



UNIVERSITY OF TARTU  
Institute of Computer Science



# SEA Group: Software Engineering Analytics

University of Tartu

ICS Day 2024

Dietmar Pfahl, Hina Anwar, Faiz Ali Shah, Kristiina Rahkema, +PhD/MSc students

[dietmar.pfahl@ut.ee](mailto:dietmar.pfahl@ut.ee)

<https://sea.cs.ut.ee>



# Software Engineering Analytics – The Team

Dietmar Pfahl

(Professor)



Hina Anwar

(Assistant Professor)



Faiz Ali Shah

(Assistant Professor)



Kristiina Rahkema

(Assistant Professor)



Aleja Duque-Torres

(PhD Student, Junior Researcher with SCCH)



Fauzia Khan

(Junior Researcher, PhD Student)



Maj-Annika Tammisto

(Junior Researcher, PhD Student)



Ali Ihsan Güllü

(Junior Researcher, PhD Student)



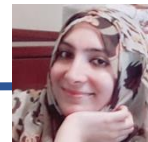
Sander Sats

(Junior Researcher, PhD Student)



Maria Khan

(Junior Lecturer, PhD Student at TLU)



## Past members of SEA:

- Ezequiel Scott (Assistant Professor)
- Mariana Falco (Visiting Researcher)
- Tarcio Bezerra (Visiting Researcher)
- Huishi Yin (former PhD Student)
- Riivo Kikas (former PhD Student)
- Laima Anna Dalbina (former MSc Student)



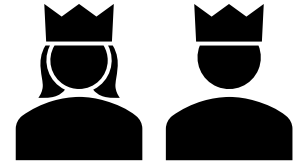
# Software Engineering Analytics – The Mission

***“To collect and analyze software engineering data with the goal to estimate, predict, and improve the quality of software and software-intensive systems”***

## Notes:

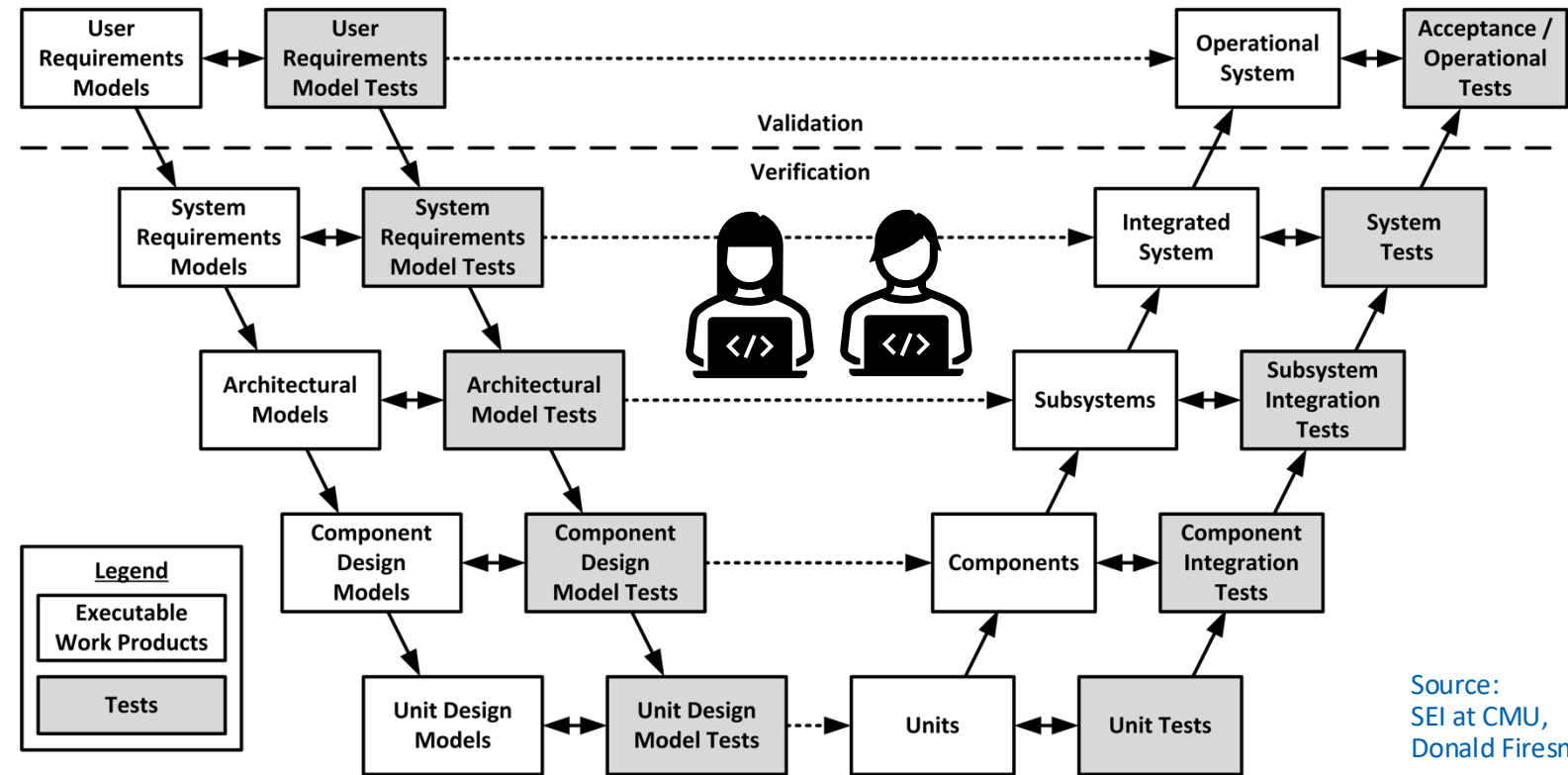
- Software Engineering = A systematic approach to develop software with the expected quality, within a given time and effort budget
- Software quality has many dimensions/facets, e.g., functionality, dependability, maintainability, performance, energy-efficiency, etc.
- The terms “estimate, predict, and improve” imply the development of models and tools (prototypes)

