

יסודות הנדסת תוכנה

היקף התכנית

- עיוני: 72 שעות

דרישות קדם

הידע המוקדם הנדרש ללימוד הפרק הזה הוא הכרת מדעי המחשב בהיקף של 5 יחידות לימוד מלימודי התיכון. תלמידים שאין להם ידע מוקדם יצטרכו להשלים; החומר **לא** יילמד במסגרת קורס זה.

אוכלוסיית היעד

כיתות י"ג במגמת הנדסת תוכנה.

מטרות הקורס

- להכיר מהי מערכת מידע, את תכונות המערכת ואת המערכת כתת-מערכת של הארגון כמערכת.
- להציג כלים גרפיים הנתמכים על ידי תיעוד מתאים, שיסייעו לתאר ולתכנן מערכות תוכנה (במיוחד מערכות תוכנה בגישה מונחה-עצמים). הכלים עוזרים לתאר את המערכת ברמת הפשטה גבוהה יותר משניתן להשיג באמצעות כתיבת קוד.
- ללמד עקרונות של עיצוב מערכת.
- להציג ולתרגל כלים חדישים לעיצוב מערכות מידע ממוחשבות. הדגש יהיה בעיצוב הלוגי תוך הבחנה.
- להציג פיתוח דגם של מערכת המידע תוך שילוב התנסות בנושאים שנלמדו במסגרת הקורס.

משרד החינוך
מנהל מדע וטכנולוגיה
מגמת הנדסת תוכנה

טבלת הפרקים וחלוקת השעות המוצעת

פרקי הלימוד	עיוני
1 ארגונים ומערכות מידע	6
2 שיטת וכלים לניתוח מערכת מידע	40
3 עיצוב מערכת מידע	16
4 פיתוח אב-טיפוס	4
5 מבחנים	6
סך הכול	72

פרק 1: ארגונים ומערכות מידע

- עיוני: 6 שעות

מטרת הפרק

להכיר מהי מערכת מידע, את תכונות המערכת ואת המערכת כתת-מערכת של הארגון כמערכת.

פירוט התכנים

- 1.1 מבוא לארגון ולמערכת המידע
עיוני: 1 שעה
מטרה: להציג מושגים בנושא הארגון חשיפת התלמיד להיכרות במערכת המידע כתת-מערכת הארגון.
- 1.2 תהליכים בארגון
עיוני: 2 שעות
מטרה: להציג ולתרגל מכלול תהליכים וישויות בארגון.
- 1.3 תהליך פיתוח מערכת מידע
עיוני: 3 שעות
מטרה: להציג את המושג 'מחזור חיי מערכת' להציג את השלבים בבניית המערכת:
 - ייזום
 - חקר מצב קיים
 - חקר ישימות
 - ניתוח ודרישות
 - יישום
 - בדיקה
 - הדרכה והטמעה

פרק 2: שיטות וכלים לניתוח מערכת מידע

- עיוני: 40 שעות

מטרת הפרק

להציג כלים גרפיים, הנתמכים על ידי תיעוד מתאים, שיסייעו לתאר ולתכנן מערכות תוכנה (במיוחד מערכות תוכנה בגישה מונחית-עצמים). הכלים עוזרים לתאר את המערכת ברמת הפשטה גבוהה יותר משניתן להשיג באמצעות כתיבת קוד.

פירוט התכנים

2.1 מהו ניתוח מונחה עצמים

עיוני: 2 שעות

מטרה: להציג את הגישה המונחית-עצמים (OO – Object Oriented).
תכנים: מושגים ראשוניים כגון מחלקה, עצם/מופץ, קשרים בין עצמים ותקשורת בין עצמים.

2.2 מבוא למתודולוגיית ה-UML

עיוני: 4 שעות

מטרה: להסביר את הגישה החדשה שמציבה את הלקוח במרכז.
להציג את ה-UML כשיטה גרפית שבעזרתה נוכל לאפיין ולעצב מערכת בטכנולוגיה מוכוונת-עצמים בצורה אחידה, ברורה, העוסקת במישרין בעולם האמיתי.

– מטרות ה-UML

– שיטת ה-UML, יתרונותיה וחסרונותיה

– שיטת הסימון (הנוטציה)

2.3 מרכיבי ה-UML

עיוני: 34 שעות

מטרה: להציג לתלמידים את כל הכלים העומדים לרשות המפתח המשתמש ב-UML.

