



## GETTING STARTED

Start by importing PhoneOS into your Unity Project

Once that's done, you can open a scene and prepare to add PhoneOS.

Navigate to **PhoneOS > Prefabs** Folder

Drag the **PHONE\_MANAGER** prefab into the Hierarchy

Right Click **PHONE\_MANAGER** and select **Prefab > Unpack Completely**

Great. You have successfully added the PhoneOS to your scene.

Now it's time for you to learn about how to configure the PhoneOS system for your custom needs.

You should continue to read the documentation to learn how to configure several of the default apps on **PhoneOS**, so that you can take full advantage of the powerful custom code that is integrated into this smartphone system.

**PhoneOS allows you to integrate a smartphone into your game.**

**You will be able to use several apps right after dragging it into your scene.**

**Some apps need to be configured or else they will NOT work properly.**

# Configuring the Phone System Information

After completing the initial setup process.

You will want to configure the operating system on **PhoneOS** to reflect your custom needs.

In the hierarchy, click on **PHONE\_MANAGER**

In the inspector, you will see a script component called **PhoneConfigure**

In this component, you will need to do the following....

## **Set the Location**

(For Example: NEW YORK)

## **Set the Battery Level**

(For Example: 100)

## **Set the Carrier Name**

(For Example: SIM)

## **Set the Phone Time**

(For Example: Hour 12, Minute 30, isAM = true)

*(This would set the default time to 12:30 AM)*

After configuring the settings, you will be able to start the scene and you'll notice that your phone now reflects the new settings that you have just configured into the phone.

# How to Receive a Voicemail Message

One of the main features that you are likely going to want to take advantage of is the ability to receive an audio message. (Voicemail)

This is a great way to have other characters communicate with the player.

To get started, you will need to have a pre-recorded audio clip that can serve as your voicemail message.

You will need to assign it to the **Voicemail Database**.

To do that, navigate to the Hierarchy....

Click on **PHONE\_MANAGER > PHONE\_OS** and Find the **Voicemail Database** script component.

Drag your sound file into any of the slots that you wish to use.  
*(Supports up to 30 by default)*

Once you've assigned the sound file to the database, you will need to go into one of your custom scripts and do the following.

*Add **using PhoneSim;** to the top of your custom script.*

*Add **public VoicemailDatabase voicemailDatabase;** above the **Start()** function*

*Put*  
**voicemailDatabase = GameObject.FindObjectOfType<VoicemailDatabase>();**  
*inside the **Start()** function*

*You have now connected your custom script with the **VoicemailDatabase**.*

*Please continue reading on the next page to proceed.*

# How to Receive a Voicemail Message (PART 2)

Add **public AppCalls appCalls;** above the **Start()** function

Add **appCalls = GameObject.FindObjectOfType<AppCalls>();**  
inside of the **Start()** function

Now you've connected your custom script with the Voicemail app

Now you can use the following line of code to setup a new voicemail.

**appCalls.SetNewVoicemail("Bob", voicemailDatabase.voicemail1);**

*(Sets a new voicemail, with the contact name “Bob”, and uses the audio file from **Slot 1** of the **Voicemail Database**)*

**IMPORTANT:** YOU MUST USE THE SLOT WHERE YOU DRAGGED THE ORIGINAL  
SOUND CLIP AT THE BEGINNING OF THIS TUTORIAL

You can ALSO review the **ExampleVoicemail.cs** script which is included in  
**PhoneOS > Scripts > Extras** for an example of how this works.

The moment you use the **SetNewVoicemail** function, the Phone will receive a push notification, along with a notification sound, and you'll be able to go into the **Voicemail** app and listen to the voicemail instantly.

You can click on “**Clear**” in the top right corner of the app  
to clear the current voicemail.

Any questions relating to this Voicemail System should be directed to  
**ALERT@FLAGMANJEREMY.NET**

# How to Add Custom Songs to the Music Player

Adding custom songs into the music app  
is extremely quick and easy.

By default, one demo song is already added.

You can navigate to **PHONE\_MANAGER > PHONE\_SONGS**  
in the hierarchy and you'll see "**Demo Song**" listed there.

Simply drag a song file into the **AudioSource** and rename the  
object from "**Demo Song**" to your desired song name.

You have now configured your first song in the music player.

## ***Want to add more songs?***

Simply **copy and paste** the existing song multiple times, rename  
the object to your desired song name, and drag in your song file  
to each **Audio Source**.

You can do this until there are a **maximum of six songs**  
in the music player.

You can also change song names and clips in real time in your  
custom script if you wish to do so.

Any questions relating to adding custom songs  
should be directed to

**ALERT@FLAGMANJEREMY.NET**

# How to Set a Custom News Headline

You may want to customize the news headline in the **News** app.

Here's how:

*Add **using PhoneSim;** to the top of your custom script.*

Add **public AppNews appNews;** above the **Start()** function

Add **appNews= GameObject.FindObjectOfType<AppNews>();**  
inside of the **Start()** function

Add

**appNews.SetNewsHeadline("Firefighters rescue a  
neighborhood cat out of a tree as dangerous weather  
approaches");**

into any function that is called in your custom script

*(This sets an example headline of firefighters rescuing a cat, but you can type  
anything you want the headline to be!)*

And that's it, then your **News** app will reflect the new headline  
once you use **SetNewsHeadline();**

Any questions relating to setting custom **News** headlines  
should be directed to

ALERT@FLAGMANJEREMY.NET

# How to Setup the Weather App

You may want to customize the weather in the **Weather** app.

Here's how:

*Add **using PhoneSim;** to the top of your custom script.*

Add **public AppWeather appWeather;** above the **Start()** function

Add **appWeather = GameObject.FindObjectOfType<AppWeather>();**  
inside of the **Start()** function

Call the following lines in your script to set the weather

**appWeather.SetClear();**

*(Sets the weather to clear in the app)*

**appWeather.SetCloudy();**

*(Sets the weather to cloudy in the app)*

**appWeather.SetRaining();**

*(Sets the weather to raining in the app)*

**appWeather.SetTemp(84);**

*(Sets the temperature to 84 degrees Fahrenheit in the app)*

And that's it, then your **Weather** app will reflect the new weather conditions that you've set in your custom script.

Any questions relating to setting custom **Weather**  
should be directed to

ALERT@FLAGMANJEREMY.NET

## Other Default PhoneOS Apps

There are several other default applications installed on the **PhoneOS** smartphone that do not require any pre-defined setup or configuration.

This includes the **Timer** app, where you can use a stopwatch to keep track of time.

There is also a **Notes** app, where you can save notes and keep a list of things that you need to remember.

There's also a built-in **Calculator** app for simple math equations.

While there's no required setup for these apps, if you have questions about them.

Any questions relating to these built-in apps should be directed to

[ALERT@FLAGMANJEREMY.NET](mailto:ALERT@FLAGMANJEREMY.NET)



# THANK YOU FOR PURCHASING

NEED ASSISTANCE OR SUPPORT?

**CONTACT:**

alert@flagmanjeremy.net



*FlagmanJeremy.net*