Software Engineering Education in the DevOps Era

PANOS FITSILIS
Discussion points

- Challenges in DevOps education
- Needs for DevOps education
- Universities and VETs
- Cultural Change and Mindset
- Call for Action
Challenges in DevOps education

- Lack of Awareness
- Traditional Teaching Methods
- Outdated Curricula
- Limited Practical Experience
- Resistance to Change
- Resource Constraints
- Industry-Academia Gap
The gap between industry demands and academic curricula.

- Rapidly Evolving Technology Landscape: The field of DevOps is characterized by rapid advancements in tools, technologies, and best practices. However, academic curricula often have a longer development and approval cycle, making it difficult to keep pace with the ever-changing industry requirements. This time lag can result in outdated or insufficient coverage of DevOps topics in academic programs.

- Lack of Standardization: DevOps is a broad and multifaceted discipline that encompasses various tools, methodologies, and cultural aspects. Due to the lack of standardized frameworks or universally accepted practices, it becomes challenging for academic institutions to define a cohesive and comprehensive curriculum that aligns with the diverse needs of different industries and organizations.

- Industry-Specific Requirements: Different industries and organizations have their unique requirements and preferences when it comes to implementing DevOps. Academic curricula, however, often take a generalized approach to education, lacking the flexibility to address industry-specific challenges. This mismatch can make it difficult for students to apply their theoretical knowledge in practical scenarios.

- Limited Interaction with Industry Professionals: Academic institutions may have limited opportunities for students to directly interact with industry professionals or gain exposure to real-world DevOps projects. This lack of industry engagement can result in a disconnect between academic theories and the practical realities of DevOps implementation in professional settings.
The rapidly evolving DevOps technology stack

- Frequent Tool Updates
- Shifting Technology Landscape
- Limited Resources for Infrastructure (for universities)
The rapidly evolving DevOps technology stack
Lack of practical hands-on experiences and real-world scenarios

- Theoretical Emphasis
- Limited Access to Tools and Infrastructure
- Simulated Environments
- Limited Industry Exposure
Key Needs for DevOps education

- Updated Curricula
- Practical Hands-On Experience
- Industry Partnerships
- Qualified Faculty
- Interdisciplinary Collaboration
- Continuous Learning and Adaptability
Update curriculum / The body of knowledge

Culture and Collaboration
Agile Development Frameworks
Architecting for Devops
Infrastructure and Automation
Configuration Management
IT Operations
IT Security
Organizational Transformation
Software Delivery Automation
Software Economics
Software Testing
# Update curriculum / ACM curriculum

<table>
<thead>
<tr>
<th>Hours</th>
<th>Discipline</th>
</tr>
</thead>
<tbody>
<tr>
<td>24</td>
<td>Agile Development Frameworks</td>
</tr>
<tr>
<td>51</td>
<td>Infrastructure and Automation</td>
</tr>
<tr>
<td>6</td>
<td>Configuration Management</td>
</tr>
<tr>
<td>0</td>
<td>IT Operations</td>
</tr>
<tr>
<td>20</td>
<td>IT Security</td>
</tr>
<tr>
<td>23</td>
<td>Organizational Transformation</td>
</tr>
<tr>
<td>0</td>
<td>Software Delivery Automation</td>
</tr>
<tr>
<td>8</td>
<td>Software Economics</td>
</tr>
<tr>
<td>47</td>
<td>Software Testing</td>
</tr>
</tbody>
</table>

ACM 2020 Curricula Improves but Not applied yet

ACM 2014 Curricula Recommendations for Software Engineering
Update curriculum / Modules that usually are missing

<table>
<thead>
<tr>
<th>Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction to DevOps:</td>
<td></td>
</tr>
<tr>
<td>Continuous Integration and Continuous Delivery (CI/CD):</td>
<td></td>
</tr>
<tr>
<td>Introduction to CI/CD concepts and workflows</td>
<td></td>
</tr>
<tr>
<td>Infrastructure as Code (IaC):</td>
<td></td>
</tr>
<tr>
<td>Configuration Management and Version Control:</td>
<td></td>
</tr>
<tr>
<td>Monitoring, Logging, and Observability:</td>
<td></td>
</tr>
<tr>
<td>Security in DevOps:</td>
<td></td>
</tr>
<tr>
<td>Collaboration and Communication Tools:</td>
<td></td>
</tr>
<tr>
<td>Cloud Computing and Serverless Architectures:</td>
<td></td>
</tr>
<tr>
<td>DevOps Culture and Soft Skills:</td>
<td></td>
</tr>
</tbody>
</table>
Update curriculum / e-CF standard - EN16234-1:2019 e-Competence Framework

