Supplementary Material: One Size Does Not Fit All: A Grounded Theory and Online Survey Study of Developer Preferences for Security Warning Types

Anastasia Danilova
University of Bonn
danilova@cs.uni-bonn.de

Alena Naiakshina
University of Bonn
naiakshi@cs.uni-bonn.de

Matthew Smith
University of Bonn, Fraunhofer FKIE
smith@cs.uni-bonn.de

ACM Reference Format:
Anastasia Danilova, Alena Naiakshina, and Matthew Smith. 2020. Supplementary Material: One Size Does Not Fit All: A Grounded Theory and Online Survey Study of Developer Preferences for Security Warning Types. In *42nd International Conference on Software Engineering (ICSE '20), May 23–29, 2020, Seoul, Republic of Korea.* ACM, New York, NY, USA, 20 pages. https://doi.org/10.1145/3377811.3380387

1 STRUCTURE OF SUPPLEMENTARY MATERIAL

Our supplementary material is structured as follows. In Section 2, we provide information on our qualitative study and in Section 3, we provide information on our quantitative study.

The **qualitative study** material covers

(1) the **semi-structured guideline** including a description and a reference to each type of warnings, (Section 2.1)
(2) **participants' demographics** (Section 2.2),
(3) the **codebook** (Section 2.4.1) and,
(4) **themes** extracted through our Grounded Theory (GT) approach (Section 2.4.2 and Figure 1).

The **quantitative study** material covers

(1) the **visual representation of the variations of warning types** in the online survey (Section 3.1),
(2) the **invitation text** to our survey (Section 3.2),
(3) **survey questions** including the **programming test** (Section 3.3),
(4) **participants' demographics** (Section 2.2 and 2.3) and,
(5) participants' **rating of the different types of warnings** (Section 3.5).

2 QUALITATIVE STUDY

Participants were presented different types of warnings. A generic IDE drawing was compiled for all warnings, because participants were assumed to have experiences with different IDEs. Additionally, the study's focus was on warnings in general, instead of the details differing between IDEs.

ICSE '20, May 23–29, 2020, Seoul, Republic of Korea

ACM ISBN 978-1-4503-7121-6/20/05.
https://doi.org/10.1145/3377811.3380387

2.1 Semi-structured Guideline

Please state who you are and what kind of development work you do. Please also tell us what editor or IDE you use for your work. We want to investigate which warnings types would be helpful. We will present different warning approaches and we would like to hear your opinion on each of them.

- **Compiler Warning.** The first is the compiler warning. In foobar, something happens that triggers a warning. The warning is displayed in the command line output in the IDE (see Figure 15).
- **Warning Marker (Yellow).** After that, we have two markers for the warning. The first is the usual yellow marker. As you can see, the function is underlined in yellow, and there is a small caution button besides the line number (see Figure 16).
- **Warning Marker (Own Color, Blue).** Secondly, we could define an own color, we chose blue and a small lock symbol to indicate that it is a security warning (see Figure 17).
- **Pop-up Warning.** Here, you see the pop-up warning. The warning would be displayed in a pop-up as soon as it is triggered (see Figure 18).
- **Plugin Warning.** Currently, there are already plugins which display warnings. To do so, they often use a violations outline in a corner of the IDE. Further, an arrows marks the warning priority (see Figure 19).
- **Security View.** Another warning type is a security warning view in the IDE and an own button just like the debug or run button (see Figure 20).
- **Warning on Committing.** Another use case would be to warn a developer when committing code. In this case, a pop-up appears (see Figure 21).
- **[Added:] Warning on Committing: Automated Security Ticket.** Finally, this is a committing warning where you would generate an automatic security ticket (see Figure 22).

- Which warning types would you use? Why?
- (For each warning type:) What do you think about this approach?
- When should this warning be displayed? (During implementation/after implementation/before committing.)
- How often should this warning be displayed? (Once/more often:...)
- Could you think of a better approach?

Security responsibility: [Added for professional developers]

- Is there a person responsible for code security in your company? Do you have a security responsible person in your company who you could consult?
- Do you feel responsible for the security in code? While coding, when do you think about security? Is security is part of the process?
- Would you like to delegate a warning to another developer?

Experiences with breaches: [Added for professional developers]

- Have you already experienced security breaches in the past? Which ones?
- If yes: Do you think one of these warnings would have prevented the security breach?
- Are you familiar about warnings, or have you had any experience with them while programming?
- Do you have examples of cases in which security warnings could be useful?
- There are different cases where security warnings could be triggered to inform the developer that they are programming potentially non-secure code. Could you think some examples?
- Do you think security warnings for developers could be useful?
- Which color would you prefer for security warnings?
- Would you like warnings to be deactivated in case of a test project ?

2.2 Demographic Questions

- Please select your gender. *Female/Male/Other/Prefer not to say*
- Age: *[free text] years*
- What is your current occupation? *Freelance developer Industrial developer/ Industrial researcher/ Academic researcher/ Undergraduate student/ Graduate student/ Other: [free text]*
- Do you have a university degree? *Yes/ No*
- *Primarily for students:* Currently, do you have a part-time job in the field of Computer Science? If yes, please specify: *[free text]*
- At which university/universities were/are you enrolled? *[free text]*
- Were/Are you taught about IT-security at your university? *Yes/ No / I don't recall*
- Which security lectures did you pass in your Bachelor's/Master's program? *[free text]*
- Were/Are you taught about IT-security extramural? *Yes/ No*
- What was your main source of learning about IT-security? *[free text]*
- How did you gain your IT skills? *[free text]*
- How did you gain your IT-security skills? *[free text]*
- What type(s) of software do you develop? *Web applications/ Mobile applications/ Desktop applications/ Embedded applications/ Enterprise applications/ Other (please specify)*
- Which programming languages do you know? *[free text]*
- How many years of experience do you have with software development in general? *[float] years*
- Which IDEs do you use? *[free text]*
- Which tools do you use for software development? *[free text]*
- What is your nationality? *[free text]*
- Thank you for answering the questions! If you have any comments or suggestions, please leave them here: [free text]

2.3 Demographics of Participants

Our participants reported to have used or currently use the following IDEs: Eclipse [6], Netbeans [14], IDEA Intellij [9], VisualStudio [13], PyCharm [11], Sublime [12], CLion [8], PhpStorm [10], Android Studio [16], Spyder [5], Notepad++ [1], Code::Blocks [2], Geany [7], ABAP Workbench [15], Google Colaboratory [3].

Also the fields in which our participants had worked in were quite diverse: system oriented, networks, SAP-development, machine learning, monitoring, databases, web development, open source projects, security, mobile development etc.

Further, our participants reported to be familiar with different programming languages: Java (24), Python (22*), C/C++ (18), JavaScript (6*), PHP (6*), C# (3*), ABAP (3*), Go (2*), Pascal (2*), Haskell (2*), Rust (1*), VisualBasic (1*), Cobal (1*), Groovy (1*), Perl (1*), Lisp (1*) and Ruby (1*). The symbol * indicates "out of 33 participants."

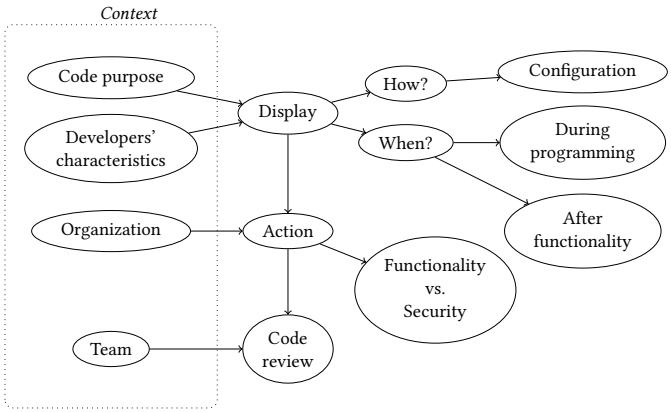

Figure 1: Themes extracted using grounded theory.

