



## פרויקט 4 – מפרט תיכון תוכנה (SDS)

### מטרת המשימה

מטרת משימה זו היא להגדיר את מפרט התכן – Software Design Specification – של הפרויקט שלכם, מפרט שעליו יתבסס בהמשך המימוש. המפרט יבוצע כאוסף דפי ויקי מקושרים באתר הפרויקט ויתבצע סקר על תכן זה. בנוסף, תעדכנו את תכניות הפיתוח ותדווחו על התקדמות בנושאי סיכונים להצלחת הפרויקט ופיתוח תשתיות.

### הנחיות הגשה

עד התאריך הנקוב לכל תת-משימה עדכון אתר הפרויקט ושליחת קישורים למייל הקורס עם כותרת הודעה: SE Project 4 SDS - <your project name>

### הנחיות עבודה

#### כללי

נפגשתם ועבדתם כבר עם הלקוחות כדי לאסוף ולנתח את דרישות המוצר. המשימה הבאה היא לתכנן את המימוש ע"י בחירת ארכיטקטורה מתאימה וניתוח מונחה עצמים ראשוני של התוכנה. פעילויות אלו יעזרו להמשיך תכנון ומימוש המוצר שלכם.

כשתשלימו את מפרט התיכון ולאחריו ואת רשימת המשימות תוכלו לשכנע את הלקוחות ש:

- אתם מבינים מהו המוצר שאתם בונים
- יש לכם רעיונות מוצקים כיצד לבנות אותו
- יש לכם את המשאבים המתאימים לעשות זאת (נרחיב זאת במשימה הבאה)

המפרט נכתב כאוסף דפי ויקי במטרה להגיע לתיכון דינמי ומשותף ככל האפשר. חישבו על דפים אלו כמצגת נושמת המאפשרת לגורם חיצוני התרשמות מהתכנון שעשיתם.

### תוצרי מסירה

1) עד להרצאה הבאה - 75% - מפרט תיכון כדפי ויקי באתר הפרויקט. המפרט יכיל דף ראשי עם תוכן כדלקמן והפניות לדפים המכילים דיאגרמות תיכון מזוויות שונות. יש לכלול לפחות ארבע דיאגרמות UML: דיאגרמת הפצה (Deployment) אחת, דיאגרמת מחלקה (Class) אחת ושתי דיאגרמות רצף (Sequence) – סעיפים 2-4 להלן. כמו כן במסגרת ה- SDS עליכם להתחיל כבר לתכנן את הבדיקות הנדרשות מהמוצר ואת התיעוד שיימסר ללקוח (בדרך זו תוכלו בהמשך להעריך יותר נכון את המאמץ הכולל הנדרש).

2) עדכון תכניות הפיתוח, טיפול בסיכונים והכנת תשתיות:  
5% - פיתוח רשימת הסיכונים ממשימת אתחול הפרויקט, כך שתכיל טבלה ממוינת לפי חומרת הסיכון, כאשר לכל סיכון מופיעים הפרטים הבאים:

- הסיכון
- רמת השפעתו אם יתרחש (גבוה, בינוני, נמוך)
- רמת הסיכוי שיתרחש (גבוה, בינוני, נמוך)
- צעדים הנלקחים כדי להנמיך את הסיכון

• תכנית פעולה חליפית למקרה שיתרחש

10% - לגבי הסיכון העיקרי דווחו על בדיקה מעמיקה יותר או אב-טיפוס לתשתית שבניתם כדי לבדוק את יכולתכם להתמודד עם הסיכון ומה תוצאות הבדיקה כרגע (במשימה הבאה תתבקשו להעלות את הקוד לאתר הפרויקט) – יש לשים הפניה לדיווח מהעמוד הראשי בוויקי. יש לכלול גם קישור למדריך בסיסי שמצאתם ואתם מתרגלים לפיו את הסביבה המתוכננת – למשל [זה](#).

