## REVISION HISTORY

<table>
<thead>
<tr>
<th>VERSION</th>
<th>DATE</th>
<th>REMARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>11-Jul-20</td>
<td>Initial Version</td>
</tr>
<tr>
<td>1.1</td>
<td>29-Jul-20</td>
<td>Consistent title. Removed topics to fit the course in 2 days</td>
</tr>
<tr>
<td>1.2</td>
<td>15-Oct-20</td>
<td>Removed Complex topics and reduced to 7 modules</td>
</tr>
<tr>
<td>1.3</td>
<td>05-Feb-21</td>
<td>Renamed module 4</td>
</tr>
<tr>
<td>1.4</td>
<td>30-Sep-21</td>
<td>Rebranded</td>
</tr>
</tbody>
</table>
Contents

REVISION HISTORY ........................................................................................................... 1
INTRODUCTION TO SYLLABUS ....................................................................................... 3
OVERVIEW .......................................................................................................................... 3
EXAMINABLE LEARNING OBJECTIVES ......................................................................... 3
EXAMINATION DETAILS ................................................................................................. 3
MODULE 1: SOFTWARE DELIVERY CHALLENGES ......................................................... 4
MODULE 2: DRIVERS AND PRINCIPLES OF DEVOPS .............................................. 4
MODULE 3: GENERATIVE CULTURE ............................................................................. 4
MODULE 4: LEAN DEVELOPMENT AND MANAGEMENT ........................................... 4
MODULE 5: CONTINUOUS SOFTWARE DELIVERY ..................................................... 4
MODULE 6: CONTINUOUS SOFTWARE RELIABILITY ............................................... 5
MODULE 7: STARTING YOUR DEVOPS JOURNEY ....................................................... 5
APPENDIX 1 - K- LEVELS EXPLANATION ................................................................. 6
ACKNOWLEDGEMENT ..................................................................................................... 7
INTRODUCTION TO SYLLABUS

OVERVIEW
This 16 hours Certified DevOps professional course provides the breadth of knowledge and best practices currently applied by high-performing organizations to build software delivery and operational performance capability holistically. This tool-agnostic course helps to understand key challenges and drivers for DevOps adoption, principles, and practices to be used throughout the journey, and the capabilities that would remove the constraints and continuously improve the value delivery.

EXAMINABLE LEARNING OBJECTIVES
The Learning Objectives support the Business Outcomes and are used to create the examination for achieving the Certified DevOps Professional. In general, all parts of this syllabus are examinable at a K1 and K2 level. That is, the candidate will recognize, remember, and recall a term or concept. The specific learning objectives at K1, K2, and K3 levels are shown at the beginning of the pertinent chapter.

EXAMINATION DETAILS

<table>
<thead>
<tr>
<th>Examination type</th>
<th>Multiple-choice questions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of questions</td>
<td>40</td>
</tr>
<tr>
<td>Pass mark</td>
<td>65%</td>
</tr>
<tr>
<td>Open book/notes</td>
<td>No</td>
</tr>
<tr>
<td>Electronic equipment/aides permitted</td>
<td>No</td>
</tr>
<tr>
<td>Time allotted for examination</td>
<td>60 Minutes</td>
</tr>
</tbody>
</table>
MODULE 1: SOFTWARE DELIVERY CHALLENGES

1.1. (K1) Recognize the evolution of software development from Waterfall to Iterative, and now to the continuous model
1.2. (K2) Understand why Agile and DevOps initiatives fail
1.3. (K1) Recognize the need for Business-IT Convergence
1.4. (K3) Understand and analyze the current state of DevOps
1.5. (K2) Define what are the goals of DevOps

MODULE 2: DRIVERS AND PRINCIPLES OF DEVOPS

2.1. (K2) Appreciate the importance of a Production-first mindset
2.2. (K1) Recognize Reliability is more important than velocity
2.3. (K1) Understand the importance of Flow, Feedback, and Continuous Experimentation
2.4. (K2) Understand the Theory of Constraints
2.5. (K1) Learn the capabilities for software delivery and operational for high-performance

MODULE 3: GENERATIVE CULTURE

3.1. (K3) Recognize the stereotypes in the IT team and their impact on software delivery
3.2. (K2) Understand types of Organizational structure based on the Westrum model
3.3. (K1) Learn the impact of Organizational structure in software development based on Conway Law
3.4. (K2) Understand the importance of Transformation leadership

