

## src

### src package

#### Subpackages

#### src.config package

#### Submodules

#### src.config.app module

```
class src.config.app.ConfigApp
```

Базовые классы: `object`

```
BASE_URL= 'https://portal.elpts.ru'
```

```
HMAC_TIMEOUT= 1
```

```
CAPTCHA_SOURCE= 'elpts-portal-web'
```

```
CAPTCHA_TIMEOUT= 30
```

#### src.config.settings module

#### Module contents

#### src.fastapi package

#### Submodules

#### src.fastapi.main module

#### src.fastapi.router module

#### src.fastapi.schemas module

#### src.fastapi.server module

## Module contents

### src.interfaces package

#### Submodules

#### src.interfaces.abstract\_captcha\_service module

```
class src.interfaces.abstract_captcha_service.AbstractCaptchaService
```

Базовые классы: `ABC`

```
abstract  
async          post_captcha(image, timeout=0)
```

Тип результата:

`dict` | `None`

```
abstract  
async          result_report(task_id, correct)
```

Тип результата:

`bool`

#### src.interfaces.abstract\_page\_parser module

#### src.interfaces.abstract\_proxy module

#### src.interfaces.abstract\_session module

## Module contents

### src.logger package

#### Submodules

#### src.logger.context\_logger module

#### src.logger.logger\_adapter module

```
class src.logger.logger_adapter.LoggingSearchKeyAdapter(logger, wrapper='|')
```

Базовые классы: `LoggerAdapter`

```
process(msg, kwargs)
```

Process the logging message and keyword arguments passed in to a logging call to insert contextual information. You can either manipulate the message itself, the keyword args or both. Return the message and kwargs modified (or not) to suit your needs.

Normally, you'll only need to override this one method in a `LoggerAdapter` subclass for your specific needs.

```
static get_context_message()
```

Возвращает контекст из текущего запроса

Тип результата:

```
str
```

## Module contents

### `src.logic` package

#### Subpackages

#### `src.logic.adapters` package

#### Submodules

#### `src.logic.adapters.response` module

#### Module contents

#### `src.logic.captcha` package

#### Submodules

#### `src.logic.captcha.captcha` module

#### `src.logic.captcha.captcha_service` module

#### Module contents

**src.logic.elpts package**

## **Submodules**

**src.logic.elpts.base\_elpts module**

**src.logic.elpts.elpts module**

**src.logic.elpts.exception\_handler module**

**src.logic.elpts.exceptions module**

**src.logic.elpts.search\_manager module**

## **Module contents**

**src.logic.page\_parsers package**

## **Submodules**

**src.logic.page\_parsers.captcha\_response module**

**src.logic.page\_parsers.main\_page module**

**src.logic.page\_parsers.post\_response module**

## **Module contents**

**src.logic.proxy package**

## **Submodules**

**src.logic.proxy.proxy module**

## **Module contents**

## **Module contents**

**src.request\_params package**

## **Subpackages**

**src.request\_params.api package**

### **Submodules**

**src.request\_params.api.captcha\_service\_get module**

**src.request\_params.api.captcha\_service\_post module**

**src.request\_params.api.captcha\_service\_put module**

**src.request\_params.api.elpts\_get\_captcha module**

**src.request\_params.api.elpts\_main\_page module**

**src.request\_params.api.elpts\_post\_captcha module**

**src.request\_params.api.elpts\_post\_data module**

### **Module contents**

**src.request\_params.interfaces package**

### **Submodules**

**src.request\_params.interfaces.base module**

**src.request\_params.interfaces.captcha\_service\_base module**

**src.request\_params.interfaces.elpts\_authed module**

**src.request\_params.interfaces.elpts\_base module**

**src.request\_params.interfaces.elpts\_html module**

**src.request\_params.interfaces.elpts\_img module**

**src.request\_params.interfaces.elpts\_signed module**

### **Module contents**

### **Module contents**

## **src.runners package**

### **Submodules**

#### **src.runners.main module**

### **Module contents**

## **src.utils package**

### **Submodules**

#### **src.utils.hmac module**

#### **src.utils.utils module**

```
src.utils.utils.now()
```

### **Module contents**

### **Module contents**