# Software Engineering Analytics – The Scope



## What we do:

- Methods and tools for static document and code analysis
- Methods and tools for dynamic code analysis (incl. testing)
- Methods and tools for quality estimation and prediction
- Methods and tools for selecting, augmenting, and generating test data, test cases, test oracles, and test code



## Note:

- All of this may or may not involve AI



Source:  
SEI at CMU,  
Donald Firesmith

# SEA Project Example 1: Mobile App Feature Analysis

## Research Topic:

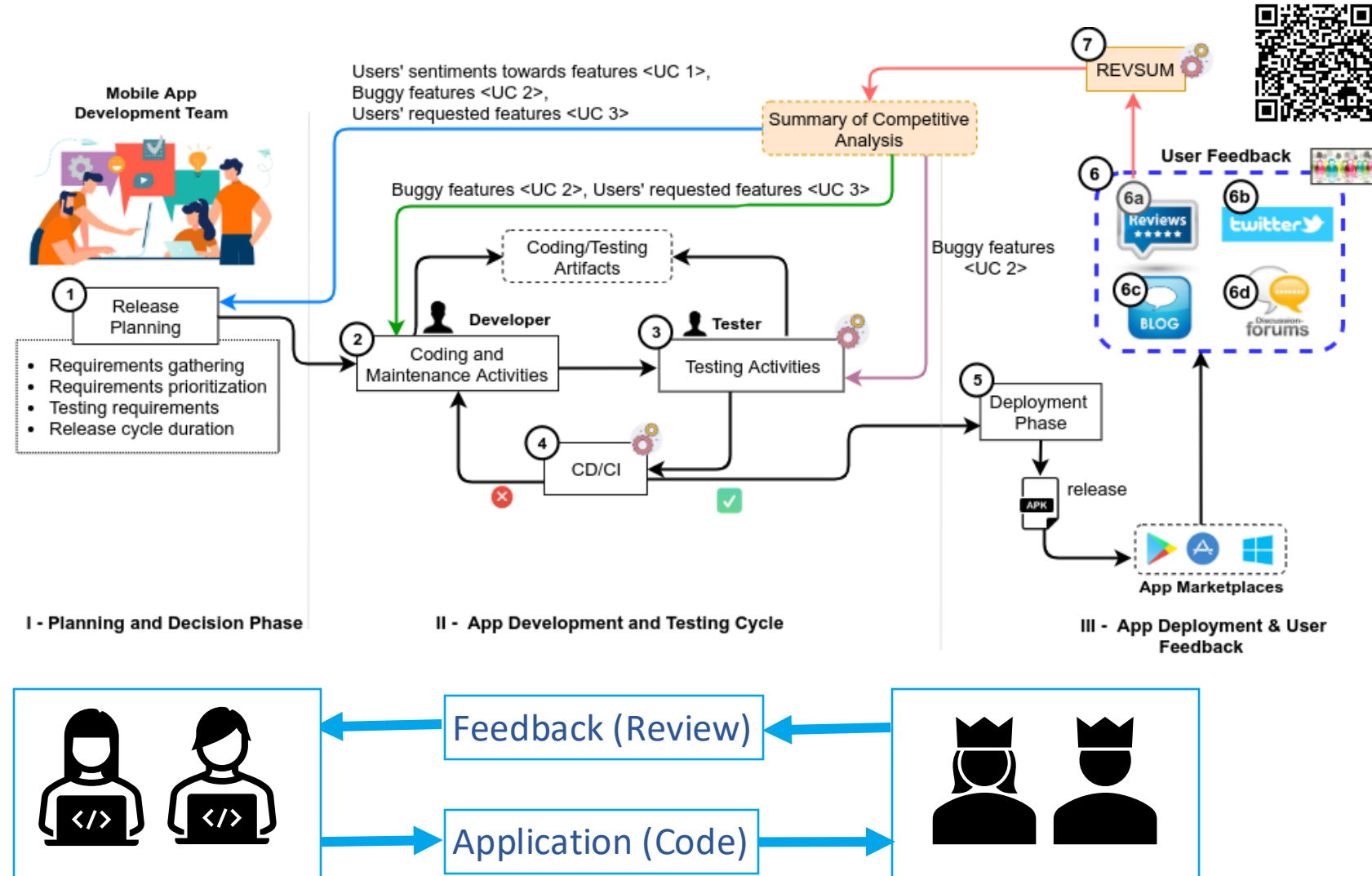
- Automated extraction, selection, and classification of end user reviews (mobile apps) → what features are good/bad/missing?

