FLOE NAVIGATION SYSTEM

DEVELOPER GUIDE

RELEASE 1.0.0

JANUARY 2019

Nadeem Gul
Rintu Daniel

Contents

Preface:

Welcome to the Floe Navigation Android Application. The application can be installed on any Android tablet (For the MOSAiC expedition the tablet being used is XSLATE D10). This application uses the periodic data from AIS transponders installed on the Sea Ice to create a coordinate system which is fixed on a moving ice floe. It creates a visual representation of the coordinate system in the form of a grid which can be used to navigate on a moving Sea Ice.

Please read through this document thoroughly before you start to customize the Floe Navigation App. The purpose of this guide is to provide developers with the necessary information about the tools required for the customization of the Floe Navigation Android App.

Audience:

This document is intended for Developers who intend to customize or extend the Floe Navigation App. Please note that this document can only be used for setting up the development environment and getting a basic understanding of the Floe Navigation App, for actual code customization you must go through the code documentation.

Related Documents:

For more information, see the following documents:

- Floe Navigation User Guide
- Floe Navigation Administrator Guide

1 Getting Started

To get started with customizing the Floe Navigation App it is recommended to have the following resources installed:

Application	**Version****
Java JRE	1.8.0_152
Android Studio	3.2.1
Eclipse IDE*	Neon.3 Release (4.6.3)
MySQL*	8.0
Postman*	6.6.0
Microsoft Internet Information Services*	10.0.17134.1

* Used only for Synchronization. For changes in the App which do not affect the Synchronization process, you will only need Android Studio.

** Versions used for developing Floe Navigation App version 1.0.0.

In addition to the above resources please note that the App needs an AIS Transponder to run. For testing purposes ensure that the tablet is connected to the Wi-Fi network of an AIS transponder. As Android does not support multiple network connections you can use Bluetooth tethering for debugging and installation purposes; therefore, the Development Machine must have a Bluetooth adapter.

2 Development Environment

The Floe Navigation System consists of the Floe Navigation App and the Synchronization Server; each of which is developed in a separate environment. The Floe Navigation App is developed in Android and the Synchronization Server consists of a MySQL Database and PHP based Webservices.

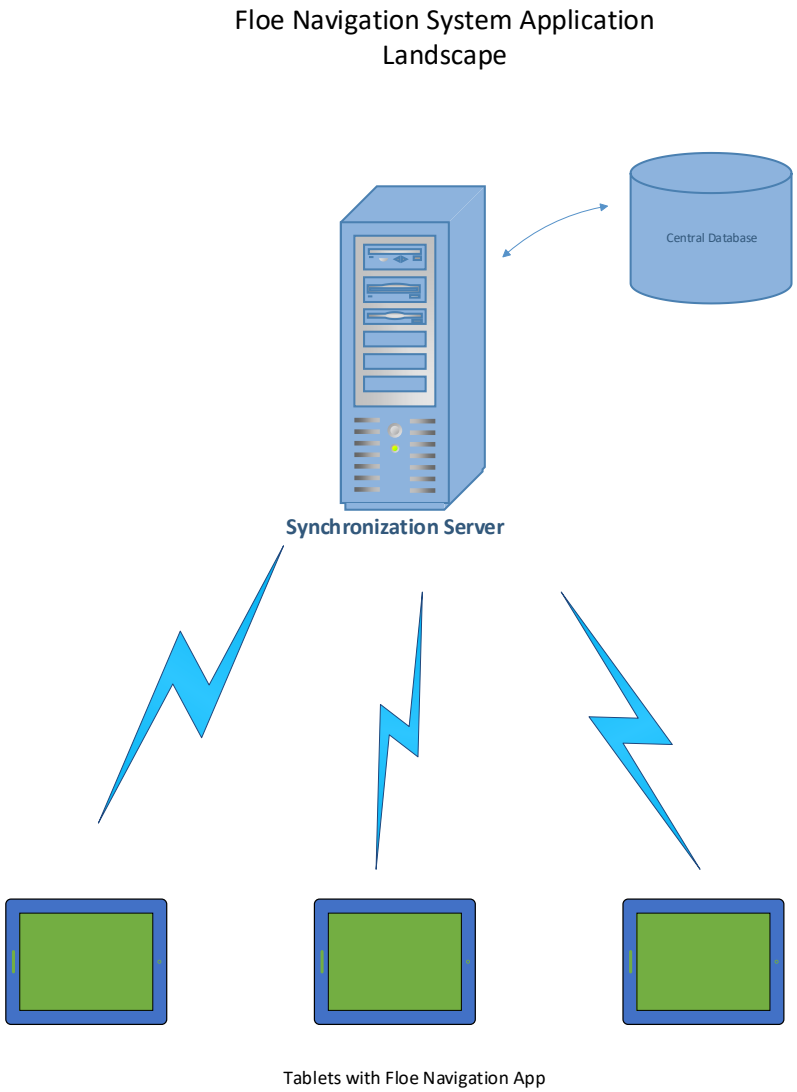


Figure 2.1 Floe Navigation Application Landscape

2.1 Setting Up Android Development Environment

The Floe Navigation App was implemented using Android Studio so any new customizations in the App must be done using the same Android Studio Project. The Android development environment is connected to the tablet using Bluetooth tethering (as mentioned in Chapter 1). This gives you the capability to deploy your code instantly to the tablet, debug it and check any log messages.

For instructions on installation and use of Android Studio check Android Documentation.

To set up Android Debugging over Bluetooth on the XSLATE D10 tablet follow these steps:

1. Enable Developer Options and Android Debugging on the tablet by following the steps mentioned here.
2. Connect your PC to the tablet via Ethernet and assign Static IPs to both the tablet and your PC.

