



שנקר – בי"ס גבוה להנדסה ועיצוב  
הפקולטה להנדסה  
להנדסת תוכנה המחלקה

# Class Line

## מערכת עזר לסטודנט

### פרויקט גמר

מגישים:  
אביתר פלג  
דביר מזרחי

מנחה:  
מר יחיאל קמחי

מוגש כחלק מהדרישות לקבלת תואר ראשון בוגר במדעים (B.Sc.)  
25/11/2013

## הכרת תודה

ברצוננו להודות למנחה הפרויקט, מר יחיאל קמחי על ליווי את פרויקט זה.

תודה מיוחדת לד"ר אמנון דקל, ראש המחלקה להנדסת תוכנה בשנקר, על ההכוונה, הייעוץ והאמונה בנושא הפרוייקט אשר תרמו לנו בהשגת מטרותינו.

תודה כללית למחלקת הנדסת התוכנה על צוותה ומשאביה.

תודה לחברינו על תמיכה נוספת:

- עומר גושן
- תומר אוחנה

## תקציר מנהלים

מערכת ה Class line היא מערכת עזר לקהל הרצאות בכלל ולסטודנט בפרט. המערכת פותחה במסגרת פרויקט גמר בשנתו הרביעית של תואר ראשון בהנדסת תוכנה במכללת שנקר.

מהות המערכת היא כלי לאירכוב וניהול מידע וחומרים פרונטאליים המובאים בפני קהל המורצים במהלך הרצאה, שיעור או ישיבת עבודה.

מערכת ה Class line באה לייעל ולפתח את משימת "מחברת הסטודנט" ולקחת אותו צעד אחד קדימה תוך התבססות על מספר הנחות יסוד עכשויות:

1. ברוב רובן של המצגות וההרצאות המתרחשות העת החדשה נעשה שימוש נרחב בעזרים ממוחשבים כגון:
  - א. מצגת שקפים.
  - ב. צפייה בתכנים רשתיים (Internet).
  - ג. שימוש בתוכנות מקצועיות.
2. רוב הסטודנטים משתמשים במחשב אישי על מנת לנהל ולשמור את החומרים המועברים אליהם.
3. חלק נרחב מהם משתמשים במחשב נישא במהלך ההרצאה עצמה על מנת להוריד אל הכתב את הנאמר.
4. קיימת מסורת ענפה של שיתוף מידע תוך שימוש במדיומים השונים לרבות:
  - א. דואר אלקטרוני.
  - ב. תיקיות משותפות.
  - ג. פורומים מקוונים.

מערכת ה Class line מאמצת הנחות יסוד ודפוסי התנהגות סטודנטאליים אלה ומעודדת אותן מתוך מחשבה כי השיטות והמנהגים המוצעים היום אינם עולים בקנה אחד עם ההתקדמות הטכנולוגית ואמצעי המחשוב הקיימים.

**תוכן העניינים**

<b>1. מבוא</b>	<b>6</b>
1.1. הקדמה	6
1.2. מוטיבציה	6
1.3. מילון מונחים ומוסכמות	7
1.3.1. שרת	7
1.3.2. לקוח	7
1.3.3. 'סשן'	7
1.3.4. גוגל כרום Google Chrome	7
1.3.5. תוסף כרום Chrome Extension	7
1.3.6. ארגז חול Sand Box	8
1.3.7. תמיכה בסביבות שונות Cross Platform	8
<b>2. מטרות הפרויקט</b>	<b>9</b>
2.1. כללי	9
2.1.1. לקוח	9
2.1.2. שרת	9
2.2. מטלות שהושגו	9
2.2.1. לקוח	9
2.2.2. שרת	10
2.2.3. מטרות אופציונאליות שלא מומשו	10
<b>3. תקציר מסמך תיכון</b>	<b>11</b>
3.1. ארכיטקטורת המערכת	11
3.2. תרשים פרישת מערכת-כללי	12
3.3. תרחיש מרכזי	13
3.3.1. המרצה	13
3.3.2. התלמיד	13
3.4. מסכים עיקריים	14
3.4.1. לקוח	14
<b>4. תקציר איפיון</b>	<b>16</b>
4.1. הקדמה לתקציר האיפיון	16
4.2. צד השרת	17
4.2.1. מחלקות	17
4.3. צד הלקוח	19
4.3.1. קונטרולים	19

21.....	<b>סקירת ספרות</b>	5.
21.....	פתרונות קיימים	5.1.
21.....	שיתוף מסך במסגרת כיתת לימוד	5.1.1.
21.....	שיתוף מסך בין מספר מחשבים עם שליטה מרחוק	5.1.2.
22.....	<b>סקירה טכנולוגית</b>	6.
22.....	שפות	6.1.
23.....	סביבות טכנולוגיות	6.2.
24.....	<b>יישום</b>	7.
24.....	מתודולוגיית הצילום – דיון חשוב.	7.1.
25.....	קובץ ה Class File.	7.2.
26.....	התנהגות המערכת	7.3.
26.....	צד השרת	7.3.1.
29.....	<b>מדריך למשתמש</b>	8.
29.....	צד שרת	8.1.
31.....	צד לקוח	8.2.
31.....	מצב ניווט הקורסים וההרצאות	8.2.1.
32.....	מצב פעולה (לכידת תצלומי מסך)	8.2.2.
33.....	<b>Project's Management-Diges</b>	9.

## 1. מבוא

### 1.1. הקדמה

תיק פרויקט זה מתאר את תהליך פיתוח מערכת "Class Line" בהנחיית מר יחיאל קמחי. פרויקט הגמר מוגש כחלק מהדרישות האקדמיות לקבלת תואר ראשון במדעים במכללת שנקר – רמת גן. פרויקט זה בוצע ע"י צמד סטודנטים אשר למדו יחדיו במהלך שנות התואר במכללה.

### 1.2. מוטיבציה

המוטיבציה לפיתוח המערכת הגיעה בטבעיות עבורנו כסטודנטים הנוטלים חלק במערך ההרצאות והמפגשים האקדמיים. ככאלה מצאנו עצמנו מתחבטים בשאלה האם שיטות סיכום החומר הנלמד העומדות לרשותנו ממלאות את הפוטנציאל הטכנולוגי בו אנו עוסקים כתלמידים בשנת 2012 בכלל וכסטודנטים להנדסת תוכנה בפרט. בשונה מהתקופה בה היינו תלמידי תיכון, חלל הלימוד של ימינו מכיל אביזרי חומרה רבים לקליטת החומר הנלמד – מהמצלמות בטלפונים החכמים שאנו נעזרים לצלם את הלוח, עד המחשבים הניידים שבהם אנו נעזרים להכין שיעורים ועבודות נדרשות. למרות ההיצע הטכנולוגי הנגיש והזמין לנו, עדיין הבחנו כי אנו ורוב שאר חברינו לספסל ממשיכים לנהל בשגרת השיעור מחברות נושאים, לצלם דפים במכונת הצילום בהפסקות ולמלא קלמרנו עטים בתחילת כל סמסטר.

אין ספק כי מסורת ארוכת שנים זו של סיכום ותיעוד החומר המועבר בהרצאה לא יכול ואלי גם לא צריך להעלם במחי יד ובכלל זוהי הרי בחירתו האישית של הסטודנט, אך האם כך הדבר באמת? האם בהינתן אלטרנטיבה ראוייה למשתתפי ההרצאה, לא יבחרו המשתתפים לזנוח את השיטות ה"ארכאיות" ולעבור לממשק נוח לסיכום החומר?

כשהתחלנו לבחון את האפשרויות העומדות בפנינו בנוגע לתחליפים לשיטות המקובלות גילינו למרבה ההפתעה שהם מועטות עד מאוד, למעשה, האפשרות היחידה הנוספת שעמדה לרשותנו הייתה החלפת הרישום שבמחברת להקלדה במעבד תמלילים. לא רע, אבל לא מספיק.

לא אמרנו נושא, והחלטנו להתבונן בצורה מעמיקה יותר במבנה השיעורים בהם לקחנו חלק והגענו למסקנה מרחיקת לכת! בעוד המורים והמרצים קיבלו לידיהם כלים אין ספור להעברת החומר הנלמד בדמות מצגות, מסמכי טקסט, אתרי תוכן ועוד, אנחנו נשארנו עם המחברת. אז נכון, ברוב המקרים הסטודנטים יוכלו להתחקות אחר החומר המוצג בפניהם במהלך השיעור ולהשיגו, הם יוכלו להוריד עותק של המצגת והמסמכים, וזה יהיה אפשרי לשמור את כתובות אתרי האינטרנט שבהם דפדפנו במהלך ההרצאה, ולשוב ולגלוש אל אותם עמודים (ולהניח שתוכנם לא השתנה עד אז). נוכל להתקין את תוכנת ה SQL שבה השתמש המרצה על מנת להמחיש לנו מבנה טבלאי, או את תוכנת ה BI דרכה למדנו על מתודולוגיות עסקיות שונות, נוכל לגשת אל קטעי הקוד שנכתבו בשיעור ולשחזרן ואפילו נוכל להסתכל במכשיר הטלפון החכם שלנו איתו צילמנו את גרף הפונקציה המסובך של המתרגלת לחדו"א.

כל זה טוב ויפה אבל דבר אחד הלך לאיבוד בתהליך וזה ההקשר, ההקשר בין התכנים הויזואליים אותם ראינו במהלך השיעור, ההקשר כפי שנהגה במוחו של זה שהעביר את החומר לנו ואולי התכוון כי כך יראה הדבר.

המסקנה המתקבלת היא ברורה, אנחנו התלמידים מפגרים כעשור לפחות מאחורי המורים בהתנהלותנו האקדמית, אנחנו משיגים המון חומר אבל נותנים לכוונת המרצה לחמוק מבעד אצבעותינו ובנוסף לכל, אנחנו מוותרים מרצון על יכולותינו המתקדמות להפעיל ציוד דיגיטלי ברגע שההרצאה מתחילה. יש צורך להכניס כלי חדש למערכה, כלי פשוט שכל תפקידו יהיה לשמור על קו ההרצאה מתחילתה ועד סופה בצורה הויזואלית והפשוטה ביותר, כלי שיאפשר לעשות אינטגרציה בין המדיומים ושיציג את התוכן הנשמר בו בצורה האינטואיטיבית ביותר.

