

# Введение

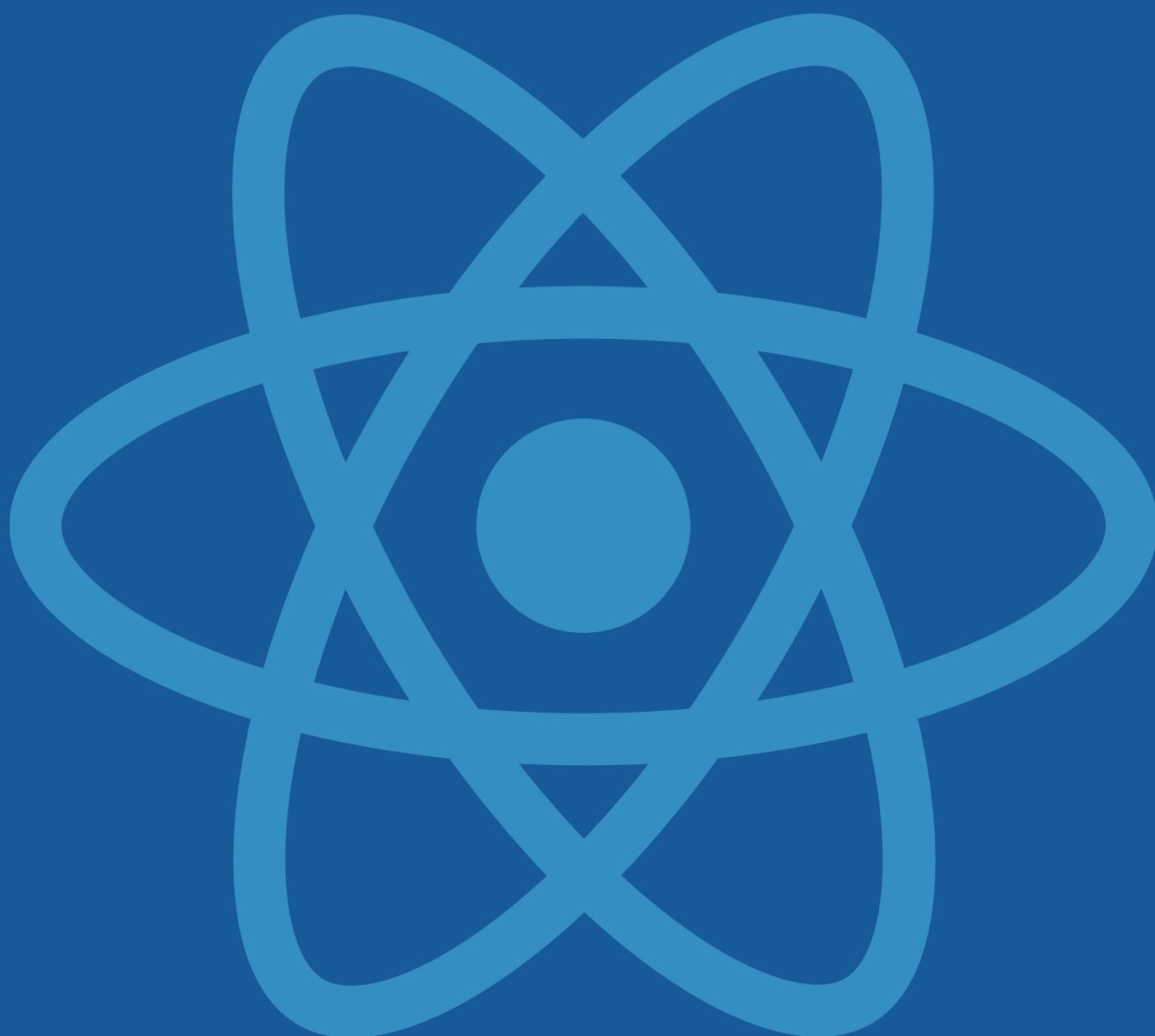
Вы почти закончили обучение и дошли до завершающей части курса: большая часть теории и практики осталась позади. Мы гордимся вашим трудолюбием и упорством! Теперь осталось совсем немного — выполнить и защитить итоговый дипломный проект.

Некоторые считают диплом формальностью: «Я получил знания, а бумажка мне не важна». Но диплом не просто свидетельство. Это ещё один кейс в ваше портфолио, который можно показывать работодателю. Это полезно, даже если прямо сейчас вы не ищете работу: например, для роста внутри компании и для закрепления ваших знаний, полученных в рамках курса.

Поэтому давайте сберёмся и сделаем это!

