

# Haskell for Everyone: An IDE Created by Students

Rajib Bastola	Barry Burd	Robert Juall	Joseph T. Pojero
Drew Univeristy New Jersey, USA	Drew Univeristy New Jersey, USA	Drew Univeristy New Jersey, USA	Drew Univeristy New Jersey, USA
rbastola@drew.edu	bburd@drew.edu	rjuall@drew.edu	jpojero@drew.edu

David Rosenblum  
Drew Univeristy  
New Jersey, USA  
drosenblum@drew.edu

As part of the assigned work for a Computer Science Capstone course, four students created an IDE to be used in their Functional Programming course. The courses ran concurrently during the Spring 2019 semester at Drew University in Madison, New Jersey, USA. The IDE has many features of an industrial-strength development tool, including a code-sharing option to help programmers work simultaneously on Haskell source files.

## 1 Introduction

In this informal report, we describe an IDE named Haskell for Everyone. The IDE was created by four students who were taking a Haskell course (CSCI 335: Functional Programming) and a software engineering-oriented Capstone course (CSCI 400). Four of the report's authors (R. Bastola, R. Juall, J.T. Pojero and D. Rosenblum) are the students who created the IDE. The other author (B. Burd) is the instructor of the Haskell course, who also served as the client for the students' Capstone course project. To avoid confusion about the various authors' roles, this paper contains many third-person references such as "the students" (or "they") and "the instructor" (or "he"). When it's used, the pronoun "we" refers to all five of the authors.

## 2 Background

The instructor taught Haskell for junior and senior undergraduates during the spring 2017 semester. In that course, the official IDE was IntelliJ IDEA [1] with a plugin for Haskell. Plugins of this kind are surprisingly difficult to install, and first week of the course was consumed with installation on the student-owned laptops.

The course was offered a second time during the spring 2019 semester. Before the start of the semester, the instructor found that the IDEA plugin that had been used in 2017 wasn't compatible with the latest version of IntelliJ IDEA. Two other Haskell plugins (IntelliJ-Haskell [2] and HaskForce [3]) claimed compatibility with IntelliJ IDEA, but neither of those plugins could be installed successfully. (The plugins' maintainers were as helpful as they could be but, in the end, nothing worked.)

Haskell plugins for the Eclipse IDE [4] were equally frustrating. The Frege programming language [5] is almost identical to Haskell, but Frege runs on the Java Virtual Machine, and Frege programs can use classes written in Java. There is a Frege plugin for Eclipse [6], but when investigating Eclipse plugins, the instructor found that compiling and running code with the Frege plugin required several repetitive

steps. Another variant of Haskell for the Java Virtual Machine, the variant named Eta [7], had no publicly available IDE.

The emphasis of Drew University's Haskell course is on pure functional thinking with no attention to Haskell's package management libraries (cabal and stack) except what is absolutely necessary. So with all of its power, the well-known Leksah IDE [8] seemed to be too complicated for novice Haskell programmers.

Other alternatives included the VS Code Haskell plugin [9], plugins for Sublime Text [10] and other such editors, and the venerable old command windows for Windows, Macintosh and Linux. But each was ruled out for one reason or another.

Near the start of the semester, two IDEs, Haskell for Mac and HEAT, were serious contenders. Haskell for Mac [11] is an intuitive IDE that's well suited for use by novice Haskell programmers. But, as its name suggests, Haskell for Mac is not a cross-platform application. Haskell for Mac has no counterpart for students whose laptops run Windows or Linux. Besides that, Haskell for Mac isn't freeware.

In the end, the HEAT IDE [12], developed at the University of Kent, seemed to be the best choice. HEAT is written in Java, so students with Windows, macOS and Linux can run it. HEAT has a few quirks that make startup a bit confusing, but once a student becomes accustomed to the flow, the eccentricities of HEAT fade into the background.

### 3 Developing a New IDE

The spring 2019 offering of the Haskell course began in mid-January. In the meantime, students in the Computer Science Capstone course were charged with finding a client to work with on a real-life, semester-long project. With all of its simplicity, the HEAT IDE has some noticeable limitations. The latest version of HEAT was released in 2013, so HEAT is missing some of the convenient features of newer IDEs. With this in mind, four students who were taking both the Haskell course and the Capstone course decided on their own to "develop a beginner-friendly Haskell code editor with an educational focus." They named it *Haskell for Everyone*. Naturally, the Haskell course instructor became the client.

The stated goals for the project were as follows:

- Haskell for Everyone will be a cross-platform desktop application.
- It will be centrally distributed and easily available to download.
- It will be easy to install.
- It will have robust file system integration with support for all CRUD operations.
- It will support easy file switching with a sidebar that displays the `.hs` and `.md` files in one or more folders.
- Help and documentation will be readily available.
- The user interface will have easy-to-use customization features with a choice of themes, font sizes and styles, and general component size adjustment.
- The editor will support syntax highlighting.

In addition to the regularly scheduled course periods (in both the Functional Programming course and the Capstone course), the Haskell for Everyone students met frequently with their client. Discussions included goals for the next iteration of the ongoing project.

