# Bachelor Thesis

## « A Leightweight Forms Engine for Querying the Mobile Web »

## Background

Mobile applications on devices such as smartphones access servers to query for information. The servers respond with search forms, which have to be filled in by the user. Once the user has filled in the information, the data of the form is transferred back to the server. In order to provide the functionality of entering information correctly, the form has to convey some sort of information what data has to be entered and how data may influence other data.

XForms are one way to do this. However, XForms is an XML based language which, compared to other languages, uses more bandwidth. Validating XForms models against their XML Schema, validating dependencies of form fields and reading different values from the form requires the use of libraries which have a considerable memory footprint.

Current smartphones communicate with servers using the mobile Internet. Every such communication can be assigned a cost which accounts for the amount of data which is transferred. The goal of this work is to reduce this cost by using the leverages *caching*, *data encoding*, *prefetching* and *data compression*. Additionally, there exist several standard libraries on the mobile devices which should be used instead of adding new ones to decrease the memory usage and reduce processing time.

## Work Description

In this work, a lightweight alternative to XForms shall be designed and implemented. Based on lessons learned from XForms and inspired by modern web applications the solution shall be realized based on JSON, scripting, caches and compression.

A Servlet container such as Tomcat will be set up as a forms provider. A client on an Android phone (emulator) will access the server and request a form. Once the form is transmitted to the client, it will reconstruct an user interface and allow the user to enter data and finally submit the form data to the server. Every form is divided into two parts: the specification of the form and the model for the form. The client initially accesses the model from the server and checks its cache whether the specification is available. If not, it will load the specification from the server as well. A form specification includes an area where the form designer is allowed to enter JavaScript code in order to dynamically handle user input.

## Requirements

**Good knowledge:**  Java, HTTP
**Interest in:**  Android, JavaScript, JSON, Servlets

**Contact: Helge Klimek**

Helge.Klimek@tu-harburg.de          Phone: +49 40 / 428 78 – 3529          Room: E 4.074

**Institute of Telematics**

**Hamburg University of Technology**

**TUHH**