SecuroGuard

E-Commerce Fake Review Detection Application

Project Policy Document

Submitted by

Faarah Khan (2578-2021)
Hafsa Nisar (1988-2021)

Supervisor

Mr. Afzal Hussain

Co-Supervisor

Mr. Maaz Ahmed

Faculty of Engineering Sciences and Technology

Hamdard Institute of Engineering and Technology

Hamdard University, Main Campus, Karachi, Pakistan

1. Introduction

This document outlines the project policies for the **SecuroGuard** system. These policies ensure effective development, collaboration, and quality control throughout the project lifecycle, from planning to deployment.

2. Purpose

The purpose of this policy is to:

- Define roles and responsibilities clearly.
- Establish rules for development, testing, and communication.
- Ensure quality, security, and ethical handling of data.
- Promote teamwork, accountability, and professionalism.

3. Scope

This policy applies to:

- All team members involved in development, testing, documentation, and deployment of SecuroGuard.
- All tasks related to requirement analysis, UI/UX design, backend logic, AI model integration, and report generation.

4. Communication Policy

- Communication must be clear, respectful, and consistent.
- Official channels:
    - **Email** – for documentation, supervisor updates, and progress sharing.
    - **WhatsApp** – for day-to-day coordination and quick responses.
    - **Weekly meetings** – for status updates, issue tracking, and planning.
- Important decisions must be documented and shared with the supervisor/team.

5. Code Management Policy

- All code will be stored and managed using **Git** (GitHub/GitLab).
- Developers will:
  - Commit changes frequently with clear commit messages.
  - Use **feature branches** for new modules or bug fixes.
  - Ensure **pull requests** are reviewed before merging into the main branch.
- Backup of the repository will be maintained weekly.

6. Testing Policy

- Each feature must be tested using **defined test cases**.
- Types of tests to be conducted:
  - **Unit Testing** – for model and backend functions.
  - **Integration Testing** – for Python-PHP interaction and database connectivity.
  - **System Testing** – full workflow testing across the web interface.
- No feature will be marked as complete until it passes all related tests.

7. Documentation Policy

- Code will include **inline comments** and **function docstrings**.
- A central README will explain project setup and usage.
- User and technical documentation will be updated alongside major changes.
- Screenshots, UI flows, and dataset details will be added where needed.

8. Issue and Risk Management Policy

- All bugs or issues will be logged in a shared issue tracker.
- Each issue will be assigned a **priority level** and **owner**.
- Any risk or critical issue will be immediately escalated to the project lead or supervisor.
- A mitigation plan will be followed for predictable risks like data errors or API failures.

9. Security and Confidentiality Policy

- Sensitive data like model files, review content, and user emails will be stored securely.
- **Google Authentication** is implemented to protect user access.
- All team members will ensure ethical handling of data and AI usage.
- Project data or credentials will not be shared with external parties without permission.

10. Compliance and Review

- All team members must follow this policy document.
- The document will be reviewed during project milestones.
- Repeated non-compliance may result in access restriction or removal from certain roles.
- The project supervisor holds authority to update or revise policies as needed.