### 1.3. מילון מונחים ומוסכמות

#### 1.3.1. שרת

המונח "שרת" הוא שם כללי למחשב הממלא פונקצייה מסוג "שרת" (Server). מחשבים מסוג אלה נפוצים בתקשורת בין מחשבים בכלל ובעולם האינטרנט בפרט. במערכת ה "Class Line" פונקציית השרת ממולאת ע"י מחשבו של המרצה. במחשב זה רצה תוכנית (רבות יפורט עליה בהמשך) שאחת מהפונקציות שהיא ממלאה היא את פונקציית השרת. אי לכך, מעתה והלאה, כאשר נדבר על ה "שרת", נדבר על מחשבו של המרצה (מחשב נישא או כיתתי) וזאת בניגוד לשרת האינטרנטי המקובל.

#### 1.3.2. לקוח

בדומה למוסכמת ה "שרת" (סעיף קודם), כאשר נתאר במהלך מסמך זה התנהגות / פונקציונאליות "לקוח", נדבר בעצם במחשבו האישי של הסטודנט (נישא או כיתתי). למעשה, כפי שעל השרת מותקנת תוכנית שחלק מתפקידה הינו לבצע את פונקציית השרת, על מחשב התלמיד תרוץ תוכנית שתפקידה הפונקציונאלי הינו "לקוח" (Client).

#### 1.3.3. 'סשן'

מאחר ולא נמצאה חלופה עברית עבור המונח "Session", נסביר אותו בקצרה ומעתה נשתמש בתעתיק לעברית של המילה Session.

המונח 'סשן' בא לתאר התקשרות רציפה וקצובת תוקף בין שני מחשבים או יותר. במצבים בהם מוגדר מחשב "שרת" ומחשב "לקוח". יוזם ה 'סשן' יהיה הלקוח, למעשה, הלקוח "יבקש" מהשרת ליצור עימו 'סשן' והשני, על פי הגדרות מקור שברשותו, יסכים / ידחה וינהל את רצף ההתקשרות ועד שאחד הצדדים יבקש לסיימה.

#### 1.3.4. גוגל כרום Google Chrome

גוגל כרום הוא דפדפן אינטרנט מתקדם מבית היוצר של גוגל.

כרום מאפשר התקנת והרצת 'תוספים' (Extensions) על גביו כמרחיבי יכולת.

נכון למועד כתיבת שורות אלו עומד נתח השוק של דפדפן כרום על למעלה מ 54 אחוזים דבר הממקם אותו בראש טבלת פופולאריות הדפדפנים העולמית ובהפרש ניכר מהמקום השני. חשוב לציין שדפדפן זה הוא Cross-Platform אשר עובד על מערכות הפעלה שונות, מה שעושה אותו אידיאלי כפלטפורמת הרצה – המונח יוסבר בהמשך.

#### 1.3.5. תוסף כרום Chrome Extension

תוסף לכרום הינה תוכנית אפליקטיבית (אפליקציה) על פי רוב בעלת נפח קטן עד בינוני אשר מופעלת ונתמכת ע"י הדפדפן עצמו. ישנן משפחות רבות לתוספי הכרום, החל מתוספים מעצימי חוויית גלישה, תוספים ייעודיים לאתר ספציפי (למשל תוסף מיוחד לאתר 'פייסבוק'), תוספים שהם משחקי מחשב ועוד סוגים רבים ומגוונים.

התוסף כתוב / ממומש בטכנולוגיה רשתית (Web technology) צד לקוח ויכול HTML, JavaScript & CSS. תוסף לכרום יכול להגיע מהתקנה פרטית או מחנות התוספים של גוגל.

### 1.3.6. ארגז חול Sand Box

ארגז חול הוא שם כללי לסביבת הרצה מבוקרת של תוכנית אחת בתוך אחרת. למשל במקרה שלנו, הדפדפן כרום מאפשר לתוספים שלו לרוץ בסביבת ארגז חול מאובטחת.

למה זה טוב? ובכן, כפי שנטען בסעיף הקודם, תוספי כרום מריצים קוד 'צד לקוח' ממשפחת JavaScript, קוד זה כמו כל שפת תכנות, הוא בעל פוטנציאל להיות מסוכן לסביבת המחשב. הקוד עשוי לרצות למחוק או לשנות קבצים במערכת הקבצים ו/או לקבל גישה למשאבי מערכת שאין במ צורך ממשי עבורו. אחד מתפקידי הדפדפן הוא לספק הגנה למשתמש בפני קוד זה המכונה קוד 'זדוני'.

לשם כך מציע כרום סביבה מאובטחת להרצת הקוד, הקוד אמנם רץ אך הוא מסוגל לבצע רק מה שהדפדפן מאפשר לו לבצע ונמנע ממנו לבצע כל מה שהדפדפן מגדיר כמסוכן.

### 1.3.7. תמיכה בסביבות שונות Cross Platform

המונח Cross Platform מבטא את יכולות תוכנית מסוימת לפעול תחת מטריית מערכות הפעלה שונות. כידוע, מערכת הפעלה היא סביבה סבוכה אשר מכילה הגבלים לא מועטים על כל מערכת המפעילה תחתיה. כיום, תוכנות רבות מציעות מספר גרסאות אשר כל אחת מהן בנויה לפעול תחת מטריית מערכת הפעלה מסוימת, האופציה לממש תוכנית אשר תפעל באופן מיטבי על כלל המערכות הינה מסובכת עד לא קיימת. בהקשרו של פרויקט זה, הוחלט לממש את תוכנית השרת בתאימות למערכת ההפעלה 'חלונות' הנפוצה במוסדות לימוד וחברות הייטק (למחשבים אישיים) ואת צד הלקוח כתוכנית מבוססת טכנולוגיית רשת (Web Application) וזאת כדי לנטרל עד כמה שניתן את התלות במערכת ההפעלה.



## 2. מטרות הפרויקט

### 2.1. כללי

פיתוח מערכת בעלת שתי צדדים: שרת ולקוח בתצורת אחד לרבים.  
תמיכה ביכולות:

#### 2.1.1. לקוח

- התקשרות ואחזקת 'סשן'.
- שליחת בקשות לשרת.
- עמידה בזמני שליחה וקבלה קצרים.
- קבלת נתונים מהשרת ואחסונם בהתאם.
- ניהול קובץ Class Line ע"פ פורמט קבוע מראש.
- 'דפדוף' מהיר ויציב.
- התנהלות מול מערכת הקבצים המקומית.
- קיום ממשק משתמש אינטואיטיבי ופשוט לתפעול.
- עבודה במסגרת ה LAN ללא התחברות לצד שלישי.
- יכולות עריכה של הנתונים.
- Cross Platform.
- התממשקות למדיומים נוספים (מצלמה, רשמקול).

#### 2.1.2. שרת

- יצירת 'סשן' (ראשי להרצאה).
- קבלת בקשות מלקוחות וטיפול\* בהן.
- \* - ניהול מסד נתונים 'קל משקל' אודות בקשות לקוחות.
- \* - השבת המידע הנדרש אל הלקוחות.
- ניהול סטטיסטיקת לקוחות עבור המרצה.
- יכולות כיבוי והדלקה מהירים.
- פשטות בתפעול ללא הפרעה למהלך ההרצאה התקין.
- תמיכה בריבוי לקוחות (עד 500).
- יצירת נתונים עבירים רשתית (נתונים בעלי נפח קל עד בינוני).

## 2.2. מטלות שהושגו

### 2.2.1. לקוח

- יצירה ואחזקת 'סשן' עם השרת.
- קיום ממשק משתמש גרפי רספונסיבי וקל לתפעול.
- יצירה ועריכת קבצים מקומיים.
- זמן 'קבלה-הצגה' קצר של המידע המתקבל מהשרת.
- Cross Platform – הושג ע"י פיתוח הלקוח כתוסף כרום.

**2.2.2. שרת**

- יציבות מערכת.
- קלות בתפעול.
- התמודדות טובה עם תעבורת רשת.
- ניהול מידע יעיל על מערכת קבצים מקומית.
- תמיכה במערכת הפעלה 'חלונות' – הנפוצה ביותר.

**2.2.3. מטרות אופציונאליות שלא מומשו****2.2.3.1. לקוח**

- יכולת עריכת קבצים מאוחרת.
- ניהול התקשרות מאובטחת.
- התממשקות עם מדיומים נוספים.

**2.2.3.2. שרת**

- תצוגת סטטיסטיקות משתמשים.
- מנשק משתמש טוב יותר.
- ניהול מנגנון אימות משתמשים Authentication.

### 3. תקציר מסמך תיכון

#### 3.1. ארכיטקטורת המערכת

שיקולים מרכזיים בתכנון המערכת:

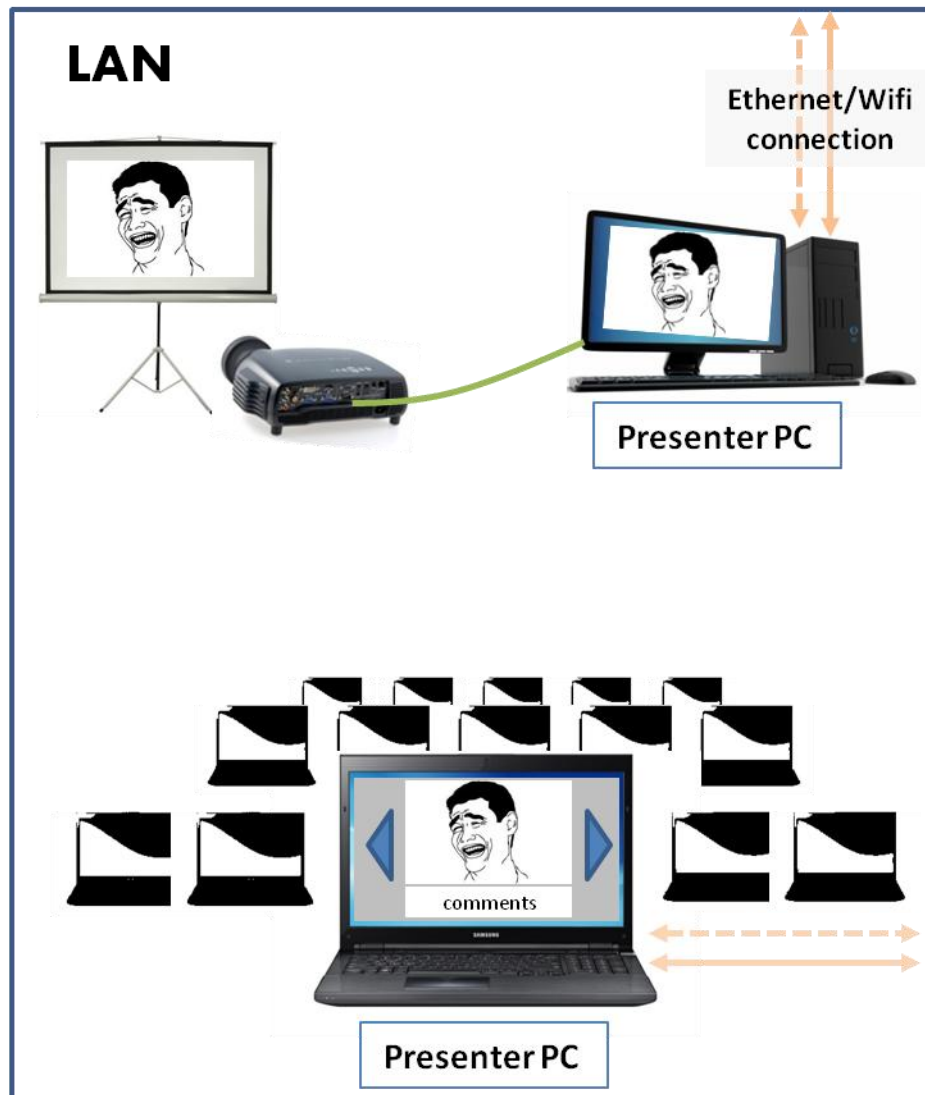
- על צד הלקוח לתמוך במגוון הגדול ביותר של מערכות (Cross Platform).
- המערכת תפעל במסגרת ה LAN ללא צורך בתקשורת אינטרנט חיצונית.
- צד השרת יפעל ב "חשאי", לא יכביד על מחשבו ולא יפריע למהלך ההרצאה התקין.
- המערכת תשמור על אנונימיות המשתמשים ולא תאגור מידע אודותיהם.

עוד מתחילתה, פורקה המערכת לשתי אבני בניין מרכזיות, צד הלקוח וצד השרת כאשר השלב הראשון שלאחר החלטת הפירוק היה הגדרת הממשק הפונקציונאלי ביניהם (API).

ההחלטה על הסתמכות על תקשורת מקומית בלבד (LAN) וללא שימוש בשרת אינטרנטי מרוחק נבעה מהרצון לשמור על המערכת זמינה לכל פורום באשר יהיה (גלוי או חסוי) וכן מתן הביטחון עבור הנוכחים, ובראשם המרצה, כי החומרים אותם מעביר לא יעשו דרכם ליעדים בלתי רצויים. בנוסף עמדה לפנינו השאיפה למערכת יציבה ככל הניתן לכל אורך ההרצאה, פרק זמן העומד בממוצע בין 20 דקות ל 180 דקות. השימוש בתקשורת אינטרנט חיצונית עשויה הייתה להוות גורם מסכל למטרה זו כיוון שזו נוטה לחוסר יציבות, לקריסות ולניתוקים.

בשלב זה של התכנון נדרשנו להתייחס לסוגיית ה Cross Platform של צד הלקוח. בסוגיה זו התחבטנו ארוכות, בעוד מערכת 'חלונות' היא הנפוצה ביותר, עדיין משתמשי 'אפל' ומערכות אחרות כגון 'אובונטו' זכו לרייטינג לא מבוטל. הנתיב הנבחר היה טכנולוגיית רשת (Web Technology), כלומר, גם שכבת הלוגיקה העסקית (Business Logic) וגם שכבת הממשק הגרפי GUI יתבססו על טכנולוגיית רשת צד לקוח (The JavaScript family). בסקר קצר שערכנו בקרב סטודנטים גילינו שאחוז המשתמשים בדפדפן כרום עמד על איזור ה 80% (בתחילת המסמך ציינו כי אחוז השימוש הכללי בדפדפן כרום עומד על 54% אך לסטטיסטיקה זו נכנסים גם משתמשים "מיושנים" ואירגונים מסורתיים). כל אלו הובילו אותנו להחלטה לממש את צד הלקוח כאפליקציה מסוג "תוסף כרום". בפשטות, השימוש בפלטפורמת תוספי כרום (Chrome Extensions) תאפשר למשתמש להשיג תפעול מלא של פונקציונאליות המערכת ללא כל תלות המערכת ההפעלה איתה הוא עובד ובתנאי שדפדפן כרום מותקן עליה.

### 3.2. תרשים פרישת מערכת-כללי



### 3.3. תרחיש מרכזי

#### 3.3.1. המרצה

השעה היא 8:02 בבוקר יום שלישי והרצאה תחת הכותרת "תיאוריית הקורלציה שבין תת-הישגיות לתהילה" של ד"ר פנחס פנחסוביץ' החלה. חדר 405 שבו התקיימה היה הומה וההתרגשות הייתה ניכרת שכן הרצאותיו של ד"ר פנחסוביץ' ידועות היו באופיין השערוניי. יש שטוענים ששיטותיו הן רדיקליות ואף מסוכנות אך תמימות הן הדעות בדבר אפקטיותן. ד"ר פנחסוביץ' היה ידוע בתזזית מיצגיו ובחדות לשונו וכן כמרצה היחיד בפקולטה המתנייד בין בנייני הקמפוס המנומנם בעזרת מערכת רקטית הקשורה אל גבו.

עם תחילת ההרצאה ולאחר שברך את היושבים בכיתה בברכת "מי ייתן והכח יהיה עימכם" המסורתית שלו, הפעיל הד"ר את מחשבו, הפעיל את בלחיצה את תוכנת ה Class Line ומיד הופיעה בצד מסכו המוקרן שקופית קטנה ובה פרטי ההתחברות למערכת. רחש תקתוקי המקלדת הפר את הדממה וההרצאה החלה. עם הזמן למד פנחס לזהות את מטרי לחיצות המקשים עם הופעת אחת משקופיותיו השערונייות ולבו נמלא בגאווה.

מתום ההרצאה, לחץ המרצה על כפתור סגירת השיעור הווירטואלי שבתוכנת ה Class Line, טמן מחשבו בתיקו, חגר חגורותיו ובסער נורא שייט מבעד לחלון הכיתה הפתוח לקול תשואות הקהל.

#### 3.3.2. התלמיד

אלעד אלעדוב נהג להגיע לשיעורים באיחור, היום הוא לא יוצא מן הכלל ובשעה 8:05 התגנב בחטף דרך הדלת האחורית אל חדר 405 שבבניין הפקולטה למדעים אבסטרקטיים וזאת כדי לקחת חלק בהרצאתו השנוייה במחלוקת של "הפרופסור המכונף", או לפחות כך כינוהו כולם. למען האמת, אלעד מעולם לא גילה עניין רב בנושא 'הקשר שבין לא לעשות כלום ללהיות מפורסם', אבל הילה, סטודנטית במחלקה שפעם בסמסטר הקודם ביקשה ממנו להחזיק לה את התיק לשני רגעים, עוררה בו תחושות סקרנות בלתי מוסברים לנושא.

משהתיישב בכיסא והפעיל את אפליקציית ה Class Line שלו, שם אלעד לב כי שקופית פרטי ההתחברות כבר נעלמה ממסך המרצה וליבו נמתק עליו. "מה הסיסמה?" שאל אלעד בלחש את הילה שבמקורות מוטלת בספק ישבה שורה אחת לפניו. הילה הסתובבה, חייכה והשיבה "123456".

"היום הולך להיות יום טוב" חשב בליבו בטרם הזין אלעד את סיסמת ההתחברות באפליקציה. לו מצליח היה להתרכז בנאמר בה, היה בטוח שהיא גם מעניינת, אך למשך כל אורך ההרצאה נעוצות היו עיניו באחורי שורה המשתפל של הילה.

הילה, שגילתה עניין גדול בנושא ההרצאה הניע עיניה לפרקים בין בימת החדר והמופיע במסך לבין מחשבה האישי ומעת לעת לחצה על כפתור ה "לכידה" שבפינת אפליקציית ה Class Line שלה. משעשתה זאת וכבדרך קסם, הופיע במרכז האפליקציה העתק מדויק של המופיע במסך של המרצה ובתיבת הטקסט סיכמה בנחישות ובזריזות את דבריו של ד"ר פנחסוביץ' שהיה ללא ספק המרצה האהוב עליה באותה תקופה.

"אני יכול לקבל את ה Class file שלך מההרצאה של היום בבוקר? פשוט, המחשב שלי עשה לי בעיות וממש חשוב לי שיהיה לי סיכום טוב שלה.", סינן אלעד לעצמו פעם אחר פעם בקול נשבר והולך בעודו ממתין לתורו בקפיטריה. "בטח! אני אשלח לך כשאגיע הביתה." נשמע קול מתוק ומאיים מאחוריו, הייתה זו הילה, אלוהים יודע כמה זמן עמדה שם אבל זה כבר לא משנה, היום באמת הולך להיות יום טוב.!

