

Базовый модуль для клиентской работы с историей запросов

Базовый модуль для клиентской работы с историей запросов предоставляет реализацию `RequestHistoryListener` для обработки и отправки данных истории запросов.

Базовая настройка

Для подключения данного сервиса необходимо:

1) Подключить данный модуль (`jprime-requesthistory-client-starter`) 2) Подключить модуль, реализующий отправку событий (`RequestHistoryEventPublisher`), например: `jprime-requesthistory-client` 3) В своём `@RestController` подключить сервис `RequestHistoryListener`:

```
@RestController
public class RestController {
    /**
     * Работа с отправкой Истории запросов
     */
    private RequestHistoryListener historyListener;

    @Autowired
    private void setHistoryListener(RequestHistoryListener historyListener) {
        this.historyListener = historyListener;
    }
}
```

4) Сформировать объект `RequestHistoryData` или воспользоваться готовым шаблоном и отправить данные:

```
historyListener
    .sendFind(                // Отправка запроса по идентификатору
        RequestHistory.CLASS_CODE, // Код класса, который мы хотим сохранить
        guid,                  // Идентификатор записи
        authInfo,              // Данные авторизации
        swe                     // Данные запроса
    );
```

Метод `send` работает в асинхронном режиме

Итоговый вид контроллера может выглядеть так:

```

@RestController
@RequestMapping("requesthistory/v1")
public class RestRequestHistoryController {
    /**
     * Обработчик JWT
     */
    private JWTService jwtService;
    /**
     * Работа с отправкой Истории запросов
     */
    private RequestHistoryListener historyListener;

    @Autowired
    private void setHistoryListener(RequestHistoryListener historyListener) {
        this.historyListener = historyListener;
    }

    @Autowired
    private void setJwtService(JWTService jwtService) {
        this.jwtService = jwtService;
    }

    @ResponseBody
    @GetMapping(value =("/{guid}", produces = MediaType.APPLICATION_JSON_VALUE)
    @PreAuthorize("hasAuthority(T(mp.jprime.requesthistory.security.Role).REQUEST_HISTORY_ADMIN)")
    @ResponseStatus(HttpStatus.OK)
    public Mono<JsonRequestHistory> getById(ServerWebExchange swe, @PathVariable String guid) {
        AuthInfo authInfo = jwtService.getAuthInfo(swe);
        historyListener
            .sendFind(
                // Отправка запроса по идентификатору
                RequestHistory.CLASS_CODE, // Код класса, который мы хотим сохранить
                guid,                      // Идентификатор записи
                authInfo,                  // Данные авторизации
                swe                        // Данные запроса
            );

        // ...
    }
}

```

Поддерживаемые типы запросов

На данный момент существуют несколько типов запросов:

Код	Описание	Обязательный параметр
FIND	Поиск по ключу	objectId

SEARCH

Код

Описание

Обязательный
requestBody
параметр

Ограничения

Для того чтобы данные корректно отправились необходимо, чтобы:

1. Были корректны данные аутентификации (если включена проверка);
 2. Была подключена реализация `RequestHistoryEventPublisher` ;
 3. Отсутствовал или был выключен `header X-JPrime-WebQuery-History-Disabled` . *
- Если вы не хотите отправлять запрос в историю запросов, вы можете включить этот `header` , указав у него флаг `true`