

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

СЕВЕРО-ВОСТОЧНЫЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНСТИТУТ ЦИФРОВЫХ ТЕХНОЛОГИЙ И ЭКОНОМИКИ

Кафедра точных и естественных наук

## **ОТЧЕТ**

**Б2.О.01(У) Учебная практика, технологическая**

начало практики 01.07.2022, длительность практики – 2 недели

**Тема «Разработка расширения «Pomodoro-таймер» для браузеров на  
основе Chromium»**

Выполнила:

Руководитель практики:

старший преподаватель  
кафедры ТиЕН  
Марсенич И. А.

---

(оценка, подпись)

Магадан  
2022

## Оглавление

Введение.....	3
1. Создание структуры расширения .....	4
2. Создание файла manifest.json .....	4
3. Создание итогового интерфейса.....	5
4. Разработка логики расширения .....	8
5. Тестирование и отладка расширения .....	10
Заключение .....	11
Список источников .....	12

## Введение

В последнее десятилетие тренд на продуктивность постепенно перерастает в одержимость. Очень важно уметь управлять своим временем, особенно в нынешнюю эпоху постоянных отвлечений, когда все делается с целью обратить на себя внимание. Развитие социальных сетей и повсеместный блоггинг особенно обострили ситуацию с нехваткой времени и отсутствием продуктивности – все больше людей тратят время на непрерывный просмотр контента в течение неограниченного времени.

Помимо виртуальной жизни, отвлечению от выполнения задач и не плодотворности реальной жизни могут способствовать следующие факторы: невозможность сосредоточиться на задаче, или же наоборот, беспрерывное выполнение задачи без отдыха от чего производительность человека оказывается минимальной.

Тайм-менеджмент актуален для всех социальных слоёв населения: от школьников до бизнесменов. В конце 1980-х годов итальянский студент решил проблему рационального распределения своего времени с помощью кухонного таймера в виде помидора. Так и появилась набравшая свою популярность техника «Pomodoro».

Благодаря этой технике вы точно знаете цену своего времени, от чего вырабатывается самодисциплина, минимизируется количество отвлечений, развивается гибкость и скорость мышления в условиях ограничения времени.

Техника «Pomodoro» является гибкой из-за возможности изменения временных интервалов: время работы и отдыха можно уменьшить или увеличить, ориентируясь на личные предпочтения.

**Целью данной работы** является разработка расширения «Pomodoro-таймер» для браузера на движке Chromium с помощью средств HTML, CSS, JavaScript.

**Результатом данной работы** является расширение для Google Chrome, позволяющее пользователю рационально выстроить рабочий процесс, отделив его от личных дел.

Для разработки данного расширения были выполнены следующие шаги:

1. Изучить литературу о JavaScript.
2. Формирования структуры расширения.
3. Описание файла-манифеста manifest.json.
4. Создание итогового интерфейса.
5. Разработка логики расширения и взаимодействия графических элементов друг с другом.
6. Отладка и тестирование разработанного расширения.

Для разработки данного расширения был выбран редактор кода «Visual Studio Code», который имеет большой функционал с работой JavaScript – подсветка синтаксиса, проверкой на ошибки и быстрого выбора команд.

## 1. Создание структуры расширения

Для создания расширения были разработаны следующие файлы:

1. `manifest.json` – данный файл создан для описания расширения и какие разрешения доступны ему;
2. `popup.html` и `popup.css` – данные файлы отвечают за внешний вид расширения. Первый файл отвечает за разметку и структуру расширения открываемого окна. Второй файл создан для определения стиля и графических элементов окна который видит пользователь.
3. `Bootstrap.min.css` – файл с набором шаблонов `html` и `css` для дизайна веб-приложений и сайтов.
4. `Popup.js` – этот скрипт отвечает за прослушивания всех элементов расширения с которыми взаимодействует пользователь. У данного скрипта есть модули который он вызывает:
  - `bootstrap.min.js` – библиотека шаблонов JavaScript для работы с интерфейсом.
5. `background.js` – фоновый скрипт, позволяющий работать расширению, при закрытии `popup.html` и браузера.
6. `Options.html` и `options.css` – файлы отвечают за внешний вид страницы настроек приложения, включает в себя информацию о приложении и настройку времени отдыха и работы.
7. `Options.js` – скрипт, работающий при открытии страницу `options.html`.

## 2. Создание файла `manifest.json`

Для каждого расширения на движке Chromium нужен специальный файл `manifest.json`.

Данный файл описывает расширение, определяет его разрешения и его структуру. Без данного файла браузер не запустит расширение (листинг 1).