אנו משתמשים ב-UML כדי לסייע לנו לרכז את המפרט של המערכת שלנו.
מחלקים את מודלים המפרט לשלוש קבוצות:

– מודל המצב

– מודל ההתנהגות

– מודל שינויי מצב

2.3.1 Use Case Diagram

עיוני: 8 שעות

מטרה: להקנות כלים לבניית מודל התנהגות.

– היכרות עם המושגים תרחיש ו-use case

– שילוב ה-use case בפרויקט בשלב הניתוח כחלק מהמודל ההתנהגות ואופן פירוט הגדרת הדרישות הפונקציונריות

– היכרות ושימוש באמצעי גרפי פשוט כדי לתאר – בד בבד עם תיאור ה-use case – את תרחישי המערכת

– תיאור הקשר בין המשתמשים (actors) לבין המערכת

– היכרות עם סוגי הקשרים: include, generalization, extends

– היכרות עם אופן התייעוד של תרשים ה-use case

2.3.2 מושגים בניתוח ועיצוב מונחה עצמים (OOAD)

עיוני: 10 שעות

מטרה: להציג את הנושא כתגובה לצורך של פיתוח תוכנה, תוך שימוש חוזר ברכיבים קיימים (Code Reuse).

שיקולי הנדסת התוכנה שהביאו לידי התפתחות תכנות מונחה-עצמים.

2.3.2.1 הגדרת המונח עצם (object)

– העצם כמודל לתפיסת נתונים והתנהגות ביחד

– העצם כאוסף של נתונים-תכונות (attributes) ופעולות המוגדרות על אותם נתונים ומתודות (methods)

– העצם במונח של קופסה.

– הגדרת המצב (state) של העצם

2.3.2.2 הגדרת המונח מחלקה (class)

– המחלקה כמודל כללי של עצמים המסווגים לאותה מחלקה

– המחלקה כמבנה נתונים ופעולות

- המחלקה ככלי ליצירת מופע של עצם
- המונח בנאי המחלקה (constructor)
- 2.3.2.3 יחס בין מחלקות
 - ארגון המחלקות כהיררכיה מסוימת. מחלקות-על (super classes) ומחלקות משנה (subclasses). הבנת היחס הזה כחוס של הורשה (inheritance)
 - מחלקת הבסיס (base class) והמחלקת המשנה כאל מחלקת מורשת (derived class)
 - הדגשת המונח 'יחס-הורשה' (is-a)
 - שימוש במחלקות המכילות מחלקות אחרות. יחס הכלה (has a)
- 2.3.2.4 ממשק (interface)
 - ממשק כמחלקה שאינה מכילה תכונות
 - הקשרים: composition, association
 - ממשק כפתרון בעיית ההורשה המרובה כדי למנוע ממחלקה לרשת תכונות זהות או מנגדודות מכמה מחלקות
- 2.3.2.5 מסרים – הודעות (messages) בין האובייקטים
 - העברת מסרים בין עצמים בתגובה על שינוי מצבו של אחד העצמים
 - טכניקה של הפעלת מתודות ככלי למימוש העברת מסרים
- 2.3.3 תרשים מחלקות (class diagram)
 - עיוני: 10 שעות
 - מטרה: לתת תיאור מבט סטטי במחלקות בתוך המודל או בחלק מהמודל. התרשים הזה הוא מפת הדרכים שתלווה את המשך הפרויקט בשלבים הבאים.
 - פרק זה מאוד חשוב ומשמש יסוד להמשך הפרויקט.
 - תיאור המחלקות השונות של הפרויקט והקשרים ביניהם
 - תכנון המחלקות, הממשקים הכולל הגדרת המאפיינים (תכונות) של המחלקות ואת השיטות שלהם
 - הכרת היחסים בין המחלקות (הורשה, composition, aggregation, ו-association) וההבדלים ביניהם
 - הגדרת ה-roles וה-multiplicity בין המחלקות והצגתם בתוך התרשים
- 2.3.4 תרשים רצף (sequence diagram)
 - עיוני: 4 שעות
 - מטרה: להציג תרשים שמתאר את ההודעות (האינטראקציות) כשהן מסודרות ברצף זמן.
 - מבנה התרשים
 - סימנים מוסכמים: אובייקטים, מחלקות, הודעות
 - אילוצים
 - הערות
 - תיאור יצירה ופירוק
 - דוגמה מסכמת
- 2.3.5 תרשים פעילויות (diagram activities)
 - עיוני: 2 שעות
 - מטרה: ללמד לבנות תרשים תהליך/אירוע כרצף פעילויות המתבצעות בו מתחילתו ועד סופו, תוך ציון תחומי אחריות (מחלקות או ישויות)
 - תיאור מצב התחלתי (initial state) ומצב סופי (final state)
 - תיאור מעבר (transition)
 - תנאי (branch)
 - פיצול, מיזוג (fork, merge)

- פעילות (activity)
- מצב אובייקט (object flow state)
- דוגמה מסכמת

פרק 3: עיצוב מערכת מידע

- עיוני: 16 שעות

מטרת הפרק

ללמד עקרונות עיצוב מערכת; להציג ולתרגל כלים חדישים לעיצוב מערכות מידע ממוחשבות. הדגש יהיה בעיצוב הלוגי תוך הבחנה בין ההיבטים הלוגיים והפיזיים של מרכבי המערכת.

