

# Schrittweise zu einer GAMP-konformen Steuerungssoftware für die vollautomatisierte ATMP-Herstellung

Simon Hort<sup>1</sup>, Julius B. Mathews<sup>1</sup>, Prof. Dr.-Ing. Robert Schmitt<sup>1,2</sup> •

<sup>1</sup> Fraunhofer-Institut für Produktionstechnologie IPT, Aachen

<sup>2</sup> Werkzeugmaschinenlabor WZL der RWTH Aachen, Aachen

**Korrespondenz:** Simon Hort, Fraunhofer-Institut für Produktionstechnologie IPT, Steinbachstraße 17, 52074 Aachen; **E-Mail:** simon.hort@ipt.fraunhofer.de



## Zusammenfassung

Dieser Beitrag stellt ein Verfahren zur Good-Automated-Manufacturing-Practice(GAMP)-konformen Auslegung einer Steuerungssoftware für die vollautomatisierte Herstellung von Arzneimitteln für neuartige Therapien (ATMP) vor. Bei der Entwicklung eines solchen computerunterstützten Systems muss insbesondere Anhang 11 der Guten Herstellungspraxis (Good Manufacturing Practice, GMP) berücksichtigt werden. Aufbauend auf GMP Anhang 11 gibt die GAMP-5-Richtlinie spezifische Handlungsempfehlungen. Diese werden in diesem Beitrag auf den vorliegenden Anwendungsfall bezogen, wobei ein anwendungsorientierter, 17-schrittiger Prozess zur GAMP-konformen Softwareentwicklung vorgestellt wird, um den Zugang und die Umsetzung des GAMP 5 zu vereinfachen.

## Key Words

- ATMP
- CAR-T
- Automatisierung
- GAMP/GMP
- Zell- und Gentherapeutika

## Ausgangssituation

Fortschritte in der Biomedizin führen zur Entwicklung von Arzneimitteln für neuartige Therapien (ATMP). ATMP beschreibt unterschiedliche Gen- und Zelltherapeutika, die zukünftig für die Behandlung von Erb- und Krebserkrankungen eingesetzt werden können. Erste zugelassene Therapien (z. B. Kymriah für die Behandlung von Leukämie [1]) zeigen vielversprechende Erfolge [2].

Herausforderungen stellen sich aber aktuell noch in der komplexen Herstellung. Die manuelle Herstellung im Labor verursacht hohe Kosten und lange Herstellungszeiten. Um ATMP einer großen Anzahl an Patienten zur Verfügung zu stellen, bedarf es einer Automatisierung und Digitalisierung des gesamten

Prozesses. Durch die Automatisierung der manuellen, repetitiven Prozessschritte und der Verkettung von bereits bestehenden Geräten auf dem Markt kann die Produktion verbessert werden. Eine digitalisierte Produktion ermöglicht zusätzlich die systematische Aufnahme und Verarbeitung aller Daten sowie eine zentrale Steuerung und Überwachung des gesamten Prozesses. So können ATMP schneller, sicherer und in einer zuverlässigen Qualität hergestellt werden.

Aufgrund des schnell wachsenden Marktes der ATMP-Herstellung werden auch immer mehr Lösungen für die Automatisierung und Digitalisierung auf dem Markt angeboten. Dabei müssen Unternehmen eine Vielzahl an Regularien zum Schutz der Patienten beachten.

Die Umsetzung der Regularien gestaltet sich dabei nicht als trivial. Zuerst müssen die relevanten Dokumente identifiziert und analysiert werden, um dann auf die entsprechenden Lösungen angewendet zu werden. Welche Gesetze und Verordnungen für die Steuerungssoftware in der automatisierten ATMP-Produktion gelten, hängt davon ab, ob es sich bei dem betroffenen Produkt um ein Medizinprodukt handelt. Dafür ist neben den Definitionen in der europäischen Medizinprodukte-Verordnung (MP-VO) die Zweckbestimmung des Produktes entscheidend [3]. Weiterhin gibt es rechtlich bindende (Arzneimittelgesetz [4], Good-Manufacturing-Practice(GMP)-Leitfaden der EU [5]) und nicht bindende Regularien (Good-Automated-Manufacturing-Practice(GAMP)-Leitfaden