2.4 Grounded Theory Analysis

2.4.1 Codebook. In Table 1 we present our codebook with at least one example quote from the interviews per category.

2.4.2 Themes. As suggested by Charmaz [4], we used the idea of theoretical sorting to develop a diagram out of our memos, categories and themes. Figure 1 provides an overview of the themes that came out of the grounded theory study. Categories and memos have informed the themes. The following themes have evolved out of the interviews. Occurring themes could change according to the theoretical sample approach within the sample process:

(1) Student participants and freelance developer:
   - **Functionality first, security second.**
     - Committing and pop-up security warnings are preferred after finishing the programming functionality.
   - **Security warnings during programming and after finished programming.**

  - Committing and pop-up security warnings are preferred after finishing the programming functionality.
  - Warning marker are desired during programming.
  - Security view should be available at any time.
- **Disabling security warnings depends on use-cases.**
  - Some believed disabling should not be allowed for security warnings and others believed it depends on situations. For instance, if developers only test their program, they do not want to be displayed security warnings.
  - However, disabled security mode should always be marked!
- **Extra view lead to an additional overhead.**
  - Participants believed that too many views could be disturbing for them, because already a lot of views exist in the IDEs.
- **Security deserves an extra warning, e.g., with an own color.**

(2) Security start-up participants:
- **Functionality and security equivalent important**
  - No merge requests should be allowed if security issues are not fixed. Additionally, others should see the security issues on merge requests as well.
  - Wish to involve team members to fix security issues.
  - Security ticket on committing requested.
  - Warnings if pulling security issues.
- **Security warnings during programming and after finished programming.**
  - Combination of warning on committing and warning marker with an own security color and symbol preferred.
- **Disabling security warnings depends on use-cases.**
  - Security warnings are important, but it should be possible to disable them in some use-cases or in a testing scenario.
- **Extra view could lead to an additional overhead.**
  - Participants believed that too many views could be disturbing for them, because new views always require familiarization each time.
- **Quality (security) costs time.**
- **Developer the weakest link.**
  - It depends on the developer how secure APIs are.
- **Workflow integration important.**
  - Developers using the command line would not see security warnings in an IDE.
  - Pop-ups intrusive.
- **Warnings keep developers up-to-date.**

(3) Academic focus group participants:
- **Trade-off between functionality and security.**
  - On the one hand, developers should be warned about security issues in the moment when they arise. On the other hand, developers should not be kept from working.
  - Security interrupts the development process.
- **Testing mode could be insecure.**
  - Testing mode without security warnings should only be available for security aware developers.
  - Developer should be warned that her/his security warnings are disabled when leaving the testing mode.
- **Configuration.**
  - There will be never one perfect warning system. It depends on habits, experience and different views.
  - Pre-defined preference profiles of warning types should be suggested to developers.
  - Pop-ups only if requested and for "are you sure" questions.
- **Security responsible person in teams requested.**
- **Text completion (secure parameter defaults) for security requested.**

(4) Government institution participants:
- **Functionality first, security second.**
  - After finished programming. Reasons: Security requires time, time pressure of the company, security not demanded by the company.
- **Warnings during programming and after finished programming.**
  - During programming. Reasons: could be too late otherwise, developers rather fix issues rather if they are shown immediately (during programming).
  - After finished programming. Reasons: security requires time, time pressure of the company, security not demanded by the company.
- **Disabling security warnings depends on use-cases.**
- **Success of the security ticket on committing warning depends on the team size.**
  - Unclear responsibility.
- **Security team/policyies in company do not have knowledge in programming and nobody checks code for security.**
- **Specific guideline for specific project requested.**
  - Security policy guidelines of the company were not developed from tech people.
  - Developers overloaded with information.
- **Configuration.**
  - Like the idea of having a list with warnings: issues, which are solved can be deleted and others will stay there.
  - Pop-up intrusive if working with key board.
  - Security view good as a summary for security issues found by different tools.
  - Would like to get additional information (source) for security warning.
  - Warning marker should be shown within source code.
  - Committing: idea of being reminded at the end of the working process about security issues developers might forgot or have overseen; "stop sign".
  - Combination of security marker warning and the security view warning.

(5) Financial sector participants:
- **Company decides about the trade-off between functionality and security.**
  - Participants perceive functionality and security as equally important.
  - Security checks could prevent from working.
  - Company with no demand for security, does not force to consider security while programming.

- **Security warnings after finished programming.**
  - For workflow integration, security is prioritized at the end.
- **Configuration.**
  - Companies with security context prefer the security ticket warning and would like to disable warnings in appropriate use-cases.
- **Good security warnings important.**
  - For end-user software outside world.
  - Current warning systems prevent from developing.
- **Better education instead of security warnings.**
- **Team integration important.**
- **Code review important matters.**
- **Security team/policies in company do not have knowledge in programming and nobody checks code for security.**

3 QUANTITATIVE STUDY

After our qualitative analysis, we built a quantitative survey to be able to quantify our results. Usually all multiple choice responses were shown randomized to our participants.

3.1 Types of Warnings

The different types of warning variations were presented to our participants:

- **Security View:** see Figure 2.
- **Warning on Committing with Security Ticket:** see Figure 3.

Figure 2: Security view

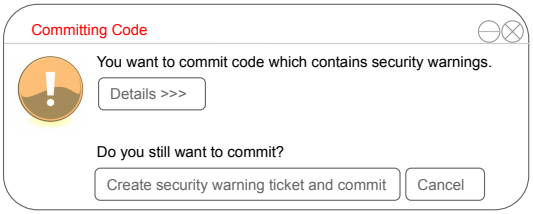

Figure 3: On committing with security ticket

3.2 Invitation

Receive a 20 euro Amazon voucher by answering a 10-15 minutes survey! We are researchers from the University of Bonn and are looking for motivated software developers who will take part in a 10-15 minutes survey. The survey is about a topic in software development and is conducted in English in order to ensure an international comparison. We look for about 25-30 participants. Interested? Find out whether you are a suitable candidate for our survey by answering two questions: LINK

3.3 Survey

- In order to make sure you are a suitable candidate for our research, we would like you to answer three questions before starting with the study.
  How old are you? [free text]
- Where are you employed?
  - ☐ Africa
  - ☐ Asia
  - ☐ Australia
  - ☐ Canada
  - ☐ New Zealand
  - ☐ UK
  - ☐ USA

```
main{
 print(func("hello world"))
}

String func(String in){
    int x = len(in)
    String out = ""
    for(int i = x-1; i >= 0; i--){
        out.append(in[i])
    }
    return out
}
```

Figure 4: Test for Software developing skills

- Please select the returned value of the pseudo code above[see Figure 4]:
  - ☐ hello world hello world hello
  - ☐ world hello

- ☐ world hello world
- ☐ hello world 10
- ☐ HELLO WORLD
- ☐ hello world
- ☐ dlrow olleh

