

פרק 1

פרויקט לפרויקט -- מסע הפיתוח המלא

מטרות הלמידה

בתום פרק זה, תוכל:

- להבין את מחוזר החיים המלא של פרויקט IA מרגע הרעיון ועד לתוצר ביצור
- לתכנן ולהוציא לפועל פרויקט IA מקופה לכמה
- להרכיב ולנהל צוות AI אפקטיבי
- לישם עקרונות eligA לניהול פרויקטי IA
- להבין יסודות spOLM לתחזקה ושיפור מתמיד
- למדוד הצלחה באמצעות IPK^s מתאים למערכות IA

פתחה

בשנת 9691, האנושות שלחה שלשה אנשים לירח והחזירה אותם בשלום לכדור הארץ. היה זה אחד המפעלים הטכנולוגיים המרשימים בהיסטוריה. אבל מה שהפך את תוכנית אפולו להצלחה לא היה רק המדע או הטכנולוגיה -- היה זה הניהול. התכנון המדוקדק, הפרויקט של יעד גדול לאלפי משימות קטנות, הוצאות הבינתיומי, והmdiידה המתמדת של ההתקדמות.

כיום, כשארגון מחייב מערכת IA, הוא עומד בפניו אתגר דומה. לא מדובר במשימה טכנית בלבד, אלא במפעל מורכב שדורש תכנון, ניהול, תאום וmdiידה. פרויקט IA שמתחליל בתלהבות יכול להיגמר בכישלון אם אין תשתיית ניהול נcona.

פרק זה מביא אתכם למסע המלא -- מהרגע שבו זיהיתם הزادנות לשימוש ב-IA, דרך בניית foorP tpecnoC (COP), ועד להפיקתו למוצר מלא ביצור. נדון בהרכבת צוות, בניהול פרויקטים בשיטת eligA המותאמת ל-IA,abisודות spOLM, ובmdiידת הצלחה.

אם הפרויקטים הקודמים למדו אתכם "מה" ו"איך", פרק זה מלמד "כיצד להצליח לעשות זאת בפועל בארגון אמיתי".

1.1 yrevocsiD -- זיהוי הזדמנויות AO בארגון

1.1.1 האמונות של לשאול את השאלות הנכונות

תארו לעצמכם ארגון בינווי בתחום השיווק הדיגיטלי. המנהלת הכללית יושבת במשרדה ושוالت את עצמה: "איפה AO יכול לעזור לנו?" זו השאלה הנכונה, אבל היא רחבה מדי. כדי לזיהות הזדמנויות אמיתיות, צריך לשאול שאלות ספציפיות יותר:

- איזה משלימות חוזרות על עצמן אנחנו עושים שוב ושוב?
 - איפה העובדים שלנו מבזבזים זמן על עבודה ידנית שאין בה ערך נוסף אמיתי?
 - איזה החלטות אנחנו מקבלים על בסיס נתונים שאפשר לנתח אוטומטית?
 - איפה אנחנו נתקלים ב"צווארי בקבוק" בתהליכי העבודה?
 - איזה שירות לקוחות אנחנו נותנים שאפשר לשפר באמצעות אוטומציה חכמה?
- תהליך ה-yrevocsiD תארו טכני -- הוא עסק. מדובר בזיהוי כאבים ארגוניים אמיתיים שאפשר לטפל בהם באמצעות IA.

2.1.1 yrevocsiD מתודולוגית

תהליך yrevocsiD מסודר כולל מספר שלבים [51], [62]:

1. **ראיון עם בעלי עניין (redlohekatI sweivretnI):** שוחחו עם מנהלים ועובדים מכל המחלקות. שאלו על יומם העבודה שלהם, המשימות החוזרות, הכאבם והאתגרים.

2. **ניתוח תהליכי (sisylanA ssecorP):** מפנו את תהליכי העבודה הקיימים. איפה יש ידנות? איפה יש חוסר יעילות?

3. **ערכת תנאים (ataD tnemssessA):** האם יש לכם תנאים? איך הם מאוחסנים? האם הם נגישים? תנאים הם הדלק של IA -- בלאיהם לא תצליחו.

4. **תעדוף הזדמנויות (noitazitiroirP ytinutropp):** לאחר שזיהיתם 01-02 הזדמנויות פוטנציאליות, דרגו אותן לפי שני צירים:

- **ערך עסקי פוטנציאלי:** כמה זה יחסוך/ירוויח לארגון?
- **מורכבות יישום:** כמה קשה להטמייע את הפתרון?

5. **בחירה esaC es ראשונית:** בחרו פרויקט ראשון עם ערך גבוה ומורכבות נוכחית-בינונית. זה יהיה "הניצחון המהיר" (QniW kciuQ) שלהם.

חברת הייטק בינוונית מהה שתהליק סיינו קורות חיים אווך זמו רב. מנהלת RH מדוחת שככל משירה פתיחה מקבלת 003-002 קורות חיים, וצוות ה-RH מבלה שעות על קריאה ידנית. **ההזדמנויות:** בניית סוכן AO שמסנן קורות חיים לפי התאמת למשרה, מדרג מועמדים, ומעבר לצוות רק את 02 המועמדים המוביילים.

הערך: חיסכון של כ-04 שעות עבודה לכל משרה (2 שעות □ 02 מועמדים שנבדקו ידנית במקום 003).

המורכבות: נוכחית-בינונית (ניתן למשם עם GAR tpmorP + Egnireenigngn).

3.1.1 **תיעוד הממצאים**

בסיום תהליך ה-D*revocsi*, תכינו מסמך "IA O'paM ytinutropp" -- מפת הזדמנויות שתכלול:

□ רשימת U sesaC מתועדת

□ הערך העסקי המשוער לכל אחד

□ הדרישות הטכנולוגיות הבסיסיות

□ סיכוני היישום

□ המלצה על esaC esU ראשון

מסמך זה יהיה הבסיס לתכנון הפרויקט הראשון.

2.1 **מ-COP ליצור -- המסע ואתגריו**

1.2.1 **הוכחת היתכנות COP (tpecnoC fo foorP) -- הוכחת היתכנות**

COP הוא שלב ראשון קրיטי [2]. מטרתו אינה לבנות מוצר מוגמר, אלא להוכיח שהרעיון ישים טכנולוגית ועונה על הצורך העסקי.

מאפייני COP טוב:

□ **מוגבל בהיקף:** פוטר בעיה אחת ספציפית

□ **זמן קצר:** 2-4 שבועות לכל היותר

□ **מדויק:** יש מודיעין הצלחה ברורים

□ **נבדק עם משתמשים:** לא רק עם מנהלים

ארגוני רבים טועים ומנסים לבנות מערכת מושלמת כבר ב-COP. התוצאה: פרויקט שנמשך חודשים, עולה הון, ולבסוף איןנועונה על הצורך. **זכור:** COP צריך להוכיח ערך, לא לספק מוצר מוגמר.

2.2.1 **מ-COP ל-COP -- הרחבת המבוקרת**

COP הצליח? מצוין! עכשו הגיע הזמן ל-COP. מהו ?**toliP** toliP הוא הרחבה מבוקרת של-h-COP לקבוצת משתמשים קטנים (01-05 משתמשים).

