



دانشگاه تهران

پردیس دانشکده های فنی

دانشکده مهندسی نقشه برداری

پروژه دوم برنامه نویسی درس خدمات مکان مبنا

AVL

مهیار داوطلب 810302025

استاد :

جناب آقای دکتر عباسپور

نیم سال دوم سال تحصیلی 1402-

1403

قسمت مربوط به بخش اندروید برنامه نوشته شده

در این پروژه لازم است اپلیکیشن نوشته شده در اندروید استودیو را به یک سرور دلخواه متصل کرده و یکسری اطلاعات مانند موقعیت کاربر روی OSM, درصد باتری گوشی، ایدی-نامبر، مدل گوشی، سرعت کاربر را بر روی نقشه ای که روی سرور قرار بفرستیم و چند آنالیز را انجام دهیم. ما در این پروژه از سرور Django که بر روی پورت و دامنه 192.168.43.107:8000 سیستم قرار دارد استفاده کرده ایم.

در بخش کد جاوا دو activity به نام های activity main و Map_activity ایجاد نموده ایم. کلاس mainactivity بعد از 5 ثانیه ما را به اکتیویتی اصلی برنامه هدایت می کند. در قسمت Map_activity اطلاعات موقعیت و اطلاعات گوشی از طریق Jason به سمت سرور ارسال شد.



قسمت مربوط به بخش وب سایت برنامه نوشته شده

در این قسمت ابتدا تنظیمات IP را در بخش setting اعمال می کنیم. همانطور که در بخش قبل اشاره شد درخواست از طریق دامنه وپورت 192.168.43.107 به سرور ارسال می شود پس می بایست این قسمت را به Setting جنگو معرفی کنیم.

تنظیم بخش MODEL

یک کلاس تعریف می کنیم و اطلاعات مربوط به Lon، lat، سرعت، درصد باتری، را در آن تعریف می کنیم. این بخش در واقع برای تعامل با دیتابیس ایجاد می شود.

تنظیم بخش view

بخش View جایی است که می توانیم توابع مورد نیاز برای آنالیز های برنامه را در آن بنویسیم. در این قسمت تابعی برای دریافت اطلاعات اندروید به صورت Jason تعبیه شده است. اگر request به درستی ارسال شود response=200 و در غیر اینصورت response=400 را دریافت می کند. تابعی برای ضبط مسیر به صورت Start/stop فراهم شده که با زدن دکمه Stop فایل مسیر به صورت شیپ فایل در مسیر برنامه به صورت خوردکار ذخیره می شود. در این قسمت تابعی برای رندر کردن نقشه نیز استفاده شده است.