<table>
<thead>
<tr>
<th>Dimension 1: e-CF area</th>
<th>Dimension 2: e-Competence: Title + generic description</th>
<th>Dimension 3: e-Competence proficiency level e-1 to e-5</th>
<th>Dimension 4: Knowledge and skills examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>A. PLAN</td>
<td>A.2. Service Level Management</td>
<td>Standard</td>
<td>Examples for inspiring</td>
</tr>
</tbody>
</table>
|                        | Defines, validates and makes applicable service level agreements (SLAs) and underpinning contracts tailored to services offered. Negotiates service performance levels taking into account the needs and capacity of stakeholders and business. | K1 SLA documentation  
K2 how to compare and interpret management data  
K3 the elements forming the metrics of service level agreements  
K4 how service delivery infrastructures work  
K5 impact of service level non-compliance on business performance | S1 analyse service provision records  
S2 evaluate service provision against SLA  
S3 negotiate realistic service level targets  
S4 use relevant quality management techniques  
S5 anticipate and mitigate against potential service disruptions |
|                        | Level 1 | Level 2 | Level 3 | Level 4 | Level 5 |
|                        | – | – | Ensures the content of the SLA. | – | – |
## Update curriculum / e-CF Profile for DevOps

https://ecfexplorer.itprofessionalism.org/

<table>
<thead>
<tr>
<th>Profile title</th>
<th>DEVOPS EXPERT ROLE</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Summary statement</td>
<td>Implements processes and tools to successfully deploy DevOps techniques across the entire solution development lifecycle.</td>
<td></td>
</tr>
<tr>
<td>Mission</td>
<td>Applies a cross-functional, collaborative approach for the creation of customer-centric software solutions. Introduces automation throughout the software production system to deliver better software faster.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Deliverables</th>
<th>Accountable</th>
<th>Responsible</th>
<th>Contributor</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• Integrated Solution</td>
<td>• Development Process</td>
<td>• Test Procedure • Release</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Main task/s</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Implement and manage continuous distribution methodologies</td>
<td>Design systems with high levels of availability and scalability</td>
<td>Manage testing across the release lifecycle</td>
</tr>
<tr>
<td></td>
<td>Facilitate cross functional collaboration and engagement</td>
<td>Design and manage process automation tools</td>
<td>Adopt an agile software development methodology</td>
</tr>
<tr>
<td></td>
<td>Manage continuous integration tool management</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
DevOps Agile Skills Association (DASA) Principles

https://www.devopsagileskills.org/

01 Customer-Centric Action
  Courage to act, innovate.

02 Create with the End in Mind
  Product & Service thinking,
  Engineering mindset,
  Collaborate.

03 End-To-End Responsibility
  Live your accountability,
  Concept to Grave, performance support.

04 Cross-Functional Autonomous Teams
  T-shaped profiles,
  Complementary skills.

05 Continuous Improvement
  If it hurts do it more often,
  Experiment, fail fast.

06 Automate Everything You Can
  Enhance quality, maximize flow.
The DASA DEVOPS Competence Model
DevOps project

An example
DevOps Competences groups

- Transversal (16)
- General IT Knowledge (9)
- Smart city related (9)
- DevOps related (6)
DevOps competences

- Introduction to DevOps
- Repository management
- Continuous integration
- Configuration management
- Deployment tools
- Using Build, Deployment and monitoring tools
- Code Analysis and Continuous testing tools
Available Certification

**Associations’ developed certification schemata**
- DASA foundation level, professional level, leadership level
- DevOps Institute Foundation
- Etc.

**Vendor specific certifications usually related with platform providers**
- Microsoft DevOps Engineer Expert
- AWS Cloud & DevOps certification
- Etc.

**Tool specific certifications**
- Certified Kubernetes Administrator (CKA), Security Specialist (CKS), etc.
- Docker Certified Associate (DCA)
- Puppet Certified Professional
- Certified Jenkins Engineer (CJE)
- etc
Addressing the challenges

- **Focus on Core Concepts:** Rather than solely focusing on specific tools, prioritize teaching the core concepts and principles of DevOps.
  - Emphasize the underlying philosophies, such as collaboration, automation, and continuous improvement. By understanding the foundational principles, students can adapt to different tools and technologies as they evolve.

- **Teach Transferable Skills:** Instead of focusing solely on teaching tool-specific skills, emphasize transferable skills that are applicable across various DevOps tools and technologies. These include skills such as scripting, infrastructure-as-code, version control, continuous integration, and deployment pipelines. Proficiency in these skills will enable students to adapt to new tools and technologies as they emerge.

- **Emphasize Learning Mindset:** Instill in students a learning mindset and the ability to self-learn. Encourage them to explore and experiment with new tools and technologies on their own. Provide them with resources such as online tutorials, documentation, and communities where they can stay updated on the latest developments in DevOps.

- **Foster Industry Collaboration:** Forge partnerships with industry professionals, organizations, or local DevOps communities. Invite industry experts as guest speakers or adjunct faculty members who can provide insights into the latest tools, technologies, and trends. Collaborate with industry partners to offer workshops, hackathons, or internships that expose students to real-world DevOps scenarios.

- **Agile Curriculum Review:** Adopt an agile approach to curriculum design and review. Regularly assess the relevance of the included tools and technologies. Consider establishing feedback loops with alumni and industry professionals to gather insights on the tools and technologies they find most valuable in their work.
Call to action

- In today's "software enabled society" we need to invest in people and win the war of talent.
  - Action 1: Continuous monitoring of labour market needs
  - Action 2: Continuous evaluation and standardisation of the knowledge set for ICT professionals
  - Action 3: Continuous promotion of modern professional profiles
  - Action 4: Development of a certification schemata for new professional profiles
  - Action 5: Promote new professional profiles.
Νέο ελληνικό βιβλίο

Free download

http://hdl.handle.net/11419/9306
https://repository.kallipos.gr/
Thank you!

FITSILIS@UTH.GR
Cultural Change and Mindset

- A cultural change required to embrace DevOps principles in education.
- A shift is needed from siloed departments to a culture of collaboration and shared responsibility.
- Continuous learning mindset among students and faculty.
- Develop an agile mindset
- Address the need for adaptability, resilience, and innovation in the face of technological advancements.