**פרק 1:**

1. הנדסת תוכנה היא מכלול של תכניות מחשב ומסמכים נלווים, מוצרי תוכנה שיש לפתח ללקוח מסוים או עבור שוק כללי. בעצם ציות הנדסי שנוגע בדבר עם כל ההיבטים של הפקת התוכנה. מערכות התוכנה הן בלתי מוחשיות מה שאומר כי התוכנה היא מופשטת.

מדעי המחשב מתעסקים בתאוריית היסודות ואילו הנדסת התוכנה עוסקת בחלק המעשי של הפיתוח ואספקת תוכנות שימושיות.

הנדסת מערכות עוסקת בכל ההיבטים של פיתוח מערכות מבוססות מחשב כולל חומרה תוכנה והנדסת התהליך. והנדסת התוכנה היא חלק כללי מהנדסת התהליך.

1. **א'**. מערכות תוכנה בזמן אמת-חייבת להיות מפורטת לחלוטין לפני תחילת הפיתוח, הן קריטיות ודורשות מפרט מלא של אנליזה מערכות אלו כוללות \*מערכות מידול וסימולציה אשר פותחו ע"י מדענים ומהנדסי מודל תהליכים פיזיקליים ,הן מערכות מחשוב אינטנסיבי וכן התוכנה כוללת אינטראקציה עם חיישנים אשר אוספים מידע הנשלח לעיבוד .

**ב'**. מערכות מסחר-המפרט מהתוכנית בד"כ מפורטות יחד והארגון שלהם משתנה בהתאם לסוג המערכת , מערכות אלו כוללות \*יישומים עצמאיים שהם לרב על מחשב מקומי שכוללים פונקציונאליות וחיבור לאינטרנט .

**ג'**. מערכות בידור- אלו הן לרב מערכות משחקים וכדומה איכות האינטראקציה עם המשתמש היא המאפיין המבדיל החשוב ביותר של מערכות בידור מערכות אלו כוללות \*שירותי אינטרנט .

1. אנשים נוטים לחשוב שתוכנה זה תכניות מחשב .אך כאשר אנו מדברים על הנדסת תוכנה , זה לא רק התוכניות עצמן אלא גם מסמכים קשורים ונתוני תצורה שנדרשים כדי להפוך את התוכניות לפועלות כראוי. מערכות תוכנה לעיתים קרובות הן יותר מאשר תכניות אחת. אלא מספר של תכניות נפרדות וקבצי תצורה שמשמשים להקמת המערכת הכוללת. זה עשוי לכלול תיעוד מערכת שמתאר את מבנה המערכת ,שימוש במערכת , ואתרים להורדת המוצר.
2. כאשר אנו מדברים על תוכנה מקצועית אנו מדברים על איכות תוכנה ז"א , לוקחים בחשבון שהתוכנה משמשת ומשתנה ע"י אנשים אחרים מלבד המפתחים שלה.

איכות לפני הכול זה לא רק מעקב אחרי מה התוכנית עושה זה כולל \*התנהגות התוכנה בזמן שהיא מתבצעת \*מבנה התוכנית \*ארגון של תכניות המערכת \* ומסמכי תיעוד.

לדוגמא זמן לתגובה של התוכנה לשאילתת המשתמש ,יכולת ההבנה של קוד התוכנה.

מה שאומר כי התוכנה לא מפותחת עבור לקוחות בלבד מכיוון שמערכות אלו עונות על צרכים שונים מלבד דרישות לקוח.

1. **מוצרים גנריים** :מדובר במערכות עצמאיות המיוצרות ע"י ארגון לפיתוח ,ונמכרו בשוק הפיתוח ללקוחות שרצו אותם לדוגמא תוכנה עבור מחשבים כגון: מסדי נתונים ,כלי ניהול פרויקטים וכן יישומים שנקראו "אנכיים" שנועדו להשגת מטרה מוגדרת כגון: מערכות מידע ספרתיות מערכת להנהלת חשבונות וכו' ... לעומת זאת

**מוצרים מותאמים אישית**: מדובר במערכות שהוזמנו מראש ע"י לקוח מסוים .

קבלן התוכנה מפתח תוכנה ספציפית עבור אותו לקוח לדוגמא מערכות בקרה עבור מכשירים אלקטרונים ,מערכות בקרה אווירית וכו' ..

* הבדל חשוב בין סוגים אלו של תוכנה זה שבמוצרים גנריים הארגון אשר מפתח את התוכנה שולט בכל מפרטי התוכנה ואילו במוצרים מותאמים אישית מפרט בד"כ מפותח ובשליטת הארגון שקונה את התוכנה ומפתחי התוכנה חייבים לעבוד למפרט זה.

