**Final Guidance: Content of Premarket Submissions for Device Software Functions**
**July 20, 2023**

**Moderator: CDR Kim Piermatteo**

**CDR Kim Piermatteo:** Hello and welcome to today's CDRH webinar. Thank you for joining us today. This is Commander Kim Piermatteo of the United States Public Health Service and I serve as the Education Program Administrator in the Division of Industry and Consumer Education in CDRH's Office of Communication and Education. I'll be your moderator for today's program.

Our topic today is the final guidance titled "Content of Premarket Submissions for Device Software Functions." This final guidance was issued on June 14, 2023, and describes the FDA's thinking on the recommended documentation sponsors should include in premarket submissions for the FDA's evaluation of the safety and effectiveness of device software functions.

This guidance replaces the FDA's guidance for the "Content of Premarket Submissions for Software Contained in Medical Devices," which was issued on May 11, 2005, and describes information that would be typically generated and documented during software development, verification, and validation.

We are holding this webinar to discuss the risk-based approach to determine the recommended documentation level for a premarket submission and to discuss the recommendations for information to be included in premarket submissions for basic and enhanced documentation levels. We will also answer your questions about this final guidance.

A few quick reminders. First, please make sure you've joined us through the Zoom app and not through a web browser. This will help avoid any technical issues. And second, the intended audience for this webinar is industry. Members of media are encouraged to consult with the FDA's Office of Media Affairs for any questions you may have.

It's my pleasure now to introduce you to our presenters, Sonja Fulmer, Acting Deputy Director for the Digital Health Center of Excellence and CDRH's Office of Strategic Partnerships and Technology Innovation; Jessica Paulsen, Digital Health Associate Director within CDRH's Office of Product Evaluation and Quality, or what is also referred to as OPEQ; and Aneesh Deoras, Assistant Director for Cardiac Ablation, Mapping, and Imaging Devices Team in the Office of Cardiovascular Devices within OPEQ as well.

We'll begin with a presentation from Sonja, Jessica, and Aneesh, and then field your questions about today's topic. Thank you all again for joining us. I'd now like to turn it over to Sonja to start today's presentation.

**Sonja Fulmer:** Thanks, Kim. And thank you all for joining today to hear about this guidance document. I'm Sonja Fulmer. I'm the Acting Deputy Director for the Digital Health Center of Excellence.

Today we'll be talking about the content of premarket submissions for device software functions guidance. We issued this final guidance on June 14, 2023. A copy of the guidance is available at the link on your screen.

There are a few key takeaways for today's webinar. We will be explaining what is covered in the guidance and why we updated this guidance. We will describe the purpose and scope of the guidance to make clear who should use it and for what types of devices. We'll also provide details on the different software documentation levels as well as software documentation elements. Finally, we'll make clear a few other regulatory considerations and plans for updates to other guidances.

We'll start today with some background on what's covered in the guidance and why we made an update.

This guidance document answers the questions of what software documentation is recommended for marketing submission and what should the documentation demonstrate. The documentation recommended in this guidance is based on FDA's experience evaluating the safety and effectiveness of device software.

Please note that this guidance does not provide recommendations on how device software should be developed, verified, and validated. It does not recommend the use of any specific software life cycle model or development methodology. Software developers are free to use the methodology that works best for them. We just provide some recommendations on how to document the information to us in a submission.

This guidance document replaces a 2005 guidance on the same topic entitled, "Guidance for Content of Premarket Submissions for Software Contained in Medical Devices." We updated the 2005 guidance to provide our current thinking related to the documentation we recommend to be included for the review of device software functions. These updates are intended to foster timely access to safe and effective software devices. The guidance takes a least burdensome approach to provide clear and simple recommendations on software documentation.

The language in this guidance also aligns with the 2016 21st Century Cures Act and is intended to harmonize with FDA recognized standards. This update completes the MDUFA V commitment as part of our work to streamline and align FDA review processes with software life cycles for digital health products.

Next, I'll take a moment to summarize some of the comments we received on the draft guidance that was issued in November of 2021. Commenters requested clarification on how to apply documentation levels for certain premarket submissions, and they requested more examples of applying those documentation levels. They also suggested more flexibility in documentation level factors to reduce the documentation burden. Commenters highlighted the opportunity to apply recommendations to device constituent parts of combination products.

Comments to the draft guidance also emphasized ways to further align with IEC 62304. They raised questions about the appropriate software development methodology and observed few recommendations for artificial intelligence and machine-learning-based devices. In addition, they noted general clarifications, editorial changes, and corrections.

We made several changes to the guidance in response to the comments received on the draft. We simplified the documentation-level considerations and how you can apply those. And we've added more

examples on how to apply the documentation level, increasing the number from 5 to 27 examples in the final guidance.

We also revised specific recommendations to address device constituent parts of a combination product and added the Center for Drug Evaluation and Research and the Office of Combination Products to the guidance.

In addition, we harmonized the risk terminology with ISO 14971 and clarified leveraging of IEC 62304 to reduce burden while maintaining differences where appropriate. We provided a fully revised system and software architecture diagram examples and included a few other general updates, corrections, and clarifications.

This guidance is intended to complement other existing guidance documents that provide recommendations related to software, such as premarket submission recommendations for interoperable medical devices and the management of cybersecurity in medical devices. This slide is meant to be a resource for you to find other related guidances, as well as related FDA-recognized consensus standards that can be useful to you as you prepare a premarket submission for device software.

Next, we'll take a moment to continue to clarify the purpose and scope of this guidance.

The purpose of this guidance is consistent with the 2005 guidance. It identifies the software information generally necessary to evaluate the safety and effectiveness of device software functions in a premarket submission. We applied the least burdensome approach to identify the minimum amount of information we generally need to support a premarket submission for a device that uses software.

