

**Fresher Academy**

**Hospital Management Application**

Requirement Specifications

Version: 1.0

***Hanoi, Feb 2023***

Contents

[1. Introduction 4](#_Toc127818341)

[1.1. Purpose 4](#_Toc127818342)

[1.2. Scope of Project 4](#_Toc127818343)

[1.3. Glossary 4](#_Toc127818344)

[2. High Level Requirements 5](#_Toc127818345)

[2.1. Use case diagram overview 5](#_Toc127818346)

[2.2. Entity Relationship Diagram 6](#_Toc127818347)

[3. Use Case Specification 6](#_Toc127818348)

[3.1. UC1: Config 6](#_Toc127818349)

[3.2. UC2: View List Hospital 7](#_Toc127818350)

[3.3. UC3: Search Hospital 7](#_Toc127818351)

[3.4. UC4: Edit Hospital 7](#_Toc127818352)

[3.5. UC5: Create Hospital 7](#_Toc127818353)

[3.6. UC6: Delete Hospital 7](#_Toc127818354)

[3.7. UC7: Import Hospital 7](#_Toc127818355)

[3.8. UC8: Export Hospital 8](#_Toc127818356)

[3.9. UC9: View List Doctor 8](#_Toc127818357)

[3.10. UC10: Search Doctor 8](#_Toc127818358)

[3.11. UC11: Edit Doctor 8](#_Toc127818359)

[3.12. UC12: Create Doctor 8](#_Toc127818360)

[3.13. UC13: Delete Doctor 9](#_Toc127818361)

[3.14. UC14: Import Doctor 9](#_Toc127818362)

[3.15. UC15: Export Doctor 9](#_Toc127818363)

[3.16. UC16: View List Patient 9](#_Toc127818364)

[3.17. UC17: Search Patient 9](#_Toc127818365)

[3.18. UC18: Edit Patient 10](#_Toc127818366)

[3.19. UC19: Create Patient 10](#_Toc127818367)

[3.20. UC20: Delete Patient 10](#_Toc127818368)

[3.21. UC21: Import Patient 10](#_Toc127818369)

[3.22. UC22: Export Patient 10](#_Toc127818370)

[3.23. UC23: View List Schedule 11](#_Toc127818371)

[3.24. UC24: Search Schedule 11](#_Toc127818372)

[3.25. UC25: Edit Schedule 11](#_Toc127818373)

[3.26. UC26: Create Schedule 11](#_Toc127818374)

[3.27. UC27: Delete Schedule 11](#_Toc127818375)

[3.28. UC28: Import Schedule 12](#_Toc127818376)

[3.29. UC29: Export Schedule 12](#_Toc127818377)

[4. Mockups Screen 13](#_Toc127818378)

[4.1. User 13](#_Toc127818379)

[4.1.1. Hospital Manage 13](#_Toc127818380)

[4.1.2. Doctor Manage 15](#_Toc127818381)

[4.1.3. Patient Manage 16](#_Toc127818382)

[4.1.4. Schedule Manage 18](#_Toc127818383)

[5. Non Functional Requirements 21](#_Toc127818384)

[5.1. Performance Requirements (optional) 21](#_Toc127818385)

[5.2. Safety Requirements (optional) 21](#_Toc127818386)

[5.3. Security Requirements (optional) 21](#_Toc127818387)

[6. Other Requirements 21](#_Toc127818388)

1. Introduction
2. Purpose

The purpose of the application is support user to manage hospitals, doctors, patients and schedule

1. Scope of Project

This software product will be the desktop application.

