Основные сущности: журнал, выпуск, статья, издательство, категория (тематика), подписка, автор.

Описание свойств:

* многоязычное - имеет разные значения на разных языках.
* множественное - имеет несколько значений
* обязательное - обязательно для заполнения
* уникальное - не может быть двух одинаковых значений
* логическое - булево, да/нет
* привязка к ... - внешний ключ, привязка к другой сущности/элементу/записи
* html - содержит форматированный текст

**Журнал.**

Свойства (поля):

* название - многоязычное, обязательное
* активность - логическое, аналог вкл/выкл, неактивные элементы как будто не существуют
* символьные код - обязательное, уникальное, на латинице, используется для формирования ссылки, мб многоязычное
* издательство - множественное, обязательное, привязка к Издательство
* категория - множественное, привязка к Категория
* ISSN - международный стандартный серийный номер, строка
* входит в перечень ВАК - html либо множественная строка
* картинка - изображение, многоязычное
* описание - html, многоязычное
* картинка для анонса - превью изображение, многоязычное, если не задано то может быть сгенерировано из "Картинка"
* описание для анонса - краткое описание, html, многоязычное
* блок Контакты (все поля многоязычные): Главный редактор, Телефон, E-mail, Формат, Объем, Периодичность, Сайт, Editor in chief, О редакторе (html), Контакты (html)
* Редсовет - многоязычное, html
* Указатель статей - многоязычное, html
* Рубрики - многоязычное, html
* Порядок рецензирования - многоязычное, html
* Правила предоставления статей - многоязычное, html

**Подписка.**

Информация о подписке на журнал. Пользователи, купившие подписку на журнал, имеют доступ к чтению этого журнала на определенный период времени.

Алгоритм работы подписки:

Пользователь выбирает тип подписки (печатная\электронная), месяц начала подписки (в конкретном году, например, ноябрь 2018), срок подписки (на сколько месяцев).

Алгоритм работы подписки.

**Срок выхода журналов, газет и справочников.**

Выходят:

А) Раз в полгода

Б) Раз в три месяца

В) Раз в 2 месяца

Г) Раз в месяц

Д) 2 раза в месяц

Е) 1 раз за все время

Бывают случаи, когда раньше журнал выходил раз в месяц, а со следующего полугодия начинает выходить раз в 2 месяца, такие моменты необходимо учитывать.

**Политика формирования цен:**

Цены устанавливаются раз в полгода перед подписной компанией на следующее полугодие

Цена бывает на печатную версию и на электронную версию.

Цена устанавливается за 1 номер журнала, за полгода и бывают случаи, когда выставляется за годовую подписку (возможно со скидкой)

При покупке одного, двух, трех и тд (но не на полугодие) номеров журнала цена формируется из стоимости одного выпуска, умноженного на кол-во журналов

При покупке журналов (газет) срок выхода 2 раза в месяц на данный момент мы принимаем оплату за месяц минимум, но это не будет касаться архивных выпусков, архивные выпуски будут продаваться по отдельности.

В дальнейшем планируется вводить комплект печатная версия + электронная версия.

Также в дальнейшем при покупке электронной версии планируется вводить количество доступов, которые будут распространяться со скидкой

Также необходимо учесть, что покупка печатных версий может быть совершена только до 20 числа предподписного месяца.

**Варианты формирования цены:**

* Журнал покупается в текущем полугодии. Цена будет текущего полугодия
* Журнал покупается, захватывая текущее полугодие и следующее полугодие
  + Если цены на следующее полугодие уже установлены, то подписка делится на 2 позиции и оплачивается по ценам из первого полугодия и из второго полугодия соответственно
  + Если цены на следующее полугодие еще не установлены, то подписка оформляется по ценам текущего полугодия
* Журнал покупается на следующее полугодие
  + Если цены на следующее полугодие уже установлены, то подписка оформляется по ценам следующего полугодия
  + Если цены еще не установлены, то подписка оформляется по ценам текущего полугодия
* Покупается архивный журнал. Цена берется за один номер того полугодия, когда он выходил (имеется ввиду тот год и то полугодие в том году, например, 1 полугодие 2015 года). Данный журнал можно купить только в электронной версии.

Свойства:

* язык - выбор языка, для которого доступна подписка
* журнал - обязательное, привязка к Журнал
* активность - логическое, аналог вкл/выкл
* тип - обязательное, список\select выбор из печатная\электронная
* год - обязательное
* полугодие - обязательное, 1е или 2е
* периодичность - с какой периодичностью будут выходить выпуски в выбранном полугодии (2 раза в месяц, 1 раз в месяц, 1 раз в 2 месяца, 1 раз в 3 месяца, 1 раз в полгода), обязательное
* цена за выпуск - обязательное
* цена за полугодие - например, может быть дешевле купить сразу на полгода
* цена за годовую подписку - если задано, то будет доступна подписка на год по этой цене

**Выпуск.**

Выпуски (номера/части) журнала.

Электронную версию выпуска можно читать на сайте. Для этого загружается архив с файлами (html, картинки, css и т.п.). Для доступа к читалке необходимо подтвердить устройство. Если пользователь открывает читалку, и у него не подтверждено устройство, то на почту отправляется код подтверждения, пользователь вводит этот код на сайте, и устройство запоминается как подтвержденное (например, запоминается в куках браузера). Максимум у пользователя может быть 2 активированных устройства. Время жизни активированного устройства - 1 неделя, т.е. через неделю нужно будет повторно активировать устройство. Если пользователь открывает читалку с неактивированного устройства открывает читалку, и уже исчерпан лимит активаций, то на почту отправляется ссылка для сброса активированных устройств.

Одновременно можно читать выпуски только с одного устройства, в том числе нельзя одновременно читать разные выпуски с разных устройств. Если пользователь открывает читалку, но она уже открыта на другом устройстве, то выводится соответствующее сообщение об ошибке.

В читалке есть содержание (собирается из заголовков читалки), закладки, библиотека (список доступных выпусков для читалки).

При загрузке архива с читалкой он парсится, и автоматически создаются статьи этого выпуска. Один html файл - это один выпуск.

Свойства:

* название - обязательное, многоязычное
* активность - логическое, аналог вкл/выкл
* дата начала активности - день выхода, с которого он доступен.
* журнал - обязательное, привязка к Журнал
* номер - номер выпуска журнала
* цена на печатную версию - используется при покупке выпуска
* цена на электронную версию - используется при покупке выпуска
* картинка - изображение, многоязычное
* описание - html, многоязычное
* картинка для анонса - превью изображение, многоязычное, если не задано то может быть сгенерировано из "Картинка"
* описание для анонса - краткое описание, html, многоязычное
* промо - логическое, доступен ли выпуск бесплатно
* цена статей - стоимость одной статьи этого выпуска, используется при генерации статей из читалки

**Статья.**

Часть выпуска, либо обособленная статья (если не задан выпуск).

Свойства:

* название - обязательное, многоязычное
* активность - логическое, аналог вкл/выкл
* дата начала активности - день выхода, по аналогии с выпуском
* символьный код - обязательное, уникальное, используется для ссылки
* сортировка - число, используется для порядка вывода
* выпуск - привязка к Выпуск
* закрепить - логическое, используется для приоритета вывода
* ограничение контента - требование, которое необходимо выполнить для чтения статьи (нет, регистрация на сайте, покупка/подписка)
* ключевые слова - теги, множественное
* авторы - множественное, привязка к Автор
* категории - множественное, привязка к Категория
* картинка - изображение, многоязычное
* описание - html, многоязычное, основной текст статьи
* картинка для анонса - превью изображение, многоязычное, если не задано то может быть сгенерировано из "Картинка"
* описание для анонса - краткое описание, html, многоязычное
* список литературы - html, многоязычное
* цена - стоимость статьи

**Категория.**

Свойства:

* название - обязательное, многоязычное
* активность - логическое
* символьный код - обязательное, многоязычное
* сортировка - число
* картинка - изображение
* описание - html

**Издательство.**

Свойства:

* название - обязательное, многоязычное
* активность - логическое
* символьный код - обязательное, многоязычное
* сортировка - число
* картинка - изображение
* описание - html

**Автор.**

Свойства:

* название - ФИО, обязательное, многоязычное
* язык - выбор языка, на котором пишет автор (может быть не выбран, значит любой)

Сайт [http://test.panor.ru](http://test.panor.ru/)

**Шапка.**

Логотип, контактные телефоны, кнопки Войти / Регистрация (для неавторизованных) и Личный кабинет / Выйти (для авторизованных). Заявка на подписку - форма, отправляется на почту (желательно где-то дополнительно сохранять). Меню - двухуровневое, настраиваемое, например, журналы по темам - это категории. Корзина - при наведении показываются товары в корзине (подписки, выпуски, статьи). При клике - переход на страницу "Моя корзина".

