

קורס הסמכת אבטחת איכות - Foundation Level (ISTQB)

מק"ט 1032 - 40 שעות

אודות הקורס

היכרות עם עולם התוכן של אבטחת האיכות קריטית לכל מפתח בארגון ומוזילה את עלויות הפיתוח והתחזוקה. יתר על כן, סקרים הראו כי כ- 40% ממשאבי הפרויקט מוקדשים לבדיקות. על כן, התמקצעות העובדים, בכל שלב במחזור חיי הפרויקט (מנהלים, מפתחים ובודקים) הינה קריטית להצלחת הארגון כולו.

ארגון ISTQB (International Software Testing Qualification Board) הינו ארגון בינלאומי שמטרתו לקדם את מקצוע בדיקות התוכנה ע"י פיתוח וקידום של סילבוס אחיד, שפה מקצועית אחידה ומסלול התקדמות מקצועית מוכר ואחיד בכל העולם. הארגון מאגד בתוכו כ- 41 מדינות (אירופה, ארה"ב, הודו, יפן, ישראל ועוד) ומדינות נוספות נמצאות בתהליכי אישור מתקדמים.

הקורס המכין להסמכת "Certified Tester - Foundation Level" כולל הכרה תיאורטית ומעשית עם עקרונות מעולם הבדיקות, שילוב הבדיקות כחלק ממחזור חיי הפיתוח, ניהול ושליטה בפרויקט, תערוף הבדיקות בהתאם לניתוח סיכוני הפרויקט, בדיקות מנקודת המבט של המפתח, ראש הצוות ומנהל הפרויקט. הקורס עוסק גם בבדיקות בשלבים מוקדמים בתהליך הפיתוח, בקרת מסמכים, בדיקות טרום כתיבת הקוד, במהלך כתיבת הקוד ובהטמעת הקוד (static testing and white box techniques).

הקורס מכין בצורה מלאה למבחן ההסמכה היוקרתית של ISTQB המעניק למסיימים בהצלחה תעודה רשמית בינלאומית.

מטרות הקורס

- הכנת המשתתפים לבחינת ההסמכה הבינלאומית ISTQB ברמת הבסיסית ולימוד כל הרקע המתודולוגי הנדרש לאנשי QA.

קהל יעד

- מנהלי פרויקטים, בודקים, מפתחים, מנתחי מערכות, אנשי QA וכל גורם המעורב בתהליכי הפיתוח והבדיקות.
- סטודנטים שנה ג' / בוגרי תואר ראשון / מהנדסים / הנדסאים במגמות המחשוב (מדעי המחשב, מערכות מידע).

דרישות קדם

- היכרות בסיסית עם עולם פיתוח התוכנה, או מינימום חצי שנת ניסיון בתחום בדיקות תוכנה או פיתוח תוכנה.
- אנגלית אקדמית בסיסית

הקדמה ל - ISTQB

- תכנית לימודים להסמכה הבסיסית - הצגת הסילבוס, מבנה הקורס, מטרת הקורס וחברת הקורס
- ISTQB - הצגת הארגון העולמי והישראלי ומטרותיו

פרק ראשון – יסודות בדיקות תוכנה

- **מהן בדיקות?**
מטרות אופייניות של בדיקות תוכנה, בדיקות וניפוי באגים.
- **מדוע הבדיקות נחוצות?**
תרומת הבדיקות להצלחה, אבטחת איכות ובדיקות, שגיאות, פגמים וכשלים, פגמים שורש הבעיה וההשפעות שלהם.
- **שבעת עקרונות הבדיקה**
עקרונות מרכזיים בבדיקות תוכנה, כגון: חשוב להתחיל את הבדיקות בשלבים מוקדמים בכדי לאתר תקלות שתיקון בשלבים מתקדמים יותר יעלה הרבה מאוד כסף, חשוב להפריד בין הבדיקות לפיתוח- כל אחד והתמחותו הוא, חובה להגדיר את התוצאות הצפויות מראש על בסיס מסמך הדרישות/ אפיון, חשוב לבצע בדיקות תקפות ולא תקפות, לא ניתן לבדוק הכל - צריך לבדוק בהתאם לתעדוף שהוגדר מראש (על בסיס ניתוח סיכונים) וכדומה.
- **תהליך הבדיקות הבסיסי**
תהליך הבדיקות בהקשר, פעילויות בדיקה ומטלות בדיקה, תוצרי העבודה בבדיקות, נעקבות בין בסיס הבדיקות ותוצרי עבודה הבדיקות.
- **הפסיכולוגיה בבדיקות**
דרך החשיבה של הבודקים והמפתחים, הפרדוקס של בודק התוכנה, היחסים בין צוות הבדיקות לצוות הפיתוח, דגש לתקשורת הבינאישית והחשיבות של בדיקות ע"י צוות עצמאי בלתי תלוי.

