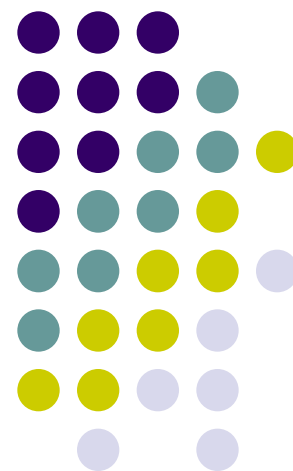


כלים וטכניקות לפיתוח מערכות מידע ממוחשבות

מרצה: שי שקרוב

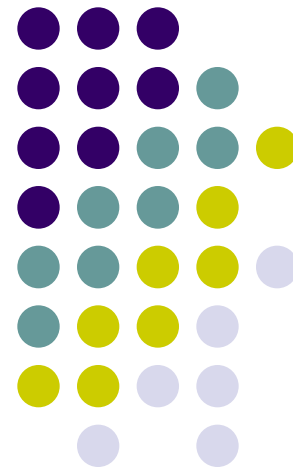




נושאי השיעור

- גישות וכלים לפיתוח ולאחזקה של מערכות מידע
 - אסטרטגיות של פיתוח מערכת מידע
 - גישות להקמת מערכות מידע
 - כלים לתמיכה בתכנון ובפיתוח מ"מ – כלי CASE
 - כלי פיתוח תכנה: שפות תכנות ומחוללי יישומים
 - נהלים ותקנים
- ניהול תהליך הפיתוח
 - פיתוח מ"מ = ניהול פרויקט
 - בעיות שכיחות בפיתוח מ"מ
 - ניהול פרויקט פיתוח מ"מ

גישות וכלים לפיתוח ולאחזקה של מערכות מידע





אסטרטגיות של פיתוח מערכת מידע

- הגדרה: מסגרת תפיסתית, שעוסקת במקומן של מ"מ באירגון, ברמה הכללית, ומציגה תכנית פעולה מתאימה לפיתוחן.

- אסטרטגיות מתאימות:

- אסטרטגיית "כיבוי שריפות"

- אסטרטגיית איסוף הנתונים

- אסטרטגיית התרשים האירגוני

- אסטרטגיות מתאימות:

- אסטרטגיית מלמעלה-למטה

- אסטרטגיית מלמטה-למעלה

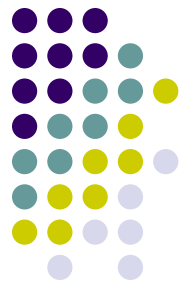
- האסטרטגיה המקבילה

אסטרטגיות של פיתוח מערכת מידע: אסטרטגיית מלמעלה-למטה



- הנחה: לאחר הגדרת כל צרכי המידע שההנהלה דורשת ניתן להתחיל בתכנון מ"מ ובעיצובה.
- האסטרטגיה והתכנון של מ"מ חייבים להיות מושלמים לפני התחלת מימושם.
- יתרון: זיהוי כל הצרכים מראש מקל על שילוב המערכות לאחר מכן.
- חיסרון: נדרש זמן רב לתכנון גלובלי לפני ירידה לפרטים

אסטרטגיות של פיתוח מערכת מידע: אסטרטגיית מלמטה-למעלה



- פיתוח מ"מ תפעוליות מהווה תשתית לבניית מ"מ ניהוליות.
- יתרון: מ"מ עונה על צרכים עכשוויים העולים מהשטח
- חסרונות:
 - חסרה הראייה הכוללת של דרישות המידע באירגון, ולכן תהליך שילוב מ"מ אינו אופטימלי ויוצר עלויות נוספות.
 - חסרה הראיה של מקבלי ההחלטות, ולכן גידול בהקף המערכות עלול לגרום לתכנון מחדש של מערכות העתיד

אסטרטגיות של פיתוח מערכת מידע: האסטרטגיה המקבילה



- שילוב של השתים הקודמות:
- פיתוח מ"מ תפעוליות מלמטה למעלה
- פיתוח מ"מ ניהוליות מלמעלה למטה
- שילוב מ"מ בהדרגה, שבסופו יש מ"מ כללית לאירגון
- ניסיון למזער חסרונות החלופות הקודמות ולמקסם יתרונותיהן.