This is the type of information that's typically generated and documented during software development, verification, and design validation. The expectation is that software developers already have this information at hand. Please note that FDA may have additional questions and request other information beyond what is listed in the guidance in order to evaluate a submission during a premarket review.

This guidance applies to all types of premarket submissions that include one or more device software functions. I'll define the term "device software functions" on the next slide. This guidance applies to device constituent parts of a combination product when that device constituent part includes a device software function. This guidance does not apply to software that is not a device.

As indicated by the title, this guidance is focused on premarket considerations and submissions. It does not apply to automated manufacturing and quality system software. It also does not apply to any software-related documentation that may be needed to evaluate post-market software device issues. Please note, there are additional guidances that address other aspects of premarket submissions for software, including, for example, cybersecurity topics.

In alignment with the 21st Century Cures Act, this guidance uses the term "function." For any given product, the term "function" is a distinct purpose of the product, which could be the intended use or a subset of the intended use of the product. For example, a product with an intended use to analyze data has one function, analysis. A product with an intended use to store, transfer, and analyze data has three functions, storage, transfer, and analysis.

FDA refers to a software function that meets the definition of a device as a device software function. And the use of these terms is consistent with other recent guidance documents that may be useful for you to review, such as the “Policy for Device Software Functions and Mobile Medical Applications” guidance and the “Multiple Function Device Products: Policy and Considerations” guidance.

Before getting into the meat of the guidance, I also want to take a moment to define the terms "software verification" and "software validation." Software verification is the confirmation by objective evidence that the output of a particular phase of development meets all the input requirements for that phase. This involves evaluating the consistency, completeness, and correctness of the software and its supporting documentation as it is being developed and provide support for a subsequent conclusion that software is validated.

Whereas software validation refers to establishing by objective evidence that the software specifications conform to user needs and intended uses and that the particular requirements implemented through software can be consistently fulfilled. Software validation is a part of design validation of the finished device. It involves checking for proper operation of the software in its actual or simulated use environment, including integration into the final device where appropriate.

Now that we're on the same page for those terms, I'll pass you on to Jessica Paulsen, who will describe the risk-based documentation levels for device software. Jessica?

**Jessica Paulsen:** Thank you, Sonja. My name is Jessica Paulsen, and I serve as the Digital Health Associate Director in the Office of Product Evaluation and Quality.

So let's go ahead and review documentation level and how to apply it.

In this final guidance, we've outlined a simplified documentation level that replaces the previous level of concern from the 2005 guidance. So there's two documentation levels defined for devices; basic and enhanced.

The purpose of documentation level is to help identify the minimum amount of software information that would support a premarket submission. The documentation level depends on a given device's risk to a patient, user of a device, or others in the environment of use. The documentation level is based on the risks of the device software functions in the context of the device's intended use, and it should reflect the device as a whole.

So let's review when an enhanced documentation level is appropriate. This should be provided for any premarket submission that includes device software functions where failure or flaw of any device software function could present a hazardous situation with a probable risk of death or serious injury, either to a patient, user of the device, or others in the environment of use. These risks should be assessed prior to implementing risk control measures, and a sponsor should consider these risks in the context of the device's intended use-- for example, impacts to safety, treatment, or diagnosis, and other relevant considerations.

Basic documentation should be provided for any premarket submission that includes device software functions where enhanced documentation does not apply.

When determining the appropriate documentation level for a given device, it's important for a sponsor to consider all known and foreseeable software hazards and hazardous situations associated with the device. This includes those resulting from reasonably foreseeable misuse, whether intentional or unintentional, prior to implementation of risk control measures. This also includes likelihood that the device's functionality is intentionally or unintentionally compromised by inadequate device cybersecurity. The sponsor is responsible for proactively and comprehensively considering these risks as part of the device's risk assessment.

It's important to individually assess devices within scope of the guidance to determine the appropriate documentation level for a given device. The guidance recommends enhanced documentation be provided for devices that are intended to test blood donations for transfusion-transmitted infections, determine blood donor and recipient compatibility, automate blood cell separator devices intended for collection of blood components for transfusion or further manufacturing use, and blood establishment computer software.

The guidance also generally recommends enhanced documentation be provided in a premarket submission for Class III devices and device constituent parts of a combination product. However, a sponsor may determine that enhanced documentation does not apply in certain cases, for which the sponsor should provide a detailed rationale as to why basic documentation is appropriate for the subject premarket submission.

So to help illustrate how to apply documentation level, we've included a number of examples in Appendix A of the guidance. And we wanted to walk through a couple of those examples today.

So here, we have a non-contact infrared thermometer intended for intermittent measurement of body temperature from the forehead. So this device is intended to measure body temperature from the forehead using an infrared sensor. It's a handheld, battery-powered, reusable device for use in the home and professional health care facilities.

So when considering documentation level for this example, in general, a failure or latent flaw of the device software functions would not present a hazardous situation with a probable risk of death or serious injury to either a patient, user of the device, or others in the environment of use prior to implementation of risk control measures. Therefore, basic documentation level is appropriate for this example device.

The next example is a facility use continuous ventilator. So this device is intended to provide continuous ventilation for adult, pediatric, and neonatal patients who require invasive or non-invasive respiratory support. The device allows clinicians to set ventilator control parameters, set alarm limits, and view monitored values and waveforms for patient management. It includes respiratory monitoring, as well as both mandatory and spontaneous ventilation modes. And the device is intended for use in professional health care facilities.

So when considering documentation level for this device, a failure or latent flaw of the device software functions, such as failure to provide appropriately timed ventilation to a patient, would present a hazardous situation with a probable risk of death or serious injury to a patient prior to implementation of risk control measures. Therefore, enhanced documentation is appropriate for this example.

