

KURSOR

Sistem Kontrol Kursor Inovatif berbasis Gestur Tangan dan Kepala untuk Aksesibilitas Digital (Proyek DAMAR - Dukungan Aplikasi Menuju Aksesibilitas Rata)

Kata Kunci: Aksesibilitas Digital, Interaksi Manusia-Komputer, Kontrol Gestur, MediaPipe, OpenCV, Teknologi Asistif, Python, Proyek DAMAR, L1O

Nama Proyek:	KURSOR
Pengusul Utama:	Ardelyo (L1O)
Kolaborator:	OurCreativity
Inspirasi & Pembimbing:	Pak Dadan Mochammad Ramdhan
Konteks Proyek:	Proyek DAMAR (Dukungan Aplikasi Menuju Aksesibilitas Rata)
Tanggal Pengajuan:	24 Mei 2024 (Asumsi)
Versi Dokumen:	1.0

2. Abstrak

2.1. Abstrak Eksekutif

Interaksi manusia-komputer (IMK) merupakan aspek fundamental dalam penggunaan teknologi digital. Namun, metode input konvensional seperti mouse dan keyboard seringkali menjadi penghalang bagi individu dengan keterbatasan fisik atau motorik. Proyek "Kursor" bertujuan untuk mengembangkan sebuah sistem kontrol kursor inovatif yang memanfaatkan gestur tangan dan gerakan kepala sebagai alternatif input. Dibangun menggunakan teknologi visi komputer canggih dari **MediaPipe** dan

OpenCV dalam lingkungan Python, **Kursor** menawarkan dua mode kontrol utama: mode tangan yang melacak jari untuk berbagai aksi mouse, dan mode kepala yang menggunakan fitur wajah sebagai pointer dengan dukungan dwell-click dan aksi berbasis kedipan mata. Aplikasi ini juga dilengkapi dengan keyboard virtual yang dapat diakses melalui metode dwell-hover, meningkatkan kemandirian pengguna. **Kursor** dirancang untuk meningkatkan aksesibilitas digital, memberikan solusi intuitif dan adaptif bagi pengguna dengan kebutuhan khusus, serta berkontribusi pada penelitian di bidang teknologi asistif. Proyek ini, sebagai bagian dari inisiatif DAMAR (Dukungan Aplikasi Menuju Aksesibilitas Rata), diharapkan dapat memberikan dampak positif yang signifikan dalam mempromosikan inklusivitas di era digital.

2.2. Abstrak Teknis

Proyek "**Kursor**" mengimplementasikan sistem kontrol kursor berbasis visi komputer. Sistem ini memanfaatkan pustaka **MediaPipe** untuk deteksi dan pelacakan landmark tangan (Hand Landmarker) dan wajah (Face Mesh) secara real-time melalui input webcam. Untuk mode tangan, landmark spesifik seperti ujung jari telunjuk (landmark #8) digunakan sebagai pointer, sementara konfigurasi jari-jari lainnya (misalnya, pinch antara ibu jari dan telunjuk, atau ibu jari dan jari tengah) diterjemahkan menjadi aksi klik kiri, klik kanan, scroll, dan drag. Pengaturan sensitivitas, threshold gestur, dan parameter lainnya dapat dikonfigurasi melalui antarmuka grafis (GUI) berbasis **Tkinter** dan disimpan dalam format JSON. Mode kepala menggunakan landmark wajah (misalnya, hidung, landmark #1) sebagai pointer, dengan implementasi dwell-click untuk klik kiri dan deteksi kedipan mata (berdasarkan Eye Aspect Ratio - EAR) untuk aksi yang dapat dikustomisasi. Sebuah keyboard virtual QWERTY juga diintegrasikan, memungkinkan input teks melalui dwell-hover pada tombol virtual yang ditampilkan di layar. Arsitektur perangkat lunak terdiri dari modul-modul utama: `config_manager.py` untuk manajemen konfigurasi, `tracker.py` (HandTracker dan FaceTracker), `mouse_handler.py` untuk interaksi dengan sistem operasi (PyAutoGUI), `virtual_keyboard.py`, dan `gui_settings.py`. Proyek ini dikembangkan menggunakan Python, dengan OpenCV untuk pemrosesan citra, NumPy untuk operasi numerik, dan PyAutoGUI untuk emulasi input mouse dan keyboard.

