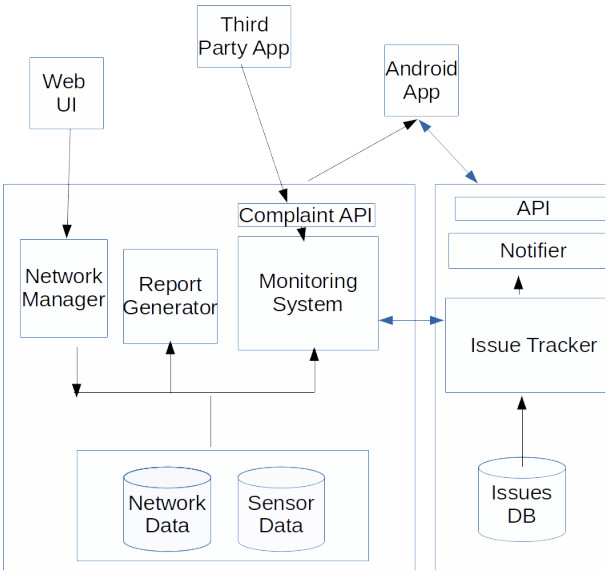
Smart Water Networks

Abhijith Madhav Kumudini Kakwani

November 1, 2014

High Level Design



Monitoring System

Scans the sensor data continuously for anomalies and reports the same to the issue tracker. Anomalies include breaches in threshold levels, leaks and external complaints. It also reports daily water requirements according to it's prediction.

Reports Generator

Provides an API to get the following types of reports, each of which can be drilled down.

- Water consumption by aggregation.
- Water consumption by time.
- Water consumption by population.
- Health of network assets.

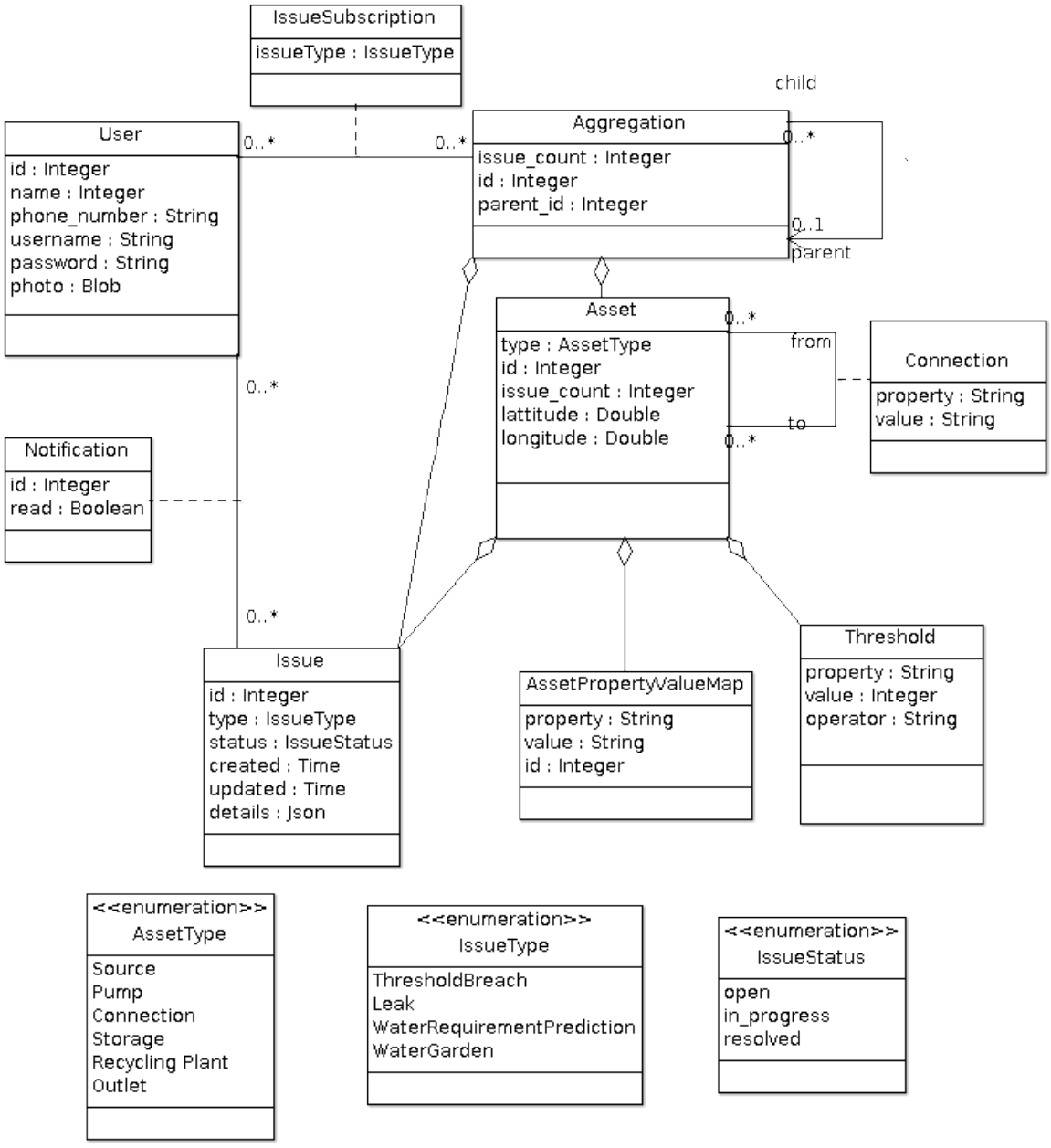
Issue Tracker

Keeps track of the status of resolution of issues. Also, notifies on creation of an issue to subscribers.

Network Manager

Provides an API for creating the network and defining hierarchical aggregations.

Database Design



Low Level Design

Following is the rough sketch of the main classes.

Class Name	Members	Methods
SensorMonitor	issueTracker	monitor(), predictWaterRequirement(), DetectLeaks()
ComplaintListener	issueTracker	getAggregations(), createComplaint()
IssueTracker	notifier	add(issue), resolve(issue)
Notifier		notify()
ReportGenerator		createAggregationBasedReport() createTimeBasedReport() createHealthBasedReport()
NetworkManager		createNetwork(), createConnections() modifyProperty(), createAggregation()