



Universidade Federal de Pernambuco  
Centro de Informática

Graduação em Ciência da Computação

# **Extensão e avaliação funcional da biblioteca CEPlin**

Proposta de Trabalho de Graduação

**Aluno:** Jonathan Gomes dos Santos (jgs@cin.ufpe.br)

**Orientador:** Kiev Santos da Gama (kiev@cin.ufpe.br)

Recife, 31 de agosto de 2018

## **Sumário**

<b>1. Contextualização</b>	<b>3</b>
<b>2. Objetivos</b>	<b>4</b>
<b>3. Cronograma</b>	<b>5</b>
<b>4. Possíveis Avaliadores</b>	<b>6</b>
<b>5. Referências</b>	<b>7</b>

# 1. Contextualização

Algo que tem se tornado cada vez mais comum é o processamento de fluxos de informação em aplicações. Tal necessidade acaba por dar vida a várias ferramentas que, embora diferentes, seguem um modelo de processamento bastante similar [3].

Surgida dessa demanda, vem a área de Processamento de eventos complexos (*Complex Event Processing*, CEP). Sistemas desse tipo funcionam filtrando e combinando eventos mais simples, com o intuito de detectar eventos de mais alto nível, e propagando tais ocorrências para as partes interessadas. Esses sistemas comumente utilizam linguagens de consulta, que normalmente são pouco intuitivas e conduzem a alguns erros [3].

Outra forma de lidar com essa demanda são as Linguagens Reativas (RLs) que, por sua vez, têm como principal funcionalidade o tratamento de valores que mudam ao longo do tempo [5], de forma a estender o padrão Observer. Porém, ao contrario deste, tais linguagens propagam esses dados às partes interessadas de forma implícita. Linguagens reativas também integram-se ao ambiente utilizado, tornando o processo de desenvolvimento mais simples e oferecendo recursos de análise estática [4].

Ambas as abordagens tem a capacidade de receber mudanças advindas de múltiplas fontes, aplicar combinações sobre as mesmas e propagar a informação, o que caracteriza um fluxo muito similar nas duas.

Dados esses fatos, foi desenvolvida a biblioteca CEPlin, que traz a abordagem reativa proveniente das RLs para o processamento de eventos complexos na linguagem de programação Kotlin [8].

## 2. Objetivos

Devido ao tempo reduzido de desenvolvimento da biblioteca CEPLin, seu escopo foi dimensionado para o tempo existente, o que deixou margem para a sua continuação em novos trabalhos.

Este trabalho dará continuidade aos trabalhos dessa biblioteca. Inicialmente será realizada a implementação de um subconjunto de operadores CEP, com o intuito de expandir o a gama de funcionalidades providas pela biblioteca. Além disso, será realizada a validação com a realização de testes dos operadores implementados em streams, visto que, dado o caráter temporal do processamento de eventos complexos, testes unitários simples não são suficientes. Com isso teremos os novos operadores implementados e testados de modo que sejam tidos como uma extensão válida da biblioteca CEPLin, incrementando suas possibilidades como desejado.

### 3. Cronograma

Atividades	Mês															
	Agosto			Setembro			Outubro			Novembro			Dezembro			
Definição do escopo																
Formulação da proposta																
Revisão bibliográfica																
Estudo e análise das ferramentas																
Implementação																
Escrita do documento																
Preparação para a defesa																
Defesa																

## **4. Possíveis Avaliadores**

Os seguintes professores são considerados como possíveis avaliadores do trabalho proposto:

- Márcio Lopes Cornélio (mlc2@cin.ufpe.br)
- Ricardo Massa Ferreira de Lima (rmfl@cin.ufpe.br)

## 5. Referências

- [1] PASCHKE, Adrian; KOZLENKOV, Alexander; BOLEY, Harold. A homogeneous reaction rule language for complex event processing. arXiv preprint arXiv:1008.0823, 2010.
- [2] AGUILERA, Marcos K. et al. Matching events in a content-based subscription system. In: Proceedings of the eighteenth annual ACM symposium on Principles of distributed computing. ACM, 1999.
- [3] CUGOLA, Gianpaolo; MARGARA, Alessandro. Processing flows of information: From data stream to complex event processing. ACM Computing Surveys (CSUR), v. 44, n. 3, 2012.
- [4] MARGARA, Alessandro; SALVANESCHI, Guido. Ways to react: Comparing reactive languages and complex event processing. REM, 2013.
- [5] ReactiveX, RxKotlin Project. Disponível em <<https://github.com/ReactiveX/RxKotlin>>. Acessado em 24 de agosto de 2018.
- [6] Kotlinlang.org, Kotlin Reference. Disponível em <<https://kotlinlang.org/docs/reference/>>. Acessado em 24 de agosto de 2018.
- [7] TEYMOURIAN, Kia; PASCHKE, Adrian. Enabling knowledge-based complex event processing. In: Proceedings of the 2010 EDBT/ICDT Workshops. ACM, 2010.
- [8] LINS, Jonas. CEPlin: A Complex Event Processing Framework for Kotlin. TG. 2018

# Assinaturas

---

Jonathan Gomes dos Santos  
(Aluno)

---

Kiev Santos da Gama  
(Orientador)