קבוצה 10 – תרגיל בית 1, חלק תיאורטי

שאלה 1

בשאלה זו נדרשנו לתאר את ארבעת מאפייני ה-ACID כפי שנלמד בהרצאה. עבור כל מאפיין, ניתן דוגמה לאיך הוא יכול להיות מופר בהינתן ואין מנגנון אשר מבטיח את הקיום התמידי של אותו המאפיין:

1. אטומיות (Atomicity) - כל פעולת הטרנזקציה מתבצעת יחד או ששום דבר לא קורה (סוג של "הכל או כלום").   
     
   דוגמה טובה לכך היא משיכה של כסף מכספומט. נניח כי יובל מבקש למשוך 500 ש"ח. טרם המשיכה נכתב ל-DB כי נמשכו על ידי כרטיס האשראי שלו 500 ש"ח וכן משתנה רישום היתרה הכללית בכספומט ואז מתרחשת תקלה, החשמל בכספומט נופל.  
   במצב זה, במידה ולא מתקיים עקרון האטומיות, כרטיס האשראי שלו יחויב וכן לא תיהיה הלימה בין היתרה הכללית של יובל בפועל בבנק אל מול היתרה הרשומה בDB של הבנק, וזאת למרות שיובל לא קיבל (כלומר משך פיזית) את הכסף. כלומר, בבנק יופיע כי יש לו 500 שקלים פחות מהיתרה שאמורה להיות לו בפועל.
2. עקביות (Consistency) – כל מצב של המערכת חייב להיות מצב לוגי תקין. כלומר, לאורך כל פעילות המערכת בהכרח מתקיים מעבר ממצב לוגי תקין אחד למצב לוגי תקין אחר.  
     
   עבור אותה הדוגמה כמו מקודם, ובהתייחס ליתרת הכסף שנמצאת בכספומט עצמו, נניח כי רפאל מבקש למשוך סכום של 5,000 ש"ח, כאשר היתרה בתוך הכספומט הינה 4,000 ש"ח. הוצאה לפועל של טרנזקציה זו, ללא וידוא היתרה הזמינה, תביא למצב בו היתרה הרשומה בכספומט הינה מינוס 1,000 ₪. זהו מצב לוגי לא תקין, מכיוון שהכספומט לא יכול להיות ביתרה שלילית, ובנוסף יובל יקבל בפועל רק חלק מהסכום אותו ביקש למשוך.
3. בידוד (Isolation) - אין השפעה של טרנזקציה אחת על טרנזקציה אחרת המתבצעת באותו הזמן.  
     
   נתבונן ברכישה מקוונת של כרטיס למופע (בדומה לדוגמת רכישת כרטיס הטיסה עליה דיברנו בהרצאה). נניח כי יובל ורפאל מעוניינים לרכוש כרטיס, כל אחד ממחשבו האישי, באותו הזמן ממש. בנוסף נניח, כי לרוע המזל, נותר רק כרטיס פנוי אחד. ללא יישום עיקרון הבידוד, לשניהם יוצג כי קיים כרטיס אחרון והמערכת תאפשר לשניהם לרכוש אותו אף כי בפועל זמין רק כרטיס אחד.
4. עמידות (Durability) - השינויים כתוצאה מטרנזקציה שבוצעה (commit), ישמרו לעד (כלומר לא יאבדו).  
     
   נניח כי יותם רוכש מבעוד מועד כרטיס נסיעה לרכבת. תהליך הרכישה הושלם בהצלחה ויותם קיבל אישור עסקה. לצער כולם, בשל וירוס אשר תקף את עובדי הרכבת וגם את השרתים שלה, השרתים קרסו. לאחר הטיפול בוירוס וכאשר השרתים עלו, רשומת הרכישה נמחקה. כאשר יותם הגיע לתקף את כרטיס הרב קו שלו במכונה האוטומטית, מופיעה הודעת שגיאה כי לא קיים הסדר נסיעה בתוקף. בכך, למרות שהטרנזקציה בוצעה טרם קריסת השרתים, המידע נשמר ועיקרון העמידות הופר.

שאלה 2

תיאור שלreplication ו fragmentation יחד עם יתרונות וחסרונות של כל אחד -

1. replication - המערכת שומרת מספר עותקים של אותו ה-data, ומאחסנת את אותו הדאטה במיקומים פיזיים שונים.  
   יתרונות - גיבוי ושרידות, מקבול, זמינות, קרבה למיקום הגיאוגרפי בו נמצאים המשתמשים (הן בימד המהירות והן בכך שבכל אזור ניגשים למידע שרלוונטי אליהם - הפחתת המידע המועבר בפועל - "read").  
   חסרונות – מגדיל עלויות (אחסון וזיכרון), מוסיף צורך בפעולות זהות של עדכון ותחזוקהֿ במקומות שונים.
2. fragmentation - מתבצעת הפרדה (Partition) של המידע לחלקים. כל חלק נשמר במיקום אחר. ההפרדה יכולה להיעשות במימד השורות (Horizontal) או במימד העמודות (Vertical).  
   יתרונות – אפשרות מסוימת לפעולות במקביל, התאמת תוכן ה-data למיקום בו נדרש, צמצום נזק מטעויות (לדוג' רק חלק מה-data נמחק במקרה של טעות), זמינות חלק מה-data במקרה שאתר מסוים אינו זמין. עלויות ביחס ל-replication.  
   חסרונות - אין מקביליות בעבודה על כל ה-data, כאשר נדרש מידע רחב היקף (מכלל האתרים) פעולות join יקרות במקרה של vertical ו/או פעולות union במקרה של horizontal, עומס על הרשת. שרידות המידע חלקית ביחס ל-replication.

*תיאור של 2 דוגמאות עבור fragmentation אנכי ואופקי מהעולם האמיתי:*

1. Horizontal - לחברה בינלאומית המציעה שירותים לחברות פיננסיות גדולות, יש לקוחות במספר מוקדים, לדוג' ביפן ובגרמניה. כללי הרגולציה במדינות אלו מחייבים שמידע רגיש יעובד וישמר במדינת המוצא. כפועל יוצא החברה מבצעת Horizonal Fragmentation, כאשר הסכמה זהה עבור שתי המדינות - אולם פרטי לקוחות יפניים נשמרים באתר ביפן ופרטי לקוחות גרמניים באתר בגרמניה.
2. Vertical - חברת טכנולוגיה שומרת מידע רב על לקוחותיה. מידע זה משרת 2 קבוצות שונות בחברה - קבוצת המחקר וקבוצת ה-support, כאשר הראשונה מתעניינת במידע אודות מאפייני המשתמשים והשנייה בפרטי ההתקשרות עמם. בעבר המידע על הלקוחות נשמר בטבלה אחת. אולם, עם מעבר קבוצת ה-Support למדינה במזרח אירופה, בוצע Vertical Fragmentation, כך שהעמודות אודות פרטי ההתקשרות הועברו לאתר אחר במזרח אירופה.

שאלה 3

יתרונות

1. גמישות בכל הנוגע לסקיילינג (כיוון שהדרישות משתנות מעת לעת)
2. ביצועים (מאחורי הקלעים - מנגנוני דחיסה, cache, ו-recovery)
3. זמינות גבוהה

מגבלות

1. התשתית אינה מאפשר replications
2. ה-UI - מקשה על משתמשים להבין איך להשתמש בפלטפורמה בצורה מיטבית
3. אינדוקס - תמיכה הבאינדקס ראשי בלבד
4. \*תחזוקה - לא קיים מוצר מדף / SaaS לניהול ותחזוקה של ה-clusters