גישות להקמת מערכות מידע

- האסטרטגיה היא מסגרת תפישתית
- גישת הפיתוח מציגה תכנית פעולה מעשית
- שתי גישות מקובלות:
 - גישת מפל המים
 - גישת אב-טיפוס

גישות להקמת מערכות מידע: גישת מפל המים



- מתבססת על יישום סדרתי של שלבי מחזור החיים: הגדרה - בניה - יישום - תפעול; מבלי לפסוח או לשנות סעיפים.
- בגישה מתודולוגיות שונות:
 - כוללות חלק מהשלבים במחזור החיים או את כולם;
 - אסטרטגיה של מלמעלה-למטה או מלמטה-למעלה.
 - המטרה של כל המתודולוגיות להכניס סדר בתהליך הפיתוח שהיה אינטואיטיבי.

גישות להקמת מערכות מידע: גישת מפל המים



● חסרונות:

- תהליך ארוך
- תהליך יקר
- נתק מסוים בין המפתחים למשתמשים (בגלל אורך התהליך ומורכבותו).
- כאשר מדובר במערכות שאינן מובנות (כמו אלו המיועדות למנהלים בכירים) המשתמשים אינם יודעים מראש מהן הדרישות שלהן מהמערכת, והמקצוענים אינם יודעים כיצד לתרגם צורך לא-מובנה.

גישות להקמת מערכות מידע: גישת מפל המים



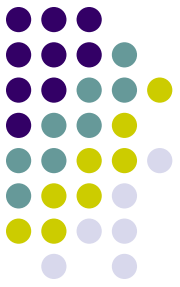
- לאילו מצבים הגישה מתאימה?
- לפיתוח מ"מ מורכבות בעלות היקף גדול שתומכות בשגרות פעולה מובנות ומוגדרות היטב.
- לפיתוח מ"מ גדולות (פרויקט בהיקף גדול)
- לפיתוח מ"מ מובנות, כגון מערכות לעיבוד תנועות.
- כאשר יש זמן לפיתוח.

גישות להקמת מערכות מידע: גישת אב-טיפוס



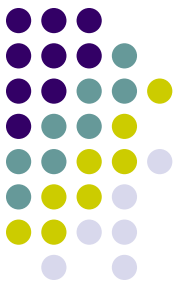
- הגדרה: תהליך איטרטיבי של יצירה מהירה וזולה של מודל חי ועובד שנועד להגדיר ולבחון הנחות ודרישות.
- גישת אב-טיפוס מבוססת על העיקרון שלפיו משתמשים יכולים להצביע על יתרונות וחסרונות של מערכת קיימת ביתר קלות ביחס למערכת תיאורטית.
- אין הסכמה לגבי מהות האב-טיפוס:
 - יכול להיות דגם או מערכת שלמה;
 - יכול להחליף את גישת מפל המים או להתווסף לה.

גישות להקמת מערכות מידע: גישת אב-טיפוס



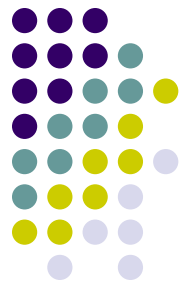
- סוגים של אב-טיפוס
- אב-טיפוס לזריקה
- אב-טיפוס מתפתח

גישות להקמת מערכות מידע: גישת אב-טיפוס לזריקה



- אב הטיפוס הוא מערכת זמנית המסיימת את חייה בתום שלב הבחינה שלה.
- את הדגם בונים בכלים המאפשרים תצוגה ברורה ומהירה אח"כ בונים את המערכת התפעולית בכלים האמיתיים.
- אינו מחליף את גישת מפל המים, אלא מתווסף לה.

