

# **Software Design**

# **Description**

## Gaming System

Version 0.13

## IMPORTANT CONFIDENTIALITY NOTICE

This document contains confidential information of SOFTEQ Development Corporation ("SOFTEQ"). It is provided for the sole purpose of permitting the recipient to evaluate the requested information submitted herewith. **In consideration of receipt of this document, the recipient agrees to maintain such information in confidence and to not reproduce or otherwise disclose this information to any person outside the group directly responsible for evaluation of its contents,** except that there is no obligation to maintain the confidentiality of any such information which was known to recipient prior to receipt of this document as evidenced by written business records; which becomes publicly known through no fault of recipient; or which is received by recipient without obligation of confidentiality from a third party owing no duty of non-disclosure or confidentiality to SOFTEQ.

SOFTEQ has endeavored to include in the information in this document the materials that are believed to be reliable and relevant for the purpose of recipient's evaluation. Neither SOFTEQ nor its representatives make any warranties as to the accuracy or completeness of the information. Accordingly, this document is provided for information purposes only in the hope that SOFTEQ may be considered to receive your business. Neither SOFTEQ nor its representatives shall have any liability to recipient or any of its representatives as a result of the use of the information provided. **Only a mutually agreed-upon written definitive agreement, signed by the authorized representatives of the parties, shall be binding on SOFTEQ or its affiliates.**

Unless otherwise agreed in writing pricing estimates are valid for 90 days from date of submission of this proposal.

If there are any concerns, questions, or issues regarding this Confidentiality Notice, please contact your SOFTEQ sales representative.

# REVISION HISTORY

Date	Version	Description	Author
10/07/2015	Draft	Initial creation	Anton Tonkovich
11/02/2015	0.2	System limitations for third party developers have been added	Anton Tonkovich
11/10/2015	0.3	Update process description has been added	Anton Tonkovich
11/16/2015	0.4	Factory Settings section has been added	Anton Tonkovich
11/17/2015	0.5	Factory Settings section has been edited	Anton Tonkovich
11/27/2015	0.6	Management Console section has been added	Alexei Barchenko
11/17/2015	0.7	Factory Settings section has been modified	Timofey Tenushko
12/11/2015	0.8	Management Console and Launcher sections have been updated	Anton Tonkovich
12/29/2015	0.9	Factory Configuration Process section has been modified	Anton Tonkovich
12/29/2015	0.10	Edited for spelling and grammar	Paul Fruia
2/2/2016	0.11	Added Update Example section and updated the screen shots	Paul Fruia
3/18/2016	0.12	Updated Screensaver section with more detail about when timeout is applied and supported image file types.	Paul Fruia
4/08/2016	0.13	Updated launcher_settings.json and update.json example scripts to reference customer servers. Added more detail and examples for the update process.	Paul Fruia

## Contents

REVISION HISTORY .....	3
INTRODUCTION .....	5
Purpose of the document .....	5
Intended Audience and Reading Suggestions .....	5
OVERALL DESCRIPTION .....	6
Project Description .....	6
Target Audience .....	6
Benefits and Importance for Users .....	6
System Environment .....	6
System Design .....	7
Hardware .....	7
Firmware .....	9
System limitations .....	9
Management Console .....	10
<a href="#">How to open Management Console</a> .....	10
<a href="#">Password Management</a> .....	11
<a href="#">Setup WiFi</a> .....	12
<a href="#">Set Time Zone</a> .....	14
<a href="#">Set Volume</a> .....	14
<a href="#">Screensavers</a> .....	15
<a href="#">System Status</a> .....	17
<a href="#">Installed Applications</a> .....	19
<a href="#">Manual Update</a> .....	20
Software Update .....	21
Factory Settings .....	28
Launcher Settings .....	29
Default Settings .....	33
Factory Configuration Process .....	35
REFERENCES .....	38

# INTRODUCTION

## **Purpose of the document**

The purpose of the Software Requirements Specification is to provide a clear and consistent vision of the Gaming System project.

This document is intended to ensure a common understanding of the software goals and tasks both by Softeq developers and NunoErin business experts.

The document is to be continuously updated to reflect the functionality changes that happen during the implementation of the Gaming System project, so that an accurate vision of what has been done is available at any time.

## **Intended Audience and Reading Suggestions**

This document is intended for use by the Gaming System team members and interested stakeholders to assist them in understanding the software requirements of the product. The expected audience of this document also includes Softeq developers. The document is not intended to be any sort of User Guide or Manual.

# OVERALL DESCRIPTION

## Project Description

Development of software for furniture game system

## Target Audience

TBD

## Benefits and Importance for Users

TBD

## System Environment

### Device

The device runs on a customized Android 4.2.2 OS. The customization includes support for WiFi, touch screen, and HDMI output.

# System Design

## Hardware

### General

TBD

### Component list

- A20-OLinuXino-MICRO-4GB board

<http://www.amazon.com/A20-OLINUXINO-MICRO-4GB-single-Arduino-Debian-Computer/dp/B00EBCP7VU>

- USB 2.0 A to Mini-B Cable

<http://www.amazon.com/AmazonBasics-USB-2-0-Cable-A-Male/dp/B00NH11N5A>

- 110-240 AC to 12V 1A DC power adapter with 5.5x2.1 mm jack

<http://www.amazon.com/Wall-Adapter-Power-Supply-12VDC/dp/B006GEPUYA>

- Micro SD card with 8GB capacity class 10
- USB SD Card Reader Any model that handles MicroSD SDXC format should suit, just for example <http://raspberrypi.au/collections/sd-cards-readers/products/usb-3-sd-card-reader>
- DELL P2314T touch screen monitor (the package includes an HDMI cable, a USB cable for connection of a touch screen and a power supply)

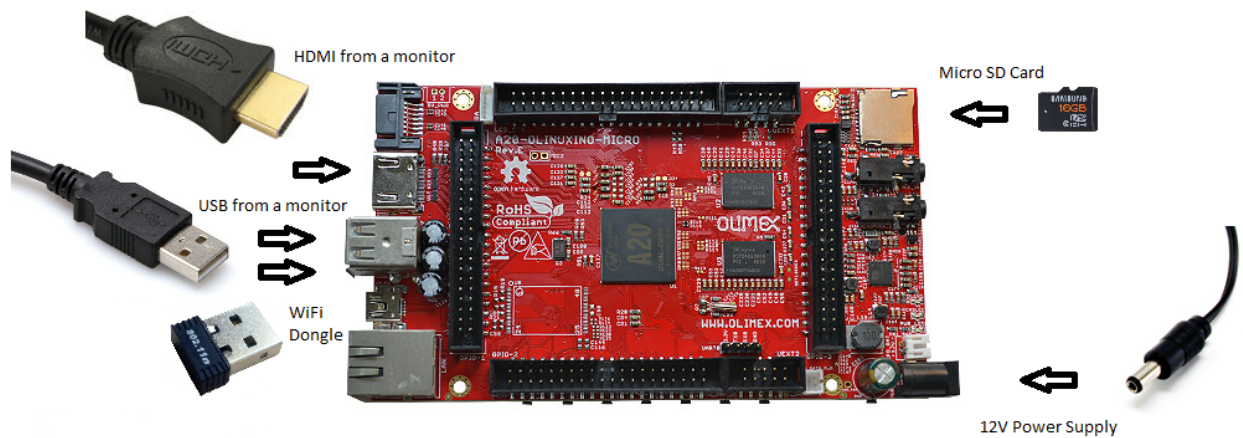
[http://www.amazon.com/DELL-P2314T-Dell-Touch-Monitor/dp/B00OZ4X414/ref=sr\\_1\\_12?s=pc&ie=UTF8&qid=1442409348&sr=1-12&keywords=P2314T](http://www.amazon.com/DELL-P2314T-Dell-Touch-Monitor/dp/B00OZ4X414/ref=sr_1_12?s=pc&ie=UTF8&qid=1442409348&sr=1-12&keywords=P2314T)

- WiFi dongle

<http://www.amazon.com/Edimax-EW-7811Un-150Mbps-Raspberry-Supports/dp/B003MTTJOY>

## Connection diagram

1. Connect HDMI, USB and Power supply cables to a DELL P2314T touch screen monitor according to the user manual [http://downloads.dell.com/Manuals/Common/dell-p2314t\\_User's%20Guide\\_en-us.pdf](http://downloads.dell.com/Manuals/Common/dell-p2314t_User's%20Guide_en-us.pdf)
2. Connect HDMI, USB, and Power Supply cables to an Olimex board according to the picture shown below.
3. Insert a Micro SD card with a flashed image (please see the firmware section for guidance on how to prepare a Micro SD card).
4. Insert a WiFi Dongle in a USB port.
5. Power on the board. (Please note that booting up takes about a minute, there is no picture on the screen during that time.)





# Firmware

In order to flash a micro SD card, please follow the steps below.

1. Install 7-Zip archiver (you can obtain it at <http://7-zip.org/a/7z920.exe>)
2. Download the firmware image archive gamesystem-image-x.x.x.7z from appropriate source.
3. Unpack image archive with 7z. You'll get the one file of type .img.
4. Download and install Win32 Disk Imager utility from <http://sourceforge.net/projects/win32diskimager/>
5. Prepare a Micro SD card with at least 8GB capacity and speed class of 10.
6. Insert the card into a card reader and plug it into a PC.
7. Launch the Win32 Disk Imager utility and select an appropriate image file and the drive letter Windows has assigned to the card.
8. Click the "Write" button and wait for disk image writing to complete.
9. When it's done, safely eject the card and insert it into the micro SD cardholder on an Olimex board.
10. Power on the Olimex board and wait while the OS boots up.
11. The monitor should start to display an image after boot up time (approximately one minute).

