

Richtlinie XX.2023: Überwachungs- und Berichterstattungssystem für Wetten

Dieses Dokument enthält die Anforderungen, die Buchmacher bei der Meldung von Wettdaten an die Nationale Wettbehörde Zyperns erfüllen müssen.

API-Datenanforderungen – Benutzerhandbuch

Versionshistorie

Version	Datum	Beschreibung
1.0	7.8.2023	-

Inhaltsverzeichnis

Inhalt

Inhaltsverzeichnis.....	3
Richtlinie XX.2023: Überwachungs- und Berichterstattungssystem für Wetten.....	4
Anhang A: Technische Anforderungen.....	5
1.0 Einleitung.....	5
2.0 Das BMRS-Datenmodell.....	6
2.1 Buchmacher, autorisierte Vertreter und Geschäfte (Lizenznehmer).....	8
2.2 Spieler und Konten.....	8
2.3 Der Wettschein.....	9
3.0 Die API-Prozesse des BMRS.....	12
3.1 Spielerkonten.....	14
3.1.1 Methode „Konten speichern“.....	14
3.2 Kontotransaktionen.....	20
3.2.1 Methode „Kontotransaktionen erstellen“.....	20
3.3 Kontobeschränkungen.....	26
3.3.1 Methode „Kontobeschränkungen speichern“.....	27
3.4 Wettaktivitäten.....	32
3.4.1 Methode „Wettscheine erstellen“.....	33
3.4.3 Methode „Wettscheine aktualisieren“.....	44
3.4.5 Methode „Wettlinien speichern“.....	53
3.5 Systemaufrufe.....	57
3.5.1 Methode „Heartbeat“.....	57
3.5.2 Methode Fehlerprotokolle abrufen.....	58
4.0 BMRS-Kommunikationsprotokolle.....	62
4.1 Primäres Kommunikationsprotokoll – Die BMRS-API.....	62
4.2 Sekundäres Kommunikationsprotokoll – XML-Dateien.....	64

Richtlinie XX.2023: Überwachungs- und Berichterstattungssystem für Wetten

Diese Richtlinie wird von der Nationalen Wettbehörde (nachfolgend „NBA“) gemäß den Bestimmungen von Artikel 15 Buchstabe b des Wettgesetzes von 2009 (L.37(I)/2019) in der geänderten oder ersetzten Fassung ausgearbeitet und herausgegeben und enthält alle technischen Anforderungen und Verfahren, die Lizenznehmer der Klassen A und B anwenden müssen, um sich mit der NBA zu verbinden und über das Überwachungs- und Berichterstattungssystem für Wetten (Betting Monitoring and Reporting System der NBA, im Folgenden „BMRS“) Informationen an sie zu übermitteln. Sie schreibt die Arten von Daten vor, die übertragen werden müssen, sowie die Instanzen oder Zeitrahmen, die die Übertragung auslösen.

2. Alle Lizenznehmer der Klassen A und B müssen —

- a) alle technischen Anforderungen und Verfahren nach Anhang A annehmen und umsetzen;
- b) sich gemäß Anhang A an das BRMS anschließen,
- c) gemäß den in Anhang A vorgeschriebenen Daten und Instanzen Daten übermitteln.

3. Diese Richtlinie ist verbindlich und ihre Umsetzung ist für alle Lizenznehmer der Klassen A und B verpflichtend.

Anhang A: Technische Anforderungen

1.0 Einleitung

Das BMRS ist eine Überwachungsplattform, die von der NBA entwickelt wurde, um ihren datenzentrierten Regulierungsrahmen zu verbessern, indem sie die Aktivitäten aller Teilnehmer des Wettmarktes in nahezu Echtzeit genau verfolgt. Von Buchmachern, die Wettaktivitäten entweder online (Lizenznehmer der Klasse B) oder durch autorisierte Vertreter und Einzelhandelsgeschäfte (Lizenznehmer der Klasse A) erleichtern, wird erwartet, dass sie über eine geeignete API eine kontinuierliche Kommunikation mit der Plattform pflegen und alle Informationen übermitteln, die die NBA für notwendig hält, um ihre regulatorische Aufgabe zu erfüllen.

Dieses Dokument beschreibt detailliert die verschiedenen Prozesse, die das System den Endnutzern (Lizenznehmern) zur Verfügung stellt, den Zweck und die Datenfelder, die jeder dieser Prozesse benötigt, die erwartete Form, die Datenübertragungen erfüllen müssen (XML-Schema), Antworten und mögliche Fehler, die sich aus der Datenvalidierung ergeben können, sowie eine allgemeine Beschreibung der verwendeten Kommunikationsprotokolle.

2.0 Das BMRS-Datenmodell

Das BMRS erfasst und verarbeitet Daten in einer Weise, die sowohl einen aufschlussreichen Überblick über Markttrends und -verhalten als auch gegebenenfalls eine detaillierte Aufschlüsselung der Wettaktivitäten auf individueller Ebene ermöglicht. Sein Design basiert auf der Modellierung bestimmter Schlüsseleinheiten und der Prozesse, durch die sie miteinander interagieren, um gemeinsam die Abläufe des Wettgeschäfts zu gestalten. Ein Umriss des Modells ist in Abbildung 2.1 dargestellt.

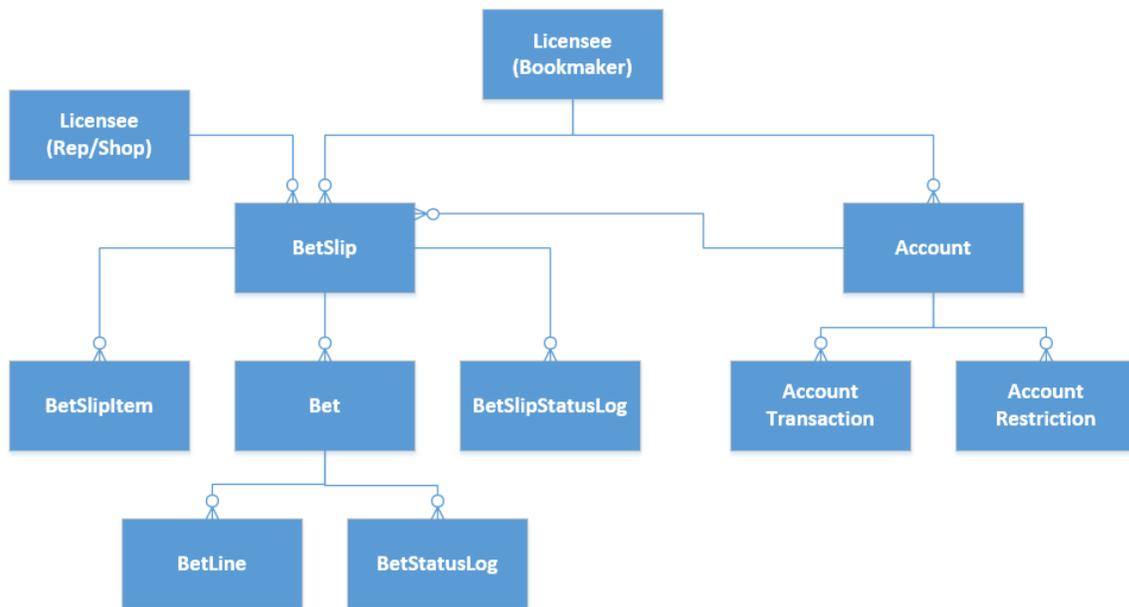


Abbildung 2.1: Überblick über das BMRS-Datenmodell.

Licensee (Rep/Shop)	Lizenznehmer (Vertreter/Geschäft)
Licensee (Bookmaker)	Lizenznehmer (Buchmacher)
BetSlip	Wettschein
Account	Konto
BetSlipItem	Wettscheinelement
Bet	Wette
BetSlipStatusLog	Wettscheinstatusprotokoll
Account Transaction	Kontotransaktion
Account Restriction	Kontoeinschränkung
BetLine	Wettlinie
BetStatusLog	Wettstatusprotokoll

Für jede Wette gibt es zwei Gegenparteien, nämlich den Buchmacher (Lizenznehmer), der bestimmte Quoten für ein bestimmtes Ergebnis eines zukünftigen zugrunde liegenden Ereignisses anbietet, und den Spieler, der diese Quoten akzeptiert und beschließt, einen bestimmten Betrag darauf zu setzen. Alle Informationen in Bezug auf eine bestimmte Wette sind auf einem Wettschein enthalten, der eine

gültige Form der Vereinbarung wird, sobald er vom Spieler eingereicht und vom Buchmacher angenommen wird. Danach folgt der Wettschein einem bestimmten Lebenszyklus, der entweder endet, wenn alle Ereignisse, auf die der Spieler gewettet hat, ohne Auszahlung abgeschlossen sind, oder nach der Auszahlung des Gewinns (oder der Rückerstattung).

2.1 Buchmacher, autorisierte Vertreter und Geschäfte (Lizenznehmer)

Buchmacher erleichtern Wettaktivitäten entweder online (Klasse B) oder durch **bevollmächtigte Vertreter** und **Einzelhandelsgeschäfte** (Klasse A). Im ersten Fall stellen sie Plattformen zur Verfügung, über die Spieler ein **Konto** registrieren und anschließend **Wettscheine** einreichen können. Im Falle eines landgestützten Betriebs registrieren die Spieler ihre Daten nicht über ein Konto, sondern reichen ihre Wetten über die Kassen ein.

Da alle drei dieser Einrichtungen gesetzlich geregelt sind, werden Details zu ihrer Struktur und ihrem Betriebsaufbau von einer Reihe von Systemen außerhalb des BMRS verfolgt. Während des Datenaustauschs durch die in Abschnitt 3.0 aufgeführten Methoden identifiziert das BMRS jeden Absender eindeutig, aber die NBA verlässt sich nicht auf diesen speziellen Kommunikationskanal, um die detaillierten Aufzeichnungen dieser Einrichtungen zu verwalten.

2.2 Spieler und Konten

Im Falle von Online-Wetten kann eine Person erst nach Registrierung eines **Kontos** eine Wette bei einem lizenzierten Buchmacher der Klasse B abschließen. Das Konto, das auf der Plattform des Buchmachers geführt wird, enthält bestimmte Identifikationsdaten seines Inhabers und führt eine Aufzeichnung der Gelder, die ihm für die Wetten zur Verfügung stehen. Die Buchmacher-Plattform ermöglicht eine Reihe von **Transaktionen**, die sich auf das Kontoguthaben auswirken, wie die Einzahlung von zusätzlichen Geldern durch den Kontoinhaber, die Auszahlung von Geldern und die Verfolgung von Boni, die der Buchmacher dem Spieler unter bestimmten Regeln und Bedingungen möglicherweise angeboten hat. Darüber hinaus erhält ein Spieler die Möglichkeit, seinen Wettaktivitäten durch **Beschränkungen** bestimmte Grenzen aufzuerlegen. In einigen Fällen kann der Buchmacher auch **Beschränkungen** für das Konto eines Spielers verhängen, wenn der Verdacht auf einen Regelverstoß besteht.

Bei landgestützten Wettanbietern benötigt der Spieler derzeit kein Konto, um an Wetten teilzunehmen, und der vorhergehende Absatz ist hier größtenteils nicht relevant. Innerhalb des BMRS-Modells stellt das Konto eines Spielers im Rahmen von Einzelhandelswetten den Kassierer des Geschäfts dar, über das ein Wettschein eingereicht wurde.

2.3 Der Wettschein

Die Wurzel des BMRS-Datenmodells zur Darstellung des Wettverkehrs ist der Wettschein (**Betslip**). Ein Wettschein kann im Falle einer Lizenz der Klasse A in einem Geschäft oder im Falle der Klasse B online ausgestellt werden. Ein Wettschein umfasst ein oder mehr **Wettscheinelemente**, die alle Informationen über ein bestimmtes zugrunde liegendes Ereignis, einen Markt, der mit diesem Ereignis verbunden ist, und das Ergebnis dieses Marktes (Auswahl), auf das der Spieler eine Wette gegen die vom Buchmacher angebotenen Quoten platzieren möchte, enthalten. Eine **Wette**, auf die der Spieler einen Einsatz macht, kann ein oder mehrere dieser Wettscheinelemente (Ereignisse, Märkte und Spielerauswahl), die zusammen eine Reihe von möglichen Kombinationen bilden, oder **Wettlinien** betreffen. Da jede dieser Kombinationen aus verschiedenen Bestandteilen besteht, unterscheiden sich die Quoten von einer Linie zur nächsten und somit könnte einer Wette eine Reihe von angebotenen Quoten zugewiesen werden.

Sobald ein Wettschein erstellt wurde, folgt ein Wettschein einem bestimmten Lebenszyklus, der endet, nachdem alle enthaltenen Wetten abgerechnet wurden, d. h. alle seine zugrunde liegenden Ereignisse sind vorbei, und alle realisierten Gewinne werden entweder ausgezahlt oder dem Konto des Spielers gutgeschrieben. Zwischen diesen Anfangs- und Endpunkten existieren mehrere Zwischenschritte, und diese Schritte können von einem Buchmachersystem zum anderen variieren. Das BMRS-Modell für den Wettscheinlebenszyklus ist in Abbildung 2.2 dargestellt. Jedes externe System, das Daten an das BMRS sendet, sollte angepasst werden, um diesem Zyklus so genau wie möglich zu folgen, indem es seinen eigenen Status eines Wettscheins den in der Abbildung dargestellten Status zuordnet, die in Tabelle 3.18 näher erläutert werden.

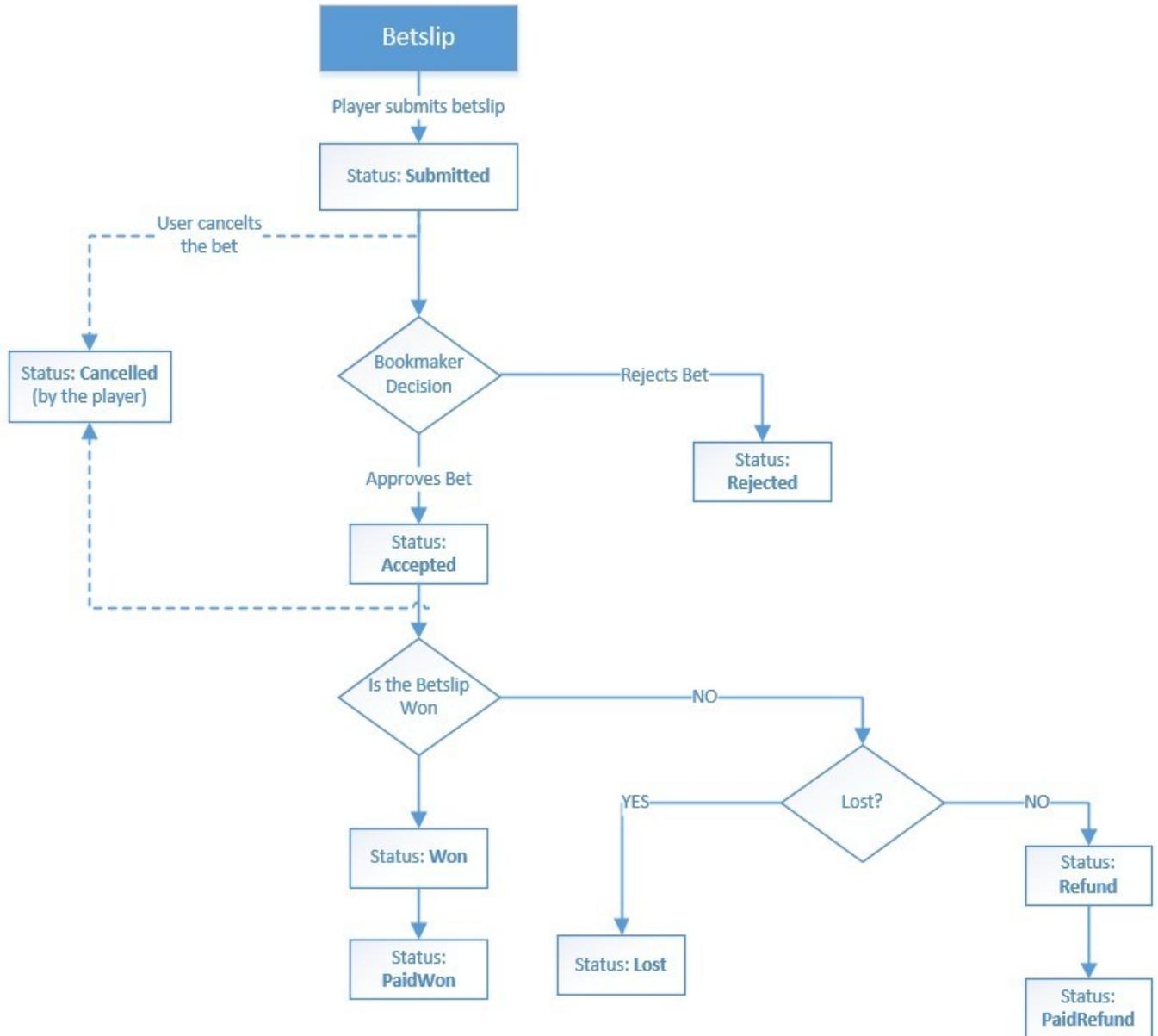


Abbildung 2.2: Der Wettscheinlebenszyklus im BMRS.

Betslip	Wettschein
Player submits betslip	Spieler reicht Wettschein ein
Status: Submitted	Status: Eingereicht
User cancels the bet	Benutzer storniert die Wette
Status: Cancelled (by the player)	Status: Storniert (durch den Spieler)
Bookmaker Decision	Buchmacher-Entscheidung
Rejects Bet	Lehnt Wette ab
Status: Rejected	Status: Abgelehnt
Approves Bet	Genehmigt die Wette
Status: Accepted	Status: Angenommen
Is the Betslip Won	Wettschein erzielt Gewinn

Status: Won	Status: Gewonnen
Status: PaidWon	Status: Gewinn ausgezahlt
YES	JA
NO	NEIN
Status: Lost	Status: Verloren
Lost?	Verloren?
Status: Refund	Status: Rückerstattung
Status: PaidRefund	Status: Rückerstattung ausgezahlt

3.0 Die API-Prozesse des BMRS

Jede Datenübertragung durch die beaufsichtigten Einrichtungen an die NBA über das BMRS erfolgt mittels einer Reihe von API-Methoden in Form von SOAP-Anfragen (die Kommunikationsprotokolle sind in Abschnitt 4.0 ausführlich beschrieben). In diesen Methoden sind die Datenmodelle kodiert, die das BMRS bei der Strukturierung der verschiedenen Objekte verwendet, die gemeinsam die Prozesse und Aktivitäten beschreiben, die es überwachen soll.

