

Schnittstellenbeschreibung

MDV BerSy Ticket Check

Versionshistorie

Version	Datum	Änderungen
1.0	06.12.2010	Initiale Version
1.1	12.04.2013	Schnittstellenerweiterung Semesterticket hinzugefügt
1.2	09.10.2013	Fehler in der Abfrage-URL korrigiert
1.3	12.11.2013	Übermittlung produktID hinzugefügt
1.4	19.05.2014	Fehlercodes für nicht bekannte Applikationen bzw. Berechtigungen hinzugefügt
1.5	28.11.2014	Erweiterung checkticketextended und Fehler-Response
1.6	26.07.2016	Änderung des Präfixes der Semesterticket Berechtigungs-ID
1.7	19.09.2017	Neue Anfrage checkTicketExtendedSTB
1.7.1	27.09.2017	checkTicketExtendedSTB Ergänzung um Felder „geburtsdatum“ und „name“
1.7.2	04.10.2017	Fehlercodes für checkticketextendedSTB definiert
1.7.3	19.10.2017	Überarbeitung RequestBody zur Kenntlichmachung der benötigten Felder je Request
1.7.4	21.11.2017	Redaktionelle Überarbeitung: Platzierung Kapitel 2.3.3 korrigiert
1.7.5	04.01.2018	Korrektur: Appinstanz-ID des Semestertickets
1.7.6	14.12.2018	Datentypen in RequestBody Beschreibung ergänzt; Verarbeitung der Daten von Semestertickets genauer spezifiziert

Inhaltsverzeichnis

<u>1</u>	<u>SICHERHEIT/ZERTIFIKATE</u>	<u>4</u>
<u>2</u>	<u>REQUEST</u>	<u>5</u>
2.1	REQUEST URL	5
2.2	REQUEST BODY	6
2.3	RESPONSE	8
2.3.1	RESPONSE FÜR ‚CHECKTICKET‘	8
2.3.2	RESPONSE FÜR ‚CHECKTICKETEXTENDED‘	9
2.3.3	RESPONSE FÜR ‚CHECKTICKETEXTENDEDSTB‘	11
2.3.4	RESPONSE FÜR ‚INVALIDATETICKET‘	11
2.3.5	FEHLER-RESPONSE	11
2.3.6	FEHLER-RESPONSE FÜR ‚CHECKTICKETEXTENDED‘	12
2.3.7	FEHLER-RESPONSE FÜR ‚CHECKTICKETEXTENDEDSTB‘	12

1 Sicherheit/Zertifikate

Das Midlet sendet sämtliche Anfragen über eine https-Verbindung an den Server. Damit diese Verbindung aufgebaut werden kann, müssen auf beiden Seiten geeignete Zertifikate installiert sein. Da auf mobilen Geräten keine weiteren Zertifikate installiert werden können, muss auf ein bereits vorhandenes zurückgegriffen werden. Auf Nokia-Geräten sind im Allgemeinen Root-Zertifikate der Firma VeriSign vorinstalliert, so dass serverseitig ein VeriSign Secure Site Zertifikat¹ benutzt werden muss. Mit Hilfe dieser Zertifikate wird der Handshake beim Verbindungsaufbau und somit eine sichere Kommunikation ermöglicht.

Die Initialisierung der Kommunikation muss vom Prüfer vor jedem Verbindungsaufbau explizit erlaubt werden. Dies ist eine Sicherheitsmaßnahme seitens J2ME, die verhindern sollen, dass unsigned Midlets ohne das Wissen des Anwenders eine Internetverbindung aufbauen und somit Kosten verursachen. Da das zu entwickelnde Midlet ebenfalls nicht digital signiert wird, muss diese Abfrage in Kauf genommen werden. Die Alternative ist ein digitales Codezertifikat der Firma VeriSign², dessen Kosten jedoch in keinem Verhältnis zu der erwarteten Verwendung des Midlets stehen.

Das Midlet kommuniziert ausschließlich über https-Verbindungen mit dem Server und muss sich über die eigene Telefonnummer des Endgerätes authentifizieren. Die Telefonnummer des Endgerätes, muss beim erstmaligen Programmstart eingegeben werden, sollte sie nicht automatisiert über JSR-177 ausgelesen werden können.

