

GrokBot: סוכנים שעובדים כמו עמיתים, מהצ'אט ועד PR שנפתח בזמן שישנים

שני אנשי Cursor מדגימים בתוכנית MTS את GrokBot, את לולאת תיקון הבאגים
האוטומטית ואת הטענות על Grok 4.6

2026-10-08 · (פורסם מחדש ב-X) · Lauren (Tan?), Roshan

משך: כ-27 דקות · X

סיכום אוטומטי — AI Briefing (aibriefing.dev)

על המסמך הזה

זה ראיון שצולם באולפן של התוכנית MTS, יום אחרי ההשקה של GrokBot. מתראיינים בו שניים מ-Cursor: Lauren, מהנדסת שבנתה חלק גדול מהמוצר, ו-Roshan מצד המוצר. המראיינת לא מציגה את עצמה בשמה. לאורך הראיון יש שני דמואים חיים מתוך GrokBot. הכיתוב של ה-repost ב-X מציג את Lauren כ-"SpaceX AI engineer", אבל בתמליל היא אומרת במפורש שהיא עובדת ב-Cursor. ה-SpaceX AI team מוזכר רק כשותף לאימון מודלי Grok.

הטענה המרכזית: את ה-agents צריך לנהל כמו עמיתים ולא כמו כלים. לכל בוט יש זהות, מחשב משלו ו-routines, ומדברים איתו בממשק צ'אט פשוט. מכאן נגזרת גם טענה על תפקיד המהנדס, שהופך למנהל של צוות סוכנים, ועל כלכלת הטוקנים: מודל זול שמגיע לאותה איכות מאפשר להפעיל יותר סוכנים.

איך לקרוא את המסמך

- 1. **תקציר מנהלים:** הטענות המרכזיות, הראיות שלהן ומה שלא נאמר.
- 2. **מה זה GrokBot ואיך נולד:** PMF פנימי וה-design brief.
- 3. **דמו ראשון:** צוות בוטים עם chief of staff, ופוסט ב-LinkedIn שעלה בשידור חי.
- 4. **Benny ולולאת תיקון הבאגים:** משוב, cloud agents, triage, ובסוף PR.
- 5. **תפקיד המהנדס:** מנהל, head chef וקוד-בייס שמותאם לסוכנים.
- 6. **Grok 4.6 וכלכלת הטוקנים:** מספרי ה-benchmark שהוקראו בשידור.
- 7. **מה עושים עם זה:** צעדים מעשיים ושאלות פתוחות.
- 8. **מילון מונחים.**

הערות מקור המסמך הופק אוטומטית מהתמליל של ההקלטה ומפריימים של המסך. ה-ASR שיבש שמות: GrokBot מופיע בתמליל גם כ-Grock-Bot, GraphBot, ו-Rocklot, ו-Grok מופיע כ-Grock ו-Rock. ייתכן ששמות ומונחים נוספים שובשו. שם המשפחה Tan מופיע רק בכיתוב ה-repost. שם המשפחה של Roshan לא ודאי: בדמו מוצג פרופיל LinkedIn שנראה כמו roshanadani, אבל הוא קשה לקריאה. הציטוטים מועתקים מילה במילה מהתמליל, ונבדקו מול חותמות הזמן. לא הוספנו שום מידע ממקורות חיצוניים.

1. תקציר מנהלים

זה ראיון השקה ולא הרצאה טכנית. מה שמעניין בו הוא ה-workflow של Lauren מציגה בפועל, ופחות ההכרזה עצמה: משוב על המוצר הופך ל-PR בלי שאדם כותב שורת קוד, ותפקיד המהנדס מצטמצם לסקירה ולבקרת איכות. הסעיפים הבאים מסכמים את הטענות ואת מה שמגבה אותן.

