

Introduction:

In today's healthcare landscape, the ethical use of electronic health records (EHR) is paramount to ensuring patient privacy, data security, and equitable healthcare delivery. The goal is to design a user-friendly AI application that empowers healthcare professionals to upload and analyse EHR datasets for ethics issues, providing actionable insights and suggestions for mitigating risks.

Problem Statement:

Healthcare providers face challenges in navigating the complex landscape of EHR data ethics. Without adequate tools and guidance, they may struggle to identify and address potential risks such as patient consent, data security, and demographic biases. Existing solutions may lack user-friendliness and fail to provide actionable recommendations for mitigating ethical concerns.

Outcome:

The AI application will streamline the process of analysing EHR datasets for ethics issues, providing healthcare professionals with actionable insights and suggestions for mitigating risks. By integrating AI algorithms and ethical frameworks, the application will empower users to make informed decisions and uphold ethical standards in EHR data management.

Challenges:

- Understanding user needs and pain points in navigating EHR data ethics.
- Designing an intuitive user interface for uploading datasets and interpreting analysis results.
- Integrating AI algorithms and ethical frameworks to automate the analysis process.
- Ensuring data security and privacy throughout the application's lifecycle.

- Providing actionable recommendations that are tailored to the user's context and preferences.

User Profiles:

1. Medical Practitioner:

- Role: Physicians, Nurses, Medical Assistants
- Experience: Limited time for data analysis, may lack expertise in data ethics
- Goals: Quickly upload and analyse EHR datasets, receive actionable insights for patient care improvement
- Pain Points: Complex user interfaces, lack of guidance on ethical considerations

2. Healthcare Administrator:

- Role: Hospital Administrators, Healthcare Executives
- Experience: Oversight of data management and compliance
- Goals: Ensure ethical data practices, mitigate risks of data breaches and compliance violations
- Pain Points: Identifying and addressing ethics issues, integrating ethical frameworks into existing workflows

Process

Research:

- Understanding User Needs: Conduct user interviews, surveys, and market research to identify the pain points and needs of potential users regarding ethical AI analysis.
- Competitive Analysis: Evaluate existing tools and platforms that offer similar functionalities to understand their strengths, weaknesses, and user experiences.
- Ethical Frameworks: Research and select widely accepted AI ethics frameworks (e.g., IEEE Ethically Aligned Design, EU AI Ethics Guidelines) to guide the analysis process.
- AI Algorithms: Research and select different types of algorithms that could be applied and used to automate the analyse of the reports.

Ideation:

- Brainstorming Features: Generate ideas for features and functionalities based on user needs and research findings, such as dataset upload, framework selection, analysis, result visualisation, and mitigation suggestions.
- Information Architecture: Organize the application's structure and content logically, ensuring easy navigation and access to key features.
- Flow Mapping: Create user flow and user journey to help mockups sketches to visualize the user interface, focusing on simplicity, clarity, and ease of use.

Design:

- User Interface Design: Design a clean and intuitive interface that guides users through the analysis process step by step, with clear instructions and visual cues.
- Design Concepts: Create wireframes and mockups to visualize the user interface and user flow, focusing on simplicity, clarity, and ease of use.
- Prototyping: Develop interactive prototypes to test key features and gather feedback from potential users.

User Upload Dataset file and set analyses settings

AI.EHR

- Dashboard
- Analyse
- Reports
- Data Management
- Data Privacy & Security
- Appendices
- Help & Support
- Settings

Dr. Monica Rembeau
 Physician

Analyse

Upload
Run Analysis
Generated Report

Upload Area

Drag and drop file here
Drag and drop or browse your file to get started

Browse your file

XLSX
Records 1 1.4 GB
X

Fine Tune

Parameters

Select Parameters

Ethics Frameworks

Select Frameworks

Algorithms

Select Algorithms

Timeframe

Select date range

Analyse Dataset

Dataset Preview

Records 1

Patient	ID	Age	Gender	Race	Diagnose	Procedure
22/10/2022	Bridge of clay	Markus Suzak	George R.R Martin	George R.R Martin	George R.R Martin	George R.R Martin
22/05/2022	Do Epic Shit	Ankur Warikoo	George R.R Martin	George R.R Martin	George R.R Martin	George R.R Martin
14/07/2020	My Sister's Keeper	Jodi Picoult	George R.R Martin	George R.R Martin	George R.R Martin	George R.R Martin

