

# Attendance system using Raspberry Pi and NFC Tag reader

por [Yim](#)

 (/id/Attendance-system-using-Raspberry-Pi-and-NFC-Tag-r/?lang=es)

4 Pasos

► (/id/Attendance-system-using-Raspberry-Pi-and-NFC-Tag-r/?lang=es)

▼ (/id/Attendance-system-using-Raspberry-Pi-and-NFC-Tag-r/?lang=es)



with my new toy raspberry pi i've done several testing projects from instructables, adafruit etc. But after a while it became boring and I've started looking for something actually useful for me. My previous playground was my new phone several weeks ago which came with new feature NFC reading. Thought and thought together gave me idea for attendance system for our small (family) company using NFC tags. The additional kick was my interest in those systems before, but discovering the cheapest are for around \$750 I decided it is too much for such small company as 6 employees.

As I don't have any experience with "mature" attendance systems, I've decided only to implement basic features. These consist of logging incoming people, outgoing people, start and end of a break and deleting last inserted action (in case of mistype during logging).


All these actions are logged into local MySQL database from where I can display it and manipulate with my front-end application. Because the SD card is not such safe data medium, especially when loosing power unexpectedly, I'm uploading all data daily to my local server, where I keep backup in case of corruption of the SD card.


During normal operation of the logging station, display shows current date and time and calls for action selection. When you choose appropriate action on the keyboard, display shows selected action and calls for attaching the TAG to the reader. Also the LED under display also turns on.

When the TAG is read, the LED turns off and beep signal comes out the speaker. For a brief moment display shows action and name of the owner of the





## Sobre este Instructable

 **36.205** vistas

 **75** favoritos

**Publicado el:**  
Dic 31, 2013

**Licencia:**  




**Yim**  
(/es/member/Yim/)  
([ver perfil](#))

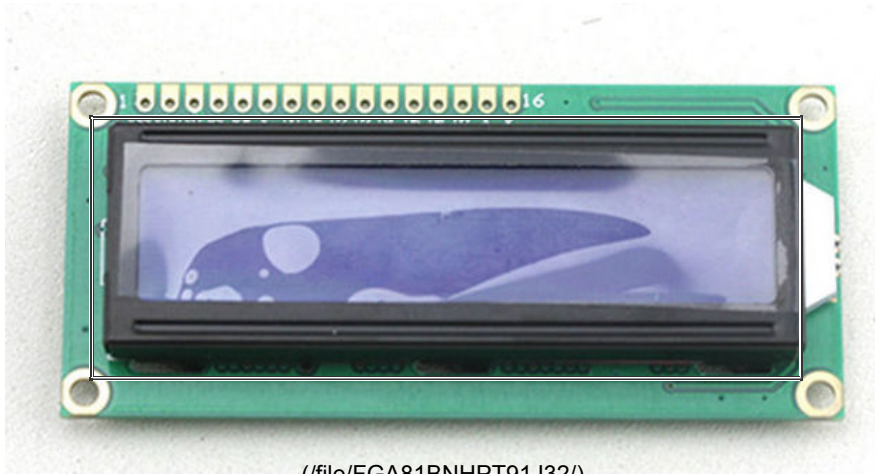
[Seguir](#)

card. Then everything returns to the default state waiting for another entrance.

For foot notice, this whole project including source codes is licensed under Beer-ware licence as follows:

Jakub Dvorak wrote this file. As long as you retain this notice you can do whatever you want with this stuff. If we meet some day, and you think this stuff is worth it, you can buy me a beer in return.

## Paso 1: Materials list



Everything you need for this project can be bought on Ebay or in many Chinese e-shops like dealextreme etc.

For this project you'll need following items:

1. Raspberry Pi model B (<http://www.adafruit.com/products/998>)
2. 16x2 display ([http://www.ebay.com/sch/i.html?\\_trksid=p2050601.m570.l1313&\\_nkw=16x2+display&\\_sacat=0&\\_from=R40](http://www.ebay.com/sch/i.html?_trksid=p2050601.m570.l1313&_nkw=16x2+display&_sacat=0&_from=R40))
3. Mifare RFID Reader ([http://www.ebay.com/sch/i.html?\\_trksid=p2050601.m570.l1313&\\_nkw=rfid+rc522&\\_sacat=0&\\_from=R40](http://www.ebay.com/sch/i.html?_trksid=p2050601.m570.l1313&_nkw=rfid+rc522&_sacat=0&_from=R40))
4. USB numeric keypad ([http://www.ebay.com/sch/i.html?\\_trksid=p2050601.m570.l1313.TR0.TRC0&\\_nkw=usb+numeric+keypad&\\_sacat=0&\\_from=R40](http://www.ebay.com/sch/i.html?_trksid=p2050601.m570.l1313.TR0.TRC0&_nkw=usb+numeric+keypad&_sacat=0&_from=R40))
5. LED diode holder ([http://www.ebay.com/sch/i.html?\\_trksid=p2050601.m570.l1313.TR0.TRC0.XLED+Lamp+Holder+for+Light-](http://www.ebay.com/sch/i.html?_trksid=p2050601.m570.l1313.TR0.TRC0.XLED+Lamp+Holder+for+Light-)

