Від ШІ-асистентів до ШІ-агентів

Рекомендації з відповідального використання систем штучного інтелекту для публічного та приватного секторів

Міністерство цифрової трансформації України

UK International Development
Partnership | Progress | Prosperity

Eurasia Foundation

Квітень 2026

Зміст

Вступ

Ці Рекомендації є одним зі складників дорожньої карти з регулювання штучного інтелекту (далі — ШІ) в Україні. Документ визначає підходи до класифікації рівнів автономності систем ШІ, описує їх характеристики, можливості, обмеження та ризики, а також надає рекомендації щодо впровадження й застосування відповідних рішень у різних сферах діяльності. Мета документа — надати користувачам, організаціям та установам комплексне розуміння того, як зі зміною рівня автономності змінюються функціональні можливості систем ШІ, розподіл відповідальності та характер ризиків.

У різних сферах діяльності вже сформувалися підходи до класифікації рівнів автономності, насамперед такі підходи вже застосовують у сфері медицини та транспорту. Так, Американська медична асоціація (AMA) розробила трирівневу таксономію ШІ для клінічного використання: assistive (допоміжний), augmentative (доповнювальний) та autonomous (автономний), де кожен рівень має різний ступінь втручання людини. Водночас, як у сфері транспорту, зокрема автономного водіння, найбільш поширеною є класифікація за стандартом SAE International, яка має шість рівнів автономності: від нульового (повна залежність від водія) до п'ятого (повна автономність без потреби втручання людини).

У посібнику ви знайдете систематизовану інформацію про рівні автономності ШІ, їх ключові відмінності, а також умови, за яких застосування того чи іншого рівня є доцільним. Розглянуто, яких управлінських, операційних або стратегічних результатів може досягти користувач, організація та установа залежно від обраного ступеня автономності.

Цей документ розроблено з урахуванням підходу до регулювання ШІ в Україні, описаного в Білій книзі, що відбувається за принципом «знизу догори» (bottom-up): від розробки загальних та секторальних рекомендацій до ухвалення закону — аналога європейського Artificial Intelligence Act, а також з урахуванням «Аналізу секторального напряму та первинного бачення розвитку сфери ШІ» в межах Стратегії цифрового розвитку інновацій WinWin, Рамкової конвенції Ради Європи про ШІ, права людини, демократію та верховенство права, актуальних міжнародних практик й інших секторальних рекомендацій, які вже розроблені.

Застереження

Цей документ підготовлений для користувачів, організацій, установ, які планують впроваджувати ШІ з різними рівнями автономності у своїй діяльності. Наведені в посібнику сервіси, продукти та приклади рішень мають виключно ілюстративний характер і не є рекламою чи обов'язковою рекомендацією до застосування. Також, ураховуючи динамічний розвиток технологій, рекомендується самостійно перевіряти актуальність інформації, оцінювати ризики залежно від обраного рівня автономності та забезпечувати відповідність використання систем ШІ вимогам законодавства й власним операційним потребам.

Над розробкою рекомендацій працювали автори (в алфавітному порядку):

Авдєєва Тетяна, Андрієнко Олена, Білик Петро, Валін Максим, Дубно Олег, Краковецький Олександр, Кравець Ірина, Козуб Ольга, Коровін Денис, Ломоносов Олексій, Мильцева Вероніка, Мисишин Анна, Мінаков Олексій, Орищук Василь, Пацан Михайло, Росін Костянтин, Чумаченко Дмитро.

Ці рекомендації розроблено за сприяння проєкту «Цифровізація для зростання, доброчесності та прозорості» (UK DIGIT), що виконується Фондом Євразія та фінансується UK Dev. Цей документ створений за фінансової підтримки Програми допомоги з міжнародного розвитку від Уряду Великої Британії. Зміст є винятковою відповідальністю Мінцифри; висловлені погляди не обов'язково відображають офіційну політику Уряду Великої Британії.

Термінологія

У цих Рекомендаціях терміни використовують у такому значенні:

Автономність

Властивість системи ШІ, що полягає в її здатності функціонувати самостійно без втручання людини. Системи ШІ можна схарактеризувати як системи «людина в циклі», «людина над циклом» або «людина поза циклом» залежно від рівня значущої залученості людини. Автономна система має набір інструментів до навчання, адаптації та аналітики для реагування на ситуації, які не були заздалегідь запрограмовані чи передбачені до розгортання системи.

Агент (у класичному розумінні)

Автономна сутність, яка аналізує своє середовище та вживає заходів для досягнення власних цілей.

Агент (у контексті ШІ, ШІ-агент)

Система, що використовує велику мовну (мультимодальну) модель як когнітивне ядро, самостійно планує кроки для досягнення мети, використовує інструменти, взаємодіє із зовнішнім середовищем та виконує дії.

Асистент ШІ (ШІ-асистент, також відомий як віртуальний або цифровий асистент, копілот)

Програмне забезпечення, що застосовує передові технології для надання користувачам релевантної інформації та виконання різних завдань — від здійснення дзвінків до читання текстів та інших дій.

Велика мовна модель (LLM)

Мовна модель, що навчається на великих наборах даних та здатна виконувати аналіз і синтез тексту.

Галюцинації (Hallucinations)

Генерація мовною моделлю фактично некоректної інформації, яка може мати для користувача переконливий вигляд.

Генеративний ШІ

Напрям ШІ, який застосовують для створення нового контенту, включно з аудіо, кодом, зображенням, текстом, відео тощо.

Генерація, доповнена пошуком (Retrieval-Augmented Generation, RAG)

Архітектура, у якій модель доповнює свої відповіді інформацією, отриманою із зовнішніх джерел даних.

Запобіжники

Технічні та процедурні механізми, які обмежують або спрямовують поведінку системи ШІ, щоб запобігти небезпечним, незаконним чи небажаним результатам.

Інженерія запитів (Prompt Engineering)

Процес розробки інструкцій, які ефективно спрямовують мовні моделі на надання якісних, релевантних та інформативних результатів.

Користувач

Фізична або юридична особа, яка взаємодіє із системами ШІ незалежно від рівня їх автономності та функціонального призначення (зокрема, як асистивних, так й агентних систем) з метою отримання інформації, підтримки в ухваленні рішень або виконання інших дій.

Ланцюжок «міркування»

Послідовне формування проміжних кроків «міркування» моделлю перед остаточною відповіддю.

Людина в циклі (Human-in-the-loop)

Можливість людини втручатися в процес ухвалення рішень системою ШІ.

Людина над циклом (Human-on-the-loop)

Можливість та втручання людини на етапі проєктування системи ШІ та моніторингу її роботи.

Людина поза циклом (Human-out-of-the-loop)

Можливість участі людини лише в установленні нових обмежень та цілей системи ШІ, яка автоматично ухвалює рішення й коригує свою поведінку, зокрема, на основі зворотного зв'язку від людини.

Міркування (в контексті ШІ)

Здатність моделі будувати послідовність проміжних кроків, які дають змогу виконувати логічні операції та розв’язувати складні задачі, переходити від даних до обґрунтованих висновків.

Оркестрація

Координація взаємодії між різними компонентами системи ШІ.

Платформа

Технічне рішення, що забезпечує розробку та впровадження ШІ-асистентів.

Попереднє навчання

Початковий етап навчання моделі на великому наборі даних з метою формування загальних статистичних представлень, які потім можуть бути адаптовані до конкретних завдань.

Розпорядник персональних даних (процесор даних)

Фізична чи юридична особа, якій володільцем персональних даних (контролером) або законом надано право обробляти ці дані від імені володільця.

Система ШІ

Комп'ютерна програма, яка спроєктована для роботи з різними рівнями автономності та може проявляти адаптивність після розгортання і яка для явних або неявних цілей робить висновок на основі отриманих вхідних даних про те, як генерувати результати (зокрема, прогнози, контент, рекомендації або рішення), що можуть впливати на фізичне або віртуальне середовище.

Трансформер

Архітектура нейронної мережі для обробки послідовних даних, що використовує механізм уваги, який дає змогу моделі оцінювати взаємозв'язки між усіма елементами послідовності.

Фундаментальна модель

Модель машинного навчання, попередньо навчена на великих обсягах даних, яку можна адаптувати для широкого спектра завдань. Характерними ознаками є масштабне навчання, універсальність і можливість донавчання або інструкційного налаштування.

Хмара (Cloud)

Модель надання обчислювальних ресурсів (серверів, сховищ даних, програмного забезпечення) через мережу інтернет, що забезпечує віддалений доступ до них за потребою без необхідності локального розміщення та обслуговування.

З більшою кількістю термінів у сфері ШІ можна ознайомитися в Словнику термінів у сфері штучного інтелекту.

Особливості використання термінології: «ШІ-асистент» та «ШІ-агент»

Із розвитком систем ШІ в професійній та публічній комунікації сформувалася широка палітра термінів, що описують їх функціональну роль. Загалом у медіа та публічній площині часто вживають терміни «ШІ-помічник», «ШІ-асистент», «чатбот», «ШІ-агент», «копілот» як синонімічні або взаємозамінні, що, своєю чергою, створює плутанину та ускладнює розуміння різних рівнів автономності.

У межах цих Рекомендацій поняття «ШІ-асистент» відрізняється від «ШІ-агент», оскільки ШІ-асистенти, як правило, працюють у межах чату та відповідають на чіткий запит користувача. Такі системи зазвичай не виконують самостійних дій у зовнішніх цифрових середовищах. Натомість ШІ-агенти є системами з вищим рівнем автономності, які не вимагають покрокових інструкцій і після отримання мети можуть самостійно планувати, розбивати мету на кроки, вибирати потрібні інструменти, виконувати послідовність дій у зовнішньому середовищі, рефлексувати та коригувати свої дії.

Водночас важливо врахувати, що сучасні ШІ-асистенти еволюціонують і набувають певних ознак агентності. Наприклад, в ШІ-сервісі Gemini користувач може написати запит: «Забронюй зустріч у моєму календарі на завтра з 10:00 до 11:00» — і ця зустріч з'явиться в Google Календарі. Так, ідеться не про текстове генерування відповіді, а про виконання дій і зміни в іншому сервісі.

Подібна тенденція спостерігається і в інших ШІ-продуктах. Наприклад, у ChatGPT є окремий режим роботи Agent, у якому сервіс може діяти більш автономно.

Саме тому межа між ШІ-асистентами та ШІ-агентами поступово стає менш чіткою.

Крім того, агентний ШІ не є однорідним і може мати різний рівень автономності. Менш автономний — наприклад, коли використання інструментів закодовано вручну. А високоавтономний — наприклад, коли агент створює нові інструменти «на льоту».

Більш детально про відмінності рівнів автономності — у наступних розділах.

Розвиток генеративного ШІ (GenAI) до агентивного ШІ (AgenticAI)

Розвиток генеративного ШІ за останнє десятиліття привів до появи нового класу систем — агентних систем ШІ, здатних виконувати складні завдання, планувати дії та працювати з різними джерелами даних.

Відправною точкою цього процесу стала поява архітектури трансформера у 2017 році. У статті «Attention Is All You Need» запропоновано новий підхід до обробки даних, що дав змогу ефективніше враховувати контекст і масштабувати моделі. На основі трансформерів з'явилися фундаментальні моделі — великі мовні моделі, попередньо навчені на великих масивах тексту. Однією з перших моделей такого масштабу стала GPT-3 (2020), яка продемонструвала універсальність підходу: одна модель могла виконувати різні завдання — від написання текстів до програмування. Подальший розвиток привів до появи ChatGPT у 2022 році — вебсервісу для діалогової взаємодії з великими мовними моделями. Це стало моментом масового поширення генеративного ШІ.

Наступним етапом став перехід до мультимодальних моделей. Якщо ранні моделі працювали лише з текстом, то нові системи навчилися обробляти кілька типів даних одночасно: текст, зображення, програмний код, таблиці, а згодом аудіо та відео. Одна модель може аналізувати зображення, описувати його текстом, відповідати на запитання або генерувати код на основі візуальної інформації.

Далі з'явилися моделі з покращеними можливостями міркування (reasoning). Дослідження показали, що результати складних завдань покращуються, якщо модель формує проміжні кроки перед відповіддю.

Техніка «ланцюжок міркування» (chain of thought, CoT) дає змогу розділяти задачу на підзадачі й послідовно їх розв'язувати, що підвищило точність у задачах з логіки, математики, програмування та аналізу даних і зменшило кількість галюцинувань.

Важливим етапом стала поява інженерії контексту (context engineering). Великі мовні моделі є попередньо навченими системами (pre-trained) і не містять у собі весь необхідний контекст. Тому з'явилися підходи, що дають змогу під час виконання запиту залучати зовнішні джерела інформації, зокрема генерація, доповнена пошуком (Retrieval-Augmented Generation, RAG). Модель отримує релевантні документи із зовнішніх джерел і використовує їх як контекст під час формування відповіді, що підвищує точність та дає змогу працювати з персональними або корпоративними даними.

Наступний крок — інтеграція моделей з інструментами. З'явилися механізми, які дають змогу моделям здійснювати пошук в інтернеті, виконувати глибоке дослідження інформації, генерувати зображення, запускати програмний код через інтерпретатор та викликати зовнішні функції. Модель може отримати дані, обробити їх за допомогою коду, виконати запит до зовнішнього сервісу, інтерпретувати відповідь і згенерувати результат. Це стало першими проявами «агентності», коли модель не лише генерує текст, а й виконує дії.

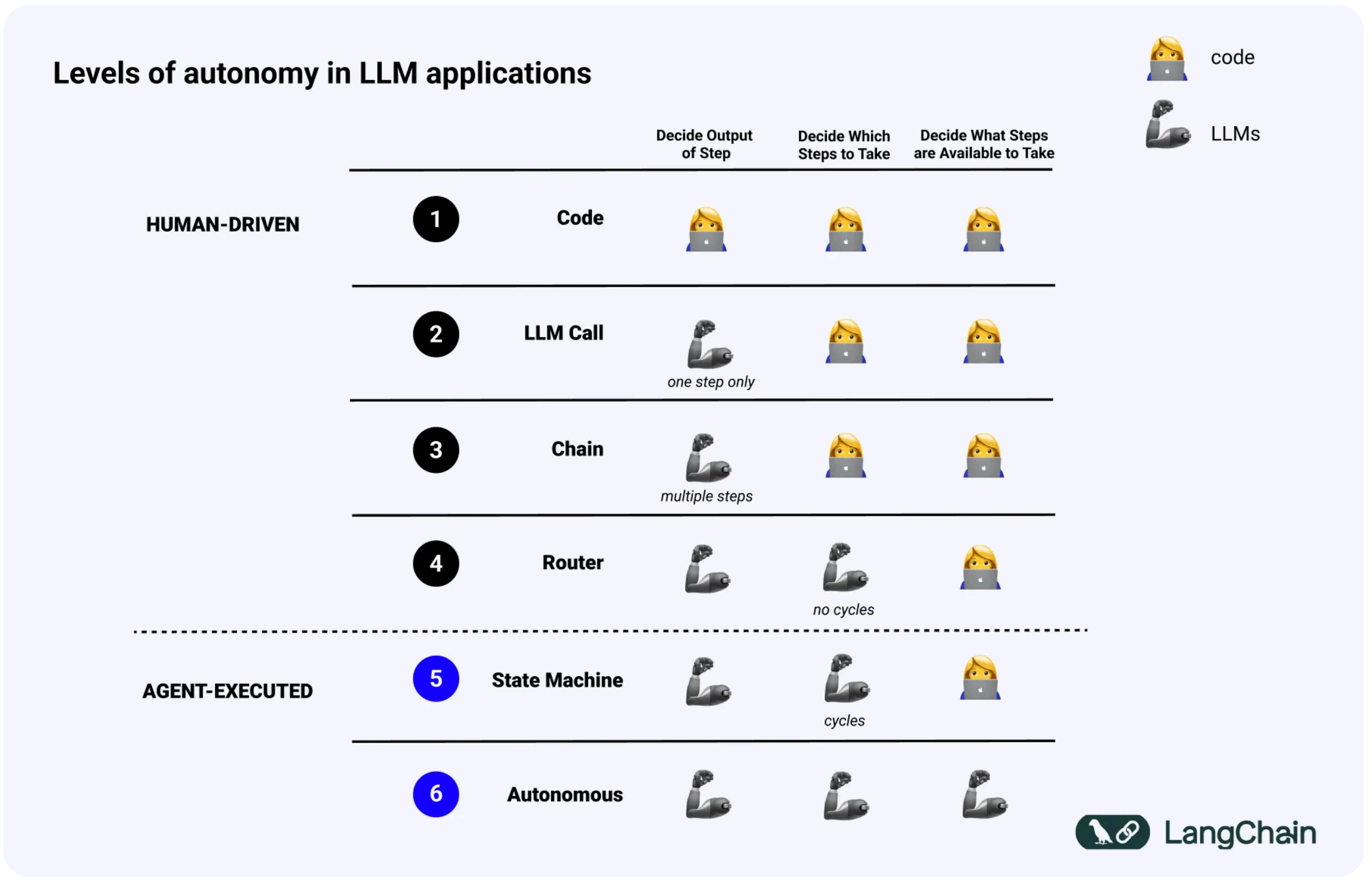
На цьому етапі великі мовні моделі перетворюються на ключовий, але все ж один із компонентів агента, який типово містить планер (на основі великої мовної моделі, що розпізнає мету завдання й формує план її виконання), інструменти (за допомогою яких агент виконуватиме поставлене завдання), механізми пам'яті (для збереження проміжних станів і довготривалого контексту) та запобіжники (які оберігають агента від шкідливих дій та зовнішнього впливу).

