AstroFarm

User Guide

Ver 0.44.14

Table of Contents

Table of Contents 2

Introduction 4

Key Features 4

A Quick Look At The AstroFarm Console 5

Device Enrollment 9

Android Device Enrollment 9

Prerequisites 9

Setup ADB on the host machine 9

ADB Setup On Windows Devices 9

ADB setup on Linux machine 10

ADB Setup on macOS Device 10

Enable Developer Option and USB Debugging on the device 11

Use Windows Machine to Enroll Devices Using an AstroFarm Agent 12

Use Linux machine to enroll devices using an AstroFarm Agent 14

Use Mac to enroll devices using an AstroFarm Agent 15

iOS Device Enrollment 18

View the Contributor Status 18

Manage Android Devices on the AstroFarm 19

Remote Actions on Android Devices 19

Remote Screen Recording of a Device 22

Remotely Launch Shortcut Applications 24

Install an Application on a Device 24

Remotely Debug a Device 25

Remove Device(s) from the AstroFarm Console 25

Remotely Monitor CPU and Memory Usage 26

Manage iOS Devices on the AstroFarm 28

Remote Actions on iOS Devices 28

Remotely Launch Shortcut Applications 30

Remotely Debug a Device 31

Remove Device(s) from the AstroFarm Console 31

Weekly Device Usage Updates By Email 33

Integrate AstroFarm with Appium 35

Manual Method 35

Use Appium to Enable Automation in AstroFarm for Android Devices 35

Use Appium to Enable Automation in AstroFarm for iOS Devices 39

Generate Reports 43

Actions in Reports 43

User Management 44

Add a User to AstroFarm 44

Remove a User from AstroFarm 45

Grant Admin Privileges to an AstroFarm User 46

Remove Admin Privileges from an AstroFarm User 46

Add a Contributor to AstroFarm 46

Remove a Contributor from AstroFarm 47

License Management 48

Release Notes Updates 49

Introduction

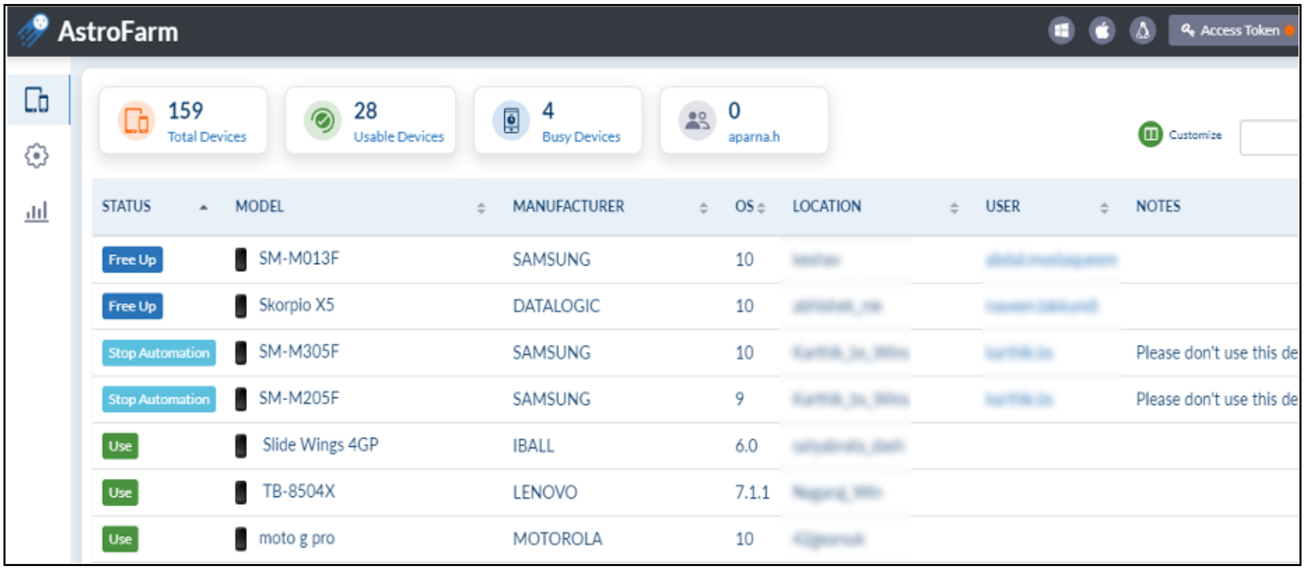
AstroFarm is a private device sharing platform using which you can share, manage, and control the company's Android/iOS devices that are used for testing, development, and demo purposes. It empowers your employees to access your company's devices from anywhere and at any time, without the need for physical access.

Key Features

- Easy and Quick Device Enrollment to Your Device Farm
- Remote Access to Device Settings
- Remote Source-Level Debugging
- Automation Testing on Devices
- Device System Logs Capture
- Remote File Upload Using File Explorer
- Enable Automation Using Appium

A Quick Look At The AstroFarm Console

AstroFarm is a web-based console that provides a centralized view of all enrolled devices and allows users to remotely manage these devices.



159 Total Devices

28 Usable Devices

4 Busy Devices

0 aparna.h

STATUS	MODEL	MANUFACTURER	OS	LOCATION	USER	NOTES
Free Up	SM-M013F	SAMSUNG	10	[redacted]	[redacted]	
Free Up	Skorpio X5	DATALOGIC	10	[redacted]	[redacted]	
Stop Automation	SM-M305F	SAMSUNG	10	[redacted]	[redacted]	Please don't use this de
Stop Automation	SM-M205F	SAMSUNG	9	[redacted]	[redacted]	Please don't use this de
Use	Slide Wings 4GP	IBALL	6.0	[redacted]		
Use	TB-8504X	LENOVO	7.1.1	[redacted]		
Use	moto g pro	MOTOROLA	10	[redacted]		

Settings	Description
Devices	Displays all enrolled devices on the console. The user can assume control of a single or multiple devices from here.
Settings	The settings section has the following options: ● **General** - This allows the user to set the general settings such as Language, Date, Device Grid Settings (Show Only Online Devices/ Show All devices), and Email Address separator.

	● **Keys** - You can generate access tokens and ADB key from this section. ● **Account** - View your AstroFarm account details. To know more, click here. ● **Devices** - Lists all the devices enrolled in the AstroFarm console. ● **Users** - Lists all the users who have/don’t have admin privileges. ● **Contributors** - Lists all the contributors who have enrolled the devices to the AstroFarm console. ● **iOS Agent** - You need to follow the steps of the tool you use, for the resigning process. This is mandatory for enrolling iOS devices in AstroFarm. ● **Release History** - Users can find the latest release updates in this section.
📊 **Reports**	Generates device usage report of a specific user or all users enrolled in the AstroFarm console. To know more, click here.

Status	Displays the device’s status: ● **Enrolling** - The device is in enrolling state. Once enrolled, the device status will change to **Use**. ● **Free Up** - The device is being used by another user. By clicking this option, the device will be freed, and the status will change to **Use**. ***Note:*** *Only users with admin privileges can free up devices that are being used by other users.* ● **Use** - The device is free to use. By clicking this option, the user can take remote control of the device and start using it. ● **Unplugged** - The device is not connected to the host machine or the USB debugging option is disabled. ● **In Use** - The device is being used for automation. This option will be available for users who don’t have admin privileges. ● **Stop Automation** - The device is being used for automation. Type the text **Automation** in the **Notes** column of the device that you want to use for automation. The device status will change to **Stop Automation** when you take the device using the **Use** button**.** To know how to automate using Appium, click here. ***Note:*** *Users with admin privileges can free up devices when they are in* ***Stop Automation*** *status.*

Customize	Users can customize the required columns on the device grid.
Search	Narrow your search based on the fields displayed on the device grid.

Device Enrollment

In this section, you will learn how to enroll Android and iOS devices to the AstroFarm.

Android Device Enrollment

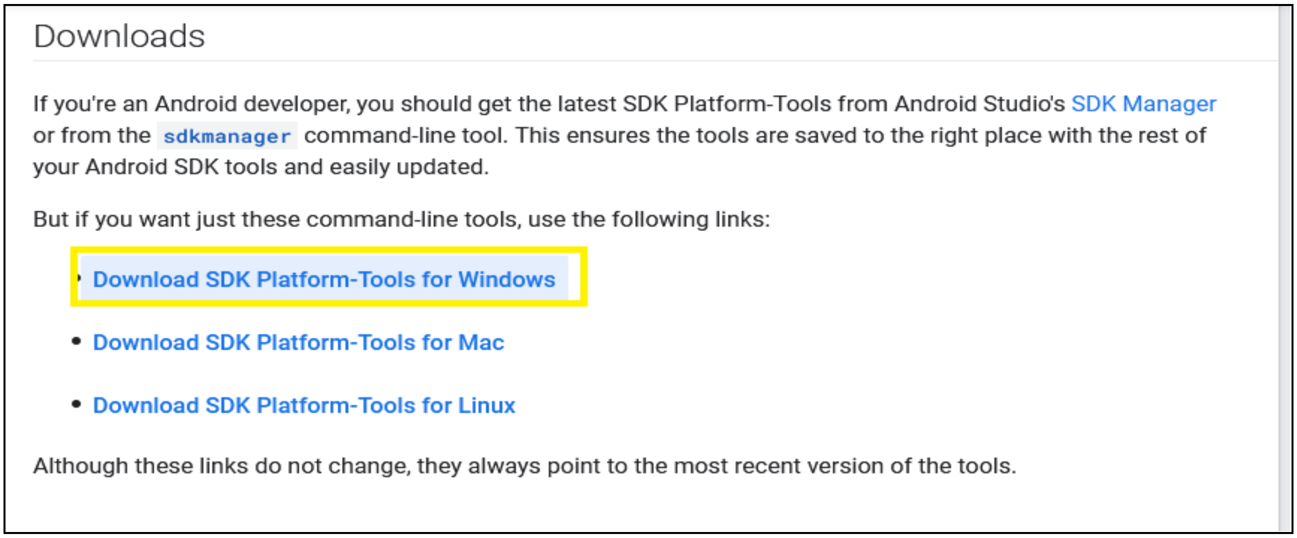
Prerequisites

Setup ADB on the host machine

First, you need to set up ADB on the host machine from where you will be enrolling Android devices. You can use Windows, Linux, or macOS devices as host machines.

a. ADB Setup On Windows Devices

1. Go to https://developer.android.com/studio/releases/platform-tools.
2. Select **Download SDK Platform-Tools for Windows** and download the platform-tools.