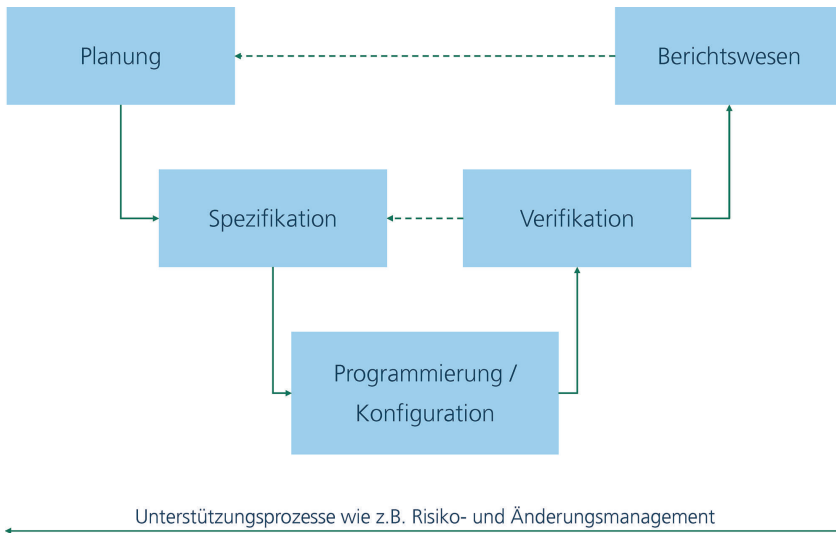


Abbildung 1: V-Modell der Projektphase des GAMP-5-Lebenszyklus (Quelle: Fraunhofer IPT nach [6]).

[6]). GAMP 5 zeigt, wie GMP-konform computergestützte System entwickelt, geprüft und betrieben werden können. Ein computergestütztes System ist das Zusammenspiel von Personal und Maschine zur Herstellung eines pharmazeutischen Produkts.

Am Fraunhofer-Institut für Produktionstechnologie (IPT) beschäftigt man sich seit mehr als 10 Jahren mit der vollautomatisierten ATMP-Herstellung. In Forschungsprojekten werden dabei Anlagen für die CAR-T-Zellproduktion [7] und die

Stammzellenkultivierung [8–10] entwickelt. Gesteuert werden diese Anlagen durch eine flexible Steuerungssoftware, die die einzelnen Geräte zentral steuert, den Prozess überwacht und die Daten verwaltet [9–12]. Im Folgenden wird ein schrittweiser Prozess zur GAMP-Umsetzung beschrieben, der im Rahmen der Entwicklung der Steuerungssoftware entstanden ist.

## GAMP-konforme Entwicklung der Steuerungssoftware

Das zentrale Ziel von GAMP 5 ist die Entwicklung, Prüfung und der Betrieb von computergestützten Systemen, sodass diese den GMP-Anforderungen in Bezug auf Patientensicherheit, Produktqualität und Datenintegrität genügen.

Der GAMP 5 basiert auf einem Lebenszyklusmodell, welches sich in Konzept-, Projekt-, Betriebs- und Stilllegungsphase unterteilen lässt. Die Konzeptphase ist Teil der inter-

## Autoren



Simon Hort

Simon Hort studierte im Bachelor Wirtschaftsingenieurwesen Fachrichtung Maschinenbau mit Schwerpunkt Produktionstechnik an der RWTH Aachen. Im Master studierte er Wirtschaftsingenieurwesen am KIT in Karlsruhe mit Schwerpunkt Digitalisierung in der Produktion. Seit 2019 ist er wissenschaftlicher Mitarbeiter am Fraunhofer-Institut für Produktionstechnologie IPT im Bereich Produktionsmesstechnik. Seit 2021 leitet er dort die Gruppe Adaptive Produktionssteuerung.



Julius B. Mathews

Julius B. Mathews studierte Wirtschaftsingenieurwesen Fachrichtung Maschinenbau mit Schwerpunkt Produktionstechnik an der RWTH Aachen. Seit 2021 ist er wissenschaftlicher Mitarbeiter am Fraunhofer-Institut für Produktionstechnologie IPT in der Abteilung für Produktionsmesstechnik mit Arbeitsschwerpunkt Adaptive Produktionssteuerung.



