



Project Report On “Predico – Disease Prediction System”

Name	Regd No	Branch
Aman Kumar Pattanaik	2301030051	CSE
Prasuna Dash	2301030048	CS-IT
B Raj Kumar Patro	2301030023	CS-IT
Subham Rout	2301030098	CS-IT
Aliva Priyadarshini Sahoo	2301030069	CS-IT
Baijayanti Mahanta	2301030125	COE

Contents

1. Introduction
2. Objective
3. Scope
4. Features
5. Technology Stack
6. System Architecture
7. Dataset and Data Collection
8. Machine Learning Model
9. System Interfaces
10. Challenges
11. Learnings
12. Future Scope
13. Conclusion
14. References

1.Introduction:

Predico is an innovative web-based platform that bridges the gap between early disease detection and timely medical consultation. Built using Django and powered by advanced machine learning algorithms, Predico empowers users to actively participate in their healthcare journey. By allowing individuals to input their symptoms and instantly receive potential diagnoses, the system plays a crucial role in early intervention and preventive care. In addition to disease prediction, the platform offers seamless access to online consultations with recommended doctors, thus ensuring that medical advice is just a click away. Predico is thoughtfully designed to cater to the unique needs of patients, healthcare professionals, and administrators, all while maintaining a user-friendly interface and secure data environment.

2. Objective:

- To harness the power of machine learning to accurately predict potential diseases based on diverse and dynamic symptom inputs from users.

- To enhance healthcare accessibility by offering real-time online consultations with qualified medical professionals.
- To implement a secure, role-based digital ecosystem that maintains data integrity and ensures appropriate user access across patients, doctors, and administrators.
- To reduce delays in diagnosis and encourage early intervention, ultimately contributing to improved patient outcomes and reduced burden on traditional healthcare systems.

3. Scope:

Predico has been carefully structured to address the healthcare interaction needs of multiple user categories:

- **Patients:** Users can effortlessly register and securely log in to access a guided interface where they can input their symptoms. Based on the inputs, the system utilizes its trained machine learning model to generate a probable disease prediction. Patients can then view a curated list of suggested doctors and initiate online consultations.

- **Doctors:** After logging in, medical professionals can access patient consultation requests, evaluate the inputted symptoms, and provide medical advice or follow-up actions. Doctors can also maintain records and review consultation histories for better continuity of care.
- **Administrators (Admins):** Responsible for the platform's integrity, administrators have the authority to manage user accounts, approve doctor credentials, monitor consultation logs, and ensure overall system performance.

4. Features:

Predico offers a rich set of functionalities that enhance the healthcare experience for all user roles:

- **Advanced Disease Prediction:** Users can enter symptoms from a predefined list, and the system leverages a trained machine learning model to provide potential diagnoses.
- **Smart Doctor Suggestions:** Once a disease is predicted, relevant doctors are recommended to the patient, streamlining the consultation process.

- **Role-Based Access:** With separate access levels for patients, doctors, and admins, the system ensures that sensitive data is protected and accessible only by authorized users.
- **Seamless Online Consultation:** The platform supports digital interaction between patients and doctors through a dedicated interface, facilitating remote healthcare delivery.
- **Secure User Management and Logging:** The system tracks user activities and maintains history logs for future reference and accountability.

5. Technology Stack:

The development of Predico integrates a range of modern web technologies and data science tools to ensure a seamless and efficient user experience:

- **Frontend Technologies:** HTML, CSS, and Bootstrap form the layout and design framework, while JavaScript and jQuery manage dynamic content and user interactions.
- **Backend Framework:** Django, a robust Python-based framework, handles all server-side logic, including routing, authentication, and form handling.

- **Database:** PostgreSQL is used for storing structured data, including user profiles, consultation logs, and system configurations.
- **Supporting Tools:** PgAdmin is employed for database management, while Orange is used for initial machine learning experimentation and visualization.
- **Machine Learning Libraries:** scikit-learn is used for model training and testing, and joblib for model serialization and loading.

6. System Architecture:

Predico's system architecture is designed to ensure high reliability, modularity, and scalability:

- **Frontend Interface:** Separate user dashboards are implemented for patients, doctors, and administrators, each with specific functionalities relevant to the user's role.
- **Backend Services:** The Django framework handles core logic, data manipulation, authentication, and serves machine learning predictions via API endpoints.