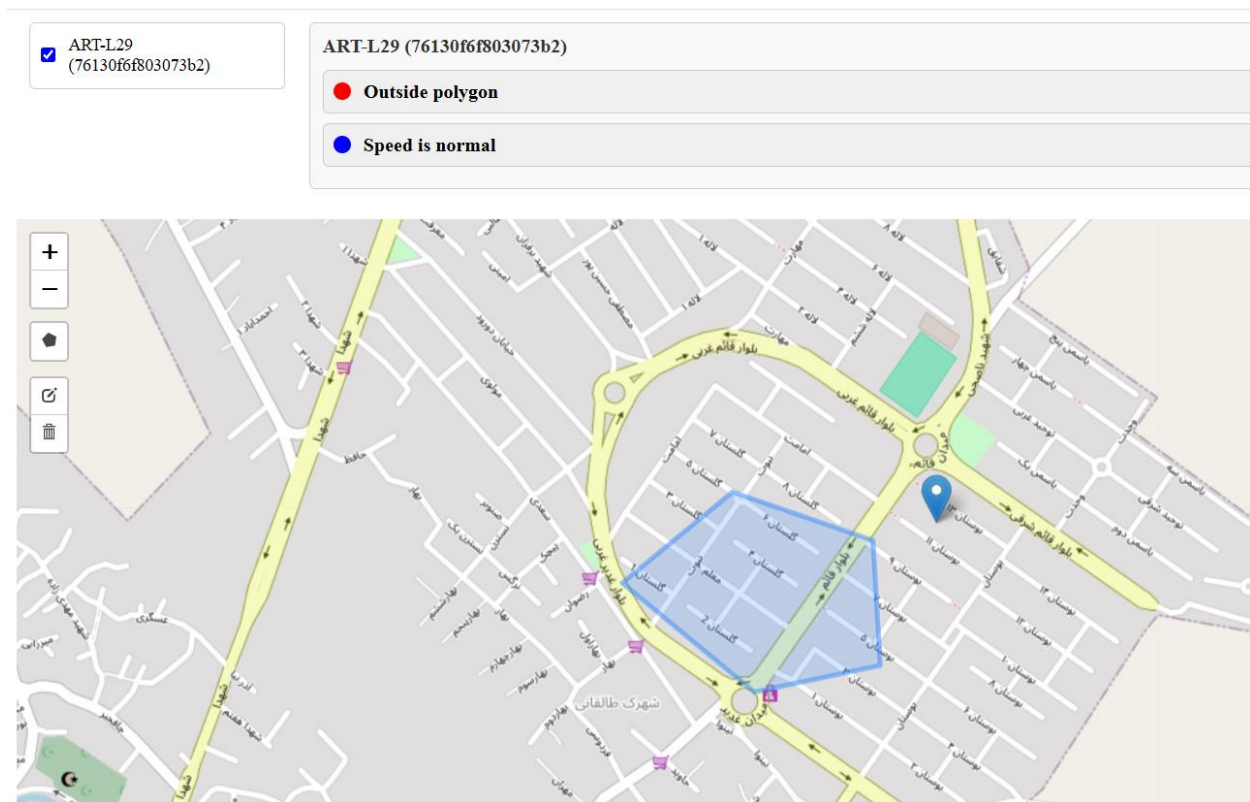
بخش shapefile

یک فایل برای درست کردن شیپ فایل مسیر در مسیر app برنامه درست کردیم که فایل های dbf،.shx،.shp را با استفاده از فایل ضمیمه شده ESRI به صورت دستی

و بدون استفاده از کتابخانه در ان نوشته می شود و در نهایت این فایل را به بخش View معرفی می کنیم تا با دکمه های start/stop وضبط مسیر در تعامل باشد.

