

16. Dezember 2009



BANKDATENCHECK.DE
MIT SICHERHEIT VERKAUFEN!



mehr Zahlungssicherheit

Portunity Webrequest Kontonummern & Bankleitzahlencheck

Dokumentations- Version 2.9 [05.09.2011]

Webservice- Version 1.19.2568 [und höher]

Programmierhandbuch (Deutsch)

PORTUNITY

Portunity GmbH
Werner-Seelenbinder Str.23
42477 Radevormwald
Germany

<http://www.portunity.de>

Copyright © 2002 - 2010 by Portunity GmbH, Germany

Jede nicht genehmigte Vervielfältigung dieses Handbuches wird strafrechtlich verfolgt. Die Rechte an dieser Dokumentation und die Rechte an der Software liegen ausschließlich bei der Portunity GmbH.

Warenzeichen werden ohne Gewährleistung der freien Verwendbarkeit benutzt. „Portunity“, „Portalsuite“ und „Easybuilder“ sind eingetragene Warenzeichen der Portunity GmbH Deutschland. „Microsoft“, „Windows“ und „Internet Explorer“ sind eingetragene Warenzeichen der Microsoft Corporation USA.

Die in diesem Handbuch abgedruckten Beispielprogramme demonstrieren verschiedene Eigenschaften der Webservices. Sie dienen ausschließlich zu Lernzwecken. Sie dürfen diese Beispielprogramme bearbeiten und für den eigenen Gebrauch modifizieren. Sie dürfen die Beispielprogramme jedoch nicht als Ganzes oder in Teilen in kompilierter Form oder als Quelltext weitergeben oder verkaufen. Sie dürfen die Beispielprogramme in eigenen Anwendungen verwenden, solange die resultierenden Programme nicht einem Beispielprogramm oder einem Teil eines Beispielprogrammes in Erscheinung oder Funktion gleichen.

Es wird ferner keine Garantie für die Richtigkeit und Vollständigkeit des Inhaltes dieses Handbuches übernommen. Da sich Fehler, trotz aller Bemühungen, nie vollständig vermeiden lassen, sind wir für Hinweise jederzeit dankbar.

Herausgeber:

Portunity GmbH
Werner Seelenbinder Str.23
42477 Radevormwald
Germany

Telefon: +49 202 / 69555-0
Fax: +49 202 / 69555-190

eMail: info@portunity.de
Internetseite Deutschland: <http://www.portunity.de>

Dokumentation: Björn Rücker, Bernd Schnell, Christian Dahmen, Dennis Felderhoff
Korrektur und Qualitätssicherung: Christian Dahmen
Satz und typographische Gestaltung: Björn Rücker
Bilder: Björn Rücker

Inhaltsverzeichnis:

1	VORWORTE & VORABINFORMATIONEN	4
1.1	AUFBAU DES HANDBUCHES	4
1.2	SYMBOLE UND TYPOGRAPHIE IN DIESEM HANDBUCH	4
2	FUNKTIONSWEISE	5
2.1	SINN, ZWECK UND VORTEILE DES WEBSERVICES	5
2.2	ALLGEMEINE FUNKTIONSWEISE.....	6
2.3	HTTP-REQUEST & ANTWORT	7
2.4	UNTERSCHIED ZWISCHEN DEMOMODUS UND VOLLVERSION	16
2.5	TEST-BANKVERBINDUNGSDATEN	17
3	ANHANG	18
3.1	LINKS.....	18
3.2	ABBILDUNGSVERZEICHNIS	18
3.3	TABELLENVERZEICHNIS	18

1 Vorworte & Vorabinformationen

1.1 Aufbau des Handbuchs

Dieses Handbuch ist im Wesentlichen in zwei Bereiche aufgeteilt: Einem, in dem wir sprach- und plattformunabhängig die Funktionsweise und Ansteuerung unseres Webservices vorstellen und einem, in dem wir konkrete Umsetzungen und Beispielscripte in verschiedenen Programmiersprachen vorstellen.

Es richtet sich dabei im Wesentlichen an Programmierer, welche Vorkenntnisse in CGI-Sprachen haben.

1.2 Symbole und Typographie in diesem Handbuch

In dieser Dokumentation werden verschiedene Konventionen zur eindeutigen Kennzeichnung von Informationen verwendet. In der folgenden Tabelle finden Sie eine Beschreibung der typographischen Konventionen:

<i>Kursiv</i>	Kursive Schrift wird verwendet, um Programm Variablenamen und – Inhalte im Text hervorzuheben. Diese Typographie wird ebenfalls verwendet, wenn einzelne Wörter betont werden, wie beispielsweise wichtige oder neue Begriffe.
Programmcode	Programmcode steht in der Regel für beispielhaften HTML- oder Templatecode und ist zusätzlich noch eingerückt dargestellt. Aber auch im Installationskapitel zum Beispiel wird diese Typographie für die Konfigurationsdateiauszüge des Apacheservers verwendet. Die wichtigen Passagen, und bei den Templatecodes, die jeweils zum Vorbeispiel geänderten Passagen, sind dabei fett formatiert.