So I'd encourage anyone interested in more examples of how documentation level is applied to review Appendix A of the guidance. And I'm now going to turn the presentation over to my colleague Aneesh to continue.

**Aneesh Deoras:** Thanks, Jessica. My name is Aneesh Deoras. I am the Assistant Director for Cardiac Ablation, Mapping, and Imaging Devices in the Office of Cardiovascular Devices, Office of Product Evaluation and Quality.

Next, we'll be talking about the software documentation elements that the guidance recommends be included in submissions containing software.

Here, we have a list of the software documentation elements. And we'll be going through these items one by one.

First is the documentation level evaluation. The guidance recommends that you provide a statement indicating the documentation level and a description of the rationale for that level. The guidance encourages sponsors to account for the device's intended use and to leverage their device's risk assessment when preparing the rationale for choosing a documentation level. The examples in Appendix A are provided to demonstrate how to implement the documentation level. During premarket review, FDA may request additional information that is needed to evaluate the submission.

Next is the software description. The guidance recommends that you provide a comprehensive software description, including an overview of significant software features, analyses, inputs, outputs, and hardware platforms. The guidance provides a curated set of questions to help readers prepare focused device description information. The guidance encourages the inclusion of additional information if needed to help further FDA's understanding of the device's functionality, such as images, flow charts, and state diagrams.

For a modified device, the guidance recommends a description of the software changes from the previous premarket submission, which may include providing the document number for that previous submission and highlighting pertinent software changes that have happened since the last authorization.

The next documentation element is the risk management file. For both basic and enhanced documentation levels, the guidance recommends that you include three components in your risk management file. The first is the risk management plan, which demonstrates how a manufacturer plans to approach a risk assessment for their device and evaluate the overall residual risk against the benefits of the intended use of the device.

The second element is a risk assessment, which documents in a tabular format the known or foreseeable hazards and resulting hazardous situations, an initial risk evaluation of the hazardous situation, risk control measures, residual risk evaluation after the implemented risk control measures, and traceability of risk control measures. Finally, a risk management report, which shows how the risk management plan has been appropriately implemented.

The guidance recommends that sponsors refer to an FDA-recognized version of ISO 14971, an account for the recommendations provided in the guidance "Multiple Function Device Products: Policy and Considerations."

Next is the software requirements specification. The guidance recommends that you include complete documentation of your software requirements, describing the needs or expectations for a system or software, presented in an organized format and with sufficient information to understand the traceability of the information with respect to the other software documentation elements. The recommendations acknowledge modern development practices and allow for different forms of software requirements to be included in the submission, such as well-elaborated stories, use cases, textual descriptions, screen mockups, and control flows.

The section includes considerations when preparing SRS documentation to help facilitate a timely premarket review, such as tips for formatting and labeling, inclusion of traceability information, and to highlight requirements that you believe are most critical to a device's safety and effectiveness or those that were modified since a previous clearance or approval. For additional details on what can be included in the software requirements specification, please refer to the guidance "General Principles of Software Validation."

The next documentation element is the system and software architecture diagram. The guidance recommends that you provide detailed diagrams of the modules, layers, and interfaces that comprise the device, the data inputs, outputs, and flow, and how users or external products, including IT infrastructure and peripherals, interact with the system and software. The guidance recommends that sponsors provide an appropriate level of detail to convey this information in a manner that facilitates an efficient premarket review. The guidance includes visual language and reference considerations that can be leveraged when developing the diagrams for a premarket submission.

Appendix B of the guidance includes an example system and software architecture diagram.

Appendix B, System and Software Architecture Diagram Examples, describes how the recommendations in the guidance can be implemented into a system and software architecture diagram. The examples are intended for illustration purposes only and are not intended to document a comprehensive system and software architecture diagram for a specific device or system. The approaches illustrated can be applied to any system and software architecture diagram. And the examples do not preclude the use of off-the-shelf architectural modeling, platforms, or languages.

Appendix B includes three example diagrams. The second is for an implantable therapeutic device that includes an external reader and a programmer, a cloud service provider, and a patient-facing mobile platform. As you can see here, there is a legend with very clear, outlined ways of describing each individual function. We can see data flows here with arrows. And the arrows themselves are described in the legend.

Another useful aspect is a box for descriptive information that can really help outline what's happening in the diagram and then also link the diagram to other documentation, like the system requirements.

The next element is the software design specification. For basic and enhanced, the recommendations are different. For basic, FDA is not recommending that you include the software design specification in

your premarket submission. However, we do recommend that you document this information in your design history file.

For enhanced documentation, we do ask that you include the software design specification, which will be a singular or set of SDS documents that provide the technical design details of how the software functions, how the software design completely and correctly implements all of the requirements of the software requirements specification, and how the software design traces to the SRS in terms of intended use, functionality, safety, and effectiveness.

The next element is the software development, configuration management, and maintenance practices section. There are two options for this section. The first option is, you may provide a Declaration of Conformity to the currently FDA-recognized version of ANSI/AAMI IEC 62304, Medical Device Software - Software Life Cycle Processes. The guidance includes specific sections that are applicable to each basic and enhanced to align with IEC 62304.

Alternatively, you do not need to declare conformity to the standard. You can instead, for basic documentation, provide a summary of the processes and procedures that are in place to manage the software life cycle development, software configuration and change management, and software maintenance activities. For enhanced documentation, we recommend providing a complete configuration management and maintenance plan, in addition to the summary documentation requested for basic documentation level.

The next element is software testing as part of verification and validation. This documentation element is different for basic and enhanced documentation levels. For basic, the guidance recommends providing a summary description of the testing activities at the unit, integration, and system levels, and then provide the complete system-level test protocols and reports. For enhanced documentation level, the guidance recommends providing the same information, but also include the full unit and integration level test protocols and reports.