تنظیم بخش template

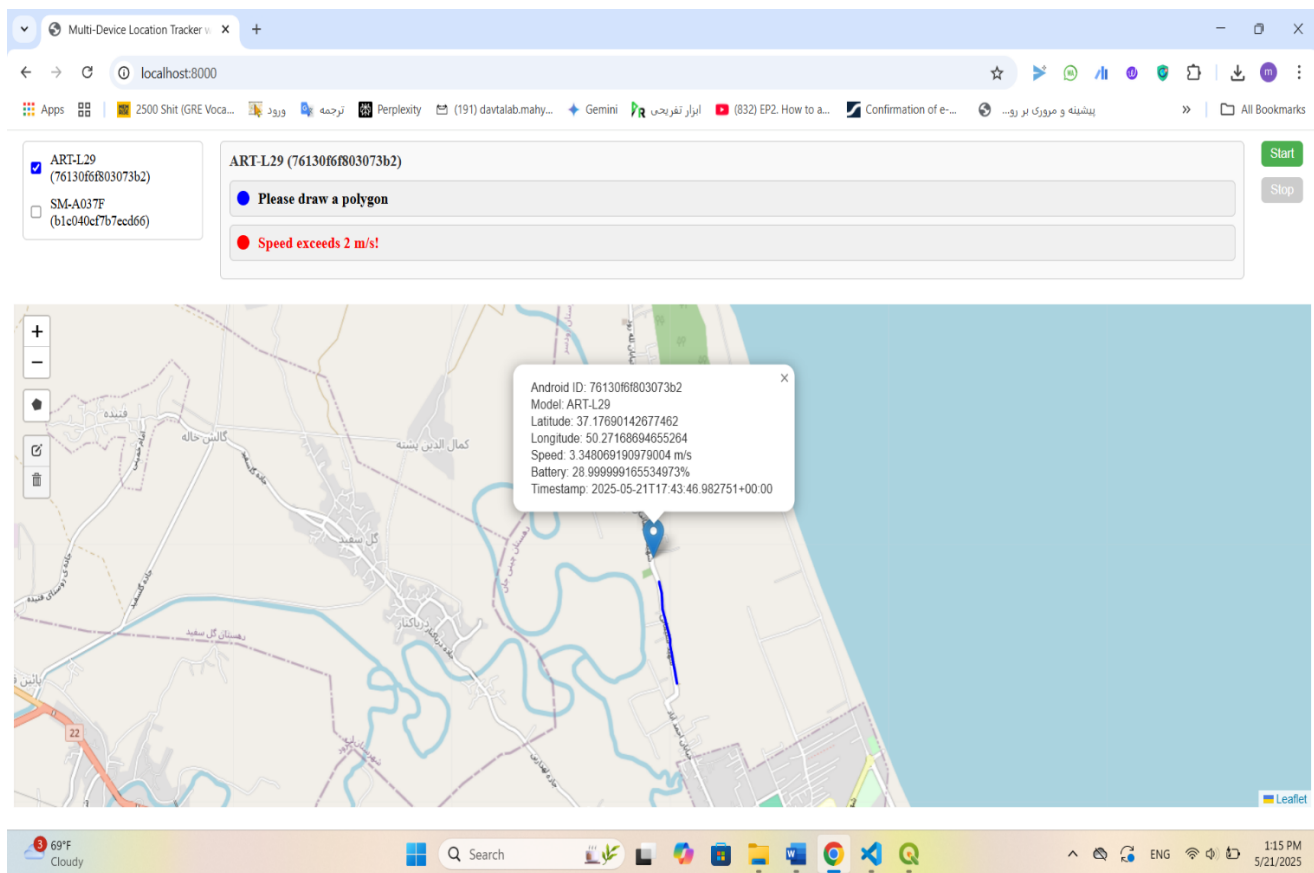
یک فولدر به نام templates ایجاد کرده و فایل index.Html را در ان بارگذاری می کنیم. این فایل امکان ردیابی دستگاه های اندرویدی را به صورت Real-time فراهم می کند. این فایل شامل کتابخانه متن باز Leaflet، ابزارهای کشیدنی Leaflet، کاشی های OSM، فایل های جاوااسکریپت و CSS است. در این فایل یک چک باکس برای نمایش و فعال سازی یا عدم فعال سازی دستگاه هایی که ارسال موقعیت کرده اند فراهم شده است که از این طریق کاربر می تواند انتخاب کند که اطلاعات کدام دستگاه بر روی نقشه نمایش داده شود. همچنین دکمه های ضبط مسیر به صورت شیب فایل در این فایل آورده شده که با بخش View در تعامل است و کاربر می تواند مسیر track دستگاه ها را با این دکمه ها تحت کنترل بگیرد. دو بخش جاوا اسکریپت برای چک کردن سرعت و چک کردن نقطه درون یک پلیگون تعبیه شده که اگر سرعت از یک حدی بیشتر شود (مثلا 2 متر بر ثانیه در حالت پیاده) خطاری در باکس مربوط به ان دستگاه اندرویدی نمایش داده می شود همچنین در قسمت انالیز Point In Polygon یک باکسی تعبیه شده که اگر موقعیت کاربر از پولیگونی که در نقشه کشیده می شود خارج شود به کاربر خطاری داده می شود.



انالیز point in polygon

بخش URL

در این بخش URL های موجود برای فراخوانی توابع یاد شده در بخش view آورده می شود. در جنگو می توان با دستور include می توان تمام URL هایی که برای توابع مختلف در اپ اصلی برنامه نوشتیم را به صورت یکجا در Localhost نمایش دهیم. در قسمت نهایی نمایی از کد در محیط سرور ملاحظه می شود.

