# Explainer Rubric

CS 7480: Categories for PL, Fall 2025
Steven Holtzen and John M. Li

October 28, 2025

## 1 Overview

As the final project for this course you will be creating an *explainer* that serves as a self-contained introduction to a topic. This explainer will be written in the style of the course notes and will build on the material in them to explore a new topic. Our goal is ultimately for these explainers to form the final part of the course notes and serve as a reference for future students interested in these topics. We encourage you to take this as an opportunity to contribute meaningfully to a useful resource for future students interested in learning category theory. It will consist of the following components:

- The *main body* will be a chapter written in the style of the course notes. It will motivate the topic, provide background on it, and pedogically introduce the main ideas through detailed examples. The main body is expected to correspond to the amount of material presentable in roughly two 100-minute lectures, and should be about 15-20 pages in length in the course notes latex template.
- The *exercises portion* will provide 3-5 exercises along with solutions that reinforce key concepts introduced by the explainer.

You may work with a partner on your explainer, but if you do, you are expected to provide a significantly more detailed explainer than those working alone, and you will be held to a higher standard for the final product. If working with a partner, we expect roughly double the length of explainer and roughly double the number of exercises.

## 2 Timeline

- By **November 7**, upload to Gradescope a roughly 1-page *explainer proposal* that contains the following information:
  - *Overview*: A 1-paragraph overview of your topic and why you chose it.
  - *Prerequisites*: A 1-paragraph summary of the prerequisites to your topic. These are things you are assuming anyone reading your explainer to already know. They should refer to specific chapters/sections of the notes, or you should be prepared to provide material on these prerequisites for your explainer.

  - *Possible homework problems*: A list of potential exercises that you expect someone who reads your explainer to be able to solve after they read it. Briefly justify why each exercise was chosen (i.e., which concept it is meant to reinforce or explore).
  - *Resources*: A list of resources you will rely on for preparing your explainer (they can be online lecture series, notes, surveys, etc.)
  - *Team*: If working with a partner, state so. Otherwise, say you are working by yourself.

- The final explainer is due **December 8**. You will submit a PDF copy of your explainer to gradescope, along with a pull request for your explainer to the main note document.

## 3 Evaluation

We will evaluate your explainer according to the following criteria:

- *Self-containedness*: Does the explainer contain within it all required material for understanding it? Does it clearly situate itself within the broader context of the course notes?
- *Pedagogy*: Are there adequate worked examples? Are topics broken down in sensible ways, and are key ideas identified and clearly presented? Is a reasonable sequence of topics? Are appropriate figures and visualizations used to make the presentation easier to follow?
- *Writing quality*: Is the writing clear, easy to follow, and free from significant grammatical mistakes?
- *Technical clarity*: Are there technical errors or typos in mathematical statements? Are key theorems, definitions, and lemmas stated clearly?
- *Reference quality*: Are relevant related works cited and contextualized?
- *Exercise quality*: Do the exercises properly reinforce the topics covered in the explainer? Are there a variety of exercise difficulties? Do the solutions to the exercises provide adequate explanation?

Ultimately, the explainer will receive a score of *Check* if we feel it is ready to be included into the notes with minor modifications; *Check plus* if it is ready to be included into the notes with no modifications; and *Check minus* if it would require significant effort to include in the course notes. Check and check-plus are considered full marks.

## 4 Suggested topics

You are encouraged to choose a topic that is relevant to your research or you are interested in personally. Here are some suggested topics you may consider. Feel free to post on Discord or reach out to the instructors if you need more help selecting a topic.

Here are some example topics we've thought of, but you are encouraged to think of your own:

- Decipher ”First Steps in Synthetic Guarded Domain Theory: Step-Indexing in the Topos of Trees” [Birkedal et al., 2012]

- The Day convolution in separation logic [Dongol et al., 2016, Day, 2006, Li et al., 2024]
- Monads and algebras: effectful programming edition [Moggi, 1991, Bauer, 2018, Plotkin and Pretnar, 2013]
- Monoidal closed categories as models of linear logic [Benton and Wadler, 1996]
- Quantum categorical semantics [Baez, 2006, Abramsky and Coecke, 2009, Selinger, 2004]
- A topos as a model of dependent type theory [Seely, 1984, Maietti, 2005]
- Nominal sets [Pitts, 2013]
- Categorical probability (Steven will likely also choose this!): [Heunen et al., 2017, **?**]

## References


Samson Abramsky and Bob Coecke. Categorical quantum mechanics. *Handbook of quantum logic and quantum structures*, 2:261–325, 2009.

John Baez. Quantum quandaries: a category-theoretic perspective. *The structural foundations of quantum gravity*, pages 240–265, 2006.

Andrej Bauer. What is algebraic about algebraic effects and handlers? *arXiv preprint arXiv:1807.05923*, 2018.

Nick Benton and Philip Wadler. Linear logic, monads and the lambda calculus. In *Proceedings 11th Annual IEEE Symposium on Logic in Computer Science*, pages 420–431. IEEE, 1996.

Lars Birkedal, Rasmus Ejlers Møgelberg, Jan Schwinghammer, and Kristian Støvring. First steps in synthetic guarded domain theory: step-indexing in the topos of trees. *Logical Methods in Computer Science*, 8, 2012.

Brian Day. On closed categories of functors. In *Reports of the Midwest Category Seminar IV*, pages 1–38. Springer, 2006.

Brijesh Dongol, Ian J. Hayes, and Georg Struth. Convolution as a unifying concept: Applications in separation logic, interval calculi, and concurrency. *ACM Trans. Comput. Logic*, 17(3), February 2016. ISSN 1529-3785. doi: 10.1145/2874773. URL `https://doi.org/10.1145/2874773`.

Chris Heunen, Ohad Kammar, Sam Staton, and Hongseok Yang. A convenient category for higher-order probability theory. In *2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–12. IEEE, 2017.

John M Li, Jon Aytac, Philip Johnson-Freyd, Amal Ahmed, and Steven Holtzen. A nominal approach to probabilistic separation logic. In *Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 1–14, 2024.

Maria Emilia Maietti. Modular correspondence between dependent type theories and categories including pretopoi and topoi. *Mathematical Structures in Computer Science*, 15(6):1089–1149, 2005.

Eugenio Moggi. Notions of computation and monads. *Information and computation*, 93(1):55–92, 1991.

Andrew M Pitts. *Nominal sets: Names and symmetry in computer science*. Cambridge University Press, 2013.

Gordon D Plotkin and Matija Pretnar. Handling algebraic effects. *Logical methods in computer science*, 9, 2013.

Robert AG Seely. Locally cartesian closed categories and type theory. In *Mathematical proceedings of the Cambridge philosophical society*, volume 95, pages 33–48. Cambridge University Press, 1984.

Peter Selinger. Towards a quantum programming language. *Mathematical Structures in Computer Science*, 14(4):527–586, 2004.