

Raspberry Pi Parcel Locker

2 in 1 dual system for online order pickup + parcel locker for multiple apartment tenants.

We required a 24h pickup possibility for online-orders and decided to build a locker based system. Unfortunately there were no tutorials or even ideas online, only commercial ready to use products. So this project can be pretty much regarded as first open source parcel locker + (online) order vending solution.

The online order will kept available for pickup at the locker and the customer can release/open the locker by a credit card payment. If she/he already pre-paid the locker can be unlocked by a pin code.

Additionally, to this we added a second locker for our apartment tenants in order to receive parcels from courier services such as DHL, DPD... Any courier company can unlock and use the locker to leave the shipment in there. In Austria there are currently several "Paketbox" systems, but are mainly only accessible for a single service provider. The main idea is to have an independent system with access to all services. We didn't care about the monopolism policy of the available parcel boxes such as from Austrian Post or Burg Wächter eBoxx. Also those come with ridiculous monthly costs.

The place of setup is semi-public and **heavily** monitored by CCTV. Sturdiness and vandalism durability is somewhat important, but not highest priority. Availability of the materials on the local market was primary importance. Budget is more low-end.

If you plan to rebuild we recommend to adjust materials depending to your requirements (eg. for outdoor use...) Items may be cheaper on Aliexpress/eBay

Material list:

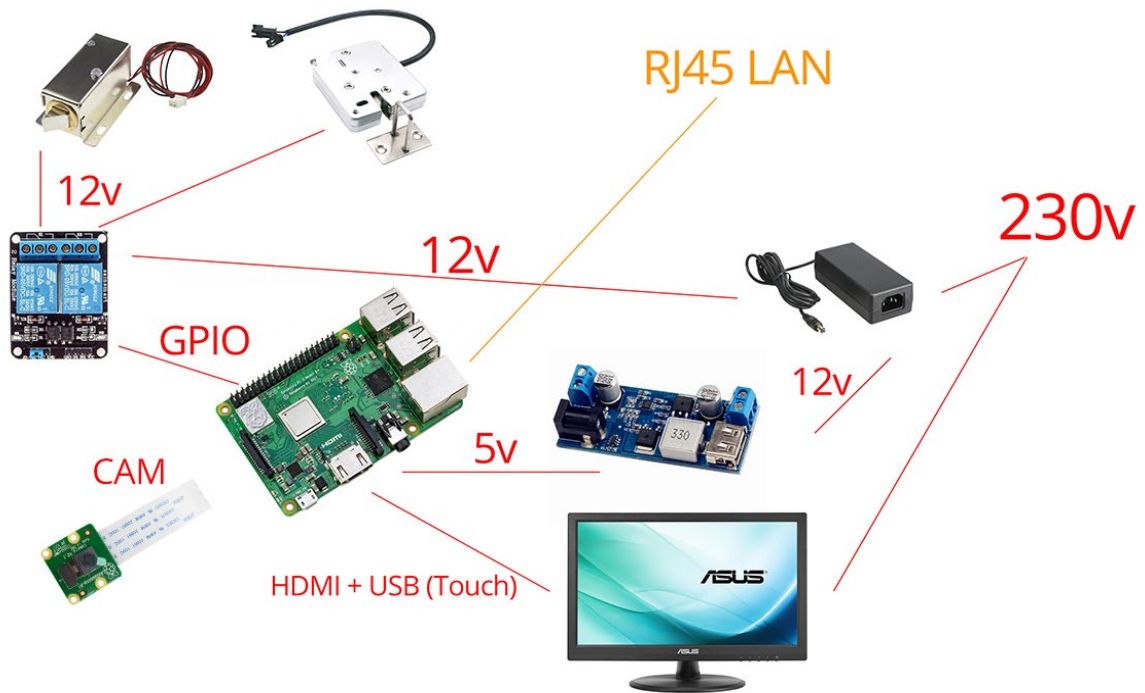
Locker #1: IKEA IVAR 40x160cm <i>(Online-Order Pickup, height required due to items with 110cm)</i>	79€
Locker #2: IKEA IVAR 80x83cm <i>(Tenant Parcel Locker)</i>	79€
Raspberry Pi 3 B+	35€
Raspberry Pi Case	5€
Raspberry Pi Camera + Case	12€
2x 5V Relay Module	5€
32GB Class 10 SD	8€
12V 6A Power Supply	12€
12V -> 5V Step Down Converter <i>(Decided to build a 12V based system since locks operate with 12V. RPi 5V via step down)</i>	7€
2x 12V Cabin Lock <i>(One mortice-style and one shackle lock)</i>	15€
1x ASUS Touch Monitor 15" (VT168N)	145€
LED Lightning, Screws, Electric Wires, Vinyl Labels...	30€