- **Introduction:**
  We are researchers from the University of Bonn and are investigating security warnings for developers. There are tools such as static analyzers which can inform developers about security issues in their code. To do so, these tools need to display warnings to the developer. We are investigating how and when security warnings can be displayed in an Integrated Development Environment(IDE). By taking part in our study you will help us understand when and how developers would prefer to be warned about security issues in their code. For this we will show you different types of warning and ask for your opinion on these. There are no right or wrong answers since we are purely interested in your opinion.
- **Consent:** I have read and understood the informed consent form. I consent that the data from the survey can be used for research purposes. Researchers will have access to this data only if they agree to preserve the confidentiality of the data and if they agree to the terms specified in the consent form. Only anonymised data will be published.
  - ☐ I consent
- Which IDE (integrated development environment) do you use most?
  - ☐ Netbeans
  - ☐ Eclipse
  - ☐ Microsoft Visual Studio
  - ☐ IntelliJ IDEA
  - ☐ Other:
- Compiler Warning:
  Below you see an example of how a security warning could be presented by the compiler. The warning would be displayed in the command line output in your IDE .
- Please state your agreement to the following items.*1: Strongly Disagree - 7: Strongly Agree*
  - I would like to be informed about security issues in my code with this type of warning.
  - I would be quickly annoyed by this type of warning
  - It would be easy to overlook this type of warning.
  - IDEs already have too many warnings of this type.
  - I am familiar with this type of warning.
- Markers:
  Below you see examples of how markers can be used to present a warning. The warning is displayed within the code editor of the IDE. The piece of code triggering the warning is underlined and there is an icon next to the line of code. Different colours can be used for the underlines. Yellow markers are commonly used to signify warnings and red is used to signify errors. Security warnings could also use these colours or use their own colour. You can hover over the icon to get more information about the warning.
- In which colour would you like your IDE to underline code to highlight security warnings?
  - ☐ Black
  - ☐ Brown
  - ☐ Blue
  - ☐ Red
  - ☐ Yellow
  - ☐ Green
  - ☐ Grey
  - ☐ Pink
  - ☐ White
  - ☐ Violet
  - ☐ Orange
  - ☐ Other:
- Assuming the IDE uses the colour you chose above: Please state your agreement to the following items.*1: Strongly Disagree - 7: Strongly Agree*
  - I would like to be informed about security issues in my code with this type of warning.
  - I would be quickly annoyed by this type of warning
  - It would be easy to overlook this type of warning.
  - IDEs already have too many warnings of this type.
  - I am familiar with this type of warning.
- Pop-up:
  Below you see an example of a pop-up warning. The warning would be displayed by your IDE as soon as you complete a code statement that triggers a security warning.
- Plugin:
  Below you see an example of how an IDE view can be used to show warnings. Such a view commonly already exists to give an overview of non-security/general warnings and errors. Security warnings could be added to this view or receive a dedicated security warnings view. You can click on the security-warning to receive additional information.
- I would prefer...(only one selection possible)
  - ☐ security warnings to be displayed in a dedicated security warnings view.
  - ☐ security warnings to be displayed in the same view as the general warnings and errors.
- For the view you selected above: Please state your agreement to the following items.*1: Strongly Disagree - 7: Strongly Agree*
  - I would like to be informed about security issues in my code with this type of warning.
  - I would be quickly annoyed by this type of warning
  - It would be easy to overlook this type of warning.
  - IDEs already have too many warnings of this type.
  - I am familiar with this type of warning.
- Warning on Committing:
  Below you see an example of a pop-up security warning which would be displayed in your IDE when you try to commit code which contains a security issue.
- Please state your agreement to the following items.*1: Strongly Disagree - 7: Strongly Agree*
  - I would like to be informed about security issues in my code with this type of warning.
  - I would be quickly annoyed by this type of warning
  - It would be easy to overlook this type of warning.
  - IDEs already have too many warnings of this type.
  - I am familiar with this type of warning.

- Warning on Committing Ticket:
  Below you see an example of a pop-up security warning which would be displayed in your IDE when you try to commit code which contains a security issue. If the developer proceeds with the commit, a ticket is created to keep track of the issues.
- For whom should the ticket be created? *Someone responsible for security issues; Myself; I don't know what a ticket is.; Other:*
  Assuming the ticket is created for the person you selected above: Please state your agreement to the following items. *1: Strongly Disagree - 7: Strongly Agree*
  - I would like to be informed about security issues in my code with this type of warning.
  - I would be quickly annoyed by this type of warning
  - It would be easy to overlook this type of warning.
  - IDEs already have too many warnings of this type.
  - I am familiar with this type of warning.
- When would you like to be warned about security issues in your code? (Multiple answers possible)
  - ☐ While coding - right away
  - ☐ While coding - after completing a function
  - ☐ While coding - in regular intervals
  - ☐ Before running
  - ☐ Before committing
  - ☐ Before release
  - ☐ On demand (e.g. by clicking a button or opening a view)
  - ☐ Never
- Do you think an IDE should offer different kinds of warnings so developers can choose their preferred way of being warned? *Yes; No*
- If you are using or in the past have used a static analyser please select the one you have the most experience with. If you have never used a static analyser please select None.
  - ☐ Checkmarx Static Code Analysis
  - ☐ Clang
  - ☐ CodeSonar
  - ☐ FindBugs
  - ☐ Fortify
  - ☐ PMD
  - ☐ None
  - ☐ Other:
- Please state your agreement with the following statements with respect to the static analyser you have the most experience with. *1: Strongly Disagree - 7: Strongly Agree*
  - It was easy to collaborate as a team when using the tool.
  - The output results of the tool usually offered me a fix for the problem.
  - The output results of the tool were easy to understand.
  - I feel that the tool helped me with my programming work.
  - The amount of output of the tool was reasonable. The percentage of false-positive was acceptable.
  - I would recommend this tool to other programmers.
  - It was possible to customize the tool.
  - Using the tool could be easily integrated into my workflow.
- Do you currently use static analyzers or other code checking tools? *Yes; No*
- What are the reasons you don't use / stopped using static analyzers or code checking tools? (Multiple selection possible)
  - ☐ I don't know how to use them
  - ☐ They are too annoying
  - ☐ I don't know any tools
  - ☐ Using them is too time consuming
  - ☐ I am not interested in security
  - ☐ I never thought about using tools
  - ☐ They are not effective
  - ☐ I am not responsible for code security
  - ☐ They are too expensive
  - ☐ Other:
- Please state your agreement: *1: Strongly Disagree - 7: Strongly Agree*
  - I would like to be able to snooze security warnings.
  - Security issues should be treated as blocking errors and not warnings.
  - Security issues should block merge requests.
  - I would like to be able to turn off security warnings entirely.
  - This is an attention check question. Please select "1" as your answer.
  - I would like to be able to disable all security for periods of time (e.g. prototyping stage)
- In your current position how many security warnings do you see while programming?
  - ☐ More than 5 a day
  - ☐ Less than 5 a day
  - ☐ Less than 5 a week
  - ☐ Less than 5 a month
  - ☐ Less than 5 a year
  - ☐ none
- Who do you think is responsible for fixing functionality bugs?
  - ☐ Me
  - ☐ Someone else:
- Who do you think is responsible for fixing security bugs?
  - ☐ Me
  - ☐ Someone else:
- Please select what is more important ...(Slider between Functionality and Security , Equally important at 50%, Not Applicable)
  - ... to you:
  - ... to your organization:
  - ... your team:
  - ... your direct supervisor:
- Does your organization have security coding policies? *Yes; No*
- Is there a person or team in your organization who is responsible for code security? *Yes; No*
- Are there colleagues who you can ask on an informal base in cases of security issues in your code? *Yes; No*
- How are your project team's total software security efforts divided among the following stages? [Must add up to 100]
  - The design stage
  - While implementing the code
  - During testing by developers

  - During code analysis (e.g. using static analysis tools)
  - During code review
  - During testing that is done by someone other than the code owner
- Have you already experienced a security incident .... *Yes;No*
  - involving code you created?
  - involving code created by a colleague?
  - involving code created by a third party which was integrated into your software?
- I have a good understanding of security concepts. *1: Strongly Disagree - 7: Strongly Agree*
- Please rate the following items: *1: Never - 7: Every time*
  - How often do you ask for help when faced with security problems?
  - How often are you asked for help when others are faced with security problems?
  - How often do you need to add security to the software you develop?
- What percentage of your programming time do you spend on security? (0-100)
- **Demographics:**
  - Age: *.. years*
  - Please select your gender. *Female/Male/Prefer not to say/ Diverse:*
  - What is the highest education level you have completed?
    - ☐ High school graduate (high school diploma or equivalent including GED)
    - ☐ Some college but no degree
    - ☐ Associate degree in college
    - ☐ Bachelor's degree in college
    - ☐ Master's degree
    - ☐ Doctoral degree
    - ☐ Professional degree (JD, MD)
    - ☐ None
  - In which country do you mainly work? *[free text]*
  - What type(s) of software do you develop? *Web applications/ Mobile applications/ Desktop applications/ Embedded applications/ Enterprise applications/ Other (please specify)*
  - How many years...
    - .. of programming experience do you have?
    - ... have you been working in a job where software development was a substantial part of your activity?
    - ... was security relevant for your programming work?
  - How many employees work in your organization?
    - ☐ 1-4
    - ☐ 5-9
    - ☐ 10-19
    - ☐ 20-49
    - ☐ 50-99
    - ☐ 100-249
    - ☐ 250-499
    - ☐ 500-999
    - ☐ 1000 or more
    - ☐ I don't work in any organization.
    - ☐ I prefer not to answer.
  - How many people work in your team? Please enter 1 if you work on your own.