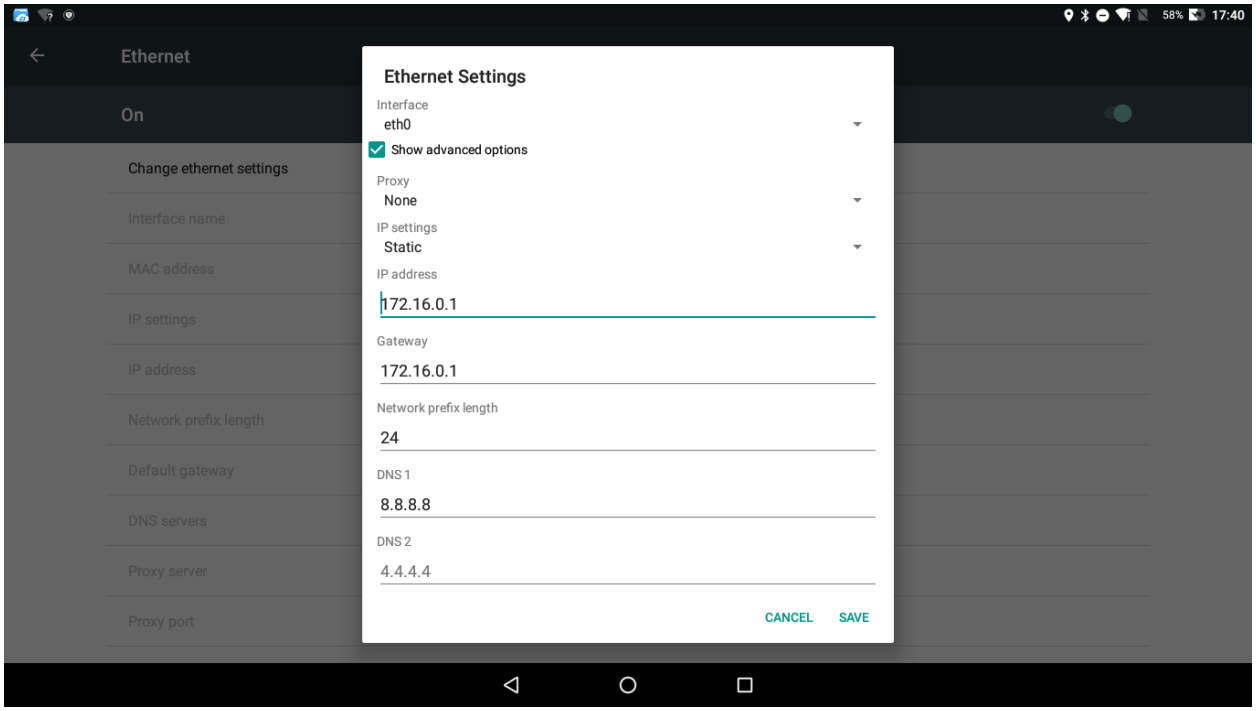


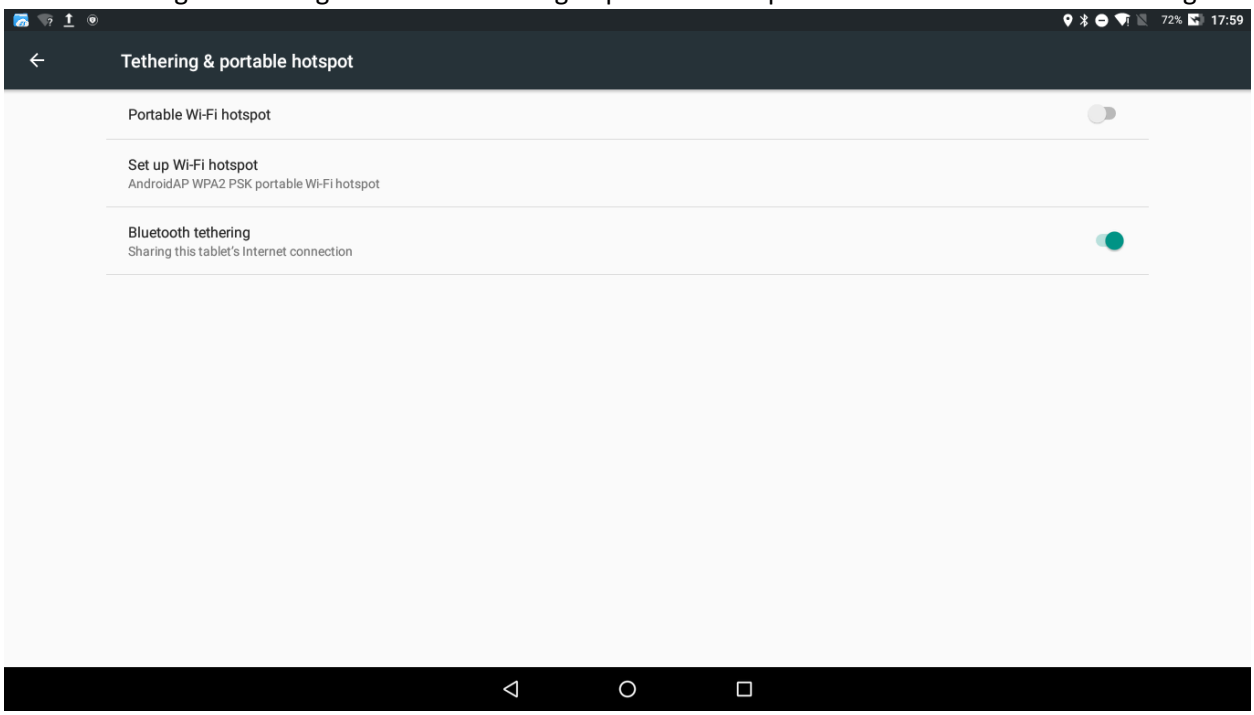
Figure 2.2 Assigning Static IP to Ethernet Connection on the Tablet

3. Open a Command Prompt window on your PC and start Android Debugging Bridge (ADB) in TCP IP mode by entering the following commands (Enter the IP of the tablet):
   >adb connect ***.***.***.***

   >adb tcpip 4455

4. Enable Bluetooth on the tablet and pair the tablet with your PC.

5. On the tablet go to Settings->More->Tethering & portable hotspot and enable Bluetooth tethering.



6. On your PC go to Control Panel -> Devices and Printers and right on the tablet and click on Connect Using -> Access Point.