## System limitations

The system has hardware and software limitations that should be taken into account during application development.

- **Allwinner A20**
  - CPU: ARM® Cortex™-A7 Dual-Core
  - GPU: ARM Mali400MP2, OpenGL ES 2.0/1.1
- **1GB DDR3**
- **8GB micro SD (5.61GB in storage partition)**
- **Android 4.2.2**
- **Graphic resolution: HDMI, 720p60 (1280x720)**
- **Landscape and portrait modes**
- **Custom applications icon 245x245 (\*.png, for four icons in a row)**
- **No system controls for applications run in full screen mode (kiosk mode)**

# Management Console

The Management Console allows controlling various device settings. The following sections are available in the Management Console:

- **Password Management**
- **Setup WiFi**
- **Set Time Zone**
- **Set Volume**
- **Screensavers**
- **System Status**
- **Serial Number**
- **Installed Applications**
- **Manual Update**

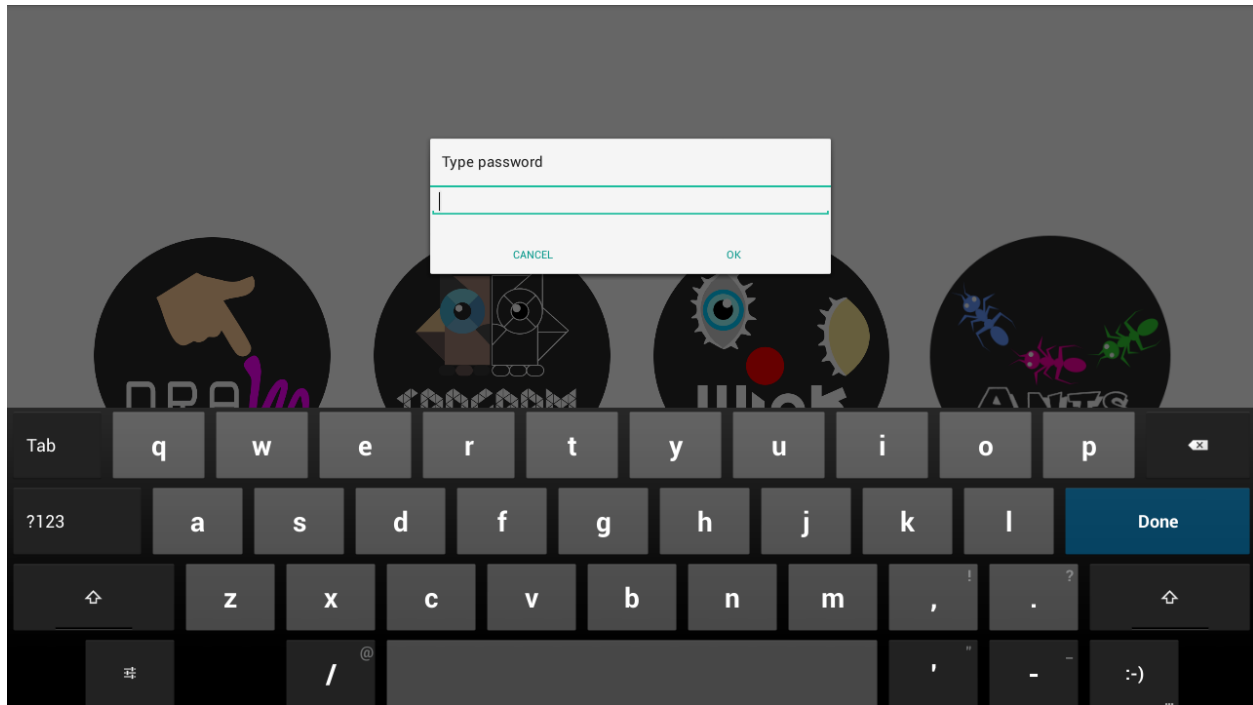
## How to open Management Console

On the main screen, tap 10 times on the NunoErin logo. Then enter the password.

---



NunoErin®



The default password is: nunoerin

The master password to reset the current password is: ZcuvTo?1CujY

## Password Management

The screen allows changing a current password for the Management Console.

To change the password, enter your current password, then enter a new password and confirmation of the new password, and then tap the “Apply” button.

← EXIT

- Password Management
- Setup WiFi
- Setup Timezone
- Setup Volume
- Screensavers
- System Status
- Serial Number
- Installed Applications
- Manual Update

Current password

New password

Confirm new password

Apply

Ucreate Management Console

## Setup WiFi

Allows connecting to WiFi networks.

To enable/disable WiFi, tap the on/off toggle button (button is grey when WiFi is disabled, aqua when WiFi is enabled).

← EXIT

- Password Management
- Setup WiFi
- Setup Timezone
- Setup Volume
- Screensavers
- System Status
- Serial Number
- Installed Applications
- Manual Update

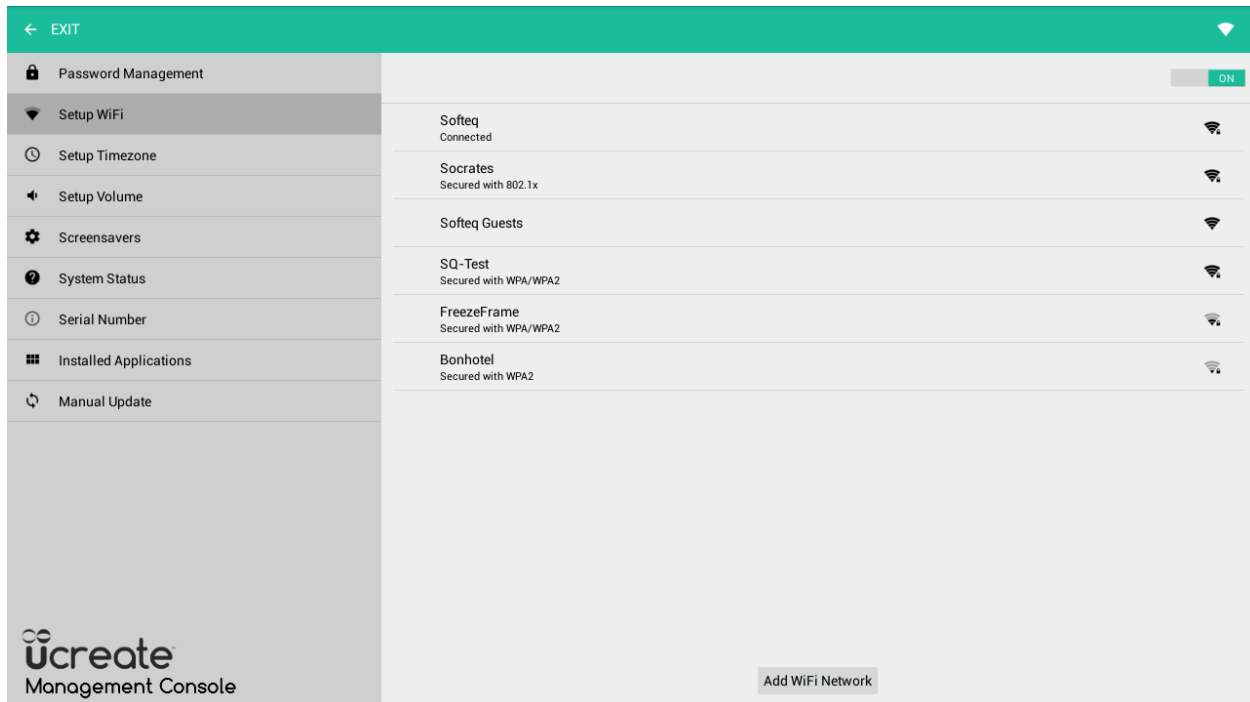
OFF

Add WiFi Network

Ucreate Management Console

To connect to a network, WiFi should be enabled. Select an available WiFi network from the list below and enter any required security credentials. The status of the WiFi connection is displayed below each WiFi network name.