- **GrokBot הוא ממשק צ'אט לסוכנים מתמידים:** לכל בוט מחשב משלו, הוא מחובר לכלים של המשתמש וממשיך לרוץ גם כשהמחשב של המשתמש כבוי [0:00, 3:01]. ההשראה לממשק הייתה [4:32] iMessage.
- **המוצר הוכיח את עצמו קודם בתוך Cursor:** אנשי go-to-market, ops, הנדסה ומוצר מצאו לו שימושים עוד לפני ההשקה [1:30]. לפי Roshan זה התהליך הקבוע אצלם: קודם מחפשים PMF פנימי.
- **סוכן coordinator מנהל צוות של סוכנים מתמחים:** ה-chief of staff בודק את הסטטוס של סוכני Marketing, Social, Shoppay ו-EM. בשידור עצמו סוכן ה-Marketing פרסם פוסט מחשבון ה-LinkedIn של Roshan ותייג את חשבון MTS הנכון [3:01–4:32].
- **לולאת תיקון באגים בזמן שישנים:** כל משוב על GrokBot מגיע לבוטים של Lauren. הם מבצעים triage, מחלקים משימות, מפעילים cloud agents של Cursor ומחזירים PR עם וידאו וצילום מסך. לדבריה היא רק עושה review [10:41, 12:13, 13:47].
- **GrokBot נבנה בעזרת Roshan: GrokBot** מתאר לולאת משוב שבה הסוכנים עושים self-triage לבאגים וכותבים קוד [10:41].
- **תפקיד המהנדס עובר לניהול ולהכנת הקוד-בייס:** לדברי Lauren, צריך לקודד את כללי הסגנון lint rules ו-constraints שמכשילים את ה-CI, כדי שגם סוכנים חלשים ייצרו קוד טוב. ריפקטורים שהיו לוקחים לצוות חודשים או שנים היא עשתה כמעט לבד [16:51].
- **Grok 4.6 הוא טיעון של עלות:** המראיית הקריאה תוצאות CursorBench 3.2: 70.8% בעלות ממוצעת של 2.81 למשימה, מול 70.5% ב-17.32 דולר ל-Fable 5 Max [21:32, 23:02]. הדוברים לא הציגו את הנתונים האלה בעצמם.
- **מה הלאה:** GrokBot הושק כ-beta. משתמשים מבקשים לא מעט תמיכה ב-iMessage, ו-Lauren כבר מימשה אותה בגרסה מוקדמת, אבל לא התחייבה לכך [20:00, 21:32].

מה לא נאמר

- **אין מדדי הצלחה ללולאת התיקון:** לא נאמר כמה מה-PRs האוטומטיים מתמזגים, כמה נדחים או כמה זמן לוקח ה-review.
- **אין מספרי אימוץ:** PMF פנימי וקבלה חיובית מתוארים במילים בלבד, בלי מספר משתמשים או שימוש.
- **אין מודל אבטחה:** בוט שמחובר ל-LinkedIn ומבצע תשלומים דרך Stripe מעלה שאלות של הרשאות, אישורים וביקורת. אף אחת מהן לא נידונה.
- **ה-benchmark לא מתועד:** המספרים הוקראו מהבמה. המתודולוגיה לא מוסברת, וגם יחידת העלות (2.81 בלי סימן מטבע) לא ברורה.
- **לא ברור מה זה "מחשב משלו":** לא נאמר אם מדובר ב-VM בענן, איך מנוהלות ההתחברויות וכמה זה עולה.
- **לא הוסבר הקשר בין Cursor ל-SpaceX AI,** וגם לא הוצגה התוצאה של משימת ה-Figma slides שהתחילה בדמו.

2. מה זה GrokBot ואיך נולד

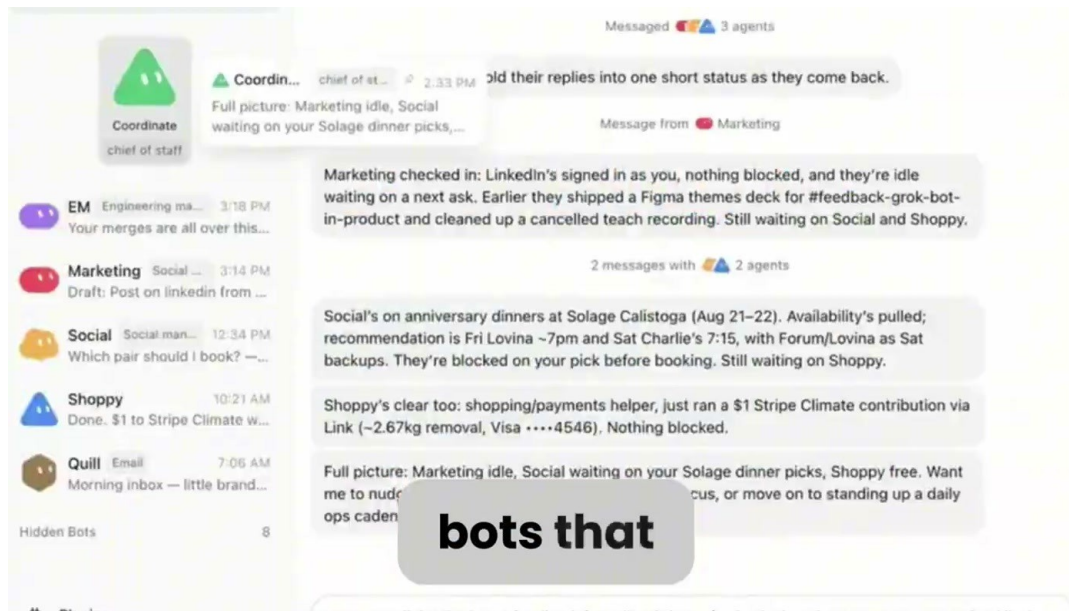
לפי Roshan, צוות קטן התחיל לבנות את GrokBot לשימוש עצמי, ומשם הוא התפשט בתוך החברה. כל מה ש-Cursor בונה עולה קודם לעובדים, שהם מגדירים "high taste", ומצפים לראות PMF פנימי [1:30]. מה שהיה שונה ב-GrokBot הוא שאנשים בכל תפקיד מצאו לו שימוש, כי הוא מתחבר לכלים שכבר עובדים איתם. Lauren, למשל, הזמינה דרכו כרטיסי IMAX ל-Odyssey [0:00].