3. Pendahuluan

3.1. Latar Belakang

Di era digital saat ini, kemampuan untuk berinteraksi secara efektif dengan komputer menjadi krusial dalam berbagai aspek kehidupan, mulai dari pendidikan, pekerjaan, hingga hiburan. Metode input standar seperti mouse dan keyboard fisik, meskipun umum digunakan, dapat menjadi tantangan signifikan bagi individu dengan keterbatasan fisik, cedera, atau kondisi neurologis yang mempengaruhi kontrol motorik halus. Keterbatasan ini dapat menghambat partisipasi penuh mereka dalam masyarakat digital.

Teknologi asistif telah berkembang pesat untuk mengatasi tantangan ini, menawarkan berbagai solusi alternatif. Salah satu pendekatan yang menjanjikan adalah penggunaan visi komputer untuk menerjemahkan gerakan tubuh atau ekspresi wajah menjadi perintah kontrol komputer. Dengan kemajuan dalam algoritma machine learning dan ketersediaan pustaka seperti MediaPipe dari Google, deteksi dan pelacakan gestur secara real-time menjadi semakin akurat dan mudah diakses.

Proyek "Kursor" diinspirasi oleh kebutuhan akan solusi kontrol kursor yang lebih intuitif, adaptif, dan aksesibel. Dengan memanfaatkan kekuatan gestur tangan alami dan gerakan kepala, Kursor bertujuan untuk memberdayakan pengguna dengan berbagai tingkat kemampuan fisik untuk berinteraksi dengan komputer secara lebih mandiri dan efisien.

Inisiatif ini sejalan dengan semangat proyek DAMAR (Dukungan Aplikasi Menuju Aksesibilitas Rata) yang berfokus pada pengembangan solusi teknologi yang bermanfaat bagi masyarakat luas, khususnya dalam meningkatkan kualitas hidup dan inklusivitas.

3.2. Rumusan Masalah

Proyek ini berupaya menjawab beberapa tantangan dan pertanyaan utama:

- Bagaimana merancang dan mengimplementasikan sistem kontrol kursor yang andal dan responsif menggunakan gestur tangan dan gerakan kepala melalui webcam standar?
- Bagaimana mengintegrasikan berbagai aksi mouse (gerakan, klik kiri/kanan, double klik, drag, scroll) secara intuitif melalui gestur yang mudah dipelajari dan dilakukan?
- Bagaimana menyediakan mekanisme input teks yang aksesibel melalui keyboard virtual yang dapat dikontrol dengan metode yang sama (misalnya, dwell-hover)?
- Bagaimana memastikan sistem dapat dikonfigurasi dengan mudah oleh pengguna untuk menyesuaikan sensitivitas, preferensi gestur, dan parameter lainnya sesuai kebutuhan individu?

- Seberapa efektif sistem Kursor dalam meningkatkan aksesibilitas komputer bagi pengguna dengan keterbatasan fisik dibandingkan metode input konvensional atau solusi asistif lainnya? (Evaluasi ini mungkin di luar lingkup pengembangan awal, namun menjadi pertimbangan desain).

3.3. Tujuan Proyek

Tujuan utama dari proyek "Kursor" adalah:

1. **Mengembangkan Aplikasi Fungsional:** Membuat aplikasi Kursor yang stabil dan berfungsi penuh.
2. **Implementasi Mode Ganda:** Menyediakan mode kontrol tangan dan kepala.
3. **Integrasi Keyboard Virtual:** Mengintegrasikan keyboard QWERTY yang aksesibel.
4. **Konfigurabilitas Tinggi:** Merancang sistem konfigurasi yang fleksibel (GUI & JSON).
5. **Dokumentasi Komprehensif:** Menyediakan dokumentasi teknis dan pengguna.
6. **Kontribusi pada Proyek DAMAR:** Menghasilkan produk teknologi bermanfaat.

3.4. Manfaat Proyek

- Peningkatan Aksesibilitas bagi individu dengan disabilitas motorik.
- Peningkatan Kemandirian pengguna dalam tugas komputasi.
- Solusi Inovatif untuk kontrol komputer hands-free.
- Kontribusi Edukasi dan Penelitian di bidang IMK dan teknologi asistif.
- Pemberdayaan Komunitas melalui teknologi inklusif.

