SOFTWARE DEVELOPMENT INTERFACE TYPES

Contributed by Svetlana Podchasova

INSIGHT

IIBA® International Institute of Business Analysis™

Software Development Interface Types

Different types of communications lead to different types of interfaces. In this document, we will review interface types referring to software development and how the peculiarities of each type should be reflected in the interface specification.

User interface

The user interface is the point of human-computer interaction. This is the space where interaction between humans and machines occurs. It can include the use of display screens, keyboards, a mouse, or other devices.

The user interface design is a specific knowledge area. Often in an initiative, especially when human-computer interaction is crucial or has specific non-functional requirements (NFRs), the dedicated user experience (UX) specialist or user interface (UI) designer is involved. In other initiatives, a business analysis professional assumes these responsibilities.

In most software applications, the user interface is implemented as a Graphical User Interface (GUI). A GUI is a form of interface that allows users to interact through direct manipulation of graphical elements such as windows, menus, and icons. GUI specifications are often replaced with visualizations of screen contents, such as wireframes, prototype screenshots, or UI mockups. This is a simple and powerful approach to describe solutions and receive feedback from stakeholders.

To avoid ambiguity and simplify information, accompanying text descriptions and flow diagrams are useful. They should help the reader understand the relationships between screens and the sequence of actions users must take to accomplish business tasks. In addition, the fields and controls presented on the screen should be listed with a short notation of the control's content and functions.

Below are some examples.

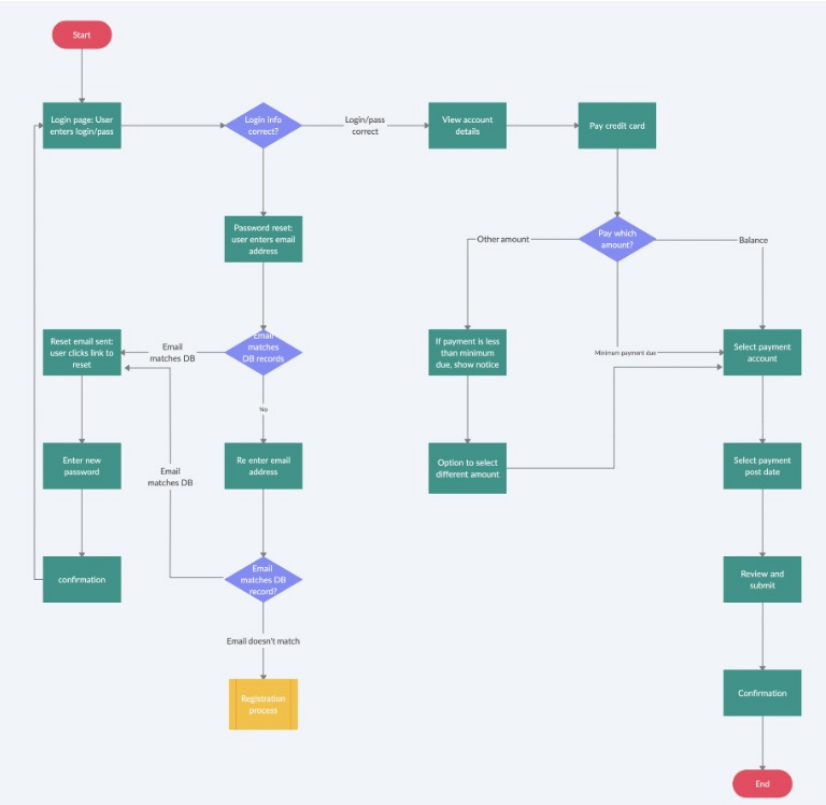

Figure 1: UI Flow Diagram of a Login (Example)[1]

The difference between wireframes, prototypes, and user interface mockups is mainly in the level of elaboration, as seen below.

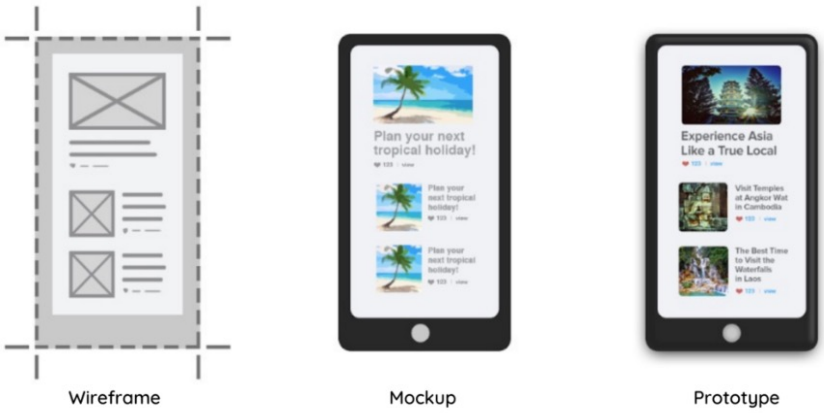

Figure 2: Wireframe, Mockup, and Prototype for the Travelling Platform (Example)[2]

Avoid developing polished screen mockups as implementation details or requirements may change. To minimize time spent on mockup preparation, standards and guidelines are usually incorporated and referenced. They can include generic rules and libraries of reusable components and controls.