MODULE 4: LEAN DEVELOPMENT AND MANAGEMENT

4.1. (K1) Recognize the need for Value-Driven Enterprise
4.2. (K3) Distinguish Exploring and Exploiting Phases
4.3. (K3) Learn and apply hypothesis-driven development
4.4. (K2) Understand Value Stream Mapping Analysis to maximize value and reduce waste in the flow

MODULE 5: CONTINUOUS SOFTWARE DELIVERY

5.1. (K3) Recognize the key practices for Continuous Delivery
5.2. (K2) Understand the Continuous Testing model
5.3. (K2) Learn the value of Continuous Integration
5.4. (K1) Learn Continuous Security
5.5. (K2) Understand Continuous Delivery
MODULE 6: CONTINUOUS SOFTWARE RELIABILITY

6.1. (K2) The Dickerson Hierarchy of Reliability
6.2. (K2) Recognize the value of Continuous Monitoring
6.3. (K2) Managing Incident Response & Blameless Post-mortem
6.4. (K2) Understand Resilience and Chaos Engineering models
6.5. (K1) Recognize the value of Continuous Feedback

MODULE 7: STARTING YOUR DEVOPS JOURNEY

7.1. (K3) Understand DevOps topologies and patterns
7.2. (K2) Build Metrics that matter
7.3. (K2) Analyze DevOps at Scale
7.4. (K2) Building and Scaling High Performing Teams
7.5. (K3) Learn and Map DevOps Journey
APPENDIX 1 - K- LEVELS EXPLANATION

Source: “Writing Learning Objectives” prepared by Raoul A. Arreola, Ph.D., The University of Tennessee, Memphis.

What is a Learning Objective?

A learning objective is a statement of what students will do when they have completed instruction. A learning objective has three major components:

1. A description of what the student will be able to do
2. The conditions under which the student will perform the task
3. The criteria for evaluating student performance

Cognitive Learning

K1. Basic Knowledge:

To recall and memorize - Assess by direct questions. The object is to test the students’ ability to recall facts and identify and repeat the information.

Recall, identify, recognize, acquire, distinguish

K2. Comprehension:

To translate from one form to another - Assess by having students’

- restate material in their own words,
- reorder or extrapolate ideas, predict or estimate.

Assessments must provide evidence that the students have some understanding or comprehension of what they are saying.

Translate, extrapolate, convert, interpret, abstract, transform

K3. Application:

To apply or use information in a new situation - Assess by presenting students with a unique situation (i.e., one not identical to that used during instruction) and have them apply their knowledge to solve the problem or execute the proper procedure.

Apply, sequence, carry out, solve, prepare, operate, generalize, plan, repair, explain, recognize

K4. Analysis:

To examine a concept and break it down into its parts - Assess by presenting students with a unique situation of the same type but not identical to that used during instruction, and have them analyze the situation and describe the appropriate procedure or solution to the problem.

Analyze, estimate, compare, observe, detect, classify, discover, discriminate, identify, explore, distinguish, catalog, investigate, breakdown, order, determine
K5. Synthesis:

**To put information together in a unique or novel way to solve a problem** – Assess by presenting students with a unique situation NOT of the same type used during instruction, and have them solve a problem by selecting and using appropriate information.

*Write, plan, integrate, formulate, propose, specify, produce, organize, theorize, design, build*

K6. Evaluation:

**To make quantitative or qualitative judgments using standards of the appraisal** - Assess by presenting the students with a situation that includes both a problem and a solution to the problem and justify or critique the solution.

*Evaluate, verify, assess, test, judge, rank, measure, appraise, select, check*

ACKNOWLEDGEMENT

A diverse team created this syllabus. We want to thank the following people (in no particular order) for their contributions in writing and reviewing this document:

**DEVON**

Hari K Gurumoorthi, Phani Bhushan, Rahul Sah, Leo van der Aalst, Markus van Duijn, and Krishan Bhagwandas

**ISQI**

Erika Paasche, Christian Kinne

At DevOn, we believe that organizations need to radically improve their software development to stay ahead of the competition. We do not make step-by-step improvements, on the contrary, we implement innovative solutions that radically change software development for the better. We help organizations solve their impediments and move beyond traditional ways of work.

Moreover, with over fifteen years of experience with Distributed Agile Software Development, DevOn is a trusted partner of various organizations. Since 2004 we have been combining best practices in the field of Lean and Agile Software Development with our experience from Agile transformations. Our Agile Software Development Lab in India consists of self-managing teams that deliver new functionalities every sprint, which are ready for production.

This document contains information that may be privileged or confidential and is the property of DevOn.

Brassersplein 1, 2612 CT Delft, The Netherlands . 015 241 1900 . info@devon.nl, http://www.devon.nl