

# Elm как язык разработки веб-интерфейсов

Симоненко Е.А.  
easimonenko@mail.ru

2018

# Содержание

- 1 Мотивация и история появления Elm
- 2 Пример приложения
- 3 Основные инструменты
- 4 Управление кодом на Elm
- 5 Дополнительные инструменты
- 6 Среды разработки
- 7 Интероперабельность
- 8 За и против, и что в будущем
- 9 Что почитать
- 10 Ссылки

# Мотивация

Требования:

- функциональный
- статически типизированный

Варианты:

- *GHCJS* <https://github.com/ghcjs/ghcjs>
- *PureScript* <http://www.purescript.org/>
- *Haste* <https://haste-lang.org/>
- *Fay* <https://github.com/faylang/fay/>
- *Idris* <https://www.idris-lang.org/>
- *Elm* <http://elm-lang.org/>

# История появления Elm

- 2012
- дипломная работа студента
- автор: Эван Чаплицкий (Evan Czaplicki)
- функционально-реактивный язык

# Как Haskell, только проще, и чтобы в браузере

- синтаксис как в Haskell
- отсутствуют классы типов, а заодно и монады
- компилируется в JavaScript для браузера

# Особенности

- чистый функциональный язык
- собственная система модулей
- сам по себе
- специфические способы интероперабельности
- сборка в один файл

## Пример приложения



## Импорт модулей

```
module Main exposing (main)

import Html exposing (Html, text, div, button)
import Html.Attributes exposing (class)
import Html.Events exposing (onClick)
```



# Главная функция

```
main : Program Never Model Msg
main =
    Html.beginnerProgram
        { model = initialModel
        , update = update
        , view = view
        }
```

# Модель

```
type alias Model =  
    { value : Int  
    }
```

```
initalModel : Model  
initalModel =  
    { value = 0  
    }
```

## Обновление

```
type Msg
  = Increment
  | Decrement
```

```
update : Msg -> Model -> Model
```

```
update msg model =
```

```
  case msg of
```

```
    Increment ->
```

```
      { model | value = model.value + 1 }
```

```
    Decrement ->
```

```
      { model | value = model.value - 1 }
```

# Представление

```
view : Model -> Html Msg
view model =
  div []
    [ div [ class "counter" ]
      [ text (toString model.value) ]
    , div [ class "controls" ]
      [ button [ onClick Increment ] [ text "+1" ]
      , button [ onClick Decrement ] [ text "-1" ]
      ]
    ]
```

## На стороне HTML

```
<script src="/app.js"></script>  
<script>  
  var app = Elm.Main.fullscreen();  
</script>
```

# Установка Elm

- `https://guide.elm-lang.org/install.html`
- `npm install -g elm`
- `elm -version`
- 0.18.0

# Базовые инструменты Elm

- `elm-make`
- `elm-package`
- `elm-reactor`
- `elm-repl`

# Управление пакетами Elm

- `elm package install elm-community/list-extra`
- <http://package.elm-lang.org/>



# Установка и обновление

```
elm package install
```

```
Some new packages are needed. Here is the upgrade plan.
```

```
Install:
```

```
elm-community/list-extra 7.1.0  
elm-community/string-extra 1.4.0  
elm-lang/core 5.1.1  
elm-lang/dom 1.1.1  
elm-lang/html 2.0.0  
elm-lang/http 1.0.0  
elm-lang/virtual-dom 2.0.4
```

```
Do you approve of this plan? [Y/n]
```

```
Starting downloads...
```

```
● elm-community/string-extra 1.4.0  
● elm-community/list-extra 7.1.0  
● elm-lang/dom 1.1.1  
● elm-lang/core 5.1.1  
● elm-lang/http 1.0.0  
● elm-lang/html 2.0.0  
● elm-lang/virtual-dom 2.0.4
```

```
Packages configured successfully!
```

## Файл пакета Elm

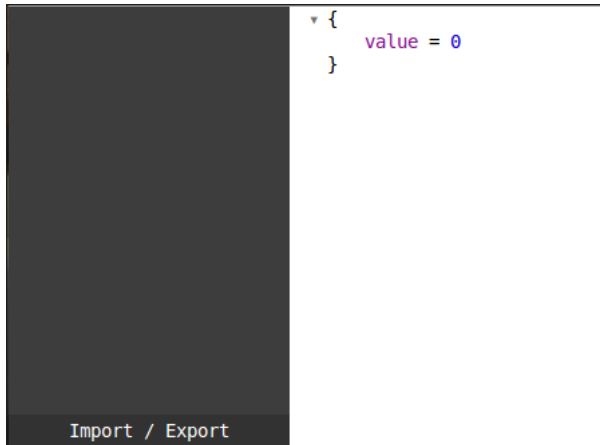
elm-package.json:

```
{
  "version": "1.0.0",
  "summary": "helpful summary of your project, less than 80 characters",
  "repository": "https://github.com/user/project.git",
  "license": "BSD3",
  "source-directories": ["app/elm"],
  "exposed-modules": [],
  "dependencies": {
    "elm-lang/core": "5.0.0 <= v < 6.0.0",
    "elm-lang/dom": "1.1.1 <= v < 2.0.0",
    "elm-lang/html": "2.0.0 <= v < 3.0.0"
  },
  "elm-version": "0.18.0 <= v < 0.19.0"
}
```