מטרה: לבדוק את הפתרון בתנאי עבודה אמיתיים, לפחות משוב, ולהבין מה צריך לשפר. **שלבי :** toliP

1. **בחירה משתמשי toliP:** בחרו משתמשים מייצגים שימושיים לחתם משוב

2. **הזרפה:** הסבירו להם איך להשתמש במערכת

3. **תקופת ריצה:** 4-8 שבועות

4. **איסוף משוב:** סקרים, ראיונות, ניתוח שימוש

5. **שיפורים:** תקנו בעיות ושברו על בסיס המשוב

3.2.1 "eviL oG" -- (noitcudorP)

הגענו לשלב הクリティי ביותר: העברת המערכת ליעוץ מלא [71].
eviL oG tsilkcehC

- ✓**בדיקות מקיפות:** resU ,stseT noitargetnI ,stseT tinU ,stseT ecnatpeccA
- ✓**ביצועים:** המערכת עומדת בעומסים צפויים?
- ✓**אבטחה:** noitarteneP ,stseT ytiruceS ,tiduA
- ✓**גיבויים:** תחוליך pukcaB ו-yrevoceR מוגדר ונבדק
- ✓**גיטור:** gniggoL-1 gnirotinoM פעילים
- ✓**תיעוד:** מדריכי משתמש, תיעוד טכני
- ✓**הזרבה:** כל המשתמשים עברו הדרכה
- ✓**תמיכה:** ksedpleH מוכן לטיפול בפניות

חברת סחר אלקטרוני ביןונית החליטה להטמע מערכת AI לתמיכת לקוחות.
COP (שבועיים):

- בנוC פשוט עם GAR על 05 שאלות נפוצות
- בדקנו עם 3 נציגי שירות
- הצלחה: 07% מהשאלות קיבלו תשובות נכונות
- הרחיבו ל-005 שאלות נפוצות
- שילבו עם מערכת MRC
- חשו ל-01% מהלקוחות (כ-005 לקוחות ביום)
- אספו משוב: בקשר יכולת העברה לנציג אנושי מהר יותר
- (לאחר 3 חודשים מההתחלת): noitcudorP
- השקעה מלאה לכל הלקוחות
- הפחיתה של 04% בפניות לנציגים אנושיים
- חיסכון של כ-08,000 ש"ח בחודש בעליות שירות
- שביעות רצון לקוחות: 85% (לעומת 87% קודם)

4.2.1 אתגרי המעבר ליעוץ

המשע COP ליעוץ אינו חלק [31]. הנה האתגרים העיקריים:

1. **Selac**: מה שעבד על 01 משתמשים לא בהכרח יעבד על 000,1
2. **נתונים**: ב-POLC היו נתונים נקיים. בייצור -- נתונים אמיתיים מლוכלים
3. **אינטרגרציה**: נדרש להתחבר למערכות קיימות (PRE, MRC וכו')
4. **אבטחה**: דרישות אבטחה מחמירות יותר בייצור
5. **תחזוקה**: מי יתחזק? מי יטפל בבעיות?
6. **שינויי ארגוני**: התנגדות של עובדים לשינוי

3.1 הרכבת צוות AI

1.3.1 תפקידים בצוות AI

צוות AI מוצלח הוא רב-תחומי [5]. להלן התפקידים המרכזיים:

דיקפת	תפקיד	ומ%
AI Product Manager	וינע ילعب סע תרושקת, פודעת, תושירד תרדגה	100%
Data Scientist	סילודם תריכב, רקחם Prompt Engineering	60-100%
ML Engineer	היצרטניא, חותיפ MLOps	100%
Data Engineer	ETL, סינונן תיתשת Vector DB	40-60%
Backend Developer	הסירפ, הייצרטניא API	60%
Frontend Developer	UX, שימוש קשمام	40-60%
DevOps Engineer	רוטין, CI/CD, תיתשת	40%
QA Engineer	פוקית, תוקידב, Evaluation	60%

טבלה 1.1: תפקידים בצוות AI טיפוסי

לא כל ארגון יכול להרשות לעצמו צוות של 8 אנשים. לפרויקט קטן, די ב:

- 1 reganaM tcudorP IA
 - 1 repoleveD kcatS-lluF IA
 - 1 reenignE LM/tsitneicS ataD IA
- שלושת אנשים אלה יכולים להוציא לפועל COP וtolIP. כশמרחבים לייצור -- תגיסו עוד.

2.3.1 בניית הצוות -- rentrap sv yuB sv dliuB

יש שלוש אסטרטגיות לבניית צוות AI [61]:

1. B dliuB (בנייה פנימית):

- **יתרונות**: שליטה מלאה, ידע נשאר בארגון
- **חסרונות**: גiros קשה, עלות גבוהה
- **מתי**: כשייש תקציב ופרויקטים ארוכי טווח

2. By (השכרת חברה חיצונית):

- **יתרונות:** מומחיות מיידית, ללא גiros
 - **חרוגנות:** עלות גבוהה, תלות בספק
 - **מתי:** לפROYיקטים חד-PURמיים או COP

3. rentraP (שותפות היברידית):

- **יתרונות:** העברת ידע, בניית יכולת פנימית
 - **חסרונות:** דורש תיאום, לוקח זמן
 - **מתי:** המודל המומלץ ברוב המקורים

המחלקה: התחלו עם `rentraP` (אחות חיצוני + 1-2 אנשים פנימיים). ובנו בהדרגה אחות פנימי.

4.1 ניהול פרויקט IA בגישה eligA

1.4.1 **למה eligible מתאים לא-IA?**

פרויקט IA הם בעלי אינדיקציות גבואה [7]. אתה לא יודע מראש אם מודל מסוים יעבד, או איך המשתמשים יגיבו. לכן, גישת Wafraella המסורתיות (תכנו מלא מראש, פיתוח אrox, בדיקה בסוף) אינה מתאימה.

- **איתרכיות קצרות (stnirps):** כל 2 שבועות יש תוצר עבודה
 - **מושב מהיר:** בודקים עם משתמשים כל הזמן
 - **גמישות:** אפשר לשנות כיוון על סמך למידה
 - **שיטתי פאולה:** צוות עובד יחיד. לא בסילואים

מגנה 2.4.1 tnirps IA-2 tcejorP

סימני צירופי ראורד שרוועיים:

סוי	תוליעוף	סיפתתשם
(ב) 1 סוי	?هزאה Sprint-B הנקן המ -- Sprint Planning	תוווצה לכ
9-2 סמי	(רקעב תוקד 15 Daily Standups) + חותיפי תדובע	תוווצה לכ
(ה) 10 סוי	וינע ילבול רצotta תגצה -- Sprint Demo	תוווץ + Stakeholders
**(ה) 10 סוי	?רפלט המ ?דבע המ -- Retrospective	תוווצה לכ

טבלה 4.2: מבנה tnirp דו-שבועי

****omeD-1 ו-2 באותו יום. אחד אחרי השני**

IA- seirotS resU 3.4.1

ב-eligA, כל דרישת מנוסחת כ-"[\[4\]](#) "yrotS resU".
תבניות:

"כ-[תפקיד], אני רוצה [פעולה], כדי [ערץ עסק]"

דוגמאות:

- "כנציג שירות, אני רוצה שהצ'אטבוט ימליץ לי על מוצר מתאים ללקוח, כדי לשפר את שביעות הרצון"
 - "כמנהל RH, אני רוצה לקבל רשימת 20 מועמדים מובילים לכל משרה, כדי לחסוך זמן סינון"
 - "כמנהל, אני רוצה דashboard עם IPK של מערכת IA, כדי לעקוב אחר ביצועים"
- לכל yrotS יש:
- airetirC ecnatpeccA: מתי IA-S yrotS נחשבת גמורה?
 - (...31, 8, 5, 3, 2, 1, :stnrioP yrotS) הערכת מאמצן

tnemeganaM golkcaB 4.4.1

ה-IA reganaM tcudorP הוא רשימת כל IA-seirotS. להזק את IA-golkcaB tcudorP אחראי:

1. לתחזק את IA-golkcaB (להוסיף, למחוק, לעדכן)
2. לתעדף לפי ערך עסק
3. להבטיח שה-IA seirotS ברורות וモבנות

בלחץ להוציא תוכנות, צוותים לפעים "מקקרים דרך" ולא עושים ,gnitseT ,gnirotcafeR .זה נקרא lacinhceT noitatnemucoD .[22].

