

Anhang

A.11 main.go-Quellcode

```

1  /*
2  #####
3  Projekt: CAMT053-Webservice
4  Autor: Marvin Viedt
5  Copyright: a.b.s. Rechenzentrum GmbH
6  Version: 1.0 (still WIP)
7  Release: Sommer 2020
8  #####
9  */
10
11 package main
12
13 import (
14     "fmt"
15     "log"
16     "errors"
17     "net/http"
18     "html/template"
19     "time"
20     "strings"
21     "regexp"
22     "os"
23     "github.com/antchfx/xquery/xml"
24     "path/filepath"
25 )
26
27 var (
28     Map = make(map[string]string)
29 )
30
31 func main() {
32     log.Println("Server wird gestartet")
33
34     http.HandleFunc("/", handler)
35     http.HandleFunc("/Camt053/", kauszughandler)
36
37     http.Handle("/static/", http.StripPrefix("/static/", http.FileServer(http.Dir("static"))))
38
39     err := (http.ListenAndServe(":3000", nil))
40
41     log.Fatal(err)
42 }
43
44 func handler(w http.ResponseWriter, r *http.Request) {
45     t, err := template.New("Webservice").Parse(webseite)
46     fmt.Println("CAMT053-Webservice wird ausgeführt")
47
48     expiration := time.Now().Add(10 * time.Second)
49     cookie := http.Cookie{Name: "Session", Value: "Session", Expires: expiration}
50     http.SetCookie(w, &cookie)
51
52     if err != nil {
53         http.Error(w, err.Error(), http.StatusInternalServerError)
54     }
55
56     err = t.Execute(w, nil)
57
58     if err != nil {
59         http.Error(w, err.Error(), http.StatusInternalServerError)
60     }
61 }
62
63 func kauszughandler (w http.ResponseWriter, r *http.Request) {
64
65     if r.Method != http.MethodPost {
66         http.Error(w, "Das ist nicht gültig", http.StatusMethodNotAllowed)
67         return
68     }
69
70     formvalue, err := getformvalue(r)
71
72     if err != nil {
73         http.Error(w, fmt.Sprintf("Formeingabe(n) konnte(n) nicht geladen oder bearbeitet werden: %s", err), http.StatusBadRequest)
74         return
75     }
76
77     if r.URL.Path == "/Camt053/Kontoauszugs-View" {

```

Anhang

```

79     if r.URL.Path == "/Camt053/Kontoauszugs-View" {
80         var kontoauszug = true
81         fmt.Println("Test 1")
82         err = readxmlfile(kontoauszug, formvalue, r, w)
83         fmt.Println("\nTest 3")
84         if err != nil {
85             http.Error(w, "Einlesen der XML Datei ist fehlgeschlagen", http.StatusBadRequest)
86             return
87         }
88     }
89
90     if r.URL.Path == "/Camt053/XML-View" {
91         var kontoauszug = false
92         fmt.Println("Test 1")
93         err = readxmlfile(kontoauszug, formvalue, r, w)
94         fmt.Println("\nTest 3")
95         if err != nil {
96             http.Error(w, "Einlesen der XML Datei ist fehlgeschlagen", http.StatusBadRequest)
97             return
98         }
99     }
100    http.Redirect(w, r, "/", http.StatusFound)
101 }
102
103 func getformvalue(r *http.Request) (string, error) {
104     err := r.ParseForm()
105
106     if err != nil {
107         log.Println("Error:", err)
108     }
109
110     formvalue := r.FormValue("XMLFile")
111     fmt.Println(formvalue)
112
113     if err != nil {
114         errors.New(fmt.Sprintf("Parse des Eingabefeldes XMLFile fehlgeschlagen: %s", err))
115     }
116
117     return formvalue, nil
118 }
119
120 func checkstring(s string) bool {
121
122     if (s >= "A" && s <= "Z") || (s >= "0" && s <= "9") || (s == "\n") {
123         return true
124     }
125
126     return false
127 }
128
129 func gettagname(get_tag string, w http.ResponseWriter) string {
130
131     value := regexp.MustCompile("(<.*?>))")
132
133     tags := value.FindString(get_tag)
134     rmrightbracket := strings.ReplaceAll(tags, ">", "")
135     tagname := strings.ReplaceAll(rmrightbracket, "<", "")
136
137     return tagname
138 }
139
140 func getxmlvalues(xmlcontent *xmlquery.Node, r *http.Request, w http.ResponseWriter) [200]string {
141
142     var i int
143     var xmlarr [200]string
144     for ; xmlcontent != nil; xmlcontent = xmlcontent.NextSibling {
145
146         get_xmlcontent := xmlcontent.InnerText()
147         get_xmltag := xmlcontent.OutputXML(true)
148
149         trim_xmlcontent := strings.Trim(get_xmlcontent, "\t \n")
150         rmnewlines := regexp.MustCompile("\n\n\n")
151         content_rmnewlines := rmnewlines.ReplaceAllString(trim_xmlcontent, "")
152         rmtab := regexp.MustCompile("\t")
153         content_rmtab := rmtab.ReplaceAllString(content_rmnewlines, "")
154         rmspaces2 := regexp.MustCompile("\n\n")
155         content_rmspaces2 := rmspaces2.ReplaceAllString(content_rmtab, "\n")
156         rmspaces3 := regexp.MustCompile("\n\n\n")
157         content_rmspaces3 := rmspaces3.ReplaceAllString(content_rmspaces2, "\n")