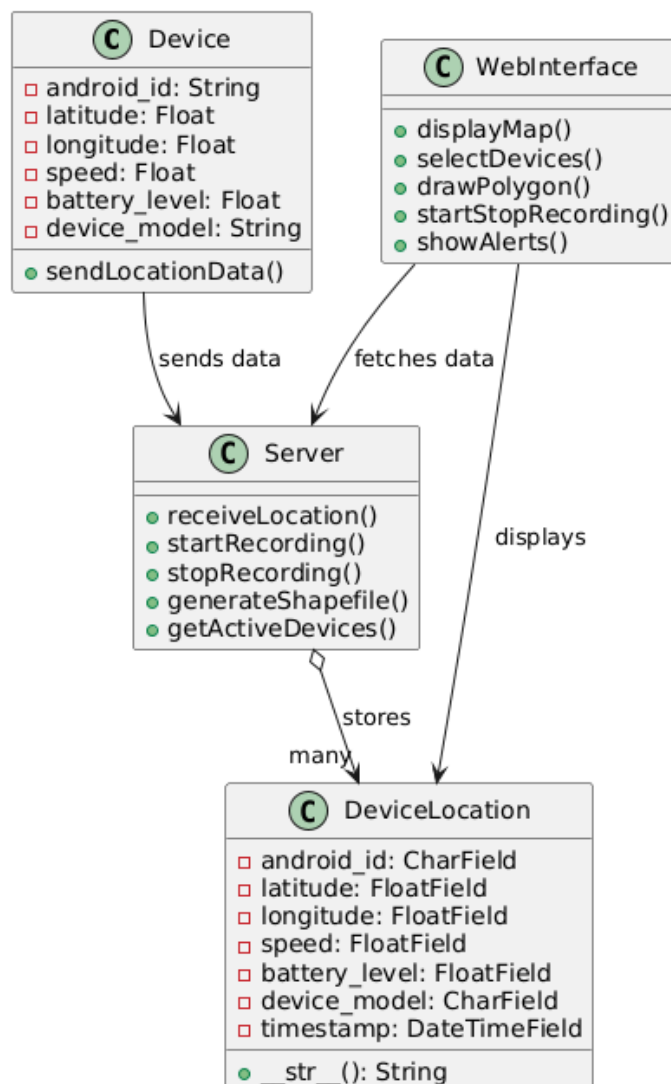


شمای کلی برنامه در محیط سرور

UML Diagram

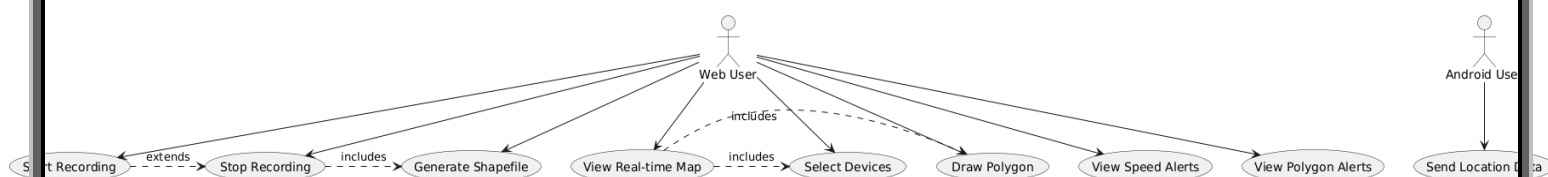
Class_Diagram

با این دیاگرام می توانیم کلاس هایی که در طول پروژه چه در بخش اندروید و چه بخش سرور و ایندکس ایجاد کردیم خیلی ساده نشان دهیم. 4 کلاس اصلی داشتیم. یکی سمت اندروید که اطلاعات ارسال می کرد (Device)، یک کلاس برای تعامل با دیتابیس که در بخش مدل تعریف می شد (DeviceLocation)، توابعی که در بخش View و سمت جنگو داشتیم را می توانیم به صورت یک کلاس با نام Server در این قسمت ر در نظر بگیریم و یک کلاس هم برای تعاملات کاربر با نقشه که با نام index در جنگو می شناختیم (Webinterface).