Prof. Dr.-Ing. Robert Schmitt

Prof. Dr.-Ing. Robert Schmitt war nach dem Studium der Elektrischen Nachrichtentechnik an der RWTH Aachen wissenschaftlicher Mitarbeiter am Lehrstuhl für Fertigungsmesstechnik und Qualitätsmanagement. Sein Arbeitsschwerpunkt lag im Bereich der fertigungsnahen Mess- und Kommunikationstechnik im automatisierten Umfeld. 1997 wechselte er zur MAN Nutzfahrzeuge AG in München, wo er leitende Positionen im Qualitätsbereich und in der Produktion innehatte. Ab 2002 war er für die LKW-Produktion am Standort Steyr (A) verantwortlich. Zum 1. Juli 2004 wurde er als Professor an die RWTH Aachen berufen. Er ist als Inhaber des Lehrstuhls für Fertigungsmesstechnik und Qualitätsmanagement, Mitglied des Direktoriums des Werkzeugmaschinenlabors WZL und des Fraunhofer-Instituts für Produktionstechnologie IPT.

nen Regelungen des Unternehmens. Die Projektphase orientiert sich an dem V-Modell der Softwareentwicklung (Abb. 1). Die Stufen des V-Modells nach GAMP 5 unterscheiden sich in Planung, Spezifikation, Konfiguration, Programmierung, Verifikation sowie Berichtswesen und Freigabe. Die Betriebsphase inkludiert u. a. den Übergabeprozess an den Kunden, Störfallmanagement sowie Änderungsmanagement. Elemente der Stilllegungsphase sind

Rücknahme, Außerbetriebnahme, Entsorgung sowie Datenmigration. Da der Fokus dieses Beitrags auf der Softwareentwicklung liegt, wird im Folgenden die Projektphase weitergehend erläutert. [6]

Der notwendige Ablauf für die Umsetzung befindet sich aufgrund der Skalierbarkeit des GAMP-Ansatzes nicht zentral an einer Stelle im Leitfaden, sondern ist über den Leitfaden dezentral verteilt. Deswegen können konkrete schrittweise Pro-

zesse nicht direkt aus dem GAMP 5 abgeleitet werden, weshalb die einzelnen Tätigkeiten zunächst zu einem Gesamtprozess zusammengefügt werden müssen. Die Anzahl der Spezifikations- und Verifikationslevel hängt von der Neuheit, Komplexität und dem Risiko des eingesetzten Systems ab.

Das Ziel dieses risikoorientierten Ansatzes ist, die Menge der Tests entsprechend dem Risiko für die Produktqualität zu skalieren. Die

**Tabelle 1**

**Schrittweise zu einer GAMP-konformen Steuerungssoftware für die vollautomatisierte ATMP-Herstellung.**

	Nr.	Prozessschritt	Dokumentation
Planung	1	Validierungsrichtlinie verfassen	[6] S. 81–82
		Validierungsmasterplan verfassen	[6] S. 82–83
		Validierungsplan für computergestützte Systeme durchführen	[6] S. 83–87
	2	Rollen auf Prozessbeteiligte verteilen und Verantwortlichkeiten übertragen	[6] S. 58–61, S. 83–86, S. 106, S. 196
	3	Teststrategie verfassen	[6] S. 196–197 und [13] S. 61
	4	Initiale Risikobewertung durchführen	[6] S. 110–112
GAMP-Kategorien auswählen		[6] S. 127–132	
Lieferantenbewertung durchführen		[6] S. 89–103	
Spezifikation	5	User Requirement Specification (URS) definieren	Anhang D1 des GAMP-Leitfadens in [6] S. 163–173
		Testspezifikationen für Anforderungstest aus URS ableiten und verfassen	[6] S. 199–200 und [13] S. 91–92
	6	Funktionsspezifikation verfassen	[6] S. 175–178
		Design Review der Funktionsspezifikation durchführen	[6] S. 133–137
		Testspezifikation für Funktionstest verfassen	[6] S. 199–200 [13] S. 91–92
		Funktionale Risikobewertung durchführen	[6] S. 113
	7	Designspezifikation verfassen	[6] S. 179–185
		Design Review der Designspezifikation durchführen	[6] S. 133–137
		Testspezifikation für den Integrationstest verfassen	[6] S. 199–200 [13] S. 91–92
		Funktionale Risikobewertung aufgrund Designspezifikation ergänzen	[6] S. 113
	8	Modulspezifikation verfassen	Module bzw. Moduleinheiten werden in der Designspezifikation definiert.
		Design Review der Modulspezifikation durchführen	Ein separates Design Review ist nicht nötig.
		Testspezifikation für Modultest verfassen	[6] S. 199–200 [13] S. 91–92
		Funktionale Risikobewertung aufgrund Modulspezifikation ergänzen	Die funktionale Risikobewertung der Module bzw. Moduleinheiten findet in Teilprozess-Schritt Nr. 7 statt, wenn die funktionale Risikobewertung aufgrund der Designspezifikation ergänzt wird.