```

Anhang

```

157     content_rmspaces3 := rmspaces3.ReplaceAllString(content_rmspaces2, "\n")
158
159     check := checkstring(content_rmspaces3)
160
161     if check == true {
162
163         //An dieser Stelle alle Elemente und vllt der Tag Ausgabe soll die Zuweisung sein.
164
165         tagname := gettagname(get_xmltag, w)
166         tagvalue := content_rmspaces3
167
168         Map["tagname"] = tagname
169
170         //fmt.Fprintf(w, "%s\n", tagname)
171         //fmt.Fprintf(w, "%s\n", tagvalue)
172
173
174         xmlarr[i] = tagvalue
175         i++
176
177     }
178 }
179
180 //fmt.Fprintf(w, "%s\n", xmlarr)
181
182 return xmlarr
183 }
184
185 func readxmlfile(check bool, filename string, r *http.Request, w http.ResponseWriter) error {
186
187     fmt.Println("Test 2\n")
188
189     pwd, err := filepath.Abs("./XML/" + filename + ".xml")
190     if err != nil {
191         fmt.Println("Die XML Datei konnte nicht gefunden werden: %s", err)
192         return err
193     }
194
195     fmt.Println("Der Dateipfad ist: %s\n", pwd)
196     getxmlcontent, _ := os.Open(pwd)
197
198     parsexmlcontent, err := xmlquery.Parse(getxmlcontent)
199     if err != nil {
200         fmt.Println("Öffnen der XML Datei ist fehlgeschlagen: %s", err)
201         return err
202     }
203
204     fmt.Println("-----")
205
206     grphdr := xmlquery.FindOne(parsexmlcontent, "//BkToCstmrStmt/GrpHdr/MsgId")
207     grphdrarr := getxmlvalues(grphdr, r, w)
208
209     stmt := xmlquery.FindOne(parsexmlcontent, "//BkToCstmrStmt/Stmt/Id")
210     stmtarr := getxmlvalues(stmt, r, w)
211
212     //fmt.Fprintf(w, "%s\n", Map)
213     fmt.Println("%s\n", stmtarr)
214
215     if check == true {
216         printauszugcontent(w, grphdrarr, stmtarr)
217     }
218     if check == false {
219         printxmlcontent(w, grphdrarr, stmtarr)
220     }
221
222     return nil
223 }
224
225 func printauszugcontent(w http.ResponseWriter, grphdrarr [200]string, stmtarr [200]string) {
226     t := template.Must(template.New("Kontoauszugs-View").Parse(KontoauszugsView))
227
228     Nm := strings.Fields(grphdrarr[2])
229     IBAN := strings.Fields(stmtarr[4])
230     stmt1 := strings.Fields(stmtarr[6])
231
232     stmt2 := strings.Fields(stmtarr[8])
233
234     //fmt.Fprintf(w, "%s\n", IBAN)
235