432€ – 450€

Build Time Hardware approx. 3-4h | Software development approx. 3-4h

***cheapest possible single cabin system based on RPi ZERO w/ smaller display = under 130€**

Hardware Schematic

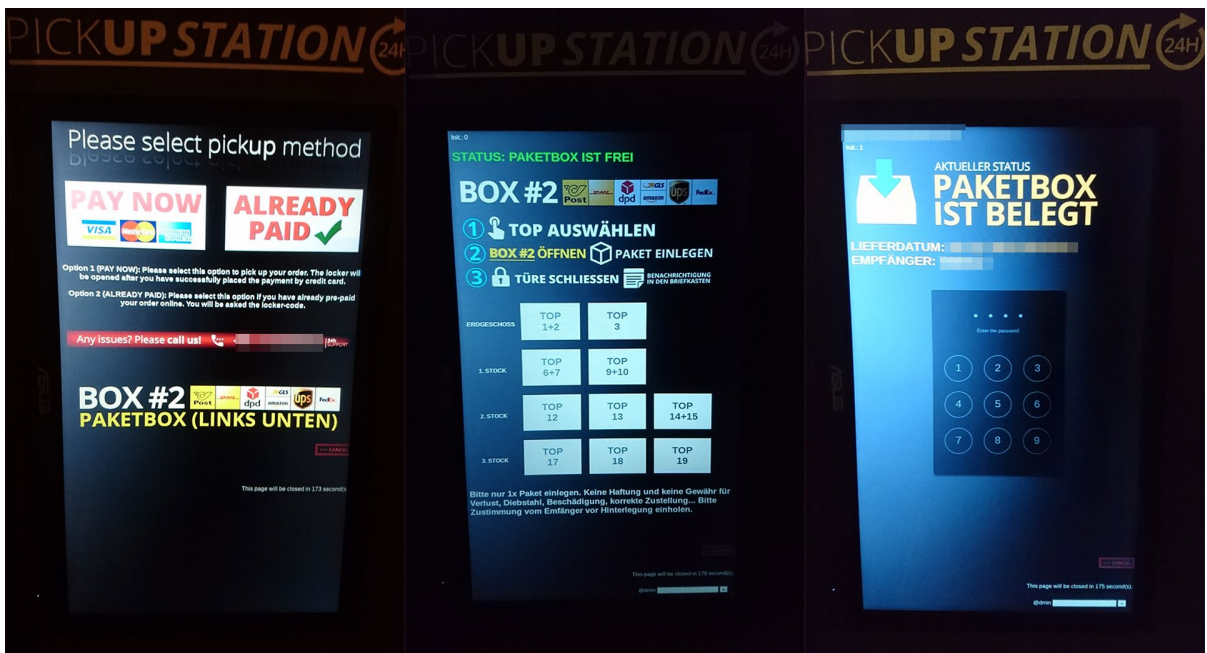


Software Schematic



created with www.bubbl.us

Finished Project Pics



Hardware Setup

Basic hardware knowledge is required. I won't get too deep into the process of assembling the hardware for this project. Having good soldering skills will make the whole process easier.

Wire all modules together. For the relay we will use GPIO pins of the Raspberry. If you don't know about GPIO please check online. If you buy a Zero Pi you will need to solder a GPIO header to it.

Since the electronic locks are operated by 12v we used as power supply a 12v PSU with 6A. Make sure to shield the wiring of the locks properly since the electromagnetic impulses of the lock might interfere with your hardware. *Didn't think of this first and was wondering why the display blacking out when unlocking :-O* In order to power your Pi a 12v -> 5v step-down-converter (with USB) is required. All the hardware is packed in a plastic box, which is mounted in one of the cabins. IKEA IVAR does not provide standardized lock mounts so some improvement is needed to fit the 12v locks. 40x160cm model can use mortice-style lock. The 80x83cm model requires a cabin-shackle lock due to the double door. Since the locks are impulse based it is a full failsafe solution.

The display is mounted to the 40x160 model by rotating it 90°. You could also use a smaller display rather than a 15". There are wires for 230v routed to the main cabin. Make sure to ground it properly in order to prevent kids and idiots to get shocked.

Software

1. Install Raspbian/NOOBS
2. Install Apache + PHP
3. Configure RPI (raspi-config), Enable SSH + Camera
4. Compile MJPG Streamer
5. Setup UI

Below is pretty much the whole code used for this project. Starting from HW related codes to the UI.

Rotate Display 90°:

```
/boot/config.txt  
lcd_rotate=2
```

Rotate Touch 90°:

```
/etc/rc.local  
xinput --set-prop 'ELAN Touchscreen' 'Coordinate Transformation Matrix' 0 -1 1 1 0 0 0 0 1
```

MJPG Streamer:

```
/etc/rc.local  
/etc/init.d/webcam start
```

Webcam Script (/etc/init.d/webcam):

```
#!/bin/sh  
# /etc/init.d/webcam  
# Carry out specific functions when asked to by the system  
case "$1" in  
  start)  
    echo "Starting web"  
    export LD_LIBRARY_PATH=/home/pi/projects/mjpg-streamer/mjpg-streamer-experimental  
    sudo /home/pi/projects/mjpg-streamer/mjpg-streamer-experimental/mjpg_streamer -o  
"/home/pi/projects/mjpg-streamer/mjpg-streamer-experimenta$  
  ;;  
  stop)  
    echo "Stopping webcam script"
```

```
killall mjpg_streamer
;;
*)
echo "Usage: /etc/init.d/webcam {start|stop}"
exit 1
;;
esac
exit 0
```

Disable Screensaver

```
sudo apt-get install xscreensaver
```

Chromium Kiosk Mode

```
sudo apt-get install xdotool unclutter sed
```

```
nano /home/pi/kiosk.sh
```

```
#!/bin/bash
xset s noblank
xset s off
xset -dpms

unclutter -idle 0.5 -root &

/usr/bin/chromium-browser --noerrdialogs --disable-infobars --kiosk http://localhost/

done
```

Thanks to: <https://pimylifeup.com/raspberry-pi-kiosk/>

UI

The user interface is fully based on PHP, HTML, JAVA. It requires no MySQL database but uses a small .txt file in order to store the status of the parcel locker. Also if the power goes out it will not affect the status of the locker. Each tenant has an own PIN code. It also stores the delivery time and date and you can backup the protocol. For credit card payment we have an online POS website which was simply cloned and the confirmation page was modified to redirect to local UNLOCK.php

