

# Healthcare Patient Management System Documentation

## Overview

The Healthcare Patient Management System allows managing patient records with operations like adding, removing, updating, searching, and displaying patient data. Additionally, this system can save data into a JSON file and send emails to patients.

## Functionalities:

### 1. Add Patient Data

Allows the user to add new patient records. Each record may contain: - Patient

ID

- Name

- Age

- Medical History

*Method Signature:*

```
@app.route('/patients',methods=['POST']) def
patients_create():
    body = request.get_json()    new_patient = Patient(name=body['name'],
age=body['age'],disease=body['disease'])    id = createPatient(new_patient)    patient =
readPatientById(id)    patient_dict = {'id':patient.id, 'name':patient.name,
'age':patient.age,'disease':patient.disease}    return jsonify(patient_dict)
```

### 2. Remove Patient Data

Removes a patient's record from the system using the unique Patient ID.

*Method Signature:*

```
@app.route('/patients/<id>',methods=['DELETE']) def
patients_delete(id):
    old_record = readPatientById(id)    if not old_record:        return
jsonify({'message': 'Record is not found', 'is_error': 1})
deletePatient(id)    return jsonify({'message': 'Record is successfully
deleted', 'is_error': 0})
```

### 3. Search Patient Data

Allows searching for a patient by their ID or Name.

*Method Signature:*

```
def patients_search(disease):
    patient = searchByDisease(disease)    patient_dict = {'id':patient.id, 'name':patient.name,
'age':patient.age,'disease':patient.disease}    return jsonify(patient_dict)
```

### 4. Display All Patients

Displays all patient records stored in the system.

*Method Signature:*

```
@app.route('/patients',methods=['GET'])
def patients_read_all():
    patients = readAllPatients()
    patients_dict = []
    for patient in patients:
        patients_dict.append({'id':patient.id, 'name':patient.name, 'age':patient.age,'disease':patient.disease})
    return jsonify(patients_dict)
```

### 5. Update Patient Data

Allows updating existing patient records.

*Method Signature:*

```
@app.route('/patients/<id>',methods=['PUT']) def
patients_update(id):
    body = request.get_json()
    old_record = readPatientById(id)    if
not old_record:
        return jsonify({'message': 'Record is not found!'})    old_record.name = body['name']
old_record.age = body['age']    old_record.disease=body['disease']
updatePatient(old_record)    patient = readPatientById(id)    patient_dict = {'id':patient.id,
'name':patient.name, 'age':patient.age,'disease':patient.disease}    return jsonify(patient_dict)
```

### 6. Send Email to Business Owner

Sends an email to a patient (e.g., for appointment reminders or medical follow-ups).

*Method Signature:*

```
import smtplib as smtplib    connection =
smtplib.SMTP_SSL('smtp.gmail.com', 465)    email_addr =
'bhavana9678@gmail.com'    email_passwd = 'oqtc
afjh sgop eglz'    connection.login(email_addr,
email_passwd)    patients=app.read_all()
```

```
patient_json_str=json.dumps(patients)      subject =  
"Patient Records"      body = f'Hi! all the patient records  
are sent in json format.\n{patient_json_str}'      message  
= f'Subject: {subject}\n\n{body}'  
connection.sendmail(from_addr=email_addr,  
to_addrs='bhavanaa9678@gmail.com',msg=message)  
connection.close()  
print("Email sent successfully!")
```

## 7. Save Patient Data to JSON File

Saves the entire patient dataset to a JSON file for persistent storage.

*Method Signature:*

```
def save_to_json(file_path: str) -> None:
```

## 8. Load Patient Data from JSON File

Loads patient data from a JSON file during system initialization or as needed.

*Method Signature:*

```
def load_from_json(file_path: str) -> None:
```

## User Interface and Interaction

Menu Interface

Description: The application provides a menu-driven interface for users to interact with the system.

Menu Options:

1. Add Patient
2. Display all Patient
3. Update Patient
4. Searching patient records
5. Remove Patient Records
6. Save flights to JSON
7. End

## Example Interaction

- **\*\*Adding a Patient: \*\***

```plaintext

Enter Patient Id: 22

Enter Patient name: Bhavz

Enter Patient age: 35

Enter Patient Disease: MU

## Sample Output:

### Add Patient Records:

```
OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Choose one of the options below...
1.Display Patient Records
    2.Get Patient Record By ID,
    3.Add a patient record,
    4.Update an existing patient record,
    5.Delete an existing record,
    6.Search Patient record by the disease
    7.Mail the patient records to Business Owner
    8.Web Scraping
    9.Write Patients Records to JSON file
    10.Display ID's using Threading
    11.Exit

ENTER YOUR CHOICE:3

Enter Name:arthi
Enter age:24
Enter disease that you are diagnosed with:TU
{'age': 24, 'disease': 'TU', 'id': 10, 'name': 'arthi'}
```

### Display Patient Records:

```
OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\work\python\healthcare_patient_management_system\frontend> python patient_app.py
<-----Healthcare Patient Management System----->

Choose one of the options below...
1.Display Patient Records
    2.Get Patient Record By ID,
    3.Add a patient record,
    4.Update an existing patient record,
    5.Delete an existing record,
    6.Search Patient record by the disease
    7.Mail the patient records to Business Owner
    8.Web Scraping
    9.Write Patients Records to JSON file
    10.Display ID's using Threading
    11.Exit

ENTER YOUR CHOICE:1
[{'age': 22, 'disease': 'MU', 'id': 7, 'name': 'Bhavana'}, {'age': 22, 'disease': 'Migrane', 'id': 8, 'name': 'Charulatha'}]
```

### Get Patient Records:

```
OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

<-----Healthcare Patient Management System----->

Choose one of the options below...
1.Display Patient Records
    2.Get Patient Record By ID,
    3.Add a patient record,
    4.Update an existing patient record,
    5.Delete an existing record,
    6.Search Patient record by the disease
    7.Mail the patient records to Business Owner
    8.Web Scraping
    9.Write Patients Records to JSON file
    10.Display ID's using Threading
    11.Exit

ENTER YOUR CHOICE:2

Enter ID:7
{'age': 22, 'disease': 'MU', 'id': 7, 'name': 'Bhavana'}
<-----Healthcare Patient Management System----->
```

## Updating the Existing Record:

```
OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

<-----Healthcare Patient Management System----->

Choose one of the options below...
1.Display Patient Records
    2.Get Patient Record By ID,
    3.Add a patient record,
    4.Update an existing patient record,
    5.Delete an existing record,
    6.Search Patient record by the disease
    7.Mail the patient records to Business Owner
    8.Web Scraping
    9.Write Patients Records to JSON file
    10.Display ID's using Threading
    11.Exit

ENTER YOUR CHOICE:4

Enter the ID that you wish to update:7
Name(Bhavana): Bhavz
Age(22): 32
Enter disease patient is diagnosed with(MU):Ulc
Updated Successfully
{'age': 32, 'disease': 'Ulc', 'id': 7, 'name': 'Bhavz'}
<-----Healthcare Patient Management System----->
```

## Deleting the Patient Record:

```
OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Enter disease patient is diagnosed with(MU):Ulc
Updated Successfully
{'age': 32, 'disease': 'Ulc', 'id': 7, 'name': 'Bhavz'}
<-----Healthcare Patient Management System----->

Choose one of the options below...
1.Display Patient Records
    2.Get Patient Record By ID,
    3.Add a patient record,
    4.Update an existing patient record,
    5.Delete an existing record,
    6.Search Patient record by the disease
    7.Mail the patient records to Business Owner
    8.Web Scraping
    9.Write Patients Records to JSON file
    10.Display ID's using Threading
    11.Exit

ENTER YOUR CHOICE:5

Enter ID you wish to delete:8
{'age': 22, 'disease': 'Migrane', 'id': 8, 'name': 'Charulatha'}
Are you sure you want to delete this patient record?yes
Deleted Successfully!
```

## Search Patient Record By the Disease

```
OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

1.Display Patient Records
    2.Get Patient Record By ID,
    3.Add a patient record,
    4.Update an existing patient record,
    5.Delete an existing record,
    6.Search Patient record by the disease
    7.Mail the patient records to Business Owner
    8.Web Scraping
    9.Write Patients Records to JSON file
    10.Display ID's using Threading
    11.Exit

ENTER YOUR CHOICE:6
Enter Name:Charulatha
Enter age:22
Enter disease that you are diagnosed with:Migrane
{'age': 22, 'disease': 'Migrane', 'id': 9, 'name': 'Charulatha'}
<-----Healthcare Patient Management System----->
```

**Mail the Patient Record to Business Owner:**

```
OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

<-----Healthcare Patient Management System----->

Choose one of the options below...
1.Display Patient Records
    2.Get Patient Record By ID,
    3.Add a patient record,
    4.Update an existing patient record,
    5.Delete an existing record,
    6.Search Patient record by the disease
    7.Mail the patient records to Business Owner
    8.Web Scraping
    9.Write Patients Records to JSON file
    10.Display ID's using Threading
    11.Exit

ENTER YOUR CHOICE:7
Email sent successfully!
```

**Web Scraping:**

```
OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

<-----Healthcare Patient Management System----->

Choose one of the options below...
1.Display Patient Records
    2.Get Patient Record By ID,
    3.Add a patient record,
    4.Update an existing patient record,
    5.Delete an existing record,
    6.Search Patient record by the disease
    7.Mail the patient records to Business Owner
    8.Web Scraping
    9.Write Patients Records to JSON file
    10.Display ID's using Threading
    11.Exit

ENTER YOUR CHOICE:8
INFORMATION GATHERED!!!
```

**Write Patients Records to JSON file:**



```
patient_app.py ...\backend M  patient_db.py  patient_app.py ...\frontend M  {} patients.json M ●
healthcare_patient_management_system > frontend > {} patients.json > {} 1
1 [{"age": 22, "disease": "MU", "id": 7, "name": "Bhavana"},
2 [{"age": 22, "disease": "Migrane", "id": 8, "name": "Charulatha"}]]
```

## Display ID's using Threading:

```
OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

ENTER YOUR CHOICE:9
JSON File created using Patients data!
<-----Healthcare Patient Management System----->

Choose one of the options below...
1.Display Patient Records
2.Get Patient Record By ID,
3.Add a patient record,
4.Update an existing patient record,
5.Delete an existing record,
6.Search Patient record by the disease
7.Mail the patient records to Business Owner
8.Web Scraping
9.Write Patients Records to JSON file
10.Display ID's using Threading
11.Exit

ENTER YOUR CHOICE:10

0@10908
1@10908
2@10908
0@4924
0@13744
1@4924
1@13744
2@13744
2@4924
0@22708
1@22708
2@22708
```

## Exit:

```
{ "age": 24, "disease": "MU", "id": 10, "name": "Archi" }
<-----Healthcare Patient Management System----->

Choose one of the options below...
1.Display Patient Records
2.Get Patient Record By ID,
3.Add a patient record,
4.Update an existing patient record,
5.Delete an existing record,
6.Search Patient record by the disease
7.Mail the patient records to Business Owner
8.Web Scraping
9.Write Patients Records to JSON file
10.Display ID's using Threading
11.Exit

ENTER YOUR CHOICE:11
Thanks for using the Application!
```