6. Red LED diode ([http://www.ebay.com/sch/i.html?\\_trksid=p2050601.m570.l1313.TR0.TRC0.Xred+LED+led+emitting+diode&\\_nkw=red+LED+led+emitting+diode&\\_sacat=0&\\_from=R40](http://www.ebay.com/sch/i.html?_trksid=p2050601.m570.l1313.TR0.TRC0.Xred+LED+led+emitting+diode&_nkw=red+LED+led+emitting+diode&_sacat=0&_from=R40))
7. Active 3V buzzer ([http://www.ebay.com/sch/i.html?\\_trksid=p2047675.m570.l1313.TR0.TRC0.Xactive+buzzer+3v&\\_nkw=active+buzzer+3v&\\_sacat=0&\\_from=R40](http://www.ebay.com/sch/i.html?_trksid=p2047675.m570.l1313.TR0.TRC0.Xactive+buzzer+3v&_nkw=active+buzzer+3v&_sacat=0&_from=R40))
8. Small breadboard and various F-F and M-F jumper cables ([http://www.ebay.com/sch/i.html?\\_odkw=breadboard+cable&\\_osacat=0&\\_from=R40&\\_trksid=p2045573.m570.l1313.TR11.TRC1.A0.Xbreadboard&\\_nkw=breadboard+cable&\\_sacat=0&\\_from=R40](http://www.ebay.com/sch/i.html?_odkw=breadboard+cable&_osacat=0&_from=R40&_trksid=p2045573.m570.l1313.TR11.TRC1.A0.Xbreadboard&_nkw=breadboard+cable&_sacat=0&_from=R40))
9. Box ([http://www.ebay.com/sch/i.html?\\_odkw=abs+box&\\_osacat=0&\\_from=R40&\\_trksid=p2045573.m570.l1313.TR5.TRC2.A0.Xabs+plastic+box&\\_nkw=abs+plastic+box&\\_sacat=0&\\_from=R40](http://www.ebay.com/sch/i.html?_odkw=abs+box&_osacat=0&_from=R40&_trksid=p2045573.m570.l1313.TR5.TRC2.A0.Xabs+plastic+box&_nkw=abs+plastic+box&_sacat=0&_from=R40))
10. Various screws and/or glue gun, LED resistor, display 10k potentiometer

You can use any display, but I used HD44780 (the same display as found on [adafruit](#)), because it has well documented usage and uses only few pins from Raspberry.

Any numeric keypad can be used. Those with high buttons might be better due to thickness of the box. For installation just remove any extra buttons with small knife or screwdriver.

The same is with the buzzer. I felt such a device should have it's own voice for letting people know it is working.

You'll also need few screws and/or glue gun for attaching all the pieces together. For LED you'll need small resistor. For my red LED I've used 68 ohm one which should limit current to around 19 mA. Depending on your LED you can calculate value for the resistor for example on this address

## Paso 2: Wiring it together





On the picture from [mazing \(http://mazing.org/moments/\)](http://mazing.org/moments/) above you can see connections between each element. You can find the fzz file as download link down the page.

For troubleshooting of display connection you can refer to nice manual on adafruit website (<http://learn.adafruit.com/drive-a-16x2-lcd-directly-with-a-raspberry-pi>).

Connection is pretty straightforward so there shouldn't be any trouble.

I've cut holes in the lid of the box for display, LED and keypad, NFC reader remains under the lid for safety and it has strong enough signal to read tags through the plastic.

I used glue gun for the alignments in the box and it holds every component firm enough. And after testing even the wires are glued to the connectors to prevent accidental disconnecting during operation.

Also don't forget to drill holes for power and network cable (or just one hole, if you use WiFi dongle or PoE).

As final touch, mark spot on the outside of the lid where NFC reader is for easier reading.

## Paso 3: The software

The software for this project is mixed from several tutorials and HOWTOs so there might be some leftovers from that, but it works for the time being and I might will fix that later (AKA never ;) )

First of all you need default Raspbian image installed on your raspberry (LINK (<http://www.raspberrypi.org/downloads>) and HowTo ([http://elinux.org/RPi\\_Easy\\_SD\\_Card\\_Setup](http://elinux.org/RPi_Easy_SD_Card_Setup))).

