Choosing the Right Controls for AI Risks

The Company Ethos

AI Risk	Control Purpose	Design-Time	Run-Time
Model Drift and Data Distribution Shift Model performance degrades over time as real-world data diverges from training data. This can lead to increasing errors or unfair outcomes if not managed through robust design and ongoing monitoring.	Prevention	☐ Use diverse and up-to-date training data ☐ Apply data augmentation and simulation ☐ Plan for periodic retraining ☐ Build in robustness to variation	☐ Deploy adaptive systems with online learning ☐ Pre-schedule automatic retraining jobs ☐ Include fallback mechanisms for high-risk scenarios
	Detection	☐ Embed drift detection hooks or components in system architecture ☐ Stress test with simulated distribution shifts	☐ Monitor live data with drift metrics (e.g. PSI, KL divergence) ☐ Track model performance via outcomes or feedback loops
	Response	☐ Define thresholds for retraining and escalation ☐ Design retraining pipelines into architecture	☐ Trigger model retraining or fine-tuning ☐ Rollback to stable models if needed ☐ Escalate to human review or rules-based system in critical cases
Hallucinations in Generative Models Generative models can produce plausible-sounding but false or fabricated outputs. These hallucinations can mislead users, spread misinformation, or cause bad decisions. Controls are needed to promote truthfulness and catch inaccuracies during generation.	Prevention	☐ Fine-tune on high-quality, verified, domain-specific data ☐ Use Retrieval-Augmented Generation (RAG) to ground responses in trusted sources ☐ Apply Reinforcement learning from human Feedback (RLHF) ☐ Design abstention mechanisms (e.g. "I'm not sure" responses)	☐ Apply confidence thresholds to suppress or disclaim low-certainty outputs ☐ Use known prompt handling rules to avoid hallucination traps ☐ Use human-in-the-loop review in high-risk contexts
	Detection	☐ Train with human feedback to improve truthfulness metrics ☐ Build in self-consistency or ensemble checking mechanisms	☐ Run real-time fact-checking on generated content against trusted sources ☐ Use secondary models or filters to evaluate output accuracy ☐ Allow user reporting of suspected hallucinations
	Response	☐ Incorporate fallback logic for uncertain outputs (e.g. human/structured answer)	☐ Flag or block hallucinated content ☐ Escalate to human review if fact-checking fails ☐ Log and learn from hallucination cases to improve future accuracy
Bias and Fairness Issues AI systems may exhibit or reinforce bias, leading to unfair or discriminatory outcomes. This can result from biased training data, model design, or unintended feedback loops. Addressing this risk is essential to ensure ethical, legal, and reputational integrity.	Prevention	☐ Curate diverse and representative training datasets ☐ Apply fairness-aware algorithms (e.g., re-weighting, fairness-constrained optimization) ☐ Define and validate against fairness metrics (e.g., demographic parity, equalised odds) ☐ Conduct pre-deployment audits and fairness stress tests ☐ Include diverse perspectives in design reviews	☐ Enforce fairness rules in real-time (e.g., content balancing in recommendations) ☐ Use human-in-the-loop to review high-impact decisions ☐ Apply dynamic fairness constraints based on live inputs and user demographics
	Detection	☐ Simulate outcomes for different groups during validation ☐ Audit model performance across subpopulations ☐ Identify proxy variables that may encode bias	☐ Continuously monitor outcomes by group (e.g., acceptance rates by race/gender) ☐ Perform regular fairness audits on system outputs ☐ Track feedback loops or data shifts that may introduce bias over time
	Response	☐ Establish remediation plans for failed fairness tests ☐ Iterate model or feature set to reduce disparate impact	☐ Provide explanations and recourse for affected users ☐ Retrain or adjust models when bias is detected in operation ☐ Trigger human review or escalation protocols for flagged cases
Adversarial Inputs and Robustness Vulnerabilities Malicious actors may exploit AI systems using crafted inputs or attacks (e.g., adversarial examples, prompt injections, data poisoning). These attacks can bypass or mislead models, especially in security-critical applications. Defences must anticipate, detect, and respond to such threats.	Prevention	☐ Use adversarial training with attack examples ☐ Preprocess inputs to remove adversarial noise (e.g., input normalization) ☐ Perform red-team testing and security reviews ☐ Design ensemble or redundant model architectures ☐ Implement out-of-distribution detection components	☐ Apply rate limiting and throttling to prevent attack probing ☐ Sanitise or validate user inputs before model access ☐ Limit access to sensitive model capabilities (e.g., restricted prompts)
	Detection	☐ Simulate adversarial scenarios during evaluation ☐ Test models with known adversarial patterns	☐ Monitor input patterns for anomalies or adversarial characteristics ☐ Use confidence-based or activation-based anomaly detectors ☐ Log inputs and outputs for forensic analysis and threat pattern recognition
	Response	☐ Plan recovery paths for adversarial failure modes ☐ Embed fallback models or decision logic for high-risk inputs	☐ Trigger alerts or shutdown mechanisms if attack is suspected ☐ Isolate or block malicious inputs in real time ☐ Patch models or filters in response to discovered vulnerabilities ☐ Engage AI security teams for live response and threat mitigation
Loss of Personal or Confidential Information AI systems may inadvertently expose sensitive, personal, or internal data through memorization, output generation, or insecure access. This raises legal, ethical, and trust concerns — especially under data protection laws.	Prevention	☐ Apply data minimisation: exclude unnecessary PII ☐ Audit training datasets for sensitive content ☐ Use differential privacy or regularisation to reduce memorisation ☐ Anonymise or mask data before training ☐ Define strict access roles and separation of duties in system design	☐ Filter outputs in real-time for PII or sensitive patterns ☐ Limit access to models and outputs via authentication and roles ☐ Use input sanitisation to block confidential user submissions ☐ Warn users not to share sensitive information in prompts