## Листинг 1. Содержимое файла manifest.json

```
{
  "background": {
    "persistent": false,
    "scripts": [ "js/background.js" ]
  },
  "browser_action": {
    "default_popup": "popup.html",
    "default_title": "DoroDoro Timer"
  },
  "default_locale": "ru",
  "description": "__MSG_ext_description__",

  "manifest_version": 2,
  "name": "__MSG_ext_name__",
  "options_page": "options.html",
  "permissions": [ "management", "notifications", "storage", "idle",
"activeTab", "\u003Call_urls>" ],
  "version": "1.0.1",
  "web_accessible_resources": [ "img/red/icon48.png",
"img/red/icon128.png" ]
}
```

Сначала были записаны базовые метаданные о расширении, такие как его имя, версия и описание. После чего, в соответствии с задачей, были описаны следующие строки:

- Background – поле, в котором указывается список скриптов, работающих в фоне.
- Browser\_action – поле, в котором указывается html файл, который открывается при запуске расширения.
- Permissions – указывает список разрешений для работы расширения, например, локальное хранилище, работа с вкладками, уведомлениями и тд.

### 3. Создание итогового интерфейса

Разметка страницы rorup.html состоит из следующих блоков:

- Head - в данном блоке располагается название расширения, подключения CSS – стилей и скриптов.
- 3 кнопки, 1 из которых всегда отображается на экране, а 2 другие сменяют друг друга при нажатии.
- 3 контейнера, содержащие в себе часы, минуты и секунды.
- 2 элемента, отображающие текущий промежуток времени (работа или отдых).

Для разработки дизайна интерфейса использовались средства CSS, позволяющие работать с документами, описанными с помощью разметки HTML (листинг 2) (рис. 1).

### Листинг 2. Содержимое файла popup.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>DoroDoro Timer</title>
  <link rel="stylesheet" href="css/bootstrap.min.css">
  <link rel="stylesheet" href="css/popup.css">
</head>
<body>
  <div class="popup-wrapper">
    <p><span id="work-text" class="hidden">Работаем!</span> <span
id="work-minutes"></span> <span class="minutes-text"></span></p>
    <p><span id="break-text" class="hidden">Отдыхаем!</span> <span
id="break-minutes"></span> <span class="minutes-text"></span></p>
    <p>
      <span id="t-hours">--</span>
      <span>:</span>
      <span id="t-minutes">--</span>
      <span>:</span>
      <span id="t-seconds">--</span>
    </p>
    <button class="btn btn-success" id="start">Старт</button>
    <button class="btn btn-danger hidden" id="stop">Стоп</button>
    <button class="btn btn-warning"
id="restart">Перезапустить</button>
    <hr>
    <div class="flex-between">
      <a href="options.html" target="_blank" class="btn btn-light"
id="settings">Настройки</a>
    </div>
  </div>
  <script src="js/popup.js"></script>
</body>
</html>
```

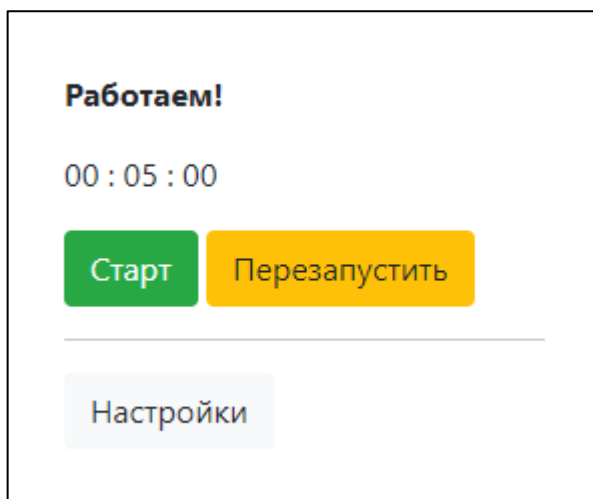


Рис. 1. Интерфейс расширения

Разметка страницы options.html состоит из следующих блоков (рис. 2):

- Кнопка «Сохранить» для сохранения настроек времени работы и отдыха в локальное хранилище.
- Вкладка «Время», в которой с помощью поля input типа range настраивается время работы и отдыха.
- Вкладка «О приложении», в которой описано, как пользоваться расширением и информация о разработчике.

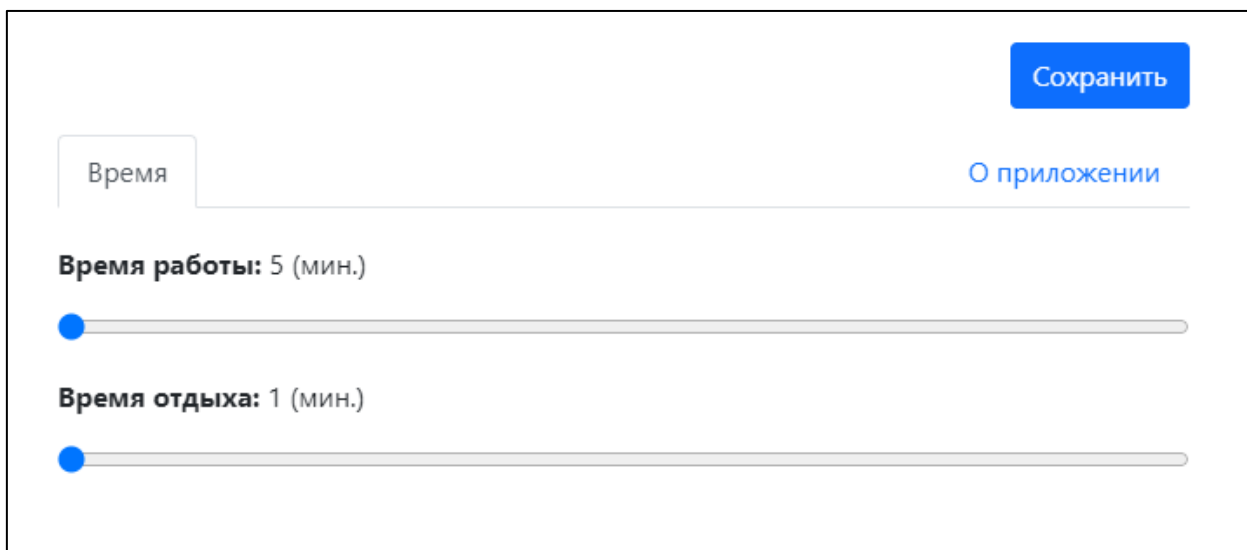


Рис. 2. Интерфейс страницы options.html

## 4. Разработка логики расширения

Перед началом работы расширения, необходимо перейти в настройки, указать время работы и отдыха, после чего сохранить их, нажав на кнопку «Сохранить». При сохранении, данные записываются в локальное хранилище (листинг 2).

### Листинг 3. Фрагмент сохранения времени работы и отдыха в LocalStorage

