Translating Data Protection into Software Requirements

Ralf Kneuper[a]

IUBH University of Applied Sciences—Distance Learning, Kaiserplatz 1, 83435 Bad Reichenhall, Germany

Keywords: Data Protection, Privacy, GDPR, Software Requirements.

Abstract: With the growth of data processing and digitalisation in many environments, data protection is also growing more and more important. This is for example reflected by the General Data Protection Regulation (GDPR) which came into effect in May 2018 and defines what organisations need to do to protect individuals and their personal data. This paper provides a summary of the main data protection concepts, using GDPR as an example, and from these derives the resulting software requirements that apply to software systems which process private data within the European Union (and to some extent beyond). This way, the paper supports software developers as well as requirements analysts in their task of identifying and defining the data protection requirements, even though they will have to be adapted and additional detail provided for any specific case.

1 INTRODUCTION

Data protection, i.e. the protection of individuals against inadequate and unwanted use of their personal data, is growing increasingly important, and needs to be taken into account in software development where many decisions are taken about how (personal) data are processed. In the European Union, the expectations about data protection are mostly defined in the General Data Protection Regulation (GDPR), which became applicable law in May 2018. The main goal of this regulation is to define and ensure a largely uniform level of protection across the European Union. Similar laws and regulations apply in many other countries.

The GDPR contains a set of demands on organisations and other entities that process any form of personal data, leading to organisational as well as technical measures that need to be taken. Many of these demands lead to requirements on any software used to process personal information. However, GDPR does not express these demands as software requirements. Instead, the software requirements need to be derived from the general demands on the handling of personal data, which can be a challenging task for software developers without relevant legal training.

Therefore, the goal of the current paper is to identify the requirements resulting from GDPR (r similar laws and regulations) that apply to software development, and to express them as software requirements. These software requirements are to be expressed as specific as possible, accepting that this will only be possible to a limited extent since they apply to very different software systems, with very different personal data processed.

Note that the requirements described here refer to the resulting product and therefore are independent of the life cycle model and the general approach (plan-driven, agile or hybrid) used.

Although the GDPR is used in this paper as the reference model for data protection, many of the demands as well as the resulting software requirements are not specific to GDPR but can be found in a similar form in most other data protection laws.

To better explain the meaning of the software requirements identified and provide some ideas about their implementation, the example of an online shop will be used throughout this paper.

Structure of Paper. This paper is structured as follows: after the current introduction, Sect. 2 introduces the terminology and main concepts used. Next, Sect. 3 provides an overview of the main principles of data protection as defined by GDPR, and how these principles can be translated into software requirements. A brief summary of the rights of the data subjects is given in Sect. 4. Some additional concepts of data protection and GDPR relevant in this context are discussed in Sect. 5. Finally, Sect.6 gives an overview of the validation performed on the results, and Sect. 7 summarises the main conclusions of this work.

In this paper, we will not even try to give a com-

[a] https://orcid.org/0000-0003-3225-5895

Kneuper, R.
Translating Data Protection into Software Requirements.
DOI: 10.5220/0008873902570264
In *Proceedings of the 6th International Conference on Information Systems Security and Privacy (ICISSP 2020)*, pages 257-264
ISBN: 978-989-758-399-5; ISSN: 2184-4356

Copyright © 2022 by SCITEPRESS – Science and Technology Publications, Lda. All rights reserved

plete introduction to data protection and GDPR, but focus on those aspects that are more or less directly relevant for software development. Other important topics such as the nomination and role of the data protection officer or the role of the supervisory bodies will not be covered here. Also, the current paper does not address the implications of data protection on the software processes. This topic was for example discussed in (Kneuper, 2019).

2 TERMINOLOGY AND BASIC CONCEPTS

2.1 Data Protection

Before starting with the identification of requirements, we first need to define the main terminology used. Since the goal of data protection is the protection of individuals, data protection only refers to *personal data*, defined as "any information relating to an identified or identifiable natural person ('data subject')" (Art. 4(1) GDPR). (In other contexts, for example ISO/IEC 29100:2011, the name "personally identifying information" (PII) is used to describe the same concept.) Therefore, when talking about "data" in the following, this refers to personal data.

