



# Universal Design of Interactive Mathematical Notebooks on Programming

Bin Guo  
McMaster University  
Canada  
guob15@mcmaster.ca

Jason Nagy  
McMaster University  
Canada  
nagyj2@mcmaster.ca

Emil Sekerinski  
McMaster University  
Canada  
emil@mcmaster.ca

## ABSTRACT

This work presents the rationale behind tools and a guideline for the Universal Design of Jupyter notebooks containing programs, explanations, graphics, algorithms, and proofs, all of which may have mathematical symbols. The tools qualitatively improve accessibility and ease the authoring of such notebooks at the same time. The tools and guidelines are currently being used for a course on concurrent system design and a course on formal languages and compiler construction at McMaster University.

## CCS CONCEPTS

• **Social and professional topics** → **Software engineering education**.

## KEYWORDS

Jupyter notebook, accessibility, alt-text, mathematical symbols

### ACM Reference Format:

Bin Guo, Jason Nagy, and Emil Sekerinski. 2022. Universal Design of Interactive Mathematical Notebooks on Programming. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education V. 2 (SIGCSE 2022)*, March 3–5, 2022, Providence, RI, USA. Providence, Rhode Island, USA, 1 page. <https://doi.org/https://doi.org/10.1145/3478432.3499102>

## 1 INTRODUCTION

Jupyterlab and Jupyterhub is open-source web-host software that allows accessing a remote file system and can be accessed through a web browser<sup>1</sup>. A Jupyter notebook combines prose, executable code, and output of execution into one document. It allows a loop of reading, evaluation, and printing, where the output can be textual or graphical. Jupyter can also be customized to users' needs by the installation of extensions.

*Universal Design* is a “way of thinking” to ensure that all features can be performed by all users regardless of ability<sup>2</sup>. Universal Design identifies “barriers” which obstruct normal operation for some users. The goal of Universal Design is to anticipate, plan around, and accommodate all barriers, such as limited vision or poor motion

<sup>1</sup><https://jupyter.org/about>

<sup>2</sup><http://universaldesign.ie/what-is-universaldesign/definition-and-overview>

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).  
SIGCSE 2022, March 3–5, 2022, Providence, RI, USA.  
© 2022 Copyright held by the owner/author(s).  
ACM ISBN 978-1-4503-9071-2/22/03.  
<https://doi.org/https://doi.org/10.1145/3478432.3499102>

control with effective and non-stigmatizing methods. Jupyter noticeably have accessibility issues related to screen reader usability, and keyboard usability, which is not in compliance with WCAS<sup>3</sup>.

This work draws from the experience of using Jupyter notebooks in teaching courses on introduction to programming, concurrent system design, and formal languages and compiler construction in suggesting how to improve the Universal Design of notebooks to make using notebooks easier for both authors and readers. You can try out our notebook<sup>4</sup> or install the JupyterLab extensions<sup>5</sup>.

## 2 OVERVIEW

*Math Symbols* are commonly used for notions in computing and software. Jupyter notebooks allow LaTeX code for mathematical expressions in markdown cells. The drawbacks are that the LaTeX command cannot be consequently spelled out by screen readers and cannot work inside the markdown code block. An alternative is to use Unicode math symbols. We developed a Jupyter extension that allows a set of pre-configured Unicode characters to be inserted through a menu item, a command, a context menu, or a keyboard shortcut.

*Diagrams* are ubiquitous in course notes. Images can be inserted in Jupyter notebooks. One issue is that the alt-text used by screen readers is not always kept in sync with the image; the other issue is that images have to be in separate files in a subfolder. We offer a solution to this for diagrams (not pictures). Our tools take a declarative description of a diagram and render that as an SVG image while at the same time generating alt-text for screen readers, thus ensuring that alt-text is always exact. The tool supports graphs like flowcharts, trees, sequence diagrams, Gantt diagrams, state diagrams, entity-relationship diagrams, requirement diagrams, and others, as supported by mermaid<sup>6</sup>.

*The principles of Universal Design* should be incorporated into the development process from the beginning to ensure the best accessibility to created notebooks. We summarize a checklist covering common accessibility issues which should be considered when creating a Jupyter notebook.

## 3 CONTRIBUTIONS

The Jupyter extensions are available for JupyterLab and JupyterHub. They qualitatively simplify the authoring of notebooks and make them accessible by default. Our checklist further assists in authoring notebooks. We are currently working on improving the layout of diagrams produced by mermaid and the shortcuts of the keyboard.

<sup>3</sup><https://github.com/jupyterlab/jupyterlab/issues/9399>

<sup>4</sup><https://mybinder.org/v2/gh/nagyj2/jupyternotebooks/HEAD>

<sup>5</sup><https://gitlab.cas.mcmaster.ca/nagyj2/unicodesnippets-ts>

<sup>6</sup><https://mermaid-js.github.io/mermaid/>