1. Glossary

|  |  |
| --- | --- |
| Term | Definition |
| User | Users manage hospital |

1. High Level Requirements
2. Use case diagram overview

|  |  |  |
| --- | --- | --- |
| **#** | **UC Name** | **Description** |
| 1 | Config | Allow user to adjust database connection configuration |
| 2 | View List Hospital | Allow user to view hospital list |
| 3 | Search Hospital | Allow user to search hospital by name |
| 4 | Edit Hospital | Allow user to edit hospital information |
| 5 | Create Hospital | Allow user to add a new hospital |
| 6 | Delete Hospital | Allow user to detele a hospital |
| 7 | Import Hospital | Allow user to import hospital list from excel file |
| 8 | Export Hospital | Allow user to export hospital list to excel file |
| 9 | View List Doctor | Allow user to view doctor list |
| 10 | Search Doctor | Allow user to search doctor by name |
| 11 | Edit Doctor | Allow user to edit doctor information |
| 12 | Create Doctor | Allow user to add a new doctor |
| 13 | Delete Doctor | Allow user to detele a doctor |
| 14 | Import Doctor | Allow user to import doctor list from excel file |
| 15 | Export Doctor | Allow user to export doctor list to excel file |
| 16 | View List Patient | Allow user to view Patient list |
| 17 | Search Patient | Allow user to search Patient by name |
| 18 | Edit Patient | Allow user to edit Patient information |
| 19 | Create Patient | Allow user to add a new Patient |
| 20 | Delete Patient | Allow user to detele a Patient |
| 21 | Import Patient | Allow user to import Patient list from excel file |
| 22 | Export Patient | Allow user to export Patient list to excel file |
| 23 | View List Schedule | Allow user to view Schedule list |
| 24 | Search Schedule | Allow user to search Schedule by name |
| 25 | Edit Schedule | Allow user to edit Schedule information |
| 26 | Create Schedule | Allow user to add a new Schedule |
| 27 | Delete Schedule | Allow user to detele a Schedule |
| 28 | Import Schedule | Allow user to import Schedule list from excel file |
| 29 | Export Schedule | Allow user to export Schedule list to excel file |

1. Entity Relationship Diagram

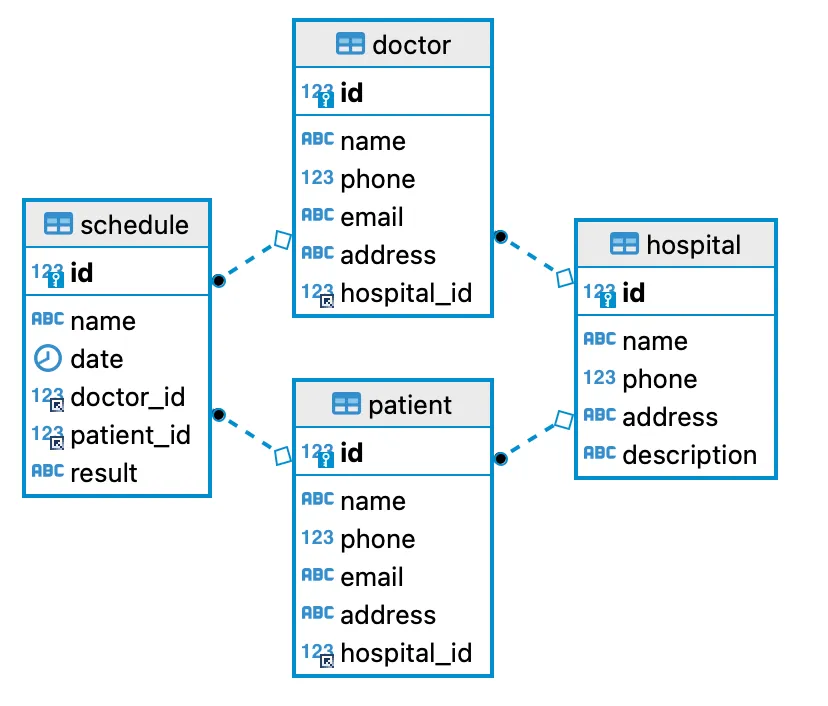


Figure 1: Entity Relationship Diagram

1. Use Case Specification

This section covers the system’s functional requirements which details what the system must do in terms of input, behavior and the expected output. It elicits the interaction between the actor(s) and the system, the system’s behavior and the results of their interactions.

1. UC1: Config

The application which read the configuration file named **database\_config.json** is located in Config folder to get the connection information to the database.

User can edit the aforemention configuration file by notepad.

The Config option is located on the application menu bar. When the user select Config -> file **database\_config.json** is opened by notepad and user can edit the information. After the file is saved, the application will use the new information to connect to the database.

1. UC2: View List Hospital

Hospital list is loaded from the Database and displayed on the table.