(3) יומיים אחרי ההרצאה - סקר תיכון אישי (הגשה אישית 3) – כל חבר צוות בוחר פרויקט שונה מרשימת הפרויקטים ומבצע עליו סקר תיכון לפי התבנית והמחווון שבהמשך המסמך **TBD** – את הסקר יש לפרסם בוויקי שלכם ולהגיש כמשימה אישית וגם להפנות אותו לצוות הפרויקט.


(4) אופציונלי: ארבעה ימים אחרי ההרצאה - 10% - עדכון מסמכי התיכון לאחר הסקר (שליחת מייל עם קישור לדפי היסטוריה שבהם ניתן לראות בהערות את השינוי שהתבצע).

קישורים להרחבה

- Amber, [Introduction to Object-Orientation and the UML](#), Especially see: [deployment](#) (and [guidelines](#)), [class](#) (and [guidelines](#)), [sequence](#) (and [guidelines](#)) diagrams, [CRC Models](#) (and another [good explanation](#)).
- Hiranabe, [Modeling in the Agile Age: What to Keep Next to Code to Scale Agile Teams](#)

בהצלחה!


## מפרט תיכון תוכנה - SDS

 השפעה על התהליך: מסמכי התיכון מגדירים את מוצר התוכנה שיספק את הדרישות. ההחלטות המתקבלות בהם נסמכות על הדרישות וכן על ההבנה של האפשרויות הטכנולוגיות והרכיבים הזמינים. העבודה על התרשימים השונים עוזרת לתכנן את הרכיבים השונים במערכת, הממשקים והקשרים ביניהם ולאחר מכן ניתן לעבור כבר למימוש ובדיקות.

### 1. דף ראשי

תיאור התוכן העיקרי של המידע  
כאן ניתן לתאר את המטרות העיקריות של דפי התיכון (בסדר יורד), למשל הכנה למימוש, בדיקת נכונות, בדיקות, יעילות וכדו'.

### 2. תרשימים (ארכיטקטורת) הפצה – Deployment Diagrams

 ספקו לפחות: דיאגרמת UML אחת המתארת את הרכיבים הפיסיים (jar, exe, dll, כגון) שמתוכננים למוצר שלכם ואת סוג הממשקים ביניהם. את הדיאגרמות מומלץ להכין במוצר ייעודי ל-UML, לשמור את תוכנו ולייצא ממנו תמונות של דיאגרמות הניתנות להצגה בוויקי. בנוסף לתרשימים יש לתאר בקצרה את הרכיבים השונים ולצרף דיון באשר לשיקולים העיקריים

בבחירתם. בשלב זה אין צורך לפרט עד לרמה של כל מחלקה ואובייקט אלא להתרכז בעיקריים שנראים לכם כרגע. בפרט דיאגרמה זו צריכה להיות מפורטת וטכנית יותר מאשר זו שהגשתם במשימה ראשונה של הצעת הפרויקט.

עקיבות: לכל דיאגרמה שבחרתם לשרטט (גם בסעיפים הבאים) פרטו על אלו דרישות ממפרט הדרישות היא עונה. בד"כ יש להראות שכל הדרישות מכוסות ע"י התיכון, אך אנו נסתפק בעיקריות שבהן.

### 3. מחלקות ותרשימי מבנה סטטי – CRC Cards & Class Diagrams

ספקו לפחות: 3-4 "כרטיסי CRC" למחלקות עיקריות שזיהיתם במוצר שלכם, לכל אחת רשימת האחריות והקשרים ביניהן. כמו כן דיאגרמת UML אחת המתארת את הקשרים העיקריים בין המחלקות. אין צורך לפרט ברמה של שדות וארגומנטים אלא בעיקר מתודות חיצוניות (=אחריות) של כל מחלקה. לצרכי הקורס – חובה שלפחות מחלקה אחת תהיה אחראית על לוגיקת חישוב שאינה טריוויאלית

### 4. תרשימי רצף התנהגותי – Sequence Diagrams

ספקו לפחות: שתי דיאגרמות רצף המציגות את האינטראקציה בין חלקים שונים של המערכת תוך כדי ביצוע פונקציונליות מסוימת.