הכלל: הקדישו 20% מכל tnirps לתשלום "חוב טכני" -- ,stseT ,gnirotcafeR ,gnitseT ,gnirotinoM gniggoL .gniniarteR

5.1 יסודות spOLM -- תחזקה ושיפור מתמיד

1.5.1 מהו spOLM ?

spOLM (gninraeL enihcaM) spOveD snoitarepO IA ,gnirotinoM gniggoL gniniarteR .[11][42]. מטרתו: להבטיח spOLM spOLM רביי: spOLM רכזות באופן יציב, מתוחזקות ומשתפרות לאורך זמן.

1. spOLM: גרסה של קוד, מודלים, נתונים, spOveD snoitarepO gninraeL enihcaM .gnirotinoM gniggoL .gniniarteR
2. DC/IC: אינטגרציה ופרישה אוטומטיות
3. spOLM: ניטור ביצועים, שגיאות, שימוש gnirotinoM .gniggoL .gniniarteR
4. spOLM: רישום אירועים לניתוח
5. gnitrelA: התראות כמשמעות לא תיקון
6. spOveD snoitarepO gninraeL enihcaM : עדכון מודלים/spOLM לפי צורך

IA lortnoC noisreV 2.5.1

בפיתוח תוכנה רגיל, אתה עושה IA lortnoC noisreV רק בלבד. במערכות IA, צריך גם [52], [72]:

□ כל שינוי במודול (sretemaraP ,tpmorP) מותועד gninoisreV ledoM □

□ מעקב אחר גרסאות של stesatad gninoisreV ataD □

□ תיעוד של כל ניסוי (מודול, tpmorP, תוצאות) gnikcarT tnemirepxE □

כלים:

□ tiG לקוד □

□ (lortnoC noisreV ataD CVD) IC לנתונים □

□ sesaiB & sthgieW / wolfLM לניסויים □

(DC/IC) tnemyolpeD suounitnoC / noitargetnI suounitnoC 3.5.1

:[01],[9] IC (noitargetnI suounitnoC): כל שינוי בקוד עבר אוטומטית :

1. dliuB (בנייה הקוד)

2. stseT (בדיקות אוטומטיות)

3. (nacS ytiruceS ,gnitniL) skcehC ytilauQ

DC (tnemyolpeD suounitnoC): אם הכל עבר, הקוד נפרש אוטומטית לטביבות ייצור.
טיפוסי: enilepiP

(snoitcA buHtiG) DC/IC buHtiG hsuP edoC
duolC ot yolpeD egamI rekcoD dliuB stseT nuR □

gniggoL-1 gnirotinoM 4.5.1

מה לנטר? [3],[41]

□ ביצועי מודל: ycnetaL tuphguorhT (זמן תגובה), (בקשות לדקה)

□ איקות תשובות: ecnaveleR ,ycaruccA (באמצעות noitaulavE)

□ שגיאות: sepyT noitpecxE ,etaR rorrE

□ שימוש: כמה משתמשים, כמה שאלות, sruoH kaeP

□ עלויות: כמה טוκנים, כמה זה עולה

כלים:

□ anafarG + suehtemorP : ניטור ודאשborדים

□ gniggoL : (anabiK ,hsatsgoL ,hcraescitsalE) kcatS KLE וCHIP

□ yrtnes : מעקב שגיאות

□ enocileH / htimsqnaL : ניטור ספציפי ל-MLL

gniniarteR leoM tfirD -- והצריך ב- 5.5.1

בזמן, ביצועי מערכות AI יכולם להידדר. זה נקרא M[8], [21] tfirD leoM. למה זה קורה?

□ הנתונים משתנים (למשל, ל��שות שואלים שאלות חדשות)

□ הממציאות משתנה (ሞצרים חדשים, מחירים)

□ stpmorP שפעם עבדו כבר לא מספיק טובים

פתרונות: etadpU tpmorP / gniniarteR

1. ניטור ביצועים קבוע

2. כשאיכות יורדת מתחת ל- -- התרעעה

3. עדכון gnut-eniF / stpmorP / החלפת מודול

4. בדיקה ופרישה מחדש

6.1 מדידת הצלחה -- sIPK למערכות AI

1.6.1 מה sIPK קרייטיים?

"מה שלא נמדד, לא מנוהל" -- פיתר דראקר בלי מדידה, אתה לא יודע אם הפרויקט הצליל, איפה לשפר, או אם להמשיך להשקיע. הם המציגן שלך [81].

2.6.1 שלושה סוגים sIPK

1. sIPK עסקי (ssenisubB): מודדים ערך עסקי ישיר [6], [91].

$$\text{AI_Project_ROI} = \frac{\text{תלוק העקשה} - \text{רצון צרע}}{\text{תלוק העקשה}} \times 100\%$$

דוגמיה:

□ ערך נוצר: 000,005 ש"ח (חיסכון שנתי בעלות שירות)

□ השקעה: 000,002 ש"ח (פיתוח + תחזוקה שנה ראשונה)

□ IOR = 000,002 / (000,005 - 000,002) = 0.051

sIPK עסקי נוספים:

□ חיסכון בעלות: כמה סף חסכנו?

□ הגדלת הבנסות: כמה הכנסות נוספות הניבה המערכת?

□ שיפור שביעות רצון לקוחות (TASC): לפני ואחרי

□ **eulaV ot emiT**: כמה זמן עד שראינו ערך ראשון?

$\text{Time_to_Value} = \text{First Value} - \text{Kickoff}$ דע (סידוח/תועובשב)

דוגמא:

□ 1 ינואר ffokciK

□ COP) eulaV tsriF 51 בפברואר

□ = 6 שבועות eulaV ot emiT

הערה: ככל ש-*TTT* קצר יותר, כך הפרויקט מהיר יותר בהניבת ערך.

2. SIPK טכניות (sIPK lacinhceT): מודדים ביצועים טכניים.

□ **ycnetaL**: זמן תגובה ממוצע (בשניות)

□ **tuphguorhT**: בקשות לדקה

□ **etaR rorrE**: % שגיאות

□ **emitpU**: % זמנים זמינים (ALS: 99.99%)

□ **egasU nekoT**: צריכת טוקנים ממוצעת לבקשה

3. SIPK של אימוץ (sIPK noitpodA): מודדים עד כמה המשתמשים משתמשים במערכת.

$$\text{Adoption_Rate} = \frac{\text{סיליפ סישמתשム}}{\text{סילאייצנטופ סישמתשム}} \times 100\%$$

דוגמא:

□ משתמשים פוטנציאליים (כל נציגי השירות): 05

□ משתמשים פעילים (השתמשו לפחות פעם אחת בשבוע): 04

□ $04 / 05 = 0.8 = \text{etaR noitpodA}$

sIPK אימוץ נוספים:

□ **yriaD evitcA sresU**: כמה משתמשים ביום

□ **etaR tnemegagnE**: % מהמשתמשים שחוזרים

□ **egasU erutaeF**: איזה תוכנות בשימוש ואיזה לא

3.6.1 **sIPK draobhsaD ל-**

draobhsaD טוב מציג את כל ה-sIPKs במקומות אחד, בזמן אמיתי.
מבנה draobhsaD מומלץ:

1. סקירה כללית (weivrevO):

- IOR
- etaR noitpodA
- emitpU

2. ביצועים טכניים:

- גראן ycnetaL לאורך זמן
- גראן rorrE
- גראן nekoT

3. שימוש:

- (sresU evitcA yliaD) UAD
- sruoH kaeP
- תוכנות פופולריות

4. עליות:

- עלות IPA חודשית
- עלות לבקשה
- תחזית עליות

כליים לבניית draobhsaD:

- anafarG: פתוח, חזק, מCustomLabel
- desab-nohtyP: פשוט, מהיר, חזותי
- uaelbaT / rewoP: עסקי, חזותי

7.1 דוגמאות מעשיות

1.7.1 **esaC esdS ydutS : פרויקט IA מוצלח מ-A עד Z**

ארגון: בנק בגין עם 002 עובדים
אתגר: תהליכי אישור הלואאות לעסקים קטנים אורך 2-3 שבועות ודורש עבודה ידנית רבה של אנליסטים.

