Техническое задание на личный проект «Дела в порядке» для базового PHP-интенсива

Оглавление

[Общая концепция 2](#_Toc480378606)

[Основные сценарии использования сайта: 2](#_Toc480378607)

[Техническое описание 2](#_Toc480378608)

[Описание процессов 3](#_Toc480378609)

[Регистрация нового аккаунта 3](#_Toc480378610)

[Создание новой задачи 3](#_Toc480378611)

[Авторизация на сайте 4](#_Toc480378612)

[Добавление проекта 4](#_Toc480378613)

[Уведомления о предстоящих задачах 5](#_Toc480378614)

[Поиск 5](#_Toc480378615)

[Выполнение задачи 5](#_Toc480378616)

[Список всех экранов 6](#_Toc480378617)

[Главная страница 6](#_Toc480378618)

[Регистрация аккаунта 7](#_Toc480378619)

[Авторизация на сайте 7](#_Toc480378620)

[Добавление проекта 7](#_Toc480378621)

[Добавление задачи 7](#_Toc480378622)

[Основные сущности 9](#_Toc480378623)

[Проект 9](#_Toc480378624)

[Задача 9](#_Toc480378625)

[Пользователь 9](#_Toc480378626)

[Роли пользователей 10](#_Toc480378627)

# Общая концепция

«Дела в порядке» - это веб приложение для удобного ведения списка дел.   
Сервис помогает пользователям не забывать о предстоящих важных событиях и задачах.

После создания аккаунта, пользователь может начать вносить свои дела, деля их по проектам и указывая сроки.

## Основные сценарии использования сайта:

* Создание проектов
* Добавление новых задач с привязкой к проекту и дате
* Просмотр дел на ближайшие дни
* Получение уведомлений о предстоящих задачах

# Техническое описание

Для разработки сайта предлагается уже готовая верстка, от программиста требуется лишь написать так называемый бэкенд сайта, т.е. сделать сайт динамическим, реализовать возможности по добавлению, просмотру задач и проектов.

Разработка бэкенда должна вестись на языке программирования PHP 7 и выше, база данных – MySQL 5.7 и выше.  
Не предполагается использование языка JavaScript для клиентского программирования.

При разработке схемы базы данных необходимо принимать во внимание какие сущности необходимо хранить, их поля и возможные связи друг с другом.

# Описание процессов

Процесс – это автономная операция, выполняющая полезное действие. К примеру: добавление нового комментария или регистрация аккаунта.

## Регистрация нового аккаунта

Выполняется после заполнения формы на странице «Регистрация аккаунта».  
Последовательность действий:

1. Проверить, что отправлена форма
2. Убедиться, что заполнены все обязательные поля
3. Проверить, что указанный email уже не используется другим пользователем
4. Если есть ошибки заполнения формы, то сохранить их в отдельном массиве
5. Если ошибок нет, то сохранить данные формы в таблице пользователей

Если были допущены ошибки (не заполнены все поля, email занят и т.д.), то не сохранять данные, а показать ошибки в форме под соотвествующими полями.

Если данные были сохранены успешно, то переадресовать пользователя на страницу входа, где показать сообщение над формой «Теперь вы можете войти, используя свой email и пароль».

**Примечания**

Необходимо проверять, что значение из поля “email” действительно является валидным E-mail адресом. Для этих целей использовать встроенную функцию **filter\_var** и фильтр **FILTER\_VALIDATE\_EMAIL**

Для хранения пароля в БД его предварительно нужно обработать встроенной функцией **password\_hash**

## Создание новой задачи

Страница доступна только аутентифицированным пользователям.

Выполняется после заполнения формы на странице «Добавление задачи».   
Последовательность действий:

1. Проверить, что отправлена форма
2. Убедиться, что заполнены все обязательные поля
3. Выполнить все проверки
4. Если есть ошибки заполнения формы, то сохранить их в отдельном массиве
5. Если ошибок нет, то сохранить новую задачу (учитывая выбранный проект)
6. Если к задаче был прикреплен файл, то перенести его в публичную директорию и сохранить ссылку

При успешном сохранении формы переадресовывать пользователя на главную страницу.

