

# **Titel** more titel

17. Dezember 2013

put abstract here

---

## **Inhaltsverzeichnis**

# 1 Robust Programming

Folgende

## 1.1 Model

### 1.1.1 XSS, SQL-Injection und Validierung

### 1.1.2 Schutz der Model-Klasse vor schädlichem Missbrauch

Die Schwierigkeit einer korrekten Login-Implementation liegt darin, den Ablauf so zu gestalten, dass nur dieser Ablauf zu einer validen Änderung führt. Heisst nur beim vorgesehenen Ablauf können nur korrekte Daten geschrieben werden, alles andere darf die Datenbank nicht verändern. So sind beispielsweise setters auf das Passwort problematisch. Zusammen mit der `.save()` Methode könnte ein fehlerhafter oder bössartiger Code das Passwort eines beliebigen Users überschreiben. Usernamen und ID sind ebenfalls Felder, auf die nur lesend zugegriffen werden soll.

Umsetzung:

### 1.1.3 Postleitzahl Validierung

Die Postleitzahl wird wie verlangt, durch einen externen Service überprüft. Wir überprüfen nur, ob die Postleitzahl existiert und nicht ob die angegebene Stadt mit der Postleitzahl übereinstimmt.

## 1.2 Controller

Der Controller wurde so simpel wie möglich gehalten,

### 1.2.1 SSL

# 2 Weitere Mögliche Massnahmen

## 2.1 DOS-Attacke