Eye Tracking Technologies to Analyze and Visualize the Behavior of Secure Coders

Daniel Davis

Doctor of Philosophy – Computer Science – UAH – 2023

Research Advisor / Committee Chair – Dr. Feng Zhu

THE UNIVERSITY OF
ALABAMA IN HUNTSVILLE

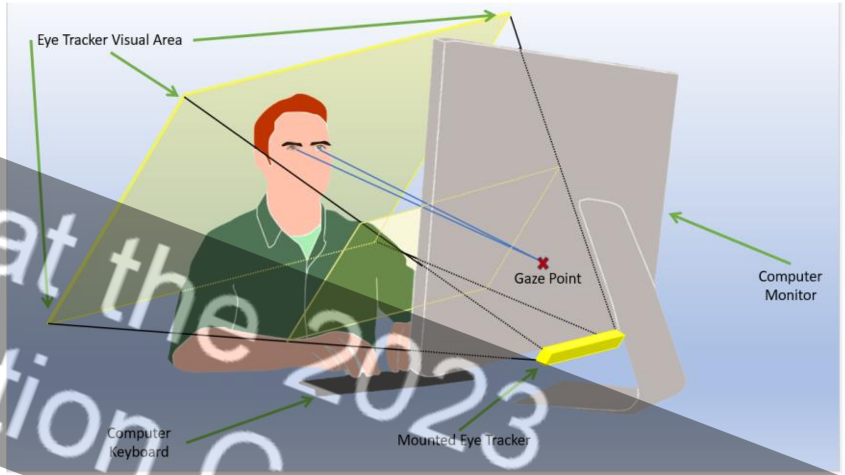
Introduction

- Secure coders techniques and procedures
- Costly coding security flaws in applications
- Common Weakness Enumeration (CWE) repository
- Build hands-on secure coding learning modules
- Eye tracking technologies to capture behaviors
- Literature survey of existing visualization analysis
- Static vs (linear and non-linear) dynamic stimuli
- Eye tracking design and analysis framework
- Decision matrix for designing experiment

- Limitations of current visualization methods used in eye tracking
- Scrolling stimuli
- Participant-editable stimuli
- Creating Areas of Interests (AOIs)
- How do coders find and mitigate security flaws
- How developers read/write code, utilize security tools, and read instructions
- Transitions among eye tracking stimuli and between software application

