Infotechnoprom

Introductory Assignment

Position: Junior Software Developer

ID: 001ia

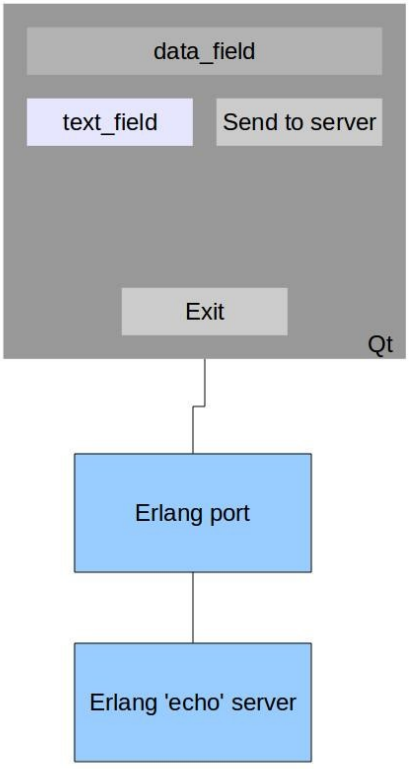
Date: 31.05.2012

1. Version history

Version	Implemented by	Reason/Rationale	Date
1.0	Andrey Pogodin	Document creation	31.05.2012

2. Description

A customer asked us to create a GUI (graphical user interface) to an Erlang/OTP application that is used in his company. The team leader and the architect decided to implement the GUI in Qt (Python/C++) and to connect the GUI to the existing Erlang/OTP application via Erlang Ports functionality. Your task is to write the actual code: a simple Erlang server (which we imagine to be an existing Erlang/OTP application) and a simple GUI. (This software system is completely made-up, we imagined it specifically for this assignment).



3. Requirements

3.1. Functional requirements

- the Erlang server should be a simple 'echo' server: once started it should open a port to a Qt GUI window and wait data from it;
- a user should be able to enter any data (via keyboard) in the **text_field** of the GUI window and click '**Send to server**' button. In that case the GUI window should send the data entered by the user to the Erlang server;
- when the data arrives at the Erlang server via port, the server should just simply send the same data back to the GUI window;
- the GUI window should print the data it has got from the server in the **data_field**;
- if the user clicks the **Exit** button, the server should stop (and close the GUI window as well).

3.2. Non-functional requirements

- the server should be written in Erlang;
- the GUI window should be made with Qt/Python;
- the whole system should be able to run on a Linux-compatible system (cross platform is not a requirement, but you should provide a README with the code explaining how to compile and run it on Debian/Ubuntu).

4. Deadline

The deadline is 14 calendar days starting after the day you got this assignment.

5. Bonus points

You can get bonus points for your assignment if you provide some or all of the following:

- The Erlang server is written as an OTP gen_server;
- The Erlang server is supervised with an OTP supervisor;
- The whole software system is packed as an OTP application and can be started and stopped with application:start()/application:stop() (including the graphical user interface);
- Unit, integration and system tests: code coverage (cover for Erlang & whatever for Qt) is more than 50%;
- rebar is used to compile, run and test the application;
- comprehensive documentation for the code (Edoc, Doxygen or any other tool(s) of your choice);
- In order to complete the assignment you will obviously do some research. One of the great research tools is the Stackoverflow community. When handing in the assignment you can attach a link to your profile at Stackoverflow which shows the questions you asked during the work on this assignment (if the questions are good and balanced, you might get bonus points);
- instead of sending by email you put your code in GitHub.