### 3.4. מסכים עיקריים

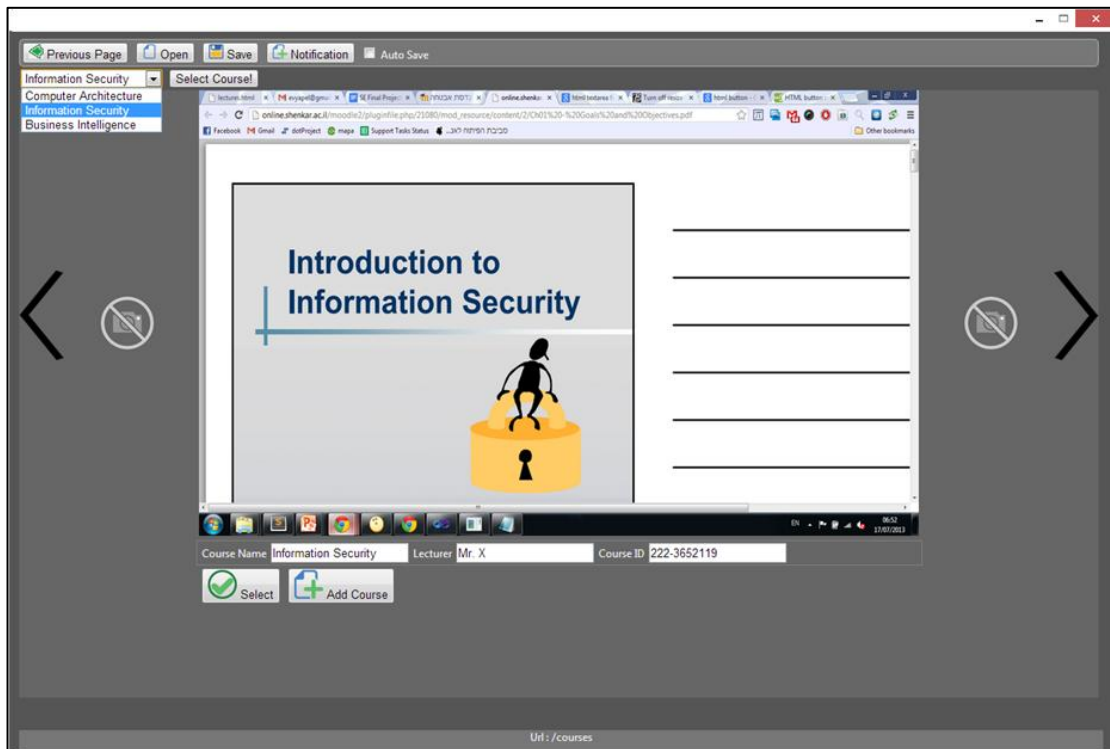
#### 3.4.1. לקוח

##### 3.4.1.1. מסך הפתיחה של האפליקציה.

מסך הפתיחה מאפשר לסטודנט לנווט ולרוחבו של הקובץ ובכך להתחיל, בין ובתוך שתי הרמות העליונות:

- רמת הסמסטר – דפדוף בין קורסים.
- רמת הקורסים – דפדוף בין הרצאות.

את הרמה עצמה ניתן לערוך ע"י הוספה או מחיקה של יחידות מידע (ברמת הסמסטר, הוספה / מחיקה של קורסים וברמת הקורס, הוספה / מחיקה של הרצאות).

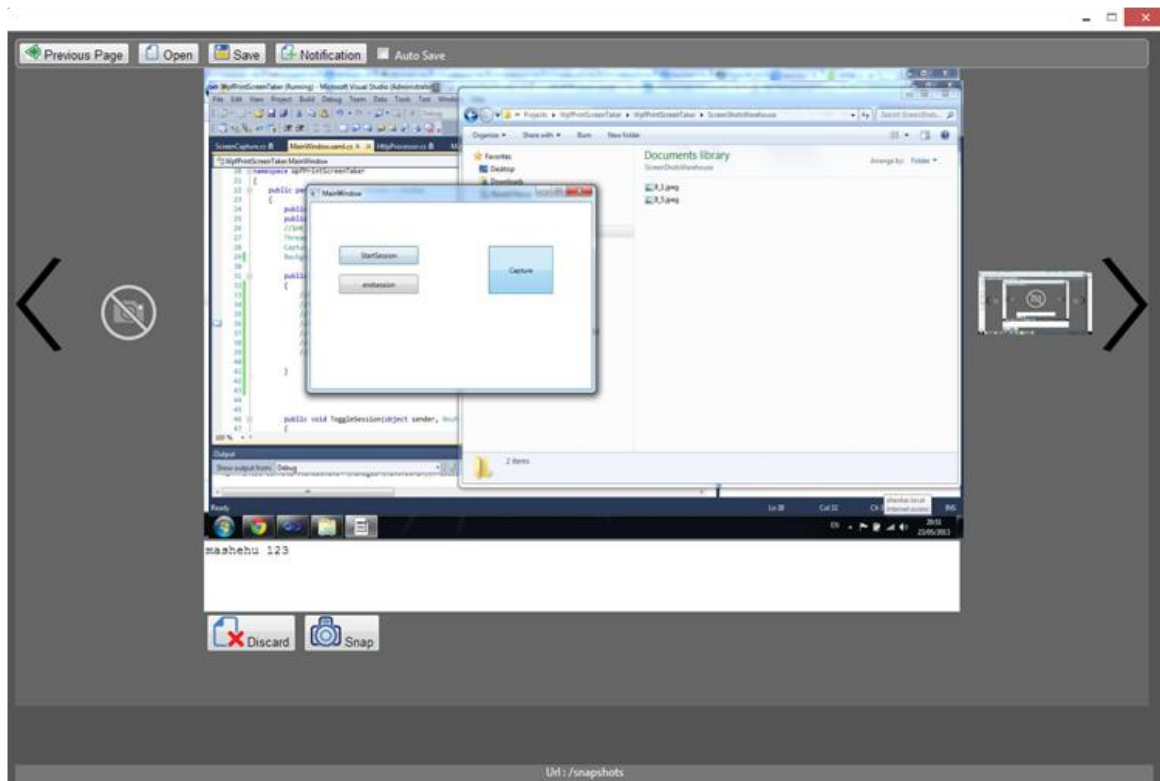


מסך הפתיחה של האפליקציה

### 3.4.1.2. מסך ניהול ההרצאה.

מסך ניהול ההרצאה הוא המסך העיקרי של האפליקציה, המסך תומך בפונקציונליות הבאה:

- שליחת בקשות צילום מסך מהשרת והצגתן.
- הוספה/עריכת הערות Cliff notes עבור הצילום הספציפי.



מסך ניהול ההרצאה

## 4. תקציר איפיון

### 4.1. הקדמה לתקציר האיפיון

מערכת ה Class Line הינה מערכת שפיתוחה מאופיין במימוש פשוט, 'קל משקל' וכזה הנשען על פלטפורמות קיימות כגון מערכת חלונות ודפדפן כרום. במהלך הפיתוח הושם הדגש על ההיבטים הבאים:

- ביצועי מערכת (מערכת ה Class Line).
- ניצול נמוך ככל הניתן של משאבי מערכת (מערכת ההפעלה המקומית).
- פעילות צד שרת בחתימה נמוכה.
- העדר תלות בצד שלישי (אינטרנט).

מערכת ה Class Line אינה מתיימרת להציג חידוש טכנולוגי ואינה משתמשת בהכרח בחזית טכנולוגיית פיתוח התוכנה. המערכת אינה מפעילה מערכת רגשים פריפריאלית ואינה מנהלת מסד נתונים מורכב. מערכת ה Class Line מציעה פתרון פשוט לבעיה אמיתית, פתרון שלדעתנו, צריך היה להיות בנמצא כבר הרבה זמן. לדעתנו, הסיבה לכך שלא היה טמונה בצורך הבלתי מוסבר של מפתחי תוכנה רבים ל "סיבוך יתר", לכניעה למוסכמות וקונבנציות שמקורן בתאגידי התוכנה ואולי אף חדלות מעוף.

לאחר שאמרנו את כל אלה, בתקציר מסמך האיפיון נביא את מימוש המערכת, את הדינאמיקה שבין הישויות השונות ואת זרימת האירועים במערכת.



## 4.2. צד השרת

### 4.2.1. מחלקות

- **מנשק משתמש גרפי.**  
מנשק המשתמש הגרפי המיועד לצד השרת הינו פשוט והוא מציע למרצה שליטה על הבאים:
  - יצירה וסיום 'סשן'.
  - קביעת תדירות צילום המסך.
  - חסימה / איפשור קבלת בקשות צילום.
  - הזנת פרטי הרצאה גלובליים.
  - הצגת סטטיסטיקת משתמשים (לא ממומש בגרסה זו).
- **שירות הקריאות (Restful Service)**  
שירות הקריאות מהווה את השער שבין קריאות המשתמשים ללוגיקת השרת. השירות למעשה פותח 'צוהר' (Port) על גבי המחשב המארח תחת כתובת IP ומאזין לבקשות הלקוחות. שירות הקריאות תומך במנשק הפונקציונאלי הבא (API):
  - **לחיצת היד המשולשת** (Three way hand-shake) – פונקציה זו מהווה את תחילת תהליך ההיקשרות שבין הלקוח לשרת. השרת מאזין לבקשות התחברות מצד לקוחות, כשאחת מגיעה הוא רושם אותה, מבצע פעולת אימות פשוטה, מיצר מזהה ייחודי ללקוח ומשיב לו אותו ולבסוף מקבל מהלקוח אישור על קבלת המזהה הייחודי.
  - **טיפול בבקשות צילום** (Snapshot application) – פונקציה זו מהווה את עיקר ההתכתבות שבין הלקוח לשרת. כאשר הלקוח מעוניין לקבל העתק של תצלומו מסך המרצה הוא לוחץ על כפתור ייעודי במנשק הלקוח, הפקודה נשלחת לשרת בתוספת חותמת זמן, שירות הקריאות מקבל את הבקשה ומפעיל את שאר קומפוננטות המערכת הרלוונטיות. לבסוף, מחזיר השרת ללקוח את התשובה בתצורת מידע טקסטואלי מקודד בבסיס 64 ביט המייצג תמונה.
  - **בקשת ניתוק** (Disconnection request) – בהנחה כי הלקוח מבצע תהליך התנתקות סדור, תשלח הפקודה מהלקוח להוציא אותו ממאגר הלקוחות הרשומים לאותו 'סשן'. לאחר ביצוע הפעולה, מחזיר השרת ללקוח אישור בדבר הניתוק. יש לציין כי פעולה דומה מתבצעת עבור כל הלקוחות בעת סיום ה'סשן' ע"י השרת.
- **המתזמן הראשי** (Sync Master) – המתזמן הראשי היא המחלקה הדומיננטית בתוכנת השרת ודרכה מופעלות כל שאר זרועותיה. המתזמן הראשי מפעיל שעון במופע 'סינגלטון' שהוא שעון המערכת וממנו נגזרות כל הפונקציות המתוזמנות לרבות ישות ה'צלם'.
- **הצלם** (Screenshot maker) – מחלקת הצלם היא מחלקה לה מופעים רבים במהלך ריצת תוכנית השרת כשהיא רצה תחת מעטה של תהליכון (Thread) שמנפיקו הוא המתזמן הראשי. התהליכון 'נורה' לראשונה בתחילת ריצת התוכנית ובו לולאה פשוטה שמחכה לאיתות מהמתזמן הראשי לבצע צילום. אחת לזמן קבוע מראש (3 שניות בברירת מחדל), יאותת המתזמן הראשי לתהליכון הצלם לבצע צילום מסך וזה יבצעו. ביצוע הצילום ושמירתו למערכת הקבצים נעשה תוך שימוש בפונקציות 'חלונות' סטנדרטיות.