## Why?

- To support app feature maintenance and evolution decision-making

## Who benefits?

- App developers and (indirectly) app end users





# SEA Project Example 2: Mobile App Energy Efficiency

## Research Topic:

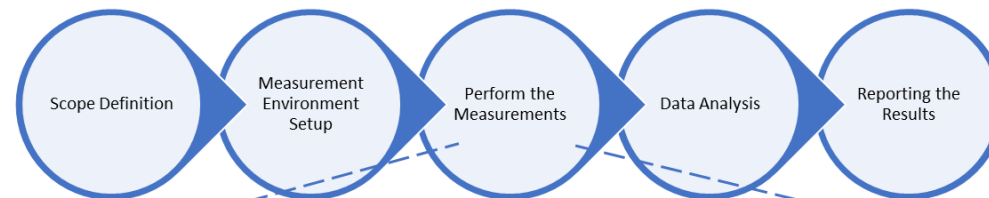
- Analysis of energy efficiency of code refactoring -> trade-off between maintainability and efficiency
- Analysis of energy efficiency of third-part library code -> recommend what library to use for specific tasks, if energy efficiency is important

## Why?

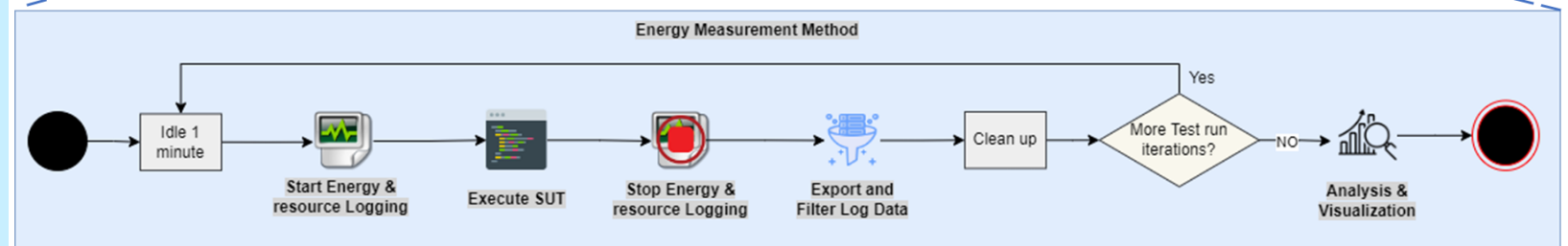
- To recommend energy-efficient refactoring decisions and library choices