Tabelle 1: Typographietabelle

Weiterhin werden folgende Begriffe vereinbart:

- Browser bezeichnet, wie allgemein üblich, ein Programm, um im Internet zu surfen und Webseiten darzustellen bzw. den dahinter stehenden HTML-Code zu interpretieren.
- PHP ist eine serverseitige, plattformunabhängige HTML-embedded Scriptsprache. PHP selbst steht für „PHP: Hypertext Preprocessor“. Sofern nicht anders angegeben bezieht sich dieses Handbuch ausschließlich auf die aktuelle Version „5.x“.
- OOP ist eine Abkürzung und steht für „objektorientierte Programmierung“.

2 Funktionsweise

2.1 Sinn, Zweck und Vorteile des Webservices

Die Bezahlung per Bankeinzug ist in Deutschland die verbreitetste und beliebteste Zahlweise im Internet - weit vor Kreditkarten. Falsche Eingaben von Kontonummern und Bankleitzahlen können jedoch für Sie als Händler bzw. Serviceanbieter eine Rückbelastung Ihrer Bank mit zusätzlichen Bankgebühren von mehreren Euros bedeuten. Bei dem Einzug von Kleinstbeträgen kann so der Profit schnell wegschmelzen - viel schlimmer sind aber in einem Onlineshop Verluste durch bereits versendete Waren und der Ärger und Zeitaufwand diese zurückzuerhalten. Dabei ist es unerheblich, ob jemand absichtlich falsche Daten eingibt oder sich nur jemand aus Versehen vertippt.

Egal, ob Sie also nur einen Kleinanzeigenmarkt, ein Branchenbuch mit Anmeldefunktion, einen Onlineshop oder sonstige abzurechnende Onlineservices anbieten - die Prüfung der von Kunden angegebenen Bankverbindungsdaten macht auf jeden Fall Sinn.

Die deutschen Banken setzen jedoch über 100 verschiedene Prüfalgorithmen ein - und jede Bank wählt einen anderen aus diesem Pool. Mehrere tausend Programmzeilen sind nötig, um alle Prüfalgorithmen abzubilden. Dazu kommt, dass es auch in der Bankenlandschaft immer wieder zu Änderungen kommt.

Die Lösung ist der Portunity Webservice "Kontonummern- & Bankleitzahlencheck". Unser Server prüft nach den jeweils aktuellen Algorithmen Ihre Kontonummern und Bankleitzahlen. Dies geschieht dabei völlig unsichtbar im Hintergrund. Die Prüffunktion des Webservices wird dabei von Ihrem CGI-Script (Perl, ASP, PHP u.a.) über einen http-request abgefragt und Ihr Script auf Ihrem Server erhält in Sekundenbruchteilen das Ergebnis vom Webservice von unserem Server zur weiteren Verarbeitung.

Die Vorteile liegen auf der Hand: Auf Ihrer Seite werden nur wenige Programmzeilen zur Einbindung des Webservices benötigt (ca. 15-20 beispielsweise in der Sprache PHP), durch die zentrale Vorhaltung durch uns haben Sie automatisch immer die aktuellen Prüfroutinen im Einsatz, Bankänderungen werden ebenfalls automatisch von uns nachgeführt.

Die Webservice-Lösung ist dazu völlig plattform- und CGI-Sprachenunabhängig. In jede Sprache, aus der Sie einen http-Request absetzen können, kann der Webservice-Dienst eingebunden werden. Bestehende, gekaufte oder eigene CGI-Scripte können so einfach und schnell erweitert werden. Unsere eigene Applikation, die Portalsuite, bringt ab der Version 3.0.3 bereits eine Unterstützung für diesen Webservice "Out of the Box" mit.

Der Hauptvorteil ist jedoch, dass durch die Prüfung weit über 90% von Falscheingaben, Tippfehlern und Zahlendrehern in den Bankdaten von Ihnen rausgefiltert werden können.

Sie können den Webservice völlig kostenfrei und unverbindlich mit Hilfe dieser Dokumentation ausprobieren und testweise in Ihre Scripte integrieren. Dazu stehen mehrere Bankverbindungsdaten zu Testzwecken bereit. Um beliebige Bankverbindungsdaten prüfen zu können, müssen Sie dann nur noch Ihre IP-Adresse Ihres Servers für den Webservice freischalten lassen, und schon können Sie in sämtlichen Projekten auf dem freigeschalteten Server den Webservice ansteuern. Übermitteln Sie uns hierzu einfach abschließend einen ausgefüllten Nutzungsauftrag, welchen Sie online auf unserer Website finden.

2.2 Allgemeine Funktionsweise

Der Ablauf einer Bankverbindungsdaten-Überprüfung, z.B. innerhalb eines Bestellformulars in einem Onlineshop, und das Zusammenspiel der einzelnen Systeme haben wir in Abbildung 1 dargestellt:

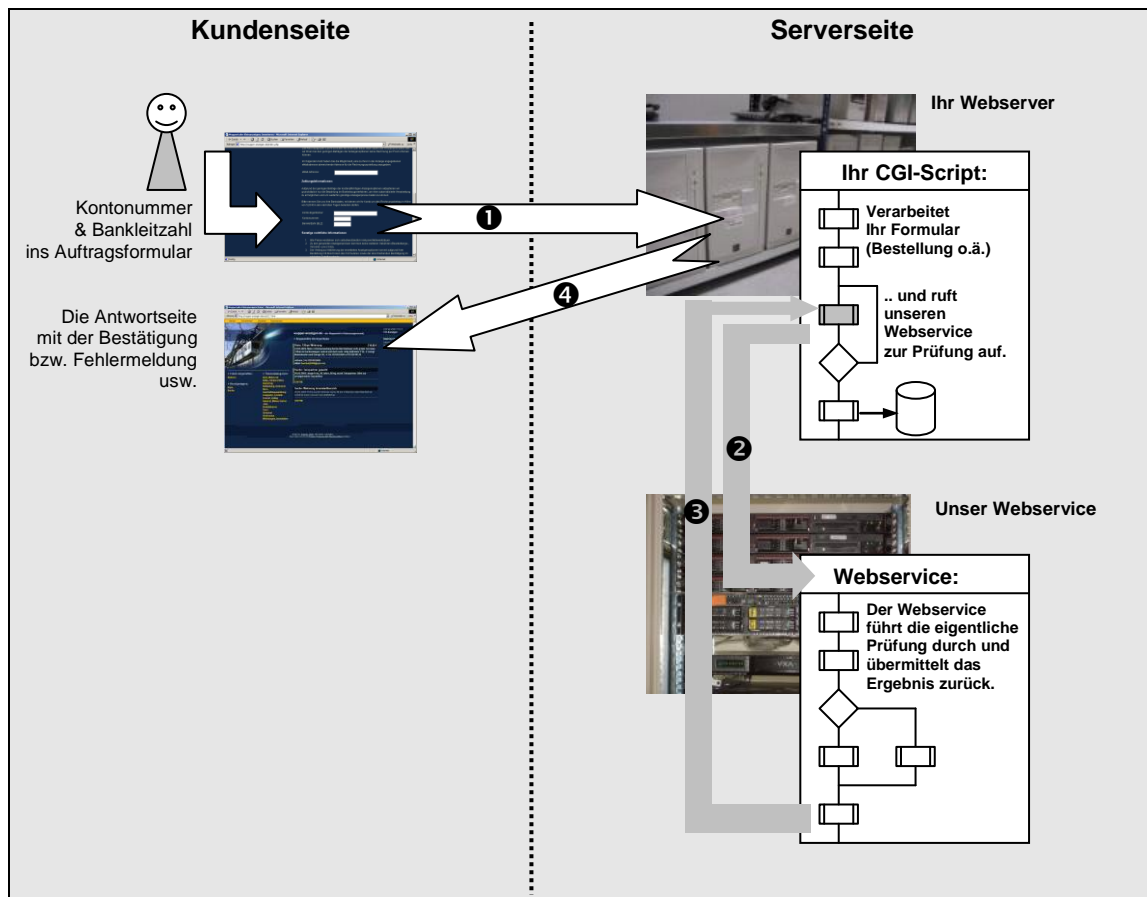


Abbildung 1: Allgemeine Funktionsweise des Webservices und des Zusammenspiels der Systeme

Der Kunde füllt ganz normal das Zahlungsdaten-Formular Ihres z.B. Shop-Systemes aus, u.a. mit den Daten „Kontonummer“ und „Bankleitzahl“. Durch Klick auf den Absendebuttton werden die Daten zusammen mit einer Anforderung einer Bearbeitungs-CGI-Seite zu Ihrem Server übertragen (1).

Die Daten werden auf Ihrem Server wie gewohnt von Ihrem CGI-Script in Empfang genommen. Irgendwo innerhalb Ihres Skriptes befindet sich dann auch die Prüfroutine für die Bankverbindungsdaten. Diese setzt nun einen Request – für den Kunden völlig unsichtbar – an unseren Webserver an das Webservice-Script ab und übermittelt die zu prüfende Kontonummer und Bankleitzahl (2).

Unser Server überprüft in Sekundenbruchteilen die erhaltenen Daten und erzeugt eine Antwortseite, welche an Ihr Script auf Ihrem Server zurückgesendet wird (3).

Die erhaltene Antwortseite kann dann von Ihnen ausgewertet werden und je nach Prüfergebnis können von Ihrem Script die Daten weiterverarbeitet werden oder der Kunde kann entsprechend von Ihrem Script informiert werden usw. Ihr Script schickt dann entsprechend eine Antwortseite für den weiteren Verlauf an den Kunden als Antwort auf das ausgefüllte Formular (4).

2.3 HTTP-Request & Antwort

Um eine deutsche Bankverbindung, bestehend aus Kontonummer und Bankleitzahl oder einer IBAN auf Plausibilität zu überprüfen, müssen Sie einen sogenannten http(s)-Request gegen unseren Webserver absetzen und die zu überprüfende Kontonummer und Bankleitzahl bzw. die IBAN übermitteln. Die URL des Webservices lautet:

Unverschlüsselt:

`http://wr.ispsuite.portunity.de/webrequests/product-bankcheck/`

Verschlüsselt:

`https://wr.ispsuite.portunity.de/webrequests/product-bankcheck/`

Zur Übermittlung von Parametern gibt es zwei Verfahren, welche Sie wahrscheinlich bereits von der Formularverarbeitung von HTML und CGI-Scripten kennen: GET und POST. Beim GET-Verfahren werden die Parameter an die URL in einer speziellen Syntax angehängt, beim POST-Verfahren werden die Parameter in dem Body des Http-Requests – für den Nutzer unsichtbar – mitgesendet. Da ein http-Request im GET-Verfahren in den meisten Programmiersprachen einfacher abzusetzen ist, haben wir uns für das GET-Verfahren entschieden, d.h. unser Webservice erwartet alle Parameter im GET-Verfahren übermittelt zu bekommen. Insgesamt gibt es folgende Parameter:

Parametername:	Beschreibung:
sKonto	Ist die zu überprüfende Kontonummer, bestehend aus numerischen Zahlen und ist ohne Leerzeichen und sonstige Trennzeichen zu übermitteln.
sBLZ	Ist die zur Kontonummer gehörige Bankleitzahl, bestehend aus numerischen Zahlen und ist ohne Leerzeichen und sonstige Trennzeichen zu übermitteln. Länge min. & max. = 8 Zahlen
sIBAN	Alternativ zur Kontonummer (sKonto) und BLZ (sBLZ) können Sie die IBAN (International Bank Account Number) des Kontos übergeben.
sProductLogin	Ist der Login-Name des Produktes. Wird auf dem Server benötigt um den Service im Vollversions-Modus aus zu führen.
sCode	Ist das Kennwort, welches Sie von uns erhalten, wenn Sie die volle Version verwenden möchten. Sie können diesen Parameter weglassen, in dem Fall arbeitet der Webservice dann im Demo-Modus und kann nur einige, von uns vorgegebene Bankverbindungsdaten überprüfen.
sFormat	Steuert das Ausgabeformat des Webservices. Derzeit gibt es für die Ausgabe das ASCII- und XML-Format, wobei das ASCII-Format voreingestellt ist und Antworten in der Regel einfacher zu parsen sind. sFormat kann entweder „ASCII“ oder „XML“ sein – andere Inhalte werden ignoriert und die Defaultbesetzung bleibt erhalten.

Tabelle 2: Parameter für den http-Request

Um einen Request beispielsweise abzusetzen, muß folgende URL mit den folgenden Parametern verwendet werden:

Vollversion: `https://wr.ispsuite.portunity.de/webrequests/product-bankcheck/?sProductLogin=12345&sCode=passwort&sBLZ=33080030&sKonto=7876543100&sFormat=ASCII`

Demo: `https://wr.ispsuite.portunity.de/webrequests/product-bankcheck/?sBLZ=33080030&sKonto=7876543100&sFormat=ASCII`

Tip: Sie können den Webservice auch von Ihrem Browser aus testweise ansteuern, indem Sie die URL samt Parameter einfach als Internetadresse eingeben. Bitte beachten Sie dabei nur, daß Sie dann wahrscheinlich meistens den Webservice nur im Demomodus ansprechen, es sei den Sie surfen auf einem IP - mäßig freigeschalteten Server.

Hinweis: Sie sollten in Ihrem Script bereits Sonderzeichen aus den Kundeneingaben entfernen und uns nur „saubere“ Daten einliefern. So sollten in der BLZ und Kontonummer nur Zahlen [0-9] enthalten sein. Sonderzeichen können den GET-Request ansonsten auch zerstören.

Als Antwort erhalten Sie im ASCII-Format vom Webservice zurück:

```
Version: 1.19.2568
VersionDatum: 2011-09-06
AbfrageFormat: ASCII
AbfrageIP: 127.0.0.1
AbfrageModus: VOLLVERSION
AbfrageID: 12345
AbfrageErgebnis: OK
CheckDatum: 2011-09-07
CheckZeit: 09:33:18
CheckBLZ: 33080030
CheckKontoNummer: XXXXXXXX
BundesbankDatenDatum: 2011-09-05
BundesbankDatenGueltig: 2011-12-04
ErgebnisBLZ: 1
ErgebnisKontoNummer: 1
ErgebnisIBAN: 1
ErgebnisKontoIBAN: DE6533080030XXXXXXXXXX
ErgebnisMethode: 00
ErgebnisBankName: Dresdner Bank Wuppertal
ErgebnisBankBezeichnung: Dresdner Bank Wuppertal
ErgebnisBankPLZ: 42003
ErgebnisBankStadt: Wuppertal
ErgebnisKontoIBAN: XXXXXXXXXXXXXXXXXXXX
ErgebnisBankPAN: 28330
ErgebnisBankBIC: DRESDEFF332
ErgebnisKumuliert: 1
```

und im XML-Format:

```
<Abfrage Format="XML" IP="127.0.0.1">
  <Info>
    <Version>1.19.2568</Version>
    <VersionDatum>2011-09-06</VersionDatum>
    <ID>12345</ID>
    <Modus>VOLLVERSION</Modus>
    <Ergebnis>OK</Ergebnis>
    <BundesbankDatenDatum>2011-09-05</BundesbankDatenDatum>
    <BundesbankDatenGueltig>2011-12-04</BundesbankDatenGueltig>
  </Info>
  <Check>
    <Datum>2011-09-07</Datum>
    <Zeit>09:33:18</Zeit>
    <BLZ>34051350</BLZ>
    <KontoNummer>XXXXXXXXXX</KontoNummer>
  </Check>
  <Ergebnis>
    <BLZ>1</BLZ>
    <KontoNummer>1</KontoNummer>
    <ErgebnisKumuliert>1</ErgebnisKumuliert>
    <IBAN>1</IBAN>
    <KontoIBAN>DE6533080030XXXXXXXXXX</KontoIBAN>
    <BankName>Sparkasse Radevormwald-Hückeswagen Zw</BankName>
    <BankKurzBezeichnung>00</BankKurzBezeichnung>
    <BankPLZ>00</BankPLZ>
    <BankStadt>00</BankStadt>
    <BankPAN>00</BankPAN>
    <BankBIC>00</BankBIC>
    <Methode>00</Methode>
  </Ergebnis>
</Abfrage>
```

Bitte beachten Sie, daß die einzelnen Zeilen mit einem Zeilenumbruch versehen sind (\n). Falls Sie den Request über einen Webbrowser abgesetzt haben, werden Sie das Ergebnis natürlich in einer langen Zeile hintereinander sehen, da die Zeilenumbruch-Tags (
) nicht vorhanden sind!