**Футер.**

Логотип. Меню Список соц.сетей (настраивается). Адрес, подписка, реклама (настраиваются). Наши партнеры (настраиваются).

**Главная.**

Форма для поиска. Поисковый запрос, где искать. Расширенный поиск - переход на страницу поиска. Журналы по категориям - переход на страницу со списком категорий.

Баннеры - настраиваются. Свойства: картинка, ссылка, название (многоязычное), привязка к языку (множественное), сортировка.

Новые журналы - выпуски, с наибольшей датой выхода.

Новые статьи - статьи, с наибольшей датой выхода. Журналы по темам - список категорий, при клике на категории в блок "Новые статьи" загружаются статьи с выбранной категорией.

**Журналы по алфавиту.**

Форма для поиска, аналогичная форме со странице поиска.

Список журналов. Сортировка (по названию, по дате). Кнопки переключения отображения (список, плитка). На каждом превью журнала есть чекбокс, список выпусков (при нажатии на "Перейти" открывается выбранный выпуск), получить доступ (переход на страницу журнала с открытой вкладкой "Подписка").

Справа блок "Действия с выбранными". Получить доступ - если отмечен один журнал, то переход на вкладку "Подписка". В избранное - добавление в избранное. Рекомендовать - отправка на введенный email ссылок на отмеченные журналы. Цитировать - вывод текста с информацией о выбранных элементах, при нажатии на "Копировать" происходит копирование в буфер обмена. Внизу пагинация. И список новых выпусков (как на главной).

**Страница журнала.**

Форма для поиска по статьям журнала.

**Вкладка "Журнал".**

Хлебные крошки. Выводятся картинка, название, описание, ISSN, входит в перечень BAK.

Справа блок, аналогичен блоку "действия с выбранными", применительно к текущему журналу.

Внизу блок "Новые журналы", как на главной.

**Вкладка "Все номера".**

Хлебные крошки. Выводятся картинка, название.

Выводится список выпусков данного журнала, сгруппированные по годам, отсортированные от новых к старым. У каждого выпуска - чекбокс (для действий с выбранными), номер (ссылка на страницу выпуска), иконка "добавить в избранное/удалить из избранного", иконка "поделиться" (открывается блока для шаринга в соц сетях, можно использовать от яндекса), получить доступ (добавление в корзину, если есть доступ, то переход на страницу выпуска).

Справа блок "действия с выбранными", применяется к отмеченным выпускам.

Внизу блок "Новые журналы", такой же как на главной.

**Вкладка "Свежий номер".**

Хлебные крошки.

Список статей последнего (самого нового) выпуска этого журнала. Переключатели отображения, сортировка, справа блок "действия с выбранными".

Превью статьи. Чекбоск, авторы (ссылка на поиск с фильтром по этому автору), название (ссылка на страницу статьи), журнал (информация о выпуске: название журнала, номер выпуска, год выпуска) - ссылка на выпуск, ниже добавить в избранное/удалить из избранного, поделиться, получить доступ (добавление в корзину).

Внизу блок "Новые журналы", такой же как на главной.

**Вкладка "Все статьи".**

То же что и вкладка "Свежий номер", только статьи всех выпусков журнала.

**Вкладка "Подписка".**

Хлебные крошки, изображение.

Выбор типа подписки, начала, срока, количество экземпляров подписки.

Оформление по абонементу - форма с данными для генерации PDF-файла.

Справа часть блок "действия с выбранными" из вкладки "Журнал", только без кнопки "Получить доступ" (т.к. мы уже находимся на ней).

Внизу блок "Новые журналы", такой же как на главной.

**Вкладка "Прислать статью".**

То же что и на вкладке "Журнал", только вместо информации о журнале - форма для отправки статьи. Данные с формы сохраняются на сервере, и отправляется письмо на почту.

**Вкладка "О журнале".**

Хлебные крошки, справа блок "действия с выбранными", внизу "новые журналы".

Посередине выводится информация:

* блок Контакты (все поля многоязычные): Главный редактор, Телефон, E-mail, Формат, Объем, Периодичность, Сайт, Editor in chief, О редакторе (html), Контакты (html)
* Редсовет - многоязычное, html
* Указатель статей - многоязычное, html
* Рубрики - многоязычное, html
* Порядок рецензирования - многоязычное, html
* Правила предоставления статей - многоязычное, html

Каждый блок в отдельном "аккордеоне".

**Выпуск.**

Страница выпуска.

Вверху хлебные крошки.

Слева название журнала (ссылка на журнал), номер выпуска, год, картинка выпуска.

Справа список статей этого выпуска (превью стандартное для статьи).

Справа блок "действия с выбранными" статьями.

Внизу "новые журналы".

**Статья.**

Выводятся хлебные крошки, название, краткое описание (описание для анонса).

Авторы - ссылки для поиска с фильтром по этому автору.

Если статья доступна текущему пользователю, то выводится детальное описание.

Справа блок "действия".

Если статья недоступна текущему пользователю, то выводится кнопка "получить доступ" - добавление в корзину. Если "ограничение доступа" - за регистрацию, то выводится окно авторизации.

В избранное, рекомендовать, цитировать, поделиться - все стандартное.

Если статья доступна, то выводится "Распечатать" - ссылка на страницу печати статьи. На этой странице статья парсится и формируется разметка для вывода на печать.

**Поиск.**

Поиск по журналам и статьям.

Фильтр для поиска. Тематика (категории), поисковый запрос, где искать, журнал (для статей), автор (первая буква, и строка) (для статей), дата публикации (для статей), УДК (для статей, забыл указать это свойство), тип (искать по журнала, статьям, везде), только доступные для чтения (журналы, на которые куплена подписка, статьи купленные, по подписке, за регистрацию, и бесплатные), только в избранном (те что добавлены пользователем в избранное, только для авторизованных пользователей).

Сбросить поиск, свернуть, развернуть расширенный режим.

Сохранить поиск - доступно при непустом фильтре. Сохраняется с привязкой к текущему пользователю, недоступно для неавторизованных.

Мои поиски - список сохраненных пользователем поисков. Крестик - удаление из сохраненных поисков, самый первый крестик - удалить все сохраненные поиски. Клик на название сохраненного поиска - заполнение фильтра данными из сохраненного поиска.

Результаты поиска. Количество результатов, переключатель отображения, действия с выбранными, внизу пагинация.

Превью элемента зависит от его типа (журнал или статья). Для журнала как на странице "Журналы по алфавиту", для статей как на странице "Статьи".

**Издательства.**

Ссылка <http://test.panor.ru/publishers/>

Хлебные крошки.

Список издательств. Картинка, название, описание. При клике на название или картинку переход на страницу издательства.

**Издательство.**

Список журналов данного издательства. Внешний вид и функционал как на странице "Журналы по алфавиту".

**Категории.**

Список категорий. Картинка, название, описание. При клике на название или картинку переход на страницу поиска с фильтром по выбранной категории и типу "журналы".

**Избранное.**

Список добавленных в избранное элементов (журналы, выпуски, статьи).

Фильтр по типу (журналы (к ним относятся и выпуски) и статьи).

Действия с выбранными, как и везде, только вместо "добавить в избранное" - "удалить" из избранного. Превью зависит от типа (журнал, выпуск, статья).

**О системе.**

Статическая страница, текст можно менять.

**Личный кабинет.**

Стартовая страница личного кабинета. Слева меню (присутствует во всем личном кабинете).

**Статус заказов.**

Выводится список заказов.

Номер заказов, подробнее (ссылка на страницу заказа), список товаров в заказе, дата оформления, статус (ожидает оплаты, оплачен, отменен).

**Заказ.**

Состав заказа (картинка, название, для подписки параметры подписки, для выпуска тип, количество, цена), тип оплаты, статус. Если заказ оплачен и товар типа электронная подписка, то картинка и название - ссылка на "Мои журналы" с открытым текущим журналом.