Список проверок

**Проверка даты**Содержимое поля «Дата выполнения» должно быть датой в формате «ДД.ММ.ГГГГ» **либо** датой в «человеческом формате» (см. ниже). Проверять, что указанная дата больше или равна текущей дате

Описание обработки даты в человеческом формате:  
система должна принимать и корректно обрабатывать следующие форматы дат:

* «завтра», «послезавтра»
* «сегодня»
* «<день недели>» - вторник, среда, четверг и т.д.
* «завтра в <время HH:SS>», «<день недели>в <время HH:SS>»

**Проверка проекта**

Если был выбран проект, то проверять, что идентификатор выбранного проекта ссылается на реально существующий проект

## Авторизация на сайте

Выполняется после заполнения формы на странице «Авторизация на сайте».  
Последовательность действий:

1. Проверить, что отправлена форма
2. Убедиться, что заполнены все обязательные поля
3. Если есть ошибки заполнения формы, то сохранить их в отдельном массиве
4. Найти в БД пользователя с переданным email
5. Проверить, что переданный в форме пароль совпадает с сохраненным
6. Если пользователь найден и пароли совпадают, то открыть новую сессию, в которую записать идентификатор пользователя

После успешной авторизации пользователь попадает на главную страницу.

**Примечания**

Для проверки пароля использовать встроенную функцию **password\_verify**

## Добавление проекта

Выполняется после заполнения формы на главной странице.  
Последовательность действий:

1. Проверить, что пользователь залогинен
2. Убедиться, что заполнено поле «название проекта»
3. Добавить новый проект в таблицу проектов

## Уведомления о предстоящих задачах

Сервис должен уметь отправлять пользователю email уведомления о предстоящих задачах.   
Уведомление должно отправляться на почтовый адрес, указанный при регистрации, за один час до наступления времени выполнения задачи.  
**Тема письма:** Уведомление от сервиса «Дела в порядке»  
**Текст письма:** Уважаемый, <имя пользователя>. У вас запланирована задача <имя задачи> на <время задачи>

Вся логика работы должна содержаться в отдельном файле сценария, который будет автоматически вызываться операционной системой в начале каждого нового часа.

Алгоритм работы:

1. Выполнить SQL запрос на получение всех невыполненных задач у которых дата выполнения больше или равна текущей дате/времени минус один час
2. Если у одного пользователя задач больше, чем одна, то объединить их названия и даты в один список, который будет вставлен в уведомление
3. Сформировать для каждого найденного пользователя письмо, указав тему и текст сообщения
4. Отправить каждому найденному пользователю это письмо

## Поиск

Выполняется после отправки формы поиска с главной страницы.

Последовательность действий:

1. Проверить, что была отправлена форма
2. Получить текст из поля поиска и убрать пробелы функцией **trim**
3. Проверить, что получившаяся строка не пустая
4. Выполнить поиск в таблице задач по полю «имя задачи», используя SQL оператор «LIKE»
5. Показать результаты поиска на соответствующей странице

## Выполнение задачи

Выполняется после отметки пользователем задачи как выполненной со страницы списка задач.

Последовательность действий:

1. Получить из базы задачу по её идентификатору
2. Составить и выполнить SQL запрос, который инвертирует статус задачи (выполнена -> не выполнена, не выполнена -> выполнена)
3. Переадресация на страницу списка задач

# Список всех экранов

Сайт состоит из следующих экранов:

* Главная страница
* Регистрация аккаунта
* Авторизация на сайте
* Добавление проекта (модальное окно)
* Добавление задачи (модальное окно)

## Главная страница

Страница делится на две колонки в соотношении 1/3

Левая часть состоит из блока со списком проектов. Первым в этом списке идет пункт «Входящие», который предназначен для хранения задач без проекта.  
Под списком проектов находится кнопка «Добавить проект», открывающая модальное окно добавления проекта.

