

UNIVERSITAT POLITÈCNICA DE VALÈNCIA



ESCOLA POLITÈCNICA SUPERIOR D'ALCOI

PROJECTE FINAL DE GRAU

SOFTWARE DE  
VIDEOVIGILÀNCIA PER A  
RASPBERRY PI, AMB ACCÉS  
SEGUR PER A CONTROL  
D'UN ACTUADOR

DIRECCIÓ: D. JOSEP SALVADOR BLANES I DOMÈNECH

AUTORA: LAURA QUILIS VAYÀ

TITULACIÓ: GRAU EN INGENYERIA INFORMÀTICA

JULIOL 2015

# Resum

Aquest projecte naix de la idea d'ajudar a la gent amb poc recursos que disposa d'una empresa o d'una casa la qual vol tindre protegida amb un sistema de seguretat. Es pretén fer un dispositiu barat, com és la Raspberry Pi, i fàcil d'implantar i instal·lar. A més a més, per a controlar aquest sistema de seguretat, s'ha fet una pàgina web, localitzada al servidor creat a la Raspberry Pi.

El sistema de seguretat és molt fàcil d'explicar. Una vegada engegada la Raspberry Pi, sols s'ha d'executar el programa principal. Quant aquest programa esta executant-se, espera que es polse un botó. Una vegada s'ha polsat el botó, s'envia un correu electrònic a la direcció donada i comença a streamejar la càmera. El client, al rebre el correu, entra a la pàgina web i des de allí pot veure el stream i, a més a més, pot donar ordres, com tancar i obrir la porta, així com parar i començar el stream.

En lloc d'una porta, per al projecte s'han utilitzat dos LEDs, un de color verd per a indicar quant va a obrir-se la porta, i un de color roig per a indicar que va a tancar-se la porta.

# Índex general

<a href="#">Resum</a>	2
1. <a href="#">Avantprojecte</a>	5
1.1. <a href="#">La Raspberry Pi</a>	5
1.2. <a href="#">Objectius del projecte</a>	6
1.3. <a href="#">Estudi de mercat</a>	6
1.3.1. <a href="#">Videovigilància de Prosegur</a>	6
1.3.2. <a href="#">Videovigilància de Securitas Direct</a>	7
1.3.3. <a href="#">Videovigilància de Sony</a>	8
1.3.4. <a href="#">Conclusions</a>	8
1.4. <a href="#">Metodologia i recursos necessaris</a>	8
1.5. <a href="#">Justificació</a>	9
1.5.1. <a href="#">Dispositiu</a>	10
1.5.2. <a href="#">Llenguatges de programació</a>	11
1.5.3. <a href="#">Servidor web</a>	13
1.6. <a href="#">Fiabilitat</a>	13
1.7. <a href="#">Política d'usabilitat</a>	14
2. <a href="#">Projecte</a>	15
2.1. <a href="#">Descripció</a>	15
2.1.1. <a href="#">Prestacions</a>	15
2.1.2. <a href="#">Pàgina web</a>	16
2.1.3. <a href="#">Sistema de streaming</a>	17
2.1.4. <a href="#">Sistema d'avís</a>	18
2.2. <a href="#">Instal·lació</a>	19
2.3. <a href="#">Compatibilitat</a>	19
2.4. <a href="#">Facilitat d'ús</a>	19
3. <a href="#">Conclusions i futures millores</a>	21
3.1. <a href="#">Conclusions</a>	21
3.2. <a href="#">Futures millores</a>	23
4. <a href="#">Pressupost</a>	24
5. <a href="#">Muntatge i plànols</a>	25

6.	<a href="#"><u>Annexos</u></a>	.35
6.1.	<a href="#"><u>Guia ràpida d'ús</u></a>	.35
6.2.	<a href="#"><u>Codi font</u></a>	.36
6.3.	<a href="#"><u>Informació tècnica</u></a>	.58
6.4.	<a href="#"><u>Contingut distribuït</u></a>	.63

# 1. AVANTPROJECTE

## 1.1. La Raspberry Pi

La Raspberry Pi es una placa computadora de baix cost desenvolupada al Regne Unit per la Fundació Raspberry Pi, amb el objectiu de estimular l'aprenentatge de ciències de la computació a les escoles.

La Fundació Raspberry Pi és una organització sense ànim de lucre que dona els seus primers passos com a fundació l'any 2008, però que en realitat portava ja dos anys gestant-se els primers models de Raspberry Pi. En 2011 es va desenvolupar la Raspberry Pi, però no va ser fins a 2012 que va començar a fabricar-se. La fundació rep suport del laboratori d'informàtica de la Universitat de Cambridge.

En aquest projecte s'utilitza el model B+, que té unes dimensions de 84x56 mm, un SoC (System on a chip) BCM 2835 amb 512 MB de RAM, una CPU ARM 1176JZF-S que funciona a 700 MHz, una GPU Broadcom VideoCore IV, 4 connexions USB 2.0, 1 eixida àudio mini jack de 3.5mm, 1 eixida de àudio/vídeo HDMI, 1 microUSB que serveix per a connectar la font d'alimentació de 5V/600 mA (3.5 W), un connector RJ45 10/100 Ethernet i un slot microSD per a targetes. També consta de 40 pin GPIO, un CSI per a connectar la càmera de Raspberry Pi i un DSI (connector sèrie), a més de suportar tecnologia WiFi mitjançant un adaptador.

La Raspberry Pi no porta cap Sistema Operatiu instal·lat. Per aquest motiu necessitem una targeta microSD per a que funcione com a emmagatzematge. Es poden descarregar diferents SO des de la pàgina oficial de Raspberry Pi, així com Noobs, una aplicació que facilita la instal·lació de diverses distribucions Linux. Aquesta aplicació et dona l'opció de instal·lar Raspbian, Arch Linux, RaspBMC, Pidora i OpenELEC.

En aquest projecte s'ha instal·lat Raspbian, què està basat en Debian Wheezy. Aquest SO està optimitzat per a càlculs en coma flotant per hardware, el que permet donar més rendiment.

## 1.2. Objectius del projecte

Es pretén dissenyar un sistema de videovigilància amb control remot que es pugui instal·lar fàcilment tant en cases com en empreses i així facilitar un poc la vida amb la domòtica. Mitjançant una simple pàgina web pots obrir i tancar la porta, així com veure qui ha pres el botó.

D'aquesta forma podem controlar el accés de forma remota des de qualsevol part on ens trobem, per exemple, podem obrir despaxos o zones privades d'una empresa sense la necessitat d'estar presents, així com obrir la porta de casa remotament, sense la necessitat d'anar fins a ella.

## 1.3. Estudi del mercat

S'ha realitzat un estudi de mercat amb relació al projecte que es vol dur a terme. Per això, s'han buscat diferents sistemes de videovigilància que també donen l'opció de interactuar mitjançant Internet. A continuació es mostren alguns resultats trobats.

### 1.3.1. Videovigilància de *Prosegur*

Aquesta empresa ofereix solucions de videovigilància als negocis i comerços que vulguin controlar el que ocorre al seu negoci d'una forma directa i constant.

Ofereixen diferents solucions de videovigilància:

- Circuit tancat de televisió
- Circuit obert de televisió
- Videoseguretat IP

A més, tenen les següents característiques:

- Sistemes equipats amb la última tecnologia
- Gravació de vídeo en alta qualitat
- Alt poder de dissuasió
- Solucions adaptades a cada negoci

- Vigilància continua, les 24 hores, els 365 dies de l'any
- Compatibilitat amb altres sistemes de seguretat

Tot açò te un preu de 327,58€ de instal·lació, a més de una cota fixa de 53,97€ al mes, durant 4 anys, com a mínim, el que fa un total de 2918,14€, a data 25 de Juny de 2015.

### 1.3.2. Videovigilància de *Securitas Direct*

Aquesta empresa ofereix seguretat de videovigilància per a casa i per a la empresa.

Per a la seguretat de casa i de l'empresa ofereix un sistema antirobatori, que es caracteritza per:

- La Central Receptora d'Alarmes de Securitas Direct és capaç de verificar el salt d'una alarma i decidir com actuar en menys de 33 segons.
- Una vegada verificat l'intent de robatori o situació d'emergència, envien un vigilant de Securitas i avisen a la Policia.
- El comandament Verisure amb botó SOS, permet sol·licitar ajuda immediata: Policia, Bombers, Serveis d'emergències.
- Disposen d'un servei tècnic les 24 hores del dia.
- Els dispositius de l'Alarma Verisure compten amb garantia per a tota la vida perquè la seva alarma estigui sempre en perfectes condicions.
- L'aplicació My Verisure de Securitas Direct et permet prendre el control de la teva seguretat. Pots activar o desactivar l'alarma des de qualsevol part del món. També obrir o tancar portes de forma remota o controlar qui entra o surt.

I consta amb la següent tecnologia:

- Alarma amb detector de inhibició
- Tauler de control
- Detectores de moviment
- Lector de claus, que inclou 6 claus
- Detector de obertures
- Sirena d'alta potencia
- Cartells dissuasius
- Comandament a distancia
- Detector perimetral
- Detector de fum
- Detector d'inundació
- Polsador SOS

- Càmera de videovigilància

Tot açò te un preu de 482,79€, a més de una cota de 56,97€ al mes durant 36 mesos per a tindre servici tècnic, és a dir, un total de 2533,71€, a data 24 de Juny de 2015.

### 1.3.3. Videovigilància de *Sony*

Aquesta empresa ofereix als seus clients videovigilància tant per a empreses, com per a educació.

Per a aquestes dues solucions t'ofereixen una càmera inalàmbrica de 720p/30 fps en red amb il·luminadors LED de llum blanca i tecnologia IPELA ENGINE EX™, sèrie C.

Les característiques d'aquest servei amb aquesta càmera son les següents:

- La càmera grava amb una resolució de 1280x720.
- Compta amb la tecnologia IPELA ENGINE EX.
- Té un detector de infrarojos que pot detectar gent per la seva temperatura.
- Sensor que pot activar una alerta per veu, iniciar una gravació o enviar un correu electrònic.
- Compta amb un estabilitzador.
- Red inalàmbrica
- Gravació automàtica
- Compatible amb ONVIF

A més, aquesta empresa t'ofereix càmeres IP, càmeres analògiques, codificadors, gravadores, software de monitorització i càmeres híbrides.

Tot açò te un preu de 278,5€ de instal·lació, a més de una cota fixa de 45,7€ al mes, durant 3 anys com a mínim, és a dir, un total de 1923,7€, a data de 25 de Juny del 2015.