את ה-design brief מסכם Roshan כך: "[7:35] keep it simple to be a little cheeky". לדבריו, בהרבה אפליקציות AI הכוח בא על חשבון הפשטות. Lauren מוסיפה ש-Cursor מלא כפתורים ומיועד ל-power users, ואילו GrokBot מתוכנן כעמית: "you don't necessarily want to micromanage your colleague and say, what".

[7:35] "tool calls are you using right now?" OpenClaw: הגדרה ביתית על גבי Telegram שלדבריה "breaks all the time" [7:35].

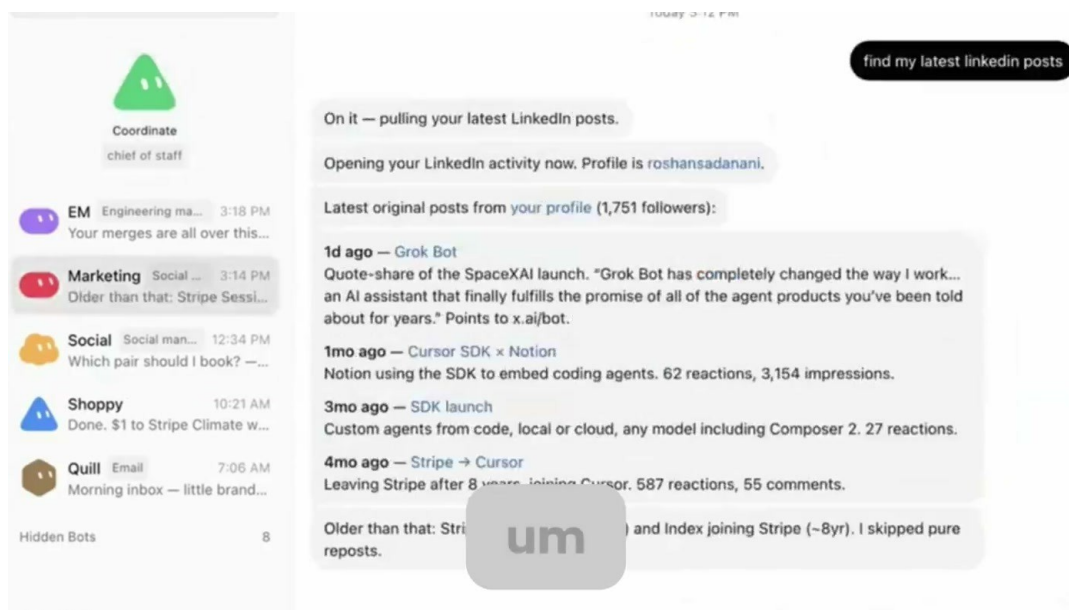
Lauren [0:00] and grokbot has its own computer so it's like super powerful it runs even if you're not — at your computer

