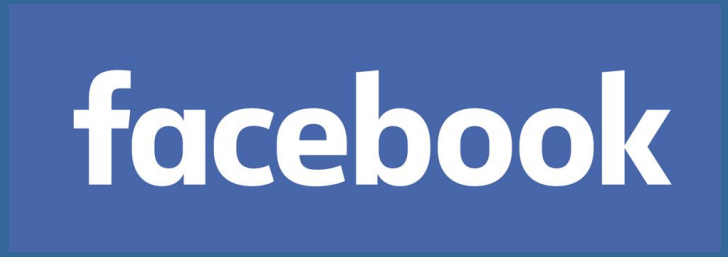
HASKELL
SIN MITOS

Haskell es académico

CODE WRITTEN IN HASKELL IS GUARANTEED TO HAVE NO SIDE EFFECTS.
...BECAUSE NO ONE WILL EVER RUN IT?

Fuente: xkcd

Haskell es utilizado en la industria



stackbuilders

at&t

¿Por qué Haskell?

Razonar sobre el código

Tipificado estáticamente

```
userExist :: Int -> Int -> Bool
userExist userId groupId = …
```

```
newtype UserId = UserId  Int

newtype GroupId = GroupId  Int


userExist :: UserId -> GroupId -> Bool

userExist userId groupId = …
```

Pureza

Evitar errores

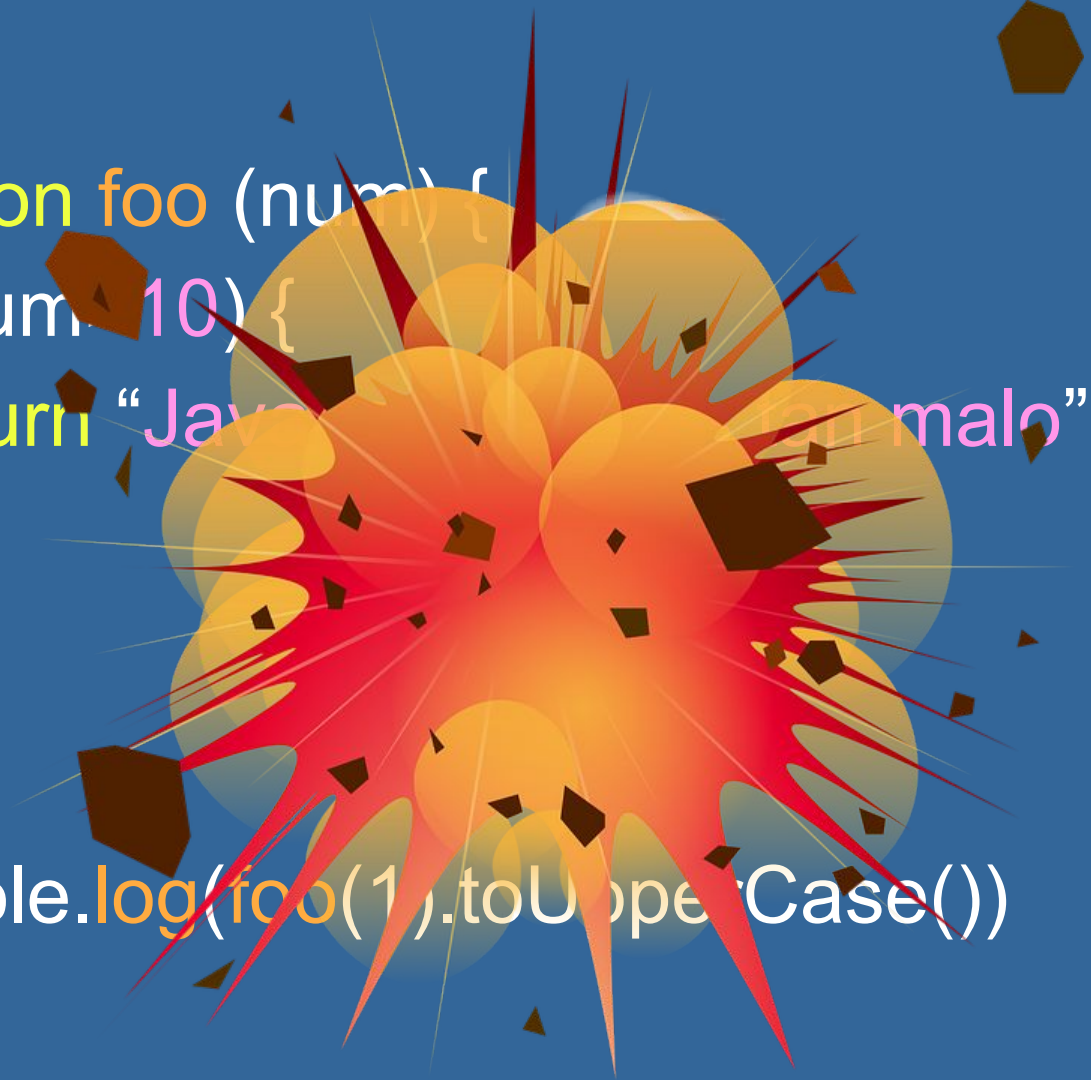
```
function foo (num) {
  if (num>10) {
    return “JavaScript no es tan malo”;
  }
}
```

```
function foo (num) {
  if (num>10) {
    return “JavaScript no es tan malo”;
  }
}

console.log(foo(1).toUpperCase())
```

```
function foo (num) {
  if (num < 10) {
    return “Java [illegible] malo”
  }
}

console.log(foo(1).toUpperCase())
```