Die Rückgabewerte sind im ASCII-Modus folgendermaßen formatiert: Je Zeile (Trennzeichen wie gesagt \n) steht grundsätzlich ein Rückgabewert. Der **Name** des Rückgabewertes steht vorne. Rückgabename und Rückgabewert sind von einem „:“ (Doppelpunkt), gefolgt von einem Leerzeichen getrennt. Leerzeilen sind zu ignorieren. Sie sollten in Ihrem Script unbedingt den Namen vergleichen und dann den Inhalt übernehmen, da sich die Reihenfolge der Parameter auch mal ändern kann.

Hier nun eine Beschreibung der einzelnen Rückgabewerte im ASCII-Format:

Parametername:	Beschreibung:
Version	Gibt die Versionsnummer des Webservices zurück. <i>Sie können somit automatisch mitbekommen, wenn wir ein Update einspielen!</i>
AbfrageFormat	Gibt das verwendete Antwortformat zurück, also entweder „ ASCII “ oder „ XML “ und sollte normal mit dem angeforderten Format übereinstimmen. <i>Wird das XML-Format verwendet, weicht die gesamte Ausgabe natürlich erheblich von der ASCII-Ausgabe ab, siehe weiter unten.</i>

Parametername:	Beschreibung:
AbfrageIP	Die IP-Adresse, von der unser Webservice abgefragt wurde. Also die IP-Adresse Ihres Servers.
AbfrageModus	Der Modus, in dem der Webservice läuft. Also entweder „ DEMO “, falls die Anfrage mit einem falschen ProductLogin eingeht. Ansonsten „ VOLLVERSION “
AbfrageID	Die Kunden-ID-Nummer (bzw. Ihr Produktlogin) z.B. „4711“. Im Demomodus ist das Feld leer.
AbfrageErgebnis	Das generelle Ergebnis der Abfrage. <u>Nicht</u> das Ergebnis der Plausibilitätsprüfung der Kontodaten! Abfrage korrekt = „OK“ Bei falschem Passwort = „WRONG_PASSWORD“ Bei falscher IP-Adresse = „IP_NOT_ALLOWED“ Bei internen Problemen = „FAILED“
CheckDatum	Das Systemdatum des Webservers auf dem der Webservice läuft
CheckZeit	Die Systemzeit des Webservers auf dem der Webservice läuft
CheckBLZ	Falls Bankdaten (BLZ & Kontonummer) übergeben wurden - Die Bankleitzahl, welche überprüft wurde – sollte mit der angefragten Bankleitzahl übereinstimmen. <i>Wurde eine IBAN angefragt, so gibt es diesen Rückgabewert nicht.</i>
CheckKontoNummer	Falls Bankdaten (BLZ & Kontonummer) übergeben wurden - Die Kontonummer, welche überprüft wurde – sollte mit der angefragten Kontonummer übereinstimmen. <i>Wurde eine IBAN angefragt, so gibt es diesen Rückgabewert nicht.</i>
CheckIBAN	Falls eine IBAN übergeben wurde - Die IBAN, welche überprüft wurde – sollte mit der angefragten IBAN übereinstimmen. <i>Wurde keine IBAN angefragt gibt es diesen Rückgabewert nicht.</i>
BundesbankDatenDatum	Datum an dem die Daten herausgegeben wurden.
BundesbankDatenGueltig	Datum wie lange die Daten gültig sind.
ErgebnisBLZ	Das Ergebnis der Bankleitzahlenprüfung. Es handelt sich hierbei um einen Integerwert mit folgenden Rückgabewerten: -99 = nicht geprüft, da Abfrage im Demomodus 0 = Bankleitzahl ist ungültig 1 = Bankleitzahl ist gültig <i>Wenn die BLZ ungültig ist, kann die Kontonummer natürlich nicht überprüft werden. Sie wird dann auch „0“ zurückliefern.</i>

