

# Assessing Developer Contribution with Repository Mining-Based Metrics

**Jalerson Lima**

Federal University of Rio Grande do  
Norte

Federal Institute of Education at Rio  
Grande do Norte  
jalerson.lima@ifrn.edu.br

**Christoph Treude, Fernando Figueira  
Filho and Uirá Kulesza**

Federal University of Rio Grande do Norte  
Norte

{ctreude,fernando,uira}@dimap.ufrn.br

# Agenda

---

1. Introduction and Motivation
2. Repository Mining-based Metrics
3. Research Method
4. Preliminary Results
5. Discussion and Future Work

# Introduction and Motivation

---

- Assessing developer's contribution is a challenging task
  - Many potential sources of contribution have to be considered
- Several metrics have been proposed
  - Few have been evaluated by project leaders (potential users)

Our goal is to design a suite of developer contribution metrics based on empirical evidence obtained from project and team leaders

# Repository Mining-based Metrics

---

Code  
Contribution

Average  
Complexity  
per Method

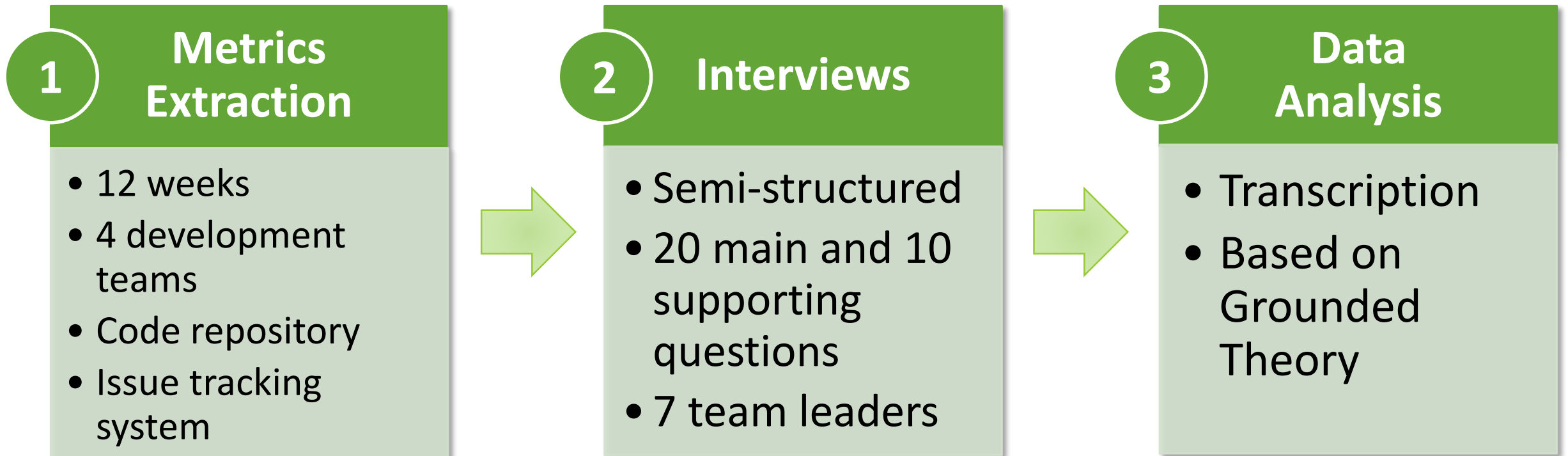
Bug Fixing  
Contribution

Introduced  
Bugs

# Research Method

---

- **Location:** SINFO/Federal University of Rio Grande do Norte, Brazil



# Preliminary Results

## Code Contribution

---



Useful information



*“May be useful with the complexity metric” (PL1)*



May penalize developers for using modern technologies or techniques

# Preliminary Results

## Average Complexity per Method

---

- ✅ *“Allows to perform a technical analysis” (PL3)*
- ✅ *“Helps to identify a developer who needs training” (PL2)*
- ⚠️ Should be followed with task list

# Preliminary Results

## Introduced Bugs

---



Useful information



Can't be used in isolation



May penalize developers who have been on the project for longer



# Preliminary Results

## Bug Fixing Contribution

---

- ✗ The metric only quantifies commits, however, some tasks don't require coding
- ✗ The number of commits is not a reliable attribute to measure effort

# Preliminary Results

## Overall Benefits and Limitations

---

- ✅ Useful to perform a quantitative contribution assessment
- ✅ May reduce the amount of time to evaluate developers
- ✅ Technical and objective criteria to evaluate developers
- ⚠️ Can't replace the subjective evaluation

# Discussion and Future Work

---



- ✓ **Best evaluated metrics**
  - Code contribution
  - Average complexity per method



- ✓ **Worst evaluated metrics**
  - Introduced bugs
  - Bug fixing contribution

# Discussion and Future Work

---



## ✓ Evaluate further metrics

- Communication
- Collaboration
- Task distribution



## ✓ Interview developers about being evaluated by these metrics

- Compare their answers with team leaders' answers

# Discussion and Future Work

---



- ✓ Investigate the impact of measuring contribution on developers' behavior
  - Hawthorne effect



- ✓ Metrics-based reward mechanisms
  - Gamification

# Assessing Developer Contribution with Repository Mining-Based Metrics

**Jalerson Lima**

Federal University of Rio Grande do  
Norte

Federal Institute of Education at Rio  
Grande do Norte  
jalerson.lima@ifrn.edu.br

**Christoph Treude, Fernando Figueira  
Filho and Uirá Kulesza**

Federal University of Rio Grande do Norte  
Norte

{ctreude,fernando,uira}@dimap.ufrn.br