```

Anhang

```

235     Map["MsgId"] = grphdrarr[0]
236     Map["CreDtIm"] = grphdrarr[1]
237     Map["Nm"] = Nm[0] + " " + Nm[1] + " " + Nm[2] + " " + Nm[3]
238
239     Map["IBAN"] = IBAN[0]
240
241     CreDtIm := grphdrarr[1]
242
243     Map["CreateDate"] = getdate(CreDtIm)
244     Map["Buchdt"] = getdate(stmt2[3])
245     Map["valdt"] = getdate(stmt2[4])
246
247     kntnr := IBAN[0]
248     Map["Kntnr"] = string(kntnr[len(kntnr)-7:])
249
250     blz := strings.Fields(strings.Trim(IBAN[0], "DE25 7404000"))
251     Map["BLZ"] = blz[0]
252
253     Map["BIC"] = IBAN[12]
254     Map["Kntst"] = stmt1[1]
255
256     Map["ums"] = stmt2[0]
257
258     Map["zweck"] = strings.ToUpper(stmt2[len(stmt2)-1])
259     Map["zwecknm"] += stmt2[41] + " " + stmt2[42] + " " + stmt2[43]
260     for i := 47; i < len(stmt2)-1; i++ {
261         Map["info"] += " " + stmt2[i]
262     }
263     Map["kndref"] = stmt2[11]
264     Map["manref"] = stmt2[13]
265
266     err := t.Execute(w, Map)
267     if err != nil {
268         http.Error(w, err.Error(), http.StatusInternalServerError)
269     }
270
271     Map = make(map[string]string)
272 }
273
274 func printxmlcontent(w http.ResponseWriter, grphdrarr [200]string, stmtarr [200]string) {
275     t := template.Must(template.New("XML-View").Parse(XMLView))
276
277     var grphdrcontent string
278     var stmtcontent string
279     var newgrphdrarr []string
280     var newstmtarr []string
281
282     newgrphdrarr = remove_empty_value (grphdrarr)
283
284     for i:=0; i < len(newgrphdrarr); i++ {
285         grphdrcontent += "\n" + newgrphdrarr[i]
286     }
287
288     newstmtarr = remove_empty_value (stmtarr)
289
290     for i:=0; i < len(newstmtarr); i++ {
291         stmtcontent += "\n" + newstmtarr[i]
292     }
293
294     Map["grphdr"] = grphdrcontent
295     Map["stmt"] = stmtcontent
296
297     err := t.Execute(w, Map)
298     if err != nil {
299         http.Error(w, err.Error(), http.StatusInternalServerError)
300     }
301 }
302
303 func remove_empty_value (arr [200]string) []string {
304     var newarr []string
305     for _, str := range arr {
306         if str != "" {
307             newarr = append(newarr, str)
308         }
309     }
310     return newarr
311 }
312
313 }

```

```

314 func getdate(datearr string) string {
315     getdate := string(datearr[0:10])
316     layout := "2006-01-02"
317     resdate, _ := time.Parse(layout, getdate)
318     dateString := resdate.Format("2.01.2006");
319
320     return dateString
321 }
322
323 }
324
325

```

A.12 webseite.go (Go-Quellcode)

```

1 package main
2
3 const webseite = `
4 <html>
5   <center>
6     <head>
7       <div class="header-container">
8         <center>
9           <link rel="stylesheet" type="text/css" href="/static/css/Webseite.css">
10          <title>CAMT053-Webservice</title>
11          <h1>CAMT053 - Webservice</h1>
12        </center>
13      </div>
14    </head>
15    <body>
16      <br></br>
17      <div class="container">
18        <form class="text-center" action="/CAMT053" method="post">
19          <fieldset>
20            <legend>Bitte die XML Datei angeben:</legend>
21            <label for="XMLFile">XML Datei:</label>
22            <input id="XMLFile" name="XMLFile" type="text" size="60" placeholder="Dateiname">
23          </fieldset>
24          <br></br>
25          <button type="submit" name="Einlesen" formaction="/Camt053/Kontoauszugs-View">Kontoauszug anzeigen</button>
26          <button type="submit" name="Einlesen" formaction="/Camt053/XML-View">XML anzeigen</button>
27        </form>
28      </div>
29    </body>
30  </center>
31 </html>
32 `

```

A.13 Webseite.css (css-Quellcode)

```

1 .header-container {
2   width: 100%;
3   border: 3px solid C42A70;
4   margin-left: auto;
5   margin-right: auto;
6 }
7
8 .container {
9   width: 100%;
10  margin-left: auto;
11  margin-right: auto;
12 }
13
14 .box {
15   float: left;
16   padding: 20px;
17   background: #eee;
18   margin-left: auto;
19   margin-right: auto;
20   box-sizing: border-box;
21   width: 30%;
22 }
23
24 .box3 {
25   float: right;
26   padding: 20px;
27   background: #eee;
28   margin-left: auto;
29   margin-right: auto;
30   box-sizing: border-box;
31   width: 70%;
32 }
33
34 button {
35   float: center;
36   width: 300px;
37   height: 40px;
38   font-weight: bold;
39   font-size: 115%;
40   color: darkmagenta
41 }
42
43 legend {
44   color: C42A70;
45   font-family: Arial;
46 }
47
48 fieldset {
49   border: 1px solid C42A70;
50   font-family: Arial;
51 }
52
53 label {
54   color: black;
55   font-style: Bold;
56   font-size: 83%;
57   font-family: Arial;
58 }
59
60 h1 {
61   color: C42A70;
62   font-family: Arial;
63 }
64
65 h4 {
66   color: C42A70;
67   font-family: Arial;
68 }
69
70 body {
71   background-color: C189A3;
72 }
73
74 p {
75   text-align: left;
76   font-family: Arial;
77 }
78