Если не оплачен и не отменен, то выводится ссылка для оплаты заказа - переход на страницу оплаты, и выводится кнопка отменить, которая переводит заказ в статус "отменен".

**Моя корзина.**

Список товаров в корзине. Картинка (если есть), название, количество, цена.

Внизу "Итого" и кнопка "Оформить заказ".

**Оформление заказа.**

Переключатель "физическое/юридическое лицо", для каждого типа свой список параметров. При нажатии "Оформить заказ" создается заказ (со статусом "ожидает оплаты"), и в него добавляются товары из корзины.

**Мои подписки.**

Список товаров типа "подписка" в заказах пользователя со статусом "оплачен".

Фильтр по типу (все, печатные, электронные).

Картинка журнала, название журнала, тип подписки, первый месяц.

Если текущий месяц позже периода подписки, то выводится "истекла", иначе "активна".

Ссылка "перейти к журналам" - переход на страницу "мои журналы" с открытым журналом.

**Личные данные.**

Страница редактирования данных пользователя.

**Мои журналы.**

Список доступны пользователю выпусков, сгруппированные по журналам. Журналы для ознакомления - все выпуски, которые пользователь получил бесплатно (например, по промокоду). При нажатии на выпуск - переход на страницу читалки этого выпуска. Функционал по читалке описан в сущности "Выпуск".

**API для мобильного приложения.**

Обмен данными между приложением и сервером происходит в формате json.

Данные для авторизации пользователя передаются от приложения серверу в заголовке в ключе Authorization, в котором логин и пароль разделены пробелом, а пароль зашифрован RSA.

Стандартное сообщение об ошибке имеет формат

|  |
| --- |
| {  'message' => 'Текст описания ошибки',  'code' => '404' } |

Фильтр передается в параметре запроса в ключе filter в формате json

Пагинация.

В запросе к серверу 2 может быть параметра: page - номер страницы, count - количество элементов на одной странице.

В ответе от сервера передается в ключе pagination, в формате json, и имеет один обязательный ключ - is\_more (есть ли еще страницы), 2 необязательных: item\_count - общее количество записей (на всех страницах), page\_count - количество страницы.

**Запросы и форматы ответов.**

! означает обязательность

? означает необязательность

первая строка - название метода (используется в коде приложения)

вторая строка - URL запроса

третья строка - описание запроса

далее список входных параметров

потом формат ответа от сервера

GetLatestNumbers

/api/numbers/latest/

список последних номеров <http://joxi.ru/J2bBkXJi4D9KZ2>

без параметров

формат ответа:

[

{

id: string!,

name: string!,

price: double!,

release\_number: int?,

date: date?,

preview\_image: url?,

in\_basket: bool?,

in\_favorite: bool?

},

...

]

GetMainBanner

/api/slider/

данные для баннера на главной <http://joxi.ru/5mdBynGivoGwQA>

без параметров

формат ответа:

[

{

image: url!, ссылка на картинку

url: url? ссылка для перехода по клику

},

...

]

без пагинации

GetLoggedBlockInfo

/api/profile/menu/

данные пользователя в менюшке <http://joxi.ru/BA0G8baIBQELx2>

без параметров.

401 для неавторизованных

формат ответа:

{

name: string!,

email: string?,

image: url?

}

GetFreeNumbers

/api/numbers/free/

данные журналов (выпусков?) в свободном доступе <http://joxi.ru/a2XnLY7Cye6Njr>

без параметров

формат ответа

[

{

id: string!,

name: string!,

release\_number: int?,

date: date?,

preview\_image: url?

},

...

]

без пагинации

GetPopularMagazines

/api/magazines/popular/

данные популярных изданий (журналы) <http://joxi.ru/4AklMEZiMg3yKr>

без параметров

формат ответа:

[

{

id: string!,

name: string!,

price: double!,

preview\_image: url?,

in\_basket: bool?,

in\_favorite: bool?

},

...

]

Login

api/login/

авторизация

входные параметры:

метод: POST или PUT

параметры: login, password (RSA закодированный)

- логин\емейл

- пароль

ответ:

200 если ок и 400 если не ок

при ошибке в теле стандартная ошибка

Register

/api/register/

регистрация

метод: POST или PUT

параметры: email, name, password (RSA закодированный), confirm (RSA закодированный)

входные данные:

- емейл

- имя

- пароль

- подтверждение пароля

ответ:

200 если ок и 400 если не ок

при ошибке в теле стандартная ошибка

GetMagazines

/api/magazines/

список всех журналов

пример: /api/magazines/?filter={"publishers":"781"}&sort={"date\_create":"asc"}&count=5&page=1

входные параметры:

-фильтр:

название - name (подстрока)

направление - categories (строка (ID) или массив строк)

издательство - publishers (строка (ID) или массив строк)

- сортировка. Сортировка забита на клиенте, я буду передавать 2 параметра: код по которому нужна сортировка и направление (asc\desc). Нужны параметры куда их подставлять и их значение. Сейчас предусмотрено: По алфавиту, Дешевле , Дороже , По дате выхода (позже) , По дате выхода (раньше)

аналогично фильтру, GET-параметр sort

По алфавиту - NAME

Дешевле - PROPERTY\_PRICE\_DIGITAL : asc

Дороже - PROPERTY\_PRICE\_DIGITAL : desc

По дате выхода (позже) - DATE\_CREATE : desc

По дате выхода (раньше) - DATE\_CREATE : asc

- пагинационные параметры.

формат ответа:

[

{

id: string!,

name: string!,

publisher: {

id: string!,

name: string!

},

price: double!,

preview\_image: url?,

in\_basket: bool?,

in\_favorite: bool?

},

...

]

GetMagazinesFilterValues

/api/magazines/filter/

данные для фильтрации всех журналов

без входных параметров

формат ответа:

{

directions: [

{

id: string!,

name: string!

},

...

],

publishers: [

{

id: string!,

name: string!

},

...

]

}

GetMagazinesFilterSuggestions

/api/magazines/suggestions/

с фильтрацией: /api/magazines/suggestions/?filter={"name":"<подстрока>"}

список подсказок при вводе имени журнала в фильтре

входные параметры:

- строка

формат ответа - массив строк:

[

string!,

string!,

...

]

GetMagazine

/api/magazines/<ID\_журнала>/

детальная информация о журнале

входные параметры:

- ид журнала

формат ответа:

{

id: string!,

detail\_image: url?,

description: string?, небольшое описание

price: double!,

is\_promo: bool?,

text: string?, большое описание

in\_basket: bool?,

in\_favorite: bool?

}

GetMagazineArchive

/api/numbers/?filter={"magazine":"<ID\_журнала>"}&count=<кол-во\_выпусков>

архив журналов (выпуски журнала) <http://joxi.ru/5mdBynGivoNE4A>

входные параметры

- ид журнала

формат ответа:

[

{

id: string!,

name: string?,

release\_number: int?,

date: date?,

preview\_image: url?

},

...

]

Ответ: объект, в котором есть ключ "items", значением которого является массив требуемых объектов.

GetMagazineSimilarMagazines

/api/magazines/<ID\_журнала>/similar/

список похожих журналов <http://joxi.ru/xAeB7X8iYO0obr>

входные параметры:

- ид журнала

формат ответа:

[

{

id: string!,

name: string!,

price: double!,

preview\_image: url?

},

...

]

GetNumbers

/api/numbers/

список выпусков

входные параметры:

-фильтр:

название - name

ид журнала - magazine

дата начала - date\_from

дата конца - date\_to- сортировка: По алфавиту , Дешевле , Дороже , По дате выхода (позже) , По дате выхода (раньше)

По алфавиту - name asc

Дешевле - property\_price\_digital asc

Дороже - property\_price\_digital desc

По дате выхода (позже) - active\_from desc

По дате выхода (раньше) - active\_from asc

- пагинация

формат ответа:

{

----items: [

--------{

------------id: string!,

------------name: string!,

------------release\_number: int?,

------------date: date?,

------------price: double!,

------------preview\_image: url?,

------------in\_basket: bool?,

------------in\_favorite: bool?

--------},

--------.....

----],

----is\_more: bool?