However, the exact interpretation of this term varies with the applicable legislation. In the case of GDPR, personal data includes data where the reference to a particular person may not be immediately visible, as for example an IP address.

In contrast to personal data, anonymous data are defined as data that do not refer to identifiable individuals, and therefore do not count as personal data. Anonymous data need to be distinguished from pseudonymous data which do refer to identifiable individuals but where this identification needs additional information which is stored separately and protected. Pseudonyms may therefore be used as a form of protecting personal data.

In times of big data and data analysis, it is important to take into account that there are many cases where seemingly anonymous data turn out to not be anonymous at all, for example because they can be related (joined) to some other data set via common (pseudo-)identifiers or keys.

A special case of personal data are the *special categories of personal data* which need particular protection, as defined in Art. 9 GDPR. These include "personal data revealing racial or ethnic origin, political opinions, religious or philosophical beliefs, or trade union membership, and [. . .] genetic data, biometric data for the purpose of uniquely identifying a natural person, data concerning health or data concerning a natural person's sex life or sexual orientation" (Art. 9(1) GDPR).

Following GDPR, this paper will count as "processing" any form of handling data, including (but not limited to) the collection, storage, reading and editing of data as well as their transfer to other systems or organisations. A wording like "collecting and processing of data" will occasionally be used to emphasise the collection of data but keeping in mind that collection is actually part of processing.

Roles Involved. In GDPR and data protection in general, one usually distinguishes three main roles:

- The *data subject* is the (natural) person[1] whose data are processed and who needs adequate protection. This may include (individual) customers, employees, visitors to a company website, and many other individuals. In other frameworks, for example in ISO/IEC 29100:2011, this role is also called the "PII principal".
- The *controller* is the entity that "determines the purposes and means of the processing of personal data" (Art. 4 (7) GDPR) within an organisation, and therefore is fully responsible for this processing. The controller may either perform the processing itself, or subcontract it to a separate processor.
- The *processor* is the entity that performs the actual processing of data, on behalf of and following the rules set by the controller. If processing is performed by the controller itself, there is no processor. If a processor exists, GDPR requires a contract between the controller and the processor to ensure that the controller does actually control the processing, and the processor follows the rules set by the controller.

A fourth role, which is not described in GDPR but important in the current context, is the software producer, which may overlap with one of the roles mentioned or be a completely different entity.

2.2 Requirements

In the following, three different groups of requirements will be distinguished following IREB (International Requirements Engineering Board (IREB), 2017; Pohl and Rupp, 2015): functional requirements (marked with identifiers F-*n*) define the functions to

[1] In GDPR, like in most legislations, data protection only applies to *natural* persons. However, in some countries such as Switzerland, data protection also applies to so-called legal persons such as companies.

be provided by the system under development. Quality requirements (identifiers Q-*n*), also known as non-functional requirements, describe the required quality properties of the system, such as performance or usability. Constraints (identifiers C-*n*) define requirements on the environment within which the system is to be developed rather than the system itself, for example the use of a certain software process life cycle model or certain documentation to be created. Some constraints do not apply to software and the development organisation at all but to the controller, for example defining the purpose of the processing performed. Nevertheless, the development organisation should in most cases be involved, either because they need information such as the purpose as input, or because they need to support the controller.

To support precise formulation of the functional requirements, they follow the template described by (Pohl and Rupp, 2015), which in a simple variant states "*The system shall/should/– provide <actor> with the ability to/be able to <process verb>*". However, following this template is not always possible, for example when the requirement describes what may *not* happen.

Note the difference between requirements expressed using "shall" and "should". If requirements need to be satisfied in order to conform to GDPR, "shall" is used. If requirements are considered very helpful but there are alternative ways to conform to GDPR, "should" is used.

