

**DEPARTMENT**

**OF**

**COMPUTER**

**APPLICATIONS**

National

Institute

of

Technology

Kurukshetra

Haryana, India

**Report**

**for**

**Semester**

**Project Assignment**

**Topic:**

***“***

***FitSense AI : “AI Based Personalized Workout Plan.”   
Workout Plan***

**MCA**

**-**

**202**

**(Artificial Intelligence)**

**Submitted**

**By:**

**Submitted**

**to:**

Shubham Karmveeer

523

110049

Dr. Anshu Parashar

**Title: FitSense AI : *AI Based Personalized Workout Plan*.**

<https://fit-sense-ai.vercel.app/> [Live Deploymnet]

<https://github.com/karmveershubham/FitSense-AI> [CodeBase]

**Problem Statement:**

In today's fast-paced lifestyle, individuals increasingly struggle to maintain a consistent and effective fitness routine tailored to their unique body metrics, goals, and experience levels. The abundance of generic workout plans and one-size-fits-all solutions often leads to ineffective training, loss of motivation, and even injury.

There is a clear need for a personalized fitness recommendation system that intelligently considers individual attributes—such as age, weight, height, gender, experience level, and calorie goals—to suggest optimal workout types.

This **FitSense AI** addresses the gap by leveraging data-driven insights and predictive modeling to deliver customized workout recommendations. It empowers users with tailored fitness guidance, enhances user engagement, and promotes healthier lifestyle habits through a responsive, intuitive, and user-centric digital experience.

**Objectives:**

1. **Analyze Fitness Tracker Data**
   * Study key metrics from the dataset, such as steps, calories burned, heart rate, and workout durations.
   * Identify trends in user activity, including workout frequency and intensity.
2. **Highlight Areas for Improvement**
   * Use the data to pinpoint actionable insights, such as optimizing workout duration or improving hydration habits.
   * Provide suggestions for enhancing daily activity levels and overall fitness consistency.
3. **Provide Basic Recommendations**
   * Offer simple, data-backed advice to improve workout routines, such as adjusting session duration or focusing on specific workout types.
   * Suggest realistic goals based on current performance trends in the dataset.

**Methodology**:

**1. Data Collection & Preprocessing**

* A curated fitness dataset containing user demographic and biometric attributes (age, gender, weight, height), along with calories burned, and experience level was used.
* Engineered additional features:
  + Workout Intensity (Calories / (Weight × Duration))
  + Calories per Minute
  + BMI and BMI Category
* Handled missing values and outliers using statistical techniques to ensure data quality.

2**. Exploratory Data Analysis (EDA)**

* Univariate and bivariate analysis conducted to understand:
  + Distributions of key variables (age, weight, calories burned, workout intensity).
  + Relationships between demographic features and calories burned.
  + Correlations using heatmaps and pairplots.
* Insights guided model features and form design.

3**. Model Building**

* Built a supervised machine learning model to predict Workout Type based on:
  + Input Features: Age, Gender, Weight, Height, Experience Level, Calories to Burn.
* Used classification algorithms (e.g., Decision Tree, Random Forest) with cross-validation.
* Evaluated performance using accuracy, precision, recall, and confusion matrix.

4**. Web App Development**

* Developed using Next.js frontend with Tailwind CSS for a sleek, responsive UI.
* User input form collects all necessary parameters for prediction.
* On form submission:
  + Sends a POST request to the backend API (/predict) with the user data in JSON format.
  + Receives and displays personalized workout recommendations.

**5. Deployment & Integration**

* Model exposed via REST API Flask.
* Deployed the full-stack app on platforms (Render)
* Used .env.local for managing API endpoint securely.

6. Validation & Feedback

* Collected user feedback on prediction relevance and interface usability.
* Adjusted feature weighting and model tuning based on validation results.

**Data Source:** [**Kaggle Fitness Tracker Dataset**](https://www.kaggle.com/datasets/nadeemajeedch/fitness-tracker-dataset/data?select=gym_members_exercise_tracking_synthetic_data.csv).

