Kevin Dsa

CWID: 20009000

1. During the lecture, I gave several examples where as a development leader I would use "Empirical software engineering" to gain insight into key leadership decisions. For this assignment, detail one use case (not one that I covered in class) where you would use the empirical software engineering process to find an answer to a leadership question. Make sure you include the following:

What is your exact hypothesis?

What are your independent and dependent variables?

What are your control variables?

What are your threats to conclusion validity, construct validity, internal validity, and external validity?

How would you collect your data?

How would you use this data to make a decision?

Answer:

Hypothesis: will Implementing continuous integration in a software development team will result in a decrease in the number of bugs discovered in production ?

Independent Variable: Continuous integration implementation

Dependent Variable: Amount of bugs discovered in production

Control Variables: Number of developers, development environment, software complexity

Threats to conclusion validity: There could be insufficient data to make a reliable decision.

Threats to construct validity: The level of expertise among developers may vary and this can have an impact on the result

Threats to internal validity: Uncontrolled variables that could influence the outcomes, such as the development environment, the complexity of the software, Tests written by developers to identify bugs

Threats to external validity: Results may only apply to service or product based software engineering teams

Data Collection: The data should be collected by keeping track of the quantity of problems found in production both before and after continuous integration was put into place. The number of developers, the development environment, and the complexity of the product should all be included as data points. Most of this data can be generated automatically and can be logged into a file every time the tests are run with and without continuous integration.

Making a Decision:

The data should be analyzed to determine if there is a significant difference i

If the data shows that implementing continuous integration results in a decrease in the number of bugs discovered in production, then the development leader can decide to implement continuous integration in the software development team. If the data shows that implementing continuous integration does not result in a decrease in the number of bugs discovered in production, then the development leader can decide to look for alternative solutions to reduce the number of bugs discovered in production.

https://semaphoreci.com/continuous-integration

https://www.ibm.com/topics/continuous-integration

https://github.blog/2022-02-02-build-ci-cd-pipeline-github-actions-four-steps/

https://www.jenkins.io/

https://www.exoscale.com/syslog/what-is-continuous-integration/

https://www.softwaretestinghelp.com/continuous-integration/

https://www.cloudbees.com/blog/reduce-production-bugs-with-continuous-integration