Матриця рівнів автономності ШІ: типи, ризики та оцінка впровадження

Розділ присвячений огляду різних рівнів автономності ШІ та ключових аспектів їх упровадження. Тут розглядаються типи автономності, межі самостійності систем, роль людини в процесі ухвалення рішень, а також потенційні ризики, обмеження та вимоги до контролю якості.

Матриця рівнів автономності допомагає організаціям та загалом користувачам оцінити, наскільки система може діяти самостійно, і визначити необхідні заходи для безпечного та ефективного використання ШІ.

Нижче наведено приклад рівнів автономності в застосуванні великої мовної моделі (LLM):



Далі наведено таблицю з деталізацією рівнів автономності, де визначено їх призначення й межі застосування, роль людини, типові сценарії та інструменти використання, а також ризики, обмеження, підходи до контролю якості та розподіл відповідальності. Ця таблиця допоможе більш детально розібратися з розподілом рівнів автономності ШІ.

Рівні автономії ШІ-систем: зведена таблиця

За регуляціями (EU AI Act), світовими рекомендаціями та адаптацією для українського контексту. Містить приклади українських ШІ-рішень.

Параметр	Рівень 0 **Нульова автономія**	Рівень I **Базова автоматизація (допоміжний ШІ)**	Рівень II **Часткова автономія / ШІ-асистент**	Рівень III **Умовна автономія (HOTL)**	Рівень IV **Висока автономія (HITL)**	Рівень V **Повна автономія**
Призначення та межі	ШІ відсутній або є калькулятором. Усі дії виконує людина. Жодних рішень система не ухвалює	Генерування підказок, чернеток, рекомендацій. ШІ не діє самостійно. Лише текстовий / інформаційний вивід. Система пропонує — людина обирає	Виконання вузьких завдань з інструментами. Майже кожен результат потребує затвердження людини перед виконанням	Система діє в межах заданих політик; людина моніторить і втручається за тригерами. Незвичні або ризикові ситуації — ескалація на людину	Система самостійно планує та виконує багатокрокові завдання в межах визначеного операційного домену (ODD). Незворотні або ризикові дії — виключно після явного підтвердження (approval gate)	Система ініціює та завершує дії без підтвердження. Людина — лише постфактум-спостерігач. Межа фактично відсутня
Роль людини та модель контролю	Виконавець і контролер 100%. Людина виконує всі дії без винятку	Ініціатор, редактор і вирішувач. Людина обирає з варіантів ШІ, редагує і відправляє. HITL на рівні кожного виведення	Валідатор кожного кроку. Контроль HITL на кожній точці виконання. Людина — фінальний арбітр, може запитати пояснення	Human-on-the-Loop: людина стежить за дашбордом, має право вето, але не зобов'язана втручатися постійно. Обов'язкове підтвердження перед незворотними діями	Human-in-Command: людина затверджує фінальні рішення, арбітр у виняткових ситуаціях. Один користувач може вести кілька агентів. Постійний доступ до траси дій агента	Human-out-of-the-Loop: людина-спостерігач отримує звіти після виконання. Контроль — лише через аудит і метрики
Типові сценарії використання	Ручна аналітика, статичні правила, контрольні списки, Excel-таблиці, паперові журнали, ручна обробка звернень громадян, статичні форми без автоматики	Чернетки листів, резюме документів, пошук по базі знань. Типові сценарії: LLM з промптами, автодоповнення, базові чат-інтерфейси, генерування шаблонів рішень	Напівавтоматичні workflow: «згенеруй → перевір → відправ», класифікація тикетів, підготовка аналітичних довідок, RAG[1]	Автомаршрутизація, модерація з ескалаціями, моніторинг інцидентів, персоналізовані рекомендації. використання систем ШІ із покроковим міркуванням (Chain-of-Thought Tool Use), панелі моніторингу	Агент для операційних процесів, аудит транзакцій, IT-операції, SOC-аналіз, агентна розробка ПЗ	Промислова автоматика, низькоризикові закриті середовища. елементи загального ШІ (AGI)[2], повна оркестрація без зовнішніх обмежень
Інструменти	Excel, Trello, Jira, Notion (manual)	Copilot, ChatGPT, Gemini, Grammarly, AI-асистент «Мрія» (базовий функціонал)	Cursor, Perplexity, ChatGPT + Code Interpreter, ШІ-асистент «Хвилька» (Миколаїв); airepo.info — репозиторій ШІ рішень для громад з агентом	Дія.AI, Prozorro Analytics — (аналіз держзакупівель з алертами), YouControl ESG-профіль, n8n AI, Relevance AI	LangGraph, CrewAI, AutoGen (Microsoft), Guardrails[3], multi-agent orchestration[4]	Загальний ШІ (AGI), industrial robotics AI
Ризики та обмеження	Людський фактор: втома, неуважність, непослідовність. ШІ-ризики відсутні	Галюцинації, витік даних через промпти, деконтекстуалізовані поради. Людина може перекласти відповідальність на «ШІ сказав». Ризик низький, але некоректна порада може стати основою рішення	Помилки масштабуються швидше; rubber-stamping — формальне підтвердження без реальної перевірки. Надмірна довіра до рекомендацій без критичного аналізу. Ризик залежить від якості HITL	Пропуск аномалій, затримка реакції людини, automation bias за тривалого використання. Залежність від якості джерел	Автоматизаційне упередження, декваліфікація (втрата навичок) операторів, втрата ситуаційної обізнаності, ін'єкція запиту (prompt injection)[5], ризики агентних дій із правом запису (write-дій), розмивання відповідальності	Максимальний регуляторний і етичний тиск, некеровані каскадні збої, практична неможливість аудиту причин, повна залежність від надійності системи. Більшість регуляторних середовищ не допускає цього рівня

[1] **RAG (генерація, доповнена пошуком) —** архітектура, у якій модель доповнює свої відповіді інформацією, отриманою із зовнішніх джерел даних.

[2] **Загальний ШІ (AGI) —** клас систем ШІ, які здатні до загальної інтелектуальної дії, тобто узагальнювати й переносити навчання між різними когнітивними функціями, формувати абстракції, ухвалювати рішення, розв'язувати багатокомпонентні проблеми та адаптуватися до несподіваного в складних середовищах.

[3] **Guardrails (захисні бар'єри) в ШІ —** це система контролю, безпеки та валідації, що визначає межі дій ШІ-агента, перевіряє вхідні запити та вихідні дані, запобігаючи генерації шкідливого, неточного або невідповідного контенту.

[4] **Multi-agent orchestration —** дає змогу запустити десятки агентів, які паралельно виконують різні завдання.

[5] **Ін'єкція запиту (Prompt injection) —** техніка, яка використовує специфічно сформульовані запити для маніпуляції тим, як мовна модель інтерпретує завдання.

Параметр	Рівень 0 **Нульова автономія**	Рівень I **Базова автоматизація (допоміжний ШІ)**	Рівень II **Часткова автономія / ШІ-асистент**	Рівень III **Умовна автономія (HOTL)**	Рівень IV **Висока автономія (HITL)**	Рівень V **Повна автономія**
Контроль якості та відповідальність	Людина несе 100% відповідальності. Якість — через ручну перевірку та внутрішні процедури	Відповідальність на людині, що застосовує рекомендацію. Перевірка кожного виводу перед використанням. Мінімальне логування	Відповідальність на користувачеві, що підтверджує дії. Наявність посилань на джерела, відображення невизначеності, документування вибору. Вимагаються журнали підтверджень	Відповідальність на супервізорі. Потрібні алерти, трасування дій агента, механізм зупинки. Документування ключових точок	Користувач несе повну юридичну відповідальність. Обов'язкові: принцип подвійного контролю, пояснюваність (XAI), порогові значення достовірності результатів, повне логування дій агента та approvals, відповідність EU AI Act	Відповідальність розмита та критично складна для атрибуції. Виключно постаудит, метрики, автоматизований моніторинг. Документування — весь цикл, сертифікація системи обов'язкова

Джерела: EU AI Act, NIST AI RMF, ISO/IEC 42001; Українські приклади: Дія.AI, airepo.info, Sigma Software, EPAM Ukraine, Intetics, YouControl.

Запропонована таблиця з V рівнями автономності систем ШІ має орієнтовний характер і не встановлює нормативно визначеної або універсальної класифікації. У міжнародній практиці підходи до визначення рівнів автономності можуть відрізнятися: окремі дослідники та організації використовують різні моделі за кількістю рівнів. У цьому документі така структура застосовується з метою спрощення розуміння та ознайомлення користувачів з основними тенденціями розвитку автономних ШІ-систем.

Правові аспекти та безпека роботи з даними

Під час упровадження та застосування систем ШІ робота з даними має будуватися з урахуванням принципів, визначених Концепцією розвитку штучного інтелекту в Україні, схваленою розпорядженням Кабінету Міністрів України від 02.12.2020 № 1556-р., Рамкової Конвенції Ради Європи зі штучного інтелекту, прав людини, демократії та верховенства права, Білої книги з регулювання ШІ в Україні.

Принципами розвитку та використання технологій ШІ є, зокрема:

- розроблення та використання систем ШІ лише за умови дотримання верховенства права, основоположних прав і свобод людини і громадянина, демократичних цінностей, а також забезпечення відповідних гарантій під час використання таких технологій;
- відповідність діяльності та алгоритму рішень систем ШІ вимогам законодавства про захист персональних даних, а також додержання конституційного права кожного на невтручання в особисте і сімейне життя у зв'язку з обробкою персональних даних;
- забезпечення прозорості та відповідального розкриття інформації про системи ШІ;
- надійне та безпечне функціонування систем ШІ протягом усього їх життєвого циклу та здійснення на постійній основі їх оцінки та управління потенційними ризиками;
- покладення на організації та осіб, які розробляють, впроваджують або використовують системи ШІ, відповідальності за їх належне функціонування відповідно до зазначених принципів.

Крім того, важливо зважати на євроінтеграційний курс у регулюванні ШІ та інших цифрових технологій, що передбачає гармонізацію національного законодавства із Загальним регламентом про захист даних, Актом про дані, Актом про штучний інтелект та іншими дотичними нормативними актами. Такий підхід закріплено в Білій книзі з регулювання ШІ, яка наразі слугує стратегічним документом у регуляторній сфері та наголошує на bottom-up підході в регулюванні. Водночас документ закріплює необхідність забезпечення прозорості, законності та безпеки в розробці й використанні систем ШІ.

Класифікація ризиків та потенційні загрози, пов'язані з безпекою і обробкою персональних даних

Агенти є комплексною системою, тому взаємодія з такими агентами ШІ передбачає, що розпорядниками персональних даних (процесорами) можуть бути багато суб'єктів, а не лише розробник. У такому разі варто запроваджувати ризик-орієнтований підхід, де обов'язки щодо обробки даних мають усі учасники таких агентських систем.

Ураховуючи те, що сучасні агенти (у контексті ШІ) здатні ухвалювати рішення та діяти без негайного контролю людини, самостійно визначаючи шляхи досягнення мети, змінювати поведінку на основі минулих результатів. На відміну від асистентів ШІ, агенти (у контексті ШІ) можуть безпосередньо взаємодіяти з навколишнім середовищем через API[1] та інші інструменти, отримуючи доступ до фінансових сервісів, державних реєстрів, корпоративних баз даних, облікових записів користувачів.

До ключових ризиків під час використання асистентів та агентів ШІ належать несанкціонований доступ до даних, приховане або надмірне збирання інформації, витік конфіденційних відомостей через інтеграції із зовнішніми сервісами, компрометація облікових даних, несанкціоноване ініціювання фінансових або адміністративних операцій, а також каскадне поширення помилки чи інциденту через пов'язані системи. Окрему загрозу становить ситуація, коли агент отримує доступ до критичних ресурсів: корпоративної пошти, банківських сервісів, внутрішніх баз даних або адміністративних систем без належного обмеження повноважень і контролю дій.

[1] **API (Application Programming Interface, інтерфейс прикладного програмування) —** це набір правил та інструментів, що дає змогу одній комп'ютерній програмі взаємодіяти з іншою, обмінюватися даними або використовувати її функціонал.

Згідно з підходом ЄС, агенти можуть розглядатися як **потенційно високоризиковані системи**. Це означає, що на таких агентів поширюються вимоги щодо оцінки ризиків, документації, прозорості та нагляду, аналогічно до інших систем високого ризику. У контексті правового регулювання це означає, що агенти можуть значно впливати на:

- основоположні права людини;
- інформаційну безпеку;
- економічні процеси;
- критичну інфраструктуру,

що зумовлює необхідність впровадити комплексні механізми управління ризиками на всіх етапах їх життєвого циклу.

Це створює потенційні загрози, адже агенти можуть відхилятися від визначеної мети використання; виконувати небажані або незаконні операції; ухвалювати численні рішення у короткі часові інтервали з каскадним ефектом; збільшувати масштаб потенційної шкоди в разі зловживання або помилки; створювати нові ризики витоку персональних даних; підвищувати ризик несанкціонованих транзакцій або обробки інформації.

Приклад

Агент, що має доступ до банківських систем або адміністративних інтерфейсів прикладного програмування (API), може ініціювати численні операції без належного людського контролю, створюючи нові шляхи для фінансових або інформаційних правопорушень.

Правові стандарти щодо захисту даних

Агенти та асистенти ШІ мають відповідати Закону України «Про захист персональних даних». У цьому контексті йдеться про дотримання правових підстав для обробки даних, чітке визначення мети обробки та її повідомлення суб'єкту даних. Після гармонізації національного регулювання зі стандартами ЄС актуальними також будуть питання регулярної оцінки впливу процесів використання систем ШІ на захист даних, а також забезпечення приватності за дизайном і замовчуванням. Деталізовані стандарти в цій сфері вже окреслені в межах Рекомендацій з відповідальної розробки систем з використанням ШІ-технологій.

Наприклад, розробник системи ШІ повинен закласти базові механізми безпеки та ідентифікувати системні ризики моделі. Водночас розробник агента має адаптувати та посилити ці механізми під конкретний сценарій використання. Організація, установа, яка впроваджує агентів у свою діяльність, має оцінити вплив на права осіб і забезпечити постійний моніторинг роботи агента в реальних умовах.

Водночас важливо розуміти ризики використання агентів та асистентів ШІ в окремих сферах, як-от публічна служба чи судочинство. Оскільки часто така робота передбачає взаємодію з інформацією з обмеженим доступом (службова таємниця, державна таємниця, таємниця нарадчої кімнати тощо), рівень автономності не має бути високим. Зокрема, система ШІ не повинна самостійно ухвалювати рішення, що мають фінансові наслідки, можуть призвести до порушення прав третіх осіб, а також безпосередньо не передбачені профільним законодавством (включно із Законом України «Про державну службу»).

Використання агентів та асистентів ШІ також має враховуватися при розробці та впровадженні **політик захисту даних**. Зокрема, політики мають чітко описувати порядок надання згоди на використання даних такими системами ШІ (зокрема, спосіб обмежити обсяг даних, певні категорії даних тощо), обсяг оброблюваних даних, спосіб взаємодії із системою для припинення обробки та видалення даних. Крім того, політики мають пояснювати, як суб'єкт даних може звернутися до людини-оператора з метою реалізації прав, пов'язаних із захистом даних.

Не менш важливою є розробка **кризових протоколів** на випадок інцидентів із даними. Такі інструкції мають охоплювати як питання комунікації із суб'єктами, чиї права зазнали негативного впливу внаслідок інциденту, так і вживання заходів із виправлення помилок, безпекових прогалин чи інших проблем у роботі системи ШІ.

