

CT30A9300 Code Camp on communications Engineering
Qt Codecamp

Whitepaper

QtFish

31.10.2010

Team members

Antti Jäppinen	antti.jappinen@lut.fi
Jukka Stranden	jukka.stranden@lut.fi
Marko Suhonen	marko.suhonen@lut.fi

Problem statement

Fishermen and fishing enthusiasts love to travel around in search for good fishing spots. Traditionally they've remembered them by either marking the spot in a map, written down instructions in notes or have just committed the place to their memory. All of these methods, with the possible exception of the map, can be very inaccurate though. Nature is in constant change so a special tree that marked that good special fishing spot may have been struck down by lightning. And the fisherman is lost.

In addition, carrying notes about the captured fish and their location requires space and care as the paper is often exposed to the elements. QtFish aims to solve that problem.

Vision

QtFish is an application that uses GPS positioning to get the location of the captured fish. It also lets the user to write notes about the fish such as weight, length, location description and other notes. It automatically adds date and time to the data but even that can be edited if the notes are written afterward. QtFish is also intended to have a camera functionality to allow the fisherman to take a photo of his catch and link it to the note. However, camera support is not yet in official Qt Mobility release so it was omitted.

Because cellphones are prone to accidents like falling and breaking, falling into water or just general upgrade, the data must be saved in a format that can be retrieved and saved elsewhere. QtFish saves the data in xml format which is an open format and has good support. In addition QtFish lets the user to upload the data to a webserver using ftp. Ftp is available with almost every service offering website space. Uploading the data to a server works as an effective backup, enabling the user to download it for viewing at computer and allows easy sharing with friends.

Technical description

QtFish is implemented with Qt 4.6.3 and Qt Mobility 1.0.2. It is cross platform and should compile to both Maemo 5 and Symbian S60 operating systems. However, because it uses self signed UID and location services, it will only work in Symbian 3.2, 5th Edition and Symbian^3. It won't work in Symbian S60 3.1 and older devices. The application user interface is designed for Symbian touch so recommended device is Symbian 5th Edition phone or Symbian^3 phone with touchscreen. Maemo

build has not been tested but it should work. Because Maemo does not support soft keys, `_MAEMOBUILD` must be defined in project file to enable the use of buttons instead of soft keys.

Architecture

QtFish is divided roughly in user interface and and model layers. The model layer consists of LocationKeeper, FishSpot and DataStorage classes. Other classes are part of the user interface although some of them handle controller options like FtpDialog which transfer files from FTP Server as well as shows the results. Most of the application is run by MainWindow which shows the mainview of the application. All the options are accessed through MainWindow. The image below shows the class diagram of the application.

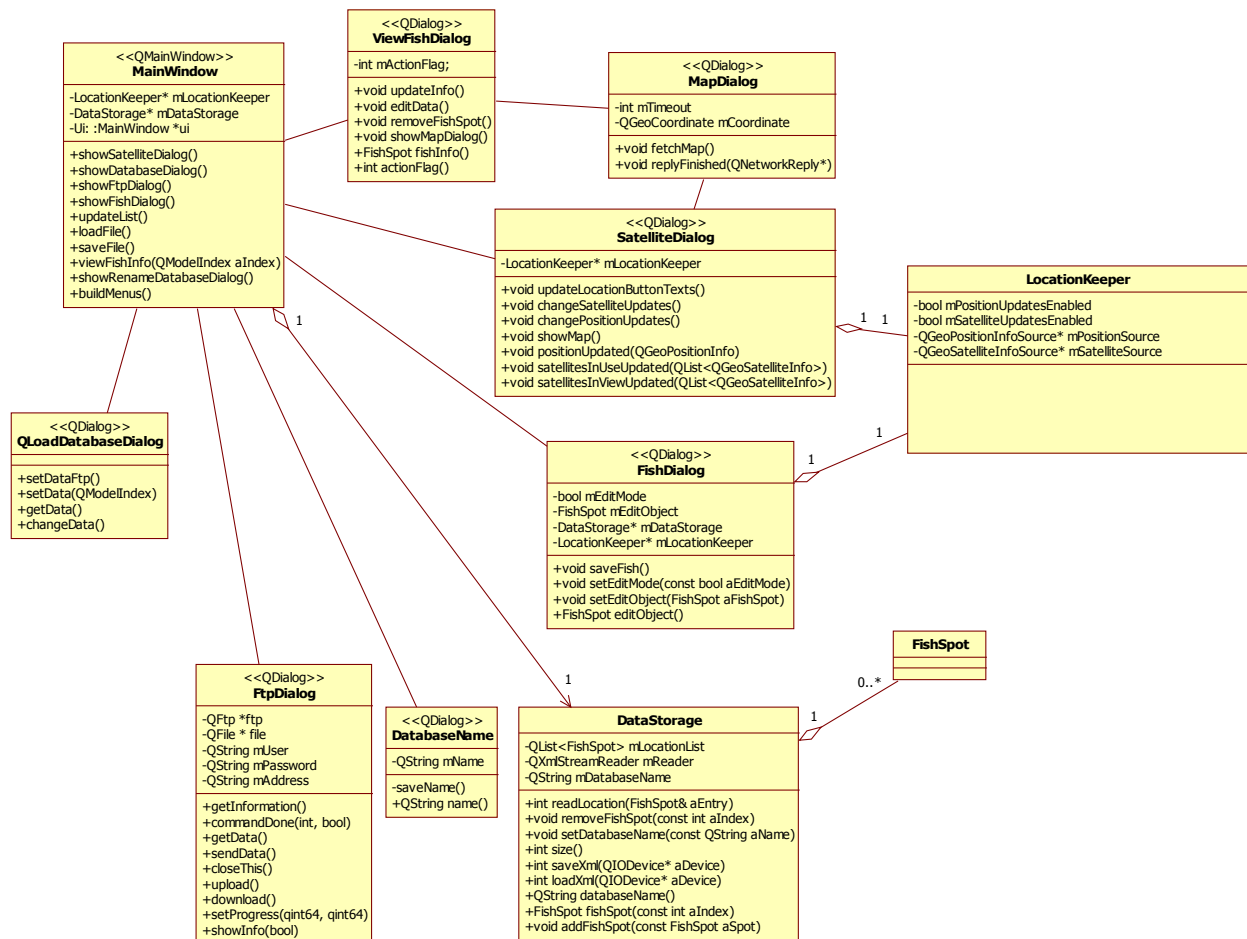


Illustration 1: Class diagram of the application