1. UC3: Search Hospital

Step1: User selects tab **search**

Step2: User inputs the name of the hospital into the search textbox.

Step3: User clicks button **Search Hospital**.

Step4: The hospitals which have the name similar to the input text are displayed on the table.

Step5: User clears textbox and clicks **Search Hospital**. List of all hospital is shown.

1. UC4: Edit Hospital

Step1: Every box on the table is editable except for the ID. User can edit hospital information.

Step2: User selects Save button to save the changes to the database.

1. UC5: Create Hospital

Step1: User selects tab **Add new**.

Step2: The application displays information form.

Step3: User inputs the information to the form.

Step4: User clicks button **Add** to save the changes to the database.

Step5: The newly created hospital is shown on the hospital list.

1. UC6: Delete Hospital

Step1: User clicks button **Detele** at the end of each record to delete a hospital information.

Step2: A confimation message is displayed in the form of QMessageBox(QMessageBox.question) with 2 button: Yes, Cancel.

Step3: If the user selects **Yes**, the selected record is deleted.

Step4: If the User clicks **Cancel**, the confirmation message is closed.

1. UC7: Import Hospital

Step1: User clicks **Import** button.

Step2: The file selection is opened. User selects a file, only allow excel file.

Step3: The System will import the hospital information in the file into the Database and display them to the table.

Step4: The system will bypass the invalid records and import the valid records into the database.

1. UC8: Export Hospital

Step1: User click **Export** button.

Step2: List of all hospital is saved to the excel file located in **Output** folder (same location with the Application).

Step3:The name of the exported file will follow the format **Hospital\_dd\_mm\_yyyy\_hh\_mm\_ss.xlsx** with **dd\_mm\_yyyy\_hh\_mm\_ss** according to the current time. For example, output file name will be **Hospital\_20\_02\_2023\_19\_35\_00.xlsx** if the current time is 20/02/2023 19:35:00.

1. UC9: View List Doctor

Step1: From the menubar of Hospital Manage screen, User selects **Programs -> Doctor**.

Step2: The **Doctor Manager** screen is displayed, doctor list is loaded from the database and shown on the table.

1. UC10: Search Doctor

Step1: User selects tab **search**

Step2: User inputs the name of the doctor into the search textbox.

Step3: User clicks button **Search Doctor**.

Step4: The doctors which have the name similar to the input text are displayed on the table.

Step5: User clears textbox and clicks **Search Doctor**. List of all doctor is shown.

1. UC11: Edit Doctor

Step1: Every box on the table is editable except for the ID. User can edit doctor information.

Step2: User selects Save button to save the changes to the database.

1. UC12: Create Doctor

Step1: User selects tab **Add new**.

Step2: The application displays information form.

Step3: User inputs the information to the form.

Step4: User clicks button **Add** to save the changes to the database.

Step5: The newly created doctor is shown on the doctor list.

1. UC13: Delete Doctor

Step1: User clicks button **Detele** at the end of each record to delete a doctor information.

Step2: A confimation message is displayed in the form of QMessageBox(QMessageBox.question) with 2 button: Yes, Cancel.

Step3: If the user selects **Yes**, the selected record is deleted.

Step4: If the User clicks **Cancel**, the confirmation message is closed.

1. UC14: Import Doctor

Step1: User clicks **Import** button.

Step2: The file selection is opened. User selects a file, only allow excel file.

Step3: The System will import the doctor information in the file into the Database and display them to the table.

Step4: The system will bypass the invalid records and import the valid records into the database.

1. UC15: Export Doctor

Step1: User click **Export** button.

Step2: List of all hospital is saved to the excel file located in **Output** folder (same location with the Application).

Step3:The name of the exported file will follow the format **Doctor\_dd\_mm\_yyyy\_hh\_mm\_ss.xlsx** with **dd\_mm\_yyyy\_hh\_mm\_ss** according to the current time. For example, output file name will be **Doctor\_20\_02\_2023\_19\_35\_00.xlsx** if the current time is 20/02/2023 19:35:00.

1. UC16: View List Patient

Step1: From the menubar of Hospital Manage screen, User selects **Programs -> Patient**.

Step2: The **Patient Manager** screen is displayed, doctor list is loaded from the database and shown on the table.