**Technologies Used**

The project leverages a modern technology stack to deliver a robust, scalable, and user-friendly fitness recommendation system. The primary technologies used are:

**Backend: Python**

* Language: Python was used for building the backend logic and model-serving API due to its simplicity and extensive machine learning ecosystem.
* Model Creation:
  + Utilized scikit-learn for building and training the classification model.
  + Joblib was employed for efficient model serialization and deserialization, enabling fast loading in production environments.
* API Layer: Flask (or FastAPI, if used) served as the REST API to handle prediction requests from the frontend.

**Machine Learning**

* Libraries:
  + scikit-learn: Model training and evaluation
  + pandas, numpy: Data manipulation and feature engineering
  + joblib: Model persistence and deployment-ready storage

**Frontend: Next.js**

* Framework: Next.js was chosen for its hybrid static & server-side rendering capabilities, enhancing performance and SEO.
* Styling:
  + Tailwind CSS for responsive, utility-first UI design.
  + Fully mobile-responsive with consistent layout and theming.
* Form Features:
  + Real-time validation
  + Responsive dropdowns and number inputs
  + Visual feedback during loading states

**Integration**

* The frontend communicates with the backend using secure POST requests to a .env-configured API endpoint.
* Responses from the backend are used to display dynamic, personalized workout recommendations.

**Design Summary**

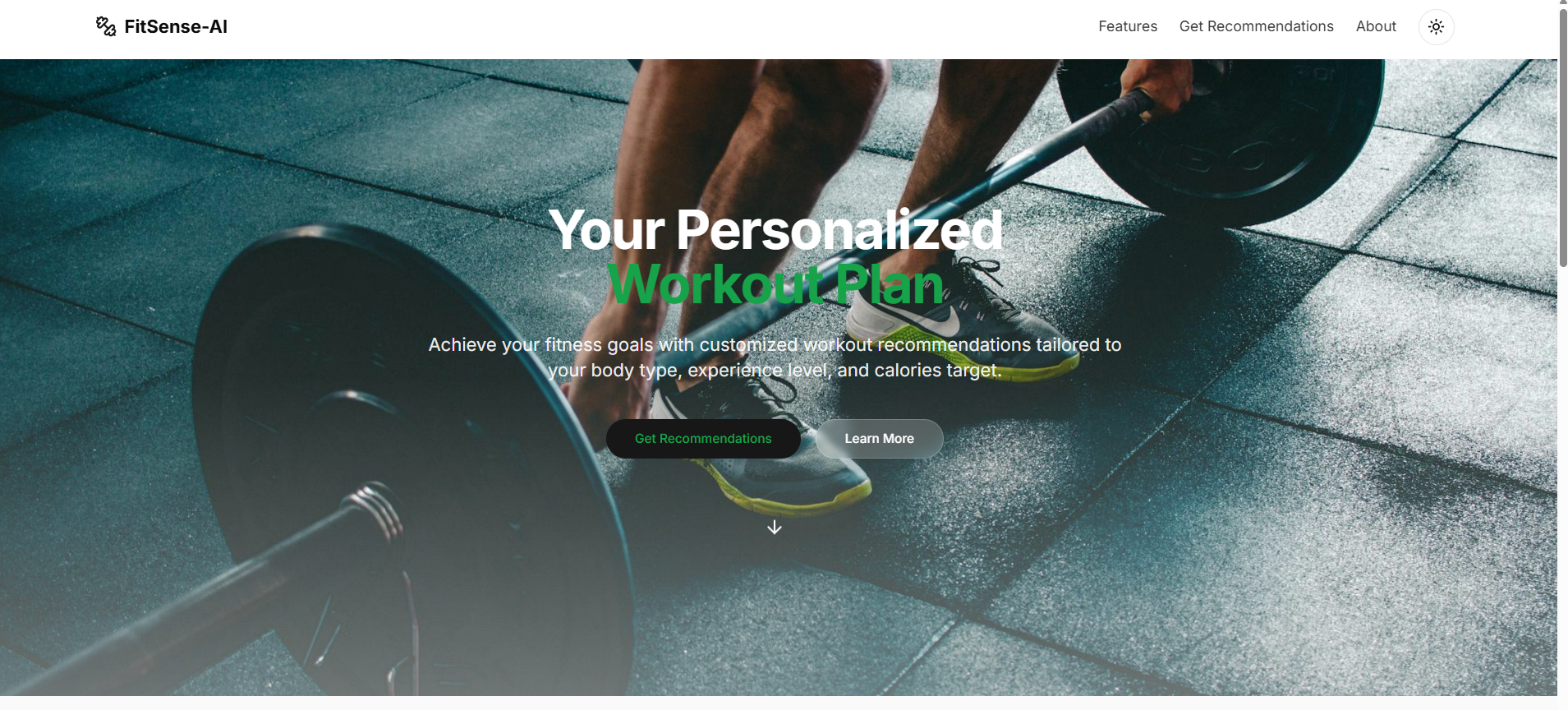
The application is designed with a strong focus on **user experience, responsiveness, and clarity**, making it both functional and aesthetically appealing. The core pages include:

**Landing Page**

**Purpose**:  
To create an impactful first impression and immediately convey the value of the app.

**Key Design Elements**:

* **Hero Section**: Bold typography with dynamic text ("Your Personalized Workout Plan") over a high-quality background image of fitness activity (lifting weights), setting the tone.
* **Call to Action**: Dual buttons – *Get Recommendations* and *Learn More* – styled with Tailwind for high contrast and visibility.
* **Navigation Bar**: Clean, minimal, and aligned right; includes links to *Features*, *Get Recommendations*, and *About* with a dark/light mode toggle.

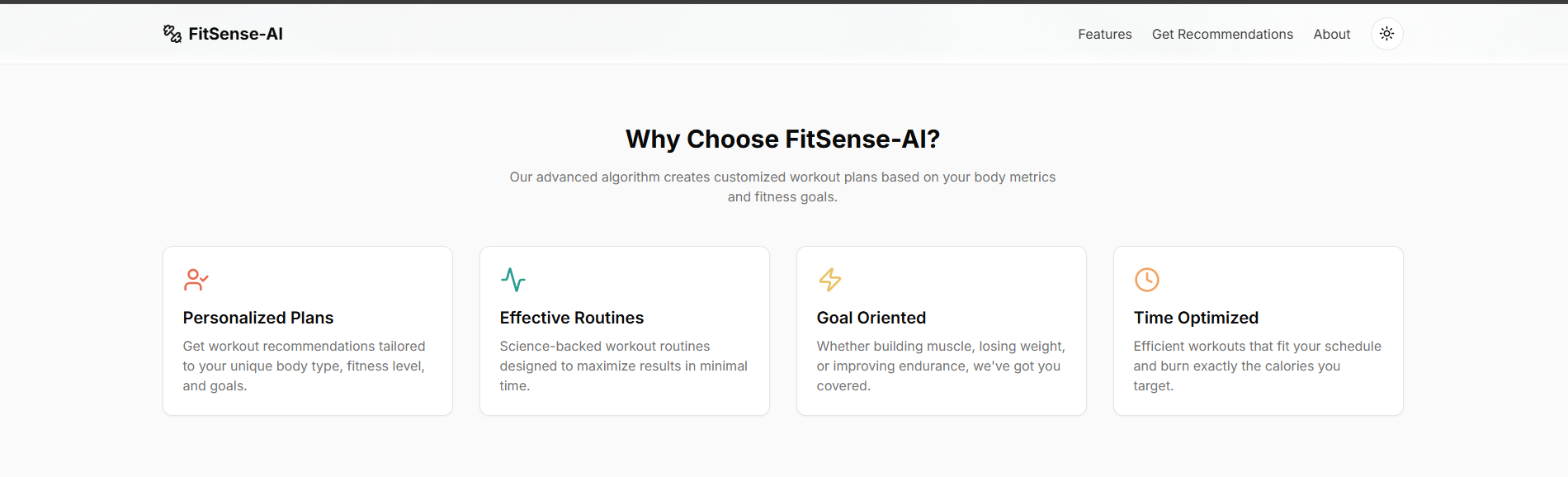


**Features Page**

**Purpose**:  
To inform users of the application’s capabilities and how it benefits them.

**Key Design Elements**:

* **Sectioned Layout**: Cards or section blocks each explaining a feature (e.g., Real-Time Predictions, Personalized Workouts, Responsive Design).
* **Icons & Illustrations**: Simple, fitness-themed visuals or vector icons for engaging UX.
* **Consistent Spacing**: Clear separation of sections for readability.
* **Interactive Feedback**: Hover states and transitions implemented using Tailwind for smoother experience

