|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |

**Институт информационных технологий (ИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

**КУРСОВАЯ РАБОТА**

по дисциплине: Разработка серверных частей интернет-ресурсов

по профилю: Разработка программных продуктов и проектирование информационных систем

направления профессиональной подготовки: 09.03.04 «Программная инженерия»

Тема: Серверная часть веб-приложения «Инвестиционный аналитик»

Студент: Миронов Алексей Дмитриевич  
Группа: ИКБО-02-19

Работа представлена к защите (дата) Миронов А.Д. / C:\Users\qwed1\AppData\Local\Microsoft\Windows\INetCache\Content.Word\подпись.png /

(подпись и ф.и.о. студента)

Руководитель: доцент Алпатов Алексей Николаевич

Работа допущена к защите (дата) /Алпатов А.Н./

(подпись и ф.и.о. рук-ля)

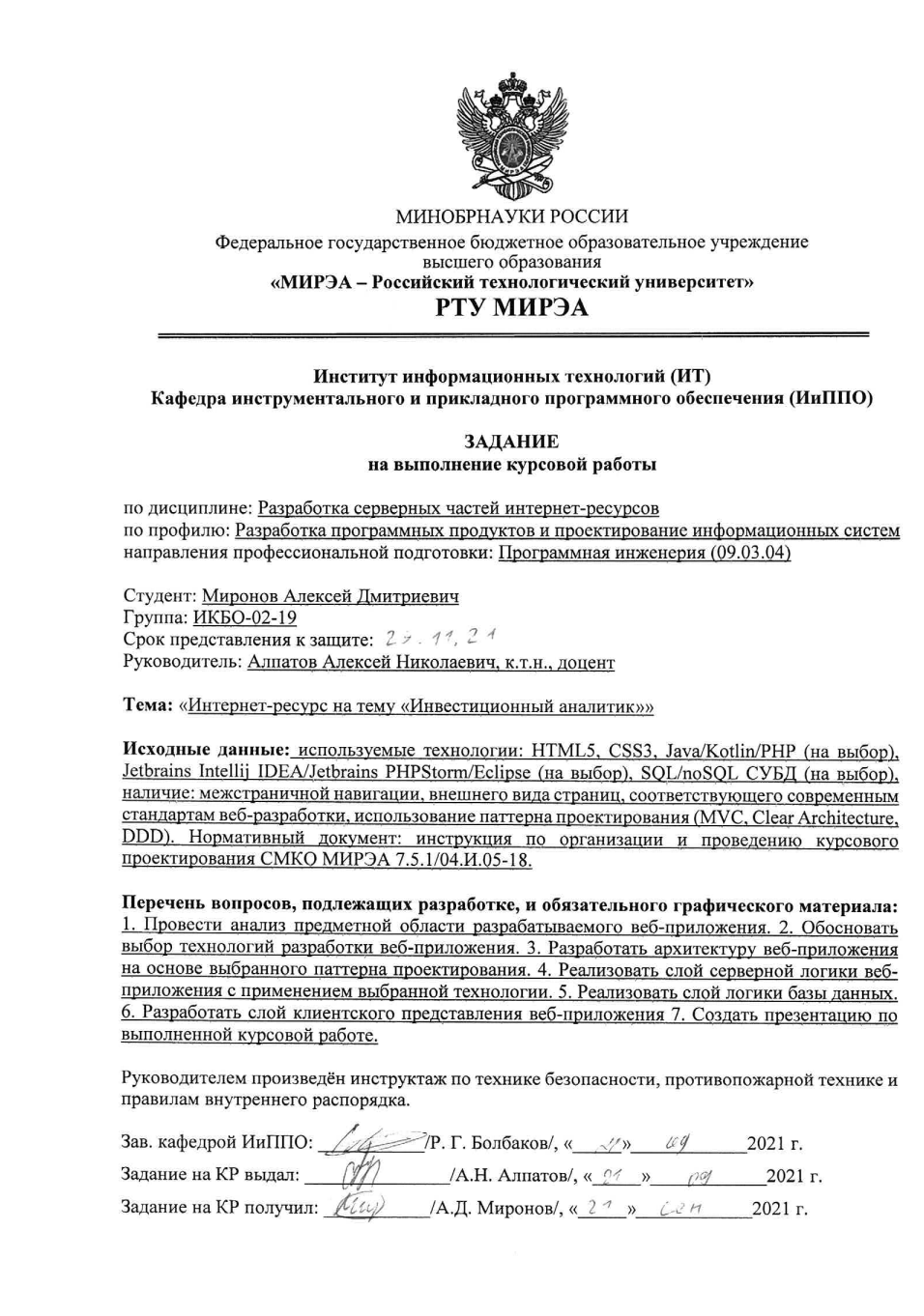
Оценка по итогам защиты:

/ /

/ /

(подписи, дата, ф.и.о., должность, звание, уч. степень двух преподавателей, принявших защиту)

М. РТУ МИРЭА. 2021



**ГЛОССАРИЙ**

Оглавление

[АННОТАЦИЯ 5](#_Toc89093082)

[ВВЕДЕНИЕ 6](#_Toc89093083)

[ОСНОВНАЯ ЧАСТЬ 8](#_Toc89093084)

[1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ РАЗРАБАТЫВАЕМОГО ВЕБ-ПРИЛОЖЕНИЯ 8](#_Toc89093085)

[2. ВЫБОР И ОБОСНОВАНИЕ ИСПОЛЬЗУЕМЫХ ТЕХНОЛОГИЙ РАЗРАБОТКИ 9](#_Toc89093086)

[3. РАЗРАБОТКА АРХИТЕКТУРЫ ПРИЛОЖЕНИЯ НА ОСНОВЕ ВЫБРАННОГО ПАТТЕРНА 10](#_Toc89093087)

[4. РАЗРАБОТКА СЕРВЕРНОЙ ЧАСТИ ИНТЕРНЕТ-РЕСУРСА 12](#_Toc89093088)

[5. РАЗРАБОТКА КЛИЕН-СЕРВЕРНОЙ ЧАСТИ ИНТЕРНЕТ-РЕСУРСА 16](#_Toc89093089)

[6. РАЗРАБОТКА БАЗЫ ДАННЫХ 23](#_Toc89093090)

[ЗАКЛЮЧЕНИЕ 24](#_Toc89093091)

[СПИСОК ЛИТЕРАТУРЫ 25](#_Toc89093092)

## АННОТАЦИЯ

В данной курсовой работе была реализована серверная часть интернет ресурса «Инвестиционный аналитик». Для разработки данного приложения был использован язык программирования PHP. На нем была реализована вся основная логика программы. Для отображения в браузере использовались такие технологии как HTML, JavaScript, CSS. С их помощью удалось сделать вывод информации, а также удобно ее расположить на страницу браузера.