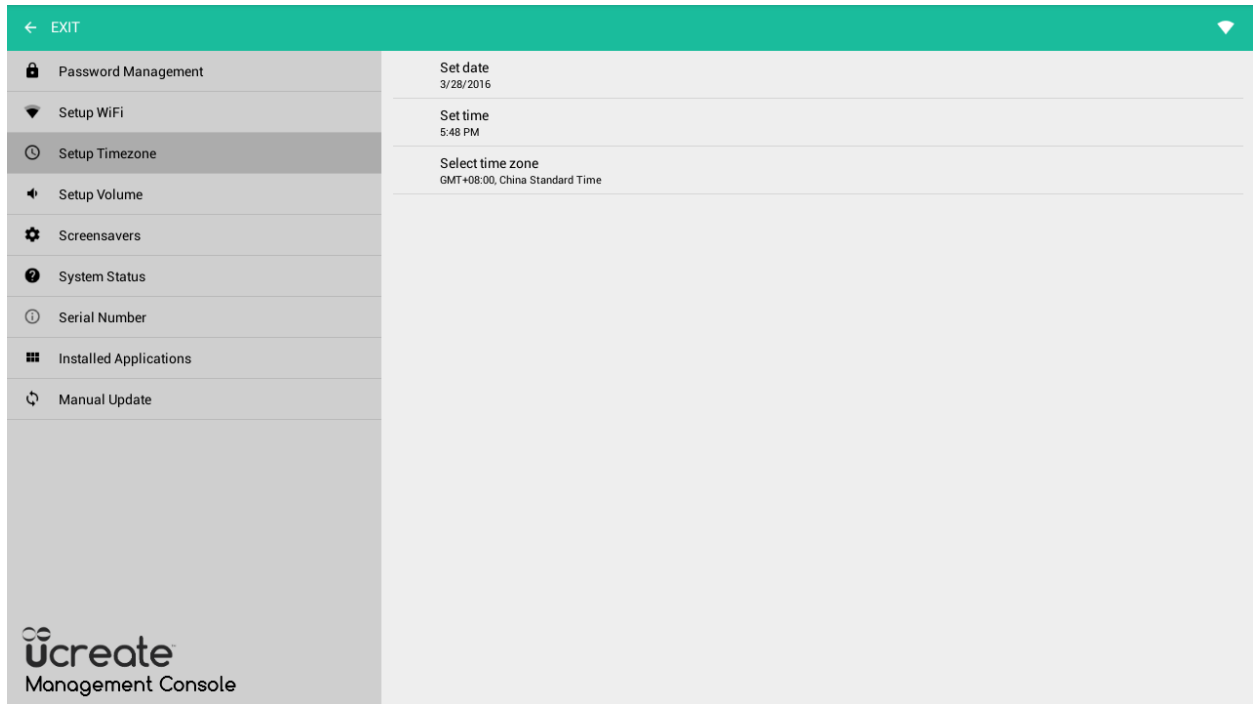
WiFi networks can also be added manually. To do this, press the “ADD NETWORK” button at the bottom of the screen, enter network parameters, and tap “Save”.



*Please Note: Even though a WiFi connection may be successfully established, it is still possible for software updates to fail. Please ensure that the WiFi network you are connecting to also has a properly functioning connection to the Internet.*

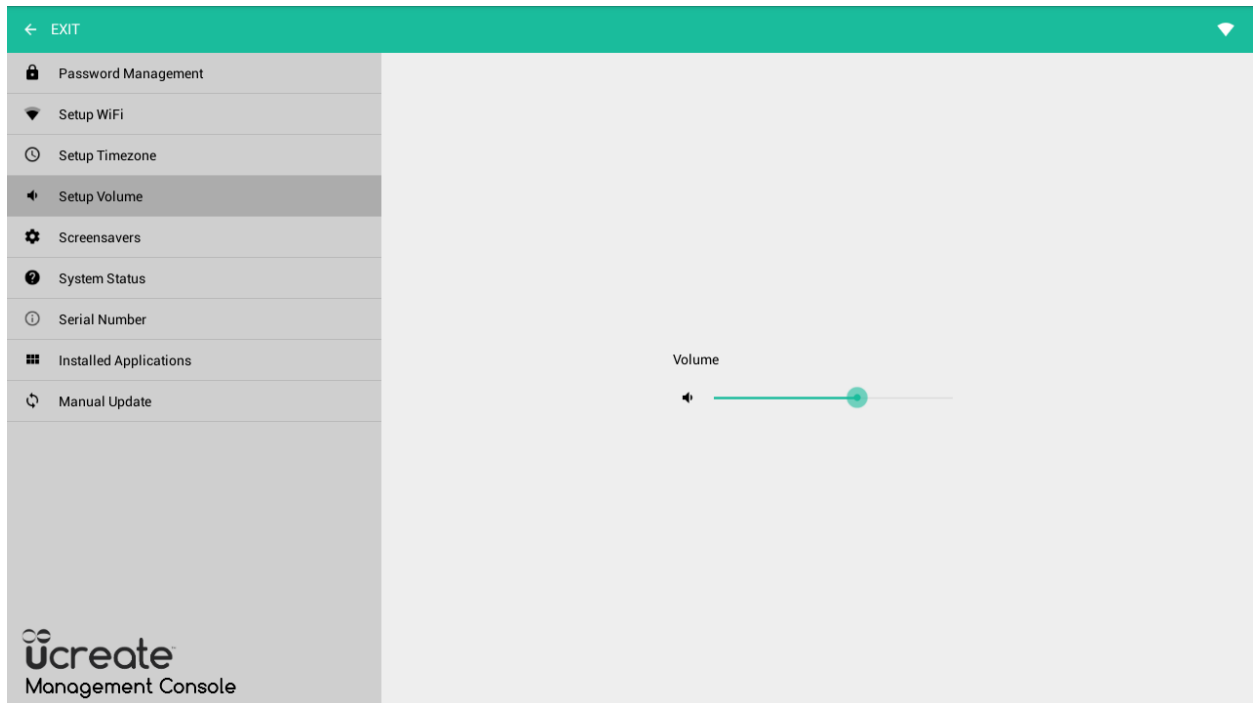
## Set Time Zone

This screen allows for manual setup of date, time, and time zone.



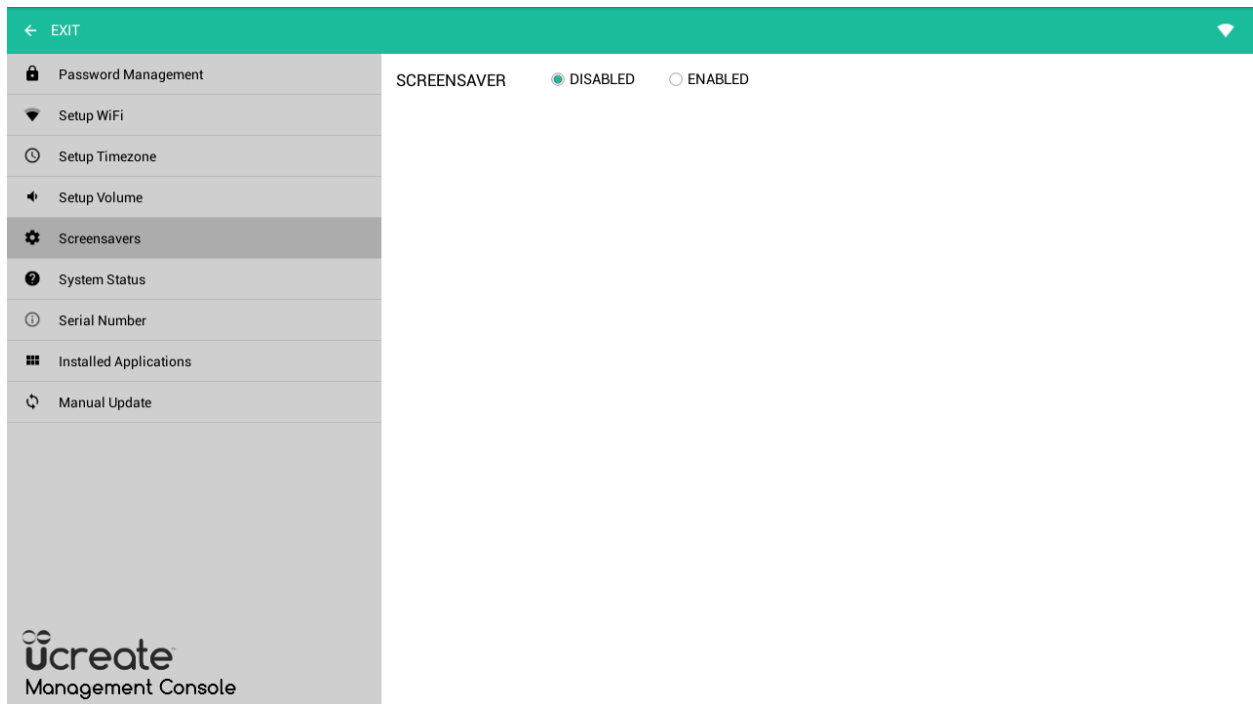
## Set Volume

This screen allows for adjustment of the Master volume of the device.