### 1.3.4. Conclusions

Tal i com s'ha vist anteriorment, en aquest projecte es vol fer un sistema de videovigilància per a poder controlar una porta. Si comparem aquest projecte amb les diferents ofertes que hi ha al mercat, no hi ha cap que es centre solament amb el control remot de una porta. Els altres



productes, al incloure altres serveis, son molt més cars que el que costaria aquest projecte.

## 1.4. Metodologia i recursos necessaris

En aquest projecte, podem dir, què és comprenen dos camps, la electrònica i la programació. A continuació, s'explica la metodologia utilitzada en cada camp.

En el camp de l'electrònica s'ha utilitzat una placa de proves per a poder connectar tots el components que s'han fet servir. Els components utilitzats són un polsador, per a simular un timbre; dos LEDs, un verd per a senyalitzar que la porta esta oberta i un roig per a senyalitzar que la porta esta tancada; dos resistències, una per a cada LED; i cables per a unir aquest components als GPIO de la Raspberry Pi.

En el camp de la programació s'han utilitzat tres llenguatges diferents de programació. S'ha utilitzat Python per a fer els scripts de la Raspberry Pi; també s'ha utilitzat Shell, amb comandaments basics de UNIX per al correcte funcionament de l'aplicació; per últim, s'ha utilitzat PHP per a fer la programació web.

Aquest tres llenguatges s'han incorporat també uns dins d'altres, per exemple, s'ha integrat un .sh dins de scripts de Python; i scripts de Python s'han integrat dins e codi amb PHP, es a dir, els tres llenguatges de programació són compatibles entre ells.

## 1.5. Justificació

En aquest apartat es justifiquen les decisions més rellevants que han afectat a la realització del projecte.

### 1.5.1. Dispositiu

Hi havia varies opcions de dispositius que podiem utilitzar per a realitzar el projecte. A continuació, es citen algunes d'elles, deixant a part la Raspberry Pi, ja explicada anteriorment.

- **Cubieboard 2:** porta un processador ARM DualCore Cortex-A7 a 1 Ghz amb una GPU Mali-400MP2, també inclou 1GB de memòria RAM DDR3 a 960 Mhz i sortida HDMI per reproduir vídeo a 1080p. Porta una connexió de xarxa Ethernet a 100 MB, dos connectors USB, un connector SATA i un slot per a una targeta MicroSD. A més de funcionar amb Android també pot funcionar amb Ubuntu o amb qualsevol altra distribució que suport processadors amb arquitectura ARM.
- **Hackberry A10 Dev Board:** Porta un processador ARM Cortex-A8 a 1 Ghz amb una GPU Mali-400MP, porta 512MB o 1GB de memòria RAM, encara que 100MB es reserven per a la targeta gràfica i també porta eixida HDMI. Porta una targeta de xarxa Ethernet a 100 MB amb xip Realtek, 2 connectors USB i un slot per a una targeta SD, encara que també porta un emmagatzematge intern de 4GB on va instal·lat Android 4.0 ICS.
- **BeagleBone Black:** Porta un processador ARM Cortex-A8 a 1Ghz amb 512MB de memòria RAM DDR3 i 2GB d'emmagatzematge intern. Inclou connector HDMI, una connexió USB, 1 connector Ethernet i un USB addicional per a alimentació. El BeagleBone Black pot executar Android o Ubuntu, fins i tot algunes distribucions o sistemes operatius més.
- **A13-OLinuX-Wifi:** Integra un xip capaç de connectar-se per Wifi (o distribuir Wifi) a través d'un xip RTL8188CU. També porta un processador ARM Cortex-A8 a 1Ghz amb una GPU Mali400 i 512 MB de memòria RAM, a més porta 4GB d'emmagatzematge intern. En aquest cas aquesta placa porta 3 connectors USB + 1 connector addicional per al Wifi, també porta connexió VGA per a connectar un monitor i connectors Jack de 3,5 per a altaveus i micròfon. Ve preparat per a executar Android, encara que també pot executar algunes distribucions Linux.
- **ODROID-O3 Community Edition:** El maquinari d'aquesta placa és un equivalent al maquinari que podem trobar en un Samsung Galaxy S3. Porta un processador QuadCore Exynos 4412 a 1,7 Ghz i 2GB de memòria RAM DDR3, a més porta connector RJ45 Ethernet a 100MB i 3 connectors USB 2.0. Ve de sèrie amb dos sistemes operatius, Xubuntu 13.10 i Android 4.X, evidentment també porta connexió HDMI i un slot per a una targeta MicroSD.

Fent una comparació entre tots els dispositius, es va decidir utilitzar la Raspberry Pi ja que cobreix totes les necessitats del projecte i s'acobla al presupost, a més de ser la més utilitzada pels usuaris amb diferència, amb la qual cosa pots obtenir moltíssima més informació i ajuda que de qualsevol altre dispositiu.

## 1.5.2. Llenguatges de programació

Aquest són alguns llenguatges que es van tindre en compte a l'hora de començar a desenvolupar el projecte:

- **C#:** és un llenguatge creat per Microsoft, i que amb la Raspberry Pi pot ser utilitzat tant amb el mode de compilació com amb el mode de interacció. Aquest llenguatge deriva de C/C++.
- **Forth:** és un llenguatge de programació de computadores procedimental, estructurat, imperatiu, reflexiu, basat en pila i sense comprovació de tipus. Forth ofereix tant l'execució interactiva de comandos (fent-ho convenient com shell per als sistemes que manquen d'un sistema operatiu més formal) com la capacitat de compilar seqüències de comandos per a l'execució posterior. Algunes implementacions del Forth (usualment les versions primerenques o les escrites per a ser extremadament portable) compilen codi enfilat (threaded code), però moltes implementacions d'avui generen codi de màquina optimitzat com altres compiladors de llenguatges.
- **Java:** És un llenguatge senzill d'aprendre. La seua sintaxi és la de C++ "simplificada". Java és un llenguatge orientat a objectes, encara que no dels denominats purs; en Java tots els tipus, a excepció dels tipus fonamentals de variables (int, char, long...) són classes. No obstant açò, en els llenguatges orientats a objectes purs fins i tot aquests tipus fonamentals són classes, per exemple en Smalltalk. El codi generat pel compilador Java és independent de l'arquitectura: podria executar-se en un entorn UNIX, Mac o Windows.
- **Python:** Python és un llenguatge de scripting independent de plataforma i orientat a objectes, preparat per a realitzar qualsevol tipus de programa, des d'aplicacions Windows a servidors de xarxa o fins i tot, pàgines web. És un llenguatge interpretat, la qual cosa significa que no es necessita compilar el codi font per a poder executar-ho, la qual cosa ofereix avantatges com la rapidesa de desenvolupament.

- **Go:** és un llenguatge compilat amb algunes característiques peculiars. És un llenguatge concurrent que suporta canals de comunicació basats en el llenguatge CSP. La concurrència en Go és diferent als criteris de programació basats en bloquejos com pthreads. La senzillesa és la característica principal. Admet la tipificació dinàmica de dades, també coneguda com duck Typing, present en multitud de llenguatges dinàmics com, per exemple, Javascript, Ruby o Python. Un struct pot implementar una interfície de forma automàtica, la qual cosa és una característica potent i nova. La comunicació entre goroutines es realitza a través d'una característica del llenguatge anomenada canals.
- **HTML:** És un estàndard que serveix de referència per a l'elaboració de pàgines web en les seues diferents versions, defineix una estructura bàsica i un codi (denominat codi HTML) per a la definició de contingut d'una pàgina web, com a text, imatges, videos, entre uns altres. És un estàndard a càrrec de la W3C. El llenguatge HTML basa la seua filosofia de desenvolupament en la referenciació.
- **Javascript:** És un llenguatge de programació que s'utilitza principalment per a crear pàgines web dinàmiques. És un llenguatge de programació interpretat, per la qual cosa no és necessari compilar els programes per a executar-los, a més a més, no requereix de compilació ja que el llenguatge funciona del costat del client, els navegadors són els encarregats d'interpretar aquests codis. Amb Javascript podem crear diferents efectes i interactuar amb els nostres usuaris.
- **PHP:** És un llenguatge de programació del costat del servidor gratuït i independent de plataforma, ràpid, amb una gran llibreria de funcions i molta documentació. PHP s'escriu dins del codi HTML, la qual cosa ho fa realment fàcil d'utilitzar, igual que ocorre amb el popular ASP de Microsoft, però amb alguns avantatges com la seua gratuïtat, independència de plataforma, rapidesa i seguretat. Gaudeix de compatibilitat amb les bases de dades més comunes, com MySQL, mSQL, Oracle, Informix, i ODBC, per exemple. Inclou funcions per a l'enviament de correu electrònic, upload d'arxius, crear dinàmicament en el servidor imatges en format GIF, i una llista interminable d'utilitats addicionals.

Amb tota aquesta informació, es va decidir treballar per una part amb Python per als scripts en la Raspberry Pi, i amb PHP, Javascript i HTML per a crear la pàgina web. Els motius en tots dos casos són: la senzillesa i el fàcil aprenentatge que tenen aquest llenguatges de programació. A més, són els llenguatges més utilitzats per a la programació amb Raspberry Pi i per a la programació web, respectivament, i això fa que tingues més opcions d'aprenentatge i ajuda que en altres llenguatges de programació.

### 1.5.3. Servidor web

Per a fer el servidor que allotge la pàgina web s'ha fet servir el software lliure Apache, què és un servidor web HTTP de codi obert, per a plataformes UNIX, Windows, Macintosh, i altres que implementen el protocol HTTP/1.1 i la noció de lloc virtual.

S'ha triat aquesta opció per dos motius principals: és tracta de software lliure i funciona en plataformes UNIX. A més a més, hi ha facilitat per a instal·lar-lo a la Raspberry Pi i hi ha molta ajuda ja que és un dels software més utilitzats pels usuaris.

## 1.6. Fiabilitat

El producte ha sigut provat i s'han realitzat algunes proves per a demostrar què és un producte segur i fiable.

Una de les proves realitzades ha sigut mantindre el dispositiu encès durant 72 hores per a comprovar que no hi ha sobrecaentament.

També s'ha realitzat una prova d'estrès a la pàgina web. S'han realitzat varies connexions a la pàgina web simultàniament per a comprovar el aguant del servidor Apache, i comprovar així que la pàgina web no 'cau' i el servidor aguanta on-line amb un gran nombre de visites. Encara que la prova a resultat satisfactòria, no es massa rellevant, ja que la idea principal es que sols entre a la pàgina una persona, és a dir, sols es produeix una connexió simultàniament.

Per últim, s'han realitzat proves de lluminositat, és a dir, s'ha provat el streaming en diferents tipus de llum (llum natural, llum al·lògena, etc). El resultat es que sempre que hi haja un mínim de llum, es pot veure perfectament el streaming.