¹<http://www.verisign.de/ssl/buy-ssl-certificates/secure-site-services/index.html>

²<http://www.verisign.com/products-services/security-services/code-signing/digital-ids-code-signing/index.html>

2 Request

2.1 Request URL

Anfragen zum Ticket Check mittels des BerSy Mobile Ticket Check müssen an folgende URL entsprechend dem MDV BerSy Server gesendet werden:

```
https://<BersyServerUrl>/bersy/ticketValidation?action=<action>
```

Die <BersyServerUrl> lautet momentan:

- für das Produktiv-System: <https://bersy.perdata.de:444/>
(nicht ohne vorherige Rücksprachen mit dem MDV verwenden)
- für das Test-System: <https://bersy.perdata.de:9444/>

Für <action> sind folgende Werte gültig:

- `checkticket` - zum Abrufen der Ticketdaten und des Ticketstatus
- `checkticketextended` - zum Abrufen der Ticketdaten und des Ticketstatus mit erweiterten Inhalten
- `checkticketextendedstb` - zum Abrufen der Ticketdaten einer statischen Berechtigung
- `invalidateticket` - zum Markieren der Karte als ‚nicht lesbar‘

Ein gültiger URL-Aufruf zur Abfrage der Ticketdaten lautet damit beispielsweise

<https://bersy.perdata.de:9444/bersy/ticketValidation?action=checkticket>

bzw.

<https://bersy.perdata.de:9444/bersy/ticketValidation?action=checkticketextended>

oder für statische Berechtigungen

<https://bersy.perdata.de:9444/bersy/ticketValidation?action=checkticketextendedstb>

und ein URL-Aufruf zum Markieren einer Karte als nicht lesbar wie folgt

<https://bersy.perdata.de:9444/bersy/ticketValidation?action=invalidateticket>

2.2 Request Body

Der übermittelte Request Body muss zeilenweise folgende Daten enthalten, um die zu prüfende Karte festzulegen:

				checkticket	checkticketExtended	checkticketExtendedSTB
Beschreibung	Feld	Datentyp ³	Beispiel			
Telefonnummer des anfragenden Handys	phoneNumber=	String	0172666666	X	X	X
Applikations-ID	appInstanceId=	Long	17445891	X	X	
Berechtigungs-ID	entitlementId=	Long	123456789			X
Prüfziffer (nach Luhn)	checkDigit=	Integer	0	X	X	
Organisations-ID	orgId=	Integer	6053	X	X	X

Beispiel für einen Request-Body (checkTicket, checkTicketExtended):

```
phoneNumber=0172666666
appInstanceId=17445891
checkDigit=0
orgId=6053
```

³ Die Ausprägung der Datentypen entspricht den Standards in der Programmiersprache Java 8. Der Wertebereich für Long liegt bei -2^{64} bis $2^{64}-1$. Der Wertebereich für Integer liegt bei -2^{31} bis $2^{32}-1$.

Beispiel für einen Request-Body (checkTicketExtendedSTB):

```
phoneNumber=0172666666  
entitlementId=123456789  
orgId=6053
```

Besonderheit bei der Übermittlung von Semesterticketdaten

Der Request-Body hat für die Semestertickets dieselbe Form. Da jedoch hier nur die *UID* aus der Karte ausgelesen bzw. abgetippt wird, wurden für die übrigen Felder folgende Standardwerte festgelegt. Für die Prüfziffer wird die -1 festgelegt. Für die Organisations-ID wird die 6053 festgelegt. Der Applikations-ID wird zudem die Ziffer 3 vorangestellt.

```
phoneNumber= [wie bei Umweltcard]  
appInstanceId=3.....  
checkDigit=-1  
orgId=6053
```

2.3 Response

2.3.1 Response für ‚checkticket‘

Der Response-Body enthält für jede Berechtigung und jede Aktion die zur angefragten Applikation gehört eine Zeile. In einer Zeile sind jeweils die Daten für die Berechtigung oder Aktion mittels ‚|‘ separiert enthalten. Die Daten sind wie folgt abgelegt:

für eine Berechtigung auf einer lesbaren Karte:

```
0|<infotext1>|<infotext2>|<zonen1>|<zonen2>|<zonen3>|<status>|<gültig von>|<gültig  
bis>|<name>|<geburtsdatum>|||produktID
```

für eine Berechtigung auf einer als bereits als nicht-lesbar markierten Karte:

```
1|<infotext1>|<infotext2>|<zonen1>|<zonen2>|<status>|||<name>|<geburtsdatum>|<ungültig ab>|<ungültig  
seit>|produktID
```

für eine Aktion:

```
2|<infotext1>|<infotext2>|<zonen1>|<zonen2>|<zonen3>|<status>|<gültig von>|<gültig  
bis>|<name>|<geburtsdatum>|||produktID
```