# Brunch

- `npm install -g brunch`
- `brunch -version`
- `2.10.12`
- `brunch new -skeleton MattCheely/elm-brunch-skeleton demo-application`
- `cd demo-application`
- `npm build`
- `npm start`

# Отладка



# Отладка



# Отладка

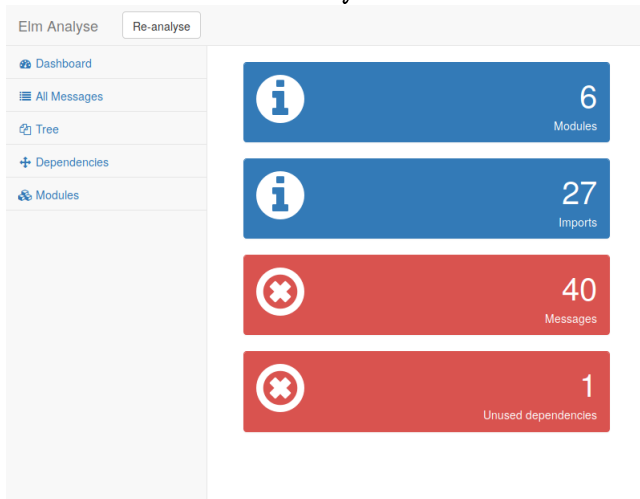
Increment	0	▼ {
Increment	1	value = 2
Decrement	2	}
Resume		
Import / Export		

## Дополнительные инструменты

- *elm-format* <https://github.com/avh4/elm-format>
- *elm-format-short* <https://github.com/nukisman/elm-format-short>
- *elm-analyse* <https://www.npmjs.com/package/elm-analyse>

# elm-analyse: главное окно

elm-analyse -s





# elm-analyse: список проблем

elm-analyse -s

The screenshot shows the Elm Analyse web interface. On the left is a sidebar with navigation links: Dashboard, All Messages, Tree, Dependencies, and Modules. The main area is titled 'Messages' and shows a list of 5 messages for the file 'app/elm/API.elm'. The messages are numbered #1 to #5. The first message is 'Expose All' with the description 'Exposing all at ((0,21),(0,23))'. The second is 'Import All' with 'Importing all from module `String.Extra` at ((2,30),(2,32))'. The third is 'Single Field Record' with 'Record has only one field. Use the field's type or introduce a Type. At ((19,8),(20,9))'. The fourth and fifth are both 'Unused Pattern Variable' with 'Unused variable `params` inside pattern at ((123,34),(123,40))' and 'Unused variable `params` inside pattern at ((126,32),(126,38))' respectively. The interface also includes a 'Re-analyse' button and a 'Group by' dropdown menu.

Elm Analyse Re-analyse

Dashboard All Messages Tree Dependencies Modules

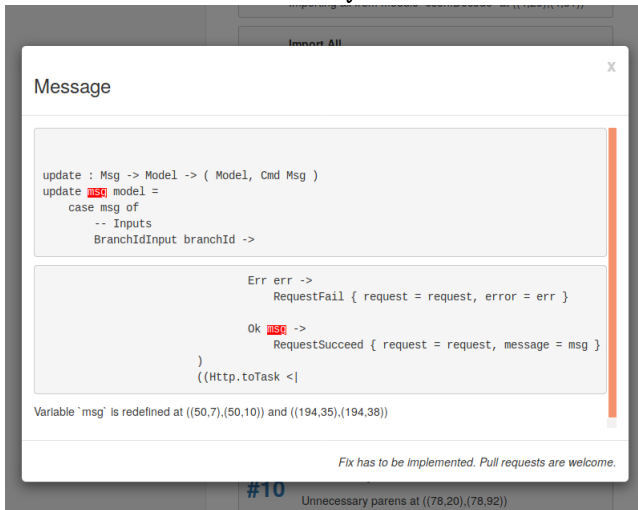
Messages Group by file Group by type

app/elm/API.elm

- #1 **Expose All**  
Exposing all at ((0,21),(0,23))
- #2 **Import All**  
Importing all from module `String.Extra` at ((2,30),(2,32))
- #3 **Single Field Record**  
Record has only one field. Use the field's type or introduce a Type. At ((19,8),(20,9))
- #4 **Unused Pattern Variable**  
Unused variable `params` inside pattern at ((123,34),(123,40))
- #5 **Unused Pattern Variable**  
Unused variable `params` inside pattern at ((126,32),(126,38))