Правая часть относится к просмотру и управлению задачами. Состоит последовательно из следующих элементов:

* Поиск задачи – поисковая строка, выполняющая операцию поиска по имени задачи
* Блок фильтров – блок, состоящий из ссылок для быстрой фильтрации задач:
  + Все задачи – показывает все задачи в выбранном проекте
  + Повестка дня – показывает все задачи на сегодня
  + Завтра – показывает все задачи на завтра
  + Просроченные – показывает все задачи, которые не были выполнены и у которых истек срок
* Блок задач – основной блок, который содержит список всех невыполненных задач в выбранном представлении.   
  Каждая задача в блоке состоит из таких элементов (слева направо):
  + Чекбокс – клик по чекбоксу вызывает операцию выполнения задачи
  + Название задачи
  + Приложенный файл (если есть) – иконка-ссылка, ведущая на скачивание файла
  + Срок выполнения (если он указан)
  + Кнопка-ссылка, вызывающая всплывающее меню контекстных действий.  
    Меню содержит пункты: «Выполнить», «Удалить», «Дублировать»

Под блоком задач должна быть ссылка «Показать выполненные», клик по которой обновляет страницу и дополняет список в выбранном представлении выполненными задачами.

* Кнопка добавления задачи – кнопка, которая открывает модальное окно добавления новой задачи

## Регистрация аккаунта

Чтобы пользователь имел возможность пользоваться сайтом и управлять своими задачами, ему необходимо пройти процедуру регистрации на этой странице.

Форма состоит из трех обязательных полей и кнопки «регистрация». Форма отправляется методом «POST».

Пользователю надлежит заполнить все обязательные поля, указав свой действительный email, пароль для входа на сайт, своё имя.

После заполнения формы, пользователь нажимает кнопку «*Регистрация*» для отправки данных формы на сервер.   
Сама процедура регистрации описана в отдельном разделе – «*Регистрация нового пользователя*».

Если по итогам выполнения процесса регистрации возникли ошибки заполнения формы, то эти ошибки должны быть показаны красным текстом под необходимыми полями, а над самой формой появится сообщение «*Пожалуйста, исправьте ошибки в форме*».

## Авторизация на сайте

Все поля, кроме галочки «*запомнить меня*» обязательны к заполнению.

Процедура авторизации описана в отдельном разделе – «Авторизация пользователя».

Если по итогам выполнения процесса авторизаци возникли ошибки заполнения формы, то эти ошибки должны быть показаны красным текстом под необходимыми полями, а над самой формой появится сообщение «*Пожалуйста, исправьте ошибки в форме*».

Если пользователь ввел неверные данные аккаунта (несуществующий пользователь или неверный пароль), то в сообщении над формой должен быть текст «*Вы ввели неверный email/пароль*».

## Добавление проекта

Модальное окно, содержит только одно (обязательное) поле для указания имени нового проекта и кнопку отправки.

## Добавление задачи

Модальное окно, содержит форму из трех полей и кнопку «Добавить».

|  |  |  |
| --- | --- | --- |
| Лейбл | Тип поля | Обязательность |
| Название | Простое текстовое поле | **Да** |
| Дата выполнения | Простое текстовое поле | **Нет** |
| Файл | Поле типа «файл» | **Нет** |

Под полем «Дата выполнения» размещается подсказка с описание формата даты. Можно сделать всплывающей подсказкой при наведении на иконку.

После заполнения формы, пользователь нажимает кнопку «*Добавить*» для отправки данных формы на сервер.   
Сама процедура добавления описана в отдельном разделе – «*Добавление задачи*».

Если по итогам выполнения процесса отправки возникли ошибки заполнения формы, то эти ошибки должны быть показаны красным текстом под необходимыми полями, а над самой формой появится сообщение «*Пожалуйста, исправьте ошибки в форме*».

# Основные сущности

Список всех сущностей

* Проект
* Задача
* Пользователь

## Проект

Состоит только из названия. Каждая задача может быть привязана к одному из проектов.

## Задача

Центральная сущность всего сайта.  
Поля:

* **дата создания** – дата и время, когда задача была создана
* **дата выполнения** – дата и время, когда задача была выполнена
* **название** – задается пользователем
* **файл** – ссылка на файл, загруженный пользователем
* **срок** – дата и время, до которого задача должна быть выполнена

Связи:

* **автор** – пользователь, создавший задачу
* **проект** – проект, к которому принадлежит задача

## Пользователь

Представляет зарегистрированного пользователя.

Поля:

* **дата регистрации** – дата и время, когда этот пользователь завел аккаунт
* **email**
* **имя**
* **пароль** – хэшированный пароль пользователя
* **контакты** – контактная информация для связи с пользователем

Связи:

* **созданные задачи**
* **созданные проекты**

# Роли пользователей

Сайт могут использовать только зарегистрированные пользователи.

Анонимный пользователь всегда видит только страницу входа на сайт.