Zur Verwendung mit freundlicher Genehmigung des Verlages / For use with permission of the publisher

**Tabelle 1 – Fortsetzung**

	Nr.	Prozessschritt	Dokumentation
Programmierung/ Konfigurierung	9	Software programmieren bzw. konfigurieren	[6] S. 187–192
	10	Prüfung des Quellcodes	[6] S. 187–194
	11	Funktionale Risikobewertung aufgrund Programmierung bzw. Konfigurierung erweitern	[6] S. 113
Verifikation	12	Testskript und Testfälle für Modul(einheit)test verfassen	[6] S. 199–200
		Testrealisierung der Modultests durchführen	[14] S. 39–40 und [13] S. 115
		Modul(einheit)tests durchführen	[13] S. 117–118
		Testergebnisse der Modultests dokumentieren	[6] S. 200
		Testbericht der Modultests verfassen	[6] S. 200
	13	Testskript und Testfälle für Integrationstest verfassen	[6] S. 199–200
		Testrealisierung der Integrationstests durchführen	[14] S. 39–40 und [13] S. 115
		Integrationstests durchführen	[13] S. 117–118
		Testergebnisse der Integrationstests dokumentieren	[6] S. 200
		Testbericht der Integrationstests verfassen	[6] S. 200
	14	Testskript und Testfälle für Funktionstests verfassen	[6] S. 199–200
		Testrealisierung der Funktionstests durchführen	[14] S. 39–40 [13] S. 115
		Funktionstests durchführen	[13] S. 117–118
		Testergebnisse der Funktionstests dokumentieren	[6] S. 200
		Testbericht der Funktionstests verfassen	[6] S. 200
	15	Testskript und Testfälle für Anforderungstests verfassen	[6] S. 199–200
		Testrealisierung der Anforderungstests durchführen	[14] S. 39–40 und [13] S. 115
		Anforderungstests durchführen	[13] S. 117–118
		Testergebnisse der Anforderungstests dokumentieren	[6] S. 200
		Testbericht der Anforderungstests verfassen	[6] S. 200
Berichtswesen	16	Testzusammenfassungsbericht aus allen Testberichten erstellen	[6] S. 200
	17	Validierungsbericht verfassen	[6] S. 145–147

Zur Verwendung mit freundlicher Genehmigung des Verlages / For use with permission of the publisher

aus dem GAMP 5 extrahierten Kriterien werden in eine für den betrachteten Anwendungsfall sinnvolle Reihenfolge gebracht und als Schrittfolge von Prozessen dargestellt. In Tab.1 wird der Gesamtprozess anhand des V-Modells aufgebaut und in 17 Schritte gegliedert. Zu jedem Schritt werden die Seitenzahlen aus dem GAMP 5 und weiterführender Literatur [13, 14] für die Umsetzung angegeben.

### Teilprozess Planung

Zu Beginn des Projekts zur Qualifizierung des computergestützten Systems wird die Organisation und Planung des Projektes in der Form eines Validierungsmasterplans festgehalten. Dieser gibt einen Überblick über den Anwendungsbereich der vollautomatisierten ATMP-Anlage, den Qualifizierungsablauf im Unternehmen, Hauptakzeptanzkriterien, Verantwortlichkeiten und Ver-

antwortliche und einen zeitlichen Projektablauf. Die Beschreibung der Verantwortlichkeiten der einzelnen GAMP-Rollen ist im GAMP-Leitfaden klar vorgenommen, auch wenn die Informationen dezentral im Dokument verteilt sind. In manchen Fällen ist es zulässig, dass ein Prozessbeteiligter mehrere Rollen innerhalb des Validierungsprozesses innehat. So können z. B. System Owner und Process Owner dieselbe Per-

son sein, wohingegen die Person Software Autor kein Tester sein darf.

Im Anschluss muss eine initiale Risikobewertung für die Steuerungssoftware inklusive der ATMP-Anlage durchgeführt werden. Es muss zuerst festgelegt werden, welche Geschäftsprozesse des computergestützten Systems Auswirkungen auf kritische Qualitäts- und Prozessparameter haben und ein Risiko darstellen. Je höher die Auftrittswahrscheinlichkeit und Auswirkungen auf kritische Parameter sind, desto stärker muss die Funktion später getestet werden.