## 1.7. Política d'usabilitat

La usabilitat es refereix a la facilitat amb la qual les persones poden utilitzar una ferramenta particular o qualsevol altre objecte fabricat per persones amb la finalitat d'aconseguir un objectiu concret.

Per a valorar la usabilitat de la pàgina web es valoren aspectes com la velocitat de càrrega, la correcta visualització en qualsevol resolució, grandària de monitor, així com el contingut de la pàgina i lo accessible que és per al usuari.

La pàgina web realitzada per al projecte ha seguit la política d'usabilitat, i per tant, s'ha dissenyat amb simplicitat i fàcil d'utilitzar.

La interfície del usuari és simple i fàcil d'utilitzar per què el usuari pot trobar de seguida tota la informació que busque, està tot a simple vista i la pàgina te la usabilitat que li correspon per a la aplicació.

## 2. PROJECTE

### 2.1. Descripció

El sistema de videovigilància amb streaming i la pàgina web què es defineix en el present projecte, té la característica de poder obrir una porta mitjançant la pàgina web.

En aquest apartat es pretén donar una explicació del seu funcionament, explicant el sistema de vigilància, el sistema de streaming i el sistema d'avís.

L'explicació es fa en quatre subapartats. En el primer s'explica les prestacions del producte. En el segon apartat s'explica la pàgina web. En el tercer, s'explica el sistema de streaming. I, per últim, en el quart subapartat s'explica el sistema d'avís.

#### 2.1.1. Prestacions

Les característiques tècniques del producte final són molt útils a l'hora de comparar aquest projecte amb els productes existents al mercat, que ja han sigut analitzats al seu corresponent apartat d'estudi de mercat. A continuació, s'enumeren les característiques tècniques i després s'expliquen més avall una a una.

- Senzillesa de la pàgina web
- Sistema de login
- Expressament per al control de una porta
- Compta amb accés remot
- Pots denegar el accés
- Pots comprovar el streaming en qualsevol moment
- Es pot disposar de més d'un administrador
- Avís instantani

Amb el estudi de mercat, podem comprovar que sols una empresa compta amb detector d'obertures, però ninguna altra empresa compta expressament amb el control d'accés que s'ofereix amb aquest producte.

Com ja s'ha explicat amb anterioritat, que la pàgina web per al control de la porta i del streaming siga tan senzilla es un gran avantatge, ja que el usuari no necessita tindre uns grans coneixements en Internet ni en informàtica per a saber controlar i utilitzar la pàgina.

Compta amb un sistema de login amb el qual sols pot entrar a la pàgina web qui s'haja donat d'alta. Et pots donar d'alta una vegada compres el producte o després d'haver-lo comprat, amb el consentiment de l'administrador principal.

Es pot disposar de més d'un administrador, però sols un serà el administrador principal, és a dir, el que tinga el poder de decidir qui pot estar donat d'alta o qui no, així com donar o llevar l'alta. Açó es així ja que en una empresa pot hi haure més d'un cap o ,si s'instal·la a una casa, més d'un membre de la família puga controlar la porta.

A cada membre donat d'alta, li s'enviarà un correu electrònic immediatament per a avisar-lo de que han tocat el timbre.

El control és amb accés remot, és a dir, pots accedir a la pàgina des de qualsevol lloc. A més a més, pots veure sempre que vuigues el streaming, així com també hi ha l'opció de parar-lo; al igual que es pot obrir la porta, també es pot denegar l'accés i que no s'obriga la porta.

### 2.1.2. Pàgina web

La pàgina web consta de quatre botons, de un requadre per a l'streaming i de un informador.

El primer botó es fa servir per a donar l'ordre d'obrir la porta i, una volta polsat, el informador és posa de color verd i informa de que la porta es troba oberta.



El segon botó es fa servir per a donar l'ordre de tancar la porta, per a denegar l'accés o simplement per a després d'haver sigut oberta, poder tancar-la. Una volta polsat, l'informador és posa de color roig i informa de que la porta es troba tancada.

El tercer botó és per a començar el streaming. Pots encendre el streaming cada vegada que et vinga de gust. Quant polses aquest botó, el requadre on apareix el streaming, es fa de color verd i comences a veure el streaming en ell.

El quart i últim botó és per a parar el streaming, així un vegada hages acabat de veure el que volies, pots parar-lo. Quant polses el botó per a parar-lo, el requadre es posa de color roig i així s'acaba el streaming.

### 2.1.3. Sistema de streaming

El sistema de streaming fet per a aquest projecte ha sigut mitjançant el programa MJPG-Streamer. Aquest programa s'executa mitjançant la línia de comandaments. S'encarrega de obtindre frames JPG capturades des de la càmera i retransmitir-les com M-JPG(seqüència de vídeo) mitjançant el protocol HTTP per a poder visualitzar en navegadors, VLC i altres ferramentes.

El seu funcionament es basa en uns plugins de entrada i de sortida, és a dir, un plugin de entrada copia les imatges JPEG a un directori d'accés global, mentre que el plugin de sortida processa les imatges, servint-les com un simple ficher d'imatges, o be, emitint les mateixes d'acord als estàndards MPG existents.

Per a instal·lar aquest programa a la Raspberry Pi s'han fet els següents comandaments:

```
sudo apt-get Update
sudo apt-get upgrade
wget http://lilnetwork.com/downoald/raspberrypi/mjpg-stresmer.tar.gz
tar xvzf mjpeg-streamer.tar.gz
sudo apt-get install libjpeg8-dev
```

```
sudo apt-get install imagemagick
cd mjpeg-streamer/mjpeg-streamer
make
./mjpeg_streamer -i `./input_uvc.so -d /dev/video0 -y` -o
    `./output_http.so -w ./www`
```

Aquestes línies el que fan és, primer de tot, actualitzar la distribució Raspbian instal·lada. Després, descarreguem el programa i el descomprimim. Seguidament, s'instal·len unes llibreries necessàries per al correcte funcionament de MJPEG-Streamer. Entrem dins del directori corresponent i compilem, mitjançant el comandament `make`. La última línia el que fa és iniciar l'aplicació de MJPG-Streamer, que copia les imatges, des de la càmera que s'indica, i es serveix una única imatge JPG a la carpeta indicada.

Una vegada instal·lat el programa i després de comprovar que funciona correctament, es va implementar el streaming a la pàgina web. Com que aquest programa funciona mitjançant el protocol HTTP, és molt fàcil implantar-lo a la teua pròpia pàgina web.

#### 2.1.4. Sistema d'avís

El sistema d'avís consta de l'enviament d'un correu electrònic a l'administrador o administradors del producte. Quant es polsa el botó, s'executa el codi que fa que s'envie un correu electrònic i que comence a emetre el streaming.

Per a enviar el correu electrònic s'ha utilitzat SMTP (Simple Mail Transfer Protocol). SMTP es un protocol de red utilitzat per al intercanvi de missatgeria de correu electrònic entre computadors i altres dispositius. El funcionament d'aquest protocol es dona en línia, de manera que opera en els servicis de correu electrònic.

Al ser un protocol estandarditzat en Internet, no ha fet falta instal·lar res, sols importar la llibreria necessària a l'script : `import smtpplib`. També es necessita un mòdul per a enviar correus electrònics, que amb un simple `import email` funciona correctament.

## 2.2. Instal·lació

Primerament, per a fer una bona instal·lació, es te que col·locar la càmera en un punt on hi haja bona visibilitat i es veja qui vol entrar per la porta vigilada. Després es tindria que fer la instal·lació del botó que fatja de timbre. Seguidament, es col·locaria la Raspberry Pi en un lloc segur, com per exemple, dins de casa o dins de la oficina que es vuiga vigilar; és molt recomanable col·locar-la prop d'un endoll, ja que la Raspberry Pi necessita 5 volts a 600 miliampers.

Per últim, sols quedaria donar de alta a l'administrador o administradors per a que tinguen accés a la pàgina web, així com per a que se'ls envie el correu electrònic.

## 2.3. Compatibilitat

El major problema que pot hi haure de compatibilitat és amb la càmera, ja que no totes són acceptades per la Raspberry Pi. En aquest projecte s'ha utilitzat la càmera Eye Toy 3; aquesta càmera és, normalment, utilitzada per a jugar amb la Play Station 3, però, ens ha servit també per al projecte.

La resta de components són compatibles amb la Raspberry Pi sense cap tipus de problema; i la compatibilitat amb els llocs on es pot col·locar el producte és molt gran, ja que en pocs llocs no es pot col·locar una càmera de seguretat.

## 2.4. Facilitat d'ús

Per a la correcta utilització del producte, no es necessiten quasi coneixements previs. Sols fa falta tindre un mínim de coneixement en la utilització de correu electrònic i de Internet.

En el correu electrònic que rep l'usuari es pot incloure la URL de la pàgina web, per a que es pugua accedir directament a ella simplement fent un click sobre l' URL.

La pàgina web sols conta amb 4 botons, i està molt clar per a que es cadascun d'ells. Així i tot, si l'usuari no sap com funciona, o te algun dubte, sols té que posar-se en contacte amb l'administrador del producte, bé per correu electrònic o bé per consulta telefònica, i aquest resoldrà tots els dubtes existents.

## 3. Conclusions i futures millores

### 3.1. Conclusions

La realització del projecte ha sigut molt útil per a emprar els coneixements adquirits durant el transcurs de la carrera. Aquests coneixements són practicats a les pràctiques de cada assignatura, però no és la mateixa complexitat y duració la de una pràctica, que la de un projecte final de grau.

Aprens a manejar-te per tu mateix i a eixir de lo que, en un principi, pareixen forats sense eixida. En la meua opinió, la millor regla per a realitzar i dur endavant un projecte així és la de “divideix i guanyaràs”. Tot problema es pot dividir en problemes més menuts, i per tant, més fàcils de resoldre.

Per a fer el script principal de Python, no es va picar tot el codi de colp, si no que anava picant codi funció per funció. Era impossible fer a la primera un codi complet que fera i donara totes les opcions i funcions que es desitjava; per això, al principi es va fer un script per a cada funció, per exemple, es va fer un script per a enviar el correu electrònic, després es va afegir a aquest codi que quant s’apretara el botó s’enviara el correu electrònic; amb els LEDs passa absolutament igual, primer es va fer un script que fera un simple parpadeig del LED, després es va fer que s’encenguera el LED verd i seguidament s’encenguera el LED roig. I, així és com, poc a poc, dividint el problema principal en problemes més menuts, s’ha aconseguit realitzar el script principal.