3. דמו ראשון: צוות בוטים עם chief of staff

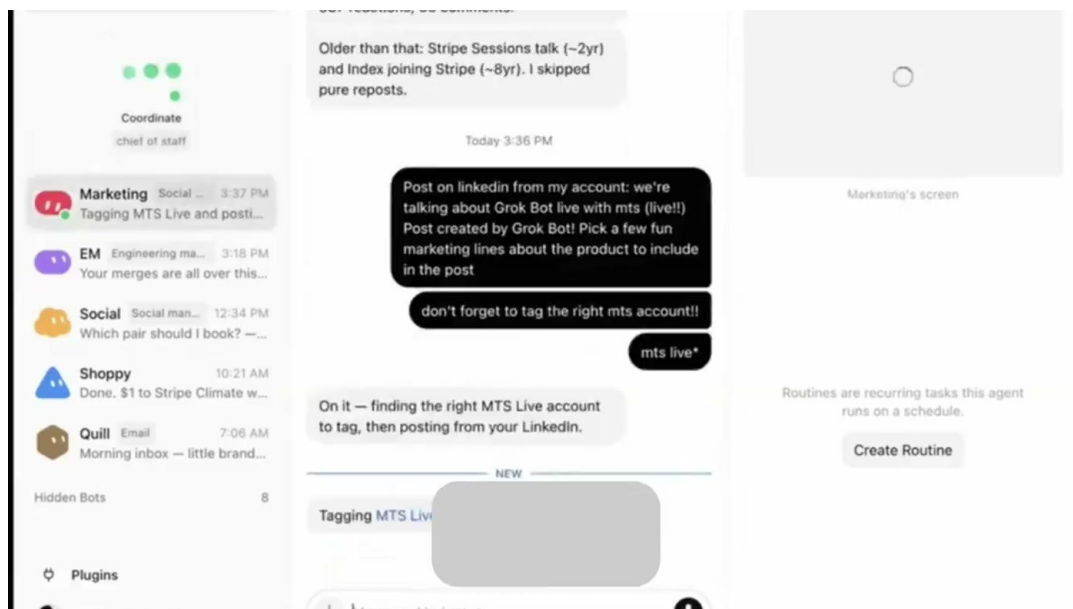


מסך ה-coordinator (chief of staff) מרכז את סטטוס הבוטים EM, Marketing, Social, Shoppy ו-Quill. רואים, למשל, ש-Marketing מחובר ל-LinkedIn ושסוכן Shoppy ביצע תרומה של 1 דולר דרך Stripe Climate.

Roshan פותח במסך הצוות שלו. ה-coordinator מכיר את היומן שלו, ויכול לשלוח הודעות לשאר הסוכנים ולבדוק מה מצבם. בדמו הוא פונה ל-Shoppy, בוט הקניות, ושואל על רכישות אחרונות [1:30]. אחר כך Roshan פותח את סוכן ה-Marketing, שהוקם ממש לפני השידור. יש לו מסך משלו, מחובר ל-LinkedIn של Roshan, ואת הפוסטים האחרונים הוא שולף פשוט בגלישה [3:01].

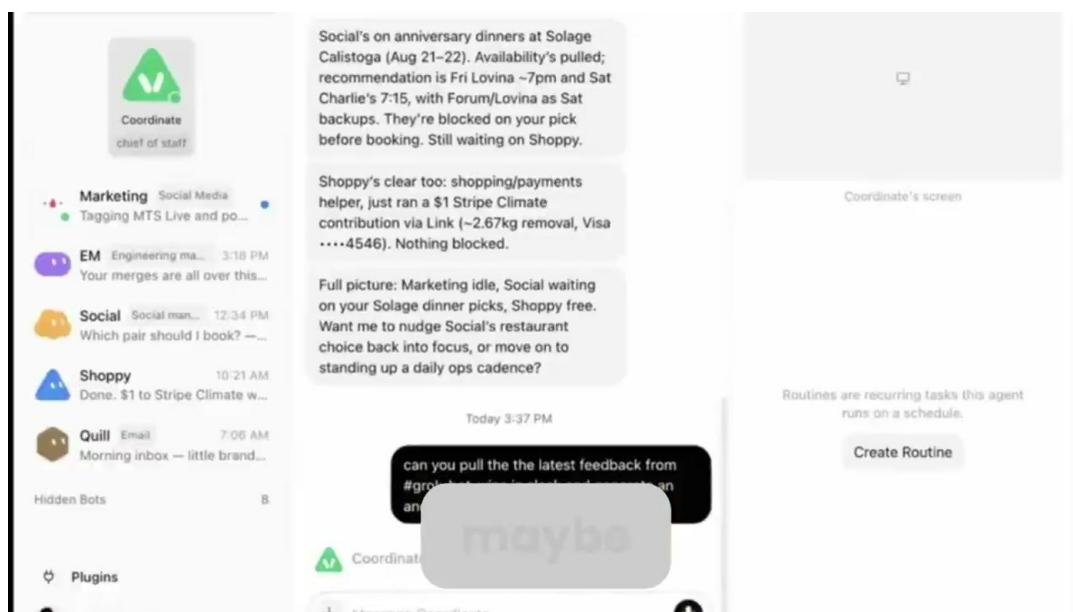


סוכן ה-Marketing עונה לבקשה find my latest linkedin posts ומחזיר רשימה של פוסטים מהפרופיל, כל אחד עם מספר תגובות וחשיפות. בצד מוצג חלוץ ה-screen של הסוכן.



הבקשה לפרסם פוסט ב-LinkedIn על השידור החי, עם התיקון `don't forget to tag the right mts account` שהוא מחפש את חשבון MTS Live הנכון לתיג ואחר כך מפרסם.