INDEX.PHP / INDEX.HTML

Intro of your choice. We use fancy parallax particles animation 8-)

SELECT.HTML

Here the user can choose between PARCEL LOCKER or ORDER PICKUP (CREDIT CARD FORM OR PIN CODE)

CREDITCARD.PHP

Our usual credit card form is used. The last page redirects to UNLOCK.PHP

PINCODE.HTML

PIN Code UI is displayed. When code is correct, redirect to UNLOCK.PHP – The styling and java is inline...sorry for the mess. PIN in the sample code is set to 1234:

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>PIN CODE</title>

<style>
body {
height: 100vh;
```

```

background: #494949;
display: -webkit-box;
display: -webkit-flex;
display: -ms-flexbox;
display: flex;
-webkit-box-align: center;
-webkit-align-items: center;
-ms-flex-align: center;
align-items: center;
-webkit-box-pack: center;
-webkit-justify-content: center;
-ms-flex-pack: center;
justify-content: center;
font-family: Verdana;
}

body.wrong {
-webkit-animation: bg-red 1s ease-in;
animation: bg-red 1s ease-in;
}

body.correct {
-webkit-animation: bg-green 1s ease-in;
animation: bg-green 1s ease-in;
}

#inspiration {
position: fixed;
right: 1em;
bottom: 1em;
}

#inspiration a {
display: inline-block;
float: left;
text-decoration: none;
font-weight: bold;
color: white;
-webkit-transition: all 1s ease;
transition: all 1s ease;
}

#inspiration a:hover { color: #212121; }

#inspiration p {
margin: 0;
padding-left: .4em;
display: inline-block;
float: right;
color: rgba(255, 255, 255, 0.6);
}

#pin {
background: #212121;
width: 20em;
display: -webkit-box;
display: -webkit-flex;
display: -ms-flexbox;
display: flex;
-webkit-box-pack: center;
-webkit-justify-content: center;
-ms-flex-pack: center;
justify-content: center;
-webkit-box-align: center;
-webkit-align-items: center;
-ms-flex-align: center;
align-items: center;
-webkit-box-orient: vertical;
-webkit-box-direction: normal;
-webkit-flex-direction: column;
-ms-flex-direction: column;
flex-direction: column;
padding: 1em;
border-radius: .3em;
box-shadow: 4px 4px 8px rgba(0, 0, 0, 0.3);
margin: auto;
color: rgba(255, 255, 255, 0.2);
}

.dots {
width: 50%;
display: -webkit-box;
display: -webkit-flex;
display: -ms-flexbox;
display: flex;
-webkit-justify-content: space-around;
-ms-flex-pack: distribute;
justify-content: space-around;
padding: 1em;
padding-top: 3em;
}

.dot {
position: relative;
background: rgba(255, 255, 255, 0.2);
border-radius: 0.8em;
width: 0.8em;
height: 0.8em;
-webkit-transform: scale3d(0.7, 0.7, 0.7);
transform: scale3d(0.7, 0.7, 0.7);
}

```

```

.dot.active {
  -webkit-animation: growDot .5s ease;
  animation: growDot .5s ease;
  -webkit-animation-fill-mode: forwards;
  animation-fill-mode: forwards;
}

.dot.wrong {
  -webkit-animation: wrong .9s ease;
  animation: wrong .9s ease;
}

.dot.correct {
  -webkit-animation: correct .9s ease;
  animation: correct .9s ease;
}

#pin p { font-size: .8em; }

.numbers {
  display: -webkit-box;
  display: -webkit-flex;
  display: -ms-flexbox;
  display: flex;
  -webkit-flex-flow: row wrap;
  -ms-flex-flow: row wrap;
  flex-flow: row wrap;
  -webkit-box-align: center;
  -webkit-align-items: center;
  -ms-flex-align: center;
  align-items: center;
  -webkit-justify-content: space-around;
  -ms-flex-pack: distribute;
  justify-content: space-around;
  -webkit-align-content: flex-end;
  -ms-flex-line-pack: end;
  align-content: flex-end;
  margin: 2em 0;
}

.number {
  position: relative;
  width: 2.5em;
  height: 2.5em;
  margin: 0.5em;
  border-radius: 2.5em;
  border: 2px solid rgba(255, 255, 255, 0);
  text-align: center;
  line-height: 2.5em;
  font-weight: 400;
  font-size: 1.8em;
  -webkit-user-select: none;
  -moz-user-select: none;
  -ms-user-select: none;
  user-select: none;
  -webkit-transition: all .5s ease;
  transition: all .5s ease;
}

.number:hover { color: rgba(255, 255, 255, 0.5); }

.number:hover:before { border: 2px solid rgba(255, 255, 255, 0.5); }

.number:before {
  content: "";
  position: absolute;
  left: -2px;
  width: 2.5em;
  height: 2.5em;
  border: 2px solid rgba(255, 255, 255, 0.1);
  border-radius: 2.5em;
  -webkit-transition: all .5s ease;
  transition: all .5s ease;
}

.number.grow:before {
  -webkit-animation: grow .6s ease;
  animation: grow .6s ease;
}

@-webkit-keyframes
growDot { 100% {
  background: white;
  -webkit-transform: scale3d(0.9, 0.9, 0.9);
  transform: scale3d(0.9, 0.9, 0.9);
}
}

