 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Standards zur Softwareentwicklung
Stand: 01.10.2020

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Inhaltsverzeichnis

1	Ziel	3
2	Geltungsbereich	4
3	Begriffe.....	5
4	Projektorganisation.....	11
5	Systemlandschaft.....	12
5.1	Kodierung und Zeichensätze.....	12
5.2	Clients und Barrierefreiheit.....	12
5.3	Applikationsserver & Frameworks	13
5.4	Relationale Datenbanken	14
5.5	Identitätsmanagement und Zugriffsrechte	17
5.5.1	Authentifizierung.....	17
5.5.2	Autorisierung	17
5.6	Schnittstellen und Datenaustausch	17
5.7	Netztrennung und Verwaltungsnetz.....	18
6	Entwicklung.....	19
6.1	Entwicklungsumgebung	19
6.2	Quellcode-Organisation.....	19
6.3	Dokumentation der Softwareprojekte	19
6.4	Namensgebung.....	21
6.5	Internationalisierung.....	21
6.6	Strukturierte Ablage von Primärdaten.....	22
6.7	BVL-interne Server- und Entwicklungsorganisation	22
6.8	Extern entwickelte Software	23
7	Tests und Abnahme der Software	24
8	Freigabe von Anwendungen für den Produktivbetrieb	25
9	Nachweisdokumente	26
10	Mitgeltende Unterlagen	27
11	Änderungen an diesem Dokument	28
	Anhang: Quellcode-Organisation.....	30
	Anhang: Architekturkonzept	34

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

1 Ziel


Für die Entwicklung und den Betrieb von Software im BVL und für das BVL werden Standards definiert.

Ziel dieser Standards ist ein vereinfachter Entwicklungsprozess, bei dem der Einarbeitungsaufwand der Entwickler in neue Projekte möglichst geringgehalten wird und Vor- bzw. Nachteile einzelner Aspekte der Softwareentwicklung nicht jedes Mal neu gegeneinander abgewogen werden müssen. Außerdem soll die Zusammenarbeit im Team dadurch erleichtert werden, da die Kommunikation der Entwicklerinnen und Entwickler auf den gleichen Technologien basiert. Durch den Wegfall eines großen Teils der Einarbeitung soll eine bessere Vertreterregelung und eine bessere Übergabe der Produkte von Entwicklungsprojekten an den Betrieb erreicht werden.

Die in diesem Dokument getroffenen Entscheidungen zur (möglichst einheitlichen) Infrastruktur und zu den verwendeten Technologien dienen dazu, verbindliche Vorgaben zu treffen, so dass im Vorfeld der Betrieb und die spätere Wartung der Software in Z32 geplant werden kann.

Die Standards sind sowohl für interne Projekte als auch Projekte mit externen Partnern verbindlich - Projekttypen nach V-Modell XT Bund: 3.1 Systementwicklungsprojekt (AN). 3.2 Systementwicklungsprojekt (AN/AG). Es werden keine Alternativen zu den aufgeführten Produkten und Technologien verwendet. Sollten sich bestimmte Anforderungen technisch nicht umsetzen lassen, so wird eine alternative Technologie in der AG Programmierung besprochen und gemeinsam beschlossen. Gleiches gilt für neue Technologien und neue Versionen der verwendeten Werkzeuge/Software. Die Abstimmung kann im Bedarfsfall auch kurzfristig per E-Mail oder per Telefon mit Protokoll erfolgen.

Die hier getroffenen Festlegungen beziehen sich folgend meist auf Individualsoftware – sie gelten aber auch (wenn anwendbar) für Standardsoftware. Abweichungen von hier definierten Standards für die Einführung einer Standardsoftware sind nur per Einzelfallentscheidungen unter Einbindung von RefL Z32 möglich.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

2 Geltungsbereich

BVL gesamt	<input checked="" type="checkbox"/>
Organisationseinheit	<input type="checkbox"/>

Die einheitlichen Vorgaben dieses Dokumentes gelten für die hausinterne Entwicklung, und sollen auch bei einer externen Auftragsvergabe verwendet werden.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

3 Begriffe

Allgemeine Begriffe

Begriff	Bedeutung
BMEL	Bundesministerium für Ernährung und Landwirtschaft
RefL	Referatsleiter/in im BVL

Spezifische Begriffe

Begriff	Bedeutung
Active Directory (AD)	Verzeichnisdienst von Microsoft
AG Programmierung	IT-Sitzung, in der Entwicklerinnen und Entwickler die Standards für die Softwareentwicklung im BVL erarbeiten.
ARIA	Accessible Rich Internet Applications. Initiative zur Verbesserung von webbasierten Benutzeroberflächen, um die Bedienung für Menschen mit Behinderungen zu vereinfachen.
Authentifizierung	Überprüfung der tatsächlichen Identität des Endbenutzers
Autorisierung	Zuweisung und Überprüfung von Zugriffsrechten auf Daten und Dienste an Systemnutzer
BITV	Barrierefreie-Informationstechnik-Verordnung
Build	Steht in der Programmierung für die Entwicklungsstufe einer Software. Mittels eines Build-Scripts wird durch die automatisierte Kompilierung/Übersetzung des Codes eine neue lauffähige Softwareversion erzeugt.
Cronjob	Auf unixoiden bzw. unixähnlichen Betriebssystemen werden wiederkehrende Aufgaben über den sogenannten Cron-Dienst zeitbasiert abgewickelt. Eine solche Aufgabe bezeichnet man als Cronjob.
CSS	Cascading Style Sheet, deklarative Sprache für Stilvorlagen
DBMS	Datenbankmanagementsystem ist eine Verwaltungssoftware, die die technischen Voraussetzungen für den Betrieb eines DBS liefert.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Begriff	Bedeutung
DBS	Das Datenbanksystem besteht aus zwei Teilen: der Verwaltungssoftware, genannt (DBMS) und der Menge der zu verwaltenden Daten, der Datenbank im engeren Sinn, zum Teil auch „Datenbasis“ genannt.
DDL	Data Definition Language.
DMZ	Demilitarisierte Zone. Hierbei handelt es sich um ein Computernetz mit sicherheitstechnisch kontrollierten Zugriffsmöglichkeiten auf die angeschlossenen Server. Dort aufgestellte System werden durch eine oder mehrere Firewalls gegen andere Netzer abgeschirmt.
EJB	Enterprise JavaBeans (EJB) sind standardisierte Komponenten innerhalb eines Jakarta-EE-Servers (Java Enterprise Edition). Sie vereinfachen die Entwicklung komplexer mehrschichtiger verteilter Softwaresysteme mittels Java. Mit Enterprise JavaBeans können Konzepte für Unternehmensanwendungen, z. B. Transaktions-, Namens- oder Sicherheitsdienste, umgesetzt werden, die für die Geschäftslogik einer Anwendung nötig sind.
ERM	Entity-Relationship-Model. Hierbei handelt es sich um ein Objektbeziehungsmodell, um im Rahmen der semantischen Datenmodellierung einen in einem gegebenen Kontext relevanten Ausschnitt der realen Welt zu beschreiben.
FHS	Filesystem Hierarchy Standard. Standard für die Organisation der Verzeichnisstruktur unixoider Betriebssysteme entsprechend deren Einsatzzweck.
Flash	Adobe Flash ist der Name einer Plattform zur Programmierung und Darstellung multimedialer und interaktiver Inhalte
HTML	Hypertext Markup Language
IDMS	Spezialisiertes System für das Identitätsmanagement. Im BVL wird für die Kommunikation mit diesem System LDAPv3 und RFC 3062 eingesetzt.
Java	ist eine objektorientierte Programmiersprache. Zum Ausführen von Java-Programmen wird eine Java-Laufzeitumgebung auf dem Rechner benötigt
Jakarta EE	Java Plattform, Enterprise Edition, abgekürzt Jakarta EE oder früher Java EE bzw. J2EE, ist die Spezifikation einer Softwarearchitektur für

