Rhys Keepence

🌐 British, Australian

📞 +44 7515 353 356

✉️ rhys@rhyskeepence.com

🐙 http://github.com/rhyskeepence

Professional Summary

Over 18 years experience in a variety of industries, developing distributed systems and web applications. I am passionate about improving the way that software is developed, with a particular focus on simplicity and quality. I have plenty of experience of delivering high-quality applications which have been developed in a sustainable manner.

I like to work on simple systems, refactoring ruthlessly to evolve the design. I'm a big believer in starting small, demoing & releasing often, and iterating based on user feedback.

In 2008 I began learning Scala, in 2009 I used it on a small production application, and at Sky I introduced it to the team, switching all new development to Scala. In recent years I have enjoyed learning pure functional programming in Haskell. Through this journey I have found that fully utilising the type system results in clearly documented code, that is simple to test and easy to reason about. At Deutsche Bank I championed Haskell and Elm for two production applications, which was well received within the team.

Key Skills

Functional Programming   Continuous Delivery   XP/Agile   Pair Programming

Scala   ScalaJS   React   Haskell   Elm   CSS3   SQL

Open Source

⫸ Elm Websocket—a library for creating a websockets server in Haskell, which automatically generates the Elm client code. This includes JSON codecs and data types, derived from the Haskell types.

☰ Clairvoyance—a specification-by-example testing framework that produces living documentation on the system's behaviour.

Experience

Senior Scala Developer—Contract   03.2018-present
Citi

I am part of a four-person team developing a new structured notes sales platform.
Rather than develop the platform in isolation, we began by finding existing sales and
trading users, tried to understand the challenges they faced, and built systems to
improve their efficiency. To date, we have replaced the existing error prone Excel pricing
spreadsheet with a web UI, and automated the generation of term-sheets. The resulting
increase in sales year-over-year has been attributed to the automation.

We are a small, co-located team, with a large degree of autonomy over technical and
project management decisions. Our technology stack is functional Scala using Typelevel
libraries, with Typescript and ReactJS to build the UI. To minimise risk and improve our
ability to react to changes, we completely automated our continuous delivery pipeline,
enabling production releases twice a week.

Senior Scala Developer—Contract   09.2013 - 03.2018
Deutsche Bank

Developed a front-office application for the lifecycle management of trades within
Structured Finance. The project consisted of a small team of developers, responsible for
talking to senior members of the business, building the system to capture the
transactions, and supporting it in production. The project has been a large success, and
we had a good relationship with front and middle office.

On this application I championed a number of technologies within the team. The front-
end was a single page app written in ScalaJS (on top of React). This allowed us to
shared data types, safely refactor across the entire stack, and write some pretty
amazing abstractions to scrap boilerplate code. The server was written in functional
scala using Typelevel libraries.

Senior Software Developer—Contract   05.2010 - 09.2013
BSKYB SNS

Part of a small team to rewrite an application to communicate with telephone switches. I
was the main driver behind the team adopting Scala. I wrote a port our chosen
acceptance testing framework to run with Specs2 tests, which I maintain on Github and
has a number of users outside of Sky.

I also lead the development of rewriting an address matching application, using Lucene,
Scala, Scalaz, Dispatch and Spray, which was delivered ahead of all estimates, despite
most of the team having to come up to speed on Scala. The application is written in a
pure functional style and has no exceptions, no 'if' statements and had only one small
bug once going live. It has a full set of automated tests and can be deployed to
production in minutes.

Java Software Developer—Contract Wiley Online Library	**07.2009 - 05.2010**
Java Software Developer—Contract BSKYB	**09.2007 - 12.2008**
Agile Software Developer—Contract—Brisbane, Australia Suncorp (retail banking and insurance)	**03.2006 - 06.2007**
Software Developer—Contract—Melbourne, Australia GE Energy	**01.2006 - 03.2006**
Software Developer—Contract—Brisbane, Australia Queensland Government (Department of Industrial Relations)	**02.2005 - 02.2006**
Software Developer—Contract—Singapore, Singapore	**11.2004 - 02.2005**
Team Lead—Brisbane, Australia Digicon	**12.2001 - 10.2004**
Software Developer—Sydney, Australia Secure Interactive	**06.2000 – 10.2001**

Education

University of Queensland, Master of Information Technology Studies, 2004-2005

ACU, Bachelor of Information Technology, 1997-1999

Sun Certified Java Developer, 2000