Non-user interface

A non-user interface is a point where interactions occur without user involvement. It includes data interfaces between systems, APIs between components and third-party applications, and interfaces with hardware devices.

These types have various subtypes and may require additional technical details to be included in the specification, such as hardware configuration, operating system, timing issues, and failure recovery process. They are visualized below.

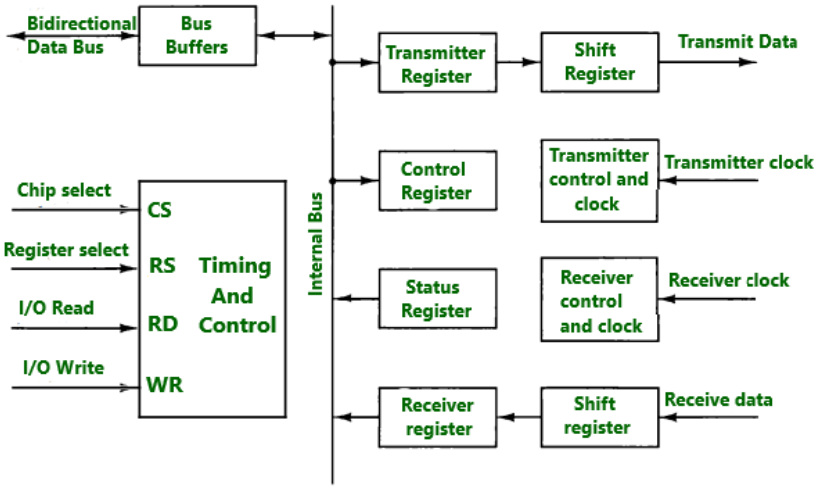

Figure 3: Asynchronous Data Transmitter and Receiver Diagram (Example)[3]

Hardware interface

Hardware interfaces help various hardware devices connect and communicate, as illustrated below.

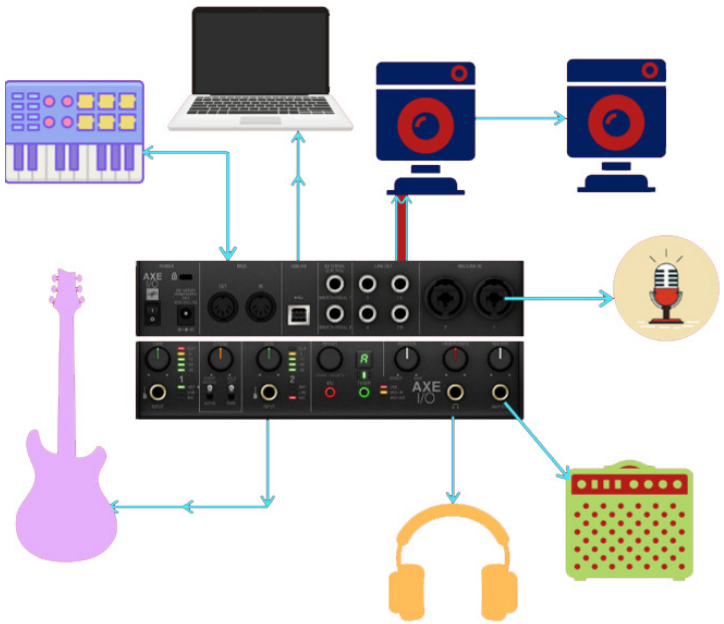

Figure 4: Audio Interface Setup Diagram (Example)[4]

Data interface

Interface descriptions usually focus on the data used in the system and the interaction of components. These are referred to as data interfaces. As part of a formal approach, these primarily apply to non-user interfaces, though certain elements can also be valuable for user interfaces.

