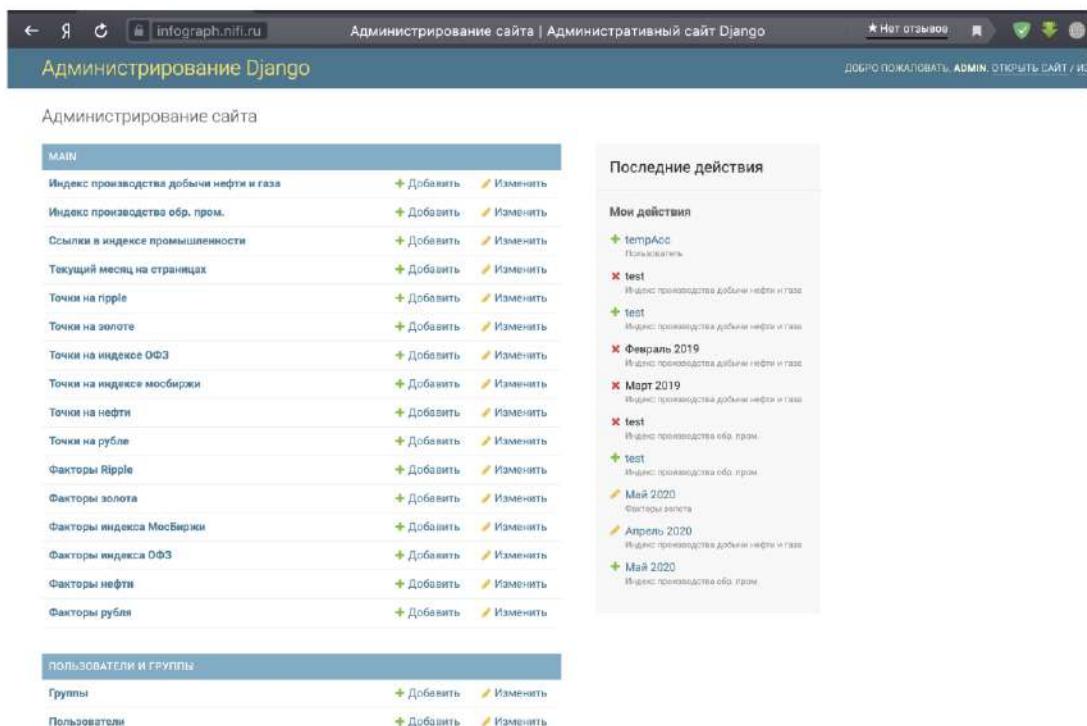


Рисунок 32 – Главная страница админки

Здесь все понятно – в разделе main находятся все объекты, которые показываются на сайте, в разделе ниже находится управление пользователями. Поскольку данный аккаунт является superuser, то у него самые высокие права. Именно с этого аккаунта и нужно добавлять других администраторов (субадминистраторов). Перед этим нужно создать специальную группу прав для них. Для этого переходим в «Группы» в разделе пользователи и группы, и нажимаем кнопку «Добавить группы» в верхнем правом углу, затем открывается страница создания группы, как показано на Рисунке 33:

Рисунок 33 – Страница создания группы

Добавляем нужные нам права, прописываем название группы и сохраняем. Далее необходимо вернуться на главную страницу, затем в том же разделе нажать «Пользователи», на новой странице в верхнем правом углу нажать «Добавить пользователя», откроется окно как на Рисунке 34:

Добавить пользователя

First, enter a username and password. Then, you'll be able to edit more user options.

Имя пользователя: Обязательное поле. Не более 150 символов. Только буквы, цифры и символы @/./+/_-

Пароль: Ваш пароль не может быть слишком похож на другие персональные данные.
Ваш пароль должен содержать как минимум 8 символов.
Ваш пароль не может быть слишком распространенным.
Ваш пароль не может быть полностью числовым.

Подтверждение пароля: Для подтверждения введите, пожалуйста, пароль еще раз.

Сохранить и добавить другой объект **Сохранить и продолжить редактирование** **СОХРАНИТЬ**

Рисунок 34 – Страница создания пользователя

Нужно отметить, что внешней регистрации на портале нет. Все пользователи исключительно создаются в админке. После создания пользователя, нужно вернуться на страницу назад. Он появится в списке. Сверху, кстати, есть поиск по пользователем. Итак, нажимаем на созданного пользователя, и открывается страница редактирования пользователя, как на Рисунке 35, 36 и 37:

Изменить пользователя

Имя пользователя: Nazarov
Обязательное поле. Не более 150 символов. Только буквы, цифры и символы @/./+/_-

Пароль: алгоритм: pbkdf2_sha256 итерации: 180000 соль: LNI7JM***** хэш: потЕК*****
New passwords are not stored; so there is no way to see this user's password, but you can change the password using this form.

Персональная информация

Имя:
Фамилия:
Адрес электронной почты:

Права доступа

Активный
Установите эту отметку, если пользователю должны считаться активными. Уберите эту отметку вместо удаления учётной записи.

Статус персонала
Установите, если пользователь может входить в административную часть сайта.

Статус суперпользователя
Указывает, что пользователь имеет все права для выполнения назначений.

Группы: Доступные группы: Выбранные группы: SubAdmins
Фильтр:

Рисунок 35 – Страница редактирования пользователя

Рисунок 36 – Страница редактирования пользователя

Рисунок 37 - Страница редактирования пользователя

На данной странице можно изменять логин пользователя, пароль. Можно добавить персональную информацию, настроить права доступа, группы и конкретные права пользователя. Так же есть поля с датой и временем регистрации и последним входом.