В процессе работы были освоены такие важные навыки, как комбинирование различных языков программирования. А именно вывод html кода с помощью php, передача переменных из языка php в язык JavaScript посредством html кода. Также в данном проекте была использована технология MySql, которая позволила реализовать базу данных в приложении. Всё приложение было реализовано в docker контейнере что позволит существенно облегчить процесс загрузки приложения на удаленный сервер.

Большинство из перечисленных технологий были изучены в ходе курса «Разработка серверных частей интернет-ресурсов», что позволило облегчить процесс разработки, а сама курсовая позволила закрепить данный материал.

При разработки данного web-сервиса были использованы различные информационные источники, такие как: статьи на форумах, официальная документация и обучающие видео.

## ВВЕДЕНИЕ

Данная тема всегда была очень актуальна. Практически каждый человек желает сохранить свои сбережения или же преумножить их. Для этого есть множество способов, таких как положить деньги в банк под процент, открыть собственный бизнес или же инвестировать деньги во что-либо. Каждый из этих способов действительно хорошо справляется со своей задачей при правильном подходе и хорошем стечении обстоятельств. Однако есть и недостатки, который могут быстро лишить предпринимателя всех его ресурсов.

Например, открытие собственного бизнеса один из наиболее эффективных способов. Однако он самый рискованный и энергозатратный. Без должного опыта бизнес не выйдет на должный оборот, а скорее всего обанкротится в ближайшее время. Даже опытные бизнесмены не всегда могут удачно открыть собственный бизнес.

Положить деньги в банк под процент на первый взгляд кажется самым надежным и простым способом сохранения или преумножения своих средств, однако это имеет смысл только если делать вклады в банк на довольно крупные суммы. Но и даже в этом случае не всё так гладко. Зачастую инфляция делает вклад малоэффективным.

Инвестиции чем-то похожи на вклады в банке. Этот процесс не требует постоянного присутствия. Достаточно лишь в нужное время купить и продать. При инвестировании процентная ставка зачастую гораздо выше, но и риск потерять сбережения тоже присутствует. Как начинающие так и опытные инвесторы зачастую нуждаются в удобном аналитическом сервисе, так как искать нужную информацию самому довольно сложно. Именно такое сервис был разработан в данной курсовой работе.

Целью данного web-сервиса является удобное предоставления данных о валютах пользователю. Проект был реализован на языка программирования PHP с использованием технологий HTML, CSS3, JavaScript. А также Docker и MySQL. В ходе разработки были выполнены следующие задачи:

* Анализ предметной области;
* Разметка страниц с помощью HTML5;
* Организация межстраничной навигации сайта;
* Реализация клиентской логики с применением языка программирования JavaScript;
* Проведение оптимизации размещения контента.

## ОСНОВНАЯ ЧАСТЬ

# АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ РАЗРАБАТЫВАЕМОГО ВЕБ-ПРИЛОЖЕНИЯ

В разработанном веб-сервисе реализован такой функционал как парсинг данных в валютах с сайта Центробанка России с последующей передачей этих данных пользователю через HTML страницу. Реализована функция открытия конкретной валюты по ссылке. На открывшейся странице пользователь увидит график изменения курса данной валюты за несколько лет. Исторические данные о курсах подгружаются из MySQL сервера. Также пользователь может выбрать конкретный временной диапазон и веб-сервис выведет график с курсом валют именно за этот период.

Данное веб-приложение разработано для удобного предоставления пользователю данных о текущих, а также об исторических курсах различных валют. Пользователь может в подробностях изучить интересующие его валюты детально изучая поведение графиков на различных промежутках времени.

Веб-сервис хранит в базе данных информацию более чем о 30 валютах за промежуток в 5 лет. Пользователь может получить доступ к этим данным посредством браузера или GET запроса. Сервер выдаст пользователю HTML код странице в зависимости от выбранной странице. Также пользователь получает код на JavaScript который отобразит график по нужному диапазону дат, которые вводятся в URL при выборе соответствующих дат на календаре.

# ВЫБОР И ОБОСНОВАНИЕ ИСПОЛЬЗУЕМЫХ ТЕХНОЛОГИЙ РАЗРАБОТКИ

• HTML5, данная технология используется для первичного размещения контента в окне браузера

• CSS3, используется для создания различных стилей для объектоа на HTML Странице

• JavaScript, для создания анимации, отображения конкретных значений при наведении мышкой, а также для вывода анимированного графика в окно браузера

• PHP, для реализации серверной логики веб-клиента

• MySQL, удобный sql сервер для проекта. Позволяет создать базу данных и обрабатывать запросы.

• Паттерн проектирования «Model View Controller». Этот патерн был использован в связи с удобством разработки. Каждый из файлов в проекте выполняет свою роль. Model – доступ к базе данных, хранение информации, View – вывод различных объектов в окно браузера. Controller – передает запросы от пользователя к Model.

# РАЗРАБОТКА АРХИТЕКТУРЫ ПРИЛОЖЕНИЯ НА ОСНОВЕ ВЫБРАННОГО ПАТТЕРНА

Для разработки данного веб-приложения был выбран паттерн программирования «Model View Controller».

Model-View-Controller (MVC, «Модель-Представление-Контроллер», «Модель-Вид-Контроллер») — схема разделения данных приложения, и управляющей логики на три отдельных компонента: модель, представление и контроллер — таким образом, что модификация каждого компонента может осуществляться независимо.

Модель предоставляет данные и реагирует на команды контроллера, изменяя своё состояние

Представление отвечает за отображение данных модели пользователю, реагируя на изменения модели.