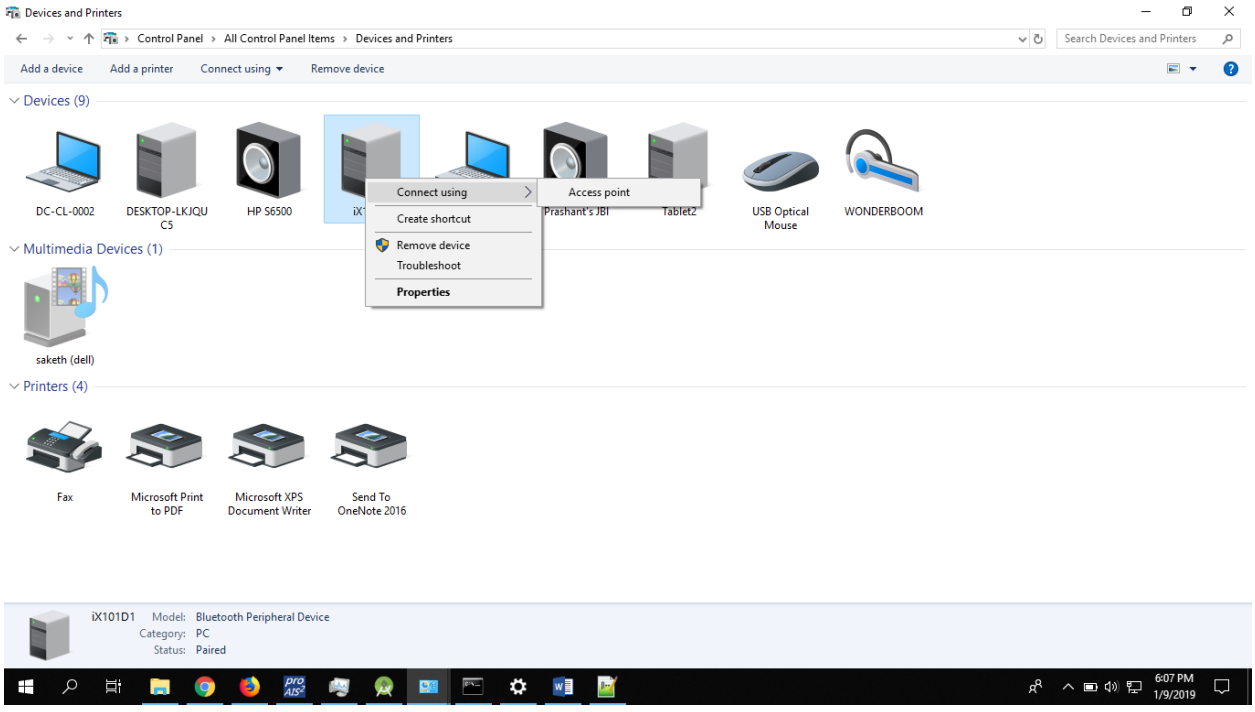


Figure 2.3 Connecting to the Tablet's Bluetooth Network

7. Now go back to the command prompt and enter:
>adb connect 192.168.44.1:4455

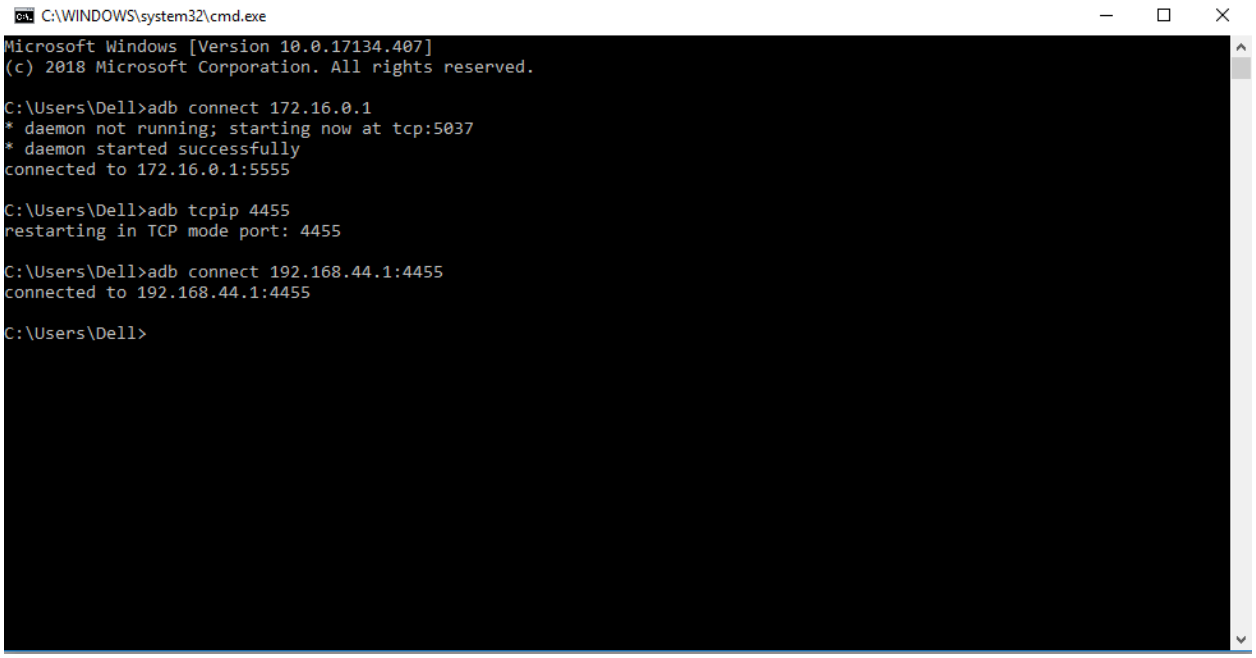


Figure 2.4 Adb Connection Over Bluetooth using Command Prompt

The PC is now connected to the tablet and you can deploy your App to the tablet and/or debug it.