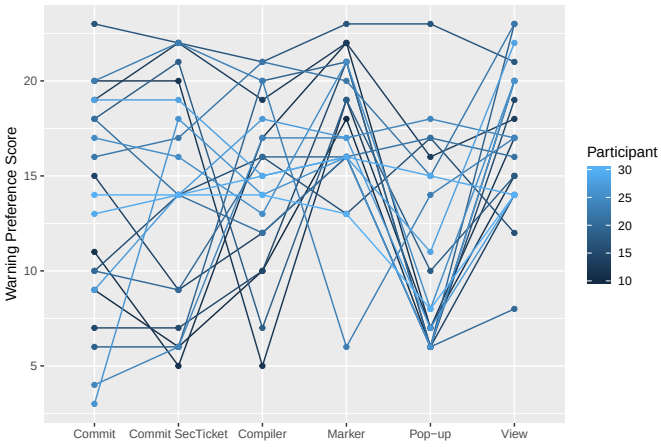

Figure 5: Preference score for 20 random participants [batch 2]

- What is your current occupation?
  - ☐ Freelance developer
  - ☐ Industrial developer
  - ☐ Industrial researcher
  - ☐ Academic researcher
  - ☐ Undergraduate student
  - ☐ Graduate student
  - ☐ Other: [free text]
- Thank you for taking part in our study! We really appreciate your time and effort. We hope our results will help improve how IDEs assist developers in their work. If you have any comments or suggestions, please leave them here and then please click on "Continue" to complete the survey.

3.4 Demographics of Participants

IDEs

32% (16/50) reported to use Eclipse, 24%(12/50) used IntelliJ IDE and 24%(12/50) Microsoft Visual Studio. 6 % indicated to not use any IDE right now, 6% used Pycharm, 1 participant used Jade, 1 participant used Visual Studio Code and 1 participant reported to use an internal IDE. Finally, one participant used monodevelop.

3.5 Warning Rating

Participants rated warnings on a 7-point Likert scale. The resulting rating can be found in Figure 7 for **compiler warnings**, in Figure 8 for **warning marker** warnings, in Figure 9 for **warning view** warnings including general and security warnings, in Figure 10 for **warning view** warnings including **only security warnings**, in Figure 11 for **pop-up warnings**, in Figure 12 for **warnings on committing** and in Figure 13 for **warnings on committing with security ticket**. Figure 14 shows the statements regarding ignoring security warnings.

In the figures 5 and 6 we plotted the preference score ranking of 20 random participants for each type of warning. Each spike in the graph indicates a preference jump within one participant.

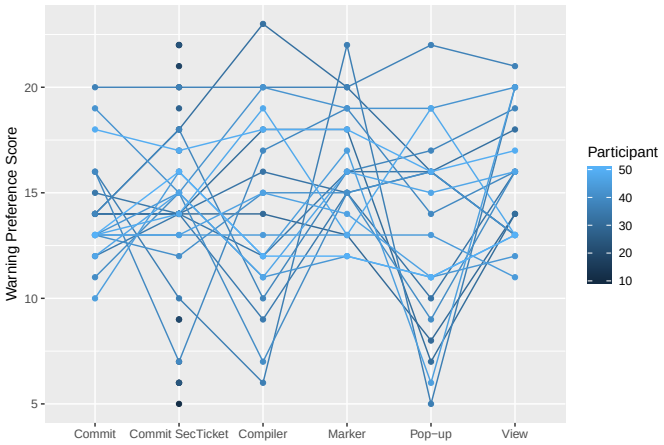

Figure 6: Preference score for 20 random participants [batch 3]

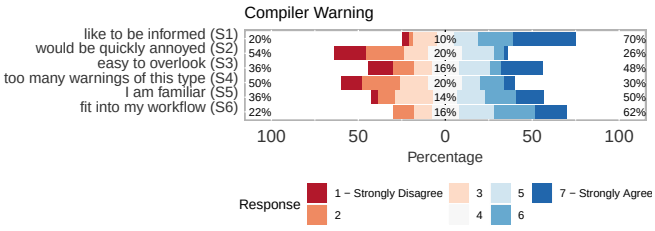

Figure 7: Compiler warning

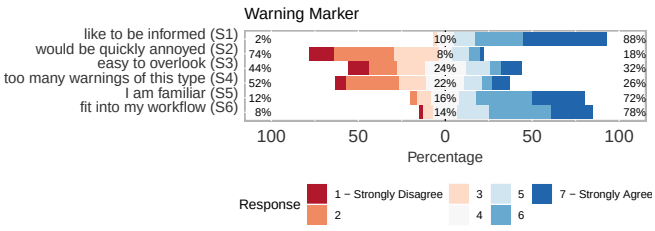

Figure 8: Warning marker

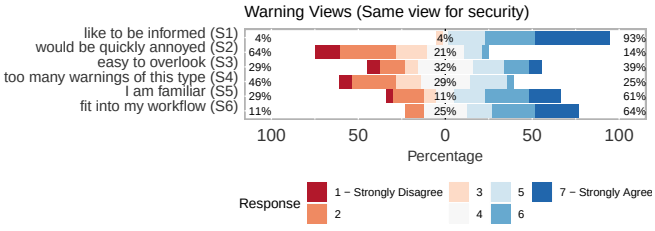

Figure 9: Warning view including security warnings

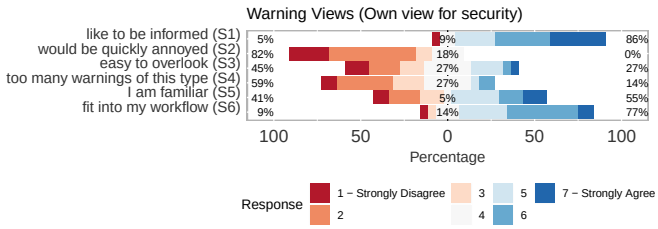

Figure 10: Warning view only for security warnings

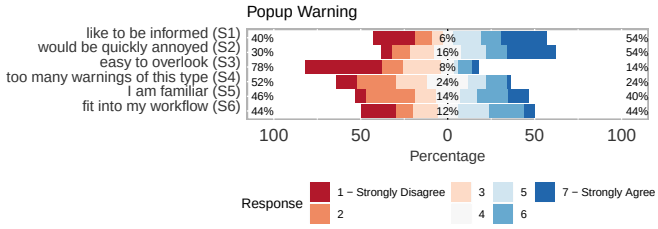

Figure 11: Pop-up warning

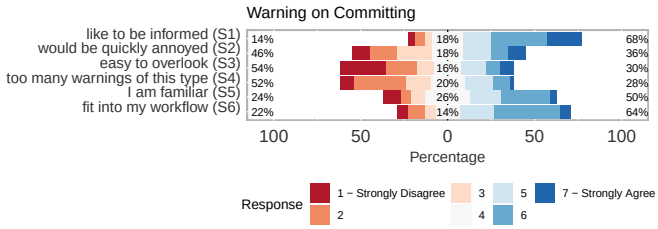

Figure 12: Warning on committing

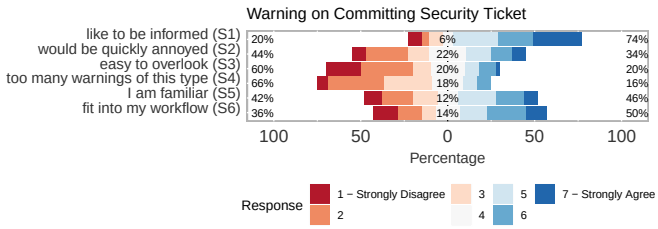

Figure 13: Warning on committing security ticket

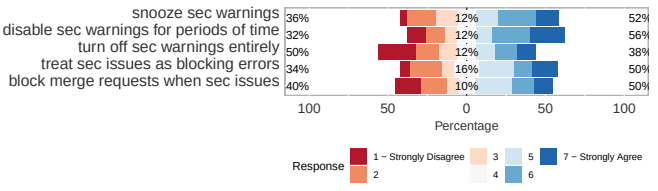

Figure 14: Ignoring security warnings

Figure 15: Compiler Warning
The compiler warning is displayed in the command line output of the IDE.