Контроллер интерпретирует действия пользователя, оповещая модель о необходимости изменений. [1]



Рисунок 1 - оригинальная схема паттерна Model-View-Controller

В данной курсовой работе проект был реализован на основе именно этого паттерна. Итоговая архитектура получилась следующая.

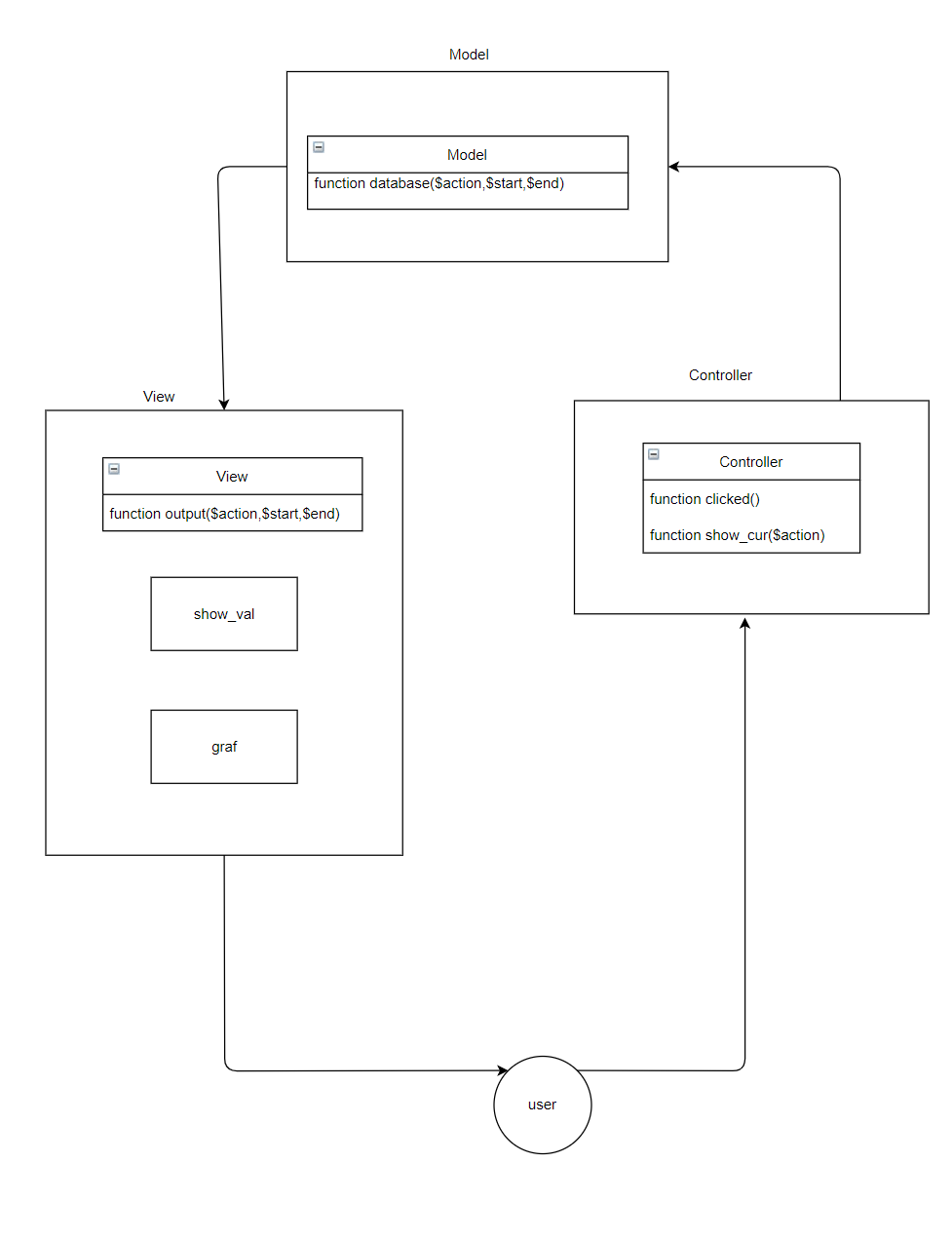


Рисунок 2 - оригинальная реализации проекта

# РАЗРАБОТКА СЕРВЕРНОЙ ЧАСТИ ИНТЕРНЕТ-РЕСУРСА

В данной работе серверная логика реализована на языке программирования PHP. Всё приложение было размещено в Docker контейнере.

Весть код представлен в листингах ниже

Листинг 1 – код реализации index.php.

<html lang="en">

    <head>

        <title>Hello world page</title>

        <link rel="stylesheet" href="/protected/style.css" type="text/css"/>

    </head>

    <body>

        <div class="container">

        <nav class="navbar">

            <h1>Инвестиционный анализ</h1>

            <?php

            include 'main/controller/controller.php';

            include 'main/model/model.php';

            include 'main/view/view.php';

     $mysqli = new mysqli("courseDB", "user", "password", "courseDB");

            // $result = $mysqli->query("SELECT \* FROM courseDB.vals");

            // foreach ($result as $row)

            // {

            //     echo "{$row['date']} {$row['KRW']} </br>";

            // }

            $model = new Model();

            $controller = new Controller($model);

            $view = new View($controller, $model);

            if (isset($\_GET['action']) && !empty($\_GET['action'])) {

                // echo "action!!";

                $controller->show\_cur($\_GET['action']);

                // echo $\_GET['start\_date'],"<br>";

                // echo $\_GET['end\_date'];

            }

            // echo "3";

            // if(empty($\_GET['action'])){

            //  echo '<h3><a href="http://localhost:8080/?action=show\_cur">курсы валют</a><br><h3>';

            // }

            echo $view->output($\_GET['action'],$\_GET['start\_date'],$\_GET['end\_date']);

            ?>

        </nav>

    </div>

    </body>

</html>

Листинг 2 – код реализации view.php.