- **Machine Learning Integration:** The ML model is trained separately and then integrated with Django using joblib for runtime predictions.
- **Database Layer:** PostgreSQL serves as the relational database for managing users, symptoms, disease predictions, and consultation records in a normalized structure.

7. Dataset & Data Collection:

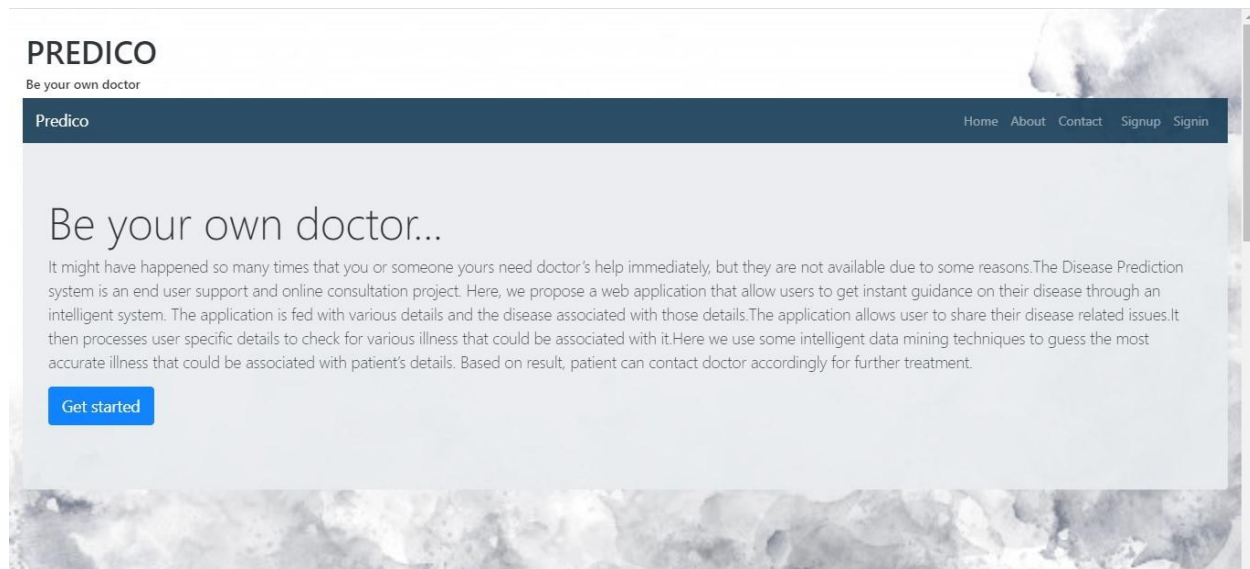
- Dataset collected from [Kaggle](#) and other verified medical websites.
- **Total Records:** 5000 patient entries.
- **Symptoms:** 132 types
- **Disease Classes:** 40 general diseases (e.g., Diabetes, Malaria, Jaundice, etc.)
- Data is real and medically validated, ensuring no dummy values.

8. Machine Learning Model:

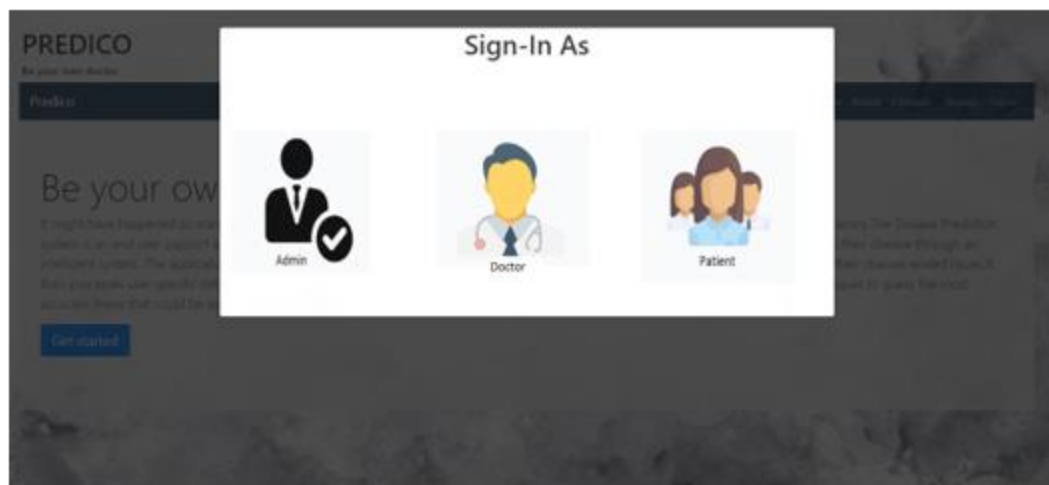
- Trained using **scikit-learn** models like Decision Tree, Random Forest.
- Model evaluated using cross-validation.