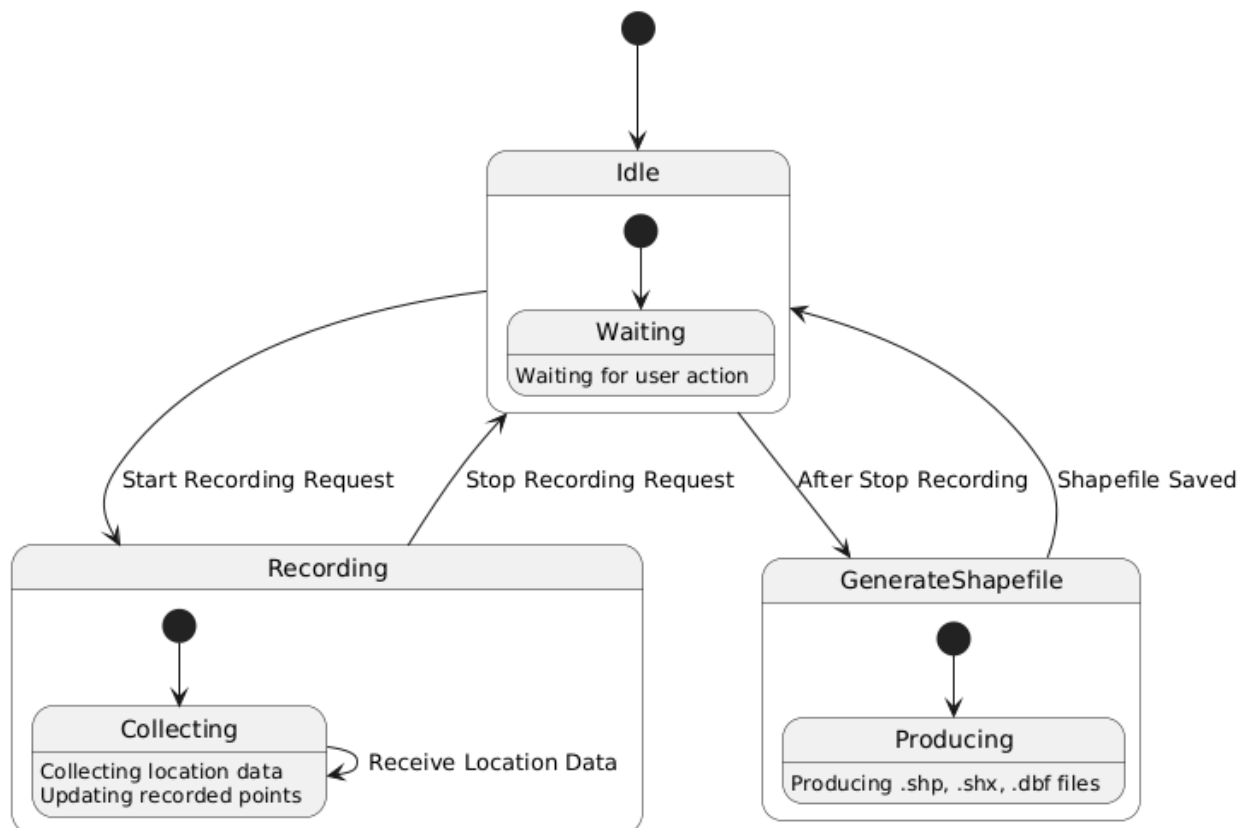
Use_case Diagram

با این دیاگرام می توان تعاملات Actor ها با سیستم را نشان دهیم. 2 اکتور اصلی داریم. Actor اول در یک طرف در سمت اندروید اطلاعات را به سمت سرور میفرستد و Actor دیگر در سمت سرور اطلاعات را دریافت می کند و میتواند یکسری عملیات مثل تولید شیپ فایل ،انالیز سرعت و ... را انجام دهد.



State_Diagram

با این دیاگرام می توان نشان داد که سیستم تو چه وضعیت هایی می تواند باشد. وضعیت اصلی برای پروژه ما حالت های on/off کردن شیپ فایل و ضبط نقاط و ذخیره فایل های مربوط به شیپ فایل بود. در حالت off یا idle فقط اطلاعات دریافتی را می توان روی نقشه نمایش داد و به محض فعال شدن فرایند برداشت نقاط شروع شده و بعد از استاپ کردن شیپ فایل نقاط ساخته شده و دوباره به حالت اول یا وضعیت idle بر می گردد



Activity_ Diagram

هدف این دیاگرام توضیح فعالیت های اصلی پروژه است. در واقع با این دیاگرام جریانات اصلی برنامه که شامل ارسال اطلاعات از گوشی، دریافت اطلاعات در سمت سرور، ضبط مسیر، تولید شیپ فایل، نمایش، و هشدارهای لازم را می توانیم نمایش دهیم.