הפוסט עלה תוך דקה בערך, והחשבון הנכון תוג [4:32]. במקביל Roshan ביקש מה-`chief of staff` לאסוף משוב מערוץ Slack פנימי בשם `GrokBotWins` ולהכין ממנו מצגת ב-Figma slides [3:01]. התוצאה הזאת לא הוצגה בשידור. Roshan מסביר שהעיקרון הוא שיחה בזמן אמת, כמו עם עמית, `versus having to like wait your turn` [3:01].

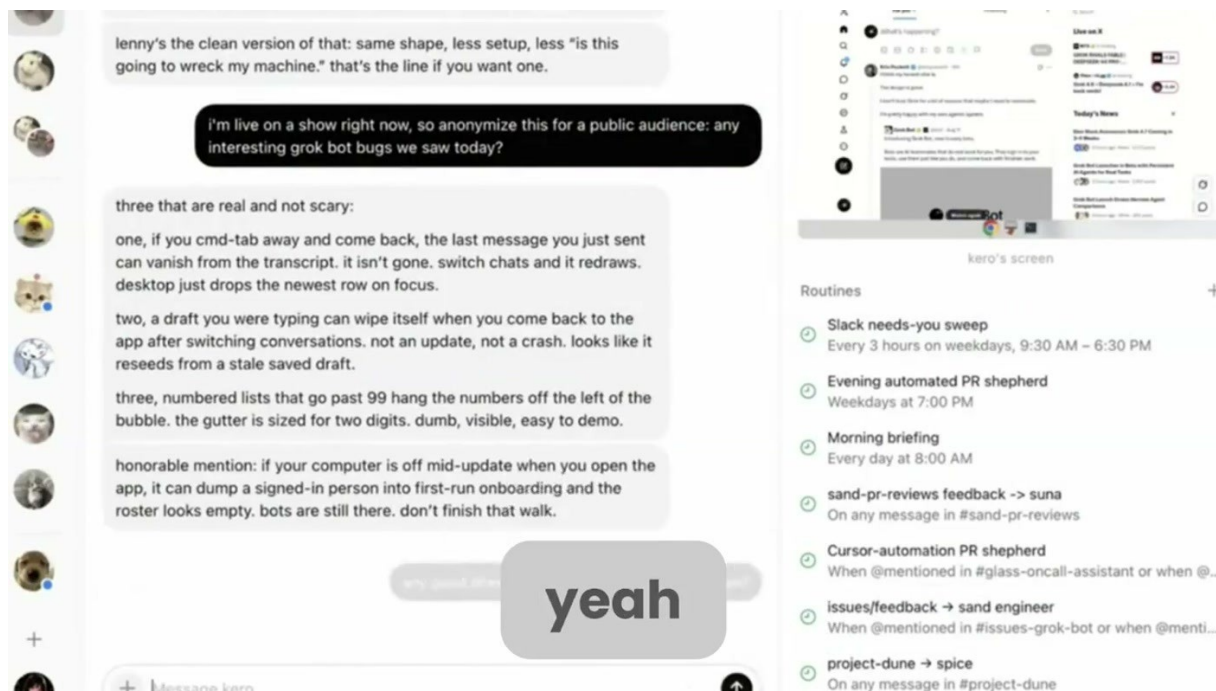


פנייה ל-coordinator לשלוח את המשוב האחרון מערוץ פנימי. בתחתית רואים את רשימת הבוטים ואת אפשרות ה-Plugins.