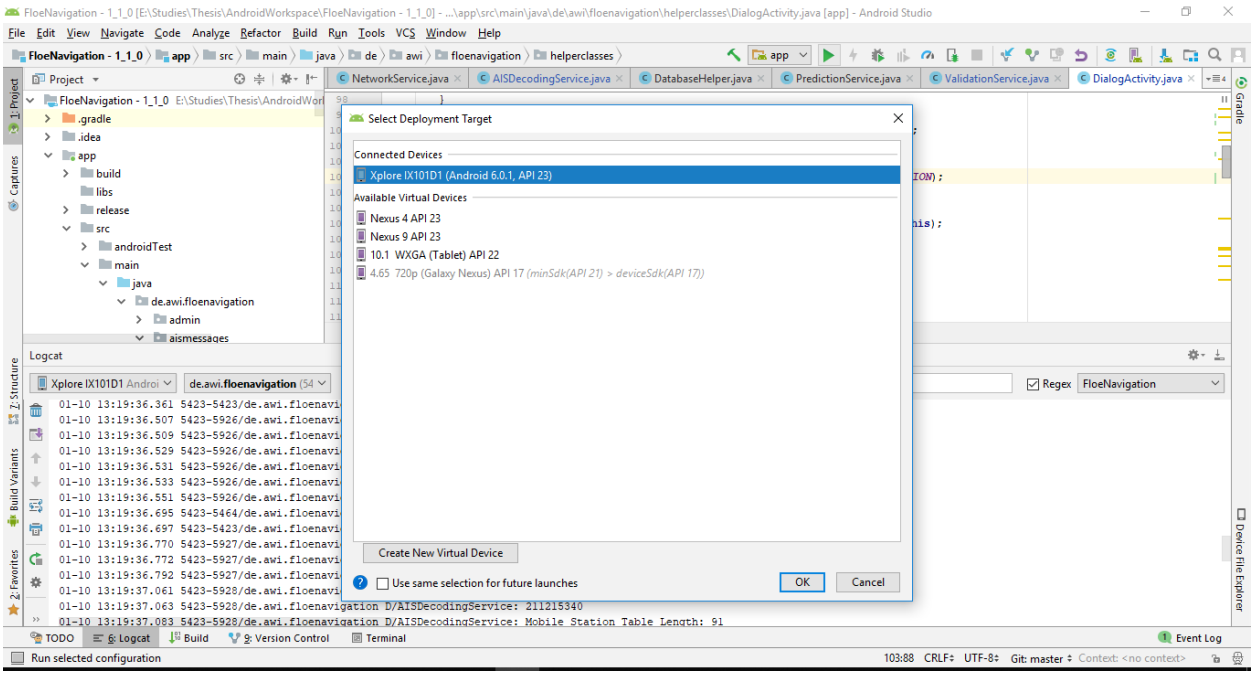


Figure 2.5 App Deployment from Android Studio using ADB over Bluetooth

You can also view the logs from the tablet using the Logcat window in Android Studio.

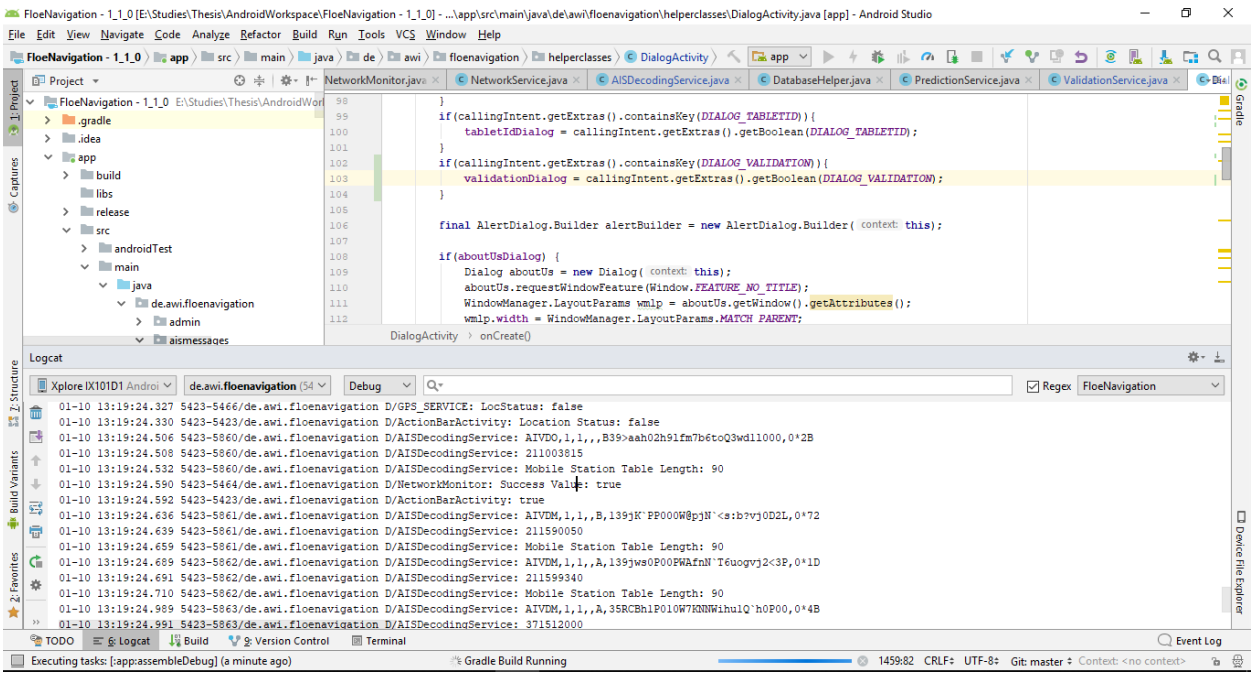


Figure 2.6 Logcat Window in Android Studio displaying the Logs from the tablet

2.2 Setting Up Synchronization Server Development Environment

The Synchronization Server consists of a database and Webservices that are used to synchronize the data. The Database is created using MySql; for details check their documentation here. The Webservices are created using PHP. The IDE used for developing the PHP scripts is Eclipse which is an open source IDE and can be downloaded from here. The Webservices are hosted on Windows IIS.

To turn on Windows IIS on your development machine follow the steps mentioned here.

To configure PHP on IIS follow the steps mentioned here.

For more details on installation and configuration of PHP and IIS check their documentation.

3 App Architecture

The Floe Navigation App is designed to ensure that each instance of the App is capable of creating and maintaining the Coordinate System on its own without any dependency on an external system. The Apps must be synchronized to each other so that different users can see the same results, however, each instance of the App can work independently.