- Model serialization done using **joblib** for quick deployment.
- Integrated with Django to provide real-time predictions on symptom input.

9. System Interfaces:



Login Modal-



Login as Patient-

The screenshot shows the 'Login as Patient' page of the PREDICO application. The header includes the PREDICO logo with the tagline 'Be your own doctor' and a navigation bar with links for Home, About, Contact, Signup, and Login. The main heading is 'Login as Patient'. Below this, there are input fields for 'Username' (with a person icon) and 'Password' (with a lock icon), both labeled 'Type your [username/password]'. A 'Forgot password?' link is positioned to the right of the password field. A large, colorful 'LOGIN' button is at the bottom.

PREDICO
Be your own doctor

Predico Home About Contact Signup Login

Login as Patient

Username
Type your username

Password
Type your password

[Forgot password?](#)

LOGIN


Patient UI-

The screenshot shows the 'Patient Profile' page. The header is identical to the login page. The main heading is 'Patient Profile'. On the left, there is a profile card featuring an illustration of three people, the title 'Patient', and the following details: 'Patient name : itishree', 'Patient ID : 2', and 'Patient email : itishreebheera2019@gmail.com'. A 'View Profile' button is at the bottom of the card. On the right, there is a table with three rows: 'check disease', 'view consultation history', and 'Give Feedback', each with a right-pointing arrow icon.

PREDICO
Be your own doctor

Predico Home About Contact **Help, Review** Profile Logout

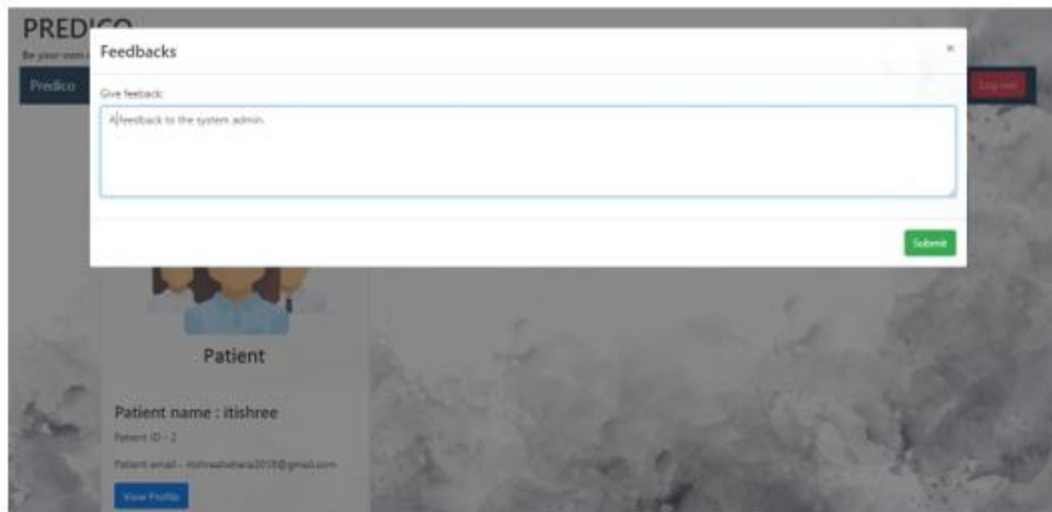
Patient Profile


Patient

Patient name : itishree
Patient ID : 2
Patient email : itishreebheera2019@gmail.com
[View Profile](#)

check disease	→
view consultation history	→
Give Feedback	→

Feedback Form-



The screenshot shows a web application interface for 'Predico'. A 'Feedbacks' modal window is open, allowing a user to provide feedback. The modal has a title bar with 'Feedbacks' and a close button. Inside, there is a text input field with the placeholder text 'Give feedback:' and a pre-filled value 'A feedback to the system admin.'. Below the input field is a green 'Submit' button. In the background, a 'Patient' profile is visible, showing the patient's name 'Itishree', ID '2', and email 'itishreea2019@gmail.com'. There is also a 'View Profile' button.