Teil der initialen Risikobewertung ist die Festlegung einer Kategorie für die verwendete Software. Hierbei reicht das Spektrum von „Kategorie 1: Infrastruktursoftware“ bis „Kategorie 5: Kundenspezifische Anwendungen“. Letztere hat aufgrund eines höheren Grads an Neuheit und Individualität ein inhärent höheres Risiko. Üblicherweise wird eine Steuerungssoftware in den Kategorien 4–5 eingeordnet. Dies ist hängt davon ab, ob die Software nur konfiguriert werden muss oder tatsächliche Änderungen am Quellcode notwendig sind.

### Teilprozess Spezifikation

Wenn die Planung abgeschlossen ist, werden die Spezifikationen erstellt. Bei der GAMP-Software-Kategorie 5 sind 4 Spezifikationsebenen vorgesehen:

- Benutzeranforderung (URS),
- Funktionsspezifikationen (FS),
- Designspezifikation (DS) und
- Modulspezifikationen.

Diese erfolgen aus unterschiedlichen Perspektiven und mit zunehmendem Detaillierungsgrad. In der URS werden vergleichbar mit einem Lastenheft die Anforderungen des Anlagennutzers an das computergestützte System zur Sicherstellung der Produktqualität spezifiziert. Im Anschluss wird die Umsetzung der beschriebenen Nutzeranforderungen in der FS, DS und Modulspezifikation festgehalten. Zusammengekommen sind diese 3 Dokumente

vergleichbar mit einem Pflichtenheft.

Hierbei fokussiert sich die FS auf das funktionale Design, also die Beschreibung der Haupt- und Unterkomponenten und der System- und Unterfunktionen. In vertiefender Weise beschreibt die DS dagegen das Systemdesign. Hier geht es u. a. um die Dokumentation der gewählten System- und Applikationssoftware und die Hardwarekonfiguration des Systems, also z. B. welche Geräte zur Zellkultivierung verbaut werden. Aus diesem Grund wird die DS häufig in die 2 Dokumente Hardwaredesignspezifikation (HDS) und Softwaredesignspezifikation (SDS) aufgeteilt. Abschließend beschreibt die *Modulspezifikation* die Umsetzung des Systemdesigns auf höchster Detailebene. Hierfür ist die Definition der Schnittstellen und des Programmablaufs einer speicherprogrammierbaren Steuerung (SPS) ein Beispiel. Aufgrund des risikobasierten Ansatzes muss aber nicht zwangsweise für jedes Modul eine Modulspezifikation erstellt werden, sondern nur bei entsprechend hohem Risiko.

Als Abschluss der Spezifikation erfolgt die Designqualifizierung (DQ). Das Hauptziel dieser Entwurfsprüfung ist es, festzustellen, ob alle Anforderungen der URS in den 3 anderen Dokumenten der Spezifikation umgesetzt wurden. Als Best Practice hat sich die Erstellung einer Checkliste mit allen Anforderungen und der systematische Abgleich mit den Spezifikationen herausgestellt. Falls es nicht erfüllte Anforderungen gibt, muss die entsprechende Spezifikation ergänzt werden.

Darüber hinaus ist es sinnvoll, parallel zur Erstellung der Spezifikationen die funktionale Risikobewertung durchzuführen und die entsprechenden Testspezifikationen zu verfassen. Das Ziel der Risikobewertung ist die Identifikation und Bewertung von Fehlermöglichkeiten auf funktionaler Ebene. So sollen Maßnahmen zur Risikoreduktion bereits in der Entwicklungsphase

Berücksichtigung finden. Falls Fehlermöglichkeiten im Hardware- oder Software-Design nicht vermeidbar sind, müssen diese Systemfunktionen mit höchster Detailtiefe (also in einer Modulspezifikation) dokumentiert werden und die Menge an Code-Prüfungen und Tests muss entsprechend umfangreich sein. Bei der Testspezifikation ist an dieser Stelle nicht das Ziel, bereits konkrete Testfälle oder -skripte aufzustellen, sondern viel mehr den grundlegenden Aufbau von Tests und Testprotokollen festzulegen. Dieses Vorgehen soll sicherstellen, dass die Implementierung der Nutzeranforderungen grundsätzlich testbar ist.