(שבועיים): yrevocsiD

- ריאיון עם 01 אנליסטים
- מיפוי תהליכי אישור הלואאות (41 שלבים!)
- זיהוי: 6 שלבים ניתנים לאוטומציה עם IA

◻ ערך פוטנציאלי: חיסכון 60% בזמן

COP (3 שבועות):

◻ בניית סוכן IA שמנתח מסמכים פיננסיים (דוחות רווח והפסד, מאזורים)

◻ מחלץ נתונים מרכזיים ומיצר "ציוון סיכון" אוטומטי

◻ נבדק על 20 הלוואות קודומות

◻ תוצאה: דיק 58% בהשוואה לanelistim אונשיים

Poli (חודשיים):

◻ הרחבת ל-100 הלוואות חדשות

◻ 5anelistim משתמשים במערכת

◻ זמן אישור ממוצע ירד מ-41 ימים ל-7 ימים

◻ משוב: המערכת טובה, אך צריכה שיפור באיזוי מסמכים סרוקים

Poitcudor (לאחר 4 חודשים):

◻ השקעה מלאה לכל מחלקת הלוואות

◻ 100% מהלוואות עוברות דרך המערכת

◻ זמן אישור ממוצע: 5 ימים (שיפור 46%)

◻ חיסכון שנתי: 2.1 מיליון ש"ח

◻ IOR: 4004% בשנה הראשונה

לקחים:

◻ revocsiD מסודר חוסך זמן בהמשך

◻ COP קוצר עם מדדים ברורים מקטין סיכונים

◻ שיתוף משתמשי קצה מוקדם מבטיח הצלחה

◻ שיפורים על בסיס משוב Poli קרייטיים

2.7.1 esaC ydutS : ניתוח כישלון -- מה לא לעשות

ארגוני: חברת ביטוח גדולה

רקע: בניית "סוכן IA אוניברסלי" שיטפל בכל פניות הלקוחות.

מה השتبש?

1. revocsiD חסר:

◻ הנהלה החליטה לבנות IA בלי לשאול את נציגי השירות

◻ לא זיהו את הכאבים האמיתיים

◻ תוצאה: פתרון לבעה שלא הייתה

2. אין COP -- ישר לפיתוח מלא:

- החליטו לפתח מערכת ענקית של 6 חודשים
- לא בדקו היתכנות טכנולוגית
- תוצאה: גילו רק בסוף שהמודל לא מספיק טוב

3. צוות לא מתאים:

- שכחו חברה חיצונית יקרה בלי ידע פנימי
- לא העבירו ידע לארגון
- תוצאה: תלות מוחלטת בספק

4. אין sIPK:

- לא הגדרו מדי הצלחה
- לא ניטרו ביצועים בזמן הפתוח

□ תוצאה: גילו רק בסוף שהמערכת לא עובדת

5. peerC epochS:

- כל פעם הוטיפו דרישות חדשות
- הפרויקט התנפח מ-6 חודשים ל-81 חודשים
- תוצאה: תקציב התפוצץ (5.2 מיליון במקום 800,000)

תוצאה סופית:

- המערכת נזנחה לאחר 81 חודשים
- הפסד 5.2 מיליון ש"ח
- תדמית AI בארגון נפגעה
- לקח 3 שנים עד שנכונו לנסות שוב

לקיים:

- **אל תדלו על revocsiD** -- זה לא בזבוז זמן, זה חיסכון
- **תתחילו ב-COP קטן** -- הוכיחו ערך לפני השקעה גדולה
- **בנו ידע פנימי** -- אל תהיו תלויים לחלוין בספקים
- **הגדרו KIPs מראש** -- תדעו متى להמשיך ומתי לעצור
- **שמרו על epochS קבוע** -- תגידו "לא" לדרישות חדשות

3.7.1 **תכנון pamdaR למחלקה IA**

רבעון 1 (חודשים 1-3): pamdaR

- **sesaC esU 5 זיהוי + yrevocsiD**
- COP לשני sesaC esU הטוביים ביותר
- **בנייה צוות ליבה (3 אנשים)**
- **הגדרת sIPK**

רבעון 2 (חודשים 4-6):

- **toliP ל-U esaC esU** הראשון עם 02 משתמשים
- הרחבת הצוות (2 אנשים נוספים)
- **בנייה תשתיית spOLM בסיסית**
- **הדרכת משתמשים**

רבעון 3 (חודשים 7-9):

- **eviL oGesaC esU** השני
- **toliP ל-U esaC esU** השני
- **בנייה sIPK-draobhsaD**
- **תיעוד ושיתוף** לקרים

רבעון 4 (חודשים 10-21):

- **eviL oG esaC esU** השלישי
- **COP לשני COP esaC esU**
- **הערכת IOR** של שנה ראשונה
- **תכנון pamdaR** לשנה השנייה

התוצאה לאחר שנה:

- **2 מערכות IA ביצור**
- **צוות של 5 אנשים עם ידע מעמיק**
- **IOR מוכן**
- **תשתיית spOLM מוכנה להרחבה**
- **תרבות IA מתחילה להיבנות בארגון**

8.1 תרגילים

1.8.1 תרגיל 1: כתיבת retrahC tcejorP לפרויקט IA

retrahC tcejorP הוא מסמך הפתיחה של פרויקט. הוא מגדר מה, למה, מי ומתי. משימה: כתבו retrahC tcejorP לפרויקט IA בארגון שלכם (או בארגון דמיוני). מבנה: retrahC tcejorP

1. **שם הפרויקט**
2. **רקע ומטרה:** למה אנחנו עושים את זה?
3. **היקף (epoch):** מה פנים, מה בחוץ
4. **מטרות (Goal):** מדידות וספקטיביות
5. **sIPK:** איך נמדד הצלחה?
6. **בעלי עניין (Stakeholders):** מי מעורב?
7. **צוות:** מי עובד על הפרויקט?
8. **לוח זמנים:** متى זה יגמר?
9. **תקציב:** כמה זה עולה?
10. **סיכום:** מה יכול להשתבש?

