T.C. MALTEPE UNIVERSITY FACULTY OF ENGINEERING AND NATURAL SCIENCES DEPARTMENT OF SOFTWARE ENGINEERING

SE40301 Software Project Management Project

Vision And Scope Document

GiggleLab: Artificial Intelligence Based Joke Generator

11 May 2025

Instructor: Prof. Dr. Ensar Gül
ensargul@maltepe.edu.tr

Prepared By:

Student Name	Student ID
Berkay Şahin	21 07 06 034
Selen Bingöl	22 07 06 17
Beyza Çelebi	20 07 06 031
Mustafa Öztürk	21 07 06 016

TABLE OF CONTENT

SE40301 SOFTWARE PROJECT MANAGEMENT PROJECT DOCUMENT 1

1. PROJECT OVERVİEW 3
2. PROJECT OBJECTİVES 3
3. DELİVERABLES 3
4. IN SCOPE 3
5. OUT OF SCOPE 4
6. CONSTRAİNTS 4
7. ASSUMPTİONS 4
8. STAKEHOLDERS 5
9. TİMELİNE 5
10. RESOURCES 5
11. SUCCESS CRİTERİA 6
12. CHANGE MANAGEMENT 6

1. Project Overview

GiggleLab is an AI-powered joke generation project focused on delivering culturally appropriate and humorous Turkish content. The frontend is developed using HTML, CSS, and JavaScript, while the backend uses Python scripts to generate and serve jokes based on a curated dataset.

2. Project Objectives

- Generate Turkish jokes that are meaningful, safe, and humorous
- Ensure fast and smooth interaction between frontend and backend
- Filter inappropriate content and manage prompt moderation
- Build a usable and responsive web-based UI for user interaction

3. Deliverables

- Clean and structured dataset (`fikra_dataset.json`) of Turkish jokes
- Backend scripts: `main.py`, `fikra_llm.py`, `dataset_creator.py`
- Complete frontend UI in HTML, CSS, and JavaScript
- Risk Report, Scope Document, Final Project Report
- Deployed and testable joke generation system (local/demo)

4. In Scope

- Joke dataset creation and cleaning
- Backend AI logic implementation (joke generation engine)
- Frontend design and interaction with backend

- Data filtering and basic content safety measures
- UI/UX compatibility testing

5. Out of Scope

- Advanced deep learning model training from scratch
- Multi-language support
- Integration with commercial or enterprise APIs
- Deployment to scalable cloud services (only local or limited hosting)

6. Constraints

- Limited computational power (no GPU cluster or high-end server)
- Time-bound delivery (~6–8 weeks)
- Dataset limited to available Turkish jokes (public domain only)
- Team size: 4 students with distinct roles

7. Assumptions

- Dataset quality is sufficient to fine-tune or prompt a joke model
- Users understand that content is intended for humor, not offense
- Project can be developed and tested using local machines
- Ethical filtering rules can catch most inappropriate outputs

8. Stakeholders

- Project Team (4 developers)
- Course Instructor
- End Users (interface testers or demo users)
- Presentation Committee

9. Timeline

Week	Task
1–2	Dataset collection and cleanup
3	Backend script development
4	Frontend UI design
5	Integration and internal testing
6	Risk & Documentation writing
7	Final review and testing
8	Submission and project delivery

10. Resources

- Tools: Python, VS Code, JSON, Flask (or raw Python), HTML/CSS/JS
- Data: 1000+ Turkish jokes in structured JSON format
- Platforms: GitHub, Local Python server, Browser testing
- Personnel: 4 developers (data, backend, frontend, coordination)

11. Success Criteria

- Jokes are contextually appropriate and humorous
- The web interface is responsive and functional
- Backend model responds within 1-2 seconds
- System passes 10+ test scenarios without failure
- Ethical filtering catches >90% of problematic prompts

12. Change Management

Any changes to the project scope must be approved by all team members and logged in the project documentation. Major changes (e.g., data format, architecture) must be reviewed with the instructor before implementation.