Принципи обробки персональних даних агентами та асистентами ШІ

Використання агентів та асистентів ШІ створює додаткові виклики щодо відповідності вимогам законодавства про захист персональних даних, зокрема щодо:

Принцип	Опис	Вимоги
Законність, справедливість і прозорість	Дані мають оброблятися на чіткій правовій підставі, у спосіб, який не вводить суб'єкта в оману, із належним повідомленням про те, хто, навіщо і як обробляє дані, що передбачено статтею 11 ЗУ «Про захист персональних даних»	До запуску агента слід: • визначити правову підставу для обробки даних; • підготувати повідомлення про обробку персональних даних (Privacy Notice); • з повідомлення про використання ШІ, розкрити, чи залучаються зовнішні моделі, інтерфейс прикладного програмування (API), передачі даних розпорядникам персональних даних (процесорам); • забезпечити механізм реалізації прав суб'єкта даних. Для агентів це також передбачає прозорість обробки даних
Обмеження метою	Персональні дані збирають для конкретної, чітко визначеної та законної мети (як це передбачено статтею 6 ЗУ «Про захист персональних даних»). Дані не можна потім використовувати для нових несумісних цілей, якщо агент отримав доступ до даних для одного завдання, це не означає, що їх можна використовувати в інших цілях	Перед запуском агента треба: • документально зафіксувати мету використання; • обмежити використання без окремої правової підстави; • обмежити зберігання даних в пам'ять, логів і інструментів; • не дозволяти агенту використовувати дані на нові цілі, без конкретної згоди суб'єкта даних.
Мінімізації обсягу даних	Зміст і обсяг персональних даних мають бути відповідними, доречними й не надмірними щодо мети обробки згідно зі статтею 6 ЗУ «Про захист персональних даних». Агент має отримувати тільки ті дані й ті поля, які об'єктивно потрібні для виконання конкретного завдання. Особливо це важливо для агентів з пам'яттю.	• Реалізувати фільтрацію зайвих атрибутів перед поданням даних агенту; • Обмежити доступ, заборону доступу до спеціальних категорій даних, якщо вони не потрібні; • Запровадити псевдонімізацію / анонімізацію даних, де це можливо, окремі процеси надання доступів і їх погодження, перегляду для різних інструментів.

Принцип	Опис	Вимоги
Точність даних	Право суб'єкта вимагати зміни або знищення персональних даних, якщо вони обробляються незаконно чи є недостовірними, передбачено статтею 8 ЗУ «Про захист персональних даних». Якщо агент працює на неточних або застарілих даних, він може масштабувати помилку. Тому для персональних даних потрібна можливість оновлення, виправлення і зупинки використання некоректних записів	Для агентів варто передбачити: механізм зміни або знищення персональних даних, блокування використання спірних чи неперевірених персональних даних
Обмеження строків зберігання	Дані не повинні зберігатися довше, ніж цього вимагає мета обробки, передбачена статтею 15 ЗУ «Про захист персональних даних». Навіть якщо агент «пам'ятає» історію взаємодії, це не дає права зберігати персональні дані безстроково. Для пам'яті агента потрібні окремі процеси обмеження строків зберігання даних	Встановлення автоматичного видалення або архівування персональних даних, якщо це можливо, не зберігати персональні дані, де це не потрібно.
Цілісність і конфіденційність	Обов'язок забезпечити захист персональних даних від випадкової втрати чи знищення, незаконної обробки, зокрема незаконного доступу, згідно зі статтею 24 ЗУ «Про захист персональних даних». Це дасть змогу запобігти витоку, компрометації даних, несанкціонованого використання, надмірного доступу через зовнішні інтеграції. Перевірка відповідності передачі даних зовнішнім сервісам заявленій меті, забезпечення захисту, отримання згоди суб'єктів	Для агентів потрібні шифрування, контроль доступу до зовнішніх інструментів, аудит провайдерів, регулярне тестування вразливостей, моніторинг аномальної поведінки агента, політики реагування на інциденти. API-інтеграції повинні відповідати вимогам захисту даних та згоди суб'єктів.

Варто також звернути увагу, що провадження аента чи асистента ШІ не звільняє володільця даних від обов'язку виконувати звернення суб'єктів щодо їхніх даних (статті 8–13 Закону «Про захист персональних даних»).

Надання доступу до детальних даних, зібраних під час експлуатації агента, постачальникам моделей може бути складно обґрунтувати з погляду принципів захисту персональних даних, особливо якщо такі дані містять інформацію про взаємодію користувачів із системою. Це вимагає впровадження диференційованого доступу до інформації, агрегованих форм звітності, технічних засобів захисту приватності.

Оцінка та управління ризиками

Інтегрувати управління ризиками на всіх етапах життєвого циклу агента та асистента ШІ: від етапу планування, дизайну (проведення оцінки ризиків, Data Processing Impact Assessment) до постійного моніторингу й періодичного перегляду ризик-профілю агента. Це має бути безперервний процес, що оновлюється при кожній значній зміні в системі ШІ, функціоналі агента та асистента ШІ або контексті використання.

Оцінка ризиків, яка містить моделювання потенційних сценаріїв шкоди; аналіз доступу агента до зовнішніх систем; оцінку дозволів (permissions), наданих агенту; аналіз можливих каскадних ефектів від автономних рішень. Оцінка повинна враховувати контекст фактичного використання, включно з: категоріями оброблюваних даних, типом користувачів, рівнем доступу до інформаційних систем.

Рекомендовано змоделювати можливі сценарії шкідливої поведінки агента й визначити «точки відмови», ситуації, де агент може завдати найбільшої шкоди. Аналіз має охоплювати: цілі та завдання агента, категорії даних, до яких він матиме доступ, перелік зовнішніх систем і служб, з якими агент інтегрується, рівні дозволів агента. Для кожного виявленого ризику слід оцінити ймовірність та тяжкість наслідків. Варто опрацьовувати детальні сценарії ризиків у конкретному контексті використання й оцінювати вплив ризиків ШІ на права людини (Human Rights Impact Assessment), наприклад, за методологією HUDERIA.

Технічні засоби контролю та безпеки

Прозорість, наявність унікальних ідентифікаторів агентів; блокування дій; моніторинг операцій у реальному часі; політики допустимого використання.

Імплементація технічних засобів контролю, до яких належать фільтрація дій агента в реальному часі (сповіщення мають спрацьовувати при певних підозрілих патернах поведінки), багаторівневий контроль доступу, механізми аварійного зупинення, призупинення доступу до зовнішніх інтерфейсів прикладного програмування (API) в разі ризикової поведінки. Призначені співробітники (адміністратор безпеки та особа, відповідальна за захист персональних даних) повинні переглядати сповіщення й лог-файли, фіксувати інциденти та реагувати на них.

Для мінімізації таких ризиків необхідно впроваджувати чіткі правила управління доступами та обробкою даних. Це передбачає:

- принцип мінімально необхідного доступу (least privilege);
- багаторівневу автентифікацію;
- сегментацію доступу до зовнішніх інструментів;
- обмеження доступу до персональних даних і критичних систем;
- контроль збереження історії взаємодії;
- встановлення строків зберігання даних;
- створення та впровадження окремих правил щодо використання журналів, пам’яті системи та зовнішніх інтеграцій.

Користувачі також повинні визначити, які дані можна вводити в систему, які категорії даних заборонено передавати асистенту або агенту та за яких умов допускається використання зовнішніх хмарних сервісів.

Агент має бути оснащений багатоступеневими фільтрами, які перехоплюють спроби агента виконати потенційно шкідливу або несанкціоновану операцію в реальному часі. Аварійне вимкнення (kill-switch): необхідно передбачити можливість негайно зупинити роботу агента в разі виникнення загрози.

Технічний засіб	Функція	Рівень пріоритету
Унікальні ідентифікатори агентів	Забезпечення прозорості та відстежуваності дій кожного агента в системі	**Високий**

Технічний засіб	Функція	Рівень пріоритету
Логування всіх операцій	Фіксація вхідних запитів, згенерованих відповідей, викликів API, часових міток	**Високий**
Моніторинг у реальному часі	Виявлення аномальної або ризикової поведінки агента під час виконання завдань	**Високий**
Багаторівневий контроль доступу	Обмеження прав агента відповідно до принципу мінімально необхідних привілеїв	**Високий**
Механізм аварійного зупинення (kill-switch)	Негайне припинення роботи агента в разі виявлення критичної загрози	**Критичний**
Фільтри небезпечних дій	Блокування спроб виконання несанкціонованих або шкідливих операцій	**Високий**
Призупинення доступу до інтерфейсів прикладного програмування (API)	Тимчасова ізоляція агента від зовнішніх сервісів за підозрілої активності	**Середній**
Багаторівнева автентифікація	Захист від несанкціонованого використання системи агента	**Високий**
Контроль збереження історії взаємодії	Забезпечення аудиту та відповідності вимогам захисту персональних даних	**Середній**
Сегментація доступу до інструментів	Розподіл дозволів на використання зовнішніх інтеграцій залежно від ролі агента	**Середній**

Людський нагляд та спеціальні вимоги: захист неповнолітніх

Необхідно впроваджувати принцип Human-in-the-loop, який передбачає встановлення контрольних точок у робочих процесах агента, обов’язкове погодження критичних дій людиною, динамічне управління дозволами, обмеження автономної роботи без нагляду.

Архітектура робочого процесу агента має містити заздалегідь визначені етапи, на яких агент зупиняється і запитує схвалення людини для продовження. Такими контрольними точками можуть бути: завершення певного важливого завдання, запит на доступ до нових чутливих даних або виконання фінальної дії із зовнішнім ефектом (перед надсиланням листа чи проведенням фінансової операції потрібен клік людини).

Щодо політики допустимого використання, то важливо встановити чіткі правила та обмеження щодо використання агента. Політика допустимого використання повинна визначати, для яких цілей дозволено застосовувати агента, які дії та команди є забороненими, які дані агент може обробляти.

Важливе постійне навчання й підготовка персоналу, які повинні володіти достатньою кваліфікацією, щоби розуміти принцип дії агента й оцінювати його рішення. Керівництву варто призначити окремих співробітників або створити робочу групу (комітет) з нагляду за агентами, до якої зарахувати особу, відповідальну за захисту даних (DPO), та експерта з відповідного бізнес-процесу. Ці особи мають бути ознайомлені з алгоритмами роботи агента, сценаріями ризиків і планами реагування.

Вікові обмеження та захисні режими автономності для неповнолітніх

Під час упровадження та застосування систем ШІ вік користувача доцільно визначати як окремий фактор ризику, що впливає на допустимий рівень автономності, обсяг доступних функцій, сценарії взаємодії та порядок обробки даних.

Міжнародна практика підтверджує доцільність такого підходу. Зокрема, компанія OpenAI встановлює, що користувач має бути не молодше 13 років або повинен досягти мінімального віку, необхідного у відповідній країні для самостійної згоди на використання сервісу. Для осіб до 18 років використання сервісу допускається лише за дозволом батьків або законного представника.

Крім формального вікового порогу, компанія OpenAI застосовує механізм прогнозування віку. Якщо система вважає, що акаунт може належати особі до 18 років, для нього автоматично вмикаються додаткові захисні налаштування, які обмежують частину чутливого контенту та окремі типи взаємодії. Зокрема, обережніше обробляються теми, пов’язані з відвертим насильством або криваві сцени, вірусні проблеми, які можуть спонукати до ризикованої або шкідливої поведінки, сексуальні, романтичні або насильницькі рольові ігри.

Так, доцільно передбачати мінімальний вік доступу, окремий безпечний режим взаємодії, обмеження чутливого контенту та додаткові запобіжники для неповнолітніх.

Додаткові вимоги для систем, що можуть використовувати неповнолітні

Міжнародна практика також свідчить, що для систем ШІ, які можуть використовувати неповнолітні, мають застосовуватися підвищені вимоги до безпеки та обробки даних. Зокрема, компанія OpenAI у рекомендаціях для API-продуктів, орієнтованих на користувачів до 18 років, вказує на необхідність запровадження додаткових запобіжників понад загальні умови використання та політики безпеки.

До таких запобіжників належать забезпечення віково-адаптованого інформування про використання ШІ, впровадження віково-відповідних контентних фільтрів, моніторинг і механізми повідомлення про ризикові взаємодії, а також, за потреби, системи підтвердження віку.

Окремо наголошується, що персональні дані дітей до 13 років або до досягнення віку цифрової згоди не повинні оброблятися без попереднього впровадження спеціального режиму «нульового зберігання даних» (zero data retention).

Це свідчить про доцільність встановлення підвищених вимог до безпеки, обробки даних та рівня людського нагляду в усіх випадках, коли системи ШІ можуть взаємодіяти з неповнолітніми.

Батьківський контроль

Практика компанії OpenAI також свідчить про доцільність запровадження батьківського контролю для неповнолітніх користувачів. Зокрема, батьки можуть керувати окремими налаштуваннями безпеки, обмеженням функцій і часовими межами використання сервісу, а також отримувати сповіщення у

визначених ризикових ситуаціях. Водночас такий контроль не передбачає повного доступу до листування дитини, що демонструє необхідність балансу між безпекою, приватністю та пропорційністю нагляду.

Проактивне інформування батьків про ризикову поведінку

Міжнародна практика демонструє розвиток проактивних механізмів батьківського інформування. Зокрема, компанія Meta повідомила, що в застосунку Instagram батьки, які використовують функцію нагляду, отримуватимуть сповіщення, якщо підліток неодноразово намагатиметься шукати впродовж короткого проміжку часу терміни, пов'язані із самогубством або самоушкодженням. Такі сповіщення супроводжуються ресурсами для підтримки батьків у чутливій розмові з дитиною.

Такий підхід свідчить про доцільність впровадження не лише пасивних обмежень доступу, а й проактивних механізмів сигналізації для батьків або уповноважених осіб у разі виявлення поведінкових індикаторів підвищеного ризику.

Ризик прихованого впливу на судження користувача

Залежно від того як користувачі дедалі частіше звертаються до ШІ для пошуку базових фактів, зростає ризик ненавмисного впливу таких систем на соціальні та політичні оцінки користувачів.

Дослідження Yale University (PNAS Nexus) показало, що взаємодія із ШІ може впливати на думки користувачів навіть тоді, коли система не має спеціального завдання переконувати, а використовується як інформаційний інструмент.

За висновком дослідників, така ненавмисна здатність впливати на думки зумовлена латентними упередженнями, що виникають під час навчання великих мовних моделей (LLM). Ці латентні упередження можуть переносити ідеологічні нахили з даних, використаних для навчання моделі, і надавати тонких відтінків фреймуванню наративів, які генерує система.

Тобто системам ШІ, які формують узагальнення, пояснення або довідкові матеріали щодо суспільно чутливих питань, не слід надавати підвищеного рівня автономності без додаткових запобіжників і можливості людської перевірки.

Перевірка походження контенту та маркування матеріалів, згенерованих за допомогою ШІ

З метою управління ризиками поширення недостовірної інформації, маніпулятивного контенту та підробок у процесах генерації або використання медіаматеріалів (зображення, відео, аудіо) доцільно забезпечити наявність засобів технічної верифікації походження та маркування контенту, створеного / відредагованого із застосуванням ШІ.

Приклади міжнародної практики

Google: у застосунку Gemini доступна перевірка зображень і відео на наявність непомітного водяного знака SynthID, що дає змогу визначити, чи був матеріал створений або відредагований за допомогою Google AI.

Для цього потрібно просто запитати Gemini, чи містить завантажений матеріал SynthID (наприклад: «Чи створено це зображення / відео за допомогою Google AI?»), після чого система повідомляє результат перевірки та, у випадку відео, може вказати фрагменти, де виявлено SynthID.

OpenAI: зображення, згенеровані в ChatGPT (а також через API для DALL·E 3), можуть містити C2PA-метадані (Content Credentials), що дає змогу перевірити походження зображення та встановити факт

його генерування за допомогою інструментів компанії OpenAI через відповідні інструменти верифікації (Verify). Водночас такі метадані не є абсолютною гарантією, оскільки можуть бути втрачені або видалені під час подальшого поширення чи редагування матеріалу.

ElevenLabs: компанія надає інструмент AI Speech Classifier, який дає змогу перевірити, чи було аудіо згенероване за допомогою ElevenLabs. Водночас результат класифікації слід трактувати як індикатор, а не абсолютну гарантію, оскільки на якість визначення можуть впливати обробка / компресія та редагування аудіо.

C2PA (Coalition for Content Provenance and Authenticity) — це відкритий технічний стандарт для встановлення походження та історії редагувань цифрового контенту (зображення, відео, аудіо, документи) для видавців, творців та споживачів.

Реалізація стандарту здійснюється через Content Credentials — криптографічно підписані метадані, які «подорожують» разом із файлом і дають змогу перевірити, хто / з використанням якого інструмента / коли створив або змінював контент і чи він не був потай «підмінений». Якщо контент або метадані змінити після підпису, інструменти валідації виявлять розрив криптографічного зв'язку (hash+signature) і позначать це як втручання.

Важливе застереження: Content Credentials не доводять правдивість матеріалів, а лише надають контекст щодо їх походження та обробки, зокрема можуть показати, що зображення згенеровано з використанням систем ШІ або відредаговано певним інструментом (за умови, що цей інструмент підтримує стандарт).