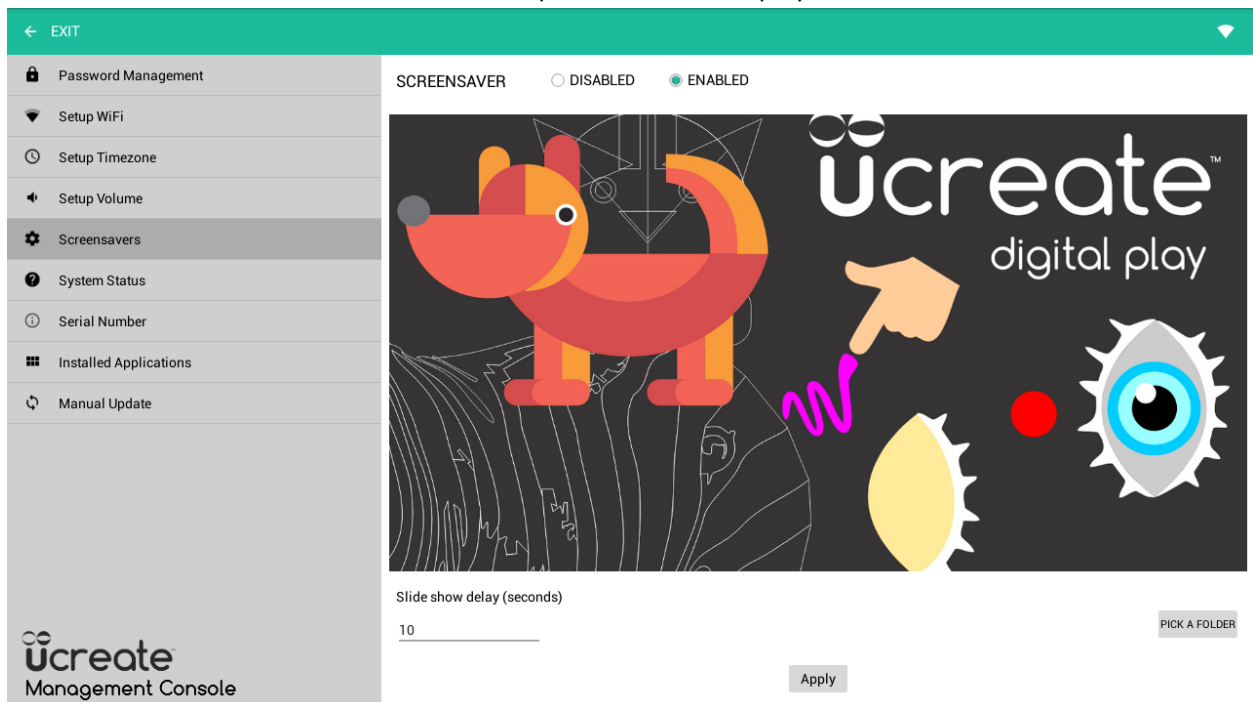


## Screensavers

This section allows configuration of the system screen saver. When the Disabled button is selected, no screensaver slideshow will be shown.



When the Enabled button is selected, the user will have the option to specify the location (folder) of the slideshow to be used for the screensaver. A preview will be displayed.



All changes in configuration should be confirmed by pressing the “Apply” button.

Supported image types for use in slideshows are .png, .jpg, and .gif (at 1280x720 resolution). If only one image file exists in a slideshow folder, the timeout property will not be utilized. However, if more than one image file is present, the timeout will be applied as the amount of time each image stays on the screen before the next image is displayed.

**Please note:** GIF animations to be used as a screensaver should be stored individually in a folder under the "Slideshows" folder on the SD card. This will prevent flashing or screen blanking during playback. If you wish to mix GIF and JPG files in the same Slideshows folder, then you will need to do two things to make sure the GIF animations run properly. First, if you have multiple GIF animations in the same folder, they will all need to run for the same amount of time. Second, you will need to set the slideshow timeout to equal the length of time required to play the animations to completion. There is no way for the Launcher to know how long the GIF images should be displayed to complete one animation cycle. Therefore, they must all be the same length and you must set the slideshow timeout appropriately.

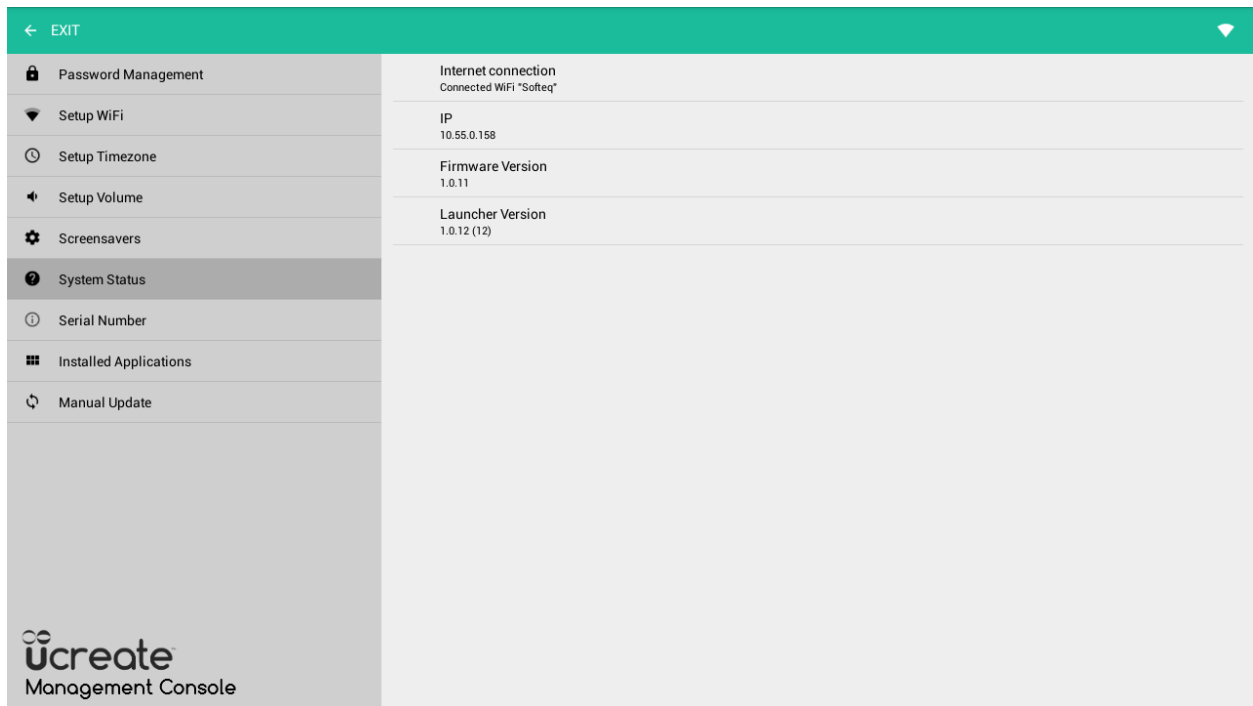
When mixing animated GIF and static images (GIF, JPG, or PNG), the static images will be displayed for the same length of time as required to run the GIF animations for one full animation cycle (or whatever slideshow timeout value you set).



## System Status

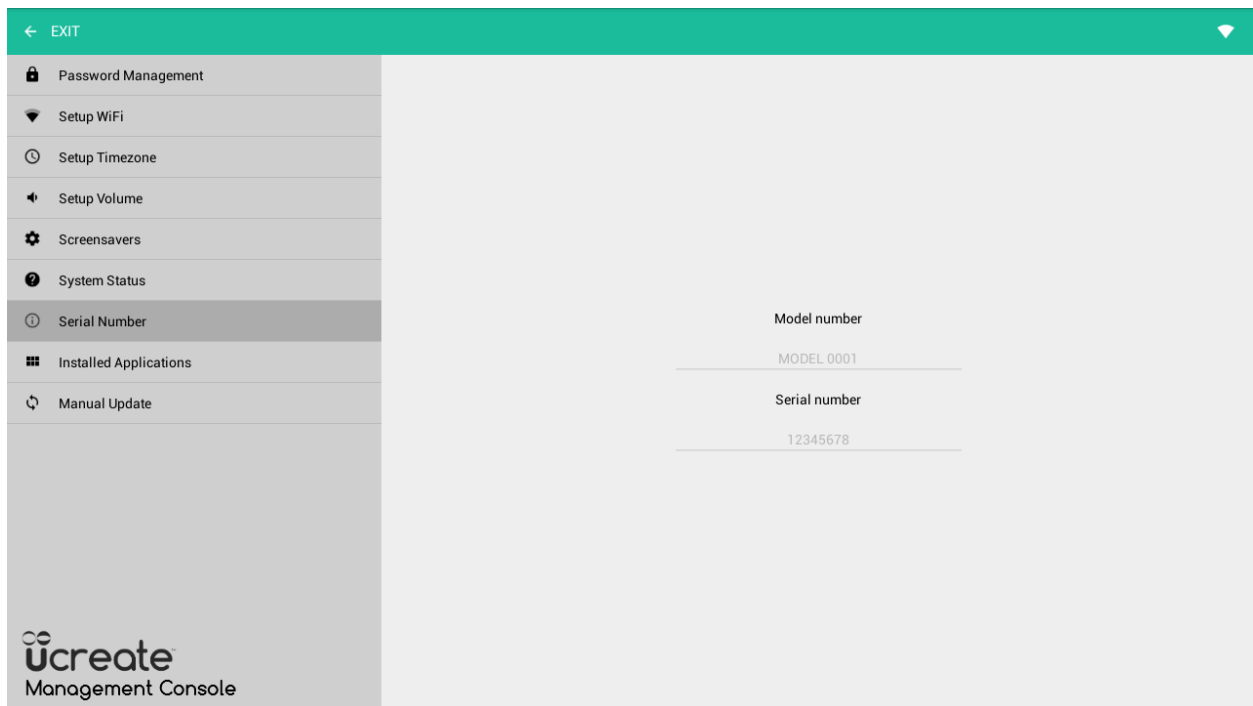
On this screen, the following information is displayed:

- Internet connection – name of the WiFi network that device is connected to.
- IP – IP-address of the device.
- Firmware Version – version of Firmware.
- Launcher Version – version of Launcher application.