Diese Aktivitäten können auf der Grundlage der Objekte, auf die sie wirken, gruppiert werden:

- Spielerkonten
 - Erstellen und Aktualisieren von Konten.
 - Erstellen von Kontotransaktionen (Einzahlungen, Auszahlungen usw.).
 - Überwachen des Hinzufügens, Entfernens oder Ablaufens von Kontobeschränkungen.
- Wettaktivitäten
 - Wettscheine erstellen.
 - Aktualisieren von Wettscheinen während ihres gesamten Lebenszyklus.
- BMRS-bezogene Daten
 - Heartbeat-Aufruf.
 - Fehlerprotokoll.

Das BMRS erwartet, dass die Daten nicht mehr als **90 Sekunden**, nachdem sie im System des Buchmachers registriert wurden, eingehen. Die API-Methoden ermöglichen den Versand solcher Daten in Stapeln, was bedeutet, dass mehrere Objekte innerhalb eines einzigen Aufrufs erstellt oder aktualisiert werden können. Die Wahl des tatsächlichen Prozesses und des Zeitplans liegt jedoch beim Absender und wird nur durch die 90-Sekunden-Grenze eingeschränkt.

Die Datenstruktur einer BMRS-Methode hat die Form eines XML-Baums. Einige Elemente dieses Baums sind einfach Container, unter denen eine Sammlung von einem oder mehreren gleichartigen Unterelementen aufgelistet werden kann, während andere Unterelemente enthalten, die Daten darstellen.

Allen Methoden gemeinsam ist das Element `<Header>`, das grundlegende Informationen zur Identifizierung des Absenders der Daten gemäß Tabelle 3.1 enthält. Die Parameter unter `<Header>` werden von der NBA zur Verfügung gestellt, bevor eine Verbindung zwischen einem externen System zum BMRS initiiert wird. Spätere Änderungen dieser Kennungen werden von der Behörde rechtzeitig mitgeteilt.

Header				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Dateneingabeschlüssel	Zeichenfolge	Ein einzigartiger Schlüssel, der vom BMRS generiert und jedem Buchmacher von der NBA (GUID) zur Verfügung gestellt wird.	Dieser Wert kann bei Bedarf ersetzt (neu ausgegeben) werden.
2	Lizenznummer	Zeichenfolge	Die Lizenznummer des Buchmachers.	
3	Lizenznehmerkennung	Zeichenfolge	Eine einzigartige ID, die vom BMRS generiert und jedem Buchmacher von der NBA zur Verfügung gestellt wird.	Dieser Wert ist <u>statisch</u> . Einmal ausgegeben, kann er nicht geändert werden, da er den Buchmacher eindeutig identifiziert.

Tabelle 3.1: Datenfelder (Unterelemente) unter dem Header-Element.

3.1 Spielerkonten

Die meisten vom BMRS überwachten Prozesse stammen entweder von einem Spielerkonto (z. B. die Platzierung einer Wette) oder gelten einem Spielerkonto (z. B. eine Transaktion, die sich aus dem Gewinn einer Wette ergibt). Daher ist das Konto eines Spielers von zentraler Bedeutung für das vom System verwendete Datenmodell und muss jedes Mal referenziert werden, wenn eine das Konto betreffende Aktion aufgerufen wird. In Bezug auf den Datenfluss ist die Folge, dass die Erstellung eines Spielerkontos jeder Berichterstattung, die dieses Konto betrifft, vorausgehen muss.

Spielerkonten werden erstellt und können nach ihrer Erstellung mit der gleichen Methode, *Konten speichern*, aktualisiert werden.

Hinweis: Konten sind sowohl für Betreiber der Klasse A als auch für Betreiber der Klasse B erforderlich, aber die Informationen, die sie im ersten Fall enthalten, beziehen sich auf Ladenkassen, nicht auf Wettkunden.

3.1.1 Methode „Konten speichern“

3.1.1.1 Beschreibung

Spielerkonten, die im System des Buchmachers erstellt oder aktualisiert werden, müssen dem BMRS innerhalb eines 90-Sekunden-Zeitfensters nach der Methode *Konten speichern* mitgeteilt werden. Die Parameter, die durch diesen Aufruf weitergegeben werden, folgen der XML-Baumstruktur, die in Abbildung 3.1 dargestellt ist.

Der *Header* und sein Inhalt wurden bereits in Tabelle 3.1 erläutert.

Der Knoten *Konto speichern Anfrage* enthält Unterelemente, die Datenfelder im Zusammenhang mit dem Konto darstellen; diese Felder sind in Tabelle 3.2 aufgeführt. Ein Konto kann als aktiv oder deaktiviert bezeichnet werden, wie in Tabelle 3.3 beschrieben.

Hinweis: Das **Benutzername**-Datenfeld für ein Konto muss eine eindeutige Kennung enthalten. Diese wird in nachfolgenden Aufrufen verwendet, um das betreffende Konto zu referenzieren (z. B. *Wettscheine erstellen*, *Kontotransaktionenerstellen* etc.).

Bis auf den **Benutzernamen** (eine eindeutige Kennung) und den Parameter **Registriert am** können alle anderen Felder, die an die Methode *Konten speichern* weitergegeben werden, nach der Erstellung des Spielerkontos geändert werden. Wenn also zu irgendeinem Zeitpunkt ein Spieler einige dieser Informationen im System des Buchmachers ändert, sollten die Änderungen in dem nachfolgenden Aufruf an die BMRS-API widergespiegelt werden, indem der vollständige Satz seiner Kontodaten in einem Element *Konto speichern Anfrage* unter dem Knoten *Konten* aufgenommen wird.

3.1.1.2 Methode der Anfrage

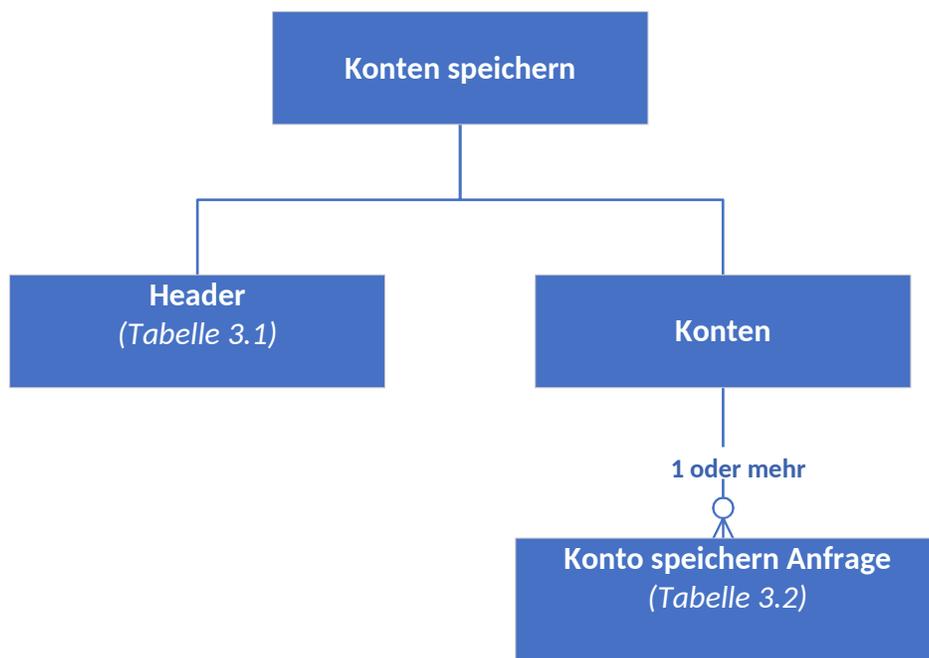


Abbildung 3.1: XML-Struktur der Methode „Konten speichern“.

Konto speichern Anfrage (SaveAccountRequest)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Benutzername	Zeichenfolge	Der Benutzername des Kontos. Die eindeutige Kennung des Kontos.	Dies muss nicht unbedingt ein tatsächlicher Benutzername sein, z. B. könnte es sich um eine Referenznummer handeln.
2	Registriert am	Datum Zeit	Das UTC-Datum und die Uhrzeit, zu der dieses Kontos im System des Buchmachers erstellt wurde.	Diese Einstellung wird beim Erstellen des Kontos festgelegt und kann nicht durch nachfolgende Aufrufe aktualisiert werden.
3	Vollständiger Name	Zeichenfolge	Name und Nachname des Kontoinhabers.	
4	Art des Ausweisdokuments	Zeichenfolge	I Die Art des Ausweisdokuments des Kontoinhabers (Personalausweis oder Reisepass)	
5	Identitätsnummer	Zeichenfolge	Die Identitätsnummer des Kontoinhabers (gemäß dem gewählten Ausweisdokument)	
6	Ausstellungsland des Ausweisdokuments	Zeichenfolge	Der Code des Landes, das das Ausweisdokument (Personalausweis oder Reisepass) ausgestellt hat.	ISO-Alpha-2 oder ISO-Alpha-3.
7	Geburtsdatum	Datum Zeit	Das Geburtsdatum des Kontoinhabers.	
8	Status	Aufzählung [Aktiv] [Deaktiviert]	Der aktuelle Kontostatus.	Weitere Einzelheiten finden Sie in Tabelle 3.3.
9	Verifiziert	Boolean	Ein Flag, das angibt, ob die Identität/Daten des Kontoinhabers vom Buchmacher überprüft wurden.	
10	Beschreibung	Zeichenfolge	Alle zusätzlichen Informationen.	

Tabelle 3.2: Datenfelder (Unterelemente) unter dem Element Konto speichern-Anfrage.

Status	Beschreibung
Aktiv	Das Konto ist für Transaktionen (Einzahlung, Auszahlung usw.) offen und/oder kann an Wettaktivitäten beteiligt werden.
Deaktiviert	Das Konto kann nicht für Transaktionen oder Wetten verwendet werden.

Tabelle 3.3: Konto speichern Anfrage – **Status-Feldwerte.**

Hinweis: Mehrere Knoten *Konto speichern Anfrage* können in einem einzigen Aufruf an die Methode *Konto speichern* enthalten sein, sowohl neue Konten als auch bestehende Konten, die aktualisiert wurden. Ein Konto sollte nur einmal in den Stapel aufgenommen werden; mehrfaches Auftreten desselben Benutzernamens innerhalb desselben Aufrufs würde zu einem Fehler führen.

Ein Beispiel für die SOAP/XML-Anfrage für diese Methode ist unten dargestellt:

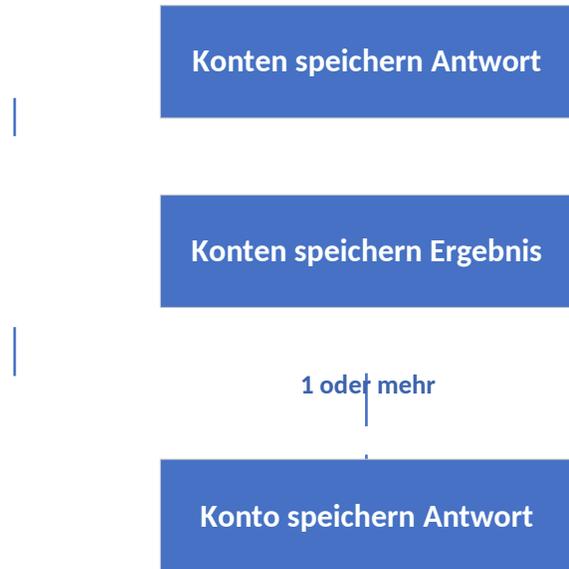
```

Konto speichern Beispielanfrage
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tem="http://tempuri.org/"
xmlns:int="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Request"
>
  <soapenv:Header/>
  <soapenv:Body>
    <tem:SaveAccounts>
      <tem:header>
        <int:DataEntryKey>F2D64583-DFC1-4DF1-A95D-02BCD1CBA489</int:DataEntryKey>
        <int:LicenseNumber>A-UT-0001</int:LicenseNumber>
        <int:LicenseeIdentifier>19666639-C7F8-4942-876D-CA1088B75EFE</int:LicenseeIdentifier>
      </tem:header>
      <tem:accounts>
        <!--One or more repetitions!-->
        <int:SaveAccountRequest>
          <int:DateOfBirth>1999-09-27T19:07:15.4993989+03:00</int:DateOfBirth>
          <int:Description>DESC_SPNTOSPIRG</int:Description>
          <int:FullName>First Last</int:FullName>
          <int:IsVerified>1</int:IsVerified>
          <int:IdentityDocumentType>NationalIdentityCard</int:IdentityDocumentType>
          <int:IdentityNumberIssuingCountry>CYP</int:IdentityNumberIssuingCountry>
          <int:IdentityNumber>00112233</int:IdentityNumber>
          <int:RegisteredOnDate>2018-09-27T19:07:15.5004083+03:00</int:RegisteredOnDate>
          <int>Status>Active</int>Status>
          <int:Username>UT-User_XBBQARPETW</int:Username>
        </int:SaveAccountRequest>
      </tem:accounts>
    </tem:SaveAccounts>
  </soapenv:Body>
</soapenv:Envelope>

```

3.1.1.3 Methode der Antwort

Nach dem Aufruf einer BMRS-Methode gibt das System dem Absender Feedback in Form einer XML-Antwort. Im Falle der Erstellung und/oder Aktualisierung von Spielerkonten enthält diese Antwort Feedback für jedes der Konten im ursprünglichen Aufruf, die unter dem Wurzelement *Konten speichern Ergebnis*, wie in Abbildung 3.2 dargestellt. Beachten Sie, dass jedes Konto durch seinen zugewiesenen Benutzernamen referenziert wird.



Element	Datentyp
Fehlercode	Integer
Fehlermeldung	Zeichenfolge
Erfolg	Boolean
Benutzername	Zeichenfolge

Abbildung 3.2: XML-Struktur von *Konten speichern Antwort*.

Die Arten möglicher Fehler und deren Beschreibung sind in Tabelle 3.4 aufgeführt.

Fehlercode	Fehlermeldung	Beschreibung
1500	„Kein Lizenznehmer gefunden für die Lizenznummer: ‚{0}‘ Lizenznehmerkennung: ‚{1}‘ Dateneingabeschlüssel: ‚{2}‘“	Informationen über den im Header angegebenen Lizenznehmer stimmen mit keinem Lizenznehmer im BMRS überein (z. B. Ungültige Lizenznummer, ungültiger Dateneingabeschlüssel etc.)
1504	„Feld ‚{0}‘: Wert ist null oder leer“	Ein obligatorischer Wert oder ein Element fehlt (z. B. der Header der Anfrage)
1505	„Feld ‚{0}‘: Wert ‚{1}‘ ist ungültig“	Der übergebene Wert ist ungültig (z. B. ein ungültiger Ländercode, undefinierte Aufzählungen, nicht erwartete negative Werte oder nicht erwartete Nullwerte)
1506	„Feld ‚{0}‘: Wert ‚{1}‘ ist doppelt“	Es gibt doppelte Einträge in derselben Anfrage (z. B. Wettscheine, Wettscheinelemente, Wettlinien, Konten, Kontotransaktionen, Beschränkungen)
1515	„Lizenznehmer {0} akzeptiert keine Konten. Lizenznehmertyp: {1}“	Es sind keine kontobezogenen Anfragen zulässig, es sei denn, der Lizenznehmer ist entweder Klasse A oder B.

Tabelle 3.4: Fehlerarten in Konten speichern Antwort.

Das Folgende ist ein Beispiel für die SOAP/XML-Antwort:

```
Konto speichern Beispielantwort:

Beispielantwort:

<s:Envelope xmlns:s="http">
<s:Body>
<SaveAccountsResponse xmlns="htt">
<SaveAccountsResult xmlns:a=" nse" xmlns:i="htt">
<a:SaveAccountResponse>
<a:ErrorCode>0</a:ErrorCode>
<a:ErrorMessage i:nil="true"/>
<a:Success>true</a:Success>
<a:Username>UT-User_XBBQARPETW</a:Username>
</a:SaveAccountResponse>
</SaveAccountsResult>
</SaveAccountsResponse>
</s:Body>
</s:Envelope>
```

3.2 Kontotransaktionen

3.2.1 Methode „Kontotransaktionen erstellen“

3.2.1.1 Beschreibung

Hinweis: Kontotransaktionen gelten nur für Lizenznehmer der Klasse B. Solche Daten werden von einem Buchmacher der Klasse A nicht erwartet.

Spielertransaktionen spiegeln jede Überweisung von Geld auf das oder von dem Konto eines Spielers, das Engagement von Geldern in Wetten und jede Umwandlung von Boni in echtes Geld wider. Die vollständige Liste der Transaktionstypen ist in Tabelle 3.5 unten aufgeführt. Bevor eine Transaktion jeglicher Art im BMRS eingeleitet wird, muss das betreffende Konto erstellt worden sein. Einmal gesendet, kann eine Kontotransaktion nicht aktualisiert werden.

Transaktionstyp	Beschreibung	Pflichtfelder
Einzahlung	Es wurde Geld auf das Konto eingezahlt.	Geldbetrag
Auszahlung	Es wurde Geld von dem Konto abgeboben.	Geldbetrag
BonusReal	Ein Bonusbetrag wurde in echtes Geld umgewandelt.	Bonusbetrag UND Geldbetrag
Wettgewinn	Ein Betrag (entweder echtes Geld oder Bonus oder beides) wurde dem Konto als Ergebnis von Wettaktivitäten gutgeschrieben.	Geldbetrag UND/ODER Bonusbetrag
Wetteinsatz	Ein Betrag (entweder echtes Geld oder Bonus oder beides) wurde bei einer Wettplatzierung eingesetzt.	Geldbetrag UND/ODER Bonusbetrag
Bonus hinzugefügt	Ein Bonusbetrag wurde dem Konto gutgeschrieben.	Bonusbetrag
Bonus entfernt	Ein Bonusbetrag wurde von dem Konto genommen.	Bonusbetrag
Rückerstattung	Ein Betrag (entweder echtes Geld oder Bonus oder beides) wurde auf das Konto zurückerstattet.	Geldbetrag UND/ODER Bonusbetrag

Tabelle 3.5: Kontotransaktionen erstellen — **Transaktionstyp**-Feldwerte.

