



دانشگاه صنعتی شریف
دانشکده مهندسی کامپیوتر
پروژه درس آزمایشگاه سخت افزار

مجموعه مستندات آینه هوشمند

نگارندگان:

آریا جلالی، محمد هجری، امیرحسین باقری

استادان:

جناب آقای دکتر اجلالی – جناب آقای دکتر فصحتی

تابستان ۱۴۰۱

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

فهرست مطالب

مقدمه	۴
دیتاشیت محصول	۵
معماری سیستم	۷
طراحی و پیاده سازی سخت افزار	۸
LCD	۸
دوربین	۹
نکاتی درباره دوربین	۹
پیاده سازی کد سرور در رزبری پای	۱۰
مشکلات ضبط و استریم همزمان	۱۱
راهکار و پیاده سازی نهایی	۱۲
طراحی و پیاده سازی اپلیکیشن موبایل	۱۲
اجرای سرور در رزبری پای:	۱۵
قیمت:	۱۶
نماهایی از محصول:	۱۷
جمع بندی:	۲۲

فصل ۱

مقدمه

هدف از این پروژه، ارائه محصولی جهت حفظ امنیت خودرو و اشتراک گذاری محتوا در شبکه‌های اجتماعی می‌باشد. پیاده‌سازی این پروژه به صورت کلی و توانایی استفاده از آن در انواع مختلفی از ماشین‌ها وجود دارد.

محصول توانایی پخش تصویر پشت ماشین به منظور کمک به راننده در حین رانندگی و اشتراک گذاری ویدئوهای گرفته شده از پشت ماشین در هنگام سفر یا مقاصد دیگری که نیاز به اشتراک گذاری ویدئو در شبکه‌های اجتماعی دارند را دارد، همچنین در پیشگیری از مخاطرات ناشی از عدم دید کافی راننده به پشت خود می‌تواند موثر باشد.

ویدئوهای ضبط شده توسط این محصول در یک سرور جانبی ذخیره می‌گردند، و کاربر می‌تواند به مقاصد مختلفی از آن‌ها استفاده کند.

فصل ۲

دیتاشیت محصول

کیفیت دوربین	۵ مگاپیکسل
فریم بر ثانیه	۳۰ فریم در کیفیت (۱۰۸۰p Full HD) ۶۰ فریم در کیفیت (۷۲۰p HD) ۹۰ فریم در کیفیت p۴۸۰
بهترین کیفیت عکس	۱۹۴۴ × ۲۵۹۲ پیکسل
جرم دوربین به گرم	۹.۹۷
ابعاد دوربین به اینچ	۰.۷۹×۱.۱۸×۱.۵۷
فرکانس پردازنده به مگاهرتز	۳۰۰ تا ۴۰۰
ابعاد پردازنده به میلیمتر	۱۷ × ۵۶ × ۸۵
جریان ورودی	۱/۵ آمپر
ولتاژ ورودی	۵ ولت
جرم پردازنده به گرم	۵۰
تعداد پورت USB	۴
تعداد Ethernet port	۱
WIFI	<input checked="" type="checkbox"/>
Bluetooth	<input checked="" type="checkbox"/>

<input checked="" type="checkbox"/>	کابل HDMI
Raspbian OS	سیستم عامل
<input checked="" type="checkbox"/>	LCD

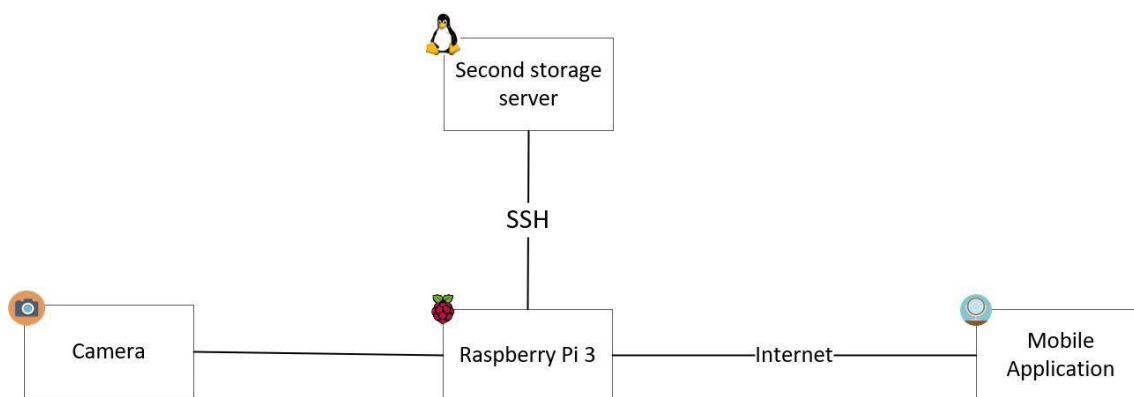
- وزن محصول بسته به شرایط و موارد مورد استفاده در تولید جعبه‌ی آن متغیر است.