3 DATA PROTECTION PRINCIPLES

This section discusses the basic principles of data protection as defined in Art. 5 GDPR, together with the resulting software requirements.

3.1 Lawfulness, Fairness and Transparency

The principles of lawfulness, fairness and transparency are defined in Art. 5(1)a GDPR. The principle of lawfulness starts from the rule that processing of personal data is prohibited unless there is an explicitly defined legal basis that allows it. In Art. 6(1), GDPR defines a list of six conditions which may form such a legal basis. In practice, the most relevant of these lawfulness conditions are a) consent by the data subject, b) processing for the performance of a contract to which the data subject is party, and f) the legitimate interest of the controller or a third party, where this interest must be balanced against the interests of the data subjects. Note that within any one system, different types of processing of the same data may take place in parallel but with different legal bases. For example, performing a contract may form a legal basis for processing the address of a customer, while using the same address for marketing purposes may require the consent of the customer. This leads to the following software requirement:

Requirement (C-1: Identify Legal Basis)**.** *Before performing any processing of personal data, the relevant legal basis according to Art. 6 GDPR shall be identified and documented.*

In general, the appropriate place for this documentation will be the *records of processing activities* as introduced in Sect. 5.1 below.

When processing data on the basis of consent, the following properties of valid consent must be taken into account: consent must be *given freely*, implying that data subjects that do not consent may not be put at a disadvantage, for example by not giving them access to information on a website where this consent is not factually necessary.

Since genuine consent is given freely, it must also be possible to change one's mind and *withdraw consent* (at least for the future). Furthermore, the consent given must be *specific*, referring to a specific kind of processing and purpose. It must be *informed*, since data subjects need to know and understand what their consent infers. Consent must be *unambiguous*, which implies for example that users must actively select a consent box, while a pre-ticked box is not sufficient. Consent must also be collected in *adequate granularity*, for example allowing customers to consent to the use of their address for sending an annual catalogue but not a weekly newsletter.

These demands on consent lead to the following software requirements:

Requirement (F-1 Freely given Consent)**.** *The system shall provide users with access to data and functionality even if they did not provide consent, unless this consent is factually necessary.*

An example of factually necessary consent regards the storage of user preferences for a system. If users do not consent to this storage, then of course they will have to set relevant preferences from scratch every time they use the system.

Requirement (F-2 Collecting Consent)**.** *If the processing of personal data is based on consent, the system shall collect consent* before *the start of processing. Before collecting consent, the system shall inform data subjects about the purpose of the processing and the implications of giving or refusing consent. The* *system shall collect consent on an adequate level of granularity. The system shall only collect consent that is given actively and unambigously.*

Requirement (F-3 Consent for Children)**.** *If consent is to be given for a child and refers to an "offer of information society services directly to a child", the system shall ensure that consent is given by the holder of parental responsibility. (Art. 8 GDPR)*

Note that the exact definition of "child" varies between legislations, even within the EU.

Requirement (F-4 Withdrawal of Consent)**.** *The system shall provide the data subject with the ability to withdraw consent previously given, and to give consent previously refused.*

Requirement (F-5 Traceability of Consent)**.** *The system shall provide the controller with the ability to trace consent given, refused or withdrawn, including information about when and in what form this was done.*

Requirement (F-6 Storage of Consent)**.** *The system should provide the data subject with the ability to store consent given or refused beyond the individual session.*

The following requirement seems obvious, but can become very complex, particular if consent—as required—is collected in a granular form, and then is partly given, partly refused, and maybe even changed over time.

Requirement (F-7 Compliance to Consent)**.** *The system shall comply to the consent given or refused in its functionality, always based on the current status of consent.*

Coming back to the example of an online shop, consent may for example be collected for sending of marketing material such as special offers. Any system function for sending marketing material, for example in the customer relationship management (CRM) system, will then have to start by checking whether consent for this type of processing is available.

To implement this last requirement, it is therefore sometimes recommended to introduce an architecture layer where relevant processing requests are checked to verify that consent has been given, similar to and possibly in combination with checking user authorisations.