3.2.1.2 Methode der Anfrage

Alle Transaktionen der in Tabelle 3.5 aufgeführten Arten, die auf der Buchmacherplattform stattgefunden haben, müssen der Überwachungsbehörde innerhalb eines 90-Sekunden-Fensters mitgeteilt werden. Die Methode *Kontotransaktionen erstellen* erleichtert die Übermittlung dieser Daten in Stapeln, wobei jeder Stapel mehrere solche Transaktionen (jeweils mit einer anderen *Transaktionsreferenznummer*) von mehreren Spielerkonten enthalten kann. Die entsprechende Datenstruktur ist in Abbildung 3.3 dargestellt. Die Elemente unter Transaktion sind in Tabelle 3.6 beschrieben.

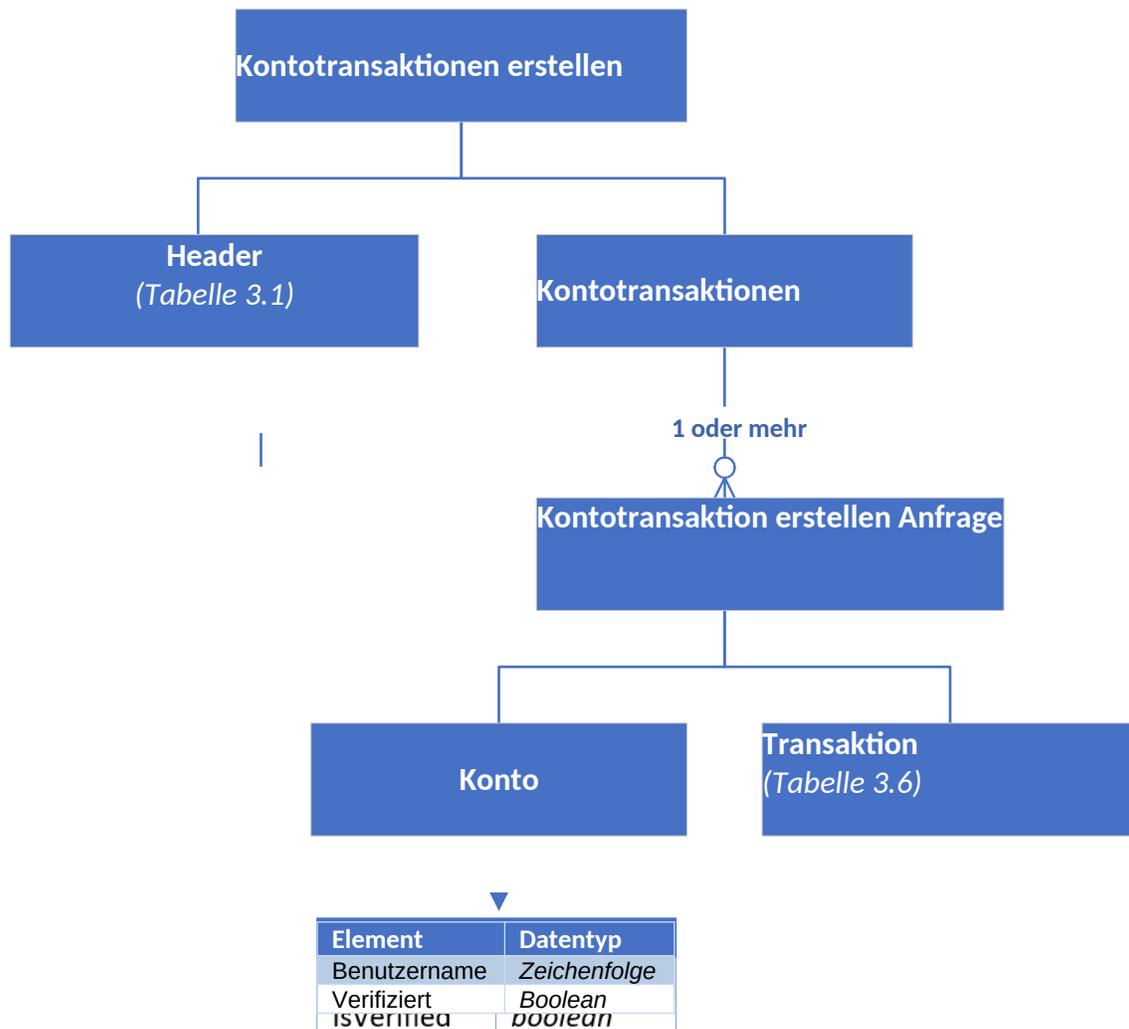


Abbildung 3.3: Die XML-Struktur der Methode *Kontotransaktionen erstellen*.

Transaktion (Transaction)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Transaktionsreferenznummer	Zeichenfolge	Eine eindeutige Kennung für diese Transaktion.	Diese wird vom Buchmachersystem zugewiesen.
2	Transaktionstyp	Aufzählung [Einzahlung] [Auszahlung] [BonusReal] [Wettgewinn] [Wetteinsatz] [Bonus hinzugefügt] m [Bonus entfernt] [Rückerstattung]	Die Art der Transaktion.	Siehe Einzelheiten in Tabelle 3.6.
3	Geldbetrag	Dezimal	Der Betrag an echtem Geld, der an dieser Transaktion beteiligt ist.	
4	Bonusbetrag	Dezimal	Der Bonusbetrag, der an dieser Transaktion beteiligt ist.	
5	Geldstand	Dezimal	Der resultierende Kontostand von echtem Geld <u>nach</u> dieser Transaktion.	Dies schließt jegliches echtes Geld aus, das derzeit eingesetzt ist (Feld 7).
6	Bonusstand	Dezimal	Der resultierende Bonus-Kontostand <u>nach</u> dieser Transaktion.	Dies schließt jeden Bonus aus, der derzeit eingesetzt ist (Feld 8).
7	Eingesetztes Geld	Dezimal	Der Gesamtbetrag an echtem Geld von diesem Konto, der <u>nach</u> dieser Transaktion in ausstehenden Wetten eingesetzt ist.	
8	Eingesetzter Bonus	Dezimal	Der Gesamtbetrag des Bonus von diesem Konto, der <u>nach</u> dieser Transaktion in ausstehenden Wetten eingesetzt ist.	
9	Erstellt am	Datum Zeit	Das UTC-Datum und die Uhrzeit, zu der diese Transaktion im System des Buchmachers erstellt wurde.	

Tabelle 3.6: Datenfelder (Unterelemente) unter dem Transaktionselement.

Ein Beispiel für die SOAP/XML-Anfrage für diese Methode ist unten dargestellt:

Kontotransaktionen erstellen Beispielanfrage
<pre> <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tem="http://tempuri.org/" xmlns:int="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Requ est"> <soapenv:Header/> <soapenv:Body> <tem:CreateAccountTransactions> <tem:header> <int:DataEntryKey>F2D64583-DFC1-4DF1-A95D-02BCD1CBA489</int:DataEntryKey> <int:LicenseNumber>A-UT-0001</int:LicenseNumber> <int:LicenseeIdentifier>19666639-C7F8-4942-876D-CA1088B75EFE</int:LicenseeIdentifier> </tem:header> <tem:accountTransactions> <!-- One or more repetitions:--> <int:CreateAccountTransactionRequest> <int:Account> <int:IsVerified>1</int:IsVerified> <int:Username>UT-User_QVNXIXOLGG</int:Username> </int:Account> <int:Transaction> <int:AmountBonus>77</int:AmountBonus> <int:AmountMoney>2319</int:AmountMoney> <int:BalanceBonus>263</int:BalanceBonus> <int:BalanceMoney>4083</int:BalanceMoney> <int:CreatedOnDate>2019-09-27T19:11:54.3972355+03:00</int:CreatedOnDate> <int:TransactionReferenceNumber>TR-UT_KGLDDUNFMR_122312313</int:TransactionReferenceNumber> <int:TransactionType>BetWinnings</int:TransactionType> <int:WageredBonus>60</int:WageredBonus> <int:WageredMoney>53</int:WageredMoney> </int:Transaction> </int:CreateAccountTransactionRequest> </tem:accountTransactions> </tem:CreateAccountTransactions> </soapenv:Body> </soapenv:Envelope> </pre>

3.2.1.3 Methode der Antwort

Das BMRS reagiert auf einen Aufruf *Kontotransaktionen erstellen* mit der XML-Struktur in Abbildung 3.4. Die Antwort gibt Feedback zu jeder einzelnen Transaktion, die über die API empfangen wird, über das Feld *Transaktionsreferenznummer*. Die Fehlerarten, die zurückgegeben werden könnten, sind in Tabelle 3.7 beschrieben.

Kontotransaktionen erstellen Antwort

Kontotransaktionen erstellen Ergebnis

1 oder mehr

Kontotransaktion erstellen Antwort

Element	DataType
ErrorCode	<i>Integer</i>
ErrorMessage	<i>String</i>
Success	<i>Boolean</i>
TransactionReferenceNumber	<i>String</i>
Username	<i>String</i>

Abbildung 3.4: Die XML-Struktur von Kontotransaktionen erstellen Antwort.

Fehlercode	Fehlermeldung	Beschreibung
1500	„Kein Lizenznehmer gefunden für die Lizenznummer: ‚{0}‘ Lizenznehmerkennung: ‚{1}‘ Dateneingabeschlüssel: ‚{2}‘“	Informationen über den im Header angegebenen Lizenznehmer stimmen mit keinem Lizenznehmer im BMRS überein (z. B. Ungültige Lizenznummer, ungültiger Dateneingabeschlüssel etc.)
1504	„Feld ‚{0}‘: Wert ist null oder leer“	Ein obligatorischer Wert oder ein Element fehlt (z. B. der Header der Anfrage)
1505	„Feld ‚{0}‘: Wert ‚{1}‘ ist ungültig“	Der übergebene Wert ist ungültig (z. B. ein ungültiger Ländercode, undefinierte Aufzählungen, nicht erwartete negative Werte oder nicht erwartete Nullwerte)
1506	„Feld ‚{0}‘: Wert ‚{1}‘ ist doppelt“	Es gibt doppelte Einträge in derselben Anfrage (z. B. Wertscheine, Wertscheinelemente, Wettlinien, Konten, Kontotransaktionen, Beschränkungen)
1515	„Lizenznehmer {0} akzeptiert keine Konten. Lizenznehmertyp: {1}“	Es sind keine kontobezogenen Anfragen zulässig, es sei denn, der Lizenznehmer ist entweder Klasse A oder B.

Tabelle 3.7: Fehlertypen in Kontotransaktionen erstellen Antwort.

Das Folgende ist ein Beispiel für die SOAP/XML-Antwort, die mit dieser Methode zurückgegeben wird:

Kontotransaktionen erstellen Beispielantwort
<pre> <s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"> <s:Body> <CreateAccountTransactionsResponse xmlns="http://tempuri.org/"> <CreateAccountTransactionsResult xmlns:a="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Respo nse" xmlns:i="http://www.w3.org/2001/XMLSchema-instance"> <a:CreateAccountTransactionResponse> <a:ErrorCode>0</a:ErrorCode> <a:ErrorMessage i:nil="true"/> <a:Success>true</a:Success> <a:TransactionReferenceNumber>TR-UT_KGLDDUNFMR_122312313</a:TransactionReferenceNumber> <a:Username>UT-User_QVNXIXOLGG</a:Username> </a:CreateAccountTransactionResponse> </CreateAccountTransactionsResult> </CreateAccountTransactionsResponse> </s:Body> </s:Envelope> </pre>

3.3 Kontobeschränkungen

Hinweis: Kontobeschränkungen gelten nur für Lizenznehmer der Klasse B. Solche Daten werden von einem Buchmacher der Klasse A nicht erwartet.

Kontobeschränkungen setzen Limits für bestimmte Transaktionen oder Wettaktivitäten, die vom Konto eines Spielers erlaubt sind, und diese Limits können entweder selbst auferlegt oder vom Buchmacher verhängt werden. Die Aktivierung, Deaktivierung und, falls es eine Zeitdimension gibt, der Ablauf einer Beschränkung sind die drei Ereignisse, die vom BMRS überwacht werden, und wenn eines von ihnen zu einem beliebigen Zeitpunkt auftritt, sollte innerhalb der nächsten 90 Sekunden ein Aufruf an die Funktion *Kontobeschränkungen speichern* ausgelöst werden. Die Arten der verfügbaren Beschränkungen innerhalb des BMRS sind in Tabelle 3.8 beschrieben.

Beschränkungsart	Beschreibung
Kurzfristiger Selbstausschluss	Der Spieler hat sich selbst für einen bestimmten Zeitraum von der Platzierung von Wetten ausgeschlossen.
Langfristiger Selbstausschluss	Der Spieler hat sich selbst auf unbestimmte Zeit von der Platzierung von Wetten ausgeschlossen.
Wettbetragsbeschränkung	Der Gesamtbetrag des Geldes, der für Wettaktivitäten von diesem Konto zulässig ist, wird für einen bestimmten Zeitraum beschränkt.
Wettverlustbeschränkung	Der Gesamtbetrag, den das Konto innerhalb eines bestimmten Zeitraums verlieren kann, wird beschränkt.
Einzahlungsbeschränkung	Der Geldbetrag, der über einen bestimmten Zeitraum auf dieses Konto eingezahlt werden kann, wird beschränkt.
Sonstiges	Jede andere Beschränkungsart, die nicht unter die vorstehenden Bestimmungen fällt. Einzelheiten können über das Feld „Beschreibung“ angegeben werden (siehe Tabelle 3.9).

Tabelle 3.8: *Kontobeschränkungen speichern* — **Beschränkungsart**-Feldwerte.

3.3.1 Methode „Kontobeschränkungen speichern“

3.3.1.1 Beschreibung

Die XML-Struktur, die für diese Methode verwendet wird, ist in Abbildung 3.5 dargestellt. Die Datenelemente unter dem Knoten *Beschränkungsträger* des Baumes sind in Tabelle 3.9 einzeln beschrieben. Wie bei allen BMRS-Methoden, kann ein einziger Aufruf *Kontobeschränkungen speichern* Datenstapel umfassen, d. h. in diesem Fall mehrere beschränkungsbezogene Ereignisse für ein oder mehrere Konten.

3.3.1.2 Methode der Anfrage

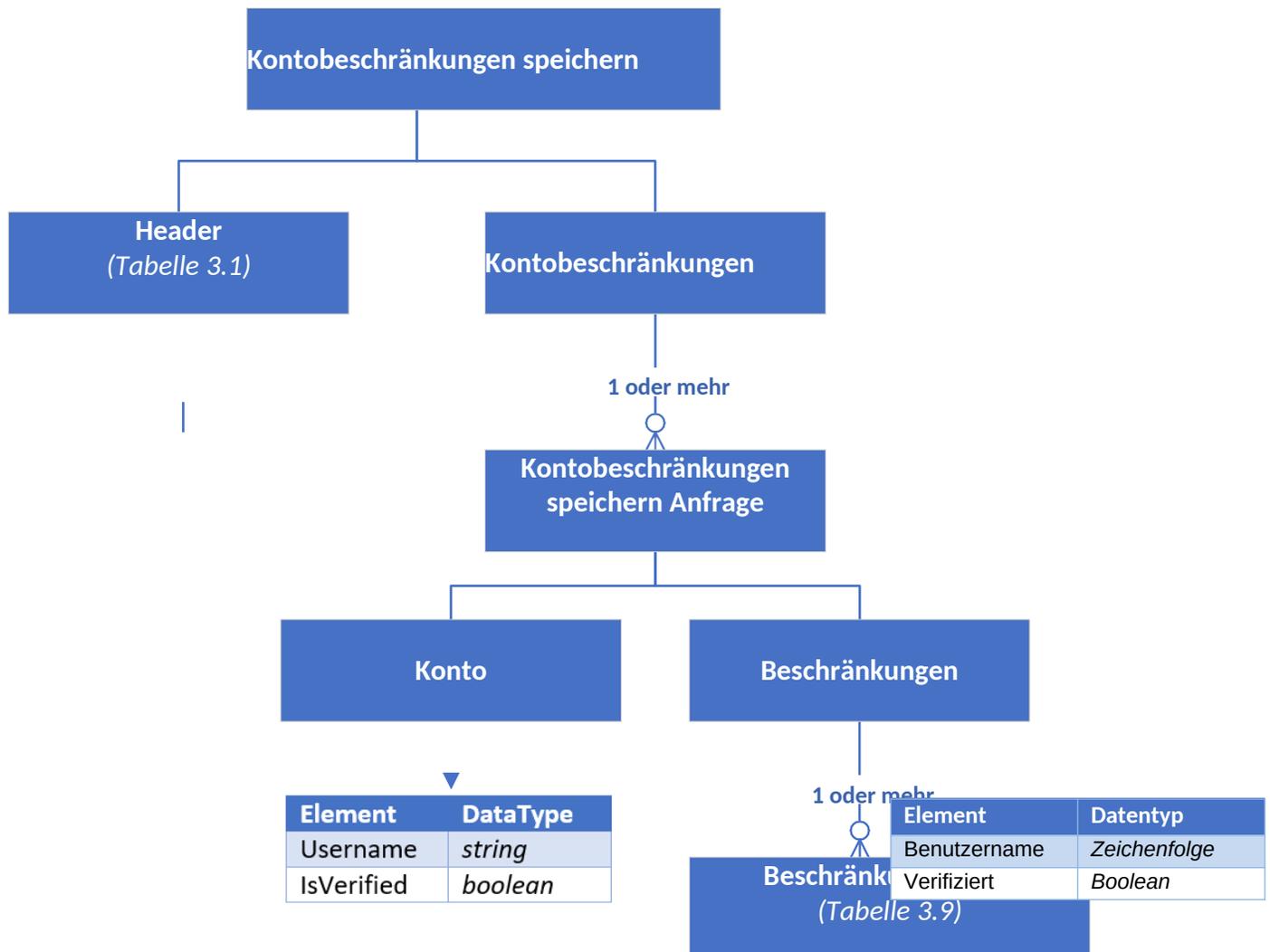


Abbildung 3.5: Die XML-Struktur der Methode *Kontobeschränkungen speichern*.