```
const save = () => {
  const timer = {
    workTime: workTime.value*60,
    breakTime: breakTime.value*60,
    type: 'Work',
    isClockRunning: false
  }
  const json = JSON.stringify(timer);
  localStorage.setItem('DoroDoroTime', json);
  return timer;
};
```

Расширение начинает работу при нажатии на кнопку «Старт», останавливает работу при нажатии на кнопку «Стоп» и начинает работу с самого начала при нажатии на кнопку «Перезапустить». При нажатии определенной кнопки, из popup.js в background.js посылаются соответствующее сообщение (листинг 4).

### Листинг 4. Фрагменты кода с прослушиванием событий нажатий на кнопки «Старт», «Стоп», «Перезапустить»

```
startButton.addEventListener("click", () => {
  isClockRunning = true;
  changeStartIcon(isClockRunning);
  chrome.runtime.sendMessage('Start');
  countdownTimer();
});

stopButton.addEventListener("click", () => {
  isClockRunning = false;
  chrome.runtime.sendMessage('Stop');
  changeStartIcon(isClockRunning);
});

resetButton.addEventListener("click", () => {
  isClockRunning = false;
  type = 'Work';
  time = workTime;
  save();
  chrome.runtime.sendMessage('Reset');
  putTime(time);
});
```

```

changeStartIcon(isClockRunning);
workLabel.classList.remove('hidden');
restLabel.classList.add('hidden');
})

```

Время указанных промежутков переводится из минут в секунды, после чего с помощью метода `setInterval()` с интервалом в 1000 миллисекунд вычитает из полученного времени 1 секунду. Когда время заканчивается, тип промежутка меняется на другое значение и интервал очищается (листинг 5).

**Листинг 5.** Функция, отвечающая за смену значений таймера и уведомление при переходе в следующий промежуток времени

```

const countdownTimer = () => {
  var timer = setInterval(() => {
    if (time <= 0) {

      if (type === 'Work') {
        type = 'Rest';
        time = breakTime;
        var myNotification = new Notification('Давайте
отдохнем!', {body: ""});
      } else if (type === 'Rest') {
        type = 'Work';
        time = workTime;
        var myNotification = new Notification('Пора за
работу!', {body: ""});
      }
      clearInterval(timer);

      countdownTimer();
    } else if (isClockRunning === false) {
      console.log(isClockRunning);
      clearInterval(timer);
    }
    else {
      --time;
      save();
    }
    console.log(time);