4. Tinjauan Pustaka

Interaksi Manusia-Komputer (IMK) terus berkembang, mencari cara yang lebih alami dan efisien bagi manusia untuk berkomunikasi dengan mesin. Visi komputer telah muncul sebagai bidang kunci dalam evolusi ini, memungkinkan komputer untuk "melihat" dan menginterpretasikan dunia visual. Dalam konteks teknologi asistif, visi komputer menawarkan potensi besar untuk menerjemahkan gerakan fisik atau ekspresi menjadi perintah kontrol.

Salah satu teknologi terdepan dalam hal ini adalah **MediaPipe** dari Google. MediaPipe menyediakan serangkaian solusi machine learning lintas platform yang siap pakai untuk deteksi wajah, pelacakan tangan, estimasi pose, dan lainnya. Solusi Hand Landmarker-nya mampu mendeteksi hingga 21 titik kunci (landmark) 3D pada tangan, sementara Face Mesh dapat mendeteksi ratusan landmark pada wajah. Akurasi dan performa real-time dari MediaPipe menjadikannya pilihan ideal untuk aplikasi kontrol berbasis gestur seperti Cursor.

Penggunaan gestur tangan untuk kontrol komputer bukanlah konsep baru. Berbagai penelitian telah mengeksplorasi pengenalan gestur statis dan dinamis. Cursor mengadopsi pendekatan berbasis landmark, di mana posisi relatif dari landmark jari (misalnya, jarak antara ujung ibu jari dan telunjuk) digunakan untuk memicu aksi seperti klik. Pendekatan ini umumnya lebih robust terhadap variasi kecil dalam penampilan tangan dibandingkan dengan metode pengenalan gestur berbasis template citra penuh.

Kontrol berbasis gerakan kepala juga telah banyak diteliti, seringkali menggunakan hidung atau titik tengah antara mata sebagai pointer. Aksi klik dalam sistem seperti ini sering diimplementasikan melalui dwell-time, di mana cursor yang diam pada satu titik selama durasi tertentu akan memicu klik. Alternatif lain adalah deteksi kedipan mata, yang dapat diukur menggunakan metrik seperti Eye Aspect Ratio (EAR), yang menghitung rasio antara jarak vertikal dan horizontal landmark mata.

Integrasi keyboard virtual adalah komponen penting untuk menyediakan solusi input yang komprehensif. Keyboard virtual yang ditampilkan di layar dapat dioperasikan melalui dwell-hover, di mana menahan pointer di atas sebuah tombol selama interval waktu tertentu akan "menekan" tombol tersebut.

Proyek Cursor membangun di atas fondasi penelitian dan teknologi yang sudah ada ini, dengan fokus pada integrasi fitur-fitur tersebut ke dalam satu aplikasi yang kohesif, dapat dikonfigurasi, dan ramah pengguna, dengan memanfaatkan pustaka Python seperti **OpenCV** untuk pemrosesan citra, **NumPy** untuk manipulasi data, dan **PyAutoGUI** untuk interaksi dengan sistem operasi.

5. Metodologi Pengembangan

Proyek Cursor akan dikembangkan menggunakan pendekatan iteratif dan modular, berdasarkan fondasi kode "Cursor (kode basis awal)" yang telah ada. Fokus utama adalah pada penyempurnaan, integrasi fitur, pengujian, dan dokumentasi.

5.1. Tahapan Pengembangan

1. Fase 1: Analisis Kebutuhan dan Studi Literatur Lanjutan (Penyempurnaan)

- Menganalisis kode sumber Kursor (kode basis awal).
- Identifikasi perbaikan, optimasi, fitur baru.
- Studi literatur tambahan (teknik gestur, UI/UX asistif).
- Spesifikasi detail fungsionalitas Kursor.
- *Luaran:* Dokumen spesifikasi kebutuhan dan desain sistem diperbarui.

2. Fase 2: Desain Sistem dan Antarmuka (Iterasi)

- Desain arsitektur final (modular, skalabel).
- Penyempurnaan GUI pengaturan (Tkinter).
- Desain tata letak keyboard virtual.
- Finalisasi set gestur tangan dan kepala.
- *Luaran:* Mockup UI/UX (konseptual), diagram arsitektur, definisi gestur final.

Mockup UI/UX untuk GUI Pengaturan dan Keyboard Virtual akan disajikan di sini.