Downloads

If you're an Android developer, you should get the latest SDK Platform-Tools from Android Studio's SDK Manager or from the `sdkmanager` command-line tool. This ensures the tools are saved to the right place with the rest of your Android SDK tools and easily updated.

But if you want just these command-line tools, use the following links:

- Download SDK Platform-Tools for Windows
- Download SDK Platform-Tools for Mac
- Download SDK Platform-Tools for Linux

Although these links do not change, they always point to the most recent version of the tools.

Once downloaded, extract it at an accessible location. This package contains an adb.exe file, the path of which we will need to configure at a later stage.

b. ADB setup on Linux machine

1. Open Terminal on your Linux machine.
2. Execute one of the following commands based on the Linux system from where you want to complete the installation of ADB on your system:
   a. Debian-based Linux users can type the following command to install ADB:

      ```
      sudo apt-get install adb
      ```

   b. Fedora/SUSE-based Linux users can type the following command to install ADB:

      ```
      sudo yum install android-tools
      ```

c. ADB Setup on macOS Device

1. Go to https://developer.android.com/studio/releases/platform-tools.
2. Select **Download SDK Platform-Tools for Mac** and download the platform-tools.
3. Once Downloaded, extract it to an accessible location.
4. Move it to the location where you won't accidentally delete them.

   ```
   mkdir ~/.android-sdk-macosx

   mv platform-tools/ ~/.android-sdk-macosx/platform-tools
   ```

5. Add platform-tools to the path.

   ```
   echo 'export PATH=$PATH:~/.android-sdk-macosx/platform-tools/' >>
   ~/.bash_profile
   ```

6. Refresh your bash_profile (or restart the Terminal app).

```
source ~/.bash_profile
```

Enable Developer Option and USB Debugging on the device

Ensure USB debugging is enabled on the Android device that you want to connect with AstroFarm.

To enable USB debugging on a device,

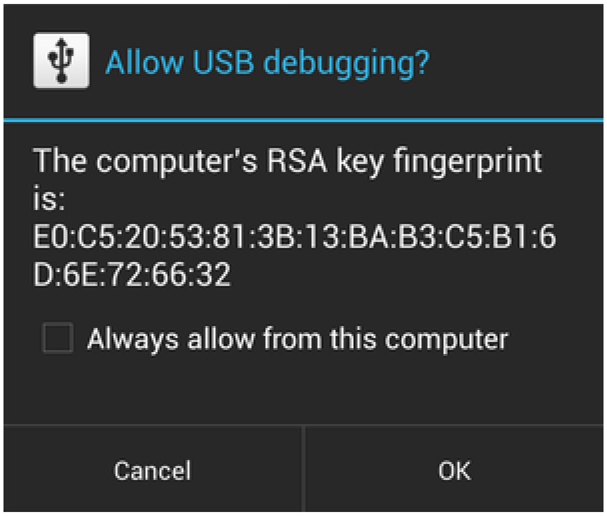
1. On the Android device, open the **Settings** application.
2. Tap **About Phone** option at the bottom of the list.

   In case it is not present, go to **System** > **About Phone**.

3. Tap the **Build Number** option 7 times to enable **Developer Mode**.

   You can see a toast message once it is enabled.

4. Now go back to the initial **settings** screen and tap **Developer Options**.
5. Enable the **USB Debugging mode** option on the device.



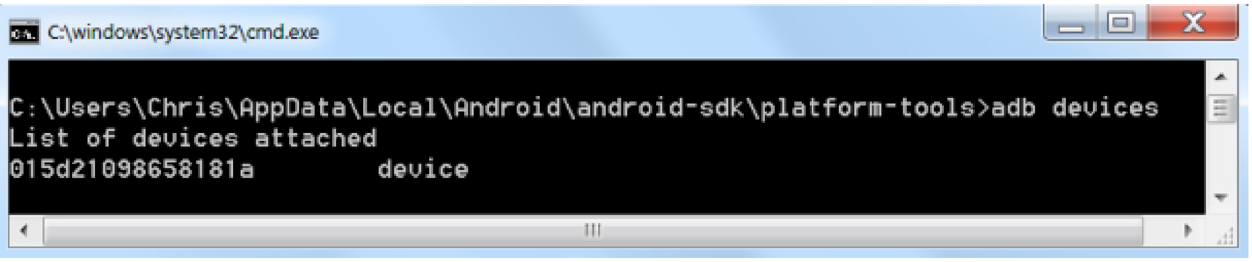
Allow USB debugging?

The computer's RSA key fingerprint is:
E0:C5:20:53:81:3B:13:BA:B3:C5:B1:6
D:6E:72:66:32

□ Always allow from this computer

6. Open the Command window and enter **adb devices**.

   Watch the device's screen for authorization messages and allow the connection.



```
C:\Users\Chris\AppData\Local\Android\android-sdk\platform-tools>adb devices
List of devices attached
015d21098658181a        device
```

Use Windows Machine to Enroll Devices Using an AstroFarm Agent

To enroll devices using an AstroFarm agent,

1. Log into the **AstroFarm** portal.
2. Download AstroFarm agent for Windows OS by clicking the Windows icon on the right top of the console. A .exe file will download.
3. Install downloaded .exe file on your host computer running Windows OS.

4. Go back to the **AstroFarm** portal and click **Copy Access Token** (located at the top right of the console) and click **Copy** to copy the access token.

   Note: *Access Token is a 36 character string that will be auto-generated while adding a Contributor in the AstroFarm console.*

5. Launch the **AstroFarm** application (shortcut will be present on the desktop).

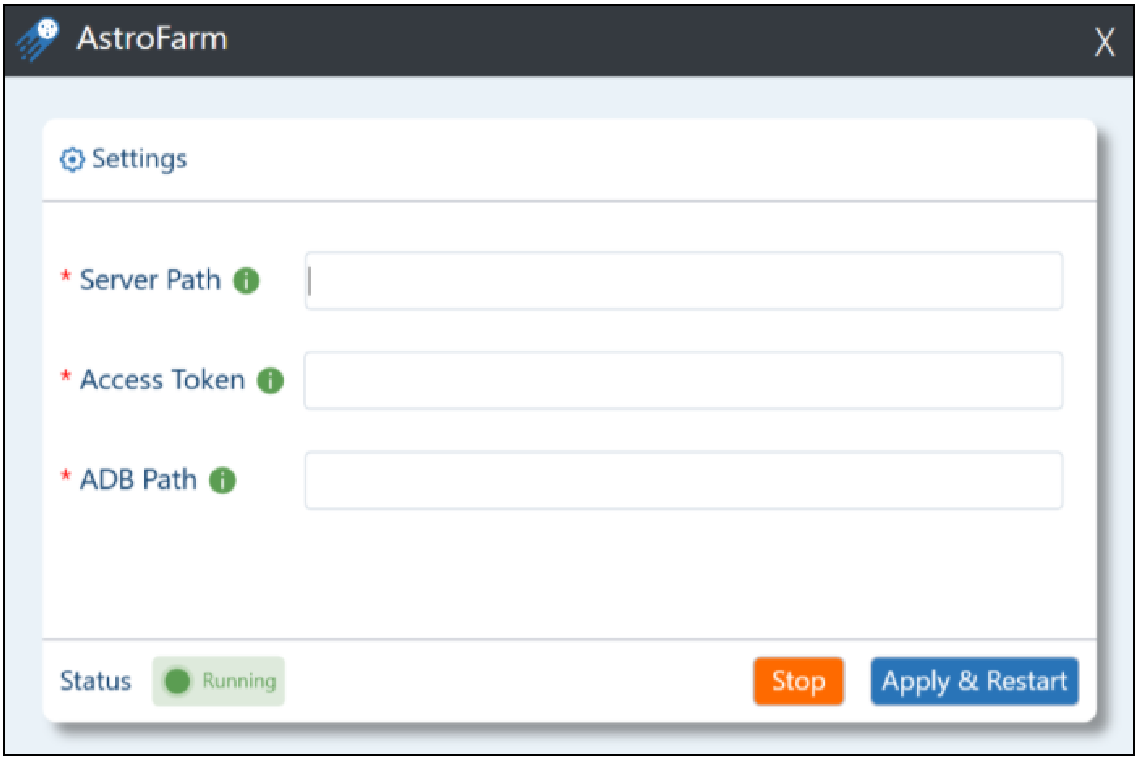
6. Enter the following details:

   a. **Server Path** - URL of the AstroFarm server. Please contact your administrator if you have any trouble finding it.

   b. **Access Token** - Paste the access token that you copied in step no.4.

   c. **ADB Path** - Enter the path of the folder in which the ADB.exe file is saved on your machine (downloaded in prerequisite).

For example: D:\adb\platform-tools_r30.0.4-windows\platform-tools\adb.exe.



7. Click **Apply** & **Restart**.

   When the device is getting enrolled to the AstroFarm console, it shows the status as Enrolling and, after 2 to 3 minutes the status will change to Use. Now, you can start controlling the devices remotely.

Use Linux machine to enroll devices using an AstroFarm Agent

To enroll devices using an AstroFarm agent,

1. Log into the **AstroFarm** portal and click the Linux icon on the right top of the console.

2. Click **Copy** on the prompt shown (choose Linux or Linux ARM based on your host machine system).

3. Open the Terminal and paste the copied command and execute it.

4. Enter the required sudo credentials, if requested.

5. Go back to the **AstroFarm** portal and click **Copy Access Token** (located at the top right of the console) and click **Copy** to copy the access token.

6. Enter **Access Token** on the Terminal when requested.

7. Plug in the Android device to your system and enable USB Debugging on the device.

8. Open the **Command** window on the Linux machine and type **adb devices**.

   An authorization message will appear on the screen.

9. Select **Always allow from this computer** and click **Ok** to allow the connection.

   The attached device will auto-enroll into AstroFarm.

   When the device is getting enrolled to the AstroFarm console, it shows the status as Enrolling and, after 2 to 3 minutes the status will change to Use. Now, you can start controlling the devices remotely.

Use Mac to enroll devices using an AstroFarm Agent

To enroll devices using an AstroFarm agent,

1. Log into the **AstroFarm** portal.

2. Download AstroFarm agent for macOS by clicking the MAC icon located at the top right of the console.

   A package file will download.