```

A.14 Kontoauszugs-View.go (Go-Quellcode)

```

1 package main
2
3 const KontoauszugsView = `
4     <html>
5         <center>
6         <head>
7             <div class="header-container">
8                 <link rel="stylesheet" type="text/css" href="/static/css/Kontoauszugs-View.css">
9                 <center>
10                    <title>CAMT053 - Webservice</title>
11                    <h1>CAMT053 - Webservice</h1>
12                </center>
13            </div>
14        </head>
15        </center>
16        <body>
17            <h4>Kontoinformationen</h4>
18            <div class="kontoinfo-container">
19                <div class="box2">
20                    <h4>Konto-Nr.</h4>
21                    <p>{{.Kntnr}}</p>
22                    <h4>BLZ</h4>
23                    <p>{{.BLZ}}</p>
24                </div>
25                <div class="box3">
26                    <h4>IBAN</h4>
27                    <p>{{.IBAN}}</p>
28                    <h4>BIC</h4>
29                    <p>{{.BIC}}</p>
30                </div>
31                <div class="box4">
32                    <h3>Kontostand am {{.CreateDate}}</h3>
33                </div>
34                <div class="box4">
35                    <h3>{{.Kntst}} EUR</h3>
36                </div>
37            <div style="clear:both;"></div>
38        </div>
39        <h4>Auswahlkriterien</h4>
40        <div class="kriterien-container">
41            <p>{{.CreateDate}}</p>
42        </div>
43        <h4>Umsätze im gewählten Zeitraum</h4>
44        <div class="table-container">
45            <table>
46                <tr>
47                    <th>Buchung</th>
48                    <th>Valuta</th>
49                    <th>Verwendungszweck</th>
50                    <th>Betrag</th>
51                </tr>
52                <tr>
53                    <td>{{.buchdt}}</td>
54                    <td>{{.valdt}}</td>
55                    <td>
56                        <b>{{.zweck}}</b>
57                        <br>
58                        {{.zwecknm}}
59                        <br>
60                        {{.info}}
61                        <br>
62                        KUNDENREFERENZ {{.kndref}}
63                        <br>
64                        MANDATSREFERENZ {{.manref}}
65                    </td>
66                    <td>{{.ums}} EUR</td>
67                </tr>
68            </table>
69        </div>
70        <br><br>
71        <center>
72            <form class="text-center" action="/Camt053/Kontoauszugs-View" method="post">
73                <button type="submit" name="Zurück" formaction="/">Zurück</button>
74            </form>
75        </center>
76    </body>
77 </html>
78 `

```

Anhang

A.15 Kontoauszugs-View.css (css-Quellcode)

```

1  .header-container {
2      width: 100%;
3      border: 3px solid C42A70;
4      margin-left: auto;
5      margin-right: auto;
6  }
7
8  .kontoinfo-container {
9      width: 100%;
10     border: 3px solid C42A70;
11     background-color: #eee;
12     margin-left: auto;
13     margin-right: auto;
14 }
15
16 .kriterien-container {
17     width: 100%;
18     border: 3px solid C42A70;
19     margin-left: auto;
20     margin-right: auto;
21 }
22
23 .table-container {
24     width: 100%;
25     border: 3px solid C42A70;
26     margin-left: auto;
27     margin-right: auto;
28 }
29
30 .box2 {
31     float: left;
32     padding: 20px;
33     background: #eee;
34     margin-left: auto;
35     margin-right: auto;
36     box-sizing: border-box;
37     width: 30%;
38 }
39
40 .box3 {
41     float: right;
42     padding: 20px;
43     background: #eee;
44     margin-left: auto;
45     margin-right: auto;
46     box-sizing: border-box;
47     width: 70%;
48 }
49
50 .box4 {
51     float: left;
52     padding: 20px;
53     background: #eee;
54 }
55
56 h1 {
57     color: C42A70;
58     font-family: Arial;
59 }
60
61 h3 {
62     font-family: Arial;
63 }
64
65 h4 {
66     color: C42A70;
67     font-family: Arial;
68 }
69
70 body {
71     background-color: C189A3;
72 }
73
74 button {

```

```

73
74 button {
75     float: center;
76     width: 200px;
77     height: 40px;
78     font-weight: bold;
79     font-size: 115%;
80     color: darkmagenta;
81 }
82
83 table {
84     width: 100%;
85 }
86
87 table, th, td {
88     border: 1px solid C42A70;
89     font-family: Arial;
90     background: #eee;
91 }
92
93 th {
94     background-color: BFAAB2;
95     color: #eee;
96 }
97
98 p {
99     text-align: left;
100    font-family: Arial;
101 }
102