3. Fase 3: Implementasi dan Pengembangan Modul (Iterasi pada Kode Ada)

- Refaktorisasi & optimasi kode Kursor (kode basis awal).
- Implementasi/penyempurnaan modul HandTracker & FaceTracker.
- Pengembangan/penyempurnaan MouseHandler (termasuk cooldown).
- Pengembangan keyboard virtual (dwell-typing, interaksi pyautogui).
- Pengembangan/penyempurnaan GUI pengaturan & manajemen konfigurasi.
- *Luaran:* Kode sumber Kursor terimplementasi & teruji unit.

4. Fase 4: Integrasi Sistem dan Pengujian Awal

- Integrasi semua modul menjadi aplikasi Kursor.

- Pengujian fungsional menyeluruh.
- Pengujian performa (FPS, latensi).
- Debugging dan perbaikan bug.
- *Luaran:* Versi alpha/beta aplikasi Kursor stabil.

5. Fase 5: Pengujian Pengguna dan Evaluasi (Jika Memungkinkan)

- Skenario pengujian pengguna.
- Melibatkan pengguna target (jika memungkinkan).
- Pengumpulan umpan balik (usabilitas, intuitivitas).
- Analisis hasil pengujian.
- *Luaran:* Laporan pengujian pengguna, rekomendasi perbaikan.

6. Fase 6: Iterasi Penyempurnaan dan Finalisasi

- Perbaikan berdasarkan umpan balik Fase 5.
- Optimasi akhir kode dan performa.
- *Luaran:* Versi rilis kandidat aplikasi Kursor.

7. Fase 7: Dokumentasi dan Pelaporan Akhir

- Dokumentasi pengguna (instalasi, penggunaan, kustomisasi).
- Dokumentasi teknis (arsitektur, modul).
- Laporan akhir proyek.
- *Luaran:* Dokumentasi lengkap, laporan akhir.

5.2. Teknologi yang Digunakan

- **Bahasa Pemrograman:** Python 3.7+
- **Visi Komputer & Pemrosesan Citra:**
 - **OpenCV (opencv-python):** Akuisisi frame, pemrosesan citra.
 - **MediaPipe (mediapipe):** Deteksi & pelacakan landmark.
- **Emulasi Input:** **PyAutoGUI (pyautogui):** Kontrol sistem.
- **Manipulasi Data:** **NumPy (numpy):** Operasi array.

- **Antarmuka Grafis (GUI):** Tkinter (Python).
- **Manajemen Konfigurasi:** Modul json (Python).
- **Lainnya:** Modul os, time, math.

6. Rencana Implementasi dan Jadwal Proyek

Estimasi jadwal ini mengasumsikan proyek dimulai dari adaptasi kode "Kursor (kode basis awal)".

6.1. Rincian Fase dan Aktivitas (Estimasi Durasi)

No	Fase	Aktivitas Utama	Estimasi Durasi
1	Inisiasi & Perencanaan Lanjut	Analisis kode Kursor (kode basis awal), finalisasi ruang lingkup, perencanaan detail.	1 Minggu
2	Desain Sistem & Antarmuka	Arsitektur, UI/UX GUI & Keyboard, finalisasi gestur.	1-2 Minggu
3	Pengembangan Modul Inti	Refaktorisasi, HandTracker, FaceTracker, MouseHandler.	2-3 Minggu
4	Pengembangan Keyboard & GUI	Implementasi VirtualKeyboard, gui_settings.py, config_manager.py.	2-3 Minggu
5	Integrasi & Pengujian Awal	Penggabungan modul, pengujian fungsional, debugging.	1-2 Minggu
6	Pengujian Pengguna & Umpan Balik	(Opsional) Skenario, sesi pengujian, analisis umpan balik.	1-2 Minggu
7	Iterasi Penyempurnaan	Perbaikan berdasarkan pengujian, optimasi akhir.	1-2 Minggu
8	Dokumentasi & Pelaporan	Dokumentasi pengguna, teknis, laporan akhir.	1-2 Minggu

No	Fase	Aktivitas Utama	Estimasi Durasi
Total Estimasi Durasi Proyek			10-17 Minggu