]

GetNumbersFilterValues

/api/numbers/filter/

данные для фильтра в выпусках

без входных параметров

формат ответа:

{

----magaines: [

--------{

------------id: string!,

------------name: string!

--------},

--------.....

----]!,

----min\_date: date!,

----max\_date: date!

}

GetNumbersFilterSuggestions

/api/numbers/suggestions/?filter={"name":"<подстрока>"}

поиск по подстроке

входные параметры:

- строка

формат ответа:

[

----string!,

----string!,

----string!,

----....

]

GetNumber

/api/numbers/<ID\_выпуска>/

информация об 1 выпуске

входные параметры:

- ид выпуска

формат ответа:

{

----id: string!,

----name: string!,

----detail\_image: url?,

----description: string!? (немного текста),

----text: string? (много текста),

----price: double!,

----in\_basket: bool?,

----in\_favorite: bool?

}

GetNumberOtherMagazineNumber

/api/numbers/<ID\_выпуска>/other\_numbers/?count=<кол-во\_выпусков>

список других выпусков журнала

входные параметры:

- ид выпуска

-? ограничение

формат ответа:

[

----{

--------id: string!,

--------name: string!,

--------release\_number: int?,

--------date: date?,

--------preview\_image: url?

----}

]

GetNumberArticles

/api/numbers/<ID\_выпуска>/?count=<кол-во\_статей>

список статей в выпуске

входные параметры:

- ид выпуска

-? ограничение

формат ответа:

[

----{

--------id: string!,

--------name: string!,

--------authors: [

------------{

----------------id: string!,

----------------name: string!

------------}

--------]?,

--------magazine: {

------------id: string!,

------------name: string!

--------}?,

--------number: {

------------id: string!,

------------name: string!,

--------}?,

--------date: date?,

--------price: double!,

--------in\_basket: bool?,

--------in\_favorite: bool?

----}

]

GetArticles

/api/articles/

Пример: /api/articles/?filter={"number":228299,"date\_from":"2017-06-01"}&sort={"price":"asc","active\_from":"desc"}&count=10&page=1

список статей

входные параметры:

фильтр.

- name - название (подстрока)

- category (id) - тематика (название\ид)

- magazine (id) - журнал (название\ид)

- number (id) - выпуск (название\ид)

- author (id) - автор (название\ид)

- date\_from - дата с

- date\_to - дата по

- keywords - теги (их может быть несколько, строки)

сортировка

- name asc - По алфавиту

- price desc - Дороже

- price asc - Дешевле

- active\_from desc - по дате выхода (позже)

- active\_from asc - по дате выхода (раньше)

- пагинация

формат ответа:

{

items: [

{

id: string!,

name: string!,

number: {

id: string!,

name: string!,

release\_number: int?

}?,

magazine: {

id: string!,

name: string!

}?,

authors: [

{

id: string!,

name: string!

},

.....

]?,

price: double!,

date: date?,

in\_basket: bool?,

in\_favorite: bool?

},

.....

],

info: {

is\_more: bool!

}

}

GetArticlesFilterValues

/api/articles/filter/

информация для фильтра статей

без входных параметров

формат ответа:

{

subjects: [

{

id: string!,

name: string!

},

.....

],

magazines: [

{

id: string!,

name: string!

},

.....

],

authors: [

{

id: string!,

name: string!

},

.....

],

date\_from: date!,

date\_to: date!

}

GetArticlesFilterSuggestions

/api/articles/suggestions/?filter={"name":"<подстрока>"}

поиск по строке в фильтре статей

входные параметры:

- строка

формат ответа:

[

string,

.....

]

GetArticlesFilterTagsSuggestions

/api/articles/suggestions\_tag/?filter={"keywords":"<строка>"}

подсказки при выборе тегов

входные параметры:

- строка

формат ответа:

[

string,

.....

]

GetArticle

/api/articles/<ID\_статьи>/

информация о статье

входные параметры:

- ид

формат отввета:

{

id: string!,

name: string!,

text: string!,

price: double!,

keywords: [

string,

.....

],

in\_basket: bool?,

in\_favorite: bool?

}

GetArticleNumber

/api/articles/<ID\_статьи>/number/

выпуск статьи

входные параметры:

- ид статьи

формат ответа:

{

id: string!,

name: string!,

release\_number: int?,

date: date?,

price: double!,

preview\_image: url?,

in\_basket: bool?,

in\_favorite: bool?

}

GetArticleSimilarArticles

/api/articles/<ID\_статьи>/similar/

список похожих статей статьи

входные параметры:

- ид статьи

формат ответа:

[

{

id: string!,

name: string!,

number: {

id: string!,

name: string!,

release\_number: int?

}?,

magazine: {

id: string!,

name: string!

}?,

authors: [

{

id: string!,

name: string!

},

.....

]?,

price: double!,

date: date?,

in\_basket: bool?,

in\_favorite: bool?

},

.....

]

AddToFavoriteMagazine

/api/favorites/

добавить журнал в избранное

входные параметры:

- ид журнала

метод POST или PUT

тело запроса:

|  |
| --- |
| {  "object": <ID\_журнала>,  "type": "magazine"  } |

формат ответа:

не задан (200 или стандартная ошибка)

RemoveFromFavoriteMagazine

/api/favorites/<ID\_журнала>/

убрать журнал из избранного

метод DELETE

входные параметры:

- ид жунала

формат ответа:

не задан

GetFavoriteMagazines

/api/favorites/?filter={"type":"magazine"}

список избранных журналов

входные параметры:

-фильтр:

- object\_name - название (подстрока) ?

- category - направление (id направления) ?

- publisher - издательство (id издательства) ?

сортировка. Сортировка забита на клиенте, я буду передавать 2 параметра: код по которому нужна сортировка и направление (asc\desc). Нужны параметры куда их подставлять и их значение. Сейчас предусмотрено: По алфавиту, Дешевле , Дороже , По дате выхода (позже) , По дате выхода (раньше)

sort={"property\_object.name":"asc"}

sort={"property\_object.property\_price\_digital":"asc"}

sort={"property\_object.property\_price\_digital":"desc"}

sort={"property\_object.created\_date":"desc"}

sort={"property\_object.created\_date":"asc"}

- пагинационные параметры.

формат ответа:

[

{

id: string!,

name: string!,

publisher: {

id: string!,

name: string!

},

price: double!,

preview\_image: url?

},

...

]

GetFavoriteMagazinesFilterValues

/api/favorites/filter/?filter={"type":"magazine"}

данные для фильтрации избранных журналов

без входных параметров

формат ответа:

{

directions: [

{

id: string!,

name: string!

},

...

],

publishers: [

{

id: string!,

name: string!

},

...

]

}

GetFavoriteMagazinesFilterSuggestions

/api/favorites/suggestions/?filter={"type":"magazine"}

подсказки к названию в фильтре избранных журналов

входные параметры:

- строка

формат ответа - массив строк:

[

string!,

...

]

AddToFavoriteNumber

/api/favorites/

добавить выпуск в избранное

входные параметры:

- ид выпуска

метод POST или PUT

тело запроса

{

"object": <ID\_выпуска>,

"type": "number"

}

формат ответа:

не задан

RemoveFromFavoriteNumber

/api/favorites/<ID\_выпуска>/

убрать выпуск из избранного

метод DELETE

входные параметры:

- ид выпуска

формат ответа:

не задан

GetFavoriteNumbers

/api/favorites/?filter={"type":"number"}

список избранных выпусков

входные параметры:

-фильтр:

- object\_name--— название (подстрока)

- magazine--— ид журнала

- object\_date\_from--— дата начала

- object\_date\_to--— дата конца

сортировка: По алфавиту , Дешевле , Дороже , По дате выхода (позже) , По дате выхода (раньше)

sort={"property\_object.name":"asc"}

sort={"property\_object.property\_price\_digital":"asc"}

sort={"property\_object.property\_price\_digital":"desc"}

sort={"property\_object.active\_from":"desc"}

sort={"property\_object.active\_from":"asc"}

- пагинация

формат ответа:

{

----items: [

--------{

------------id: string!,

------------name: string!,

------------release\_number: int?,

------------date: date?,

------------price: double!,

------------preview\_image: url?,

------------in\_basket: bool?,

------------in\_favorite: bool?

--------},

--------.....

----],

----is\_more: bool?