פרק שני – בדיקות לאורך מחזור חיי התוכנה

נתייחס למחזור החיים של פרויקט הפיתוח ולשלבי וסוגי הבדיקות השונות לאורך מחזור זה

- **מודלים למחזור חיי תוכנה**
הצגת המודלים השונים של מחזור חיי הפיתוח, בדגש על ה- V model ועל המודל האיטרטיבי והיתרונות שלו בפיתוח "גמיש" לשינויים. הצגת פיתוח תוכנה ובדיקות תוכנה. התאמת מודל מחזור חיי התוכנה לפרויקט ולמוצר.
- **רמות בדיקה**
הצגת שלבי הבדיקות השונים המתבצעים לאורך מחזור חיי הפרויקט, כולל התייחסות לסוג הבדיקות, סביבת הבדיקות, מי עושה את הבדיקות ומה הן צריכות לכלול.
- **Test types**
סוגי הבדיקות נחלקים לארבעה:
 - בדיקות פונקציונליות - בדיקות שמטרתן לוודא שהמערכת עושה מה שהיא צריכה לעשות ולא עושה מה שהיא לא צריכה. בדיקות אלו תמיד יתבססו על מסמך הדרישות/ האפיון.
 - בדיקות לא פונקציונליות - בדיקות שמטרתן לוודא איך המערכת מתפקדת. קיימים סוגי בדיקות רבים, כגון: Load, Stress, Performance, volume, data, configuration, Usability, Backup & Recovery.**(software quality characteristics) ISO 9126** - הצגת ששת מאפייני האיכות הבסיסיים למערכות IT בהתאם ל- ISO 9126, כולל: functionality, reliability, usability, efficiency, maintainability, portability
- קופסה לבנה - בדיקות המבוססות על הבנת מבנה הקוד/ התוכנית, כולל טכניקות למדידת כיוסי הבדיקות

- בדיקות הקשורות בשינויים (coverage) - בדיקות המשויכות לשינויים/ תיקונים כגון: retesting, regression - sanity.
- **בדיקות תחזוקה**
גורמי התחזוקה, ניתוח והשפעה עבור התחזוקה.

פרק שלישי – יסודות הבדיקות הסטטיות

בדיקות סטטיות הן בדיקות המבוצעות ללא הרצת קוד בפועל. בהיבט זה נתייחס לנושאים הבאים:

- **יסודות הבדיקות הסטטיות**
הצגת הבדיקות הסטטיות, מיקומן במחזור חיי הפרויקט, יתרונות וחשיבות של הבדיקות הסטטיות, תוצרים שניתן לבחון על ידי בדיקות סטטיות, ההבדלים בין בדיקות סטטיות לדינמיות.
- **תהליך הסקירה**
תהליך הסקירה של תוצרי תהליך הפיתוח, תפקידים והתחומי אחריות בסקירה רשמית, סוגי הסקירה, יישום טכניקות סקירה, גורמי הצלחה לסקירות.

פרק רביעי – טכניקות בדיקה

- **קטגוריות בדיקה**
קיימות קטגוריות וטכניקות רבות ליצירה ותכנון של תרחישי בדיקה לבדיקות פונקציונאליות. בחלק זה נלמד לבחור טכניקות בדיקה וזאת לאחר הכרת קטגוריות של טכניקות בדיקה והמאפיינים שלהן.
- **טכניקות בדיקה מסוג קופסה שחורה**
מחלקת שקילות, ניתוח ערכי גבול, טבלאות החלטה, החלף מצבים, מקרי שימוש.
- **טכניקות בדיקה מסוג קופסה לבנה**
בדיקות וכיסוי ומשפטים, בדיקות וכיסוי החלטות. הערך של בדיקות משפטים והחלטות.
- **טכניקות בדיקה מבוסס ניסיון**
ניחוש שגיאות, בדיקות חוקרות, בדיקות מבוססות רשימת תיוג.