- **המחלקה המשותפת (Shared)** – המחלקה המשותפת, כשמה כן היא, מחלקה ובה 'חברי' מידע (Data members) במופע 'סטטי' שתפקידם הוא להחזיק את המידע המשותף לכלל המערכת לרבות:
  - דגל אינדיקציית הצילום.
  - דגל אינדיקציית ה 'סשן' הפעיל.
  - נתיבים מקומיים שימושיים (למשל נתיב שמירת התמונות).
  - חותמת הזמן הפעילה כעת (חותמת הזמן הפעילה נקבעת ע"י המתזמן ונשמרת במחלקה המשותפת).
  - טבלת המיפוי הכללית (Indexed mapping table) טבלה זו שומרת על הקשר בין:
    - חותמת הזמן.
    - הנתיב הפיזי לתמונה הרלוונטית.
- **מנהל מערכת הקבצים (File system handler)** – מחלקה זו, בדומה למחלקת הצלם מופעלת ע"י המתזמן הראשי בתצורת תהליכון ותפקידיה העיקריים כוללים את:
  - בדיקות תקניות ואימות נתיבים וקבצים שוטפת.
  - מחיקת קבצים חסרי דרישה (תמונות שצולמו ללא דורש).
  - מחיקת קבצים מתום ריצה.
- **מחלקת "פרטי הסשן" (Session Details)** – מכילה נתונים אודות ה 'סשן' כגון:
  - מזהה ייחודי.
  - שם המרצה.
  - שם ההרצאה.
  - הגבלת משתמשים.
  - אינטרוול הצילום.
  - זמני התחלה וסיום.
  -
- **מחלקת רשומת הבקשה (Request Detail)** – משמשת כמחלקת עזר למתזמן הראשי אשר מופעיה נוצרים עבור הבקשות השונות ומוחזקים במבנה נתונים. היא ממומשת 'כמחלקה בת' של 'המחלקה המשותפת' והיא מכילה נתונים אודות הבקשה הספציפית כגון:
  - מזהה ייחודי.
  - חותמת הזמן.
  - שם קובץ מלא לתמונה.
  - דגל 'התקיימה בקשה'.
  - מונה בקשות.

### 4.3. צד הלקוח

כדי להבין את מבנה צד הלקוח אנו נסביר בקצרה ממה הוא מורכב – מכיוון שמימוש לוגיקת צד הלקוח מורכב רובו ככולו ב JavaScript על פלטפורמת אנגולר, המקבילה הרוחנית של המחלקות המוכרות מ NET. הינה ההפרדה הלוגית לקונטרולרים. אנו עשינו הפרדה מקובלת לקבצי Javascript אשר מכילים כל אחד פונקציונאליות מסויימת ייחודית לאותו הקובץ.

#### 4.3.1. קונטרולרים

- **app.js**

קונטרולר האפליקציה אחראי על הניווט של העמודים השונים. בו אנחנו ניעזר כדי לדעת היכן אנחנו נמצאים, ולכן לעשות redirect כאשר האופציה המתאימה נבחרה.

- **background.js**

זהו הקובץ אשר אחראי על העלאת ה Extension ויצירת המנשק הגרפי. פה אנו נמצא את כל הגדרות התצוגה של חלון האפליקציה. חלון האפליקציה מוגדר כ Singleton – ניתן לייצר רק instance אחד מה Extension.

- **main.js**

זהו הקונטרולר הראשי אשר מרכז את כלל הלוגיקה של האפליקציה ברמה עילית. בו אנו נשתמש לקרוא לפונקציות השונות כגון שמירת הנתונים, מעבר אחורה לדף הקודם, פתיחה של קובץ, שמירה של קובץ חדש ועדכון מיקום האפליקציה. כל פונקציונאליות אחרת שתפותח לאפליקציה תיקרא מהקונטרולר הזה.

- **data-service.js**

זהו "המוח" של האפליקציה. בקובץ זה אנו נמצא את המימוש של הפונקציות השונות – מטיפול בשגיאות, טעינה של המידע לקריאה ב UI, שמירה אוטומטית ושמירה ידנית ועוד. חשוב להבין שאפליקציה זו הינה כולה מורכבת מ Data Binding אשר חוסך לנו פונקציות מניפולציה על הקלט של המשתמש – האפליקציה מעדכנת את הקלט ואת ה UI בצורה אוטומטית.

- **courses-controller.js**

הקונטרולר אשר אחראי על חלון הקורסים המוצגים באפליקציה. קונטרולר זה אחראי על ניווט ה UI בתוך חלון הקורסים ומה שמטפל בכל הלוגיקה של הצגת הקורסים השמורים (הקודם, הנוכחי והבא) בקובץ ה Class Line. כמו כן קונטרולר זה מטפל גם בלוגיקה של קליטה של קורס חדש, הכנסת הנתונים לתוך מאגר הקורסים (אשר אותו שומרים במידת הצורך לקובץ).

- **lectures-controller.js**

הקונטרולר אשר אחראי על חלון ההרצאות עבור קורס מסויים. קונטרולר זה אחראי על ניווט ה UI בתוך חלון ההרצאות ומה שמטפל בכל הלוגיקה של הצגת ההרצאות השמורות (הקודמת, הנוכחית והבאה) בקובץ ה Class Line תחת קורס מסויים. כמו כן קונטרולר זה מטפל גם בלוגיקה של קליטה של הרצאה חדשה, הכנסת הנתונים לתוך מאגר ההרצאות בתוך הקורס (אשר אותו שומרים במידת הצורך לקובץ).

**• snapshots-controller.js**

הקונטרולר אשר אחראי על חלון הצילומי מסך (Snapshots) עבור הרצאה מסויימת. קונטרולר זה אחראי על ניווט ה UI בתוך חלון צילומי המסך ומה שמטפל בכל הלוגיקה של הצגתם (הקודם, הנוכחי והבא) בקובץ ה Class Line תחת הרצאה מסויימת. כמו כן בקונטרולר זה אנו מבצעים את קריאת ה Rest עבור צד השרת כדי ללכוד צילום מסך בצורה מרוחקת (מה שבעצם מטפל ב Capture של צילום המסך מצד השרת). כמו כן הקונטרולר מכיל לוגיקה אשר מוציאה Notification window על מסך התלמיד עבור כל צילום מסך אשר הצליח.

## 5. סקירת ספרות

### 5.1. פתרונות קיימים

אחד משלבי ניתוח התכנות הפרוייקט הינו סריקה אחר פתרונות קיימים עם דגש על פתרונות דומים. לאחר סקירה ארוכה הגענו למסקנה כי קיימים בשוק פתרונות רבים המציעים יכולות דומים ל ClassLine ובתחומים מסוימים אף טובים ממנה אך לא מצאנו אף מערכת המתמקדת בניהול חומרי ההרצאה כ ClassLine ושמציעה את הקלות, הפשטות והפונקציונאליות הספציפית בה המערכת מתמקדת.

#### 5.1.1. שיתוף מסך במסגרת כיתת לימוד

כיום קיימות מספר תוכנות המאפשרות שיתוף מסך בין מחשב מרכזי (מחשב המרצה) ובין מחשבי התלמידים. השיתוף נעשה על בסיס הזרמת התמונה (Streaming). בנוסף, המרצה יכול לעיתים לקבל אל מסכו שיתוף של מסכי התלמידים לבחירתו.

דוגמאות:

- Netop - Classroom Management



- Join.me



#### 5.1.2. שיתוף מסך בין מספר מחשבים עם שליטה מרחוק

ישנן מערכות המאפשרות שליטה מרחוק על המחשב אליהן הן מחוברות. שליטה זו באה לידי ביטוי בשליטה בעכבר ובמקלדת המחשב המארח. בנוסף, ישנן אופציות הקלטת המתרחש.

דוגמאות:

- TeamViewer



- Oneeko



## 6. סקירה טכנולוגית

### 6.1. שפות



#### 6.1.1. C#

שפת תכנות מונחית עצמים מתקדמת ממשפחת C. השפה מציעה מגוון אינסופי של ספריות לכל מטרה וביניהן ממשקי משתמש גרפיים, תכנות מקבילי, עבודה עם תהליכונים, תכנות צד שרת ועוד..



#### 6.1.2. JavaScript

שפת תכנות צד לקוח. את הקוד הכתוב מריץ הדפדפן והשפה מאפשרת שליטה בפקדי ה HTML, וולידציות טפסים מקוונים, אנימציות, תכנות עיצוב CSS והתנהגות מסמך אינטרנטי כללית.



#### 6.1.3. JQuery

Jquery הוא הרחבה של ספת ה Javascript הותיקה. JQuery עוטף את Javascript, מעניק לה סינטקס חדש, יעיל ואינטואיטיבי יותר, מוסיף יכולות תיכנות מונחי עצמים ומספק ספריות משתמש חדשות ובעלות יכולות מתקדמות.



#### 6.1.4. AngularJS

העצמה נוספת לשפת ה Javascript מבית גוגל. מעטפת זו מאפשרת כתיבה קריאה, יעילה וחזקה יותר ומתמקדת באלמנטים של HTML 5. אנגולר מתמחה בדפי אינטרנט דינמיים.



#### 6.1.5. HTML 5

הדור האחרון של שפת התגיות HTML. HTML5 מספקת תגיות חדשות, חכמות יותר הנותנות מענה אינטואיטיבי לצרכי תכנות האתרים המתקדמים. כמו כן, HTML5 מספקת תכונות (Attributes) נוספים לתגיות הבסיסיות המוכרות לנו מגרסאות ה HTML המוקדמות יותר.



#### 6.1.6. CSS 3

הדור האחרון של שפת עיצוב דפי האינטרנט CSS3. CSS3 מספקת יותר תכונות עיצוב וכן יכולות עיצוב דפי אינטרנט תגובתיים (Responsive design) עבור פלטפורמות המובייל השונות.

## 6.2. סביבות טכנולוגיות



### 6.2.1 Visual Studio 2010

המהדר המפורסם מבית מיקרוסופט מציע תמיכה בשפות שונות, שימש בפרוייקט זה לכתובת קוד צד השרת.



### 6.2.2 SublimeText3

עורך טקסט קוד פתוח מתקדם בעל יכולות תמיכה ויזואלית והשלמת מילים במגוון עצום של שפות וזאת ע"י תמיכה בחבילות הרחבה חנימיות.