```

A.16 XML-View.go (Go-Quellcode)

```

1 package main
2
3 const XMLView = `
4     <html>
5         <center>
6             <head>
7                 <div class="header-container">
8                     <link rel="stylesheet" type="text/css" href="/static/css/XML-View.css">
9                     <center>
10                        <title>CAMT053 - Webservice</title>
11                        <h1>CAMT053 - Webservice</h1>
12                    </center>
13                </div>
14            </head>
15            </center>
16            <body>
17                <h4>XML-View</h4>
18                <div class="xmlview-container">
19                    <h4>GRPHDR:</h4>
20                    <pre>{{.grphdr}}</pre>
21                    <br>
22                    <h4>STMT:</h4>
23                    <pre>{{.stmt}}</pre>
24                </div>
25                <br>
26                <center>
27                    <form class="text-center" action="/Camt053/XML-View" method="post">
28                        <button type="submit" name="Zurück" formaction="/">Zurück</button>
29                    </form>
30                </center>
31            </body>
32        </html>
33 `

```

A.17 XML-View.css (css-Quellcode)

```

1
2 .header-container {
3     width: 100%;
4     border: 3px solid C42A70;
5     margin-left: auto;
6     margin-right: auto;
7 }
8
9 .xmlview-container {
10     width: 100%;
11     border: 3px solid C42A70;
12     background-color: #eee;
13     margin-left: auto;
14     margin-right: auto;
15 }
16
17 h1 {
18     color: C42A70;
19     font-family: Arial;
20 }
21
22 h3 {
23     font-family: Arial;
24 }
25
26 h4 {
27     color: C42A70;
28     font-family: Arial;
29 }
30
31 body {
32     background-color: C189A3;
33 }
34
35 button {
36     float: center;
37     width: 200px;
38     height: 40px;
39     font-weight: bold;
40     font-size: 115%;
41     color: darkmagenta;
42 }
43
44 p {
45     text-align: left;
46     font-family: Arial;
47 }
48

```


A.18 Browseransicht: Hauptseite + Eingabefeld

CAMT053 - Webservice

Bitte die XML Datei angeben:

XML Datei:

Kontoauszug anzeigen

XML anzeigen

A.19 Browseransicht: Kontoauszugs-View

CAMT053 - Webservice

Kontoinformationen

Konto-Nr.	IBAN
<div style="background-color: black; width: 100px; height: 15px;"></div>	<div style="background-color: black; width: 150px; height: 15px;"></div>
BLZ	BIC
<div style="background-color: black; width: 100px; height: 15px;"></div>	<div style="background-color: black; width: 100px; height: 15px;"></div>
Kontostand am <div style="background-color: black; width: 100px; height: 15px;"></div> <div style="background-color: black; width: 50px; height: 15px;"></div> EUR	

Auswahlkriterien

Umsätze im gewählten Zeitraum

Buchung	Valuta	Verwendungszweck	Betrag
<div style="background-color: black; width: 100px; height: 15px;"></div>	<div style="background-color: black; width: 100px; height: 15px;"></div>	<div style="background-color: black; width: 500px; height: 15px;"></div>	<div style="background-color: black; width: 100px; height: 15px;"></div>

Zurück

A.20 Browseransicht: XML-View

CAMT053 - Webservice

XML-View

GRPHDR:

STMT:

Zurück

A.21 CAMT053 Datei (Auszug)

```

1  <?xml version="1.0" encoding="UTF-8" ?>
2  <Document xmlns="urn:iso:std:iso:2002:tech:xsd:camt.053.001.02" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
3    <BkToCstmRstmt>
4      <GrpHdr>
5        <MsgId>[REDACTED]</MsgId>
6        <CreDtTm>[REDACTED]</CreDtTm>
7        <MsgRcpt>
8          <Nm>[REDACTED]</Nm>
9          <PstlAdr>
10             <AdrLine>[REDACTED]</AdrLine>
11             <AdrLine>[REDACTED]</AdrLine>
12          </PstlAdr>
13        </MsgRcpt>
14        <MsgPgntn>
15          <PgNb>[REDACTED]</PgNb>
16          <LastPgInd>[REDACTED]</LastPgInd>
17        </MsgPgntn>
18      </GrpHdr>
19      <Stmt>
20        <Id>[REDACTED]</Id>
21        <ElectrncSegNb>[REDACTED]</ElectrncSegNb>
22        <CreDtTm>[REDACTED]</CreDtTm>
23        <FrToDt>
24          <FrDtTm>[REDACTED]</FrDtTm>
25          <ToDtTm>[REDACTED]</ToDtTm>
26        </FrToDt>
27        <Acct>
28          <Id>
29            <IBAN>[REDACTED]</IBAN>
30          </Id>
31          <Ccy>[REDACTED]</Ccy>
32          <Ownr>
33            <Nm>[REDACTED]</Nm>
34            <PstlAdr>
35              <AdrLine>[REDACTED]</AdrLine>
36              <AdrLine>[REDACTED]</AdrLine>
37            </PstlAdr>
38          </Ownr>
39          <Svcr>
40            <FinInstnId>
41              <BIC>[REDACTED]</BIC>
42              <Nm>[REDACTED]</Nm>
43              <Othr>
44                <Id>[REDACTED]</Id>
45                <Issr>[REDACTED]</Issr>
46              </Othr>
47            </FinInstnId>
48          </Svcr>
49        </Acct>
50        <Bal>
51          <Tp>
52            <CdOrPrtry>
53              <Cd>[REDACTED]</Cd>
54            </CdOrPrtry>
55          </Tp>
56          <Amt Ccy="[REDACTED]">[REDACTED]</Amt>
57          <CdtDbtInd>[REDACTED]</CdtDbtInd>
58          <Dt>
59            <Dt>[REDACTED]</Dt>
60          </Dt>
61        </Bal>
62        <Bal>
63          <Tp>
64            <CdOrPrtry>
65              <Cd>[REDACTED]</Cd>
66            </CdOrPrtry>

```

Anhang

```

67     </Tp>
68     <Amt Ccy=███></Amt>
69     <CdtDdtInd>███</CdtDdtInd>
70     <Dt>███</Dt>
71     <Dt>███</Dt>
72     </Dt>
73   </Bal>
74   <Bal>
75     <Tp>
76       <CdOrPrtry>
77         <Cd>███</Cd>
78       </CdOrPrtry>
79     </Tp>
80     <Amt Ccy=███></Amt>
81     <CdtDdtInd>███</CdtDdtInd>
82     <Dt>███</Dt>
83     <Dt>███</Dt>
84     </Dt>
85   </Bal>
86   <Ntry>
87     <Amt Ccy=███></Amt>
88     <CdtDdtInd>███</CdtDdtInd>
89     <Sts>███</Sts>
90     <BookgDt>
91       <Dt>███</Dt>
92     </BookgDt>
93     <ValDt>
94       <Dt>███</Dt>
95     </ValDt>
96     <AcctSvcrRef>███</AcctSvcrRef>
97     <BkTxCd>
98       <Domn>
99         <Cd>███</Cd>
100       <Fmly>
101         <Cd>███</Cd>
102         <SubFmlyCd>███</SubFmlyCd>
103       </Fmly>
104     </Domn>
105     <Prtry>
106       <Cd>███</Cd>
107       <Issr>███</Issr>
108     </Prtry>
109   </BkTxCd>
110   <NtryDtls>
111     <TxDtls>
112       <Refs>
113         <EndToEndId>███</EndToEndId>
114         <TxId>███</TxId>
115         <MndtId>███</MndtId>
116         <ClrSysRef>███</ClrSysRef>
117       </Refs>
118       <BkTxCd>
119         <Domn>
120           <Cd>███</Cd>
121         <Fmly>
122           <Cd>███</Cd>
123           <SubFmlyCd>███</SubFmlyCd>
124         </Fmly>
125       </Domn>
126       <Prtry>
127         <Cd>███</Cd>
128         <Issr>███</Issr>
129       </Prtry>
130     </BkTxCd>
131     <RltdPties>
132       <Dptr>