@keyframes
growDot { 100% {
  background: white;
  -webkit-transform: scale3d(0.9, 0.9, 0.9);
  transform: scale3d(0.9, 0.9, 0.9);
}
}

@-webkit-keyframes
grow { 50% {
  -webkit-transform: scale3d(1.5, 1.5, 1.5);
  transform: scale3d(1.5, 1.5, 1.5);
}
  100% {
  -webkit-transform: scale3d(1, 1, 1);
  transform: scale3d(1, 1, 1);
}
}

```

```

}
@keyframes
grow { 50% {
  -webkit-transform: scale3d(1.5, 1.5, 1.5);
  transform: scale3d(1.5, 1.5, 1.5);
}
  100% {
  -webkit-transform: scale3d(1, 1, 1);
  transform: scale3d(1, 1, 1);
}
}
@-webkit-keyframes
wrong { 20% {
  background: crimson;
}
  40% {
  -webkit-transform: translate(-15px, 0);
  transform: translate(-15px, 0);
}
  60% {
  -webkit-transform: translate(10px, 0);
  transform: translate(10px, 0);
}
  80% {
  -webkit-transform: translate(-5px, 0);
  transform: translate(-5px, 0);
}
}
@keyframes
wrong { 20% {
  background: crimson;
}
  40% {
  -webkit-transform: translate(-15px, 0);
  transform: translate(-15px, 0);
}
  60% {
  -webkit-transform: translate(10px, 0);
  transform: translate(10px, 0);
}
  80% {
  -webkit-transform: translate(-5px, 0);
  transform: translate(-5px, 0);
}
}
@-webkit-keyframes
correct { 20% {
  background: limegreen;
}
  40% {
  -webkit-transform: translate(0, -15px);
  transform: translate(0, -15px);
}
  60% {
  -webkit-transform: translate(0, 10px);
  transform: translate(0, 10px);
}
  80% {
  -webkit-transform: translate(0, -5px);
  transform: translate(0, -5px);
}
}
@keyframes
correct { 20% {
  background: limegreen;
}
  40% {
  -webkit-transform: translate(0, -15px);
  transform: translate(0, -15px);
}
  60% {
  -webkit-transform: translate(0, 10px);
  transform: translate(0, 10px);
}
  80% {
  -webkit-transform: translate(0, -5px);
  transform: translate(0, -5px);
}
}
@-webkit-keyframes
bg-red { 50% {
  background: crimson;
}
}
@keyframes
bg-red { 50% {
  background: crimson;
}
}
@-webkit-keyframes
bg-green { 50% {
  background: limegreen;
}
}
@keyframes
bg-green { 50% {
  background: limegreen;
}
}
</style>
</head>

```



```

<body>
<div id="inspiration"><a href="index.html"></a>
</div>
<div id="pin">
  <div class="dots">
    <div class="dot"></div>
    <div class="dot"></div>
    <div class="dot"></div>
    <div class="dot"></div>
  </div>
  <p>Enter the password</p>
  <div class="numbers">
    <div class="number">1</div>
    <div class="number">2</div>
    <div class="number">3</div>
    <div class="number">4</div>
    <div class="number">5</div>
    <div class="number">6</div>
    <div class="number">7</div>
    <div class="number">8</div>
    <div class="number">9</div>
  </div>
</div>
<script>(function () {
  var input = '', correct = '1234';
  var dots = document.querySelectorAll('.dot'), numbers = document.querySelectorAll('.number');
  dots = Array.prototype.slice.call(dots);
  numbers = Array.prototype.slice.call(numbers);
  numbers.forEach(function (number, index) {
    number.addEventListener('click', function () {
      number.className += ' grow';
      input += index + 1;
      dots[input.length - 1].className += ' active';
      if (input.length >= 4) {
        if (input !== correct) {
          dots.forEach(function (dot, index) {
            dot.className += ' wrong';
          });
          document.body.className += ' wrong';
        } else {
          dots.forEach(function (dot, index) {
            dot.className += ' correct';
          });
          document.body.className += ' correct';
        }
      }
      document.body.className += ' correct';
    });
  });
  document.ready(window.setTimeout(location.href = "unlock.php",10));

  }
  setTimeout(function () {
    dots.forEach(function (dot, index) {
      dot.className = 'dot';
    });
    input = '';
  }, 900);
  setTimeout(function () {
    document.body.className = '';
  }, 1000);
}
setTimeout(function () {
  number.className = 'number';
}, 1000);
});
})();
//# sourceMappingURL=pen.js
</script>
</body>
</html>

```

UNLOCK.PHP

This page automatically triggers GPIO 29. First it sets GPIO Mode to out, then closes the relay and opens it again. User has the possibility to manually unlock again. The page automatically closes after 180 seconds back to the index.html – Also you can trigger it remote by the URL .../?trigger=1

```

<?php
  if(isset($_GET['trigger']) && $_GET['trigger'] == 1) {
    error_reporting(E_ALL);
    exec('gpio mode 29 out');
    usleep(100);
    exec('gpio write 29 1');
    usleep(9000000);
    exec('gpio write 29 0');
  }
?>
<!DOCTYPE html>
<html>
<script type="text/javascript">
  function codeAddress() {
    $.get("/?trigger=1");
  }
  window.onload = codeAddress;
</script>
<head>
  <title>PickUpStation Unlock</title>
  <link rel="stylesheet" href="style.css" type="text/css">

```