]

GetFavoriteNumbersFilterValues

/api/favorites/filter/?filter={"type":"number"}

данные для фильтра в избранных выпусках

без входных параметров

формат ответа:

{

----magaines: [

--------{

------------id: string!,

------------name: string!

--------},

--------.....

----]!,

----min\_date: date!,

----max\_date: date!

}

GetFavoriteNumbersFilterSuggestions

/api/favorites/suggestions/?filter={"type":"number","object\_name":"<строка>"}

список подсказок по имени в фильтре избранных выпусок

входные параметры:

- строка

формат ответа:

[

----string!,

----....

]

AddToFavoriteArticle

/api/favorites/

добавить статью в избранное

метод POST или PUT

входные параметры:

- ид статьи

тело запроса:

{

"object": <ID\_статьи>,

"type": "article"

}

формат ответа:

не задан

RemoveFromFavoriteArticle

/api/favorites/<ID\_статьи>/

убрать статью из избранного

метод DELETE

входные параметры:

- ид статьи

формат ответа:

не задан

GetFavoriteArticles

/api/favorites/?filter={"type":"article"}

список избранных статей

входные параметры:

- фильтр.

- object\_name - название (подстрока)

- category - тематика (название\ид)

- magazine - журнал (название\ид)

- number - выпуск (название\ид)

- author - автор (название\ид)

- object\_date\_from - дата с

- object\_date\_to - дата по

- keywords (string or array) - теги (их может быть несколько, строки)

сортировка

sort={"property\_object.name":"asc"} - По алфавиту

sort={"price":"asc"} - Дешевле

sort={"price":"desc"} - Дороже

sort={"property\_object.active\_from":"desc"} - по дате выхода (позже)

sort={"property\_object.active\_from":"asc"} - по дате выхода (раньше)

- пагинация

формат ответа:

{

items: [

{

id: string!,

name: string!,

number: {

id: string!,

name: string!,

release\_number: int?

}?,

magazine: {

id: string!,

name: string!

}?,

authors: [

{

id: string!,

name: string!

},

.....

]?,

price: double!,

date: date?,

in\_basket: bool?,

in\_favorite: bool?

},

.....

],

info: {

is\_more: bool!

}

}

GetFavoriteArticlesFilterValues

/api/favorites/filter/?filter={"type":"article"}

информация для фильтра избранных статей

без входных параметров

формат ответа:

{

subjects: [

{

id: string!,

name: string!

},

.....

],

magazines: [

{

id: string!,

name: string!

},

.....

],

authors: [

{

id: string!,

name: string!

},

.....

],

date\_from: date!,

date\_to: date!

}

GetFavoriteArticlesFilterSuggestions

/api/favorites/suggestions/?filter={"type":"article"}

подсказка по имени в фильтре избранных статей

входные параметры:

- строка

формат ответа:

[

string,

.....

]

GetFavoriteArticlesFilterTagsSuggestions

/api/favorites/suggestions\_tag/?filter={"type":"article","keywords":"<строка>"}

подсказки при выборе тегов в избранных статьях

входные параметры:

- строка

формат ответа:

[

string,

.....

]

BasketAddNumbe

/api/numbers/<ID\_выпуска>/r

добавить выпуск в корзину

метод POST или PUT

входные параметры:

- ид выпуска

- type - тип (электронный, печатный)

- quantity - количество

ответ:

ид товара в корзине

BasketAddArticle

/api/articles/<ID\_статьи>/

добавить статью в корзину

метод POST или PUT

входные параметры:

- ид статьи

ответ:

ид товара в корзине

BasketDeleteItem

/api/basket/<basket\_id>/

удалить товар из корзины

метод DELETE

входные параметры:

- ид товара в корзине

формат ответа:

не задан

BasketDeleteAll

/api/basket/

очистить корзину

метод DELETE

без входных параметров

формат ответа не задан

BasketGetItems

/api/basket/

список товаров в корзине

без входных параметров

формат ответа:

{

magazines: [

{

basket\_id: string!,

id: string!,

name: string!,

preview\_image: url?,

subscribe\_info: {

price: double!,

end: date!,

left: time!

}!

},

.....

],

numbers: [

{

basket\_id: string!,

id: string!,

name: string!,

preview\_image: url?,

release\_number: int?,

date: date?,

price: double?

},

.....

],

articles: [

{

basket\_id: string!,

id: string!,

name: string!,

number: {

id: string!,

name: string!,

release\_number: int?

}?,

magazine: {

id: string!,

name: string!

}?,

authors: [

{

id: string!,

name: string!

},

.....

]?,

date: date?,

price: double!

}

]

}

GetMyLibraryMagazines

/api/magazines/?filter={"my":true}

список журналов в моей библиотеке

фильтрация, сортировка и все остальное работает как в GetMagazines

просто дополнительно применяется фильтр по доступным пользователю журналам

входные параметры:

-фильтр:

- название (подстрока) ?

- направление (id направления) ?

- издательство (id издательства) ?

- сортировка. Сортировка забита на клиенте, я буду передавать 2 параметра: код по которому нужна сортировка и направление (asc\desc). Нужны параметры куда их подставлять и их значение. Сейчас предусмотрено: По алфавиту, Дешевле , Дороже , По дате выхода (позже) , По дате выхода (раньше)

- пагинационные параметры.

формат ответа:

[

{

id: string!,

name: string!,

publisher: {

id: string!,

name: string!

},

price: double!,

preview\_image: url?

},

...

]

GetMyLibraryMagazinesFilterValues

/api/magazines/filter/?filter={"my":true}

данные для фильтрации журналов в моей библиотеке

без входных параметров

формат ответа:

{

directions: [

{

id: string!,

name: string!

},

...

],

publishers: [

{

id: string!,

name: string!

},

...

]

}

GetMyLibraryMagazinesFilterSuggestions

/api/magazines/suggestions/?filter={"my":true,"name":"<строка>"}

подсказки к названию в фильтре журналов в моей библиотеке

входные параметры:

- строка

формат ответа - массив строк:

[

string!,

...

]

GetMyLibraryNumbers

/api/numbers/?filter={"my":true}

список выпусков в моей библиотеке

то же что и GetNumbers, только с фильтром filter={"my":true}

входные параметры:

-фильтр:

--— название (подстрока)

--— ид журнала

--— дата начала

--— дата конца

- сортировка: По алфавиту , Дешевле , Дороже , По дате выхода (позже) , По дате выхода (раньше)

- пагинация

формат ответа:

{

----items: [

--------{

------------id: string!,

------------name: string!,

------------release\_number: int?,

------------date: date?,

------------price: double!,

------------preview\_image: url?,

------------in\_basket: bool?,

------------in\_favorite: bool?

--------},

--------.....

----],

----is\_more: bool?

]

GetMyLibraryNumbersFilterValues

/api/numbers/filter/?filter={"my":true}

данные для фильтра в выпусках в моей библиотеке

без входных параметров

формат ответа:

{

----magaines: [

--------{

------------id: string!,

------------name: string!

--------},

--------.....

----]!,

----min\_date: date!,

----max\_date: date!

}

GetMyLibraryNumbersFilterSuggestions

/api/numbers/suggestions/?filter={"my":true,"name":"<строка>"}

список подсказок по имени в фильтре выпусков в моей библиотеке

входные параметры:

- строка

формат ответа:

[

----string!,

----....

]

GetMyLibraryArticles

/api/articles/?filter={"my":true}

список статей в моей библиотеке

параметры как у GetArticles

входные параметры:

- фильтр.

- название (подстрока)

- тематика (название\ид)

- журнал (название\ид)

- выпуск (название\ид)

- автор (название\ид)

- дата с

- дата по

- теги (их может быть несколько, строки)

- сортировка

- По алфавиту

- Дешевле

- Дороже

- по дате выхода (позже)

- по дате выхода (раньше)

- пагинация

формат ответа:

{

items: [

{

id: string!,

name: string!,

number: {

id: string!,

name: string!,

release\_number: int?

}?,

magazine: {

id: string!,

name: string!

}?,

authors: [

{

id: string!,

name: string!

},

.....

]?,

price: double!,

date: date?,

in\_basket: bool?,

in\_favorite: bool?

},

.....

],

info: {

is\_more: bool!

}

}

GetMyLibrarbyArticlesFilterValues

/api/articles/filter/?filter={"my":true}

информация для фильтра статей в моей библиотеке

без входных параметров

формат ответа:

{

subjects: [

{

id: string!,

name: string!