<style>

.hiden\_zn {

  display: none;

}

.hiden\_lb {

  display: none;

}

</style>

<?php

class View

{

    private $model;

    private $controller;

    public function \_\_construct($controller,$model) {

        echo '<h3><a href="http://localhost:8080/?action=show\_cur">курсы валют</a><br><h3>';

        $this->controller = $controller;

        $this->model = $model;

    }

    public function output($action,$start,$end) {

        if($action == "show\_cur"){

        $list = $this->model->vals;

        for ($x=0; $x<count($list); $x+=10){

            echo "<tr><td>{$list[$x]}</td> <td>{$list[$x+2]}</td> <td>{$list[$x+4]}</td> <td>{$list[$x+6]}</td> <td><b> {$list[$x+8]}</b>  руб.</td><td> <a href='http://localhost:8080/index.php?action={$list[$x+2]}'> More</a></td> </tr><br>";

        }

        }else if($action == ""){

            echo "";

        }else{

            $ret = $this->model->database($action,$start,$end);

            // foreach ($ret as $row){

            //     echo $row;

            // }

            $halved = array\_chunk($ret, ceil(count($ret)/2));

            $ar = $halved[0];

            $lb = $halved[1];

            for ($i=0; $i<count($ar); $i+=1){

                echo "<div class='hiden\_zn'> {$ar[$i]} </div>";

            }

            for ($i=0; $i<count($ar); $i+=1){

                echo "<div class='hiden\_lb'> {$lb[$i]} </div>";

            }

            echo "<form>

            <p>Код валюты: <input type='text' name='action' value={$action}></p>

            <p>Начальная дата: <input type='date' name='start\_date' value={$\_GET['start\_date']}></p>

            <p>Конечная дата: <input type='date' name='end\_date' value={$\_GET['end\_date']}></p>

            <p><input type='submit' value='Отправить'></p>

           </form>";

            echo '<canvas id="myChart"></canvas>';

            // echo '<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.1/Chart.js"></script>';

            echo '<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.6.0/chart.js"></script>';

            echo '<script src="main/view/graf.js"></script>';

            // echo 'рекомендации';

        }

    }

}

?>

Листинг 3 – код реализации model.php.

<?php

class Model

{

    public $string;

    public $vals;

    public function \_\_construct(){

        $this->string = "MVC + PHP = Awesome, click here!";

        $this->vals =  array();

    }

    public function database($action,$start,$end){

        $mysqli = new mysqli("courseDB", "user", "password", "courseDB");

        // echo $start;

        if($start!='' and $end!=''){

            // echo 'in';

            $result = $mysqli->query("SELECT \* FROM courseDB.vals WHERE courseDB.vals.date >= '$start' AND courseDB.vals.date <= '$end'");

            // $result = $mysqli->query("SELECT \* FROM courseDB.vals WHERE courseDB.vals.date >= '2020-07-08 00:00:00' AND courseDB.vals.date <= '2020-09-08 00:00:00'");

        }else{

            // echo 'else';

            $result = $mysqli->query("SELECT \* FROM courseDB.vals");

        }

        $ar = array();

        $lb = array();

        foreach ($result as $row)

        {

            // echo "{$row['date']} {$row['KRW']} </br>";

            array\_push($ar, $row[$action]);

            array\_push($lb, $row['date']);

        }

        return array\_merge($ar, $lb);

    }

}

?>

Листинг 4 – код реализации controller.php.

<?php

class Controller

{

    private $model;

    public function \_\_construct($model){

        $this->model = $model;

    }

    public function clicked() {

        $this->model->string = "Updated Data, thanks to MVC and PHP!";

    }

    public function show\_cur($action) {

        if($action == "show\_cur"){

        $this->model->string = "cur";

        error\_reporting(E\_ERROR | E\_PARSE);

            $code = file\_get\_contents('https://www.cbr.ru/currency\_base/daily/');

            $dom = new DOMDocument();

            $dom->loadHTML($code);

            $tables = $dom->getElementsByTagName('td');

            $list = array();

            foreach ($tables as $book) {

                array\_push($list, $book->nodeValue, PHP\_EOL);

            }

            $this->model->vals = $list;

        }else{

            echo "";

        }

    }

}

?>

# РАЗРАБОТКА КЛИЕН-СЕРВЕРНОЙ ЧАСТИ ИНТЕРНЕТ-РЕСУРСА

Для реализации клиентской части приложения были использованы такие технологии как HTML, CSS и JavaScript

Главная страница приложения имеет только одну ссылку, которая ведет к различному функционалу данного сервиса.

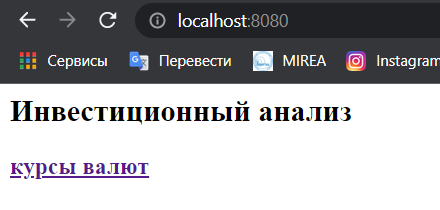


Рисунок 3 – Главная страница

Данная ссылка выводится непосредственно из файла View, посредством команды echo, что соответствует архитектуре MVC. Стоит отметить что в данном приложении все межстраничные переходы выполнены с помощью передачи параметров в URL, что позволяет не использовать прямые ссылки на файлы и придерживаться подхода MVC.



Рисунок 4 – Вывод ссылки в браузер

После нажатия кнопки «курсы валют» пользователь попадает на страницу, которая отображает текущие курсы всех доступных валют.



Рисунок 5 – Вывод курсов валют

Помимо самого курса валюты пользователю доступна и другая информация касающаяся непосредственно конкретной валюты, а именно: цифровой код валюты, буквенный код валюты, количество (сколько единиц данной валюты эквивалентно цене указанной в курсе), а также само название валюты.

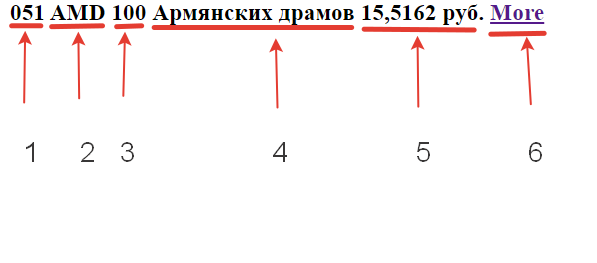


Рисунок 6 – Описание вывода курсов валют

На данном рисунке подробно изображено расположение значений в строке.

1 - Цифровой код валюты

2 - Буквенный код валюты

3 - Количество

4 – Название валюты

5 - цена за единицу данной валюты \* количество (3)

6 – Ссылка на дополнительную информацию о валюте

Вывод данной таблицы происходит с помощью команды echo в файле View, которая принимает информацию от Model. В свою очередь модел принимает эту информацию от Controller, что соответствует паттерну MVC.