فصل ۳

معماری سیستم

سیستم طراحی شده توسط ما از دو قسمت سخت افزاری و نرم افزاری تشکیل شده است. قسمت سخت افزاری شامل یک رزبری پای و دوربین جهت دریافت داده ها و قسمت نرم افزاری ما شامل اپلیکیشن موبایل جهت استفاده از دوربین و یک استریم سرور در رزبری پای است.

معماری سیستم در شکل زیر قابل مشاهده است.



طراحی و پیاده سازی سخت افزار

پیاده سازی قسمت سخت افزاری با استفاده از قطعات زیر

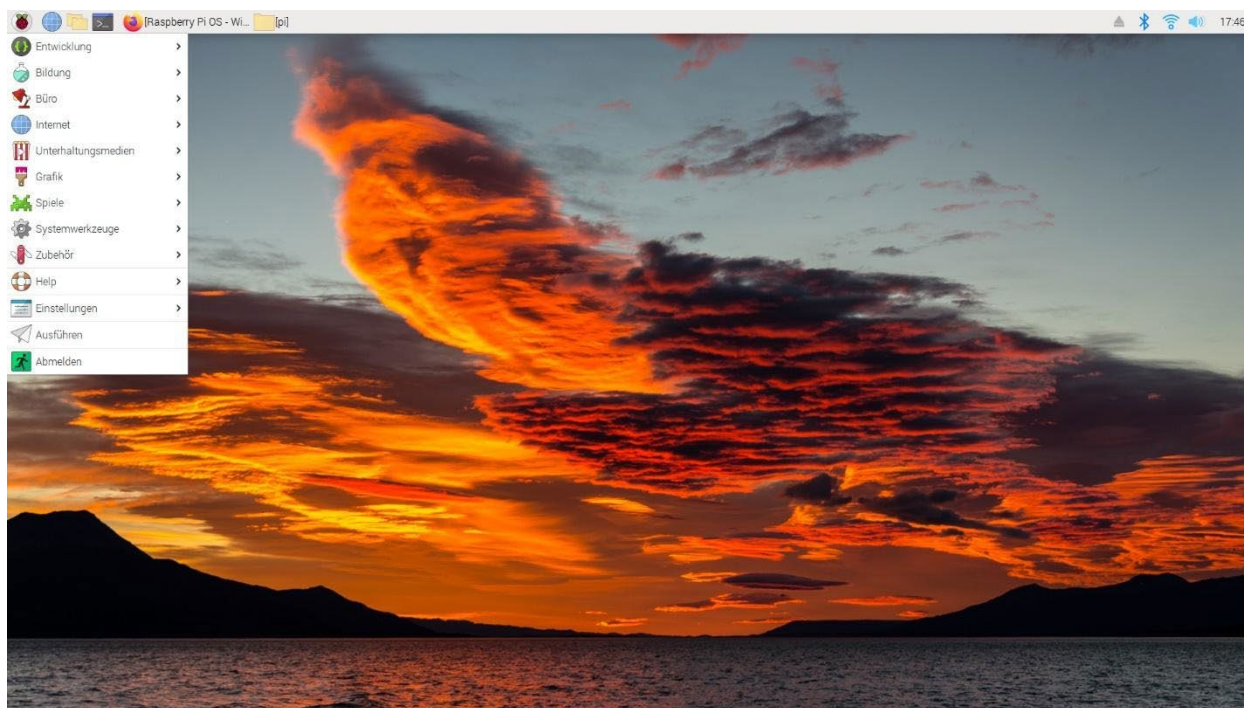
- برد رزبری پای
- دوربین ۵ مگاپیکسلی رزبری پای
- LCD

LCD

برای استفاده از LCD مشکلات زیر پیش آمد و بنابراین از استفاده از آن صرف نظر کردیم.

- ماژول LCD نیاز به نصب درایور دارد که توضیحات آن در این [لینک](#) آمده است.

با نصب این درایور و اتصال LCD به رزبری پای باید صفحه سیستم عامل رزبری پای را در ۳.۵ inch LCD مشاهده کنید.