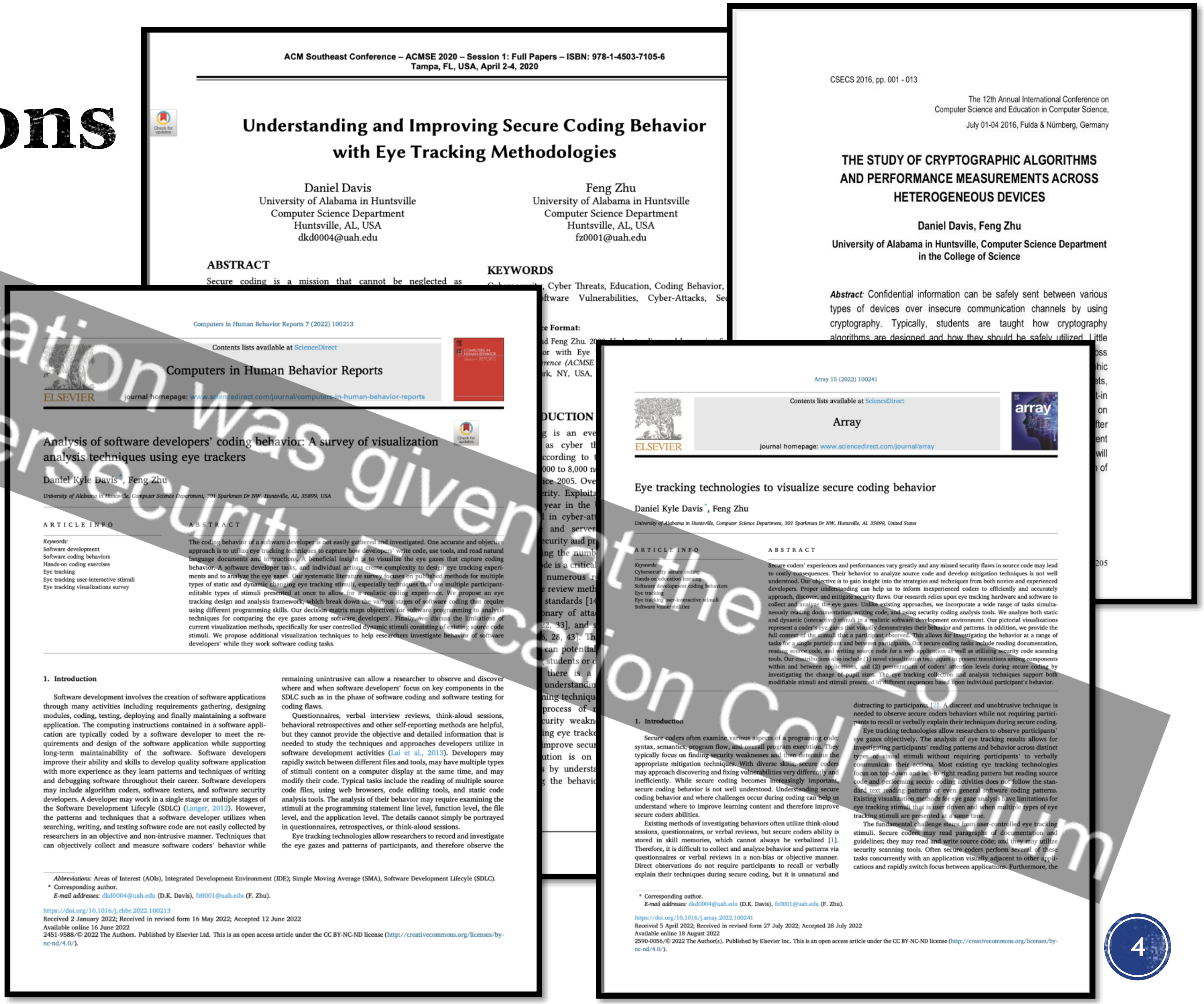
Research Contributions

- Classification of the goals, objectives, participant tasks, and visualization techniques in distinct stages of the SDLC for eye tracking
- Understanding of secure coders' behaviors with multiple types of visualizations of distinct aspects in secure coding and over a timeline
- At the low level, we process eye movements, the speed of movement, the duration of eye fixation, and changes in pupil sizes
- At the medium level, we examine participants' eye gaze at the application and source code files or function level
- At the high level, we present participants' secure coding patterns and strategies
- We propose swimlane diagrams, state transition diagrams, and pupil size fluctuation diagrams
- Developed our Eye Tracking Design and Analysis Framework for software development with a focus on secure coding
- A decision matrix for mapping objectives/tasks in the SDLC to specific aspects of eye tracking design, analysis, and comparison
- Guide on the type of software tasks and eye tracking stimuli to present to participants



Publications

- The Study of Cryptographic Algorithms and Performance Measurements Across Heterogeneous Devices
  - Computer Science and Education in Computer Science (CSECS)
  - July 2016
  - Pages 205 - 219
- Understanding and Improving Secure Coding Behavior with Eye Tracking Methodologies
  - Association for Computing Machinery Southeast (ACMSE)
  - April 2020
  - Pages 107 - 114
- Analysis of Software Developers' Coding Behavior: A Survey of Visualization Analysis Techniques Using Eye Trackers
  - Computers in Human Behavior Reports
  - August 2022
  - Pages 1 - 28
- Eye Tracking Technologies to Visualize Secure Coding Behavior
  - Array
  - September 2022
  - Pages 1 - 34