To create the IDE, students used the following technologies:

- React [13] and Bootstrap [14] for the UI
- The MobX React architecture [15]
- Electron [16] and Electron-Builder [17] to package the IDE as a desktop application
- Node.js [18], Express [19] and MongoDB [20] for the web server
- WebSockets [21] for online collaboration rooms
- TypeScript [22] for the server
- Heroku [23] for delivery of the software to users

The Haskell for Everyone source code is available as a Git repository [24].

## 4 Using the Haskell for Everyone IDE

Installing Haskell for Everyone is a simple two-step process:

1. Download and install the Haskell Platform [25].
2. Download and install the IDE [26].

On first launch, Haskell for Everyone is ready to use. The IDE automatically finds the path to the computer's Haskell Platform installation. No configuration is required. On subsequent launches, Haskell for Everyone connects to a server that checks for application updates. If an update is available, the user sees a popup window with advice about installing the newest version.

The IDE's user interface is shown in Figure 1. In that interface, the lower right panel houses the IDE's real-time, online collaboration feature. When a student clicks the Create button, a Create Room dialog appears (see Figure 2). When the student clicks the dialog box's Create button, the student's editor window becomes sharable. Other students joining the same room see the same code in their editor windows. There are options to make the code editable or non-editable by newcomers to the room. In a classroom setting, an instructor can create a room to demonstrate new coding techniques. Each student can run the code by clicking the Run icon near the top of the screen.

## 5 Evaluating the IDE

As the main client for the Capstone project, the Functional Programming course instructor used Haskell for Everyone for all classroom instruction during the second half of the semester. Students in the Functional Programming course used the IDE in the final weeks of the semester. They gave immediate verbal feedback to the students who developed the project, and gave written feedback using a Google Form. On the positive side, several students who tested the IDE made similar comments:

- They appreciated the ability to customize the look of the UI.
- They liked the modern, familiar programming interface.
- They liked the ability to share code interactively.
- They were happy to have a list of files in the window's sidebar.

On the negative side, students reported the following problems:

- The IDE’s panels are adjustable in size, but the panels can’t be moved around within the window.
- There isn’t appropriate autofocus to parts of the window at appropriate times.
- When the user selects certain font sizes, there can be a problem scrolling to the bottom of the editor window.
- Useful as they are, the online collaboration rooms can be a bit buggy.

Overall, 90% of the respondents said that they would prefer to use Haskell for Everyone over HEAT. We call that a win.

## 6 Future Work

Three of the four students who created Haskell for Everyone were graduating seniors. They’ve volunteered to continue refining the IDE after they graduate (as time permits, of course). The instructor will be using Haskell for Everyone as the main IDE in subsequent offerings of the Functional Programming course, and will use the IDE for one or more public workshops to be offered in the coming months.

We will continue to maintain and grow the Haskell for Everyone IDE. In future Functional Programming courses, the instructor will formally evaluate the IDE’s effectiveness as a tool for teaching Haskell.

## References

- [1] *IntelliJ IDEA (Ultimate Edition)*, Copyright ©2000-2019 JetBrains. Available at <https://www.jetbrains.com/idea/>.
- [2] *IntelliJ-Haskell*. Available at <https://plugins.jetbrains.com/plugin/8258-intellij-haskell>.
- [3] *Haskforce: The Haskell Plugin for IntelliJ IDEA*. Available at <http://haskforce.com/>.
- [4] *Eclipse*, Copyright ©Eclipse Foundation. Available at <https://www.eclipse.org/>.
- [5] *The Frege Programming Language*. Available at <https://github.com/Frege/frege>.
- [6] *The Frege Eclipse Plugin*. Available at <https://github.com/Frege/eclipse-plugin>.
- [7] *Eta Programming Language*, Copyright ©TypeLead. Available at <https://eta-lang.org/>.
- [8] *Leksah: Haskell IDE in Haskell*. Available at <http://leksah.org/>.
- [9] *Visual Studio Code*, Copyright ©2019 Microsoft. Available at <https://code.visualstudio.com/>.
- [10] *Sublime Text*, Copyright ©Sublime HQ Pty Ltd. Available at <https://www.sublimetext.com/>.
- [11] *Haskell for Mac*, Copyright ©2019 Applicative. Available at <http://haskellformac.com/>.
- [12] *HEAT: The Haskell Educational Advancement Tool*, Copyright ©University of Kent. Available at <https://www.cs.kent.ac.uk/projects/heat/>.
- [13] *React A JavaScript library for building user interfaces*. Available at <https://reactjs.org/>.
- [14] *The Bootstrap Library*. Available at <https://getbootstrap.com/>.
- [15] *MobX*. Available at <https://mobx.js.org/>.
- [16] *Electron: Build cross platform desktop apps with JavaScript, HTML, and CSS*. Available at <https://electronjs.org/>.
- [17] *electron-builder*. Available at <https://www.electron.build/>.
- [18] *Node.js*. Available at <https://nodejs.org/>.
- [19] *Express: Node.js web application framework*. Available at <https://expressjs.com/>.

