HTWK Leipzig
Fachbereich IMN
Sommersemester 2014

Traincard

Beleg im Smartcard Programmierung
bei Prof. Dr. rer. nat. Uwe Petermann

Kurt Junghanns, B.Sc.
Marcel Kirbst, B.Sc.
Michael Reher, B.Sc.

10. Juli 2014

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

Diese Arbeit befasst sich mit der Vorstellung der Belegarbeit im Fach Smartcard Programmierung mit dem Thema “Traincard”. Ziel des Belegs ist die prototypische Implementierung eines Trainingssystems auf Basis JCOP-fähiger Smartcards.

Im ersten Kapitel Anwendungsfall wird der Use-Case für die prototypische Implementierung beschrieben, sowohl unter Anwendungsfall aus Sicht des Trainierenden aus Sicht des Trainierenden, wie auch unter Anwendungsfall aus Sicht des Trainers aus Sicht des Trainers.

2 Anwendungsfall

In vielen Fitnessstudios werden heute noch bei der Neuanmeldung Trainingspläne auf Papier ausgegeben. Der Trainierende führt den Trainingsplan über die Trainingsperiode ständig bei sich und verzeichnet den Trainingsfortschritt in diesem Dokument. Im Verlauf der Trainingsperiode kann der Trainer beurteilen wie effektiv das Training beim Trainierenden ist.

Die Verwendung altmodischer Trainingspläne auf Papier hat jedoch einige Nachteile. Beispielsweise wird das Dokument vom Trainierenden manchmal vergessen, die betreffenden Trainingsfortschritte müssen also nachgetragen werden. Weiterhin sind Traingspläne in Papierform nicht besondern resistent gegen Abnutzung und verschleißen mit fortdauernder Verwendung. Um den genannten Nachteilen zu begegnen soll im Rahmen dieser Belegarbeit der Einsatz von Trainingsplänen auf Basis so genannter Smartcards abgebildet werden. Smartcards können sehr leicht in einer Brieftasche mitgeführt werden und sind weniger anfällig für Abnutzung. Weiterhin bieten sich viele weitere Vorteile wie ... (Datenlogging für das Studio, Kopieren bzw. verteilter Zugriff auf die Daten ....)

2.1 Anwendungsfall aus Sicht des Trainierenden

Aus Sicht des Trainierenden bietet die Verwendung der Smartcard einige der folgenden Vorteile:

- leichte Transportierbarkeit
- robuster als Trainingspläne aus Papier

2.2 Anwendungsfall aus Sicht des Trainers

Trainer profitieren beim Einsatz von Smartcards unter anderem von folgenden Vorteilen:

- leichte und lesbare Auswertung der vorgegebenen Trainingspläne
- zusätzliche Metainformationen wie beispielsweise zu welchem Zeitpunkt welcher Datensatz geschrieben wurde
- statistische Auswertungen lassen sich sehr leicht erstellen

3 Grundlagen

In diesem Kapitel wird auf die Grundlagen der in diesem Beleg verwendeten Technologien eingegangen.

3.1 Grundlagen Smartcard

Die in diesem Beleg verwendete Smartcard basiert auf der Java Card Technologie. Die Java Card Technologie bietet eine Teilmenge der Java Programmiersprache, sowie eine hinsichtlich der Anforderungen an Smartcards optimierte Laufzeitumgebung.
Die Verwendung von Java-basierten Smartcards bietet einige Vorteile, von denen nachfolgend eine Auswahl beispielhaft genannt sei: [1]

plattformunabhängig: Java Card Applets, die der Java Card API entsprechen, lassen sich plattformunabhängig und herstellerübergreifend nutzen.

Multiapplikationsfähig: es können mehrere Applikationen gleichzeitig auf einer Smartcard ausgeführt werden

hohe Flexibilität: die Verwendung der objektorientierten Programmiersprache Java erlaubt die Erstellung komplexer Anwendungen für die Smartcard.

Post-Aktualisierbarkeit: die Möglichkeit, nachträglich Code auf der Smartcard zu modifizieren und auszutauschen erhöht die Flexibilität weiter

Standardkonformität: die Java Smartcards entsprechen dem ISO7816 Standard [2]

Eine Java Smartcard besteht im Wesentlichen aus den Bestandteilen Kommunikationsschnittstelle, Speicher und einem Prozessor zur Durchführung von Berechnungen. Bei Verwendung der Smartcard wird diese in ein Lesegerät eingelegt. Das Lesegerät wird in einschlägiger Literatur auch als Card Acceptance Device, abgekürzt CAD, bezeichnet. Der Speicher auf Java Smartcards besteht aus zwei Typen, RAM und EEPROM. Der RAM-Speicher ist flüchtig und kann beliebig oft beschrieben werden. Der EEPROM-Speicher ist nichtflüchtig und kann nur endlich oft beschrieben werden, je nach Hersteller und Modell bis zu 100.000 mal pro Speicherzelle.