דוגמיה:

1. **שם הפרויקט:** metsyS gnineercS VC derewoP-IA
2. **רקע ומטרה:** מחלוקת RH מבלח כ-04 שעות בחודש על סינון יdni של קורות חיים. המטרה: להפחית זמן סינון ב-07% באמצעות AI.
3. **היקף:**
 - **בפנים:** סינון קורות חיים, דירוג מועדים, המלצת pot 02
 - **בחוץ:** ריאיון אוטומטי, הצעת עבודה אוטומטית
4. **מטרות:**
 - הפחיתת זמן סינון מ-04 04-21 שעות בחודש
 - שיפור איצות מועדים שעוברים לריאיון ב-20%
 - eviL eviL תוך 3 חודשים
5. **sIPK:**
 - זמן סינון ממוצע למשרה
 - % מועדים שעברו לשלב ריאיון והתקבלו
 - etar noitpodA (כמה משתמשי RH משתמשים)

□ IOR לאחר 6 חודשים

6. בעלי עניין:

□ ניהול RH (rosnops)

□ צוות sretrourceR (משתמשי קצה)

□ OTC (תומך טכני)

7. צוות:

□ reganaM tcudorP (חצי משורה)

□ reenignE LM (משורה מלאה)

□ reenignE AQ (רביע משורה)

8. לוח זמנים:

□ COP: 3 שבועות

□ :tolip 4 שבועות

□ eviL oG: 3 חודשים מהיום

9. התקציב:

□ שכר צוות: 1,050,000 ש"ח

□ :stsoC IPA 5,000 ש"ח

□ כלים: 1,000 ש"ח

□ סה"כ: 1,561,000 ש"ח

10. סיכונים:

□ קשי בגישה לנוטרי קורות חיים (הסתברות: בינונית)

□ התנגדות של sretrourceR (הסתברות: נמוכה)

□ דיק המודל נמוך מ-80% (הסתברות: בינונית)

2.8.1 תרגיל 2: תכנון צוות AI אידיאלי

משימה: תכננו את הצוות האידיאלי לפרויקט AI בארגון שלכם.
שאלות מוחות:

1. מהו גודל הפרויקט? (קטן / בינוני / גדול)

2. מהו התקציב?

3. האם יש ידע פנימי ב-AI?

4. האם עדיף ?rentraP / yuB / dliuB

5. אילו תפקידים נדרשים?

6. כמה זמן (חلكי/מלא) לכל תפקיד?

פלט מצופה: טבלה עם:

שם תפקיד

אחריות

% זמן

פנימי / חיצוני

עלות חודשית

3.8.1 תרגיל 3: בניית draobhsaD של sIPK

משימה: עצבו draobhsaD ויזואלי (על נייר או כלי עיצוב) שמציג sIPK למערכת AI.
דרישות:

4 אזוריים: עסק, טכני, אימוץ, עלויות

לפחות 3 מדדים בכל אזור

כולל גרפים (עמודות, קו, עוגה)

סימנו אזוריים קרייטיים באדום/ירוק

כלים מומלצים:

hctekS / amgiF (עיצוב)

steehS elgooG / lecxE (מוקאף מהיר)

anafarG (אם יש זמן ללמידה)

4.8.1 תרגיל 4: ניתוח esaC seidutS yduS ולמידה לקחים

משימה: קראו את שני ה-esaC seidutS esaC (הצלחה וכישלון) ומלאו טבלה:

אשוו	הchèה Case	וולשיך Case
Discovery	תועובש 2, רדוסם	היה אל
POC	סירורב סיידם, תועובש 3	וילע וגליד
תוצ
KPIs
Scope
האצת	ROI 400%	2.5M דספה

טבלה 3.1: השוואת esaC seidutS

שאלות להרחבה:

1. מה היה הגורם המרכזי להצלחה בesaC הראשון?

2. מה היה הגורם המרכזי לכישלון בesaC השני?

3. אילו לקחים תיקחו לפרויקט שלכם?

5.8.1 תרגיל 5: תכנון IA pamdaR ל-21 חודשים

משימה: תכננו pamdaR של 21 חודשים לפרויקט IA בארגון שלכם.
מבנה:

□ חלקו ל-4 רביעונים

□ כל רביעון: מטרות, פעילותות, selbarevileD, משאבים

דוגמה לרבעון 1:

□ **מטרות:** זיהוי הזרמוויות, COP

□ **פעילויות:** COP, yrevocsiD, בניית צוות

□ **淼:** מסמך pamD COP, selbarevileD

□ **משאבים:** 3 אנשים, K001 ש"ח

6.8.1 תרגיל 6 (קוד nohtyP): מערכת ניטור sIPK בסיסית

משימה: כתבו סкриיפט nohtyP שמחשב ומציג sIPK בסיסיים למערכת IA.
נתונים לדוגמה (NOSJ):

```

1 {
2     "total_requests": 10000,
3     "successful_requests": 9500,
4     "failed_requests": 500,
5     "total_tokens": 5000000,
6     "total_cost_usd": 250,
7     "potential_users": 100,
8     "active_users": 75,
9     "value_generated_usd": 10000,
10    "total_investment_usd": 5000
11 }
```

לчисוב: sIPK

001 □ stseuquer_latot / stseuquer_lufsseccus = etaR sseccuS .1

stseuquer_latot / tsoc_latot = tseuqeR rep tsoC .2

001 □ sresu_laitnetop / sresu_evitca = etaR noitpodA .3

001 □ tnemtsevni_latot / (tnemtsevni_latot - detareneg_eulav) = IOR .4

stseuquer_latot / snekot_latot = tseuqeR rep snekoT egarevA .5

פתרונות:

Listing 1.1: kpi_monitor.py

```

1 import json
2
3 def calculate_kpis(data):
4     """
```

```

5     Calculate basic KPIs for AI system
6
7     Args:
8         data (dict): System usage data
9
10    Returns:
11        dict: Calculated KPIs
12    """
13    kpis = {}
14
15    # Success Rate
16    if data['total_requests'] > 0:
17        kpis['success_rate'] = (
18            data['successful_requests'] /
19            data['total_requests'] * 100
20        )
21    else:
22        kpis['success_rate'] = 0
23
24    # Cost per Request
25    if data['total_requests'] > 0:
26        kpis['cost_per_request'] = (
27            data['total_cost_usd'] /
28            data['total_requests']
29        )
30    else:
31        kpis['cost_per_request'] = 0
32
33    # Adoption Rate
34    if data['potential_users'] > 0:
35        kpis['adoption_rate'] = (
36            data['active_users'] /
37            data['potential_users'] * 100
38        )
39    else:
40        kpis['adoption_rate'] = 0
41
42    # ROI
43    if data['total_investment_usd'] > 0:
44        kpis['roi'] = (
45            (data['value_generated_usd'] -
46             data['total_investment_usd']) /
47            data['total_investment_usd'] * 100
48        )
49    else:
50        kpis['roi'] = 0
51
52    # Average Tokens per Request
53    if data['total_requests'] > 0:
54        kpis['avg_tokens_per_request'] = (
55            data['total_tokens'] /

```

```

56         data['total_requests']
57     )
58 else:
59     kpis['avg_tokens_per_request'] = 0
60
61 return kpis
62
63 def display_kpis(kpis):
64     """Display KPIs in clean format"""
65     print("=" * 50)
66     print("AI System KPIs Dashboard")
67     print("=" * 50)
68     print(f"Success Rate: {kpis['success_rate']:.2f}%")
69     print(f"Cost per Request: ${kpis['cost_per_request']:.4f}")
70     print(f"Adoption Rate: {kpis['adoption_rate']:.2f}%")
71     print(f"ROI: {kpis['roi']:.2f}%")
72     print(f"Avg Tokens/Request: {kpis['avg_tokens_per_request']:.0f}")
73     print("=" * 50)
74
75 def main():
76     # Sample data
77     data = {
78         "total_requests": 10000,
79         "successful_requests": 9500,
80         "failed_requests": 500,
81         "total_tokens": 5000000,
82         "total_cost_usd": 250,
83         "potential_users": 100,
84         "active_users": 75,
85         "value_generated_usd": 10000,
86         "total_investment_usd": 5000
87     }
88
89     # Calculate and display KPIs
90     kpis = calculate_kpis(data)
91     display_kpis(kpis)
92
93 if __name__ == "__main__":
94     main()

```

פלט מצופה:

```

=====
draobhsaD sIPK metsyS IA
=====
%00.59 :etaR sseccuS
0520.0$ :tseuqeR rep tsoC
%00.57 :etaR noitpodA
%00.001 :IOR
005 :tseuqeR/snekoT gvA
=====
```

шиפורים אפשריים:

1. הוספת צבעים (ירוק/אדום) לפי סעיפים
2. שמיות היסטוריה ב-VSC
3. גרפים עם biltolptam
4. התראות כ-IPK יורד מתחת ל-סעיף

7.8.1 תרגיל 7 (קוד gniggoL ו-gnirotinoM): למערכת IA

משימה: בנו מערכת gniggoL בסיסית שמתעדת כל שימוש במערכת IA ושומרת goL elif.

