

17-313

Foundations of Software Engineering

Spring 2023

Introductions



Michael Hilton



B.S. San Diego State University - 2002

Software Engineer at DoD - 2002 to 2011



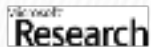
M.S. Cal Poly San Luis Obispo - 2013



PhD at Oregon State - 2017



Internship at Microsoft Research - Summer 2017



Assistant Teaching Professor at CMU - Fall 2017



Associate Teaching Professor at CMU - Fall 2020



Hyrum Wright



B.S. Brigham Young University - 2006



M.S. Univ. of Texas at Austin - 2010



PhD at Univ. of Texas - 2012



Apache Subversion - 2006 - 2012

WanDisco - 2010 - 2012



Duolingo - 2017 - 2018



Google - 2012 - ?



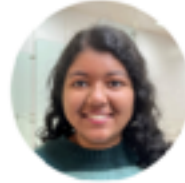
Visiting Lecture, CMU - 2015 - ?



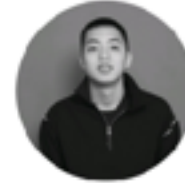
Teaching Assistants



Anuda Weerasinghe



Mahima Pannala



Tak-Ho Lee



Victor Andres Alfonzo



Antara Pal



**Software is
everywhere**



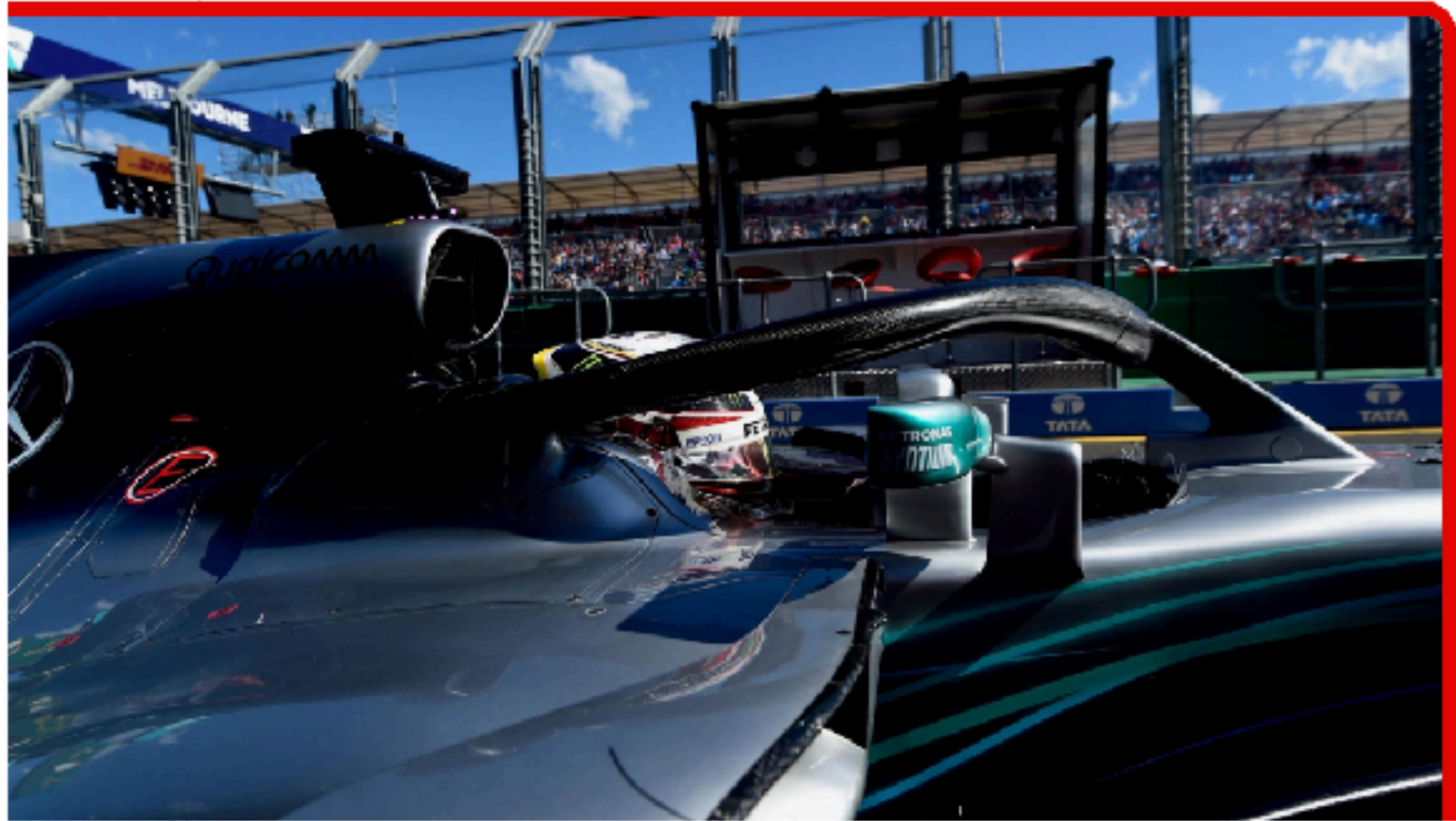
Software glitch cost Hamilton victory - Mercedes