פרק חמישי – ניהול הבדיקות

- **ארגון הבדיקות**
בדיקות עצמאיות, מאילו תפקידים צריך להיות מורכב צוות הבדיקות, מה רמת החופש/ העצמאות הדרושה לאנשי הבדיקות (האם כפיפות למנהל הפיתוח או גוף עצמאי?). מה תפקידו של מנהל הבדיקות ומה תפקידו ואחריותו של הבודק עצמו.
- **תכנון בדיקות ואומדן הבדיקות**
תכנית על לתכנון הבדיקות הינה מסמך בדיקות ראשון ובעל חשיבות רבה. בפרק זה יוצג המבנה הפורמאלי של תכנית STP על בסיס הסטנדרט המקובל לתיעוד פרויקט בדיקות - IEEE 829. כולל דוגמאות ודגשים לפרקים מרכזיים במסמך תכנון הבדיקות (מה בודקים ומה לא, אסטרטגיית הבדיקות, תנאי סף להתחלה ולסיום הבדיקות וכדומה). מטרת תכנית הבדיקות ותוכן תכנית הבדיקות, אסטרטגיית בדיקות וגישה לבדיקות, קריטריון כניסה וקריטריון יציאה (הגדרה נקודת התחלה ונקודת סיום), לוח זמנים לביצוע בדיקות, גורמים המשפיעים על מאמץ הבדיקות.
- **ניטור ובקרת בדיקות**
כמו בכל פרויקט גם בפרויקט בדיקות ישנה חשיבות רבה מאוד לבקרה ודיווח במהלך חיי הפרויקט. נציג את החשיבות של מעורבות אנשי הבדיקות בדיווח מדויק ונדגים סוגים שונים של דיווחים לאורך שלבי פרויקט הבדיקות. מדדים בשימוש הבדיקות, מטרות, תוכן וקהל היעד של דוחות הבדיקות.
- **ניהול תצורה**
הסבר קצר מה זה ניהול תצורה, למה הוא חשוב לארגון, אילו תקלות נובעות מניהול תצורה לא תקין ומאילו אלמנטים בנוי ניהול תצורה נכון
- **סיכון ובדיקות**

זיהוי הסיכונים (product and project risks), הגדרת פרמטרים המשפיעים על הסיכון, בניית טבלת סיכונים ומטריצת סיכונים.

▪ **ניהול פגמים**

מחזור החיים של הטיפול בתקלות חדשות כולל קביעת סוג התקלה (תוכנה, אפיון, תשתית), הסטטוס שלה, החומרה שלה, תעדוף לתיקון וכיו"ב. בפרק זה נסביר את החשיבות בתיעוד מדויק של כל תקלה/ אירוע, כיצד צריך להיות בנוי טופס דיווח תקלות (שדות קלט מרכזיים) ובנוסף, יוצגו הגורמים השונים המטפלים בכל תקלה והשלבים אותם היא עוברת עד לסגירתה בסטטוס כזה או אחר.

פרק שישי – כלים תומכים בבדיקות

▪ **שיקולים בבחירת הכלי**

הסבר קצר על תפקידם של הכלים האוטומטיים בבדיקות. סיווג הכלים, יתרונות וסיכונים של אוטומציה בבדיקות, שיקולים מיוחדים לכלי ביצוע בדיקות וכלי ניהול הבדיקות.

▪ **שימוש יעיל בכלים**

עקרונות מרכזיים לבחירה של כלים, פרויקט ניסיוני להכנסת כלי לארגון, גורמי ההצלחה לכלים

תרגול והכנה למבחן ההסמכה

בסיום הקורס תבוצע הכנה למבחן ההסמכה אשר תכלול:

- תרגולים על מבחנים לדוגמא
- טיפים למעבר מוצלח
- טיפים כיצד יש להתכונן למבחן