



Permission setting up Technical Documentation

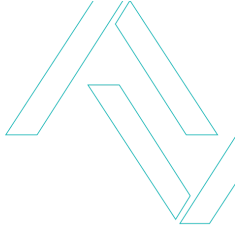
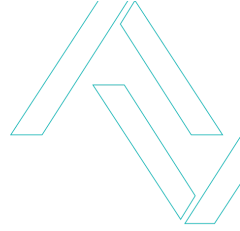


Table of contents

Table of contents	2
1. Methods and Parameters	3
2. Implementation and method calling	3
2.1 Method requestPermissions()	3
2.1.1 Code Implementation	3
2.2 Method checkPermissions()	4
2.2.1 Code Implementation	4
2.2.2 Callback Return Payload	4





1. Methods and Parameters

Callback Function	Parameters	Data Type	Callback Return	Description
requestPermissions	-	void	-	Helps to manage permission sequentially
checkPermissions	-	Object	{ "callback": "check_permission", "state": true }	If permission allow then state is true else return false

2. Implementation and method calling

As before, you can follow the same steps to install the SpeedTest plugin. Below are the implementation and execution steps for **requestPermissions** and **checkPermissions** methods

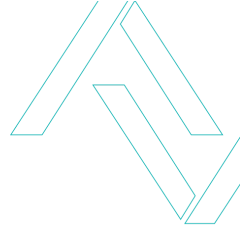
2.1 Method requestPermissions()

In the previous plugin, the **startTesting** method was used to initiate the speed test. Before the test execution, all required permissions were requested, and permission handling was integrated into the process. A single method is now exposed to handle the permission request

2.1.1 Code Implementation

```
window.plugins.SpeedTest.requestPermissions(  
  null,  
  (success) => console.log(success),  
  (error)    => console.error(error)  
);
```





2.2 Method checkPermissions()

2.2.1 Code Implementation

```

window.plugins.SpeedTest.checkPermissions (
    null,
    (success) => {
        console.log(success);
        //Add you custom logic
    },
    (error) => {
        console.error(error);
    }
);

```

2.2.2 Callback Return Payload

Returns true if all permissions are granted; otherwise, returns false.

```

{
  "callback": "check_permission",
  "state": true
}

```

```

{
  "callback": "check_permission",
  "state": false
}

```

Using these two methods, permissions can be managed manually based on requirements