Die einzelnen Datenfelder enthalten dabei folgende Werte:

Datenobjekt	Inhalt
0,1,2	Typ des Datensatzes 0 - Berechtigung 1 - Berechtigung auf einer bereits als nicht-lesbar markierten Karte 2 - Aktion
<infotext1>, <infotext2>	Info-Texte der Berechtigung oder Aktion
<zonen1>, <zonen2>, <zonen3>	Gültigkeitszonen der Berechtigung oder Aktion jeweils durch ein Leerzeichen getrennt. Eine nicht vorhandene Zone ist entweder leer oder durch eine ‚0‘ gekennzeichnet.
<status>	Status der Berechtigung oder Aktion
<gültig von>, <gültig bis>	Gültigkeitsdatum der Berechtigung oder Aktion in der Form ‚DD.MM.YYYY‘
<name>	Codierter Name des Karteninhabers
<geburtsdatum>	Geburtsdatum des Karteninhabers
<ungültig ab>	Gibt den Zeitpunkt an, zu dem eine nicht-lesbar gemelde-

	te Karte erstmalig als nicht-lesbar gemeldet wurde im Format ,DD.MM.YYYY HH:MM:SS‘
<ungültig seit>	Aktuelle Zeitdifferenz zum <ungültig ab> Datum im Format ,DD d HH h MM m‘ (z.B. „30 d 10 h 53 min“)
<produktID>	Produkt-ID aus der Produktdatenbank

Beispiele:

```
0|Abo-Monatskarte|Jedermann|210 210 0 |0 0 ||aktiv|03.08.2009|10.12.2013|Hxxxxxs,
Mxxxxy|09.09.1986|||5843
```

```
1|Abo-Monatskarte|Jedermann|210 210 0 |0 0 |aktiv|||Hxxxxxs, Mxxxxy|09.09.1986|08.11.2010
02:34:56|30 d 10 h 53 min|5843
```

2.3.2 Response für „checkticketextended“

Der Response-Body ist ähnlich aufgebaut wie der Response-Body von „checkticket“. Das Feld <orgid_pv> wird bei allen Datensatztypen(0,1,2) zusätzlich hinter dem Feld <produktID> mitgeliefert. Desweiteren soll am Ende aller Datensatztypen ein neues Feld <ueberzonen> hinzugefügt werden.

Speziell im Datensatztyp1 (Berechtigung auf als nicht-lesbar markierten Karte) werden hier nun auch die Felder <gültig von> und <gültig bis> befüllt und das Feld <zonen3> wird hinter dem Feld <zone2> hinzugefügt.

Für eine Berechtigung auf einer lesbaren Karte:

```
0|<infotext1>|<infotext2>|<zonen1>|<zonen2>|<zonen3>|<status>|<gültig von>|<gültig
bis>|<name>|<geburtsdatum>|||<produktID>|<orgid_pv>|<ueberzonen>
```

für eine Berechtigung auf einer als bereits als nicht-lesbar markierten Karte:

```
1|<infotext1>|<infotext2>|<zonen1>|<zonen2>| <zonen3>|<status>|<gültig von>|<gültig bis>
|<name>|<geburtsdatum>|<ungültig ab>|<ungültig seit>|<produktID>|<orgid_pv>|<ueberzonen>
```

für eine Aktion:

```
2|<infotext1>|<infotext2>|<zonen1>|<zonen2>|<zonen3>|<status>|<gültig von>|<gültig
bis>|<name>|<geburtsdatum>|||<produktID>|<orgid_pv>|<ueberzonen>
```

für eine Aktion auf einer als bereits als nicht-lesbar markierten Karte:

```
3|<infotext1>|<infotext2>|<zonen1>|<zonen2>|<zonen3>|<status>|<gültig von>|<gültig
bis>|<name>|<geburtsdatum>|<ungültig ab>|<ungültig seit>|<produktID>|<orgid_pv>|<ueberzonen>
```

Die einzelnen Datenfelder enthalten dabei folgende Werte:

Datenobjekt	Inhalt
0,1,2,3	Typ des Datensatzes 0 - Berechtigung 1 - Berechtigung auf einer bereits als nicht-lesbar markierten Karte 2 - Aktion 3- Aktion auf einer bereits als nicht-lesbar markierten Karte
<infotext1>, <infotext2>	Info-Texte der Berechtigung oder Aktion
<zonen1>, <zonen2>, <zonen3>	Gültigkeitszonen der Berechtigung oder Aktion jeweils durch ein Leerzeichen getrennt. Eine nicht vorhandene Zone ist entweder leer oder durch eine ‚0‘ gekennzeichnet.
<status>	Status der Berechtigung oder Aktion
<gültig von>, <gültig bis>	Gültigkeitsdatum der Berechtigung oder Aktion in der Form ‚DD.MM.YYYY‘
<name>	Codierter Name des Karteninhabers
<geburtsdatum>	Geburtsdatum des Karteninhabers
<ungültig ab>	Gibt den Zeitpunkt an, zu dem eine nicht-lesbar gemeldete Karte erstmalig als nicht-lesbar gemeldet wurde im Format ‚DD.MM.YYYY HH:MM:SS‘
<ungültig seit>	Aktuelle Zeitdifferenz zum <ungültig ab> Datum im Format ‚DD d HH h MM m‘ (z.B. „30 d 10 h 53 min“)
<produktID>	Produkt-ID aus der Produktdatenbank
<orgid_pv>	Gibt die DL_ORG_ID des Verkehrsunternehmens des aktuellen Produktes an
<ueberzonen>	Bei bis zu 5 Zonen setzt sich dieses Feld jeweils aus: arealID-Zone1 arealID-Zone2 ... arealID-Zone5 zusammen. Bei mehr als 5 Zonen setzt sich dieses Feld aus: corridorId-Zeigerüberzone corridorId-0 corridorId-0 zusammen. ArealID = 16 corridorID= 17

Beispiel für Datensatztyp 0 mit bis zu 5 Zonen:

```
0|Abo-Monatskarte|Jedermann|210 210 0 |0 0 ||aktiv|03.08.2009|10.12.2013|Hxxxxxxs,  
Mxxxxy|09.09.1986|||5843 |6055 | 16-210 16-210 16-0
```

Beispiel für Datensatztyp1 mit mehr als 5 Zonen:

```
1|Abo-Monatskarte|Jedermann| 147 152 155 |242 258 261||aktiv|12.10.2012|01.03.2014|Hxxxxxxs,  
Mxxxxy|09.09.1986|08.11.2010 02:34:56|30 d 10 h 53 min|5843 | 6055 | 17-10058 17-0 17-0
```

2.3.3 Response für „checkticketextendedSTB“

Der Response-Body ist äquivalent dem Response-Body von „checkticketextended“ aufgebaut.

Für eine statische Berechtigung:

```
0|<infotext1>|<infotext2>|<zonen1>|<zonen2>|<zonen3>|<status>|<gültig von>|<gültig  
bis>|<name>|<geburtsdatum>|||produktID|<orgid_pv>|<ueberzonen>
```

2.3.4 Response für ‚invalidateticket‘

Der Response-Body enthält keine Daten.

2.3.5 Fehler-Response

Im Fall eines auftretenden Fehlers werden durch das BerSy folgende HTTP-Error-Codes zurückgesendet:

HTTP Error Code	Bedeutung
BAD_REQUEST(400)	Falscher oder fehlender Parameter
BAD_REQUEST(400)	Die angefragte Applikation/Karte ist dem BerSy nicht bekannt
BAD_REQUEST(400)	Zur angefragten Applikation/Karte liegen keine Berechtigungen vor
SC_FORBIDDEN(403)	Ungültige oder unbekannte Handynummer
SC_INTERNAL_SERVER_ERROR(500)	Fehler beim Lesen des Requests oder interner Fehler

2.3.6 Fehler-Response für ‚checkticketextended‘

Im Falle eines auftretenden Fehlers werden in diesem Anfragetypen nun keine HTTP-Error-Codes zurück-gesendet sondern neue Datensatztypen eingeführt, diese setzen sich wie folgt zusammen:

Response	Bedeutung
900 invalid parameters	Falscher oder fehlender Parameter
901 unknown application	Die angefragte Applikation/Karte ist dem BerSy nicht bekannt
902 no entitlement	Zur angefragten Applikation/Karte liegen keine Berechtigungen vor
903 invalid phonenumber	Ungültige oder unbekannte Handynummer
904 internal error	Fehler beim Lesen des Requests oder interner Fehler

2.3.7 Fehler-Response für ‚checkticketextendedstb‘

Ähnlich zu checkticketextended werden die folgenden Datensatztypen für die Fehlerantwort genutzt:

Response	Bedeutung
900 invalid parameters	Falscher oder fehlender Parameter
901 unknown entitlement	Die angefragte statische Berechtigung ist dem BerSy nicht bekannt
903 invalid phonenumber	Ungültige oder unbekannte Handynummer
904 internal error	Fehler beim Lesen des Requests oder interner Fehler