PREDICO
Be your own doctor

Predico

Feedbacks

Give feedback:

A feedback to the system admin.

Submit

Patient

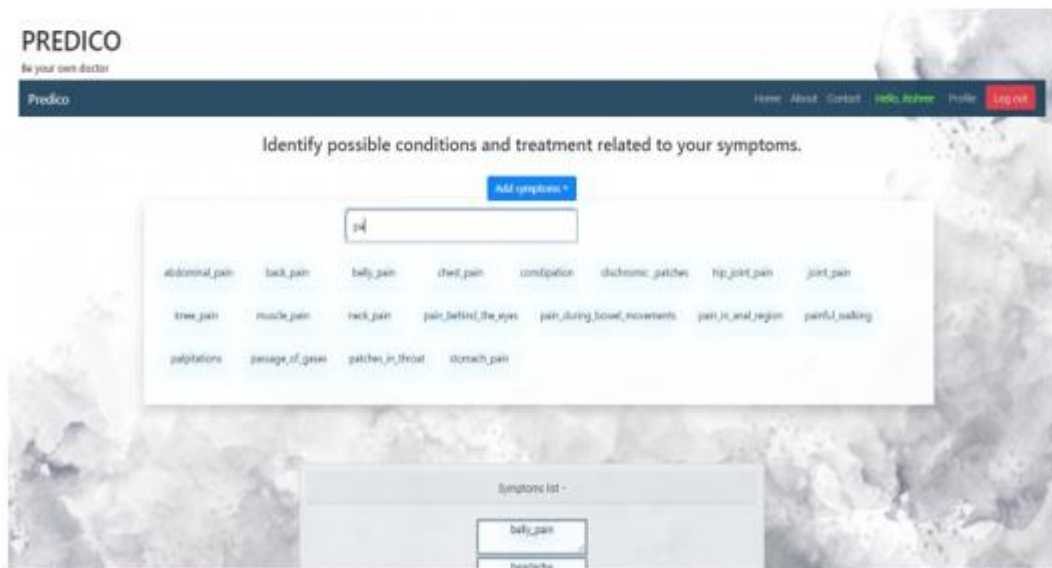
Patient name : Itishree

Patient ID : 2

Patient email : itishreea2019@gmail.com

View Profile

Check Disease- Entering symptoms



The screenshot shows the 'Check Disease' interface of the 'Predico' web application. The header includes the 'Predico' logo and navigation links: 'Home', 'About', 'Contact', 'Hello, Itishree', and 'Profile'. A 'Log out' button is also present. The main heading is 'Identify possible conditions and treatment related to your symptoms.' Below this is an 'Add symptoms +' button. A search input field contains the text 'p4'. Below the input field is a grid of symptom buttons: 'abdominal_pain', 'back_pain', 'belly_pain', 'chest_pain', 'constipation', 'dichromic_patches', 'hip_joint_pain', 'joint_pain', 'knee_pain', 'muscle_pain', 'neck_pain', 'pain_behind_the_eyes', 'pain_during_bowel_movements', 'pain_in_anal_region', 'painful_swallow', 'palpitations', 'passage_of_gas', 'patches_in_throat', and 'stomach_pain'. At the bottom, there is a 'Symptoms list -' section with a table containing 'belly_pain' and 'headache'.

PREDICO
Be your own doctor

Predico

Home About Contact Hello, Itishree Profile Log out

Identify possible conditions and treatment related to your symptoms.

Add symptoms +

p4

abdominal_pain back_pain belly_pain chest_pain constipation dichromic_patches hip_joint_pain joint_pain

knee_pain muscle_pain neck_pain pain_behind_the_eyes pain_during_bowel_movements pain_in_anal_region painful_swallow

palpitations passage_of_gas patches_in_throat stomach_pain

Symptoms list -

belly_pain
headache

Predictions-

chills

vomiting

headache

nausea

muscle_weakness

Predict

Patient name : itishree Age : 22
predicted disease is : **Malaria**
confidence score of: 75%

[Click here to know more about: Malaria](#)

This tool does not provide medical advice. It is intended for informational purposes only.
It is not a substitute for professional medical advice, diagnosis or treatment.