גישות להקמת מערכות מידע: גישת אב-טיפוס לזריקה



- יתרונות:
- מאפשר השתתפות אינטנסיבית של המשתמשים בהגדרת הצורך וביצירת אב הטיפוס - מקטין את סיכוני הכשלון.
- השילוב עם גישת מפל המים - מאפשר להנות מיתרונותיה.
- היבטים החשובים למשתמש תופסים מקום מרכזי.
- חסרונות: ההשקעה באב הטיפוס יורדת לטמיון
- מתאים למערכות TPS

גישות להקמת מערכות מידע: גישת אב-טיפוס מתפתח



- זהו אב טיפוס שנועד לעבור שינויים, הרחבות ותוספות, ולבסוף להפוך למערכת עצמה.
- מחליף את גישת מפל המים.
- השלבים (שלבים 3-4 מחזוריים):
 1. זיהוי דרישות ראשוניות של המשתמש
 2. פיתוח אב טיפוס
 3. שימוש באב הטיפוס והערכתו
 4. ביצוע רויזיה ושיפור של אב הטיפוס.
- בד"כ נעשות 4-6 איטרציות עם המשתמשים

גישות להקמת מערכות מידע: גישת אב-טיפוס מתפתח



יתרונות:

- המשלב הראשוני מקטין את סיכוני הכשלון.
- בשיטת הניסוי והטעיה יגלו המשתמשים נקודות לשיפור ולשינוי
- לרוב גישה זו זולה מגישת מפל המים
- הלקוח יכול להשתמש בחלקים מהמערכת כבר בשלב הפיתוח
- ההשקעה באב הטיפוס אינה מתבזבזת

חסרונות:

- לוקח יותר זמן לבנות דגם מסוג זה (ביחס לדגם מתכלה)

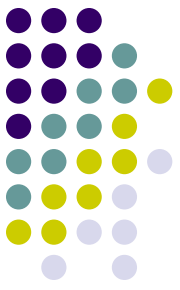
גישות להקמת מערכות מידע: גישת אב-טיפוס מתפתח



מתאים ל:

- מ"מ ניהוליות DSS / MIS. כיוון שאלו מ"מ שאינן עתירות נתונים ותנועות, ולא צורכות בהכרח משאבי מחשב רבים, ולפיכך ניתן ליישמן על מיקרו מחשבים.
- כאשר מהירות הפיתוח חשובה יותר מיעילות האב טיפוס (כאשר אין זמן לפיתוח)
- כאשר ישנה סבירות גבוהה שאב הטיפוס לא יהיה מושלם או לא יתאים (כאשר הצרכים לא ברורים)

כלים לתמיכה בתכנון ובפיתוח מ"מ – כלי CASE



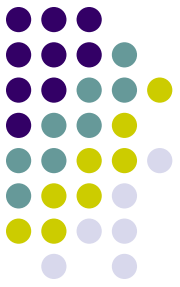
- כלים ממוחשבים שמטרתם סיוע בתכנון והפיתוח תכנה
- Upper CASE: כלים לתיאור האירגון ותכניותיו
- מבנה האירגון, תכניות, מטרות, מדדים להשגת יעדים
- Middle CASE: כלים התומכים בניתוח מ"מ
- הגדרת ישויות, פעילויות, זרימת מידע, מאגרי מידע
- Lower CASE: כלים היוצרים תכניות מחשב לתהליכים ולתיעודם

כלי פיתוח תכנה: שפות תכנות ומחוללי יישומים



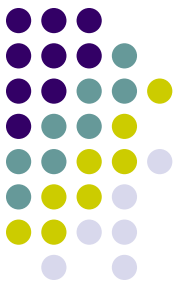
- שפות תכנות מושתתות על אוסף של פקודות וכללי תחביר שבאמצעותם מעבירים הוראות ביצוע למחשב
- 4 דורות עיקריים של שפות תכנות:
 - דור ראשון – שפת מכונה
 - דור שני – שפות סף
 - דור שלישי – שפות עיליות
 - דור רביעי – מחוללי יישומים

כלי פיתוח תכנה: דור ראשון – שפת מכונה



- שנות ה-40-50 של המאה ה-20
- כל הפקודות נכתבות בשפת המכונה, ע"י שימוש בסימנים 0,1.

כלי פיתוח תכנה: דור שני – שפות סף



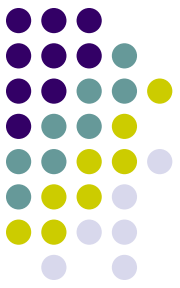
- שפות אסמבלר
- סוף שנות ה-50 של המאה ה-20
- התכניות בנויות מקודים שמסמלים את פעולות המחשב
- לאחר כתיבתה עוברת התכנית תהליך של הידור – תרגום התכנית לשפת מכונה
- משמשות כיום כשיש צורך ביעילות מרבית
- דוגמאות: תוכנת המערכת; תוכנות אבטחה; תוכנות לניהול מסדי נתונים