The other legal bases for processing of personal data do not lead to any specific software requirements beyond those of requirement C-1 above.

Fairness and Transparency. The principle of fairness is rather vague and therefore results in the following, also rather vague, software requirement:

Requirement (F-8 No Surprising and Unexpected Processing)**.** *The system shall not perform any processing of personal data that is surprising or unexpected to the data subject. It shall not perform any processing that unfairly exploits the imbalance of power between controller and data subject.*

Finally, the principle of transparency aims at making sure that data subjects know what data about them are processed and how. Although the principle itself is rather vague as well, the rights of the data subjects listed in Sect. 4 help to translate it into specific requirements.

3.2 Purpose Limitation

The principle of purpose limitation, in GDPR specified in Art. 5(1)b, states that the purpose of processing personal data must be defined in advance, before collecting the data, and the data may not be used for any purpose inconsistent with the initial purpose. Processing data for a new, different purpose therefore is not excluded altogether, but there must be a direct relationship between the original and the new purpose.

Requirement (C-2 Document Purpose)**.** *Before any personal data are collected or processed by the system, the controller shall define and document the purpose to be achieved.*

Again, the adequate place for this documentation in general is the records of processing activities. Furthermore, this purpose should be available to the development organisation before starting development.

Requirement (C-3 Purpose Limitation)**.** *If the system performs some form of processing different from the one initially defined, the controller shall ensure that it is analysed, justified and documented why the new purpose is considered consistent with the old one.*

Of course, there must also be an adequate legal basis for the new processing. Coming back to the case of an online shop, a common objective is to perform various analyses on the customer data collected. To a very limited extent, this may be done based on the condition of legitimate interest, but beyond that, consent by the customer is usually needed. In many cases, it will be easier and more efficient to work with anonymised data instead.

3.3 Data Minimization

According to the principle of data minimization, personal data must be "adequate, relevant and limited to what is necessary in relation to the purposes for which they are processed" (Art. 5(1)c GDPR). Therefore, only those data may be collected and processed that are necessary for the defined purpose of processing, while storing data “just in case” is not allowed. This can be quite a challenge in the context of big data and data science where one often collects data to find out later what information may be gained from them.

The principle of data minimization leads to the following software requirement:

Requirement (C-4 Data Minimization)**.** *The system may only collect and process data that are necessary for the defined and documented purpose. This includes each individual data attribute as well as the overall data set.*

In the case of an online shop, the shop may offer to answer customer questions via a contact page. To answer any such questions, it is usually sufficient to know either the email address or the telephone number, and collecting any personal data beyond that must therefore at least be optional, stating clearly what these data are used for, or better not done at all.

As discussed in Sect. 2.1, it may be useful to convert personal data into anonymous or at least pseudonymous data in order to protect them against unauthorised usage:

Requirement (C-5 Data Anonymisation and Pseudonymisation)**.** *The system should store data in anonymised or at least pseudonymised format where this is possible without limitation for the intended purpose.*

As mentioned before, pseudonymous data are still personal data and need to be protected accordingly, and even with anonymous data, protecting the data against unauthorised access still is recommended where possible since experience shows that anonymisation often can be broken.

In order to analyse the movement of visitors to the website of an online shop (web analytics, as opposed to user tracking), the IP addresses of the visitors are usually anonymised by deleting part of them. This is still sufficient to identify subsequent movements by the same visitor, but does not allow the identification of visitors.

3.4 Accuracy

The principle of accuracy (Art. 5(1)d GDPR) may be summarised as follows:

Requirement (Q-1 Accuracy)**.** *Data processed and stored shall be accurate and, where relevant, kept up to date.*

Additionally, the following requirements are implied:

Requirement (F-9 Rectifying Data)**.** *The system shall provide the controller with the ability to rectify or update data.*

In the case of an online shop, this mainly refers to the fairly obvious need to be able to update or correct customer data such as their address, but also information such as customer interests and preferences, or about customer not having paid previous bills if this turns out to be a mistake.

