High-level documentation provides an overview of a system, software application, or product, without getting into the technical details. It's aimed at a non-technical audience and is designed to provide a broad understanding of the system's purpose, features, and capabilities.

Examples of high-level documentation include:

1. **User manuals**: These provide instructions on how to use a system or software application, often in a step-by-step format. They are designed to be easily understandable by users who may not have technical expertise.
2. **Marketing materials:** These include brochures, flyers, and other materials that are used to promote a system or product. They may include high-level descriptions of the product's features, benefits, and use cases.
3. **Project proposals**: These are documents that outline the purpose, scope, and objectives of a project. They provide a high-level overview of the project's goals and objectives, without getting into the technical details.
4. **Executive summaries**: These are brief summaries of reports or proposals that provide a high-level overview of the key points and findings.
5. **Business requirements document (BRD**): This document outlines the high-level business requirements and objectives for a new product or system. It typically includes information about the target market, business goals, user needs, and other high-level considerations.
6. **System requirements specification (SRS):** This document describes the high-level functional and non-functional requirements for a product or system, including user requirements, system capabilities, and performance requirements.
7. **Project plan**: This document outlines the high-level plan for executing a project, including the project scope, timeline, budget, and resource requirements.
8. **Design brief:** This document provides a high-level overview of the design requirements and goals for a product or system, including the design principles, user experience goals, and design constraints. It is typically used to guide the design process and ensure that the final product meets the intended design goals.
9. **Executive summary**: This document provides a high-level overview of a product or system, including its key features, benefits, and market potential. It is typically used to communicate with executives, investors, or other high-level stakeholders.

Overall, high-level documentation provides a general understanding of a system or product, and is useful for non-technical stakeholders who need to understand the purpose and capabilities of the system.

Low Level:

A low-level document is a technical document that provides detailed information on the design, implementation, and operation of a system, software, or hardware component.

Low-level documents are often created by developers or engineers as a reference for themselves or other team members. These documents typically contain information such as detailed descriptions of algorithms, data structures, system architecture, and software/hardware interfaces.

Low-level documents are essential for understanding how a system or component works and for making modifications or improvements. They are often used as part of the development process, for troubleshooting, and for maintenance and support.

For example, in software development, low level documents may refer to design documents, code documentation, or technical specifications that provide a detailed description of the software's architecture, components, and functions. In engineering, low level documents may include blueprints, schematics, and technical drawings that provide detailed information about the design and construction of a system or product.

In general, low level documents provide granular, detailed information that is intended for use by technical experts or specialists. They may be used to guide development, construction, or maintenance processes, and are often not suitable for general audiences or end-users.

Here are a few examples of low-level documents in technical writing:

1. Technical specifications: This document describes the detailed technical specifications of a product or system, including the materials used, dimensions, tolerances, performance requirements, and other technical details.
2. Design documents: These documents provide detailed information about the design of a system or product, including the architecture, components, interfaces, and relationships between them.
3. Code documentation: This document provides detailed information about the source code of a software application, including the programming language used, the purpose of each function, and the relationships between different modules.
4. Test plans: These documents describe the testing procedures that will be used to ensure that a product or system meets its specifications and requirements.
5. User manuals: While user manuals are typically intended for end-users, they may also include low-level information for advanced users or system administrators. For example, a user manual for a network router might include instructions for configuring advanced network settings or troubleshooting common issues.

High-level documentation typically provides an overview of a system or a product, without going into the technical details. It's aimed at a non-technical audience and may include descriptions of features, user manuals, and marketing materials. High-level documentation is generally less detailed, but it is important for helping users understand the overall purpose and functionality of a system.

Low-level documentation, on the other hand, provides detailed technical information about a system or a product. It's aimed at a technical audience and may include software specifications, API documentation, and programming guides. Low-level documentation is typically more detailed and provides a deeper understanding of how a system works and how to interact with it.

In general, high-level documentation is more accessible and easier to understand, while low-level documentation is more technical and detailed. Both types of documentation are important for different purposes, and they complement each other. High-level documentation provides an overview of a system's capabilities, while low-level documentation provides the technical details needed to implement or integrate with a system.