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Begriff	Bedeutung
	die transaktionsbasierte Ausführung von in Java programmierten Anwendungen und insbesondere Webanwendungen. Sie ist eine der großen Plattformen, die um den Middleware-Markt kämpfen. Größter Konkurrent ist dabei die .NET-Plattform von Microsoft.
JavaDoc	JavaDoc ist ein Software-Dokumentationswerkzeug, das aus Java-Quelltexten automatisch HTML-Dokumentationsdateien erstellt.
JavaScript	ist eine Skriptsprache, die für dynamisches HTML in Webbrowsern entwickelt wurde, um Benutzerinteraktionen auszuwerten, Inhalte zu verändern, nachzuladen oder zu generieren und so die Möglichkeiten von HTML und CSS zu erweitern
JAX-WS	Java API for XML Web Services. Hierbei handelt es sich um eine Programmierschnittstelle zum Erstellen von Webservices.
JDBC	Java Database Connectivity. Hierbei handelt es sich um eine standardisierte Programmierschnittstelle für den Zugriff auf relationale DMBS. Alle großen Datenbankhersteller bieten passende Treiber an, die den technischen Zugriff auf die Datenbank kapseln.
JPA	Die Java Persistence API (JPA) ist eine Schnittstelle für Java-Anwendungen, die die Zuordnung und die Übertragung von Objekten zu Datenbankeinträgen vereinfacht.
JSF	Java Server Faces: ist ein Framework-Standard zur Entwicklung von grafischen Benutzeroberflächen für Webapplikationen
Kerberos	Bei Kerberos handelt es sich um ein Netzwerkprotokoll zur Authentifizierung von Benutzern, welches von der Annahme ausgeht, dass die Authentifizierung über ein unsicheres Netzwerk abgewickelt wird. Bei Kerberos handelt es sich um einen Industriestandard nach RFC 4120. ¹
MVC	Modell-View-Controller Pattern, ein Architekturmuster zur Strukturierung von Software in die drei Einheiten Datenmodell (engl. model), Präsentation (engl. view) und Programmsteuerung (engl. controller).
NdB	Netze des Bundes bezeichnet die hochverfügbare Netzwerkinfrastruktur, die von Bundesbehörden in Deutschland für den sicheren Datenaustausch genutzt wird.

¹ Siehe hierzu <https://tools.ietf.org/html/rfc4120>

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0


Begriff	Bedeutung
OR-Mapper	Objektrelationaler Mapper. Beim objektrelationalen Mapping (ORM) werden Objekte einer objektorientierten Programmiersprache auf Zeilen einer relationalen Datenbank abgebildet. Für das Programm erscheint die Datenbank dann als objektorientierte Datenbank.
PDF	Portable Document Format. Plattformunabhängiges Dateiformat gemäß ISO 32000-1:2008.
PDF/UA	Portable Document Format, Universal Accessibility. Substandard des PDF-Standards für barrierefreie PDF-Dokumente gemäß ISO 14289-1:2014.
RFC	Request for Comments. Hierbei handelt es sich um eine Reihe technischer und organisatorischer Dokumente, deren Erstellung am 7. April 1969 begonnen wurde. Aus historischen Gründen behalten die Diskussionsdokumente ihren Namen auch wenn sie final verabschiedet und zum Standard erklärt wurden.
RWD	Responsive Webdesign. Entwicklung einer Webanwendung für den Einsatz auf beliebigen End- und Anzeigegeräten. Ziel ist die dynamisch optimierte Darstellung der grafischen Benutzeroberfläche unter Berücksichtigung der zur Verfügung stehenden Auflösung und der vorhandenen Eingabemethoden.
SAGA 5.0	Standards und Architekturen für E-Government-Anwendungen.
SAML	Security Assertion Markup Language. Hierbei handelt es sich um ein XML-Framework zum Austausch von Authentifizierungs- und Autorisierungsinformationen.
Schema-Manager	Hierbei handelt es sich um ein Framework zum Nachverfolgen, Verwalten und Anwenden von Datenbankschemata. Die Kommunikation mit der Datenbank erfolgt über JDBC und ist prinzipiell unabhängig von einem konkreten DBMS. Alle Änderungen werden in Konfigurationsdateien im XML-Format gespeichert und können inkrementell auf eine bestehende Datenbank angewendet werden. Die Konfigurationsdateien lassen sich versionieren.
SQL	Structured Query Language. Hierbei handelt es sich um eine Datenbanksprache zur Definition von Datenstrukturen in relationalen Datenbanken sowie zum Bearbeiten (Einfügen, Verändern, Löschen) und Abfragen von darauf basierenden Datenbeständen.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Begriff	Bedeutung
SSO	Single Sign-on. Der Benutzer kann nach einmaliger Anmeldung an seinem Arbeitsplatz auf alle Rechner und Dienste zugreifen, für die er lokal autorisiert ist.
TCP	Transmission Control Protocol. Hierbei handelt es sich um ein zuverlässiges, verbindungsorientiertes und paketvermitteltes Netzwerkprotokoll für den Datenaustausch zwischen Netzwerkkomponenten.
Pilotphase	<p>Der Betrieb von Anwendungen durchläuft verschiedene Phasen.</p> <p>Testphase: Für die Anwendung kann noch kein fehlerfreier Betrieb garantiert werden, meist haben nur wenige interne Nutzer Zugriff auf die Anwendung, die Anwendung ist hinsichtlich der Vertraulichkeit (oft nur fiktive Testdaten), der Verfügbarkeit (Ausfälle müssen akzeptiert werden) und der Integrität (jederzeit könnten Daten manipuliert werden oder verloren gehen) nur wenig geschützt.</p> <p>Pilotphase: Für die Anwendung gelten die gleichen Anforderungen wie für den Produktivbetrieb, wobei jedoch der Nutzerkreis und/oder die Verwendung noch eingeschränkt werden.</p>
Produktivbetrieb	Produktivbetrieb: Die Entwicklung der Anwendung ist abgeschlossen, Fehler wurden behoben, die Infrastruktur für die Anwendung wird entsprechend der IT-Sicherheitsanforderungen betrieben. Die Anwendung wird vollumfänglich verwendet.
V-Modell XT Bund	Das Vorgehensmodell „V-Modell 97“ wurde im Zuge neuer Erkenntnisse im Februar 2005 durch das „V-Modell XT“ ersetzt. Das „XT“ steht hierbei für „Extreme Tailoring“ und beschreibt die Möglichkeit, sich den Bedürfnissen durch Zuschnitt optimal annähern zu können.
WAF	Web Application Firewall. Verfahren zum Schutz von Webanwendungen über dem Hypertext Transfer Protocol (HTTP). Die WAF überwacht auf der Anwendungsebene die Kommunikation und erfordert in der Regel keine Anpassung der zu schützenden Anwendung.
WAI	Web Accessibility Initiative. Hierbei handelt es sich um einen Bereich innerhalb des W3C, der sich in mehreren Arbeits- und Interessengruppen mit dem barrierefreien Zugang zum Web und seinen Inhalten beschäftigt.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Begriff	Bedeutung
Wiki	Mit diesem hawaiianischen Wort wird eine Website beschrieben, deren Inhalte von den Besuchern nicht nur gelesen, sondern auch direkt im Webbrowser bearbeitet und geändert werden können.
XML	IT: XML ist die Kurzbezeichnung für „extensible markup language“, einer Dokumentenbeschreibungssprache.
XÖV	„XML in der Öffentlichen Verwaltung“ – (umfangreiche) Vorgaben für die XML-Modellierung und Standardisierung zum Austausch von Daten.
XSD	XML Schema Definition. Hierbei handelt es sich um eine Empfehlung des W3C zum Definieren von Strukturen für XML-Dokumente. Diese Definitionen sind selbst auch in XML geschrieben.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

