RISK TABLE: Joke Generating LLM

Muharrem Şimşek – 210706001

Taha Demirhan – 210706002

Simay Aydın – 210706040

Aslımay Mısra Kandar – 210706043

Gökhan Mert Demirok – 220706311

07.05.2025

Risk No	Risk Title	Description	Likelihood	Impact	Mitigation
1	Inappropriate / Offensive Content Generation	The model may unexpectedly generate profanity, offensive jokes, or sensitive content.	Medium	High	Implement content filters, curate clean training data, and add human review mechanisms.
2	Cultural or Religious Sensitivities	Jokes acceptable in one culture may cause offense or misunderstanding in another.	High	High	Use region-specific filtering, apply cultural checks, and state intended audience.
3	Incorrect or Illogical Output	The model may produce low-quality, nonsensical, or meaningless jokes.	Medium	Medium	Fine-tune regularly, use user feedback, and evaluate coherence metrics.
4	Tokenizer / Model Errors	Tokenizer or model files may be corrupted, missing, or incompatible.	Low	High	Apply version control, perform file integrity checks, and test compatibility.
5	API or Streamlit Crash	The web interface or backend may crash under heavy loads.	Medium	High	Introduce rate-limiting, monitor load, and implement autoscaling if needed.
6	Excessive Memory or Computation Usage	Long requests or many users may cause GPU memory overflows or slowdowns.	Medium	High	Limit input length, set request caps, and optimize memory usage.
7	Failure to Meet User Expectations	Users may expect every prompt to result in a funny joke.	High	Medium	Add disclaimers, display multiple options, and allow retries.

8	Bias and Stereotypes in Training Data	The model may replicate biased expressions from its data.	Medium	High	Audit data, apply debiasing, and include ethical guidelines.
9	Legal and Copyright Violations	Jokes or datasets may involve copyright infringement or legal issues.	Low	Medium	Use open-license datasets, document sources, and consult legal experts.
10	Adversarial User Attacks	Malicious users may attempt to break or misuse the system.	Medium	High	Sanitize inputs, validate input, and monitor system logs.
11	Security Vulnerabilities	Unexpected inputs might attempt attacks on the app.	Low	High	Escape all content, update dependencies, and conduct security tests.
12	Scalability Issues	If the app becomes popular, server capacity may fall short.	Low	High	Design scalable architecture, use cloud services, and monitor health.
13	Insufficient Computational Resources	Resources may not meet requirements, causing slow or unresponsive models.	Medium	Medium	Assess needs early, upgrade hardware, and optimize efficiency.
14	User Data Mismanagement or Leakage	If user data is mishandled or leaked, privacy breaches may occur.	Low	High	Apply strict privacy policies, avoid storing unnecessary data, and encrypt sensitive information.