Pflichtenheft 1920Parser

# 1. Zielbestimmung

Die Phoenix Group IT verwendet für gemeinse COBOL-Variablendefinitionen und als Schnittstelle zum Senden von Logdaten an den Lagerrechner eigene Dateischemas. Anhand dieser Schemas werden Datenströme aus den Logdateien interpretiert. Diese Aufgaben fallen bei Programmänderungen sowie Kundenreklamationen an. Bisher analysieren die Entwickler die Logdaten von Hand. Datenströme sollen in Zukunft automatisch anhand eines angegebenen Schemas zergliedert werden.

## 1.1 Musskriterien

* Datenstrom und Schema müssen frei angegeben werden können.
* Das Programm muss den Datenstrom anhand des Schemas zergliedert anzeigen.

## 1.2 Wunschkriterien

* Schemas, die ein Benutzer einmal angegeben hat sollen gespeichert werden.
* Anhand des Transaktionscodes im Datenstrom soll das richtige Schema - falls vorhanden - automatisch ausgewählt werden.
* Alle Datenströme sind 1920 Byte lang und sequentiell in Logdateien abgespeichert.

Der Benutzer soll ein Kriterium angeben können, anhand dessen die passenden Datenströme aus der Logdatei ausgewählt werden.

* Länge-Felder können Angaben zur Anzahl Nachkommastellen haben. Das Programm soll an der richtigen Stelle ein Komma einfügen.
* Manche Redefine-Felder sind von größerem Interesse als andere. Es soll Funktionalität bereitgestellt werden, um Redefine-Felder als wichtig zu kennzeichnen und nur diese anzuzeigen.

## 1.3 Abgrenzungskriterien

* Das Programm soll Schema und Datenstrom nicht auf Korrektheit oder Plausibilität prüfen

(z. B. doppelte Variablennamen, Redefines ohne Vorgänger, ungültige Stufennummern).

# 2. Produkteinsatz

2.1 Anwendungsbereiche

* Als Tool für die Programmierer der Phoenix Group IT zur Unterstützung bei der Logdatei-Analyse

2.2 Zielgruppe

* Die Programmierer der Abteilung Warehouse der Phoenix Group IT GmbH

# 3. Produktumgebung

3.1 Software

* Das Programm muss unter Windows laufen

3.2 Hardware

* Entwickler-PCs der Phoenix

# 4. Produktfunktionen

4.1 Angabe von Datenstrom und Schema

4.2 Parsen des Schemas in eine Baumstruktur

4.3 Werte aus dem Datenstrom den Baumknoten zuweisen

4.4 Ausgabe des Baumes als String

# 5. Produktdaten

# 6. Produktleistungen

# 7. Benutzeroberfläche

* 2 Textboxen zur Angabe von Datenstrom und Schema, Ausgabe auch in einer Textbox

# 8. Globale Testszenarien und Testfälle