3. Install the downloaded package file on your host computer running macOS.

4. Go back to the **AstroFarm** portal and click **Copy Access Token** (located at the top right of the console) and click Copy to copy the access token.

5. Launch the **AstroFarm** application (shortcut will be present on the desktop).

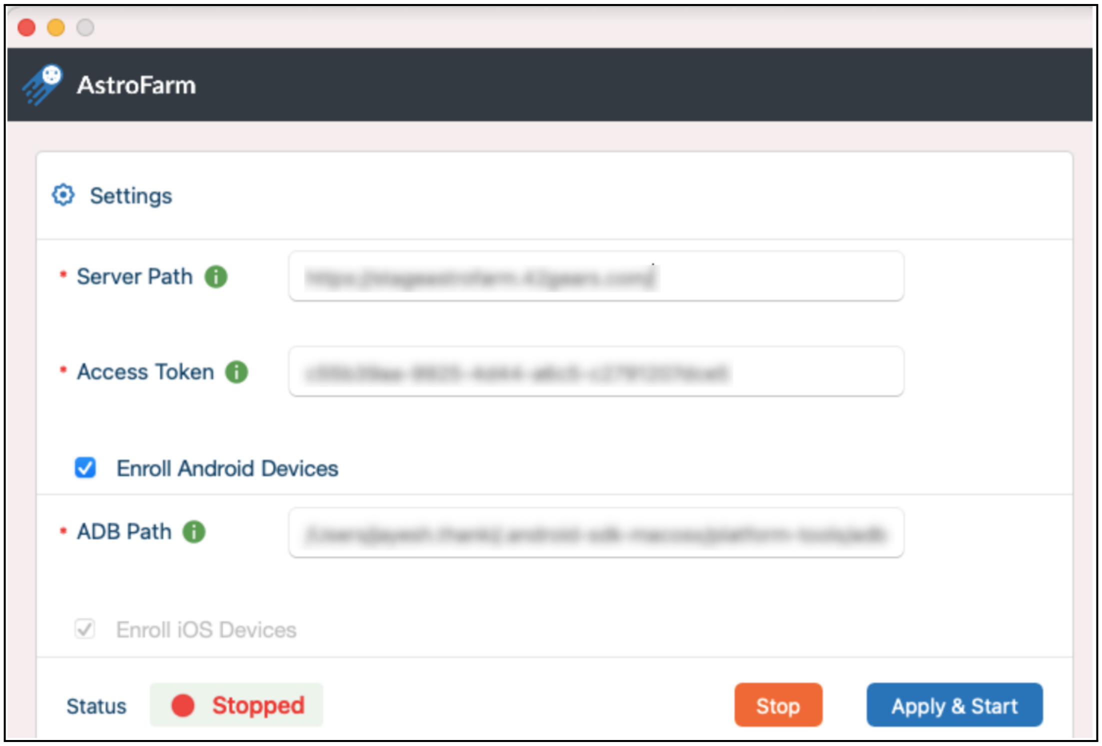
6. Enter the following details:

   a. **Server Path** - URL of the AstroFarm server. Please contact your administrator if you have any trouble finding it.

   b. **Access Token** - Paste the access token that you copied in step no.4.

   c. **Enroll Android Devices** - Select this option for Android device enrollment

   d. **ADB Path** - Enter the path of the folder in which the ADB.exe file is saved on your machine.



AstroFarm

Settings

* Server Path: [redacted]

* Access Token: [redacted]

☑ Enroll Android Devices

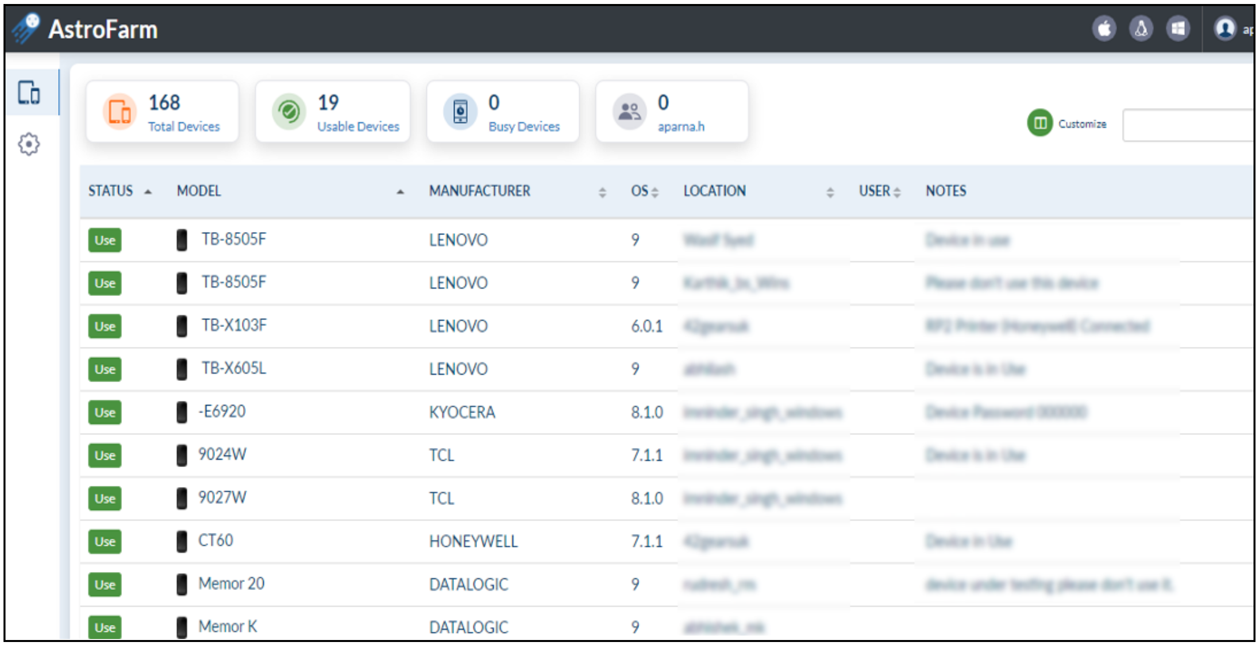
* ADB Path: [redacted]

☑ Enroll iOS Devices

Status: ● Stopped

7. Click **Apply** & **Restart**.

When the device is getting enrolled to the AstroFarm console, it shows the status as **Enrolling** and, after 2 to 3 minutes the status will change to **Use**. Now, you can start controlling the devices remotely.



STATUS	MODEL	MANUFACTURER	OS	LOCATION	USER	NOTES
Use	TB-8505F	LENOVO	9	[redacted]		[redacted]
Use	TB-8505F	LENOVO	9	[redacted]		[redacted]
Use	TB-X103F	LENOVO	6.0.1	[redacted]		[redacted]
Use	TB-X605L	LENOVO	9	[redacted]		[redacted]
Use	-E6920	KYOCERA	8.1.0	[redacted]		[redacted]
Use	9024W	TCL	7.1.1	[redacted]		[redacted]
Use	9027W	TCL	8.1.0	[redacted]		
Use	CT60	HONEYWELL	7.1.1	[redacted]		[redacted]
Use	Memor 20	DATALOGIC	9	[redacted]		[redacted]
Use	Memor K	DATALOGIC	9	[redacted]		

Once the device is enrolled successfully on the AstroFarm console, the status on the host machine will show as **Running**.

iOS Device Enrollment

To know how to enroll iOS devices into AstroFarm, click here.

View the Contributor Status

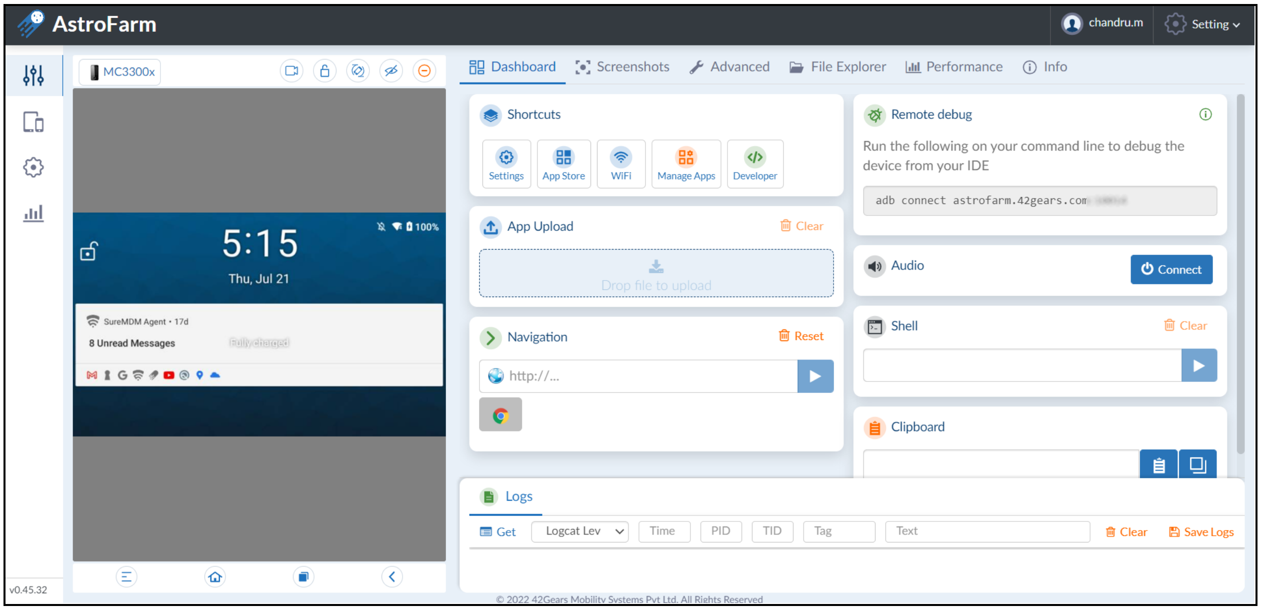
Once the device is added to AstroFarm through the host machine (Windows/Mac/Linux) as a contributor, it indicates the following contributor status on the console:

- **Green** - Host machine is connected
- **Orange** - Host machine is disconnected

Manage Android Devices on the AstroFarm

This section contains topics about managing Android devices remotely on the AstroFarm console. It covers topics such as launching shortcut applications, installing an application, and more.