Beschränkungsträger (RestrictionCarrier)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Ereignisname	Aufzählung [Aktiviert] [Deaktiviert] \[Abgelaufen]	Das Ereignis, durch das diese Beschränkung ausgelöst wurde.	
2	Beschränkungsart	Aufzählung [Kurzfristiger Selbstausschluss] [Langfristiger Selbstausschluss] [Wettbetragsbeschränkung] [Wettverlustbeschränkung] [Einzahlungsbeschränkung] [Sonstige]	Die Art der Beschränkung.	Siehe Einzelheiten in Tabelle 3.8.
3	Erstellt von	Aufzählung [Kunde] [Buchmacher]	Gibt an, ob diese Beschränkung vom Spieler oder vom Buchmacher auferlegt wurde.	
4	Erstellt am	Datum Zeit	UTC-Datum und Uhrzeit, zu der das auslösende Ereignis (Feld 1) aufgetreten ist.	
5	Begründung	Zeichenfolge	Der Grund für diese Beschränkung.	Vor allem für Fälle, in denen die Beschränkung vom Buchmacher auferlegt wird.
6	Beschreibung	Zeichenfolge	Alle zusätzlichen Informationen.	Wenn beispielsweise die Art der Beschränkung „Sonstige“ ist, kann dieses Feld eine kurze Beschreibung dessen enthalten, was durch diese Beschränkung begrenzt ist.

Tabelle 3.9: Datenfelder (Unterelemente) unter dem Element Beschränkungsträger.

Das Folgende ist ein Beispiel für die SOAP/XML-Anfrage dieser Methode:

Kontobeschränkungen speichern Beispielanfrage

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tem="http://tempuri.org/"
xmlns:int="http://schemas.datacontract.org/2004/07/InteliScope.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Req
uest">
  <soapenv:Header/>
  <soapenv:Body>
    <tem:SaveAccountRestrictions>
      <tem:header>
        <int:DataEntryKey>F2D64583-DFC1-4DF1-A95D-02BCD1CBA489</int:DataEntryKey>
        <int:LicenseNumber>A-JT-0001</int:LicenseNumber>
        <int:LicenseeIdentifier>19666639-C7F8-4942-876D-CA1088B75EFE</int:LicenseeIdentifier>
      </tem:header>
      <tem:accountRestrictions>
        <!--One or more repetitions:-->
        <int:SaveAccountRestrictionRequest>
          <int:Account>
            <int:IsVerified>true</int:IsVerified>
            <int:Username>UT-User_CFMHOFYMVN</int:Username>
          </int:Account>
          <int:Restrictions>
            <!--One or more repetitions:-->
            <int:RestrictionCarrier>
              <int:CreatedBy>Bookmaker</int:CreatedBy>
              <int:CreatedOnDate>2019-09-27T18:59:56.3519795+03:00</int:CreatedOnDate>
              <int:Description>Description</int:Description>
              <int:EventName>Activated</int:EventName>
              <int:Rationale>Rationale</int:Rationale>
              <int:RestrictionType>BettingLossLimit</int:RestrictionType>
            </int:RestrictionCarrier>
          </int:Restrictions>
        </int:SaveAccountRestrictionRequest>
      </tem:accountRestrictions>
    </tem:SaveAccountRestrictions>
  </soapenv:Body>
</soapenv:Envelope>
```

3.3.1.3 Methode der Antwort

Das BMRS reagiert auf einen Aufruf für *Kontobeschränkungen speichern* mit der XML-Struktur in Abbildung 3.6. Die Antwort gibt Feedback zu jedem Beschränkungsereignis, das über die API empfangen wird, und diese Beschränkungen werden vom Benutzernamen des betroffenen Kontos lose referenziert. Die Fehlerarten, die zurückgegeben werden könnten, sind in Tabelle 3.10 beschrieben.

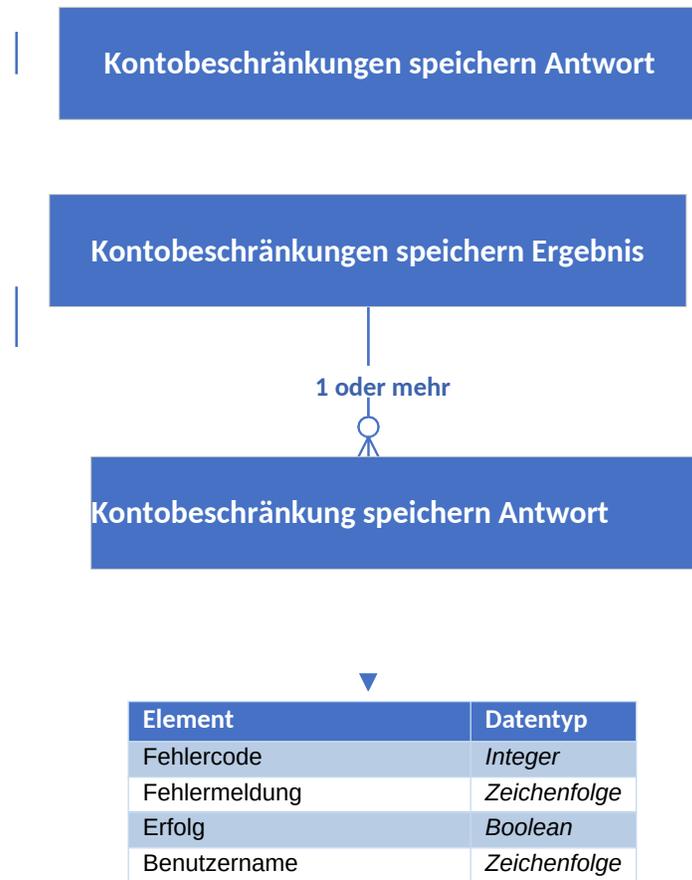


Abbildung 3.6: Die XML-Struktur von *Kontobeschränkungen speichern Antwort*.

Fehlercode	Fehlermeldung	Beschreibung
1500	„Kein Lizenznehmer gefunden für die Lizenznummer: ‚{0}‘ Lizenznehmerkennung: ‚{1}‘ Dateneingabeschlüssel: ‚{2}‘“	Informationen über den im Header angegebenen Lizenznehmer stimmen mit keinem Lizenznehmer im BMRS überein (z. B. Ungültige Lizenznummer, ungültiger Dateneingabeschlüssel etc.)
1504	„Feld ‚{0}‘: Wert ist null oder leer“	Ein obligatorischer Wert oder ein Element fehlt (z. B. der Header der Anfrage)
1505	„Feld ‚{0}‘: Wert ‚{1}‘ ist ungültig“	Der übergebene Wert ist ungültig (z. B. ein ungültiger Ländercode, undefinierte Aufzählungen, nicht erwartete negative Werte oder nicht erwartete Nullwerte)
1506	„Feld ‚{0}‘: Wert ‚{1}‘ ist doppelt“	Es gibt doppelte Einträge in derselben Anfrage (z. B. Wettscheine, Wettscheinelemente, Wettlinien, Konten, Kontotransaktionen, Beschränkungen)
1515	„Lizenznehmer {0} akzeptiert keine Konten. Lizenznehmertyp: {1}“	Es sind keine kontobezogenen Anfragen zulässig, es sei denn, der Lizenznehmer ist entweder Klasse A oder B.

Tabelle 3.10: Fehlerarten in Kontobeschränkungen speichern Antwort.

Das Folgende ist ein Beispiel für eine SOAP/XML-Antwort für diese Methode:

Kontobeschränkungen speichern Beispielantwort:
<pre> <s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"> <s:Body> <SaveAccountRestrictionsResponse xmlns="http://tempuri.org/"> <SaveAccountRestrictionsResult xmlns:a="http://schemas.datacontract.org/2004/07/IntelIScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Respo nse" xmlns:i="http://www.w3.org/2001/XMLSchema-instance"> <a:SaveAccountRestrictionResponse> <a:ErrorCode>0</a:ErrorCode> <a:ErrorMessage i:nil="true"/> <a:Success>true</a:Success> <a:Username>UT-User_CFMHOFYMVN</a:Username> </a:SaveAccountRestrictionResponse> </SaveAccountRestrictionsResult> </SaveAccountRestrictionsResponse> </s:Body> </s:Envelope> </pre>

3.4 Wettaktivitäten

Wettaktivitäten im Rahmen des BMRS umfassen die Platzierung von Wetten durch Spielerkonten, eine detaillierte Beschreibung dieser Wetten und die Verfolgung ihres Lebenszyklus durch ihren Status. Zentral für das Datenmodell, das für den Empfang von Wettdaten verwendet wird, ist das *Wettschein*-Objekt, das eine Vielzahl von Informationen enthält, einschließlich mehrerer Ergebnisse, auf die ein Spieler wetten möchte, und der tatsächlichen *Wetten* zu diesen Ergebnissen. Eine Übersicht über die Beziehung zwischen diesen Objekten finden Sie in Abschnitt 2.2.

3.4.1 Methode „Wettscheine erstellen“

3.4.1.1 Beschreibung

Neue Wettscheine, die von einem oder mehreren Konten erstellt werden, können dem BMRS mit der Methode *Wettscheine erstellen* in Stapeln übermittelt werden. Die entsprechende XML-Struktur ist in Abbildung 3.7 dargestellt. Die verschiedenen Datenelemente und die Werte, die sie annehmen können, sind in den Tabellen 3.11 bis 3.19 beschrieben.

3.4.1.2 Methode der Anfrage

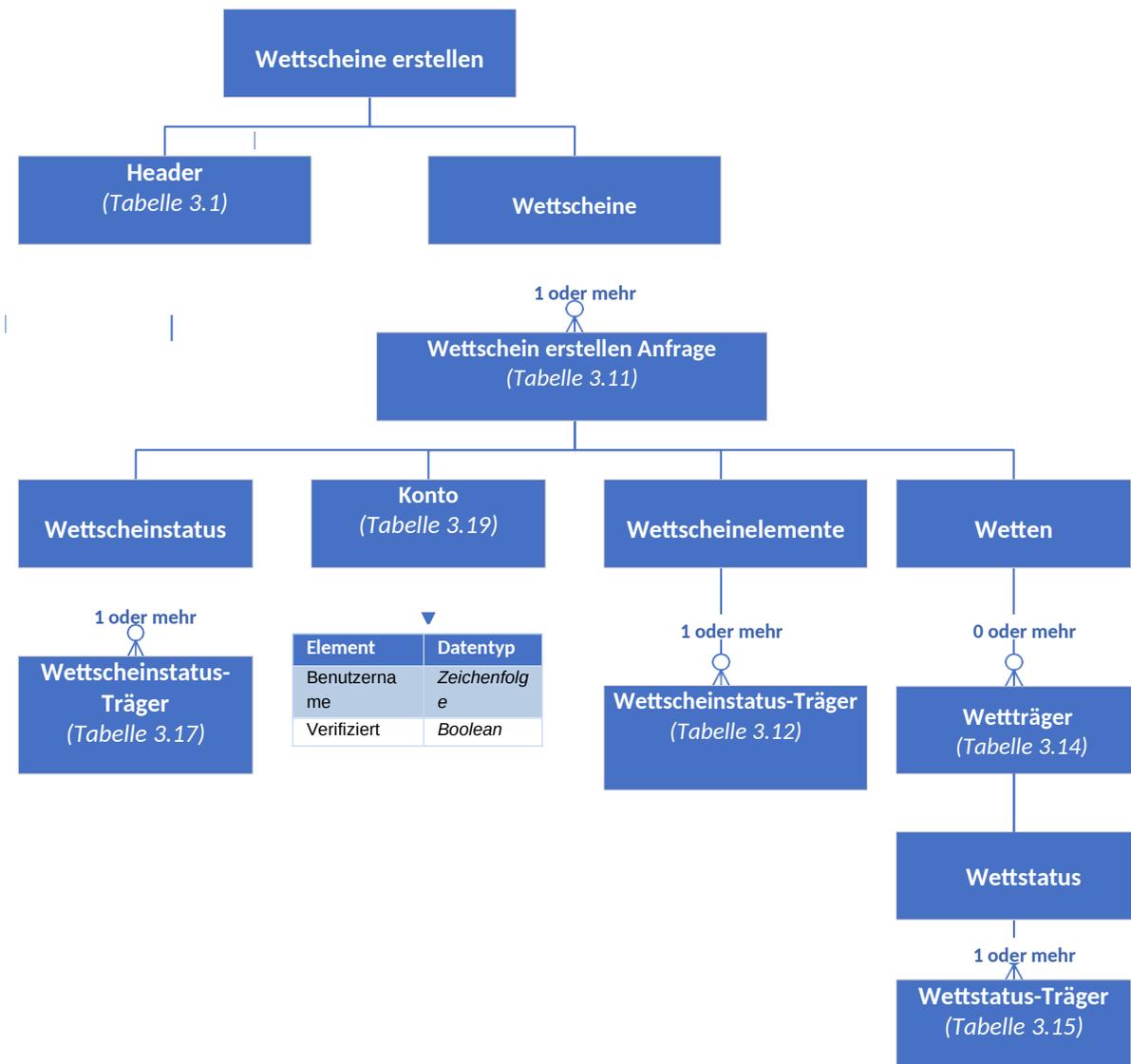


Abbildung 3.7: Die XML-Struktur der Methode *Wettscheine erstellen*.

Wettschein erstellen Anfrage (CreateBetSlipRequest)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Referenznummer	Zeichenfolge	Eindeutige Kennung, die vom Buchmachersystem zugewiesen wird. Keine zwei Wettscheine können den gleichen Wert haben.	Eindeutige Kennung. Wird verwendet, um einen Wettschein in folgenden Aktualisierungsanfragen zu referenzieren.
2	Lizenznummer des Emittenten	Zeichenfolge	Die Lizenznummer des Emittenten.	<u>Klasse A:</u> Geschäftslizenz <u>Klasse B:</u> Buchmacherlizenz
3	Ursprünglicher Einsatz	Dezimal	Der gesamte ursprüngliche Einsatz auf diesem Wettschein.	<u>Inklusive</u> Bonus.
4	Ursprünglicher Einsatz mit Bonus	Dezimal	Jeder Betrag des Bonus auf den ursprünglichen Einsatz.	
5	Auszahlung	Dezimal	Die gesamte endgültige Auszahlung dieses Wettscheins nach Abrechnung aller Wetten.	<u>Inklusive</u> Bonus.
6	Bonusauszahlung	Dezimal	Jeder Bonusbetrag in der gesamten endgültigen Auszahlung dieses Wettscheins.	
7	Max. Auszahlung	Dezimal	Die maximale mögliche Gesamtauszahlung.	
8	Min. Auszahlung	Dezimal	Die minimale mögliche Gesamtauszahlung.	
9	Beschreibung	Zeichenfolge	Alle zusätzlichen Informationen.	
10	Provision	Dezimal	Absoluter Betrag der an den Bevollmächtigten gezahlten Provision.	
11	TERMINAL-ID	Zeichenfolge	Kennung des Terminals, mit dem dieser Wettschein erstellt wurde.	Nur Klasse A.
12	Gesamtzahl der Kombinationen	Integer	Gesamtzahl der Kombinationen, die sich aus den zugrunde liegenden Wetten ergeben.	
13	Erstellt am	Datum Zeit	UTC-Datum und Uhrzeit der Erstellung des Wettscheins im Buchmachersystem.	

Tabelle 3.11: Datenfelder (Unterelemente) unter dem Element Wettschein erstellen Anfrage.

Wettscheinstatus-Träger (BetSlipItemCarrier)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Elementreferenznummer	Zeichenfolge	Referenznummer, die diesem Wettscheinelement vom System des Buchmachers zugewiesen wird.	Eindeutige Kennung nur in diesem Wettschein.
2	Wetttyp	Aufzählung [Direkt] [Vorspiel] [Live] [Sonstige]	Der Wetttyp dieses Wettscheinelements.	
3	Sport	Zeichenfolge	Der Name der Sportart der zugrunde liegenden Veranstaltung.	
4	Wettbewerb	Zeichenfolge	Name des Wettbewerbs, in dem die zugrunde liegende Veranstaltung stattfindet.	
5	Region	Zeichenfolge	Die geographische Region, in der der Wettbewerb stattfindet.	
6	Ereignisname	Zeichenfolge	Der Name des zugrunde liegenden Ereignisses.	
7	Datum des Veranstaltungsbeginns	Datum Zeit	Das UTC-Datum und die Uhrzeit, zu der das zugrunde liegende Ereignis beginnt.	
8	Ereignis Ergebnis	Zeichenfolge	Das Endergebnis des zugrunde liegenden Ereignisses.	
9	Veranstaltungsschlüssel	Zeichenfolge	Die eindeutige Kennung, die dem zugrunde liegenden Ereignis vom System des Buchmachers zugewiesen wird.	
10	Wettbewerber A	Zeichenfolge	Der Name des Wettbewerbers A.	Falls anwendbar.
11	Wettbewerber B	Zeichenfolge	Der Name des Wettbewerbers B.	Falls anwendbar.
12	Markttyp	Zeichenfolge	Der Name des Markttyps.	Beispiele: „Endstand“ „Unter/Über 2,5“
13	Markttypschlüssel	Zeichenfolge	Die eindeutige Kennung, die diesem Markttyp vom Buchmachersystem zugewiesen wird.	
14	Markttypschlüssel BMRS	Zeichenfolge	Die eindeutige Kennung, die diesem Markttyp durch das BMRS zugewiesen wird.	Bereitgestellt von der NBA.
15	Auswahl Name	Zeichenfolge	Der Name der Auswahl des Spielers.	Beispiele: „Heimsieg“, „Über“
16	Auswahlschlüssel BMRS	Zeichenfolge	Die eindeutige Kennung, die der Auswahl des Spielers durch das BMRS zugewiesen wird.	Bereitgestellt von der NBA.

17	Quoten	Dezimal	Die vom Buchmacher angebotenen Quoten für die Auswahl des Spielers.	
18	Beschreibung	Zeichenfolge	Alle zusätzlichen Informationen.	
19	Status	Aufzählung: [Ausstehend] [Gewonnen] [Verloren] [Ungültig]	Der Status der Auswahl des Spielers für den Markt dieses Elements.	Siehe Tabelle 3.13.

Tabelle 3.12: Datenfelder (Unterelemente) unter dem Element Wettscheinelement-Träger.

Status	Beschreibung
Ausstehend	Das Ergebnis des Marktes ist noch nicht entschieden.
Gewonnen	Das Ergebnis des Marktes ist das gleiche wie die Auswahl des Spielers.
Verloren	Das Ergebnis des Marktes unterscheidet sich von der Auswahl des Spielers.
Ungültig	Das Ergebnis des Marktes kann nicht ermittelt werden (z. B. Ereignis storniert).

Tabelle 3.13: Wettscheinelement-Träger – Status-Feldwerte.

Wettträger (BetCarrier)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Wettreferenznummer	Zeichenfolge	Referenznummer dieser Wette. Vom Buchmachersystem zugewiesen.	Eindeutige Kennung nur innerhalb dieses Wettscheins.
2	Wettscheinelemente-Referenznummern	Zeichenfolge	Durch Komma getrennte Zeichenfolge der Referenznummer (Tabelle 3.12 Feld 1) jedes Wettscheinelements, das in dieser Wette enthalten ist.	
3	Beschreibung	Zeichenfolge	Alle zusätzlichen Informationen.	
4	Anzahl der Kombinationen	Integer	Die Gesamtzahl der Kombinationen, die aus dieser Wette abgeleitet werden.	

5	Mindestquoten	Dezimal	Mindestquoten für diese Wette.	
6	Max. Quoten	Dezimal	Maximale Quoten für diese Wette.	
7	Ursprünglicher Einsatz	Dezimal	Der gesamte ursprüngliche Einsatz auf diese Wette. Inklusive Bonus.	
8	Auszahlung	Dezimal	Die gesamte endgültige Auszahlung dieser Wette. Inklusive Bonus.	

Tabelle 3.14: Datenfelder (Unterelemente) unter dem Element Wettträger.

Wettstatus-Träger (BetStatusCarrier)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Status	Aufzählung [Ausstehend] [Gewonnen] [Verloren] [Ungültig]	Der aktuelle Status dieser Wette.	Siehe Tabelle 3.16.
2	Erstellt am	Datum Zeit	UTC-Datum und Uhrzeit, zu der dieser Status im Buchmachersystem erreicht wurde.	

Tabelle 3.15: Datenfelder (Unterelemente) unter dem Element Wettstatus-Träger.

Status	Beschreibung
Ausstehend	Die Wette steht noch aus.
Gewonnen	Mindestens eine der zugrunde liegenden Kombinationen der Wette wurde realisiert.
Verloren	Keine der zugrunde liegenden Kombinationen der Wette wurde realisiert.
Ungültig	Die Wette ist nicht mehr gültig.

Tabelle 3.16: Wettstatus-Träger – Status-Feldwerte.

Wettscheinstatus-Träger (BetSlipStatusCarrier)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Status	Aufzählung [Eingereicht] [Angenommen] [Storniert] [Abgelehnt] [Gewonnen] [Verloren] [Rückerstattung] [Gewinn ausgezahlt] [Rückerstattung ausgezahlt]	Der aktuelle Status dieses Wettscheins.	Siehe Abbildung 2.2 und Tabelle 3.18 für den Wettscheinlebenszyklus.
2	Aktuelle Auszahlung	Dezimal	Gesamtauszahlung bereits durch diesen Wettschein gesichert. Inklusive Bonus.	
3	Abgewickelter Einsatz	Dezimal	Jeder Betrag des Einsatzes, der abgewickelt wurde. Inklusive Bonus.	
4	Erstellt am	Datum Zeit	Das UTC-Datum und die Uhrzeit, zu der diese Änderung in „Status“ oder in „Aktuelle Auszahlung“ oder „Abgewickelter Einsatz“ im System des Buchmachers aufgezeichnet wurde.	

Tabelle 3.17: Datenfelder (Unterelemente) unter dem Element Wettscheinstatus-Träger.

Status	Beschreibung
Eingereicht	Der Wettschein wurde im Buchmachersystem erstellt.
Angenommen	Der Buchmacher hat den Wettschein angenommen.
Storniert.	Der Spieler hat den Wettschein storniert.
Abgelehnt	Der Buchmacher hat den Wettschein abgelehnt.
Gewonnen	Eine oder mehrere Wetten auf dem Wettschein haben gewonnen. Der Betrag wurde noch nicht an den Spieler ausgezahlt.
Verloren	Keine der Wetten auf dem Wettschein hat gewonnen.
Rückerstattung	Der Einsatz wird dem Spieler zurückerstattet.
Gewinn ausgezahlt	Die Gewinne wurden an den Spieler ausgezahlt.
Rückerstattung ausgezahlt	Die Rückerstattung wurde an den Spieler ausgezahlt.

Tabelle 3.18: Wettscheinstatus-Träger – **Status-Feldwerte.**

Hinweis: Jeder Wettschein, der durch die Weitergabe von Daten an die Methode *Wettscheine erstellen* erstellt wird, kann mehrere damit verbundene *Wettscheinstatus-Träger* und/oder *Wettstatus-Träger* haben (Abbildung 3.7). Diese Funktion ermöglicht die Einbeziehung von mehr als einer Änderung in den zugrunde liegenden Parametern dieser Knoten, innerhalb desselben Aufrufs. Zum Beispiel, wenn ein neu erstellter Wettschein innerhalb desselben 90-Sekunden-Fensters eingereicht und angenommen wurde, kann der Absender beide Änderungen in einem einzigen Aufruf an diese Methode übermitteln, indem er zwei verschiedene und angemessen zeitgestempelte *Wettscheinstatus-Träger* in seine XML-Struktur aufnimmt.

Konto (Account)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Benutzername	Zeichenfolge	Benutzername (UID) des Kontos, das diese Wette platziert hat.	Für Klasse A wird dies ein Kassierer sein.
2	Verifiziert	Boolean	Ein Flag, das angibt, ob das Konto, das diesen Wettschein platziert hat, ein verifiziertes Konto ist oder nicht.	

Tabelle 3.19: Datenfelder (Unterelemente) unter dem Element „Konto“.

Im Folgenden finden Sie ein Beispiel für die SOAP/XML-Anfragestruktur für diese Methode:

```

Wettscheine erstellen Beispielanfrage:
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tem="http://tempuri.org"
xmlns:int="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Requ
est">
  <soapenv:Header/>
  <soapenv:Body>
    <tem:CreateBetSlips>
      <tem:header>
        <int:DataEntryKey>F2D64583-DFC1-4DF1-A95D-02BCD1CBA489</int:DataEntryKey>
        <int:LicenseNumber>A-UT-0001</int:LicenseNumber>
        <int:LicenseeIdentifier>19666639-C7F8-4942-876D-CA1088B75EFE</int:LicenseeIdentifier>
      </tem:header>
      <tem:betSlips>
        <!--One or more repetitions:-->
        <int:CreateBetSlipRequest>

          <int:BetSlipStatus>
            <!--One or more repetitions:-->
            <BetSlipStatusCarrier>
              <int:CreatedOnDate>2019-09-27T19:19:13.6183845+03:00</int:CreatedOnDate>
              <int:CurrentPayout>412</int:CurrentPayout>
              <int:SettledStake>26</int:SettledStake>
              <int:Status>Submitted</int:Status>
            </BetSlipStatusCarrier>
          </int:BetSlipStatus>
        </tem:betSlips>
      </tem:CreateBetSlips>
    </soapenv:Body>
  </soapenv:Envelope>

```

```

<int:CreatedOnDate>2019-09-27T19:20:13.6183845+03:00</int:CreatedOnDate>
<int:CurrentPayout>412</int:CurrentPayout>
<int:SettledStake>26</int:SettledStake>
<int:Status>Accepted</int:Status>
</BetSlipStatusCarrier>
</int:BetSlipStatus>

<int:Commision>1</int:Commision>
<int:Description>Unit Test APSOHAUYUI</int:Description>
<int:Payout>55</int:Payout>
<int:PayoutBonus>55</int:PayoutBonus>
<int:ReferenceNumber>UT-38115dcca4ce4-1277-4c56-b48b-7ac46018123</int:ReferenceNumber>
<int:IssuerLicenseNumber>SHOP-UT-0002</int:IssuerLicenseNumber>

<int:Account>
  <int:IsVerified>true</int:IsVerified>
  <int:Username>UT-User332014</int:Username>
</int:Account>

<int:BetSlipItems>
  <!-- One or more repetitions:-->
  <int:CreateBetSlipRequest.BetSlipItemCarrier>
    <int:EventResult>1</int:EventResult>
    <int:EventStartDate>2019-10-07T19:20:13.6193833+03:00</int:EventStartDate>
    <int:ItemReferenceNumber>1</int:ItemReferenceNumber>
    <int:Status>Pending</int:Status>
    <int:BetType>Outright</int:BetType>
    <int:Competition>UEFA Champions League</int:Competition>
    <int:CompetitorA>Team 01</int:CompetitorA>
    <int:CompetitorB>Team 02</int:CompetitorB>
    <int:Description>Unit Test PTEQMSOCPU</int:Description>
    <int:EventKey>1</int:EventKey>
    <int:EventName>Team 01 Vs Team 15</int:EventName>
    <int:MarketType>Correct Score</int:MarketType>
    <int:MarketTypeKey>Correct Score</int:MarketTypeKey>
    <int:MarketTypeKeyBMRS>Correct Score</int:MarketTypeKeyBMRS>
    <int:Odds>8</int:Odds>
    <int:Region>Europe</int:Region>
    <int:SelectionKeyBMRS>1</int:SelectionKeyBMRS>
    <int:SelectionName>1</int:SelectionName>
    <int:Sport>Soccer</int:Sport>
  </int:CreateBetSlipRequest.BetSlipItemCarrier>

  <int:CreateBetSlipRequest.BetSlipItemCarrier>
    <int:EventResult>1</int:EventResult>
    <int:EventStartDate>2019-10-07T19:20:13.621383+03:00</int:EventStartDate>
    <int:ItemReferenceNumber>2</int:ItemReferenceNumber>
    <int:Status>Pending</int:Status>
    <int:BetType>Outright</int:BetType>
    <int:Competition>UEFA Champions League</int:Competition>
    <int:CompetitorA>Team 01</int:CompetitorA>
    <int:CompetitorB>Team 02</int:CompetitorB>
    <int:Description>Unit Test YGPIJAEKNL</int:Description>
    <int:EventKey>1</int:EventKey>
    <int:EventName>Team 01 Vs Team 16</int:EventName>
    <int:MarketType>Correct Score</int:MarketType>
    <int:MarketTypeKey>Correct Score</int:MarketTypeKey>
    <int:MarketTypeKeyBMRS>Correct Score</int:MarketTypeKeyBMRS>
    <int:Odds>6</int:Odds>
    <int:Region>Europe</int:Region>
    <int:SelectionKeyBMRS>1</int:SelectionKeyBMRS>
    <int:SelectionName>1</int:SelectionName>
    <int:Sport>Soccer</int:Sport>
  </int:CreateBetSlipRequest.BetSlipItemCarrier>

</int:BetSlipItems>

<int:Bets>
  <!-- Zero or more repetitions:-->
  <int:CreateBetSlipRequest.BetCarrier>

```

```

<int:BetReferenceNumber>1</int:BetReferenceNumber>
<int:BetStatus>
  <!-- One or more repetitions:-->
  <int:BetStatusCarrier>
    <int:CreatedOnDate>2019-09-27T19:20:13.6223832+03:00</int:CreatedOnDate>
    <int:Status>Pending</int:Status>
  </int:BetStatusCarrier>
</int:BetStatus>
<int:Payout>100</int:Payout>
<int:BetSlipItemReferenceNumbers>1</int:BetSlipItemReferenceNumbers>
<int:Description>Unit Test GGVRFPBDS</int:Description>
<int:InitialStake>22</int:InitialStake>
<int:MaxOdds>2</int:MaxOdds>
<int:MinOdds>5</int:MinOdds>
<int:NumberOfCombinations>86</int:NumberOfCombinations>
</int:CreateBetSlipRequest.BetCarrier>

<int:CreateBetSlipRequest.BetCarrier>
  <int:BetReferenceNumber>2</int:BetReferenceNumber>
  <int:BetStatus>
    <!-- One or more repetitions:-->
    <int:BetStatusCarrier>
      <int:CreatedOnDate>2019-09-27T19:20:13.6233824+03:00</int:CreatedOnDate>
      <int:Status>Pending</int:Status>
    </int:BetStatusCarrier>
  </int:BetStatus>
  <int:Payout>100</int:Payout>
  <int:BetSlipItemReferenceNumbers>2</int:BetSlipItemReferenceNumbers>
  <int:Description>Unit Test ARMUBSNYNQ</int:Description>
  <int:InitialStake>67</int:InitialStake>
  <int:MaxOdds>1</int:MaxOdds>
  <int:MinOdds>9</int:MinOdds>
  <int:NumberOfCombinations>28</int:NumberOfCombinations>
</int:CreateBetSlipRequest.BetCarrier>