Для получения информации о валютах Controller посылает запрос на сайт центробанка России, получая в ответ актуальную информацию о курсах валют на текущий день.

В конце каждой строки также присутствует ссылка “More”, которая отобразит график с курсом валют за весь имеющийся в базе данных промежуток времени.

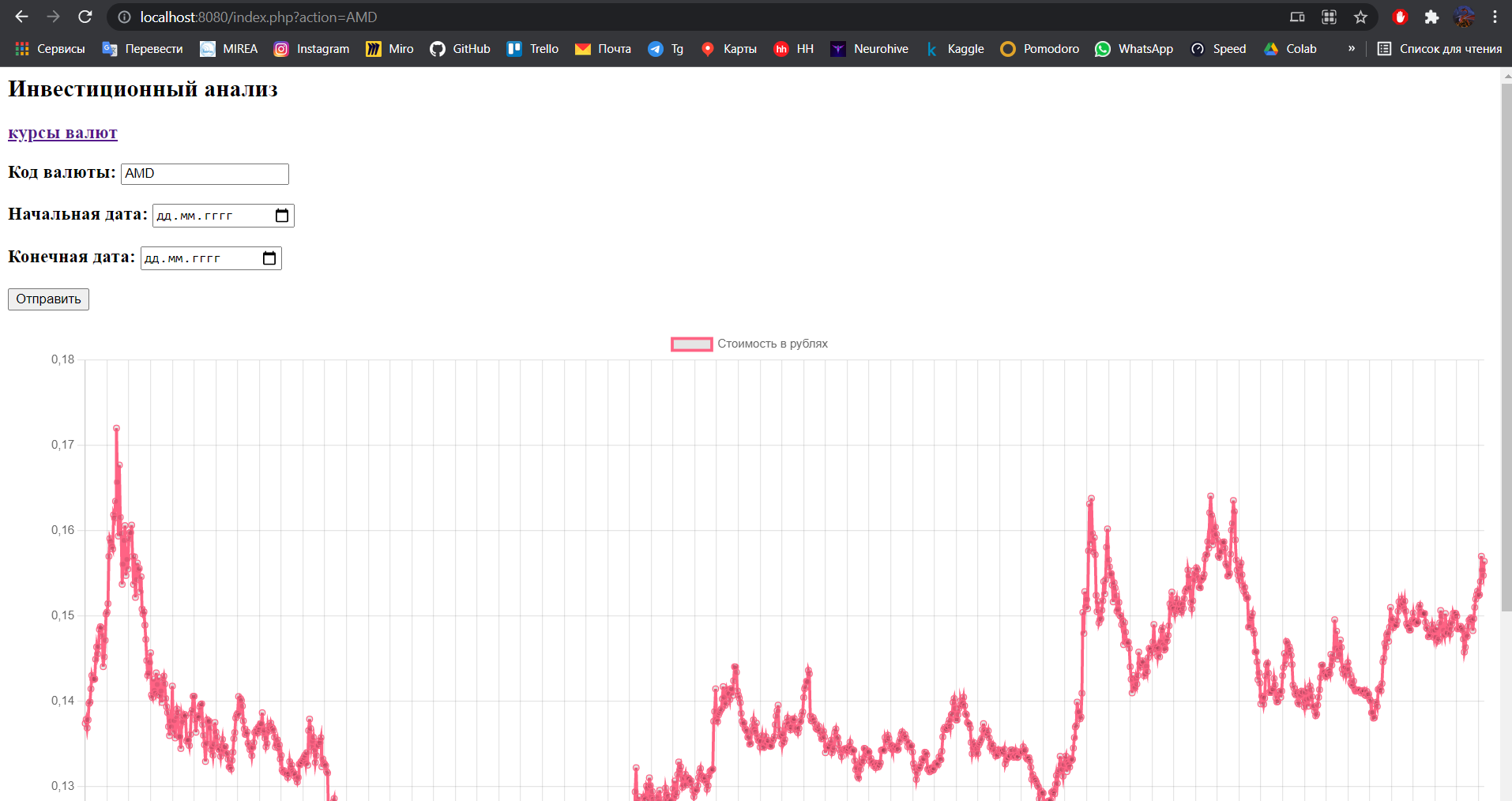


Рисунок 7 – Демонстрация отображения графика.

Прокрутив страницу вниз пользователь может увидеть конкретные даты, к которым относится значение курса.