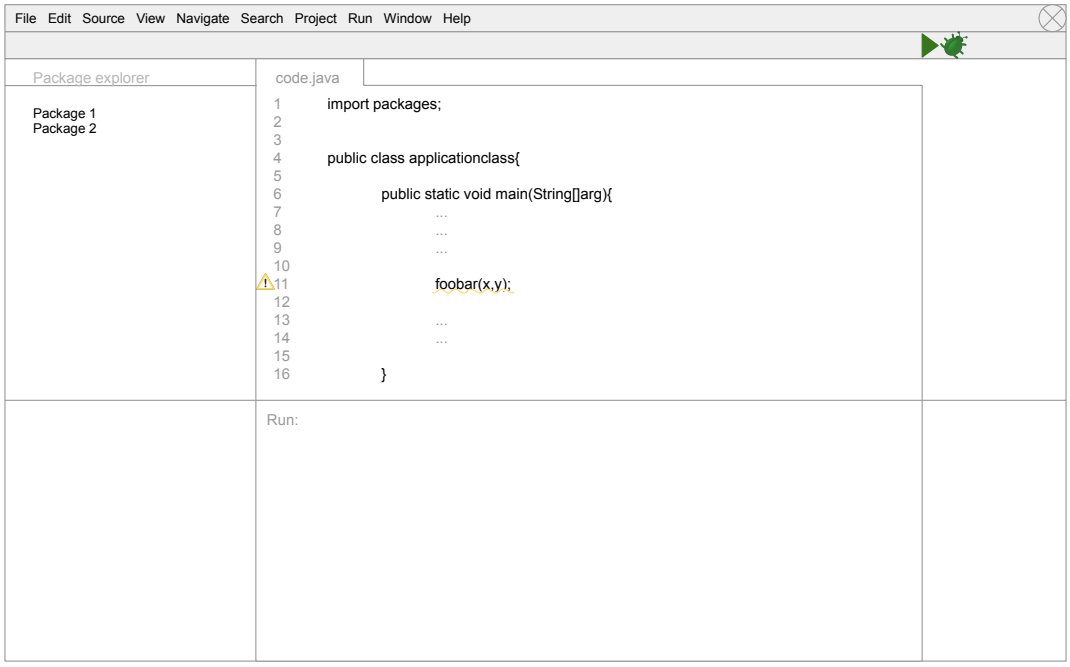

Figure 16: Warning Marker Yellow
The yellow marker is displayed in the code editor marking the exact line.

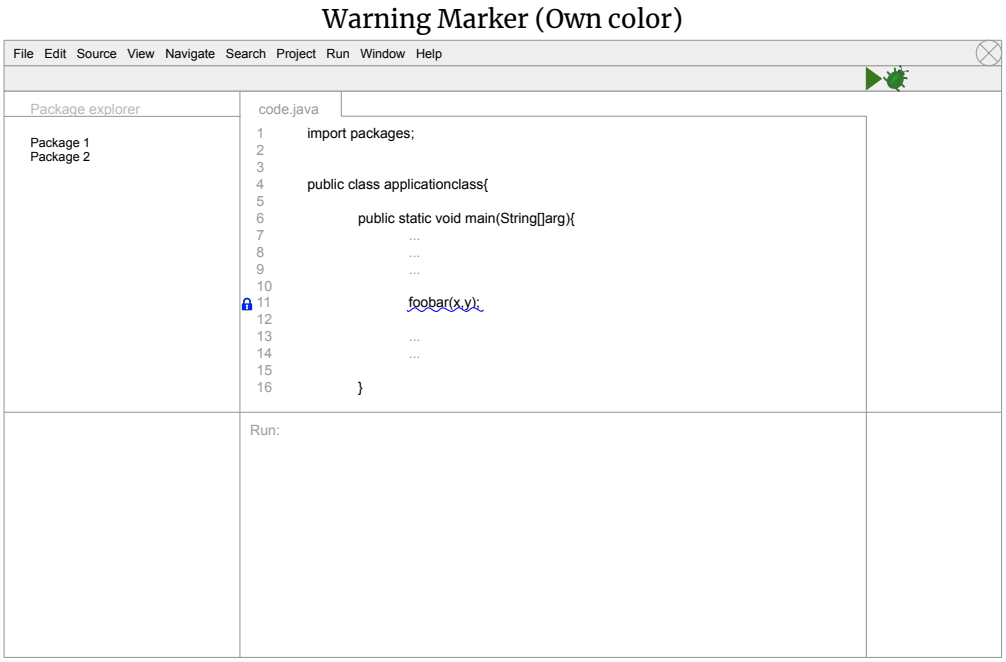

Figure 17: Warning Marker Own Color
The blue marker is displayed in the code editor marking the exact line.

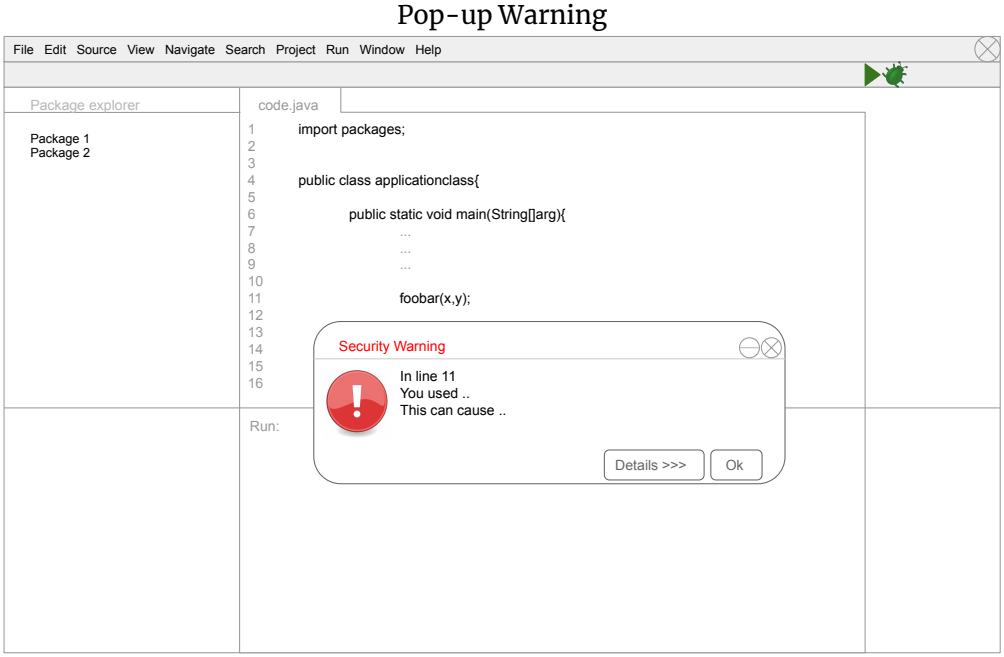

Figure 18: Pop-up Warning
The pop-up overlays the IDE and code editor.