[Consult a Allergist/immunologist doctor](#)

Consult a Doctor-

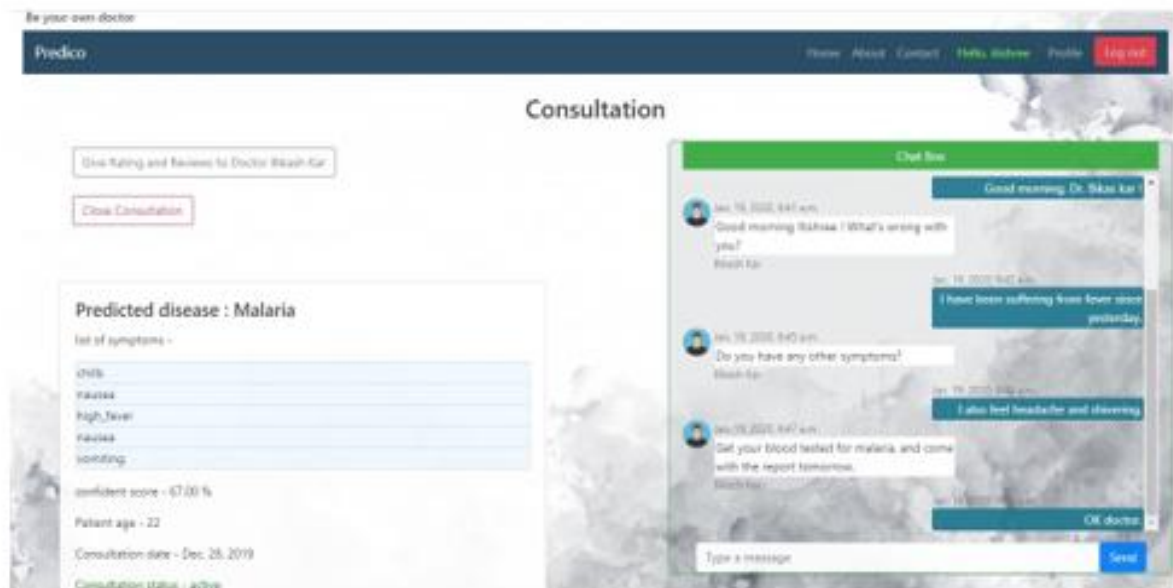
PREDICO
Be your own doctor

Predicto [Home](#) [About](#) [Contact](#) [Help](#) [Admin](#) [Profile](#) [Log out](#)

Consult a Doctor

Doctor name	Specialization	Email	Rating	View profile	Consult
Shash Kar	Dermatologist	shashkar1@gmail.com	2/5	view profile	Consult
anj	Cardiologist	anugredhan444@gmail.com	3/5	view profile	Consult

Consultation UI-



Consultation history- (Doctor)

PREDICO
Be your own doctor

Predico Home About Contact Hello Dr. Ekash Kar Profile Log out

Consulation History


Patient name	Patient Email	View Patient's profile	Predicted Disease Name	Consultation Date	Consultation Status	Resume Consultation
Ekash	ekashbhatia2018@gmail.com	view profile	Vertigo/Parosmal Positional Vertigo	Dec. 27, 2019	active	Consult
Ekash	ekashbhatia2018@gmail.com	view profile	Malaria	Dec. 26, 2019	active	Consult
Ekash	ekashbhatia2018@gmail.com	view profile	Acidosis	Jan. 11, 2020	closed	Consult
Ekash	ekashbhatia2018@gmail.com	view profile	Heart attack	Jan. 16, 2020	active	Consult

Admin Sign in-

PREDICO
Be your own doctor

Predico Home About Contact Signin Signup

Admin Signin



Username

Password

Signin

[Forgot your password?](#)

Admin Interface-

Django administration HELLO! admin user info / Django Admin - Log Out

Site administration

Authentication and authorization	
Groups	Add Change
Users	Add Change

CMS	
Chats	Add Change
Feedbacks	Add Change

Main app	
Consultations	Add Change
Examinations	Add Change
Doctors	Add Change
Patients	Add Change
Patient reviews	Add Change