After that we need few adjustments to enable correct function:

First of all we install files needed by the NFC reader as follows on previously mentioned page (<http://fuenteabierta.teubi.co/2013/07/utilizando-el-lector-nfc-rc522-en-la.html>) (<http://fuenteabierta.teubi.co/2013/07/utilizando-el-lector-nfc-rc522-en-la.html>))

1. Enable SPI device by editing file `/etc/modprobe.d/raspi-blacklist.conf` and comment the line **blacklist spi-bcm2708** so it will read as follows:

```
# blacklist spi and i2c by default (many users don't need them)
# blacklist spi-bcm2708
blacklist i2c-bcm2708
```

2. Then download SPI-Py package to enable SPI communication for Python

```
git clone https://github.com/lthiery/SPI-Py (https://github.com/lthiery/SPI-Py)
```

and install it via

```
sudo python setup.py install
```

We will also need the MFRC522 package from <https://github.com/mxgxw/MFRC522-python> (<https://github.com/mxgxw/MFRC522-python>) but I had to edit it a little bit, so it's already in the source folder of this project.

The last think to get it work is installing GPIO for Python (if you wish, update the number in package version to the newest one):

```
wget https://pypi.python.org/packages/source/R/ (https://pypi.python.org/packages/source/R/)RPi.GPIO-0.5.4.tar.gz
tar xzf RPi.GPIO-0.5.4.tar.gz
cd RPi.GPIO-0.5.4
sudo python setup.py install
```

After that we can download the source code for this project

```
git clone https://github.com/Yimbo/attendance (https://github.com/Yimbo/attendance)
```

Create table in your MySQL Database and upload structure to that database from file `mysql.sql`.

Next update file `mysql.py` according to your settings (server, username, password and database name).

After this step you should be able to run the program:

```
sudo python attendance.py
```

## Paso 4: Automatization

Now you should have fully functioning system for reading NFC tags and storing information about them in MySQL database.

For daily usage your should be ready to accept power failures, data errors etc.

To avoid power failures you can use UPS, but your raspberry might still get rebooted from time to time. Also you want to use the integrated keyboard instead of the keyboard from console where you run program from.

First of all change your attendance file attributes so it can run without external calling python:

```
chmod +x attendance.py
```

In next step enable autologin on TTY1. This will automatically login your first console after reboot as root.

First install mingetty:

```
apt-get install mingetty
```

Then edit the file `/etc/inittab` and find line



```
1:2345:respawn:/sbin/getty --noclear 38400 tty1
```

and update it to:

```
1:2345:respawn:/sbin/minigetty --autologin root --noclear tty1
```

Then you need NumLock activated everytime system logs onto console tty1, so create file leds.sh with following content (note it probably won't affect the actual LEDs on the keypad, but it will turn Numlock ON anyway):

```
setleds -D +num < /dev/tty1
```

and change its permissions to executable:

```
chmod +x leds.sh
```

Then you want to create watchdog for your python script so even if it crashes, this watchdog will start it again. Go to your home directory and create file runattendance.sh with following text:

```
while [ 1=1 ];
do
    if [ ! "$(pgrep attendance.py)" ];
    then
        /root/leds.sh && cd /root/Attendance/ && /root/Attendance/at
tendance.py
    fi
    sleep 10
done
```

and change it's permissions to be executable the same way as before.

After that go to your root home folder (/root/) and update file .bash\_profile and add following line:

```
[[ $(tty) == '/dev/tty1' ]] && /root/runattendance.sh
```

This will run runattendance.sh every time system logs into first console (which is now automatically after booting).

If you finished all those steps, you can now reboot your raspberry and after successful boot, display should light and attendance system should work.






Tenemos una política de comentarios respetuosos.  
Por favor, se positivo/a y constructivo/a.

 I Made it!

 editar imagen

Publicar Comentario



**jjjdz (/member/jjjdz)** (/member/)13 days ago

hmm.. when running attendance.py

File "/home/pi/RPi.GPIO-0.5.7/attendance/MFRC522.py", line 11, in  
<module>

import spi

ImportError: No module named spi

install RPi.GPIO-0.5.7.tar.gz

Responder



**SébastienJ** (/member/SébastienJ) (/member/)

2 months ago

Other skill with PNEV512R :

<http://domotique-info.fr/2014/08/passerelle-domotique-nfc/>

Responder



**asadkundi** (/member/asadkundi) (/member/)

6 months ago

Quotation Required of Tags Attendance system.

Responder



**zacy5000** (/member/zacy5000) (/member/)

7 months ago

This is great! Exactly what I have been looking for! Thanks so much. Im going to be getting my NFC reader from hong kong soon. I got it for £2.50 so its going to be fun. Thanks.

Responder



**audreyobscura** (/member/audreyobscura) (/member/)

10 months ago

Good idea! I am just starting to noodle around with NFC, nice project, and thank you for sharing.

Responder



Hagamos:

Español