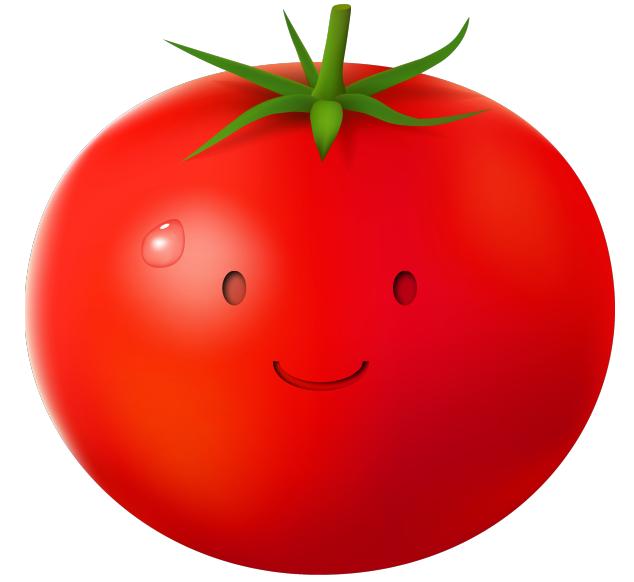


**Будет круто!**



# Тема вашего проекта

Разработка трекера задач  
с таймером по методу  
**«Помодоро» — pomodoro\_box**



Перед вами подробное описание будущего проекта, который станет отличным дополнением вашего портфолио. Здесь есть всё, что вам нужно, чтобы справиться с поставленной задачей.

А именно:

- [Описание задачи.](#)
- [ТЗ на дипломную работу.](#)
- [Описание поведения.](#)
- [Состав продукта.](#)
- [Описание страниц и разделов проекта.](#)
- [Технические требования.](#)
- [Дополнительные задания к проекту \(по желанию\).](#)
- [Формат сдачи материалов и оценивание.](#)
- [Рекомендации по выполнению проекта.](#)
- [Полезные ссылки и ресурсы.](#)

**Последовательно изучите каждую часть.**  
Периодически обращайтесь к актуальным для вас разделам  
в этом документе.

# Описание задачи

Вы работаете в студии, занимающейся разработкой веб-приложений на React. У компании есть постоянный заказчик — школа повышения личной эффективности **Простор**.

Команда вашей студии очень любит работать с этим заказчиком и старается делать всё, чтобы «Простор» был доволен сотрудничеством с вами. Это приносит неплохие бонусы и компании, и самой команде: заказчик увеличивает количество задач, разработчикам платится повышенная премия, а ваша компания зарабатывает больше денег 😊

**От вашей качественной работы выигрывают все!**

Недавно методист Андрей Иванов из «Простора» принял решение создать новый курс по обучению студентов технике «Помодоро». Это требует от вас создания веб-приложения для студентов, реализующего механику этой техники.



**Андрей Иванов**  
Методист



Дизайнер Катя подготовила несколько макетов и UIKit, которые помогут сделать приложение с продуманным и стильным дизайном. В общем, почти всё готово для старта работ 😊



**Катя Миронова**  
Дизайнер



В следующем месяце у заказчика запланирована обширная рекламная кампания, в том числе запуск роликов на ТВ. На курс придёт огромное количество студентов, и нам нужно работающее приложение, чтобы рекламный бюджет не ушёл впустую.

**Сроки запуска рекламы смещать нельзя — это огромные денежные потери!**

Технический писатель уже обновил техническое задание, а дизайнер уже нарисовал макеты. Теперь дело за вами: нужно разобраться в предметной области, а затем выполнить задачи по ТЗ. Заодно проверить себя — всё ли в приложении работает так, как надо.

Как вы поняли, задачи интересные, их много и «Простор» очень рассчитывает, что благодаря вашей работе новый курс получится интересным и полезным.

**Готовы? Поехали!** 

# ТЗ на дипломную работу

Метод «Помодоро» — это техника управления временем, разработанная студентом колледжа Франческо Чирилло в 1980-х годах. В основе техники, названной от итальянского слова «помидор», лежат многочисленные временные интервалы, распределённые в течение дня и обозначенные на кухонном таймере в форме помидора, который Чирилло использовал сам, когда оттачивал свой метод управления временем.

Базовая структура техники «Помодоро», призванной быть простой в применении и в то же время давать большие преимущества с точки зрения производительности, заключается в следующем:

- ➔ Установите в своём списке дел цель или задачу, которую необходимо выполнить.
- ➔ Установите таймер «Помодоро» 
- ➔ Работайте в течение заданного периода времени.
- ➔ Когда таймер зазвонит, отметьте галочкой свой рабочий интервал.
- ➔ После каждого рабочего сеанса делайте 5-минутный перерыв.
- ➔ После четвёртого сеанса сделайте более длительный перерыв — 15–30 минут.
- ➔ Переустановите таймер и начните процесс снова.