Теперь в группах добавляете пользователю созданную группу и ставите галочку напротив статуса персонала, как на Рисунке 38:

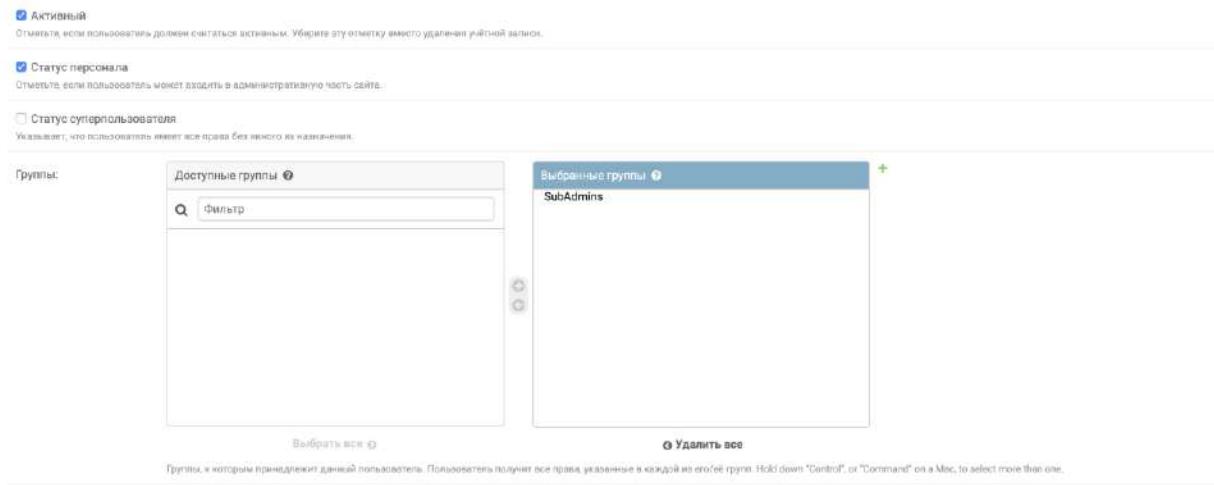


Рисунок 38 - Группы

Внизу нажимаем на кнопку «Сохранить». Вы создали суб-администратора. У данного администратора будет возможность производить изменение данных, делать рассылку, но вот пользователей он изменять не сможет.

2.9 Руководство администратора

Основной функцией администратора является добавление новых данных и рассылка уведомлений. Начнем с добавления данных. Все данные графиков, таблиц и т.д. расположены в блоке main, это показано на Рисунке 39:

Администрирование сайта

Последние действия

Мои действия
Март 2020 Индекс производства добычи нефти и газа
Февраль 2020 Индекс производства добычи нефти и газа
Февраль 2020 Индекс производства добычи нефти и газа
Январь 2020 Индекс производства добычи нефти и газа
Декабрь 2019 Индекс производства добычи нефти и газа
Декабрь 2019 Индекс производства добычи нефти и газа
Март 2020 Факторы нефти
Февраль 2020 Факторы нефти
23.07.03.2020 Факторы нефти
16.10.03.2020 Факторы нефти

Рисунок 39 – Вид админки обычного администратора

Все данные подписаны довольно понятно. Например, нам нужно добавить новый прогноз по золоту. Сначала переходим в Точки на Золоте, и там уже добавляем точку. Откроется страница, как показано на Рисунке 40:

The screenshot shows a Django administrative page titled 'Добавить Точки на золоте | Административный сайт Django'. The page has several input fields: 'Период:' (Period), 'Цена:' (Price), 'PMid:' (PMid), 'ИРО:' (IRO), 'Описание PMI:' (Description PMI), and 'Описание IRO:' (Description IRO). At the bottom are three buttons: 'Сохранить и добавить другой объект' (Save and add another object), 'Сохранить и продолжить редактирование' (Save and continue editing), and a large blue 'СОХРАНИТЬ' (Save) button.

Рисунок 40 – Добавление новой точки

Вводим все необходимые данные, и сохраняем. Так делаем с каждой точкой, которую надо будет добавить. В итоге получится результат, показанный на Рисунке 41:

The screenshot shows a Django administrative page titled 'Выберите Точки на золоте для изменения | Административный сайт Django'. It displays a list of four gold points: 'точки на золоте', '25-29.05.2020', '18-22.05.2020', '11-15.05.2020', and '04-08.05.2020'. A large blue '+' button is visible at the top right.

Рисунок 41 – Точки на золоте

Отлично. Эти точки уже появились на графике, однако при попытке перехода на IRO будет происходить ошибка, т.к. факторы точек не были добавлены и связаны с точками. Для добавления факторов, из главного меню переходим в

Факторы на Золоте, там создаем объект. Важно – фактор на месяц всегда один.

Страница создания фактора показана на Рисунке 42 и 43:

The screenshot shows the 'Добавить Факторы золота' (Add Gold Factors) page in the Django Admin interface. The top navigation bar includes links for 'Начало / Main : Факторы золота : Добавить Факторы золота'. The main form has a field 'Месяц и год, к которым относятся факторы:' with a dropdown menu showing '04-08.05.2020', '11-15.05.2020', '18-22.05.2020', and '25-29.05.2020'. Below this is a note: 'Hold down "Control", or "Command" on a Mac, to select more than one.' A section titled 'Прикреплен к точкам:' contains a list of four items: '04-08.05.2020', '11-15.05.2020', '18-22.05.2020', and '25-29.05.2020', each with a small green plus sign icon. There are also several text input fields for various factors like 'Ссылка на Мировую Экономику:', 'Повышение в Мировой Экономике:', etc., each preceded by a small 'g' icon.