4 Projektorganisation

Die Projektorganisation und der Projektablauf im BVL und in der Zusammenarbeit mit externen Partnern sind in der **BVL_PA_07_0003_000 Projekte** detailliert beschrieben. Folgend werden die Mindestanforderungen für die Projektdurchführung dargestellt:

- Vor der Erstellung von Leistungsbeschreibungen für Softwareprojekte wird normalerweise in einem Fachkonzept das gesamte im Projekt Umzusetzende zusammengestellt. Dabei werden auch diejenigen Projektziele beschrieben, die im Rahmen der Projektarbeit nicht durch Software umgesetzt werden sollen (z.B. Organisationsänderungen, Schulungen, etc.).
- Auf der Basis des Fachkonzepts wird entschieden, ob die Umsetzung über eine oder mehrere Softwareentwicklungen erfolgen soll und welcher Teil des Fachprozesses IT gestützt umgesetzt werden soll.
- Die konkreten Anforderungen (Leistungsbeschreibung) an eine neue Software werden schriftlich in Form eines Lastenhefts erstellt. Die detaillierten Anforderungen werden anschließend vom Auftragnehmer in ein Pflichtenheft umgearbeitet, welches von Fachseite und IT abgenommen werden muss. Erst danach erfolgt die Umsetzung. Bei späteren Änderungswünschen werden die jeweiligen Änderungsanforderungen schriftlich niedergelegt und priorisiert.
- Eine Zeit- und Ressourcenplanung wird schriftlich verfasst und vom Projektleiter fortgeschrieben.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

5 Systemlandschaft

Anwendungen werden in der Regel als Webanwendungen implementiert: Dabei wird generell dem Standard SAGA 5.0 gefolgt. Außerdem werden Anwendungen als Jakarta-EE-Webanwendungen umgesetzt.

Die Architektur der Webanwendungen folgt dem 4-Schichten-Modell (ggf. auch 5-Schichten möglich):

- Client
- Web-Schicht
- Modell-Schicht
- Datenbank
- (optionale 5. Schicht) Sicherungsschicht

Die genauen Produkte (Software und Version) sind verbindlich für das gesamte BVL im Dokument **BVL_TA_04_9311_Z3_Systemumfeld–Entwicklung** geregelt.


5.1 Kodierung und Zeichensätze

Die Standardkodierung auf allen Abstraktionsschichten ist UTF-8. Dies gilt ebenso für Ressourcen und Mediadateien, die im Rahmen der Dokumentation erstellt werden. Aufgrund der zahlreichen Möglichkeiten Zeichenketten zu verarbeiten, muss die Prüfung auf unzulässige Zeichen auf Applikationsebene z. B. durch Bean Validation erfolgen. Es besteht zwar die Möglichkeit, dies mit Spaltenbeschränkung (constraint) in die Schemadefinitionen aufzunehmen, jedoch müsste in einem solchen Fall die Fehlerbehandlung auf allen Abstraktionsschichten der Applikation durchgeführt werden – dies sollte daher vermieden werden bzw. in Ergänzung zu der Validierung auf Applikationsebene erfolgen.

5.2 Clients und Barrierefreiheit

Als Clients werden Webbrowser eingesetzt. Plug-Ins (z.B. Java Laufzeitumgebung, Flash) dürfen für die Nutzung nicht nötig sein. Falls Plug-Ins verwendet werden sollen, muss hierzu eine Genehmigung durch RefL Z31 und RefL Z32 eingeholt werden.

Die Anwendungen sollen mindestens im Internet Explorer und im Mozilla Firefox lauffähig sein. Für Webbrowser auf mobilen Geräten wie Smartphones und Tablet Rechnern soll eine Unterstützung von Android und iOS erfolgen, sofern sie nicht per Einzelfallentscheidung durch RefL Z32 für Verwaltungsanwendungen ausgeschlossen werden.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Außerdem ist folgendes zu beachten:

- Für die Darstellung der Inhalte ist der Online-Style-Guide der Pressestelle umzusetzen. Sofern Anforderungen an die Barrierefreiheit es erfordern, kann formlos eine Ausnahmegenehmigung zur Abweichung vom Guide bei der Pressestelle beantragt werden.
- Es soll eine Trennung von Design (CSS) und Inhalt (HTML) erfolgen.
- Die Anwendung muss barrierearm gemäß BITV 2.0 und WAI-ARIA sein, dies betrifft insbesondere:
 - Die Umsetzung der Anforderungen der BITV ist im Geschäftsbereich des BMEL auch für alle internen Anwendungen verbindlich.
 - Für den Einsatz auf unterschiedlichen Endgeräten muss responsives Webdesign (RWD) zur Anwendung kommen.
 - Zur Sicherstellung einer möglichst barrierefreien Anwendung sind darüber hinaus die Normen DIN EN 301549:2018-08 (Richtlinien für barrierefreie Webinhalte) und ISO 14289-1:2014 (PDF/UA) maßgeblich und müssen beachtet werden.

5.3 Applikationsserver & Frameworks

Die Web-Schicht und Modell-Schicht laufen auf einem Jakarta-EE-zertifizierten Applikationsserver als Jakarta-EE-Anwendung. Als Alternative zum Jakarta-EE-Applikationsserver ist ggf. der Einsatz anderer Webserver bzw. HTTP-Server möglich (Einzelfallentscheidung durch RefL Z32).

Neben der Sprache werden auch strategische Frameworks oder Bibliotheken festgeschrieben. Für Datenbanken mit Weboberfläche gelten dabei folgende Vorgaben:

- Verwendung von Jakarta EE. Der bereitgestellte Funktionsumfang durch den Applikationsserver entspricht dem Full Profile. Empfohlen wird jedoch ein möglichst sparsamer Einsatz der bereitgestellten Dienste.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

- Als Frontend-Technologie wird JSF mit Facelets eingesetzt, um W3C-konformes HTML und CSS zu generieren.² Abweichungen sind ggf. nach Absprache möglich.
- Als Persistenztechnologie wird JPA vorausgesetzt. Für das objektrelationale Mapping muss der mitgelieferte OR-Mapper des Jakarta-EE-Applikationsservers verwendet werden. Wird im Ausnahmefall ein anderer Webserver bzw. HTTP-Server eingesetzt, muss die Auswahl des OR-Mapper in Rücksprache mit RefL Z32 erfolgen.
- Sowohl für die Model- als auch die Controller-Schicht müssen EJB zum Einsatz kommen.
- Für regelmäßige Aufgaben sind zeitgesteuerte Beans (timer beans) zu verwenden, um Datenbankoperationen transaktionssicher und über die Geschäftslogik der Anwendung durchführen zu können. Der Einsatz von Cronjobs ist zu vermeiden und muss in Ausnahmefällen durch RefL Z32 genehmigt werden. Falls Cronjobs eingesetzt werden, müssen diese im Installations- und Betriebshandbuch vollständig dokumentiert sein.
- Für die statische Quellcodeanalyse müssen geeignete Werkzeuge zum Einsatz kommen, die in Kombination mit dem Build-Werkzeug verwendet werden. Die hierfür notwendigen Konfigurationen werden durch Ref Z32 bereitgestellt bzw. mit Ref Z32 abgestimmt.
- Für die statische Schwachstellenprüfung der Abhängigkeiten der Anwendung muss ein geeignetes Werkzeug zum Einsatz kommen, das in Kombination mit dem Build-Werkzeug verwendet wird.
- Generell wird mit dem MVC-Pattern gearbeitet.