</int:Bets>

<int:CreatedOnDate>2019-09-27T19:20:13.6166518+03:00</int:CreatedOnDate>
<int:InitialStake>16</int:InitialStake>
<int:InitialStakeBonus>20</int:InitialStakeBonus>
<int:MaxPayout>100</int:MaxPayout>
<int:MinPayout>20</int:MinPayout>
<int:TerminalId>FLWVE</int:TerminalId>
<int:TotalNumberOfCombinations>2</int:TotalNumberOfCombinations>
</int:CreateBetSlipRequest>
</tem:betSlips>
</tem:CreateBetSlips>
</soapenv:Body>
</soapenv:Envelope>

```

3.4.2.3 Methode der Antwort

Wenn die Methode *Wettscheine erstellen* aufgerufen wird, antwortet die API mit der XML-Struktur in Abbildung 3.8. Die Antwort gibt über das Feld *Referenznummer* Feedback zu jedem einzelnen empfangenen Wettschein. Die Fehlerarten, die zurückgegeben werden könnten, sind in Tabelle 3.20 beschrieben.

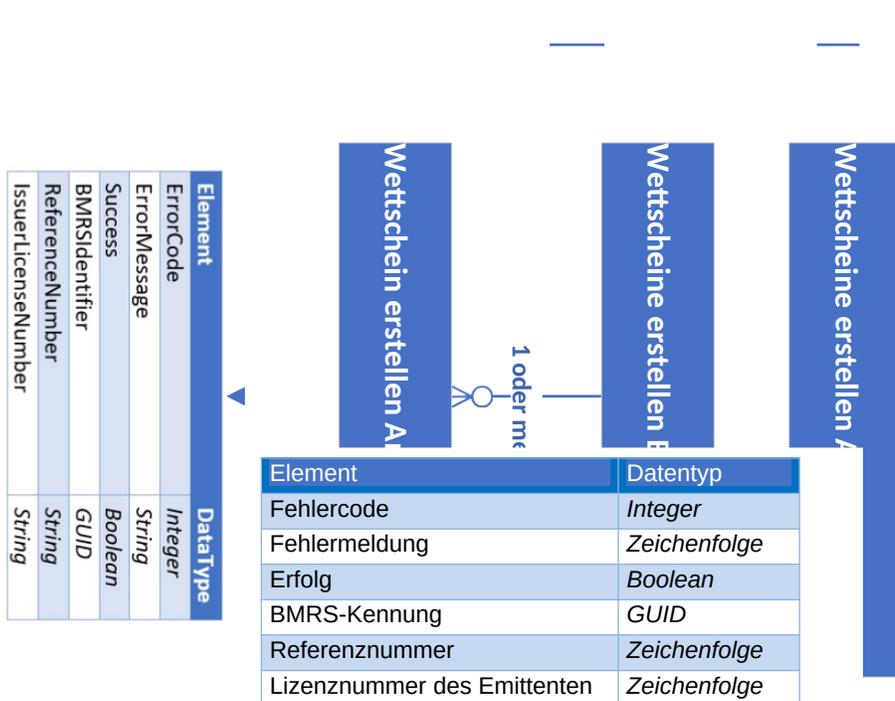


Abbildung 3.8: Die XML-Struktur von *Wettscheine erstellen* Antwort.

Fehlercode	Fehlermeldung	Beschreibung
1500	„Kein Lizenznehmer gefunden für die Lizenznummer: ‚{0}‘ Lizenznehmerkennung: ‚{1}‘ Dateneingabeschlüssel: ‚{2}‘“	Informationen über den im Header übergebenen Lizenznehmer stimmen mit keinem Lizenznehmer in BMRS überein (z. B. Ungültige Lizenznummer, Dateneingabeschlüssel etc.).
1501	„Vertreter des Lizenznehmers ‚{0}‘ nicht gefunden“	Vertreter des Lizenznehmers unbekannt
1503	„Lizenznehmer {0} repräsentiert nicht den Lizenznehmer {1}“	Der Vertreter des Lizenznehmers repräsentiert nicht den Buchmacher.
1504	„Feld ‚{0}‘: Wert ist null oder leer“	Ein obligatorischer Wert oder ein Element fehlt (z. B. der Header der Anfrage).
1505	„Feld ‚{0}‘: Wert ‚{1}‘ ist ungültig“	Der übergebene Wert ist ungültig (z. B. ungültige Ländercodes, undefinierte Aufzählungen, nicht erwartete negative Werte oder nicht erwartete Nullwerte).
1506	„Feld ‚{0}‘: Wert ‚{1}‘ ist doppelt“	Doppelte Elemente existieren in derselben Anfrage (z. B. Wettscheine, Wettscheinelemente, Wettlinien, Konten, Kontotransaktionen, Beschränkungen).
1514	„Lizenznehmer {0} nimmt keine Wetten an. Lizenznehmertyp: {1}“	Wettscheine dürfen nur für Klasse B oder GESCHÄFT sein. Klasse A und Vertreter müssen ihre Wetten mit einem Geschäft verbinden.

Tabelle 3.20: Fehlertypen in Wettscheine erstellen Antwort.

Das Folgende ist ein Beispiel für die SOAP/XML-Antwort dieser Methode:

Wettscheine erstellen Beispielantwort
<pre> <s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"> <s:Body> <CreateBetSlipsResponse xmlns="http://tempuri.org/"> <CreateBetSlipsResult xmlns:a="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Respo nse" xmlns:i="http://www.w3.org/2001/XMLSchema-instance"> <a:CreateBetSlipResponse> <a:ErrorCode>0</a:ErrorCode> <a:ErrorMessage i:nil="true"/> <a:Success>true</a:Success> <a:BMRSIdentifier>f7971e7c-09e5-47a6-a980-38822fe35597</a:BMRSIdentifier> <a:ReferenceNumber>UT-38115dcca4ce4-1277-4c56-b48b-7ac46018123</a:ReferenceNumber> <a:IssuerLicenseNumber>SHOP-UT-0002</a:IssuerLicenseNumber> </a:CreateBetSlipResponse> </CreateBetSlipsResult> </CreateBetSlipsResponse> </s:Body> </s:Envelope> </pre>

3.4.3 Methode „Wettscheine aktualisieren“

3.4.3.1 Beschreibung

Einmal erstellt, folgen Wettscheine ihrem Lebenszyklus bis zur Abrechnung, woraufhin der Spieler, der einen Wettschein erstellt hat, entweder verloren hat oder seine Gewinne erhalten hat (oder eine Rückerstattung, falls der Schein ungültig wurde). Eine Änderung des Wettscheinstatus (Tabelle 3.18), das Ende und das Ergebnis eines Wettscheinelements, eine teilweise Auszahlung durch einen Spieler oder die Änderung des Status einer Wette auf dem Wettschein sind Beispiele, bei denen der Status des Wettscheins geändert wurde.

Die API des BMRS bietet die Methode *Wettscheine aktualisieren*, um solche Änderungen im Status eines Wettscheins widerzuspiegeln. Die Methode akzeptiert Daten, die verwendet werden, um einen Wettschein zu referenzieren, zusammen mit Daten für die Felder, die eine Aktualisierung erfordern. Wie bei den zuvor erläuterten BMRS-Funktionen, kann die Funktion *Wettscheine aktualisieren* Daten in Stapeln aufnehmen, d. h. mehrere Belege können über einen einzigen Aufruf aktualisiert werden. Jeder Stapel kann eine *Wettschein-Referenznummer* nur einmal enthalten.

3.4.3.2 Methode der Anfrage

Die XML-Struktur dieser Methode ist in Abbildung 3.9 dargestellt. Einzelne Datenelemente sind in den Tabellen 3.21 bis 3.28 beschrieben.

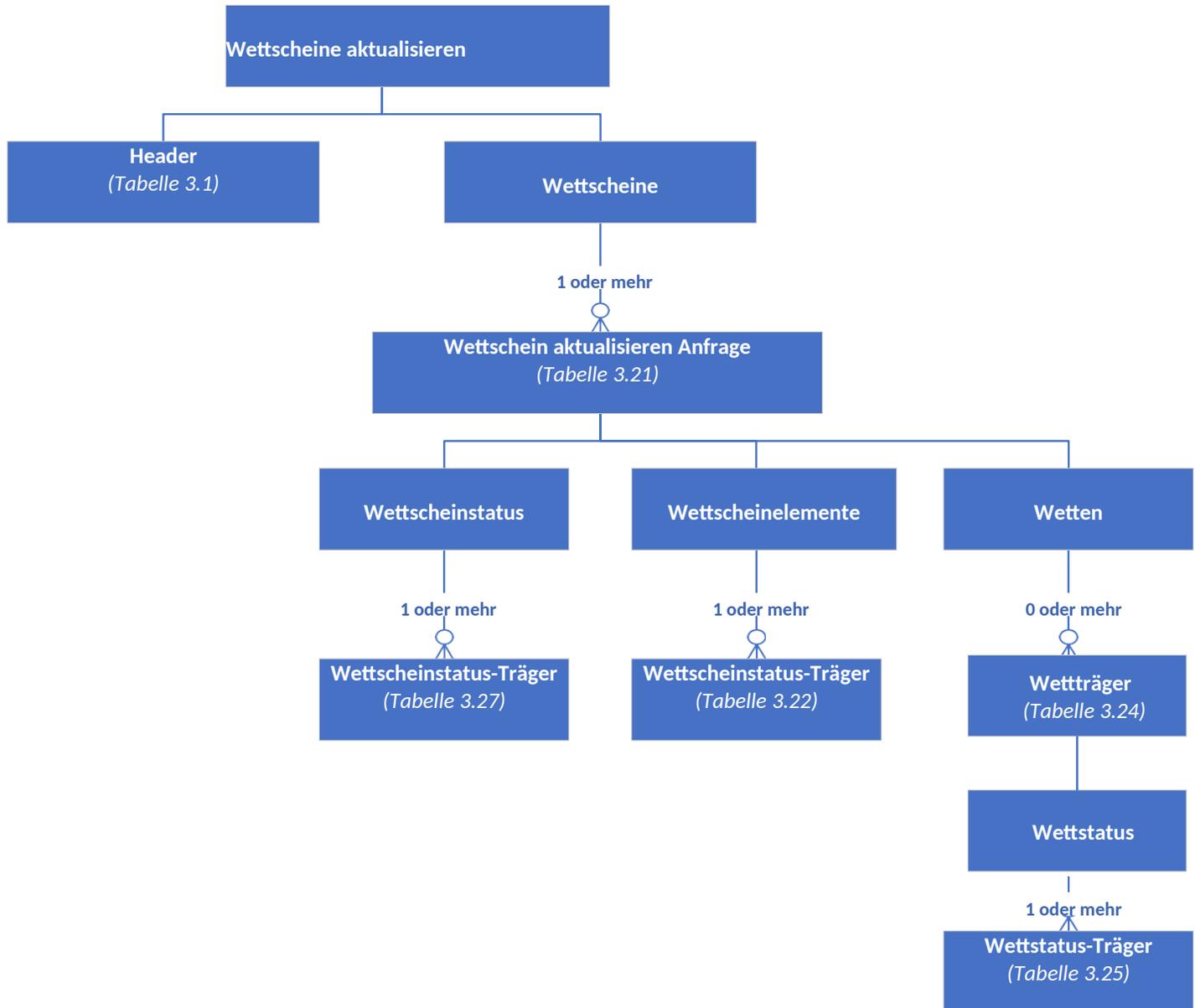


Abbildung 3.9: Die XML-Struktur der Methode *Wettscheine aktualisieren*.

Wettschein aktualisieren Anfrage (UpdateBetSlipRequest)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Referenznummer	Zeichenfolge	Eindeutige Kennung, die vom Buchmachersystem zugewiesen wird.	Wird verwendet, um auf den Wettschein zu verweisen, der dieser Aktualisierung unterliegt.
2	Lizenznummer des Emittenten	Zeichenfolge	Die Lizenznummer des Wettscheinemittenten.	
3	Auszahlung	Dezimal	Die gesamte endgültige Auszahlung dieses Wettscheins nach Abrechnung aller Wetten.	Inklusive Bonus.
4	Bonusauszahlung	Dezimal	Jeder Bonusbetrag in der gesamten endgültigen Auszahlung dieses Wettscheins.	
5	Beschreibung	Zeichenfolge	Alle zusätzlichen Informationen.	
6	Provision	Dezimal	Absoluter Betrag der an den Bevollmächtigten gezahlten Provision.	

Tabelle 3.21: Datenfelder (Unterelemente) unter dem Element Wettschein aktualisieren Anfrage.

Wettscheinstatus-Träger (BetSlipItemCarrier)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Elementreferenznummer	Zeichenfolge	Referenznummer, die diesem Wettscheinelement vom System des Buchmachers zugewiesen wird.	Eindeutige Kennung nur in diesem Wettschein. Wird hier verwendet, um auf das Element zu verweisen, das dieser Aktualisierung unterliegt.
2	Datum des Veranstaltungsbegins	Datum Zeit	Das UTC-Datum und die Uhrzeit, zu der das zugrunde liegende Ereignis beginnt.	
3	Ereignis Ergebnis	Zeichenfolge	Das Endergebnis des zugrunde liegenden Ereignisses.	
4	Status	Aufzählung: [Ausstehend] [Gewonnen] [Verloren] [Ungültig]	Der Status der Auswahl des Spielers für den Markt dieses Elements.	Siehe Tabelle 3.23.

Tabelle 3.22: Datenfelder (Unterelemente) unter dem Element Wettscheinelement-Träger.

Status	Beschreibung
Ausstehend	Das Ergebnis des Marktes ist noch nicht entschieden.
Gewonnen	Das Ergebnis des Marktes ist das gleiche wie die Auswahl des Spielers.
Verloren	Das Ergebnis des Marktes unterscheidet sich von der Auswahl des Spielers.
Ungültig	Das Ergebnis des Marktes kann nicht ermittelt werden (z. B. Ereignis storniert).

Tabelle 3.23: Wettscheinelement-Träger – **Status-Feldwerte.**

Wetträger (BetCarrier)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Wettreferenznummer	Zeichenfolge	Referenznummer dieser Wette. Vom Buchmachersystem zugewiesen.	Eindeutige Kennung nur innerhalb dieses Wettscheins. Wird hier verwendet, um auf die Wette zu verweisen.
2	Auszahlung	Dezimal	Die gesamte endgültige Auszahlung dieser Wette. Inklusive Bonus.	

Tabelle 3.24: Datenfelder (Unterelemente) unter dem Element Wetträger.

Wettstatus-Träger (BetStatusCarrier)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Status	Aufzählung [Ausstehend] [Gewonnen] [Verloren] [Ungültig]	Der aktuelle Status dieser Wette.	Siehe Tabelle 3.26.
2	Erstellt am	Datum Zeit	UTC-Datum und Uhrzeit, zu der dieser Status im Buchmachersystem erreicht wurde.	

Tabelle 3.25: Datenfelder (Unterelemente) unter dem Element Wettstatus-Träger.

Status	Beschreibung
Ausstehend	Die Wette steht noch aus.
Gewonnen	Mindestens eine der zugrunde liegenden Kombinationen der Wette wurde realisiert.
Verloren	Keine der zugrunde liegenden Kombinationen der Wette wurde realisiert.
Ungültig	Die Wette ist nicht mehr gültig.

Tabelle 3.26: Wettstatus-Träger – Status-Feldwerte.

Wettscheinstatus-Träger (BetSlipStatusCarrier)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Status	Aufzählung [Eingereicht] [Angenommen] [Storniert] [Abgelehnt] [Gewonnen] [Verloren] [Rückerstattung] [Gewinn ausgezahlt] [Rückerstattung ausgezahlt]	Der aktuelle Status dieses Wettscheins.	Siehe Abbildung 2.2 und Tabelle 3.28 für den Wettscheinlebenszyklus
2	Aktuelle Auszahlung	Dezimal	Jede Auszahlung, die durch diesen Wettschein gesichert wird. Inklusive Bonus.	
3	Abgewickelter Einsatz	Dezimal	Jeder Betrag des Einsatzes auf diesen Wettschein, der abgewickelt wurde. Inklusive Bonus.	
4	Erstellt am	Datum Zeit	Das UTC-Datum und die Uhrzeit, zu der diese Änderung in „Status“ oder in „Aktuelle Auszahlung“ oder „Abgewickelter Einsatz“ im System des Buchmachers aufgezeichnet wurde.	

Tabelle 3.27: Datenfelder (Unterelemente) unter dem Element Wettscheinstatus-Träger.

Status	Beschreibung
Eingereicht	Der Wettschein wurde im Buchmachersystem erstellt.
Angenommen	Der Buchmacher hat den Wettschein angenommen.
Storniert.	Der Spieler hat den Wettschein storniert.
Abgelehnt	Der Buchmacher hat den Wettschein abgelehnt.
Gewonnen	Eine oder mehrere Wetten auf dem Wettschein haben gewonnen. Der Betrag wurde noch nicht an den Spieler ausgezahlt.
Verloren	Keine der Wetten auf dem Wettschein hat gewonnen.
Rückerstattung	Der Einsatz wird dem Spieler zurückerstattet.
Gewinn ausgezahlt	Die Gewinne wurden an den Spieler ausgezahlt.
Rückerstattung ausgezahlt	Die Rückerstattung wurde an den Spieler ausgezahlt.

Tabelle 3.28: Wettscheinstatus-Träger – **Status**-Feldwerte.

Hinweis: Jeder Wettschein, der durch Übermittlung von Daten an die Methode *Aktualisierung von Wettscheinen* aktualisiert wird, kann mehrere damit verbundene *Wettscheinstatus-Träger* und/oder *Wettstatus-Träger* haben (Abbildung 3.9). Diese Funktion ermöglicht die Einbeziehung von mehr als einer Änderung in den zugrunde liegenden Parametern dieser Knoten, innerhalb desselben Aufrufs. Zum Beispiel, wenn ein Wettschein innerhalb desselben 90-Sekunden-Fensters gewonnen und bezahlt wurde, kann der Absender entscheiden, beide Änderungen in einem einzigen Aufruf an diese Methode zu übertragen, indem er zwei verschiedene und angemessen zeitgestempelte Elemente *Wettscheinstatus-Träger* in seine XML-Struktur aufnimmt.

Das Folgende ist ein Beispiel einer SOAP/XML-Anfrage für diese Methode:

Wettscheine aktualisieren Beispielanfrage:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tem="http://tempuri.org/"
xmlns:int="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Requ
est">
  <soapenv:Header/>
  <soapenv:Body>
    <tem:UpdateBetSlips>

      <tem:header>
        <int:DataEntryKey>F2D64583-DFC1-4DF1-A95D-02BCD1CBA489</int:DataEntryKey>
        <int:LicenseNumber>A-UT-0001</int:LicenseNumber>
        <int:LicenseeIdentifier>19666639-C7F8-4942-876D-CA1088B75EFE</int:LicenseeIdentifier>
      </tem:header>

      <tem:betSlips>
        <!--One or more repetitions-->
        <int:UpdateBetSlipRequest>
```

```

<int:BetSlipStatus>
  <!--One or more repetitions:-->
  <BetSlipStatusCarrier>
    <int:CreatedOnDate>2019-09-27T19:20:12.6183845+03:00</int:CreatedOnDate>
    <int:CurrentPayout>412</int:CurrentPayout>
    <int:SettledStake>26</int:SettledStake>
    <int:Status>Won</int:Status>
  </BetSlipStatusCarrier>
  <BetSlipStatusCarrier>
    <int:CreatedOnDate>2019-09-27T19:20:13.6183845+03:00</int:CreatedOnDate>
    <int:CurrentPayout>412</int:CurrentPayout>
    <int:SettledStake>26</int:SettledStake>
    <int:Status>PaidWon</int:Status>
  </BetSlipStatusCarrier>
</int:BetSlipStatus>

<int:Commision>1</int:Commision>
<int:Description>description</int:Description>
<int:Payout>100</int:Payout>
<int:PayoutBonus>10</int:PayoutBonus>
<int:ReferenceNumber>UT-38115dcacfae4-1277-4c56-b48b-7ac46018123</int:ReferenceNumber>
<int:IssuerLicenseNumber>SHOP-UT-0002</int:IssuerLicenseNumber >

<int:BetSlipItems>
  <!--One or more repetitions:-->
  <int:UpdateBetSlipRequest.BetSlipItemCarrier>
    <int:EventResult>1</int:EventResult>
    <int:EventStartDate>2019-10-07T19:20:13.6193833+03:00</int:EventStartDate>
    <int:ItemReferenceNumber>1</int:ItemReferenceNumber>
    <int:Status>Won</int:Status>
  </int:UpdateBetSlipRequest.BetSlipItemCarrier>

  <int:UpdateBetSlipRequest.BetSlipItemCarrier>
    <int:EventResult>2</int:EventResult>
    <int:EventStartDate>2019-10-07T19:20:13.621383+03:00</int:EventStartDate>
    <int:ItemReferenceNumber>2</int:ItemReferenceNumber>
    <int:Status>Won</int:Status>
  </int:UpdateBetSlipRequest.BetSlipItemCarrier>

</int:BetSlipItems>

<int:Bets>
  <!--Zero or more repetitions:-->
  <int:UpdateBetSlipRequest.BetCarrier>
    <int:BetReferenceNumber>1</int:BetReferenceNumber>
    <int:BetStatus>
      <!-- One or more repetitions:-->
      <int:BetStatusCarrier>
        <int:CreatedOnDate>2019-09-27T20:20:13.6233824+03:00</int:CreatedOnDate>
        <int:Status>Won</int:Status>
      </int:BetStatusCarrier>
    </int:BetStatus>
    <int:Payout>120</int:Payout>
  </int:UpdateBetSlipRequest.BetCarrier>

  <int:UpdateBetSlipRequest.BetCarrier>
    <int:BetReferenceNumber>2</int:BetReferenceNumber>

    <int:BetStatus>
      <!-- One or more repetitions:-->
      <int:BetStatusCarrier>
        <int:CreatedOnDate>2019-09-27T20:20:13.6233824+03:00</int:CreatedOnDate>
        <int:Status>PaidWon</int:Status>
      </int:BetStatusCarrier>
    </int:BetStatus>

    <int:Payout>100</int:Payout>
  </int:UpdateBetSlipRequest.BetCarrier>

```

```

        </int:Bets>
    </int:UpdateBetSlipRequest>
  </tem:betSlips>
</tem:UpdateBetSlips>
</soapenv:Body>
</soapenv:Envelope>

```

3.4.4.3 Methode der Antwort

Wenn die Methode *Wettscheine aktualisieren* aufgerufen wird, antwortet die API mit der XML-Struktur in Abbildung 3.10. Die Antwort gibt über das Feld *Referenznummer* Feedback zu jedem einzelnen empfangenen Wettschein. Die Fehlertypen, die zurückgegeben werden können, sind in Tabelle 3.29 beschrieben.