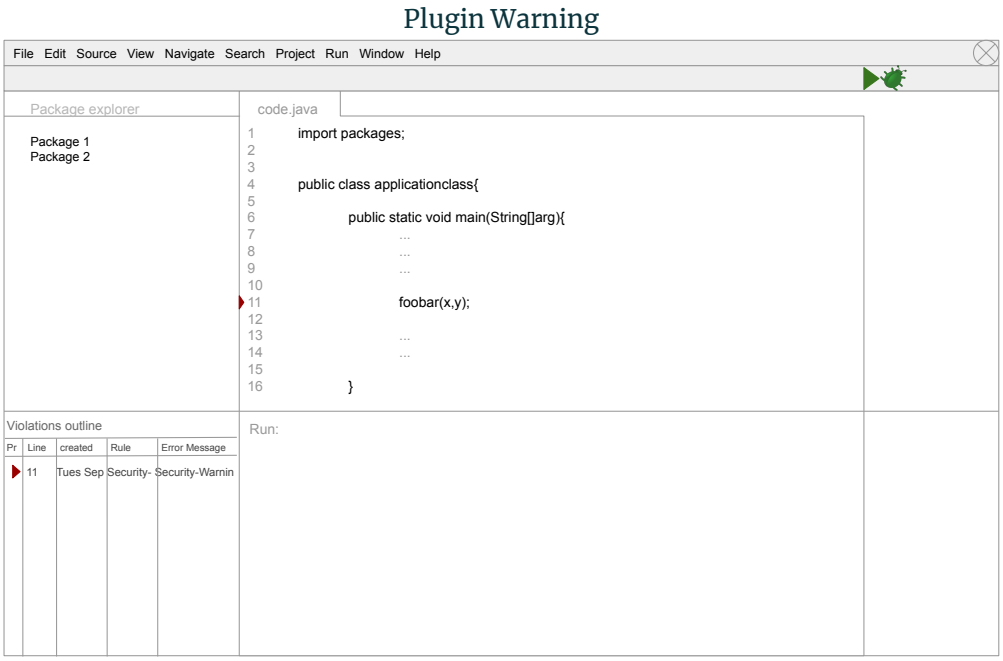

Figure 19: Plugin Warning
The plugin warning is displayed, by default the view can be found in the left corner of the IDE. It usually contains different other warnings than security warnings.

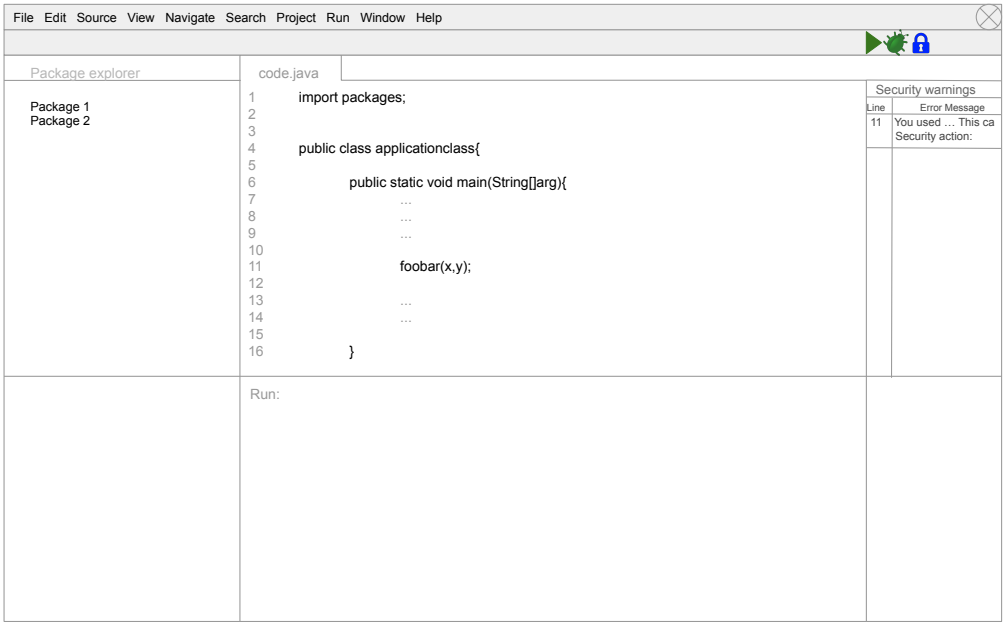

Figure 20: Security View
The security view warning is displayed in a view, by default the view can be found in the right corner of the IDE. Only security warnings would be displayed in the security view.

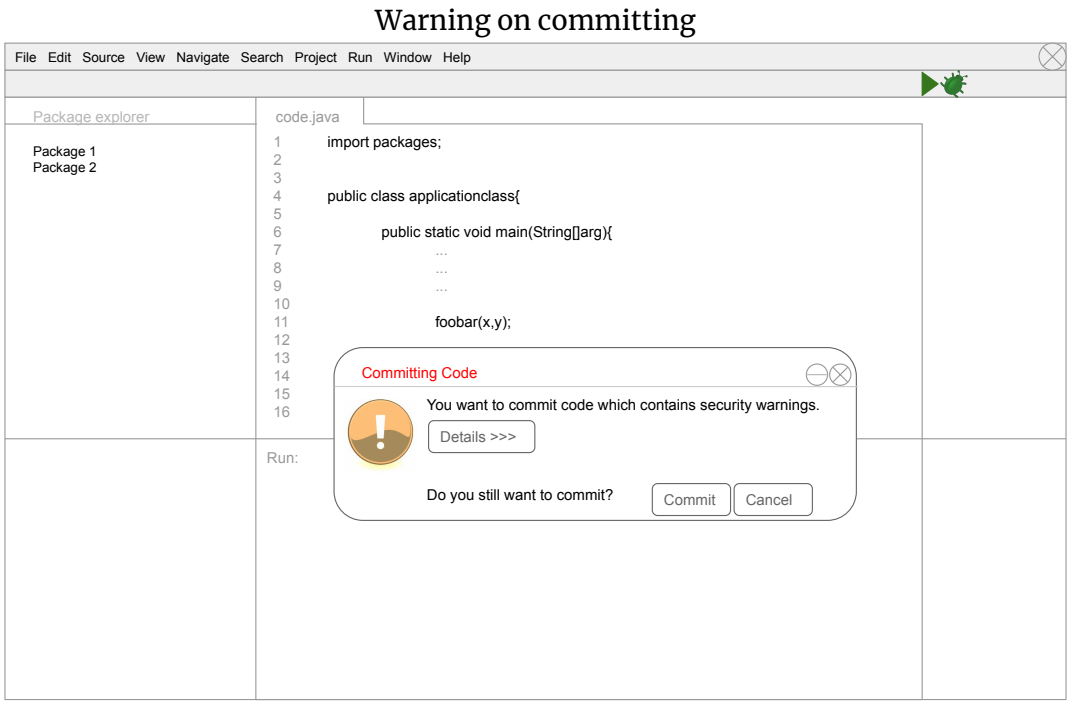

Figure 21: Warning on Committing
The warning is displayed on committing.