## Who benefits?

- App developers and (indirectly) app end users



Measured Metrics	Unit
Duration	s
Mean power draw	W
Energy	Wh or J
CPU usage	%
RAM usage	MB or %



Software-Based Energy Measurement Method

# SEA Project Example 3: Mobile App Security Analysis

## Research Topic:

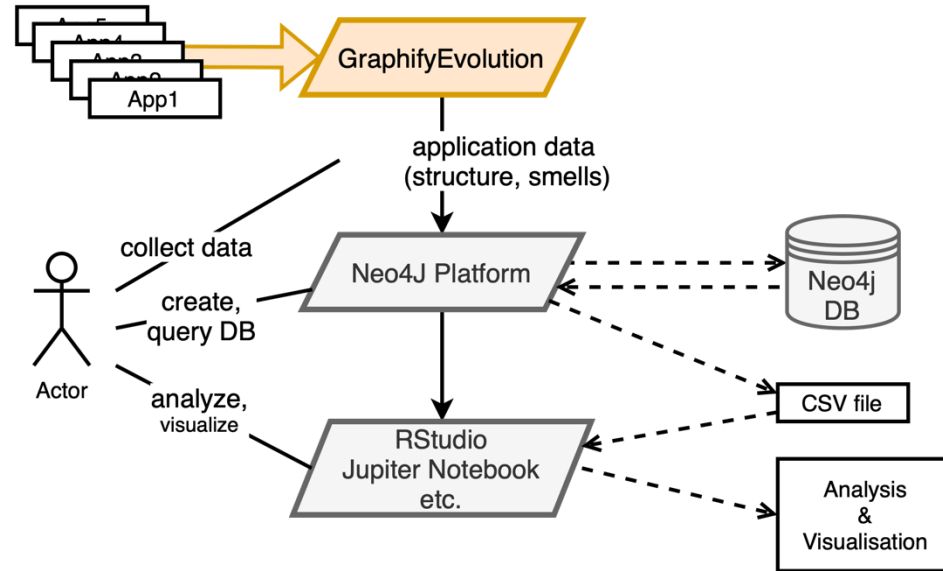
- Analysis of code smells in iOS mobile app code
- Analysis of dependencies to libraries with known security vulnerabilities

## Why?

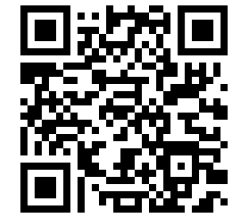
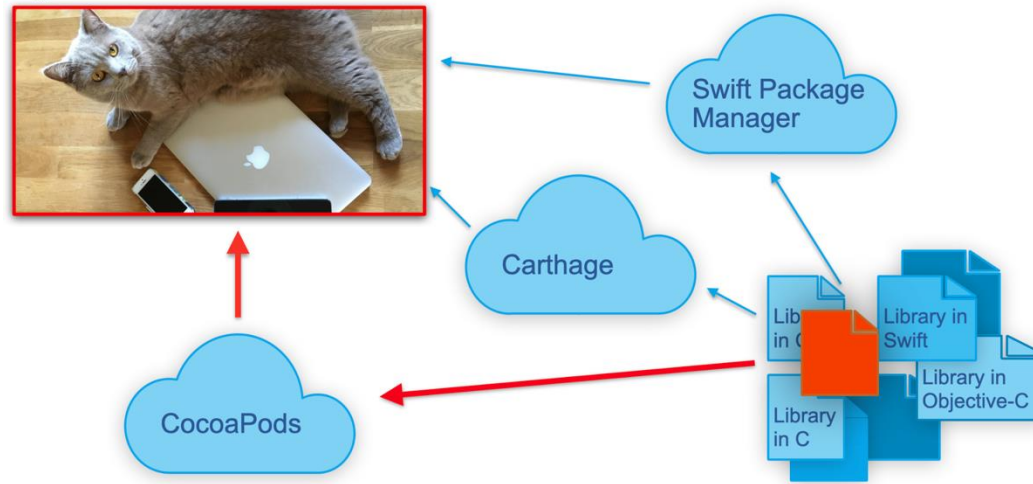
- To reduce the risk of high maintenance cost and dependency on insecure code