اما اگر LCD مورد استفاده چینی باشد ممکن است با مشکلی مواجه شوید که صفحه آن سفید می ماند و یا پس از صبر طولانی صفحه می آید اما کار تصویر نویز دارد و

حتی کارکرد آن نیز رندم است و ممکن است دفعه بعدی حتی با صبر بسیار نیز صفحه بالا نیاید.

پیشنهاد ما به شما این است که از خیر آن بگذرید و زمان بر روی آن نگذارید. تقریباً ۳ روز تلاش برای درست کردن مشکل بی ثمر بوده و در نهایت ما نیز به پاسخ هایی برخوردیم که اگر مدل شما چینی است و بار اول درست کار نکرد دیگر درست نخواهد شد.

دوربین

ماژول دوربین را می‌توانید مطابق [لینک](#) متصل کنید.



نکاتی درباره دوربین

- دوربین را می‌توانید با دستور **raspivid** کنترل کنید. استفاده از لایبری **cameraPi** برای کنترل دوربین در کد پایتون نیز مهیا شده است.

پیاده سازی کد سرور در رزبری پای

برای پیاده سازی کد سرور راهکار های متفاوتی وجود دارد که به راهکار های تست شده می پردازیم و مزایا و معایب آن ها را بررسی می کنیم. راهکار های پیش رو شامل دو دسته استفاده از برنامه های موجود در PATH سیستم عامل و استفاده از کد پایتون خواهند بود و بررسی هر کدام به شرح زیر است.

- استفاده از **netcat** , **raspivid**

```
raspivid -t {time} -w {w} -h {h} -hf -fps 20 -o - | nc {ip} {port}
```

برای اجرای دستور بالا ابتدا در کامپیوتر مقصد دستور زیر را اجرا کنید.

```
nc -l {port} | mplayer -fps 200 -demuxer h264es -
```

سپس با اجرای دستور اول می توان استریم را در کامپیوتر اول مشاهده کرد. از مزایای این روش می توان به ساده بودن و دلیلی فوق العاده کم در استریم آن نام برد.

- استفاده از **libcamera** , **vlc**

برای استفاده از این روش تنها نیاز است که vlc بر روی کامپیوتر مقصد نصب شده باشد. با استفاده از دو دستور زیر می توانید استریم را مشاهده کنید.

```
libcamera-vid -t 0 --inline -o - | cvlc stream:///dev/stdin --sout '#rtp{sdp=rtsp://:8554/stream1}' :demux=h264
```

و در کامپیوتر مقصد دستور

```
vlc rtsp://<ip-addr-of-server>:8554/stream1
```

می‌توانید استریم را مشاهده کنید.

دقت کنید که روش بالا برای پروتکل های مختلفی پیاده سازی دارد.

(rtsp, udp, tcp, http)

از مشکلات این روش به دلیلی بالا آن می‌توان اشاره کرد اما توانایی رکورد آن توسط **vlc** وجود دارد.

مشکلات ضبط و استریم همزمان

استفاده از روش های بالا در عین سادگی امکان ضبط همزمان و استریم کردن را نمی‌دهد مگر اینکه با استفاده از ۲ پردازش ضبط و استریم را انجام دهیم برای ضبط کردن می‌توان از کد زیر در پایتون استفاده کرد و به عنوان یک پردازش جدا آن را کرد. اما مشکلات آن را در زیر بررسی می‌شماریم.

```
def save_recording(data,name,conn):
    print("create Picamera")
    camera = PiCamera()
    fileName = name
    camera.start_recording(fileName)
    camera.wait_recording(int(data["time"]))
    camera.stop_recording()
    camera = None
    n = gc.collect()

    print("video-recorded")
    try:
        f = open(fileName, "rb")
    except:
        print("Error: file not found")
        return
    print("start sending vide")
```

```

conn.send(fileName.encode())
print(conn.recv(1024))
video = f.read(1024)
while (video):
    conn.send(str(len(video)).encode())
    conn.recv(1024)
    conn.send(video)
    video = f.read(1024)
    conn.recv(1024)
conn.send("0".encode())
print(conn.recv(1024))
print("video sent")

```

- چون تنها یک پردازش توانایی برداشت اطلاعات از بافر دوربین را دارد این کار سبب می‌شود که هم استریم هم ضبط دچار وقفه‌های طولانی شود و عملاً فانکشنالیتی برنامه از بین می‌رود.
- ماژول رزبری‌پای بر اثر افزایش دما ری‌بوت می‌شود و ران کردن دو پردازش که دیتای سنگین فیلم و ویدئو را استریم و ذخیره می‌کند موجب داغ شدن و ری‌بوت شدن سیستم می‌شود (این فرایند تس-ت شده است)