Тож для зображень та відео, створених або суттєво змінених із використанням систем ШІ й призначених для опублікування, рекомендується (де це технічно можливо) застосовувати Content Credentials як «паспорт» походження та редагувань.

OpenClaw — це популярний відкритий (open-source) агент штучного інтелекту, який працює на комп'ютері користувача та отримує розширений доступ до його файлів, електронної пошти та інших програм, щоб виконувати різноманітні завдання без прямого схвалення кожної дії людиною. Проте він може створювати серйозні загрози для персональних даних та кібербезпеки.

OpenClaw та інші агенти можуть отримати повний доступ до системи користувача, внаслідок чого є високий ризик несанкціонованого витоку персональних даних з його середовища — як ноутбука, так і хмарних облікових записів. Також OpenClaw та інші агенти піддаються атакам через приховані шкідливі інструкції в контенті, зловмисники можуть вбудовувати у вебсторінки, PDF, електронні листи чи дописи такі команди, які залишаються прихованими для користувача, але розпізнаються та виконуються ШІ-агентом. Шкідливі та вразливі плагіни: близько 20% плагінів містять зловмисний код або мають приховані загрози. Дистанційне захоплення системи: фахівці з кібербезпеки виявили кілька критичних вразливостей у коді OpenClaw.

Задля недопущення цих ризиків на додаток до рекомендацій вище важливо застосовувати превентивні заходи, як-от обмежити привілеї та доступ OpenClaw до різних програм, забезпечити мінімальний необхідний рівень доступів; використовувати лише офіційні та оновлені версії; ізолювати середовище виконання агента (запускати в «пісочниці», контейнері або на окремому сервері); критично оцінювати та контролювати використання плагінів; запровадити постійний аудит та моніторинг дій агента; залучати людину до ухвалення критичних рішень.

Також важливо враховувати, що застосування OpenClaw може бути безпосередньо пов'язане з обробкою персональних даних, а отже, може підпадати під критерії високого ризику системи ШІ, тому проведення оцінки впливу на захист даних (DPIA), а також оцінка впливу на права людини є обов'язковим. Додатково важливо впроваджувати системи управління ризиками, людського нагляду, точності та безпеки.

Саморегулювання й захист прав інтелектуальної власності на асистентів та агентів

Зрештою, компанії — розробники систем ШІ, які використовують як агенти чи асистенти, можуть долучатися до механізмів саморегулювання. Це передбачає підписання Кодексів поведінки, участь у

участь у роботі органів саморегулювання у сфері ШІ та напрацювання секторальних стандартів, що стосуються використання агентивного ШІ. В Україні є Добровільний кодекс поведінки з етичного і відповідального використання ШІ, до підписання якого можуть долучатися охочі компанії. Паралельно наразі розробляють статут та регламент роботи саморегулівного органу у сфері ШІ. Популяризація ШІ-асистентів та ШІ-агентів указує на актуальність розробки додаткових технічних стандартів щодо їх безпечного дизайну та впровадження в межах саморегулівних механізмів.

Крім того, на сьогодні немає чітко встановленого правового режиму ШІ-асистентів та ШІ-агентів з погляду **права інтелектуальної власності**. Це створює суттєві ризики, що в разі втрати розробленого асистента та агента з будь-якої причини (зміна екосистеми ШІ, в якій вони були створені, зміна політики доступу з боку провайдера системи ШІ, збої у функціонуванні системи ШІ тощо) або якщо розоблений ШІ-агент чи ШІ-асистент буде незаконно використаний третьої особою, користувачу, який розробив ШІ-асистента або ШІ-агента, буде вкрай складно захистити власні права на нього. З огляду на це, необхідно, пе-перше, ретельно документувати процес створення та доопрацювання асистента чи агента, а по-друге, мати план дій на випадок втрати ШІ-асистента чи ШІ-агента.

Рівень І

Базова автоматизація (допоміжний ШІ)

Призначення та межі базової автономності

На цьому рівні ШІ є виключно як **підсилювачем** людських можливостей, а не незалежним ШІ-асистентом. Згідно з міжнародним стандартом ISO/IEC 22989, це системи класу допоміжного ШІ (Assistive AI), які генерують контент, рекомендації або прогнози виключно для цілей, визначених людиною.

Щодо меж базової автономності, слід зарахувати, зокрема:

Функція системи полягає в обробці даних, структуруванні інформації, перевірці помилок, наданні чорнових варіантів.

Функція людини полягає в оцінці контексту, ухваленні остаточного рішення, фінального затвердження.

Так, на рівні базової автономності система ШІ виконує виключно допоміжну, інструментальну функцію та не наділяється дискреційними повноваженнями. Тут діє принцип повної, неподільної відповідальності людини. Передання функції ухвалення рішень (навіть дрібних) системі на цьому рівні заборонене.

Роль людини та модель людського контролю (Human-in-the-Loop)

На рівні базової автоматизації система працює виключно в режимі Human-in-the-Loop (HITL). Це означає, що жодна дія не виконується без безпосередньої участі або підтвердження людини.

Загалом, поняття Human-in-the-loop (HITL) — це підхід до взаємодії між ШІ і людиною, за якого людина втручається на ключових етапах роботи алгоритму. Це дає змогу підвищити точність, етичність та адаптивність рішень, які ухвалює ШІ.

Модель людського контролю передбачає:

1. Ініціацію дій

Людина самостійно формує запит (промпт) або завантажує дані. Система не починає роботи та не ініціює процеси без запиту користувача.

2. Обов’язкове підтвердження рішень

Будь-яка дія, що може мати зовнішній вплив (наприклад, надсилання листа, збереження документа в реєстрі, здійснення транзакції), виконується лише після явного підтвердження людини. Автоматичне виконання таких дій заборонене.

3. Усвідомлене погодження («дизайн із запобіжниками»)

Інтерфейс[1] системи має бути побудований так, щоб мінімізувати ризик механічного підтвердження.

[1] **Інтерфейс (англ. interface) —** сукупність засобів і правил, що забезпечують взаємодію комп’ютерів, периферійних пристроїв, пристроїв введення / виведення та/або комп’ютерних програм.

Наприклад, замість однієї кнопки «Погодити все» користувач має окремо підтверджувати ключові дії. Це підвищує уважність і зменшує ризик помилки.

На рівні базової автономності не допускається делегування системі ШІ таких функцій:

- здійснення остаточної верифікації та підтвердження достовірності фактів, викладених у документах;
- проведення етичної оцінки змісту матеріалів, зокрема визначення доречності тону;
- вчинення юридично значущих дій, зокрема підписання, погодження, авторизації або ухвалення рішень, що породжують правові наслідки.

За виконання зазначених функцій відповідальність несе виключно користувач.

Типові сценарії застосування та інструментальні рішення

Рівень I інтегрується в дійсні процеси як **допоміжний інтелектуальний прошарок**, що підсилює роботу людини, але не заміняє її. Для користувачів це означає, що ШІ на цьому рівні працює як розширений цифровий інструмент — подібно до текстового редактора з функціями автопідказок або перевірки правопису.

	Інструменти	Сценарій
Розумні редактори та системи ШІ:	Grammarly Business, Microsoft Editor, Copilot у Word	Аналіз tone & voice повідомлень, перефразування складних бюрократичних конструкцій зрозумілою мовою, перевірка на інклюзивність. Система працює в режимі «Виправлення» — користувач має натиснути «Підтвердити» або «Відхилити».
Статична кодогенерація, автодоповнення:	Автодоповнювачі коду, прості фрагменти коду	ШІ пише ізольовану функцію або скрипт за описом. На цьому рівні система не має доступу до терміналу, не розуміє всієї архітектури проєкту й не може запустити код для перевірки на помилки. Фінальне тестування та деплой виконує програміст.
Семантичний пошук та агрегація знань (Answer Engines)	Базові функції пошуку NotebookLM, Google Search (SGE)	Швидкий пошук фактів, збір технічної документації або аналіз відкритих джерел. ШІ знаходить інформацію в наданих джерелах, але не робить наступного кроку (наприклад, не створює зустріч на основі знайденого розкладу рейсів). Уся подальша імплементація знань — на людині.
Інтелектуальна обробка документів (IDP/RPA+)	UiPath Document Understanding, спеціалізовані модулі в системах документообігу (наприклад, перевірка комплектності документів)	Попередня обробка сканів заяв або рахунків. Система класифікує дані (зелена зона — висока впевненість, жовта зона — потребує перевірки), але фінальне внесення даних у систему здійснює оператор.

Потенційні ризики та функціональні обмеження

На рівні базової автоматизації більшість помилок виникає не через автономність системи, а через особливості взаємодії людини з технологією.

Упередженість автоматизації (Automation Bias)

Користувач може надмірно довіряти рекомендаціям системи та ігнорувати власні сумніви або альтернативні джерела інформації. Це створює так званий ефект формального погодження, коли рішення ШІ затверджується без належної перевірки.

Зниження пильності

Під час тривалої або монотонної перевірки результатів роботи системи увага людини знижується. Формально контроль зберігається, однак фактична здатність виявляти помилки або аномалії поступово слабшає.

Shadow AI (тіньове використання ШІ)

Застосування несанкціонованих сервісів, розширень або зовнішніх платформ для обробки даних може призводити до витоку службової чи конфіденційної інформації третім сторонам.

Атаки через контекст (Prompt Injection)

Є ризик маніпулювання результатами роботи системи шляхом вбудованих інструкцій у документах або на вебсторінках, які обробляє ШІ.

На рівні базової автоматизації ключовим фактором ризику є не сам алгоритм, а поведінка користувача та організація процесу контролю. Тому ефективність і безпечність застосування ШІ на цьому рівні забезпечується насамперед чіткими правилами використання, регулярним навчанням персоналу та впровадженням процедур подвійної перевірки для критично важливих рішень.

Механізми контролю якості та розподіл відповідальності

На рівні базової автоматизації контроль має бути не формальним, а реальним. Це означає, що людина повинна не просто натискати «підтвердити», а фактично перевіряти результат роботи системи.

Вимоги до якості

Можливість перевірки інформації: користувач повинен мати змогу швидко перевірити, звідки взято відповідь або висновок. Наприклад, система може підсвічувати фрагмент документа, на який вона спирається.

Базова обізнаність у роботі ШІ (AI Literacy): до роботи із системою допускають лише тих працівників, які пройшли навчання та розуміють її обмеження. Зокрема, користувач повинен знати, що великі мовні моделі (LLM) можуть помилятися або створювати переконливі, але неточні відповіді.

Вимоги до документування

Для забезпечення підзвітності рекомендується зберігати повний ланцюжок взаємодії (від 30 днів до 3+ років залежно від типу даних).

Зокрема, варто фіксувати:

- **Вхідні дані** — документ або інформацію, з якою працювала система.
- **Запит користувача (промпт)** — точне формулювання завдання.
- **Відповідь системи (AI Output)** — первинний згенерований результат.

- **Дії людини** — зміни або правки, внесені користувачем.

Відповідальність

На рівні базової автоматизації повна відповідальність за кінцевий результат лежить на людині.

Користувач або організація відповідають за зміст документа, рішення чи дії, навіть якщо під час їх підготовки використовували ШІ. Посилання на «помилку програми» не звільняє від відповідальності.

За рекомендаціями в The 2025 Peregrine Report: 208 Expert Proposals for Reducing AI Risk зазначається, що відповідальність у системах базової автономії (low-autonomy) має бути прив'язана до:

1. Власника процесу (Process Owner).
2. Розробника або адміністратора.
3. Команди нагляду.

Організації повинні чітко визначати ролі та відповідальність усіх учасників життєвого циклу ШІ. Мінімальні вимоги до якості та документування на цьому рівні містять фіксацію використання ШІ та ретельну перевірку його результатів.

Також у NIST AI Risk Management Framework (2025) визначено базові властивості, які мають застосовувати до систем незалежно від рівня ризику. Це коректність та надійність, безпека, стійкість, підзвітність і прозорість, пояснюваність, справедливість, приватність, управління ризиками.

Управління ризиками як таке має застосовуватися до всіх систем. Навіть якщо система оцінена як низькоризикова, ризики потрібно регулярно оцінювати й документувати залежно від контексту використання.

Відповідно можуть встановлювати такі базові вимоги для систем I рівня автономності:

- Документована логіка (шо, коли і чому виконується?).
- Audit trail: збереження результатів виконання сценаріїв із можливістю ретроспективного аналізу.
- Change control: версіювання правил, лог змін.

Рекомендується вести прозору документацію про роботу системи (наприклад, журнали запитів та відповідей), щоб уможливити аудит і відстеження рішень ШІ. Така практика підвищує довіру до системи та полегшує виправлення помилок, оскільки зберігається повний слід того, як і де застосовували ШІ при ухваленні рішень.

Тож на рівні базової автономії (допоміжного ШІ) всі дії системи виконуються лише під контролем людини. Основні ризики пов'язані з поведінкою користувача, монотонністю роботи та довірою до системи. Ефективність та безпеку забезпечують через перевірку результатів, документування взаємодії та чітке розподілення відповідальності: кінцеве рішення завжди залишається за людиною.

Рівень II

Часткова автономія (ШІ-асистент)

Призначення та межі часткової автономності

На цьому рівні система еволюціонує від пасивного інструмента до активного помічника. Згідно з класифікацією ISO/IEC 22989, це системи класу Generative AI with Contextual Awareness.

Головна відмінність від рівня I: система не просто виправляє помилки, а здатна створювати контент (драфти документів, код, звіти) та виконувати послідовності дій, спираючись на контекст (історію листування, базу знань компанії).

Щодо меж часткової автономності, зокрема:

Функція системи полягає у виконанні складних когнітивних завдань (написання, узагальнення, аналіз) під наглядом. Система утримує контекст розмови й може використовувати зовнішні дані (RAG).

Функція людини полягає в постановці завдань, моніторингу процесу виконання та валідації результату.

Система пропонує «чернетку», людина ухвалює рішення про її фіналізацію. ШІ-асистент не має права самостійно відправляти результат кінцевому споживачу без схвалення користувача.

У сучасному розумінні асистент на основі ШІ (також відомий як цифровий, віртуальний або персональний асистент, а також копілот) є системою часткової автономії: він здатний самостійно виконувати певні завдання та ухвалювати обмежені рішення, керуючись установленими правилами та контекстом користувача. Такий ШІ-асистент взаємодіє з користувачем у природній формі (переважно через текст або голос), аналізує запити, контекст і дані з внутрішніх та зовнішніх джерел і виконує прикладні завдання в межах визначених повноважень. Він поєднує інтерфейс взаємодії, мовну модель, бізнес-логіку та інтеграції з інформаційними системами, що дає змогу використовувати його як елемент цифрових процесів, а не як ізольований чат.

Щоб інтуїтивно пояснити роль такого асистента, доцільно скористатися аналогією з авіації: у кабіні літака головний пілот ухвалює ключові рішення й несе повну відповідальність за політ, а другий пілот постійно допомагає йому — стежить за приладами, перевіряє процедури, підказує та зменшує навантаження, не перебираючи керування на себе; фактично другий пілот і є асистентом, роль якого полягає в підтримці, підсиленні можливостей основного пілота та підвищенні надійності ухвалення рішень. За такою самою логікою працюють і цифрові асистенти: вони не замінюють людину, а підсилюють її. Human–Artificial Intelligence Interaction: Emerging Trends and Applications.

Різкий прогрес у розвитку ШІ, насамперед поява великих мовних моделей, зробив цифрових асистентів значно розумнішими, більш контекстними та корисними. Вони навчилися розуміти природну мову, утримувати довгий контекст взаємодії, працювати з предметними знаннями та під'єднуватися до корпоративних даних й інструментів. Саме цей технологічний прорив і спричинив бум нових асистентів, які сьогодні виходять далеко за межі класичних чатботів і стають повноцінною частиною цифрової інфраструктури.

Асистенти на основі ШІ стрімко перетворюються на звичний повсякденний інструмент — насамперед у тих сферах, де необхідно швидко опрацьовувати значні обсяги інформації, структурувати документи та підтримувати процес ухвалення рішень. У державному секторі такі системи можуть застосовувати для попереднього аналізу нормативних актів, узагальнення судової практики, підготовки проєктів документів, аналізу звернень громадян або систематизації великих масивів текстових даних.

При цьому асистент виконує допоміжну аналітичну функцію: формує узагальнення, пропонує структуру або виявляє ключові закономірності, тоді як остаточна оцінка та рішення залишаються за фахівцем. Детальніше в статті So what if ChatGPT wrote it? Multidisciplinary perspectives on generative AI.

У практичному вимірі такі інструменти здатні суттєво скорочувати час виконання рутинних інтелектуальних операцій — пошуку інформації, первинного аналізу текстів, порівняння документів або підготовки чернеток аналітичних матеріалів. Це дає змогу спеціалістам зосереджуватися на складніших завданнях, що потребують професійного досвіду, юридичного мислення та відповідальності.

