

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
INE5426 - CONSTRUÇÃO DE COMPILADORES

PROJETO 2

NativeIDE

Alunos:

Diego da Silva Marques
Matheus Faustino Demetrio

Florianópolis, 19 de Outubro de 2016

1- Introdução

NativeIDE tem como proposta prover um editor para aplicativos que utilizam React Native, gerando um “live preview” do componente sendo editado no momento convertendo o componente para HTML, CSS e JavaScript e mostrando a partir de uma WebView ou utilizando um emulador do sistema operacional desejado.

2- Motivação

O desenvolvimento utilizando React Native tem crescido muito nos últimos tempos, e a única ferramenta disponível que fornece a funcionalidade de live preview (utilizando apenas o emulador) é a Deco IDE, porém a mesma só funciona em sistemas operacionais macOS.

3- Projeto

O projeto consiste em prover uma interface ao desenvolvedor, se assemelhando a um editor de texto, onde o desenvolvedor possa escrever os seus aplicativos e visualizá-los em tempo real.

Para tal, é necessário realizar uma análise léxica, sintática e semântica do código que é inserido na área de inserção de texto no programa, e interpretá-lo de uma forma onde cada componente nativo da biblioteca React Native seja mapeado a uma tag correspondente em HTML, ainda levando em conta o mapeamento de funções onde o estado do componente seja alterado para JavaScript e aplicando as regras de estilo que existem no componente através do CSS em sua contraparte em HTML.

Neste projeto, serão utilizadas as seguintes ferramentas:

- **Electron:** Ferramenta que permite a criação de aplicativos para desktop utilizando tecnologias baseadas em JavaScript
- **React:** Biblioteca para criação de interfaces de usuário baseada em JavaScript, e que serve como base para a biblioteca no qual este projeto irá lidar (React Native).
- **JSX:** Linguagem utilizada pela biblioteca React para representar seus componentes, semelhante ao HTML.

- **Esprima:** Biblioteca que contém um parser da linguagem JavaScript, gerando um objeto JSON que representa a árvore sintática do código, além de seus tokens (análise léxica e sintática).

Tais ferramentas foram escolhidas pois como a biblioteca destino é baseada no próprio React, e a visualização dos componentes será feita em HTML, essas ferramentas se adequam e auxiliam a diminuir a complexidade do desenvolvimento da interface com o usuário, pois a renderização do HTML é nativa.

4- Divisão do Projeto

Parte 1 - Implementação da GUI do editor

- Nessa parte será implementada a interface mínima do editor (campo para entrar com código) e um espaço para conter o componente capaz de renderizar o HTML do componente

Parte 2 - Análise semântica

- Implementar as regras de análise semânticas, baseadas na biblioteca alvo, na linguagem JavaScript e na proposta do projeto.

Parte 3 - Feedback das análises

- Adicionar a GUI as capacidades de feedback sobre as análises léxica, sintática e semântica no editor (coloração de palavras baseadas na tokenização, marcação de erros sintáticos).

Parte 4 - Mapeamento dos componentes nativos

- Definir o mapeamento para todos os componentes nativos visuais do React Native para componentes correspondentes em HTML, CSS e JavaScript.

Parte 5 - Tradução

- Implementar o tradutor capaz de traduzir o resultado da análise feita anteriormente para HTML, CSS e JavaScript

Parte 6 - Renderização

- Aprimorar a GUI para renderizar o resultado da tradução.

Parte 7 - Gerenciamento de Arquivo

- Fornecer um pequeno gerenciador de arquivos ao usuário através da GUI, e funções de IO para os arquivos, além da capacidade de executar direto no emulador/dispositivo, ao invés do live preview.

5 - Papéis

- Gerente: Diego da Silva Marques e Matheus Faustino Demetrio
- Arquiteto de Sistema: Diego da Silva Marques
- Testador: Matheus Faustino Demetrio

Referências

<http://esprima.org/>

<http://electron.atom.io/>

<https://facebook.github.io/react/>

<https://facebook.github.io/react-native/>

<https://www.decosoftware.com/>