5.4 Relationale Datenbanken

Die Speicherung von fachlichen Daten erfolgt in normalisierter Form in einem relationalen Datenbankmanagementsystem (DBMS) unter Verwendung der JDBC. Diese Festlegung gilt für alle Neuimplementierungen und ist bei Anpassungen von bestehender Individualsoftware zu prüfen. Ausnahmen vom Standard für das DBMS und ggf. die Verwendung von nicht-relationalen DBMS erfordern eine Genehmigung von RefL Z32.

² Siehe hierzu auch <http://validator.w3.org/> und <http://jigsaw.w3.org/css-validator/>

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Außerdem gelten folgende Vorgaben:

- Die Datenbankzugriffe müssen über Container-verwaltete Datenquellen (data sources) erfolgen. Die Verwendung von Deployment-spezifischen Konfigurationen für Datenbankverbindungen (z. B. via Deployment-Deskriptoren, Quelltextkodierungen etc.) sind unzulässig.
- Der Einsatz von herstellereigenen Erweiterungen des SQL-Standards ist unzulässig. Gelieferte SQL-Skripte (siehe Abschnitt 6.2) müssen ohne Änderungen im zu unterstützenden DBMS ausführbar sein. Der Einsatz von Prüfroutinen, um diese Anforderung zu umgehen, ist unzulässig.³
- Der Einsatz von Triggern, Stored Functions, Stored Procedures, speziellen Index-Typen, speziellen Datentypen etc. ist unzulässig.
- Technische Primärschlüssel müssen durch die Datenbank vergeben werden (identity columns).⁴ Hierfür ist die Verwendung des JPA-Erzeugungstyps **IDENTITY** zu verwenden, sofern es sich um einen numerischen Primärschlüssel handelt. Die Verwendung von Identifikatoren wie UUIDs im Sinne von synthetischen Schlüsseln⁵ ist zulässig, falls technisch sichergestellt ist, dass es zu keinen Kollisionen (Mehrfachzuweisungen) kommt.
- Technische Zeitstempel müssen mit einer Auflösung in Millisekunden eingesetzt werden. Hierzu ist der SQL-Datentyp **TIMESTAMP(3)** zu verwenden. Für Zeitstempel auf fachlicher Ebene mit geografischem Bezug (Zeitzone) ist der SQL-Datentyp **DATETIME** zu verwenden. Für die technische Darstellung von Zeitangaben ist der Standard ISO-8601 anzuwenden.


³ Nach Rücksprache mit RefL Z32 sind Konfigurationskommandos zulässig, um das korrekte Parsen bei Datenimports steuern zu können, bspw. bei Oracle: **ALTER SESSION SET NLS_TIMESTAMP_FORMAT = 'YYYY-MM-DD HH24:MI:SS.FF';**

⁴ Bezüglich Oracle: <https://oracle-base.com/articles/12c/identity-columns-in-oracle-12cr1>

Bezüglich PostgreSQL: <https://www.postgresql.org/docs/10/static/sql-createtable.html>

Bezüglich MySQL: <https://dev.mysql.com/doc/refman/8.0/en/example-auto-increment.html>

⁵ Siehe hierzu <https://de.wikipedia.org/wiki/Surrogatschl%C3%BCssel>

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

- Bei der Speicherung großer binärer Objekte in der Datenbank (**@Lob**) muss im Rahmen von Konzeptworkshops festgelegt werden, wo die Primärdaten gespeichert werden und mit welchen Mechanismen sie vor Manipulationen geschützt werden, z. B. durch eine kryptografische Hashfunktion (Fingerprint) oder eine elektronische Signatur. Erfordert die Anwendung die Speicherung zahlreicher oder großer Binärdaten, müssen diese dateibasiert entsprechend dem Filesystem Hierarchy Standard⁶ und transaktionsgeschützt⁷ abgelegt werden. Der Speicherort muss konfigurierbar sein, i.d.R. handelt es sich um **/opt/data**. Abweichungen davon erfordern die Zustimmung von RefL Z32.
- Der Einsatz von nativen SQL-Kommandos im Quelltext ist unzulässig. Sämtliche Datenbankabfragen müssen über den OR-Mapper abgewickelt werden.
- Für häufig auftretende Datenbankabfragen stehen folgende Technologien zur Verfügung: EJB Named Query, EJB-QL oder die JPA Criteria API.


Der Aufbau der initialen Datenbankstruktur (Schema) und die Befüllung mit initialen Datenbankinhalten (Konstanten) werden über den Schema-Manager durchgeführt. Die Konfigurationsdateien des Schema-Managers werden im Unterordner **src/main/database** des Projektes gespeichert. Bei Änderungen durch eine neue Lieferung der Software werden die Konfigurationsdateien ebenfalls dort abgelegt, so dass der Schema-Manager in Kombination mit dem Build-Werkzeug das Datenbankschema aktualisieren kann.

- Auslagerung von Geschäftslogik auf die Datenbank ist generell unzulässig.
- Zur Sicherstellung der Synchronizität zwischen Datenbankschema und Modell-Schicht ist im Deployment-Deskriptor **persistence.xml** die Schema-Validierung zu konfigurieren, z. B.:

```
<property name="hibernate.hbm2ddl.auto" value="validate" />
```

⁶ Siehe hierzu auch <http://refspecs.linuxfoundation.org/fhs.shtml>

⁷ Siehe hierzu auch <https://narayana.io/sub-projects/index.html>, Abschnitt „Transactional File I/O“

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

5.5 Identitätsmanagement und Zugriffsrechte

5.5.1 Authentifizierung

Für das Identitätsmanagement wird Red Hat SSO in allen Netzsegmenten (Intranet, DMZ) eingesetzt. Die Installation von Red Hat SSO erfolgt über die Paketverwaltung des eingesetzten Betriebssystems. Erfolgt in Ausnahmefällen die Pflege der Benutzer über die Anwendung, muss eine geeignete Bibliothek für den LDAPv3-basierten Zugriff auf Red Hat SSO eingesetzt werden, die RFC 3062 implementiert.⁸ Passwortrichtlinien werden zentral über Red Hat SSO durchgesetzt. Die Konfiguration von einem oder mehreren Parametern der Passwortrichtlinie in der Fachanwendung ist unzulässig.

5.5.2 Autorisierung

Die Autorisierung muss in Red Hat SSO durch Konfiguration entsprechender Sicherheitsbereiche erfolgen, die von der Anwendung als Vorbedingung für die implementierten Anwendungsfälle ausgewertet werden. Zusätzlich muss in Red Hat SSO auch eine Gruppe für jede Anwendung eingerichtet werden, der nur die Benutzer zugeordnet werden, die sich in der jeweiligen Anwendung anmelden dürfen. Über Red Hat SSO ist SSO über SAML zu realisieren. In begründeten Ausnahmefällen ist auch das Kerberos-Protokoll möglich. Die hierfür notwendigen Maßnahmen werden in Absprache mit Referat Z32 festgelegt.