25 March 2018

MERCEDES

AUSTRALIA

HAMILTON



Toyota Case: Single Bit Flip That Killed

Junko Yoshida

10/25/2013 03:35 PM EDT

During the trial, embedded systems experts who reviewed Toyota's electronic throttle source code testified that they found Toyota's source code defective, and that it contains bugs -- including bugs that can cause unintended acceleration.

"We did a few things that NASA apparently did not have time to do," Barr said. For one thing, by looking within the real-time operating system, the experts identified "unprotected critical variables." They obtained and reviewed the source code for the "sub-CPU," and they "uncovered gaps and defects in the throttle fail safes."

The experts demonstrated that "the defects we found were linked to unintended acceleration through vehicle testing," Barr said. "We also obtained and reviewed the source code for the black box and found that it can record false information about the driver's actions in the final seconds before a crash."

Stack overflow and software bugs led to memory corruption, he said. And it turns out that the crux of the issue was these memory corruptions, which acted "like ricocheting bullets."

Barr also said more than half the dozens of tasks' deaths studied by the experts in their experiments "were not detected by any fail safe."

Bookout Trial Reporting

http://www.eetimes.com/document.asp?doc_id=1319903&page_number=1
(excerpts)

**"Task X death
in combination
with other task
deaths"**

HealthCare.gov Diagnosis: The Government Broke Every Rule Of Project Management



Loren Thompson Senior Contributor

Aerospace & Defense

I write about national security, especially its business dimensions.

f After 400 software fixes and major hardware upgrades, the Obama Administration is claiming to have achieved its goal of transforming HealthCare.gov into a web-site that will operate smoothly for "the vast majority of users." That's important, because the site is central to implementation of the most



The Patient Protection and Affordable Care Act, better known as Obamacare, will probably be remembered as President Obama's most important domestic policy initiative. However, inept federal management of the HealthCare.gov web-site that is central to implementing Obamacare has left many users with a negative first impression of the program. (Image credit: AFP/Getty Images via @daylife)



implementation of the most

daylife)

THE VERGE

Figure 1.

121
121
135
92
121
92

BOEING

Probabilistic Compliance Gra



REDLINE

The many human errors that brought down the Boeing 737 Max

Catastrophic

(12.35 M)