	Detection	☐ Test for memorisation using known PII probes ☐ Run red-team attacks to surface potential leakage ☐ Monitor training for overfitting to rare or personal data	☐ Use automated output scanning (NER, PII detectors) ☐ Log and monitor for suspicious output or access behavior ☐ Conduct audits for data exposure or anomalies
	Response	☐ Establish privacy risk thresholds for retraining or model blocking ☐ Document potential leakage risks in system governance reviews	☐ Takedown or suppress harmful outputs ☐ Notify affected users or regulators if required ☐ Trigger incident response protocols for potential breaches ☐ Update training and filtering strategies based on incidents
Harmful Content Generation or Exposure AI systems, especially generative models, may produce or disseminate content that is offensive, abusive, or otherwise harmful. This includes hate speech, explicit material, misinformation, or content that incites violence. Such outputs can lead to user harm, legal repercussions, and reputational damage.	Prevention	☐ Implement strict data curation practices to exclude harmful content during training ☐ Apply content moderation policies aligned with legal and ethical standards ☐ Design models with built-in safety constraints to limit the generation of harmful content ☐ Use Reinforcement Learning from Human Feedback (RLHF) to limit undesirable outputs	☐ Employ real-time content filtering systems to detect and block harmful outputs ☐ Implement user input sanitisation to prevent the generation of harmful content ☐ Establish user reporting mechanisms for harmful content ☐ Apply rate limiting and monitoring to detect and prevent abuse patterns
	Detection	☐ Conduct adversarial testing to identify potential for harmful content generation ☐ Develop classifiers to detect harmful content within model outputs ☐ Establish benchmarks for acceptable content and test models against these standards	☐ Monitor content outputs continuously for signs of harmful material ☐ Utilize automated tools to detect and flag harmful content in real-time ☐ Analyze user feedback and reports to identify patterns of harmful content generation
	Response	☐ Create protocols for content takedown and user notification ☐ Develop incident response plans for addressing harmful content-related crises	☐ Implement immediate content removal or correction mechanisms ☐ Suspend or modify AI system functionalities when harmful content is detected ☐ Engage human moderators to review and address flagged content promptly ☐ Update models and filters based on incidents to prevent future occurrences
Feedback Loops and Behaviour Amplification AI systems can unintentionally reinforce behaviors or biases by acting on and shaping their own input data — creating self-reinforcing cycles. This can lead to echo chambers, polarisation, or instability in dynamic environments like social media or financial markets.	Prevention	☐ Introduce diversity-promoting mechanisms in algorithms (e.g., explore-exploit strategies) ☐ Apply multi-objective optimization (e.g., engagement + diversity + well-being) ☐ Conduct simulation studies to test long-term behavior patterns ☐ Enable user control features (e.g., reset or broaden profile options) ☐ Set policy constraints to prevent repetitive or overly narrow recommendations	☐ Periodically inject randomised or diverse content to break loops ☐ Allow users to actively request content variation ("show me something new") ☐ Reset or perturb model states on a scheduled basis
	Detection	☐ Run simulations during development to observe feedback effects ☐ Analyse how model behavior evolves across interaction cycles	☐ Monitor metrics like content diversity, engagement concentration, or user polarization over time ☐ Detect signs of degenerative cycles or user behavior anomalies ☐ Audit system impact on group behaviors and preferences
	Response	☐ Tune algorithms based on simulation findings ☐ Embed decay factors or reset mechanisms into the system logic	☐ Adjust model parameters in real time to counter amplification trends ☐ Escalate to human review when feedback effects exceed thresholds ☐ Periodically retrain on external or unbiased data to rebalance the system
Overreliance on Automation / Erosion of Human Oversight As AI systems become more capable, users may place too much trust in them, leading to passivity or failure to challenge incorrect outputs. This "automation bias" is especially dangerous in high-stakes domains like healthcare, finance, or aviation. Effective controls preserve human judgment and ensure the AI is treated as a decision aid, not a final authority.	Prevention	☐ Design transparent, explainable AI outputs ☐ Include cognitive forcing functions (e.g., rationale for accepting AI recommendations) ☐ Provide alternative perspectives (e.g., second opinion models) ☐ Train users during rollout on AI limitations and failure cases ☐ Build systems that defer to humans when uncertain ☐ Establish rules requiring human sign-off for critical decisions	☐ Include mandatory review steps for high-risk recommendations ☐ Provide in-system prompts or nudges to encourage critical evaluation ☐ Limit duration of fully automated operation before requiring human check-in
	Detection	☐ Evaluate interface design for undue trust tendencies ☐ Test user workflows for blind acceptance of AI suggestions	☐ Monitor user behavior for signs of overreliance (e.g., always accepting AI outputs without edits) ☐ Track model confidence and flag low-certainty cases for human review ☐ Conduct regular oversight audits and operator feedback reviews
	Response	☐ Adjust system design based on user testing outcomes ☐ Reinforce human-in-the-loop workflows in high-risk contexts	☐ Trigger alerts or warnings when overreliance is detected ☐ Escalate questionable outputs for human validation ☐ Share examples of caught AI errors to reinforce vigilance culture ☐ Retrain staff or adjust incentive structures to reward thorough review