File Analyses

AI.EHR

- Dashboard
- Analyse
- Reports
- Data Management
- Data Privacy & Security
- Appendices
- Help & Support
- Settings

Dr. Monica Rembeau
 Physician

Analyse

Upload
Run Analysis
Generated Report

Upload Area

File: Report 1 here
Drag and drop or browse your file to get started

Browse your file

XLSX
Records 1 1.4 GB
X

Running Analysis

File: Report 1 here
Drag and drop or browse your file to get started

33% 33%

Analyse Dataset

Data Preprocessing

Ethical Framework Analysis 18 / 18
Algorithm Analysis 18 / 18

Dataset Preview

Records 1

Patient	ID	Age	Gender	Race	Diagnose	Procedure
22/10/2022	Bridge of clay	Markus Suzak	George R.R Martin	George R.R Martin	George R.R Martin	George R.R Martin
22/05/2022	Do Epic Shit	Ankur Warikoo	George R.R Martin	George R.R Martin	George R.R Martin	George R.R Martin
14/07/2020	My Sister's Keeper	Jodi Picoult	George R.R Martin	George R.R Martin	George R.R Martin	George R.R Martin

Generate Report

AI.EHR

- Dashboard
- Analyse
- Reports
- Data Management
- Data Privacy & Security
- Appendices
- Help & Support
- Settings

Dr. Monica Rembeau
Physician

Analyse

Upload Area

Drag and drop file here
Drag and drop or browse your file to get started

Browse your file

Records 1 1.4 GB

Fine Tune

Parameters Select Parameters

Ethics Frameworks Select Frameworks

Algorithms Select Algorithms

Timeframe Select date range

Analyse Dataset

Dataset Preview

Records 1

Report List

File	Size	Actions
Report 1	12KB	🔗

Report Visualisation

AI.EHR

- Dashboard
- Analyse
- Reports
- Data Management
- Data Privacy & Security
- Appendices
- Help & Support
- Settings

Dr. Monica Rembeau
Physician

Reports

Download Report

Overview

Lorem ipsum dolor sit amet consectetur. Orci faucibus facilisis suspendisse vestibulum. Hac donec duis risus nisi vitae. Eleifend tellus tellus turpis nunc nam nibh at. Nunc cursus dolor accumsan morbi amet morbi.

Issues graphs

Data Privacy Violations

Bias in Decision Making

Security Breaches

Inaccurate Data

Lack of Consent

Unintended Data Use

Chart of the Month - October

Development:

- Frontend Development: Implement the user interface design using HTML, CSS, and JavaScript, ensuring responsiveness and compatibility across devices and browsers.
- Backend Development: Develop the backend logic using Python Flask to handle dataset uploads, analysis based on selected frameworks, and generation of actionable insights and recommendations.
- Integration of AI Algorithms (Natural Language Processing, Machine Learning Classification, Deep Learning for Image Analysis and Cluster Analysis) and AI Ethics Frameworks (Principle of Biomedical Ethics, Fair Information Practices, Healthcare Ethics Committees, Ethical, Legal, and Social Implications): Incorporate predefined AI Algorithms and AI Ethics frameworks into the application's backend to guide the analysis process and ensure alignment with established ethical principles.