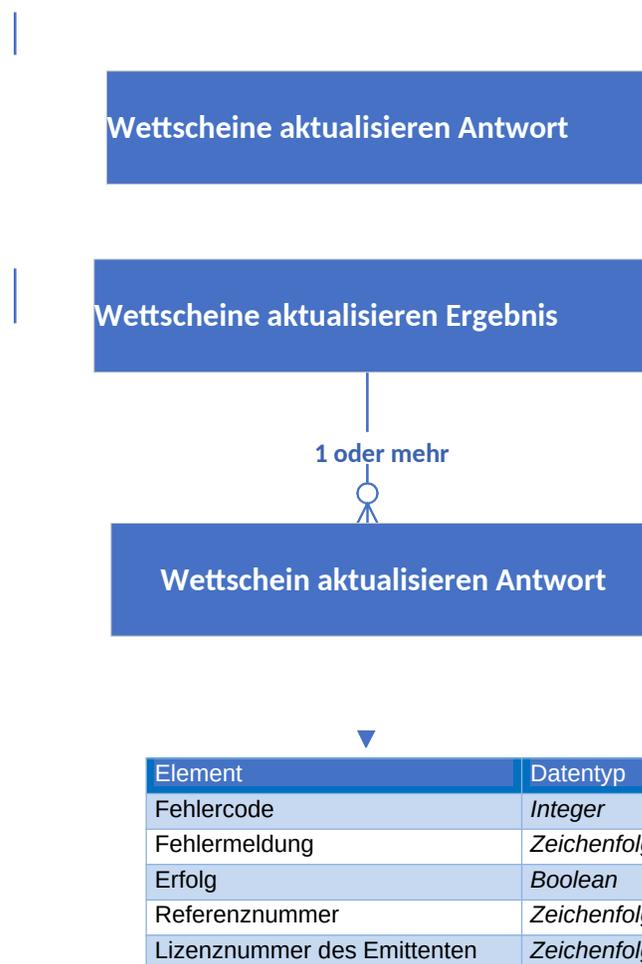


Abbildung 3.10: Die XML-Struktur von *Wettscheine aktualisieren Antwort*.

Fehlercode	Fehlermeldung	Beschreibung
1500	„Kein Lizenznehmer gefunden für die Lizenznummer: ‚{0}‘ Lizenznehmerkennung: ‚{1}‘ Dateneingabeschlüssel: ‚{2}‘“	Informationen über den im Header angegebenen Lizenznehmer stimmen mit keinem Lizenznehmer im BMRS überein (z. B. Ungültige Lizenznummer, ungültiger Dateneingabeschlüssel etc.)
1501	„Vertreter des Lizenznehmers ‚{0}‘ nicht gefunden“	Vertreter des Lizenznehmers unbekannt
1503	„Lizenznehmer {0} repräsentiert nicht den Lizenznehmer {1}“	Vertreter des Lizenznehmers repräsentiert nicht den Buchmacher
1504	„Feld ‚{0}‘: Wert ist null oder leer“	Ein obligatorischer Wert oder ein Element fehlt (z. B. der Header der Anfrage)
1505	„Feld ‚{0}‘: Wert ‚{1}‘ ist ungültig“	Der übergebene Wert ist ungültig (z. B. ungültige Ländercodes, undefinierte Aufzählungen, nicht erwartete negative Werte oder nicht erwartete Nullwerte)
1506	„Feld ‚{0}‘: Wert ‚{1}‘ ist doppelt“	Es gibt doppelte Einträge in derselben Anfrage (z. B. Wettscheine, Wettscheinelemente, Wettlinien, Konten, Kontotransaktionen, Beschränkungen)
1514	„Lizenznehmer {0} nimmt keine Wetten an. Lizenznehmertyp: {1}“	Wettscheine dürfen nur für Klasse B oder GESCHÄFT sein. Klasse A und Vertreter müssen ihre Wetten mit einem Geschäft verbinden.

Tabelle 3.29: Fehlertypen in Wettscheine aktualisieren Antwort.

Das Folgende ist ein Beispiel für die SOAP/XML-Antwort:

Wettscheine aktualisieren Beispielantwort:
<pre> <s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"> <s:Body> <UpdateBetSlipsResponse xmlns="http://tempuri.org/"> <UpdateBetSlipsResult xmlns:a="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Respo nse" xmlns:i="http://www.w3.org/2001/XMLSchema-instance"> <a:UpdateBetSlipResponse> <a:ErrorCode>0</a:ErrorCode> <a:ErrorMessage i:nil="true"/> <a:Success>true</a:Success> <a:ReferenceNumber>UT-38115dcca4-1277-4c56-b48b-7ac46018123</a:ReferenceNumber> <a:RepresentativeLicenseNumber>SHOP-UT-0002</a:RepresentativeLicenseNumber> </a:UpdateBetSlipResponse> </UpdateBetSlipsResult> </UpdateBetSlipsResponse> </s:Body> </s:Envelope> </pre>

3.4.5 Methode „Wettlinien speichern“

3.4.5.1 Beschreibung

Hinweis: Die NBA verlangt nicht die Linien einer jeden Wette innerhalb des regulären 90-Sekunden-Fensters für die Datenübermittlung. Diese Methode sollte nur auf Antrag der Behörde für einen oder mehrere Wettscheine aufgerufen werden.

Die Kombinationen oder Linien, die von Wetten abgeleitet werden, können von der NBA unter außergewöhnlichen Umständen verlangt werden, in denen ein Wettschein bis ins kleinste Detail untersucht werden muss. Für solche Fälle bietet die API des BMRS die Funktion *Wettlinien speichern*, deren Struktur in Abbildung 3.11. dargestellt ist. Die Datenfelder unter dem Element *Wettlinien-Träger* sind in Tabelle 3.30 beschrieben.

3.4.5.2 Methode der Anfrage

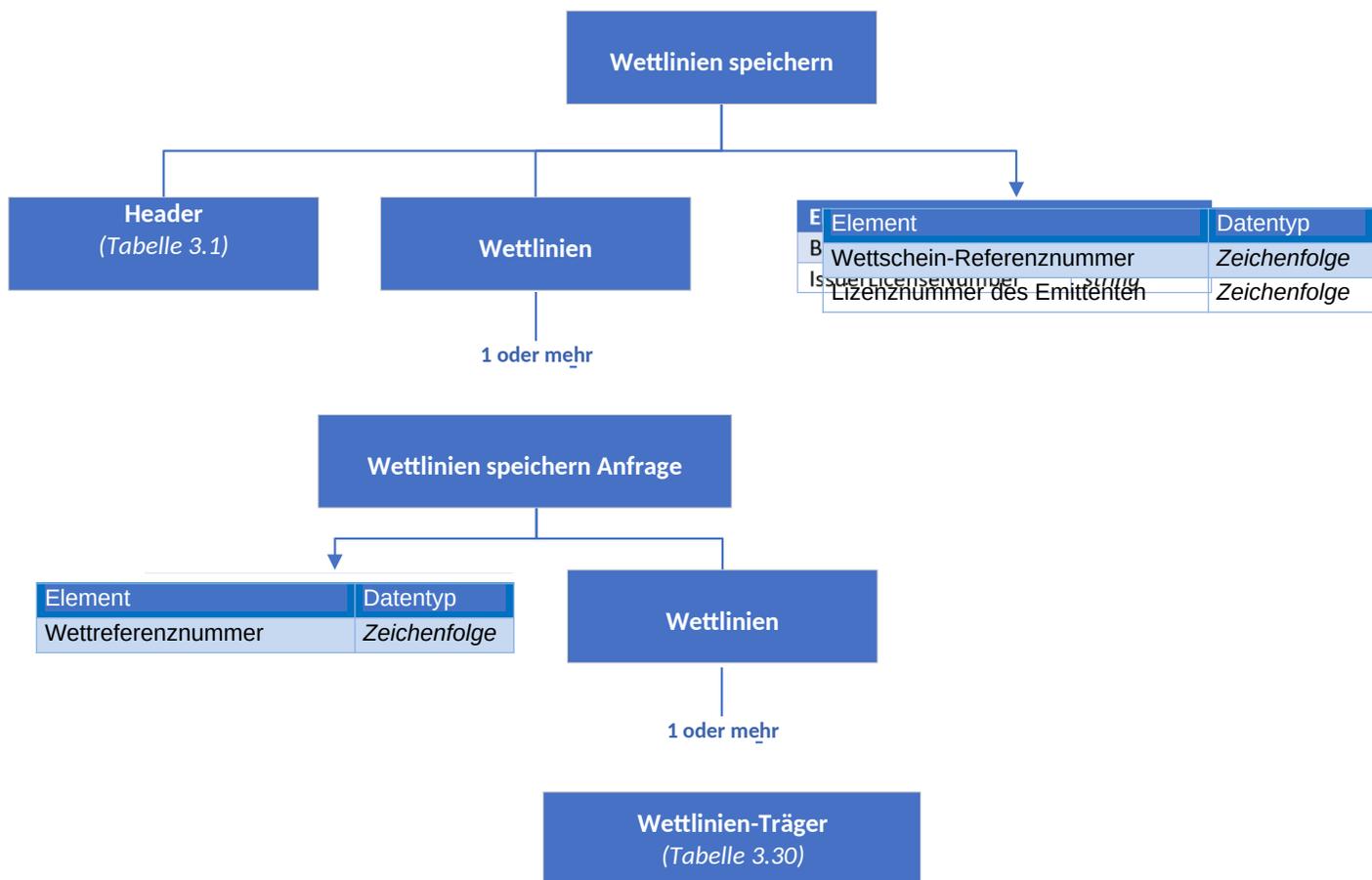


Abbildung 3.11: Die XML-Struktur der Methode *Wettlinien speichern*.

Wettlinien-Träger (BetLineCarrier)				
Nr.	Feldname	Datentyp	Beschreibung	Bemerkung
1	Linienreferenznummer	Integer	Eine eindeutige Kennung unter anderen Linien dieser Wette, die vom Buchmachersystem zugewiesen wird.	Vorzugsweise ein sequenzieller Integer.
2	Wettscheinelemente-Referenznummern	Zeichenfolge	Eine durch Kommata getrennte Zeichenfolge, die sich auf die Wettscheinelemente dieser Linie bezieht.	
3	Quoten	Dezimal	Die kombinierten Quoten dieser Linie.	
4	Einsatz	Dezimal	Der Betrag des Einsatzes auf dieser Linie. Inklusive Bonus.	

Tabelle 3.30: Datenfelder (Unterelemente) unter dem Element Wettlinien-Träger.

Das Folgende ist ein Beispiel für die SOAP/XML-Anfrage dieser Methode:

```

Wettlinien speichern Beispielanfrage

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tem="http://tempuri.org/"
xmlns:int="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Requ
est">
  <soapenv:Header/>
  <soapenv:Body>
    <tem:SaveBetLines>

      <tem:header>
        <int:DataEntryKey>C2EDF452-9B0E-4B75-85BA-D8C95F242371</int:DataEntryKey>
        <int:LicenseNumber>B-UT-0001</int:LicenseNumber>
        <int:LicenseeIdentifier>6254122F-CA27-4034-A629-B855DD0D25EC</int:LicenseeIdentifier>
      </tem:header>

      <tem:betSlipReferenceNumber>UT-38115dcca4e4-1277-4c56-b48b-7ac460188ff4</tem:betSlipReferenceNumber>
      <tem:IssuerLicenseNumber>B-UT-0001</tem:IssuerLicenseNumber>

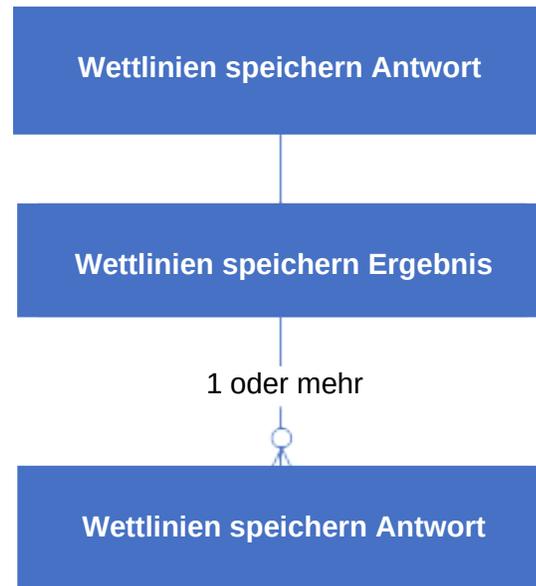
      <tem:betLines>
        <!--One or more repetitions-->
        <int:SaveBetLinesRequest>
          <int:BetLines>
            <!--One or more repetitions-->
            <int:SaveBetLinesRequest.BetLineCarrier>
              <int:BetSlipItemReferenceNumbers>1</int:BetSlipItemReferenceNumbers>
              <int:LineReferenceNumber>1</int:LineReferenceNumber>
              <int:Odds>12</int:Odds>
              <int:Stake>100</int:Stake>
            </int:SaveBetLinesRequest.BetLineCarrier>
          </int:BetLines>
          <int:BetReferenceNumber>?</int:BetReferenceNumber>
        </int:SaveBetLinesRequest>
      </tem:betLines>
    </tem:SaveBetLines>

  </soapenv:Body>
</soapenv:Envelope>

```


3.4.5.3 Methode der Antwort

Die API-Antwortstruktur nach einem Aufruf der Methode *Wettlinien speichern* ist in Abbildung 3.12. dargestellt. Die Fehlerarten, die zurückgegeben werden könnten, sind in Tabelle 3.31 beschrieben.



Element	Datentyp
Fehlercode	<i>Integer</i>
Fehlermeldung	<i>Zeichenfolge</i>
Erfolg	<i>Boolean</i>
Wettreferenznummer	<i>Zeichenfolge</i>

Abbildung 3.12: Die XML-Struktur von *Wettlinien speichern Antwort*.

Fehlercode	Fehlermeldung	Beschreibung
1500	„Kein Lizenznehmer gefunden für die Lizenznummer: ‚{0}‘ Lizenznehmerkennung: ‚{1}‘ Dateneingabeschlüssel: ‚{2}‘“	Informationen über den im Header angegebenen Lizenznehmer stimmen mit keinem Lizenznehmer im BMRS überein (z. B. Ungültige Lizenznummer, ungültiger Dateneingabeschlüssel etc.)
1501	„Vertreter des Lizenznehmers ‚{0}‘ nicht gefunden“	Vertreter des Lizenznehmers unbekannt
1503	„Lizenznehmer {0} repräsentiert nicht den Lizenznehmer {1}“	Vertreter des Lizenznehmers repräsentiert nicht den Buchmacher
1504	„Feld ‚{0}‘: Wert ist null oder leer“	Ein obligatorischer Wert oder ein Element fehlt (z. B. der Header der Anfrage)
1505	„Feld ‚{0}‘: Wert ‚{1}‘ ist ungültig“	Der übergebene Wert ist ungültig (z. B. ungültige Ländercodes, undefinierte Aufzählungen, nicht erwartete negative Werte oder nicht erwartete Nullwerte)
1506	„Feld ‚{0}‘: Wert ‚{1}‘ ist doppelt“	Es gibt doppelte Einträge in derselben Anfrage (z. B. Wettscheine, Wettscheinelemente, Wettlinien, Konten, Kontotransaktionen, Beschränkungen)
1514	„Lizenznehmer {0} nimmt keine Wetten an. Lizenznehmertyp: {1}“	Wettscheine dürfen nur für Klasse B oder GESCHÄFT sein. Klasse A und Vertreter müssen ihre Wetten mit einem Geschäft verbinden.

Tabelle 3.31: Fehlertypen in Wettlinien speichern Antwort.

Das Folgende ist ein Beispiel für die SOAP/XML-Antwort dieser Methode:

Wettlinien speichern Antwort:
<pre> <s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"> <s:Body> <SaveBetLinesResponse xmlns="http://tempuri.org/"> <SaveBetLinesResult xmlns:a="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Respo nse" xmlns:i="http://www.w3.org/2001/XMLSchema-instance"> <a:SaveBetLinesResponse> <a:ErrorCode>0</a:ErrorCode> <a:ErrorMessage i:nil="true"/> <a:Success>true</a:Success> <a:BetReferenceNumber?</a:BetReferenceNumber> </a:SaveBetLinesResponse> </SaveBetLinesResult> </SaveBetLinesResponse> </s:Body> </s:Envelope> </pre>

3.5 Systemaufrufe

Diese letzte Klasse von BMRS-Methoden handeln außerhalb des Umfangs jeder Wettgeschäftstätigkeit und werden verwendet, um Feedback an beide Enden des Datenübertragungskanal zu geben, nämlich die API und das System des Buchmachers.

3.5.1 Methode „Heartbeat“

3.5.1.1 Beschreibung

Ein Aufruf der Methode Heartbeat erstellen ist ein zeitgestempeltes Signal vom System des Buchmachers zum BMRS und bestätigt, dass die Verbindung zwischen den beiden Endpunkten aktiv ist, was in Zeiten des geringen Verkehrs relevant wird, wenn andere Methoden nicht mehr aufgerufen werden, weil keine Daten zu übertragen sind.

Hinweis: Ein regelmäßiger Aufruf dieser Funktion ist obligatorisch, d. h. sie ist nicht datenabhängig, da sie keine anderen Daten als einen Zeitstempel trägt.

3.5.1.2 Methode der Anfrage

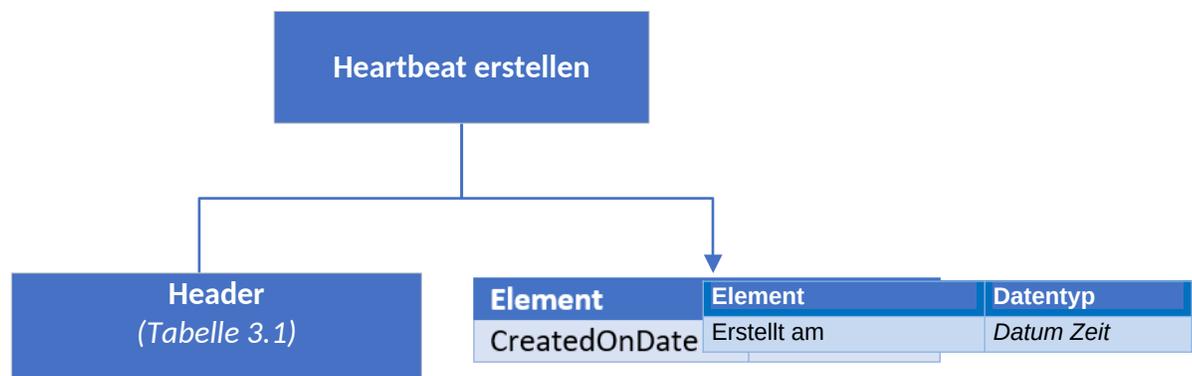


Abbildung 3.13: Die XML-Struktur der Methode Heartbeat erstellen.