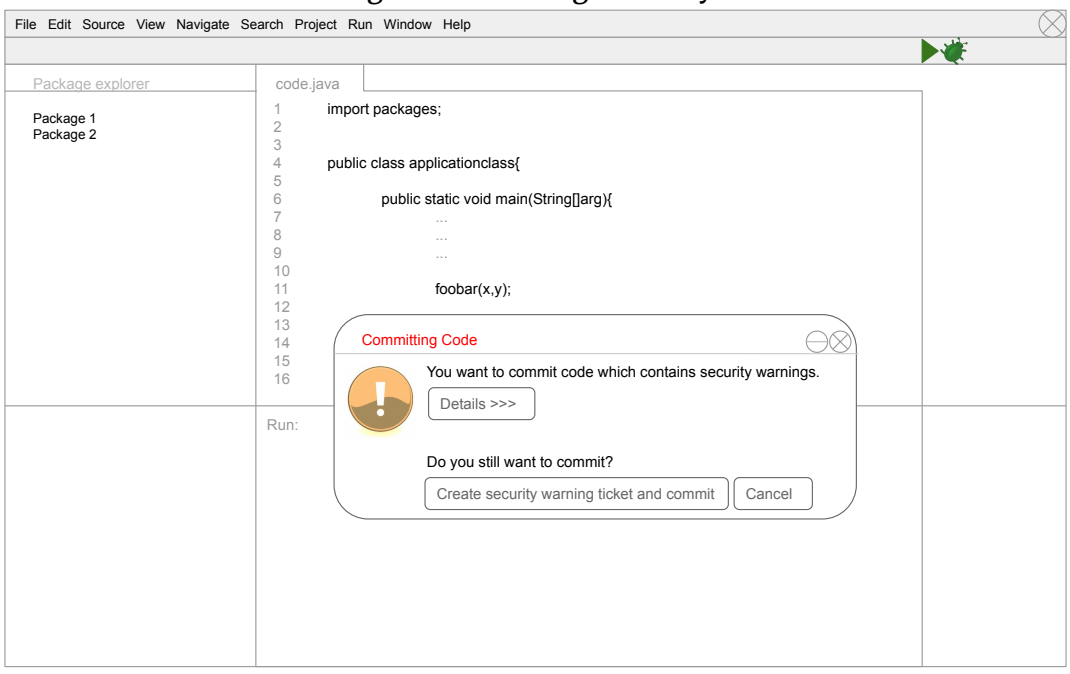

Figure 22: Warning on Committing with Automatic Security Ticket
The warning is displayed on committing. It is only possible to commit when creating a security ticket.

Example	Category	Sub-Category	Codes
	Warning Preferences		
		Security View Warning	
"That's why I like the idea of security view, it's kind of like plug-in, but you can differentiate it, we can see the plug-ins outline somewhere else and the security view warning somewhere else." (D5)			likes the idea of extra view for security
"I didn't like it that much, because I think it's not that visible, visibility purpose like in the corner, because there is lots of other stuff." (S6)			Security View Warning not visible
			Security View Warning: likes error with line number
			very interesting when using many tools
			summary in extra window - would be ok, too
			Security View Warning good overview
			can also be overkill
			bad: requires familiarization
			Security View Warning for severe security mistakes
			Security View Warning for HTTP instead of HTTPS
			Security View Warning for Static IV warning case
			Security View Warning could be combined with Warning Marker
			make sure that window is noticed
			whether there is enough space in this IDE
		Compiler Warning	
			likes Compiler Warnings
			Security Warnings as errors
			Compiler Warnings more familiar
			Compiler Warning good because of visibility
			Compiler Warning as favorite
			misleading
			could be overseen
			developers could also use command line
			dislike: it's more like an error that we get in coding
			Compiler Warnings could be not shown if code is not compiled
			Compiler Warning could be overseen in big projects
			always depends on how much scope I have
			Compiler Warning during coding
			would like to see Compiler Warning every time I compile code
			the good thing about Compiler Warnings is that you can disable them
			Compiler Warning for HTTP(S)
			Compiler Warning for SQL input validation
		Warning on Committing	
			prefers Warning on Committing
			likes Warnings on Committing
			Warning on Committing is a good idea because of visibility
			reminds me when I want to disable this Warning for testing

		but I like to have another stop sign THEN
		Warning on Committing necessary
		somebody already thought about it
		Committing Warning good idea after finished functionality task
		does not fit into my workflow
		always commit my code using the command line
		Warning on Committing for severe security issues
		allowance to commit only after solved security issues
		Warning on Committing every time
		others should see the security issue as well on merge request
		would make more sense when pushing into the master branch
	Security Ticket	
		likes the warning
		project management doesn't want to have unsolved to-dos
		go into some pot into which nobody takes a look
		I reported it, so I don't have to do anything about it anymore!
		good for security process, not for development in general
		would make security administrator happy
		good when I can disable it
		depends on how experienced your co-workers are
	Warning Marker	
		yellow: could be overseen: another color than yellow
		best: prefers Warning Marker in own color
		like icon for security
		I like this lock, this triangle
		open lock
		a special one, for example this icon
		like Warning Marker
		visible
		information directly with source code
		Warning Marker good during programming
		Warning Marker demonstrates security issues well
		Warning Marker useful
		inline is difficult if you have 10.000 lines of code
		own experience with Warning Markers negative
		strange point in time and I cannot believe it anyway
		would ignore Warning Marker in yellow
		Warning Marker meaningless
		Warning Marker during programming instead of Test Mode
		Warning Marker as addition
	Plugin Warning	
		it is unnecessary that it is always there
		likes the idea of having a list
		prefers Plugin Warning
		Plugin Warning reliable
		dislike: we already have lots of plugins in Eclipse
		it is a separate thing and therefore less important
		Plugin Warning not visible
		window for Plugin Warning could be hidden (not recommended)
		with a plugin option you would have so many errors or warnings
		bad: requires familiarization

	Pop-up Warning	
		for me "Are you sure" questions are allowed to appear as a Pop-up
		Pop-up fine if requested for more details
		Pop-ups only in context
		Pop-up on committing
		Pop-up Warning for important security issues
		so if you have a Pop-up, you're forced to check all of them
		just at the end
		Pop-up bad because jump, a link
		not enough space to show error report
		Pop-up warning intrusive particularly when one uses shorthand symbol
		not when saving, not when editing
		with keyboard and you don't want to always have a window open
		Pop-up Warnings intrusive during coding
		blocker
		habituation effect
		but no Pop-up, I will simply close it
		depend on frequencies
		depends on development lifecycle time
		functionality first, Pop-up Warning second
		Pop-up Warning when it occurs and after finished coding
	Own Warning Design	
		have statistics for coding
		possibility to get details to security issue
		where to get additional knowledge to topic
		description and maybe additional help
		for some positions a more detailed description helpful
		forum
		build infrastructure where you can get feedback from more people
		to-do in code, automatically create a view out of it
		profile of preferences, suggest those
		text completion (parameter choices) for security
		warnings should show context issues, not only single lines
		above visual warnings
		or light goes out
		Warning as audio?
		Smart watch with electric shocks
		internet disabled for 10 minutes or so
		likes suggestions for improvements
		Warnings have to be short and concise
		Warnings should be short with possibility for more details
		Warning as drop-down text
		possibility for sorting
	Color for Warnings	
		yellow
		Security warnings deserve an own color
		yellow could be overseen
		blue for Warnings
		red for Warnings
		color similar to red
		orange
		lilac for warnings