Seguridad

```
data Maybe a = Just a | Nothing
```

```
foo num =
  if num > 10
    then Just “Haskell es seguro”
    else Nothing
```

```haskell
foo :: Integer -> Maybe String
foo num =
  if num > 10
    then Just “Haskell es seguro”
    else Nothing
```

```
bar num =
  case foo num of
    Just texto ->
      String.toUpper texto
    Nothing -> “Ouch”
```

```
bar num =
  case foo num of
    Just texto ->
      String.toUpper texto
```

```
bar num =
  case foo nu
    Just texto
      String.to
```

Pattern match(es) are non-exhaustive

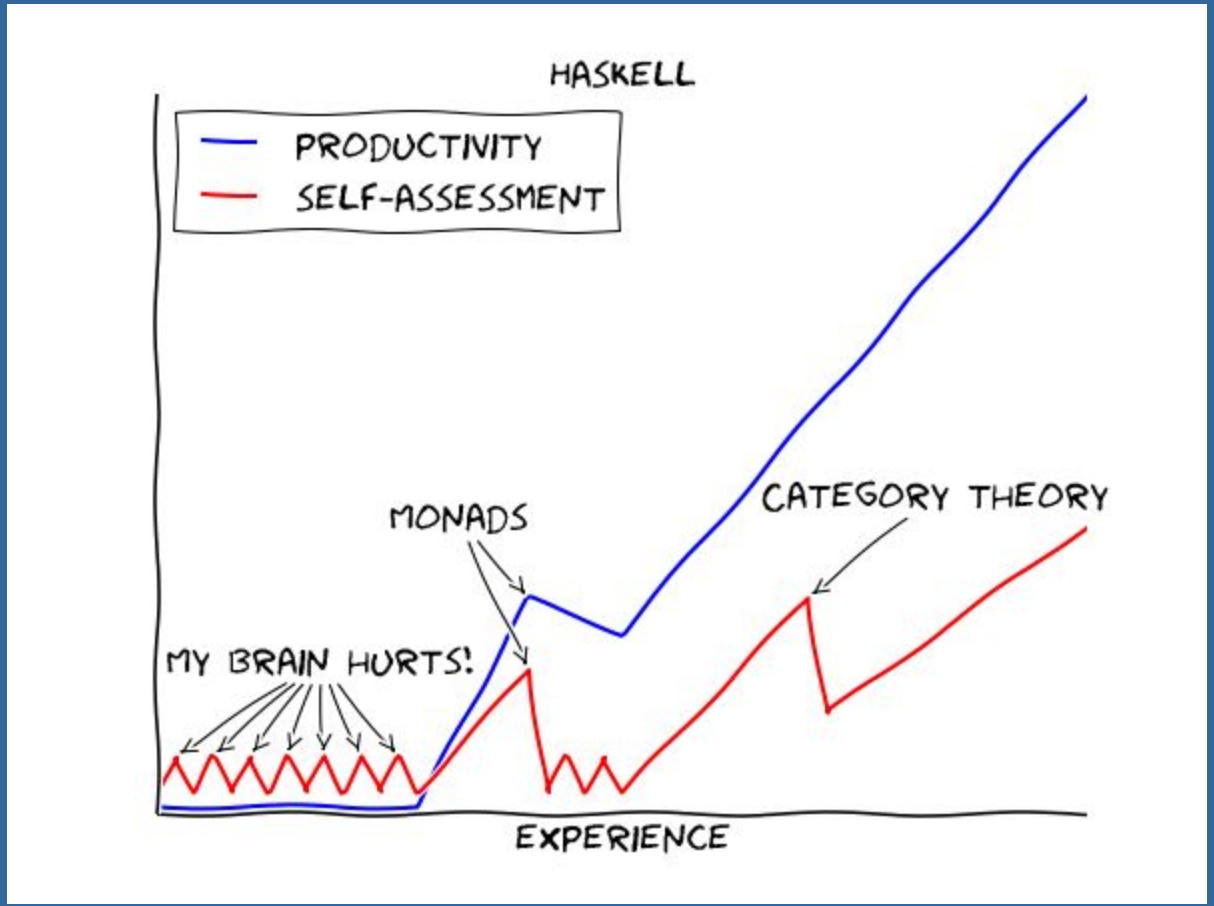
In a case alternative:

Patterns not matched: Nothing

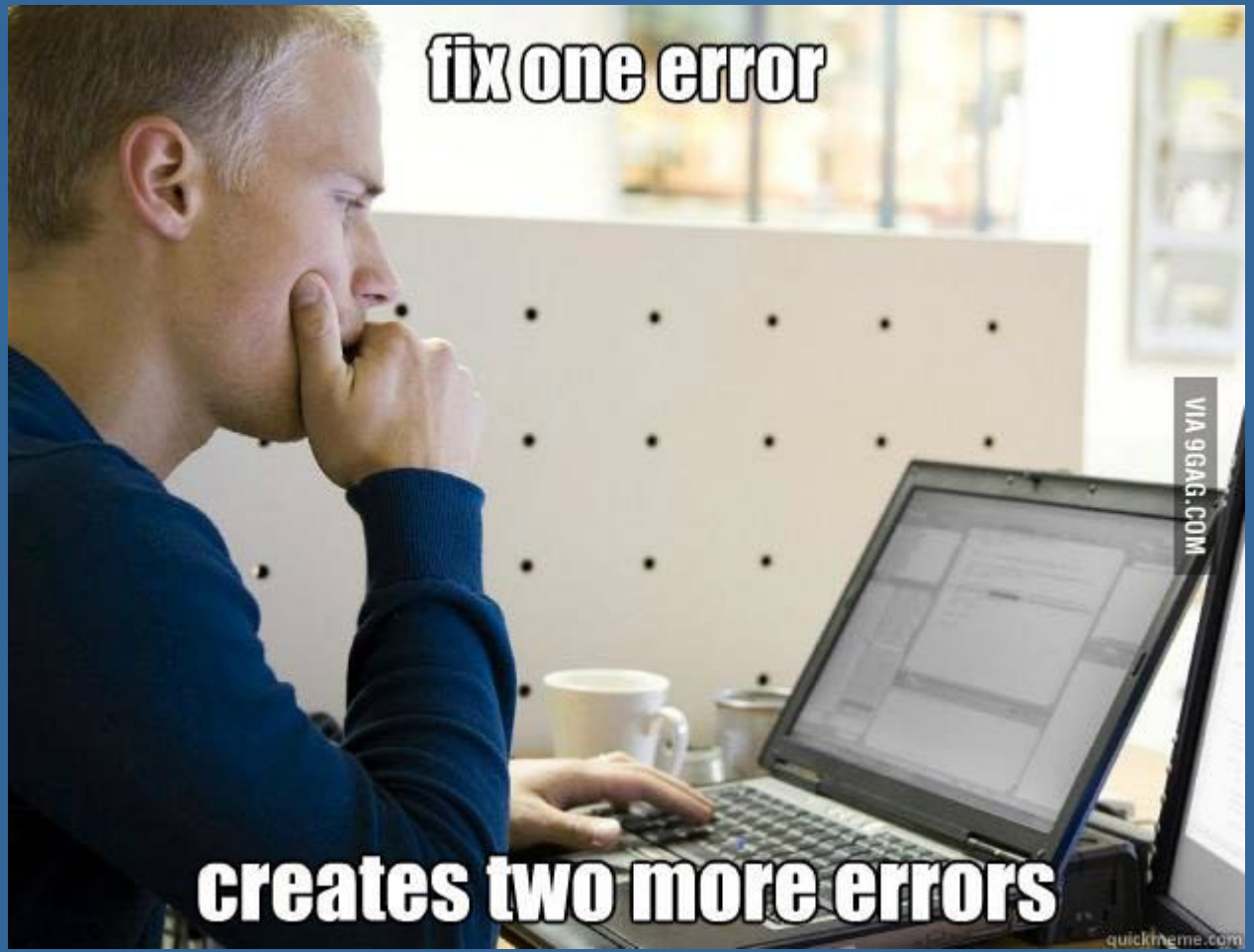
Haskell es difícil

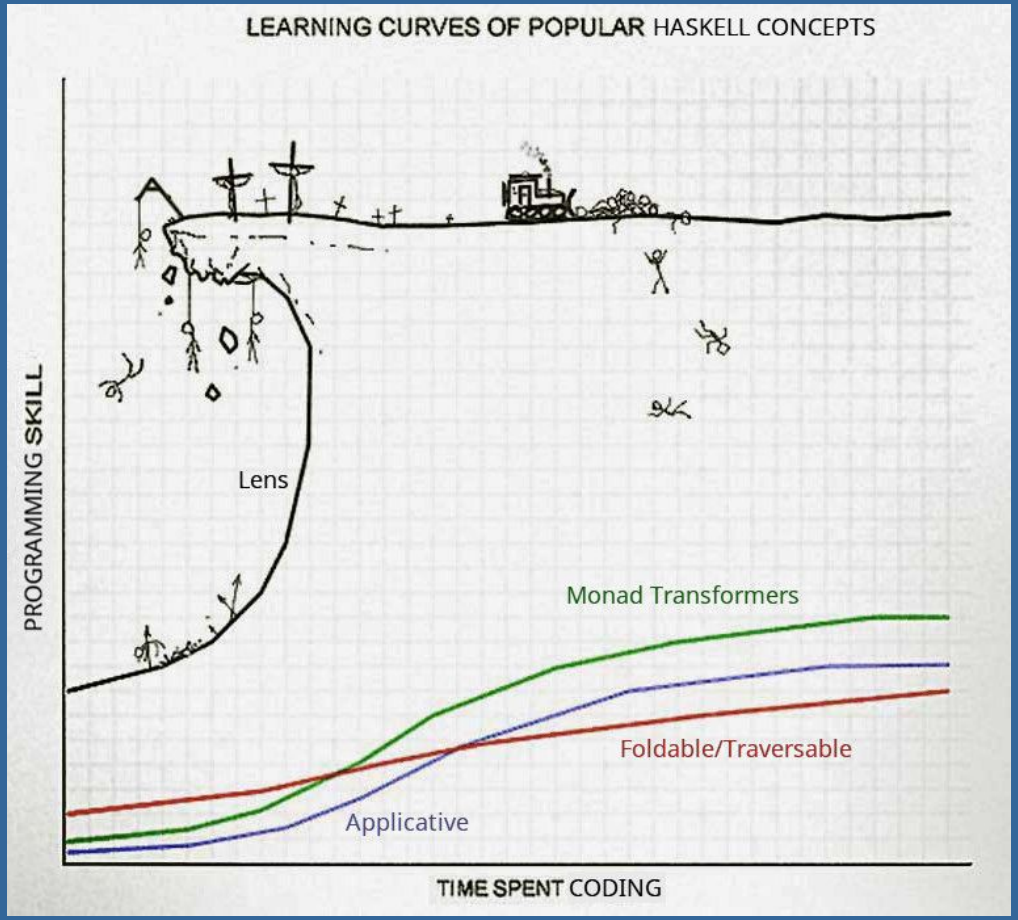
Haskell es difícil de aprender

Expectativa


HASKELL
PRODUCTIVITY
SELF-ASSESSMENT
MONADS
CATEGORY THEORY
MY BRAIN HURTS!
EXPERIENCE

Haskell es diferente


fix one error
VIA 9GAG.COM
creates two more errors
quickmeme.com


LEARNING CURVES OF POPULAR HASKELL CONCEPTS
PROGRAMMING SKILL
Lens
Monad Transformers
Foldable/Traversable
Applicative
TIME SPENT CODING

Abstracciones

"Functional programming is simple. The core ideas are simple. Abstraction can be hard. It doesn't mean it's not worth learning."

John Hughes

Es difícil pero vale la pena

Vim

Haskell es difícil de contratar

stack**builders**

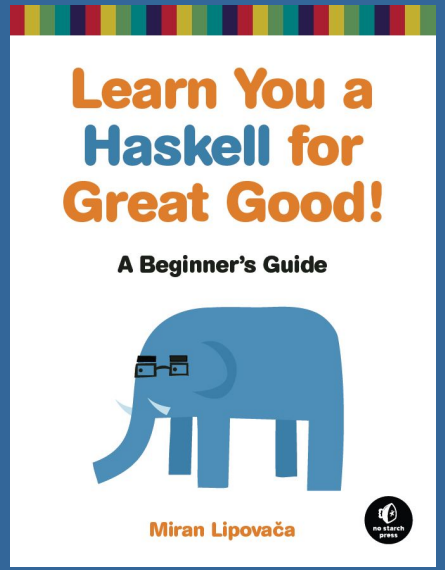
CIS 194: Introduction to Haskell (Spring 2013)

