**ТЗ ЭМШ**

В данном документе ниже будет прописано тз веб-приложения для ЭМШ. Функционал реализации разделе на смысловые пункты.

Обозначения типов пользователей:

Все – все сущности пользователей

Пользователь – неавторизованный пользователь

Ученик – учащийся ЭМШ

Бывший ученик – отчисленный ученик ЭМШ

Преподаватель – преподаватель, одобренный админом

Админ – супер-админ, всемогущая сущность

Новостник – преподаватель с ролью создания новостей, выданный админом

Изменение и управление типами пользователей доступно из сущности Админ. У Админа есть доступ ко всему функционалу.

1. НОВОСТИ
   1. Показ новостей (Доступ: все)

Получение всех новостей: список сущностей Новость. Сущность Новость – **дата публикации** (date) в классическом формате даты – 2024-11-20 (формат такой, потому что он будет отображен в url на фронте), **название** (title) – строка, **текст новости** (description) – строка, большой блок текст, **фотография новости** (photo) – 1 фотография формата .png, .jpeg, .jpg, .gif. API функции – /news.

* 1. Показ карточек новостей и самих новостей реализовано только на фронте
  2. Создание новостей (Доступ: новостник)

Создание сущности Новость:

Title – обязательно

Description – обязательно

Date – автоматически

Photo – не обязательно

API - /news/create

1. КУРСЫ
   1. Показ всех курсов (Доступ: все)

Получение всех курсов сущности Курс. Сущность Курс, описана на бэке (я потом посмотрю). API – /courses. Присылаются все сущности Курсов. Показываются курсы по годам (присабачить год к данным, я буду на бэк отправлять, на фронте сам отображу)

* 1. Показ моих курсов (Доступ: ученик, бывший ученик, преподаватель)

Получение и отображение всех курсов в виде списка. API – /courses/{userId}. Для фронта: при нажатии на курс с краткой инфой переходим на страницу курса, там есть описание курса, каждого занятия, 1 из 3 направлений, к которому принадлежит курс и т.д. Показываются курсы по годам (присабачить год к данным, я буду на бэк отправлять, на фронте сам отображу)**.**

* 1. Расписание (Доступ: ученик, преподаватель)

Показ расписания с текущего момента до конца следующей недели. Присылается объект: {currentWeek: [], nextWeek: []}, массива – списки курсов. Единица списка имеет формат: название курса (name), дата проведения (date), время (time), аудитория (classroom), вид курса: зачетный или нет (credit = true или false соответственно). API – /schedule/{userId}

* 1. Ведомость (Доступ: ученик, бывший ученик, преподаватель)

Показ ведомости по всем курсам, которые есть. API – /marks/pupil/{userId} и /marks/teacher/{userId} для получения ведомости всех видов учеников и преподавателей соответственно. Для всех видов учеников: название курса, таблица ведомости (потому что они уникальны и формулы), название курса2 и т.д. Формат полученных данных: [{courseName: [{date, mark, weight}, …]}, {…}]. Если оценки нет, присылается -1.

Для преподавателей: Название курса, таблица ведомости с учениками и датами в шапках. Для каждой отдельной ячейки таблицы можно выбрать вес оценки (из сущности Курс) и поставить оценку 0 – +бесконечность. Таких курсов столько, сколько ведет преподаватель – сколько ему добавлено. Формат аналогичен ученикам за некоторыми отличиями:

[{courseName: [{pupilName: [{date, mark, weight}, …]}, {…}], courseName2: […]}]

Показываются ведомость по годам (присабачить год к данным, я буду на бэк отправлять, на фронте сам отображу).

* 1. Добавление курса (Доступ: админ)

Указывание всех полей курса. API – /course/add. Указанные преподаватели по ФИО ищутся в бд, им в Мои курсы добавляется соответствующий курс.

1. СОБЫТИЯ (Доступ: все, кроме функции Конкурс научных работ, там доступ: ученики, преподаватели)
   1. ПОКА ЧТО ПОХУЙ
2. ОБ ЭМШ (Доступ: все)
   1. ПОКА ЧТО ПОХУЙ
3. РЕДАКТИРОВАНИЕ ПРОФИЛЯ
   1. ПОКА ЧТО ПОХУЙ
4. СУЩНОСТЬ АДМИН (доп функционал)
   1. Одобрение заявок на регистрацию от учеников, преподавателей

На страницу отображается ФИО человека, статус (ученик / преподаватель), рядом кнопки: принять и отклонить. При нажатии на всю строку, всплывает окно с данными всей формы регистрации. API – /users/add. При нажатии принять отправляется функция API – /users/add/{userData} с полной инфой о пользователе. При отклонении: API – /users/reject/{userData}

* 1. Назначение ролей преподавателям

Список преподавателей (ФИО) и несколько тоглов: новостник, ключеносец и т.д. Новостник добавляет возможность добавлять новости API – /teachers/role/news/{userId}. Ключеносец и т.д. API – /teachers/role/AA/{userId}, где вместо AA прописана роль. Если роль выключается, отправляется запрос вида API – /teachers/role/AA/reject/{userId}, где AA – роль. У преподавателя в профиле появляется должность, отображается вместе с фио на сайте. Если роли нет, на фронт присылается пустая строка.

* 1. Отчисления ученика

API – /pupils/reject/{userId}. У ученика становится недоступен функционал Расписание.