The definition section of the guidance includes important definitions for verification and validation as they pertain to this guidance. And so we recommend referring to that section. And also, sponsors are encouraged to appropriately reference performance testing material in the submission to help facilitate navigation between submission sections, reduce instances of duplication, and improve readability.

The next element is a software version history. The guidance recommends that you include a history of tested software revisions, including date, version number, and a brief description of all changes relative to the previously tested software version. FDA recommends beginning with the version that became subject to design controls, as described in 21 CFR 820.30. A version history typically takes the form of a line item tabulation, including the date, version number that was tested, including, if applicable, bench, animal, and clinical testing, and a brief description of all changes in the version relative to the previously tested version.

The last entry in a line item tabulation should be the final version to be incorporated in the released device. This entry should also include any differences between the tested version of the software and the release version, along with an assessment of the potential effect of the differences on the safety and effectiveness of the device.

The last documentation element we'll discuss is unresolved software anomalies. The guidance recommends that you provide a list of remaining unresolved software anomalies with an evaluation of the impact of each unresolved software anomaly on device safety and effectiveness. The section recommends the following information be provided in a tabular format-- a description of the problem; identification of how the anomaly was discovered, and, if possible, identification of its root causes; evaluation of the impact of the anomaly on device safety and effectiveness; the outcome of that evaluation; and a risk-based rationale for not correcting or fixing the anomaly.

The guidance encourages communication of unresolved anomalies to end users as appropriate to assist in proper device operation. The guidance also includes a reference to ANSI/AAMI SW91, classification of defects in health software, to help with the classification of unresolved software anomalies.

Next, I'll turn it over to Sonja Fulmer for remaining regulatory considerations.

**Sonja Fulmer:** Thanks, Aneesh, and thanks to you and Jessica for taking us through the details of the document.

To wrap up today's webinar, I'll provide a few other regulatory considerations and information on planned updates to other guidance documents.

First, at the beginning of this webinar, we talked about how this guidance applies to device software functions. You may have questions about whether or not your product is a device software function that is the focus of FDA's oversight. The guidance documents and websites listed on this slide provide information on what types of software functions are devices.

There's also a tool called the Digital Health Policy Navigator that can help you understand the regulatory status of a software function. These will help you understand whether or not the guidance document we're talking about today applies to your software function. All links to guidance documents, websites, and other resources are provided at the end of the slide deck, which is available online to download.

We will also be issuing a minor update to the “Off-The-Shelf Software Use in Medical Devices” guidance, as indicated in the draft guidance. The off-the-shelf guidance will be updated to be consistent with the final 2023 guidance.

There will also be a number of additional guidance updates for guidances that reference the 2005 premarket software guidance and a specific level of concern. These guidances will be updated in the near term with a cover page to indicate a documentation level aligning with the 2023 guidance, rather than a level of concern.

You may also have seen that this guidance was announced with a 60-day transition period to allow time for both FDA and sponsors to operationalize the recommendations discussed in the guidance. During this transition period from June 14 to August 13, 2023, sponsors can use either the 2005 or 2023 guidance to submit software-related documentation.

Starting August 14, our expectation is that sponsors should use the 2023 guidance to help them understand software-related documentation for premarket submission. Currently, during this transition period, there are two versions of eSTAR available. Version 3 reflects the 2005 guidance policies. Version

4 reflects the 2023 guidance and allows sponsors to select basic or enhanced documentation level for the software content in their submissions.

I'll close now with a few summary points about the guidance. The guidance is intended to simplify the organization and content of software documentation elements, as well as the categorization levels. In this guidance, we provide recommendations that should help sponsors prepare their software documentation. This guidance complements several existing software guidances, which are listed in the Resources slide at the end of this deck.

We updated this guidance to harmonize with software-related consensus standards, reflect changes made to software regulation from the 21st Century Cures Act, and ultimately to provide least-burdensome, risk-based recommendations based on our experience evaluating the safety and effectiveness of device software. As a reminder, this guidance supersedes the 2005 guidance on this topic.

Thank you all for your attention during this webinar. We're looking forward to your questions. Back to you, Kim.

**CDR Kim Piermatteo:** Thank you, Sonja, Jessica, and Aneesh, for that presentation. We will now transition to the interactive question-and-answer segment of today's webinar, where you get to ask your questions to our panel. I'd like to first introduce several additional panelists who will be joining our presenters and answering your questions today.

Samantha Collado-- she is a regulatory Policy Analyst in CDRH's Office of Policy. Also, Jason Ryans, Policy Analyst on the Regulation, Policy, and Guidance staff within CDRH's Office of Product Evaluation and Quality, or OPEQ. And Justin Post, Policy Analyst in the Office of Health Technology Number 7 for in vitro diagnostics within OPEQ. Thank you all for joining us and being on our panel today.

Before we begin, I'd like to go over how we will manage today's question-and-answer segment. To ask a question, please select the Raise Hand icon, which should appear on the bottom of your Zoom screen. For your information, all hands that were raised during the presentation were lowered. And you may now raise your hand at this time to get into the queue to answer-- or to ask a question.

Next, I'll announce your name and give you permission to talk once you've selected that Raise Hand icon. And then when prompted, please select the blue button to unmute your line and then ask your question. After you ask your question, please lower your hand. And if you have another question, please raise your hand again to get back into the queue, and I'll call on you as time permits.

A few additional reminders-- one, please remember to limit yourself to asking one question only and to keep it as short as possible. And two, we appreciate that you may have a very specific question involving your device or scenario. Please note that we may not be able to answer such specific questions today. But we'll try to frame a broader response based on what's being proposed in the guidance and the topic of today's webinar.

