



The Software Engineering SQEP Pipeline Problem: Challenges and Opportunities

Shamal Faily, James H Sharp
Software & Systems Dependability Team
Cyber and Information Systems Division

- Certification is a barrier to technology exploitation.
- We must certify software at the pace afforded by modern CI/CD infrastructure.
- Vignettes for what *good* looks like assume collaborative and professional software engineers.




UK MOD © Crown copyright 2021

- Public perceptions are changing
 - Protecting privacy matters.
 - Assuring automation matters.
- The UK should be leading safety regulation.
- A workforce that can engineer software may not be sufficient, but it is necessary.

The screenshot shows a CNBC news article. At the top, there is a navigation bar with 'MARKETS BUSINESS INVESTING TECH POLITICS CNBC TV INVESTING CLUB & PRO' and a search icon. The main headline is 'British Prime Minister Rishi Sunak pitches UK as home of A.I. safety regulation as London bids to be next Silicon Valley'. Below the headline, it says 'PUBLISHED MON, JUN 12 2023-1:26 PM EDT | UPDATED MON, JUN 12 2023-2:41 PM EDT'. The author is 'Ryan Browne @RYAN_BROWNE_' with a 'WATCH LIVE' button. A 'KEY POINTS' section follows, containing three bullet points.

British Prime Minister Rishi Sunak pitches UK as home of A.I. safety regulation as London bids to be next Silicon Valley

PUBLISHED MON, JUN 12 2023-1:26 PM EDT | UPDATED MON, JUN 12 2023-2:41 PM EDT

 Ryan Browne
@RYAN_BROWNE_ WATCH LIVE

KEY POINTS

- British Prime Minister Rishi Sunak on Monday pitched the U.K. as a global center for AI safety, seeking to position the country as a leader in regulation of the technology as countries around the world race to secure a leading role.
- “I want to make the U.K. not just the intellectual home but the geographical home of global AI safety regulation,” Sunak said, addressing a packed tech conference in London Monday.
- Sunak last week announced the first global AI safety summit in the U.K. later this year.

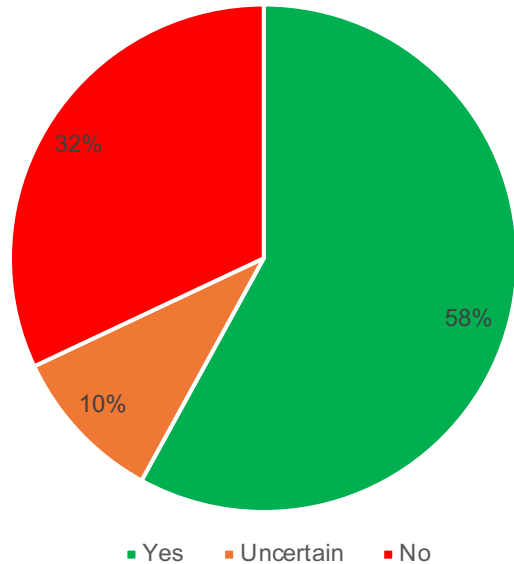
<https://www.cnn.com/2023/06/12/pm-rishi-sunak-pitches-uk-as-geographical-home-of-ai-regulation.html>

- Material reviewed from 50 full-time UK undergraduate programmes.
- Non-specialised Computer Science degrees only.
- All Russell Group HEIs included.
- Several HEIs delivering NCSC accredited Cyber Security degrees included.
- Not all programmes were BCS accredited.

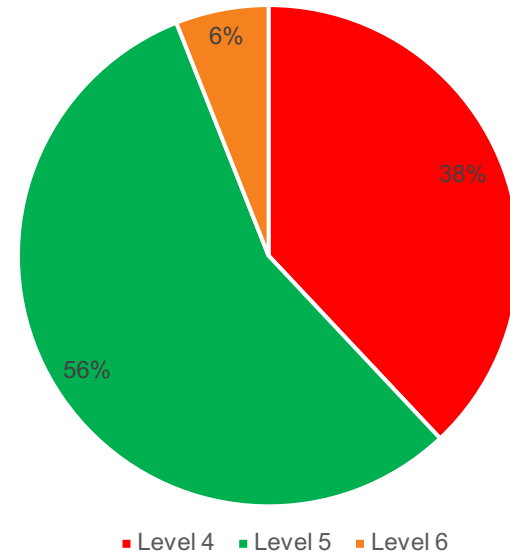


Photo by Changbok Ko on Unsplash

- Classic but relevant material on Software Engineering de-emphasised from Computing programmes.