```

Anhang

```

132 <Dbtr>
133   <Nm> [REDACTED] </Nm>
134   <PstlAdr>
135     <Ctry> [REDACTED] </Ctry>
136     <AdrLine> [REDACTED] </AdrLine>
137     <AdrLine> [REDACTED] </AdrLine>
138   </PstlAdr>
139 </Dbtr>
140 <DbtrAcct>
141   <Id>
142     <IBAN> [REDACTED] </IBAN>
143   </Id>
144 </DbtrAcct>
145 <Cdtr>
146   <Nm> [REDACTED] </Nm>
147   <PstlAdr>
148     <Ctry> [REDACTED] </Ctry>
149     <AdrLine> [REDACTED] </AdrLine>
150     <AdrLine> [REDACTED] </AdrLine>
151   </PstlAdr>
152   <Id>
153     <PrvtId>
154       <Othr>
155         <Id> [REDACTED] </Id>
156       </Othr>
157     </PrvtId>
158   </Id>
159 </Cdtr>
160 <CdtrAcct>
161   <Id>
162     <IBAN> [REDACTED] </IBAN>
163   </Id>
164 </CdtrAcct>
165 <UltmtCdtr>
166   <Nm> [REDACTED] </Nm>
167 </UltmtCdtr>
168 </RltdPties>
169 <RltdAgts>
170   <DbtrAgt>
171     <FinInstnId>
172       <BIC> [REDACTED] </BIC>
173     </FinInstnId>
174   </DbtrAgt>
175   <CdtrAgt>
176     <FinInstnId>
177       <BIC> [REDACTED] </BIC>
178     </FinInstnId>
179   </CdtrAgt>
180 </RltdAgts>
181 <Purp>
182   <Cd> [REDACTED] </Cd>
183 </Purp>
184 <RmtInf>
185   <Ustrd> [REDACTED] </Ustrd>
186 </RmtInf>
187 </TxDtls>
188 </WtryDtls>
189 <AddtlNtryInf> [REDACTED] </AddtlNtryInf>
190 </Ntry>
191 <Ntry>
192   <Amt Ccy= [REDACTED] </Amt>
193   <CdtDbtInd> [REDACTED] </CdtDbtInd>
194   <Sts> [REDACTED] </Sts>
195   <BookgDt>
196     <Dt> [REDACTED] </Dt>
197 </BookgDt>


```

A.22 Benutzerdokumentation (Auszug)

Der folgende Auszug aus der Benutzerdokumentation werden die jeweils einzelnen Schritte erläutert, die bei der Bedienung des CAMT053 - Webservice relevant sind.

Schritt 1: XML-Datei importieren

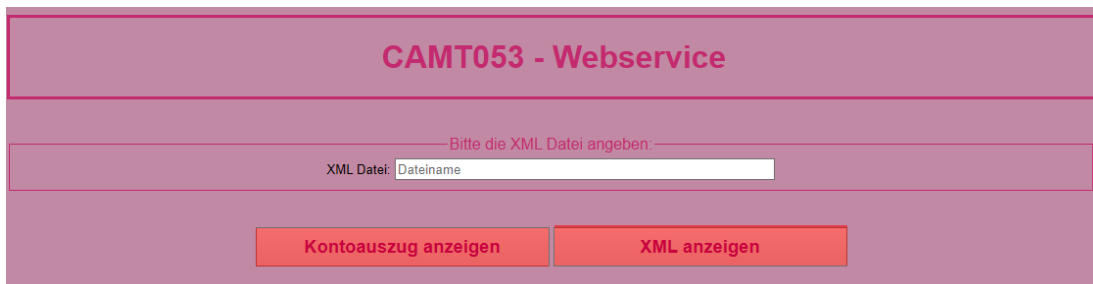
Um eine CAMT053-Datei in den Webservice importieren zu können, muss als erstes der Name der XML-Datei in das Eingabefeld eingegeben werden. Die Endung .xml ist bei der Eingabe nicht erforderlich.



The screenshot shows the 'CAMT053 - Webservice' interface. At the top, there is a header bar with the title 'CAMT053 - Webservice'. Below the header, there is a prompt 'Bitte die XML Datei angeben:'. Underneath this prompt is an input field labeled 'XML Datei:' with the placeholder text 'Dateiname'. At the bottom of the interface, there are two buttons: 'Kontoauszug anzeigen' and 'XML anzeigen'.

Schritt 2: Eingabe bestätigen

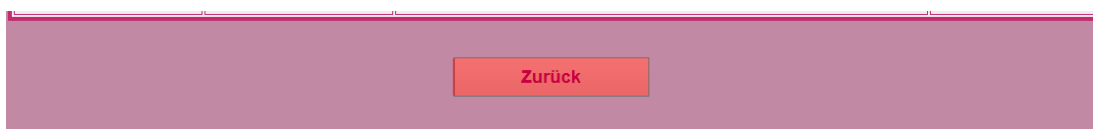
Nachdem die Eingabe der CAMT053-Datei abgeschlossen ist kann nun mit dem egl importieren begonnen werden. Dafür kann der unter dem Eingabefeld angegebene Button mit dem Namen: „XML anzeigen“ betätigt werden um die Weiterverarbeitung und die abschließende Ausgabe der Inhalte der CAMT053-Datei im Browser zu generieren. Möchte man nur die notwendigen Daten eines Kontoauszuges wie IBAN, BIC, Salden und Umsätze anzeigen, ist dafür der Button „Kontoauszug anzeigen zu betätigen“.