```

<script type="text/javascript" src="/js/jquery-1.10.2.min.js"></script>

</head>
<body>
<div align="center" style="padding-top:200px;">



</div>

<div class='awrap'>
<a href="/?trigger=1"></a>
</div>

<br /><br />
<div style="text-align: right;"><br>
<br>
<p>This page will be closed in <span id="counter">180</span> second(s).</p>
<script type="text/javascript">
function countdown() {
    var i = document.getElementById('counter');
    if (parseInt(i.innerHTML)<=0) {
        location.href = 'start.html';
    }
    i.innerHTML = parseInt(i.innerHTML)-1;
}
setInterval(function(){ countdown(); },1000);
</script>
<a href="index.html"></a></div>

</body>
</html>

```

PARCELLOCKER.PHP (BOX2.PHP)

This page is for the Parcel Locker System. It contains an overview of the tenant door numbers where delivery service selects the receiver. When the parcel locker is occupied it displays a message and features a PIN code form for the unlock. It also has an admin interface for manual unlock and database flush + backup. It also saves and displays the time when the parcel was delivered. Optionally you can integrate phpmailer in order to send out a message to the tenant once they receive a parcel. For this you will need to create new cases with the phpmailer script in it. CSS and JAVA is again inlay. **The whole system saves data to a .txt and does not require MySQL.**

Each tenant has an own .php file in order to manage PIN codes more easily.

In this case it uses again GPIO 29. Make sure to change it to your other relay.

```

<?php
    if(isset($_GET['trigger']) && $_GET['trigger'] == 1) {
        error_reporting(E_ALL);
        exec('gpio mode 29 out');
        usleep(100);
        exec('gpio write 29 1');
        usleep(9000000);
        exec('gpio write 29 0');
    }
?>

<?php

if (!isset($operation)) {
    $operation = 0;
}

$id=$_REQUEST['operation'];
$note=$_REQUEST['note'];

if (empty($note)) {
    $note = 'N/A';
}

$dataFile = "data.txt";
$time = date("d M Y H:i:s");

$record = $time.", ".$note.PHP_EOL;

switch($id) {

```

```

case 1: //insert new recordi

    $fh = fopen($dataFile, 'a') or die("can't open file");
    fwrite($fh, $record);
    fclose($fh);

        exec('gpio mode 29 out');
        usleep(100);
        exec('gpio write 29 1');
        usleep(9000000);
        exec('gpio write 29 0');

    print "<p>Inserted: ".$record."</p>";

break;

case 2: //clear file
    $fh = fopen( $dataFile, "r+" );
    ftruncate($fh, 0);
    fclose($fh);

    exec('gpio mode 29 out');
    usleep(100);
    exec('gpio write 29 1');
    usleep(9000000);
    exec('gpio write 29 0');

    print "<p> Status: Empty </p>";

break;

case 3; //list the content of the file
    $count = substr_count(file_get_contents($dataFile), "\n");
    print "<p>".$count." record(s) in the text database: <br>";

    $fh = fopen($dataFile, 'r') or die("can't open file");
    $n = 1;

    for ($i=1; $i<$count+1; $i++){
        $line = fgets($fh, 1024);
        $list = explode(" ", $line);
        $r = urlencode($line);
        print $i." " . $list[0]. " => " . $list[1]. " | " . "<a href=$self?operation=4&line_no=\$i\>return
confirm('Delete the record?')\>delete</a>".<br>";
    }
    print "</p>";
    fclose($fh);

break;

case 4; //delete line
    function outline($filename,$line_no=-1) {

        $strip_return=FALSE;

        $data=file($filename);
        $pipe=fopen($filename,'w');
        $size=count($data);

        if($line_no==-1) $skip=$size-1;
        else $skip=$line_no-1;

        for($line=0;$line<$size;$line++)
        if($line!=$skip)
        fputs($pipe,$data[$line]);
        else
        $strip_return=TRUE;
        return $strip_return;
    }

    $line_no = trim ( $_GET['line_no'], '' ) ;

    outline($dataFile, $line_no); // deletes line
break;

case 5: //Backup the file

    $d = new DateTime();
    $timeStamp = $d -> getTimestamp();

    $backupFile = "backup/dataBackup_{$timeStamp}.txt";

    if (!copy($dataFile, $backupFile)) {
        print "<p>Failed to backup the database! $dataFile, $backupFile</p>";
    } else {
        print "<p> Backup succeeded!<br> " . "<a href=\"\$backupFile\>Examine the backed up file</a></p>";
    }

break;

case 99;
    show_source(__FILE__);

```

```

        break;

        case 0: //do nothing

        break;

}

?><html>
<head>
        <meta charset="utf-8" />
        <title>PAKETBOX</title>
        <style>

body.wrong {
    -webkit-animation: bg-red 1s ease-in;
    animation: bg-red 1s ease-in;
}

body.correct {
    -webkit-animation: bg-green 1s ease-in;
    animation: bg-green 1s ease-in;
}

#inspiration {
    position: fixed;
    right: 1em;
    bottom: 1em;
}

#inspiration a {
    display: inline-block;
    float: left;
    text-decoration: none;
    font-weight: bold;
    color: white;
    -webkit-transition: all 1s ease;
    transition: all 1s ease;
}

#inspiration a:hover { color: #212121; }

#inspiration p {
    margin: 0;
    padding-left: .4em;
    display: inline-block;
    float: right;
    color: rgba(255, 255, 255, 0.6);
}