Remote Actions on Android Devices



Run the following on your command line to debug the device from your IDE

```
adb connect astrofarm.42gears.com:[redacted]
```

The following actions can be performed remotely on a device from the AstroFarm console:

Settings	Description
Touch Events	Touch actions are allowed on the device
📹	Initiate screen recording session on a device. To know more, click here.

🔓	Unlock the device.
🔄	Rotate the device screen.
👁	Show or hide the device screen.
⊖	Free up the device you are using.
Dashboard	
Shortcut Applications	Certain shortcut applications can be launched remotely on a device. To know more, click here.
App Upload	Remotely install an application on a device. To know more, click here.
Navigate to URL	Use this option to launch the specified URL on the device.
Remote Debug	Use this option to remotely debug the device from your IDE (Integrated Development Environment). To know more, click here.

Audio Streaming	Use this option to play the audio of an application on the AstroFarm console that is getting streamed on the device. 📝 ***Note:*** *The device must run Android 10 or later for this feature to work.*
Shell Commands	Use this option to execute script commands remotely on the device.
Clipboard	Use this option to copy or paste the data from or to the device.
Screenshots	Use this option to capture the screenshot of a device.
Advanced Settings	Advanced settings have the following options: ● Remotely turn on the screen of the device ● Launch the camera ● Control the volume of the device ● Manage media controls ● Restart the device
File Explorer	Use this option to download/upload the files from the device. User can create/rename the folders and upload the files.

Performance	Use this option to monitor the CPU and memory performance of the device. To know more, click here.
Device Info	Info tab provides device details such as the battery, hardware, platform, network, display, and SIM.
Logs	Use this option to view logcat messages of the device. Click **Get** to list all the logcat messages of the device.

Remote Screen Recording of a Device

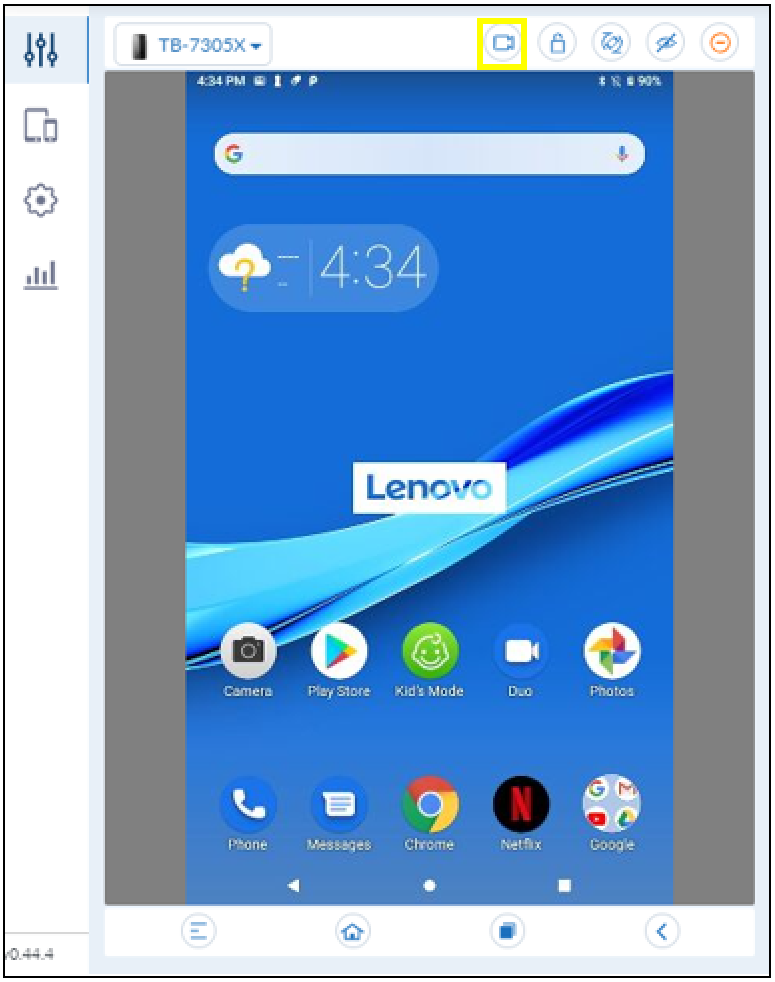
The users can initiate screen recording sessions of a device remotely.

To remotely initiate screen recording session of a device,

1. Log into the AstroFarm console.
2. Go to Devices.

   A list of enrolled devices will be displayed on the device grid with different statuses.
3. Search for the device and click Use.

   The following screen will appear.



4. Click 📹 .

   A screen recording session gets initiated.

5. Click the **Stop** button whenever you want to stop the screen recording session.

   A recording session file will download in WEBM format.

Note: *If a device is inactive for more than 30 minutes during the screen recording session, then the session will be stopped automatically.*

Remotely Launch Shortcut Applications

To remotely launch applications on a device,

1. Log into the AstroFarm console.
2. Go to Devices.

   A list of enrolled devices will be displayed on the device grid with different statuses.
3. Select the required device with the status **Use**.
4. Navigate to the **Dashboard** > **Shortcuts**.

   The following applications can be launched remotely on the device from the AstroFarm console:

   - Settings
   - App Store
   - Wi-Fi
   - Manage Apps
   - Developer

Install an Application on a Device

To install an application on a device remotely using AstroFarm,

1. Log into the AstroFarm console.

2. Go to Devices.

   A list of enrolled devices will be displayed on the device grid with different statuses.

3. Select the required device with the status **Use**.

4. Navigate to the **Dashboard** > **App Upload**.

5. Browse and select a .apk file.

   The application will be installed and launched on the device.

Remotely Debug a Device

To remotely debug a device from your IDE,

1. Log into the AstroFarm console.

2. Go to Devices.

3. A list of enrolled devices will be displayed on the device grid with different statuses.

4. Select the required device with the status **Use**.

5. Navigate to the **Dashboard** > **Remote Debug**.

Copy the command displayed under Remote Debug and run it on your Command prompt to connect the AstroFarm-enrolled device with your machine. You will get access to the device and can start controlling it using ADB commands.

Remove Device(s) from the AstroFarm Console

To remove a device from the AstroFarm console,

1. Log into the AstroFarm console.

2. Go to ⚙ **Settings** > **Devices**.

3. You can see a list of all devices enrolled in the AstroFarm console.

Search for the device and click **Remove** for the device that you want to delete.

4. If you want to remove devices in bulk from the AstroFarm console, then continue with the below steps.

   - Search for the device by name.

     A list of devices will be displayed based on the search criteria.

   - Click **Remove** > **Confirm Remove**.

   This will remove all the listed devices.

Note: *If you want to enroll the removed device again, you need to go through the complete device enrollment process.*

Remotely Monitor CPU and Memory Usage

Users can monitor the utilization of the device's CPU and memory in the **Performance** section. Based on the device usage, a detailed graph of CPU and memory performance will be displayed.

To view the usage graph, follow these steps:

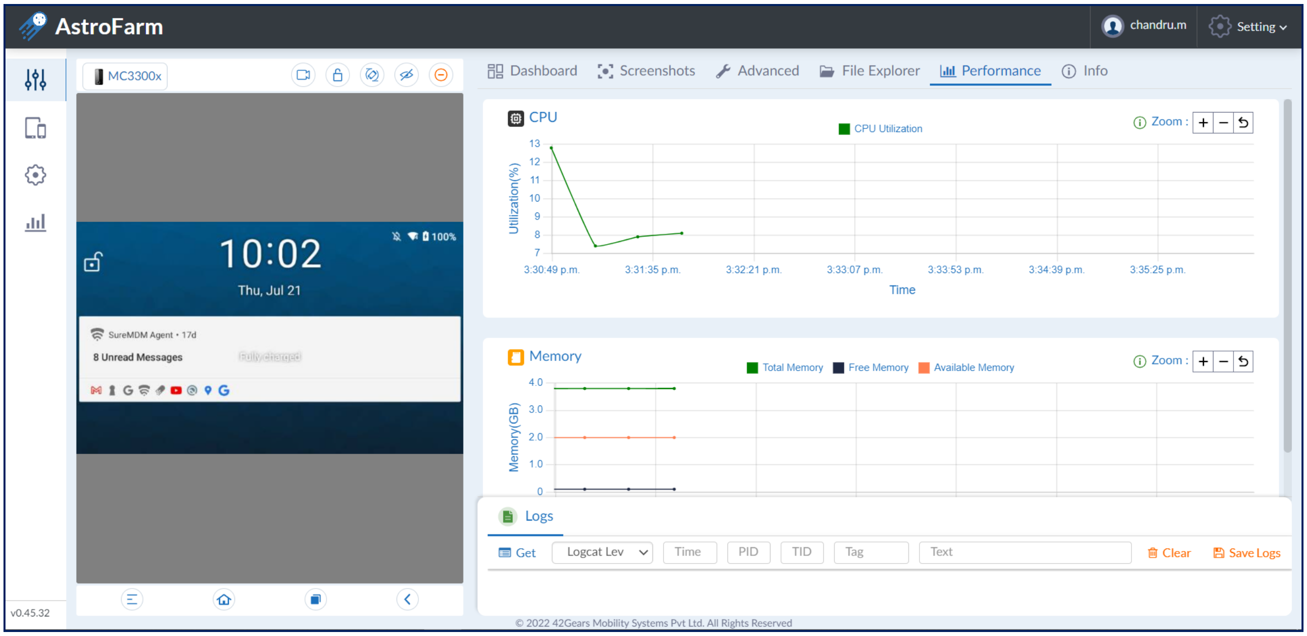
1. Log into the **AstroFarm console**.
2. Go to **Devices**.

A list of enrolled devices will be displayed on the device grid with different statuses.

3. Select the required device with the status **Use**.

4. Go to the **Performance** tab on the remote screen.

   - In this section, users can view the individual graphs for CPU and memory performance.
   - Use **Zoom In** and **Zoom Out** options to view the graph effectively.
   - Users can view the history of the usage graph for that device session under the **Reports** section.



Manage iOS Devices on the AstroFarm

This section contains topics about managing iOS devices remotely on the AstroFarm console. It covers topics such as launching shortcut applications, installing an application, remove devices from the AstroFarm and more.