### Teilprozess Programmierung/ Konfigurierung

Nach der Spezifikation wird die Software programmiert bzw. konfiguriert. Die im vorherigen Schritt verfassten Dokumente bilden den Anforderungskatalog an die Entwicklung bzw. Anpassung neuer Softwarefunktionalitäten. Dies stellt sicher, dass die Software auf einem gut strukturierten, dokumentierten und geprüften Designentwurf basiert. Grundsätzlich empfiehlt GAMP 5 für die Entwicklung von Software die Einhaltung anerkannter Programmierverfahren. Diese werden zwar nicht genauer benannt, allerdings sind damit üblicherweise die folgenden organisatorischen und operationalen Aspekte des Entwicklungsprozesses gemeint: Softwareentwicklung (agil, z. B. SCRUM, oder klassisch, z. B. Wasserfallmodell), Verwendung einer Versionskontrolle (z. B. Git), Software-Designprinzipien (z. B. Abstraktion, Datenkapselung), Styleguides und Dokumentationsrichtlinien.

Nach der Entwicklung des Quellcodes wird die Software anhand der festgelegten Programmierrichtlinien und Styleguides geprüft und Abweichungen werden korrigiert. Abschließend wird die Risikobewertung aufgrund der vorgenommenen Programmierung bzw. Konfigurierung ergänzt.

## Teilprozess Verifikation

Sind diese Prozessschritte abgeschlossen, kann mit der Verifikation der Software bzw. des computergestützten Systems begonnen werden. Entsprechend der Anzahl der Spezifikationen werden Teststufen durchgeführt. Jede Verifikation besteht aus 5 identischen Schritten:

1. Testskript und Testfälle verfassen
2. Testrealisierung durchführen
3. Tests durchführen
4. Testergebnisse dokumentieren
5. Testbericht verfassen

GAMP 5 unterscheidet dabei 4 verschiedene Teststufen: Modultests, Integrationstests, Funktionstests und Anforderungstests. Bei den Modultests handelt es sich um reine Softwaretests. Diese stellen sicher, dass Programmanweisungen ihre beabsichtigte Funktion erfüllen. Typische Inhalte sind automatisierte Unit Tests, Code Reviews und die Prüfung der Entscheidungs- und Pfadabdeckung insbesondere für Softwaremodule mit erhöhtem Risiko.

Die restlichen Tests betreffen das Zusammenspiel aus Software und Hardware. Als erstes stellen die Integrationstests sicher, dass die Installation von Software und Hardware dem Design entspricht. Dann wird durch Funktions- und Anforderungstests überprüft, ob alle spezifizierten Funktionen wie beabsichtigt funktionieren und alle Anforderungen umgesetzt sind.

## Teilprozess Berichtswesen

Im letzten Teilschritt werden ein Testzusammenfassungsbericht und ein Validierungsbericht erstellt.

Zusätzlich zu den vorgestellten Kernprozessen empfiehlt GAMP 5 die Durchführung von Unterstützungsprozessen, wie etwa Risikomanagement, Änderungs- und Konfigurationsmanagement, Dokumentenmanagement und Rückverfolgbarkeit.

## Schlussfolgerung und Ausblick

Durch die fortschreitende Forschung im Bereich ATMP bieten sich neue Möglichkeiten für IT- und Softwareunternehmen zur Entwicklung neuer digitaler Produkte. Für den Einsatz dieser Produkte im hochreglementierten Medizinbereich muss eine Reihe von Regularien und Richtlinien beachtet werden. Der in diesem Beitrag vorgestellte Prozess zur GAMP-konformen Entwicklung einer Steuerungssoftware soll den Zugang zum GAMP 5 erleichtern und die Softwareentwicklung in diesem Bereich fördern.