5.6 Schnittstellen und Datenaustausch

Als Datenaustauschformat zwischen verschiedenen Anwendungen wird XML eingesetzt, dabei sind die XÖV-Regeln zur Modellierung von XML umzusetzen, siehe Tabelle 1 auf Seite 20.

Für den Datenaustausch sind zustandslose Web-Services auf Grundlage von JAX-RS zu verwenden.⁹ Die jeweilige technische Umsetzung ist dabei für den jeweiligen Anwendungsfall zu prüfen und eine Abweichung vom empfohlenen Vorgehen mit RefL Z32 abzustimmen. Nach Absprache ist bei Bedarf auch eine nachrichtenorientierte Umsetzung gemäß JAX-WS möglich.

⁸ Für Schreibzugriffe auf den LDAP muss die vom BVL vorgeschriebene Richtlinie für die Speicherung von Passwortinformationen eingehalten werden, die beim Einrichten eines OpenLDAP-Servers konfiguriert wird. Eine RFC-3062-konforme Bibliothek verwendet die „LDAP Password Modify Extended Operation“, so dass die OpenLDAP-Konfiguration beachtet wird.


⁹ Sie hierzu auch <https://docs.oracle.com/javasee/7/tutorial/jaxws.htm>

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

5.7 Netztrennung und Verwaltungsnetz

Die netztopologische Platzierung der Anwendung ist abhängig vom Anwenderkreis und dem Schutzbedarf der verarbeiteten Daten. Diese Frage muss im Rahmen der Konzeptworkshops unter Unterstützung von Z3 und unter Beteiligung des IT-SiBe erörtert werden. Grundsätzlich gilt, dass Anwendungen für die Öffentlichkeit in der DMZ platziert werden müssen. Anwendungen in der DMZ können ausschließlich auf Ressourcen und Fremdsysteme zugreifen, die sich ebenfalls in der DMZ befinden.

Soll eine Anwendung den Mitarbeitenden des BVL oder anderen Teilnehmern des NdB zur Verfügung gestellt werden, muss diese zwingend im sicheren Verwaltungsnetz (Intranet) platziert werden. Die Bereitstellung der Anwendung für andere NdB-Teilnehmer muss darüber hinaus in enger Abstimmung mit Z33 und dem IT-SiBe erfolgen, da ggf. eine Genehmigung für die Verarbeitung von Verschlussachen des Geheimhaltungsgrades NUR FÜR DEN DIENSTGEBRAUCH (VS-NfD) beim Geheimschutzbeauftragten des BVL beantragt werden muss.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

6 Entwicklung

In diesem Kapitel werden Entwicklung, einzusetzende Werkzeuge und Vorgaben für die Umsetzung beschrieben.

6.1 Entwicklungsumgebung

Für die Übersetzung der Java Programme kommen folgende Werkzeuge zum Einsatz:

- Die Verwendung eines Build-Werkzeuges mit einem Abhängigkeitsmanagement ist zwingend notwendig.
- Die interne Entwicklung von Individualsoftware erfolgt unter Verwendung einer integrierten Entwicklungsumgebung.
- Für die Pflege der Quellcodes wird ein Versionsverwaltungssystem verwendet.

Grundsätzlich müssen frei verfügbare Programme, deren Lizenz die Nutzung im BVL nicht einschränkt, für das Kompilieren bzw. Übersetzen der Anwendungen verwendet werden. Ebenso sollen möglichst nur frei verfügbare Bibliotheken eingebunden werden.

6.2 Quellcode-Organisation

Die Quelltexte und Ressourcen-Dateien müssen anhand des vom Build-Werkzeug vorgegebenen Layouts organisiert werden. Bezugnehmend auf Abschnitt 5.4 ist hierbei insbesondere das Verzeichnis **src/main/database** zu beachten, in dem die Schema-Manager-Konfigurationen zum Erzeugen des Datenbankschemas und der initialen Befüllung der Tabellen abgelegt werden müssen.

6.3 Dokumentation der Softwareprojekte

Die Dokumentation der Software-Projekte und der neuen Software erfolgt grundsätzlich wie nachfolgend beschrieben:

- Im IT-Wiki wird das Projekt bzw. die Software nach dem vorgegebenen Schema dokumentiert.
- Der Quellcode, ggf. Schema-Manager-Konfigurationen und weitere Dokumentation wird in der Versionsverwaltung gepflegt.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Der geänderte Quellcode wird regelmäßig in die Versionsverwaltung übertragen (mindestens bei jeder neuen Lieferung). Die Daten werden nach den gängigen Standards der Versionsverwaltung strukturiert abgelegt.

Für die Dokumentation des Quellcodes müssen insbesondere aussagekräftige Kommentare eingefügt werden, so dass die generierte JavaDoc-Dokumentation verständlich und umfassend ist.

Zusätzlich zur Visualisierung der Klassenhierarchie, werden Datenbankschema-Visualisierungen und insbesondere Darstellungen der Systemlandschaft erstellt. Hierbei ist UML zu verwenden. Datenbankmodelle sind mittels des Entity-Relationship-Modells (ERM) zu modellieren. Dabei ist die Notation nach Elmasri/Navathe¹⁰ zu verwenden. Kardinalitäten sind in Min-Max-Notation anzugeben.

Beim Einsatz von UML und der Definition der XML-Schemata sind folgende Richtlinien des XÖV-Standard zu beachten:

Tabelle 1 Verpflichtende XÖV-Regelungen für XML-Protokollnachrichten.

Nr.	Verbindlichkeit	Kurzbeschreibung
NDR-9	MUSS	Nachrichten als globale Elemente
NDR-10	MUSS	Konsistente Namen in XÖV-Fachmodell und XML Schema-Definitionen
NDR-12	SOLL	Erlaubte Zeichen für Klassifikationen in Namen
NDR-22	MUSS	Unveränderte Übernahme von XÖV-Codelisten
NDR-28	MUSS	Valide W3C XML Schema-Definitionen
NDR-29	MUSS	Identifizierende Namensräume
NDR-30	MUSS	Versionierung der XML Schema-Definitionen
NDR-31	SOLL	Namensräume mit Versionen

¹⁰ R. Elmasri, S. B. Navathe: Grundlagen von Datenbanksystemen, Addison-Wesley

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

6.4 Namensgebung

Bei der Namensgebung gelten folgende Vorgaben:

- Es gelten die Java-Code-Conventions.
- Jedes Projekt erhält einen BVL-weit eindeutigen Kurznamen ohne Leer- oder Sonderzeichen. Groß- und Kleinschreibung darf nicht signifikant sein, da für Java spezielle Regeln existieren.
- Package-Namen in Java entsprechen der generischen Form


de.bund.bvl.<projekt>.<layer>.<package>

- **projekt** bezeichnet den eindeutigen Projektkurznamen.
- **layer** bezeichnet das Abstraktionsniveau der zugehörigen Pakete. Zulässige Werte sind **model**, **view**, **controller**, **services** und **utils**.
- **package** bezeichnet den eigentlichen Package-Namen.
- XSD-Schemata werden wie folgt abgelegt:
 - URIs werden nach dem Schema **http://www.bvl.bund.de/<Jahr>/<Projektkürzel>** gewählt.
 - Bei Bedarf kann, ähnlich wie beim W3C, unter der URL **http://www.bvl.bund.de/<Jahr>/<Kürzel>** eine kurze Beschreibung abgelegt werden.
 - Unter **http://www.bvl.bund.de/<Jahr>/<Kürzel>.xsd** werden die Schema-Definitionen abgelegt. Durch eine abweichende Endung können auch weitere Ressourcen abgelegt werden.