Parametername:	Beschreibung:
ErgebnisKontoNummer	<p>Das Ergebnis der Kontonummernprüfung. Es handelt sich hierbei um einen Integerwert mit folgenden Rückgabewerten:</p> <ul style="list-style-type: none"> -99 = nicht geprüft, z.B.: da Abfrage im Demomodus -9 = Schwerwiegender, interner Webservice-Fehler -8 = Schwerwiegender, interner Webservice-Fehler -7 = Schwerwiegender, interner Webservice-Fehler -6 = Schwerwiegender, interner Webservice-Fehler -5 = ungültige Bankleitzahl -4 = ungültige Kontonummer 0 = Bankleitzahl oder Kontonummer ist ungültig 1 = Bankleitzahl und Kontonummer ist gültig 2 = Kontonummer kann nicht überprüft werden¹ <p>Normalerweise sollten Sie nur die Rückgabewerte 0 oder 1 bzw. 2, oder im Demomodus noch „-99“ erhalten. Erhalten Sie -6 bis -9 als Rückgabewert, nehmen Sie bitte mit Portunity Kontakt auf und nennen dabei das zurückgegebene Systemdatum und Ihre IP-Adresse.</p>
ErgebnisKumuliert	<p>Zusammengefasstes Ergebnis: <i>(ab Version: 1.19)</i></p> <ul style="list-style-type: none"> 0 = Prüfergebnis nicht als in Ordnung zu werten 1 = Prüfergebnis als in Ordnung zu werten. <p>→ Wenn „ErgebnisBLZ“ = „1“ und „ErgebnisKontoNummer“ = „1“ oder = „2“ (siehe ggf. Fußnote).</p>
ErgebnisIBAN	<p>Das Ergebnis der IBAN-Prüfziffer - Prüfung. Es handelt sich hierbei um einen Integerwert mit folgenden Rückgabewerten:</p> <ul style="list-style-type: none"> -99 = nicht geprüft (z.B. weil nicht für Demomodus autorisiert) 0 = IBAN Prüfziffer ist ungültig 1 = IBAN Prüfziffer ist gültig <p>Hinweis: Dieser Wert ist lediglich eine Syntaxprüfung der übergebenen der IBAN, nicht die Plausibilitätsprüfung des eigentlichen Kontos! Zur Plausibilitätsprüfung des hinter der IBAN befindlichen Kontodaten fragen Sie bitte den Rückgabewert: "ErgebnisKontoNummer" ab (Momentan nur für deutsche Bankkonten verfügbar).</p>

¹ Für einige Kontonummern gibt es keine Prüfziffernkontrolle, die Kontonummern sind aber dann als richtig anzusehen. Dies gilt für die ErgebnisMethoden: **09, 99** und in Teilen für: 45, 57, 59, 68, 69, 78, 89, 95, 96, A0, B0, B7, B8, C5, D0, D2, D8 [Stand 06.09.2011, ab Version: 1.19]

Parametername:	Beschreibung:
ErgebnisKontoIBAN	Die International Bank Account Number des Kontos für den Internationalen Zahlungsverkehr. Portunity weist ausdrücklich darauf hin, das nur die Banken selber gültige IBANs herausgeben. Portunity übernimmt keine Gewähr im Falle einer falschen IBAN und für Schäden die durch die Benutzung entstehen.
ErgebnisBerechneteBLZ	Bei übergebener IBAN wird die daraus resultierende BLZ berechnet und zurückgegeben.
ErgebnisBerechneteKontoNummer	Bei übergebener IBAN wird die daraus resultierende Kontonummer berechnet und zurückgegeben.
ErgebnisMethode	Die Nummer des verwendeten Prüfalgorithmuses lt. Beschreibung der Bundesbank. <i>Diese kann aus zwei Teilen, der Basisnummer z.B. „57“ oder „A0“ sowie einem optionalen Suffix z.B.: „b“, bestehen „57b“</i>
ErgebnisBankName	Der Name der Bank von der die Bankleitzahl stammt.
ErgebnisBankKurzbezeichnung	Die Kurzbezeichnung der Bank.
ErgebnisBankPLZ	Die Postleitzahl der Bank.
ErgebnisBankStadt	Die Stadt der Bank.
ErgebnisBankPAN	Die Institutsnummer der Bank für den internationalen Kartenzahlungsverkehr mittels Bankkundenkarten.
ErgebnisBankBIC	Der Bank Identifier Code.

Tabelle 3: Ergebnisvariablen der Antwort auf einen http-Request im ASCII-Format

Hier die Beschreibung der einzelnen Rückgabewerte im XML-Format:

Element:	Beschreibung:
Container <Info>	Beinhaltet folgende Unterelemente.
<Version>	Gibt die Versionsnummer des Webservices zurück. <i>Sie können somit automatisch mitbekommen, wenn wir ein Update einspielen!</i>
<Versionsdatum>	Datum der Webservice-Version.
<ID>	Die Kunden-ID-Nummer (bzw. Ihr Produktlogin) z.B. „4711“. Im Demomodus ist das Feld leer.
<Modus>	Der Modus, in dem der Webservice läuft. Also entweder DEMO, falls die Anfrage von einer nicht autorisierten IP-Adresse gestartet wurde oder das übermittelte Kennwort nicht zur IP-Adresse paßt.
<Ergebnis>	Das generelle Ergebnis der Abfrage. <u>Nicht</u> das Ergebnis der Plausibilitätsprüfung der Kontodaten! Abfrage korrekt = „OK“ Bei falschem Passwort = „WRONG_PASSWORD“ Bei falscher IP-Adresse = „IP_NOT_ALLOWED“ Bei internen Problemen = „FAILED“
<BundesbankDatenDatum>	Datum an dem die Daten herausgegeben wurden.
<BundesbankDatenGueltig>	Datum wie lange die Daten gültig sind.
Container <Check>	Beinhaltet folgende Unterelemente.
<Datum>	Das Systemdatum des Webservers auf dem der Webservice läuft
<Zeit>	Die Systemzeit des Webservers auf dem der Webservice läuft
<BLZ>	Falls BLZ übergeben wurde - Die Bankleitzahl, welche überprüft wurde – sollte normal mit der angefragten Bankleitzahl übereinstimmen. <i>Wurde eine IBAN angefragt, so gibt es diesen Rückgabewert nicht.</i>
<KontoNummer>	Falls Kontonummer übergeben wurde - Die Kontonummer, welche überprüft wurde – sollte normal mit der angefragten Kontonummer übereinstimmen. <i>Wurde eine IBAN angefragt, so gibt es diesen Rückgabewert nicht.</i>
<IBAN>	Falls eine IBAN übergeben wurde - Die IBAN, welche überprüft wurde – sollte mit der angefragten IBAN übereinstimmen. <i>Wurde keine IBAN angefragt gibt es diesen Rückgabewert nicht.</i>