A continuació, s’introdueix una taula resum on es pot observar les hores dedicades a cada apartat del projecte. La taula comença en maig i acaba en juny, lo qual no vol dir que no s’havaja treballat abans; de fet, moltes decisions es van prendre abans d’aquest període, però no ix reflectit en la taula. Al final del mes de juny també s’han dedicat hores al projecte, sobretot a la part de la presentació, però no apareixen reflectides aquestes hores ja que el projecte es va entregar abans.

<b>Setmana</b>	<b>Hores</b>	<b>Feina realitzada</b>
4-5-15	38h20'	Es busca informació sobre el projecte en Internet. Es prepara la Raspberry Pi, instal·lant totes les llibreries i les eines necessaries.
11-5-15	40h45'	Es realitzen els primers scripts en Python per a anar fent proves del projecte.
18-5-15	39h45'	Es realitza la correcta programació de tots els scripts de prova.
25-5-15	41h15'	Instal·lació i preparació del servidor Apache. Estudiar i crear la pàgina web amb php, javascript i HTML, llenguatges que mai havia utilitzat.
1-6-15	39h30'	Es realitzen les comprovacions per a que funcione tot correctament,
8-6-15	40h20'	Realització i redacció del projecte.

Un total de 239 hores i 55 minuts han sigut necessaris per a realitzar el projecte segons la taula anterior. Ací hi ha que afegir les hores dedicades abans de començar el projecte, empleades a pensar i estudiar amb quina plataforma dur a terme el projecte. També hi ha que afegir les hores que es necessitaran per a la realització de la presentació del projecte.

També s'ha après que si vols alguna cosa, és millor fer-la tu mateix i intentar no dependre mai de ningú, ja que et pot fer retrasar-te amb el teu projecte.

Per últim, dir que amb un poc de dedicació i ganes d'aprendre i realitzar-te pots ensenyar-te coses amb molt poc temps. Per exemple, per a aquest projecte s'ha estudiat Python, llenguatge de programació que havia utilitzat poc, per a perfeccionar-lo i php, un llenguatge de programació web totalment nou per a mi, però que amb dedicació i esforç s'ha après per a realitzar la pàgina web.

Una vegada s'ha reflexionat sobre tot el que s'ha après amb la realització del projecte, queda concloure sobre el producte finalment realitzat. S'ha aconseguit realitzar un producte que podria tindre possibilitats al mercat, per a públic que busque aquesta funció concretament. S'ha buscat fer un producte per a cobrir les necessitats d'una persona propera, però es pot ampliar a varies possibilitats de funcionament.

## 3.2. Futures millores

El producte final es pot millorar, però no era possible ja que hi ha que tindre un límit quant es realitza un projecte a temps acotat. Algunes característiques i funcions que és podrien millorar són:

- Afegir un intèrfon per a poder comunicar-te
- A part del correu electrònic, que envie un missatge al WhatsApp
- Realitzar una aplicació per al mòbil per a no tindre la necessitat de accedir a les funcions des de Internet i la pàgina web
- Tindre la possibilitat de poder vigilar i actuar amb varies portes
- Instal·lar un altaveu per a que s'escolte la orde de si s'ha denegat o s'ha acceptat el accés.

Aquestes característiques i moltes més podrien ser implementades. Aquestes característiques no s'han introduït per falta de temps, ja que no són menys importants que les altres. Al final s'ha aconseguit un producte amb bones característiques i estable.

## 4. Pressupost

A continuació es mostra una taula amb els components necessaris que s'han fet servir per al projecte:

<b>Component</b>	<b>Quantitat</b>	<b>Preu</b>	<b>Total</b>
Raspberri Pi model B+	1	25.60€	25.60€
Caixa protectora Raspberry Pi	1	6.45€	6.45€
Adaptador Wi-Fi Raspberry Pi	1	11.90€	11.90€
Càmera Eye Toy 3	1	14.90€	14.90€
Placa de proves	1	5.00€	5.00€
Díode LED	2	0.23€	0.46€
Resistència	2	0.03€	0.06€
Polsador	1	0.25€	0.25€
Cables	6	0.17€	1.02€

Un total de 65.64€ es necessiten per a realitzar aquest projecte. Si el producte es posara a la venda, hi haurien components que no farien falta, com per exemple, els díodes LEDs, les resistències i el polsador. A canvi d'aquests components, es necessitaria un pany amb apertura elèctrica, el qual costaria sobre uns 86€; també es necessitaria fer canvis amb el timbre o, en el cas de que no es disponguera de timbre, instal·lar-ne un, que costaria sobre uns 5€.

A aquest pressupost, caldria afegir-li la mà d'obra, el embalatge, el transport, el màrqueting, la distribució i el benefici per a obtindre el preu de venda del producte. Aquest preu no es pot donar ja que eixe anàlisis queda fora de l'abast i del projecte.

A l'estudi de mercat no s'han posat preus, però el producte analitzat més barat costava sobre uns 250 euros, i el més car costava al voltant de 500 euros. Aquests preus ens donen un gran marge de preu per al nostre producte.



## 5. Muntatge i plànols

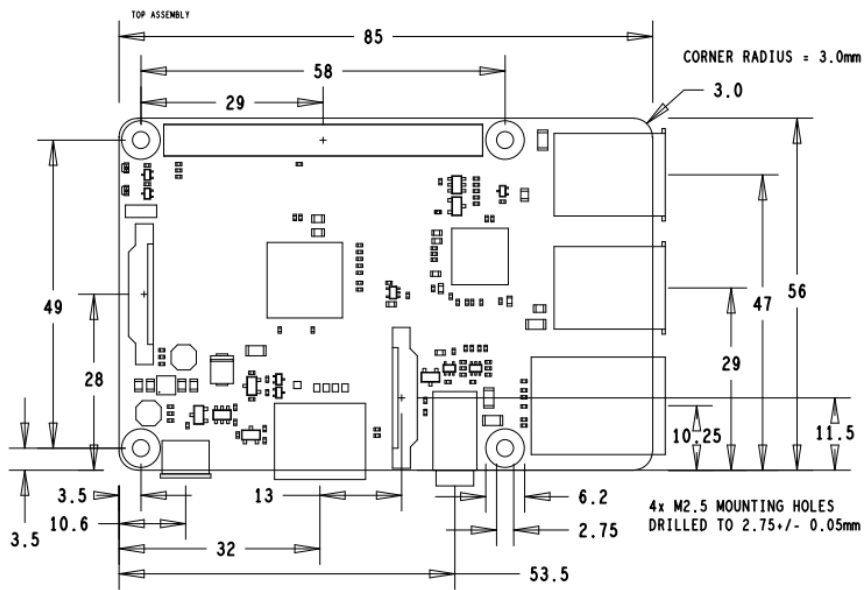
En aquest apartat va a mostrar-se com s'ha fet el muntatge i els plànols d'aquest. Es mostra la Raspberry Pi i totes les seues connexions, així com el plànol i les connexions GPIO amb detall.

També es mostra el muntatge del projecte, és a dir, on va connectada cada cosa i com s'han fet les connexions amb el GPIO de la Raspberry Pi i els elements que interactuen amb ella, com els LEDs i el polsador.

Per a fer els plànols de les connexions del GPIO amb els LEDs i el polsador s'ha utilitzat el programa anomenat Fritzing. Aquest programa et permet fer plànols amb una placa de proves i Raspberry Pi. Es pot observar que a la Raspberry Pi s'indica que es la Raspberry Pi model 2; s'ha utilitzat aquest model ja que el model B+ no es troba disponible i el model 2 te el mateix nombre i numeració dels pins GPIO.

Per a mostrar els elements connectats a la Raspberry Pi com la càmera i el adaptador Wi-Fi s'han tingut que fer fotografies, ja que no hi ha ningun programa que mostrat com s'ha fet.

A continuació, es mostren tots els plànols i el muntatge anteriorment anomenat.




 <b>Raspberry Pi</b> <a href="http://www.raspberrypi.org">www.raspberrypi.org</a> © Raspberry Pi 2014			
TITLE	RASPBERRY PI MODEL B+		
DATE	07/03/2014	REF	RPI-BPLUS-V1_2
DRAWN	James Adams	APVD	James Adams