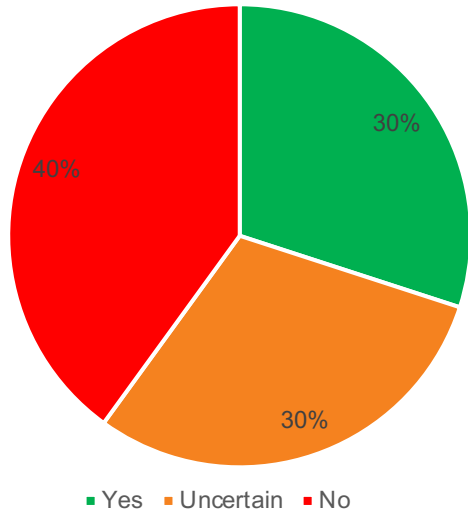


Presence of a Software Engineering module

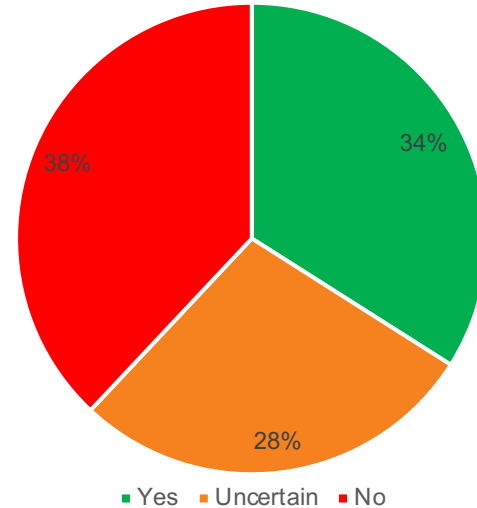


Point of Software Engineering content delivery

- BCS accreditation emphasises presence of embedded systems with safety & security implications.
- Their design should entail applying engineering principles.

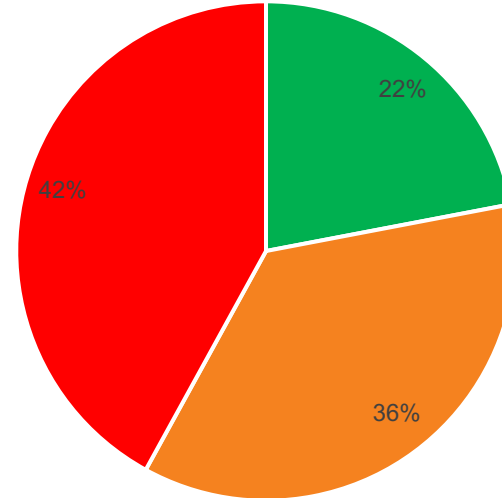


Requirements content



Compiled languages content

- Erosion of the SQEP available to deliver best practice in Software Engineering research & practice has implications.
 - What cogent material can be delivered?
 - What is Software Engineering understood to be?
 - What are the unintended consequences of teaching Software Engineering to students via group work?



■ Specialist ■ Non-specialist ■ Unknown

Level of Teaching SQEP

- Acknowledge tacit knowledge on Software Engineering education is being lost.
- Are we facing a Software Crisis 2.0?
- Is there a case for a new model 'Software Engineering' undergraduate degree?



<http://homepages.cs.ncl.ac.uk/brian.randell/NATO/>

- Degree accreditors must call out anti-patterns in Software Engineering education.
 - Software development & design <> Software Engineering
 - Teaching Software Engineering early in a group project without reinforcing theory in later years.
 - ... etc

BSiSEE

CSEE&T 2023 Workshop CFP: Bad Smells in Software Engineering Education

Aug 7, 2023

Tokyo, Japan

Contribution submissions deadline: June, 7 2023

Theme & Goals

There are a number of common bad practices that software engineering students innocently fall into. Software engineering educators need to detect ("smell") these bad practices early on and provide students with guidance on better practices before they become ingrained habits that are difficult to change. Some examples of bad practice smells include copying and pasting code without understanding it, employing few or poorly written comments, not following coding standards and formatting guidelines, over-complicating solutions, writing code with no clear purpose, jumping into coding before thinking about the design, writing large blocks of code without testing, and not seeking feedback or assistance when stuck.

Data from surveys and interviews with students indicate that bad practices are common and have real negative consequences. Bad practices can hinder the learning process. And current software engineering curricula and teaching methods may not adequately address them, or may not address them early enough, leaving students to develop bad habits that linger into their later careers.

Enabling instructors to "smell" bad practices and proactively address them can help them to help students understand why these practices do harm in the long run. Thus instructors can provide guidance and reinforcement for better practices within software engineering coursework which in turn can help to reduce or avoid the development of these bad habits.

<https://dport96.github.io/projects/BSiSEE>

- Embody material on software certification into the curriculum for all Computer Science students.
 - Gives instructors and students something of tangible value besides code.
 - Encourages innovation around how to do this at the same pace as modern software engineering.
 - Opens avenues for sharing of Teaching SQEP.



<https://agilemanifesto.org/>

[dstl] The Science Inside

Discover more