6.2. Diagram Gantt (Deskriptif)

Minggu Ke-	Fase Terkait (Merujuk pada Tabel Rincian Fase)	Fokus Utama / Aktivitas Paralel
1	Fase 1	Analisis kode basis, finalisasi ruang lingkup, perencanaan detail.
2-3	Fase 2, Mulai Fase 3	Desain arsitektur, UI/UX, finalisasi gestur; Memulai refaktorisasi & pengembangan tracker.
4-5	Lanjutan Fase 3, Mulai Fase 4	Pengembangan modul inti (HandTracker, FaceTracker, MouseHandler); Memulai pengembangan Keyboard Virtual & GUI.
6-7	Lanjutan Fase 4, Penyelesaian Fase 3	Finalisasi Keyboard Virtual & GUI; Penyelesaian modul inti dan integrasi awal.
8-9	Fase 5	Integrasi sistem menyeluruh, pengujian fungsional dan performa, debugging.
10-11	(Opsional) Fase 6	Pelaksanaan pengujian pengguna (jika dilakukan), pengumpulan dan analisis umpan balik.

Minggu Ke-	Fase Terkait (Merujuk pada Tabel Rincian Fase)	Fokus Utama / Aktivitas Paralel
12-13	Fase 7 (jika Fase 6 dilakukan, mungkin tumpang tindih atau mengikuti)	Iterasi perbaikan berdasarkan hasil pengujian, optimasi akhir kode dan performa.
14-15 (atau lebih awal/kemudian tergantung fase opsional)	Fase 8	Penyusunan dokumentasi pengguna dan teknis, pembuatan laporan akhir proyek.

Catatan: Beberapa aktivitas dalam fase yang berbeda dapat berjalan secara paralel untuk efisiensi waktu. Durasi total (10-17 minggu) memperhitungkan fleksibilitas ini dan fase opsional.

Representasi visual Diagram Gantt (misalnya, gambar atau grafik interaktif) akan ditempatkan di sini.

7. Struktur Tim dan Peran

- **Ardelyo (L10) - Pengembang Utama / Arsitek Sistem:**
 - Arsitektur teknis Cursor.
 - Pengembangan, implementasi, refaktorisasi kode inti.
 - Integrasi modul.
 - Pengujian teknis & debugging.
 - Dokumentasi teknis.
- **OurCreativity - Kolaborator (Desain UI/UX, Pengujian, Dokumentasi):**
 - Desain UI/UX (GUI Pengaturan, tema Keyboard Virtual).
 - Masukan perspektif pengguna.
 - Penyusunan skenario & pelaksanaan pengujian pengguna (jika dilakukan).
 - Dokumentasi pengguna & materi pendukung.

- Diseminasi & promosi proyek.
- **Pak Dadan Mochammad Ramdhan - Pembimbing / Penasihat Proyek:**
 - Bimbingan teknis & non-teknis.
 - Arahan strategis.
 - Bantuan mengatasi tantangan.
 - Validasi luaran proyek.

8. Luaran Proyek (Deliverables)

1. Aplikasi Kursor:

- Kode sumber lengkap (Python).
- Versi aplikasi stabil & dapat dijalankan.
- File konfigurasi default (`settings.json`).

2. Dokumentasi Teknis:

- Deskripsi arsitektur sistem (termasuk diagram).
- Penjelasan fungsionalitas modul utama.
- Panduan pengembang.

3. Dokumentasi Pengguna:

- Panduan instalasi.
- Panduan penggunaan (mode tangan & kepala).
- Penjelasan gestur.
- Panduan keyboard virtual & GUI pengaturan.
- Tips troubleshooting.

4. Laporan Akhir Proyek:

- Dokumen komprehensif siklus hidup proyek, termasuk hasil pengujian (jika ada) dan analisis.

5. (Opsional) Materi Presentasi/Demonstrasi Proyek.