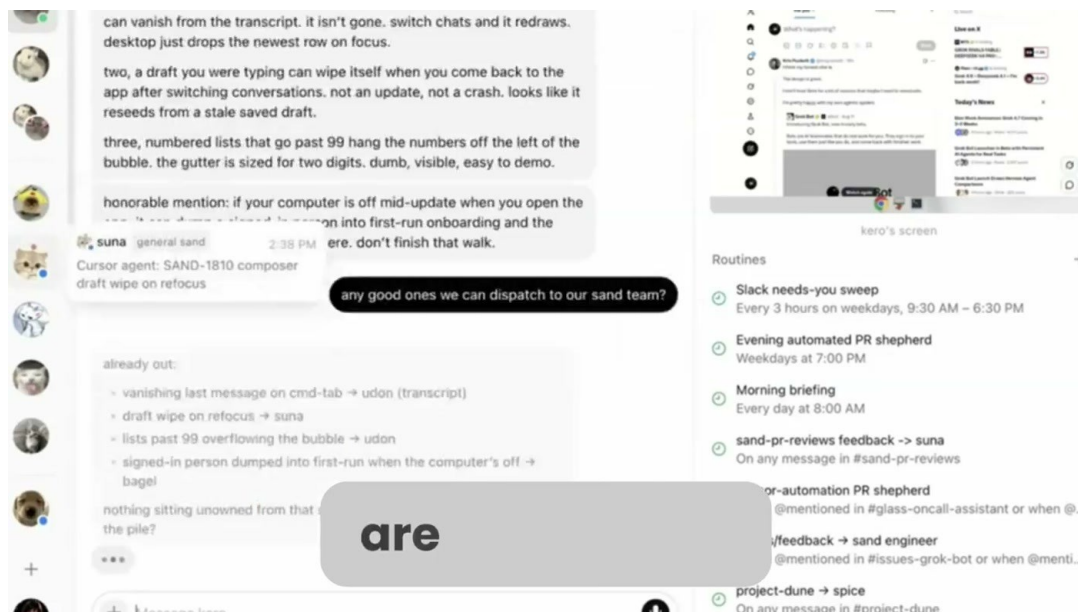
4. Benny ולולאת תיקון הבאגים האוטומטית

לפני `GrokBot` הייתה ל-Lauren בוט Slack פנימי בשם Benny, עם אווטאר של כלב. לפי Roshan הוא די מפורסם בחברה [9:10]. `Cursor` קיבלה הרבה דיווחי באגים, ו-Lauren, שמגדירה את עצמה `"fairly lazy"`, רצתה לתקן את כולם. הרבה עובדים שאלו אותה איך בונים בוט כזה, ומשם הגיע הרעיון של `GrokBot`: כל אחד מגדיר בוט עם זהות, מחשב, routines ואוטומציות [10:41]. היום, לדבריה, `"every time someone gives us feedback on GrokBot"`, `"I'm almost like a manager for my digital colleagues."` [10:41]. את

Lauren [10:41] and so I started thinking about how would I make it so my agents can just — automatically fix bugs for me while I sleep

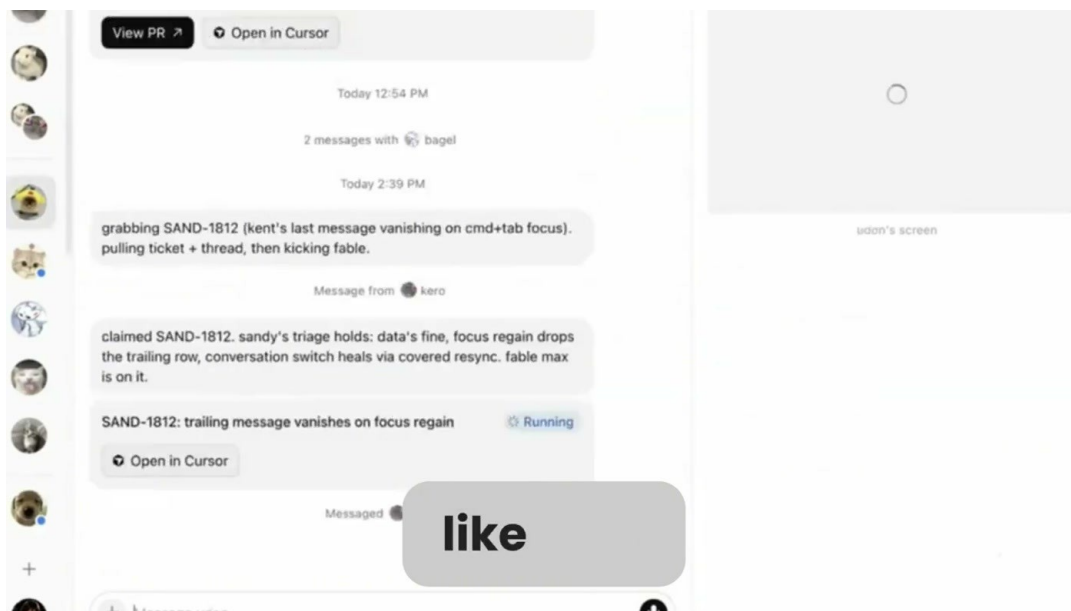


Lauren מבקשת מה- chief of staff רשימה אוניוויט של באגים מעניינים מהיום ומקבלת שלושה. בצד מופיעות ה-routines המוגדרות: Slack needs-you sweep, Evening automated PR shepherd, Morning briefing, issues/feedback → sand engineer ועוד.