מה לתעד?

pmatsemiT □

DI resU □

(tpmorP) tseuqeR □

esnorseR □

desU snekoT □

(sm) ycnetaL □

rorrE / sseccuS □

פתרונות:

Listing 1.2: ai_logger.py

```
1 import json
2 import logging
3 from datetime import datetime
4 from pathlib import Path
5
6 class AISystemLogger:
7     """Logging system for AI applications"""
8
9     def __init__(self, log_dir="logs"):
10        """Initialize Logger
11
12        Args:
13            log_dir (str): Logs directory
14        """
15        self.log_dir = Path(log_dir)
16        self.log_dir.mkdir(exist_ok=True)
17
18        # Configure Logger
19        self.logger = logging.getLogger("AISystem")
20        self.logger.setLevel(logging.INFO)
21
22        # File Handler
```

```

23     log_file = self.log_dir / f"ai_system_{datetime.now() .
24         strftime('%Y%m%d')}.log"
25     file_handler = logging.FileHandler(log_file, encoding='utf
26         -8')
27     file_handler.setLevel(logging.INFO)
28
29     # Format
30     formatter = logging.Formatter(
31         '%(asctime)s - %(name)s - %(levelname)s - %(message)s'
32     )
33     file_handler.setFormatter(formatter)
34
35     self.logger.addHandler(file_handler)
36
37     def log_request(self, user_id, prompt, response,
38                     tokens_used, latency_ms, success=True, error_msg
39                     =None):
40         """
41             Log AI system request
42
43             Args:
44                 user_id (str): User identifier
45                 prompt (str): The prompt sent
46                 response (str): The response received
47                 tokens_used (int): Token count
48                 latency_ms (int): Response time in milliseconds
49                 success (bool): Whether request succeeded
50                 error_msg (str): Error message (if any)
51
52             """
53
54         log_entry = {
55             "timestamp": datetime.now().isoformat(),
56             "user_id": user_id,
57             "prompt": prompt[:100], # First 100 chars only
58             "response": response[:100] if response else None,
59             "tokens_used": tokens_used,
60             "latency_ms": latency_ms,
61             "success": success,
62             "error_msg": error_msg
63         }
64
65         if success:
66             self.logger.info(json.dumps(log_entry, ensure_ascii=
67                             False))
68         else:
69             self.logger.error(json.dumps(log_entry, ensure_ascii=
70                             False))
71
72     def get_daily_stats(self):
73         """
74             Calculate daily statistics from log file
75
76             Returns:
77                 dict: Statistics
78

```

```

69      """
70      log_file = self.log_dir / f"ai_system_{datetime.now() .
71      strftime('%Y%m%d')}.log"
72
73      if not log_file.exists():
74          return {"error": "No log file for today"}
75
76      total_requests = 0
77      successful_requests = 0
78      failed_requests = 0
79      total_tokens = 0
80      total_latency = 0
81
82      with open(log_file, 'r', encoding='utf-8') as f:
83          for line in f:
84              if 'prompt' in line:
85                  total_requests += 1
86
87                  # Extract JSON from line
88                  json_start = line.find('{')
89                  if json_start != -1:
90                      try:
91                          entry = json.loads(line[json_start:])
92
93                          if entry.get('success', False):
94                              successful_requests += 1
95                          else:
96                              failed_requests += 1
97
98                          total_tokens += entry.get('tokens_used',
99                                         0)
100                         total_latency += entry.get('latency_ms',
101                                         0)
102
103                         stats = {
104                             "date": datetime.now().strftime('%Y-%m-%d'),
105                             "total_requests": total_requests,
106                             "successful_requests": successful_requests,
107                             "failed_requests": failed_requests,
108                             "success_rate": (successful_requests / total_requests *
109                                         100) *
110                                         if total_requests > 0 else 0,
111                             "total_tokens": total_tokens,
112                             "avg_tokens_per_request": (total_tokens /
113                             total_requests) *
114                                         if total_requests > 0 else 0,
115                             "avg_latency_ms": (total_latency / total_requests) *
116                                         if total_requests > 0 else 0
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
599
600
601
602
603
604
605
606
607
608
609
609
610
611
612
613
614
615
616
617
618
619
619
620
621
622
623
624
625
626
627
628
629
629
630
631
632
633
634
635
636
637
638
639
639
640
641
642
643
644
645
646
647
648
649
649
650
651
652
653
654
655
656
657
658
659
659
660
661
662
663
664
665
666
667
668
669
669
670
671
672
673
674
675
676
677
678
679
679
680
681
682
683
684
685
686
687
688
689
689
690
691
692
693
694
695
696
697
698
699
699
700
701
702
703
704
705
706
707
708
709
709
710
711
712
713
714
715
716
717
718
719
719
720
721
722
723
724
725
726
727
728
729
729
730
731
732
733
734
735
736
737
738
739
739
740
741
742
743
744
745
746
747
748
749
749
750
751
752
753
754
755
756
757
758
759
759
760
761
762
763
764
765
766
767
768
769
769
770
771
772
773
774
775
776
777
778
779
779
780
781
782
783
784
785
786
787
788
789
789
790
791
792
793
794
795
796
797
797
798
799
799
800
801
802
803
804
805
806
807
808
809
809
810
811
812
813
814
815
816
817
818
819
819
820
821
822
823
824
825
826
827
828
829
829
830
831
832
833
834
835
836
837
838
839
839
840
841
842
843
844
845
846
847
848
849
849
850
851
852
853
854
855
856
857
858
859
859
860
861
862
863
864
865
866
867
868
869
869
870
871
872
873
874
875
876
877
878
879
879
880
881
882
883
884
885
886
887
888
889
889
890
891
892
893
894
895
896
897
898
899
899
900
901
902
903
904
905
906
907
908
909
909
910
911
912
913
914
915
916
917
918
919
919
920
921
922
923
924
925
926
927
928
929
929
930
931
932
933
934
935
936
937
938
939
939
940
941
942
943
944
945
946
947
948
949
949
950
951
952
953
954
955
956
957
958
959
959
960
961
962
963
964
965
966
967
968
969
969
970
971
972
973
974
975
976
977
978
979
979
980
981
982
983
984
985
986
987
987
988
989
989
990
991
992
993
994
995
996
997
998
999
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1079
1080
1081
1082
1083
1084
1085
1086
1087
1087
1088
1089
1089
1090
1091
1092
1093
1094
1095
1095
1096
1097
1098
1099
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1138
1139
1140
1141
1142
1143
1144
1144
1145
1146
1147
1148
1149
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1187
1188
1189
1189
1190
1191
1192
1193
1194
1195
1195
1196
1197
1198
1199
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1287
1288
1289
1289
1290
1291
1292
1293
1294
1295
1295
1296
1297
1298
1299
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1387
1388
1389
1389
1390
1391
1392
1393
1394
1395
1395
1396
1397
1398
1399
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1488
1489
1489
1490
1491
1492
1493
1494
1495
1495
1496
1497
1498
1499
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1588
1589
1589
1590
1591
1592
1593
1594
1595
1595
1596
1597
1598
1599
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1688
1689
1689
1690
1691
1692
1693
1694
1695
1695
1696
1697
1698
1699
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1788
1789
1789
1790
1791
1792
1793
1794
1795
1795
1796
1797
1798
1799
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1888
1889
1889
1890
1891
1892
1893
1894
1895
1895
1896
1897
1898
1899
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1988
1989
1989
1990
1991
1992
1993
1994
1995
1995
1996
1997
1998
1999
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2088
2089
2089
2090
2091
2092
2093
2094
2095
2095
2096
2097
2098
2099
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2188
2189
2189
2190
2191
2192
2193
2194
2195
2195
2196
2197
2198
2199
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2229
2230
2231
2232
2233
2234
2235
223
```

```

115     }
116
117     return stats
118
119 # Usage example
120 def main():
121     # Initialize Logger
122     logger = AISystemLogger()
123
124     # Simulate requests
125     logger.log_request(
126         user_id="user_123",
127         prompt="What is GPT-4?",
128         response="GPT-4 is a large language model...",
129         tokens_used=150,
130         latency_ms=1200,
131         success=True
132     )
133
134     logger.log_request(
135         user_id="user_456",
136         prompt="Calculate something complex",
137         response=None,
138         tokens_used=50,
139         latency_ms=500,
140         success=False,
141         error_msg="Timeout"
142     )
143
144     logger.log_request(
145         user_id="user_789",
146         prompt="Write me an email",
147         response="Hello, I am writing to you...",
148         tokens_used=200,
149         latency_ms=1500,
150         success=True
151     )
152
153     # Display daily statistics
154     stats = logger.get_daily_stats()
155     print("\nDaily Statistics:")
156     print("=" * 50)
157     for key, value in stats.items():
158         if isinstance(value, float):
159             print(f"{key}: {value:.2f}")
160         else:
161             print(f"{key}: {value}")
162     print("=" * 50)
163
164 if __name__ == "__main__":
165     main()

