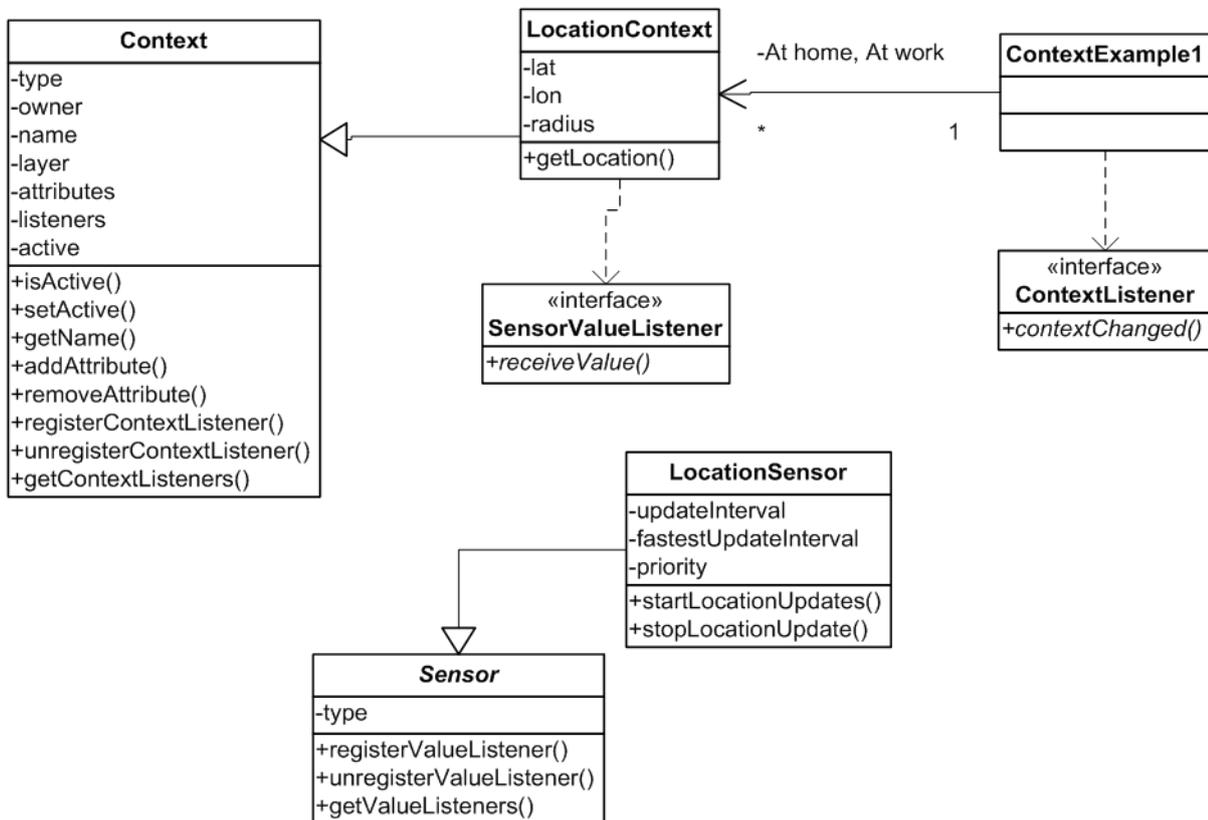


ContextExample1.java

This example creates two location-based contexts named "At home" and "At work". The location contexts take the coordinates (lat, lon) and radius (in meters) as input, which values define the area (circle) where the context is active. Further, the example shows how to receive updates to the contexts active value.



LocationContext

This example uses the LocationContext class (eu.h2020.helios_social.core.context.ext.LocationContext), which is a subclass of the base class Context (eu.h2020.helios_social.core.context.Context), see the diagram above.

E.g., creating a location-based context named "At home": (code samples from ContextExample1.java)

```
locationContext1 = new LocationContext(user1, "At home", home_lat, home_lon, 1000.0);
```

ContextListener

Then, register a listener (that implements the ContextListener interface) to obtain changes in the context active value:

```
locationContext1.registerContextListener(this);
```

The ContextListener interface (eu.h2020.helios_social.core.context.ContextListener) includes the method contextChanged, which informs that the active value (true/false) of the context was changed.

```

@Override
public void contextChanged(boolean active) {
    // do something with the received active value of the context. For example, update the UI:
    updateUI();
}

```

Also, it is always possible to check whether the context is currently active by using the `isActive` method defined in the Context base class:

```
Boolean active = locationContext1.isActive();
```

LocationSensor

LocationSensor class (`eu.h2020.helios_social.core.sensor.ext.LocationSensor`) obtains GPS position from phone with certain update intervals (5s), which can be set by the application.

```
mLocationSensor = new LocationSensor(this);
```

The LocationSensor class is a subclass of the Helios core class Sensor (`eu.h2020.helios_social.core.sensor.Sensor`) and implements the SensorValueListener interface (`eu.h2020.helios_social.core.sensor.SensorValueListener`).

Then, set the context to receive updates in GPS position from the LocationSensor using the `registerValueListener` method of Sensor:

```
mLocationSensor.registerValueListener(locationContext1);
```

The LocationContext class implements the SensorValueListener interface (`eu.h2020.helios_social.core.sensor.SensorValueListener`). Further, the LocationSensor include the methods to start and stop the location updates:

```
mLocationSensor.startLocationUpdates(); // start updates
```

```
mLocationSensor.stopLocationUpdates(); // stop updates
```

When the context receives updated position, it also checks whether the position is within the area (circle) where the context is defined to be active, and update the active value correspondently. Also, each registered context listener receive the update in active value.