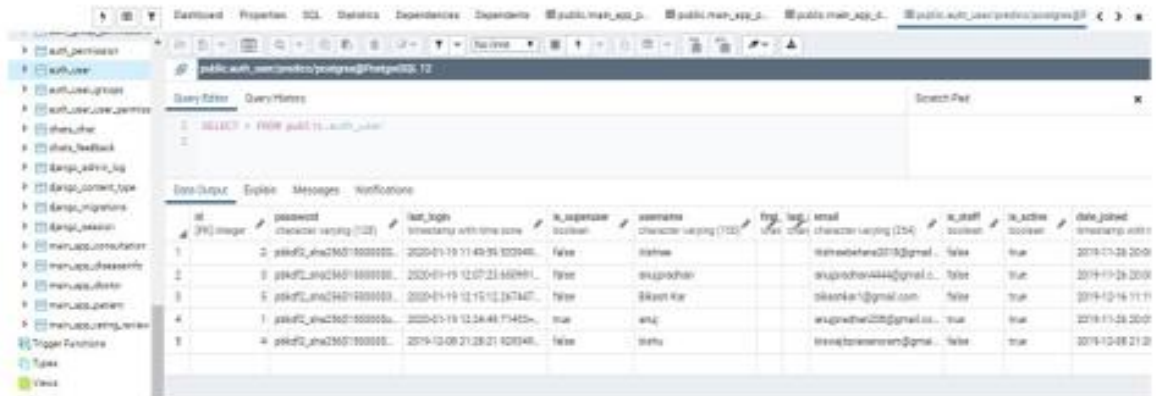
Recent actions

My actions

- ❌ Chat object (16)
- ✓ Consultation object (2)
- ✓ Feedback object (13)
- ❌ Feedback object (12)
- ❌ Feedback object (1)
- ❌ Feedback object (3)
- ❌ Feedback object (3)
- ❌ Feedback object (4)
- ❌ Feedback object (5)
- ❌ Feedback object (6)

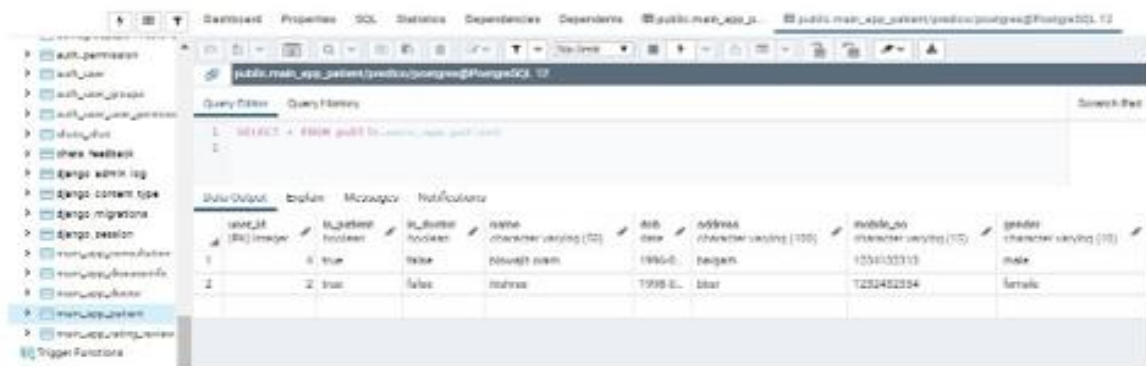
6. Database ~~predico~~

Users table-



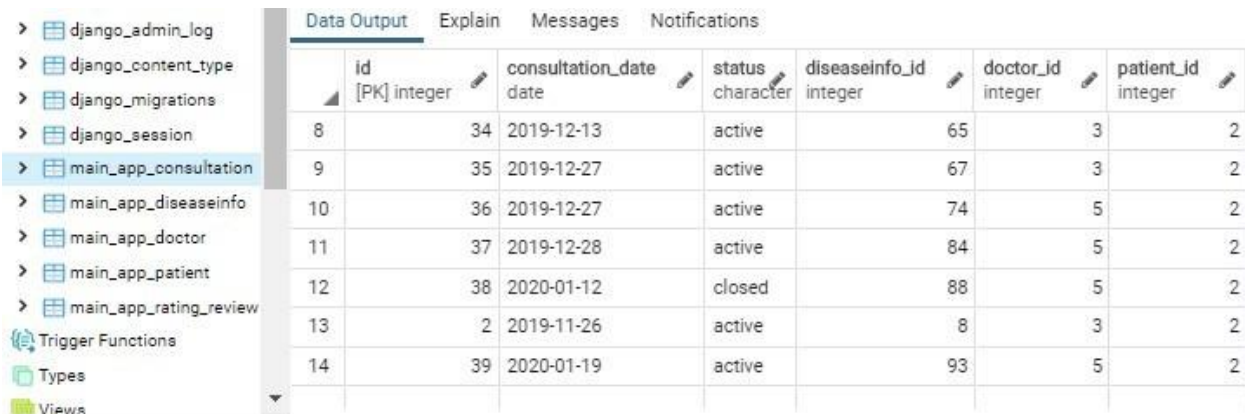
id	password	last_login	is_superuser	username	first_name	last_name	email	is_staff	is_active	date_joined
1	pbkdf2_sha256\$100000\$...	2020-01-19 11:49:55.000000	False	admin			admin@admin.com	True	True	2019-01-28 20:00
2	pbkdf2_sha256\$100000\$...	2020-01-19 12:07:23.000000	True	superadmin			superadmin444@gmail.com	True	True	2019-01-28 20:00
3	pbkdf2_sha256\$100000\$...	2020-01-19 12:15:12.000000	True	Elkann Kar			elkanna1@gmail.com	True	True	2019-01-16 11:11
4	pbkdf2_sha256\$100000\$...	2020-01-19 12:24:48.000000	True	anj			anjanthar20@gmail.com	True	True	2019-01-28 20:00
5	pbkdf2_sha256\$100000\$...	2019-12-08 21:28:21.000000	False	delia			delia@delia.com	False	True	2019-12-08 21:28