		Combination	color for warnings should depend on how important (security) is background color own color plus own symbol Warning sign more significant than color combination of warnings is a good idea Compiler Warning + Warning Marker Compiler + Committing Compiler + Committing + Marker Warning Compiler + Plugin Warning Committing + Pop-up Committing + Warning Marker Committing + Warning Marker (own color for security/symbol) Committing + Warning Marker + Security View committing + Security View Warning Marker + own color + Plugin Warning Marker + Pop-up Warning Warning Marker own color + Security View Warning does not like the idea of combined warnings
	Disabling warnings		
"You have a testing environment for which I am currently implementing new features. In this case I would say that warnings can be clicked away." (D4)			ignore in test cases should be possible
			dismiss until next commit Reminder! after ignoring should be marked Test-Mode should be catched if forgotten Configuration! whether this is useful in the intended purpose administration costs! for security unavoidable ignore it for the run likes the idea of the possibility to ignore warnings for production disabling warning function usefulness depends on situtation Warnings should always be shown warning disable function is not useful for security issues via SVN/Git solvable, e.g. Brunch
	Warning in the context		
"Compiler warnings are not intrusive enough." (D3)			important security issues should be warned intrusively
			better education instead of warnings! current warning system prevents from developing and does not help More than one prefect system Depends on habits, experience and different views Warnings important in java context or wherever in the outer world Warnings should be always up-to-date for you! APIs are only as secure as the developer warnings should be obligatory for security I would force the developer to actually solve the issue

			these warning cases easy to prevent really good, if you can get the warnings for basic steps Asks for static analyzer never seen a security warning in the IDE warnings for secure password storage useful warning appears directly after pulling IntelliJ pretty good for development quality control to include a last obstacle
	Company		
"We do have a security team in our company but they are not familiar with coding. In the end they are only responsible for imposing security instructions on us." (D13)			security team in company
"There is a security officer. But I could not directly approach him, he is over-strained because he needs to make sure that we comply with security policies. There is a security guideline but it is not precisely made for all individual cases. So I cannot look up and see which code I have to use. That is what the developer decides on his own." (D9)			security officer available, but cannot be asked
			superordinated security check impose security guideline are not familiar with coding security responsible person decides about security issues for inspector-tool if somebody is responsible and delegates no security responsible person Code Review I think this should be done for all projects. I am also one of the initiators of this code review of course doubled resources Testing! no demand for security in company, but policy/guideline general policy/guide, no specific guideline for current project security responsible person not relevant for small team
	Team		
"If I am working with a productive team where everybody works fine, then they would look at it. But usually, there are different people in a team. The people working system-oriented would look at it closely. The front-end developer would not likely do that."(D14)			colleagues with skills could check the issues one more time
			Warnings from other people would like to see warnings of other people maintainer responsible for security I reported it, so I don't have to do anything about it anymore! to-dos are often not comprehensible after a while depends on how the team is organized it is nothing which is easy to realize

			depends on the team size relatively small set it is difficult during vacation depends on how frequently colleagues change would fix others warnings if working on same program section usually developer develops alone delegating warnings to colleagues with other context no sense
	Responsibility for Security		
"Of course. Also, we are a government entity. It is very important for us and we get reports and information sources regularly telling us which rules we need to follow. Within the scope of a project me as a project manager has to make sure that the security policies are met." (D6)			feel responsible for security
			if you program something, you are responsible for it personal preference that I additionally get pmd's opinion consider security while programming project management government agencies have security guidelines ok this is not my code could use frameworks which ensure security if there is a mistake in it, I don't feel âĂę of course people developing security guidelines not technique there exists the owasp-guideline and this needs to be fulfilled developer overloaded
	Security		
"Also I have some negative experience because it hinders (laugh). When we program they need to release the code for production or the testing environment, there is a programming running checking for specific criteria. Some criteria is too specific. It does not fit everything in the code and that hinders you a bit." (D12)			security checks prevent from working
			extra warnings for security security warnings for developers important security should be there by default often security is only used, because no extra steps needed relies on tools which deal with security
	Security Breach		
"Personally, I am not affected, but if you work for a company you hear some things. For example if there are some updates we need to make sure the systems are up-to-date and security vulnerabilities are always patched."(D6)			did not experience security breach as developer
			experienced security breach as end-user used honeypots to find vulnerabilites fetched scripts security breaches are in the php context right now firewall system word macro direct access to SAP database security updates ignored certificate security as developer already
	Time		

"I would like to see directly if the IDE thinks there is a problem anywhere." (D2)			Warnings during programming best
			immediate prompt when IDE sees problem
			during and after finished programming
			no time to work to learn new tools
			Depends on time, not on warning type!
			warnings could be too late
			So it's really good, if you come to know before committing
			2-3 times a day at most
			depends on how severe security issues are
			warnings during programming and on committing
			education time security trade off
			because in between.. I think it is disturbing then
			at the end
			warnings need extra time
			you need a lot of time for security
			quality costs time
	Frequency		
"I mean too many warnings are annoying but being able to check it makes sense." (D2)			too many warnings are annoying
			warnings everywhere (habitation)
	Used tools		
"Yes. The standard rules. That is the case. You see a lot of warnings most of them get deactivated because they are not applicable." (D10)			pmd: many warnings, most is deactivated, not applicable
			sonarqube: I thought it was interesting
			sonarqube: feedback via pop-up
			FindBugs
			Virtual Forge
			SAP-Code Inspector
			guideline and then you can derive things
			pmd: if you load everything, they sometimes disagree with each other
			pmd: default settings are not really helpul
			pmd: we would use it afterwards
	Functionality vs. security		
"When the code is not running, I will first try to fix the code before addressing security warnings ... so before saving my work, let me go through all the warnings, not while I'm coding." (D5)			functionality first, security second
"In project I worked on it is not really checked. That is my personal preference that I would like to hear the opinion of pmd." (D10)			on the projects I am working on, it is not really checked
"You need a lot of time for security. But when you have time pressure everything needs to be done fast so then how much more time would you like to invest?" (D7)			time pressure, security needs time
"When you are developing for a bank you internalized it (security)." (D12)			functionality and security equally important

"If there are security relevant findings in the code, they are displayed and if they are critical, the code is not released." (D13)			if critical, code is not released
"Checking code and reviewing code is not a pleasant task. That's how it is. But it is required. Therefore, I would say merge should not work [if there is a warning]. Then you need to really look why it does not work." (D4)			no merge request allowed if security issues are not fixed
"On one hand, you want to bother people at the right time. On the other hand, you don't want to prevent people from working" (A2)			functionality, security trade off
			distracts a lot from what you actually want to code Errors first, warnings second yes that come while conceptualizing security does not matter often for us other things are more important... performance

Table 1: Codebook

REFERENCES

[1] [n.d.]. Notepad++. https://notepad-plus-plus.org/. Accessed: 28-01-19.
[2] 2018. Code::Blocks IDE. www.codeblocks.org/. Accessed: 28-01-19.
[3] 2018. Hello, Colaboratory - Colaboratory - Google. https://colab.research.google.com/. Accessed: 28-01-19.
[4] Kathy Charmaz. 2006. *Constructing grounded theory: A practical guide through qualitative analysis.* sage.
[5] The Spyder Website Contributors. 2018. Spyder Website. https://www.spyder-ide.org/. Accessed: 28-01-19.
[6] Eclipse. 2018. Eclipse. https://www.eclipse.org/. Accessed: 28-01-19.
[7] Geany. 2018. Geany. https://www.geany.org/. Accessed: 28-01-19.
[8] JetBrains. 2018. CLion: A Cross-Platform IDE for C and C++ by JetBrains. https://www.jetbrains.com/clion/. Accessed: 28-01-19.
[9] Jetbrains. 2018. IntelliJ IDEA. https://www.jetbrains.com/idea/. Accessed: 28-01-19.
[10] JetBrains. 2018. PhpStorm: The Lightning-Smart IDE for PHP Programming by JetBrains. https://www.jetbrains.com/phpstorm/. Accessed: 28-01-19.
[11] Jetbrains. 2018. PyCharm: the Python IDE for Professional Developers by JetBrains. https://www.jetbrains.com/pycharm/. Accessed: 28-01-19.
[12] Sublime HQ Pty Ltd. 2018. Sublime Text - A sophisticated text editor for code, markup and prose. https://www.sublimetext.com/. Accessed: 28-01-19.
[13] Microsoft. 2018. Visual Studio IDE, Code Editor, VSTS, & App Center - Visual Studio. https://visualstudio.microsoft.com/. Accessed: 28-01-19.
[14] Netbeans. 2018. Netbeans. https://netbeans.org/. Accessed: 28-01-19.
[15] SAP SE or an SAP affiliate company. 2018. SAP Documentation - ABAP Workbench Tools. https://help.sap.com/saphelp_nw73EhP1/helpdata/en/ef/d94b78ebf811d295b100a0c94260a5/frameset.htm Accessed: 28-01-19.
[16] Android Studio. 2018. Android Studio and SDK tools Android Developers. https://developer.android.com/studio/. Accessed: 28-01-19.