### 5. שמירת נתונים - Persistence

תארו כיצד נשמרים נתונים של המערכת, אלו קבצים ייווצרו ומה יהיה המבנה שלהם (ייתכן שתצטרכו תרשים מחלקות נוסף). במקרה שאתם גם מתכננים בסיס נתונים רלאציוני, ספקו תכנון ראשוני של הטבלאות והקשרים ביניהן.

### 6. דרישות לא-פונקציונליות

תארו בכמה משפטים כיצד התיכון שהצעתם במסמך זה עונה על דרישות כאלו.

### 7. ניהול סיכונים

עכשיו כשהתמונה ברורה יותר אתם נדרשים לפתח את תכנית ניהול הסיכונים. הוסיפו בעמוד הפיתוח טבלה הכוללת לכל סיכון:

- הסבירות שהסיכון יתרחש (1-3)
- חומרת ההשפעה על הפרויקט (1-3)
- אלו צעדים בוצעו להנמכת הסיכון, בעיקר ברמת התיכון – יש לפרט במיוחד לגבי הסיכון העיקרי
- מה תעשו אם הסיכון יתממש

## 8. תוכנית בדיקות ותיעוד (ראשונית)

### תוכנית בדיקות



תארו אלו מאפיינים של המערכת מתוכננים להיבדק ומדוע זה מספיק. פרטו כיצד יעשו בדיקות אלו. תארו בדיקות יחידה (עבור רכיבים בודדים אך משמעותיים במערכת), בדיקות מערכתיות (אינטגרציה) ובדיקות שמישות (usability). כמו כן נסו לקשר את הבדיקות ככל האפשר לדרישות המופיעות במסמך ה-SRS. תארו כיצד תעקבו אחרי תקלות המתגלות תוך כדי שימוש ובדיקות.

הערה: מכיוון שנושא הבדיקות טרם נלמד, עשו כמיטב יכולתכם ותכננו לעדכן דף זו באיטרציה שבהמשך הפיתוח. הנה דוגמא אפשרית למבנה הדף:

1. תכנון בדיקות יחידה (לרכיבים משמעותיים)
- א. מהי הבדיקה (מטרה וכיסוי)
- ב. אופן פיתוח הבדיקה ועל ידי מי
- ג. תדירות שימוש בבדיקה
2. תכנון בדיקות מערכתיות
- א. כנ"ל
3. תכנון בדיקות שמישות \ חווית משתמש
4. כיצד תעקבו אחרי תקלות שייתגלו

### 6.1 תוכנית תיעוד

הגדירו את התיעוד המתוכנן להימסר עם המערכת, למשל מדריך למשתמש, הוראות התקנה, תפריטי עזרה וכדומה

# סיכום סקר תיכון תוכנה - SDS

## הסבר והוראות לסוקר:

סקר התיכון משמש כמה מטרות:

1. לוודא שהפרויקט נישל לפתור את הבעיה שלפניו בצורה סבירה ומתקדם לעבר ההצלחה. אמנם ייתכן שגם המנהלים של חברת סבב"ה יסקרו את המפרט וינסו לתת משוב, אך תמיד כדאי לשמוע דעות נוספות.
2. בהמשך פיתוח המוצר, בין שחרור גרסת הבטא לשחרור הסופי, תשמשו גם כבודקי אינטגרציה ומשתמשי קצה. הסקר הזה הוא הזדמנות להכיר את המוצר ולספק קלט משמעותי לקבוצת הפיתוח.
3. הזדמנות לחשיבה ביקורתית ונתינת משוב בונה (יכולות שתזדקקו להם בהמשך הדרך בקריירה של מהנדסי תוכנה).

## תוכן

שם הפרויקט הנסקר	
נציג טכני של הלקוח (הסוקר)	
מועד ומקום הפגישה	
שמות משתתפים	

## נקודות עיקריות שעלו במפגש

1.	
2.	

## סיכום קצר של הסוקר

כתוב כמה משפטים על מצב התיכון בפרויקט. בפרט התייחס למידת העמידה בארבעת כללי התיכון הפשוט שלמדנו (Beck)

---



---



---

## טבלת משימות הנובעות מהמפגש

נושא	פעולה נדרשת	באחריות