## Who benefits?

- Developers of tools that support software developers, software developers, and (indirectly) app end users



```
81 func analyseAppVersion(appVersion: AppVersion) {
82     var finalClasses: [Class] = []
83     var methodsToBeHandled: [Method] = []
84     var variablesToBeHandled: [Variable] = []
85
86     var pathWithoutGit = appVersion.directories
87     if pathWithoutGit.contains(".git") {
88         pathWithoutGit = "\\(appVersion.directories)
89     }
90
91     self.syntaxAnalyser.reset(with: pathWithoutGit)
92     self.fileManager.updateDependencies(pathWithoutGit)
93
94     var includePaths: [String] = fileManager.getIncludePaths()
95     //print("all include paths: \(includePaths)")
96
97     var isMerge = false
98     if appVersion.parent != nil && appVersion.parent.isMerge == true
99         isMerge = true
100 }
101
102 var newClassVersions: [Class] = []
103
104 if !self.noSourceCodeAnalysis {
105     if let parent = appVersion.parent {
106         print("has parent")
107         if let parentApp = appVersion.parentApp {
108
109             let parentClasses = parentApp.classes
110             var altParentClasses: [Class] = []
111
112             var intersectionChanged = [String]()
113             var intersectionParentNew = [String]()
114             var intersectionAltParentNew = [String]()
115             var intersectionNew = [String]()
116
117             if let altParent = appVersion.parentApp {
118                 if let altParentApp = altParentApp {
119                     altParentClasses = altParentApp.classes
120                     for classInstance in altParentClasses {
121                         } else {
122                         }
123                     }
124                 }
125                 intersectionChanged = Array()
126                 intersectionParentNew = Array()
127                 intersectionAltParentNew = Array()
128                 intersectionNew = Array()
129             }
130             var addedClasses: [String] = []
131
132             var combinedPaths: [String] = []
133             combinedPaths.append(content)
134             combinedPaths.append(content)
135             combinedPaths.append(content)
136             combinedPaths.append(content)
137             combinedPaths.append(content)
138             combinedPaths.append(content)
139             combinedPaths.append(content)
140             combinedPaths.append(content)
```



```
124 }
125 }
126 }
127 }
128 }
129 }
130 }
131 }
132 }
133 }
134 }
135 }
136 }
137 }
138 }
139 }
140 }
```

# SEA Project Example 4: Test Oracle Generation

## Research Topic:

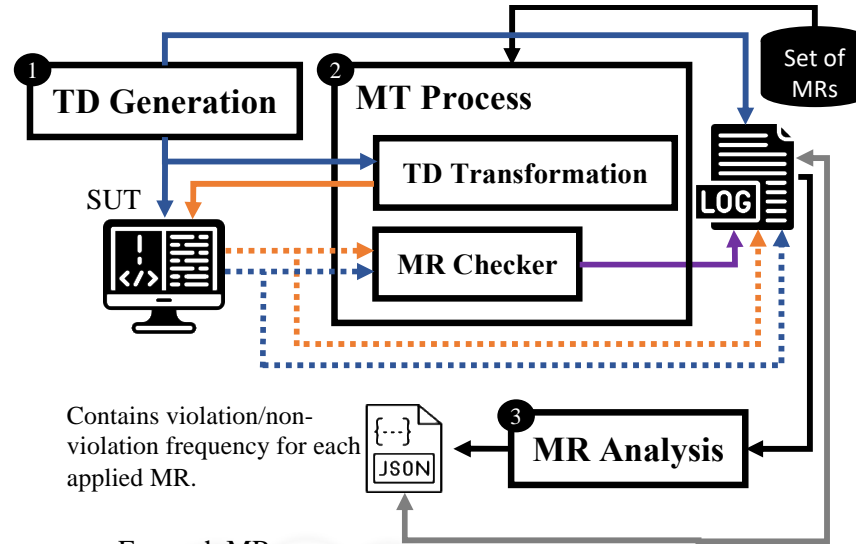
- Combine fuzzing (test data generation) and metamorphic testing to generate effective test oracles

## Why?

- To (semi-)automatically select and constrain metamorphic relations for test suite generation and augmentation

## Who benefits?

- Testers and (indirectly) software developers and software end users



Contains violation/non-violation frequency for each applied MR.

