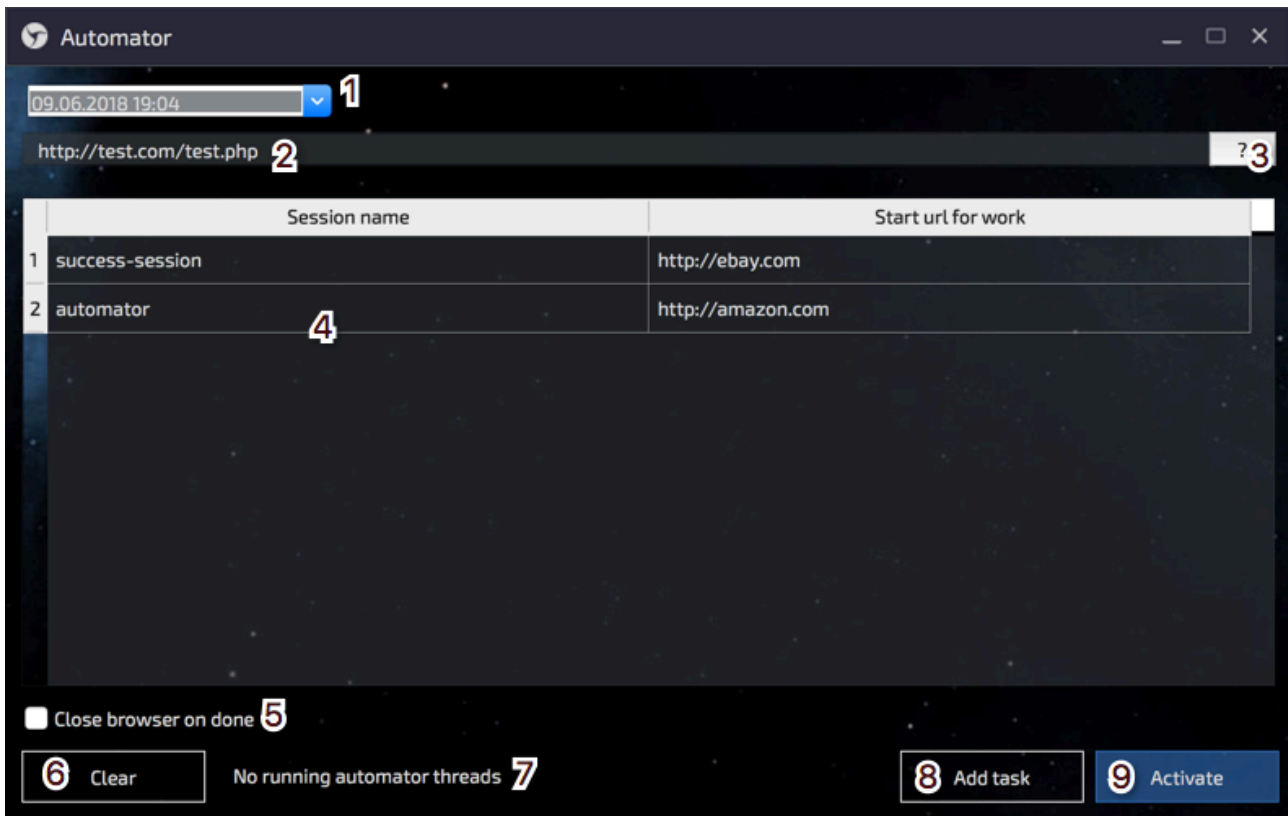


Автоматор v1.0



1. Описание

Автоматор - модуль для автоматизированного управления машинного серфинга по веб-сайтам с возможностью имитации человеческих действий. Модуль доступен в быстром меню. Краткое описание функционала модуля:

1. Селектор для установки времени запуска автоматизации, в данной версии браузер не должен быть закрыт для запуска по данному таймеру.
2. Управляющий URL. После каждого успешного (или неуспешного) запроса, модуль отправляет POST-запрос со следующими данными.

html - контент который получила вкладка или пусто соответственно если ничего не загрузилось по каким то причинам

url - вебсайт который загрузила вкладка

sess - имя сессии данной вкладки

Приводим пример php кода:

```
1 <?
2
3 $resp = array(
4     array("method"=>"scrollTo", "params"=>array(0,100)),
5     array("method"=>"mouseTo", "params"=>array(500,200)),
6     array("method"=>"mouseTo", "params"=>array(100,100)),
7     array("method"=>"waitFor", "params"=>array(6)),
8     array("method"=>"clickToElement", "params"=>array("#my-link")),
9 );
10
11 if (isset($_POST["html"])){
12     $filename = md5($_POST["sess"] . $_POST["url"]);
13
14     $h = $_POST["html"];
15     $s = $_POST["sess"];
16     $u = $_POST["url"];
17
18     file_put_contents($filename, $h."\n".$s."\n".$u);
19     echo json_encode($resp);
20 }
21
22 ?>
```