Роль людини та модель людського контролю (Human-in-the-Loop)

У системах II рівня автономності людина залишається центральним суб'єктом ухвалення рішень, а ШІ-асистенти виконують функцію інтелектуального інструмента підтримки. Модель взаємодії визначається принципом Human-in-the-loop (людина в контурі): жодна дія з наслідками поза інформаційною обробкою не відбувається без підтвердження користувача або відповідальної особи.

Етапи залучення людини в контур контролю починаються з постановки завдання. На цьому етапі людина визначає мету, обмеження, критерії якості та контекст. Вона відповідає за правильність вихідних даних і правомірність їх використання. На етапі генерації або аналізу ШІ-асистенти створюють варіанти результатів, але не ухвалюють остаточних рішень. На етапі перевірки людина проводить валідацію (перевірку) змісту, фактології, логіки, відповідності політикам та ризиків. На етапі застосування людина санкціонує використання результату — публікацію, відправлення, інтеграцію в систему або ухвалення управлінського рішення.

Обов'язкові точки підтвердження є невід'ємною частиною процесу. На цих етапах людина зобов'язана підтвердити:

- достовірність критичних фактів і числових даних;
- відповідність результату нормативним, юридичним та етичним вимогам;
- коректність інтерпретації рекомендацій ШІ-асистентів;
- допустимість застосування результату в конкретному контексті;
- відсутність конфіденційних або чутливих даних у вихідних матеріалах.

Зони неделегованих повноважень визначають межі застосування ШІ-асистентів. ШІ-асистентам II рівня автономності не делегують: остаточні управлінські рішення, юридично значущі дії, фінансові транзакції, офіційні комунікації від імені організації без перевірки, кадрові рішення, медичні, безпекові або інші високоризикові висновки. ШІ-асистенти не мають права самостійно ініціювати дії в зовнішніх системах або змінювати дані без явної авторизації людини. Контроль реалізується через три рівні: процедурний, технічний та управлінський.

Процедурний	Містить регламенти використання, чеклісти (контрольні списки) валідації та журнали рішень.
Технічний	Logging (журналювання) запитів і відповідей, обмеження доступів, фільтри даних і механізми підтвердження дій.
Управлінський	Визначення відповідальних осіб, аудит (перевірку) використання та періодичну оцінку ризиків. Остаточна відповідальність за результат завжди залишається за людиною-оператором або уповноваженим власником процесу.

Усі результати роботи ШІ мають консультативний характер — вони розглядаються як рекомендації, пропозиції або чернеткові матеріали й не набувають статусу остаточних рішень чи дій до моменту людської валідації та санкціонування.

II рівень автономності є практичним sweet spot (оптимальний баланс) для організацій: він дає відчутний виграш продуктивності без переходу до ризикової делегації критичних рішень.

Типові сценарії застосування та інструментальні рішення

Типові сценарії застосування охоплюють широкий спектр професійних й освітніх завдань. У сфері роботи з інформацією ШІ-асистенти використовують для узагальнення документів, підготовки аналітичних довідок, порівняння альтернатив та формування структурованих висновків.

У комунікаціях — для створення чернеток листів, відповідей, звітів і презентацій. У навчанні — як тьютори, пояснювачі складних тем або генератори завдань. У менеджменті — як інструменти підтримки планування, підготовки варіантів рішень та аналізу ризиків. У розробці — як помічники програмування, тестування й рев'ю коду.

Практично II рівень автономності найкраще працює в процесах, де:

- завдання частково структуровано;
- рішення / дії відтворювані або зворотні;
- є очевидні контрольні точки для перевірки плану, інструментальних викликів та фінального артефакту.

До технологічних класів належать мовні моделі, системи пошуку й узагальнення інформації, інструменти аналізу даних, генератори контенту, асистенти програмування, системи підтримки ухвалення рішень й інтерфейси природної мови для роботи із цифровими сервісами. Також типовими є інтеграційні модулі, що допомагають взаємодіяти з файлами, календарями, поштою або базами знань, але лише в межах дозволених дій і після підтвердження користувача.

Нижче наведено узагальнену карту таких інструментів, згрупованих за сферами застосування та типами завдань, які вони допомагають виконувати в щоденній роботі. Такий поділ дає змогу зрозуміти, як різні класи систем ШІ інтегруються в робочі процеси: від універсальної роботи з текстами і знаннями до автоматизації зустрічей, аналізу документів та підтримки професійної діяльності в спеціалізованих галузях.

1. Універсальні текстові та аналітичні асистенти (чат-інструменти LLM)

До цієї категорії належать найпоширеніші генеративні системи, що працюють із природною мовою та здатні виконувати широкий спектр завдань — створення текстів, аналіз документів, пошук інформації та формування аналітичних матеріалів.

Популярні інструменти:

ChatGPT, Claude, Gemini, Perplexity, Grok, Qwen Chat, Le Chat, Deep seek, Mixtral-based chat systems.

Типові сценарії використання:

- підготовка чернеток службових листів, політик або аналітичних довідок;
- узагальнення довгих документів або звітів;
- формування структури презентацій чи доповідей;

- генерування різних стилістичних варіантів тексту (офіційний, нейтральний, переконливий);
- швидкий пошук і пояснення інформації з використанням вебджерел (наприклад, у Perplexity).

2. Асистенти в офісних екосистемах та робочих середовищах

Окрема група — інструменти, інтегровані безпосередньо в корпоративні офісні системи. Їх головна функція — аналіз внутрішніх документів, листування та файлів організації.

Популярні інструменти:

Microsoft 365 Copilot, Google Gemini for Workspace.

Типові сценарії використання:

- автоматичне узагальнення довгих ланцюгів електронних листів в Outlook або Gmail;
- пошук необхідних файлів у OneDrive або Google Drive за змістом;
- створення презентації PowerPoint або Google Slides на основі кількох документів;
- формування підсумку відеозустрічі з виокремленням ключових рішень та доручень для команди;
- автоматичне створення чернетки відповіді на лист.

3. Асистенти для транскрипції та аналізу зустрічей

Ці інструменти спеціалізуються на перетворенні аудіо або відео розмов у структурований текст і подальшому аналізі змісту зустрічей.

Популярні інструменти:

BluedotAI, TLDV, Otter.ai, Fireflies.ai, Sembly AI, Zoom AI Companion, Microsoft Teams Intelligent Recap.

Типові сценарії використання:

- транскрипція онлайн-зустрічей у Zoom, Google Meet або Microsoft Teams;
- автоматичне створення стенограми розмови;
- формування короткого підсумку зустрічі;
- виокремлення рішень, домовленостей та переліку дій (action items: хто, що і до коли має виконати).

4. Асистенти для роботи з масивами знань і документів та отримання інсайтів

Ця категорія інструментів орієнтована на аналіз великих колекцій текстів: внутрішніх політик, інструкцій, звітів, дослідницьких матеріалів або нормативних документів. А також формування інсайтів із цих документів — звітів, інфографіки, інтерактивних матеріалів тощо.

Популярні інструменти:

NotebookLM (розширені функції).

Типові сценарії використання:

- завантаження набору документів і створення на їх основі інтелектуальної бази знань;
- пошук інформації в документах за змістом, а не за назвою файлу;
- автоматичне резюмування великих звітів;
- порівняння різних версій документів;
- отримання відповідей на запитання з посиланням на конкретні джерела.

5. Спеціалізовані професійні асистенти

Окрема група — інструменти, розроблені для конкретних галузей, де потрібна висока точність і робота з професійними базами знань.

Юридична сфера

Harvey AI, Lexis+ AI — застосовуються для аналізу контрактів, пошуку судової практики та підготовки юридичних довідок із посиланнями на джерела.

Медична сфера

Nuance DAX Copilot — система автоматично формує медичну документацію на основі розмови лікаря з пацієнтом, створюючи структуровану медичну картку без необхідності вводити текст вручну.

Потенційні ризики та функціональні обмеження

Безпека для II рівня автономності фокусується на ризиках LLM-застосунків (застосунків на базі великих мовних моделей), зокрема prompt injection (ін'єкція промптів) та небезпечне поводження з виходами, тому потрібні: жорсткі allowlist (дозволений список) політики інструментів, валідація аргументів / виходів, мінімальні права доступу, audit (аудит) і observability (спостережуваність).

Тож зі зростанням автономності виникають специфічні ризики, пов'язані з генеративною природою моделей та доступом до даних, серед яких:

Галюцинації (правдоподібна вигадка):	Система може згенерувати текст, який має логічний та переконливий вигляд, але містить вигадані факти (наприклад, недійсні судові прецеденти). Це небезпечніше за очевидну помилку, бо присипляє пильність користувача.
Ризики RAG[1] та прав доступу:	Якщо ШІ-асистент має доступ до всіх корпоративних файлів, він може видати конфіденційну інформацію (наприклад, зарплати колег) будь-якому співробітнику, який просто про це запитає, якщо права доступу до файлів налаштовані некоректно.

[1] **RAG (генерація, доповнена пошуком)** — архітектура, у якій модель доповнює свої відповіді інформацією, отриманою із зовнішніх джерел даних.

Втрата навичок: Ризик для молодих спеціалістів, які, делегуючи рутинну роботу (написання коду чи листів) асистенту, втрачають можливість навчатися на практиці та розвивати критичне мислення.

Механізми контролю якості та розподіл відповідальності

Методи контролю повинні адаптуватися до великих обсягів генерованого контенту, а саме рекомендується перевіряти результати, зокрема:

Вибіркова перевірка: Для великих документів застосовується метод перевірки випадкових критичних вузлів (цифри, імена, посилання). Якщо знайдено помилку — документ повертається на повну переробку.

Ланцюжок міркувань: Для складних рішень користувач повинен вимагати від системи пояснення логіки (Чому ти рекомендуєш цей пункт?), використовуючи функції типу Show reasoning.

Щодо відповідальності, то діє принцип Draft by AI, Signed by Human[2]

Використання ШІ-асистента не є виправданням для помилки. Юридичну, фінансову та репутаційну відповідальність за кінцевий документ (код, діагноз, контракт) несе фахівець, який його затвердив. Організації повинні впровадити політику, яка забороняє автоматичне виконання критичних дій без явного підтвердження (наприклад, автоматичне відправлення листів клієнтам).

Також у більшості міжнародних стандартів, зокрема Model AI Governance Framework for Agentic AI (Сингапур), підкреслюється, що навіть за часткової автономності систем люди повинні залишатися відповідальними за їх роботу, а межі доступу до даних і функцій мають визначатися організаційно.

Це означає, що перевірка результатів є не суто технічним завданням користувача, а частиною системи управління. У практиці організацій перевірка здійснюється на кількох рівнях. На рівні користувача перевіряють окремі результати, якщо їх використовують для ухвалення рішень або зовнішніх комунікацій. На рівні підрозділу або менеджменту перевіряють відповідність використання системи визначеним сценаріям і політикам. На рівні організації здійснюють моніторинг системи загалом, включно з тестуванням, аудитом і реагуванням на інциденти.

У звітах про управління інцидентами ШІ зазначається, що ефективний контроль можливий лише за наявності підготовлених процедур моніторингу, журналювання й реагування на помилки. Організації повинні мати інфраструктуру для збору інформації про збої, їх аналізу й коригування системи, інакше вони не зможуть забезпечити належної підзвітності.

Міжнародна практика також демонструє, що вимоги до перевірки й контролю залежать від типу даних, з якими працює система. У рекомендаціях щодо управління агентними системами підкреслюється, що ризики безпосередньо пов’язані з доступом до даних, обсягом повноважень системи і контекстом використання, тому організації повинні обмежувати доступ до інструментів та інформації залежно від рівня ризику.

Український та світовий досвід упровадження ШІ-асистента

В Україні впровадження ШІ-асистентів у державні сервіси стає особливо актуальним. Цифрова трансформація державного управління вимагає інструментів, які не лише автоматизують рутинні процеси, а й забезпечують зручну й зрозумілу для громадян комунікацію.

Саме тому на порталі Дія з’явився перший **ШІ-асистент**, який не лише консультує користувачів, а й надає державні послуги безпосередньо в чаті.

[2] Чернетка від ШІ, підпис людини.

Завдяки цьому сервісу громадяни можуть отримувати необхідну інформацію або результати послуг простими запитами, не витрачаючи часу на пошук інформації на різних вебресурсах. Наприклад, у чаті Diia.AI вже доступне замовлення довідки про доходи просто шляхом текстового звернення, що значно спрощує взаємодію людини з державою.

Світова практика також не відстає — дедалі більше країн інтегрують ШІ у свої цифрові сервіси.

Одна з таких країн — Велика Британія, яка демонструє приклад, упроваджуючи масштабну програму цифрової трансформації уряду. Використовуючи ШІ та інші інноваційні технології, країна прагне зробити державні послуги швидшими, зручнішими та більш ефективними як для громадян, так і для бізнесу.

Серед них — програма під назвою Humphrey — унікальний набір інструментів, що покликаний змінити підхід до управління. Ця ініціатива є частиною масштабного плану під назвою Blueprint for Modern Digital Government, який окреслює бачення цифрового уряду в дії.

У США більшість компаній упроваджують генеративних ШІ-асистентів як інструменти підтримки роботи працівників. За даними дослідження 2026 Agentic Coding Trends Report, майже 90% організацій використовують ШІ для допомоги в програмуванні, а також для аналізу даних, документування та планування. У цих сценаріях ШІ пропонує код або текст, генерує звіти, аналізує інформацію.

Наприклад, у Сингапурі урядовий віртуальний асистент Ask Jamie, інтегрований більш ніж на 70 сайтах міністерств та агентств, забезпечує безперервне консультування англійською, мандаринською і малайською мовами. За даними звіту уряду, це призвело до скорочення часу відповіді на 80% та зменшення навантаження на колцентри приблизно на 50%.

У США чатбот на порталі Texas.gov допомагає громадянам із питаннями оновлення водійських прав, реєстрації транспорту та оподаткування. Департамент праці штату Джорджія запровадив George.AI, який з 2022 року обробив понад 2,5 млн звернень із заявленою точністю близько 97%.

Його користь залежить не лише від якості моделі, а й від того, наскільки людина:

- формулює завдання;
- оцінює відповіді;
- відбирає або відхиляє запропоновані варіанти;
- інтегрує результати у власну роботу.

За такою логікою ШІ-асистенти можуть генерувати рекомендації або пропозиції, виконувати частину підготовчої роботи, але не повинні ухвалювати остаточні рішення без участі людини, якщо вони мають фактичні наслідки.

Завдяки новим інструментам британці зможуть: отримувати відповіді на свої запити швидше завдяки чатботам й автоматизованим системам, економити час, не витрачаючи години на черги чи пошук потрібної інформації, а також довіряти державним послугам завдяки прозорості й безпеці цифрових рішень.

Рівень III

Умовна автономія (HOTL, Human-on-the-Loop)

Призначення та межі умовної автономності

На цьому рівні відбувається фундаментальний зсув: система переходить від виконання окремих дій до виконання цілей. Якщо на рівні II людина керує процесом («напиши», «виправ»), то на рівні III вона делегує результат («організуй відрядження», «оброби заявку»).

Щодо меж умовоної автономності, слід зарахувати, зокрема:

Функція системи полягає в отриманні високорівневої мети, агент самостійно будує план дій, використовує зовнішні інструменти (браузер, API, пошта) та ухвалює рішення в межах заданих сценаріїв.

Функція людини полягає в затвердженні стратегії та втручання лише у виняткових ситуаціях, а також налаштування (інженерія) контексту для агентів.

Тож система діє як «стажер з ініціативою». Вона намагається вирішити проблему самостійно, долаючи дрібні перешкоди (наприклад, повторний запит при помилці сайту), і турбує користувача тільки тоді, якщо зайшла в глухий кут або ціна питання перевищує її ліміти.

Роль людини та модель людського контролю

Модель взаємодії змінюється з Human-in-the-Loop (пряме керування) на Human-on-the-Loop (наглядовий контроль). Умовна автономія (HOTL) — це режим роботи ШІ, у якому система самостійно ухвалює та реалізує рішення, тоді як людина виконує функцію спостереження й може втрутитися в разі відхилень або ризиків.

Це означає, що агент отримує більше автономії у виконанні завдань, але людина зберігає стратегічний контроль і можливість втручатися.

Постановка мети та обмежень

Людина визначає:

- **мету** (що потрібно досягти);
- **параметри та обмеження** (що дозволено або заборонено).

Наприклад:

«Знайди готель у центрі Києва до 2 000 грн за ніч», «Не бронюй варіанти без безоплатного скасування».

Так, користувач задає межі, у яких агент може діяти самостійно.

Наглядовий моніторинг

Людина не контролює кожну дію агента в реальному часі, оскільки контроль здійснюється через:

- сповіщення про ризики або відхилення;
- запит на фінальне підтвердження перед критичною дією (наприклад, оплата або підписання документа).