Remember, this is your chance to better understand and get clarity on what we intend in the final guidance. So we ask you to try to frame your questions with that in mind. And then now, as we wait to

receive your questions, I'd like to welcome our newest panelists with some questions we have gotten over the past few weeks about the guidance.

For our first question, I'll be directing that to Samantha. Samantha, the question is, how does level of concern from the previous guidance map to documentation level?

**Samantha Collado:** Hi, Kim. Thank you so much for that question. In both the 2005 version of the guidance and the final 2023 version of the guidance, the amount of documentation a sponsor should provide in a premarket submission is based on the level of risk to the patient or the user of the device. In the 2005 version of the guidance, this level was called level concern and was determined through a series of questions to determine whether a device was major, moderate, or minor level of concern. Answering yes to a question such as, "Is the software device intended to be used in combination with a drug or biologic," meant that the device was automatically a major level concern.

In this, the 2023 version of the guidance, the documentation level of device, whether enhanced or basic, is based on the risks of its device software functions in the context of the device's intended use, such that the documentation level reflects the device as a whole. Given this approach, in many cases, devices who were previously considered to have a moderate level of concern will likely have a basic documentation level. And devices with a major level of concern will have an enhanced documentation level. Regardless, it is important to consider the risks of the device software functions in the context of the device's intended use, as outlined in the 2023 guidance, to determine the appropriate documentation level.

**CDR Kim Piermatteo:** Great. Thanks, Samantha. Now, for our next question, I'll be directing that to Jason. Jason, the question is, "How long will manufacturers be able to use the policies in the 2005 guidance, including the level of concern framework?"

**Jason Ryans:** Thanks, Kim. As mentioned in the guidance and earlier in this presentation, there is a 60-day implementation period running from June 14 to August 13. During the implementation period, sponsors can either utilize the 2005 guidance or the 2023 guidance. Starting August 14, there is an expectation that all sponsors will utilize the 2023 guidance to submit software related to documentation for premarket submission.

With regards to eSTAR, Version 3 of the template maps to the 2005 version of the guidance and the level of concern. Version 3 of eSTAR will be retired on August 13. Version 4 that is also currently available maps to the 2023 guidance and the documentation level. As an additional note, on October 1, 2023, all 510(k) submissions, unless exempted, must be submitted via eSTAR.

**CDR Kim Piermatteo:** Thanks, Jason. Alright, for our next question, that's coming to you, Justin. Justin, the question is, "In contrast to the 2005 version of the guidance, a traceability analysis document is no longer recommended to be part-- or to be provided as part of software documentation provided with a premarket submission. Can you speak to FDA's perspective on that?"

**Justin Post:** Yes. Thanks, Kim. While FDA is not recommending a specific traceability analysis document as part of a premarket submission, there is a general expectation that sponsors document this analysis in their design history file for the device in alignment with the quality system regulations in 21 CFR Part 820. In the 2023 guidance, FDA has taken a least-burdensome approach and has decided to evaluate the

adequacy of a sponsor's traceability procedures via other software documentation provided in this premarket submission. For example, in the risk assessment provided as part of the risk management file, we intend to evaluate how identified risk control measures trace to product requirements, software testing, and labeling, where applicable.

Furthermore, we'll also review the overall traceability between requirements listed in the software requirements specification and information related to those requirements and other software documentation, such as the software design specification, system and software architecture design, and software testing documentation.

**CDR Kim Piermatteo:** Thank you very much, Justin and Jason and Samantha. Alright, our first live question is coming from Jurgen. Jurgen, I have unmuted your line. Please unmute yourself and ask your question.

**Pesara Jurgen:** Yeah, thank you. My name is Pesara. And I'm with the company Drager, and I have a question about unresolved software anomalies. What exactly is meant by the outcome of the evaluation? Is that really only a description of the impact on safety and effectiveness? Or does it also include a decision on a time frame when to fix it as it was previously. Or even is it an analysis of possible other instances, which is something that we were asked in previous submissions that we did?

**CDR Kim Piermatteo:** Thanks, Pesara. Aneesh, did you want to take that one?

**Aneesh Deoras:** Yeah, that's a really great question. So I definitely understand where that kind of confusion was coming from. The point of saying "outcome" is really just to say that not everybody is using the same kind of approach to analyzing their bugs, anomalies, et cetera. And so we just want to understand exactly what your outcome was, particularly if it's not going to be fixed in the final version of the software. Any details you can provide about how you decided to leave anything in there is really helpful, though, in helping us make our decisions. So I hope that answers your question.

**Pesara Jurgen:** OK, so time frame is no longer required when we plan to fix it.

**Aneesh Deoras:** Well, we suggest that you include your time frame, if possible. Sometimes there isn't really a time frame. And that's the case. But if you could explain why there's no time frame, that helps, too. But if you do have a time frame-- you know, the next release is going to be the fix-- then that can be helpful for us to understand your software.

**Pesara Jurgen:** OK, thank you.

**CDR Kim Piermatteo:** Thank you, Pesara and thank you, Aneesh. Alright, our next question is coming from Gergely Antalfi. I've unmuted your line. Please unmute yourself and ask your question.

**Gergely Antalfi:** Thank you for taking my question. My name is Gergely Antalfi and I'm part of Boston Scientific. And I have a question related to the difference in language between this premarket submission guidance and the cybersecurity premarket submission draft guidance that came out last year. Specifically, in this presentation, too, you mentioned that related to software hazards. This also includes the likelihood that device functionality is intentionally or unintentionally compromised by inadequate device cybersecurity. So that's from this guidance.

And in the draft guidance for cybersecurity, line 312 says, "Effective security risk management also addresses that cybersecurity-related failures do not occur in probabilistic manner and where assessment of likelihood of occurrence and of particular risk could not be estimated based on historical data or modeling." Could you comment on this language difference?