כלי פיתוח תכנה: דור שלישי – שפות עיליות

- לקראת שנות ה-60 של המאה ה-20
- שימוש בפקודות הקרובות לשפה האנגלית
- מאפשרות למתכנת להגדיר "כיצד תבוצע המשימה"
- ברמה גבוהה יותר מאשר בשפות אסמבלר
- לאחר כתיבתה עוברת התכנית תהליך של הידור
- מרבית התוכנות נכתבות בשפות מדור זה
- שפות לדוגמא: Basic, VB, C, C++, HTML, JAVA, FORTRAN, COBOL

כלי פיתוח תכנה: דור רביעי – מחוללי יישומים



- מתמקדות ב"מה צריך לעשות" ולא ב"איך לעשות"
- מיועדות למשתמשים שאינם תוכניתנים, או כדי לקצר את משך התכנות
- כוללים מרכיבים מובנים ולא-מובנים, המאפשרים למשתמש לשלב בין יעילות לקלות בתכנות.
- דוגמאות: Magic, Access, Oracle Applications, Focus



נהלים ותקנים (1)

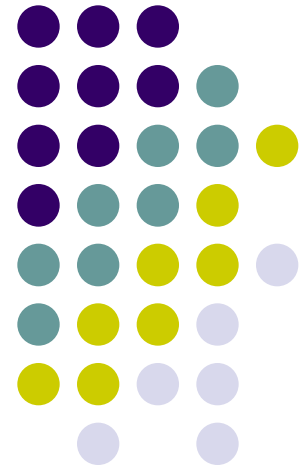
- בכל אחד משלבי מחזור החיים יש להפיק תוצרים ומסמכי לוואי שמפרטים את הפעילויות הרלוונטיות ומתעדים אותן.
- מטרת הנהלים והתקנים היא להנחות את תהליך הפיתוח ולפרט את התיעוד הנדרש
- נהלים מוכרים לפיתוח ותחזוקה של מ"מ:
 - תקן ANSI/IEEE של מכון התקנים האמריקני
 - נוהל מפת"ח של חברת מתודה
 - תקן לתיעוד של מכון התקנים הישראלי



נהלים ותקנים (2)

- השימוש בנהלים מוכנים אינו מחייב שימוש באסטרטגיה, גישה, או שפת תכנות מסויימות.
- הדגש בנהלים הוא על תהליכי עבודה ועל התוצרים, ולא על הטכניקות להשגת התוצרים.

ניהול תהליך הפיתוח





פיתוח מ"מ = ניהול פרויקט (1)

- פיתוח מ"מ כרוך ב:
 - תכנון
 - השקעת משאבים (כ"א, זמן, כסף, ...)
- פרויקט פיתוח מתבצע בסביבת סיכון ו/או אי-ודאות.
- משליך על ביצועי האירגון – יכול להביא לתוצאות הנעות בין הצלחה לפשיטת-רגל



בעיות שכיחות בפיתוח מ"מ

- חריגה בעלויות הפיתוח והיישום
- חריגה בלוחות הזמנים
- שינויים באירגון ובסביבתו במהלך הפיתוח והמשפיעים על המערכת שבפיתוח
- דרישות עיקריות למידע לא נכללו בהגדרת מ"מ
- התקנת המערכת כרוכה בתקלות לא צפויות
- הטמעת המערכת נתקלת בהנגדות אירגונית
- התועלת המתוכננת אינה מושגת



הסיבות לבעיות השכיחות

- המנהלים והמשתמשים אינם מעורבים בתהליכי ההגדרה והפיתוח
- לא הוקדשו משאבים מספיקים לשלב ההגדרה
- היקף המערכות רחב מדי
- אי-קיום מסגרת שיטתית שלמה וברורה לתכנון תהליך הפיתוח ולבקרתו



פיתוח מ"מ = ניהול פרויקט (2)

● מסקנות:

- אין להשאיר את תהליך הפיתוח בידי מפתחי מ"מ בלבד
- נדרשת מעורבות של המשתמשים בפיתוח
- נדרשת מעורבות פעילה של המנהלים בתהליך
- חלוקת העבודה בין ההנהלה, המשתמשים והצוות המפתח צריכה להיות ברורה ומוגדרת מראש
- יש להבנות את תהליך הפיתוח של מ"מ
- יש להגדיר נקודות בקרה והחלטה, ונוהל ברור לפעילות בנקודות אלו