To understand the Architecture of the App you must have a theoretical understanding of the coordinate system that is created by each instance of the App. So, it is highly recommended that you go through Chapter 2 of the Floe Navigation Administrator Guide before starting to customize the App.

The Floe Navigation App has a layered structure as can be seen in the figure below.

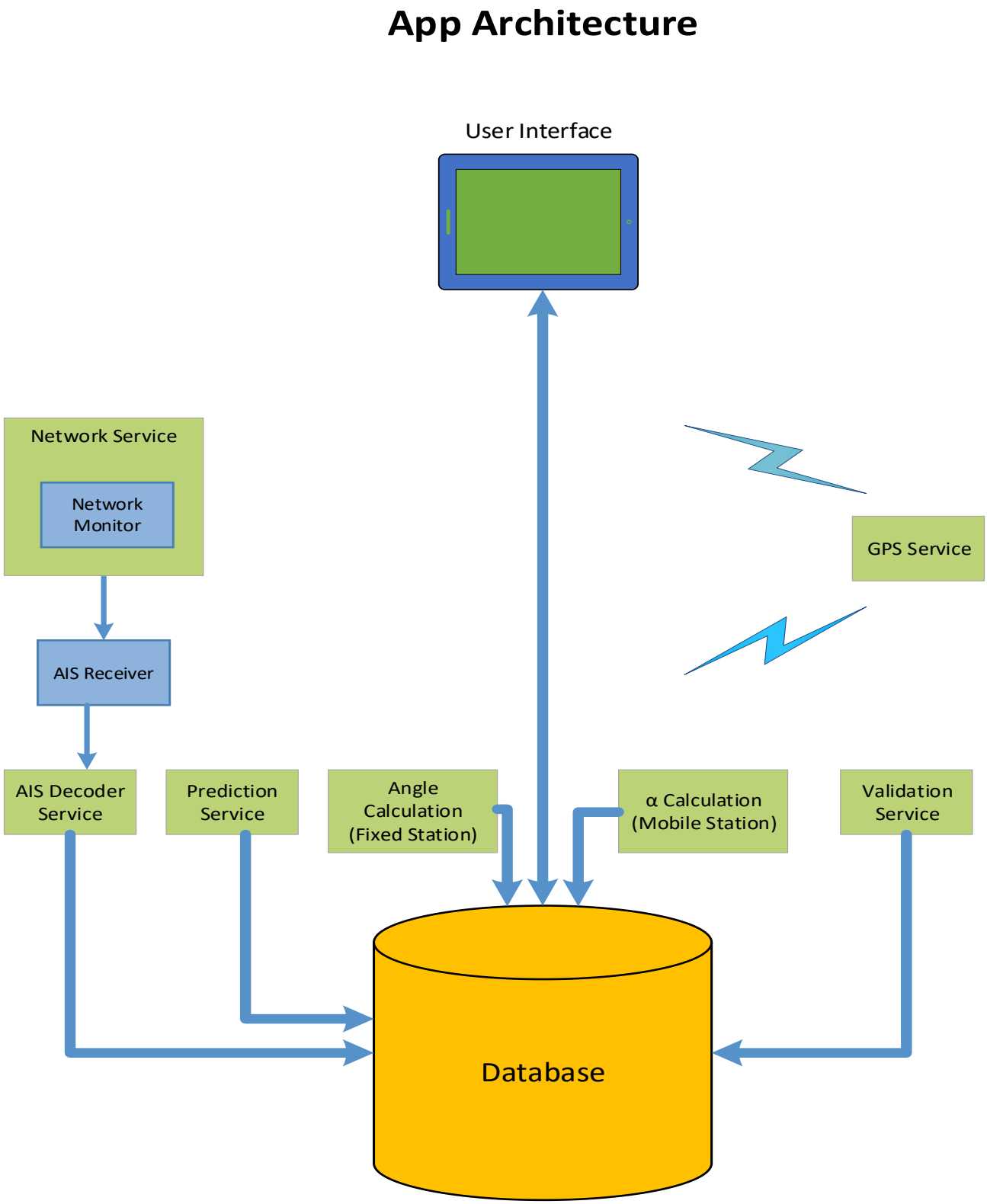


Figure 3.1 Floe Navigation App Architecture

3.1 Database

The App uses a locally created database in Android to ensure persistence of data in the tablet. The Database in Android is created using SQLite. For more details about the SQLite database check Android documentation.