**CDR Kim Piermatteo:** Thank you, Gergely. Justin, would you like to provide a first response? And then anyone else from the panel can chime in as well.

**Justin Post:** Yeah, sure. Obviously, you pointed out the draft guidance, which is in draft out for public comment. It's not final. And then the risk assessment in the cyber draft is obviously different than the risk assessment that's being called out in the premarket guidance. Obviously, when you talk about risk, there's risk from a cybersecurity perspective, which is done a certain way. And there's also risk done from a safety perspective.

So if you have a device that we would consider a cyber device per the 524(b) statute that was effective March 29 of this year, there is an expectation that you do both a cybersecurity risk assessment per that statute for any submissions submitted after March 29 of this year, and then a safety risk assessment in alignment with what's in our final guidance for software. I hope that answers your question.

**Gergely Antalfi:** Mostly. I was interested in the likelihood of cybersecurity events and how that should be considered for the patient safety perspective, because one guidance seems to be implying that it should not be considered in probabilistic manner. The other guidance seems to indicate that there is some kind of probabilistic calculation in the setup.

**Justin Post:** Yeah. Yeah, so when it comes to cybersecurity risk assessments, we're really focused on more exploitability versus the impact of that risk, whereas for software, as you know, probability is difficult to assess. So you have to decide if that's a realistic approach for you to take.

But there are differences in how you approach risk assessment for cybersecurity versus just software in general. And from a probability standpoint for cyber, without going too far away from the topic of discussion today, we're really more interested in cybersecurity for talking about exploitability, not the intentions of a malicious hacker to go after your software, but actually how is the software vulnerable to exploitation, and then looking at the impact from a cybersecurity perspective.

**Gergely Antalfi:** Thank you very much.

**CDR Kim Piermatteo:** Thank you for that question, and thank you, Justin, for that response. Our next question is coming from Kimberly. Kimberly, I've unmuted your line. Please unmute yourself and ask your question.

Kimberly, are you able to unmute your line?

**Kimberly McCarthy:** Yes, can you hear me?

**CDR Kim Piermatteo:** Yes, we can.

**Kimberly McCarthy:** OK, I'm curious to ask what level of detail are you looking for in the relationship to the traceability of the SRS to the SDS?

**CDR Kim Piermatteo:** Thank you, Kimberly, for that question. Justin or Aneesh, would you like to provide a response?

**Aneesh Deoras:** Yeah, I can take a first stab. So I think one of the things we're trying to stress here is just that you don't really need a traceability matrix like we were asking for in the past, partly because that doesn't really align with how people are developing software today using different tools. So this kind of helps things be a little bit easier.

We do want to get a sense from your SRS and SDS that you do have a traceability system in place. And as far as you'd like to describe your traceability system, that's appreciated.

Where we actually ask for a little bit more about traceability is in the risk assessment section, to make sure that safety-critical risks are being traced all the way through to their validation. So that's just the initial thought I would provide. Justin, do you have any other comments on that?

**Justin Post:** Yeah, so when we talk about software requirement specification, that's kind of what we'd say, like the high-level design and where the SDS kind of gets into that low level. So in general, we just want to understand how the high-level design gets kind of translated into the lower-level design that software programmers are using as they start the coding process. So we want to make sure that that's kind of a prospective activity, and that's being thought of during the design development process so that we can kind of see that the thinking and the procedures are there to kind of show that you've taken the requirements and translated that into the lower-level design.

**CDR Kim Piermatteo:** Thank you very much, Justin and Aneesh. Thank you, Kimberly for that question. Our next question is coming from Rhonda. Rhonda, I have unmuted your line. Please unmute yourself and ask your question.

**Rhonda:** Hi. Thank you so much. Following on what Kim was saying, for one thing, the 62304 does require trace matrix. They want to see traceability. One of the questions that we have-- well, it's kind of two parts.

The question we have is that in the enhanced documentation section, it states that we need to show-- [COUGHS] excuse me-- that we need to show SDS documentation, including sufficient information to allow FDA to understand the technical details, blah, blah, blah, which we interpret to be a trace matrix. We're not sure how else we would show this.

And then inside that trace matrix, one of the things that was changed in this draft-- in this document was the software verification validation description. It says about tracing from test cases to source code. And that's got a lot of people questioning what that means, test cases to source code.

**CDR Kim Piermatteo:** Thanks, Rhonda. I'm going to turn it over to Justin.

**Justin Post:** Yeah, so as I mentioned in the opening Q&A, we would expect as part of any software development process that you internally are documenting your traceability, from user needs all the way

down to test cases. The scope of this guidance is really about what we're looking to review as part of our review to determine safety and effectiveness of the device. And part of my explanation that I gave in the Q&A was, what we're looking at is kind of that risk assessment piece as part of that, your traceability procedures.

So we're looking at documentation. We're not looking at source code. I think you mentioned source code. But we just want to make sure as we review some of your documentation, especially risk, when you're putting together your risk control measures, that we can trace that to, OK, where is the requirement where that was implemented? Where did you test that? Because that gives us an indication that you have procedures and processes in place to adequately do that.

And as we mentioned in the guidance, we can also-- if we start to dig there and start to see some issues, then we can start asking for additional information, if needed, to serve the safety and effectiveness requirement for the device. But that's really the expectation is, we're not really looking for an Excel file with all the links between them. You should be documenting that internally. But we kind of want to see evidence of traceability throughout the software documentation that's requested in the guidance.