ACM Southeast Conference – ACMSE 2020 – Session 1: Full Papers – ISBN: 978-1-4503-7105-6
Tampa, FL, USA, April 2-4, 2020

Understanding and Improving Secure Coding Behavior with Eye Tracking Methodologies

Daniel Davis
University of Alabama in Huntsville
Computer Science Department
Huntsville, AL, USA
dkd0004@uah.edu

Feng Zhu
University of Alabama in Huntsville
Computer Science Department
Huntsville, AL, USA
fz0001@uah.edu

ABSTRACT

KEYWORDS

CSECS 2016, pp. 001 - 013

The 12th Annual International Conference on Computer Science and Education in Computer Science, July 01-04 2016, Fulda & Nürnberg, Germany

THE STUDY OF CRYPTOGRAPHIC ALGORITHMS AND PERFORMANCE MEASUREMENTS ACROSS HETEROGENEOUS DEVICES

Daniel Davis, Feng Zhu

University of Alabama in Huntsville, Computer Science Department in the College of Science

Computers in Human Behavior Reports 7 (2022) 100213

Contents lists available at ScienceDirect

Computers in Human Behavior Reports

journal homepage: www.sciencedirect.com/journal/computers-in-human-behavior-reports

Analysis of software developers' coding behavior: A survey of visualization analysis techniques using eye trackers

Daniel Kyle Davis*, Feng Zhu

University of Alabama in Huntsville, Computer Science Department, 301 Sparkman Dr NW, Huntsville, AL, 35899, USA

https://doi.org/10.1016/j.chbr.2022.100213

Received 2 January 2022; Received in revised form 16 May 2022; Accepted 12 June 2022

Available online 16 June 2022

Array 15 (2022) 100241

Contents lists available at ScienceDirect

Array

journal homepage: www.sciencedirect.com/journal/array

Eye tracking technologies to visualize secure coding behavior

Daniel Kyle Davis*, Feng Zhu

University of Alabama in Huntsville, Computer Science Department, 301 Sparkman Dr NW, Huntsville, AL, 35899, United States

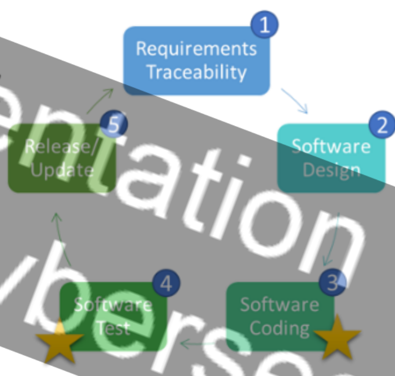
https://doi.org/10.1016/j.array.2022.100241

Received 5 April 2022; Received in revised form 27 July 2022; Accepted 28 July 2022

Available online 18 August 2022

Study Design / Methods

Software Development Lifecycle



★ ➔ *Stages that we focused on in our framework*

High Level Objectives in each SDLC Stages

Software Requirements and Design Stage	Software Coding Stage
• Requirements Definition Documents • Graphical Modeling Languages (UML) • Reading Problem Statement • Design Reviews	• Answering Problem Statements • Reading Source Code • Writing Source Code • Utilizing IDEs • Code Reviews / Code Walkthroughs
Software Testing Stage	**Overlapping Stages of the SDLC**
• Reading Source Code • Testing Code Modifications • Running Code Scanning Tools • Reviewing Coding Analysis Tools • Utilizing IDEs • Code Reviews	• Reading Documentation / User Guides • Researching Online Guidance/ Solutions