The Database schema implemented is shown in Figure 3.2 & Figure 3.3

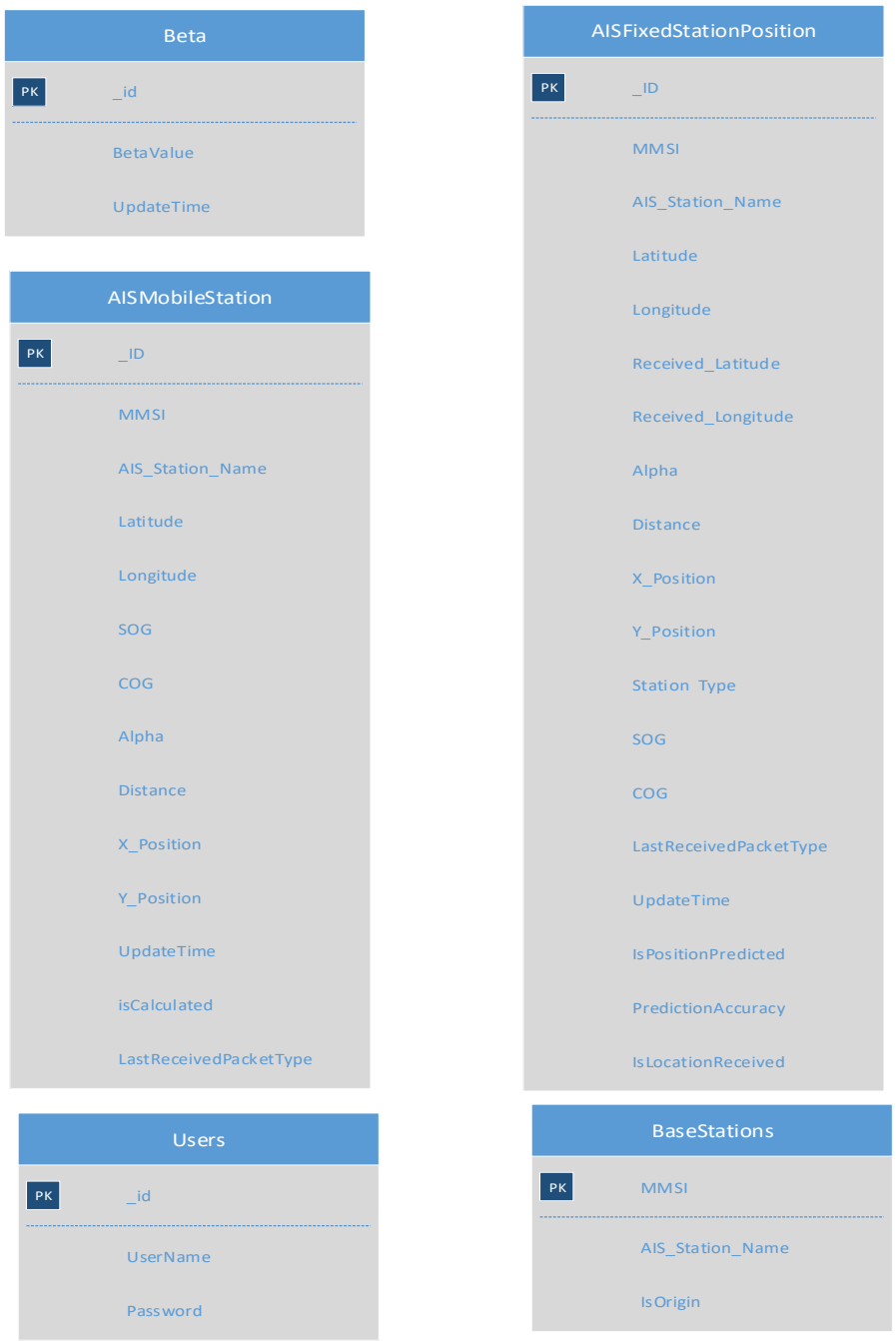


Figure 3.2 Android Database Schema

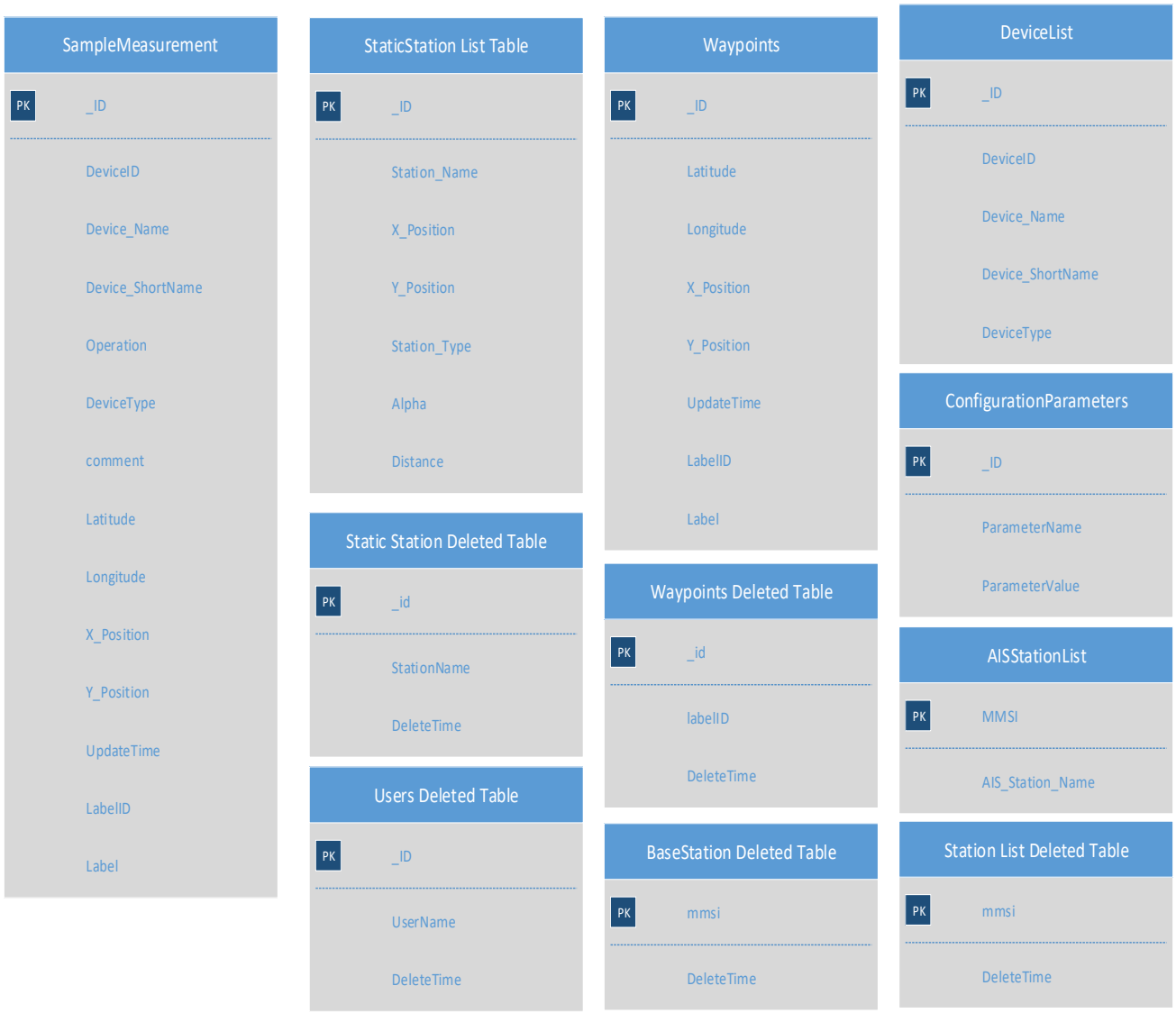


Figure 3.3 Android Database Schema Contd.

3.2 Services

As explained in Chapter 2 of the Floe Navigation Administrator Guide, the App needs to calculate several important parameters at regular intervals to ensure that the coordinate system is calculated correctly and the points installed on the coordinate system are also placed properly. The Floe Navigation App uses several Services to ensure that the calculations are done correctly and at regular intervals (For details about Android Services check Android Documentation).