Das Folgende ist ein Beispiel für die SOAP/XML-Anfrage dieser Methode:

<p>Heartbeat Beispielanfrage:</p> <pre> <soapenv:Envelope xmlns:soapenv="http" xmlns:tem="" xmlns:int="http://schemas.xmlsoap.org/soap/envelope/"> <soapenv:Header/> <soapenv:Body> <tem:CreateHeartbeat> <tem:header> <int:DataEntryKey>8F063140-4D72-3F01-1E8C-09C90F86722D</int:DataEntryKey> <int:LicenseNumber>A-90181</int:LicenseNumber> <int:LicenseeIdentifier>B3411DE4-5E4B-4199-FC08-4C4E0B49C2CF</int:LicenseeIdentifier> </tem:header> <tem:createdOnDate>2019-09-27T15:35:52.939Z</tem:createdOnDate> </tem:CreateHeartbeat> </soapenv:Body> </soapenv:Envelope> </pre>
--

3.5.2 Methode Fehlerprotokolle abrufen

3.5.2.1 Beschreibung

Wie bereits gezeigt, gibt das BMRS bei jedem Aufruf der API eine Antwort, und innerhalb dieser Antwort gibt es Feedback für mögliche Fehler, die beim Versuch, das gesendete Datenpaket zu verarbeiten, aufgetreten sein könnten. Diese Fehler sind jedoch von begrenztem Umfang, da sie das Ergebnis einer vorläufigen Datenprüfung sind, die während des Datenwarteschlangenprozesses (Dienste für Lizenznehmer) und bevor ein Vorgang auf Datenbankebene durchgeführt wird (Speicherdienste), stattfindet. Folglich nimmt das Absendersystem (Buchmacher) jegliche Fehler, die weiter im Speicherprozess aufgetreten sein könnten, nicht wahr.

Die Methode *Fehlerprotokolle abrufen* zielt darauf ab, die oben beschriebene Einschränkung zu entschärfen und auf Anfrage des Absenders Feedback auf einer tieferen Ebene zu geben. Die XML-Struktur für den Aufruf dieser Methode ist in Abbildung 3.14 und die entsprechende Antwort in Abbildung 3.15 dargestellt. In Tabelle 3.34 sind die Fehlertypen aufgeführt, die in der Antwort zurückgegeben werden könnten.

Hinweis: Im Gegensatz zu anderen Methoden der API folgt die Funktion *Fehlerprotokolle abrufen* nicht einem regelmäßigen Zyklus und wird nur aufgerufen, wenn der Absender Fehler überprüfen möchte.

3.5.2.2 Methode der Anfrage

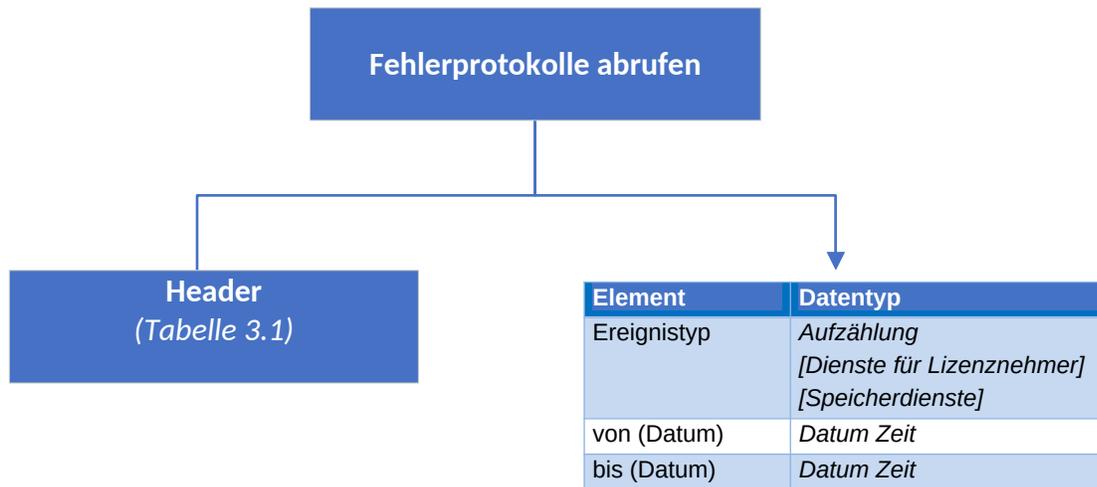


Abbildung 3.14: Die XML-Struktur der Methode Fehlerprotokolle abrufen.

Das Folgende ist ein Beispiel für die SOAP/XML-Anfrage für diese Methode:

Fehlerprotokolle abrufen Beispielanfrage:

```
<soapenv:Envelope xmlns:soapenv="http" xmlns:tem="" xmlns:int="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <tem:GetErrorLogs>
      <tem:header>
        <int:DataEntryKey>F2D64583-DFC1-4DF1-A95D-02BCD1CBA489</int:DataEntryKey>
        <int:LicenseNumber>A-UT-0001</int:LicenseNumber>
        <int:LicenseeIdentifier>19666639-C7F8-4942-876D-CA1088B75EFE</int:LicenseeIdentifier>
      </tem:header>
      <tem:eventType>StoringService</tem:eventType>
      <tem:fromDate>2019-09-20T15:00:00.000Z</tem:fromDate>
      <tem:toDate>2019-09-27T15:00:00.000Z</tem:toDate>
    </tem:GetErrorLogs>
  </soapenv:Body>
</soapenv:Envelope>
```

Ereignistyp	Beschreibung
Dienst für Lizenznehmer	Fehler, die bei der Übermittlung an die API erkannt wurden.
Speicherdienst	Fehler, die bei der Nachrichtenverarbeitung zum Speichern erkannt wurden.

Tabelle 3.33: Arten von Ereignissen in Fehlerprotokolle abrufen.

3.5.2.3 Methode der Antwort

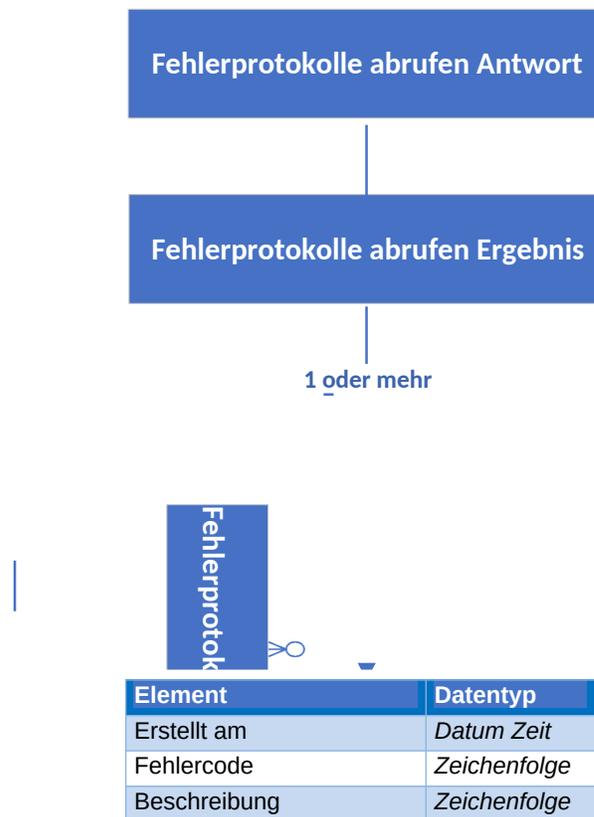


Abbildung 3.15: Die XML-Struktur von Fehlerprotokolle abrufen Antwort.

Fehlercode	Fehlermeldung	Beschreibung
1500	„Kein Lizenznehmer gefunden für die Lizenznummer: ‚{0}‘ Lizenznehmerkennung: ‚{1}‘ Dateneingabeschlüssel: ‚{2}‘“	Informationen über den im Header angegebenen Lizenznehmer stimmen mit keinem Lizenznehmer im BMRS überein (z. B. Ungültige Lizenznummer, ungültiger Dateneingabeschlüssel etc.)
1504	„Feld ‚{0}‘: Wert ist null oder leer“	Ein obligatorischer Wert oder ein Element fehlt (z. B. der Header der Anfrage)

Tabelle 3.34: Typen von Dienst für Lizenznehmer-Fehlern in Fehlerprotokolle abrufen Antwort.

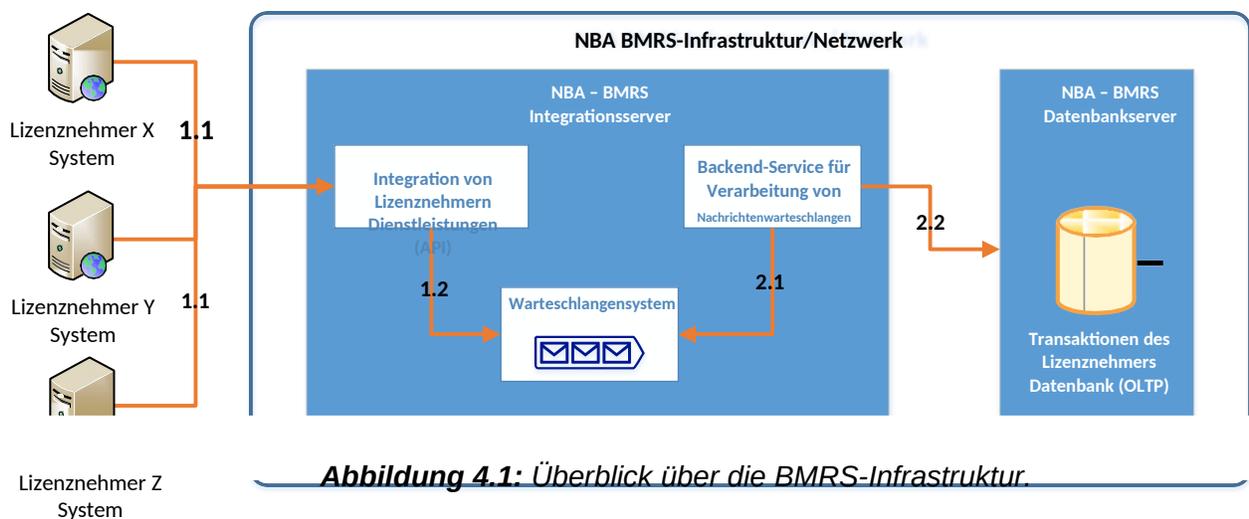
Fehlerprotokolle abrufen Beispielantwort
<pre> <s:Envelope xmlns:s="http"> <s:Body> <GetErrorLogsResponse xmlns="http"> <GetErrorLogsResult xmlns:a=" nse" xmlns:i="htt"> <a:ErrorLogCarrier> <a:CreatedOnDate>2019-09-25T15:40:38+03:00</a:CreatedOnDate> <a:Description>Storing transaction failed. Transaction with Reference Number: 'RefNo1' for Username: 'testxp1' already exist</a:Description> <a:ErrorCode>1507</a:ErrorCode> </a:ErrorLogCarrier> <a:ErrorLogCarrier> <a:CreatedOnDate>2019-09-25T15:40:51+03:00</a:CreatedOnDate> <a:Description>Storing transaction failed. Transaction with Reference Number: 'RefNo1' for Username: 'testxp1' already exist</a:Description> <a:ErrorCode>1507</a:ErrorCode> </a:ErrorLogCarrier> </GetErrorLogsResult> </GetErrorLogsResponse> </s:Body> </s:Envelope> </pre>

4.0 BMRS-Kommunikationsprotokolle

Die Kommunikation zwischen einer Buchmacherplattform und dem BMRS der NBA wird durch ein primäres und ein sekundäres Protokoll unterstützt.

4.1 Primäres Kommunikationsprotokoll – Die BMRS-API

Das primäre Kommunikationsprotokoll hat die Form einer über das Internet verfügbaren Anwendungsschnittstelle (API), über die lizenzierte Einrichtungen die erforderlichen Daten innerhalb eines 90-Sekunden-Zeitfensters an die NBA übertragen. Am Empfangsende wird vom BMRS ein Warteschlangenmechanismus verwendet, um einen ununterbrochenen Datenfluss zu gewährleisten und potenzielle Engpässe bei der Übertragung in Zeiten mit hohem Verkehrsaufkommen zu vermeiden, wie in der Abbildung unten dargestellt.



In der folgenden Liste sind die wichtigsten Abläufe für den Empfang und die Speicherung der Daten aufgeführt:

- **Schritt 1.1:** Lizenznehmersystem sendet Daten an die API.
- **Schritt 1.2:** Die API validiert die Daten und speichert sie im Warteschlangensystem. Vor dem Beenden sendet die API eine Antwort zurück, die eine Liste von Fehlern enthalten kann.
- **Schritt 2.1:** Ein separater Thread läuft ständig und liest Nachrichten, die in der Warteschlange gespeichert sind.
- **Schritt 2.2:** Nachrichten in der Warteschlange werden verarbeitet und in der Datenbank für Lizenznehmertransaktionen gespeichert.

4.1.1 Technische Spezifikationen

Das API-Schnittstellenprotokoll, das vom BMRS verwendet wird, ist die **XML/SOAP 1.1** Web Services, ergänzt durch kommunikative Metadaten über WSDL-Dateien, um die effektive Erkennung und Interpretation der verfügbaren Methoden zu ermöglichen. Dieses Protokoll wird vom „World Wide Web Consortium (W3C)“ gepflegt und basiert auf Anwendungsschichtprotokollen, die betriebssystemunabhängig sind und somit weithin übernommen werden können.

Über SOAP

SOAP ist eine Messaging-Protokollspezifikation für den Austausch strukturierter Informationen bei der Implementierung von Webdiensten in Computernetzwerken. Ihr Ziel ist es, Erweiterbarkeit, Neutralität und Unabhängigkeit zu induzieren. Sie verwendet das XML Information Set für ihr Nachrichtenformat und stützt sich auf Anwendungsschichtprotokolle, meistens Hypertext Transfer Protocol (HTTP) oder Simple Mail Transfer Protocol (SMTP), für die Aushandlung und Übertragung von Nachrichten. SOAP ermöglicht es Prozessen, die auf unterschiedlichen Betriebssystemen (wie Windows und Linux) laufen, mit der Extensible Mark-up Language (XML) zu kommunizieren. Da Webprotokolle wie HTTP auf allen Betriebssystemen installiert und ausgeführt werden, ermöglicht SOAP den Clients, Webdienste aufzurufen und Antworten unabhängig von Sprache und Plattformen zu erhalten.

Über WSDL

Die Web Services Description Language (WSDL) ist eine XML-basierte Interface-Definitionssprache, die zur Beschreibung der Funktionalität eines Webdienstes verwendet wird. Das Akronym wird auch für jede spezifische WSDL-Beschreibung eines Webdienstes (auch als WSDL-Datei bezeichnet) verwendet, die eine maschinenlesbare Beschreibung enthält, wie der Dienst aufgerufen werden kann, welche Parameter er erwartet und welche Datenstrukturen er zurückgibt. Daher ist ihr Zweck in etwa dem einer Typsignatur in einer Programmiersprache ähnlich.

Es wird empfohlen, dass Entwickler die bereitgestellte WSDL-Datei von BMRS-APIs verwenden, um die Proxyklassen in ihren Systemen zu generieren, indem sie den Netzwerkstandort der WSDL-Datei verwenden oder die Datei in ihre Projekte importieren. Dies ermöglicht die korrekte automatische Generierung aller Methoden, Klassen und verwandten Felder und eliminiert so mögliche Fehler, die beim manuellen Erstellen aller XML-Anfrage- und -Antwortbefehle auftreten.

4.1.2 Authentifizierung/Autorisierung

Die Authentifizierung der mit dem BMRS verbundenen Systeme erfolgt durch Shared Secrets (ein API-Schlüssel – zufällige UUIDs), die in den Header jeder Stapelanfrage an die Webdienste integriert werden müssen. Es wird darauf hingewiesen, dass die Webdienste zustandslos sein werden, sodass die Aufrufe bei allen Anfragen authentifiziert werden müssen.

Sowohl die Authentifizierung als auch die Autorisierung finden auf der **Lizenzebene** statt, was bedeutet, dass jedes Unternehmen ein Systemkonto pro Lizenz hat (z. B. eines für Klasse A und eines für Klasse B, falls beide vorhanden sind). Die Lizenznehmer können Daten über Wettaktivitäten senden, die mit den mit ihnen verbundenen repräsentativen Geschäften zusammenhängen.

4.1.3 Sicherheit (TLS)

Transport Layer Security (TLS), auch Secure Socket Layer (SSL) genannt, dient zum Schutz des Kommunikationskanals zwischen den Lizenznehmersystemen und den Webdiensten, die vom BMRS freigelegt werden. TLS schützt die Daten während des Transports, indem es sie verschlüsselt.

4.1.4 Netzwerkzugriffsregeln

Die BMRS-Webdienste werden auf Firewall-Ebene Netzwerkzugriffsregeln befolgen, um ihre Exposition gegenüber einer vordefinierten Whitelist von IP-Adressen zu begrenzen. Daher müssen beaufsichtigte Einrichtungen der Behörde die IP-Adressen ihrer Systeme melden, um einen angemessenen Netzzugang zu ermöglichen.

4.2 Sekundäres Kommunikationsprotokoll – XML-Dateien

Das sekundäre Integrationsprotokoll wird die Form von XML-Dateien annehmen und sollte nur nach Zustimmung der NBA eingesetzt werden.

Den Lizenznehmern wird ein vordefiniertes XML-Schema (XSD) zur Verfügung gestellt, um sicherzustellen, dass die Daten einem einheitlichen Standard entsprechen. Beaufsichtigte

Einrichtungen müssen die Daten gemäß dem XSD-Schema vorbereiten und zum Import an NBA-IT-Offiziere senden. Beaufsichtigte Einrichtungen müssen die Daten gemäß dem XSD-Schema in regelmäßigen Abständen erstellen, deren Zeitraum mit der Behörde vereinbart werden muss. Diese Dateien sollten dann entweder an einem sicheren Online-Standort gehostet werden, zu dem die NBA nach einem „Pull“-Protokoll Zugang zum Datenabruf erhält, oder über alternative Kommunikationskanäle an die Behörde gesendet werden.

Für die Vorlagen der XML-Dateien werden die gleichen Datensätze verwendet, die im vorherigen Abschnitt (Primäres Protokoll – Web Services) beschrieben sind.