## 7. יישום

מאחר ותיאור מפורט של מבנה המערכת על מחלקותיה הובא בפרק 4 ("תקציר האיפיון"), בחרנו בפרק זה להתעמק בקונפליקטים מרכזיים, מתודולוגיות פיתוח והתנהגות המערכת.

### 7.1 מתודולוגיית הצילום – דיון חשוב.

עוד מתחילת התהליך עלתה סוגיה בעלת חשיבות מכרעת לנוגעת לשיטת הפעלת הצלם. נתיב הפיתוח הראשוני בו הלכנו בהיבט זה דיבר על צלם 'פסיבי' שבסיס פעולתו יהיה המתנה שקטה בכוננות קריאה, כלומר הצלם יהיה רדום ובעת שתתקבל בקשה מהלקוח, יוזנק 'אירוע' (Event) לאוויר, הצלם יתעורר מתרדמתו, יצלם תמונת מסך, ישמרה, יכניסה לטבלת המיפוי וחוזר לישון. פעולה בשיטה זו הציפה יתרונות וחסרונות מספר, היתרון המשמעותי ביותר היה חיסכון בפעולות צילום וזאת מתוך מחשבה שבמהלך הרצאה תתכנה דקות ארוכות בהן לא יתעדכן דבר במסך המרצה (למשל בעת הסבר מילולי / דיון כיתתי) כך שבזמן זה, מיותר יהיה לצלם את המסך (פעולה הדורשת משאבי מערכת הן מבחינת מעבד והן מבחינת התקן האחסון). אך לשיטה זו היה גם חיסרון אחד משמעותי והוא 'מה יקרה באותם רגעים בהן תצוגת מסך המרצה כן מתחלף ובמוצג בה חפצים לקוחות רבים באותה העת'.

האם נכון יהיה להעביר את שרביט הניצוח על פעולה כל כך יוקרתית (צילום המסך) לידי הלקוחות?

מה יקרה אם תתקבלנה עשרות בקשות בשניה?

התשובה הייתה ברורה, המערכת תוצף בבקשות ומחשב המרצה יקרוס!

אם כן מהו הפתרון? בדרך אל הפתרון הצבנו מוסכמה והיא שברובן המוחלט של ההרצאות (לפחות כאלה בהן נכחנו), התוכן המוצג על המסך אינו מתחלף בתדירות הגבוהה מאחת בכל כמה שניות. מוסכמה זו עודדה אותנו לשנות גישה ולבוא עם מנגנון צילום אקטיבי אשר במקום 'לשבת בשקט' ולצלם כאשר נקרא לדגל בלבד, ינהל שגרת צילומים אוטונומית בקבוע זמן ידוע מראש (בפועל המערכת אותחלה לברירת מחדל של צילום כל 3 שניות). לפתרון זה הצטרף מנגנון 'ניקוי' חיוני (מעין Garbage collector) אשר אחת לכמה זמן יבצעה סריקה ומחוק את קבצי התמונה שלא נעשה בהם שימוש וזאת ע"י הוספת סמן מיוחד בטבלת המיפוי המודיע האם לתמונה מסויימת התקבלה בקשה או לא. פרק זמן הניקוי יהיה רחב מספיק על מנת לאפשר לשרת להעלות את התמונות להן הגיעה בקשה וקצר דיו על מנת לא לאפשר לתקיית התמונות המקומית לגדול למימדי ענק.

בנוסף, הוחלט, כהכנה למודול סטטיסטיקת המשתמשים, להשאיר את התמונות להן היה ביקוש על קנ ולספק למרצה פונקציית מחיקה ייעודית שתפקידה יהיה לנקות את התקנייה בגמר השימוש.

על מנת לאפשר אמיתות הבקשה, הונהג מנגנון 'נירמול הבקשה' שתפקידו להכריע לאיזה מחותמת הזמן (בעלת מרווח ה 3 שניות) שייכת הבקשה.



## 7.2. קובץ ה Class File.

קובץ הנתונים של אפליקציית Class Line נקרא בפשטות "Class File".

קובץ ה Class File הוא למעשה קובץ נתונים טקסטואלי בפורמט JSON. הקובץ בנוי במבנה היררכי בעל 3 רמות עומק:

- סמסטר
- קורס
- הרצאה
- + שקופית

מבנה הקובץ שומר על אחדות המבנה לכל אורכו ובכך מאפשר לאפליקציית הלקוח לנווט לעומקו (בין רמות העומק) וכן לרחבו (בין יחידות המידע).

הקובץ מכיל בתוכו מספר סוגי מידע:

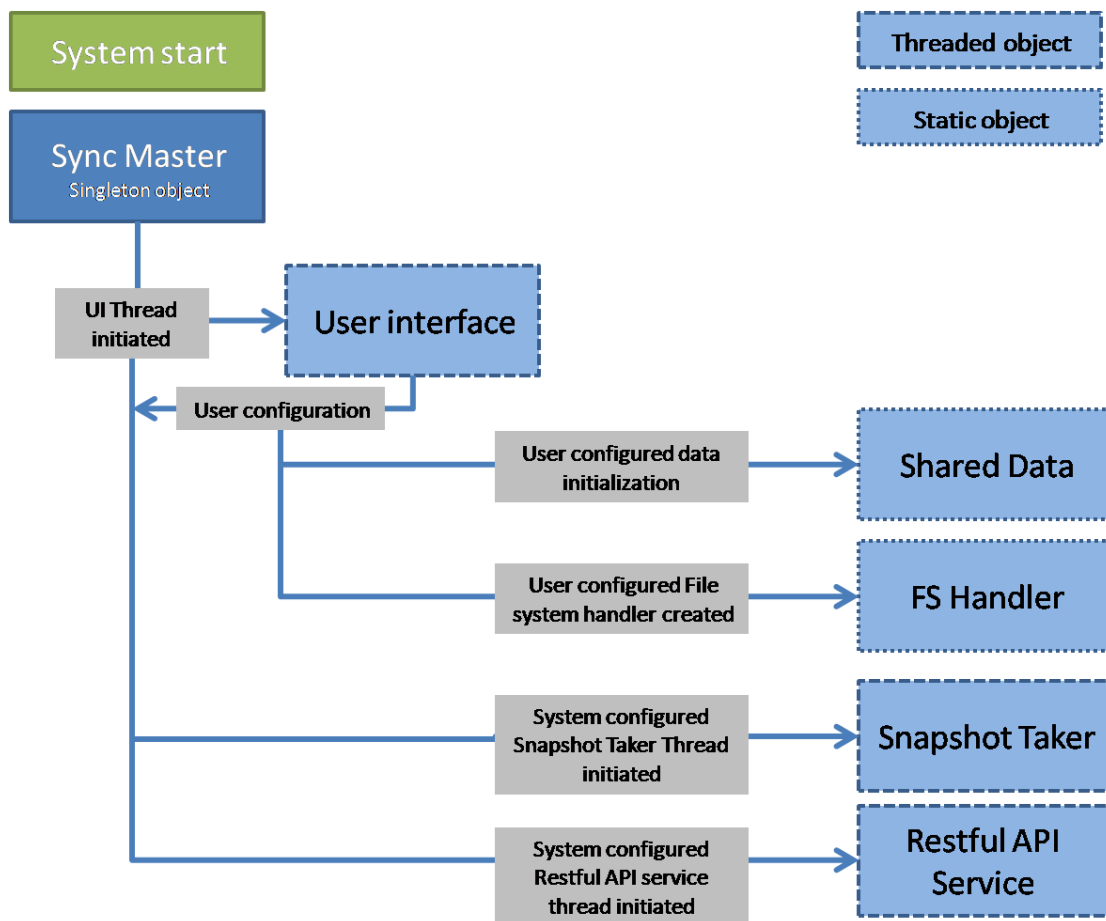
- שם המידע.
- מזהה ייחודי (ID).
- חותמת זמן.
- מידע:
- הרמה הבאה פנימה (בשתי הרמות העליונות).
- תמונה בפורמט base64 (ברמת ההרצאה).

## 7.3. התנהגות המערכת

### 7.3.1. צד השרת

#### 7.3.1.1. אתחול המערכת

- אתחול המערכת מתחיל ביצירת הישות הראשית והיא "המסנכרן הראשי" (Sync Master).
- המסנכרן יוצר תהליכון מנשק משתמש ומזניק אותו.
- בתצורה מקבילית, מחולל המסנכרן מופעים סטטיים של שתי מחלקות אשר אינן דורשות איפיון משתמש:
  - מחלקת המידע המשותף (Shared Data).
  - מנהל הקבצים (File system handler) – בתצורת תהליכון.
- עם תום פעולת האיפיון ע"י המשתמש והזנקת אופציית הפעלת ה'סשן', נוצרים מופעים של שתי מחלקות:
  - מחלקת הצלם (Snapshot Taker) – בתצורת תהליכון.
  - מחלקת שירות הקריאות (Restful API Service) – בתצורת תהליכון.



### **.7.3.1.2**

- תצלום מסך

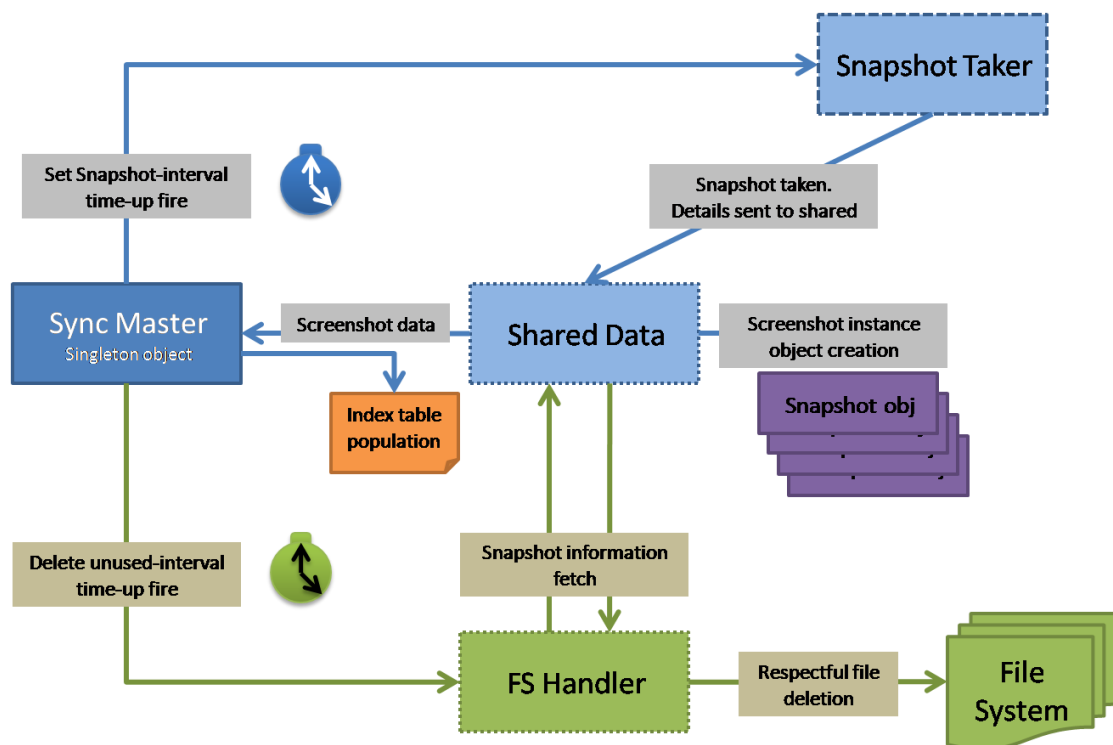
## תצלום המסך המתוזמן פועל באופן הבא:

- תהליכון הצלם נמצא במצב פעילה והוא דוגם דגל מיוחד שתפקידו לאותת לו לבצע צילום. הדגל הוא 'חבר' במחלקת המידע המשותף ומי שמפעיל אותו הוא המסנכרן הראשי.
- רגע לפני הדלקת דגל הצילום, מזין המסנכרן הראשי את מחלקת המידע המשותף בחותמת הזמן הרלוונטית העכשווית.
- עם הדלקת הדגל, הצלם מצלם את המסך ושומר את התמונה בנתיב מקומי תחת השם שהוא חותמת הזמן העכשווית אותה הצלם שואב ממחלקת המידע המשותף.
- את פרטי התצלום והקובץ מזין הצלם את מחלקת המידע המשותף והיא בתורה מחוללת מופע של "מחלקת הצילום הספציפי" אשר בתורה מזינה רשומה בטבלת המיפוי (אותה טבלה המכילה אינדוקס של חותמות הזמן, הנתיבים הפיזיים, דגלי השימוש ומונה השימוש).
- מתום פעולה זו מכבה הצלם את דגל "הזנקת הצילום" ובכך מיידע את המסנכרן כי המחזור הושלם.

- מחיקת קבצים

מחיקת הקבצים המתוזמנת פועלת באופן הבא:

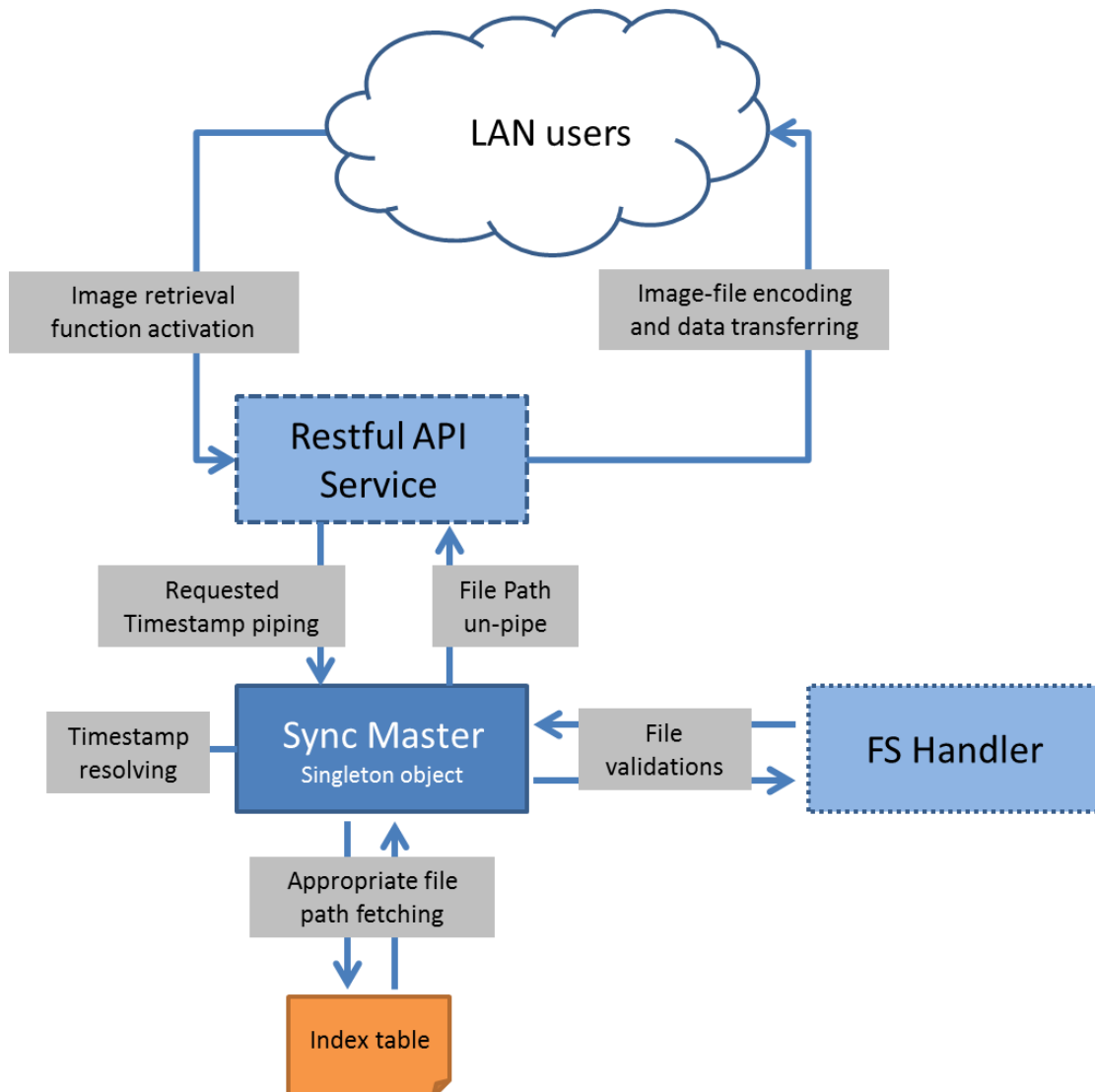
- אחת לזמן קבוע מראש, המתזמן הראשי מזניק באמצעות מנהל הקבצים תהליכון אנונימי ובו עותק זמני של טבלת המיפוי.
- התהליכון עובר בצורה סדרתית על הטבלה ומגלה בה קבצים שלא נעשה בהם שימוש (תמונות שצולמו אך לא נתבקשו).
- באותו האופן, ניגש המנהל אל אותם קבצים ומוחקם בעודו פוסח על הקבצים שסומנו ככאלה להם יש/היה ביקוש ע"י משתמש אחד או יותר.
- עם מחיקת קובץ, מעדכן תהליכון המחיקה את טבלת הקבצים על המחיקה.
- 



### 7.3.1.3 טיפול בקריאה לתצלום מסך

הטיפול בקריאת משתמש לתצלום מסך נעשה באופן הבא:

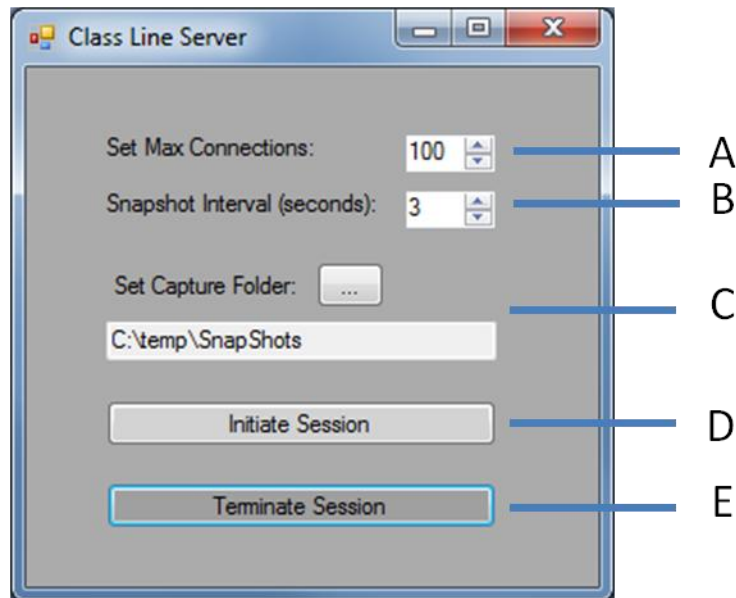
- קריאת בקשת תצלום מסך מתקבלת אצל תהליכון שירות הקריאות.
- שירות הקריאות מעביר את חותמת הזמן אל המסנכרן הראשי.
- המסנכרן ממפה את נתיב הקובץ המקומי לאור חותמת הזמן תוך שימוש בטבלת הקבצים אותה מנהל.
- הנתיב מועבר אל מנהל הקבצים לביצוע בדיקות אימות (וזאת על מנת למנוע נסיון העברה של קובץ פגום או חסר).
- מנהל הקבצים מאשר את תקינות הקובץ.
- המסנכרן מעביר לשירות הקריאות את נתיב הקובץ.
- שירות הקריאות מקודד את קובץ התמונה לטקסט ומשיבה אל הלקוח.