<?

```
$resp = array(
    array("method"=>"scrollTo", "params"=>array(0,100)),
    array("method"=>"mouseTo", "params"=>array(500,200)),
    array("method"=>"mouseTo", "params"=>array(100,100)),
    array("method"=>"waitFor", "params"=>array(6)),
    array("method"=>"clickToElement", "params"=>array("#my-link")),
);

if (isset($_POST["html"])){
    $filename = md5($_POST["sess"] . $_POST["url"]);

    $h = $_POST["html"];
    $s = $_POST["sess"];
    $u = $_POST["url"];

    file_put_contents($filename, $h."\n".$s."\n".$u);
    echo json_encode($resp);
}
```

?>

Ответ формируется в формате JSON с указанными методами API, каждая команда должна содержать название метода и параметры. Об этом чуть ниже.

3. При клике по данной кнопке можно получить краткий обзор методов API
4. Таблица команд автоматов, включает в себя название сессии и начальный URL. При вводе имени сессии работает подсказчик, то есть набирая название сессии в списке будут подставляться совпадения по существующим сессиям. Другой способ добавления сессий в автомат - это использовать менеджер сессий, отмечаете галочками нужные сессии и кнопку «отправить в автомат».
5. Флажок закрытия браузера после выполнения всех заданий
6. Статусная строка, тут будет отображаться различная информация в Runtime.
7. Очистка - данные автоматора автоматически сохраняются после активации, и при открытии браузера проставятся последние значения. Можете использовать если нужно очистить все
8. Добавить задание
9. Активировать, после нажатия есть возможность деактивировать, даже во время работы.

2. API

scrollTo

[x,y] - Скроллинг к координатам

scrollToElement

[elem] - Скроллинг к элементу - обратите внимание что указанный селектор должен быть уникален, то есть полный путь к элементу, если используются классы, ну или id если есть id.

mouseTo

[x,y] - имитация движения мыши к заданным координатам

mouseToElement

[elem] - движение мыши к элементу, правила те же что и при скроллинге

clickTo

[x,y] - клик в области с заданными координатами

clickToElement

[elem] - Данный метод использует комбинацию - скролл к элементу, движение мыши к нему и клик по нему.

waitFor

t - время в сек. Ожидание на странице в бездействии

exitWith

[code=0] - закончить работу с кодом. Если код положительный (в дальнейшем планируем описывать ситуации, для анализа) то успех, если нет то неуспех. Отрицательные коды могут быть получены в случае различных ошибок обработки DOM, к примеру не найден селектор, указанный в команде и прочее.

Пример JSON:

```
1  [
2    {
3      "method": "scrollToElement",
4      "params": [
5        "div.myElements"
6      ]
7    },
8    {
9      "method": "mouseToElement",
10     "params": [
11       "div.myElements < a:first-child"
12     ]
13   },
14   {
15     "method": "clickTo",
16     "params": []
17   }
18 ]
```

```
[
  {
    "method": "scrollToElement",
    "params": [
      "div.myElements"
    ]
  },
  {
    "method": "mouseToElement",
    "params": [
      "div.myElements < a:first-child"
    ]
  },
  {
    "method": "clickTo",
    "params": []
  }
]
```

Тоже самое в сокращенном виде:

```
1  [
2    {
3      "method": "clickToElement",
4      "params": [
5        "div.myElements < a:first-child"
6      ]
7    }
8  ]
```

```
[
  {
    "method": "clickToElement",
    "params": [
      "div.myElements < a:first-child"
    ]
  }
]
```

Все имитируемые движения не используют стандартный интерфейс - это главное отличие работы автоматора по сравнению с существующим веб-эмулятором. Все события используют корневые контролы Chrome и помечаются как **trusted**, то есть выглядит как будто это синтезировано от реального устройства.

Имейте ввиду что после ожидаемого перехода последующие методы не имеют смысла, так как после загрузки страницы автоматор снова обратится за порцией команд.

Вы можете скачать наиболее свежую версию **Linken Sphere** с нашего официального сайта <http://ls.tenebris.cc>

В случае, если у вас есть вопросы или пожелания - всегда рады слышать вас по нашим актуальным контактам:

Jabber: dev.tenebris@jabber.calyxinstitute.org

Telegram: [@devtnbrs](https://t.me/devtnbrs), [@devtenebris](https://t.me/devtenebris) (Telegram-channel), [@spherechat](https://t.me/spherechat) (Telegram-group)

E-mail: sphere@tenebris.cc