WHAT IS THIS TOOL?

This tool explains the steps of developing or adapting a coding framework for coding open, unstructured feedback. This coding framework can help you code data, whether you are coding your data in Excel, using machine learning, writing the codes on sticky notes, or using another coding method. It also mentions the need for a codebook, setting one up, and ensuring it is kept up to date.

WHAT DO WE MEAN BY ‘CODING’, ‘COPDING FRAMEWORK’ AND A ‘CODEBOOK’?

***Coding*** is the process of attaching a specific keyword or short phrase to feedback comments to track and analyse them more easily. It helps to better understand the main topics communities are talking about, which allows identifying differences between community groups and locations and changes over time. A ***coding framework*** presents the hierarchy of codes we are using for this process and shows how they are grouped into categories under the different types of feedback. Once you have your coding framework, you will develop a ***codebook***. This document describes the different types, categories and codes and explains how to use them. It is the reference document for everyone using the coding frame and ensures everyone codes the data the same way.

HOW TO USE THIS TOOL?

This tool should be used in conjunction with the module 4 of the Feedback Kit. Besides developing your framework, you will need to decide on which method you are going to use for coding your data (see [this tool](https://docs.google.com/document/d/1fukQjwD4Ia0r0FKcK-vDcjwebJV1dbWB/edit) on this). To test and adapt your framework, you will need:

* **guidance on how to code** the feedback data, this is provided in [this tool](https://docs.google.com/document/d/1G4r563U080m_p0c-TeekjCx6uVIW44xJ?rtpof=true&authuser=evaelisabeth.erlach%40gmail.com&usp=drive_fs)
* some **feedback data** that is similar to the data you will be coding in the future, as this is what you will tailor your coding framework to
* *optional*: the [**template coding framework**](https://docs.google.com/spreadsheets/d/1CSy6kTUed47r864FNp4j4prAyZhS6kkf?rtpof=true&authuser=evaelisabeth.erlach%40gmail.com&usp=drive_fs): this is a basic coding framework that can serve as a starting point for organisations receiving any type of community feedback. You can use and adapt this to your context.

# Think about what community members mainly share with you

First, you need to understand what the most common themes and topics are that community members will share with you. To do so, you can read through the feedback comments you have already collected and see if there are certain themes in the data that are coming up frequently. You can also discuss this with your colleagues who are in regular contact with communities and have been receiving community feedback.

# Adapt the template coding framework, or create your own

Now you have two options. You can either use and adapt the template coding frame or create your own coding framework from scratch. Below you can find guidance for both:

Adapting the template coding framework:

[This template coding framework](https://docs.google.com/spreadsheets/d/1CSy6kTUed47r864FNp4j4prAyZhS6kkf?rtpof=true&authuser=evaelisabeth.erlach%40gmail.com&usp=drive_fs) provides a basic structure as a starting point. It is based on the frameworks used in many different contexts and can further be refined to your specific context.

The coding framework consists of three levels of codes:

1. **Types of feedback** – which tells us if the feedback is a question, a request or suggestion, a report of a concern or incident, encouragement or praise, or a more general observation, perception or belief
2. **Categories** – this groups together the specific codes and tells us the general topic, e.g., “Requests for support”
3. **Codes** – this is the most specific level, and describes the specific topic the person is talking about, e.g., “Request for face masks”, “Request for soap”

On the second tab of the Excel file, you will find the codebook, which describes all the different types, categories, and codes and provides specific examples.

You can go through the categories and codes and take out what is not relevant in your situation. Again, this should be a discussion as part of setting up the feedback mechanism and include technical colleagues, colleagues regularly in touch with community members, and may involve community members themselves.

For example:

If you would like to use the coding framework in a non-emergency context, categories such as “Comments on the scope and scale of disaster”, or “Questions about the crisis” will not be relevant and you can take them out.

There may be important issues and trends in your context that are not represented in the template coding framework. You can add codes, and additional categories, if they are currently missing.

For example:

Maybe you are working on a health-related project, and lots of comments will relate to the health care system. In that case, you might want to consider adding a new category for “Questions about health care” and “Observation, perception, belief about health care”.