Accident

Vasa



Vasa



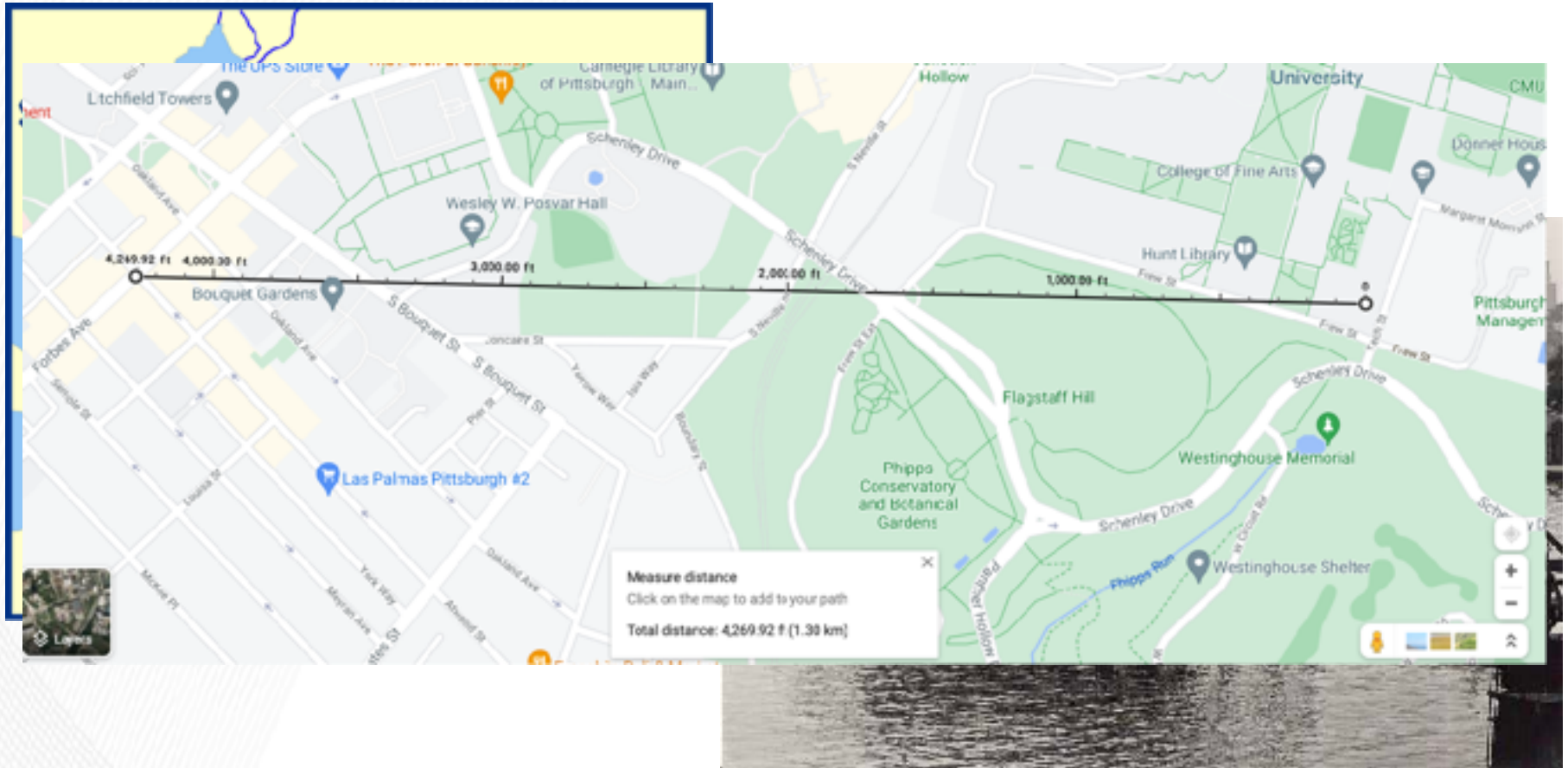
Vasa



Vasa



Vasa



What happened is now called “Vasa syndrome”

What happened is now called “Vasa syndrome”

- Changing shipbuilding orders
- No specifications for modified keel

What happened is now called “Vasa syndrome”

- Changing shipbuilding orders
- No specifications for modified keel
- Shifting armaments requirements

What happened is now called “Vasa syndrome”

- Changing shipbuilding orders
- No specifications for modified keel
- Shifting armaments requirements
- Shipwright’s death

What happened is now called “Vasa syndrome”

- Changing shipbuilding orders
- No specifications for modified keel
- Shifting armaments requirements
- Shipwright’s death
- No way to calculate stability, stiffness, or sailing characteristics

What happened is now called “Vasa syndrome”

- Changing shipbuilding orders
- No specifications for modified keel
- Shifting armaments requirements
- Shipwright’s death
- No way to calculate stability, stiffness, or sailing characteristics
- Failed pre-launch stability tests

What happened is now called “Vasa syndrome”

- Changing shipbuilding orders
- No specifications for modified keel
- Shifting armaments requirements
- Shipwright’s death
- No way to calculate stability, stiffness, or sailing characteristics
- Failed pre-launch stability tests

REQUIREMENTS

What happened is now called “Vasa syndrome”

- Changing shipbuilding orders
- No specifications for modified keel
- Shifting armaments requirements

REQUIREMENTS

- Shipwright's death

TEAMS

- No way to calculate stability, stiffness, or sailing characteristics
- Failed pre-launch stability tests

What happened is now called “Vasa syndrome”

- Changing shipbuilding orders
- No specifications for modified keel
- Shifting armaments requirements

REQUIREMENTS

- Shipwright’s death

TEAMS

- No way to calculate stability, stiffness, or sailing characteristics

METRICS

- Failed pre-launch stability tests

What happened is now called “Vasa syndrome”

- Changing shipbuilding orders
- No specifications for modified keel
- Shifting armaments requirements

REQUIREMENTS

- Shipwright's death

TEAMS

- No way to calculate stability, stiffness, or sailing characteristics

METRICS

- Failed pre-launch stability tests

QA

Software *Engineering*?

What is **engineering**? And how is it different from **hacking/programming**?