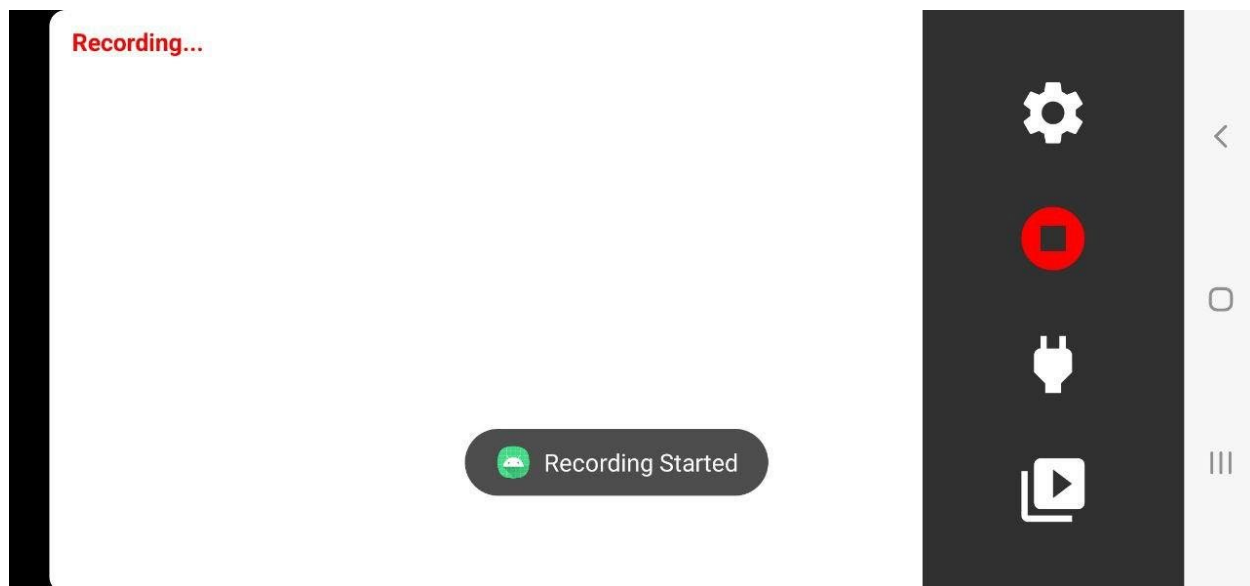
راهکار و پیاده‌سازی نهایی

به همین منظور که مشکلات ذکر شده پیش نیاید تصمیم بر آن شد که ریکورد استریم توسط اپلیکیشن موبایل انجام شود. بنابراین رزبری‌پای یک استریم سرور است که بر روی پورت ۸۰۰۰ آن بالا می‌آید.

پیاده‌سازی سرور با استفاده از **socketserver** , **webserver** , **streamserver** پایتون پیاده شده‌اند.