# elm-analyse: пример с фрагментом кода

elm-analyse -s



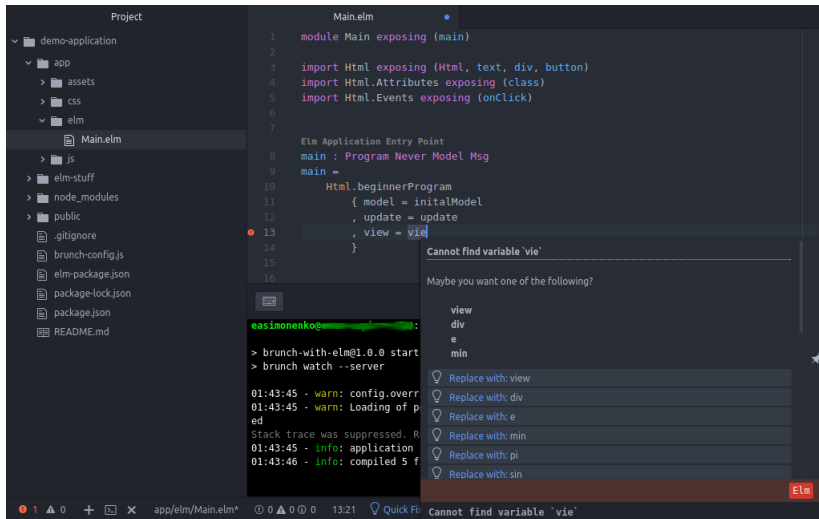
# Среды разработки

- Atom <https://habr.com/post/347730/>
- Light Table <https://habr.com/post/302154/>
- Visual Studio Code
- GNU Emacs
- другие

# Настройка Atom для работы с Elm

- `language-elm`
- `atom-ide-ui`
- `autocomplete-plus`
- `elmjutsu`
- `elm-format`
- `linter`
- `linter-elm-make`
- `elm-lens`
- `elm-instant`
- `platformio-ide-terminal`

# Atom в процессе работы с Elm



# Способы интероперабельности

- port
- native module

## Пример с port (код на Elm)

```
port title : String -> Cmd msg
...
title <| "Site_Title_::_" ++ pageTitle
```

## Пример с port (код на JavaScript)

```
app.ports.title.subscribe(function (title) {  
    document.title = title;  
});
```



# Результат опроса

<https://habr.com/post/347730/>

- не нужен в принципе: 26.9%
- годный, но на нём не пишу: 25%
- в первый раз вижу, но выглядит годным: 19.2%
- сыроват для серьёзной разработки: 15.3%
- годный, я на нём уже программирую: 13.4%

# За Elm

## Технические “за”:

- если скомпилировалось, то и работать будет
- отсутствие исключений и неопределённостей
- более понятная последовательная логика
- более простой и ясный синтаксис

## Социальные “за”:

- ???

# Против Elm

## Технические “против”:

- сложный механизм интероперабельности
- недостаточный спектр прикладных пакетов
- большой объём скомпилированного кода
- отсутствие модульности во время исполнения
- несовместимость кода для разных версий компилятора

## Социальные “против”:

- необходимость переключиться в иную парадигму
- практически полное отсутствие кадров
- мало историй успеха
- отсутствие поддержки со стороны компаний
- неясность перспективы

## Что в будущем

- совершенствование инструментов разработчика
- улучшение поддержки разработки SPA
- не планируется поддержка других платформ

Подробности: <https://github.com/elm-lang/projects/blob/master/roadmap.md>

## Что почитать

- <https://habr.com/post/347730/>
- <https://habr.com/post/347730/>
- <http://ts-soft.ru/blog/elm-part-1>
- <http://ts-soft.ru/blog/elm-part-2>
- <https://habr.com/hub/elm/>
- <http://elm-lang.org/docs>

- <http://elm-lang.org/>
- <http://brunch.io/>
- <https://github.com/halohalospecial/atom-elmjutsu>
- <https://github.com/avh4/elm-format>
- [https://vk.com/elm\\_lang\\_ru](https://vk.com/elm_lang_ru)

*Благодарю за внимание!*  
***Elm как язык разработки веб-интерфейсов***  
Симоненко Е.А.  
easimonenko@mail.ru