},

.....

],

magazines: [

{

id: string!,

name: string!

},

.....

],

authors: [

{

id: string!,

name: string!

},

.....

],

date\_from: date!,

date\_to: date!

}

GetMyLibraryArticlesFilterSuggetions

/api/articles/suggestions/?filter={"my":true,"name":"<строка>"}

подсказка по имени в фильтре статей в моей библиотеке

входные параметры:

- строка

формат ответа:

[

string,

.....

]

GetMyLibraryArticlesFilterTagsSuggetions

/api/articles/?filter={"my":true,"keywords":"<строка>"}

подсказки при выборе тегов в статьях в моей библиотеке

входные параметры:

- строка

формат ответа:

[

string,

.....

]

GetActiveSubscriptions

/api/subscriptions/?filter={"active":true}

список активных подписок

входные параметры:

- фильтр (не предусмотрено)

- сортировка (не предусмотрено)

- пагинация

формат ответа:

{

items: [

{

id: string!,

name: string!,

type: string!,

image: url?,

date\_end: date!,

month\_total: int!,

month\_left: int!

},

.....

],

pagination: {

}

}

GetExpiredSubscriptions

/api/subscriptions/?filter={"active":false}

список закончившихся подписок

входные параметры:

- фильтр (не предусмотрено)

- сортировка (не предусмотрено)

- пагинация

формат ответа:

{

items: [

{

id: string!,

name: string!,

type: string!,

image: url?

},

.....

],

pagination: {

}

}

GetSubscribers

/api/subscribers/

список подписчиков

входные параметры:

- фильтр:

- name - имя

- email - email

- сортировка

- name - по имени

- email - по емейлу

- пагинация

формат ответа:

{

items: [

{

id: string!,

name: string!,

email: string!,

image: url?

},

.....

],

pagination: {

}

}

DeleteSubscriber

/api/subscribers/<ID\_подписчика>/

удалить подписчика

метод DELETE

входные параметры:

- ид

формат ответа не задан

GetSubscribersFilterNameSuggestions

/api/subscribers/suggestions\_name/?filter={"name":"<строка>"}

подсказка к фильтру подписчиков по имени

входные параметры:

- строка

формат ответа:

[

string!,

.....

]

GetSubscribersFilterEmailSuggestions

/api/subscribers/suggestions\_email/?filter={"email":"<строка>"}

подсказка к фильтру подписчиков по email

входные параметры:

- строка

формат ответа:

[

string!,

.....

]

GetSubscriber

/api/subscribers/<ID\_подписчика>/

получить подписчика

входные параметры:

- ид

формат ответа:

{

id: string!,

name: string!,

email: string!,

phone: string!

image: url?

}

GetSubscriberSubscriptions

/api/subscribers/<ID\_подписчика>/products/

список подписок подписчика

входные параметры:

- ид подписчика

формат ответа:

[

{

id: string!,

name: string!,

type: string!,

image: url?

}

]

GetAviliableSubscriptions

/api/subscribers/<ID\_подписчика>/products/add/

получить список доступных подписок, чтобы добавить подписчику

входные параметры:

- ид подписчика

формат ответа:

[

{

id: string!,

name: string!

},

......

]

AddSubscriberSubscription

/api/subscribers/<ID\_подписчика>/

добавить подписчику подписку

метод POST или PUT

входные параметры:

- ид подписчика

в теле product = <ID\_подписки(товара)> - ид подписки

формат ответа не установлен

DeleteSubscriberSubscription

/api/subscribers/<ID\_подписчика>/

удалить у подписчика подписку

метод DELETE

входные параметры:

- ид подписчика

- в теле product = <ID\_подписки(товара)> - ид подписки

формат ответа не установлен

GetPurchaseNumberMagazineList

/api/magazines/subscribe/

список возможных журналов для покупки

без входных параметров

формат ответа:

[

{

id: string!,

name: string!

},

.....

]

GetPurchaseNumberNumberList

/api/numbers/buy/?filter={"magazine":<ID\_журнала>}

список выпусков для покупки

входные параметры:

- ид журнала

формат ответа:

[

{

id: string!,

name: string!,

prices: {

print: double?,

digital: double?

}

},

.....

]

GetPurchaseNumberNumberListWithMagazine

/api/numbers/buy/magazine/?filter={"magazine":<ID\_журнала>}

список выпусков для покупки с предустановленным журналом

входные параметры:

- ид журнала

формат ответа:

{

magazine: {

id: string!,

name: string!

},

numbers: [

{

id: string!,

name: string!,

prices: {

print: double?,

digital: double?

}

},

.....

]

}

GetPurchaseNumberNumberData

/api/numbers/<ID\_выпуска>/buy/

покупка выпуска с предустановленным выпуском

входные параметры:

- ид выпуска

формат ответа:

{

magazine: {

id: string!,

name: string!

},

number: {

id: string!,

name: string!,

prices: {

print: double?,

digital: double?

}

}

}

PurchaseNumber

/api/numbers/<ID\_выпуска>/

совершить покупку выпуска

метод POST или PUT

входные параметры:

- ид выпуска

- type - тип (print, digital)

- quantity - количество

формат ответа не задан

GetPurchaseMagazineDirections

/api/magazines/subscribe/categories/

список направлений для подписки

без входных параметров

формат ответа:

[

{

id: string!,

name: string!

},

.....

]

GetPurchaseMagazineDirectionMagazines

/api/magazines/subscribe/?filter={"categories":<ID\_направления>}

список журналов для покупки по направлению

входные параметры:

- ид направления

формат ответа:

[

{

id: string!,

name: string!

},

.....

]

GetPurchaseMagazineMagazineInfo

/api/magazines/<ID\_журнала>/subscribe/

информация для подписки на журнал

входные параметры:

- ид журнала

формат ответа:

{

types: [

{

id: string!,

name: string!,

durations: [

id: string!,

name: string!,

start\_dates: [

id: string!,

name: string!,

price: double!

]

]

}

]

}

GetPurchaseMagazineMagazinesByDirection

/api/magazines/subscribe/category/?filter={"categories":<ID\_направления>}

список журналов для покупки с предустановленным направлением

входные параметры:

- ид направления

формат ответа:

{

direction: {

id: string!,

name: string!

},

magazines: [

{

id: string!,

name: string!

},

.....

]

}

GetPurchasemagazineMagazineInfoWithDirectionAndMagazine

/api/magazines/<ID\_журнала>/subscribe/magazine/

информация для подписки на журнал c заданным журналом

входные параметры:

- ид журнала

формат ответа:

{

direction: {

id: string!,

name: string!

},

magazine: {

id: string!,

name: string!

},

types: [

{

id: string!,

name: string!,

durations: [

id: string!,

name: string!,

start\_dates: [

id: string!,

name: string!,

price: double!

]

]

}

]

}

AddSubscriber

метод POST или PUT

добавить подписчика

входные параметры:

- email

формат ответа не задан

GetOrderInfo

/api/profile/order/

сохраненная информация о пользователе при оформлении заказа

без входных параметров

формат ответа:

{

individual: {

lst\_name: string?,

first\_name: string?,

second\_name: string?,

phone: string?,

email: string?,

address: string?

},

entity: {

company: {

name: string?,

address: string?,

inn: string?,

kpp: string?

},

responsible: {

last\_name: string?,

first\_name: string?,

second\_name: string?

},

contacts: {

phone: string?,

email: string?,

delivery\_address: string?

}

}

}

GetOrders

/api/orders/

список заказов

входные параметры:

- пагинация

формат ответа:

{

orders: [

id: string!,

name: string!,

type: "prepaiment|processing|complete"!,

items: {

magazines:[

{

id: string!,

name: string!,

preview\_image: url?

},

.....

],

numbers: [

{

id: string!,

name: string!,

preview\_image: url?,

release\_number: int?,

date: date?

},

.....

],

articles: [

{

id: string!,

name: string!,

number: {

id: string!,

name: string!,

release\_number: int?

}?,

magazine: {

id: string!,

name: string!

}?,

authors: [

{

id: string!,

name: string!

},

.....

]?,

date: date?,

}

]

}

],

pagination: {}

}

GetProfile

/api/profile/

данные личного кабинета

без входных параметров

формат ответа:

{

first\_name: string?,

last\_name: string?,

email: string?,

phone: string?,

image: url?