#pin {
    background: #212121;
    width: 20em;
    display: -webkit-box;
    display: -webkit-flex;
    display: -ms-flexbox;
    display: flex;
    -webkit-box-pack: center;
    -webkit-justify-content: center;
    -ms-flex-pack: center;
    justify-content: center;
    -webkit-box-align: center;
    -webkit-align-items: center;
    -ms-flex-align: center;
    align-items: center;
    -webkit-box-orient: vertical;
    -webkit-box-direction: normal;
    -webkit-flex-direction: column;
    -ms-flex-direction: column;
    flex-direction: column;
    padding: 1em;
    border-radius: .3em;
    box-shadow: 4px 4px 8px rgba(0, 0, 0, 0.3);
    margin: auto;
    color: rgba(255, 255, 255, 0.2);
}

.dots {
    width: 50%;
    display: -webkit-box;
    display: -webkit-flex;
    display: -ms-flexbox;
    display: flex;
    -webkit-justify-content: space-around;
    -ms-flex-pack: distribute;
    justify-content: space-around;
    padding: 1em;
    padding-top: 3em;
}

.dot {
    position: relative;
    background: rgba(255, 255, 255, 0.2);
}

```

```

border-radius: 0.8em;
width: 0.8em;
height: 0.8em;
-webkit-transform: scale3d(0.7, 0.7, 0.7);
transform: scale3d(0.7, 0.7, 0.7);
}

.dot.active {
-webkit-animation: growDot .5s ease;
animation: growDot .5s ease;
-webkit-animation-fill-mode: forwards;
animation-fill-mode: forwards;
}

.dot.wrong {
-webkit-animation: wrong .9s ease;
animation: wrong .9s ease;
}

.dot.correct {
-webkit-animation: correct .9s ease;
animation: correct .9s ease;
}

#pin p { font-size: .8em; }

.numbers {
display: -webkit-box;
display: -webkit-flex;
display: -ms-flexbox;
display: flex;
-webkit-flex-flow: row wrap;
-ms-flex-flow: row wrap;
flex-flow: row wrap;
-webkit-box-align: center;
-webkit-align-items: center;
-ms-flex-align: center;
align-items: center;
-webkit-justify-content: space-around;
-ms-flex-pack: distribute;
justify-content: space-around;
-webkit-align-content: flex-end;
-ms-flex-line-pack: end;
align-content: flex-end;
margin: 2em 0;
}

.number {
position: relative;
width: 2.5em;
height: 2.5em;
margin: 0.5em;
border-radius: 2.5em;
border: 2px solid rgba(255, 255, 255, 0);
text-align: center;
line-height: 2.5em;
font-weight: 400;
font-size: 1.8em;
-webkit-user-select: none;
-moz-user-select: none;
-ms-user-select: none;
user-select: none;
-webkit-transition: all .5s ease;
transition: all .5s ease;
}

.number:hover { color: rgba(255, 255, 255, 0.5); }

.number:hover:before { border: 2px solid rgba(255, 255, 255, 0.5); }

.number:before {
content: "";
position: absolute;
left: -2px;
width: 2.5em;
height: 2.5em;
border: 2px solid rgba(255, 255, 255, 0.1);
border-radius: 2.5em;
-webkit-transition: all .5s ease;
transition: all .5s ease;
}

.number.grow:before {
-webkit-animation: grow .6s ease;
animation: grow .6s ease;
}

@-webkit-keyframes
growDot { 100% {
background: white;
-webkit-transform: scale3d(0.9, 0.9, 0.9);
transform: scale3d(0.9, 0.9, 0.9);
}
}

@keyframes
growDot { 100% {

```

```

background: white;
-webkit-transform: scale3d(0.9, 0.9, 0.9);
transform: scale3d(0.9, 0.9, 0.9);
}
}
@-webkit-keyframes
grow { 50% {
-webkit-transform: scale3d(1.5, 1.5, 1.5);
transform: scale3d(1.5, 1.5, 1.5);
}
100% {
-webkit-transform: scale3d(1, 1, 1);
transform: scale3d(1, 1, 1);
}
}
@keyframes
grow { 50% {
-webkit-transform: scale3d(1.5, 1.5, 1.5);
transform: scale3d(1.5, 1.5, 1.5);
}
100% {
-webkit-transform: scale3d(1, 1, 1);
transform: scale3d(1, 1, 1);
}
}
@-webkit-keyframes
wrong { 20% {
background: crimson;
}
40% {
-webkit-transform: translate(-15px, 0);
transform: translate(-15px, 0);
}
60% {
-webkit-transform: translate(10px, 0);
transform: translate(10px, 0);
}
80% {
-webkit-transform: translate(-5px, 0);
transform: translate(-5px, 0);
}
}
@keyframes
wrong { 20% {
background: crimson;
}
40% {
-webkit-transform: translate(-15px, 0);
transform: translate(-15px, 0);
}
60% {
-webkit-transform: translate(10px, 0);
transform: translate(10px, 0);
}
80% {
-webkit-transform: translate(-5px, 0);
transform: translate(-5px, 0);
}
}
@-webkit-keyframes
correct { 20% {
background: limegreen;
}
40% {
-webkit-transform: translate(0, -15px);
transform: translate(0, -15px);
}
60% {
-webkit-transform: translate(0, 10px);
transform: translate(0, 10px);
}
80% {
-webkit-transform: translate(0, -5px);
transform: translate(0, -5px);
}
}
@keyframes
correct { 20% {
background: limegreen;
}
40% {
-webkit-transform: translate(0, -15px);
transform: translate(0, -15px);
}
60% {
-webkit-transform: translate(0, 10px);
transform: translate(0, 10px);
}
80% {
-webkit-transform: translate(0, -5px);
transform: translate(0, -5px);
}
}
@-webkit-keyframes
bg-red { 50% {

```