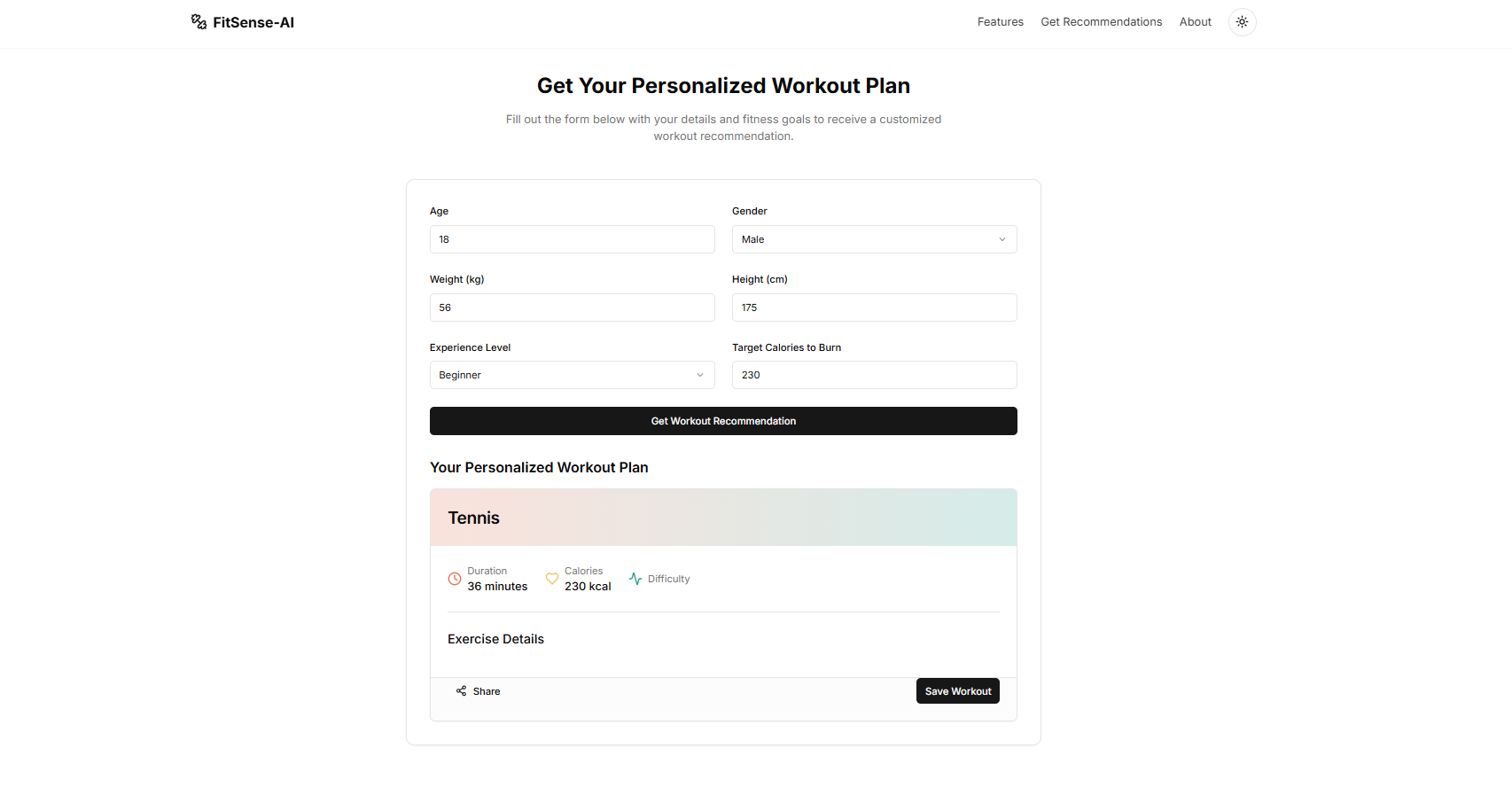


**Get Recommendations Page**

**Purpose**:  
To collect user details and provide intelligent workout suggestions.

**Key Design Elements**:

* **Form Components**: Dropdowns (for gender and experience level), number inputs (for age, weight, height, calories).
* **Validation**: Real-time input checks with error prompts.
* **Submit Button**: Clearly visible, loading state enabled while request is processing.
* **Response Display**: Once recommendation is fetched, it is presented in a clean card layout with actionable insights.
* **Responsiveness**: The layout adjusts seamlessly to desktop, tablet, and mobile screen sizes.



**Conclusion**

The **FitSense-AI** system successfully bridges the gap between personalized fitness guidance and accessible web technology. By integrating a trained machine learning model with a sleek and responsive frontend built using Next.js, the application offers users instant, data-driven workout recommendations tailored to their physical attributes and fitness goals.

The platform emphasizes **ease of use, real-time feedback**, and **user-centric design**, ensuring a smooth and engaging experience for users across all devices. The seamless interaction between the form inputs and prediction API demonstrates the power of combining modern web frameworks with intelligent backend systems.

Overall, this project showcases how AI can personalize health and fitness journeys, making workouts more effective and aligned with individual needs.

**Future Enhancements**

To further enrich the functionality and impact of **FitSense-AI**, the following feature enhancements are proposed:

**1. Workout Schedule Generator**

* Automatically generate weekly workout plans based on user inputs and goals.
* Include rest days, workout durations, and targeted muscle groups.

**2 Progress Tracking Dashboard**

* Allow users to save progress and visualize changes over time (e.g., calories burned, workouts completed).