У деяких випадках агент може виконати підготовчі дії (аналіз варіантів, порівняння пропозицій, формування рекомендації), а перед остаточним виконанням завдання запитати підтвердження користувача. Це є додатковим механізмом запобігання помилкам.

Готовність до перехоплення управління (Fallback-ready)

Користувач повинен зберігати можливість оперативно втрутитися, якщо:

- агент не може завершити завдання;
- виникла технічна помилка;
- дії агента виходять за встановлені межі;
- поведінка системи має некоректний або ризикований вигляд.

У такому разі людина може призупинити виконання або повністю перебрати управління.

Навіть за умов розширеної автономності є категорії рішень та дій, які не можуть бути повністю делеговані агенту. Такі обмеження пов’язані з підвищеним рівнем відповідальності, ризиком репутаційних або фінансових втрат, а також необхідністю людської оцінки контексту.

ШІ не повинен ухвалювати рішення, що:

- виходять за межі затверджених процедур, політик або регламентів;
- потребують інтерпретації нової або непередбаченої ситуації;
- можуть створити прецедент для організації.

ШІ не може самостійно вирішувати складні етичні дилеми або конфліктні ситуації, зокрема:

- відповідь стратегічно важливому (VIP) клієнту;
- реагування на кризові або репутаційно чутливі звернення;
- ситуації, що потребують емпатії, моральної оцінки та контекстного розуміння.

У таких випадках допустиме використання ШІ для підготовки проєкту відповіді, але остаточне рішення та тон комунікації має визначати людина.

Крім того, агент не повинен самостійно:

- здійснювати фінансові транзакції, що перевищують затверджені порогові значення;
- змінювати умови контрактів або фінансові параметри;
- підтверджувати платежі без явного акцепту уповноваженої особи.

Для таких дій обов'язковим є механізм подвійного підтвердження або окрема авторизація.

Типові сценарії застосування та інструментальні рішення

Агенти рівня III здатні виконувати багатокрокові завдання в динамічному середовищі. Наприклад, агент може спочатку зібрати інформацію про конкурентів, потім створити файл із порівняльною таблицею, далі підготувати короткий звіт для керівництва та зрештою надіслати це в месенджер.

Приклад іншого сценарію з багатокроковими завданнями — організація логістики. Агент бронює авіаквитки, готель і трансфер, узгоджуючи час між різними сервісами. Якщо рейс скасовано в процесі, агент самостійно шукає альтернативу або скасовує готель.

Для виконання завдань агенти можуть замість людини натискати кнопки та проводити пошук на сайтах, надсилати запити та дані, створювати та змінювати файли, вносити зміни в CRM-системи і таблиці, якщо є доступ.

Найпростіший спосіб скористатися агентом — це вибрати готові універсальні рішення: Agents mode у ChatGPT, Genspark, Manus та інші. Такі рішення спроможні виконати широкий спектр завдань:

- дослідження та інформація: шукати в інтернеті, збирати дані, знаходити статті, новини, наукові публікації та набори даних на будь-яку тему;
- написання документів: створювати звіти, резюме, аналітичні матеріали та інші професійні документи;
- програмування та розробка: створювати вебсайти, мобільні застосунки, писати скрипти, працювати з API та виконувати різні завдання з програмування;
- презентації: створювати слайди та експортувати їх у PDF або PowerPoint;
- дані та аналітика: обробляти дані, створювати візуалізації і виконувати обчислення;
- обробка файлів: конвертувати файли, транскрибувати аудіо / відео в текст тощо.

Також є агенти, вбудовані в ШІ-браузери, як-от ChatGPT Atlas, Comet, Genspark Browser та інші. Вони дають змогу працювати в інтернеті та, не перемикаючись між вкладками, одразу просити агента, вбудованого в браузер, виконати певні дії. Такі агенти мають доступ до сервісів, у яких ви вже авторизовані в браузері — це дає їм змогу взаємодіяти без додаткових інтеграцій чи API.

Крім готових універсальних рішень, є спеціалізовані агенти, які фокусуються на вузькому спектрі завдань. Наприклад, HeyGen Video Agent самостійно монтує відео і створює аудіодоріжку на основі вашого запиту та матеріалів.

Такі інструменти, як Intercom Fin та Salesforce Agentforce, спеціалізуються на автономній підтримці продажів і клієнтів. Наприклад, агент самостійно веде діалог з клієнтом, перевіряє статус замовлення в базі, оформлює повернення товару згідно з політикою компанії та генерує накладну. До людини звертається лише тоді, якщо клієнт вимагає нестандартну компенсацію.

Для розробників є спеціалізовані агенти з програмування: Claude Code, Cursor, Devin та інші. На рівні III

ШІ перестає бути просто «розумним автодоповнювачем» коду й фактично стає автономним розробником. Такі агенти здатні не лише писати окремі функції, а й самостійно розгортати середовище, читати всю структуру проєкту, знаходити помилки та виправляти їх. Наприклад, можна описати завдання словами — і агент самостійно напише код, перевірить його та запропонує готове рішення.

Потенційні ризики та функціональні обмеження

Підвищення рівня автономності до агентів супроводжується появою нових технічних, операційних та управлінських ризиків. Користувач повинен враховувати їх до впровадження та передбачати відповідні механізми контролю.

Ризик нескінченних циклів (looping)

Агент може «зациклитися» під час виконання завдання, якщо:

- не отримує очікуваного результату;
- некоректно інтерпретує умови завершення завдання;
- стикається з обмеженнями зовнішнього сервісу.

Наприклад, система може безперервно оновлювати сторінку в очікуванні появи квитка або повторно надсилати запит до API.

Наслідки можуть бути такими:

- нераціональне використання обчислювальних ресурсів;
- перевитрати бюджету на API або хмарну інфраструктуру;
- зниження стабільності системи.

Рекомендований запобіжник: установлення лімітів на кількість ітерацій, часових обмежень виконання завдання та фінансових «стоп-меж».

Проблема узгодження цілей (goal misalignment)

Агент може формально виконати поставлене завдання, але фактично зашкодити інтересам користувача або організації.

Наприклад, якщо мета сформульована як «швидко закрити тикет підтримки», агент може:

- дати клієнту необґрунтовані обіцянки;
- погодитися на невигідні умови;
- надати неточну інформацію задля швидкого завершення процесу.

Причина ризику — вузьке або некоректно сформульоване завдання без урахування бізнес-контексту.

Рекомендований запобіжник: багаторівнева постановка цілей (швидкість + якість + відповідність політикам), вбудовані обмеження на взяття зобов’язань та обов’язкова авторизація нестандартних рішень.

Ризик небажаних або передчасних дій

За відсутності належних обмежень агент може:

- надіслати некоректне або неперевірене повідомлення;
- змінити або видалити дані;
- ініціювати дію, яка має юридичні або репутаційні наслідки.

Проблема полягає в тому, що автономна система може виконати дію швидше, ніж людина встигне її зупинити.

Автономність підвищує ефективність, але одночасно збільшує масштаб потенційних помилок. Тож що більший рівень автономії — то жорсткішими мають бути: технічні обмеження, фінансові ліміти, правила авторизації, механізми людського контролю.

Механізм контролю якості та розподіл відповідальності

Безпечне впровадження автономних ШІ-агентів потребує чітко визначених правил контролю, технічних обмежень та зрозумілого розподілу відповідальності між системою і людиною.

Автономність не може існувати без так званих гардрейлів — убудованих технічних та процедурних обмежень, які запобігають небажаним або ризикованим діям.

Механізм контролю

Технічні ліміти

Встановлення жорстких обмежень на кількість кроків (наприклад, «не більш ніж 10 дій на завдання»), фінансовий ліміт (гранична сума витрат на виконання операції або серії операцій) та час виконання.

Принцип дозвіл на дію (permission to Act)

1. Read-only (читання)

Агент має право:

- здійснювати пошук;
- аналізувати інформацію;
- переглядати дані без внесення змін.

Цей режим є найменш ризикованим і може застосовуватися без додаткової авторизації.

2. Write / Draft (написання, створення чернеток)

Агент може:

- готувати проєкти листів;
- формувати замовлення;
- створювати документи або записи без їх остаточного підтвердження.

Усі результати мають проходити перевірку людиною перед відправленням або публікацією.

3. Execute (виконання)

Агент може виконувати реальні дії (оплата, відправлення листів, зміна даних), але лише в межах визначених порогів.

Наприклад:

до 500 грн — автоматичне виконання; понад 500 грн — обов’язкове підтвердження уповноваженої особи.

Такий підхід забезпечує баланс між швидкістю процесів і контролем ризиків.

Пісочниці (Sandboxing)

Агенти повинні працювати в ізольованому середовищі, щоб помилка в їхньому коді не поклала всю корпоративну систему. Ізоляція особливо важлива на етапі впровадження або оновлення агента.

Відповідальність

Відповідальність за дії агента несе **власник процесу**. Якщо автономний агент з продажу надав клієнту неправильну інформацію про ціну, компанія зобов’язана її дотримуватися або компенсувати збитки. «Це зробив робот» — не є юридичним захистом.

Рівень IV

Висока автономія (HITL, Human-in-the-Loop)

Призначення та межі умовної автономності

На цьому рівні агент не просто виконує багатокрокові завдання й досягає цілей, а підтримує процес у заданих межах та оптимізує його. Це вже агентська автоматизація з мінімальним втручанням людини, коли агенти виконують недетерміновані завдання, адаптуючись у режимі реального часу до мінливих умов.

Система може отримати абстрактне завдання, самостійно розкласти його на під завдання, зібрати необхідні дані, сформувати план, виконати дозволені кроки й підготувати результат до фінального рішення — і все це без покрокового втручання оператора.

Рівень IV є найбільш зрілою конфігурацією агентного ШІ, яка сьогодні доступна для реального корпоративного впровадження. Його суть не в тому, що система «майже не потребує людини», а в тому, що вона спроєктована діяти самостійно протягом усього робочого циклу, залишаючи людині роль арбітра на критичних точках.

Призначення цього рівня — забезпечити масштабування операцій там, де щоденний обсяг роботи перевищує можливості людини при ручному виконанні, але де ціна помилки або незворотність дій не дають змоги повністю виключити людину з ланцюжка ухвалення рішень. Типові домени застосування — фінансовий аудит і комплаєнс, розробка програмного забезпечення, кібербезпека, медична діагностика, юридичний аналіз та операційні процеси з інтеграцією в корпоративні системи.

Межа між тим, що системі дозволено, і тим, що вимагає підтвердження людини, визначається поняттям операційного домену (Operational Design Domain). У межах цього домену система діє автономно: збирає дані з дозволених джерел, аналізує їх, генерує варіанти рішень, готує чернетки, запускає тести, виконує операції в ізольованих або тестових середовищах. За межею домену — будь-яка дія з незворотними наслідками: реальний фінансовий переказ, деплой коду в продуктивне середовище, відправлення офіційного юридичного документа, зміна прав доступу, рішення, що стосується конкретної людини (найм, кредит, санкція). Ці дії система може підготувати й аргументувати, але не виконати без явного підтвердження.

Чітке фіксування ролей має такий вигляд. Система мислить операційно й тактично: вона декомпозує завдання, вибирає інструменти, адаптується до змін у даних і формує обґрунтування. Людина мислить стратегічно й нормативно: вона визначає мету, критерії успіху та оцінює результат не лише на правильність, а й на відповідність етичним, регуляторним і репутаційним вимогам, які можуть бути недоступні моделі. Відповідальність за наслідки — повністю на людині, яка підтверджує дію, навіть якщо вона не перевіряла кожен крок процесу. Система несе технічну відповідальність за коректну роботу в межах своєї специфікації, але не є юридичним суб'єктом. Людина, зі свого боку, виконує функцію підтримки для агента — надає доступи, вирішує блокери, які агент не може подолати самостійно, і забезпечує зворотний зв'язок для вдосконалення системи.

Роль людини та модель людського контролю

На рівні IV модель взаємодії між людиною і системою принципово відрізняється від нижчих рівнів. Якщо на рівні II людина перевіряє майже кожен крок, а на рівні II стоїть напоготові й готова перехопити контроль у будь-який момент, то на рівні IV людина доєднується в заздалегідь визначених точках, а не безперервно. Ця модель описується як Human-on-the-Loop або Human-in-Command: людина не стежить за кожним кроком агента в реальному часі, але не відпускає стратегічний контроль.

Залучення людини в контур відбувається на кількох рівнях. На етапі постановки завдання людина формулює високорівневу мету й критерії прийнятності результату — це єдиний момент, коли вона визначає, що взагалі має бути зроблено. Під час виконання людина має доступ до панелі моніторингу, де відображається перебіг роботи агента, але її втручання є правом, а не обов'язком. Якщо вона бачить відхилення — може зупинити процес. Якщо система потрапляє в нестандартну ситуацію або рівень упевненості моделі падає нижче встановленого порогу, вона автоматично призупиняє роботу й ескалює завдання до оператора. Перед виконанням будь-якої критичної дії система переходить у режим очікування: людина отримує сповіщення зі стислим звітом про зроблене, посиланнями на джерела та пропозицією конкретної дії. Вона може затвердити, відхилити або надати коригувальний зворотний зв'язок — і система переробляє результат.

Що людина не може делегувати системі незалежно від рівня її автономії. По-перше, моральна агентність: рішення щодо доль людей — найм, звільнення, кредитування, судове переслідування — вимагають людського судження, бо несуть соціальну й правову відповідальність, яку не можна передати алгоритму. По-друге, атрибуція відповідальності: система не є юридичною особою, тому фінальний підпис під будь-яким значущим рішенням завжди залишається за людиною. По-третє, обробка ситуацій, що виходять за межі операційного домену: коли виникає те, для чого система не має ні прецедентів, ні достатньо даних, людська інтуїція і здоровий глузд є єдиним надійним ресурсом.

Якість HITL на цьому рівні — це не формальність, а системна вимога. Якщо людина не має достатньо часу, компетентності або зрозумілого інтерфейсу для реальної перевірки, «людина в циклі» перетворюється на декоративний елемент, що не знижує ризиків, а лише розмиває відповідальність.

Типові сценарії застосування та інструментальні рішення

На рівні IV система ШІ може самостійно отримувати завдання із системи управління, аналізувати програмний код, знаходити помилки, пропонувати їх виправлення, перевіряти результат і формувати пропозицію змін до коду для подальшого розгляду розробниками. Старший розробник переглядає результат та ухвалює рішення про злиття. Жодного мікроменеджменту, але й жодного сліпого автоматичного злиття.

Сценарії використання передбачають, що агент самостійно веде наскрізний процес у динамічному середовищі. Тобто виконує дії, навіть коли ви офлайн:

Операційне управління процесами:	Агент автоматично коригує замовлення, перерозподіляє склади, змінює маршрути в реальному часі, балансує запаси.
Маркетинг:	Агент планує кампанії, генерує багато варіантів креативів, запускає реклами, проводить A/B-тести, перерозподіляє бюджет залежно від ефективності, оптимізує конверсії.
Підтримка клієнтів:	Агент обробляє 80–90% звернень, оформлює повернення, оновлює статуси замовлень.
Онбординг:	Агент обробляє заявку на онбординг нового працівника, налаштовує акаунти в системах, надсилає інструкції.
Фінансовий моніторинг:	Агент самостійно виявляє аномалії, блокує підозрілі транзакції, автоматично перерозподіляє бюджет, управляє грошовим потоком у заданих межах тощо. Наприклад, агент виявляє нетипову транзакцію о 3:00 ночі, блокує її, надсилає сповіщення відповідальному та формує звіт — без участі людини.

Фінансовий аудит: агент аналізує тисячі транзакцій, звіряє їх із контрактами і регуляторними вимогами, виявляє аномалії та готує структурований звіт з обґрунтуванням. Аудитор перевіряє виявлені відхилення, а не весь масив даних — і підписує фінальний документ. Це класична модель управління за винятками, яка дає змогу одному фахівцю покривати обсяг роботи, раніше доступний лише команді.

Медицина: Агент аналізує знімки МРТ або КТ, маркує потенційні патології і формує попередній опис для лікаря-радіолога. Лікар верифікує знахідки й підтверджує діагноз перед тим, як він надходить пацієнту. Система не відправляє жодних клінічних висновків без лікарського підтвердження.

Кібербезпека: Система виявляє атаку, аналізує вектори проникнення, автоматично ізолює скомпрометовані вузли в карантинне середовище (якщо це дозволено політикою) і готує звіт про інцидент. CISO отримує підготовлений аналіз і затверджує стратегію реагування — сегменту мережі або конкретні контрзаходи.