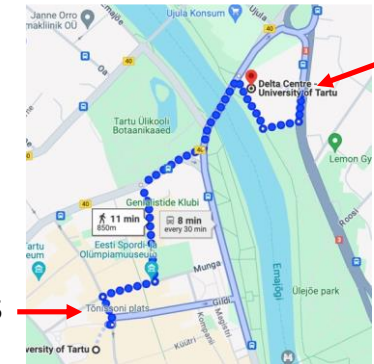
For each MR:

- What is the valid input space in which the MR is violated?
- What is the valid input space in which the MR is not violated?
- What is the input space that results in neither violation nor non-violation of the MR, but causes the system to crash?



Property of map search tool:  
Should the path change,  
if we swap s and d? **à NO**

**Metamorphic Relation (MR):**  
`assertEquals( (SUT(s,d) , SUT(d,s) )`



UT Delta building: d

UT Main building: s

- Test data
- SUT output with the test data
- Transformed test data
- SUT output with the transformed test data
- Manual inspection
- MR violation status (violated / no violated)





# SEA Project Example 5: Simulation-Based Safety Test

## Research Topic:

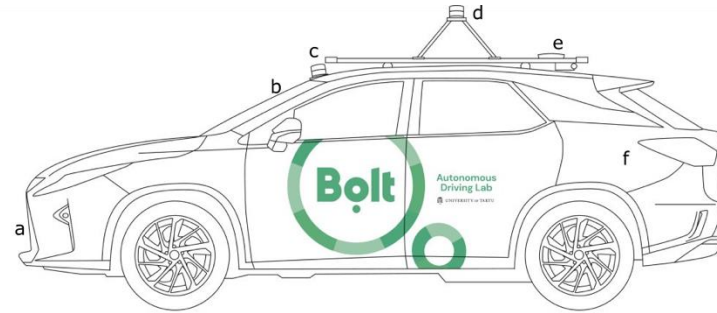
- Methods for simulation-based safety testing of Automated Driving Systems (ADS)

## Why?

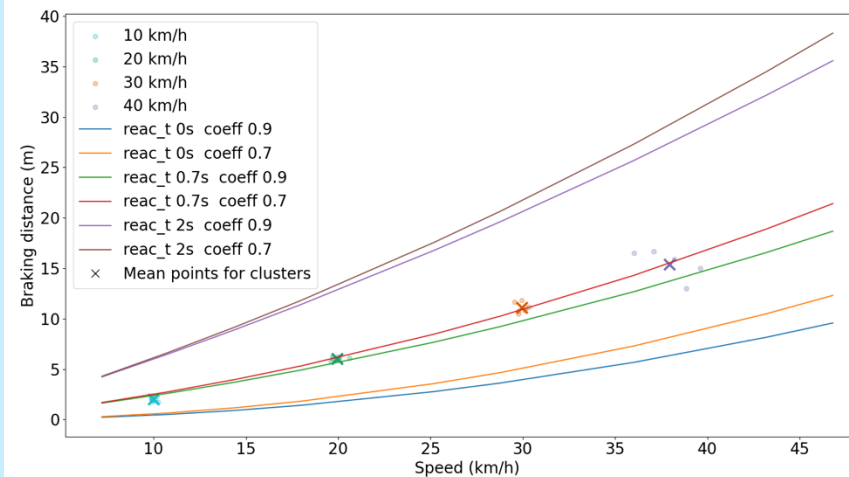
- To help improve the safety of ADS
- To complement on-road testing

## Who benefits?

- Testers and developers of ADS and (indirectly) users of ADS and traffic participants where ADS operate



Comparing the emergency braking of simulated ADS versus theoretical reference modes:



# SEA Project Example 6: Synthetic Test Data for X-tee

## Research Topic:

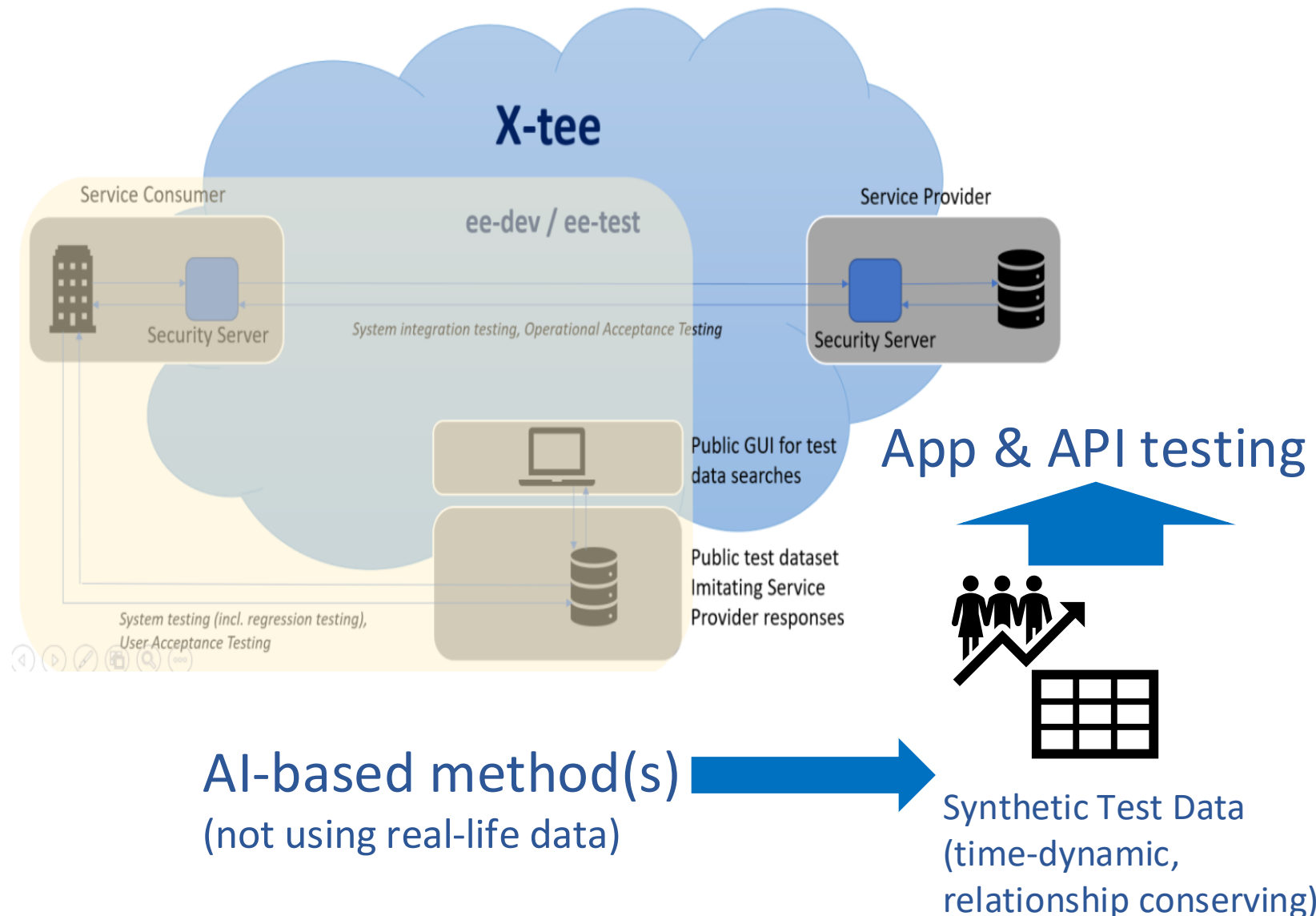
- Methods for automatic generation of relationship-conserving time-dynamic test data sets for the testing of X-tee services