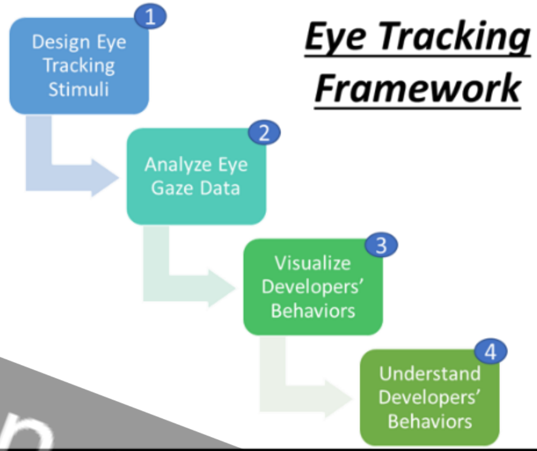
Software Coding

- Reading Source Code
- Writing Source Code
- Using IDEs/Tools

Software Testing

- Reading Source Code
- Using IDEs/Tools

Eye Tracking Framework


1
Design Eye Tracking Stimuli
2
Analyze Eye Gaze Data
3
Visualize Developers' Behaviors
4
Understand Developers' Behaviors

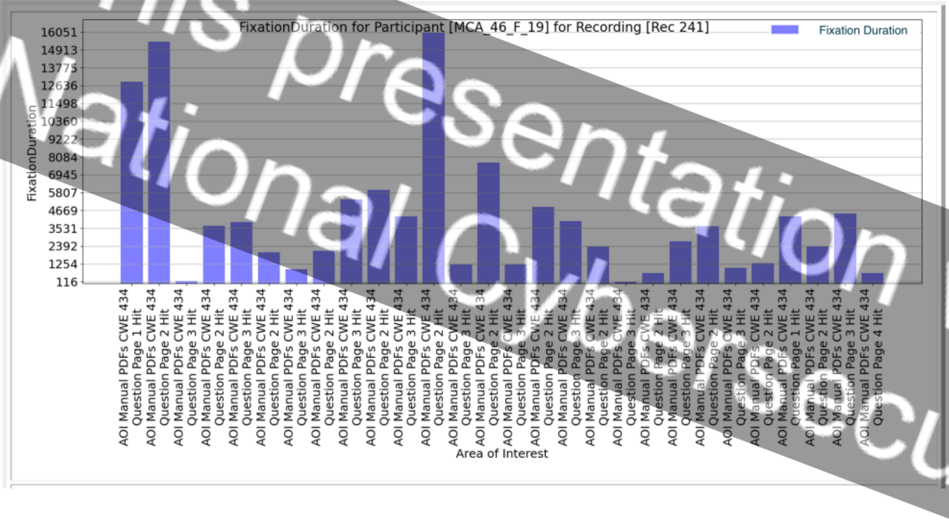
Design of Eye Tracking Stimuli

Types of Tasks	Types of Stimuli	Viewing Dimensions	Developer Interactive
• Natural Language Reading • Reading Source Code • Writing Source Code • Using IDEs/Tools	• Code Static Content • Linear Dynamic Content • Non-Linear Dynamic Content	• Single Stimuli • Side-by-Side Multiple Stimuli • Overlapping Multiple Stimuli	• Developer Non-Changeable • Developer Interactable

Analysis of Eye Gaze Data

Areas of Interest	Comparison Perspective	Quantitative vs Qualitative
• Single • Multiple • Overlaying	• Single Developer • Multiple Developers	• Descriptive Statistical • Statistical Graphic • Infographics