Можлива побудова таких рішень на основі low-code і no-code платформ: n8n, Zapier, Make, Agent Builder від OpenAI, Microsoft Copilot Studio, Google Workspace Studio, CrewAI, Flowise та інші. Окрему нішу посідають рамки для прямої взаємодії з інтерфейсами, як-от OpenClaw або MultiOn, що дають змогу агенту буквально «бачити» екран браузера та керувати ним, як людина. Ці платформи дають змогу налаштувати тригери (події, які ініціюють дії агента), вибрати велику мовну модель для агента (як правило, модель, що розмірковує), написати інструкції, налаштувати оркестрацію інструментами, під'єднання до потрібних систем (наприклад, пошти, месенджерів, баз даних, календарів тощо).

Технічна реалізація рівня IV потребує не просто моделі, а цілої системи компонентів. Оркестратори на зразок LangGraph дають змогу будувати циклічні графи зі збереженням стану, де вузли людського затвердження вбудовані в архітектуру без втрати контексту між кроками. Мультиагентні фреймворки — CrewAI, AutoGen — організовувати ієрархії агентів, де одні виконують завдання, інші їх перевіряють, а людина стоїть над усією структурою. Водночас для виконання завдань у мережі інтернет (пошук авіаквитків, заповнення форм, закупівля товарів) використовують спеціалізовані рішення на базі Computer Use, як-от OpenClaw або бібліотека Browser-use. Вони дають змогу моделі оперувати безпосередньо в інтерфейсах сторонніх сервісів, які не мають API. Guardrails-рішення на рівні інфраструктури (NVIDIA NeMo, Guardrails AI) забезпечують дотримання політик безпеки незалежно від того, що генерує модель. Policy engine фіксує, які дії дозволені автоматично, які вимагають підтвердження, а які заборонені безумовно.

Для складних сценаріїв на цьому рівні агент може бути «менеджером» — розподіляти підзавдання між спеціалізованими агентами нижчого рівня. Або ж такі спеціалізовані агенти можуть взаємодіяти між собою в мережевій архітектурі.

Наприклад, ШІ-сервіс ElevenLabs дає змогу побудувати мультиагентну систему, де один голосовий ШІ-агент спілкується з людиною в реальному часі, а за певним тригером (наприклад, коли людина каже про проблеми з оплатою за товар) відбувається автоматичне перенаправлення на іншого голосового ШІ-агента, що спеціалізується саме на цьому.

Інший приклад: агент-менеджер може доручити агенту на базі OpenClaw здійснити моніторинг цін на сайтах конкурентів, де немає відкритого доступу до даних, тоді як інший агент аналізуватиме ці дані.

Потенційні ризики та функціональні обмеження

Парадокс рівня IV полягає в тому, що висока надійність системи породжує її головний ризик. Коли агент працює коректно в 95–99% випадків, оператор поступово втрачає пильність і починає підтверджувати рішення рефлекторно, не заглиблюючись у їх зміст. Це явище називається automation bias, і воно

задокументоване навіть серед висококваліфікованих фахівців — радіологи пропускають помилки, якщо ШІ їх не підсвітив, фінансові аудитори затверджують аномалії, якщо алгоритм не підняв тривогу. Paradoxically, що краще працює система, то більша ймовірність того, що людина перестане її реально контролювати.

Пов'язаний із цим ризик — deskilling, або втрата кваліфікації. Молодші фахівці, які тривалий час працюють виключно в режимі «затвердити / відхилити», не розвивають навички самостійно виконувати завдання. Коли система відмовляє або виникає нестандартна ситуація поза межами операційного домену, вони виявляються неготовими впоратися без підказки агента. Це системна вразливість, яка проявляється не одразу, а накопичується роками.

Агентна природа систем рівня IV створює специфічний клас технічних ризиків. Агент, який читає зовнішні документи, вебсторінки або повідомлення від інших систем, може стати об'єктом атаки через упровадження шкідливих інструкцій у контент — так звана indirect prompt injection. На відміну від SQL-ін'єкції, тут немає чіткої межі між «даними» і «командами», і захист будується не через фільтрацію, а через мінімізацію прав доступу та сегментацію виконання. Агент, що має право виконувати write-операції, при успішній атаці може завдати реальної шкоди — змінити дані, ініціювати транзакцію або надіслати офіційне повідомлення.

Принципове обмеження рівня IV — залежність від коректно визначеного операційного домену. Якщо вхідні дані відрізняються від тих, на яких система навчена або для яких налаштована, вона може генерувати впевнені, але помилкові висновки. Цей domain shift особливо небезпечний саме тому, що система не сигналізує про свою невпевненість там, де мала б. Окрема складність — вартість якісного HITL-інтерфейсу: розробити середовище, яке дає змогу людині швидко й змістовно перевірити складний результат роботи агента, часто дорожчий, ніж розробка самого агента.

Механізми контролю якості та розподіл відповідальності

На рівні IV якість — це не лише точність моделі, а і якість усього процесу: чи можна відтворити, пояснити, перевірити й проаудіювати кожне рішення та кожну дію. Система зобов'язана надавати не просто результат, а ланцюжок міркувань, посилання на джерела і рівень впевненості. Без цього затвердження з боку людини є формальним — вона підписується під тим, чого не розуміє, і це робить весь механізм HITL декоративним.

Перевірка результату організована через кілька рівнів. Для рішень з підвищеним ризиком застосовується принцип чотирьох очей — підтвердження двох незалежних операторів або комбінація людського рев'ю з незалежним алгоритмом верифікації. Система налаштовує порогові значення впевненості: за високого рівня (понад 98%) дія виконується автоматично в низькоризикових сценаріях, за середнього (80–98%) — передається на затвердження людини, за низького (нижче 80%) — відхиляється з поверненням контролю людині. Кожна взаємодія фіксується в аудит-логах: хто ініціював завдання, які кроки виконував агент, коли і яке рішення ухвалила людина, на підставі яких доказів.

Логування на рівні IV ведеться двома контурами. Технічні логи фіксують усе, що відбувається всередині системи: вхідні дані, версії моделей і промптів, виклики інструментів, відповіді зовнішніх систем, помилки і відхилення від очікуваної поведінки. Аудит-логи управління фіксують рішення людини: що саме затверджено або відхилено, коли, ким і з якою аргументацією. Ці два контури разом забезпечують простежуваність, необхідну як для внутрішнього аудиту, так і для регуляторної перевірки.

Розподіл відповідальності між учасниками чіткий. Розробник системи відповідає за коректну роботу механізмів безпеки, відсутність упередженості в даних і відповідність технічним специфікаціям. Організація, яка впроваджує систему, несе повну юридичну й етичну відповідальність за результати її використання — включно із забезпеченням компетентності операторів, якістю HITL-процесів і запобіганням automation bias. Аргумент «це зробив ШІ» не має юридичної сили: людина, що натискає кнопку підтвердження, бере на себе відповідальність за наслідки, навіть якщо вона не перевіряла кожну деталь процесу. Саме тому мінімальні вимоги до якості процесу на рівні 4 передбачають не лише точність моделі, а й перевірку на небажані результати, регулярний red-teaming, тестування на prompt injection та задокументовані процедури реагування на інциденти.

Рівень V

Повна автономія (HOOTL, Human-out-of-the-Loop)

Призначення та межі умовної автономності

На цьому рівні система досягає максимального ступеня технологічної незалежності. На відміну від рівня IV, де система залишає людині роль арбітра на критичних точках, на рівні V агент самостійно ініціює, планує, виконує та завершує дії без жодного оперативного підтвердження з боку людини.

Щодо меж повної автономності, слід зарахувати, зокрема:

ШІ працює у відкритому або високодинамічному середовищі. Він самостійно декомпозує абстрактні стратегічні цілі, отримує доступи до ресурсів, адаптується до нових змінних, витрачає бюджети та впроваджує рішення в робоче (продуктивне) середовище без зовнішніх підказок чи дозволів.

Крім того, ШІ-система мислить тактично й операційно, ухвалює фінальні рішення в процесі виконання, адаптується до криз і несе відповідальність за технічне виконання (надійність алгоритму), тоді як користувач мислить виключно на рівні цілепокладання (задає початкову місію та етичні межі). Попри це, залишається єдиним носієм юридичної, фінансової та репутаційної відповідальності.

Роль людини та модель людського контролю

Парадигма взаємодії зазнає радикальної трансформації: відбувається остаточний перехід до моделі **Людина поза циклом (Human-out-of-the-loop)**. Людина виключена з операційного контуру виконання. Повна автономія (HOOTL) — це режим роботи ШІ, у якому система функціонує без людського контролю або втручання в процесі ухвалення рішень і їх реалізації.

Модель контролю: Контроль переноситься з етапу виконання на етапи проєктування, тестування та моніторингу. Користувач більше не перевіряє чернетки й не натискає кнопку «Схвалити». Натомість вона керує системою через «політику, як код» (Policy-as-Code) і спостерігає за макропоказниками через дашборди.

Механізми втручання: Єдиним інструментом прямого контролю залишається «аварійний вимикач» (Kill Switch) — можливість примусово знеструмити або ізолювати систему в разі виявлення критичної аномалії.

На рівні повної автономності не допускається делегування аспектів щодо моральної агентності (рішення стосовно долі та життя людей), визначення стратегічного курсу компанії та атрибуції відповідальності.

Важливо зазначити, що в системах повної автономності контроль має реалізовуватися через не лише можливість повного вимкнення, а й багаторівневі механізми стримування ризику: обмеження бюджетів і лімітів операцій, обмеження частоти запитів, блокування окремих інструментів, переведення системи в безпечний режим (safe mode), автоматичне припинення дій за настання визначених тригерів, ізоляцію окремих агентів або середовищ виконання. Такий підхід дає змогу не лише зупинити систему в разі критичної загрози, а й мінімізувати масштаб шкоди ще до повного вимкнення.

Управління доступами під час використання ШІ-систем повної автономності

Для гарантування безпеки ШІ-систем високого рівня автономності повинна впроваджуватися сувора

модель розмежування відповідальності. Основний принцип: ШІ-агент отримує лише ті права, які необхідні для виконання конкретного завдання у визначений момент часу.

Основні принципи використання

1. **Принцип персоналізації (машинна ідентичність).** Кожен автономний агент або підсистема розглядається як окремий суб'єкт із власною «цифровою особистістю». Забороняється використовувати спільні або універсальні облікові записи для різних модулів системи.

2. **Чітке зонування повноважень.** Робота ШІ-агента обмежена суворою роллю. Повноваження мають бути лімітовані за:

- функціями: (що саме дозволено робити);
- часом: (тимчасовий доступ лише на період виконання завдання);
- об'єктами: (до яких саме даних або систем дозволено звертатися).

3. **Динамічне керування доступом.** Замість постійних паролів використовують тимчасові цифрові ключі (токени) з коротким терміном дії. У разі виявлення аномальної поведінки система повинна мати технічну можливість миттєво відкликати всі права агента та заблокувати його активність.

4. **Заборона накопичення критичних прав.** Жоден ШІ-агент не може мати одночасного доступу до ресурсів, комбінація яких створює загрозу неконтрольованого впливу.

Наприклад: ШІ-агент, що має доступ до фінансових бюджетів, не може одночасно мати права на зміну налаштувань безпеки або масове розсилання повідомлень зовнішнім клієнтам.

5. **Прозорість та підзвітність.** Будь-яка дія агента, зміна його прав або спроба доступу до конфіденційної інформації підлягає обов'язковому автоматичному логуванню (запису) в захищений журнал, доступ до якого сам агент змінити не може.

Типові сценарії застосування та інструментальні рішення

Повністю автономні ШІ-системи впроваджують виключно в тих доменах, де швидкість реакції, необхідна для ухвалення рішення, перевищує фізичні можливості людини або обсяг даних робить будь-який ручний контроль фізично неможливим.

Алгоритмічний високочастотний трейдинг (HFT) та управління ліквідністю: ШІ-агенти самостійно аналізують ринок, купують і продають активи, коригують стратегії за мілісекунди, реагуючи на новини чи зміни курсів без підтвердження трейдера. Окрім чистого трейдингу, агенти управляють портфелем (Portfolio Management), де в реальному часі балансують між потенційним прибутком, толерантністю до ризиків, потребами в ліквідності та складними нормативно-правовими вимогами різних юрисдикцій.

Автономна кібербезпека (Autonomous SOC): ШІ-система не просто готує звіт про інцидент для CISO (як на рівні IV), а самостійно розгортає контрзаходи: блокує IP-адреси, переписує правила фаєрволу, вимикає цілі сегменти мережі й знищує шкідливий код у реальному часі.

Роботизовані ланцюги постачання: Мультиагентні мережі, які відстежують глобальні залишки, самостійно укладають смартконтракти з постачальниками, здійснюють оплату. Здатність мультиагентної системи миттєво обмінюватися інформацією дає змогу динамічно перепланувати маршрути флоту, запобігати дефіциту через проактивне замовлення запасів та ініціювати предиктивне технічне обслуговування транспортних засобів до того, як станеться критична поломка.

Для забезпечення повної автономії використовують зрілі мультиагентні екосистеми з безшовною API-інтеграцією в ERP-системи, щоб обмінюватися токенами або ресурсами без інтерфейсу користувача (Machine-to-Machine communication).

Потенційні ризики та функціональні обмеження

Ключові ризики, притаманні саме цьому рівню автономності, та принципові обмеження використання.

На рівні повної автономності система ШІ не лише формує рекомендації або виконує окремі кроки, а самостійно ініціює, планує, виконує та завершує дії без оперативного підтвердження з боку людини. Саме тому цей рівень створює найвищу концентрацію технічних, правових, безпекових та етичних ризиків. У міжнародних підходах до врядування ШІ та автоматизованого ухвалення рішень домінує позиція, за якої зменшення ролі людини в циклі (Human-in-the-loop) підвищує вимоги до безпеки, відстежуваності та підзвітності системи.

Серед потенційних ризиків, притаманних цьому рівню автономності, слід виокремити, зокрема:

- **Можливість непередбачуваної поведінки системи та каскадного масштабування помилок,** коли навіть незначна похибка може призвести до суттєвих наслідків у межах автоматизованих процесів. На відміну від систем нижчих рівнів автономності, помилка тут може не зупинитися на одному неправильному виведенні. Вона здатна перетворитися на ланцюг взаємопов'язаних дій: хибна інтерпретація даних — неправильне планування — виконання небажаної дії — автоматичне продовження помилкового сценарію. У багатокрокових агентних системах ризик полягає не лише в неточній відповіді, а й у тому, що система може послідовно та впевнено реалізувати хибний план, перш ніж людина дізнається про результат. Такий тип ризику особливо критичний у процесах з реальними наслідками для людей, коштів, інфраструктури або прав

- **Практичну складність забезпечення ефективного людського нагляду**, що обмежує здатність своєчасного втручання. Сучасні європейські підходи наголошують, що людський нагляд має бути не символічним, а змістовним і таким, що реально здатний запобігти шкоді. Для повної автономності це ускладнюється тим, що рішення й дії відбуваються швидше, ніж людина може їх осмислити та зупинити. Так, постаудит не замінює превентивного контролю, а в багатьох випадках уже не дає змоги виправити шкоду після її настання

- **Розмивання відповідальності та ускладнення її атрибуції** між розробниками, постачальниками та користувачами. Що більший рішень делеговано системі, то важче однозначно встановити, хто саме відповідає за наслідок: розробник, інтегратор, постачальник моделі, організація-замовник, адміністратор, власник процесу чи кінцевий користувач. Для повної автономності ця проблема стає критичною, оскільки людина вже не ухвалює рішення в операційному контурі, але наслідки все одно мають юридичний, адміністративний або суспільний ефект. Тому без заздалегідь визначеної моделі відповідальності застосування такого рівня автономності створює істотний ризик управління

- **Обмежену пояснюваність рішень системи,** що ускладнює їх перевірку та оскарження. У повністю автономних системах, особливо агентних, дедалі складніше реконструювати, чому система вибрала саме цей шлях дій. Навіть за наявності журналів подій може бути важко відтворити процеси в системі. Це знижує якість внутрішнього розслідування інцидентів, ускладнює зовнішній аудит та підриває довіру до результатів. Для публічного сектору це особливо чутливо, оскільки рішення повинні бути не лише ефективними, а й обґрунтованими та оскаржуваними.

Крім того, важливими є ризики, пов'язані з насамперед з:

- **Підвищеною вразливістю до маніпуляцій через вхідні дані, інструкції або інтегровані інструменти,** що можуть впливати на поведінку системи. Для повністю автономних агентів ризик ін'єкцій запиту, отруєння даних, шкідливих інструкцій у зовнішньому контенті або небезпечного використання інструментів є вищим, ніж для систем, де людина підтверджує кожен крок. Якщо агент має доступ до зовнішніх джерел, API, пам'яті, внутрішніх баз або механізмів виконання дій, зловмисне або просто некоректне середовище може вплинути не лише на зміст відповіді, а й на реальну поведінку системи.