3.2 Java Card Open Platform

Java Card Open Platform, abgekürzt JCOP, ist der Name für ein Smartcard Betriebssystem, welches initial von IBM entwickelt wurde, inzwischen aber von NXP

Semiconductors betreut wird. Es stehen sowohl reale JCOP-Karten zur Verfügung, wie auch simulierte JCOP-Karten.
Bei der Verwendung realer JCOP-Karten

3.3 APDU

Die Kommunikation zwischen dem Kartenlesegerät und der Smartcard erfolgt reaktiv und paketweise. Reaktiv bedeutet, dass die Smartcard keine Kommunikation initiiert sondern nur auf Anfragen vom Lesegerät reagiert. Der bei der Kommunikation zwischen Smartcard und Lesegerät verwendete Kommunikationsmechanismus wird als Application Protocol Data Units, abgekürzt APDU, bezeichnet. Spezifiziert ist dies ebenfalls in ISO7816 Teil 4.[2]

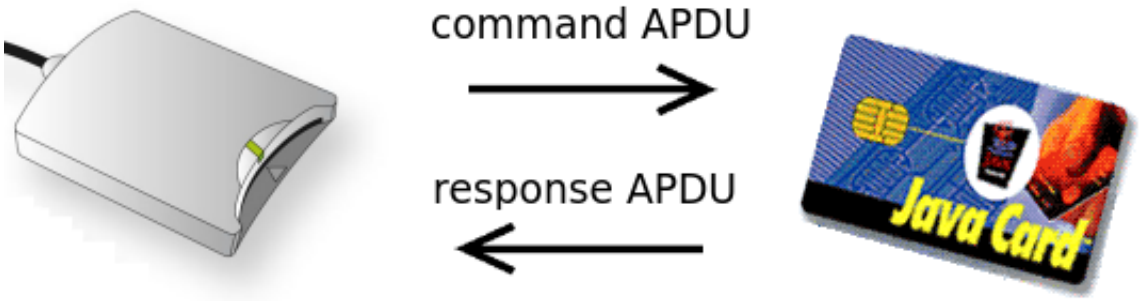

Abbildung 1: APDU-Kommunikation schematisch dargestellt, Quelle: Autoren

Die Kommunikation wird über ein so genanntes command APDU Paket initiiert, welches aus den folgenden Feldern besteht:

CLA: class,gibt die Klasse an, spezifiziert ob es sich um ein ISO7816-4 konformes Kommando handelt

INS: gibt die Instruktion an

P1: zusätzlicher Parameter

P2: zusätzlicher Parameter

Weiterhin können je nach Kommandotyp noch die folgenden, optionalen Felder an das command APDU Paket angehängt werden:

Lc: Length, gibt die Länge der Kommandodaten

Data: gibt die Kommandodaten an

Le: gibt die Länge der erwarteten Antwort an

Als Antwort erhält das Lesegerät von der Smartcard ein so genanntes response APDU Paket. Dieses Antwortpaket kann Daten enthalten, dies ist jedoch nicht obligatorisch. Der Aufbau eines response APDU ist wie folgt:

Data: optionaler Datenbestandteil

Sw1: Statusword, erstes Byte

Sw2: Statusword, zweites byte

4 Umsetzung

Dieses Kapitel legt die Umsetzung dar.

4.1 Systemumgebung

Die Anwendung besteht aus einem On-Card und einem Off-Card Teil. Entsprechend der JCOP Umgebung, ist der On-Card Teil durch ein Applet realisiert, welches auf der Smartcard installiert und gestartet wird. Als Smartcard kann eine physische oder eine emulierte zum Einsatz kommen. Im Rahmen dieses Projektes wird die Smartcard per JCOP Eclipse Umgebung emuliert und verwendet. Dabei ist in der gestarteten JSOP Shell der Befehl */close* auszuführen, wodurch die Smartcard für externe Zugriffe auf der lokalen IP des Emulator-Rechners auf dem Port 8090 erreichbar ist. Der Off-Card Teil ist ebenfalls in Java geschrieben und kommuniziert mit Hilfe des OpenCard Frameworks mit der Smartcard. Die erfolgreiche Kommunikation zwischen On-Card und Off-Card auf verschiedenen Rechnern benötigt entsprechende Regeln der Firewalls der Rechner, lokale Ausführung beider Teile auf einem Rechner benötigt keine
Da für die Umsetzung des Belegs keine Experimentalhardware zur Verfügung steht, erfolgt die Implementierung mit Hilfe eines Simulators.

4.2 On-Card Teil

Der On-Card Teil der Implementierung enthält sämtliche Trainingspläne und weitere Daten.

4.3 Off-Card Teil

5 Zusammenfassung und Ausblick

Zusammenfassung und Ausblick ohne persönliche Meinung.

6 Quellenverzeichnis

Quellenverzeichnis

[1] Sun Microsystems, Inc.: *Java Card Applet Developer's Guide*
http://www.oracle.com/technetwork/java/javacard/downloads/index.html
abrufbar am 01.Juli 2014

[2] International Organization for Standardization: *ISO 7816 - Identification cards – Integrated circuit cards* kostenpflichtig abrufbar unter http://www.iso.org/

[3] Autornachname, Autorvorname: *Buchtitel* Verlag, Erscheinungsjahr, ISBN: 978-0-07024-807-6

[4] Uwemann, Peter: *Train harder, 2. Auflage*. gpunkt-Verlag Heidelberg, 2012, ISBN: 133-7-13371-337-1