This screenshot is identical to the one in Schritt 1, showing the 'CAMT053 - Webservice' interface with the input field for the XML file name and the buttons 'Kontoauszug anzeigen' and 'XML anzeigen'.

Schritt 3: Neue Eingabe?

Möchte man als Anwender eine neue Eingabe durchführen ist unterhalb der generierten Ausgabe ein weiterer Button mit dem Namen: „Zurück“ zu finden. Nach Betätigung des Buttons kommt man zurück zur Startseite wo man eine neue Eingabe durchführen kann.



The screenshot shows a single button labeled 'Zurück' centered on a light blue background.

A.23 Entwicklerdokumentation (Read Me)

```
#####
# Titel:      Entwicklerdokumentation (Read Me)
# Programm:   CAMT053 - Webservice (Version 1.0)
# Autor:      Marvin Viedt
# Copyright:  ©a.b.s. Rechenzentrum GmbH
# Erstelldatum: 29.05.2020
#####

Einführung:

Willkommen zum CAMT053 - Webservice

Dieses Tool wurde für die a.b.s. Rechenzentrum GmbH als Resultat eines Abschlussprojektes einer Ausbildung erstellt.
Sinn und Zweck dieses Tools ist es CAMT053 Datei einzulesen und die darin enthaltenden Daten im Browser darzustellen.

Der CAMT053 Webservice bietet daher zwei Arten von Ansichten an:

    > Eine Kontoauszugs Ansicht
    > Eine XML Ansicht

#####

Installation:

Um den Webservice auf den lokalen Rechner zu installieren, muss zuerst das Verzeichnis camt053-Webservice entweder auf den Desktop/Verzeichnis kopiert werden.
Der Webservice muss dann über den Befehl ./camt053-Webservice in der Komandozeile (CLI) oder mit doppel klick auf der Ausführbaredatei (.exe) gestartet werden.
Zusätzliche Argumente werden nicht bei der Eingabe des Befehls benötigt.
#####

Ordnerstruktur:

Innerhalb des Verzeichnisses camt053-Webservice befinden sich folgende relevante .go Dateien:

main.go, webseite.go, XML-View.go und Kontoauszugs-View.go

Diese Dateien beinhalten lediglich den jeweiligen Go-Quellcode für den Webservice

In den anderen Dateien befindet sich hauptsächlich unbenutzter Code, der während der Entwicklung des Webservice archiviert wurde.

Die CSS Dateien sind im Verzeichnis static zu finden.

Einige Test CAMT053 Dateien sind im XML Verzeichnis zu finden.

#####

Die Funktionen in Main.go:

Folgende Funktionen die innerhalb der main.go Datei relevant sind:

main:
Hauptfunktion des Programms.
Hier wird mitgeteilt, dass der Server gestartet wurde und mit den Aufruf der Handlefunc und Handler Methoden wird der Webserver und die Webseiten etabliert.
Zudem werden die entsprechenden Templates für die Layouts der Webseiten geladen.

handler:
Darstellung und Etablierung der Willkommenseite mit dem Eingabefeld
Etablierung eines Cookies

kauszughandler:
Etablierung und Festlegung der beiden Browseransichten für die CAMT053 Datei

getformvalue:
Einlesen des Eingabewertes

checkstring:
Überprüfung eines Strings ob dieser nur Zahlen, Buchstaben und Newlines enthält.

gettagname:
Ermitteln und Speicherung eines Tagnamens

getxmlvalues:
Ermitteln und Speicherung der Tagnamen und Werte der CAMT053 Datei

readxmlfile:
Öffnen und Lesen der CAMT053 Datei
Festlegung der Startpunkte für den GrpHdr und Stmt Tag zum Durchlaufen der Werte
Speicherung ermittelten Werte in String Arrays
Prüfen welche Ansicht angegeben wurde.

printauszugscontent:
Speicherung der notwendigen Werte für die Kontoauszugs Ansicht in die Map
Etablierung des Layouts für die Kontoauszugs Ansicht

printxmlcontent:
Durchlaufen der Arrays für GrpHdr und Stmt für die anschließende Ausgabe
Etablierung des Layouts für die XML Ansicht

remove_empty_value:
Durchlaufen eines String Arrays
Entfernen von Leeren Werten aus einem String Array

getdate:
Umwandlung des Formats für das Datum (2019-03-25 zu 25.03.2019)

#####

#####

Todo:
Gilt für die Methoden readxmlfile und printauszugscontent
Durchlaufen der Untertags von GrpHdr und Stmt
Ermitteln der Tagnamen und Werte mit mehreren Schleifen.
Speicherung der Tagnamen und Werte in einer Map (Tagname, Tagvalue)

#####
```