- [20] *MongoDB*. Available at <https://www.mongodb.com/>.
- [21] *WebSockets*. Available at <https://www.websocket.org/>.
- [22] *TypeScript: JavaScript that scales*. Available at <https://www.typescriptlang.org/>.
- [23] *Heroku: Cloud Application Platform*. Available at <https://www.heroku.com/>.
- [24] *Haskell for Everyone: An easy-to-use educational editor for learning the Haskell programming language*. Available at <https://github.com/RJuall/haskell-for-everyone>.
- [25] *Haskell Platform*. Available at <https://www.haskell.org/platform/>.
- [26] *Haskell for Everyone*. Available at <http://www.haskellforeveryone.com/>.

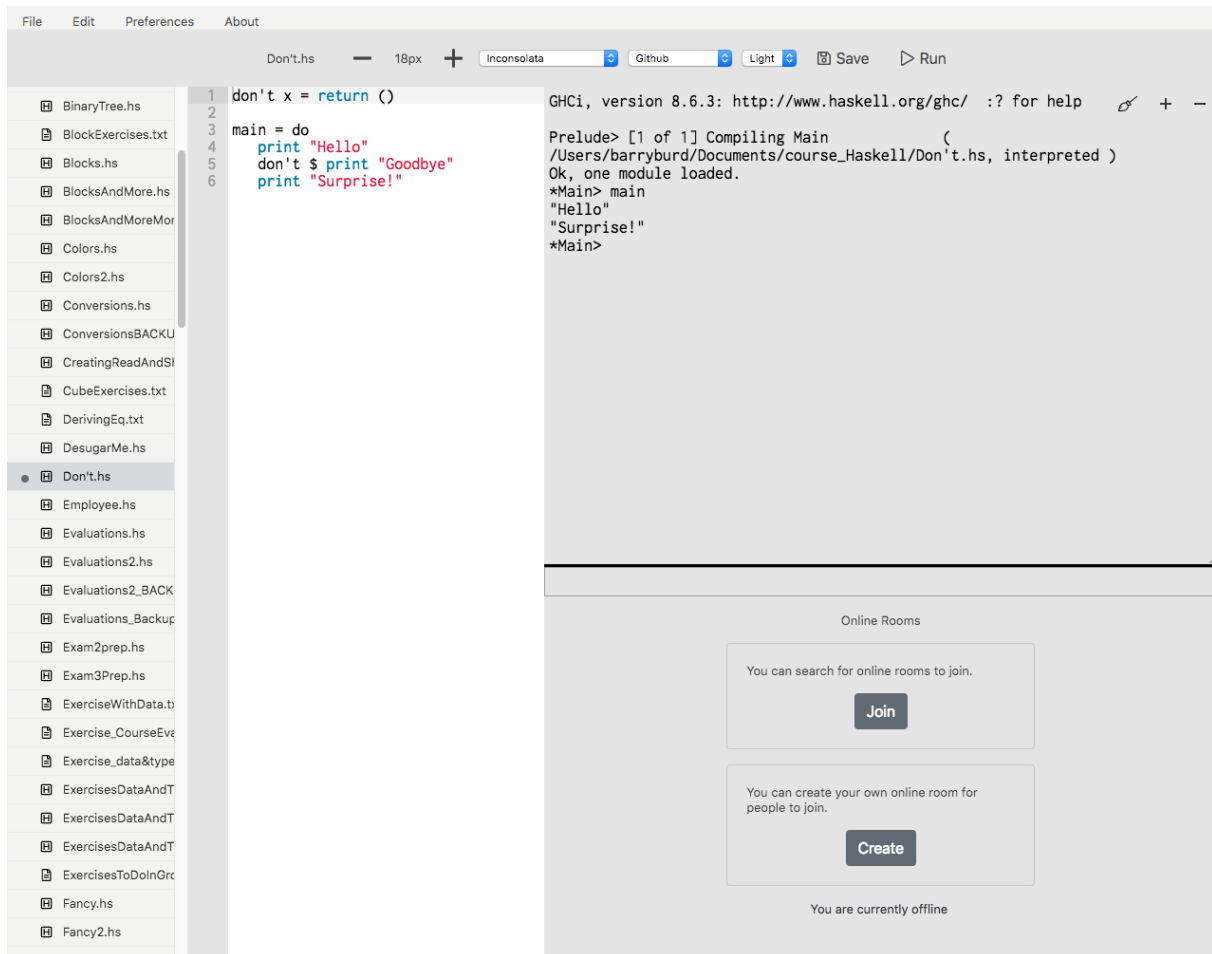


Figure 1: The Haskell for Everyone desktop application.

## Create Room ×

Your Name

Jane J. Student

Room Name

My Room

Room Description

Optional description

Initial File

Select

No file selected (optional)

Editing

Anyone can edit

Access

Public

Create

This will create an online room with you as the owner. Room name is required to be unique. Online connection is required.

Figure 2: Creating a room for real-time, online collaboration.