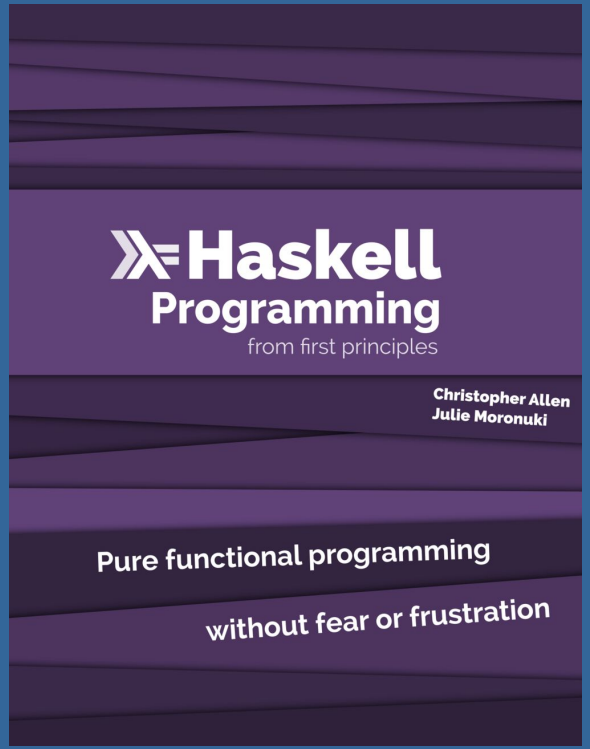
Mondays 1:30-3
Towne 309

Class Piazza site

Instructor: **Brent Yorgey**

- Email: byorgey at cis
- Office: Levine 513
- Office hours: Friday 2-4pm



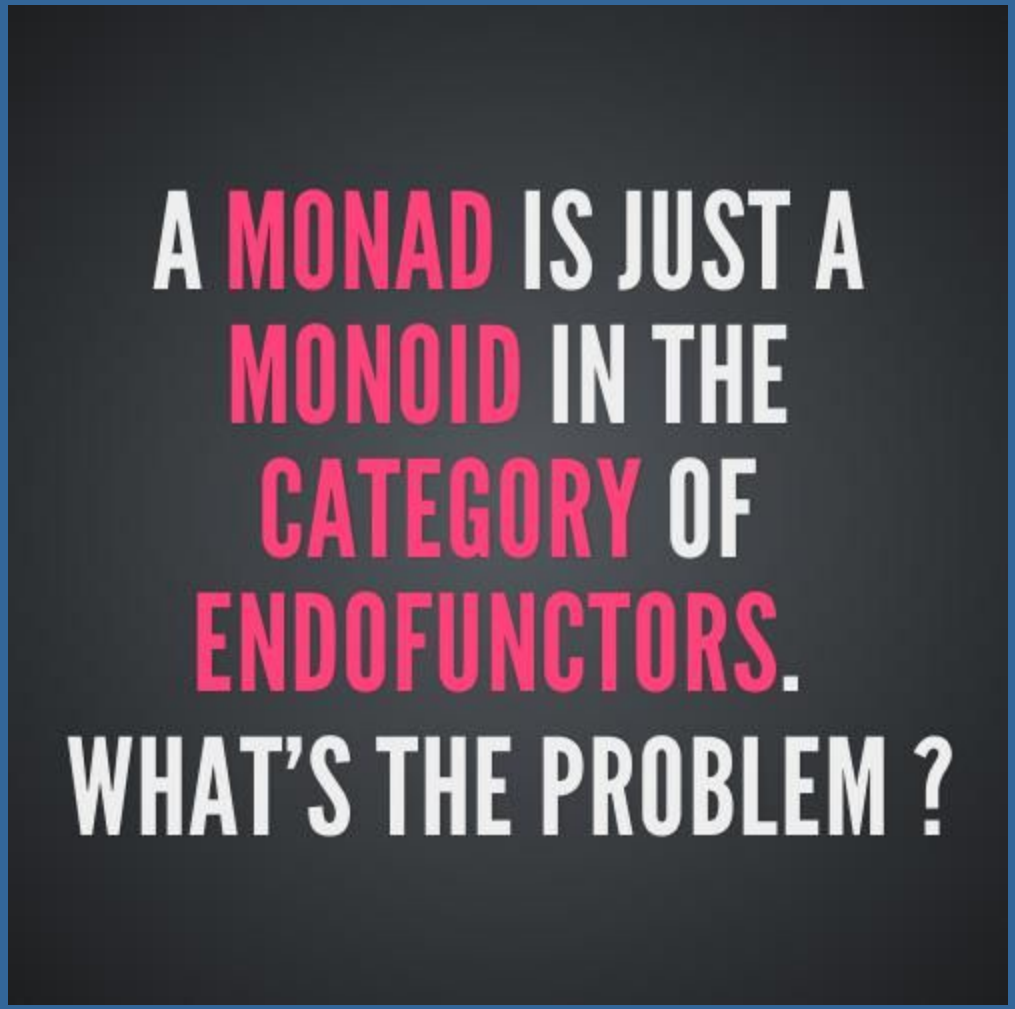


FP101x - Functional Programming
Programming in Haskell – Recursive Functions, part 1
Erik Meijer
TUDelft
Challenge the future

FREE ONLINE COURSE
Functional Programming in Haskell: Supercharge Your Coding
Get an introduction to Haskell, the increasingly popular functional programming language, with this University of Glasgow course.