פיתוח מ"מ = ניהול פרויקט (3)

- קריטריונים כלליים להצלחה בפרויקט פיתוח:
 - תקציב
 - לוחות זמנים
 - ביצועי המערכת
 - שביעות רצון המשתמשים



ניהול פרויקט פיתוח מ"מ

● ארבעה שלבים, שלעיתים יש חפיפה ביניהם:

● ייזום

● תכנון

● ביצוע

● סיום



ניהול פרויקט פיתוח מ"מ (1)

ייזום ← תכנון ← ביצוע ← סיום

- תאור הבעיה

- תיאור הסביבה

- קריטריונים למערכת (דרישות הלקוח) ואילוצים (דרישות בעלי העניין).

- פיתוח מדדי הצלחה למ"מ החדשה: זמן, עלות, ביצועים, שביעות רצון לקוח, סיכון, קריטריונים מיוחדים

- פיתוח חלופות

- RFI

- RFP



ניהול פרויקט פיתוח מ"מ (2)

ייזום ← תכנון ← ביצוע ← סיום

● הערכת חלופות

● ניתוח עלויות – עלות מחזור החיים של החלופות

● עלות פיתוח/רכש; עלות פיתוח תשתיות; עלויות תפעול ותחזוקה; ערך גרט

● ניתוח תקציבי

● השוואת קו התקציב הצפוי לפרויקט והשוואתו לתקציב שהוקצה ע"י האירגון < איתור נקודות קושי > תזמון מחדש

● ניתוחי רגישות

● בחירת חלופה



ניהול פרויקט פיתוח מ"מ (3)

ייזום ← תכנון ← ביצוע ← סיום

● מסמך ייזום

● תמצית – מטרה

● מנהל הפרויקט

● אופן הביצוע

● בעלי עניין ואפי מעורבותם

● מדדי ביצוע גלובליים

● מועדים ומשכים; תקציב; מעטפת ביצועים; שביעות רצון
לקוחות; סיכונים



ניהול פרויקט פיתוח מ"מ (4)

ייזום ← תכנון ← ביצוע ← סיום

- מבנה אירגוני לפרויקט
- מבנה תכולת העבודה
- חלוקת הפרויקט לחבילות עבודה
 - תוצר, משך הזמן לביצוע
 - אפיון בדיקות לקבלה/דחיה של החבילה
 - בדיקות מבוצעות ע"י הלקוח של החבילה
- ניהול תצורה: מערכת ניהולית שתפקידה לשמור על תצורת המערכת, תוך טיפול שיטתי בהכנסת שינויים.
- השינויים מאושרים ו/או מובאים לידיעת כל מי שנדרש



ניהול פרויקט פיתוח מ"מ (5)

ייזום ← תכנון ← ביצוע ← סיום

- תזמון הפרויקט: קביעת המועדים שבהם משאבים שונים מבצעים פעולות שונות.
- תזמון הוא שילוב של אילוצי סדר ומשכי ביצוע עם אילוצי משאבים ותקציב
- סוגים של קשרים בין פעולות (2 דוגמאות):
- סוף-התחלה: לא ניתן להתחיל בפעולה 2 לפני סיום 1
- חוסר קשר: אין תלות בין פעולות
- נתיב קריטי: הזמן המינימלי לביצוע הפרויקט
- כלים לתזמון: טבלה, תרשים גאנט, תרשים רשת, ...



ניהול פרויקט פיתוח מ"מ (6)

ייזום ← תכנון ← ביצוע ← סיום

- תכנון משאבים ותקציב
- הזמן הדרוש לביצוע פעולה תלוי במשאבים
- ניתן לשנות את משך הזמן הדרוש לכל פעולה ע"י שינוי במשאבים.
- לשינוי במשאבים יש גם השלכה תקציבית
- צריך להתחשב גם בתזרים המזומנים
- ניהול סיכונים (יפורט בנפרד)



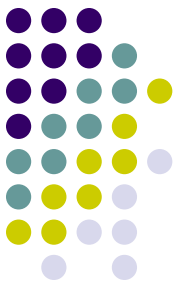
ניהול פרויקט פיתוח מ"מ (7)