Patient table-



id	is_active	is_superuser	name	dob	address	mobile_no	gender
1	True	False	Elkann Kar	1995-01-01	delia	1234567890	Male
2	True	False	delia	1999-01-01	delia	1234567890	Female

Consultation table-



	id	consultation_date	status	diseaseinfo_id	doctor_id	patient_id
	[PK] integer	date	character	integer	integer	integer
8	34	2019-12-13	active	65	3	2
9	35	2019-12-27	active	67	3	2
10	36	2019-12-27	active	74	5	2
11	37	2019-12-28	active	84	5	2
12	38	2020-01-12	closed	88	5	2
13	2	2019-11-26	active	8	3	2
14	39	2020-01-19	active	93	5	2

10. Challenges Faced:

- Data cleaning and symptom standardization.
- Model performance tuning.
- Integrating ML predictions with a live Django app.
- Designing a secure and scalable database.

11. Learnings:

- Practical implementation of ML in web applications.
- Role-based access control using Django.
- Handling structured medical data.
- Real-time model deployment and integration.

12. Future Scope:

- Addition of **voice-based symptom input**.
- Expansion to include more diseases and multilingual support.
- Integration with **wearables and IoT devices**.
- Collaboration with healthcare institutions for real-world deployment.

13. Conclusion:

- Predico represents a significant leap forward in the digitization of primary healthcare services. By integrating artificial intelligence with an intuitive web interface, the system empowers users to take a proactive approach toward health management. It simplifies the process of disease prediction by allowing users to input symptoms and instantly receive likely diagnoses—reducing dependency on traditional diagnostic cycles that are often time-consuming and resource-intensive.
- The real value of Predico lies not just in its predictive capabilities, but also in how it streamlines the entire healthcare consultation process. Patients are directly connected with appropriate doctors based on the predictions, making the system both responsive and

intelligent. For medical professionals, it offers tools to manage consultations efficiently, ensuring continuity of care through digital records.

- With a well-thought-out architecture and a robust machine learning backbone, Predico sets a strong foundation for future enhancements. It is more than a project—it is a scalable, impactful solution designed to evolve with the growing demands of digital healthcare.

14. References:

This project and its development are supported and inspired by a diverse set of tools, platforms, and educational materials. Below are the references used in crafting the Predico system:

- **Kaggle Dataset:** Disease prediction dataset used for training the machine learning model. Source: [Kaggle - Disease Prediction Dataset](#)
- **Django Documentation:** Comprehensive guide to Django's framework features, authentication, and routing systems. Available at: <https://docs.djangoproject.com>

- **scikit-learn:** The core machine learning library used for model building, training, evaluation, and deployment. Official site: <https://scikit-learn.org>
- **PostgreSQL:** The open-source relational database used to manage and store user data and consultation records. Reference: <https://www.postgresql.org>
- **Orange – Data Mining Tool:** Used for visual exploration of data and initial modeling. More information at: <https://orange.biolab.si>
- **PgAdmin:** A GUI tool for interacting with PostgreSQL, used for database setup and management. Details: <https://www.pgadmin.org>