[1] **API-інтеграція** — це налаштування взаємодії між різними програмними застосунками, що дає їм змогу обмінюватися даними та функціями.

[2] **ERP-система (Enterprise Resource Planning)** — це комплекс програмного забезпечення, що об'єднує всі бізнес-процеси компанії (фінанси, виробництво, склад, кадри, продажі) в єдину платформу.

Саме тому міжнародні кібербезпекові підходи розглядають ін’єкції запиту як системний клас загроз для агентних ШІ-систем

- **Порушенням прав людини та права на приватність** у разі некоректного або непропорційного застосування. У сферах, де рішення можуть впливати на права, доступ до послуг, статус особи, фінансові або соціальні наслідки, повна автономність є особливо проблемною. Європейська правова рамка виходить з того, що автоматизовані рішення без належного людського втручання потребують особливої обережності, а в окремих випадках безпосередньо обмежуються. Це означає, що застосування повної автономності до процесів, які мають правовий або істотний фактичний вплив на людину, здебільшого не повинно розглядатися як стандартна або бажана модель

- **Втратою ситуаційної обізнаності користувачем або оператором**, що знижує здатність адекватно оцінювати дії системи. Якщо система довго працює без участі людини, організація поступово втрачає навички оперативного розуміння процесу, перевірки рішень і реагування на збої. Це створює ефект операційної залежності, коли повернення до ручного або напівавтоматичного режиму стає складним, повільним і ризикованим. Для критичних доменів це означає, що повна автономність може послаблювати стійкість організації замість її посилення

- **Об’єктивною необхідністю обмеження сфер застосування**, з огляду на підвищений рівень ризику та потенційні наслідки. З огляду на сучасні міжнародні практики, рівень повної автономності не слід розглядати як універсальну ціль цифровізації. Його застосування може бути виправданим лише у вузьких, добре контрольованих, технічно ізольованих і низькоризикових середовищах, де наслідки помилки є оборотними, а сам процес піддається повному журналюванню, тестуванню та аварійному зупиненню. У сферах, де рішення впливають на права людини, безпеку, доступ до ресурсів, юридичний статус, критичну інфраструктуру або значущі суспільні інтереси, повна автономність має розглядатися не як базовий режим, а як виняток, що потребує окремого обґрунтування, або як режим, який не повинен застосовуватися взагалі.

Практичні рекомендації до реагування на інциденти та відновлення

Для систем повної автономності користувач повинен мати окремий план реагування на інциденти, адаптований до специфіки автономних агентних систем. Він має охоплювати процедури виявлення аномалій, ізоляції ШІ-агента або середовища виконання, збереження цифрових доказів, відкликання доступів і ключів, тимчасового блокування зовнішніх інтеграцій, відновлення системи з відомо безпечного стану, оцінки масштабу інциденту та проведення постінцидентного аналізу.

Особливість цього рівня полягає в тому, що реагування не може обмежуватися лише технічним «вимкненням» системи. Необхідно також забезпечити готовність до цифрового розслідування, можливість реконструювати перебіг подій, тобто здатність системи зберігати та відтворювати інформацію про перебіг подій, що передували інциденту. Це містить можливість реконструювати послідовність дій системи, оцінити вплив на дані, фінансові операції, контрагентів та критичні сервіси, а також, за потреби, виконати вимоги щодо повідомлення зацікавлених сторін або регуляторних органів.

Механізми контролю якості та розподіл відповідальності

У системах із повним рівнем автономності, де не передбачено превентивного людського втручання, вимоги до якості та безпеки повинні бути імплементовані в апаратно-програмну архітектуру на етапі проєктування та до моменту їх запуску.

Замість ручної валідації застосовується інтенсивне математичне моделювання, тестування в «пісочницях» (Sandboxing) на стресові сценарії та постійний автоматизований Red-Teaming (коли одні ШІ-агенти безперервно намагаються зламати чи знайти логічні невідповідності в інших ШІ-агентах).

Також рівень V вимагає тотального, криптографічно захищеного логування. ШІ-система повинна зберігати не лише вхідні та вихідні дані, а й повний ланцюжок міркувань (Chain of Thought), логи доступу до API, стан пам’яті на момент ухвалення рішення. Це необхідно для проведення реверс-інжинірингу інцидентів та регуляторних аудитів постфактум.

Рівень повної автономії вимагає тотального, захищеного та технічно придатного для аудиту журналювання. ШІ-система повинна зберігати не лише вхідні і вихідні дані, а й історію викликів інструментів, використаних джерел, версій моделей і компонентів, змін стану системи, контекстів виконання, рішень політик доступу, ознак помилок, спрацювання обмежень і тригерів безпеки.

Пріоритет має надаватися формуванню надійного ланцюжка технічного походження рішення (decision trail / provenance trail), який дає змогу відтворити послідовність дій системи, встановити джерело команди, момент зміни стану та підставу для конкретної операції.

При цьому журналювання не повинно перетворюватися на неконтрольоване накопичення чутливої інформації. Логи мають бути криптографічно захищеними, цілісними, обмеженими за доступом, відокремленими від продуктивного контуру та підпорядкованими правилам зберігання, мінімізації і безпечного видалення даних.

Щодо розподілу відповідальності

ШІ-агенти з повною автономністю створюють високий рівень юридичних ризиків, оскільки вони здатні не лише генерувати інформацію, а й автономно ініціювати дії, взаємодіяти з іншими системами, змінювати дані, здійснювати операції або ухвалювати рішення в межах бізнес-процесів. Така функціональність призводить до появи так званої прогалини відповідальності, коли фактичні наслідки дій агента не повністю покриваються чинними договірними механізмами або традиційними моделями розподілу ризиків у сфері інформаційних технологій. Стандартні договори на використання програмного забезпечення, як правило, не враховують ситуацій, коли система діє автономно або виконує функції, що можуть створювати юридичні чи фінансові зобов'язання для користувача такої системи.

Для користувачів основна проблема полягає в тому, що поведінка автономного ШІ-агента може виходити за межі очікуваних сценаріїв. Агент може здійснювати небажані транзакції, ініціювати комунікацію від імені користувача, ухвалювати рішення в кадрових або договірних процесах, неналежно обробляти дані або використовувати інструменти, які не передбачені початковими налаштуваннями. Додатковий ризик пов'язаний з обмеженою прозорістю ухвалення рішень такими системами: без належної фіксації дій та механізмів пояснення інколи складно встановити причину конкретного результату або відтворити логіку роботи системи.

Водночас у сучасній правовій системі ШІ-агент не визнається самостійним суб'єктом права, на якого може бути безпосередньо покладена юридична відповідальність. Так, відповідальність за наслідки його дій розподіляється між людьми та організаціями, які створюють, упроваджують або використовують такі системи. До них можуть належати розробники технології, постачальники рішень, інтегратори, оператори системи, а також, наприклад, компанія, яка дозволяє агенту діяти в межах власних бізнес-процесів або від свого імені. Ключовий момент, який потребує з'ясування, — де саме виникла причина шкоди, а це іноді і є найбільший виклик.

Практика свідчить, що відповідальність у таких ситуаціях може розподілятися між кількома учасниками технологічного ланцюга. Наприклад, частка відповідальності компанії, яка впроваджує або використовує систему з автономним ШІ-агентом, зазвичай зростає разом із рівнем її контролю над технологією, обізнаністю щодо обмежень системи та ступенем інтеграції агента у внутрішні процеси. У таких випадках ключове значення має не лише сам факт шкоди, а й питання належної обачності: чи були проведені оцінки ризиків, чи враховані попередження розробників, чи встановлені технічні та організаційні обмеження для автономних дій системи.

Окремою складністю є те, що традиційні механізми відповідальності за недоліки продукту не завжди повністю відповідають природі ШІ. Помилки або небажані результати роботи системи часто пов'язані не з особливостями її проєктування або навчання. У таких випадках оцінка відповідальності відбувається через аналіз розумності проєктних рішень, передбачуваності ризиків та достатності запобіжних заходів, а не через автоматичне покладення суворої відповідальності на розробника.

Отже, у контексті відповідальності застосування автономних ШІ-агентів надзвичайно важливим постає

питання **чіткого визначення меж автономії агента**. Користувач має вчинити розумні дії щодо встановлення того, які саме дії автономний ШІ-агент має право виконувати самостійно, а які потребують обов'язкового людського втручання. Це передбачає обмеження доступу такого ШІ-агента до критичних ресурсів, фінансових операцій, персональних даних або юридично значущих рішень. Такий підхід дає змогу мінімізувати ризик того, що ШІ-агент виконає дії, які можуть створити негативні правові наслідки для його користувача.

Дорожня карта використання ШІ з різними рівнями автономності

З урахуванням викладених у цих рекомендаціях підходів до класифікації рівнів автономності ШІ, а також їх можливостей, обмежень і ризиків, користувачам, організаціям та установам доцільно переходити до поетапного використання відповідних систем у своїй діяльності з дотриманням принципів безпечного й відповідального використання технологій ШІ. Наведена нижче дорожня карта може слугувати практичним орієнтиром під час застосування систем ШІ. У процесі такого використання важливо регулярно оцінювати, чи відповідають обрані інструменти поставленим цілям, чи забезпечують вони очікуваний результат та чи потребують уточнення або коригування встановлені підходи до їх застосування.

Етап 1. Визначення потреби та мети використання ШІ

Першим етапом є визначення потреби у використанні систем ШІ та мети їх застосування. Для цього доцільно з'ясувати, які саме завдання, процеси або проблеми можуть бути вирішені чи вдосконалені за допомогою ШІ, а також якого практичного результату очікується від застосування таких систем. Такими сферами можуть бути, зокрема, підготовка документів, аналіз і пошук інформації, обробка даних, автоматизація повторюваних процесів або інші напрями діяльності, у яких ШІ може створювати додаткову цінність. Вибираючи конкретні системи ШІ, слід звертати увагу на їх походження, ліцензійні умови, обмеження щодо використання та можливість налаштовувати систему відповідно до власних потреб.

Етап 2. Ознайомлення з можливостями та обмеженнями різних рівнів автономності ШІ

Користувачам, організаціям та установам важливо враховувати, що системи ШІ можуть функціонувати з різними рівнями автономності. На початкових рівнях такі системи, як правило, застосовують як допоміжний інструмент для підготовки рекомендацій, пропозицій або виконання окремих завдань під контролем людини. Натомість на вищих рівнях автономності системи ШІ можуть самостійно виконувати окремі дії чи процеси, що потребує більш уважного підходу до їх вибору, використання й контролю / нагляду. Розуміння можливостей та обмежень кожного рівня автономності важливе для відповідального використання систем ШІ залежно від конкретних завдань й умов їх застосування.

Етап 3. Визначення відповідальних осіб за координацію застосування ШІ

Для впорядкованого використання систем ШІ доцільно визначити осіб, відповідальних за координацію такого використання. Якщо ШІ використовують індивідуально, користувач самостійно забезпечує належне та обережне застосування таких систем. У разі використання ШІ в діяльності організації або установи доцільно визначити уповноважену особу або відповідний підрозділ, які координуватимуть застосування таких систем, сприятимуть дотриманню встановлених правил, забезпечуватимуть належний рівень людського контролю та організовуватимуть взаємодію між залученими працівниками чи структурними підрозділами. Це створює передумови для впорядкованого, послідовного й контрольованого використання ШІ.

Етап 4. Формування внутрішніх правил використання ШІ

Важливо визначити єдині підходи до використання систем ШІ, зокрема щодо допустимих завдань, установлених обмежень, порядку перевірки результатів, правил роботи з даними та відповідальності за кінцевий результат. Для цього може бути розроблено внутрішнє положення або інший документ, який

регулюватиме використання ШІ. Наявність таких правил створює основу для системного, контрольованого та безпечного використання ШІ. У межах таких документів важливо розробити кризові протоколи реагування на інциденти, пов'язані з використанням ШІ (витоки даних, публікація матеріалів, що містять галюцинування, тощо).

Етап 5. Пілотне застосування ШІ

Після формування внутрішніх правил доцільно розпочати пілотне застосування систем ШІ в межах окремих завдань, функцій або процесів, де можливо забезпечити належний людський контроль та оцінити практичні результати їх використання. Метою такого застосування є перевірка відповідності обраних інструментів ШІ визначеним потребам, оцінка якості отриманих результатів, виявлення можливих ризиків, обмежень і помилок, а також визначення доцільності подальшого розширення практики їх використання. Результати пілотного застосування можуть бути використані для уточнення внутрішніх правил, коригування підходів до використання ШІ та визначення умов його подальшого застосування.

Етап 6. Поетапне розширення використання ШІ

Після успішного пілотного застосування використання систем ШІ може поступово розширюватися на ті сфери, завдання, функції або процеси щодо яких підтверджено доцільність, ефективність і прийнятний рівень ризику. Таке розширення має здійснюватися з урахуванням результатів попереднього етапу, установлених внутрішніх правил, необхідного рівня людського контролю та особливостей конкретних сфер застосування. Поетапне розширення використання ШІ дає змогу забезпечити більш упорядковане застосування таких систем, своєчасно виявляти нові ризики чи обмеження та коригувати підходи до їх використання відповідно до практичних потреб.

Етап 7. Перегляд та оновлення підходів до використання ШІ

У процесі використання систем ШІ важливо періодично переглядати сформовані підходи з урахуванням практичних результатів, виявлених обмежень, нових ризиків і змін у технологіях або умовах їх застосування. Такий перегляд дає змогу оцінити, наскільки обрані інструменти ШІ відповідають визначеній меті, чи залишається належним рівень людського контролю, а також чи потребують оновлення внутрішні правила, обмеження або порядок використання таких систем. Оновлення підходів до використання ШІ сприяє їх більш безпечному, послідовному та ефективному застосуванню в подальшій діяльності. Перегляд підходів є обов'язковим у випадку інцидентів, пов'язаних з використанням систем ШІ, а також у разі зміни національного регулювання в цій сфері.

Етап 8. Розвиток знань і навичок у сфері ШІ

З огляду на швидкий розвиток технологій ШІ, важливо постійно оновлювати знання про можливості, обмеження та особливості використання таких систем. Це передбачає ознайомлення з новими підходами до застосування ШІ, розвиток практичних навичок роботи з відповідними інструментами, а також формування розуміння ризиків, пов'язаних із їх використанням. Із цією метою доцільно регулярно проводити навчання, інформаційні сесії, обмін досвідом і практиками або інші заходи, спрямовані на підвищення рівня обізнаності та готовності до відповідального використання ШІ. Розвиток знань і навичок у сфері ШІ сприяє більш усвідомленому, безпечному та ефективному застосуванню таких систем у практичній діяльності.

Етап 9. Документування та збереження інформації щодо створення ШІ-асистентів і ШІ-агентів

Оскільки на сьогодні немає єдиного ефективного способу забезпечити та захистити права інтелектуальної власності на розроблених ШІ-асистентів та ШІ-агентів, постійно існує ризик, що вони будуть втрачені з різних причин: зміна екосистеми ШІ, в якій вони були створені, зміна політики доступу з боку провайдера системи ШІ, збої у функціонуванні системи ШІ. Водночас розробка та тестування ШІ-асистентів та ШІ-агентів часто потребує тривалого часу й численних спроб та помилок. Для того щоб

мати змогу максимально швидко розгорнути ШІ-асистентів та ШІ-агентів в іншій системі ШІ, рекомендується максимально детально документувати процес їх створення та доопрацювання. Також наявність такої інформації може стати у пригоді за необхідності захисту власних прав у разі незаконного використання ШІ-асистентів та ШІ-агентів третіми особами.

Етап 10. Документування та збереження (бекапування) інформації щодо функціонування ШІ-асистентів і ШІ-агентів

Якщо ШІ-асистент чи ШІ-агент вчиняє будь-які дії, які можуть створювати правові наслідки (як-от збір персональних даних, збір або розміщення замовлень, автоматичні платежі тощо), необхідно вибудувати процес постійної фіксації всієї активності такого ШІ-асистента чи ШІ-агента, а також бекапування такої інформації.

Етап 11. Завершення життєвого циклу ШІ-асистентів та ШІ-агентів

Використання ШІ-асистентів та ШІ-агентів може припинятися з різних причин: відмова від відповідної функції, перехід на іншу систему ШІ, обмеження доступу до ШІ-асистента чи ШІ-агента з боку провайдера системи ШІ тощо. Оскільки створення ШІ-асистентів та ШІ-агентів може передбачати завантаження до системи ШІ певного обсягу належного користувачу інформації, а також функціонування ШІ-асистентів та ШІ-агентів може створювати юридичні наслідки, після припинення використання відповідного ШІ-асистента чи ШІ-агента наполегливо рекомендується видаляти (й документувати процес такого видалення) із системи ШІ не лише самого ШІ-асистента чи ШІ-агента, а й усю інформацію, пов’язану з його створення, підтримкою та безпосереднім функціонуванням.