Requirement (F-10 Consistency Check)**.** *The system should check input data for consistency to recognize and prevent invalid input.*

This requirement is not legally required but helps to implement requirement Q-1. An approach to implement F-10 is the use of drop-down lists where possible rather than allowing free text.

In most cases, implementing the principle of accuracy is also in the controller’s own interest, similar to data minimization. Defining it as a legal requirement ensures that accuracy is also aimed for in those cases where the controller has little or no own interest in keeping the data accurate.

3.5 Storage Limitation

The principle of storage limitation (Art. 5(1)e GDPR) extends data minimization by a temporal view and states that personal data may only be stored as long as they are needed for their defined purpose. After that, they must be deleted or at least turned into anonymous data.

Requirement (F-11 Identify Data no Longer Required)**.** *The system shall provide the controller with the ability to identify any personal data no longer required.*

For example, this includes identifying the data whose required retention period has passed.

Since the controller may choose between anonymising and deleting data that are no longer needed, the following two requirements are expressed as should-requirements where at least one of them *shall* be satisfied in any individual case identified by F-11.

Requirement (F-12 Anonymisation of Data)**.** *The system should provide the controller with the ability to transform stored personal data no longer required into anonymous data. The anonymous data shall be in a format that prevents de-anonymisation with realistic effort. When data are anonymised, the system shall replace all copies of the original data by the anonymised data, unless they are deleted.*

This requirement is similar to C-5, but while C-5 applies to an entire data collection, F-12 refers to selected parts of a data collection, possibly even individual records. Again looking at an online shop as an example, F-12 requires that old customer and order data are anonymised (or deleted, see F-13 below) once legal retention periods have run out, while current customer and order data of course must be preserved.

Requirement (F-13 Deletion of Data)**.** *The system should provide the controller with the ability to delete personal data identified as no longer required, including all instances of the data such as backups.*

3.6 Integrity and Confidentiality

In spite of their close relationship, IT security and data protection start from fundamentally different perspectives. The goal of IT security is to protect data and infrastructure against negative influence from outside, including protection against *unauthorised* access. Data protection, on the other hand, aims at protecting individuals against misuse of their data, which implies restricting *authorised* access as well as protecting against *unauthorised* access. To a large extent, IT security is needed to achieve data protection, but it is not sufficient.

GDPR expresses this relationship in the principle of integrity and confidentiality (Art. 5(1)f) which requires "adequate security"[2]. Taken almost literally from GDPR, the resulting software requirement states:

Requirement (Q-2 Integrity and Confidentiality)**.** *The system shall provide adequate security of personal data, including protection against unauthorised or unlawful processing and against accidental loss, destruction or damage.*

Note that, in spite of the name of the GDPR principle, this includes the third dimension of IT security, namely availability, in addition to the dimensions integrity and confidentiality.

Particularly important for data protection are measures such as the restrictive handling of access rights, the secure storage of access data (for example no storage of passwords in plain text), and encryption of data at least when transferring them between systems.

3.7 Accountability

Accountability is a fairly new principle of data protection that is not currently widespread in other data protection legislation. It states that it is not sufficient to *implement* data protection but controllers need to be able to *show* that they do so by providing adequate documentation such as the "records of processing activities" introduced in Sect. 5.1 below.

Requirement (C-6 Documentation)**.** *The system, in particular its processing of personal data, shall be documented and this documentation be versioned and kept up to date.*

Versioning is important in this context in order to show, in case of problems, what processing was performed at a certain point in time in the past.

Apart from satisfying the legal requirements, such documentation helps to identify the personal data processed and the resulting needs for data protection measures.

4 RIGHTS OF DATA SUBJECTS

Based on the data protection principles above, data protection defines a set of rights of the data subjects. These help to implement the principles and therefore in some cases lead to the same software requirements as described above.