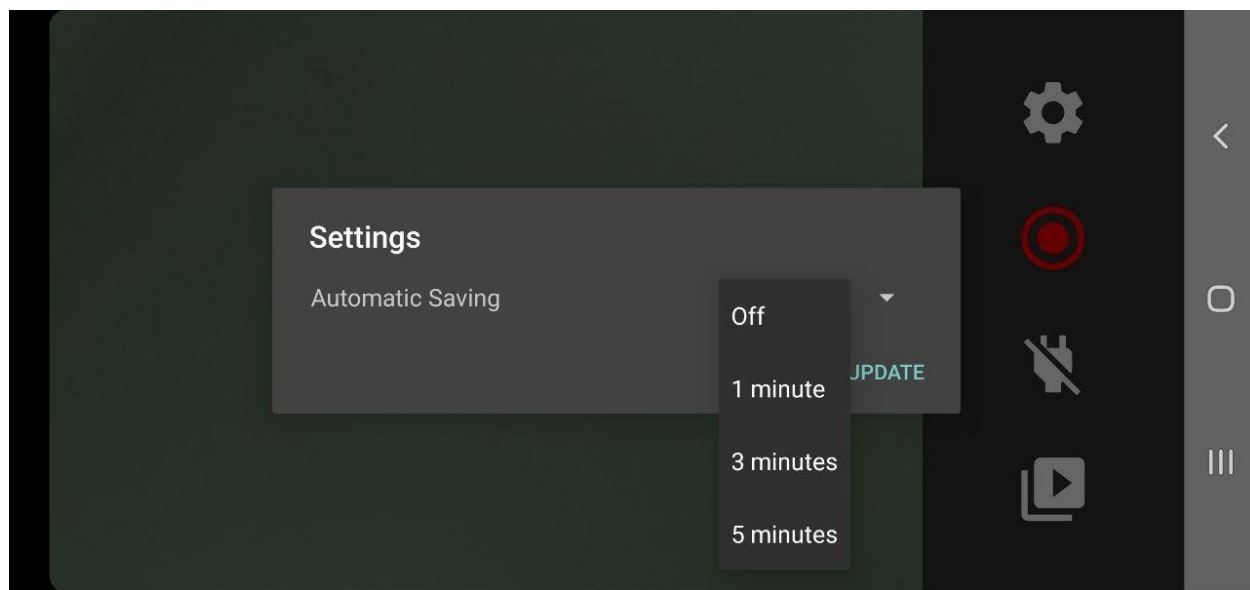
طراحی و پیاده‌سازی اپلیکیشن موبایل

نمای کلی برنامه به صورت مقابل است:



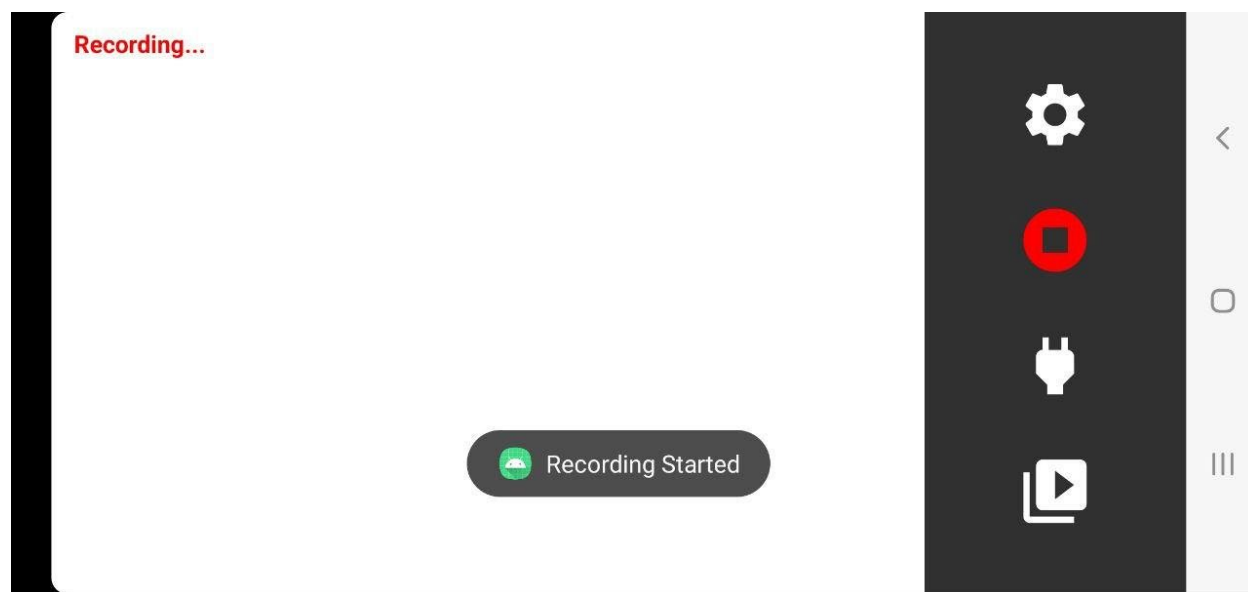
برنامه دارای ۴ دکمه‌ی اصلی است که در سمت راست تصویر بالا مشخص است. در ادامه به شرح وظایف هرکدام از این دکمه‌ها می‌پردازیم:

• دکمه‌ی تنظیمات :