6.5 Internationalisierung

Um die Webanwendungen möglichst einfach zu internationalisieren, sollen Texte in Resource Bundles hinterlegt und im Deskriptor

`/src/main/webapp/WEB-INF/faces-config.xml`

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

konfiguriert werden.¹¹ Dies soll auch eine nachträgliche Übersetzung erleichtern.

6.6 Strukturierte Ablage von Primärdaten

Wurde im Rahmen der Konzeptworkshops zwischen dem AN und dem BVL die Notwendigkeit zur Speicherung von Primärdaten auf dem Dateisystem identifiziert, so muss die Ablage der Einzeldateien im Dateisystem entsprechend folgendem Schema erfolgen:

`/opt/data/<Zweck>/<Jahr>/<Monat>/<Tag>`

`/opt/data/certificates/2019/06/03`

Hierbei wird standardmäßig angenommen, dass das Mengengerüst der Primärdaten ein stetiges Wachstum aufweist und die Dateiablage für einen effizienten Zugriff strukturiert erfolgen muss. Standardmäßig genügt eine Unterteilung in Kalendertagen. Sollte die erwartete Datei-anzahl pro Kalendertag den Schwellwert **100** überschreiten, ist eine Feinunterteilung in Stunden, Minuten oder sogar Sekunden vorzunehmen.


6.7 BVL-interne Server- und Entwicklungsorganisation

Das IT-Wiki dient als zentraler Einstiegspunkt. Dort sind alle Informationen zur technischen Infrastruktur dokumentiert. Zu jedem Softwareprojekt wird eine eigene Wiki-Seite erstellt.

Darüber hinaus gelten folgende Vorgaben:

- „Kleine“ Anwendungen können mit 3-Schichten-Architektur erstellt werden. Welche Anwendung als „klein“ zu betrachten ist, wird in projektbezogenen Workshops definiert.
- Für jede Anwendung wird neben der künftigen Produktivumgebung auch eine separierte Testumgebung aufgebaut.
- Tickets zu Programmfehlern/Änderungsanforderungen (Request for Change) werden, sobald die Anwendung in den Betrieb übergeht und das Entwicklungsprojekt abgeschlossen wurde, vom BVL-eigenen IT-Service entgegengenommen.

¹¹ Siehe hierzu auch <https://docs.oracle.com/cd/E19798-01/821-1770/gcrr/index.html>

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

6.8 Extern entwickelte Software

Vorgaben für extern entwickelte Software:

- Der Quellcode extern entwickelter Software wird nach jeder Lieferung in die Versionsverwaltung eingestellt.
- Alle Unterlagen werden zentral nach den Vorgaben des V-Modell XT Bund abgelegt.

Vorgaben für extern entwickelte Software für Auftragnehmer:

- Der Auftragnehmer stellt ein zugangsgesichertes Bugtracking-Tool über HTTPS bereit, der Export der Tickets als PDF muss möglich sein. Die BVL-Mitarbeiter/innen sind verpflichtet, regelmäßig zu Dokumentationszwecken im Zusammenhang mit einer Abnahme einen Export der Tickets durchzuführen.
- Lieferungen sollen zugangsgesichert und per HTTPS über Port 443/TCP bereitgestellt werden (andere Ports sind im BVL für Datentransfer nicht verfügbar).
- Sofern Testsysteme beim Auftragnehmer betrieben werden, müssen diese zugangsgesichert und mit HTTPS-Verschlüsselung betrieben werden.
- Alle Softwareentwicklungsprojekte werden nach V-Modell XT Bund durchgeführt.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

7 Tests und Abnahme der Software


Neue Software wird gründlichen Tests unterzogen, insbesondere sollen auch Lasttests durchgeführt werden. Handelt es sich um intern programmierte Software, müssen sowohl Entwicklertests als auch Nutzertests von der Fachseite durchgeführt werden. Soll eine Lieferung eines externen Auftragnehmers getestet werden, müssen sowohl Tests der Fachseite als auch Tests durch Mitarbeiter von Z32 durchgeführt werden.

Die Tests sollen möglichst automatisiert durchgeführt werden. Folgende Werkzeuge sollen für diese Automatisierung eingesetzt werden:

- Durchführung von Frontend- und Akzeptanztests
- Durchführung von Lasttests
- Durchführung von Modultests
- Durchführung von Schnittstellentests
- Durchführung von Integrationstests

Die Testfälle sind schriftlich zu dokumentieren in Form eines Prüfprotokolls. Dieses Prüfprotokoll ist die Voraussetzung für die Abnahme einer Softwareversion. Zusätzlich ist bei Lieferungen externer Auftragnehmer zu prüfen, ob sich die Individualsoftware fehlerfrei auf den Servern des BVLs kompilieren lässt.

Der Auftragnehmer verpflichtet sich, dem BVL qualitätsgesicherte Produkte auszuliefern. Die Qualitätssicherung beinhaltet mindestens Nutzertests, die an den Anforderungen angelehnt sind. Es wird empfohlen, geeignete Tests zu entwickeln, um die Korrektheit der Software auch nach Änderungen/Softwareanpassungen zu gewährleisten.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

8 Freigabe von Anwendungen für den Produktivbetrieb

Neue Anwendungen müssen sowohl von der Fachseite als auch von RefL Z32 für den Produktivbetrieb freigegeben werden. Für die Freigabe des Produktivbetriebes müssen ausführliche und dokumentierte Abnahmetests der in den Betrieb zu überführenden Softwareversion auf einem Testsystem erfolgt sein. Die Dokumentation dieser Tests ist Z32 bei der Prüfung auf Freigabe für den Produktivbetrieb vorzulegen.


Sind Änderungen an der im Produktivbetrieb befindlichen Software notwendig, so ist die Software erneut zu testen. Erst nach positivem Abschluss der Tests dürfen die Änderungen in den Produktivbetrieb übernommen werden.

Sollte vor dem Produktivbetrieb eine Pilotphase durchgeführt werden, so gelten diese Regelungen zusätzlich auch für die Pilotphase.