## Literatur

- [1] Kymriah: tisagenlecleucel. Online verfügbar unter <https://www.ema.europa.eu/en/medicines/human/EPAR/kymriah>
- [2] Schuster, Stephen J.; Bishop, Michael R.; Tam, Constantine S.; Waller, Edmund K.; Borchmann, Peter; McGuirk, Joseph P. et al. (2019): Tisagenlecleucel in Adult Relapsed or Refractory Diffuse Large B-Cell Lymphoma. In: *The New England journal of medicine* 380 (1), S. 45–56. DOI: 10.1056/NEJMoal804980
- [3] Europäische Union (05.04.2017): Verordnung (EU) 2017/745 des Europäischen Parlaments und des Rates vom 5. Apr. 2017 über Medizinprodukte, zur Änderung der Richtlinie 2001/83/EG, der Verordnung (EG) Nr. 178/2002 und der Verordnung (EG) Nr. 1223/2009 und zur Aufhebung der Richtlinien 90/385/EWG und 93/42/EWG des Rates (Text von Bedeutung für den EWR.), vom 05.05.2017. Fundstelle: Amtsblatt der europäischen Union. Online verfügbar unter <https://eur-lex.europa.eu/legal-content/DE/TXT/?uri=CELEX%3A32017R0745>
- [4] Bundesministerium der Justiz: Gesetz über den Verkehr mit Arzneimitteln. Online verfügbar unter [https://www.gesetze-im-internet.de/amg\\_1976/](https://www.gesetze-im-internet.de/amg_1976/)
- [5] European Commission (o. J.): EudraLex – Volume 4 – Good Manufacturing Practice (GMP) guidelines. European Union. o. O. Online verfügbar unter [https://ec.europa.eu/health/documents/eudralex/vol-4\\_en](https://ec.europa.eu/health/documents/eudralex/vol-4_en)
- [6] ISPE (2008): ISPE GAMP 5: A Risk-Based Approach to Compliant GxP Computerized Systems. Online verfügbar unter <https://ispe.org/publications/guidance-documents/gamp-5>
- [7] Hort, Simon; Herbst, Laura; Bäcker, Niklas; Erkens, Frederik; Niessing, Bastian; Frye, Maik et al. (2022): Toward Rapid, Widely Available Autologous CAR-T Cell Therapy – Artificial Intelligence and Automation Enabling the Smart Manufacturing Hospital. In: *Front. Med.* 9. DOI: 10.3389/fmed.2022.913287
- [8] Doulgkeroglou, M.-N.; Di Nubila, A.; Niessing, B.; König, N.; Schmitt, R.H.; Damen, J.; Szilvassy, S.J.; Chang, W.; Csonotos, L.; Louis, S.; et al. Automation, Monitoring, and Standardization of Cell Product Manufacturing. *Front. Bioeng. Biotechnol.* 2020, 8, 811, doi:10.3389/fbioe.2020.00811
- [9] Egri P, Csáji BC, Kis KB, Monostori L, Vánca J, Ochs J et al. Bio-inspired control of automated stem cell production. *Procedia CIRP* 2020; 88(1680):600–5. DOI: 10.1016/j.procir.2020.05.105
- [10] Jung, Sven; Ochs, Jelena; Kulik, Michael; König, Niels; Schmitt, Robert H. (2018): Highly modular and generic control software for adaptive cell processing on automated production platforms. In: *Procedia CIRP* 72, S. 1245–1250. DOI: 10.1016/j.procir.2018.03.189. DOI: 10.1007/978-3-662-56714-2\_25
- [11] Jung, Sven; Grunert, Dennis; Schmitt, Robert (2018): Service-oriented Communication and Control System Architecture for Dynamically Interconnected Assembly Systems. In: Thorsten Schüppstuhl, Kirsten Tracht und Jörg Franke (Hrsg.): *Tagungsband des 3. Kongresses Montage Handhabung Industrieroboter*. Berlin, Heidelberg: Springer Berlin Heidelberg, S. 223–229. DOI: 10.1016/j.procir.2018.03.189
- [12] Hort, Simon; Nießing, Bastian; Jung, Sven; Schmitt, Robert (2020): Smarte Steuerung flexibler Produktionssysteme. In: *wt Werkstattstechnik online* 110 (11–12), S. 768–771. Online verfügbar unter <https://publica.fraunhofer.de/entities/publication/3a811a43-ef3e-47c0-a5c9-d07208b5889d/details>. DOI: 10.37544/1436-4980-2020-11-12-32
- [13] ISPE (2012): GAMP Good Practice Guide. A Risk-Based Approach to Testing of GxP Systems. 2. Aufl. Online verfügbar unter <https://ispe.org/products/gamp-good-practice-guide-risk-based-approach-testing-gxp-systems-german-edition-individual>
- [14] Spillner, Andreas; Linz, Tilo (2019): *Basiswissen Softwaretest. Aus- und Weiterbildung zum Certified Tester: Foundation Level nach ISTQB-Standard. 6., überarbeitete und aktualisierte Auflage*. ISBN: 978-3-86490-583-4

Alle Links wurden zuletzt am 13.10.2022 geprüft.