1968 NATO Conference on Software Engineering

- Provocative Title
- Call for Action
- “Software crisis”





Margaret Hamilton



Margaret Hamilton





This Course

“...participants who multitasked on a laptop during a lecture scored lower on a test compared to those who did not multitask, and participants who were in direct view of a multitasking peer scored lower on a test compared to those who were not. The results demonstrate that *multitasking on a laptop poses a significant distraction to both users and fellow students and can be detrimental to comprehension of lecture content.*”

Computers & Education 62 (2013) 24–31

Contents lists available at SciVerse ScienceDirect

 **Computers & Education** 

journal homepage: www.elsevier.com/locate/compedu

Laptop multitasking hinders classroom learning for both users and nearby peers

Faria Sana^a, Tina Weston^{b,c}, Nicholas J. Cepeda^{b,c,*}

^aMcMaster University, Department of Psychology, Neuroscience, & Behaviour, 1280 Main Street West, Hamilton, ON L8S 4K1, Canada
^bYork University, Department of Psychology, 4700 Keele Street, Toronto, ON M3J 1P3, Canada
^cYork University, LaMarsh Centre for Child and Youth Research, 4700 Keele Street, Toronto, ON M3J 1P3, Canada

ARTICLE INFO **ABSTRACT**

Article history:

Content not available in ScienceDirect. In Risk of copyright infringement, please do not

Smoking Section

- Last full row



Course infrastructure and logistics

Infrastructure/source of truth

Course website: schedule, slides, syllabus, office hours

Canvas (and Gradescope) homework, grades, other material

Slack for communication and collaboration.

Git/Github for coding and collaboration

Logistics:

Lecture in-person only

All recitations are in-person

Office Hours are over flexible.

If you want to talk to us, DM/email ALL INSTRUCTORS at once. Trust me, it's faster.



Connect with us for the class

- All links on our course website: <https://cmu-313.github.io>
- Canvas: <https://canvas.cmu.edu/courses/33307>
- We will send you an invite for slack, please be on the lookout for it.

Hello
my name is

NAME

**HAVE YOU HAD AN INTERNSHIP, IF SO
WHERE?**

Course Themes

- Software engineering as a human process
- Process
- Requirements
- Measurement
- Quality, incl. Security
- Time and team management
- Ethics
- Software Engineering for AI/ML
- Strategic thinking about software

Prerequisites

- Assumes working knowledge of popular programming language(s)
- You will have the best experience if you have had an internship (ask us if you have any questions)
- vs 17-214
 - 17-313 largely focused on human issues and quality beyond functional correctness
 - 17-313 focused on larger scale

Readings and Quizzes

- Reading assignments for most lectures
 - Preparing in-class discussions
 - Background material, case descriptions, possibly also podcast, video, Wikipedia
 - Complement with research
- Few short and easy online quizzes on readings, count as participation
 - Goal is to prep for lecture.

Textbook

- No single textbook
- Assigned readings from different sources
 - Book chapters (library)
 - News articles
 - Lecture notes
- Recommended supplementary reading: Software Engineering at Google
 - Available for free online (legally): <https://abseil.io/resources/swe-book>

O'REILLY™

Software Engineering at Google

Lessons Learned
from Programming
Over Time



Curated by Titus Winters,
Tom Manshreck & Hyrum Wright



Gaining Experience: Central to 313!

- Case study analyses
- Team assignments
- Open source engagement

- No “survivor”-style projects –
wait till 17-413 (Capstone)

Evaluation

- Assignments (60 %)
 - Regular homework, mostly in teams with individual component
 - Open source engagement
- Midterm (20 %)
- Participation activities (20 %)
 - In-class exercises
 - Pre-class reading assignments
 - Recitation exercises

Recitations

- Practical tasks, preparation for homework, extra material, discussions
- Have your github account at the ready.
 - Bring your laptop!
- This week: GitHub (helpful for recitation 1)
- Teams will all go to the same recitations

Assignments

- Setup and test an existing software product
- Get up to speed with the technologies
- Come together as a team and decide on metrics
- Develop a design doc, and implement a machine learning microservice
- Develop a plan for evaluating the quality of the software
- Contribute to an open source project of your choice

Team Assignments

- Mirror realistic setting
- Assigned teams throughout the semester
 - Fill in team building survey before next lecture
- Teamwork surveys every week
- Conflict resolution process as needed
- Most team assignments have individual components

Professionalism

- Being a professional means you should work well with others
- The best professionals are those who make those around them better
- If you feel someone is not treating you or someone else in a professional manner, you have two options:
 - If you feel you have the standing to do so, speak up!
 - Reach out to the course staff, and we will meet with you privately to discuss it, as well as preserve your anonymity

Late day policy

- No late days
 - (simply doesn't work with team assignments)
- Accommodations in case of health issues, travel for interviews, ... on case by case base
 - Inform us at least 2 days before deadline

Academic Honesty

- Standard Collaboration Policy
- In group work, be honest about contribution of group members; do not cover for others
- HW1 will be done in 1 public repo. PLEASE reach out if you have concerns.

For next class: survey, scheduling



Two “quizzes” for Thursday: survey, scheduling

- Forming groups based on schedule availability.
 - This is ridiculously important.
- Shaping the courses based on
 - your background knowledge
 - your interests
- Identifying experience/interest.