Haskellers son difíciles

Elitismo


A MONAD IS JUST A
MONOID IN THE
CATEGORY OF
ENDOFUNCTORS.
WHAT'S THE PROBLEM ?

Documentación

Haskell no está listo para producción

Menos esfuerzo arreglando problemas

Es más sencillo reescribir

Cabal hell

No hay suficientes librerías

La documentación es pobre

Librerías de código abierto con poca documentación

Tiempo extra para probar

stackbuilders.com/tutorials

Distracción

El objetivo no es avanzar el estado del arte

Pruebas y deploy

circleci

Haskell es el remedio a todos los males

Código...

```
blockAustralians :: Haxl SyncResponses
blockAustralians = do
  textMap <- textArr
  let text = HashMap.lookupDefault "" "main_text" textMap
      numBadWords   = length $ filter (`Text.isInfixOf` text) aussieTerms
      numBadPhrases = length $ filter (`Text.isInfixOf` text) aussieSayings
  if numBadWords < 2 && numBadPhrases <= 0
  then return noResponses
  else
      if numBadWords < 4 && numBadPhrases < 2
      then return requireCaptcha
      else
          if numBadWords < 5 && numBadPhrases < 3
          then return $ responses [warnUser, requireCaptcha]
          else
              if numBadWords < 7 && numBadPhrases < 4
              then return warnUser
              else
                  if numBadWords < 8 && numBadPhrases < 5
                  then return $ responses [warnUser, blockAccess]
                  else
                      if numBadWords < 10 && numBadPhrases < 6
                      then return blockAccess
                      else
                          if numBadWords < 13 && numBadPhrases < 7
                          then return $ responses
                                  [ blockAccess
                                  , enrollInFakeAccountCheckpoint
                                  ]
                          else return $ responses
                                  [ blockAccess
                                  , enrollInFakeAccountCheckpoint
                                  , requireCaptcha
                                  ]

    where
      aussieTerms =
          [ "Acca Dacca"
          , "ambo"
          , "arvo"
          , "Aussie"
          , "bangaroo"
```

```
blockAustralians :: Haxl SyncResponses
blockAustralians = do
  textMap <- textArr
  let text = HashMap.lookupDefault "" "main_text" textMap
      numBadWords   = length $ filter (`Text.isInfixOf` text) aussieTerms
      numBadPhrases = length $ filter (`Text.isInfixOf` text) aussieSayings
  if numBadWords < 2 && numBadPhrases <= 0
  then return noResponses
  else
      if numBadWords < 4 && numBadPhrases < 2
      then return requireCaptcha
      else
          if numBadWords < 5 && numBadPhrases < 3
          then return $ responses [warnUser, requireCaptcha]
          else
              if numBadWords < 7 && numBadPhrases < 4
              then return warnUser
              else
                  if numBadWords < 8 && numBadPhrases < 5
                  then return $ responses [warnUser, blockAccess]
                  else
                      if numBadWords < 10 && numBadPhrases < 6
                      then return blockAccess
                      else
                          if numBadWords < 13 && numBadPhrases < 7
                          then return $ responses
                                  [ blockAccess
                                  , enrollInFakeAccountCheckpoint
                                  ]
                          else return $ responses
                                  [ blockAccess
                                  , enrollInFakeAccountCheckpoint
                                  , requireCaptcha
                                  ]

    where
      aussieTerms =
          [ "Acca Dacca"
          , "ambo"
          , "arvo"
          , "Aussie"
          , "bangaroo"
```

El legado

Java™