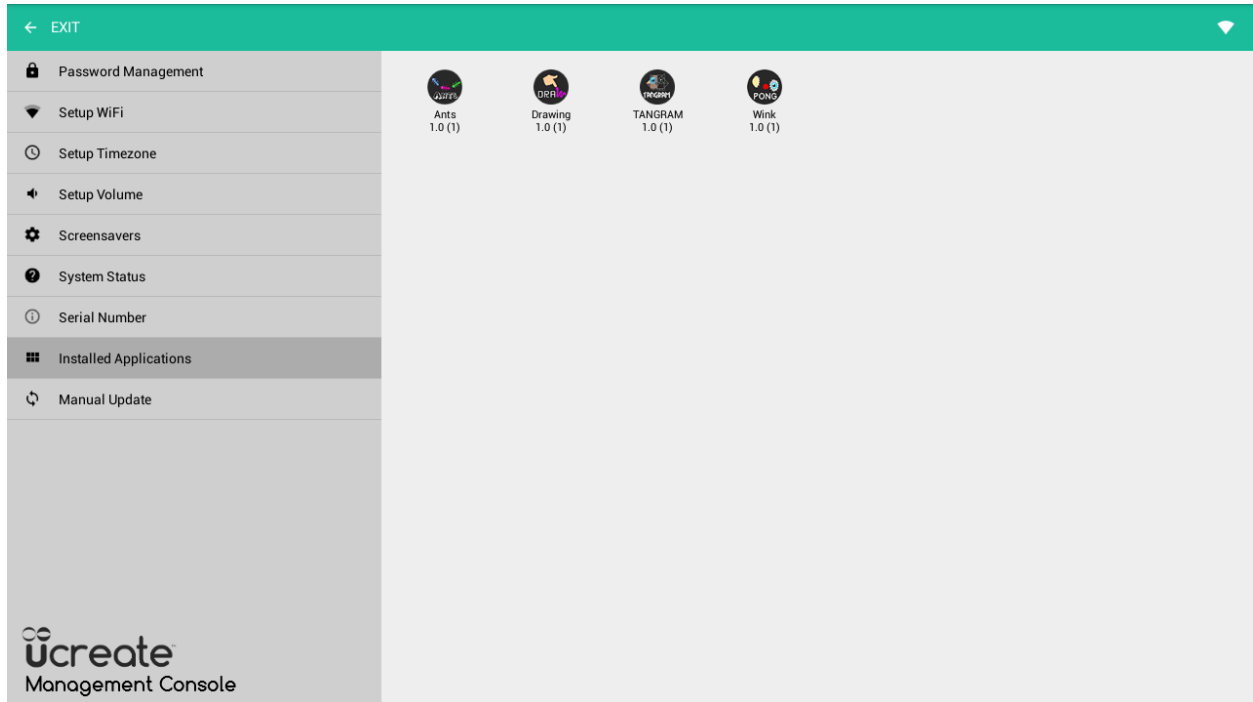
## Serial Number

On this screen the device's Model number and Serial number are displayed. They are read-only.



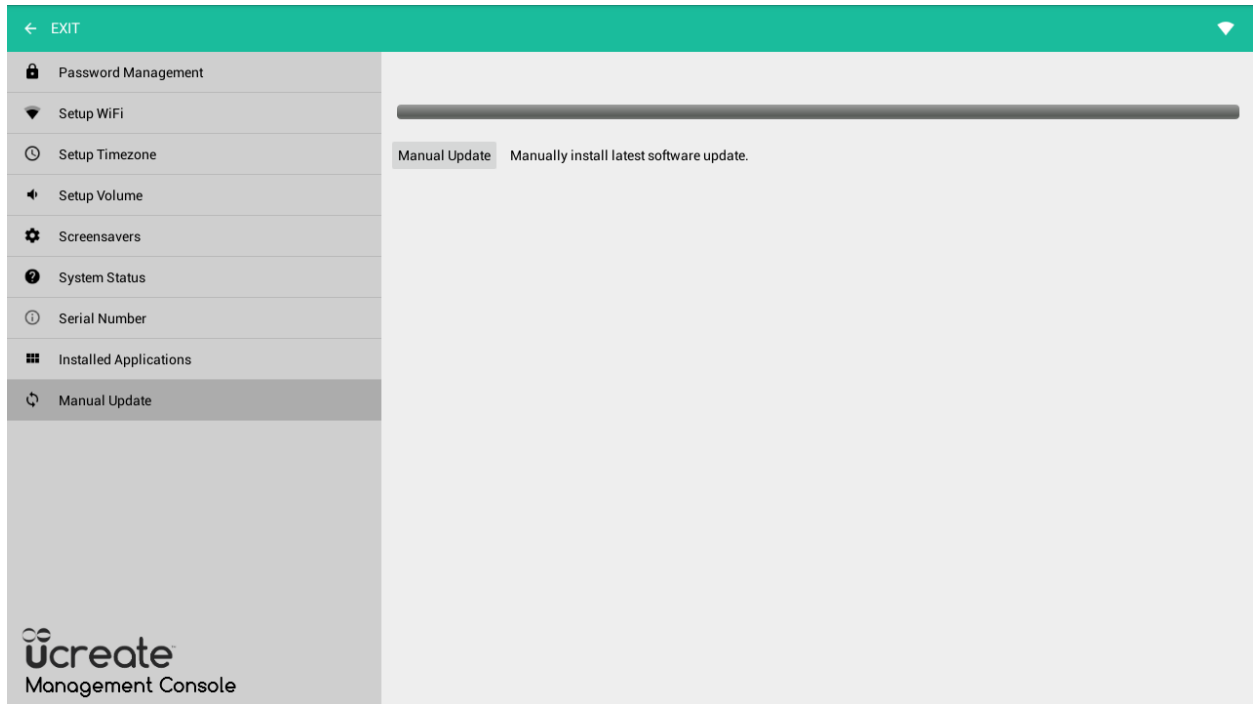
## Installed Applications

Icons for all the applications installed on the device are displayed. Under each icon there is an application name and version. Tap an application's icon to launch the application.

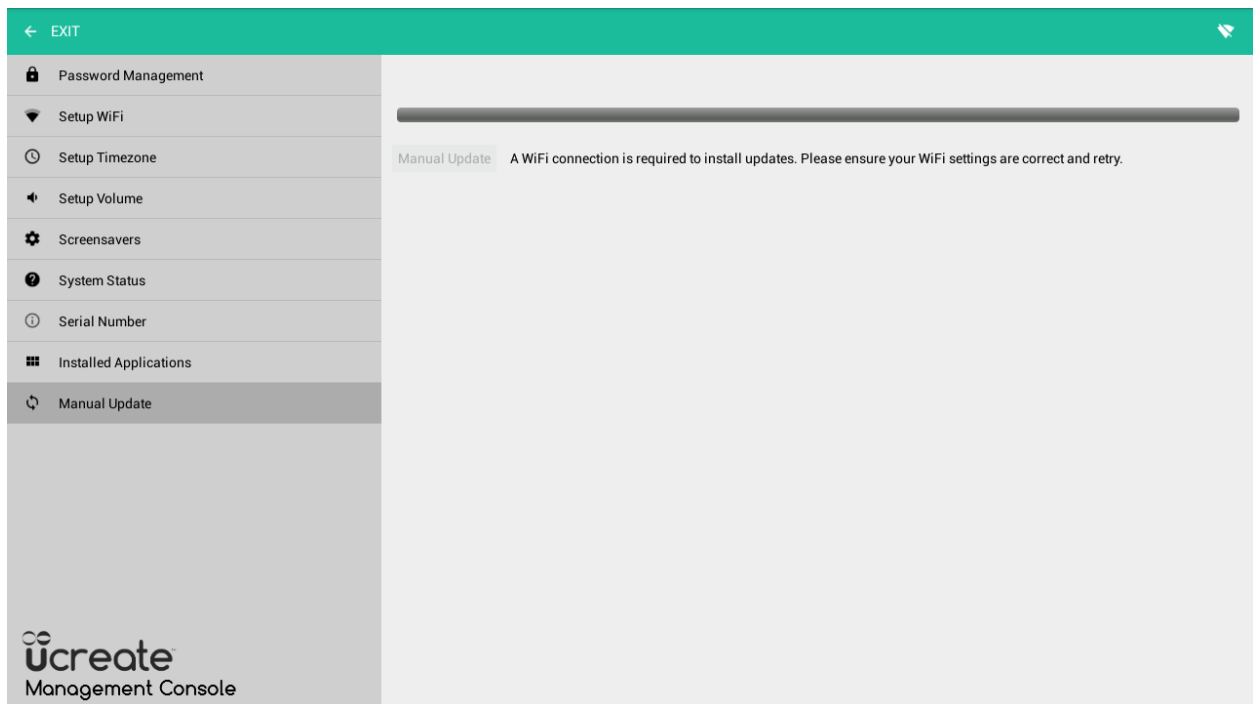


## Manual Update

The screen allows performing updates of applications and files. To update, a device should be connected to WiFi. For more information, see the Software Update section of this document.



To start the update process, tap the Update button. The progress bar advances as updates are downloaded and installed.