פירוט התכנים

3.1 עיצוב מסדי נתונים

עיוני: 8 שעות

מטרה: להכיר את המושג 'בסיס הנתונים' (ישויות וקשרים, נרמול בסיס

נתונים

ו-SQL); להציג לתלמידים את המודל הטבלאי (יחסי), ישויות, תכונות, מפתחות

– מושג המפתח לסוגי

– בניית טבלאות

– סוגי טבלאות: בסיס, פירוט, קשר, תת-טבלה

– נרמול טבלאות בעזרת מודל bcnf

– יצירת קשרים בין טבלאות

– יסודות SQL

3.2 עיצוב תוכנה

עיוני: 4 שעות

מטרה: להציג את עצי מבנה התכניות תוך שימוש בעקרונות הנדסת תוכנה;

התלמידים יכירו את מושג המודול וילמדו טכניקות לתיאור המודולים.

– הגדרת המודול

– בניית עץ מודולים

– עקרונות בניית עץ תפריטים וקשר לעץ המודולים

– כלים לתיאור המודולים (עברית מבנית, טבלאות ועצי החלטה), שיקולים

לבחירה ושילוב ביניהם

3.3 עיצוב ממשקים

עיוני: 2 שעות

מטרה: ללמד את עקרונות עיצוב הקלטים והפלטות במערכת מידע; להציג

בניית מודל אחיד של ממשקים.

– עקרונות עיצוב ממוקד-משתמש

– היררכיה של המידע

– מסמכי מקור, מסך, דוח

– שימוש בצבע ובעיצובים מיוחדים (צליל, גופן, נגישות, הדגשה, קול)

משרד החינוך
מנהל מדע וטכנולוגיה
מגמת הנדסת תוכנה

– אחידות ושילוב
3.4 בדיקות תוכנה
עיוני: 1 שעה

מטרה: להסביר את חשיבות הבדיקות; התלמידים יכירו ויתרגלו בדיקות מקובלות.

- בדיקות לוגיות
- בדיקות תקינות
- בדיקות סבירות
- תכנון בדיקות

- 3.5 גיבוי ואבטחת מידע
עיוני: 1 שעה
מטרה: להדגיש את חשיבות הגיבוי ואבטחת המידע
- | | |
|------------|---|
| מבוא | – |
| גיבוי | – |
| אבטחת מידע | – |

פרק 4: פיתוח אב-טיפוס

- עיוני: 4 שעות

מטרת הפרק

- להציג פיתוח דגם של מערכת המידע תוך שילוב התנסות בנושאים שנלמדו במסגרת הקורס.
- שלבים בפיתוח אב-טיפוס
 - חלוקה למשימות והצגת פתרונות תוך שילוב הצעות התלמידים (כאשר הדבר אפשרי)

ביבליוגרפיה

1. **מבוא למערכות מידע.** המרכז לטכנולוגיה חינוכית, 1999 (פרקים: 1, 2, 3, 4, 5.1, 6).
2. אליהו, יניב, **ניתוח מערכות מידע: כולל את מתודולוגיית ה-UML.** הרצליה: הוד-עמי, 2004 (חלק 1, פרקים 7, 8).

ספרות עזר

1. **ניתוח ועיצוב מערכות מידע.** המרכז לטכנולוגיה חינוכית, 1994 (כרך א, פרקים 1, 2.1, 2.2, 2.5, 2.6, 3, 5).
2. שובל, פרץ, **תכנון ניתוח ועיצוב מערכות מידע.** תל-אביב: האוניברסיטה הפתוחה, 1998 (כרך א, יחידות 1, 3).*
3. חוטובלי, ראובן וחוטובלי פנינה, דרכים להוראת ארגון קבצים ובסיסי נתונים. תל-אביב: הוצאת אורט, 2002.*
4. Coad, Peter, Yourdon, Edward, Object-Oriented Analysis. New Jersey: Prentice-Hall, 1990.*