  }, 1000);
};

```

## 5. Тестирование и отладка расширения

Для проверки работоспособности расширения, браузер был закрыт в ожидании завершения указанного промежутка времени работы/отдыха.

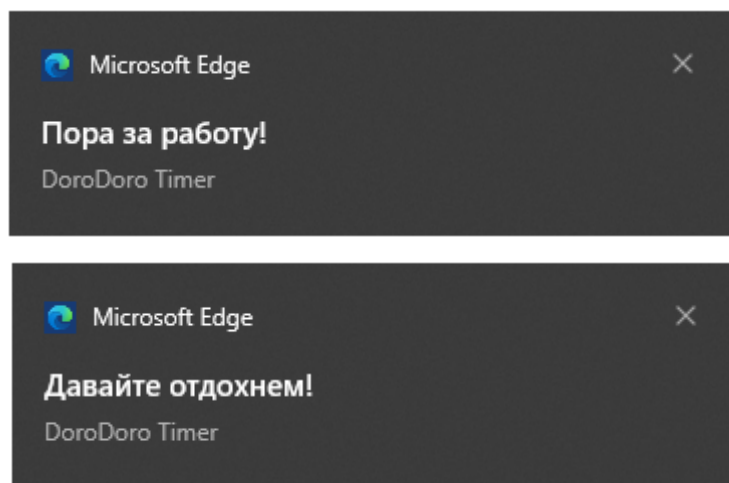


Рис. 3. Уведомления при завершении таймера. Верхнее – завершение отдыха, нижнее – завершение работы

## **Заключение**

Было разработано расширение, позволяющее пользователю рационально выстроить рабочий процесс, разделяя время работы на несколько непродолжительных промежутков с перерывами на отдых.

Расширение «DoroDoro Timer» позволяет пользователю выбрать промежутки работы и отдыха, которые чередуются, до тех пор, пока пользователь не остановит таймер. При завершении одного из промежутков браузер уведомляет пользователя о начале промежутка работы/отдыха.

Все задачи, поставленные в начале практической работы, были выполнены, следовательно, ее цель – разработка расширения для браузеров на движке Chromium «Pomodoro-таймер» с помощью средств HTML, CSS, JavaScript, была выполнена.

## СПИСОК ИСТОЧНИКОВ

1. Современный учебник JavaScript: [Электронный ресурс] URL: <https://learn.javascript.ru/> (дата обращения 01.07.2022).
2. Создание расширений для Google Chrome: [Электронный ресурс] URL: <https://legacy.gitbook.com/book/torus/sozдание-rasshirenii-dlya-google-chrome/details> (дата обращения 01.07.2022).
3. Bootstrap 4: [Электронный ресурс] URL: <https://bootstrap-4.ru/docs/5.0/getting-started/introduction/> (дата обращения 01.07.2022).
4. Chrome Developers: [Электронный ресурс] URL: [https://developer.chrome.com/docs/extensions/mv3/service\\_workers/](https://developer.chrome.com/docs/extensions/mv3/service_workers/) (дата обращения 03.07.2022).
5. Асинхронность в JavaScript: [Электронный ресурс] URL: <https://itchief.ru/javascript/settimeout-setinterval> (дата обращения 03.07.2022).
6. КОД – журнал Яндекс Практикума: [Электронный ресурс] URL: <https://thecode.media/extention/> (дата обращения 03.07.2022).
7. Nuances of programming: [Электронный ресурс] URL: <https://medium.com/nuances-of-programming/%D1%81%D0%BE%D0%B7%D0%B4%D0%B0%D0%B5%D0%BC-%D1%80%D0%B0%D1%81%D1%88%D0%B8%D1%80%D0%B5%D0%BD%D0%B8%D0%B5-%D0%B4%D0%BB%D1%8F-chrome-5933505e50fa> (дата обращения 05.07.2022).
8. Habr.com – Руководство по JavaScript, 1 часть: [Электронный ресурс] URL: <https://habr.com/ru/company/ruvds/blog/429552/> (дата обращения 05.07.2022).
9. CyberForum: [Электронный ресурс] URL: <https://www.cyberforum.ru/javascript-api/thread1217611.html> (дата обращения 05.07.2022).
10. Resources for developers: [Электронный ресурс] URL: <https://developer.mozilla.org/ru/> (дата обращения 05.07.2022).