Results



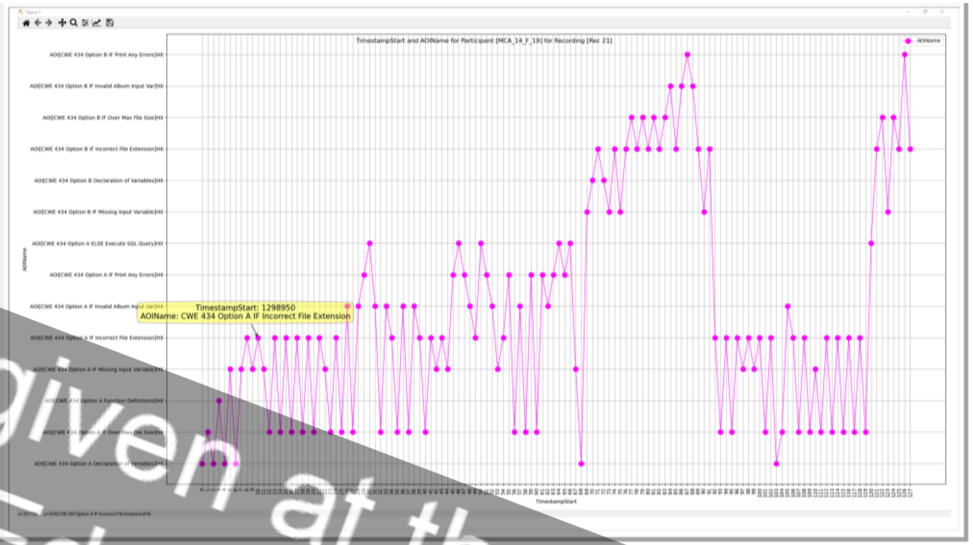
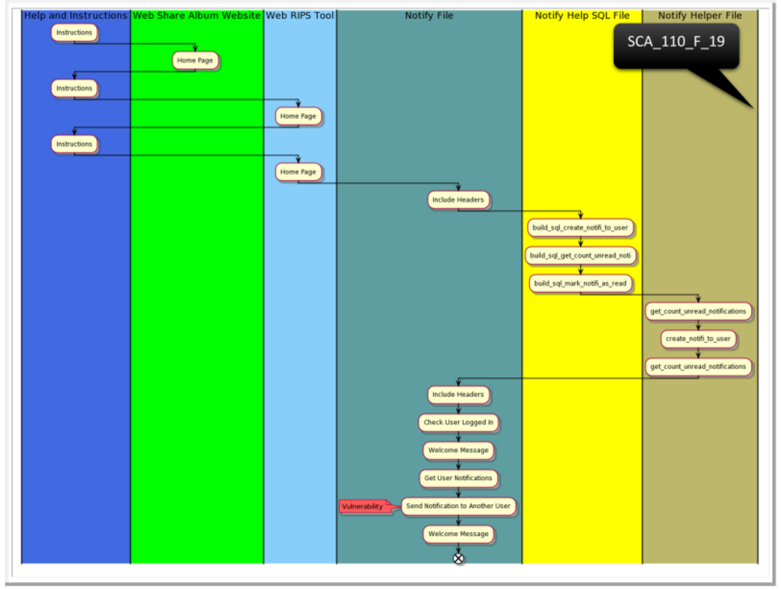
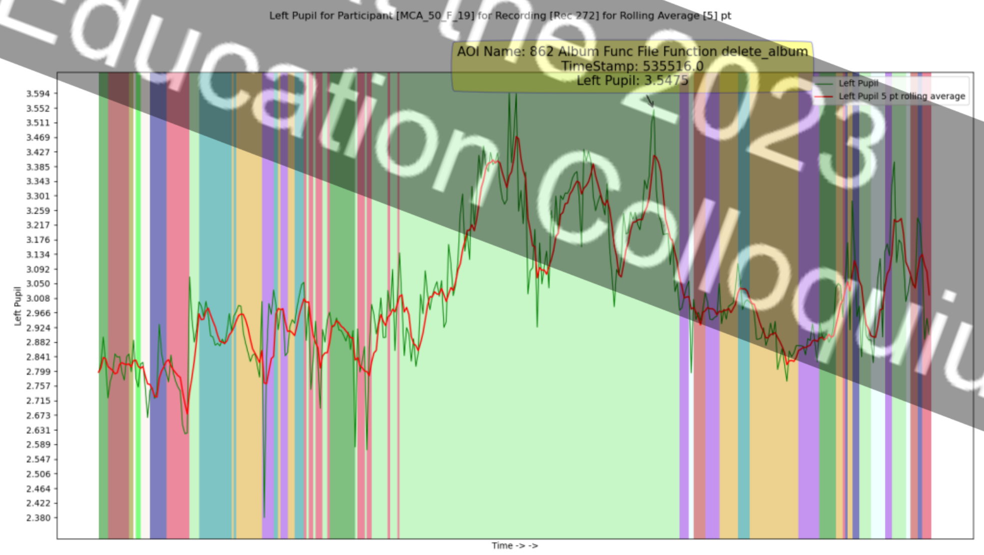