For reasons of space, only a brief list of the relevant rights can be included in the current paper. However, in most cases these rights are fairly specific as expressed in GDPR, which makes it easier to translate them into software requirements compared to the data protection principles:

- Rights to transparency and information (Art. 12–15 GDPR)
- Right to rectification (Art. 16 GDPR)
- Right to erasure (right to be forgotten) (Art. 17 GDPR)
- Right to restriction of processing (Art. 18 GDPR)
- Right to data portability (Art. 20 GDPR)
- Right to object (Art. 21 GDPR)
- Right to request manual decision-taking (Art. 22 GDPR)

In most cases, these rights of the data subjects do not apply in general but only under certain conditions. However, these are usually conditions that have to be checked on the organisational level rather than by the software. As a result, they are not directly included in the software requirements but need to be described as functions initiated by the controller (or, in some cases, the user/data subject).

[2] Of course, this immediately leads to the question of what is considered adequate. The answer obviously depends on the specific environment and the personal data involved, but in many cases, the requirements on information security management systems as defined by ISO/IEC 27001 provide a good guideline.

5 FURTHER REQUIREMENTS

In addition to the principles of data protection and the rights of the data subjects, GDPR defines a number of further data protection requirements. In the following, we summarise those that influence the software requirements.

5.1 Records of Processing Activities

The "records of processing activities" as required by Art. 30 GDPR contain a summary of the processing of personal data that is performed by the organisation, with attributes including the categories of data processed (such as customer addresses, health data of employees etc.), and the technical and organisational measures taken to protect these data. This is related to requirement C-6, but with a different focus: while C-6 is concerned with the documentation of individual systems and more detailed, requirement C-7 puts the focus on providing an overview of the different systems and processes in use, and the personal data processed by them.

Requirement (C-7 Records of Processing Activities)**.** *The controller shall document in the records of processing activities the processing performed by the system, including the purpose of processing, the categories of affected data subjects and of personal data, the recipients of the data, and the technical and organisational measures taken to protect these data.*

Although this is not a requirement to be implemented by software development itself, the controller will typically need support from software development in order to implement this requirement adequately.

5.2 Data Transfer to Third Parties

Another set of important requirements which however are too complex to be covered fully within this paper concern the transfer of data to third parties. In the context of software development, this may for example refer to the use of cloud or other external services, or to the use of certain plug-ins and SDKs which transfer personal data to the provider of the plug-in or SDK. The term "third party" includes any legally separate unit even if it may belong to the same enterprise as the controller.

Software developers should however realise that any such data transfer may lead to complex data protection questions, and should therefore be discussed with the controller as well as data protection specialists before implementation. In particular, it is important to check for unexpected or unwanted data transfer as happens commonly when using third-party SDKs in the development of web applications or mobile apps, see for example (Englehardt et al., 2018).

Requirement (C-8 Data Transfer to Third Parties)**.** *Any transfer of personal data to or from third parties shall be identified. For any such data transfer, the legal basis shall be identified and documented. In case the legal basis is a controller-processor relationship, a legal contract according to Art. 28 GDPR shall be agreed.*

Additional requirements apply if the data are to be transferred outside the European Economic Area (EEA)[3]. In this case, the transfer of personal data is only allowed if the receiving country has an "adequate level of data protection", where GDPR defines a set of conditions that may be used for verifying this property (Art. 44–50 GDPR).

Requirement (C-9 Data Transfer Outside the EEA)**.** *In case any personal data are to be transferred outside the European Economic Area, an adequate level of data protection shall be verified.*

5.3 Data Protection by Design / by Default

The concept of *data protection by design* (Art. 25(1) GDPR), also known as "Privacy by Design" (Cavoukian, 2011), may be summarised as follows:

Requirement (C-10 Data Protection by Design)**.** *Measures to address data protection shall be taken across the entire life cycle of the system, starting with analysis and design.*

The purpose of data protection by design is to ensure that data protection is taken into account from the start rather than as an "add-on" at the end when effective measures become too expensive and therefore are no longer feasible.An important step towards implementing data protection by design therefore is to address the requirements listed in this paper as part of requirements analysis.