Element:	Beschreibung:
Container <Ergebnis>	Beinhaltet folgende Unterelemente.
<BLZ>	<p>Das Ergebnis der Bankleitzahlenprüfung. Es handelt sich hierbei um einen Integerwert mit folgenden Rückgabewerten:</p> <p>-99 = nicht geprüft (da z.B. nicht autorisiert)</p> <p>0 = Bankleitzahl ist ungültig</p> <p>1 = Bankleitzahl ist gültig</p> <p><i>Wenn die BLZ ungültig ist, kann die Kontonummer natürlich nicht überprüft werden. Sie wird dann auch „0“ zurückliefern.</i></p>
<KontoNummer>	<p>Das Ergebnis der Kontonummernprüfung. Es handelt sich hierbei um einen Integerwert mit folgenden Rückgabewerten:</p> <p>-99 = nicht geprüft, z.B.: da Abfrage im Demomodus</p> <p>-9 bis -6 = Schwerwiegender, interner Webservice-Fehler</p> <p>-5 = ungültige Bankleitzahl</p> <p>-4 = ungültige Kontonummer</p> <p>0 = Bankleitzahl oder Kontonummer ist ungültig</p> <p>1 = Bankleitzahl und Kontonummer ist gültig</p> <p>2 = Kontonummer wurde nicht überprüft ²</p> <p>Normalerweise sollten Sie nur die Rückgabewerte 0 oder 1 bzw. 2, oder im Demomodus noch „-99“ erhalten.</p> <p>Erhalten Sie -6 bis -9 als Rückgabewert, nehmen Sie bitte mit Portunity Kontakt auf und nennen dabei das zurückgegebene Systemdatum und Ihre IP-Adresse.</p>
<ErgebnisKumuliert>	<p>Zusammengefasstes Ergebnis: (ab Version: 1.19)</p> <p>0 = Prüfergebnis nicht als in Ordnung zu werten</p> <p>1 = Prüfergebnis als in Ordnung zu werten.</p> <p>→ Wenn „ErgebnisBLZ“ = „1“ und „ErgebnisKontoNummer“ = „1“ oder = „2“ (siehe ggf. Fußnote).</p>

² Für einige Kontonummern gibt es keine Prüfziffernkontrolle, die Kontonummern sind aber dann als richtig anzusehen. Dies gilt für die ErgebnisMethoden: **09, 99** und in Teilen für: 45, 57, 59, 68, 69, 78, 89, 95, 96, A0, B0, B7, B8, C5, D0, D2, D8 [Stand 06.09.2011, ab Version: 1.19]

Element:	Beschreibung:
<IBAN>	<p>Das Ergebnis der IBAN-Prüfziffer - Prüfung. Es handelt sich hierbei um einen Integerwert mit folgenden Rückgabewerten: -99 = nicht geprüft (z.B. weil nicht für Demomodus autorisiert) 0 = IBAN Prüfziffer ist ungültig 1 = IBAN Prüfziffer ist gültig</p> <p>Hinweis: Dieser Wert ist lediglich eine Syntaxprüfung der übergebenen der IBAN, nicht die Plausibilitätsprüfung des eigentlichen Kontos! Zur Plausibilitätsprüfung des hinter der IBAN befindlichen Kontodaten fragen Sie bitte den Rückgabewert: "ErgebnisKontoNummer" ab (Momentan nur für deutsche Bankkonten verfügbar).</p>
<KontoIBAN>	<p>Die International Bank Account Number des Kontos für den Internationalen Zahlungsverkehr. Portunity weist ausdrücklich darauf hin, das nur die Banken selber gültige IBANs herausgeben.</p> <p>Portunity weist ausdrücklich darauf hin, das nur die Banken selber gültige IBANs herausgeben. Portunity übernimmt keine Gewähr im Falle einer falschen IBAN und für Schäden die durch die Benutzung entstehen.</p>
<BerechneteBLZ>	Bei übergebener IBAN wird die daraus resultierende BLZ berechnet und zurückgegeben.
<BerechneteKontoNummer>	Bei übergebener IBAN wird die daraus resultierende Kontonummer berechnet und zurückgegeben.
<BankName>	Name der Bank von der die BLZ stammt.
<BankBezeichnung>	Kurzbezeichnung der Bank.
<BankPLZ>	Postleitzahl der Bank.
<BankStadt>	Stadt der Bank.
<BankPAN>	Die Institutsnummer der Bank für den internationalen Kartenzahlungsverkehr mittels Bankkundenkarten.
<BankBIC>	Der Bank Identifier Code.
<Methode>	<p>Die Nummer des verwendeten Prüfalgorithms lt. Beschreibung der Bundesbank.</p> <p><i>Diese kann aus zwei Teilen, der Basisnummer z.B. „57“ oder „A0“ sowie einem optionalen Suffix z.B.: „b“, bestehen „57b“</i></p>

Tabelle 4: Ergebnisvariablen der Antwort auf einen http-Request im XML-Format

2.4 Unterschied zwischen Demomodus und Vollversion

Der Unterschied zwischen dem Demomodus und der Vollversion ist gering, aber dennoch entscheidend.

