```
def map_plot(timedata, t0, t1, gps_lon, gps_lat, alt):

    import matplotlib.pyplot as plt
    import numpy as np
    import itertools

    from mpl_toolkits.basemap import Basemap
    from matplotlib import cm

    ''' Setting up the map boundaries'''
    westlimit = 3.25
    southlimit = 51.5
    eastlimit = 6.25
    northlimit = 52.5

    '''Plotting the map data'''
    fig, ax = plt.subplots(figsize=(10,20))

    ''' Getting the time data and zeroing it out'''
    time = np.round(np.asarray(timedata.get('data')),3)

    ''' Indexing where to start plots from '''
    if t1 == "end":
        endind = -1
    else:
        endind = list(time).index(t1)
    if t0 == 'start':
        startind = 0
    else:
        startind = list(time).index(t0)

    ''' Setting up the basemap plot '''
    m = Basemap(resolution='f', # c, l, i, h, f or None
                projection='cyl',
                lat_0=(westlimit+eastlimit)/2, lon_0=(southlimit+northlimit)/2,
                llcrnrlon=westlimit, llcrnrlat= southlimit, urcrnrlon=eastlimit, urcrnrlat=northlimit, epsg=3035)

    ''' Including parameters for the map plot'''
    '''         Uncomment below for less detailed plot      '''
#    m.drawmapboundary(fill_color='#46bcec')
#    m.fillcontinents(color='#eeeeee',lake_color='#46bcec')
#    m.drawrivers(linewidth=1, color='#46bcec', antialiased=1, ax=None, zorder=None)
#    m.drawcountries(color = "black")
#    m.drawcoastlines()
    '''         Uncomment below for detailed plot           '''
    m.arcgisimage(service = "World_Topo_Map", xpixels=12000) #uncomment this for super high res image

    '''drawing the meridian and parralel lines, labels = [left,right,top,bottom] '''
    m.drawmeridians(np.arange(0, 360, 0.5), labels=[False,True,True,False])
    m.drawparallels(np.arange(-90, 90, 0.25), labels=[True,False,False,True])

    '''Plotting location of Rotterdam Airport'''
    rotlat = 51.9555
    rotlon = 4.4399

    xpt, ypt = m(rotlon, rotlat)
    lonpt, latpt = m(xpt, ypt, inverse=True)
    m.scatter(xpt, ypt, marker='D',color='r', s=10, zorder=11)

    '''Plotting Flight GPS Data '''
    lon = np.asarray(list(itertools.chain.from_iterable(gps_lon.get('data'))))[startind:endind]
    lat = np.asarray(list(itertools.chain.from_iterable(gps_lat.get('data'))))[startind:endind]
    alt = np.asarray(list(itertools.chain.from_iterable(alt.get('data'))))[startind:endind]

    xpts, ypts = m(lon, lat)
    lonpts, latpts = m(xpts, ypts, inverse=True)
    norm = 1-(((alt-np.min(alt))/(np.max(alt)-np.min(alt))))
    m.scatter(xpts, ypts, marker='o', c=cm.viridis(norm),  edgecolor='none', s=15, zorder=10)

    ''' Returning the map plot '''
    plt.savefig("sexi-map-boi", dpi=1000)
    return plt.show()
```