Closely related to data protection by design is *data protection by default* (Art. 25(2) GDPR):

Requirement (Q-4 Data Protection by Default)**.** *The system shall be configured such that only minimal personal data are collected and processed as necessary for the relevant purpose. Any further data collection and processing may only be performed if the user explicitly allows this by adapting the configuration.*

[3] The EEA consists of the European Union plus the EFTA states Iceland, Liechtenstein and Norway, and defines the territorial scope of the GDPR.

An example for this is not to save the user preferences by default, unless the user sets a flag that these data should be saved and used.

For both concepts it is true that as of today, they are not widely applied and there are many counter-examples around, partly because these concepts were only fairly recently introduced as legal requirements. Nevertheless, any current software development should adhere to these concepts, both to satisfy user expectations and to prevent legal problems, for example with the relevant supervisory authorities.

6 VALIDATION OF RESULTS

Since the GDPR requirements are expressed in a very different way compared to software and software process requirements, there is no formal way to validate the results in this paper. To ensure the correctness of the results, they were checked against the text of the GDPR itself, supported by some legal commentaries such as (Kühlung and Buchner, 2018), and reviewed by several experts.

More difficult is the completeness of the results. To validate the completeness, the results were compared against different publications that address the effects of data protection and GDPR requirements on software development, in particular (Danezis et al., 2014; Datatilsynet, 2017; Reid, 2017; Santala, 2017; Simon and Moucha, 2019). The results of these comparisons were directly integrated into the results described above, so that the version reported here already includes all requirements that had been identified as missing.

7 CONCLUSIONS

As this paper shows, data protection and in particular the GDPR leads to a set of requirements that need to be addressed in software development, including functional requirements, quality (non-functional) requirements and constraints. Although these requirements strictly speaking only apply to the processing of personal data, a close look shows that this indeed includes a large proportion of the data processed.

This implies that requirements analysts and software developers need to take these requirements into account and check which of them apply in their specific case. Although the requirements presented in this paper will of course have to be adapted and extended for any specific software system, these requirements represent an advanced starting point for the task, providing considerably more detail compared to the GDPR (or other, similar legislation) itself.

REFERENCES

Cavoukian, A. (2011). Privacy by design. The 7 foundational principles. Technical report, Information and Privacy Commissioner of Ontario.

Danezis, G., Domingo-Ferrer, J., Hansen, M., Hoepman, J.-H., Le Métayer, D., Tirtea, R., and Schiffner, S. (2014). Privacy and data protection by design—from policy to engineering. Technical report, ENISA.

Datatilsynet (2017). Software development with data protection by design and by default.

Englehardt, S., Acar, G., and Narayanan, A. (2018). Website operators are in the dark about privacy violations by third-party scripts. https://freedom-to-tinker.com/2018/01/12/website-operators-are-in-the-dark-about-privacy-violations-by-third-party-scripts/.

International Requirements Engineering Board (IREB) (2017). IREB Certified Professional for Requirements Engineering — Foundation Level. Syllabus Version 2.2.2. Technical report.

Kühlung, J. and Buchner, B., editors (2018). *Datenschutz-Grundverordnung / BDSG. Kommentar (in German)*. C.H. Beck, 2. edition.

Kneuper, R. (2019). Integrating data protection into the software life cycle. In Franch, X., Männistö, T., and Martínez-Fernández, S., editors, *Product-Focused Software Process Improvement. 20th International Conference, PROFES 2019, Barcelona, Spain, November 27–29, 2019, Proceedings*, pages 417–432, Cham. Springer International Publishing.

Pohl, K. and Rupp, C. (2015). *Requirements Engineering Fundamentals*. Rocky Nook, 2nd edition.

Reid, G. (2017). How to navigate the software development life cycle under the GDPR. International Association of Privacy Professionals (IAPP).

Santala, A. (2017). What should software engineers know about GDPR?

Simon, K. and Moucha, C. (2019). Sicherheit und Datenschutz im Lebenszyklus von Informationssystemen (in German). *DuD Datenschutz und Datensicherheit*, 43(2):97–101.