These Services run along with the App and are running as long as the App is running. However, when App is used for the first time and the Coordinate System is not established yet, not all Services are running as some of these services can only work if there is a working coordinate system.

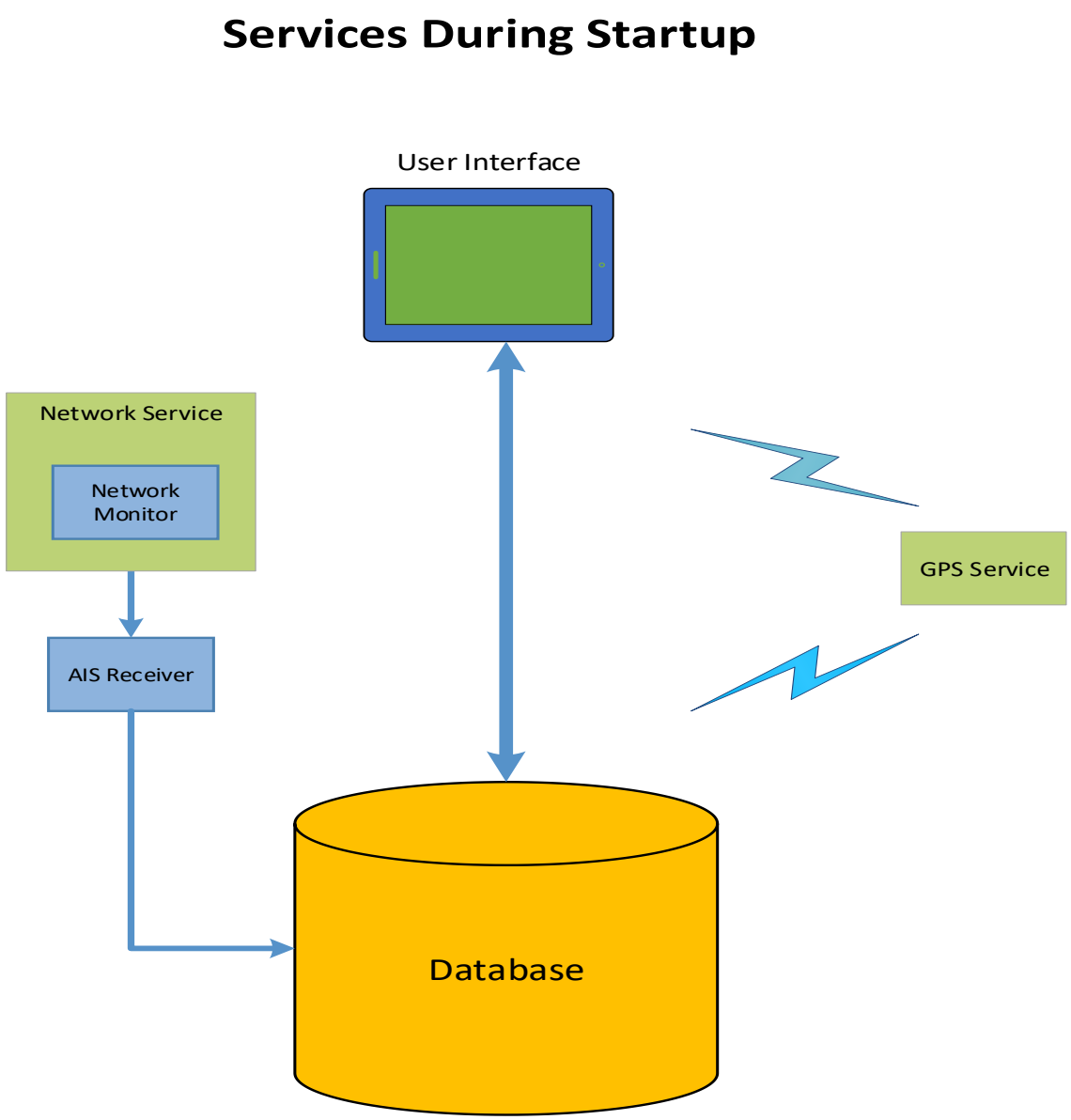


Figure 3.4 Service during First Use

3.2.1 GPS Service

GPS Service is responsible for reading the current location of the tablet and synchronizing time with the GPS Clock time. It runs in an individual thread and reads the location data from the GPS Provider within the tablet. The GPS provider does not use the Network Location Provider which uses internet and mobile network instead it uses satellite fix to determine the location of the device because of which the location fix may not work indoors.

The GPS Service does not write the location to any particular location in the database instead it broadcasts the location data and other activities or services can read the location data from the broadcast. The GPS Service runs every 30 seconds.

3.2.2 Network Service

The Network Service ensures the connectivity of the App to the Wi-Fi network of an AIS transponder. It runs a Network Monitor thread which pings the AIS transponder every 5 seconds. The Network Monitor thread, in turn, starts another thread called AIS Message Receiver which then creates a Telnet Connection with the AIS transponder and starts receiving AIS Data using a buffered reader.

If the ping drops the already running AIS Message Receiver thread is terminated, and as soon as the ping is restored, the Network Monitor thread starts a new AIS Message Receiver thread so as to reestablish the Telnet Client.

3.2.3 AIS Decoding Service

The AIS Decoding Service as its name suggests is responsible for decoding the incoming AIS packets. As explained previously the AIS Message Receiver thread creates a Telnet client which reads the incoming data using a buffered reader which can read the data line by line instead of byte by byte. As soon as a complete AIS packet is read, the AIS Message Receiver thread starts an AIS Decoding Service and passes it the packet.

The AIS Decoding Service then decodes the AIS Packet (for details on decoding check code documentation) and checks the MMSI of the incoming packet. If the MMSI is a Fixed Station it writes the data to the Fixed Station table otherwise it writes the data to the Mobile Station Table.

As the AIS Decoding Service is an Intent Service it runs in its own thread. However, as soon as its work is done and it has written the AIS data to the database it dies automatically.