Рисунок 42 – Страница создания факторов

This screenshot shows the same 'Добавить Факторы золота' (Add Gold Factors) page, but with more fields visible. It includes sections for 'Ссылка на Мировой финансовый рынок:', 'Повышение в Мировом финансдовом рынке:', 'Понижение в Мировом финансдовом рынке:', 'Ссылка на Монетарную Политику:', 'Повышение в Монетарной Политике:', 'Понижение в Монетарной Политике:', 'Ссылка на Геополитику:', 'Повышение в Геополитике:', 'Понижение в Геополитике:', 'Повышение в Весах факторов:', and 'Понижение в Весах факторов:'. At the bottom right are three buttons: 'Сохранить и добавить другой объект' (Save and add another object), 'Сохранить и продолжить редактирование' (Save and continue editing), and 'СОХРАНИТЬ' (Save).

Рисунок 43 - Страница создания факторов

Здесь очень важно – после заполнения необходимо отметить точки, которые в окошке под названием «Прикреплен к точкам». Там есть подсказка – «Hold

down “Control”, or “Command” on a Mac, to select more than one», т.е. зажмите клавишу Ctrl или Cmd на Mac, чтобы выбрать больше одного элемента. Делаем так и выбираем несколько элементов, как показано на Рисунке 44:

Рисунок 44 – Выбор прикреплённых точек

Если необходимые точки после выбора помечены серым, значит они прикреплены к фактору. Не забываем сохранить все, и теперь можем переходить к сайту. Наверху есть кнопка – Открыть сайт, на нее и кликаем. Откроется главная страница, как на Рисунке 45:

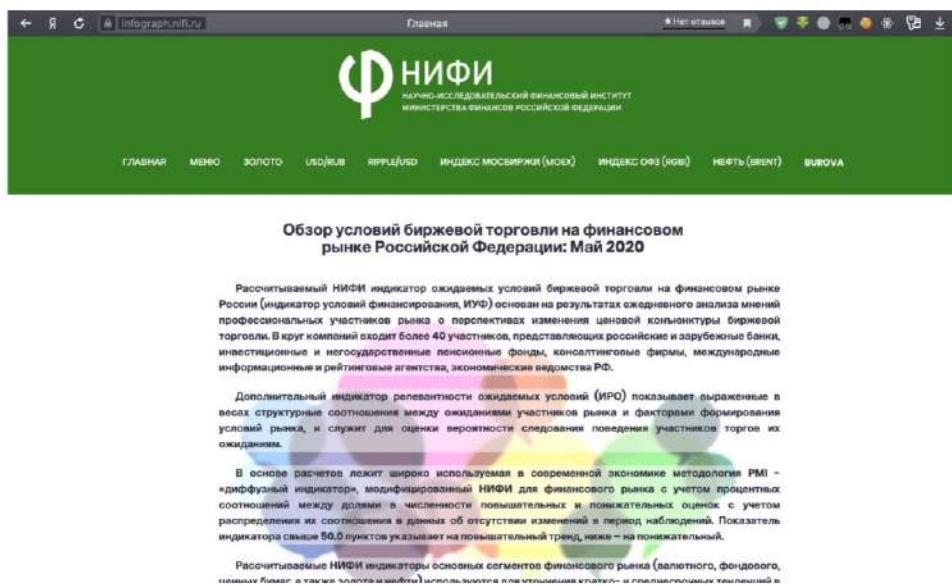


Рисунок 45 – Главная страница сайта

В верхнем навигационном баре нажимаем Меню, и переходим к нему, как на Рисунке 46:

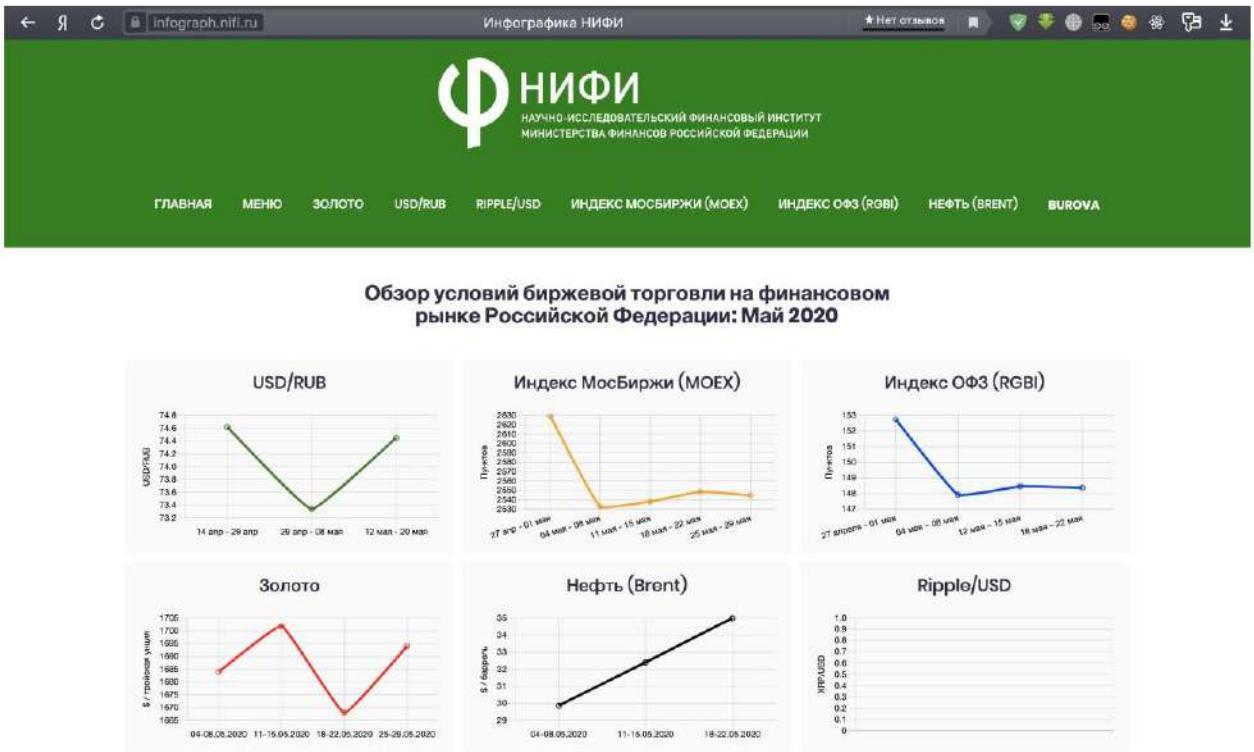


Рисунок 46 – Меню портала

Как можно заметить по меню – точки уже загружены. Можно перейти и проверить факторы, как показано на Рисунках 47 и 48:

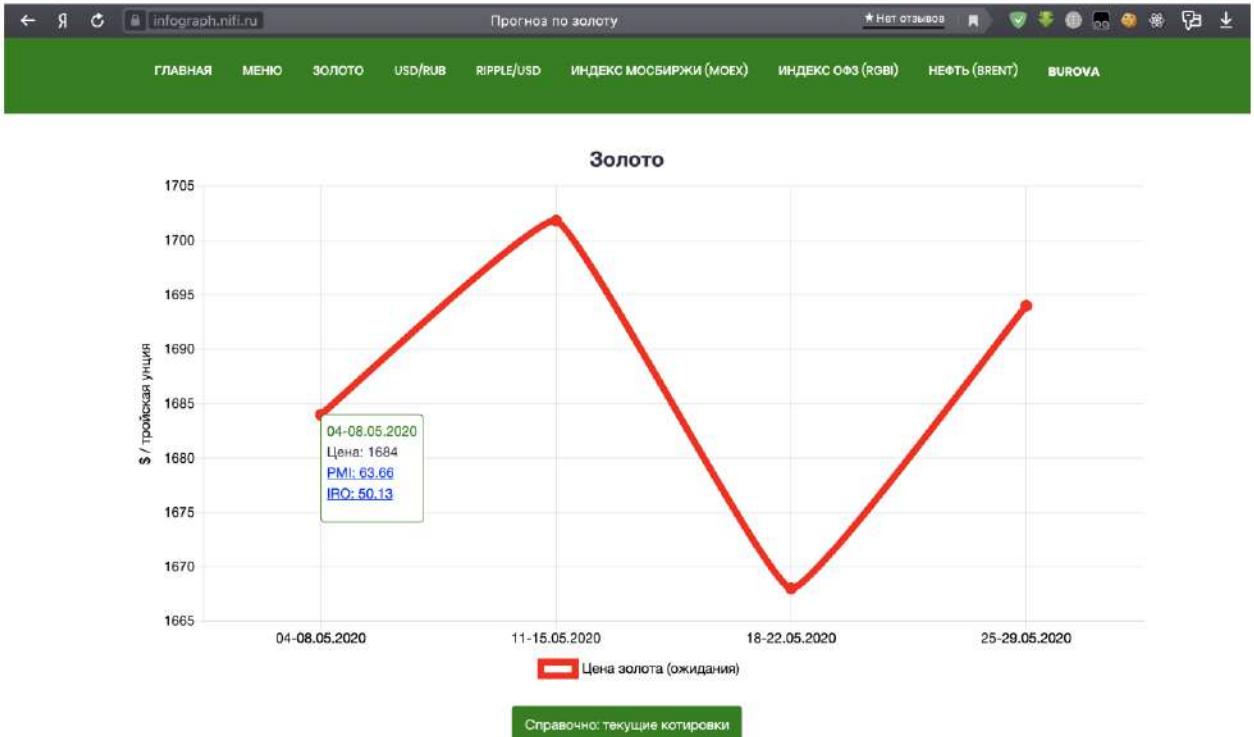


Рисунок 47 – График золота

Описание IRO на период:

Индикатор релевантности ожиданий рынка составил 50,13 л., что указывает на возможность сохранения в целом позитивных оценок профессиональными участниками российского финансового рынка факторов изменения мировой цены золота. Анализ участниками рынка влияния факторов на рынок золота свидетельствует о превалировании повышательных факторов – (0,63), что связывается участниками рынка с ожиданием замедления роста мировой экономики в условиях пандемии коронавируса, сохранением спроса на золото как надежного актива в условиях нестабильности в мировой экономике; продолжением политики денежно-кредитного смягчения центральными банками мира, ослаблением курса доллара. Вес понижательных факторов составил 0,37 л., что связывается рядом участников рынка со снижением спроса на золото, в условиях ослабления карантинных мер странами мира, в виду сокращения числа зараженных коронавирусом, укреплением доллара США.