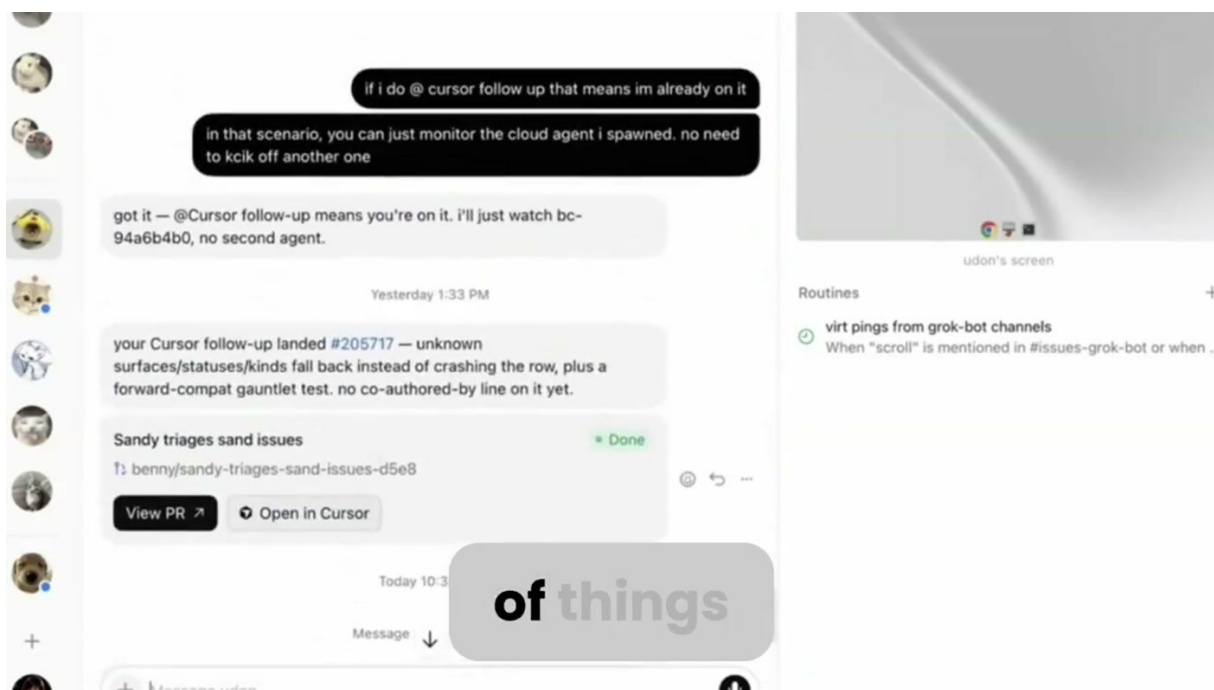


השלב הבא: any good ones we can dispatch to our sand team? ה-coordinator מחלק את הבאגים לסוכנים בצוות. בתמליל השם נשמע כמו SAN team, ונראה שזה שם פנימי.

בדמו השני ה-chief of staff כבר אסף כמה באגים מהיום, וכל סוכן בצוות אחראי על פיצ'ר אחר. לפי Lauren, בסוף כל משימה "I'll get like a pull request that i can review i'll get videos i'll get a screenshot" [12:13] מדגישה פיצ'ר שלא בולט במוצר: "if you set up cloud agents on Cursor, your GrokBots can actually kick them off." [13:47] כלומר, סוכן GrokBot מנהל cloud agent שרץ על מחשב נפרד. Roshan מוסיף ש-"It's actually remarkable how much of GrokBot was built with GrokBot, too." [10:41] החתולים Lauren הוסיפה סתם בשביל הכיף [13:47].



כרטיס SAND-1812 (הודעה אחרונה שנעלמה בחזרה לחלון עם cmd+tab). סוכן תופס את הכרטיס ומאשר את ה-triage, ומשימה בסטטוס Running נפתחת עם הכפתורים View PR ו-Open in Cursor.



תיאום בין GrokBot ל-cloud agent: Lauren עושה @Cursor follow-up, הבוט רק עוקב אחרי ה-agent הקיים ולא פותח עוד אחד. אחר כך מגיע דיווח ש-follow-up נכנס ל-PR, הכולל fallback לסטטוסים לא מוכרים ובדיקת forward-compat.

5. תפקיד המהנדס: מנהל, head chef וקוד-בייס שמותאם לסוכנים

Roshan מסכם את החזון במילה אחת: "I think that if I had to pick one word, it would be efficiency." [15:19]. לדבריו, בכל תפקיד יש צוואר בקבוק בדרך מרעיון למוצר, והמטרה היא לכווץ את המחזור הזה בעזרת צי של סוכנים שמייצרים עבודה ברמת ה-taste שלו.

Lauren נותנת את החלק המעשי. לדבריה, מהנדסים צריכים עכשיו להחזיק את הקוד-בייס במצב שבו גם PM ומעצבים יכולים לתרום קוד איכותי. זה אומר להשקיע ב-refactoring ואפילו בשכתוב, "so that agents can really thrive". [16:51] היא מעידה: "I've done a lot of big refactors and migrations at Cursor. And I did them" like mostly by myself (כשמישהו נוסף עזר לה), עבודה "which would have taken me or a team of".

Michelin kitchen: "And so engineers months or even years to finish" [16:51]
my role is becoming more like a head chef." [20:00]
הצוות היא מאמנת דרך skills וריפקטורים [20:00–18:23].