1. בתחילה האינטרנט הי בעיקר מאגר מידע והייתה לו השפעה מועטת על מערכות תוכנה ,בשנות ה-2000 האינטרנט התחיל לתפוס תאוצה ולהתפתח עוד ועוד ונתן יותר פונקציונאליות פירוש הדבר שמערכות מבוססות אינטרנט יכולות להתפתח ע"י האינטרנט הן לא צריכות ממשק ייעודי אלה הן מוצגות באמצעות דפדפן ,מצב זה הוביל לפיתוח של מגוון עצום של מוצרי מערכת ושירותים חדשניים והכל דרך האינטרנט, כמו כן גם פיתוח של דפדפני אינטרנט יכול להפעיל תוכניות קטנות ויכול לעבד גם נתונים מה שמוביל אתנו לא לכתוב את התוכנית על מחשבי המשתמשים אלא התוכניות נפרסות על שרת וזה עושה את זה יותר זול ומקל עלינו במצב של שינוי ושדרוג התוכנה זה גם הפחית עליות כמו פיתוח ממשק משתמש שהוא יקר במיוחד כתוצאה מכך בכל מקום בו אפשר לעשות זאת ,עסקים רבים עברו למערכות תוכנות מבוססות אינטרנט .

הופעתו של האינטרנט הוביל לשינוי מהותי באופן שבו תוכנות עסקיות מאורגנות לפני האינטרנט יישומים עסקיים היו בעיקר בגוון אחיד תכניות יחידים הרצים על מחשבים מקומיים ,התקשורת הייתה מקומית ובתוך הארגון . עכשיו התוכנה מופצת רבות לעיתים ברחבי העולם ,יישומים עסקיים לא מתוכנתים מאפס אבל מערבות רכיבים ותוכניות רבות נרחבות. יש שינוי קיצוני בארגון תוכנה ושינוי זה הוביל לשינוי הדרכים בהן מערכות תוכנה מבוססות אינטרנט מתבצעות. לדוגמא:

1. חוזר תוכנה- גישה אשר הפכה להיות הדומיננטית לבניית מערכות מבוססות אינטרנט ,כשאר בונים מערכות אלו אתה חושב על איך אתה יכול להרכיב את אותם רכיבים ממערכות קיימות.
2. ארגון דרישות- עכשיו זה מזוהה בד"כ וזה לא מעשי לציין את כל הדרישות עבור מערכות אלו מראש אלא מתפתחות ונמסרות באופן הדרגתי.
3. ממשקי משתמש- מוגבלים ע"י היכולות של דפדפני האינטרנט למרות שישנם טכנולוגיות אשר מספקים ממשקים עשירים לדפי אינטרנט. יישום ממשקים ע מערכות מבוססות אינטרנט בד"כ מתוכננים במיוחד.

* הרעיונות הבסיסיים של הנדסת תוכנה חלים גם על מערכות מבוססות אינטרנט ובאותו אותן.

1. 1כיק'ןועכו
2. מערכות קבוצות עיבוד- מערכות אלו נועדו לעסקים אשר עובדים בקבוצות גדולות בכדי לאפשר מספר רב של כניסות בשביל ליצור תפוקות מתאימות.
3. יישומים עצמאיים-מערכות אלו פועלות על המחשב המקומי כגון מחשב ביתי ,מערכות אלו כוללות את כל הפונקציות הדרושות ללא חיבור לרשת.
4. יישומים מבוססים~עיסקה אינטראקטיבית-מעפילים אותם המחשב מרוחק ונשלטים ע"י מחשבים פרטיים יישומים אלו כוללים יישומי אינטרנט כמו מסחר אלקטרוני.
5. מערכות בידור-אלה הם מערכות שנמצאות בעיקר לשימוש שיאי ושנועדו לבדר את המשתמש.
6. מערכות איסוף נתונים-אלה הם מערכות שאוספות נתונים מסביבתם באמצעות מערכות של חיישנים ושולחות את הנתונים למערכות אחרות לעיבוד נתונים.
7. הקוד האתי של מהנדסי תוכנה הוא התנהגות בדרך ישרה ואחראית ונדרש מהם להיות מכובדים כאנשי מקצוע ולשמור על החוק.
8. סודיות-מהנדסי תוכנה חייבים לכבד את הסודיות של המעסיקים או הלקוחות שלהם ללא שום קשר להסכם סודיות שנחתם, מכיוון שהם נחשפים למידע אישי ומסווג.
9. כשירות-מהנדסים לא צריכים לשקר על רמת הכשירות שלהם כדי להתקבל לעבודה שמחוץ למיומנויות שלהם.
10. זכויות וקניין רוחני-מהנדסים צריכים להיות מודעים לחוקים המקומיים המסדירים את השימוש הקניין הרוחני וכן צריכים להיות זהירים על מנת להבטיח כי הקניין הרוחני של הלקוחות והמעסיקים מוגן. לדוגמא: פטנטים וזכויות יוצרים .
11. שימוש לרעה במחשב-מהנדסי תוכנה לא צריכים להשתמש במיומנויות הטכניות שלהם לשימוש לרעה במחשבים של אנשים אחרים. לדוגמא: להשתלט על המחשב של המעסיק ,להפיץ וירוסים או תוכנות זדוניות.
12. תוכנה טובה צריכה לענות על דרישות הלקוח ולספק פונקציונאליות נדרשת, ביצועים קלים בפני המשתמש ,תוכנה אמינה , יעילה ושימושית.