Figure 1
TimestampStart and AOIName for Participant [MCA_14_F_19] for Recording [Rec 21]
AOIName
AOI[CWE 434 Option B IF Print Any Errors]Hit
AOI[CWE 434 Option B IF Invalid Album Input Var]Hit
AOI[CWE 434 Option B IF Over Max File Size]Hit
AOI[CWE 434 Option B IF Incorrect File Extension]Hit
AOI[CWE 434 Option B Declaration of Variables]Hit
AOI[CWE 434 Option B IF Missing Input Variable]Hit
AOI[CWE 434 Option A ELSE Execute SQL Query]Hit
AOI[CWE 434 Option A IF Print Any Errors]Hit
AOI[CWE 434 Option A IF Invalid Album Input Var]Hit
AOI[CWE 434 Option A IF Incorrect File Extension]Hit
AOI[CWE 434 Option A IF Missing Input Variable]Hit
AOI[CWE 434 Option A Function Definitions]Hit
AOI[CWE 434 Option A IF Over Max File Size]Hit
AOI[CWE 434 Option A Declaration of Variables]Hit
TimestampStart: 1298950
AOIName: CWE 434 Option A IF Incorrect File Extension
TimestampStart

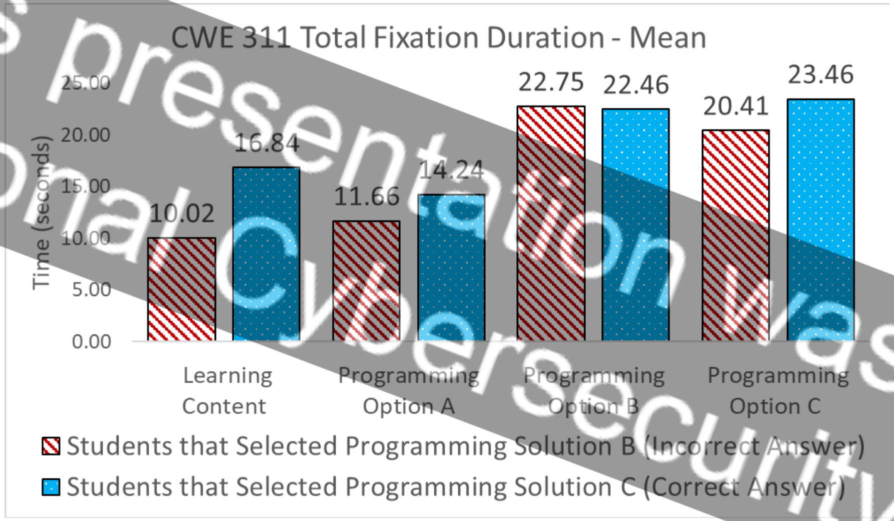

Help and Instructions | Web Share Album Website | Web RIPS Tool | Notify File | Notify Help SQL File | Notify Helper File