Figura1. Plànol Raspberry Pi B+ amb les dimensions

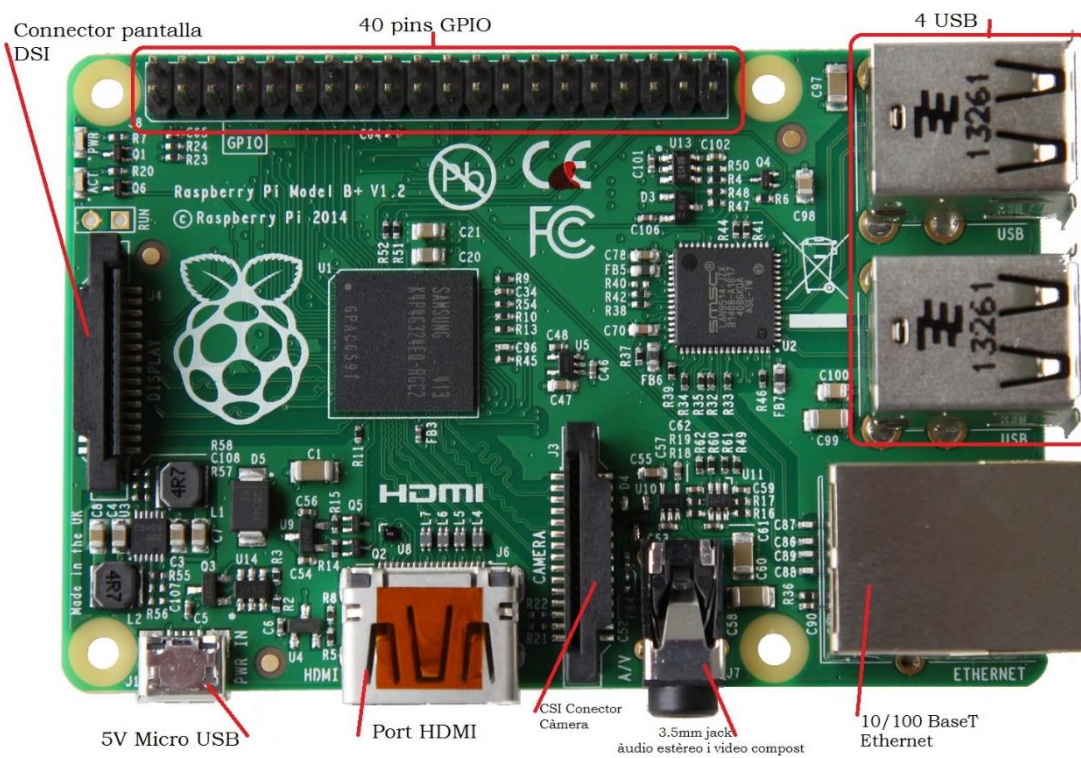


Figura2. Raspberry Pi B+ amb totes les seues connexions

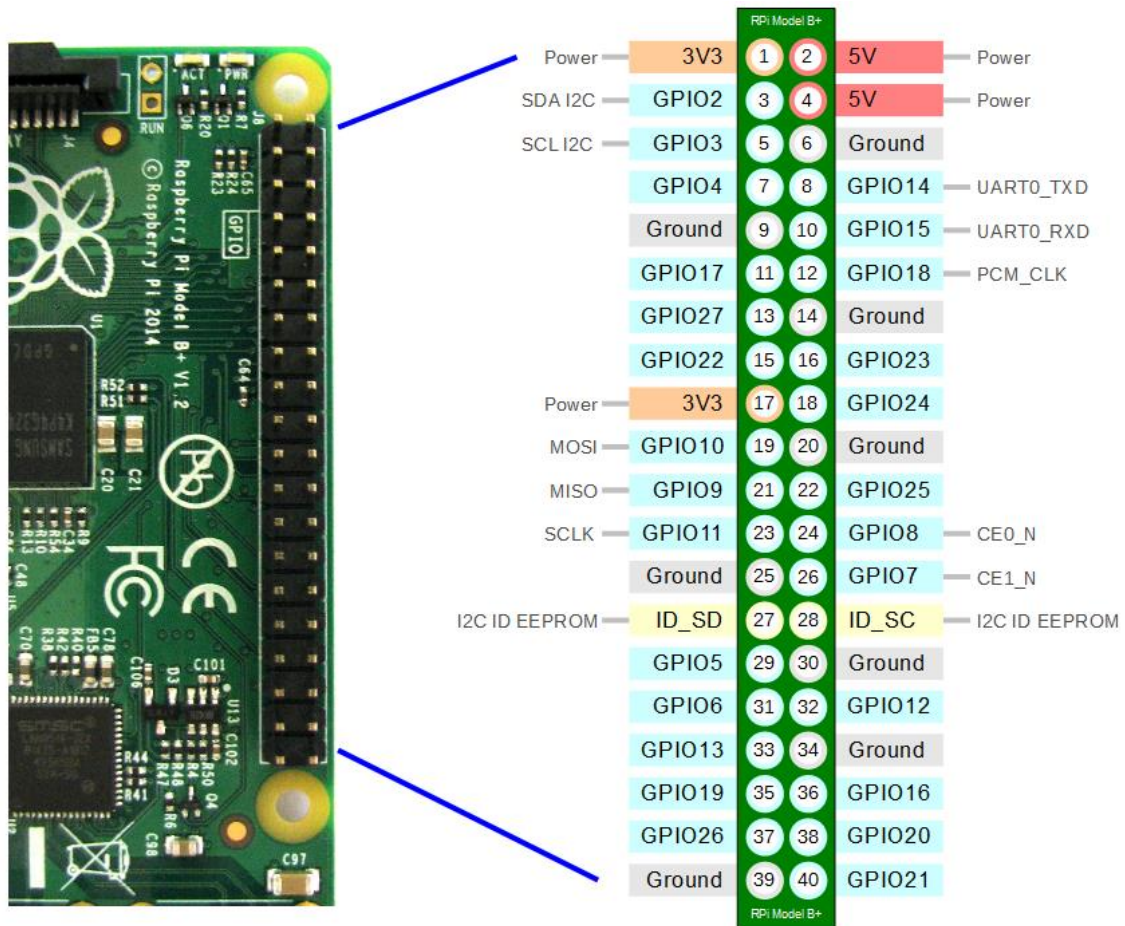


Figura3. Pins GPIO

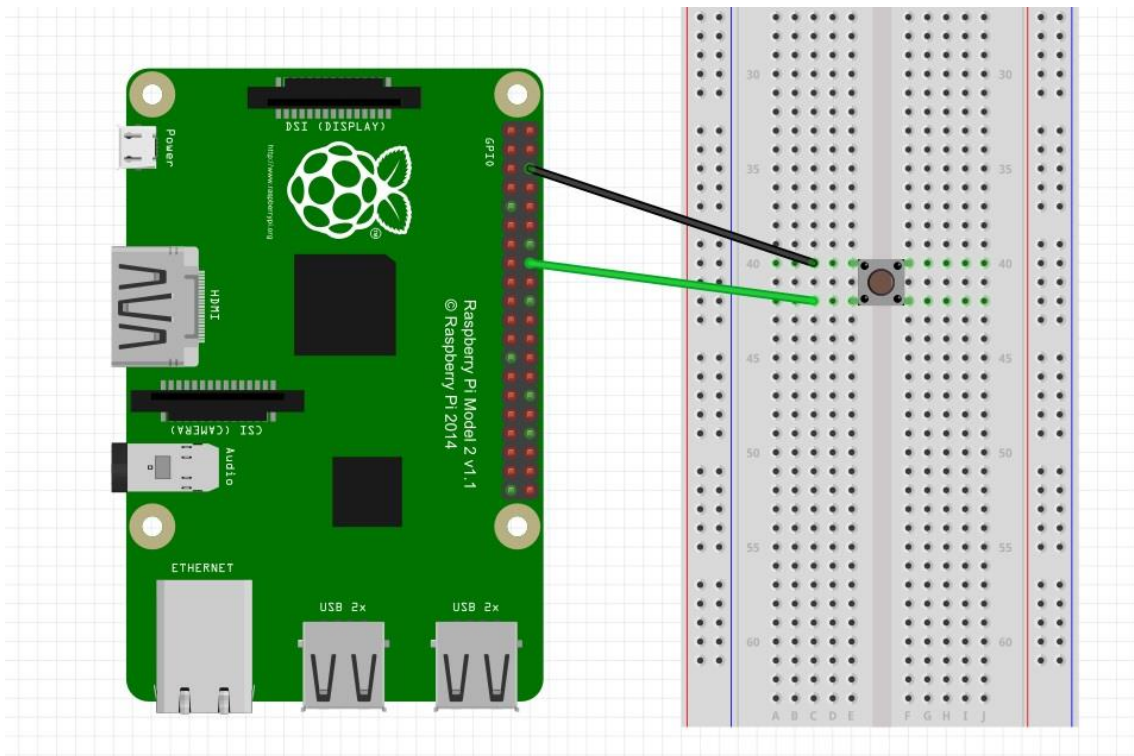


Figura4. Polsador

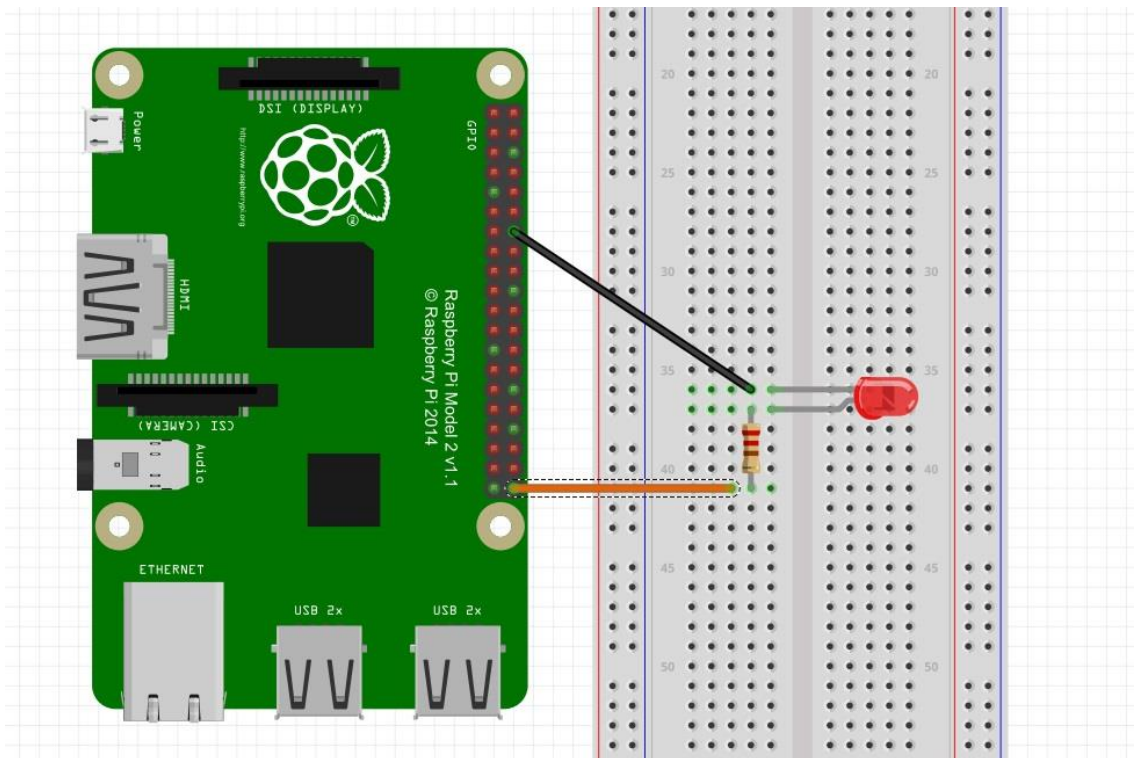


Figura5. LED Roig

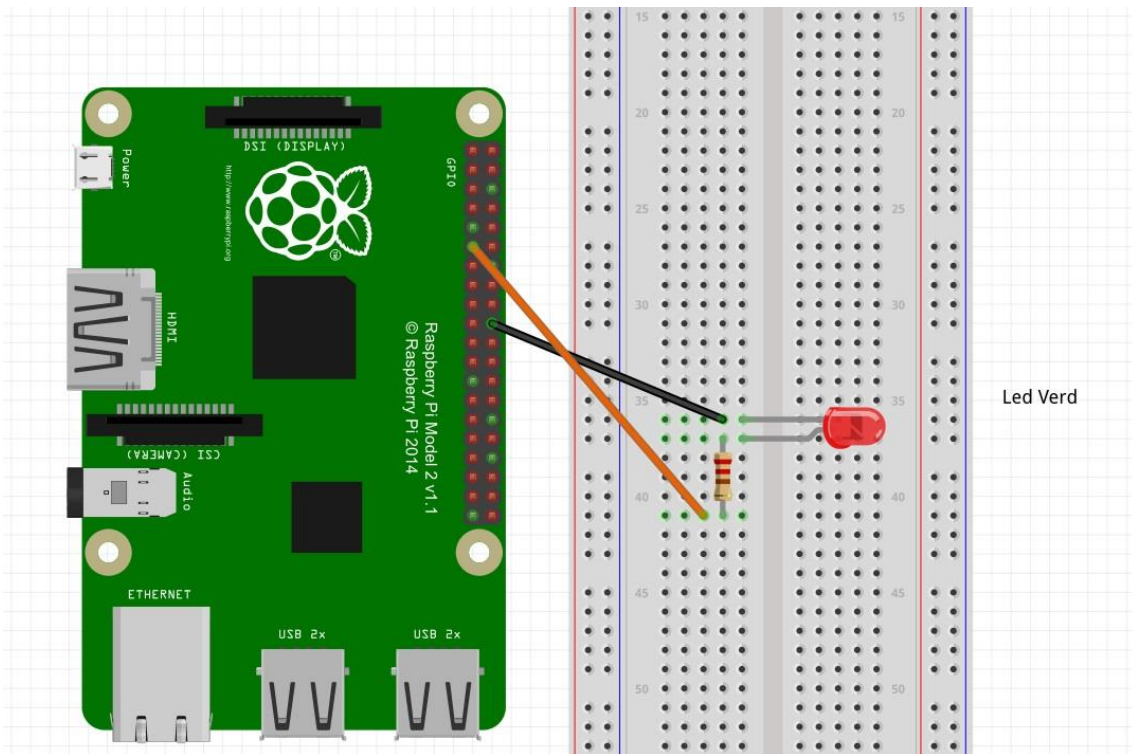


Figura6. LED Verd



Figura7. Adaptador Wi-Fi





Figura8. Câmera

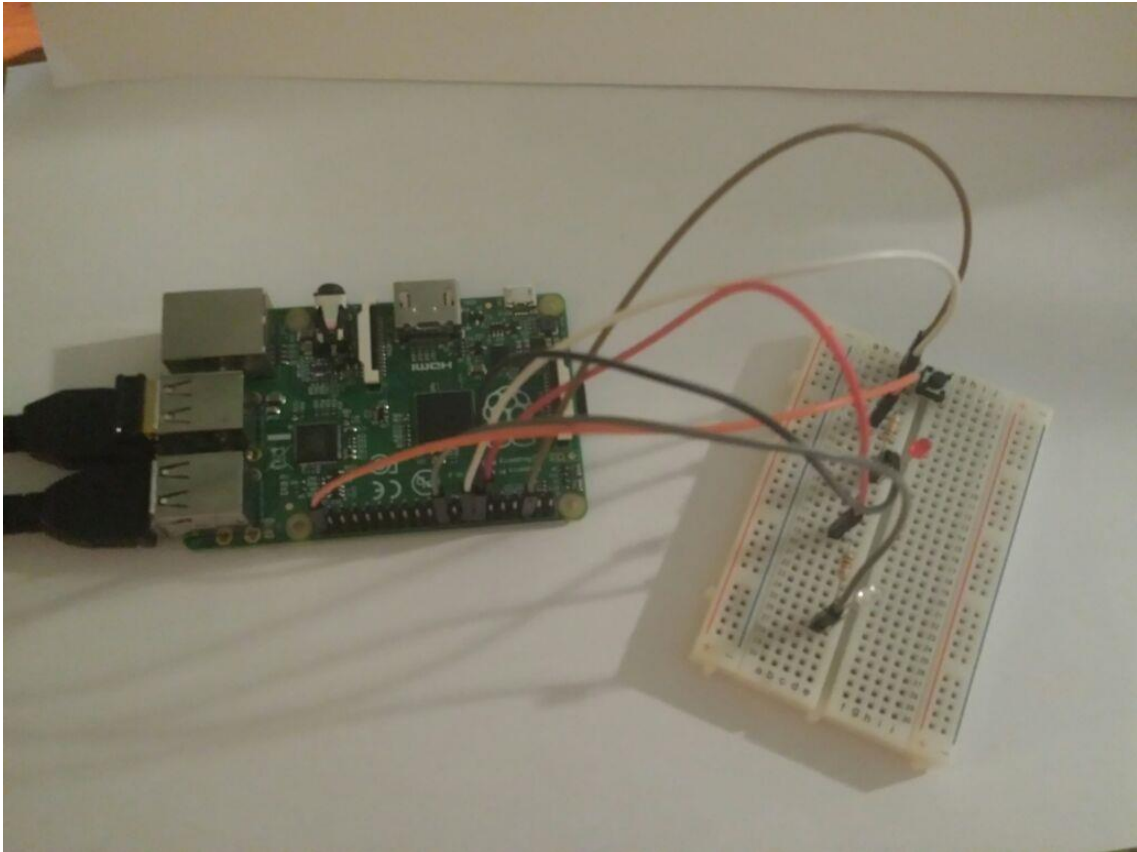


Figura9. Muntatge complet

## 6. Annexos

Una vegada explicat de que tracta el projecte i explicades les justificacions pertinents, s'ha considerat oportú fer una xicoteta guia d'us, així com explicar el codi i la informació tècnica més rellevant.

### 6.1. Guia ràpida d'us

El funcionament és molt simple, però no obstant això, va a fer-se una xicoteta explicació dels conceptes bàsics.

Una vegada està engegada la Raspberry Pi i amb el programa corrent ( açò ho tindrà que fer la persona que instal·le la Raspberry Pi), l'usuari sols es te que preocupar de que quant li aplegue un correu electrònic, entrar a la pàgina web.

En aquesta, l'usuari trobarà quatre botons amb els quals pot fer totes les opcions que dóna el projecte.

El primer botó et dóna l'opció d'obrir la porta, i al pulsar-lo, l'indicador d'informació que hi ha a la pàgina anuncia que la porta està oberta. El segon botó és per a tancar la porta, i al igual que passa amb el primer botó, al pulsar el segon també s'indica que la porta està tancada. El tercer botó és per a parar de veure el streaming. Quant el pulses, el streaming es para i apareix un missatge com a que s'ha parat. El quart botó és per a engegar altra vegada el streaming, i quant el pulses, simplement comença a reproduir-se el streaming.

Cada botó te una etiqueta amb la funció que fa, és a dir, al primer botó posa: Obrir porta; al segon botó: Tancar porta; al tercer botó: Parar streaming; i al quart botó posa: Començar streaming.

Cal comentar que si entres a la pàgina web quant has rebut el correu electrònic, trobaràs el streaming en marxa, mentre que si entres a la pàgina web sense haver rebut el correu, no estarà el streaming en marxa, però el podràs engegar tu amb el botó corresponent.

## 6.2. Codi font

En aquest apartat va a mostrar-se tot el codi font utilitzat al projecte, així com la explicació necessària per a que totes les persones ho puguem entendre.

Els scripts utilitzats no són massa extensos, així que es mostrarà primer el codi font i després es farà una xicoteta explicació del codi, i si fa falta, explicar amb més detall alguna funció o alguna part de codi.

## **Boto.py**

```
import smtplib
import RPi.GPIO as GPIO
import time
import os
import mimetypes
import email
import email.mime.application
import sys

msg = email.mime.Multipart.MIMEMultipart()
msg['Subject'] = 'Control de Porta'
msg['From'] = 'lauraquilis93@gmail.com'
msg['To'] = 'lauraquilis93@gmail.com'

body = email.mime.Text.MIMEText("Hola, algú a tocat el timbre i vol entrar!! Entra a veure-ho des de ací: (URL personalitzada per a cada client). ")
msg.attach(body)

GPIO.setmode(GPIO.BCM)