* Integrate charts and trend analysis using libraries like Chart.js or Recharts.

**3. Workout Demonstration Videos**

* Embed short tutorial videos or animated GIFs showing correct form and techniques for each recommended workout.

**4. More Intelligent Recommendations**

* Enhance the machine learning model to include more features like:
  + User goals (e.g., weight loss, muscle gain, endurance)
  + Medical conditions or injuries
  + Time available per workout

**5. User Authentication and Profile**

* Add login/signup functionality to save personalized workout history.
* Allow users to update their information and preferences.

**6. Multilingual Support**

* Add localization to support users from different language backgrounds.

**7. Mobile App Integration**

* Build a React Native version of the app or create a PWA (Progressive Web App) for better mobile accessibility.

**References**

**Machine Learning & Backend Technologies**

1. **Python Programming Language**  
   Python Software Foundation. <https://www.python.org>
2. **Joblib Library for Model Serialization**  
   Joblib Documentation. <https://joblib.readthedocs.io>
3. **Flask (for API Backend)**  
   Flask – Pallets Projects. https://flask.palletsprojects.com

**Frontend Technologies**

1. **Next.js – React Framework**  
   Vercel Inc. <https://nextjs.org>
2. **Tailwind CSS – Utility-First CSS Framework**  
   Tailwind Labs Inc. <https://tailwindcss.com>
3. **React – JavaScript Library for Building UIs**  
   Meta Platforms, Inc. <https://reactjs.org>

**Dataset**

1. **Data Source:** [**Kaggle Fitness Tracker Dataset**](https://www.kaggle.com/datasets/nadeemajeedch/fitness-tracker-dataset/data?select=gym_members_exercise_tracking_synthetic_data.csv).
2. The dataset used for training the workout recommendation model was curated manually from:
   * Public health and fitness research sources
   * Fitness tracking insights
   * User-submitted workout logs  
     (Note: Dataset preprocessing, feature engineering, and labeling were done manually to ensure relevance and consistency.)