

## Тестовое задание Yii 2.0

Необходимо разработать приложение, реализующее REST API по управлению заметками пользователей.

В разработке требуется использовать **Yii Framework** версии **2.0**. Ввод и вывод данных выполняется в формате **JSON**. Тип используемой базы данных значения не имеет.

Необходимо спроектировать и реализовать API, удовлетворяющее нижеследующим условиям.

Аутентификация пользователей в API должна быть выполнена посредством Bearer Access Token.

Для упрощения, в базе данных уже должны иметься два пользователя (например, *user1* и *user2*) с зафиксированными Access Tokens.

**! Регистрацию и управление пользователями делать *не нужно!***

Каждая заметка должна отображать во всех end points, кроме просмотра:

- заголовок;
- дату публикации;
- информацию об авторе заметки.

Каждая заметка должна отображать в end point просмотра заметки:

- заголовок;
- текст;
- дату публикации;
- информацию об авторе заметки.

Заметки могут иметь дату публикации в будущем; в этом случае до момента наступления времени публикации они выводятся только их авторам, но не другим пользователям.

Заметки должны иметь ограничение по времени на изменение (или удаление) в течение 24 часов с даты создания. Заметки, созданные более 24 часов назад, не могут быть изменены или удалены.

Заметки могут редактироваться или удаляться только автором заметки.

На момент запуска приложение должно содержать для каждого пользователя как минимум 4 заметки:

- 1 заметка с датой создания и публикации в далеком прошлом (т.е. более 24 ч.);
- 2 заметки с текущей датой создания и публикации;
- 1 заметка с датой публикации в далеком будущем (т.е. более 24 ч.).

API end points для заметок должны предусматривать:

- список заметок с страничным выводом;
- просмотр заметки с указанным идентификатором;
- создание заметки;
- изменение заметки с указанным идентификатором;
- удаление заметки с указанным идентификатором.

End point списка заметок:

- доступен всем пользователям, в том числе не аутентифицированным;
- выводит заметки, отсортированные по дате публикации в порядке ее убывания; в случае совпадения даты публикаций, первой должна быть выведена заметка, которая была добавлена ранее;

- выводит 5 заметок на каждой странице, изменение этого параметра пользователем не разрешается; номер страницы для вывода указывается GET-параметром **p**;
- выводит информацию о количестве заметок, количестве страниц и номере текущей страницы как параметры ответа **count**, **pageCount** и **currentPage** соответственно.

End point просмотра заметки:

- доступен всем пользователям, в том числе не аутентифицированным;
- не позволяет просматривать заметки с датой публикации в будущем и заметки, которые были удалены.

End point создания заметки:

- доступен только для аутентифицированных пользователей;
- принимает на вход заголовок, текст и дату публикации заметки.

End point изменения заметки:

- доступен только для аутентифицированных пользователей;
- принимает на вход заголовок, текст и дату публикации заметки;
- позволяет изменять заметки только их авторам;
- позволяет изменять только те заметки, которые были созданы менее 24 часов назад.

End point удаления заметки:

- доступен только для аутентифицированных пользователей;
- позволяет удалять заметки только их авторам;
- позволяет удалять только те заметки, которые были созданы менее 24 часов назад.

And that's all she wrote :)

***PS. Большим плюсом будут присутствующие приемочные тесты.***