At this stage, your coding frame does not have to be completely accurate, and it will never be perfect. All you need is a basic structure, and you can still tailor it to what communities will share with you as you go.

Once you have your basic coding frame, you can consider changing the wording of the categories and codes to fit your context and ensure they are easily understood.

For example:

Suppose you are using “Questions about preventative behaviours”, but you always use the term “safe and healthy practices” in your programme. In that case, you might want to change it to “Questions about safe and healthy practices” to make sure everyone will know what this refers to.

Creating your own coding framework:

If you have data that refers to a specific context or topic, and you feel that the template coding frame does not capture the topics properly, you can develop your codes from scratch. All you need to do is go through your data and/or discuss the main topics with your colleagues and create codes for the topics.

It is important to note that we are creating codes for topics that people are talking about. That means that the feedback comments include similar ideas or concepts, which might be formulated in different ways. Two people could be talking about the same topic but use different words.

*For example, all the following comments are talking about the same topic, but in different ways:*

* *“You should hire more local staff”*
* *“I think you should be working with people like myself, who know the local culture.”*
* *“It would be good if your organisation hired more people from around here.”*
* *“Don’t fly in doctors from abroad, we have great and competent people who can support the emergency response.”*

*All these comments can be grouped together using the code “Suggestions to hire more local people”*

When creating codes, you should be cautious not to create too many specific codes (as this will make your coding framework too big and complex), as well as not to create codes that are too broad and won’t be informative enough. It is always good to have codes that immediately tell you what the topic is. Sometimes you might have to start using a broader code in the beginning but can split the code in two at a later stage, if more and more people talk about a sub-topic.

For example:

You might see a lot of questions from people about how they can protect themselves against a disease. They might be sharing that they don’t know how to protect them, some will ask if it is good to keep a distance from other people, and others might be asking about how to best wash your hands.

For this, you can use a code such as “Questions about protective behaviours”

If at a later stage, you might realise that most of the questions refer to hand washing, such as people asking if you need clean water for handwashing, if you need to use soap, if you can use ash, and how often you are supposed to wash your hands. If this is the case over a longer period of time, you can introduce a new additional code, such as “Questions about hand washing”

You can then use both codes, with all questions relating to hand washing coded under “Questions about hand washing”, and any other questions about other protective behaviours under “Questions about protective behaviours”.

Another important consideration is to think if you want to capture both positive and negative comments about a certain topic.

For example:

You might see a lot of feedback comments about the denial of a disease, such as “The disease you are talking about is not real”, “This disease does not exist”. Other people might say “The disease is real and a real threat”.

These comments say the contrary. The same can be the case when people talk about people following the rules, and others say people do not comply with rules. Or people accepting the vaccine and others saying they do not accept the vaccine.

In such cases, it is recommended to use separate codes for these opposites. This will make it easier to track the different views. You don’t always have to create both. If 100 people say the disease does not exist and two people say it exists, you can have a code for “Belief that the disease does not exist” and do not need a code for “Belief that the disease does not exist”. The two comments about the denial of the disease can be coded under “Other” in the meantime. If it becomes a common topic, you can introduce the code at a later stage.

You can then make a list of the topics, for example:

|  |
| --- |
| **Codes** |
| * Concerns about the location of the latrines
 |
| * Questions about how to maintain the latrines
 |
| * Suggestions to hire local staff
 |
| * Reports of the latrines being broken
 |
| * Questions about remuneration of the community committee
 |

Once you have your list, you can start using these codes and refine them as you go. In case the list gets too long, you can group them into categories:

|  |  |
| --- | --- |
| **Category** | **Code** |
| Latrines | * Concerns about the location of the latrines
 |
| * Questions about how to maintain the latrines
 |
| * Reports of the latrines being broken
 |
| Staffing and remuneration | * Suggestions to hire local staff
 |
| * Questions about remuneration of the community committee
 |