```

פלט מצופה:

```
:scitsitats yliaD  
=====  
      51-10-5202 :etad  
      3 :stseuquer_latot  
      2 :stseuquer_lufsseccus  
          1 :stseuquer_deliaf  
          76.66 :etar_sseccus  
          004 :snekot_latot  
      33.331 :tseuquer_rep_snekot_gva  
      76.6601 :sm_ycnetal_gva  
=====
```

SHIPORIM APSEIYIM:

1. שליחת התרעות (kcalS/liamE) כшиб שגיאה
2. אחסון ב- esabataD במקום קובץ
3. draobhsaD בזמן אמת עם tilmaertS
4. ניתוח טרנדים לאורך זמן

9.1 סיכום

פרק זהלקח אתכם למסע המלא -- מזיהוי הזדמנויות IA בארגון, דרך COP ו-toliP-1, ועד לייצור מלא ותחזקה מתמדת. לממנו כיצד להרכיב צוות IA, הנהל פרויקט בגישה A, eligible, לישם עקרונות spOLM, ולמדוד הצלחה באמצעות IPK.

נקודות מפתח:

1. revocsiD **קפדי חוסך זמן וכסף** -- אל תדלגו עליו
2. COP **קטן ומהיר עדיף על פרויקט ענק ארוך**
3. **צוות היברידי (פנימי + חיצוני)** הוא לרוב הפתרון הטוב ביותר
4. eligible **מתאים ל-IA** -- איטרציות קצרות ומשוב מהיר
5. spOLM **הוא הכרח** -- בludeiy המערךת תקורס
6. IPK **המצביע** -- מה שלא נמדד, לא מנוהל

בסיומו של דבר, הצלחה בפרויקט IA היא לא עניין של טכנולוגיה בלבד. היא עניין של ניהול נכון, תוכנן מושכל, צוות טוב, ומידה מתמדת. כמו תוכנית אפולו ב-9691 -- הטכנולוגיה חשובה, אבל ניהול הוא שעושה את ההבדל.

אתם עכשו מוצדים בידע ובכלים להוביל פרויקט IA מוצלח בארגון שלכם. זכרו: התחלו קטן, והוכחו ערך, למדדו, שפרו, והרחיבו. בהצלחה במסע!

מקורות והמלצות לקריאה נוספת

koohS nhoJ -- "koobyalP IA naeL ehT" :IA rof eligA .1

[42] .la te lieverT kraM -- "spOLM gnicudortnl" :spOLM .2

[02] esoR guoD -- "ssenisuB rof IA" :tnemeganaM tcejorP IA .3

[81] retnemraP divaD -- "srotacidnI ecnamrofreP yeK" :sIPK .4

[32] tropeR xednI IA IAH drofnatS :seidutS esaC .5

מקורות

- ,gninnerG .J ,relwoF .M ,mahgninnuC .W ,nrubkcoC .A ,mukenneB nav .A ,eldeeB .M ,kceB .K [1]
.J ,rebawhcS .K ,rolleM .S ,nitram .C .R ,kciraM .B ,nreK .J ,seirffeJ .R ,tnuH .A ,htimshgiH .J
,1002 ,ecnaillA eligA .tnempoleveD erawtfoS eligA rof otsefinaM ,samohT .D dna ,dnalrehtuS
<https://agilemanifesto.org/> .:elbaliavA .[enilnO] .otsefinaM eligA lanigiro ehT
org/
- "adnegA hcraeseR A :smetsyS IA gnireenignE`` ,civoknrC .I dna ,nosslo .H .H ,hcsoB .J [2]
10.4018/978-1-7998-5101-1.ch001 :IOD .0202 ,91-1 .pp
- cirbuR A :erocS tseT LM ehT`` ,yellucS .D dna ,bilas .M ,nesleiN .E ,iaC .S ,kcerB .E [3]
:IOD .7102 ,2311-3211 .pp ",noitcudeR tbeD lacinhceT dna ssenidaeR noitcudorP LM rof
10.1109/BigData.2017.8258038
- ,lanoisseforP yelseW-nosiddA .tnempoleveD erawtfoS eligA roF :deilppA seirotS resU ,nhoC .M [4]
.1865021230-879 :NBSI ,4002
.E dna ,ovla□eP-a□craG .J .F ,atsocA-otoS .P ,sarerbmuL-odasaC .C ,soicalaP-omoloC .R [5]
,ydutS lanoitazinagro-itluM A :lennosreP erawtfoS ni spaG ecnetepmoC`` ,oraC-ravoT
10.1016/j. :IOD .3102 ,164-654 .pp ,2 .on ,92 .lov ,roivaherB namuH ni sretupmoC
chb.2012.04.021
- ,69 .lov ,8102 .dlroW laeR eht rof ecnegilletnI laicifitrA ,iknanoR .R dna tropnevaD .H .T [6]
.seigetarts noitatnemelpmi IA lacitcarP ,611-801 .pp
:seigolodohteM eligA fo edaceD A`` ,eoM .B .N dna ,yllapejilaB .V ,rureN .S ,ry□sgniD .T [7]
,58 .lov ,erawtfoS dna smetsyS fo lanruoJ ",tnempoleveD erawtfoS eligA gnnialpxE sdrawoT
10.1016/j.jss.2012.02.033 :IOD .2102 ,1221-3121 .pp ,6 .on
- tpecnoC no yevruS A`` ,aihcahcuoB .A dna ,yikzinehceP .M ,tefiB .A ,□tiaboil□ .I ,amaG .J [8]
10. :IOD .4102 ,73-1 .pp ,4 .on ,64 .lov ,syevruS gnitupmoC MCA ",noitatpadA tfirD
1145/2523813
- ,dliuB hguorht sesaeleR erawtfoS elbaileR :yrevileD suounitnoC ,yelraF .D dna elbmuH .J [9]
.9191061230-879 :NBSI ,0102 ,lanoisseforP yelseW-nosiddA .noitamotuA tnemyolpeD dna ,tseT
-dlroW etaerC ot woH :koobdmaH spOveD ehT ,elbmuH .J dna ,silliW .J ,siobeD .P ,miK .G [01]
,sserP noituloveR TI .snoitazinagrO ygolonhceT ni ytiruceS dna ,ytilibaileR ,ytiligA ssalC
.3008872491-879 :NBSI ,6102
- ,weivrevO :(spOLM) snoitarepO gnnraeL enihcaM`` ,lhcsriH .S dna ,lh□K .N ,regrebzuerK .D [11]
.3202 ,978 13-668 13 .pp ,11 .lov ,sseccA EEEI ",erutcetiherA dna ,noitinifeD