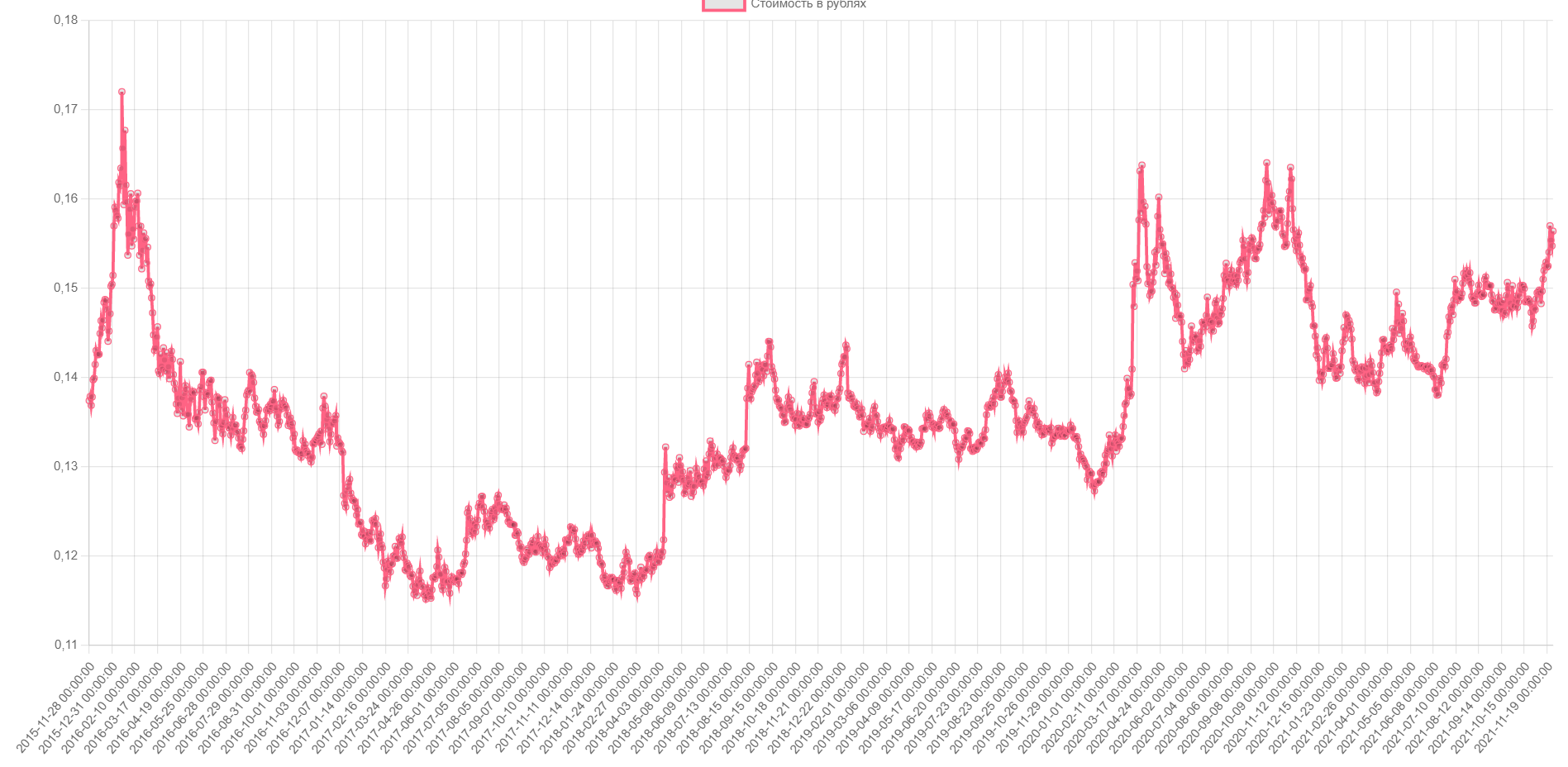


Рисунок 8 – Демонстрация отображения графика с датами.

Пользователю доступно окно выбора дат диапазона и кода валюты. В этом окне пользователь может указать интересующий его диапазон дат и саму валюту. Значения данных полей передаются в параметры URL.

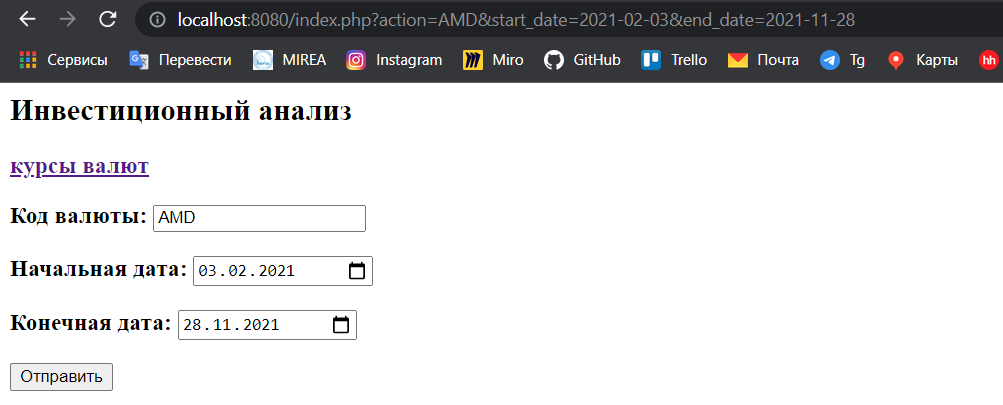


Рисунок 9 – установка диапазона дат через форму в URL.

По данному запросу программа отправит запрос в базу данных, которая возвращает значение курсов из данного диапазона.



Рисунок 10 – установка диапазона дат через форму в URL.

Вывод графика происходит с использованием языка JavaScript. С его помощью был реализован анимированный график. Анимация заключается в плавном перемещении точек от начала координат к их реальным значения во время открытия страницы с графиком. Одной из ключевых особенностей графика является отображение информации о точке на графике при наведении на нее курсора.



Рисунок 11 – JS файл для построения графика.