# Описание поведения

Пользователь может выполнить на сайте следующие действия:

## 1. Работать со списком задач (добавить, редактировать, удалить, сортировать)

Пользователь может запланировать несколько задач на свой день и для каждой задать примерное количество «помидоров», которое необходимо, чтобы её сделать. Верхняя задача из списка — это текущая задача.

## 2. Работать с таймером (старт, стоп, пауза, продлить, пропустить)

Как только пользователь готов, он запускает таймер. Если его отвлекли, то пользователь останавливает таймер, «помидорка» при этом не засчитывается. Пользователь может поставить таймер на паузу и пропустить «помидорку» или перерыв.

## 3. Просматривать статистику использования таймера

На этой странице отображается статистика по использованию приложения и некоторые полезные метрики. Пользователь может посмотреть столбчатую диаграмму с количеством часов, когда он работал с таймером. Может выбрать неделю, за которую он хочет посмотреть статистику. Может посмотреть дополнительные метрики, такие как:

- **Фокус** (отношение времени работы с таймером ко времени, потраченному на законченные «помидорки»).
- **Время на паузе.**
- **Остановки.**



# Состав продукта

Основная страница с таймером.

Страница со статистикой использования таймера.

Подробный состав — в описании страниц и разделов проекта.



## Описание страниц и разделов проекта

### 1. Основная страница с таймером

Эта страница является основной для приложения. Пользователь проводит большую часть своего времени на ней, поэтому на работу этой страницы нужно обратить максимальное внимание.

**Страница состоит из следующих блоков:**

- Шапка с логотипом pomodoro\_box и ссылкой на страницу статистики.
- Краткая инструкция о работе с приложением.
- Форма для добавления задач.
- Список задач с возможностью редактирования.
- Таймер с кнопками управления.



### 2. Страница со статистикой

#### использования таймера

Это дополнительная страница, чтобы пользователь мог следить за своим прогрессом в освоении техники «Помодоро» и видеть различные дополнительные метрики по использованию таймера.

**Страница состоит из следующих блоков:**

- Шапка с логотипом pomodoro\_box и ссылкой на страницу статистики.
- Столбчатая диаграмма времени использования таймера по дням недели.
- Выбор недели.
- Статистика работы с таймером за сегодня.
- Фокус.
- Время на паузе.
- Количество остановок.

# Технические требования

1. Приложение должно быть написано на React.
2. В работе должен использоваться ESLint.
3. Приложение не должно содержать ошибок, работа должна быть корректной.
4. Техническая реализация: блоки, кнопки, элементы, описанные в ТЗ, должны работать в соответствии с постановкой.
5. Интерфейс должен быть отзывчивым: при возникновении пользовательских ошибок пользователь уведомляется о них.

## Дополнительные задания к проекту. По желанию

Если вы уже выполнили проект по ТЗ, можете попробовать свои силы в дополнительных заданиях. Они расположены по сложности — от более простых к более сложным. По просьбе заказчика их нужно сделать как можно бюджетнее, поэтому мы не привлекаем дизайнера, также для них нет готовой вёрстки — вам нужно будет сделать задачу самостоятельно и в одиночку. 💪😎

### 1. Добавить анимации

Сложность · 1

Приложение выглядит довольно скучно. Заказчик хочет создать у пользователя вай-эффект. Давай попробуем добавить в проект несколько красивых анимаций.

- Уменьшение времени на таймере сопровождается анимацией.
- При добавлении новой задачи она появляется с анимацией.
- После остановки таймера выполненная задача исчезает с анимацией.

### 2. Добавить тёмную тему

Сложность · 2

У нас много пользователей-программистов, которые привыкли работать по ночам и любят использовать тёмную тему. Сможем добавить?

- Стилизовать приложение в «тёмную тему», инвертировав цвета.
- Добавить в шапку переключатель между темами.
- Запоминать последнюю выбранную тему между перезагрузками страницы.

### 3. Добавить уведомления

Сложность • 3

Наши пользователи иногда пропускают таймер из-за того, что приложение недостаточно их уведомляет об окончании работы или отдыха. Есть пара идей!

- По истечении времени таймера **работы** раздаётся сигнал и появляется браузерное уведомление.
- При окончании таймера **перерыва** раздаётся сигнал и показывается браузерное уведомление.

### 4. Добавить настройки таймера

Сложность • 4

Заказчик попросил, чтобы пользователи могли настроить таймер под себя. Нужно добавить раздел «**Настройки**» и вынести туда несколько параметров.

