

به نام خدا

گزارش کار اسپرینت دوم پروژه شماره ۱۹ – دستیار تشخیص حداکثر سرعت مجاز

تیم شماره ۱

- یاشار ظروفچی
- امین مقرایی
- سپهر صفری

گزارش کار اسپرینت ۲

در پایان دو هفته دوم، موارد زیر به انجام رسید:

۱. طراحی وب سرور اولیه برای پیاده سازی api برای اپلیکیشن اندروید
۲. بررسی ماژول gps
۳. پیاده سازی اپلیکیشن اولیه اندروید

مورد اول:

در این مدت تیم ما یک وب سرور اولیه مبتنی بر فریم ورک جنگو طراحی کرد که وظیفه پاسخ گویی به درخواست های اپلیکیشن اندروید و همچنین پردازش اطلاعات دریافتی از سنسورها را برعهده دارد. ما در وب سرور قالب کلی نحوه طراحی api ها را مشخص کردیم و هم اکنون صرفا باید api ها را به وب سرور اضافه کنیم. قالب کلی پردازش اطلاعات سنسورها هم مشخص شد. برای این کار در وب سرور جنگو، علاوه بر ترد اصلی که سرور بر روی آن بالا است، یک ترد دیگر هم ساختیم که وظیفه این را دارد به طور موازی در بازه های زمانی ۱ ثانیه ای اطلاعات سنسورها یعنی از عکس از دوربین و مکان از gps را دریافت کرده و پردازش کند و نتیجه آن را در آرایه ای ذخیره نماید. این آرایه اطلاعات پردازش شده مانند سرعت خودرو و ... را در زمان های مختلف در خود دارد. در ادامه وقتی اپلیکیشن به وب سرور درخواست بدهد، این اطلاعات در اختیار آن قرار می گیرد.

مورد دوم:

داده گیری از ماژول GPS

این بخش به ۳ مرحله تقسیم می شود

۱. اتصال ماژول. با توجه به عدم در دسترس بودن خروجی (USB بر خلاف مدل (AdaFruit در نتیجه اتصالات به این ماژول لحیم شده اند. با توجه به دستورالعمل های به دست آمده GPIO14,15 برای اتصال به TX, RX ماژول GPS هستند. سپس در بخش تنظیمات Raspberry واسط Serial را فعال می کنیم و سپس با دستور cat /dev/serial0 بررسی می کنیم که آیا اساسا اتصالی وجود دارد یا خیر.
۲. مانیتورینگ GPS. برای انجام این بخش ابتدا لازم است تا ماژول و کلاینت gpsd را نصب کنیم. سپس برای دریافت موقعیت فعلی ابتدا می بایست با دستورات زیر سرویس را متوقف کنیم.

```
sudo systemctl stop gpsd.socket  
sudo systemctl disable gpsd.socket
```

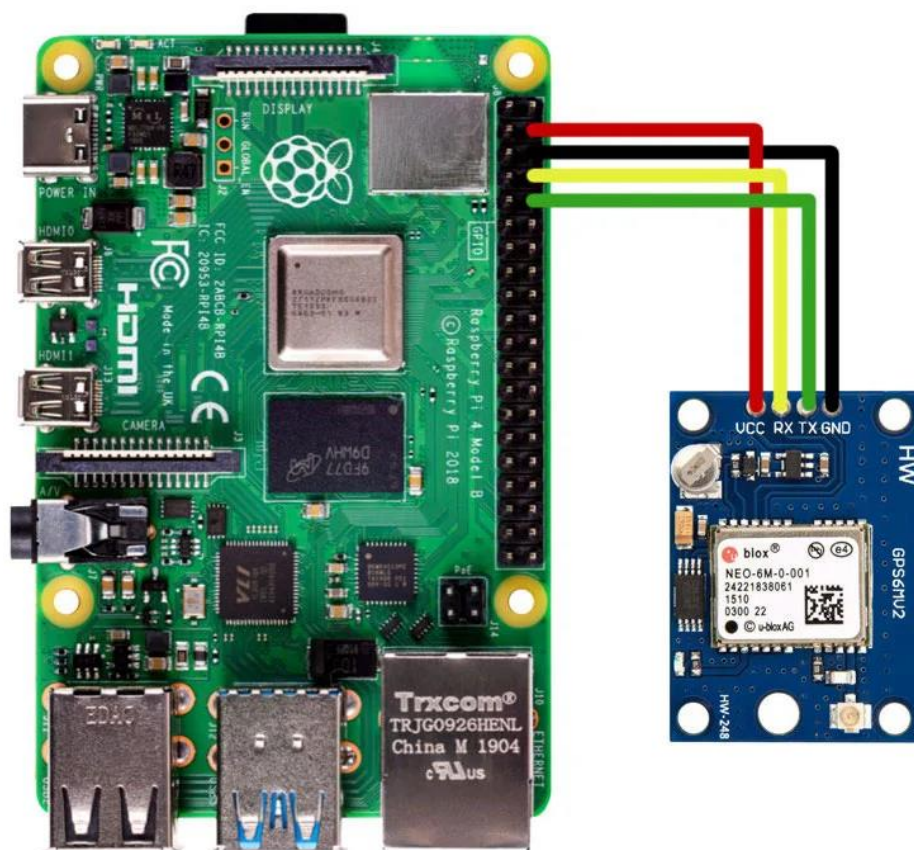
سپس سرویس را مجدداً بالا می‌آوریم

```
sudo gpsd /dev/serial0 -F /var/run/gpsd.socket
```