**Rhonda:** OK, so test cases to source code is not as literal as we're making it to be. Thank you.

**Justin Post:** Yeah, I mean it's really-- we want to see, at a minimum, test cases to requirement. If you have a requirement, obviously, requirements should be written in a manner that they're testable. We want to make sure [BREAK IN AUDIO] traceability of those requirements in a test case somewhere. So, yeah, we're not asking you to provide source code as part of a review, obviously. But yeah, that's kind of the expectation there. Hopefully, that answers your question.

**Rhonda:** Thank you so much.

**CDR Kim Piermatteo:** Thanks, Rhonda, and thanks, Justin. Alright, our next question is coming from Denise. Denise, I've unmuted your line. Please unmute yourself and ask your question.

**Denise Angwin:** Hello there. Can you hear me?

**CDR Kim Piermatteo:** Yes, we can.

**Denis Angwin:** Great. My question is related to the software history, software version history. During development, we use many different versions. We don't necessarily do a full testing on them. Some, we might use for hardware testing, standards testing, usability testing. What versions of software do you want us to be describing in the submission? Because from beginning to end of a project, that could be many different versions.

**CDR Kim Piermatteo:** Thanks, Denise. Aneesh, do you want to take this question?

**Aneesh Deoras:** Yeah, that's a great point, Denise. So generally, what we're looking for is just enough that if you have a version number that you can assign that this version number was used for XYZ testing, this version was used for XYZ testing, that kind of detail is really what we use the version history for. And we incorporate that throughout the whole review because our review of devices is not just the software. It's also going to be clinical, non-clinical, bench testing.

And so having enough detail about what version of the software is tested in each of those settings is really valuable for us, because then we can understand what changed between these. Often, we'll end up sending a question about, OK, well, you did this test two years ago, well, what was the software when you did that test? So trying to explain that level of detail is really helpful.

And it's definitely understood you might have multiple versions going in different directions. But at least if you can provide enough detail to explain what's changing in major steps, that's helpful. It doesn't have to be perfect. It doesn't have to be thousands and thousands of lines of every single Git edit that's happened, but enough that you can get some detail, that's really helpful.

**Denise Angwin:** Yes, thank you. That's actually-- I mean, I guess I want to just restate it. We don't have to tell you for every build that we made. But you want to hear about the major ones we use, like for usability testing or some standards testing, which would mean it would be a few, not thousands.

**Aneesh Deoras:** Yeah, exactly. And I kind of wish we could have written it like that. That's perfect. Thank you.

**CDR Kim Piermatteo:** Thank you, Denise, for the question. Thank you, Aneesh, for the response. Alright, next, we're moving to Sam DeMarco. Sam, I've unmuted your line. Please unmute yourself and ask your question.

**Sam DeMarco:** Yes, good afternoon. Sam DeMarco here from Stryker. Can you help me understand if there's alignment between the software safety class as defined by 62304 to the guidance's definition of risk and the recommendation for the documentation level? For example, Class B could be defined as non-serious injuries possible. Class C is defined as death or serious injuries possible. Would that mean, then, that only Class C should be submitted via the enhanced level, and Class B would be submitted via the basic level? Thanks.

**CDR Kim Piermatteo:** Thanks, Sam. I'm going to turn that over to Aneesh for a response.

**Aneesh Deoras:** Yeah, Sam, that's a great question. And thank you for answering that. I kind of wish we had suggested that in the beginning.

So one of the things to really remember is that the documentation level is pretty different from 62304 and how the software safety classification is written up. We went through line by line in each to figure out exactly how they are different. And there are subtle differences that make it really hard to directly relate them.

We try to, in some areas. So like in software development, where we're leveraging 62304 recognition, largely, A and B match up with minor, and C matches up with enhanced. But otherwise, it really doesn't work out. There are some subtle differences. So I'd really recommend independently doing your assessment per what the guidance says, because then you'll have the closest outcome.

For the most part, we did try and avoid any kind of conflicts. There are some conflicts, and that was sort of inevitable. But I think that's a great point. And hopefully, I answered your question.

**Sam DeMarco:** Yeah, thank you.

**Aneesh Deoras:** And excuse me, where said "minor" before, I did mean basic. Sorry.

**CDR Kim Piermatteo:** Thanks, Aneesh. And thanks, Sam. Alright, our next question is coming from-- I apologize, I'm not going to say this correctly-- P Zenelaj. I've unmuted your line. Please unmute yourself and ask your question.

**Patricia Zenelaj:** Hi, this is Patricia Zenelaj from Beckman Coulter. My question is within regards to the documentation that we must provide under verification and validation. Under what circumstances would the agency expect the executed protocols or the test scripts or test logs to be submitted along with the summary or the verification reports?

**CDR Kim Piermatteo:** Thanks, Patricia. Justin, would you like to provide a response?

**Justin Post:** Yeah, I can take that. So I think we outlined in the guidance the definition of basic or enhanced. So for enhanced documentation level, we kind of go into, in addition to the system level testing, that we also want to see the unit integration test protocols and reports. So that would be the situation where you have enhanced as defined in the guidance document, where we want to see that information.

Obviously, for any software development project, we expect that that's being documented internally and documented in DHF. What we're requesting for the guidance is based on the risk, this risk-based approach of what we want to see for review to determine safety and effectiveness. Thank you.

**Patricia Zenelaj:** But you wouldn't expect the actual test scripts even under the enhanced level?

**Justin Post:** What do you mean by test scripts?

**Patricia Zenelaj:** Like, the actual scripts from the testing that was done during verification for the software, in addition to the executed protocol. So the printouts from the testing.

**Justin Post:** So we want to see evidence. We want to see the test cases and how those were executed. So you have a protocol that was executed. You want to see, OK, for this particular test case, what was the pass or fail? If it failed, was there a justification for why it failed and that you deemed the overall testing for that-- maybe that test suite to be acceptable.