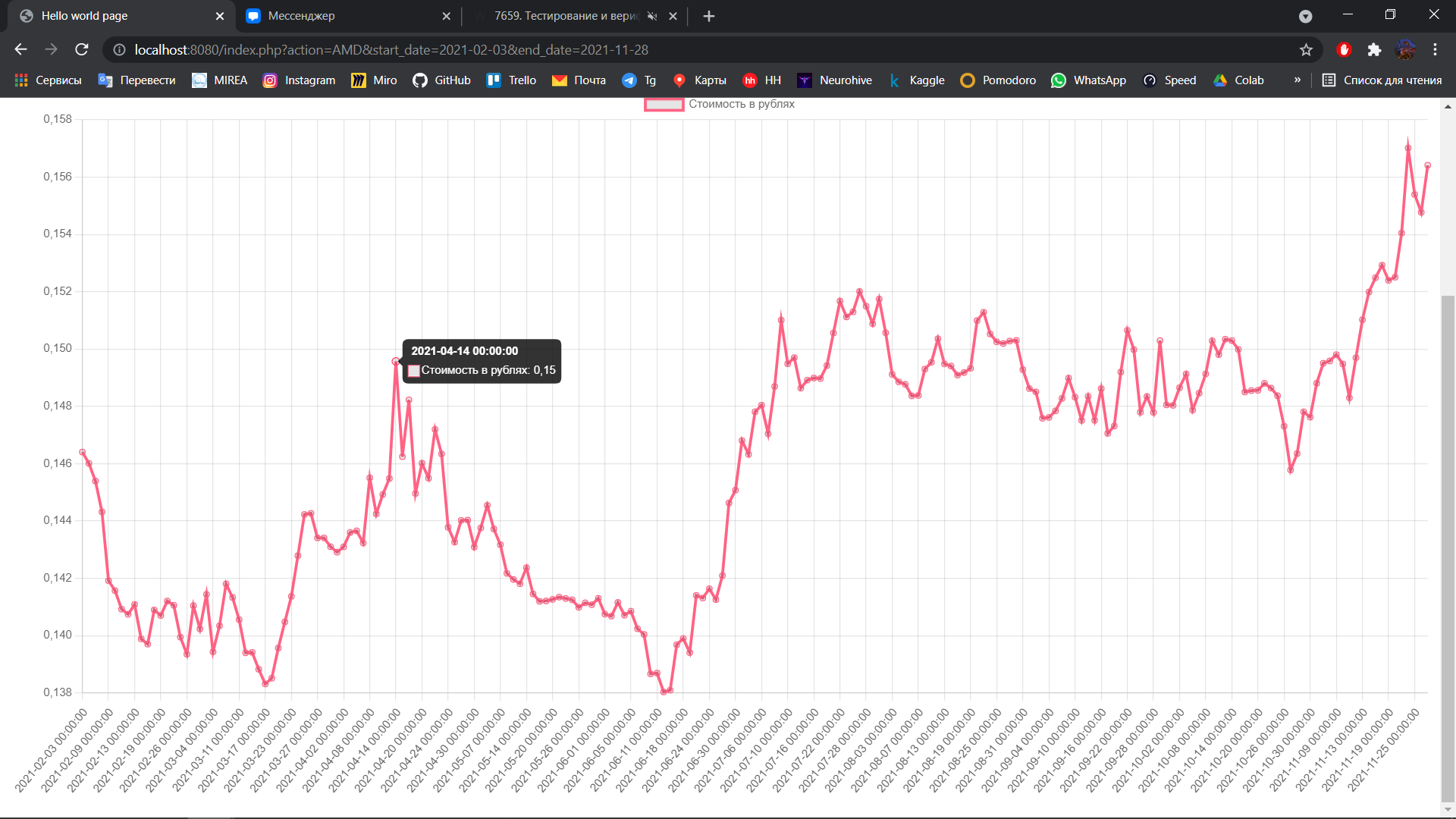


Рисунок 12 – Вывод информации о точке при наведении курсора.

Такой график позволит легко отслеживать изменение по дням, неделям и месяцам.

# РАЗРАБОТКА БАЗЫ ДАННЫХ

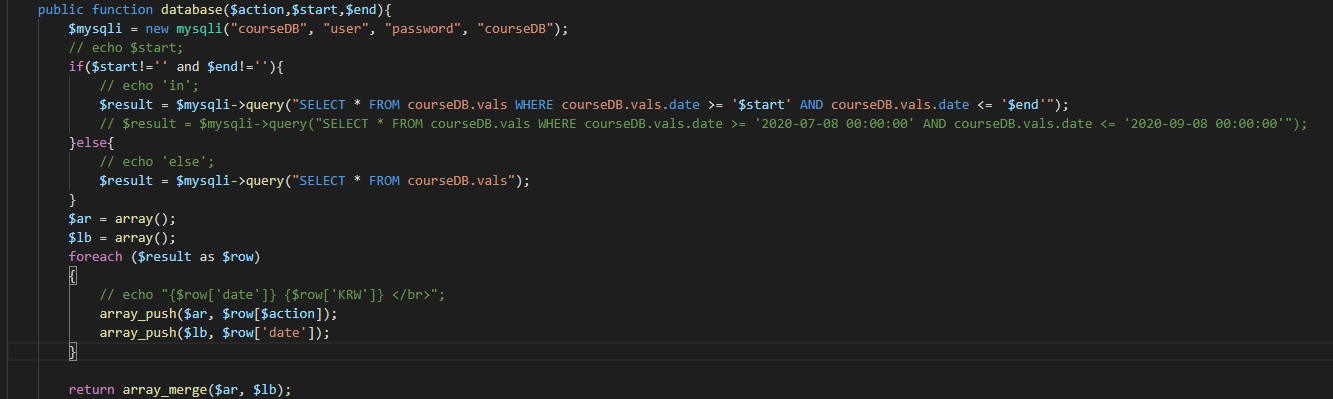
Для реализации базы данных был использован MySQL. В данном веб-приложении sql сервер необходим для хранения данных о валютах за 5 лет. База данных получает запросы от Model, который в свою очередь передает из в View. База данных имеет следующую структуру: В первом столбце находятся даты, во всех остальных столбцах – коды валют и их значения по датам. Такая таблица позволяет сделать запросы очень простыми, и ускоряет получение выборки. Все данные для базы данных были взяты с сайта центробанка России. Парсинг данных производился с помощью языка Python и модулей pandas и requests.

Рисунок 13 – Пример обращения к базе данных через Model.



Рисунок 14 – Получение данных с центробанка.



Рисунок 15 – Формирование итоговой таблицы.