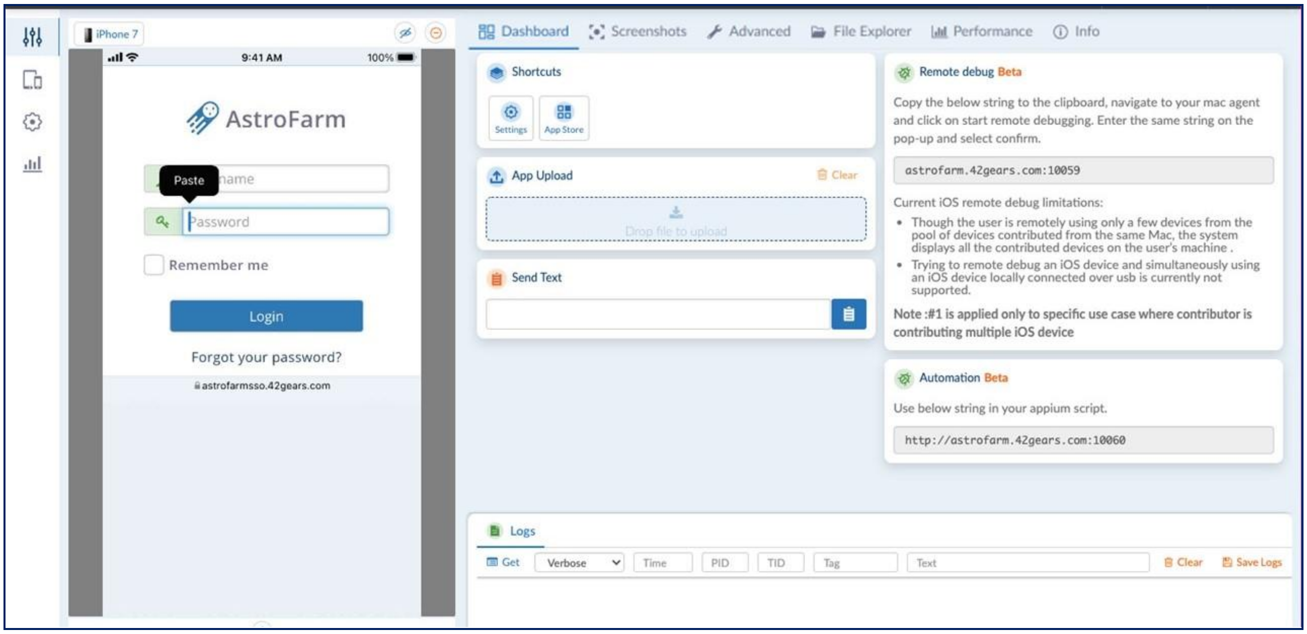
Remote Actions on iOS Devices

Following actions can be performed remotely on an iOS device from the AstroFarm console:

Settings	**Description**
Touch Events	In order to work touch events on the iOS devices, on the AstroFarm console, go to **Settings** > **iOS Agen**t and follow the steps of the tool you use for the resigning process. This is mandatory for enrolling iOS devices in AstroFarm.
⊖	Free up the device you are using.
👁	Show or hide the device screen.
Dashboard	
Settings	Launch and configure device's settings.

AppStore	Launch and configure AppStore settings.
App Upload	Install signed apps (ipa files) remotely on a device. To know more, click here.
Send Text	This option allows users to enter the text and apply it on the device.
Remote Debug	Use this option to remotely debug the device from your IDE (Integrated Development Environment). To know more, click here.
Automation	Use this option to enable the automation using appium. To know more, click here.
Screenshots	Use this option to capture the screenshot of a device.
Advanced Settings	Advanced settings have the following options: ● Remotely turn on the screen of the device ● Launch the camera ● Control the volume of the device ● Manage media controls

Info	Info tab provides device details such as the battery, hardware, platform, network, display, and SIM.
Logs	Use this option to view logcat messages of the device. Click **Get** to list all the logcat messages of the device.



Copy the below string to the clipboard, navigate to your mac agent and click on start remote debugging. Enter the same string on the pop-up and select confirm.

```
astrofarm.42gears.com:10059
```

Current iOS remote debug limitations:

- Though the user is remotely using only a few devices from the pool of devices contributed from the same Mac, the system displays all the contributed devices on the user's machine .
- Trying to remote debug an iOS device and simultaneously using an iOS device locally connected over usb is currently not supported.

Note :#1 is applied only to specific use case where contributor is contributing multiple iOS device

Use below string in your appium script.

```
http://astrofarm.42gears.com:10060
```

Remotely Launch Shortcut Applications

To remotely launch applications on a device,

1. Log into the **AstroFarm** console.
2. Go to **Devices**.

   A list of enrolled devices will be displayed on the device grid with different statuses.

3. Select the required device with the status **Use**.
4. Navigate to the **Dashboard > Shortcuts**.

    The following applications can be launched remotely on the device from the AstroFarm console:

    - Settings
    - App Store

Remotely Debug a Device

To remotely debug a iOS device follow these steps:

1. Log into the **AstroFarm** console.
2. Go to 📱 **Devices**.

    A list of enrolled devices will be displayed on the device grid with different statuses.

3. Select the required iOS device with the status **Use**.
4. Navigate to the **Dashboard > Remote Debug.**

    Copy the command displayed under **Remote Debug URL.**

5. Launch **AstroFarm mac Agent** on your mac device.
6. On the **AstroFarm mac Agent**, click on **Start Remote Debugging.**
7. Paste the **Remote Debug URL** and click on **Confirm**.

Admins can now perform the remote operations on the device using **Xcode**.

Remove Device(s) from the AstroFarm Console

To remove device(s) from the console,

1. Log into the **AstroFarm** console.
2. Go to **Settings > Devices.**

   You can see a list of all devices enrolled in the **AstroFarm** console.

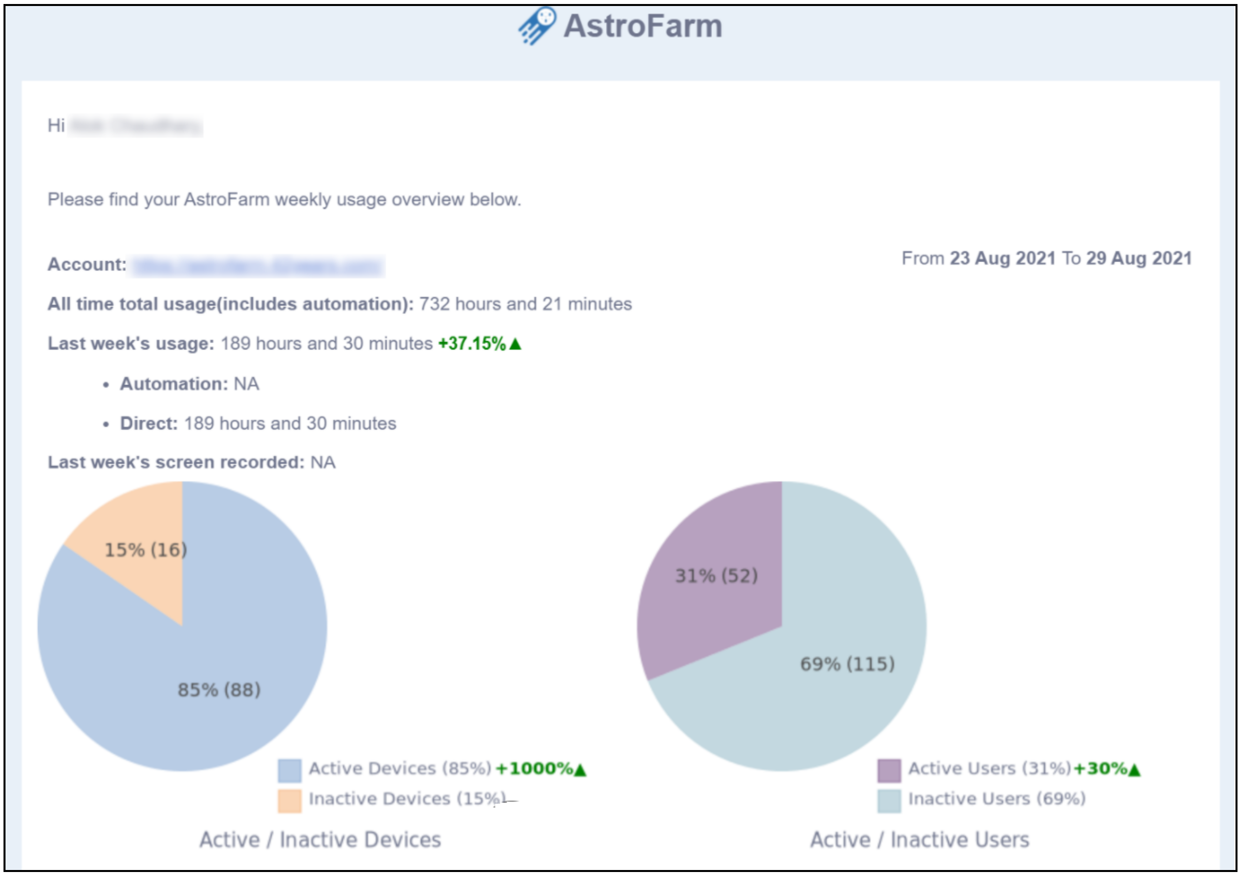
3. Search for the device and click **Remove** for the device that you want to delete.
4. If you want to remove devices in bulk from the **AstroFarm** console then continue with the below steps.
   - Search for the device by name.
   - A list of devices will be displayed based on the search criteria.
   - Click on **Remove > Confirm Remove.**

   This will remove all the listed devices.

Note: *If you want to enroll the removed device again, you need to go through the complete device enrollment process.*

Weekly Device Usage Updates By Email

The admin will automatically receive the weekly device usage details to their registered email account.



AstroFarm

Hi [redacted]

Please find your AstroFarm weekly usage overview below.

Account: [redacted]

From **23 Aug 2021** To **29 Aug 2021**

All time total usage(includes automation): 732 hours and 21 minutes

Last week's usage: 189 hours and 30 minutes **+37.15%▲**

- **Automation:** NA
- **Direct:** 189 hours and 30 minutes

Last week's screen recorded: NA

The email will have the following details:

- **Account** - Displays your domain name.
- **All time total usage (includes automation)** - Displays total device usage time for an account. This includes the time involved in automation.