# Software Update

The software update functionality allows updating existing applications and installation of new applications over a WiFi connection. The update process requires the availability of a remote HTTP, server accessible via the Internet. An update server directory should contain the following:

1. Packages of applications that should be updated or installed.
2. Files for update or upload.
3. A file in JSON format with descriptions of updating/installing applications and updating/deleting files. An example of a JSON content of a file is shown in the code section below.

```
{
  {
    "applications":[
      {
        "package":"com.draw.olimex.drawing",
        "version":"1.1(1)",
        "url":"http://ucreate.nunoerin.com/updates/applications/Nuno
ErinDraw.apk"
      },
      {
        "package":"com.pong",
        "version":"1.1(1)",
        "url":"http://ucreate.nunoerin.com/updates/applications/Nuno
ErinWink.apk"
      },
      {
        "package":"com.nunoerin.launcher",
        "version":"1.0.11(11)",
        "url":"http://ucreate.nunoerin.com/updates/applications/laun
cher-release-1.0.11(11).apk"
      }
    ],
    "files":[
      {
        "url":"http://ucreate.nunoerin.com/updates/files/ucreate720p
.jpg",
        "file":"/sdcard/Slideshows/TestFolder/ucreate720p.jpg",
        "operation":"update",
        "reboot":false
      },
      {
        "url":"http://ucreate.nunoerin.com/updates/files/ucreate720p
2.jpg",
        "file":"/sdcard/Slideshows/TestFolder/ucreate720p2.jpg",
        "md5":"1b8b8a5380b2ebece9aedce623a9993a",
        "operation":"updateIfHashEquals"
```

```

    },
    {
      "url": "http://ucreate.nunoerin.com/updates/files/ucreatescreensaver.png",
      "file": "/sdcard/Slideshows/Ucreate/ucreatescreensaver.png",
      "md5": "80a32e36bd76be8af0327268ab40fee8",
      "operation": "updateIfHashNotEquals"
    },
    {
      "file": "/sdcard/Slideshows/wallpaper4.gif",
      "operation": "delete"
    },
    {
      "file": "/sdcard/Slideshows/wallpaper5.gif",
      "md5": "e3c50e7b1996478539e5cb76c8ba12ae",
      "operation": "deleteIfHashEquals"
    },
    {
      "file": "/sdcard/Slideshows/wallpaper6.gif",
      "md5": "63c6491a8c4ae44dd7b597cbbbe39675",
      "operation": "deleteIfHashNotEquals"
    }
  ]
}

```

## Applications Section

Applications provided in this section will be uploaded and installed (or updated) on the device.

**“package”** – application package name

**“version”** – application version; please note that version contains the version number “x.x.x” and version code in brackets “(x)”. The version code is a mandatory part of the version field. The software detects whether or not an application update is required by the version code. To get the version code and version name information from the .apk it is possible to use the “aapt” tool from the Android SDK (“aapt dump badging <apk file>”; for more information about the aapt tool please see: [http://elinux.org/Android\\_aapt](http://elinux.org/Android_aapt)).

**“url”** – link to an application (.apk file)

## Files Section

It is also possible to update and delete files on the device by configuring the “files” section of the JSON file. This section contains an array of records that describe file operations.

**“operation”** – file operation. Possible values:

- **“update”** - file will be downloaded from the server and placed on the device whether or not it already exists on the system

- **“updateIfHashEquals”** - file will be downloaded from the server and placed on the device if its MD5 hash is equal to the hash specified in parameters
- **“updateIfHashNotEquals”** - file will be downloaded from the server and placed on the device if its MD5 hash is not equal to the hash specified in the parameters or if the file does not already exist on the system
- **“delete”** - file will be deleted
- **“deleteIfHashEquals”** - file will be deleted if its MD5 hash is equal to the hash specified in the parameters
- **“deleteIfHashNotEquals”** - file will be deleted if its MD5 hash is not equal to the hash specified in the parameters

**“url”** – file download url (mandatory only for the following operations: **“update”**, **“updateIfHashEquals”**, **“updateIfHashNotEquals”**)

**“file”** – local file name (mandatory, file will be created if it doesn’t already exist on the system)

**“md5”** – expected file MD5 hash (mandatory only for the following operations: **“updateIfHashEquals”**, **“updateIfHashNotEquals”**, **“deleteIfHashEquals”**, **“deleteIfHashNotEquals”**)

**“reboot”** – optional flag indicating that a device reboot is required if file operation is completed successfully. Possible values: true, false. Default value: false

## Update Process

The update description file (JSON) URL is specified in a launcher settings file (see Launcher Settings section for more information).

The updating process can be initiated, as follows:

- By pressing the update button in the Management console.
- At the specified time of a day set in Launcher settings.

The Launcher application first downloads the update description file and applies any file operations specified in JSON. After that, it creates a list of applications, which will be installed or updated:

- If an application package specified in the JSON update description file is not installed on device, then it will be installed.
- If an application package is already installed on the device, then it will be updated only if the version code specified in the JSON update description file is greater than that of the installed application.

After that Launcher creates the list of applications to be installed, the applications will be downloaded one by one and installed (or updated). Application update progress is displayed on-screen.

After applications update complete, reboot is performed if any successful file operation has **“reboot”** flag set.

#### NOTES:

1. Updating the entire system over a WiFi connection is not supported.
2. Access restriction functionality in current implementation is not supported.

## Update Process Example – Screen Saver

To properly create a slideshow, you should define a sub-folder under the Slideshows folder on the SD Card, which contains all the files you wish to include in your slide show. Then, you would select the appropriate slideshow folder that you wish to use from the Management Console.

**Please Note:** The “Selected folder” search list for screen savers shows the contents of the “Slideshows” folder, by default. Therefore, it is necessary to make sure that any slide shows or slide show updates are stored in folders beneath the “Slideshows” folder on the SD card.

It is also possible to add a slide show through the update process. For example, if you wished to create a “Test” slideshow containing three images, you could change the update.json file (on the server), as follows:

```
{
  "applications":[],
  "files":[
    {
      "url":"http://ucreate.nunoerin.com/updates/files/test1.jpg",
      "file":"/sdcard/Slideshows/Test/test1.jpg",
      "operation":"update",
      "reboot":false
    }
    {
      "url":"http://ucreate.nunoerin.com/updates/files/test2.jpg",
      "file":"/sdcard/Slideshows/Test/test2.jpg",
      "operation":"update",
      "reboot":false
    }
    {
      "url":"http://ucreate.nunoerin.com/updates/files/test3.jpg",
      "file":"/sdcard/Slideshows/Test/test3.jpg",
      "operation":"update",
      "reboot":false
    }
  ]
}
```

This would upload the three “test” images (test1, test2, and test3) to the “Slideshows/Test” folder on



the SD Card. Then, you would be able to select the “Test” slideshow folder from the Management Console. For the update.json file above to work, you would also need to upload the test1.jpg, test2.jpg, and test3.jpg files to the Media Temple server and place them in the [ucreate.nunoer.in.com/updates/files](https://ucreate.nunoer.in.com/updates/files) folder.

*Please note: This same process can be used to update existing files or upload new files that are currently not existent on the system.*

*Please note: In the previous example, if the “/sdcard/Slideshows/Test” folder does not already exist on the system, it will automatically be created and the images copied into it. Otherwise, the images will be copied into the existing folder.*

## Update Process Example – New Application

When it is desired to add a new application to the Gaming System, it will be necessary to contact Softeq to have them update the Launcher application with the new game .apk package name and icon and the Launcher settings JSON to include the new application. You will be expected to provide the application package file (.apk) and an appropriately sized icon, for testing. (See the **System Limitations** section, above.)

Once Softeq completes integration of the new application, it will then be possible to push the update to remote systems using the WiFi update process. The update JSON file required for this operation would need to include the new application and icon, as well as the updated Launcher application and Launcher settings. All of these files would be stored on the Media Temple server in an appropriate location, of your choosing.

An example update JSON file for this operation is as follows:

```
{
  "applications":[
    {
      "package":"com.new.game",
      "version":"1.0(0)",
      "url":"http://ucreate.nunoerin.com/updates/applications/newgame.apk"
    },
    {
      "package":"com.nunoerin.launcher",
      "version":"1.0.20(20)",
      "url":"http://ucreate.nunoerin.com/updates/applications/launcher-release-1.0.20(20).apk"
    }
  ],
  "files":[
    {
      "url":"http://ucreate.nunoerin.com/updates/files/newgame_icon.png",
      "file":"file:///android_asset/newgame_icon.png",
      "operation":"update",
      "reboot":true
    },
    {
      "url":"http://ucreate.nunoerin.com/updates/files/launcher_settings.json",
      "file":"/sdcard/launcher_settings.json",
      "operation":"update",
      "reboot":true
    }
  ]
}
```

This Update.json file instructs the system to install the new game, “com.new.game”, the updated Launcher, “com.nunoerin.launcher”, the updated Launcher settings file, launcher\_settings.json, and the icon for the new game, “newgame\_icon.png”. This also illustrates the process of uploading a new file that was previously not existent on the system (e.g. newgame\_icon.png) and updating the launcher settings JSON file.

## Update log files

The system supports writing log files to a remote server. A log file contains information about the status of an update: if it was successful, how many applications and files were updated or downloaded, errors encountered during the update process, etc. A log file is formed only if at least one new application or

file update is available. To support upload functionality on a server side, a remote HTTP directory should contain a PHP script and subdirectory **uploads/**.