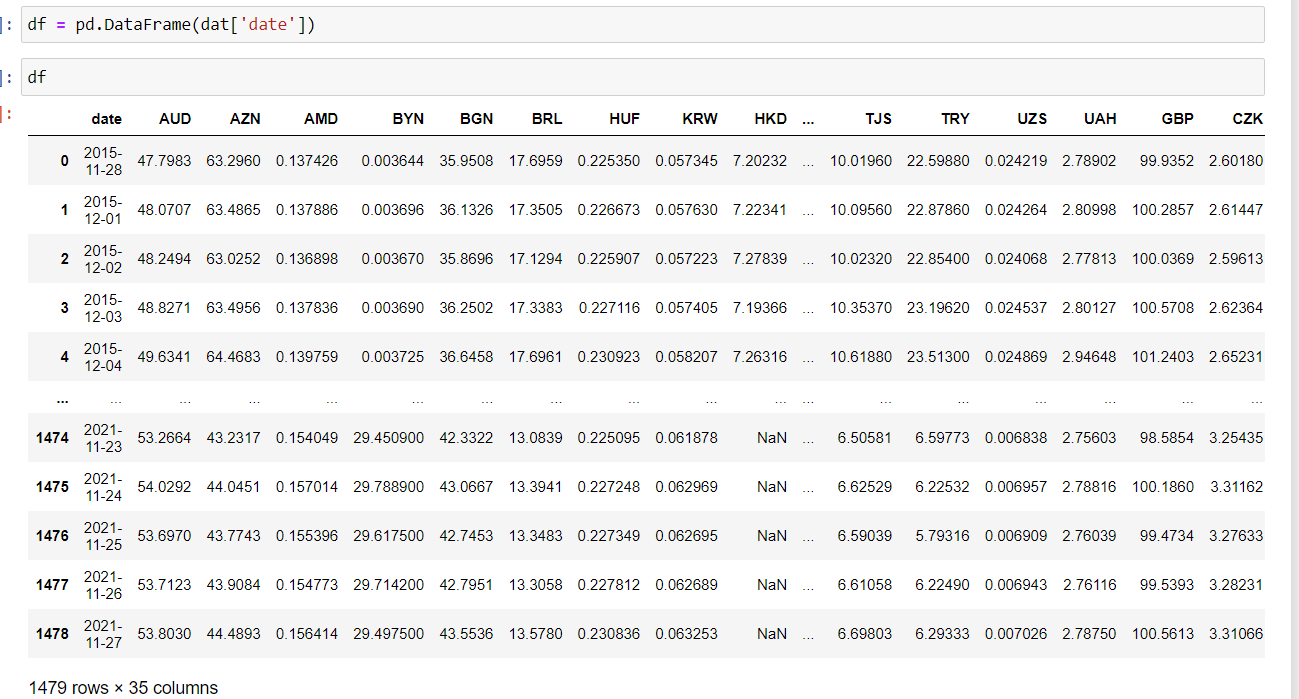


Рисунок 16 – Вид итоговой таблицы для загрузки на сервер.

# ЗАКЛЮЧЕНИЕ

В ходе выполнения данной курсовой работы были закреплены такие важные навыки как программирование на PHP, JavaScript, SQL. Материалы изученные в ходе курса «Разработка серверных частей интернет-ресурсов» позволили разработать полноценных веб-сервис, который позволяет узнавать информацию о текущих курсах валют, а также проанализировать исторические данные и сделать выводы о целесообразности той или иной инвестиции. Был реализован простой и понятный пользовательский интерфейс с возможностью выбора различных валют и диапазонов отображения.

Так же были реализованы и достигнуты следующие цели:

· Проведен анализ предметной области разрабатываемого веб-сервиса;

· Обоснован выбор используемых технологий разработки веб-сервиса;

· Разработано веб-приложения на основе паттерна проектирования;

· Реализована серверная логика веб-приложения с применением выбранной технологии;

· Реализована логика базы данных;

· Разработана клиентская часть веб-сервиса.

# СПИСОК ЛИТЕРАТУРЫ

1. Интернет-ресурс: Wikipedia // wikipedia URL: https://ru.wikipedia.org/wiki/Model-View-Controller (дата обращения: 28.11.2021).

2. Интернет-ресурс: Otus // otus URL: https://otus.ru/nest/post/787/ (дата обращения: 28.11.2021).

3. Интернет-ресурс: Learn javascript // learn javascript URL: https://learn.javascript.ru/array (дата обращения: 28.11.2021).

4. Интернет-ресурс: Сoderoad // coderoad URL: https://coderoad.ru/ (дата обращения: 28.11.2021).

5. Интернет-ресурс: Apexcharts // apexcharts URL: https://apexcharts.com/javascript-chart-demos/line-charts/basic/ (дата обращения: 28.11.2021).

6. Интернет-ресурс: Schoolsw3 // schoolsw3 URL: https://schoolsw3.com/howto/tryhow\_js\_check\_hidden.php/ (дата обращения: 28.11.2021).

7. Интернет-ресурс: Cyberforum // cyberforum URL: https://www.cyberforum.ru/php/thread539948.html/ (дата обращения: 28.11.2021).

8. Интернет-ресурс: Habr // habr URL: https://qna.habr.com/q/986757/ (дата обращения: 28.11.2021).

9. Интернет-ресурс: docs.microsoft // docs.microsoft URL: https://docs.microsoft.com/ru-ru/sql/relational-databases/import-export/import-data-from-excel-to-sql?view=sql-server-ver15/ (дата обращения: 28.11.2021).

10. Интернет-ресурс: Investments101 // investments101 URL: https://investments101.ru/academy/courses/investing-fundamentals/lessons/personal-finances / (дата обращения: 28.11.2021).

11. Интернет-ресурс: Сайт дистанционного обучения РТУ МИРЭА

online-edu.mirea.ru (Дата обращения: 04.11.2021)

12. Интернет-ресурс: php.net // php.net URL: https://www.php.net/manual/ru/index.php/ (дата обращения: 28.11.2021).

13. Никсон Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5. 5-е изд.. - СПб.: Питер, 2021. (Дата обращения: 17.11.2021).

14. Антонова И. И., Кашкин Е. В. Разработка web-сервисов с использованием HTML, CSS, PHP и MySQL [Электронный ресурс]:учебнометодическое пособие. - М.: РТУ МИРЭА, 2019. - – Режим доступа: http://library.mirea.ru/secret/15052019/2022.iso (Дата обращения: 17.11.2021).

15. Диков А. В. Клиентские технологии веб-дизайна. HTML5 и CSS3 [Электронный ресурс]:учебное пособие. - Санкт-Петербург: Лань, 2019. - 188 с. – Режим доступа: https://e.lanbook.com/book/122174 (Дата обращения: 17.11.2021).

16. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебное пособие для вузов / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2021. — 218 с. — (Высшее образование). — ISBN 978-5-534-00515-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/469982 (Дата обращения: 17.11.2021).

17. PHP 7 / Д.В. Котеров, И. В. Симдянов. - СПб.: БХВ-Петербург, 2021. - 1088 с.: ил. (Дата обращения: 17.11.2021).

18. Интернет ресурс: Что такое веб-приложения, виды и их преимущества https://www.azoft.ru/blog/web-apps/ (Дата обращения: 12.11.2021).

19. Интернет-ресурс: Docker.com // docker.com URL: https://www.docker.com/ (дата обращения: 28.11.2021).

20. Интернет-ресурс: Itglobal.com // itglobal.com URL: https://itglobal.com/ru-ru/company/glossary/nginx// (дата обращения: 28.11.2021).