- **Last week's usage** - Displays total device usage time for the specified week. This includes the time involved in automation.
- **Last week's screen recorded** - Displays the total time involved in screen recording by all users for the specified week.
- **Pie Charts** - The pie charts depict the total number of **Active/Inactive Users/Devices** on the console for the specified week.
- **Newly Added Devices** - Displays a list of devices added to the AstroFarm console for the specified week.
- **Last Week's top devices** - Displays a list of the top five devices that have the highest usage for the specified week.
- **Last week's top users** - Displays a list of the top five users of the devices for the specified week.
- **Last week's top contributors** - Displays a list of the top five contributors of the devices for the specified week.

Note: *1. Admin can unsubscribe from getting a weekly usage email by clicking the unsubscribe link given in the email.*

*2. The increase or decrease in the values (in percentage) for **Last week's usage, Active Users, Active Devices, etc.,** will be displayed in green or red color.*

Integrate AstroFarm with Appium

The devices enrolled to AstroFarm can be used for automation with Appium (automation tool used to run scripts and native applications on test devices) in following ways:

- Using Manual Method
- Using APIs

Manual Method

This section helps admin to enable Appium manually on their devices using AstroFarm console.

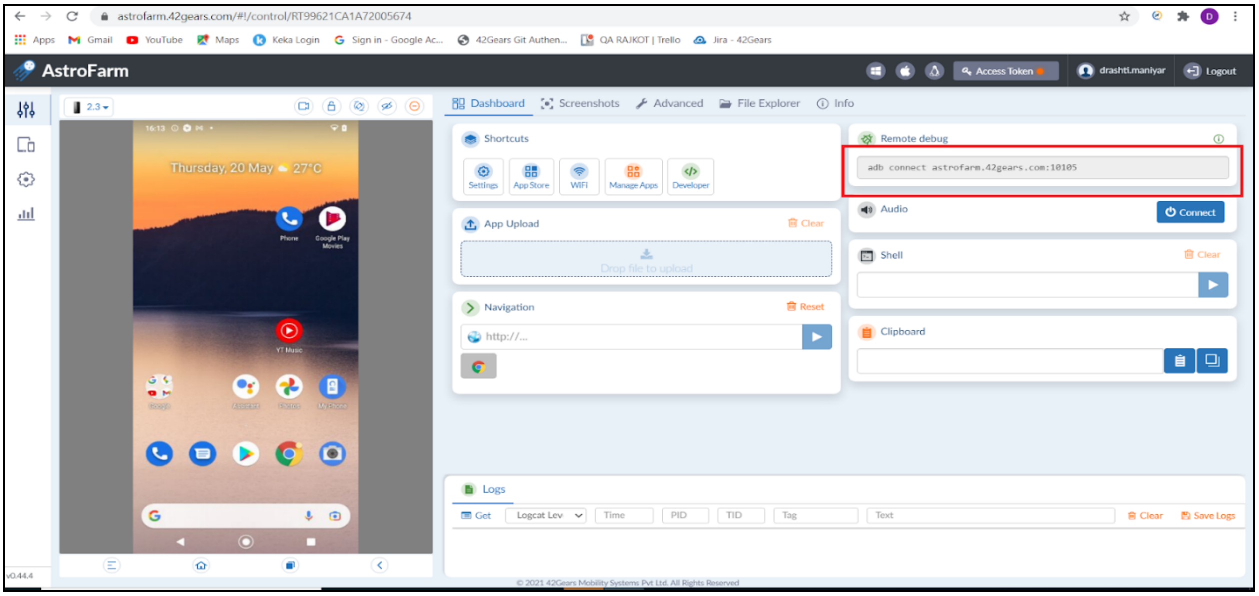
Use Appium to Enable Automation in AstroFarm for Android Devices

Use Appium to Enable Automation in AstroFarm for iOS Devices

Use Appium to Enable Automation in AstroFarm for Android Devices

To enable automation in **AstroFarm** using **Appium**,

1. Log into the **AstroFarm** console.
2. Search for the device that you want to use for automation and click **Use**.
3. Navigate to the **Dashboard** and copy the command from **Remote debug**.



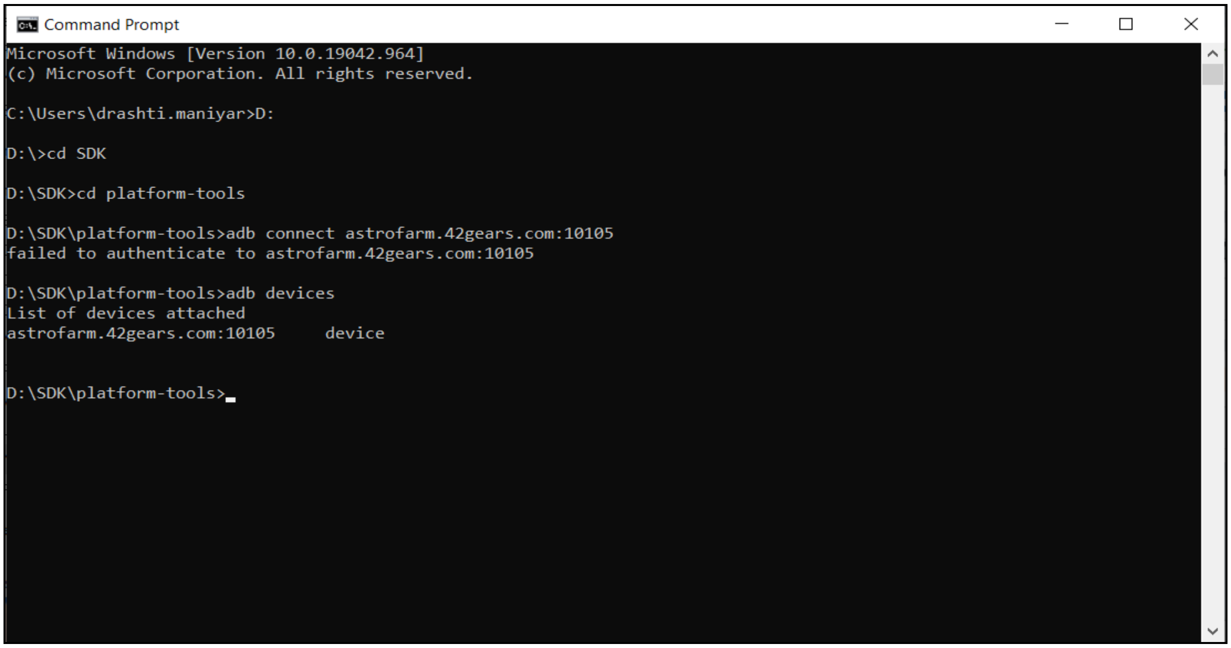
```
adb connect astrofarm.42gears.com:10105
```

4. On the Command prompt, go to the adb path and paste the command copied in the previous step.

Note: *ADB version should be the latest.*

5. Check if the device is connected through adb by typing the following command in the Command prompt:

```
adb devices
```

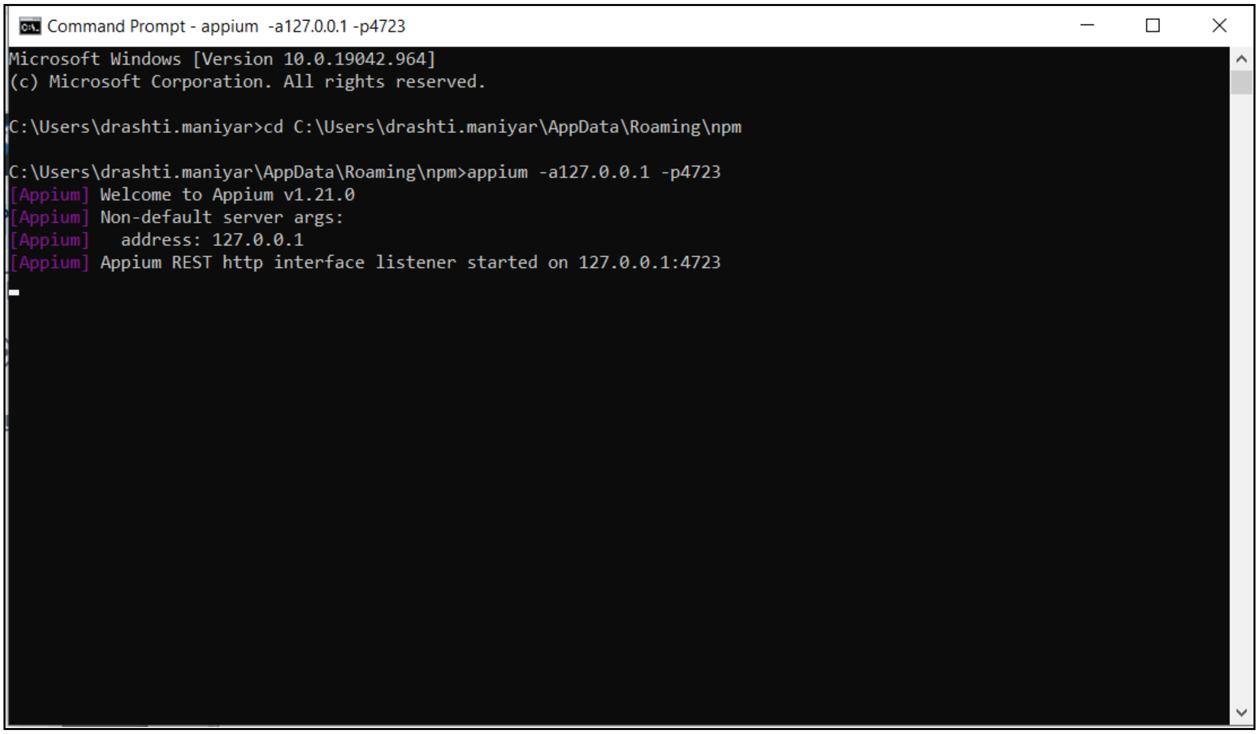


Command Prompt

```
Microsoft Windows [Version 10.0.19042.964]
(c) Microsoft Corporation. All rights reserved.

C:\Users\drashti.maniyar>D:

D:\>cd SDK

D:\SDK>cd platform-tools

D:\SDK\platform-tools>adb connect astrofarm.42gears.com:10105
failed to authenticate to astrofarm.42gears.com:10105

D:\SDK\platform-tools>adb devices
List of devices attached
astrofarm.42gears.com:10105     device


D:\SDK\platform-tools>
```

6. Start the Appium server using the below command (if Appium is installed through node and npm).

```
appium -a127.0.0.1 -p4723
```



```
Microsoft Windows [Version 10.0.19042.964]
(c) Microsoft Corporation. All rights reserved.

C:\Users\drashti.maniyar>cd C:\Users\drashti.maniyar\AppData\Roaming\npm

C:\Users\drashti.maniyar\AppData\Roaming\npm>appium -a127.0.0.1 -p4723
[Appium] Welcome to Appium v1.21.0
[Appium] Non-default server args:
[Appium]   address: 127.0.0.1
[Appium] Appium REST http interface listener started on 127.0.0.1:4723
```

(Or)

You can use the Appium desktop server to start the Appium server.