}

SaveProfile

/api/profile/

сохранить данные в личном кабинете.

метод POST

входные параметры:

-name - имя

-last\_name - фамилия

-phone - телефон

-email - емейл

-photo - фото (base64)

формат ответа:

{

first\_name: string?,

last\_name: string?,

email: string?,

phone: string?,

image: url?

}

PurchaseMagazine

/api/magazines/<ID\_журнала>/subscribe/

добавление в корзину подписки на журнал

входные параметры:

- ид журнала

- type - ид типа

- start\_date - ид даты начала

- length - ид длительности

- quantity - количество

формат ответа не задан

getMagazinesFilter1

метод по начальным данным по фильтрации. Возвращает пустой объект. На будущее.

getMagazineSuggestions

на входе тип поля для подсказки

на выходе массив select-оподобных объектов (id, name)

Для выпусков и статей по аналогии:

- фильтр для избранных и моей библиотеки в общем запросе

- get\*\*\*Filter оставить получение только тех данных, которые не будут использоваться через suggestion систему (если не ошибаюсь, то это только даты начала\конца)

- get\*\*\*Suggestions по аналогии

ReaderNumberGetInfo

GET /api/number/{id}/reader/info/

информация для читалки выпуска

вход:

- ид выпуска

выход:

- url для чтения (абсолютный)

- нужно ли обновление онлайна

- ид выпуска

- название выпуска

- в избранном ли выпуск

- ссылка для шаринга

{

url: url!,

is\_need\_online\_update: bool?,

id: string!,

name: string!,

in\_favorite: bool?,

link: url?

}

ReaderNumberUpdateOnline

POST /api/number/{id}/reader/online/

запрос на поддержание чтения

вход:

- ид выпуска

выход: —

ReaderNumberAddBookmark

POST /api/number/{id}/reader/bookmarks/

добавление закладки

вход:

- ид выпуска

- название закладки

- якорь

{

name: string!,

anchor: string!

}

выход:

- ид закладки

- название закладки

- якорь

{

id: string!,

name: string!,

anchor: string!

}

ReaderNumberDeleteBookmark

DELETE /api/number/{id}/reader/bookmarks/{bookmarkId}/

удаление закладки

вход:

- ид выпуска

- ид закладки

выход: —

по аналогии для статей:

ReaderArticleGetInfo

ReaderArticleUpdateOnline

ReaderArticleAddBookmark

ReaderArticleDeleteBookmark

для читалки

при require файла читалки нужно передать данные о содержании, закладках и самом контенте:

- $GLOBALS['HTML\_CONTENT']. Подготовленный html контент (дата-атрибуты, абсолютные ссылки на картинки...)

- $GLOBALS['CONTENTS']. Информация о содержании. Массив со структурой:

array(

array(

'name' => 'название элемента',

'anchor' => 'привязка к дата-атрибуту'

),

...

)

$GLOBALS['BOOKMARKS']. Информация о добавленных закладках. Массив со структурой:

array(

array(

'id' => 'id закладки',

'name' => 'название закладки',

'anchor' => 'привязка к дата-атрибуту'

),

...

)

ReaderNumberGetContentInfo

ReaderArticleGetContentInfo

GET /api/numbers/{id}/reader/content/

GET /api/articles/{id}/reader/content/

вход:

- ид выпуска\статьи

выход:

{

content: string!, // html-контент

contents: [

{

name: string!,

anchor: string!

}

...

]

}

ReaderNumberGetBookamarks

ReaderArticleGetBookamarks

получить список закладок

GET /api/bookmarks/?filter

вход:

- ид выпуска\статьи

filter:

item\_type - тип элемента (number, article)

item\_id - ID элемента (выпуска или статьи)

выход: массив объектов-закладок

[

{

'id': string,

'item\_id': string,

'item\_type': string,

'name': string,

'anchor': string,

},

{

...

},

...

]

GET /api/bookmarks/<ID>/ - получить информацию о закладке

возвращает объект в формате

{

'id': string,

'item\_id': string,

'item\_type': string,

'name': string,

'anchor': string,

}

POST /api/bookmarks/ - добавить закладку

в теле

item\_id - ID элемента

name - название закладки

anchor - якорь (data-id html элемента)

DELETE /api/bookmarks/<ID>/ - удалить закладку

**Доступ для партнеров.**

Описание задачи:

Необходимо разработать универсальную API для доступа к журналам через наших партнеров. Т.е. API, благодаря которому они смогут демонстрировать наши листалки через свои сайты. Каждый партнер будет иметь свой ключ для подключения к сайту, нужно продумать простой способ инеграции, чтобы для них это было не слишком затратно. Также нужна какая-то система учета по журналам и количеству доступов. Партнеры врятли захотят нам передавать контактные данные подписчиков (чтобы мы их не переманили в редакционные), но могут давать уникальный идентификационный ключ на каждого подписчика, а нам надо будет отслеживать к-во подключенных пользователей.

Блоксхема условия предоставления журнала: <https://yadi.sk/i/mXOhYKd83KBjQu>

Сущности:

**Партнер.**

Партнер, пользователям которого предоставляется доступ к выпускам.

Свойства:

* ключ - уникальный, обязательный
* активность - логическое

**Квота.**

Настройка доступа к чтению выпусков для пользователей партнеров, здесь указывается к чему есть доступ у пользователей.

При заданном журнале также дополнительно могут быть ограничения по дате выхода выпусков этого журнала. Если заданы и журнал, и выпуск, то доступ предоставляется и к указанному выпуску, и к выпускам указанного журнала.

При использовании квоты ей увеличивается на 1 свойство "использовано". Если свойство "использовано" больше или равно свойству "размер квоты", то новые пользователи не смогут ее использовать, а кто её уже использовал могут продолжать ей пользоваться.

Свойства:

* ID - уникальный, идентификатор квоты.
* партнер - обязательный, привязка к Партнер, для какого партнера предоставляется квота
* активность - логическое
* журнал - привязка к журналу, на какой журнал предоставляется квота
* начало интервала - дата, доступны выпуски журнала, вышедшие после этой даты, если не задана, то нет ограничения
* конец интервала - дата, доступны выпуски журнала, вышедшие до этой даты, если не задана, то нет ограничения
* выпуск - привязка к Выпуск, к какому выпуску предоставляется доступ
* размер квоты - сколько пользователей могут использовать эту квоту
* использовано - сколько раз эту квоту уже использовали

**Пользователь.**

Пользователь партнера, которому предоставляется доступ к чтению выпусков.

Пользователи партнера не являются пользователями Панорамы, но могут быть авторизованы на ней.

Свойства:

* идентификатор - уникальный (в рамках партнером, составной ключ), обязательный, используется для идентификации пользователя
* партнер - обязательный, привязка к Партнер, к какому партнеру относится этот пользователь
* активность - логическое
* использованные квоты - множественное, привязка к Квота, какие квоты пользователь использовал. При повторном обращении к уже использованной квоте, ее свойство "использовано" не увеличивается, т.е. использование (списание) происходит при первом обращении пользователя к этой квоте.
* доступные выпуски - множественное, привязка к Выпуск, список выпусков, к которым пользователь успешно получил доступ.
* email - поста пользователя, используется для отправки кода подтверждения. Если при создании пользователя идентификатором является email-валидная строка, то в это свойство автоматически записывается идентификатор.

Общий алгоритм работы:

По ссылке (пример <http://panor.ru/personal/reader/api/?partnerkey=foo&userkey=bar&journal=123>) открывается специальная страница, на которой проверяется доступ к читалке. Передаются ключ партнера (partnerkey), идентификатор читателя (userkey), ID квоты (journal). Проверяется ключ партнера (существует и активен), идентификатор пользователя (совпадает ли партнер. Если нет пользователя такого, то создается).

Проверяется квота: если пользователь ее уже использовал, или еще не исчерпан размер квоты, то ее можно использовать.

По ID квоты определяется к чему есть доступ. Если указан только выпуск, то он открывается. Если журнал, то выводится список доступных по квоте выпусков (выпуски журнала, возможно с ограничением по дате выхода), пользователь выбирает конкретный выпуск из предложенных, и он открывается.