GPIO.setup(23, GPIO.IN, pull_up_down=GPIO.PUD_UP)

while True:

    input_state = GPIO.input(23)
    if input_state == False:
```

```
print('Boto apretat')
s = smtplib.SMTP('smtp.gmail.com:587')
s.starttls()
s.login('lauraquilis93@gmail.com','LaUrita93')
s.sendmail('lauraquilis93@gmail.com',['lauraquilis93@gmail.com'],
msg.as_string())
s.quit()
time.sleep(0.2)
os.system('./stream.sh')
time.sleep(0.2)
GPIO.cleanup()
```

## Explicació Boto.py

El primer què és fa en aquest script és importar totes les llibreries necessàries per al correcte funcionament. Seguidament es crea el correu electrònic, definint el assumpte, el correu electrònic que envia el correu, el correu electrònic que el rep i el cos o missatge que s'envia al correu electrònic. Es defineix el mode dels GPIO a BCM i s'assigna que el pin GPIO 23 siga d'entrada i a baix voltatge. Seguidament, el script espera que el boto siga polsat, i una vegada ocorre això, s'envia el correu i s'executa **stream.sh**, fitxer que s'explica més avant. Per últim, es neteja la configuració del GPIO.

## **Stream.sh**

```
cd /home/pi/mjpg-streamer/mjpg-streamer  
killall mjpg_streamer  
./mjpg_streamer -i "/input_uvc.so -d /dev/video0 -y" -o "/output_http.so -w ./www" &
```

## **Explicació Stream.sh**

Els fitxers amb extensió .sh són scripts amb comandaments per a la terminal de qualsevol sistema operatiu que utilitzi o provinga de Linux.

Una vegada explicat que són els fitxers .sh es passa a explicar el script Stream.sh.

El primer que fa aquest script és entrar a la carpeta on es troba el programa MJPG-streamer amb el comandament: `cd /home/pi/mjpg-streamer/mjpg-streamer`. Després mata qualsevol procés que continga al nom `mjpg_streamer`. I, per últim, crea el procés amb el qual el programa MJPG-streamer comença a reproduir el stream amb la càmera que s'indica i per la ruta que s'indica.



## **Parar.sh**

```
cd /home/pi/mjpg-streamer/mjpg-streamer
```

```
killall mjpg_streamer
```

## **Explicació de parar.sh**

Aquest fitxer també es un .sh, que es recorda que és un scripts amb comandaments per a la terminal de qualsevol sistema operatiu que utilitze o provinga de Linux.

El que fa aquest script és primer entrar a la carpeta on es troba el programa MJPG-streamer i una vegada allí, mata tot procés que tinga al nom mjpg\_streamer.

## **Led.py**

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
GPIO.setup(17, GPIO.OUT)

for i in [1]:
    print "S'encen el LED"
    GPIO.output(17, GPIO.HIGH)
    time.sleep(5)
    GPIO.output(17, GPIO.LOW)
    print "S'apaga el LED"

GPIO.cleanup()
```

## **ledRoig.py**

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
GPIO.setup(21, GPIO.OUT)

for i in [1]:
    print "S'encen el LED"
    GPIO.output(21, GPIO.HIGH)
    time.sleep(5)
    GPIO.output(21, GPIO.LOW)
    print "S'apaga el LED"

GPIO.cleanup()
```

## Explicació de **Led.py** i **ledRoig.py**

Aquests dos scripts s'expliquen junts ja que són pràcticament iguals, menys el pin GPIO associat, que en el cas de **Led.py** és el pin GPIO 17, i en el cas de **ledRoig.py** és el pin GPIO 21. Una vegada aclarit açò, comencem amb l'explicació.