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

9 Nachweisdokumente

Dokument	Führung	Aufbewahrung	Zeit	Ort	Bemerkung
Freigabe Produktivbetrieb (nach V-Modell XT)	Projektleiter	Projektleiter	3 Jahre	Projektleiter	
Zeit- und Ressourcenplanung (formlos)	Projektleiter	Projekteiter	Min. 3 Jahre	Projekteiter	

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0


10 Mitgeltende Unterlagen

Dokument
BVL_PA_07_0003_000 Projekte
BVL_TA_04_9309_Z3_IT-Dokumentation
BVL_TA_04_0005_000_Anwendung_Online_Styleguide
BVL_TA_04_9311_Z3_Systemumfeld-Entwicklung
Online-Styleguide des BVL
SAGA 5.0 Standard (siehe CIO Bund)
XÖV Standards 2.0.1 http://www.xoev.de/sixcms/media.php/13/XOEV-Handbuch_2_0_1.pdf
BITV 2.0 (Barrierefreie Internet Verordnung)
V-Modell XT Bund http://www.bit.bund.de/nn_2148612/BIT/DE/Standards__Methoden/V-Modell_20XT/node.html?__nnn=true

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

11 Änderungen an diesem Dokument

- Das Dokument wurde in der Version 4.0 den aktuellen Entwicklungen entsprechend angepasst. Hierzu zählen:
 - Endgültige Umbenennung der Spezifikationssammlung Java EE in Jakarta EE.
 - Umsetzung der Organisationsverfügung bezüglich Gruppe Z3.
 - Definition der Vorgaben zur strukturierten Ablage von Primärdaten.
 - Umbenennung des Begleitdokumentes zum Systemumfeld des BVL **BVL_TA_04_9311_Z3_Systemumfeld-Entwicklung**.
 - Hinzufügen der neuen Anforderungen zur Netztrennung und dem sicheren Verwaltungsnetz (Nutzerpflichten des NdB).
 - Hinzufügen der aktualisierten Anforderungen an die Barrierefreiheit
- Das Dokument wurde in der Version 3.0 den aktuellen Entwicklungen entsprechend angepasst. Hierzu zählen unter anderem
 - Wechsel auf eine allgemeine Versionsverwaltung. Spezifizierung erfolgt über das neue Dokument **BVL_TA_04_9311_030_Systemumfeld-Entwicklung**.
 - Wechsel auf einen Jakarta-EE-zertifizierten Applikationsserver
 - Verallgemeinerung der Persistierung in relationalen DBMS via JDBC
 - Wechsel von Cronjobs auf zeitgesteuerte Beans, die die Geschäftslogik der Anwendung für die Abwicklung regelmäßiger Aufgaben verwenden
 - Einführung eines Schema-Managers
 - Einführung von Integrationstests
 - Einführung eines generalisierten IDMS mit Zugriff via RFC 3062
 - Umbenennung des Begleitdokumentes zum Systemumfeld mit dem neuen QM-Schlüssel in **BVL_TA_04_9311_030_Systemumfeld-Entwicklung**

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

- Im Abschnitt 3 wurden allgemeine und spezifische Begriffe den Unterabschnitten neu zugeordnet.
- Das Dokument wurde in der Version 2.0 komplett überarbeitet.

Wesentliche inhaltliche Änderungen:

- Kapitel „Begriffe“ erweitert um „spezifische Begriffe“
- Umstrukturierung der Kapitel: Gliederung nach Systemlandschaft und Entwicklung mit entsprechenden Unterkapiteln
- Kapitel „Tests und Abnahme“ neu eingefügt bzw. Kapitel „Test“ erweitert
- Das Dokument wurde so umgestellt, dass allgemeingültige Vorgaben beschrieben werden. Die Festlegung von konkreten Versionsvorgaben wurde ausgelagert und erfolgt in dem Dokument **BVL_TA_04_9311_903_Systemumfeld - Entwicklung**

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Anhang: Quellcode-Organisation

Nachfolgend wird eine Ordnerstruktur vorgeschlagen, die für die Organisation sämtlicher Ressourcen einer Lieferung verwendet werden kann. Die Struktur orientiert sich an den Anforderungen, die in diesem Dokument formuliert wurden.

Verzeichnis, Inhalt und Bedeutung
src Zentrales Quelltextverzeichnis
src/main Hauptverzeichnis der Implementierungsquelltexte
src/main/config Konfigurationsdateien der Anwendung, z.B. Properties-Dateien (bevorzugt im XML-Format)
src/main/java[de.bund.bvl.<Applikationsname>] Zentrales Quelltextverzeichnis der Implementierung
src/main/java/de.bund.bvl.<Applikationsname>/controller Implementierung der Controller-Schicht
src/main/java/de.bund.bvl.<Applikationsname>/controller/services Dienste, die in der Anwendung von den darüber liegenden Schichten verwendet werden. Hierzu zählen zum Beispiel Dienste für die Benutzerverwaltung, das Persistieren von Objekten, die Kommunikation mit anderen System, die Umwandlung von Objekten in ein technisches Zielformat etc.
src/main/java/de.bund.bvl.<Applikationsname>/controller/connectors Implementierungen zum Verbindungsaufbau mit externen System über Standardschnittstellen, bspw. SOAP, LDAP, REST, CMIS etc.
src/main/java/de.bund.bvl.<Applikationsname>/controller/batch Enthält alle Arbeitssaufträge, die im Rahmen des Batch Processing implementiert werden (JSR-352). ¹²
src/main/java/de.bund.bvl.<Applikationsname>/controller/batch/batchlets Enthält alle Aufträge, die nicht mit großen Datenmengen hantieren müssen.

¹² Siehe hierzu auch <http://www.mastertheboss.com/javaee/batch-api/batch-applications-tutorial-on-wildfly>

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Verzeichnis, Inhalt und Bedeutung
<p>src/main/java/de.bund.bvl.<Applikationsname>/controller/batch/chunks</p> <p>Enthält alle Aufträge, die mit großen Datenmengen hantieren (chunks).</p>
<p>src/main/java/de.bund.bvl.<Applikationsname>/controller/batch/chunks/reader</p> <p>Zu den Chunks gehörende Klassen, die von javax.batch.api.chunk.AbstractItemReader abgeleitet sind.</p>
<p>src/main/java/de.bund.bvl.<Applikationsname>/controller/batch/chunks/processors</p> <p>Zu den Chunks gehörende Klassen, die javax.batch.api.chunk.ItemProcessor implementieren</p>
<p>src/main/java/de.bund.bvl.<Applikationsname>/controller/batch/chunks/writer</p> <p>Zu den Chunks gehörende Klassen, die von javax.batch.api.chunk.AbstractItemWriter abgeleitet sind.</p>
<p>src/main/java/de.bund.bvl.<Applikationsname>/model</p> <p>Implementierung der Modellschicht</p>
<p>src/main/java/de.bund.bvl.<Applikationsname>/model/comparators</p> <p>Komparatoren, die das Interface java.util.Comparator implementieren.</p>
<p>src/main/java/de.bund.bvl.<Applikationsname>/model/converters</p> <p>Konverter, die für das typsichere Binding von CSV- und XML-Dateien verwendet werden, z.B. via JSEFA¹³ oder JAXB.</p>
<p>src/main/java/de.bund.bvl.<Applikationsname>/model/entities</p> <p>Definitionen der Persistence Entities.</p>
<p>src/main/java/de.bund.bvl.<Applikationsname>/model/predicates</p> <p>Filterregeln für das Verarbeiten von Datenstrukturen, die beispielsweise java.util.Iterator implementieren.</p>
<p>src/main/java/de.bund.bvl.<Applikationsname>/model/repositories</p> <p>Zugriffsklassen, die die CRUD-Szenarien der Persistence Entities implementieren.</p>