ייזום ← תכנון ← ביצוע ← סיום

- פיתוח מ"מ בפועל
- הפעלת מערכות פיקוח ובקרה
 - בקרת עלות
 - בקרת לו"ז
 - בקרת תצורה ואיכות
- כתוצאה מהבקרה ו/או משינויים בדרישות במהלך הפרויקט מעדכנים את הלו"ז, העלויות והתצורה.
- ניהול סיכונים (יפורט בנפרד)

ניהול פרויקט פיתוח מ"מ (8)

ייזום ← תכנון ← ביצוע ← סיום



- סיום הפרויקט
- העברת מ"מ ללקוח
- הפקת לקחים
- סגירת חשבונות

ניהול פרויקט פיתוח מ"מ (9)

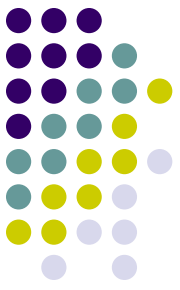
ניהול סיכונים בפרויקט פיתוח



- סיכונים המתייחסים למימוש כושל של מ"מ
- בשלב הייזום והתכנון יש לבצע:
 - ניתוח סיכונים ראשוני
 - הערכה כמותית של הנזקים
 - פיתוח תכניות פעולה להקטנת הסיכונים המזוהים
- בשלב הביצוע יש לנהל הסיכונים:
 - מימוש תכנית הפעולה להקטנת הסיכונים
 - הערכת סיכונים שוטפת
 - עדכון תכניות הפעולה להקטנת הנזקים

ניהול פרויקט פיתוח מ"מ (10)

ניהול סיכונים בפרויקט פיתוח



- ניהול הסיכונים באחריות מנהל הפרויקט
- הערכת הסיכונים מתבצעת ע"י צוות של אנשי פיתוח ומשתמשים.
- ההערכה הראשונית כוללת את השלבים:
 - זיהוי אירועים מסוכנים
 - כימות הסיכון: הסתברות ועוצמת הנזק
 - בניית תכניות פעולה מתאימות

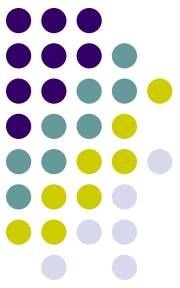
ניהול פרויקט פיתוח מ"מ (11) תהליך ניהול הסיכונים



1. זיהוי הסיכונים;
2. הערכת חומרת הסיכון והסתברות לסיכון;
3. בחירת שיטת הטיפול בסיכון:
 - א- חמיקה מסיכון (המנעות מהפעילות)
 - ב- אימוץ הסיכון (לקחת סיכון ולשלם אם יתרחש);
 - ג- הקטנת הסיכון (פעולות להקטנת חומרת הנזק, לא ההסתברות)
 - ד- העברת הסיכון (ביטוח; חכירה; קבלני משנה)
4. ניהול סיכונים שוטף (מינוי אדם או יח' לנושא)

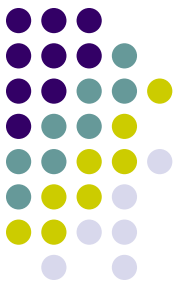
ניהול פרויקט פיתוח מ"מ (12)

ניהול סיכונים בפרויקט פיתוח



- דוגמאות לתכניות להקטנת נזקים:
- שימוש בקבלני משנה בפעולות בהן הסיכון גבוה
- פיתוח ואימוץ חלופות אחרות (למערכת, לרכיב, לתהליך)
- מעבר לטכנולוגיה חליפית זמינה ומוכחת
- רכישת ביטוח
- תכנון לאימוץ סיכון – תכניות מגירה; הכנת רזרבה תקציבית

ניהול פרויקט פיתוח מ"מ (13) בקרה ניהולית על תהליך הפיתוח



- להצלחת מ"מ דרושה מעורבות פעילה של המנהלים ושל המשתמשים בתהליכי הפיתוח והבקרה.
- כדי שמעורבות ההנהלה והמשתמשים תהיה יעילה ואפקטיבית יש להקים מסגרות אירגוניות מתאימות
 - ועדת מ"מ
 - ועדת ההיגוי של הפרויקט
 - צוות הפיתוח
 - צוות היישום



לחמאנת