При открытии выпуска для чтения, работает такой же алгоритм по проверке подтвержденного устройства, как и в читалке для обычных пользователей (описан в сущности Выпуск). Но сбрасывать подтвержденные устройства при исчерпании лимита активированных устройств нельзя. Если устройство не подтверждено, то на email этого пользователя отправляется код подтверждения, и открывается форма для ввода этого кода. Если email пользователя не задан, то выводится форма для ввода email, который запоминается в свойстве "email" этого пользователя. Также проверяется нет ли других устройств в сети.

На странице пользователя партнера в админке должна выводиться информация об устройствах этого пользователя, пример <http://prntscr.com/jroiog>

* "в сети" означает что на этом устройстве сейчас открыт журнал, "обновлен" - время когда был обновлен статус в сети
* "был в сети" - время, когда в последний раз был открыт журнал
* "устарело" означает что "время жизни" устройства истекло (сейчас стоит 1 неделя), как и в читалке, требуется повторное подтверждение устройства
* "код ... не найден" - ошибка в связях базы данных, скорее всего возникать не будет, вывел на всякий случай

**Промокоды:**

Сущности:

**Промокод.**

Свойства:

* код - уникальный, сам промокод
* активность - логическое
* вид - обязательный, список (о вариантах будет написано ниже)
* издательство - множественное, привязка к Издательство
* журнал - привязка к Журнал
* выпуски - привязка к Выпуск, множественное
* лимит использований - максимальное количеств использований промокода. Если пустое или 0, то неограниченное число использований
* использован - сколько раз использовали этот промокод
* журнал для выпусков - привязка к Журнал
* дата начала выпусков - дата
* дата окончания выпусков - дата
* журналы для выбора - привязка к Журнал, множественное, с названием группы (строка).
* сколько можно выбрать - число

**Промо-участник.**

Свойства:

* имя (фио) - обязательное
* пользователь - обязательное, уникальное, привязка к пользователю (стандартная авторизация на сайте)
* телефон
* активированные промокоды - привязка к Промокод, множественное, список активированных пользователем промокодов.
* издательство - привязка к Издательство, множественное, выбранные пользователем издательства
* открытые выпуски - привязка к Выпуск, множественное, список открытых пользователем выпусков

**Выбранные журналы по промокоду.**

Свойства:

* пользователь - обязательное, привязка к Промо-участник
* промокод - обязательное, привязка к Промокод
* журнал - привязка к Журнал, множественное

Общий алгоритм работы:

Страница активации промокода - [http://promo.panor.ru](http://promo.panor.ru/)

На ней пользователь вводит данные: ФИО, email, телефон, промокод. Все поля обязательные. При нажатии на "получить доступ" проверяется доступность промокода: существует, активен, не исчерпан лимит использований, не активировал ли уже этот пользователь этот промокод. Также выводится выбор издательства/журнала/выпуска, если того требует вид промокода. Если код доступен, то на указанный номер телефона отправляется код подтверждения, и открывается окно для ввода этого кода. Если код подтверждения введен верно, то происходит создание промо-участника и использования кода.

Если пользователь не авторизован, то ищется пользователь с таким же email. Если найден, то открывается окно для ввода пароля. Если не найден, то пользователь создается, используются данные из формы. Пользователь автоматически авторизуется. Если пользователь уже авторизован, то данные из формы записываются в его профиль.

Затем проверяется, является ли текущий пользователь промо-участником (есть запись "Промо-участник" со значением свойства "Пользователь" равным этому пользователю). Если не является, то создается запись "промо-участник" для этого пользователя.

Далее происходит активация промокода. У промокода увеличивается на 1 свойство "использован", промо-участнику в "активированные промокоды" добавляется введенный промокод и при необходимости добавляется выбранное издательство.

В зависимости от вида промокода пользователю предоставляется доступ к выпускам:

* Общий - промо-выпуски (отмеченные как промо) журналов тех издательств, которые выбрал промо-участник.
* На журнал - промо-выпуски журнала, указанного в промокоде (свойство "Журнал")
* На издательство - как и общий, но если заданы "дата начала выпусков" и "дата окончания выпусков", то они используются как ограничение по дате выхода выпусков.
* На выпуск - выпуски из свойства "Выпуски" + если заданы свойства "дата начала выпусков", "дата окончания выпусков" и "журнал для выпусков", то все выпуски этого журнала, вышедшие в указанный интервал
* На издательство + на выпуски - объединение выпусков вида "На издательство" и "На выпуск"
* Выборочный - промо-выпуски журналов из записи в "Выбранные журналы по промокоду" с привязкой к этому промокоду и этому пользователю

После активации происходит переход на страницу "Мои журналы", а для промокода вида "Выборочный" переход на страницу выбора журналов.

На странице выбора журналов (ссылка <http://panor.ru/promo/>) происходит поиск всех активированных пользователем промокодов вида "Выборочный", для которых не выбраны журналы (нет записи в "Выбранные журналы по промокоду" или для такой записи пустое свойство "Журнал"). Если таких кодов больше одного то выводится список этих промокодов. Если промокод один, или пользователь выбрал промокод из списка, то выводится список журналов для выбора из свойства "журналы для выбора", сгруппированные по названию группы, и если нет записи в "Выбранные журналы по промокоду", то она создается (такая запись с пустым свойством "Журнал" будет означать, что пользователь открывал страницу выбора журналов по этому промокоду, но не выбрал журналы). Если задано свойство "сколько можно выбрать", то пользователь может выбрать журналов не больше чем это количество. После выбора журналов и нажатия "получить доступ" выбранные журналы записываются в соответствующую запись "Выбранные журналы по промокоду" в свойство "Журнал".

При открытии выпуска промо-участником ему в свойство "открытые выпуски" дописывается этот выпуск.

На странице промо-участника в админке рядом с каждым значением свойства "Активированные промокоды" выводятся доступные по коду выпуски, и для каждого выпуска выводится "стадия доступа" промо-участника к выпуску: открывал ли пользователь этот выпуск (он есть в списке значений свойства "Открытые выпуски")

**Новости.**

Свойства:

* название - обязательное
* активность - логическое
* дата - дата публикации новости
* символьный код - уникальный, используется для ЧПУ ссылки
* издательство - множественное, привязка к Издательство, к каким издательствам относится новость
* описание - html, текст новости
* картинка - изображение
* анонс - html, краткое описание новости
* картинка для анонса - изображение, если не задано, то может быть сгенерировано из "картинка"

На станице со списком новостей (ссылка <http://panor.ru/news/>) выводятся новости, отсортированные по дате от новых к старым, с пагинацией. Для каждой новости выводится ее название (ссылка на страницу новости), картинка для анонса, анонс, издательство, дата публикации, ссылка на детальную страницу новости. Слева выводится список издательств (что-то вроде фильтра по издательству), при нажатии на издательство выводятся новости этого издательства (у которых в значениях свойства "Издательство" есть это издательство).

На детальной странице новости выводятся ее название, дата публикации, картинка, описание, внизу издательство и кнопки для шаринга в соц. сетях. Если у новости задано издательство, то внизу выводятся последние новости этого издательства. Слева выводятся последние новости, и форма для подписки на рассылку.

**Подписчики.**

Этот функционал доступен только пользователям в группе "Мастер-аккаунт".

Страница на сайте - <http://panor.ru/personal/subscribers/>

Подписчики - это пользователи, с которыми мастер-аккаунт делится покупками. Т.е. те товары (подписка, выпуск или статья), которыми мастер-аккаунт делится с подписчиками доступны им также, как будто бы они сами купили этот товар. При этом мастер-аккаунт может поделиться одним товаром только один раз.

На сайте список подписчиков текущего пользователя, и добавленные им товары.

При нажатии на "добавить подписчика" открывается форма добавления подписчика. Обязательные поля: email и издание (товар). В списке товаров выводятся все товары из оплаченных заказов пользователя (мастер-аккаунта), кроме тех, которыми мастер-аккаунт уже поделился. После отправки формы (нажатия кнопки "добавить") происходит поиск пользователя с указанным email. Если он не найден, то пользователь создается на основании данных из формы (email, фамилия, имя). Затем этому пользователю добавляется указанный товар (проверяется, что он уже не был добавлен ранее другому пользователю/подписчику).

В мобильном приложении можно добавлять и удалять подписчиков и товары им, при этом действуют те же ограничения (поделится товаром можно только один раз).