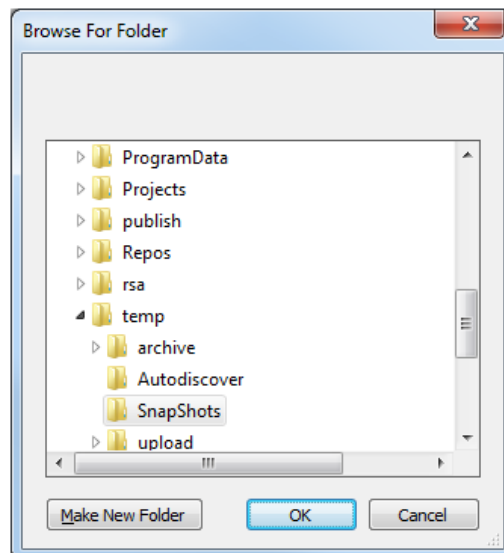
## 8. מדריך למשתמש

### 8.1. צד שרת

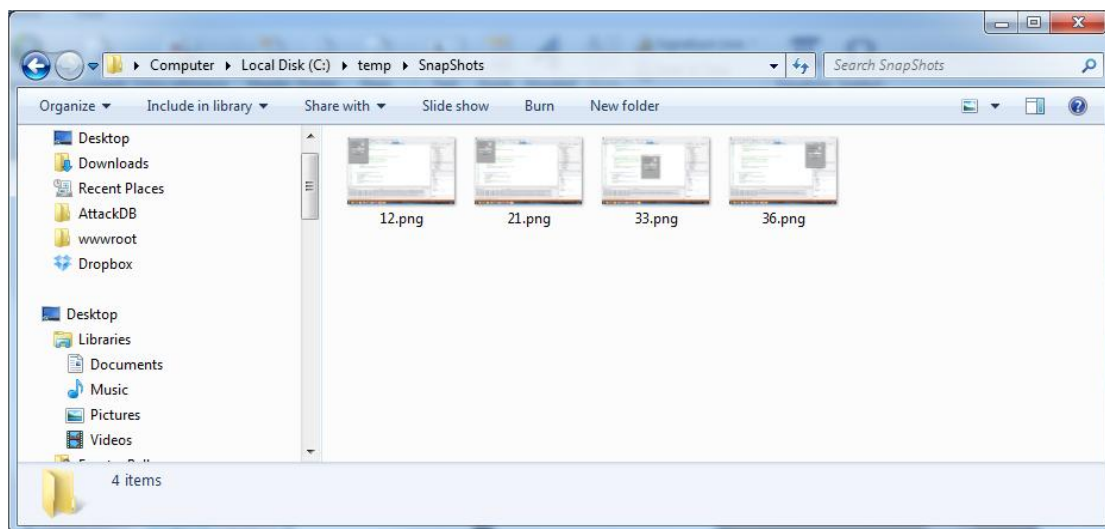
מנשק המשתמש בצד השרת הינו מינימליסטי ביותר מתוך מחשבה שממשק מורכב יצריך משך זמן ארוך לשימוש דבר שעשוי לגרוע מהזמן הכללי של ההרצאה מאיחד וכן עשוי גם להרתיע מרצים מלהפעיל את השרת מאידך. המוטיב העיקרי במנשק הוא של "שגר ושכח".



- A. הגבלת מספר המתחברים ל'סשן' הנוכחי – "Set Max Connections".  
פקד נומרי זה מאפשר למרצה להגביל את מספר המתחברים האפשרי מלמעלה.  
מספר החיבורים האפשריים נע בין 100 (כברירת מחדל) לבין 1000 ובקפיצות של 100.
- B. קביעת מרווח הזמן שבין צילום מסך אחד למשנהו – "Snapshot Interval".  
פקד נומרי זה מאפשר למרצה לקבוע באילו קבועי זמן (כלומר אחת לכמה זמן) 'צלם' המערכת יבצע את פעולת הצילום.  
הערך הנבחר מייצג שניות והוא נע בסקלה שבין 1-7. ברירת המחדל הינה 3 שניות.
- C. הגדרת תקיית קבצי התמונות – "Set Capture Folder".  
ע"י לחיצה על כפתור הניווט "... נפתח דיאלוג בחירת תיקייה. אל תיקייה זו ישלחו צילומי המסך מיד עם לקיחתם וכן משם הם ימחקו.



- דיאלוג בחירת תקיית הצילומים.



- דוגמה לתיקיית התמונות בה נשמרים התצלומים.

#### D. הפעלת ה 'סשן' – "Initiate Session".

כפתור הפעלת ה 'סשן'. מרגע שהופעל יחוללו המחלקות והתהליכונים בסדר קבוע, תצלומי המסך יחלו להתקבל וכן תחול האזנה לקריאות ממשתמשים.

#### E. סיום ה 'סשן' – "Terminate Session".

סיום ה 'סשן'.

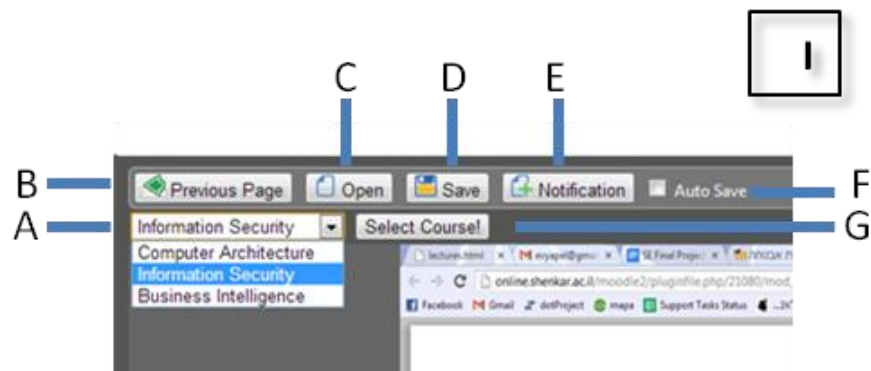
## 8.2. צד לקוח

מנשק המשתמש משתנה קלות בין מצב ניווט הקורסים וההרצאות לבין מצב הפעולה.

### 8.2.1. מצב ניווט הקורסים וההרצאות

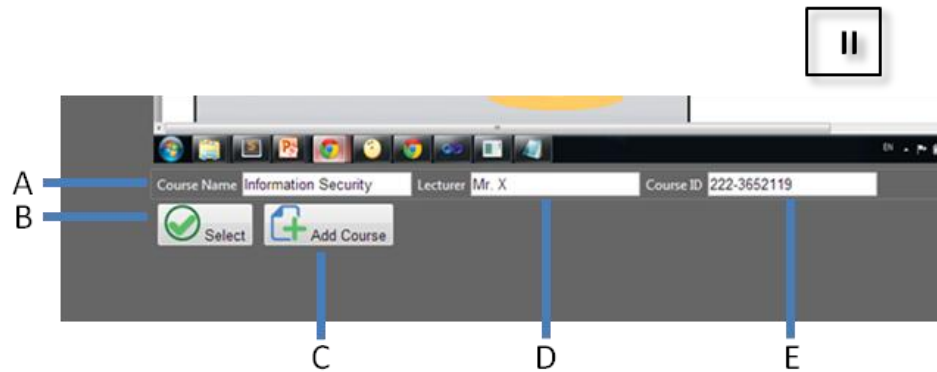


#### 8.2.1.1. תפריט כללי



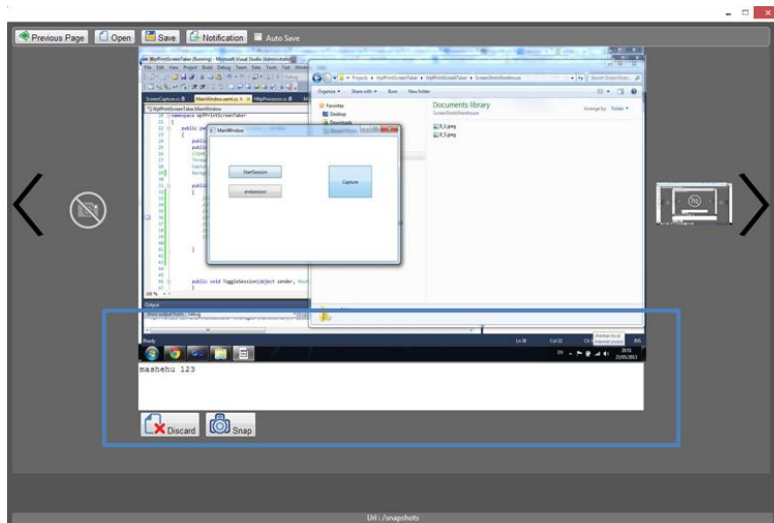
- A. לאחר טעינת קובץ (C), תופענה ברשימת הגלילה אפשרויות הרמה הבאה "Drill Down".
- B. ניווט למצב הקודם (למשל לטובת תיקון פעולה קודמת).
- C. פתיחת דיאלוג דפדפן קבצים לבחירת Class File.
- D. שמירת לקובץ פתוח או חדש.
- E. הפעלה וכיבוי הודעות צילום.
- F. הפעלה וכיבוי שמירה אוטומטית.
- G. בחירה (מרשימת הגלילה לטובת ירידה לרמה הבאה).

## 8.2.1.2 תפריט מידע



- A. תצוגה / עריכת שם קורס / ההרצאה.
- B. בחירה (מקביל לפתור G בתפריט הקודם).
- C. יצירת מידע חדש (קורס / הרצאה).
- D. שם המרצה.
- E. מזהה ייחודי.

## 8.2.2 מצב פעולה (לכידת תצלומי מסך)



### 8.2.2.1 תפריט לכידה



- A. איזור כתיבת הערות.
- B. מחיקת לכידה נבחרת (הלכידה המופיעה במרכז המסך).
- C. פתור לכידה.



## 9. Project's Management-Digest

The 'Class Line' system is a lecture-audience aid tool. It was developed during the fourth year of a bachelor degree in Software Engineering at Shenkar College as a finals project.

The systems main goal is to assist in obtaining, maintaining and sharing classroom, lecture or even meeting materials.

The 'Class Line' system upgrades the old fashioned 'Student note keeping' practice and lifts it a few steps forward into the technological presence relying on a the following basic assumptions:

1. The vast majority of today's frontal presentations are being assisted by some sort of a digital visual aid such as:
  - a. Slideshow (e.g. A PowerPoint presentation).
  - b. Internet based contents.
  - c. Some professional software (e.g. Compiler, Database engine, CG software).
2. Most students / audiences rely on personal computers to manage the material postdate.
3. Nowadays, many students prefer to use mobile computers (Laptops) to physically document lectures in opposite to the classic 'Hand written notebook'.
4. An ever-growing tradition of knowledge sharing is unfolding assistant by:
  - a. Email.
  - b. Shared folders.
  - c. Online forums and groups.

The 'Class Line' system's development process embraces, respects and encourages these behaviors while sustaining the absurd notion stating that it's aspiring (and yet simple) solution is, for some unknown reason, still not within reach.



**Shenkar College of Engineering and Design  
Faculty of Engineering  
Department Of Software Engineer**

# **Class Line**

## **A Student's aid system**

**Final Project**

**By  
Evyatar Pelleg  
Dvir Mizrahi**

**Mentored by  
Mr. Yechiel Kimchi**

**Submitted as Part of the Requirements for receiving  
The Bachelor of Science degree (B.Sc.)  
10/12/2013**