9. Potensi Risiko dan Mitigasi

Risiko	Kategori	Probabilitas	Dampak	Strategi Mitigasi
Akurasi deteksi gestur kurang optimal	Teknis	Sedang	Tinggi	Penyesuaian parameter MediaPipe, fine-tuning threshold, filtering/smoothing, dokumentasi kondisi ideal penggunaan.
Kinerja aplikasi lambat (FPS rendah)	Teknis	Sedang	Sedang	Optimasi kode, penggunaan algoritma yang lebih efisien, penyesuaian resolusi kamera, rekomendasi spesifikasi hardware minimum.
Masalah kompatibilitas OS/hardware	Teknis	Rendah	Sedang	Pengujian pada berbagai konfigurasi OS dan hardware yang umum, penggunaan pustaka cross-platform.
Kesulitan integrasi keyboard virtual dengan berbagai aplikasi	Teknis	Rendah	Sedang	Desain modular yang baik, pengujian unit intensif, penggunaan metode emulasi input yang robust (PyAutoGUI).
Keterbatasan waktu pengembangan	Proyek & Manajemen	Sedang	Tinggi	Perencanaan yang realistis dan terperinci, prioritisasi fitur inti, komunikasi tim yang efektif, pemanfaatan maksimal dari kode basis awal Cursor.
Kurangnya sumber daya untuk pengujian pengguna yang representatif	Proyek & Manajemen	Tinggi	Sedang	Pengujian internal yang komprehensif oleh tim, pengumpulan umpan balik dari komunitas terbatas, atau merencanakan pengujian

Risiko	Kategori	Probabilitas	Dampak	Strategi Mitigasi
				pengguna sebagai fase lanjutan pasca-rilis awal.
Kurva belajar yang curam bagi pengguna baru	Pengguna	Sedang	Sedang	Dokumentasi pengguna yang jelas dan mudah dipahami, tutorial video (jika memungkinkan), desain gestur yang intuitif, antarmuka konfigurasi yang ramah pengguna.
Frustrasi pengguna akibat kesalahan deteksi atau lag	Pengguna	Sedang	Tinggi	(Lihat mitigasi untuk akurasi gestur dan kinerja), memberikan umpan balik visual yang jelas kepada pengguna mengenai status deteksi, opsi untuk menonaktifkan sementara gestur tertentu.

10. Penutup

Proyek "Kursor" memiliki potensi besar untuk memberikan kontribusi signifikan dalam meningkatkan aksesibilitas digital bagi individu dengan berbagai keterbatasan fisik. Dengan memanfaatkan teknologi visi komputer mutakhir dan pendekatan pengembangan yang berpusat pada pengguna, kami yakin dapat menghasilkan aplikasi yang fungsional, intuitif, dan bermanfaat.

Melalui kolaborasi antara Ardelyo (L1O) dan OurCreativity, serta bimbingan dari Pak Dadan Mochammad Ramdhan, proyek ini diharapkan dapat berjalan lancar dan mencapai semua tujuan yang telah ditetapkan. Sebagai bagian dari inisiatif proyek DAMAR (Dukungan Aplikasi Menuju Aksesibilitas Rata), Kursor tidak hanya bertujuan sebagai solusi teknologi, tetapi juga sebagai wujud nyata dari upaya menciptakan teknologi yang inklusif dan memberdayakan.

Kami berharap proposal ini dapat memberikan gambaran yang jelas mengenai visi, ruang lingkup, metodologi, dan rencana implementasi proyek Kursor. Kami siap untuk menjalankan proyek ini dengan dedikasi dan komitmen penuh untuk menghasilkan luaran terbaik.

11. Lampiran (Referensi Kode Sumber)

Kode sumber awal ("Kursor (kode basis awal)") yang menjadi dasar pengembangan proyek Kursor terdiri dari file-file berikut (struktur dapat berubah selama pengembangan):

File Utama (Contoh Struktur Awal):

- ▮ `main.py` (Titik masuk utama aplikasi)
- ▮ `tracker.py` (Mengelola HandTracker dan FaceTracker dari MediaPipe)
- ▮ `mouse_handler.py` (Mengelola emulasi aksi mouse via PyAutoGUI)
- ▮ `virtual_keyboard.py` (Implementasi keyboard virtual dan interaksinya)
- ▮ `gui_settings.py` (Mengelola antarmuka grafis untuk pengaturan)
- ▮ `config_manager.py` (Membaca dan menyimpan konfigurasi pengguna)

File Pendukung:

- ▮ `settings.json` (Contoh file konfigurasi default/pengguna)

▮ requirements.txt (Daftar dependensi Python)

▮ readme.md (Dokumentasi awal dan panduan pengembangan)

Kode sumber ini akan direview, direfaktor, dan dikembangkan lebih lanjut sesuai dengan rencana yang telah dijabarkan dalam proposal ini untuk menjadi aplikasi Kursor yang final.

© 2024 Proyek KURSOR - Inisiatif Proyek DAMAR (Dukungan Aplikasi Menuju Aksesibilitas Rata). Sebuah Proyek oleh Ardelyo (L1O) & OurCreativity.