A :tfirD tpecnoC rednU gni.raeL `` ,gnahZ .G dna ,amaG .J ,uG .F ,gnoD .F ,uiL .A ,uL .J -6432 .pp ,21 .on ,13 .lov ,gnireenignE ataD dna egdelwonK no snoitcasnarT EEEI ",weiveR 10.1109/TKDE.2018.2876857 :IOD .9102 ,3632	[21]
erawtfoS fo ymonoxaT A `` ,civoknrC .I dna ,nossIO .H .H ,hcsoB .J ,jaR .A ,eratakawL .E .L -722 .pp ",noitagitsevnI lacipmE nA :smetsyS gni.raeL enihcaM rof segnellahC gnireenignE 10.1007/978-3-030-19034-7_14 :IOD .9102 ,342	[31]
noitcudorP gniveihcA :gnireenignE yilibavresbO ,adnariM .G dna ,senoJ-gnoF .L ,srojaM .C .N ,lluK .M ,ollarO-zednanreH .J ,irref .C ,odnahcO-sarertnoC .L ,demulP-zenitraM .F morF :retaL sraeY ytnewT MD-PSIRC `` ,hcalF .P dna ,anatniuQ-zerimaR .J .M ,ehcihcaL dna egdelwonK no snoitcasnarT EEEI ",seirotcejarT ecneicS ataD ot sessecorP gniniM ataD 10.1109/TKDE.2019.9038403 .pp ,8 .on ,33 .lov ,gnireenignE ataD 2962680	[41]
yesniKcM ",weiveR ni edaceD flaH a dna - 2202 ni IA fo etatS ehT `` ,ynapmoC & yesniKcM .eulav dna noitpoda IA no yevrus launnA ,2202 ,.peR .hceT ,etutitsnI labolG :gninraeL enihcaM gni.yolpeD ni segnellahC `` ,ecnerwaL .D .N dna ,amrU .G-.R ,seyelaP .A :IOD .2202 ,92-1 .pp ,6 .on ,55 .lov ,syevruS gnitupmoC MCA ",seidutS esaC fo yevruS A 10.1145/3533378	[61]
gninniW gnisU dna ,gnitnemelpmI ,gnipoleveD :srotacidnI ecnamrofreP yeK ,retnemraP .D .2015298111-879 :NBSI ,5102 ,yeliW .dr3 ,sIPK ,nati.nuoFaL .B dna ,uhC .M ,nolednaC .F ,noriK .D ,hednabadohK .S ,mahtobsnaR .S ,weiveR tnemeganaM naolS TIM ",gninraeL lanoitazinagrO htiW tcapmI s'IA gnidnapxE `` .GCB htiw yevrus IA launnA ,0202	[71]
.sredaeL ssenisuB dna sruenerpertnE rof IA ot ediuG lacitcarP A :ssenisuB rof IA ,esoR .D .9906724841-879 :NBSI ,1202 ,sserpA .murcS ot ediuG evitinifeD ehT :ediuG murcS ehT ,dnalrehtuS .J dna rebawhcS .K .ediug ygolodohtem murcS laiciffO ,0202 ,gro.murcS ,gnuoY .M ,yrahduahC .V ,renbE .D ,spillihP .T ,vodyvaD .E ,nivoloG .D ,tloH .G ,yellucS .D ,"smetsyS gni.raeL enihcaM ni tbeD lacinhceT neddiH `` ,nosinneD .D dna ,opserC .F-.J .segnellahc ecnanetniam metsys LM no repap lanimeS ,5102 ,1152-3052 .pp ,82 .lov -namuH rof etutitsnI drofnatS ",3202 tropeR xednI ecnegilletnI laicifitrA `` ,IAH drofnatS .troper sdnert IA launna evisneherpmoC ,3202 ,.peR .hceT ,ecnegilletnI laicifitrA deretneC ,ikazayiM .M ,ettollioVaL .A ,icitneZ .J ,nahP .D ,ervfeL .K ,canetS .C ,tnomO .N ,lieverT .M .esirpretnE eht ni gni.raeL enihcaM elacS ot woH :spOLM gnicudortnI ,nnamdieH .L dna .0923802941-879 :NBSI ,secitcarp spOLM ot ediug evisneherpmoC ,0202 ,aideM yllieR'O .M dna ,neddaM .S ,rebuH .S ,nahtanawsiv .S ,eeL .E-.W ,maynamarbuS .H ,katrav .M .6102 ,3-1 .pp ",tnemeganaM ledoM gni.raeL enihcaM rof metsyS A :BDledoM `` ,airahaZ 10.1145/2939502.2939516 :IOD	[81]
,"gniniM ataD rof ledoM ssecorP dradnatS a sdrawoT :MD-PSIRC `` ,ppiH .J dna htriW .R .gniniM ataD rof ssecorP dradnatS yrtsudnI-ssorC ,0002 ,.peR .hceT ,GA zneB-relmiaD .T ,gnihcruM .S ,jksniwnoK .A ,gnoH .A .S ,isdochG .A ,nosdivaD .A ,nehC .A ,airahaZ .M gninraeL enihcaM eht gnitareleccA `` ,ramuZ .C dna ,eiX .F ,ehkraP .M ,eivligO .P ,mydokyN .54-93 .pp ,8102 ,14 .lov ,4 ",wolfLM htiw elcycefiL	[62]
,"gniniM ataD rof ledoM ssecorP dradnatS a sdrawoT :MD-PSIRC `` ,ppiH .J dna htriW .R .gniniM ataD rof ssecorP dradnatS yrtsudnI-ssorC ,0002 ,.peR .hceT ,GA zneB-relmiaD .T ,gnihcruM .S ,jksniwnoK .A ,gnoH .A .S ,isdochG .A ,nosdivaD .A ,nehC .A ,airahaZ .M gninraeL enihcaM eht gnitareleccA `` ,ramuZ .C dna ,eiX .F ,ehkraP .M ,eivligO .P ,mydokyN .54-93 .pp ,8102 ,14 .lov ,4 ",wolfLM htiw elcycefiL	[72]