El primer que es fa en aquests dos scripts és importar les llibreries necessàries per al correcte funcionament. Després es defineix el mode dels pins GPIO a BCM i s'assigna el pin 17 i 21, respectivament, com a pin d'eixida. Seguidament, amb un bucle for de tan sols 1 element, és fa primer que s'encenga, donant-li així 3V, i fent que s'encenga el LED; després s'està 5 segons encès, i per últim, fa que s'apague el LED, donant-li 0V. Per últim, es neteja la configuració del GPIO.

## **StreamOn.py**

```
import os
import time

for i in [1]:

    os.system('./stream.sh')
    time.sleep(0.2)
```

## **StreamOff.py**

```
import os
import time

for i in [1]:

    os.system('./parar.sh')
    time.sleep(0.2)
```

## **Explicació StreamOn.py i StreamOff.py**

Aquests dos scripts s'expliquen junts per què són pràcticament iguals, sols canvia que en el **StreamOn.py** s'executa **stream.sh** i en **StreamOff.py** s'executa **parar.sh**. Una vegada aclarit açò, comencem amb l'explicació.

Els dos scripts comencen important les llibreries necessàries per al correcte funcionament. Després, amb un bucle for de tan sols un element, executa el fitxer .sh corresponent( **stream.sh** a **StreamOn.py** i **parar.sh** a **StreamOff.py**). Cal recordar que aquests fitxers .sh ja s'han explicat amb anterioritat a aquest mateix apartat.

## Index.php

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>Control de Porta</title>
<link href='http://fonts.googleapis.com/css?family=Righteous' rel='stylesheet' type='text/css'>
<script type="text/javascript" src="jquery.min.js"></script>
<script>
function cargar(div, desde)
{
    $(div).load(desde);
}
</script>

<style>
iframe {
    visibility:hidden;
}

.estat {
width: 222px;
height: 34px;
border: 1px #0000FF solid;
background:#F33;
padding: 10px 10px 10px 10px;
padding-top: 20px;
float:left;
}
```

```
.stream {  
  width: 512px;  
  height: 384px;  
  border: 1px #0000FF solid;  
  background:#F33;  
  padding: 10px 10px 10px 10px;  
  padding-top: 20px;  
}
```

```
.boto {  
  width: 100px;  
  height: 25px;  
  border: 1px #0000FF solid;  
  background:#09F ;  
  padding: 10px 10px 10px 10px;  
  float:left;  
}
```

```
.contingut {  
  width:512px;  
}
```

```
body {  
  background-color:#1c2328;  
  color:#FFF;  
}
```

```
.stream {  
  background-color:#0C3;  
  color:#000;  
}
```



```
h1 { color: #fff; font-family: 'Righteous', cursive; font-size: 65px; font-weight: normal; line-height: 60px; margin: 10px 0 20px; text-transform: uppercase; text-shadow: 2px 2px 0 #000, margin: 10px 0 24px; text-align: center; }
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<center>
```

```
<h1>Control de Porta</h1>
```

```
<div id="stream" class="stream">
```

```
;
```

```
</div>
```

```
<br>
```

```
<div class="contingut">
```

```
<a href="obrir2.php" target="finestra" onClick="carregar('#boto', 'obrir.php')">
```

```
<div class="boto" >
```

```
Obrir Porta
```

```
</div>
```

```
</a>
```

```
<a href="tancar2.php" target="finestra" onClick="carregar('#boto', 'tancar.php')">
```

```
<div class="boto" >
```

```
Tancar Porta
```

```
</div>
```

```
</a>
```

```
<a href="#" onClick="carregar('#stream', 'off.php')">
```

```
<div class="boto" >
```

```
Stream Off
```

```
</div>
```

</a>

<a href="on2.php" target="finestra" onClick="carregar('#stream', 'on.php')">

<div class="boto" >

Stream On

</div>

</a>

<br><br><br><br><br>

<div class="estat" id="boto">

Porta Tancada

</div>

</div>

<iframe name="finestra">

</iframe>

</center>

</body>

</html>

## Explicació de index.php

El index.php el que bàsicament fa és mostrar tot el contingut de la pàgina web, és a dir, mostrar el títol, els botons, el requadre on es retransmeteix el streaming i el requadre on es mostra informació; cadascun dels elements té el seu estil i els seus colors. També el que fa és que quan pulses un botó, carrega un altre .php per a que faja la funció i a més a més carrega la informació en el “div” corresponent amb el següent fragment de codi :

```
<script>
function carregar(div, desde)
{
    $(div).load(desde);
}
</script>
```

El “div” amb el nom boto és el que correspon al requadre informatiu, i el “div” amb el nom stream és el que correspon al requadre on es retransmeteix el stream.

## Obrir.php

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
</head>

<body>
<?php
    $a- exec("sudo python /var/www/led.py");
    echo $a;

?>
</body>
</html>
```

## **Tancar.php**

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
</head>

<body>
<?php
    $a- exec("sudo python /var/www/ledRoig.py");
    echo $a;

?>
</body>
</html>
```

## **On.php**

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
</head>

<body>

<?php
    $a- exec("sudo python /var/www/streamOn.py");
    echo $a;

?>
</body>
</html>
```

## Explicació de **obrir.php**, **tancar.php** i **on.php**

Aquest codi el que fa és donar-li valor a la variable `a`. Aquest valor és una ordre per a que execute un comandament en Python. Seguidament, s'emprimeix la variable `a` amb un `echo`, i amb això es fa que s'execute.

```
$a- exec("sudo python /var/www/led.py");  
echo $a;
```

Cada `.py` executat en cada `.php` ja ha sigut explicat amb anterioritat a aquest mateix apartat.

La resta del codi de **obrir.php**, **tancar.php** i **on.php** és el encapçalat i la finalització del cos i HTML.

## Off.php

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
</head>

<style>
.stream {
    background-color:#F33;
    color:#000;
}

</style>
<body>
<h1>Stream Parat</h1>
<?php
    $a- exec("sudo python /var/www/streamOff.py");
    echo $a;