7. Trigger the scripts that need to be executed.

8. Below capabilities are required for appium configuration:

```
DesiredCapabilities caps = new DesiredCapabilities();

caps.setCapability("deviceName", "DeviceName from adb device list");

caps.setCapability("platformName", "Android");

caps.setCapability("appPackage", appPackage);

caps.setCapability("appActivity", activity);
caps.setCapability("adbExecTimeout",100000);

caps.setCapability("uiautomator2ServerInstallTimeout",20000);

caps.setCapability(MobileCapabilityType.NEW_COMMAND_TIMEOUT,20000);

caps.setCapability("testdroid_testTimeout", 20000);

caps.setCapability("noReset", “true”);

caps.setCapability("fullReset", “false”);

driverAppium = new AndroidDriver<WebElement>(new

URL(“http://127.0.0.1:4723/wd/hub”),caps);
```

9. Use the following command to quit an Appium session.

   ```
   driverAppium.quit();
   ```

   Quit method should be added in afterTest() or afterClas() methods.

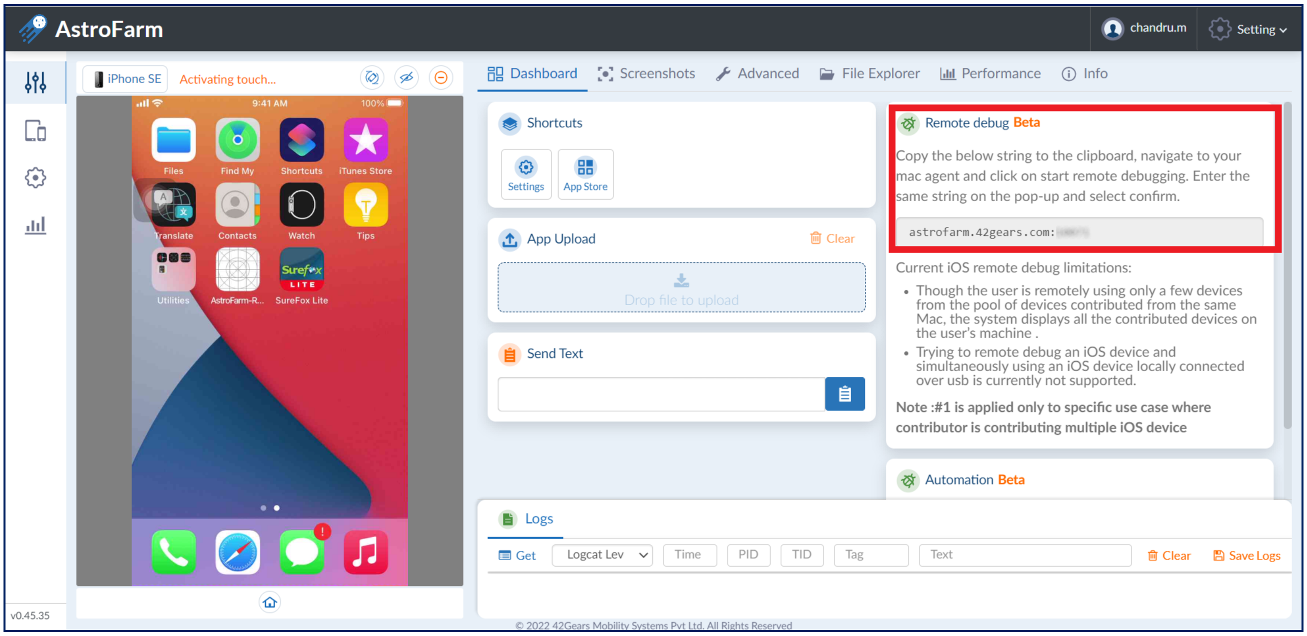
Use Appium to Enable Automation in AstroFarm for iOS Devices

Prerequisite

- Appium should be installed and configured in the mac system

To enable automation in AstroFarm using Appium, follow these steps:

1. Log into the **AstroFarm console**.
2. Search for the **iOS** device that you want to use for automation and click **Use**.
3. Navigate to the **Dashboard** and copy the command from **Remote Debug**.



Remote debug Beta

Copy the below string to the clipboard, navigate to your mac agent and click on start remote debugging. Enter the same string on the pop-up and select confirm.

```
astrofarm.42gears.com:[redacted]
```

Current iOS remote debug limitations:

- Though the user is remotely using only a few devices from the pool of devices contributed from the same Mac, the system displays all the contributed devices on the user's machine .
- Trying to remote debug an iOS device and simultaneously using an iOS device locally connected over usb is currently not supported.

Note :#1 is applied only to specific use case where contributor is contributing multiple iOS device

4. In the **Astrofarm** contributor agent, click **Start Remote Debugging** and configure the same with the **Remote Debug** command.
5. Once done, click **Apply & Start**.

Below is sample code for execution automation code on Astrofarm iOS device with Appium:

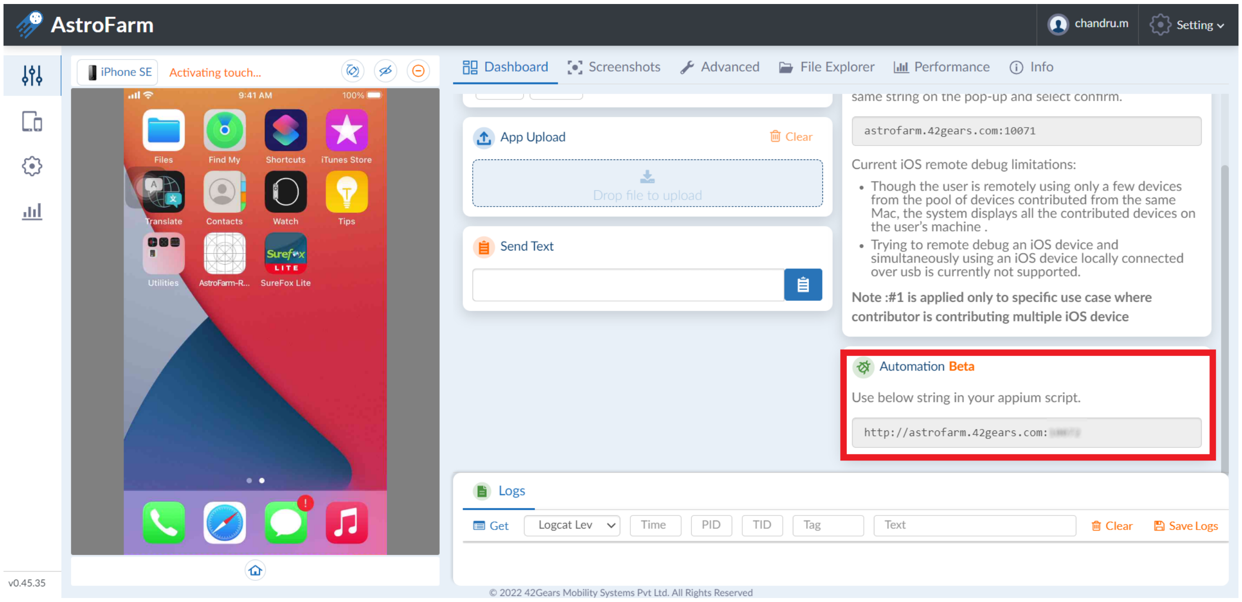
```
import java.net.URL;

import org.openqa.selenium.By;

import org.openqa.selenium.remote.CapabilityType;

import org.openqa.selenium.remote.DesiredCapabilities;

import com.google.common.collect.ImmutableMap;

import io.appium.java_client.ios.IOSDriver;
import io.appium.java_client.remote.MobileCapabilityType;

public class AstroDeviceExecution {

public static String deviceUDID="xxxxxx";

public static String xcodeOrgId="xxxxxx";

public static String platformVersion="xxxxxx";

public static String deviceName="xxxxxxx";

public static void main(String[] args) throws MalformedURLException, InterruptedException {

DesiredCapabilities capabilities = new DesiredCapabilities();

        capabilities.setCapability("platformName", "ios");

        capabilities.setCapability("platformVersion", platformVersion);

        capabilities.setCapability("deviceName", deviceName);

        capabilities.setCapability("udid", deviceUDID);

        capabilities.setCapability("automationName", "xcuitest");

        capabilities.setCapability("xcodeOrgId", xcodeOrgId);

        capabilities.setCapability("xcodeSigningId", "Apple Development");

        capabilities.setCapability("usePrebuiltWDA", true);
capabilities.setCapability("updatedWDABundleId","com.gears42.WebDriverAgent");

        capabilities.setCapability("webDriverAgentUrl", "Automation command");
```

Note:** Follow step.6 to obtain the **Automation Command.

```
        capabilities.setCapability("noReset", true);
        capabilities.setCapability("skipServerInstallation", true);

        capabilities.setCapability("app", "ApplicationPath");

        IOSDriver driver = new IOSDriver(new
URL("http://0.0.0.0:4723/wd/hub"), capabilities);
driver.findElement(By.xpath("//XCUIElementTypeButton[@name='Test']")).c
lick();
driver.findElement(By.xpath("//XCUIElementTypeTextField[@value='Test']"
)).sendKeys("Test");

  ImmutableMap pressHome = ImmutableMap.of("name", "home");

  driver.executeScript("mobile: pressButton", pressHome);

 driver.activateApp("com.apple.Preferences");

 driver.quit();

 }
```

Note:** User-defined variables are highlighted in **bold.

6. Navigate to the **Dashboard > Automation** to get the **Automation Command.**



same string on the pop-up and select confirm.

```
astrofarm.42gears.com:10071
```

Current iOS remote debug limitations:

- Though the user is remotely using only a few devices from the pool of devices contributed from the same Mac, the system displays all the contributed devices on the user's machine .
- Trying to remote debug an iOS device and simultaneously using an iOS device locally connected over usb is currently not supported.

