test_x.docx

Turnitin

Document Details

Submission ID
trn:oid:::3618:104976968

Submission Date
Jul 18, 2025, 10:34 PM GMT+5

Download Date
Jul 18, 2025, 10:35 PM GMT+5

File Name
test_x.docx

File Size
20.8 KB

4 Pages

1,130 Words

7,143 Characters

24% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Detection Groups

1 AI-generated only 24%
Likely AI-generated text from a large-language model.

0 AI-generated text that was AI-paraphrased 0%
Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

3.3 Website

We have developed a dedicated web application, locally accessible at http://127.0.0.1:8000/, which serves as the primary user interface for our Chronic Kidney Disease (CKD) prediction and test recommendation system. The website provides an intuitive, modern platform that bridges advanced machine learning research with the real-world workflows of healthcare professionals. Its design prioritizes usability, security, and scalability, ensuring both ease of access for end-users and the privacy of sensitive patient data.

The system is tailored for two main user roles: **medical staff** and **doctors**. Staff members are responsible for entering patient details through a structured form, while doctors review submitted cases, interpret CKD predictions, and manage test recommendations. All patient information is securely stored in the underlying database, accessible only to authenticated users with appropriate privileges.

Privacy and Security: The platform is designed to follow best practices for patient data privacy, including secure authentication, role-based access, and encrypted data storage. While currently deployed locally for development and testing, the system architecture supports future integration with hospital networks and compliance with regulations such as GDPR or HIPAA.

User Experience and Accessibility: The interface is responsive, ensuring usability on both desktop and mobile devices. Basic accessibility features such as clear labels, color contrast, and keyboard navigation have been considered to support diverse user needs.

Overall, the website operationalizes our machine learning research, offering a practical, user-friendly tool for early CKD prediction and clinical decision support.

3.3.1 User Interface Components

The website is organized into several core components, each designed to facilitate a smooth and secure workflow for users.

3.3.1.1 Home Page

The Home Page offers a welcoming entry point, branded as "ABC Hospital CKD Prediction System." The navigation bar at the top includes links to Home, Features, About, and Contact pages. On the right, prominent **Register** and **Log In** buttons allow new users to create accounts or returning users to securely access the system. The main banner provides an overview of the platform’s purpose.

Scrolling down, a “Why Choose Our System” section highlights three primary features through visually distinct cards:

- **Ensemble Learning-Based Analysis:** Advanced machine learning models collaboratively analyze patient data to enhance CKD prediction accuracy.
- **Progressive Patient Monitoring:** Tracks each patient’s risk status and testing history, helping clinicians respond effectively to disease progression.
- **Data Security:** Ensures patient privacy with robust encryption and secure login mechanisms.

Figure 3.5: Home Page of the CKD Prediction System (add screenshot here)

3.3.1.2 Authentication Page

The Authentication Page provides secure login and registration for all users (doctors and staff). The login form includes fields for username/email and password, along with error feedback for invalid credentials. Security features such as password hashing, account lockout after repeated failed attempts, and password reset via email can be integrated.

Figure 3.6: Login Form (add screenshot here)

3.3.1.3 Doctor Dashboard

Upon successful login, doctors are redirected to a dashboard summarizing current clinical activity. The dashboard presents:

- Quick stats on total patients, pending cases, and completed reviews.
- A searchable/filterable table of patient records.
- Notifications for new patient entries or abnormal CKD predictions.

Additional features such as data export (CSV/PDF), print, or trend charts may be included for enhanced usability.

Figure 3.7: Doctor Dashboard (add screenshot here)

3.3.1.4 Patient Input Form

Authenticated staff members access the Patient Input Form to register new patients. The form captures essential details, including name, age, contact information, clinical measurements (e.g., blood pressure, albumin), and observed symptoms.

- All fields include validation for accuracy and completeness (e.g., blood pressure must be within a realistic range).
- Dropdown menus and tooltips assist users in entering correct values.
- Upon submission, data is stored in the database and flagged for doctor review.

Figure 3.8: Patient Input Form (add screenshot here)

3.3.1.5 Prediction and Recommendation Result Page

When a doctor selects a patient record, the system displays a detailed results page featuring:

- **Patient Profile:** Name, age, contact info, and status.
- **Prediction Result:** Clearly labeled as “CKD” or “Not CKD,” with color indicators (e.g., red for positive, green for negative).
- **Key Indicators:** Values for serum creatinine, eGFR, albumin, proteinuria, and risk factors such as hypertension, diabetes, blood pressure, and blood glucose.

- **Symptoms Panel:** Displays anemia, appetite status, pedal edema, and hemoglobin levels.
- **Suggested Test Groups:** Lists recommended clinical and laboratory tests based on the patient's symptoms and risk profile.

All data is visually organized with color coding and group labels to help clinicians interpret results quickly. Optionally, doctors may add comments or print/download the results for patient records.

Figure 3.9: Prediction and Recommendation Result Page (add screenshot here)

3.3.2 Technical Stack

3.3.2.1 Backend

- **Framework:** Django (Python), selected for its robustness and scalability in web applications and easy integration with machine learning models.
- **Model Integration:** API endpoints are built to handle model inference, loading the pre-trained CKD prediction model (ckd_model.joblib) from the saved_models/ directory.
- **Security:** Backend endpoints require authentication and perform role checks to ensure only authorized actions.

3.3.2.2 Frontend

- **Technologies:** HTML, CSS, and Tailwind CSS for responsive, clean design.
- **JavaScript:** Used for interactive elements (e.g., input validation, dynamic form feedback).
- **Accessibility:** Semantic markup and ARIA labels improve accessibility.

3.3.2.3 Database

- **Database Engine:** SQLite3 is used for local development, storing patient records, user credentials, and activity logs.
- **Migration Ready:** The system is designed to support migration to PostgreSQL or MySQL for production deployment.

3.3.2.4 API Security and Error Handling

- All API endpoints are protected with authentication tokens.
- Sensitive operations are restricted by user roles (doctor, staff).
- Errors in model prediction or data storage are logged, and users are presented with user-friendly error messages.

3.3.2.5 Scalability and Deployment

- The system is modular and can be deployed on hospital intranets or cloud platforms with minimal changes.

- Static/media files, environment variables, and database settings are separated for production readiness.

3.3.3 Workflow Integration

The workflow supports seamless collaboration between staff and doctors:

1. **Data Submission:** Staff register patients and enter symptom/test data via secure forms.
2. **Model Prediction:** The backend preprocesses input (imputation, scaling), runs the ensemble LSR model, and generates CKD prediction and test recommendations.
3. **Result Review:** Doctors review predictions and recommendations, confirm or adjust as needed, and manage patient care accordingly.

A simple workflow diagram (see Figure 3.10) can be included to illustrate these steps.

3.3.4 Validation and Testing

To ensure the system's reliability and user-friendliness:

- **Unit Testing:** Validates form submissions, model output, and API endpoints.
- **Usability Testing:** Feedback from clinicians guided improvements to UI/UX, field validation, and reporting features.
- **Data Security Testing:** Ensures access control, encryption, and audit logging are functioning as intended.

3.3.5 Challenges and Future Improvements (Optional but recommended)

While the system is robust in its current form, future enhancements may include:

- Integration with hospital information systems (HIS) and cloud-based databases.
- More advanced access control and audit trails for compliance with medical data regulations.
- Support for multi-language interfaces and enhanced accessibility features.
- Ongoing usability testing with broader clinical staff and real patient data.