- Можно изменять продолжительность «помидора».
- Можно изменять продолжительность короткого/длинного перерыва.
- Можно изменять частоту длинных перерывов.
- Можно выключить уведомления.

### 5. Сохранять состояние приложения при перезагрузке страницы

Сложность • 5

Пользователям не очень нравится, что при перезагрузке таймера они теряют прогресс и статистику. Давай попробуем исправить это.

- Сохранять состояние приложения в LocalStorage или IndexedDB.
- При перезагрузке приложения восстанавливать предыдущее сохранённое состояние.



# Формат сдачи материалов и оценивание

Результат проведённой работы опубликуйте на [GitHub Pages](#) или на учебном [GitLab](#).

Также для проверки диплома передайте проверяющим исходный код приложения.

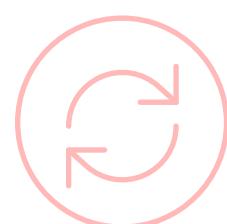
Если вы сделали дополнительные задания, то сообщите об этом своему преподавателю.

## Готовый проект мы будем проверять по следующим критериям:

- Соответствие техническому заданию. Элементы должны работать так, как они описаны в документе.
- Приложение собирается и запускается.
- Отсутствие ошибок во фронтовой части приложения (проверяем в консоли).
- Отсутствие ошибок в бизнес-логике. Пользователь должен пройти до конца по всем сценариям.
- Usability приложения. Субъективный критерий с точки зрения пользовательского опыта.
- Отзывчивость интерфейса. При наличии ошибок при вводе данных пользователь получает соответствующее уведомление.

## Проект возвращается на доработку, если:

- Приложение не собирается или не запускается.
- Приложение не соответствует техническому заданию.
- В консоли приложения есть ошибки.
- Приложение аварийно завершается.



### \* Подсказки и советы

- Не тратьте много времени на настройку webpack или любых других сборщиков. Возьмите готовый стартер, например, Create React App.
- Вы можете использовать внешний контейнер состояния, чтобы не пробрасывать стейт между компонентами. Храните в нём только основное состояние приложения.
- Вы можете воспользоваться расширениями для внешних контейнеров состояний, которые умеют сохранять данные в LocalStorage.
- Вы можете воспользоваться роутингом для переключения между таймером и статистикой.

# Рекомендации по выполнению проекта

## План работы над курсовой:

- Внимательно изучите документацию. Убедитесь, что вам всё понятно. Это первый и очень важный шаг в начале работы над любым проектом. Если у вас возникают вопросы, фиксируйте их для себя и обязательно задавайте своему куратору дипломного проекта.
- Чтобы прийти к результату и выполнить проект целиком, рекомендуем разбить работу над ним на части. Например, одна часть — одна страница: список товаров или карточка товара. Иными словами, готовая часть — это то, что можно показать заказчику как выполненную работу. В вашем случае роль заказчика исполняет куратор.
- Составьте план работы по датам. Исходя из нашего опыта, продуктивнее выделять на дипломный проект по 2-3 часа несколько дней в неделю, чем делать этот же объём за один подход. Рекомендуем придерживаться этого графика и обязательно выделять время для отдыха.
- Отмечайте свой прогресс по мере выполнения плана. Это полезно по нескольким причинам: во-первых, вы будете держать ритм, во-вторых — контролировать ситуацию. И самое главное: каждый выполненный этап — это ваша маленькая победа: чем больше таких побед вы будете замечать, тем большее удовольствие от выполненного проекта получите в итоге.
- Прежде чем отправлять каждую часть проекта на проверку преподавателю (а в реальной работе — тестировщику), всегда уделяйте время проверке. Мы рекомендуем вам протестировать каждый сценарий (смотрите описание задачи) отдельно, а также проверить весь проект целиком, когда вы его закончите. Идеально будет, если между завершением работы и проверкой пройдёт некоторое время — это позволит вам отключиться и в результате посмотреть на работу свежим взглядом.
- Качество продукта — это не метрика, а личное отношение разработчика. Попросите нескольких друзей или коллег протестировать проект вместе с вами. Обращайте внимание на поведение, просите вслух проговаривать мысли, действия и ощущения. Это поможет вам понять, где у пользователя могут возникнуть проблемы, и сделать действительно удобный интерфейс. Задавайте наводящие вопросы. Вы можете заранее подготовить для этого чек-лист, отталкиваясь от функциональных требований.

# ★ Полезные ссылки и ресурсы ★

[Документация React.js](#)

[Описание техники Помодоро](#)

[Состояния в Functional Components \(модуль 4\)](#)

[Подключаем React Router \(модуль 14\)](#)



**У вас всё получится!**