Data interface specification should contain:

- Types of data
- Valid data ranges
- Default values
- Validation rules
- Correction algorithm
- Data conversions and transformation procedures (if any)
- Encrypting algorithm
- Transitions between states
- Frequency and period of data refresh
- Rules for things like data invalidation and obsolescence

A data interface diagram presenting types of data, their storage locations, and flows will help users in reading the documentation. Here is an example:

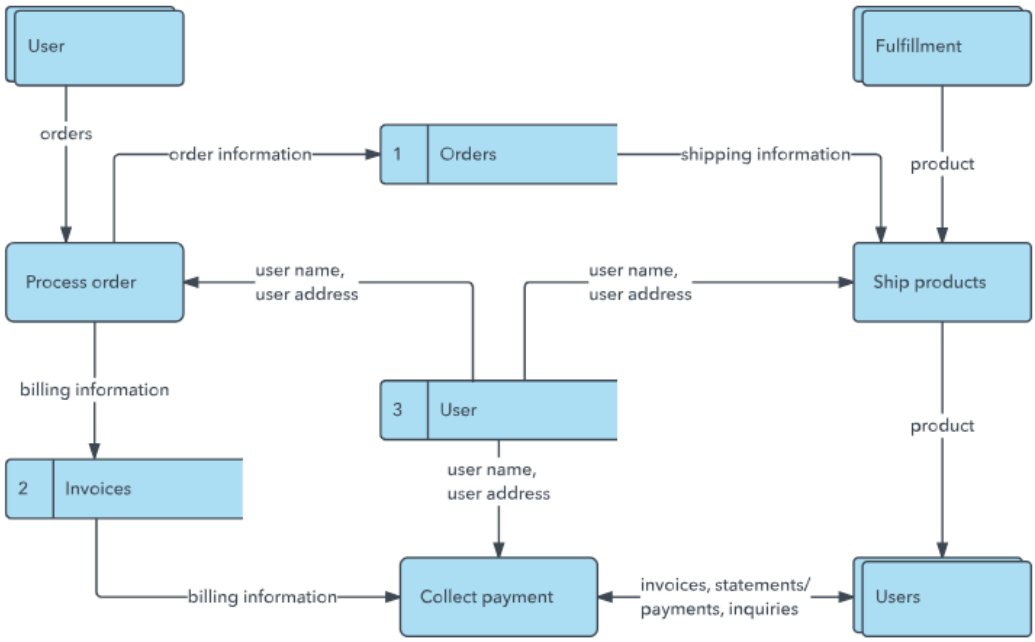

Figure 5: Data Flow Diagram for Client Orders Management (Example)[5]

Message interface

When the interface is implemented through messages exchanged between systems or components, the specification should state:

- Message types
- Formats
- Creation stimulus
- Communication method
- Communication protocol
- Synchronous or asynchronous communication type
- Encoding and security standards

To determine these technical details, contributions should be made by developers, software and hardware engineers, system or solution architects, and external consultants. See the example below:

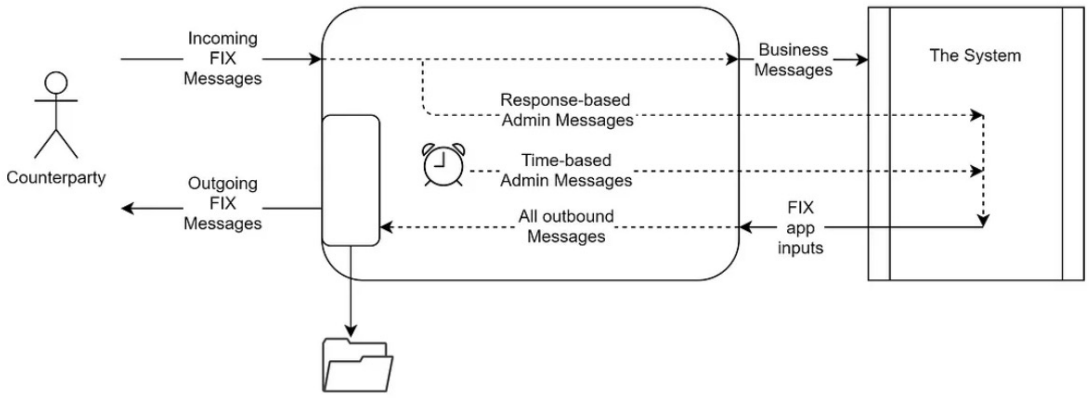

Figure 6: Message Flow Diagram Using FIX Protocol Exchange (Example)[6]

References

1. Creately. “How to Make a User Flow Diagram.” 2022.
2. Code Epsilon Services. “Difference Between Wireframes, Mockups and Prototypes.” N.d.
3. GeeksforGeeks. “Asynchronous Communication Interface.” 2020.
4. Vibeyy. “Audio Interface Setup Diagrams - A Step-by-Step Explanation!” N.d.
5. Lucidchart. “What is a Data Flow Diagram.” N.d.
6. Sanghvi, Prerak. “Proof Engineering: FIX Gateways.” 2021.