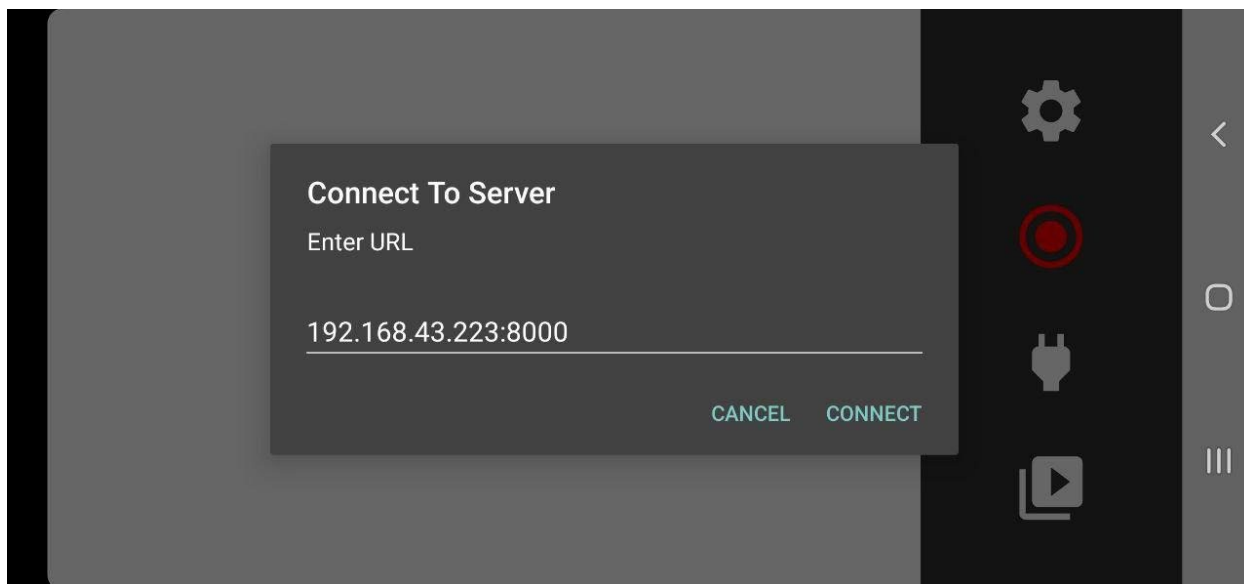
کاربر پس از فشار دادن این دکمه می‌تواند زمانی که می‌خواهیم رکورد کنیم را از بین گزینه‌های ۱، ۳ و ۵ دقیقه انتخاب کند، و اگر کاربر بخواهد می‌تواند با انتخاب گزینه‌ی Off به مقدار دلخواه صفحه را رکورد کند.

• دکمه‌ی رکورد:



پس از فشار دادن این دکمه یک Toast مبنی بر شروع شدن ضبط نشان داده می‌شود و تکست "Recording..." در بالای صفحه به نمایش در می‌آید. اگر کاربر زمان خاصی را انتخاب کرده باشد، رکوردینگ پس از زمان مشخص شده پایان می‌یابد و فایل در حافظه‌ی گوشی ذخیره می‌شود. دقت کنید در هر دو حالت (انتخاب زمان پیشفرض و عدم انتخاب زمان خاص) کاربر می‌تواند در صورت نیاز دوباره دکمه‌ی رکورد را بزند و رکوردینگ را به پایان برساند.

• دکمه‌ی Connect:



برای وارد کردن آدرس سرور به کاربر Popup، یک Connect هنگام فشردن شدن دکمه‌ی در هر وصل شدن عوض می‌شود، دقت کنید در صورت IP نمایش داده می‌شود. این عرضه‌ی این پروژه این حقیقت مشکل چندانی برای ما ایجاد نمی‌کند، زیرا در صورت برای مدت‌های بسیار IP عرضه شدن این پروژه سرور به ندرت خاموش می‌شود و ثابت برای هر کاربر با دریافت مبلغی IP طولانی ثابت می‌ماند، و یا حتی می‌توانیم یک اضافه تهیه کنیم.

• دکمه‌ی Files:

هنگام فشردن شدن دکمه‌ی Files، فولدیری که فیلم‌های ضبط شده در آن جا ذخیره شده‌اند برای کاربر باز می‌شود تا بتواند فایل مورد نظر خود را تماشا کند.

اجرای سرور در رزبری پای:

```

192.168.43.186 - - [23/Aug/2022 15:17:54] "GET /index.html HTTP/1.1" 200 -
192.168.43.186 - - [23/Aug/2022 15:17:54] "GET /stream.mjpg HTTP/1.1" 200 -
192.168.43.186 - - [23/Aug/2022 15:17:54] code 404, message Not Found
192.168.43.186 - - [23/Aug/2022 15:17:54] "GET /favicon.ico HTTP/1.1" 404 -
WARNING:root:Removed streaming client ('192.168.43.186', 45358): [Errno 32] Broken pipe
192.168.43.186 - - [23/Aug/2022 15:19:13] "GET /index.html HTTP/1.1" 200 -
192.168.43.186 - - [23/Aug/2022 15:19:13] "GET /stream.mjpg HTTP/1.1" 200 -