Code Sample:

Code Snippet (*generated on ChatGPT: I used prompts to create the code by inputting what I wanted to build and its features.)

```
from flask import Flask, render_template, request, flash, redirect, url_for
import pandas as pd

app = Flask(__name__)
app.secret_key = 'supersecretkey' # Secret key for flash messages

# Function to detect file format based on file extension
def detect_file_format(filename):
    if filename.endswith('.csv'):
        return 'csv'
    elif filename.endswith('.xlsx') or filename.endswith('.xls'):
        return 'excel'
    elif filename.endswith('.json'):
        return 'json'
    else:
        return None

# Function to validate the uploaded file
def validate_file(df):
```

```

# Check if DataFrame is empty
if df.empty:
    return False, "Uploaded file is empty"

# Add additional validation checks here as needed

# If all checks pass, return True for validation success
return True, ""

# Function for error handling
def handle_error(message):
    flash(message, 'error')
    return redirect(url_for('upload_form'))

# Function for success message
def handle_success(message):
    flash(message, 'success')
    return redirect(url_for('upload_form'))

# Render the upload form
@app.route('/')
def upload_form():
    return render_template('upload_form.html')

# Process the uploaded dataset
@app.route('/upload', methods=['POST'])
def upload():
    if 'file' not in request.files:
        return handle_error("No file part")

    file = request.files['file']

    if file.filename == "":
        return handle_error("No selected file")

    if file:
        # Detect file format
        file_format = detect_file_format(file.filename)

        if file_format is None:
            return handle_error("Unsupported file format")

        # Read the uploaded file
        try:
            if file_format == 'csv':
                df = pd.read_csv(file)
            elif file_format == 'excel':
                df = pd.read_excel(file)
            elif file_format == 'json':
                df = pd.read_json(file)
        except Exception as e:
            return handle_error(f"Error reading file: {str(e)}")

```

```

# Validate the uploaded file
is_valid, validation_message = validate_file(df)
if not is_valid:
    return handle_error(validation_message)

# Process the dataset (Example: extract first 5 rows)
processed_data = df.head()

# Convert processed data to HTML table
processed_data_html = processed_data.to_html()

# Provide success message with dataset details
success_message = f"Upload successful. Dataset contains {len(df)} rows and
{len(df.columns)} columns."
return handle_success(success_message)
else:
    return handle_error("Upload failed")

if __name__ == '__main__':
    app.run(debug=True)

```

6. Testing:

- Usability Testing: Conduct usability testing with real users to identify any usability issues, gather feedback on the user experience, and validate design decisions.
- Functionality Testing: Test all features and functionalities of the application to ensure they work as intended, including dataset upload, analysis, result visualisation, and mitigation suggestions.
- Performance Testing: Evaluate the application's performance, responsiveness, and scalability under different usage scenarios and load conditions.

7. Deployment:

- Beta Release: Deploy the application to a limited audience for beta testing and gather feedback on its performance, usability, and effectiveness in addressing user needs.

- Iterative Improvements: Continuously iterate and improve the application based on user feedback, bug reports, and emerging best practices in AI ethics.
- Full Release: Launch the fully refined and optimised version of the application to the public, accompanied by marketing efforts to promote its availability and attract users.

8. Evaluation:

- User Feedback: Collect ongoing feedback from users to assess their satisfaction with the application, identify areas for further improvement, and prioritise future development efforts.
- Impact Assessment: Evaluate the application's impact on promoting ethical AI practices, raising awareness of AI ethics issues, and facilitating informed decision-making among users.
- Iterative Development: Use insights from user feedback and impact assessment to inform iterative development cycles, ensuring the application continues to evolve and meet the changing needs of its users.

Insights

- The prototypes feature a clean and intuitive interface for uploading EHR datasets, selecting ethical frameworks, and viewing analysis results.
- Users can easily interpret analysis results, including visualisations of ethics issues identified in the dataset and recommended mitigation strategies.
- The application leverages AI algorithms to automate the analysis process, saving time and effort for healthcare professionals.

What Could Have Been Done Differently:

- More extensive user research could have been conducted to capture a broader range of user perspectives and requirements.

- Greater collaboration with healthcare organisations and regulatory bodies could have ensured alignment with industry standards and regulations.
- Continuous iteration and refinement of the application based on user feedback and real-world usage could have further improved its usability and effectiveness.