Lauren [16:51] things like encoding your rules and your style guides into actual like lint rules or —
hard constraints that you know fail ci

6. Grok 4.6 וכלכלת הטוקנים

ההשקה של GrokBot הגיעה יום אחרי ההכרזה על Grok 4.6. לפי Lauren, הוא "much more intelligent than 4.5 is at the same token cost" [4:32]. היא משתמשת בו הרבה לעבודה הנדסית, ובמקביל גם במודלים אחרים, "like Fable and Sol" [6:04]. השמות האלה כנראה שובשו בתמלול ולא ניתן לאמת אותם. Roshan מספר שכבר בהשקה של Grok 4.5 עם ה-SpaceX AI team הכוונה הייתה לאמן מודלים שמיועדים ליותר מקוד [6:04].

את המספרים הקריאה המראיית, ולא הדוברים. Lauren מודה בגלוי: "i work at cursor i have unlimited tokens" [23:02] "and so i can just you know you know go crazy with token maxing", לדבריה, רגישים לעלות. Roshan ממסגר את זה כחשבון פשוט: "if you have a budget of 20 bucks to do all of your AI tasks and the cost per task goes down, it's like you can just do more tasks." [24:35]

מודל (כפי שנשמע בתמליל)	ציון CursorBench 3.2	עלות ממוצעת למשימה
Grok 4.6 Extra High	70.8%	2.81 (מטבע לא צוין)
Fable 5 Max (השם לא ודאי)	70.5%	17.32 דולר

7. מה עושים עם זה

- מתחילים מלולאה אחת שאפשר למדוד: למשל משוב, triage ו-PR. כדאי לספור כמה PRs מתמזגים, משהו שבשידור לא נאמר.
- מכינים את הקוד-בייס לסוכנים: הופכים style guides ל-lint rules ולבדיקות שמכשילות CI, לפני שמגדילים את מספר הסוכנים.
- מפרידים בין תפקידים: coordinator אחד שמחלק משימות, וסוכנים מתמחים לפי פיצ'ר או תחום, כמו בצוות של Lauren.
- מגדירים routines ולא רק שיחות: sweeps, briefings ו-PR shepherd רצים לפי לוח זמנים, וזה מה שהופך בוט למתמיד.
- מוודאים שאין כפילויות בין סוכנים: הכלל ש-Cursor follow-up@ פירושו לא לפתוח עוד agent הוא דוגמה טובה למוסכמה כזאת.
- בודקים את מודל העלות בעצמכם: מריצים את Grok 4.6 מול המודל הנוכחי על משימות אמיתיות, ולא מסתמכים על מספרים שהוקראו בשידור.

שאלה פתוחה	למה זה חשוב
כמה מה-PRs האוטומטיים מתמזגים בלי תיקון?	זה מה שקובע אם הלולאה חוסכת זמן או רק מעבירה את העבודה ל-review.
איך מנוהלות ההרשאות של בוט שמחובר ל-LinkedIn ול-Stripe?	פעולות בשם המשתמש, כולל תשלומים, דורשות אישור וביקורת.
מה זה בפועל "מחשב משלו" ומה העלות שלו?	זה משפיע על התמחור ועל ההחלטה אם להחזיק הרבה בוטים במקביל.
על איזו מתודולוגיה מבוסס CursorBench 3.2?	בלי מתודולוגיה אי אפשר להשוות את 70.8% ל-70.5%.
מה הקשר בין Cursor ל-SpaceX AI במודל Grok?	זה משפיע על זמינות המודל ועל מפת הדרכים.

8. מילון מונחים

מונח	פירוש
agent	מודל שמבצע משימות בעצמו באמצעות כלים

פירוש	מונח
סוכן מתאם שמחלק משימות לסוכנים אחרים ועוקב אחרי הסטטוס שלהם	coordinator / chief of staff
סוכן קוד של Cursor שרץ בענן על מחשב משלו	cloud agent
משימה מתוזמנת שהבוט מריץ בעצמו לפי לוח זמנים	routine
התאמה בין המוצר לשוק. כאן גם במובן של אימוץ פנימי בחברה	PMF (product-market fit)
מיון באגים לפי חשיבות והחלטה מי מטפל בהם	triage
בקשה למזג שינוי קוד, שעוברת review	PR (pull request)
כללי סגנון שנבדקים אוטומטית, ו-pipeline שנכשל כשהם מופרים	lint rules / CI
שינוי מבנה הקוד או מעבר טכנולוגי בלי לשנות את ההתנהגות	refactor / migration
שימוש בטוקנים בלי לחשוב על העלות	token maxing
מבחן השוואתי למודלים. כאן CursorBench 3.2	benchmark
מודל מהשורה הראשונה מבחינת יכולות	frontier model