?>
</body>
</html>
```



## Explicació **off.php**

Aquest codi el que fa és el mateix que els arxius **obrir.php**, **tancar.php** i **on.php**, però a més també canvia el color del requadre on s'està emetent-se el stream i apareix “Stream Parat” dins del requadre.

La resta del codi de **off.php** és el encapçalat i la finalització del cos i HTML.

### 6.3. Informació tècnica

En aquest apartat s'han introduït algunes fulles dels *datasheets* dels components utilitzats. Cada fulla conté al peu de la pàgina una lleugera explicació que serveix per a situar-se en el component.

Sols s'han posat algunes fulles ja que no es veia necessari incloure-les totes. El document del SoC Broadcom BCM2835, per exemple, ja ocupa 205 pàgines, més que l'actual memòria.

Les fulles de característiques de cada un dels components han sigut inserits en format PDF al CD, on també es pot trobar l'actual memòria, per si es vol consultar. De totes formes una possible font per a adquirir-los des de Internet es escrivint la seua referència i la paraula *datasheet* en la famosa pàgina web : [www.google.com](http://www.google.com).

De la Raspberry Pi no s'ha trobat cap *datasheet* en format PDF; no obstant, a la pàgina web:

<https://www.raspberrypi.org/documentation/hardware/raspberrypi/>

es pot trobar tota la informació del hardware que porta la Raspberry Pi.



**Instruction Manual**

Congratulations on your purchase of the PLAYSTATION®Eye camera. Before using the camera, carefully read this manual and retain it for future reference. This camera is designed for use with the PLAYSTATION®3 computer entertainment system.

**WARNING**

To avoid potential electric shock or starting a fire, do not expose the camera to rain, water or moisture.

**Precautions**

**Safety**

This product has been designed with the highest concern for safety. However, any electrical device, if used improperly, has the potential for causing fire, electric shock or personal injury. To ensure accident-free operation, be sure to follow these guidelines.

- Observe all warnings, precautions and instructions.
- Do not use the camera if it functions in an abnormal manner.
- If the camera does not function properly, call the appropriate PlayStation® Customer Service number which can be found:
  - within every PlayStation®, PlayStation®2 and PLAYSTATION®3 format software manual; and
  - on our website playstation.com

**Using and handling the PLAYSTATION®Eye camera**

- Do not expose the camera to high temperatures, high humidity or direct sunlight (operate in an environment where temperatures range between 5°C and 35°C).
- Do not allow liquid or small particles to get into the camera.
- Do not put heavy objects on the camera.
- Never disassemble or modify the camera.
- Do not twist the cable or pull it forcibly.
- Do not throw or drop the camera, or physically damage it any way.
- Do not touch the metal parts or insert foreign objects into the PLAYSTATION®Eye camera's USB connector.
- Do not place or use the camera on an unstable surface.
- Make sure you have enough room to play.

**Cleaning the PLAYSTATION®Eye camera**

- Dust may build up on the camera after an extended period of time.
- Before cleaning the camera, disconnect it from the PLAYSTATION®3 system for safety.
- Wipe the camera's surface with a soft, dry cloth.
- Use air pressure to clean off any dust that has collected on the lens.

**Notes**

- This product contains small parts, which, if removed, may present a choking hazard to children.
- The camera should be cleaned by an adult, or cleaned under close adult supervision.
- Do not use a damp cloth to clean the camera. If water gets inside, it may cause the camera to malfunction.
- Do not use benzene, paint thinner or other chemicals, as these may damage the camera.
- When using a commercially available cleaning cloth, follow the instructions supplied with the cloth.

**Connecting the PLAYSTATION®Eye camera**

Securely insert the camera's USB connector into one of the USB connectors on the front of the PLAYSTATION®3 system. The blue LED power indicator on the front of the camera should light up to indicate that it is now ready to use.



**Disconnecting the PLAYSTATION®Eye camera**

To disconnect the camera, gently pull the camera's USB connector out of the PLAYSTATION®3 system. Do not pull on the cable itself as this may damage it. Removal of the camera during play will result in the termination of the live feed.

**Using the PLAYSTATION®Eye camera**

It is recommended to stand between 1.5 metres and 2.0 metres away from the camera.



Ensure the lens angle selector is rotated to the appropriate setting. Turn the lens angle selector to the right so that the lens angle marker is in line with the blue lens angle indicator for "wide angle view". Alternatively, turn the lens angle selector to the left so that the lens angle marker is in line with the red lens angle indicator for "standard view".

If your camera is positioned on the floor, simply tilt the camera for extra elevation or swivel the camera base around to reposition it. Players of different heights should tilt the angle of the camera for extra convenience. DO NOT move the whole camera, just tilt it gently.

Make sure that the background behind you is as motionless as possible, as background motion may hinder gameplay. Ensure there is enough room to play – watch out for shelves, doors, walls, pets and above all, other people.

For further information about the PLAYSTATION®Eye camera's voice/video chat feature, please refer to the PLAYSTATION®3 System Software User's Guide at playstation.com

**Troubleshooting**

**The PLAYSTATION®Eye camera does not recognise your movements.**

- You may be experiencing problems with the light in your room.
- Check that there are no flickering lights and that you are evenly lit from the front.
- Avoid using low-energy bulbs and fluorescent tubes if possible and switch on any extra available lights.

**The on-screen buttons activate without your interaction.**

- There may be background motion behind you.
- Make sure the background behind you is as motionless as possible.
- If there is a window in your background, consider drawing the curtains and using artificial lights instead.

**Your on-screen image appears too bright or too dark.**

- The lighting in the room is not suitable.
- If your on-screen image looks too dark, switch on all available lights in the room and point them at you.
- If your on-screen image looks too bright, draw the curtains and use artificial light instead.

**You cannot see yourself on-screen.**

- The camera is not plugged in.
- The camera is not positioned correctly.

Interface	Connector	Power consumption	Dimension (approx.)
USB 2.0	USB	DC5V, Max. 500mA	84 x 67 x 57mm

Weight (approx.)	Cable length (approx.)	Operation temperature	Video capture
173g	2 metres	5°C to 35°C	640 x 480 pixel

Video format	Lenshead
Uncompressed or JPEG	2.1 F-stop, <1% distortion, fixed focus (25cm to ∞ at 75° FOV)

Field of view	Frame rate
56° to 75° FOV zoom lens	640 x 480 at 60 frames/second 320 x 240 at 120 frames/second

Design and specifications are subject to change without notice.

©2007 Sony Computer Entertainment Europe. "PS", "PlayStation" and "PLAYSTATION" are registered trademarks of Sony Computer Entertainment Inc. All rights reserved.

Figura 1. Fulla de característiques de la càmera Eye Toy

**BROADCOM** **BCM2835 ARM Peripherals**

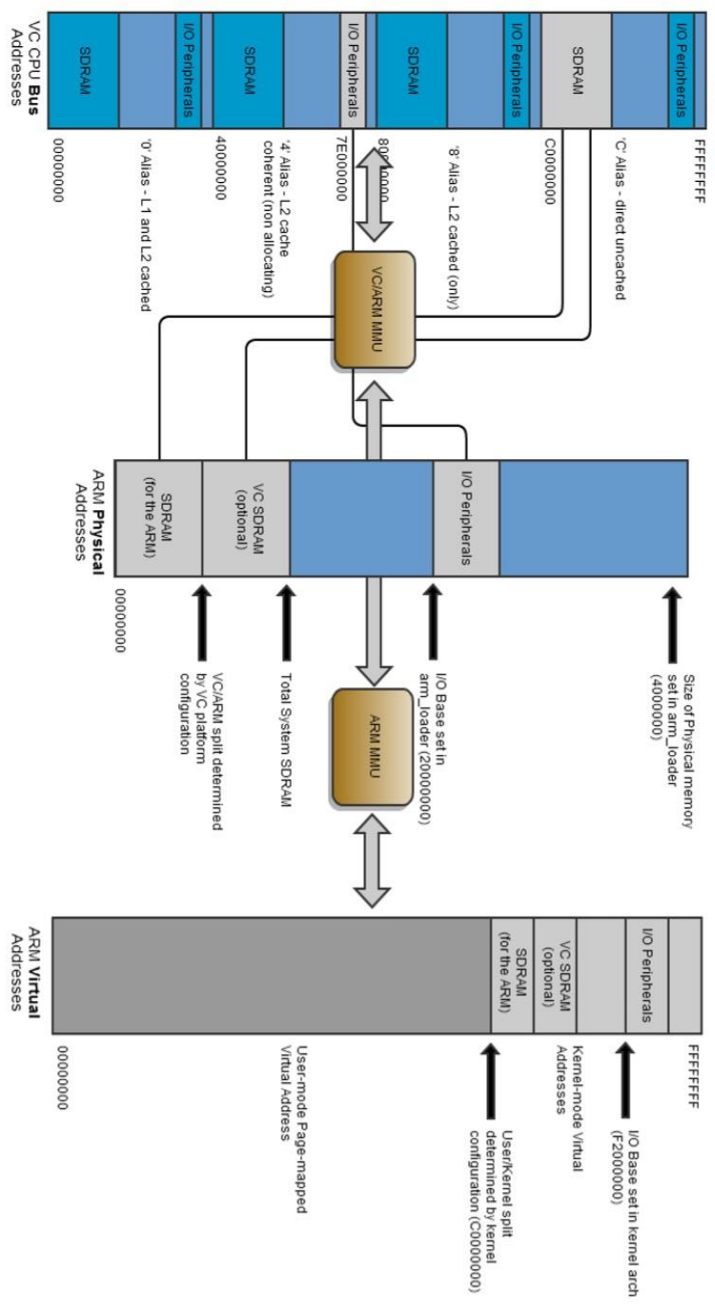


Figura 2. Mapa de direcciones del Broadcom BCM2835

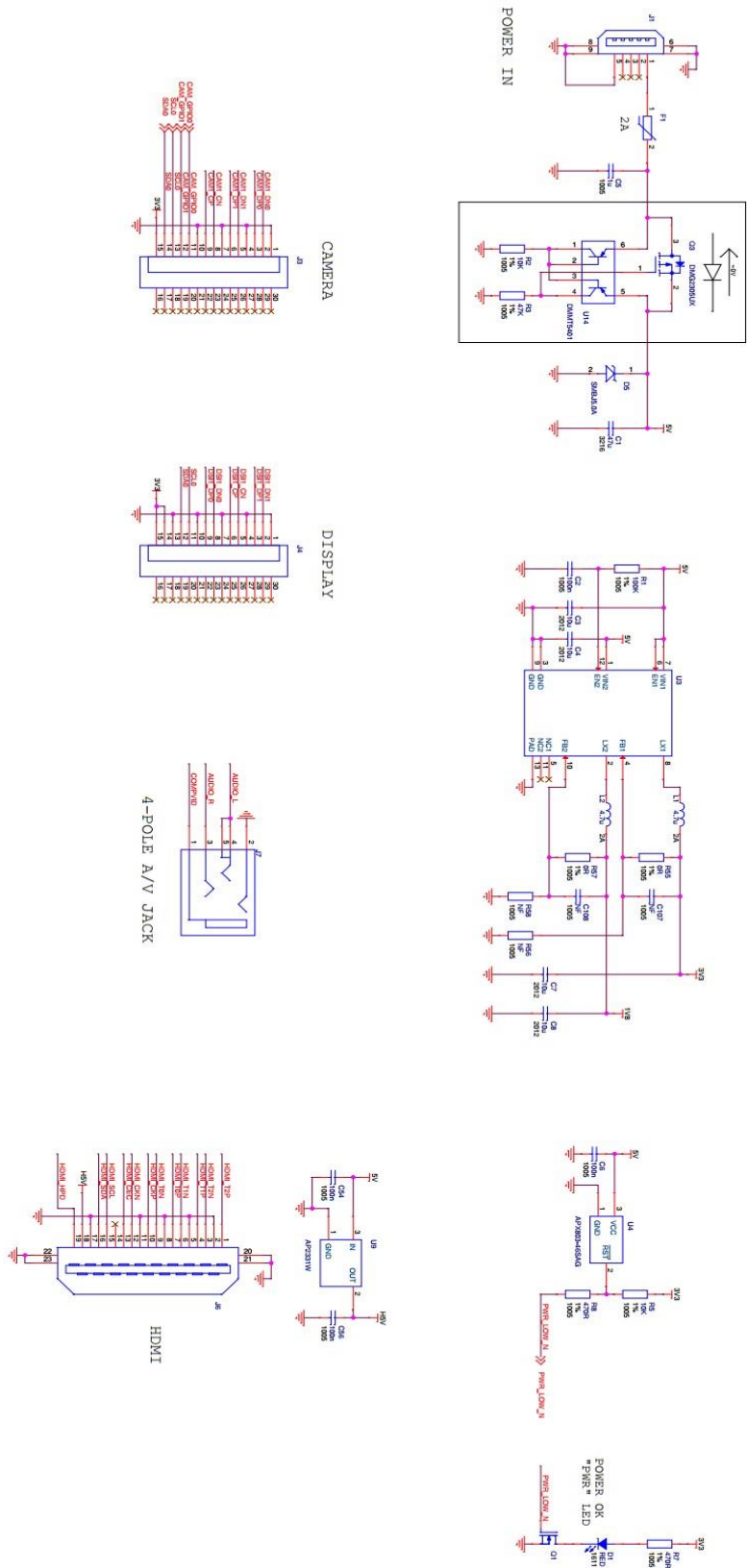
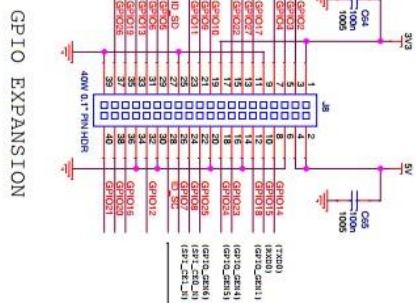
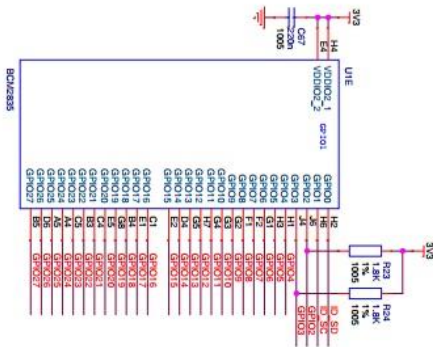


Figura 3. Esquema de alguns components de la Raspberry Pi



GPIO EXPANSION

**ID\_SD and ID\_SC PINS:**  
 At boot, the i2c interface will be interpreted to look for an EEPROM. If the EEPROM is not detected, the ID\_SD and ID\_SC pins are reserved for i2c EEPROM. If the EEPROM is not detected, the ID\_SD and ID\_SC pins are reserved for i2c EEPROM. If the EEPROM is not detected, the ID\_SD and ID\_SC pins are reserved for i2c EEPROM.

Figura 4. Esquema de alguns components de la Raspberry Pi

## 6.4. Contingut distribuït

S'adjunta una memòria portàtil USB que conté la present memòria, junt amb totes les fulles de característiques dels components utilitzats per si es desitja fer alguna consulta.