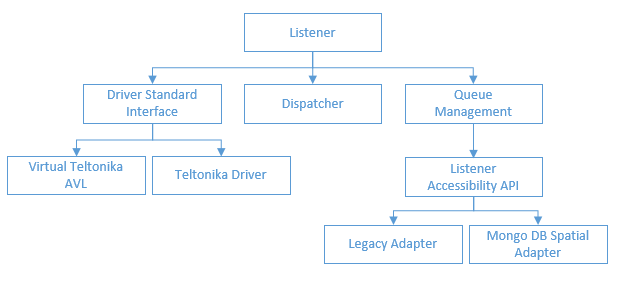
**سامانه کنترل دستگاه های AVL**

بر اساس طراحی مورد نظر این نرم افزار در چهار بخش کلی قابل تعریف و تحویل می باشد

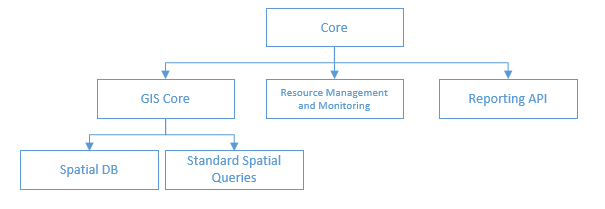
* طراحی و پیاده سازی Listener
* طراحی و پیاده سازی Core & DB
* طراحی و پیاده سازی Web Portal
* طراحی و پیاده سازی Fleet Management

1. **طراحی و پیاده سازی Listener:**

این فاز شامل موارد زیر میباشد

* Driver Standard Interface: چارچوب ارتباط با دستگاه های مختلف را مشخص می کند و به عنوان یک لایه استاندارد برای پیاده سازی درایور های دستگاه ها با پروتکل های مختلف مورد استفاده قرار می گیرد.
* Dispatcher: به عنوان سرور TCP بر پایه تکنولوژی Async I/O که موجب فشار کمتر بر روی سرور و امکان پاسخ گویی همزمان به Client های بیشتری نسبت به روش های مرسوم می دهد.
* Queue Management: برای افزایش قابلیت اطمینان و Reliability سیستم در این بخش از تکنولوژی Message Queue استفاده می شود که موجب کمتر شدن coupling بین سرویس های مختلف نرم افزار و همین طور افزایش Scalability سیستم می شود.
* Teltonika Driver : به عنوان درایور اصلی در سیستم Listener ارائه می گردد
* Virtual Teltonika AVL : دستگاه مجازی بر اساس پروتکل Teltonika FMXXX برای تست Listener تحت شرایط مختلف ( تعداد دستگاه های مختلف، مشکل در ارتباط، بار گذاری با فشار های مختلف بر روی Listener، آلارم و ...)
* Listener Accessibility API: لایه ارتباطی سبک بر روی سیستم Messaging در Listener برای ارتباط با سرویس های و Component های دیگر سیستم
* Legacy Adapter: آداپتور ارتباطی برای Listener جدید به دیتابیس سیستم فعلی برای سهولت در تغییر به سیستم جدید.
* Mongo DB Spatial Adapter: آداپتور ارتباطی برای انتقال اطلاعات دریافتی از Listener در قالب اطلاعات استاندار GIS در دیتابیس Mongo در حالت Spatial

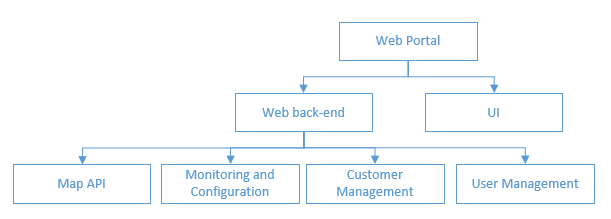
1. **طراحی و پیاده سازی Core:**



این فاز شامل موارد زیر میباشد

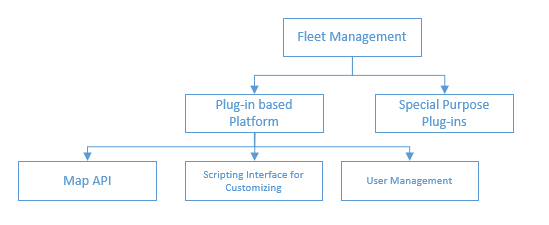
* GIS Core: لایه اصلی برای مدیریت داده ها و ارتباط با پایگاه داده Spatial
* Spatial DB: ارتباط و مدیریت با پایگاه داده Spatial در این Component پیاده سازی می شود
* Standard Spatial Queries: در این لایه Query های معمول برای بازیابی داده ها ( سرعت متوسط، مسافت، جهت حرکت، مسیر و ...) پیاده سازی می شوند و همینطور یک رابط استاندارد برای پیاده سازی Query های جدید قرار داده می شود.
* Resource Management and Monitoring: مدیریت و مانیتورینگ دستگاه های موجود در سیستم ، قابلیت اضافه، حذف و تغییر دستگاه در سیستم
* Reporting API: قابلیت گزارش گیری بر اساس داده های spatial موجود در پایگاه داده، این Component قابلیت ایجاد گزارش های جدید بر اساس Query های تعریف شده در سیستم را دارا می باشد.

1. **طراحی و پیاده سازی Web Portal:**



این فاز شامل موارد زیر میباشد

* Web back-end: سرور اصلی portal که بر اساس تکنولوژی Rest پیاده سازی می شود.
* Map API: نمایش داده های گزارش گیری شده از پایگاه داده سیستم با استفاده از امکانات موجود یکی از نقشه های موجود (Google Maps, Bing Maps, Open Street, Nokia Here )
* Monitoring and Configuration: مانیتورینگ و تنظیم دستگاه ها بر اساس دید مشتری و ادوات تجهیز شده
* Customer Management: مدیریت حساب مشتریان به صورت فردی، شرکتی و گروهی
* User Management: مدیریت کاربران، سطح دسترسی، وضعیت حساب و کنترل آن
* UI: رابط کاربری تحت وب به صورت Single page application برای سهولت استفاده و Responsive بودن آن.

1. **طراحی و پیاده سازی Fleet Management:**

این فاز شامل موارد زیر میباشد

* Plug-in based Platform: بستر نرم افزار بر اساس تکنولوژی های Plug-in based برای سهولت در شخصی سازی سیستم.
* Special Purpose Plug-ins: برای سیستم های خاص منظوره مانند اورژانس و گشت پلیس که این امکانات بدون نیاز به تغییر در کد بر روی بستر فعال می گردند.
* Map API: نمایش داده ها، وضعیت ها و گزارشات سیستم بر روی یکی از نقشه های موجود
* Scripting Interface for Customizing: قابلیت تغییر در بستر و شخصی سازی های دقیق تر با استفاده از زبان های Scripting موجود بر روی بستر
* User Management: مدیریت کاربران، سطح دسترسی، وضعیت حساب و کنترل آن

**زمان بندی:**

زمان بندی برای پیاده سازی نرم افزار. هزینه هر ساعت برنامه نویسی بر اساس تعاریف نظام صنفی رایانه ای کشور سال 1391 در نظر گرفته شده است.

|  |  |  |
| --- | --- | --- |
| **نام** | **میزان نفر ساعت** | **مبلغ** |
| طراحی و پیاده سازی Listener | 4 – 6 هفته | 80000000 |
| طراحی و پیاده سازی Core & DB | 6 تا 8 هفته | 160000000 |
| طراحی و پیاده سازی Web Portal | 4 هفته | 100000000 |
| طراحی و پیاده سازی Fleet Management | 9 تا 11 هفته | 160000000 |