```

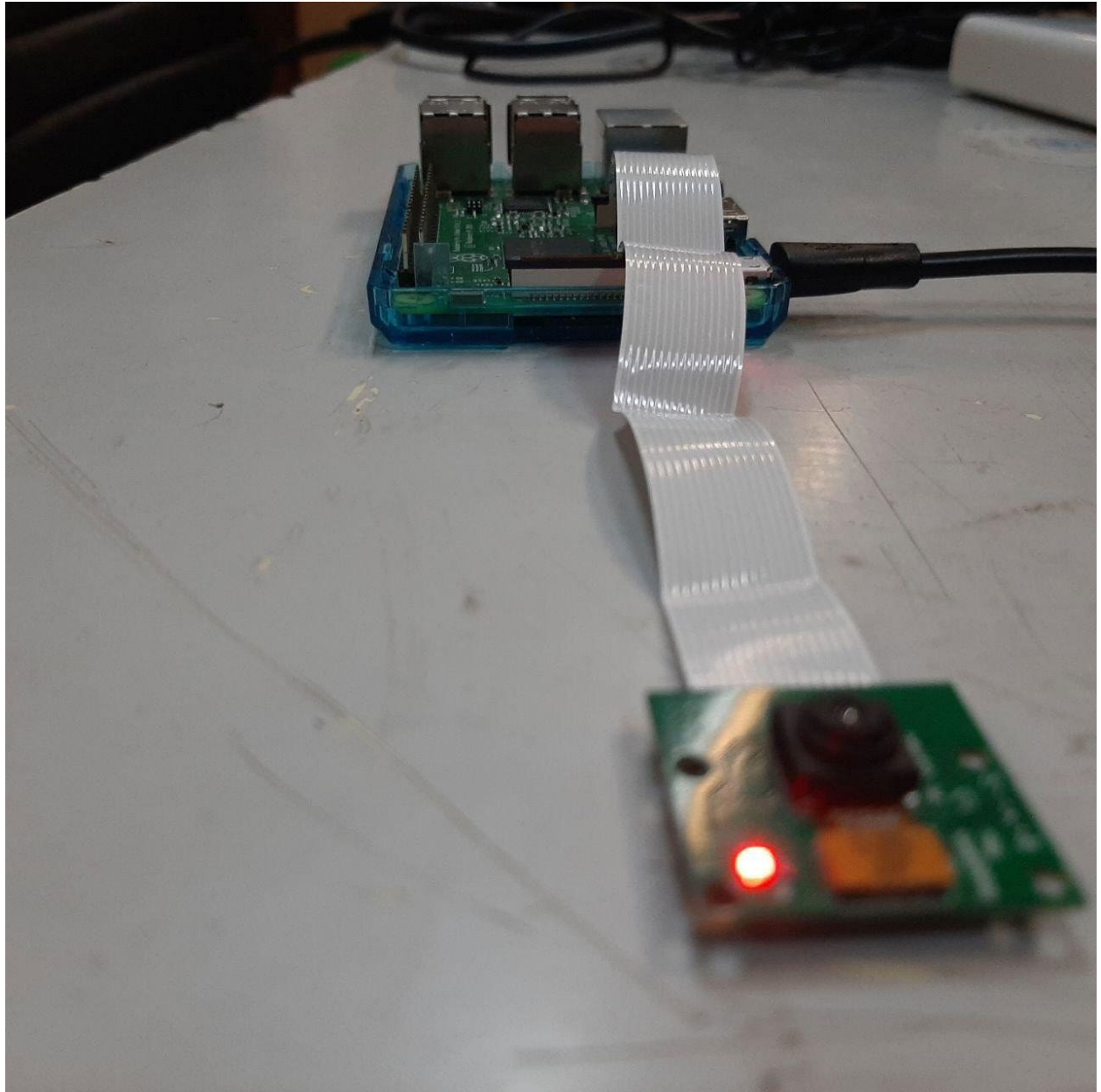
سرور اجرا شده در رزبری پای اطلاعات گرفته شده از دوربین رزبری پای را در بافر کرنل قرار این اطلاعات را از بافر می‌خواند و در قالب Streamer می‌دهد و در ادامه یک کلاس آن بافر ذکر شده است) قرار source (که img tag که دارای یک HTML یک فایل می‌دهد.

قیمت:

ردیف	قطعه	قیمت
۱	رزبری پای ۳	۴,۱۵۰,۰۰۰ تومان
۲	ماژول دوربین ۵ مگاپیکسلی	۱۵۹,۵۰۰ تومان
۳	جمع	۴,۳۰۹,۵۰۰ تومان