```

background: crimson;
}
}
@keyframes
bg-red { 50% {
background: crimson;
}
}
@-webkit-keyframes
bg-green { 50% {
background: limegreen;
}
}
@keyframes
bg-green { 50% {
background: limegreen;
}
}

#HIDDENDIV {
display: none;
}

#table td {
padding: 1em;
border: 1px solid black;
}

#table.show tr > *:nth-child(2) {
display: block;
}

#adminpanel {
position: fixed;
bottom: 0;
right: 0;
}

body,td,th {
color: #FFF;
font-family: Verdana, Geneva, sans-serif;
font-size: 16px;
}

body {
background-color: #333;
}

a:link {
color: #FFF;
}
a:visited {
color: #FFF;
}

.myButton {
background:-webkit-gradient(linear, left top, left bottom, color-stop(0.05, #f9f9f9), color-stop(1, #e9e9e9));
background:-moz-linear-gradient(top, #f9f9f9 5%, #e9e9e9 100%);
background:-webkit-linear-gradient(top, #f9f9f9 5%, #e9e9e9 100%);
background:-o-linear-gradient(top, #f9f9f9 5%, #e9e9e9 100%);
background:-ms-linear-gradient(top, #f9f9f9 5%, #e9e9e9 100%);
background:linear-gradient(to bottom, #f9f9f9 5%, #e9e9e9 100%);
filter:progid:DXImageTransform.Microsoft.gradient(startColorstr='#f9f9f9', endColorstr='#e9e9e9',GradientType=0);
background-color:#f9f9f9;
border:3px solid #dcdcdc;
display:inline-block;
cursor:pointer;
color:#666666;
font-family:Arial;
font-size:28px;
font-weight:bold;
padding:18px 40px;
text-decoration:none;
width: 160px;
}

.myButton:hover {
background:-webkit-gradient(linear, left top, left bottom, color-stop(0.05, #e9e9e9), color-stop(1, #f9f9f9));
background:-moz-linear-gradient(top, #e9e9e9 5%, #f9f9f9 100%);
background:-webkit-linear-gradient(top, #e9e9e9 5%, #f9f9f9 100%);
background:-o-linear-gradient(top, #e9e9e9 5%, #f9f9f9 100%);
background:-ms-linear-gradient(top, #e9e9e9 5%, #f9f9f9 100%);
background:linear-gradient(to bottom, #e9e9e9 5%, #f9f9f9 100%);
filter:progid:DXImageTransform.Microsoft.gradient(startColorstr='#e9e9e9', endColorstr='#f9f9f9',GradientType=0);
background-color:#e9e9e9;
}

.myButton:active {
position:relative;
top:1px;
}

}

</style>
</head>
<body alink="#FFFFFF">

```

```

<?php
$count = substr_count(file_get_contents($dataFile), "\n");
print "<p>Init.: ".$count."<br>";

$fh = fopen($dataFile, 'r') or die("can't open file");
$n = 1;
    for ($i=1; $i<$count+1; $i++){
        $line = fgets($fh, 1024);
        $list = explode(",", $line);
        $r = urlencode($line);

        print "<img src='occu.jpg' /><br><h1 style='color:#F90'>LIEFERDATUM: ".$list[0]." <br>
EMPFÄNGER: <a href='$list[1].php'>$list[1]</a></h1>";

        fclose($fh);
    }

<?php
    $t = substr_count(file_get_contents($dataFile), "\n");

if ($t < "1") {
    echo "<h1 style='color:#0F0'>STATUS: PAKETBOX IST FREI</h1><img src=info.jpg /><br><br>

        <table width='700' border='0' cellpadding='5'>
        <tr>
        <th scope='row'>ERDGESCHOSS</th>
        <td><form action=TOP1+2.php method=post>
            <button class='mybutton' type=submit>TOP 1+2</button>
            <input type=hidden name=note value=TOP1+2>
            <input type=hidden name=operation value=1>
            </form> </td>
        <td> <form action=TOP3.php method=post>
            <button class='mybutton' type=submit> TOP 3 </button>
            <input type=hidden name=note value=TOP3>
            <input type=hidden name=operation value=1>
            </form></td>
        <td>&nbsp;</td>
        </tr>
        <tr>
        <th scope='row'>1. STOCK</th>
        <td>
            <form action=TOP6+7.php method=post>
            <button class='mybutton' type=submit> TOP 6+7 </button>
            <input type=hidden name=note value=TOP6+7>
            <input type=hidden name=operation value=1>
            </form></td>
        <td>
            <form action=TOP9+10.php method=post>
            <button class='mybutton' type=submit> TOP 9+10 </button>
            <input type=hidden name=note value=TOP9+10>
            <input type=hidden name=operation value=1>
            </form></td>
        <td>&nbsp;</td>
        </tr>
        <tr>
        <th scope='row'>2. STOCK</th>
        <td>
            <form action=TOP12.php method=post>
            <button class='mybutton' type=submit> TOP 12 </button>
            <input type=hidden name=note value=TOP12>
            <input type=hidden name=operation value=1>
            </form></td>
        <td>
            <form action=TOP13.php method=post>
            <button class='mybutton' type=submit> TOP 13 </button>
            <input type=hidden name=note value=TOP13>
            <input type=hidden name=operation value=1>
            </form></td>
        <td>
            <form action=TOP14+15.php method=post>
            <button class='mybutton' type=submit> TOP 14+15 </button>
            <input type=hidden name=note value=TOP14+15>
            <input type=hidden name=operation value=1>
            </form></td>
        </tr>
        <tr>
        <th scope='row'>3. STOCK</th>
        <td>
            <form action=TOP17.php method=post>
            <button class='mybutton' type=submit> TOP 17 </button>
            <input type=hidden name=note value=TOP17>
            <input type=hidden name=operation value=1>
            </form></td>
        <td>
            <form action=TOP18.php method=post>
            <button class='mybutton' type=submit> TOP 18 </button>
            <input type=hidden name=note value=TOP18>
            <input type=hidden name=operation value=1>
            </form></td>
        <td>
            <form action=TOP19.php method=post>
            <button class='mybutton' type=submit> TOP 19 </button>
            <input type=hidden name=note value=TOP19>
            <input type=hidden name=operation value=1>
            </form></td>
        </tr>
        </table>