Факторы на период:

Факторы	Веса факторов изменения условий рынка золота	
	Повышение	Понижение
Мировая экономика	0,21	0,07
Мировой финансовый рынок	0,25	0,24
Монетарная политика	0,1	0,0
Геополитика	0,07	0,06
Веса факторов	0,63	0,37

Назад

Текущие котировки

<https://www.federalreserve.gov>

Рисунок 48 – Факторы золота

В других разделах все идентично этому, поэтому отдельно они описываются не будут. Стоит только сказать про подразделы – касательно Индекс производства добычи нефти и газа, а также индекс производства обр. пром. для изменения. Для того что бы графики были синхронизированы и не расходились, их самые старые точки должны быть одного месяца, как на Рисунках 49 и 50:

Выберите Индекс производства обр. пром. для изменения

Действие: Выполнить Выбрано 0 объектов из 14

<input type="checkbox"/> Индекс производства обр. пром.
<input type="checkbox"/> Май 2020
<input type="checkbox"/> Апрель 2020
<input type="checkbox"/> Март 2020
<input type="checkbox"/> Февраль 2020
<input type="checkbox"/> Январь 2020
<input type="checkbox"/> Декабрь 2019
<input type="checkbox"/> Ноябрь 2019
<input type="checkbox"/> Октябрь 2019
<input type="checkbox"/> Сентябрь 2019
<input type="checkbox"/> Август 2019
<input type="checkbox"/> Июль 2019
<input type="checkbox"/> Июнь 2019
<input type="checkbox"/> Май 2019
<input type="checkbox"/> Апрель 2019

Добавить Индекс производств обр. пром. +

14 Индекс производств обр. пром.

Рисунок 49 – Индекс производства обрабатывающей промышленности

← Я ⌂ infograph.nifi.ru Выберите Индекс производства добычи нефти и газа для изменения | Админ...

Администрирование Django

Начало / Main + Индекс производства добычи нефти и газа

Выберите Индекс производства добычи нефти и газа для изменения

Добавить индекс производства добычи нефти и газа +

Действие: ━━━━ Выполнить Выбрано 0 объектов из 14

индекс производства добычи нефти и газа

Май 2020

Апрель 2020

Март 2020

Февраль 2020

Январь 2020

Декабрь 2019

Ноябрь 2019

Октябрь 2019

Сентябрь 2019

Август 2019

Июль 2019

Июнь 2019

Май 2019

Апрель 2019

14 Индекс производства добычи нефти и газа

Рисунок 50 – Индекс производства добычи нефти и газа

Заметьте, что самые старые точки синхронизированы – это апрель 2019 года.

На Рисунке 51:

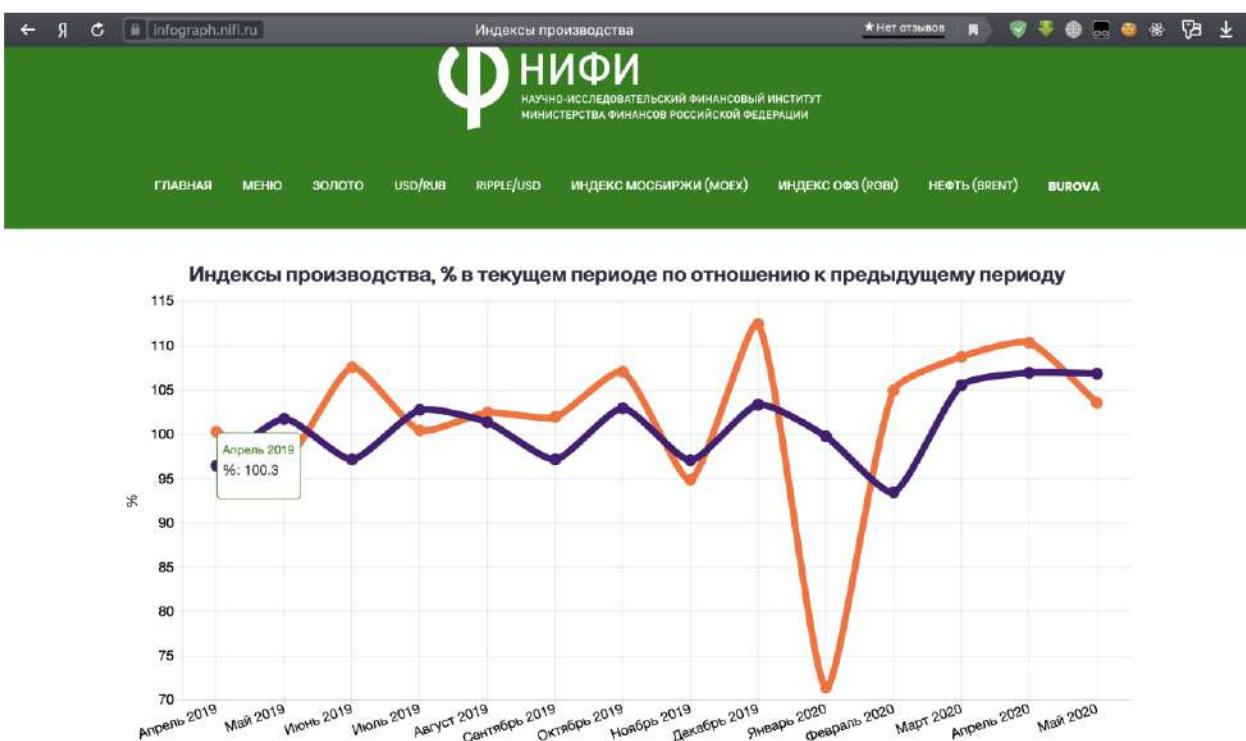


Рисунок 51 – График индексов производства

Последнее что надо сказать – рассылка. Открыть ее можно в верхнем навбаре, как на Рисунке 52:

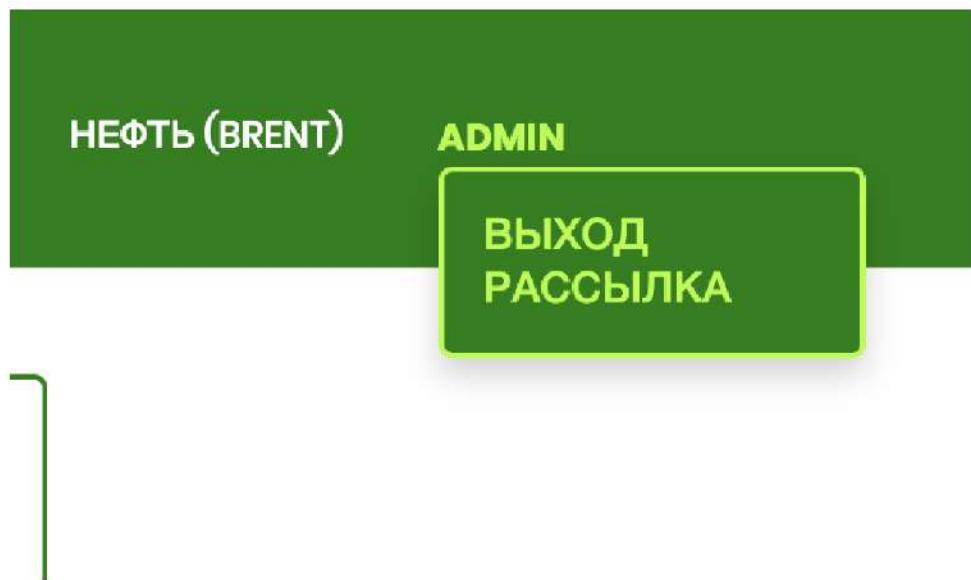


Рисунок 52 – Выпадающее меню

Здесь можно выйти из аккаунта, а также можно перейти к странице по рассылке уведомлений, как на Рисунке 53:

A screenshot of a web browser showing the 'Рассылка уведомлений' (Email Notifications) page. The URL is infograph.nifi.ru. The page header includes the logo of the National Financial Research Institute (НИФИ) and the text 'НАУЧНО-ИССЛЕДОВАТЕЛЬСКИЙ ФИНАНСОВЫЙ ИНСТИТУТ МИНИСТЕРСТВА ФИНАНСОВ РОССИЙСКОЙ ФЕДЕРАЦИИ'. Below the header is a navigation bar with links: ГЛАВНАЯ, МЕНЮ, ЗОЛОТО, USD/RUB, RIPPLE/USD, ИНДЕКС МОСБИРЖИ (МОЕХ), ИНДЕКС ОФЗ (RGBO), НЕФТЬ (BRENT), and ADMIN. The main content area is titled 'Рассылка уведомлений' and contains a text input field labeled 'Текст оповещения:' with a placeholder '...', and a green button labeled 'Начать рассылку'.

Рисунок 53 – Страница рассылки

После обновления данных, сюда заходят администраторы, и рассылают сообщения об обновлении прогнозов. Рассылка происходит автоматически, почтовые адреса берутся из аккаунтов пользователей.

2.10 Руководство пользователя

Поскольку открытой авторизации на портале нет, то аккаунт вы получаете от администратора. После того, как вы получили аккаунт то при заходе в меню сайта вас выкинет на страницу авторизации, как на Рисунке 54:

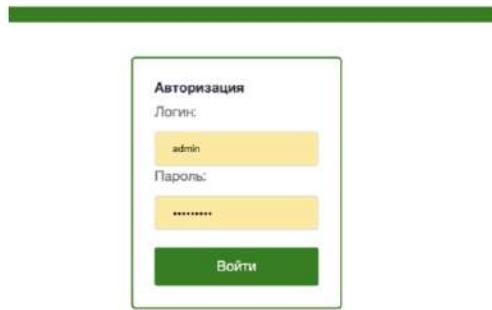


Рисунок 54 – Страница авторизации

Вводим полученные логин и пароль, и входим. Нас должно перекинуть в меню сайта, показанного на Рисунке 55:

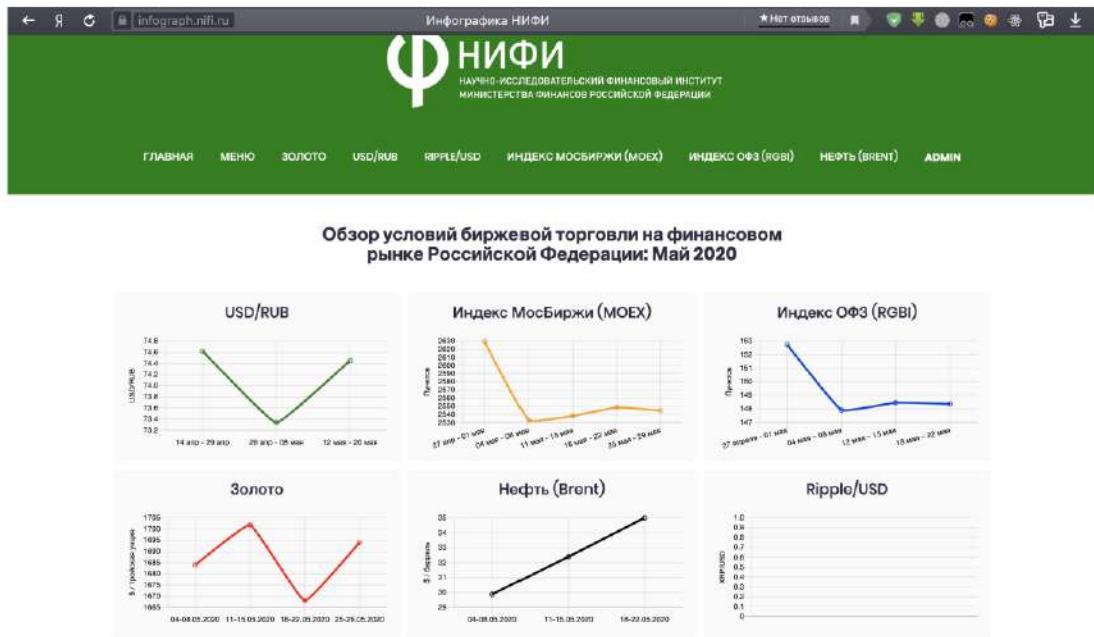


Рисунок 55 – Главное меню портала

Здесь уже видны графики прогнозов, но если нажать на них, то можно перейти в ваш сегмент. Сегмент USD/RUB показан на Рисунке 56:

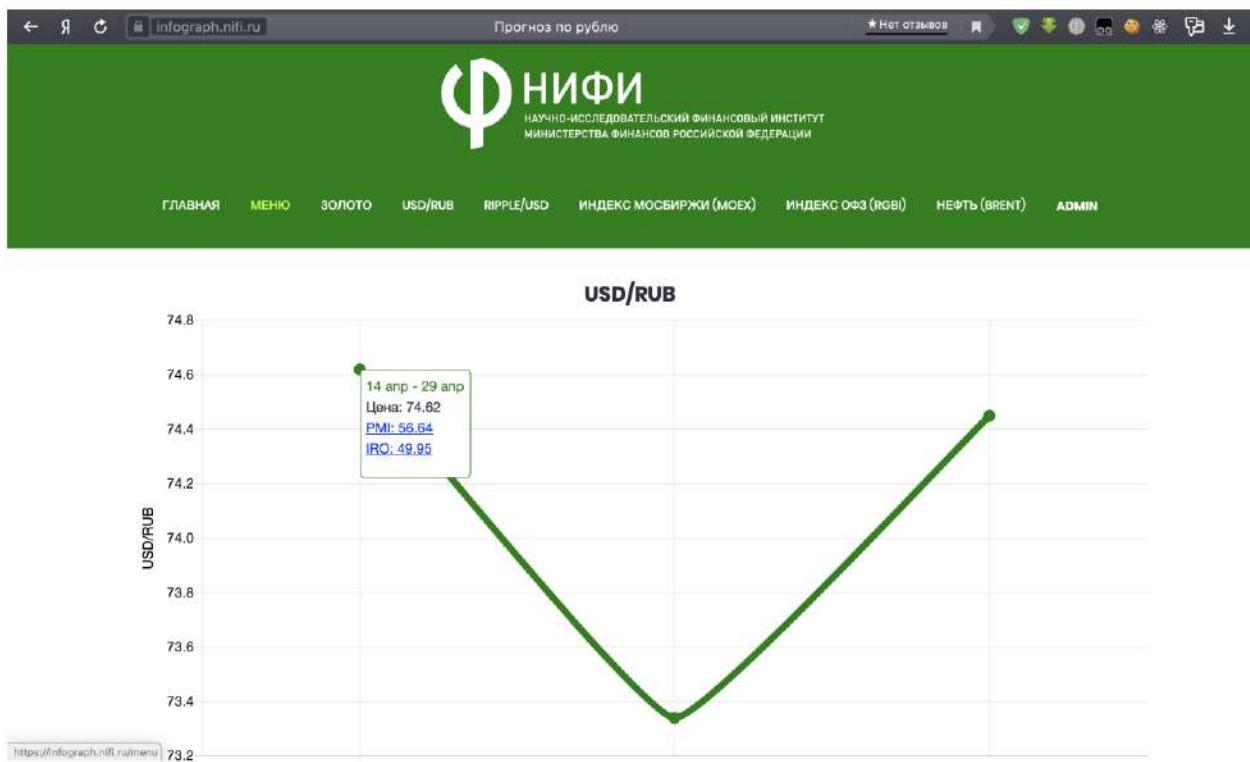


Рисунок 56 – Сегмент USD/RUB

При нажатии на PMI, нас перенесет вниз страницы, где появится описание PMI на текущий период, как на Рисунке 57:

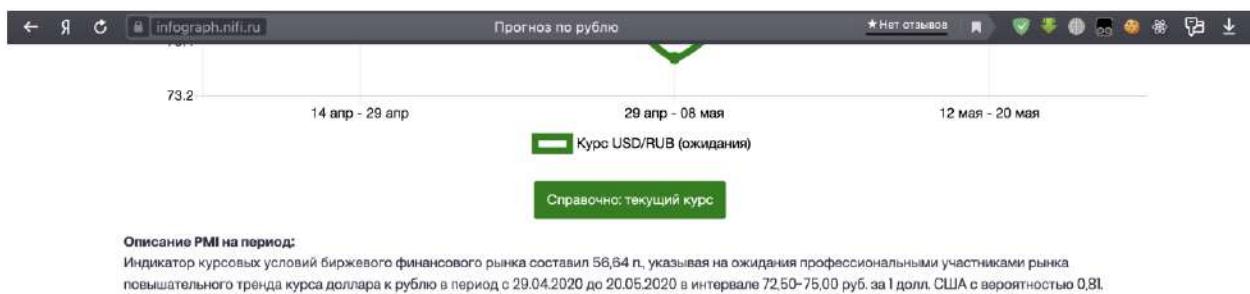


Рисунок 57 – Описание PMI

При нажатии на IRO, нас перенесет на страницу его описания, где помимо самого описания, будет таблица с факторами, как на Рисунке 58:

← Я ⌂ infograph.nifi.ru Описание IRO на период ★ Нет отзывов

Описание IRO на период:
Индикатор релевантности ожиданий составил 49,95 л, что указывает на преобладание оценки понижаительных факторов формирования ценовой конъюнктуры рубля. Внешние факторы (0,73) превалируют над внутренними (0,27). К понижающим факторам (0,55) значительная доля участников рынка относит повышенную волатильность на рынке энергоносителей, а также укрепление доллара в бивалютной корзине. К повышающим факторам (0,45) меньшая доля участников рынка относит ожидания восстановления экономики на фоне постепенного снятия ограничительных мер против распространения коронавирусной инфекции, повышение спроса к рисковым активам, валютные интервенции Банка России, аукционы ОФЗ и налоговый период.

Факторы на период:

Факторы	Веса факторов изменения условий рынка	
	Повышение	Понижение
Внутренние факторы	0,27	
Экономика	0,0	0,0
Финансовый рынок	0,11	0,04
Монетарная политика	0,09	0,04
Внешние факторы	0,73	
Экономика	0,2	0,36
Финансовый рынок	0,05	0,13
Монетарная политика	0,0	0,0
Геополитика	0,0	0,0
Веса факторов	0,45	0,55

[Назад](#) [Текущий курс](#)

Рисунок 58 – Описание IRO на период

Все ссылки в таблице кликабельны, и ведут на сторонние экономические информационные ресурсы. Все алгоритмы действий универсальны для каждого раздела портала.

Заключение

Подводя итоги выпускной квалификационной работы, можно сказать, что все поставленные задачи были выполнены и протестированы должным образом. За время работы был полностью реализован портал экономической инфографики. Некоторые идеи, которые не совсем вписывались в портал, были перенесены в другие проекты.

Были выполнены следующие задачи:

- Разработана общая схема и концепция сайта
- Разработан дизайн сайта
- Разработаны необходимые модели к базам данных
- Разработано приложение;
- Реализованы основные функциональные требования к программному обеспечению;
- Проведено тестирование приложения;
- Разработаны руководства по использованию приложения;

В результате был разработан веб-портал для удобного доступа к прогнозам, которые могут помочь при биржевой торговле. К приложению были составлены четыре вида руководств. Ошибки, выявленные в процессе тестирования приложения, были успешно устранены. В дальнейшем, для улучшения программы, будет доделана мобильная версия данного портала.

Данное приложение успешно выведено на производство, и в данный момент используется в интересах НИФИ.

Список использованных источников

Законодательные и нормативные акты

- 1) ГОСТ 7.82 – 2001. Библиографическая запись. Библиографическое описание электронных ресурсов. Общие требования и правила составления. – М.: ИПК Издательство стандартов, 2001. –21 с.
- 2) ГОСТ 7.1. – 2003. Библиографическая запись. Библиографическое описание. Общие требования и правила составления. – М.: ИПК Издательство стандартов, 2004. – 169 с.
- 3) ГОСТ 7.32 – 2001. Система стандартов по информации, библиотечному и издательскому делу. Отчёт о научно-исследовательской работе. Структура и правила оформления. – М.: ИПК Издательство стандартов, 2001. – 21 с.

Учебная и научная литература

- 1) Лутц М. Изучаем Python. – Москва, Символ-Плюс, 2011. –1280 с.
- 2) Лутц М. Программирование на Python. Том 1. – Москва, Символ-Плюс, 2011. –992 с.
- 3) Дронов В. Django 2.1. Практика создания веб-сайтов на Python - Москва: ВНВ, 2019. – 672 с.

Интернет-ресурсы

- 1) Документация Python 3– [Электронный ресурс]. – Режим доступа: <https://docs.python.org/3/index.html>
- 2) Документация Django – [Электронный ресурс]. – Режим доступа: <https://docs.djangoproject.com/en/3.0/>
- 3) ChartJS – [Электронный ресурс]. – Режим доступа: <https://www.chartjs.org/>
- 4) Bootstrap 4 – [Электронный ресурс]. – Режим доступа: <https://bootstrap-4.ru/>
- 5) Документация MySQL – [Электронный ресурс]. – Режим доступа: <https://dev.mysql.com/doc/>

Приложение А

Данная работа выполнена мной самостоятельно

05.06.20

А—