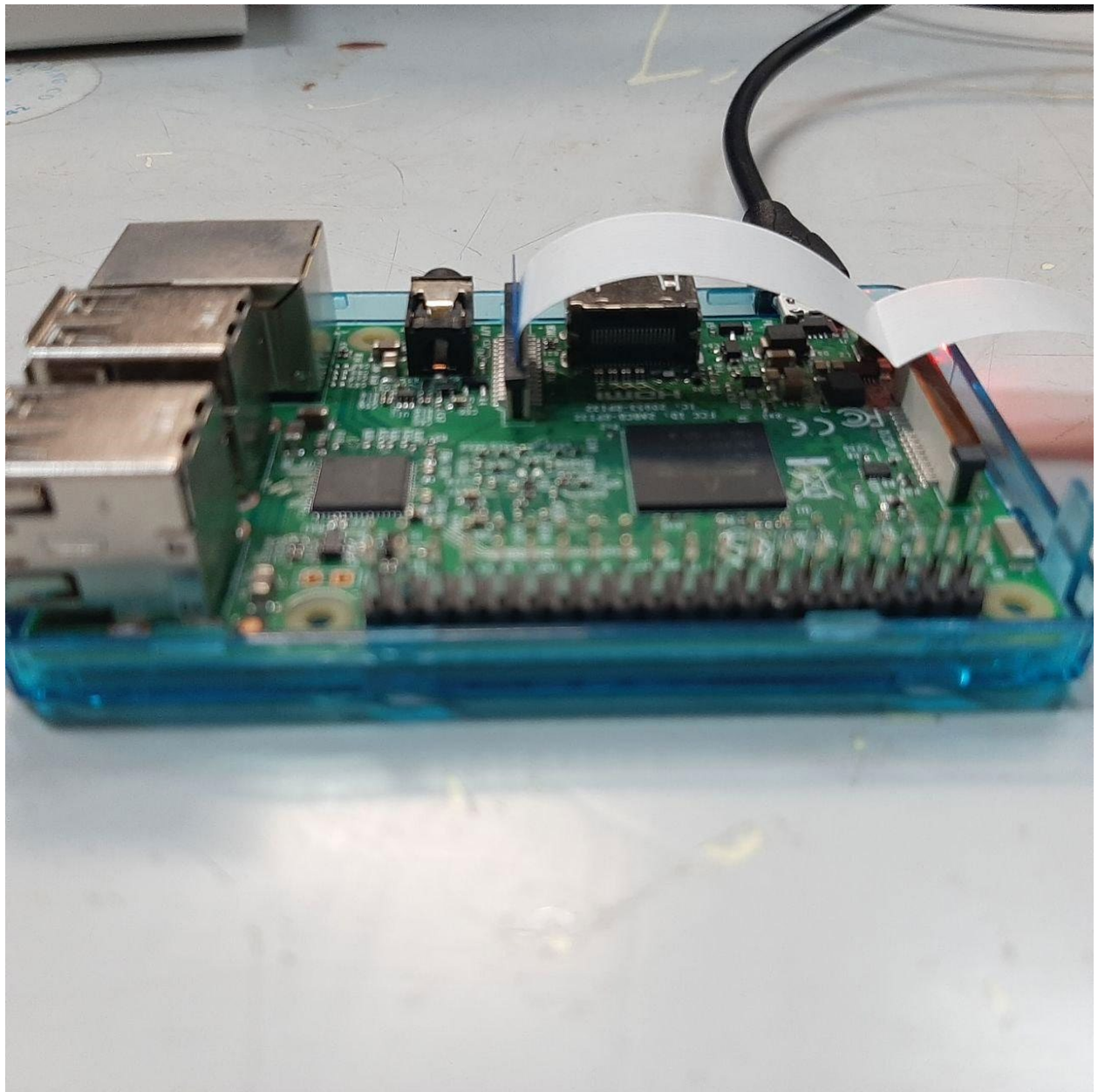
نماهایی از محصول:











فصل ۴:

جمع‌بندی:

در این پروژه به پیاده‌سازی یک سیستم هوشمند آینه‌ی پشت ماشین پرداختیم. این سیستم با استفاده از یک دوربین و یک سرور در رزبری پای و یک اپلیکیشن موبایل می‌تواند به راننده در حین رانندگی کمک کند تا در هنگام وقایع غیر مترقبه مانند مسدود شدن دید پشت توسط سرنشینان پشت یا صندوق عقب ماشین به پشت خود دید داشته باشد.

در کنار این محصول، یک اپلیکیشن موبایل فراهم گشته تا با استفاده از آن بتوانید از استریم فیلم تهیه کرده و آن را در شبکه‌های اجتماعی برای مخاطبان خود به اشتراک بذارید.

می‌توان با استفاده از ابزارهای یادگیری ماشین بر روی رزبری پای یک مدل تشخیص اشیاء تا بتواند به راننده هشدارهای لازم جهت جلوگیری از وقوع حادثه را دهد.