        <h2> Bitte nur 1x Paket einlegen. Keine Haftung und keine Gewähr für Verlust, Diebstahl, Beschädigung, korrekte
        Zustellung... Bitte Zustimmung vom Empfänger vor Hinterlegung einholen.</h2>
";

```



```

}
?>

<br><br>

<div id="adminpanel">
<p align="right"><a href="start.html"></a></p><br>
<p>This page will be closed in <span id="counter">180</span> second(s).</p>
<script type="text/javascript">
function countdown() {
    var i = document.getElementById('counter');
    if (parseInt(i.innerHTML)<=0) {
        location.href = 'start.html';
    }
    i.innerHTML = parseInt(i.innerHTML)-1;
}
setInterval(function(){ countdown(); },1000);
</script>
<table id="table">
    <tr>
        <td>@dmin
<input type="password" id="password" onKeyDown="if (event.keyCode == 13) document.getElementById('button').click()" />
<!-- IMPORTANT! this part is so if you click enter, it works. -->

        <input id="button" type="button" value="x" onClick="if (document.getElementById('password').value ==
'adminpassword') {
document.getElementById('table').classList.toggle('show'); document.getElementById('passw').style.display='none'; }
else { alert('Invalid Password!'); password.setSelectionRange(0, password.value.length); } " /></td>
        <td id="HIDDENIDIV" > <form action="box2.php" method="post">

            Add a Record: <button type="submit"> Timestamp </button>
Notes: <input type="text" name="note">
<input type=hidden name=operation value=1>
<p></p>
</form>

            <form action="box2.php" method="post">
<input type=hidden name=operation value=2>
Clear Data: <button type="submit">Erase</button>
<p></p>
</form>

            <form action="box2.php" method="post">
<input type=hidden name=operation value=3>
Records in the Database: <button type="submit">Show</button>
<p></p>
</form>

            <form action="box2.php"
method="post">
<input type=hidden name=operation value=5>
<p>Backup the database: <button type="submit">Backup</button>
</p>
</form></td>
    </tr>
</table>

</div>

</body>
</html>

```

TOPXX.PHP

UI for tenant to receive parcel. It displays a PIN code form. PIN for sample code is 1234

```

<?php include 'box2.php';?>

<div id='pin'>
    <div class='dots'>
        <div class='dot'></div>
        <div class='dot'></div>
        <div class='dot'></div>
        <div class='dot'></div>
    </div>
    <p>Enter the password</p>
    <div class='numbers'>
        <div class='number'>1</div>
        <div class='number'>2</div>
        <div class='number'>3</div>
        <div class='number'>4</div>
        <div class='number'>5</div>
        <div class='number'>6</div>
        <div class='number'>7</div>
        <div class='number'>8</div>
        <div class='number'>9</div>
    </div>
</div>

<script> (function () {

```

```

var input = '', correct = '1234';
var dots = document.querySelectorAll('.dot'), numbers = document.querySelectorAll('.number');
dots = Array.prototype.slice.call(dots);
numbers = Array.prototype.slice.call(numbers);
numbers.forEach(function (number, index) {
  number.addEventListener('click', function () {
    number.className += ' grow';
    input += index + 1;
    dots[input.length - 1].className += ' active';
    if (input.length >= 4) {
      if (input !== correct) {
        dots.forEach(function (dot, index) {
          dot.className += ' wrong';
        });
        document.body.className += ' wrong';
      } else {
        dots.forEach(function (dot, index) {
          dot.className += ' correct';
        });
        document.body.className += ' correct';
      }
    }
  });
});

document.ready(window.setTimeout(location.href = "./box2.php?operation=2",10));
  }
  setTimeout(function () {
    dots.forEach(function (dot, index) {
      dot.className = 'dot';
    });
    input = '';
  }, 900);
  setTimeout(function () {
    document.body.className = '';
  }, 1000);
}
setTimeout(function () {
  number.className = 'number';
}, 1000);
});
})(); </script>

</body>
</html>

```

That's all. Total code including some including some images was below 200kb so pretty much lightweight. It should also run very smoothly on a ZERO Pi, which is a great alternative if you want to save some space and downsize the power input. By building this project it saved us easily several thousands of euros for a commercial product. The result is very professional and high quality if built with care. Also the parcel locker is service provider independent and open to all courier services. We didn't care about the monopolism policy of the available parcel boxes such as from Austrian Post or Burg Wächter eBoxx.

Since payment solution (for Pickup) is routed to an external server it is pretty much future proof. You simply need a POS website and can use any credit card processor or even accept payments with cryptocurrencies.

Finally in our case the MJPEG stream of the RPi camera is recorded 24/7. Integration of a code which snaps a picture on unlock can be done but not needed in our case.

Any improvements to the code are happily received at roottext32@gmail.com