Der Webservice kann von jeder beliebigen IP-Adresse aus im Demomodus angesteuert werden. Im Demomodus brauchen die Parameter „sProductLogin“ und „sCode“ beim http-Request nicht mit übermittelt werden. Im Demomodus sind allerdings nur einige, von uns autorisierte Test-Bankverbindungen zugelassen, welche im Wesentlichen für die ersten Tests, sowie zur Integration des Webservices in das eigene CGI-Script bzw. Webapplikation dienen. Werden andere Bankverbindungen angefragt, erhalten Sie als Rückgabewert in den Variablen „*ErgebnisBLZ*“ und „*ErgebnisKontoNummer*“ nur –99 zurück.

Um die Vollversion, und damit beliebige Bankverbindungsdaten auf Plausibilität zu prüfen, ist im http-Request der Parameter sCode mit zu übermitteln. Ein Rückgabewert von –99 in den Variablen „*ErgebnisBLZ*“ und „*ErgebnisKontoNummer*“ sollte im Vollversionsmodus nicht mehr ausgegeben werden.

Ob der Webservice im Demo- oder Vollversionsmodus arbeitet, können Sie anhand des Inhaltes des Rückgabewertes der Variable „*AbfrageModus*“ sehen.

Um die Vollversion nutzen zu können, müssen Sie die IP-Adresse Ihres Servers für unseren Webservice freischalten und beim HTTP-Request zusätzlich mit einem Kennwort autorisieren. Ein Auftragsformular sowie Preisinformationen finden Sie online auf unserer Internetseite im Bereich Media (<http://www.portunity.de/media/projekte/service/bankdatencheck.html>).

2.5 Test-Bankverbindungsdaten

Die folgenden Bankverbindungsdaten können zum Testen, auch im Demomodus, verwendet werden. Bei allen Kombinationen ist jeweils angegeben, ob diese eine lt. Prüfroutinen plausible oder nicht plausible Bankverbindung darstellt.

Bitte beachten Sie, daß es sich hierbei nicht um reale Bankverbindungsdaten handelt – bitte verwenden Sie diese deshalb auf keinem Fall in Transaktionen oder in der weiteren Verarbeitung. Die Kontonummern stammen von der Deutschen Bundesbank.

Sollte der unwahrscheinliche Fall eintreten, daß die von der Deutschen Bundesbank veröffentlichten Bankverbindungsdaten doch reale Bankverbindungsdaten sein sollten, benachrichtigen Sie uns bitte, dann werden wir diese selbstverständlich aus dieser Dokumentation entfernen. Da die Kontonummern aus der Dokumentation der Deutschen Bundesbank stammen, übernimmt Portunity keinerlei Haftung.

Bankleitzahl:	Kontonummer:	Bankname:	Algorithmus:	Plausibel:
33070024	4974161	Deutsche Bank Privat und Geschäftskunden	63b	NEIN
33080030	7876543100	Dresdner Bank	76a	JA
33080030	9012345600	Dresdner Bank	76a	JA
33080030	9012775600	Dresdner Bank	76b	NEIN
51350025	12345008	Sparkasse Gießen	10	JA
51350025	81237612	Sparkasse Gießen	10 PZ-Soll: 8	NEIN
52069246	9141405	<i>Unbekannt</i>	-	NEIN
52064156	9245500460	Raiffeisenbank	32	JA
52064156	6323423	Raiffeisenbank	32 PZ-Soll: 1	NEIN
76026000	5796800	Norisbank	C7c	JA
76026000	5796801	Norisbank	C7b PZ-Soll: 6	NEIN
70091500	2525259	Volksbank Raiffeisenbank Dachau	88	JA
70091500	1234567	Volksbank Raiffeisenbank Dachau	88 PZ-Soll: 0	NEIN

Tabelle 5: Kontonummern und Bankleitzahlen für Tests

3 Anhang

3.1 Links

- **Bankdatencheck:** <http://www.bankdatencheck.de>
- **Portunity GmbH:** <http://www.portunity.de>
- **Deutsche Bundesbank:** <http://www.bundesbank.de>

3.2 Abbildungsverzeichnis

Abbildung 1: Allgemeine Funktionsweise des Webservices und des Zusammenspieles der Systeme.. 6

3.3 Tabellenverzeichnis

Tabelle 1: Typographietabelle	4
Tabelle 2: Parameter für den http-Request	7
Tabelle 3: Ergebnisvariablen der Antwort auf einen http-Request im ASCII-Format.....	12
Tabelle 4: Ergebnisvariablen der Antwort auf einen http-Request im XML-Format.....	15
Tabelle 5: Kontonummern und Bankleitzahlen für Tests	17