¹³ Siehe hierzu auch <http://jsefa.sourceforge.net/>

 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Verzeichnis, Inhalt und Bedeutung
<p>src/main/java/de.bund.bvl.<Applikationsname>/model/validators</p> <p>Validatoren auf der Persistenzschicht, die javax.validation.ConstraintValidator implementieren, z.B. im Kontext der Bean Validation von Jakarta EE.</p>
<p>src/main/java/de.bund.bvl.<Applikationsname>/utils</p> <p>Hilfsklassen und technische Detailimplementierungen, die Ebenen übergreifend zur Anwendung kommen.</p>
<p>src/main/java/de.bund.bvl.<Applikationsname>/view</p> <p>Die von der Implementierung nach außen präsentierten Systemschnittstellen. Dies betrifft sowohl das UI als auch Webservice- und REST-Schnittstellen.</p>
<p>src/main/java/de.bund.bvl.<Applikationsname>/view/filter</p> <p>Gegebenenfalls benötigte Servlet-Filter der Webanwendung (implementieren javax.servlet.Filter). In Kombination mit der WAF kann beispielsweise ein HTTPS-Filter implementiert werden, der Anfragen automatisch auf eine verschlüsselte Verbindung umleitet. Die Konfiguration erfolgt in src/main/webapp/WEB-INF/web.xml.</p>
<p>src/main/java/de.bund.bvl.<Applikationsname>/view/handler</p> <p>Gegebenenfalls benötigte Handler der Webanwendung (erweitern javax.faces.context.ExceptionHandler). In Kombination von @ViewScoped Backing-Beans kann ein Handler implementiert werden, der das Logging von Stack-Traces bei abgelaufenen Sessions unterdrückt, beispielsweise bei javax.faces.application.ViewExpiredException. Die Konfiguration erfolgt in src/main/webapp/WEB-INF/faces-config.xml.</p>
<p>src/main/resources</p> <p>Ressourcen, die die Anwendung verwendet oder diese konfiguriert.</p>
<p>src/main/resources/templates</p> <p>Ablageort für Freemarker-Templates (FTL).</p>
<p>src/main/resources/META-INF</p> <p>In der Jakarta-EE-Spezifikation festgelegter technischer Ordner für Metainformationen des Deployments¹⁴, enthält bspw. Deskriptor persistence.xml</p>
<p>src/main/resources/META-INF/batch-jobs</p>

¹⁴ Bezüglich zulässiger Deskriptoren siehe auch

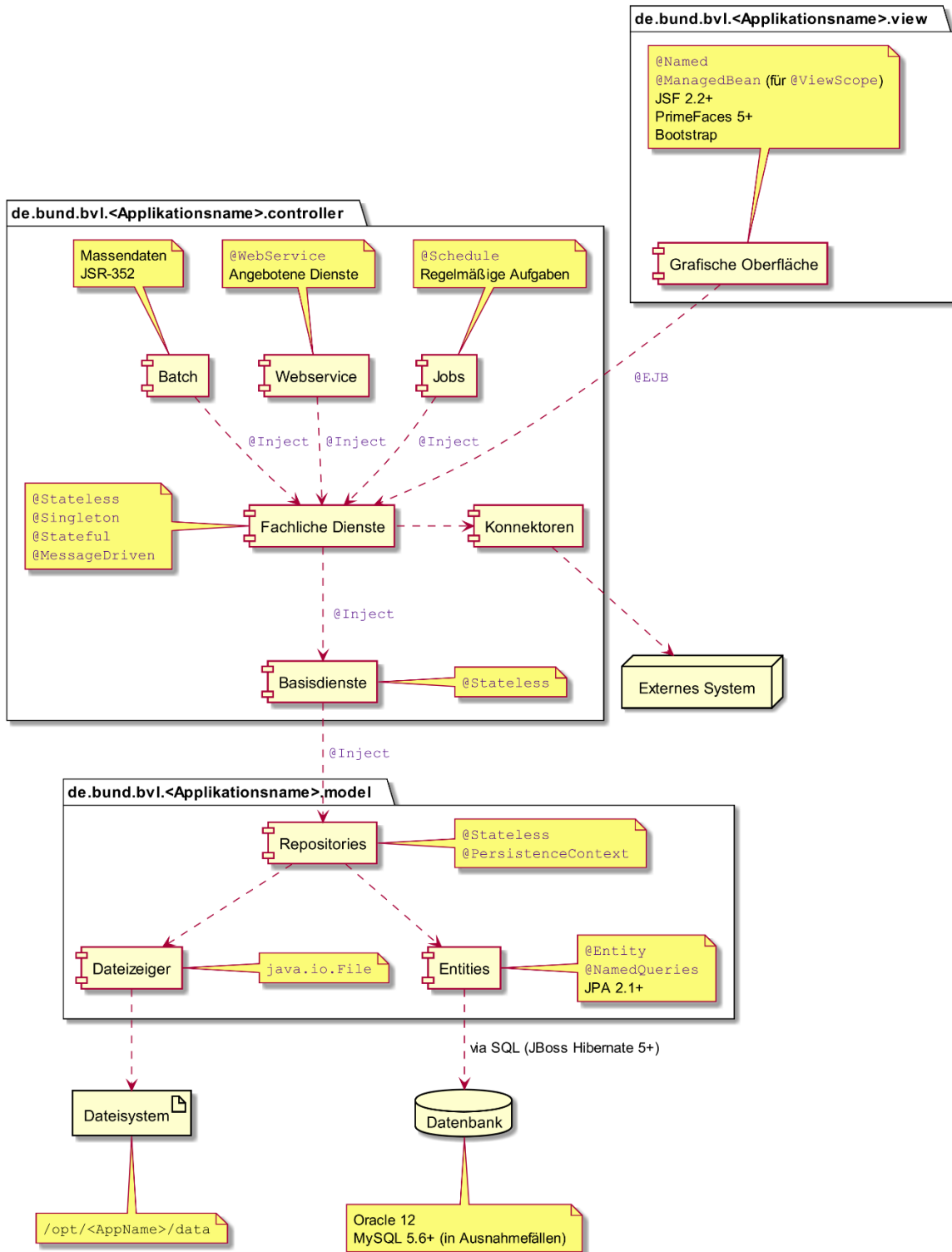
<https://docs.jboss.org/author/display/WFLY8/Deployment+Descriptors+used+In+WildFly>


 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Verzeichnis, Inhalt und Bedeutung
Ablageort für die XML-Deskriptoren der Arbeitsaufträge des Batch Processing.
src/main/scripts Ablageort für sämtliche SQL-Skripte (siehe Abschnitt 5.4).
src/main/webapp Zentrales Quellcodeverzeichnis der Webanwendung.
src/main/webapp/WEB-INF In der Servlet-Spezifikation festgelegter technischer Ordner für die Konfiguration der Webanwendung, enthält bspw. die Deskriptoren beans.xml , faces-config.xml und web.xml .
src/main/webapp/resources Ablageort der Ressourcen der Webanwendung.
src/main/webapp/resources/components Ablageort für selbstdefinierte JSF-Komponenten.
src/main/webapp/resources/templates Ablageort für selbstdefinierte JSF-Templates.
src/main/webapp/resources/css Ablageort für die Stylesheet-Dateien.
src/main/webapp/resources/js Ablageort für benötigte JavaScript-Dateien.
src/main/webapp/resources/images Ablageort für benötigte Bilddateien (JPG, PNG)
src/test Hauptverzeichnis der Testquellcodes.
src/doc Ausgabeverzeichnis für gerendertes JavaDoc und ergänzende Dokumentationsmaterialien

Anhang: Architekturkonzept

Abbildung 1 Mustervorlage für die Anwendungsarchitektur



 Bundesamt für Verbraucherschutz und Lebensmittelsicherheit	Standards zur Softwareentwicklung im BVL	
	BVL_TA_04_9306_000	Version 4.0

Entsprechend den Vorgaben für das Architekturmodell (siehe Abschnitt 5.3), den zu berücksichtigenden Spezifikationen und der Organisation des Quellcodes (siehe Abschnitt 6.2) veranschaulicht Abbildung 1 das empfohlene Grundgerüst für die Architektur der Anwendung.