Note :#1 is applied only to specific use case where contributor is contributing multiple iOS device

Automation Beta

Use below string in your appium script.

```
http://astrofarm.42gears.com:[redacted]
```

© 2022 42Gears Mobility Systems Pvt Ltd. All Rights Reserved

Generate Reports

The **Reports** section in AstroFarm helps to generate two different types of reports.

General - In this section, the report is generated based on the device usage of a specific user. The report will have details such as device details, duration of the device used by the user, Contributor details and, more.

Device Usage Logs - In this section, the report is generated based on the total device usage of all users enrolled in the AstroFarm. This option is available for the users having admin privileges. The report will have the details such as device details, duration of the device used by the user, contributor details, and more.

Actions in Reports

The following actions can be performed in the **Reports** section:

Settings	Description
Search	Narrow down the search based on the users/contributors in the list.
Show Filters	Use this option to narrow down the search based on Contributors, Devices, Users, and Duration of the device used to generate reports.
Customize	Show or hide required columns in the list.
Export	Export the filtered records of device usage details in CSV format.

User Management

In this section, you will learn more about how to add and manage users and contributors.

Two types of users can access the AstroFarm console.

- **Users** - Users having admin privileges can add and manage users. Users without admin privileges can only access the devices on the AstroFarm web console.
- **Contributors** - Contributors are users who can enroll devices to the AstroFarm console through the host machine but may not necessarily need to access the devices. Users having admin privileges can add and manage Contributors.

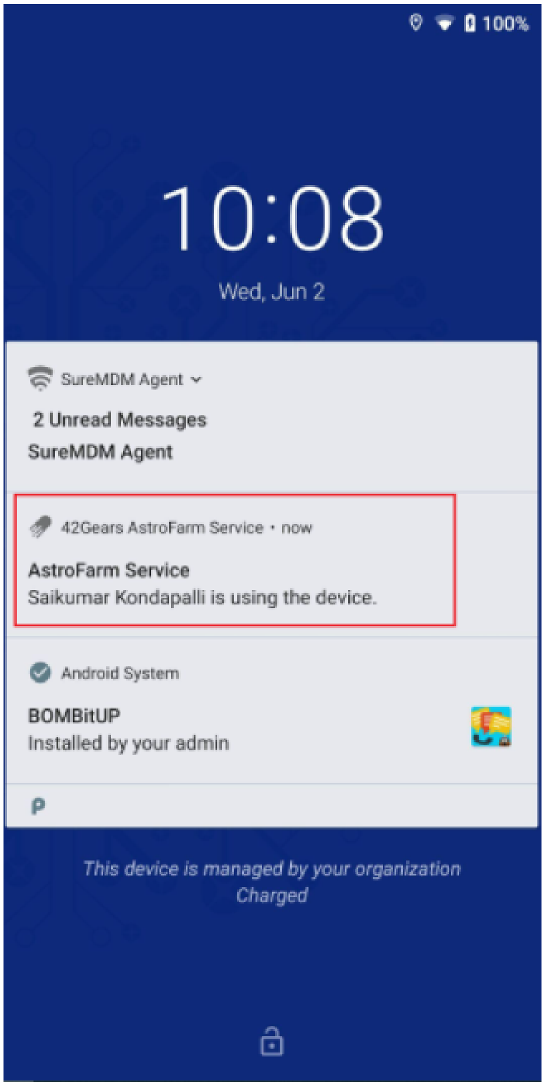
Add a User to AstroFarm

To add a user to your AstroFarm account,

1. Log into the **AstroFarm** portal and tap the **Settings** icon.
2. Select the **Users** tab.
3. Click + icon to add users.
4. Enter **Name**, **Email ID**, and select **Add as Contributor** to make the user a contributor.
5. Click **Save**.

The user will get an email invitation to access the portal with the option to set a password.

Once a user is added to the console, contributors can see the user details (like username) by dragging down the device notification panel and without logging into the AstroFarm console.



10:08

Wed, Jun 2

SureMDM Agent

2 Unread Messages

SureMDM Agent

42Gears AstroFarm Service • now

AstroFarm Service

Saikumar Kondapalli is using the device.

Android System

BOMBitUP

Installed by your admin

This device is managed by your organization

Charged

Remove a User from AstroFarm

To remove a user from your AstroFarm account,

1. Log into the **AstroFarm** portal, and tap the **Settings** icon.

2. Select the **Users** tab.

3. Select the user to be deleted and click the **Remove** icon.

Grant Admin Privileges to an AstroFarm User

To grant admin privileges to a user on your AstroFarm account,

1. Log into the **AstroFarm** portal, and tap on the **Settings** icon.
2. Select the **Users** tab.
3. Select the user and click **Give Admin Privileges** icon to grant the user admin privileges.

   Click **OK** on the warning screen to confirm.

Remove Admin Privileges from an AstroFarm User

To remove admin privileges from a selected user of your AstroFarm account,

1. Log into the **AstroFarm** portal, and tap on the **Settings** icon.
2. Select the **Users** tab.
3. Click **Remove Admin Privileges** icon to remove admin privileges for a user.

Add a Contributor to AstroFarm

To add a contributor to your AstroFarm account,

1. Log into the **AstroFarm** portal, and tap on the **Settings** icon.
2. Select the **Contributors** tab.
3. Click + icon to add contributors.
4. Enter the **Name** and **Email** (optional) of the contributor.
5. Click + **Add Contributor**.

Note: *You can create multiple contributors for a single user using the same email address.*

Remove a Contributor from AstroFarm

To remove contributors from your AstroFarm account,

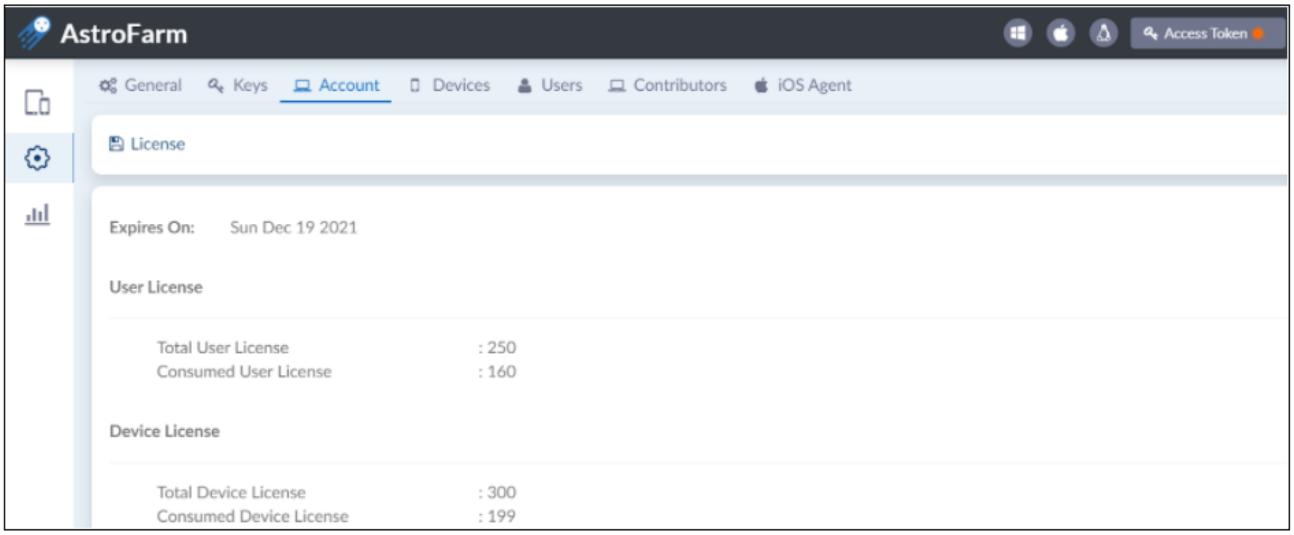
1. Log into the **AstroFarm** portal, and tap on the **Settings** icon.

2. Select the **Contributors** tab.

3. Click **Remove** for the contributor you want to remove.

License Management

In this section, administrators can view the license details of the AstroFarm account, such as account expiration date, the total number of licenses, and the number of consumed licenses for both device and user. To get these details, log into the AstroFarm console > Settings > Accounts, and you will find the details as given in the screenshot below.



Expires On: Sun Dec 19 2021

User License

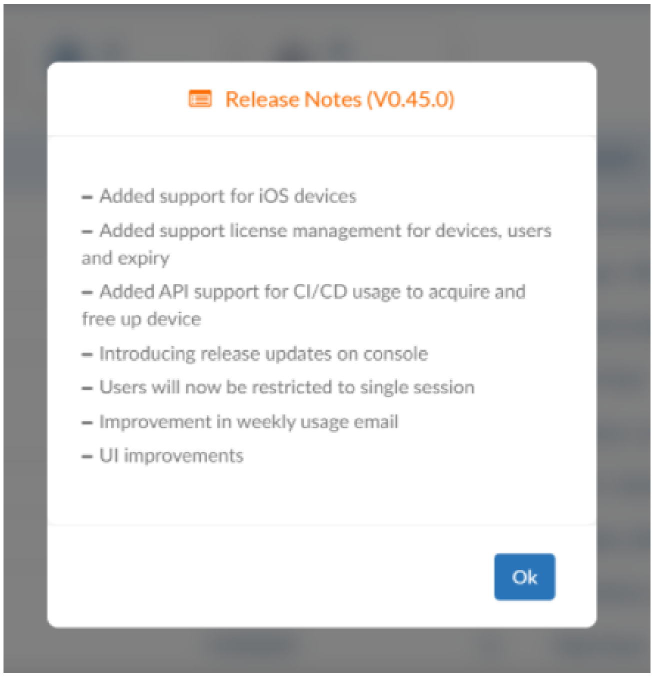
Total User License : 250
Consumed User License : 160

Device License

Total Device License : 300
Consumed Device License : 199

Release Notes Updates

You will be able to see the latest release notes of AstroFarm on the console once you login for the first time after the new release.



Release Notes (V0.45.0)

- Added support for iOS devices
- Added support license management for devices, users and expiry
- Added API support for CI/CD usage to acquire and free up device
- Introducing release updates on console
- Users will now be restricted to single session
- Improvement in weekly usage email
- UI improvements