If your list grows further and you need to structure your data even better, you can add the type of feedback in your coding frame[[1]](#footnote-1):

|  |  |  |
| --- | --- | --- |
| **Type** | **Category** | **Code** |
| Questions | Questions about latrines | * Questions about how to maintain the latrines
 |
| * Questions about remuneration of the community committee
 |
| Suggestions | Suggestions about staffing and remuneration | * Suggestions to hire local staff
 |
| Reports | Issues related to latrines | * Reports of the latrines being broken
 |
| * Concerns about the location of the latrines
 |

# Update or develop your codebook

A codebook is the reference document with all the descriptions of the types, categories and codes you are using. It can also include examples and be used for tracking changes of codes you might be making. This is essential for ensuring that the coding will always be done consistently and that everyone involved in the coding process has the same understanding of the codes.

Updating the template codebook:

You can find the template codebook on the second tab of [this file](https://docs.google.com/spreadsheets/d/1CSy6kTUed47r864FNp4j4prAyZhS6kkf?rtpof=true&authuser=evaelisabeth.erlach%40gmail.com&usp=drive_fs). You will see that it has the same structure as your coding framework but includes additional information in the columns next to the types, categories and codes. You can now make the changes you made on the first tab here as well, edit the descriptions, add to them, and add examples from your context.

Developing your own codebook:

If you are not working with Excel, and/or want to use a different coding framework, you can make your own codebook from scratch. You can create the codebook in Excel, a Word file, or any format you prefer.

The following format can be used for documenting the descriptions of the codes, providing example comments, and documenting changes.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Code | Description | Example feedback comments | Last changed | Comments |
| Eg:Request for information about cash | Eg:Requests to provide information about the cash transfers, including about selection criteria, and questions about when and where distributions are taking place. | Eg:When will the next distribution take place?You need to tell us where to go to receive the transfer.Why are only people whose houses have been destroyed receiving cash transfer?  | Eg:Introduced on 03.11.2021 | Eg:Do not code requests for information on other relief distributions under this code, they are coded under “request for information about distributions”  |

# Test it

Now you can start coding feedback data with your new coding frame. See [this tool](https://docs.google.com/document/d/1G4r563U080m_p0c-TeekjCx6uVIW44xJ/edit?rtpof=true&sd=true) for guidance on how to code data. Whenever there are comments, and you do not have a code that describes the statement, you can code them as “other”.

Remember that your coding framework is a flexible system and will never be perfect. You can still refine it, as we see in the next step.

# Refine the coding framework

Once you have coded a good amount of data (data collected over a period of at least 2 to 3 weeks, or at least a few hundred feedback comments), you can go through the comments coded as “other”. If specific topics emerge in “other”, you can consider adding an additional code to capture them. “Other” should never be one of the most commonly used codes.

You should also review the codes you have, to see if they are accurately capturing the topics shared by community members. You can further refine the code and adapt the wording. Your coding framework is a flexible system and should regularly be reviewed and adapted to the feedback that communities share with you.

Please see the section above on “[Creating your own coding framework](#_heading=h.gjdgxs)”, which provides information on new codes. It is important to keep the right balance between keeping your coding frame simple and concise, and having the codes you need to best describe the topics community members share with you.

In case there are several new codes related to one topic, you can also create a new category to group them together.

# Keep the codebook updated

Make sure to document all changes in the codebook, add example comments, and add the date of when the changes were made. The example comments and descriptions in the codebook are key to ensuring consistency in the coding. Documenting the changes of codes you are making (for example when you introduce a new code, or add a new one) is important as well, as you might be in a situation where you need to go back in the data and adjust the coding or you need to update the coding frame in different documents.

# Have regular discussions with anyone involved in coding the data

Organise regular meetings with anyone involved in coding the data to discuss the categories and codes and agree on necessary changes. Everyone using the same coding framework needs to have the same understanding of the coding. A coding frame will never be perfect as it is always a compromise to group similar comments together, but if the coding is consistent, it is possible to identify trends and track changes over time.

1. See more information on the types of feedback in the introduction of Module 4 of the Feedback Kit. [↑](#footnote-ref-1)