An example of the script is shown in the code section below:

```
<?php
$target_dir = "uploads/";
$target_file = $target_dir . basename($_FILES["fileToUpload"]["name"]);
$uploadOk = 1;
$imageFileType = pathinfo($target_file,PATHINFO_EXTENSION);
//
if(isset($_POST["submit"])) {
}
// Check if file already exists
if (file_exists($target_file)) {
    echo "Sorry, file already exists.";
    $uploadOk = 0;
}
// Check file size
if ($_FILES["fileToUpload"]["size"] > 50000) {
    echo "Sorry, your file is too large.";
    $uploadOk = 0;
}
// Allow certain file formats
if($imageFileType != "log" ) {
    echo "Sorry, only log files are allowed.";
    $uploadOk = 0;
}
// Check if $uploadOk is set to 0 by an error
if ($uploadOk == 0) {
    echo "Sorry, your file was not uploaded.";
// if everything is ok, try to upload file
} else {
    if (move_uploaded_file($_FILES["fileToUpload"]["tmp_name"], $target_file)) {
        echo "The file ". basename( $_FILES["fileToUpload"]["name"]). " has been
uploaded.";
    } else {
        echo "Sorry, there was an error uploading your file.";
    }
}
?>
```

**NOTE:** Access restriction functionality in current implementation is not supported.

# Factory Settings

The Factory Settings JSON file allows the device model and serial number to be set. A file named ***factory\_settings.json*** should be placed in the root of an internal SD card of the device.

Here is an example of the contents of ***factory\_settings.json*** file:

```
{  
  "modelName": "MODEL 0001",  
  "serialNumber": "12345678"  
}
```

## How to edit Factory Settings

1. Copy a sample factory settings file to a location on a Windows PC.
2. Open the file in a JSON editing program (for example, JSONedit [http://tomeko.net/software/JSONedit/bin/JSONedit\\_0\\_9\\_15\\_1.zip](http://tomeko.net/software/JSONedit/bin/JSONedit_0_9_15_1.zip)).
3. Make any desired modifications to the fields described above and save the file.
4. Insert a micro SD card into a card reader and plug it into a PC. (For instructions on how to prepare an SD card, please see the Firmware section of this document.)
5. Copy the file to the root directory of the card.

# Launcher Settings

The Launcher Settings JSON file allows setting of the Launcher (main application) configuration. A ***launcher\_settings.json*** file should be placed in the root of an internal SD card of the device.

Here is an example of the contents of ***launcher\_settings.json*** file:

```
{
  "applications": [
    {
      "package": "nunoerin.ants",
      "icon": "file:///android_asset/ants.png"
    },
    {
      "package": "nunoerin.drawing",
      "icon": "file:///android_asset/draw.png"
    },
    {
      "package": "nunoerin.tangram",
      "icon": "file:///android_asset/tangram.png"
    }
  ],
  "allowedForegroundApplications": [
    "com.google.android.play.games"
  ],
  "orientation": "landscape",
  "kioskMode": true,
  "applicationList": {
    "mode": "whiteList",
    "applications": [
      "nunoerin.ants",
      "nunoerin.drawing",
      "nunoerin.tangram"
    ]
  },
  "update": {
    "url": "http://ucreate.nunoerin.com/updates/update.json",
    "time": "13:10",
    "logs": "http://ucreate.nunoerin.com/upload.php"
  },
  "screenSaver": {
    "mode": "slideShow",
```

```
"path":"/mnt/sdcard/Slideshows",  
"slideShowDelay":"10",  
"timeout": "120"  
}  
}
```

## Applications Section

The section is where the list of applications to display in the Launcher is defined.

**“package”**- application package name (i.e. Ants: nunoerin.ants; Draw: nunoerin.draw)

**“icon”** – application icon, could be a local file (ex: <file:///mnt/sdcard/icons/dots.png>) or a URL (ex.

[http://t0.gstatic.com/images?q=tbn:AND9GcQ0U8q9fL-BOs61eOsZbIF-](http://t0.gstatic.com/images?q=tbn:AND9GcQ0U8q9fL-BOs61eOsZbIF-AZ6gD4h48shSHPm9NRpYPTWTAjQ)

[AZ6gD4h48shSHPm9NRpYPTWTAjQ](http://t0.gstatic.com/images?q=tbn:AND9GcQ0U8q9fL-BOs61eOsZbIF-AZ6gD4h48shSHPm9NRpYPTWTAjQ))

## AllowedForegroundApplications Section

This section contains a list of application packages that are allowed to work in kiosk mode together with applications from the “applications” list. For example, if some game requires the Google Play Games package, then add “com.google.android.play.games” here.

## ApplicationList Section

This section contains a list of applications that are displayed in **Settings->Installed Applications**

**“mode”** – sets display mode. Possible values:

- **“whiteList”** (only applications from **“applications”** list will be displayed)
- **“blackList”** (only applications that are NOT in “applications” list will be displayed)

**“applications”** – the list of application packages to display

## Update section

**“url”** – link to software update JSON file (please see Software Update section)

**“time”** – automatic application update time (in military time format)

**“logs”** – link to PHP script for uploading log files to a server

*Please Note: The links to the update JSON file and PHP script for uploading logs files may be specified so as to point to any convenient location on your server. Further, the files may be named whatever you*

wish as long as the names are **exactly** the same as they are on the server. For example, if you wish to separate updates by customer, you could specify the location of the update JSON files as follows:

```
"url": "http://ucreate.nunoerin.com/Customers/McDonalds/Store1/updates/update.json"
```

Similarly, you could specify a unique upload PHP script as follows:

```
"logs": "http://ucreate.nunoerin.com/Customers/McDonalds/Store1/upload.php"
```

Of course, both of these files must exist in the locations specified on the Media Temple server and the "Customers/McDonalds/Store1" folder should contain an "uploads" sub-folder where the logs will be stored.

## ScreenSaver Section

**"image"** – a screen saver image, can be a local file (ex: "file:///mnt/sdcard/wallpapers/wallpaper1.png") or URL (ex: "http://www.planwallpaper.com/static/images/3d-balls-hd-wallpaper.jpg")

Picture file requirements:

- PNG, JPG, GIF, animated GIF formats
- Resolution – up to 1280x720 (does not apply to animated GIF files)
- Aspect ratio 16:9

**"path"** – a slide show folder (ex: "/mnt/sdcard/wallpaper" )

**"slideShowDelay"** – delay between in seconds in showing next picture in slide show mode

**"timeout"** – idle timeout in seconds before screen saver should appear

**"mode"** – screen saver mode. Possible values:

- **"slideShow"**
- **"disabled"**

### NOTES:

1. Orientation of a picture should be the same as specified in the "orientation" option of the Launcher settings.
2. Proper function of animated GIF files should be checked on a factory system before copying them to remote systems.

## How to edit settings

1. Copy a sample factory settings file to a location on a Windows PC.

2. Open the file in a JSON editor program (for example, JSONedit [http://tomeko.net/software/JSONedit/bin/JSONedit\\_0\\_9\\_15\\_1.zip](http://tomeko.net/software/JSONedit/bin/JSONedit_0_9_15_1.zip)).
3. Make desired modifications to the fields described above and save the file.
4. Insert a micro SD card into a card reader and plug it into a PC. (For information on how to prepare an SD card, please see the Firmware section of this document.)
5. Copy the file to the root directory of the card.

## How to set development mode

Development mode can be used for installing and testing new applications on the system. This mode allows launching any application without the limitations of kiosk mode. To set development mode, make the following changes to the **launcher\_settings.json** file:

```
"kioskMode": false,  
  "applicationList": {  
    "mode": "blackList",  
    "applications": []  
  },
```

## How to update settings on existing systems

It is possible to push updated Launcher settings to remote systems over WiFi. This is accomplished by including the Launcher settings file in an update JSON script and making sure the file is stored in the correct location on the Media Temple server. Please see the section **Update Process Example – New Application** above for example update JSON script, which includes updating the Launcher settings.



# Default Settings

The Default settings JSON file allows configuration of volume, time zone, and WiFi at application startup. The **“default\_settings.json”** file should be placed in the root of the internal SD card of the device. Once the Launcher detects the **“default\_settings.json”** file is present on the SD card, it applies any required configuration changes and deletes the default settings JSON file.

Here is the example of **“default\_settings.json”**:

```
{
  "volume": 10,
  "timezone": "Etc/GMT-3",
  "wifi": {
    "ssid": "TestWifiConfig3",
    "security": "WEP",
    "password": "abcdef0123"
  },
  "screenSaver": {
    "mode": "slideShow",
    "path": "/mnt/sdcard/Slideshows",
    "slideShowDelay": "10",
    "timeout": "120"
  }
}
```

**“volume”** – volume level, valid values: 0..100

**“timezone”** – time zone, you can see a list of available time zones at:

<https://gist.github.com/arpit/1035596>

**“wifi”** – wifi network configuration section, here is the complete list of available WiFi options:

- **“ssid”** (mandatory) – WiFi access point name
- **“security”** (mandatory) – security level of WiFi network : “NONE”, “WEP”, “PSK”, or “EAP”
- **“password”** – Wifi network password (only if security is WEP, PSK, or EAP)
- **“eap”** (mandatory if security is “EAP”)
- **“phase2”** (mandatory if security is “EAP”)
- **“identity”** (mandatory if security is “EAP”)
- **“anonymous\_identity”** (mandatory if security is “EAP”)
- **“ca\_cert”** (mandatory if security is “EAP”)
- **“client\_cert”** (mandatory if security is “EAP”)
- **“ip”** (optional) –static IP address if needed
- **“gateway”** (mandatory if “ip” is specified)
- **“prefix\_length”** (mandatory if “ip” is specified)

- ***"dns1"*** (mandatory if "ip" is specified)
- ***"dns2"*** (mandatory if "ip" is specified)

**NOTE:** Before uploading any JSON file to a device, please validate it, since files with an incorrect JSON format could crash the application. You can use online JSON validators like <http://jsonlint.com/> to validate your JSON files.