برای مانیتورینگ دو ابزار `gpsmon` و `cgps` وجود دارد.

۳. اندازه‌گیری سرعت. داده‌های خروجی `GPS` فرمت مشخصی دارند که در آن طول و عرض جغرافیایی مشخص شده است. در `repository` پروژه، کد مربوط به `GPS` وجود دارد که این اطلاعات را تفکیک می‌کند. خروجی این تکه کد توسط وب‌سرور موجود دریافت شده و با بررسی اختلاف مقادیر جغرافیایی سرعت به دست می‌آید. چالش‌های پیش‌آمده.

ماژول `GPS` مطابق اتصالات زیر بسته شد.



منبع: [پیوند](#)

لکن با بررسی GPS در سه محیط اتاق، بالکن و حیاط (که فضای کاملاً بازی است) مانیتورینگ‌ها با خطای No FIX found مواجه می‌شوند. در اینجا سه احتمال مطرح است

- ماژول GPS ضعیف‌تر چیزی است که انتظار می‌رود - با توجه به اینکه ویدیوهای بررسی شده که با کابل‌های بلند ماژول در بیرون آزمایشگاه گذاشته شده و نتایج در داخل تحلیل و بررسی می‌شوند دور از انتظار است که این ماژول در فضای باز کار نکند.
- ماژول دریافت ایراد سخت‌افزاری دارد
- به هر دلیلی چون اتصالات این ماژول GPS صرفاً مبتنی بر داده‌های ماهواره‌ای است (برخلاف گوشی‌های هوشمند) ممکن است این سیگنال‌ها در داخل ایران ضعیف باشد.

مورد سوم:

برای پیاده سازی نسخه‌ی اولیه از اپلیکیشن اندروید تصمیم گرفتیم تمام توجه خود را سمت backend سیستم بگذاریم و با توجه به این نکته که در محصول نهایی باید ساز و کاری فراهم کنیم تا اپلیکیشن اندروید بتواند با رزبری ارتباط بگیرد و یکسری اطلاعات شامل سرعت خودرو در زمان‌های مختلف را دریافت کند، پس تصمیم ما نیز بر آن شد تا برای پیشبرد بهتر و دقیق‌تر کار ابتدا یک سرور رست جنگو بالا بیاوریم و همین فرآیند ارتباط و تبادل اطلاعات را یک دور بین اپ اندروید و این وب سرور طی کنیم و سپس کار را به محصول اصلی منتقل کنیم.

برای این منظور همانطور که در مخزن مربوطه می‌بینید یک اپ نسبتاً ساده‌ی اندروید زده شد که به یک سرور جنگو دسترسی پیدا می‌کند و توسط تابع Get موجود در آن وب سرور دیتای موجود در آن صفحه را می‌گیرد و بر روی صفحه‌ی اولیه‌ی این اپ نمایش داده می‌شود.

پیاده‌سازی اندروید:

سرور مذکور را بر روی یک دامنه‌ی مشترک بالا می‌آوریم و از طریق گوشی نیز به همان دامنه متصل می‌شویم؛ لینک مربوطه را در یک متغیر جهانی url نگه می‌داریم؛ در تابع doInBackground به این url ریکوئست Get ارسال می‌شود و تمام فیلدهای موجود در صفحه‌ی post مربوط به وب سرور ما در یک بافر بصورت داده‌ی JSON ریخته‌شده و برای کد اندروید ارسال می‌شود. در انتهای این تابع نیز تنها محتویات بافر را به داده‌ی String تبدیل می‌کنیم. در تابع execute مربوطه نیز این داده‌های متنی گرفته شده و هریک از فیلدهای آن پست را که خواستیم انتخاب می‌کنیم و در اپ اندروید نمایش می‌دهیم.

پیاده‌سازی وب سرور:

برای بالا آوردن یک سرور جنگو از سایت <https://www.pythonanywhere.com/> استفاده کرده و یک وب سرور ابتدایی شامل صفحه‌ی مربوط به Postها را تشکیل دادیم، هر پست هم دو فیلد name و text دارد.