## Why?

- To increase test efficiency and effectiveness of X-tee services

## Who benefits?

- Testers and developers of X-tee services and (indirectly) end users



# SEA Supervision & Teaching

## PhD (since 2014):

- 5 PhD theses completed at UT
- 2 PhD theses completed at University of Calgary
- 5 PhD theses ongoing



## 2 PhD theses started in Sep 2024:

- AI-powered methods for detecting and explaining security vulnerabilities in code
- A method for systematically assessing the safety of automated driving systems via simulation

## MSc & BSc (since 2014):

- 50+ MSc theses completed
- 30+ BSc theses completed



## Courses and Curricula we are responsible for:

- International MSc Software Engineering (program director)
- 3 MSc-level courses (6 ECTS per term / 35 students per term)
- 3 BSc-level courses (7.5 ECTS per term / 200 students per term)

# SEA Supervision & Teaching

## PhD (since 2014):

- 5 PhD theses completed at UT
- 2 PhD theses completed at University of Calgary
- 5 PhD theses ongoing



## 2 PhD theses started in Sep 2024:

- AI-powered methods for detecting and explaining security vulnerabilities in code
- A method for systematically assessing the safety of automated driving systems via simulation

## MSc & BSc (since 2014):

- 50+ MSc theses completed
- 30+ BSc theses completed



**SEA Webpage**



**Thesis Topics**

# SEA Collaboration & Service

**PROFES 2024  
In Tartu !!!**



## Collaboration:

- Within ICS/UT: ADL, Huber Flores' lab, Kairit Sirt's lab, Vesal Voidani's lab (in 4 other chairs)
- Within Estonia: joint MSc/BSc supervisions with companies / RIA
- Within Europe: Several collaborations with Universities and Research Institutes (Fraunhofer, SCCH), several EU project applications with many partners (academia, industry)
- Worldwide: Member of ISERN (International Software Engineering Research Network)

## Service:

- Organizer of international conferences (ICSSP 2015 in Tallinn / ESEC/FSE 2019 in Tallinn / PROFES 2024 in Tartu)
- Member of several OCs, PCs, and steering committees worldwide
- Editorial board member of two top-level international journals (ESEM, INFSOE)
- Volunteering in professional organisations (ACM, IEEE)
- Reviewer of grant application for several public funding agencies in Europe (Austria, Belgium, Finland, Ireland, Sweden, Switzerland, EU)



# Wrapping it up

We are serving the Research Community, the Industry, the Society

by

- Extending the body of knowledge
- Building models and tools
- Educating the future work force

**“To collect and analyze software engineering data with the goal to estimate, predict, and improve the quality of software and software-intensive systems”**

**Software Engineering Analytics – The Team**

**Current Team:**

- Dietmar Pfahl (Professor)
- Hina Anwar (Assistant Professor)
- Faiz Ali Shah (Assistant Professor)
- Kristina Rahkema (Assistant Professor)
- Aleja Duque-Torres (PhD Student, Junior Researcher with SCCH)
- Fauzia Khan (Junior Researcher, PhD Student)
- Maj-Annika Tammisto (Junior Researcher, PhD Student)
- Ali Ihsan Güllü (Junior Researcher, PhD Student)
- Sander Sats (Junior Researcher, PhD Student)
- Maria Khan (Junior Lecturer, PhD Student at TLU)

**Past members of SEA:**

- Ezequiel Scott (Assistant Professor)
- Mariana Falco (Visiting Researcher)
- Tarcio Bezerra (Visiting Researcher)
- Huishi Yin (former PhD Student)
- Riivo Kikas (former PhD Student)
- Laima Anna Dalbina (former MSc Student)

**Other Researcher:** Metamorphic P... assertEquals (SUT)

**System Architecture:** X-tee, ee-dev / ee-test, Service Provider, Security Server, Public GUI for test data searches, App & API testing, AI-based method(s) (not using real-life data), Synthetic Test Data (time-dynamic, relationship conserving)

**Diagrams and Charts:** Mobile App Development Team process flow, GraphyEvolution data flow, Energy monitoring and analysis, TD Generation and MT Process flow, Reference models chart, and a car driving on a road.





Talk to us if you wish to learn more about our research!



Hina



Faiz



Kristiina



Dietmar



unitartu



tartuuniversity