Please note that the AIS Decoding Service can only decode AIS packets of Type 1, 2, 3, 5, 18 and 24. For details on AIS Packet types and the information contained therein check here.

3.2.4 Alpha Calculation Service

This service calculates the position of each Mobile Station on the coordinate system. As explained in Chapter 2 of the Floe Navigation Administrator Guide, any point on the coordinate system can be specified by the angle it makes with the x-Axis (α) and the distance from the origin. As Mobile Stations can move on the Floe their angle α and distance need to be calculated at regular interval to ensure that the Mobile Stations are shown correctly on the Grid.

The Alpha Calculation Service reads the mobile station data from the database and for each mobile station calculates the angle θ it makes with the Geographical Longitudinal Direction (See Figure 3.4). Then it calculates the difference between the angles θ and β which gives α. The Alpha Calculation Service calculates the distance from the origin using the Haversine formula (Details of which can be found here).

The Alpha Calculation Service runs every 10 seconds. So, the position of each mobile station on the grid is updated every 10 seconds as well.

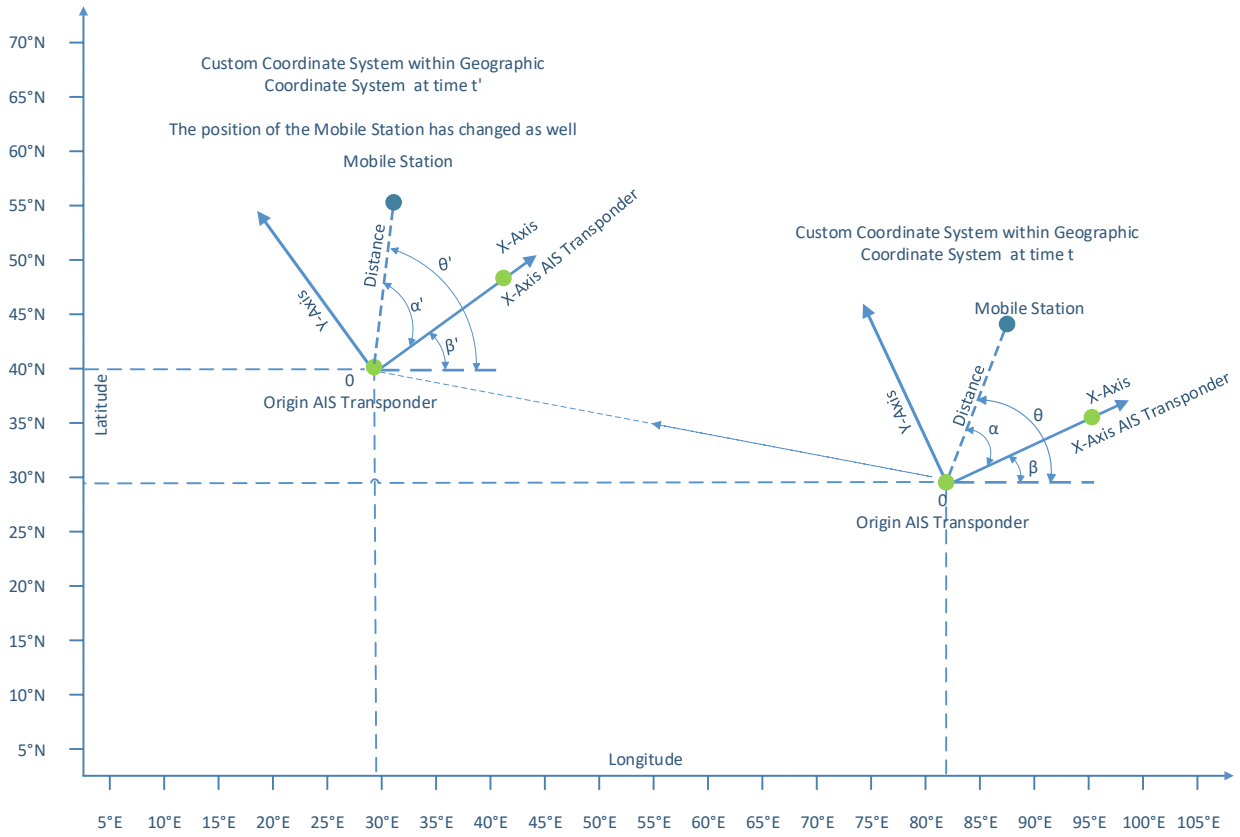


Figure 3.5 Mobile Stations within Coordinate System

3.2.5 Angle Calculation Service

This Service is responsible for maintaining the coordinate system within the Geographical Coordinates. It updates the value of the angle β which is used to create the coordinate system. For details on how the Custom Coordinate System is created and maintained go through Chapter 2 of the Floe Navigation Administrator Guide.

This Service runs every 10 seconds.

3.2.6 Prediction Service

This Service predicts the position of each Fixed Station on the Ice at a regular interval. It uses the Haversine formula to calculate the position. For each station, it reads the current Latitude, Longitude, Speed Over Ground and Course Over Ground values and predicts what the Latitude and Longitude of that station will be in 10 seconds time.

This Service also runs every 10 seconds.

3.2.7 Validation Service

This Service is used to detect if a Fixed Station has broken off from the Floe. It calculates the difference in distance between the Predicted Coordinates (from the Prediction Service) and the Received Coordinates (from the AIS Decoding Service) and if the difference is more than a specified amount it marks the prediction as a wrong prediction. If several consecutive predictions go wrong within a specified interval

of time for a given station that station is then removed from the Fixed Station table and is no longer used for calculations by other Services.

The difference in the distance which can mark a prediction wrong is defined by the ERROR_THRESHOLD configuration parameter, and the time interval is defined by the PREDICTION_ACCURACY_THRESHOLD configuration parameter. For details on the configuration parameters check Chapter 9 of the Floe Navigation Administrator Guide.

3.3 User Interface

The User Interface consists of the Android Activities and Fragments which are used to display the Grid and other forms such as Deployment, Waypoint etc. on the App. These are simple Android Activities which are used to read/write data from/to the Database.

The Grid uses a custom-built View to show the Custom Coordinate System. The rest of the activities use standard Android Views to build the Layout.

For details on how to use the User Interface check Floe Navigation User Guide, for technical details check the Code Documentation.