# Factory Configuration Process

The Factory configuration process includes two mandatory steps:

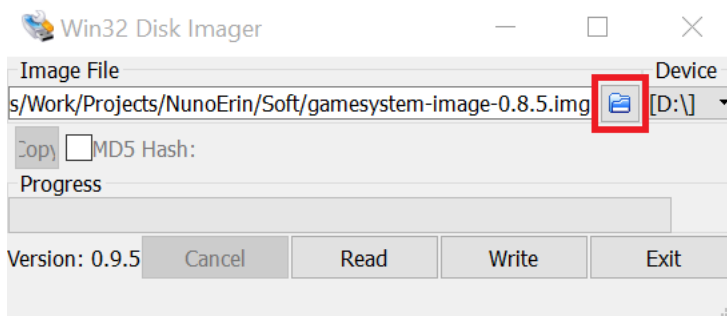
- Micro SD card creation
- Setting serial number

## Environment

- A Windows PC
- A micro SD card reader with a USB cable
- Installed Win32 Disk Imager utility <http://sourceforge.net/projects/win32diskimager/>
- Installed JSON editor program  
[http://tomeko.net/software/JSONedit/bin/JSONedit\\_0\\_9\\_15\\_1.zip](http://tomeko.net/software/JSONedit/bin/JSONedit_0_9_15_1.zip)

## Micro SD card creation

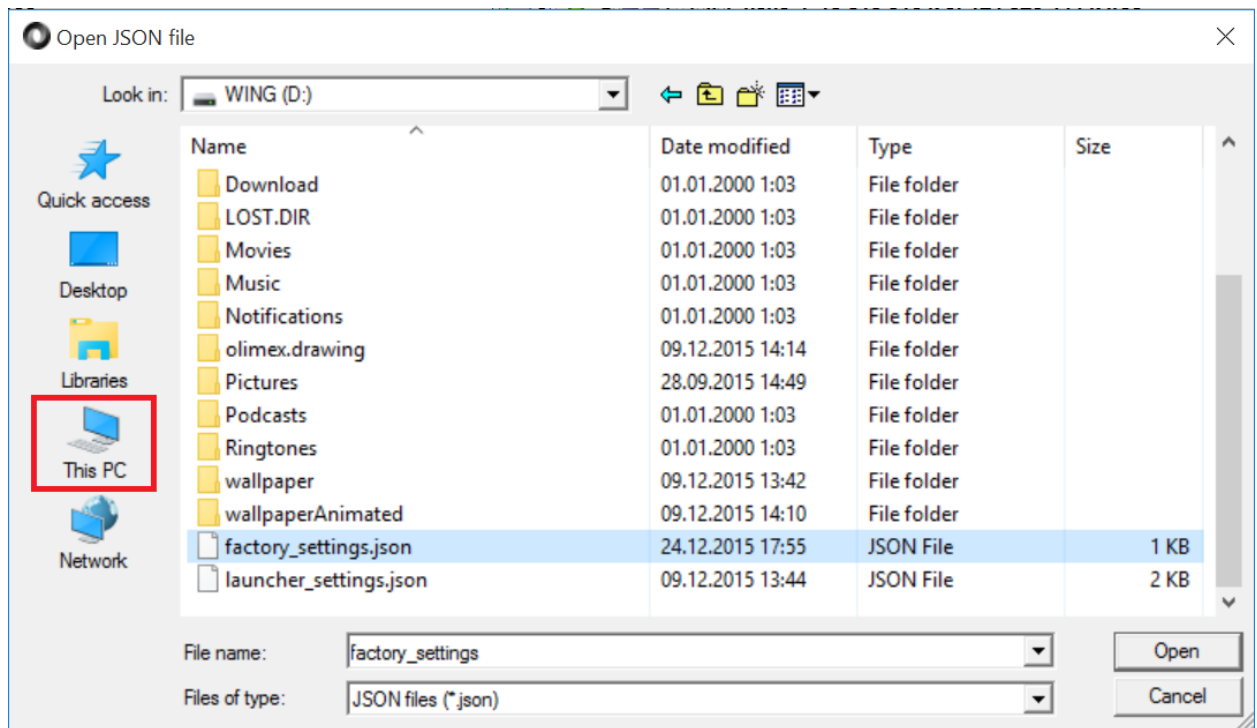
- Insert a Micro SD card in a card reader and connect it to a PC
- Start Win32 Disk Imager utility
- Select a path to an image file by pressing the folder icon (please see the image below)



- Press the “Write” button to write an image to the SD card and wait until the process is completed

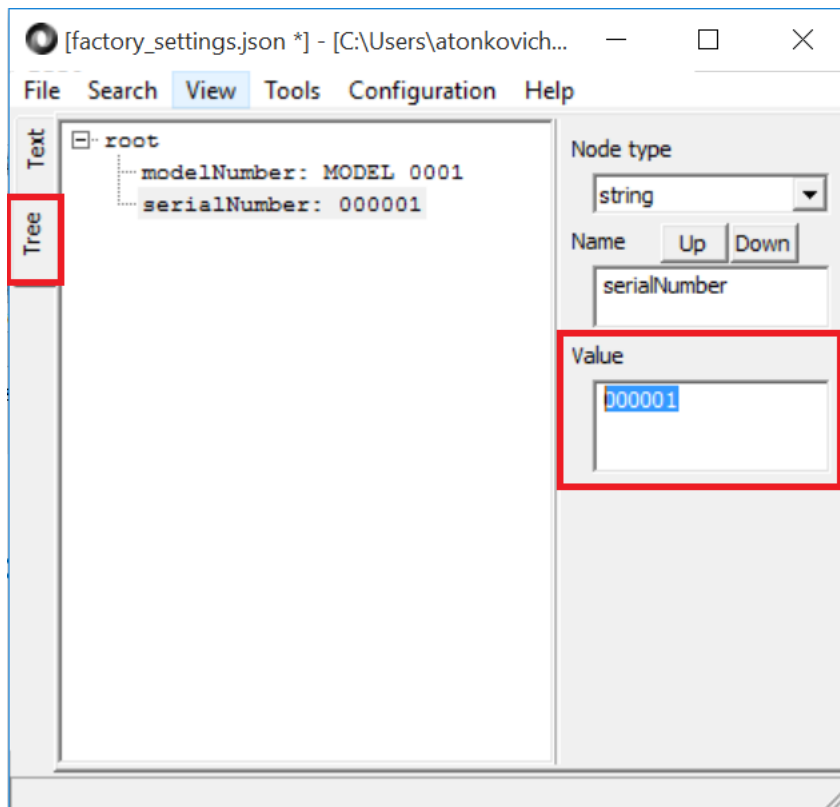
## How to set serial number

- Insert a created Micro SD card in a card reader and connect it to a PC (the action can be omitted if a created Micro SD card is already connected to a PC)
- Start JSONedit program
- Open a **factory\_settings.json** file **File->Open** (a factory\_settings.json file can be found in the root directory of a Micro SD card on the path **This PC-> WING (D:)**)



**Note:** The disk letter may be different

- If it is necessary, switch to the tree view mode
- Select serialNumber entry
- Type a new serial number in the **“Value”** field and save the file (**File->Save**)

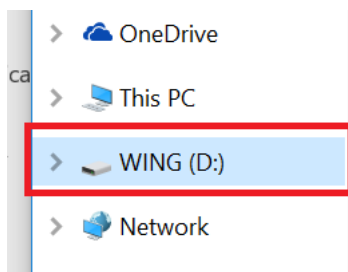


- Disconnect the Micro SD card from the PC

**Note:** The Model number may be changed in a similar manner.

## How to add a new screensaver slideshow folder (optional)

- Insert a Micro SD card in a card reader and connect it to a PC
- Open a file browser program (ex. File Explorer )



**Note:** The disk letter may be different

- Copy an entire folder or folders into the **"Slideshows"** folder

# REFERENCES

**Olimex A20-OLinuxina-MICRO User's Manual:**

<https://www.olimex.com/Products/OLinuxino/A20/A20-OLinuxino-MICRO-4GB/resources/A20-OLinuxino-Micro.pdf>

**Android AAPT Tool:**

[http://elinux.org/Android\\_aapt](http://elinux.org/Android_aapt)

**JSON Editors/Validators:**

<http://jsonlint.com/>

JSONedit [http://tomeko.net/software/JSONedit/bin/JSONedit\\_0\\_9\\_15\\_1.zip](http://tomeko.net/software/JSONedit/bin/JSONedit_0_9_15_1.zip)

**Worldwide Time Zones:**

<https://gist.github.com/arpit/1035596>

**Win32 Disk Imager Utility:**

<http://sourceforge.net/projects/win32diskimager/>