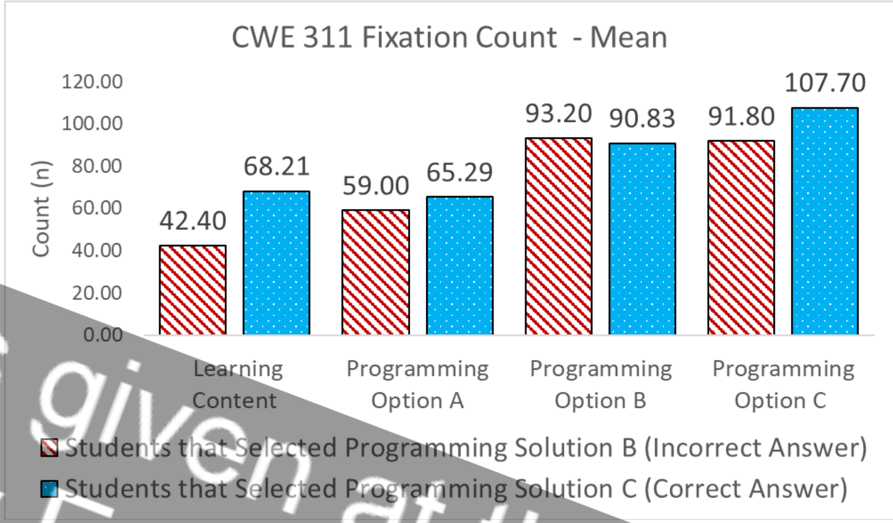
SCA_110_F_19

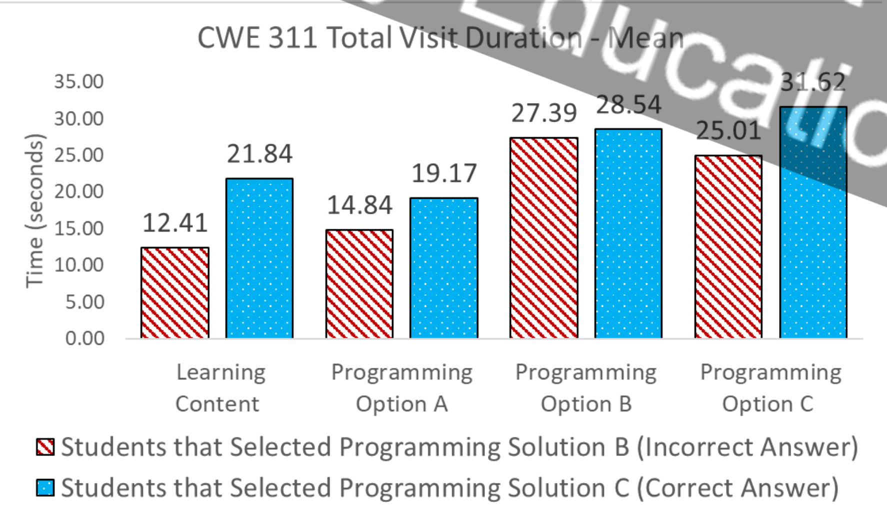
- Instructions
- Home Page
- Instructions
- Home Page
- Instructions
- Home Page
- Include Headers
- build_sql_create_notifi_to_user
- build_sql_get_count_unread_noti
- build_sql_mark_notifi_as_read
- get_count_unread_notifications
- create_notifi_to_user
- get_count_unread_notifications
- Include Headers
- Check User Logged In
- Welcome Message
- Get User Notifications
- Send Notification to Another User — Vulnerability
- Welcome Message



Secure Coder Responses CWE-311


CWE 311 Total Fixation Duration - Mean


CWE 311 Fixation Count - Mean


CWE 311 Total Visit Duration - Mean

Ongoing Work

- Our focus has been on analyzing the data at both the low-level and high-level
- One area we are investigating is to compare the behaviors of those that answered correctly with those that did not answer correctly
  - Methods being explored
    - Manually Analysis
    - Automated Analysis (low-level or high-level) (machine learning)
  - CWE Problems being explored
    - True/False if Software Flaw – CWE-443 or CWE-73
    - Programming Problem – CWE-862 or CWE-22
- Analyze reading patterns between novice and expert secure coders
  - Majority of our data is with novice secure coders

QUESTIONS / COMMENTS