So it's not just a blank protocol. It's kind of the outcome of that testing and then your final determination on whether or not that particular test passed or failed. That's the kind of evidence that we're looking for in this submission.

**Patricia Zenelaj:** Thank you.

**Justin Post:** You're welcome.

**CDR Kim Piermatteo:** Thank you, Justin. Thank you, Patricia. Next, we have Duane. Duane, I've unmuted your line. Please unmute yourself and ask your question.

**Duane Herberg:** Yeah, hi. Thank you very much for taking my question. My question is about cloud services. In the guidance document in the sample architecture diagram, or the diagrams, cloud services are shown as off-the-shelf software. And my question is, for cloud services that are far removed from directly supporting a device function-- for example, a load balancer to manage internet traffic across multiple services-- would you see such a service as still being off-the-shelf software, even though it's really deep in the technology stack and is more infrastructure?

**CDR Kim Piermatteo:** Thanks Duane. I'm going to turn that over to Aneesh.

**Aneesh Deoras:** Yeah, that's a great point. So I mean, largely, that stuff isn't regulated by FDA for our multiple-function device products and medical device data system policies. So you really wouldn't worry about that. The only case in this architecture diagram is just explaining what's going on. So you wouldn't need to explain load balancing or anything like that. That's not really expected.

To the degree you can explain, and how it's set up in the architecture diagrams is really just enough detail for us to understand, oh, this isn't happening on site. This is happening somewhere else. That's really the point. But no, you don't need to go into that much detail unless it's critical to how you've developed your system. Normally, it's not the case. But that could be relevant, but generally, no.

Yeah, it's a great point. Most of that stuff is not really things that we regulate or have authority over, so you don't need to include that.

**Duane Herberg:** Thank you.

**CDR Kim Piermatteo:** Thanks, Dwayne. And thank you, Aneesh. We have time for one more question. That question is going to come from Aly. I've unmuted your line. Please unmute yourself and ask your question.

**Aly Abayazeed:** Hi. Thank you, Aly Abayazeed. So is it fair to say that any type 2 device is automatically basic documentation? Or there could be a type II device that would require more enhanced documentation. Thank you.

**CDR Kim Piermatteo:** Thanks, Aly, for your question. I'm going to just open this up to the panel. Is anyone on the panel-- Justin, did you want to start?

Sorry. Aly, can you repeat that question? We had some static on our end.

**Aly Abayazeed:** Sure. Can you hear me?

**CDR Kim Piermatteo:** Yes.

**Aly Abayazeed:** OK, is it fair to say that any type II risk device is automatically basic documentation? Or would there be a type II device that would require enhanced documentation? And if so, if you can just give an example. Thank you.

**Aneesh Deoras:** Oh, I see. OK.

**CDR Kim Piermatteo:** OK, yeah.

**Aneesh Deoras:** We understand. You're saying, would a Class II device automatically be basic? Would a Class III automatically be enhanced? Is that right, or?

**Aly Abayazeed:** Yes, yes. I'm focused more on type II, but definitely a category is Class III, yes.

**Aneesh Deoras:** Yeah, so we tried to make it clear in the guidance, actually. So that's not a default decision. So sometimes Class II devices can be enhanced. Sometimes they can be basic. And actually, the same is for class III.

Now, we do suggest in the guidance that if it's Class III, it's likely enhanced. But we don't want to assign that across the board. There are some pieces of software within Class III devices that really aren't high risk.

So, yeah, it's not defined. For class II, you really do have both directions because Class II is so broad. You could have a patient monitor all the way down to certain types of prefilled syringes. So really, a single definition doesn't really work for Class II. So it doesn't give you a solid answer, but I hope that clarifies things.

**Aly Abayazeed:** Yes, thank you.

**CDR Kim Piermatteo:** Thanks, Aneesh. And thank you, Aly. Alright, that wraps up our live Q&A. I want to say thank you to all of you for an engaging question-and-answer segment.

I know that there were a lot of questions still in the queue. So please don't forget, you can email the digital health inbox, which I'll mention in a few minutes, and then also DICE.

But right now, I just want to turn it back over to Sonja for her final thoughts for today. Sonja?

**Sonja Fulmer:** Thanks, Kim. And thanks, everyone, for your questions today. So I hope we've clarified during this webinar that we've worked to find a least-burdensome means for premarket submissions to describe the necessary information about device software functions. This guidance provides a simplified approach that's risk based, where riskier devices should include enhanced documentation above the basic level.

Finalizing this guidance is an important milestone for us as we continue our work to streamline and align FDA review processes with software life cycles for digital health products. And I hope the clarity in this guidance helps you prepare your premarket submissions going forward. I'll turn it back to Kim now to close the webinar.

**CDR Kim Piermatteo:** Thanks again, Sonja, for those final thoughts, as well as your presentation for today. I'd also like to, again, thank Jessica and Aneesh for their presentations and our additional panelists, Samantha, Jason, and Justin. Thank you, guys. Thank you all for your participation in today's webinar.

If you have additional questions about this final guidance document or questions that I said, we didn't get to today, please email the CDRH Digital Health Center of Excellence at digitalhealth@fda.hhs.gov.

And then for your information, printable slides of today's presentation are currently available on CDRH Learn at the link provided on this slide under the section titled "Specialty Technical Topics" and the subsection "Digital Health." A recording of today's webinar and transcript will be posted to CDRH Learn under this same section and subsection in the next few weeks. A screenshot of where you can find these webinar materials is provided on this slide.

We encourage you to attend one of our future CDRH webinars. And you can find a listing of all of our upcoming webinars at the bottom link on this slide as well. For additional questions about today's webinar or an upcoming webinar, feel free to reach out to us in DICE at dice@fda.hhs.gov.

Thank you all again for joining us today. This concludes our webinar and see you next time.

**********

END