Process maps are developed in steps.

- First, specify the **purpose** of the process.
- Next, identify the **participants**.
- Third, map the so-called "**happy path**."
- Fourth, extend the model to show how **exceptions** to the happy path are to be managed.
- Next, reduce visual complexity by linking your process to **sub-processes** where appropriate.
- Finally, add **clarifying details** regarding process elements, resources, and documentation.

For example, let's consider a basic customer support process. The process starts when a support request is received. An instance of the process ends when the support request is closed successfully. The purpose of the process is to handle support requests.

In this example, there are two participants. The first is the customer. The second is the support center. Within the support center, there are two roles. The customer-facing role is the Customer Support Representative. Behind the scenes is the Technical Support Specialist.

The happy path is the scenario that works out in the easiest and best way possible. It's generally free of exceptions and features a normal progression of events, infrastructure, and the "usual" cast of characters. It's presented from the start event to the end event in the level of detail appropriate to the purpose of—and audience for—the model. In this example, a support request is received by the CSR, who identifies a solution, communicates the solution to the customer, and closes the support request.

In the real world, the happy path isn't the only path. Sometimes, exceptions occur. For instance, the CSR might not be able to identify a solution, in which case the assistance of a Technical Support Specialist may be required.

Some activities within a process constitute a sub-process. Depending on the audience, it may be appropriate to show sub-processes in their "collapsed" state to avoid unnecessary complexity.

The basic model can be extended to reflect additional information designed to help the reader better understand the process. These additional details might include specifying event and task types, highlighting the contents of messages between participants, or adding data objects to show how participants interact with components of information systems.

Taken together, our stepwise approach yielded a process map representing the collaboration between a customer and the support center to resolve and close a support request. You can apply the same stepwise approach to developing your process maps.