1. UC17: Search Patient

Step1: User selects tab **search**

Step2: User inputs the name of the patient into the search textbox.

Step3: User clicks button **Search Patient**.

Step4: The patients which have the name similar to the input text are displayed on the table.

Step5: User clears textbox and clicks **Search Patient**. List of all patient is shown.

1. UC18: Edit Patient

Step1: Every box on the table is editable except for the ID. User can edit patient information.

Step2: User selects **Save** button to save the changes to the database.

1. UC19: Create Patient

Step1: User selects tab **Add new**.

Step2: The application displays information form.

Step3: User inputs the information to the form.

Step4: User clicks button **Add** to save the changes to the database.

Step5: The newly created patient is shown on the patient list.

1. UC20: Delete Patient

Step1: User clicks button **Detele** at the end of each record to delete a patient information.

Step2: A confimation message is displayed in the form of QMessageBox(QMessageBox.question) with 2 button: Yes, Cancel.

Step3: If the user selects **Yes**, the selected record is deleted.

Step4: If the User clicks **Cancel**, the confirmation message is closed.

1. UC21: Import Patient

Step1: User clicks **Import** button.

Step2: The file selection is opened. User selects a file, only allow excel file.

Step3: The System will import the patient information in the file into the Database and display them to the table.

Step4: The system will bypass the invalid records and import the valid records into the database.

1. UC22: Export Patient

Step1: User click **Export** button.

Step2: List of all patient is saved to the excel file located in **Output** folder (same location with the Application).

Step3:The name of the exported file will follow the format **Patient\_dd\_mm\_yyyy\_hh\_mm\_ss.xlsx** with **dd\_mm\_yyyy\_hh\_mm\_ss** according to the current time. For example, output file name will be **Patient\_20\_02\_2023\_19\_35\_00.xlsx** if the current time is 20/02/2023 19:35:00.

1. UC23: View List Schedule

Step1: From the menubar of Hospital Manage screen, User selects **Programs -> Schedule**.

Step2: The **Schedule Manager** screen is displayed, doctor list is loaded from the database and shown on the table.

1. UC24: Search Schedule

Step1: User selects tab **search**

Step2: User inputs the name of the schedule or the time into the search textbox.

Step3: User clicks button **Search Schedule**.

Step4: The schedule which have the name similar to the input text and in the input time are displayed on the table.

Step5: User clears textbox and clicks **Search Schedule**. List of all schedule is shown.

1. UC25: Edit Schedule

Step1: Every box on the table is editable except for the ID. User can edit schedule information.

Step2: User selects **Save** button to save the changes to the database.

1. UC26: Create Schedule

Step1: User selects tab **Add new**.

Step2: The application displays information form.

Step3: User inputs the information to the form.

Step4: User clicks button **Add** to save the changes to the database.

Step5: The newly created schedule is shown on the schedule list.

1. UC27: Delete Schedule

Step1: User click button **Detele** ở cuối mỗi bản ghi để xoá thông tin lịch khám.

Step2: Một message confirm sẽ được hiển thị dưới dạng QMessageBox(QMessageBox.question) với 2 button confirm là Yes, Cancel

Step3: User click **Yes.** Bản ghi được chọn sẽ bị xoá.

Step4: User click **Cancel**. Thao tác xoá sẽ bị huỷ bỏ.

1. UC28: Import Schedule

Step1: User clicks **Import** button.

Step2: The file selection is opened. User selects a file, only allow excel file.

Step3: The System will import the schedule information in the file into the Database and display them to the table.

Step4: The system will bypass the invalid records and import the valid records into the database.

1. UC29: Export Schedule

Step1: User click **Export** button.

Step2: List of all patient is saved to the excel file located in **Output** folder (same location with the Application).

Step3:The name of the exported file will follow the format **Schedule \_dd\_mm\_yyyy\_hh\_mm\_ss.xlsx** with **dd\_mm\_yyyy\_hh\_mm\_ss** according to the current time. For example, output file name will be **Schedule \_20\_02\_2023\_19\_35\_00.xlsx** if the current time is 20/02/2023 19:35:00.

1. Mockups Screen
2. User
3. Hospital Manage

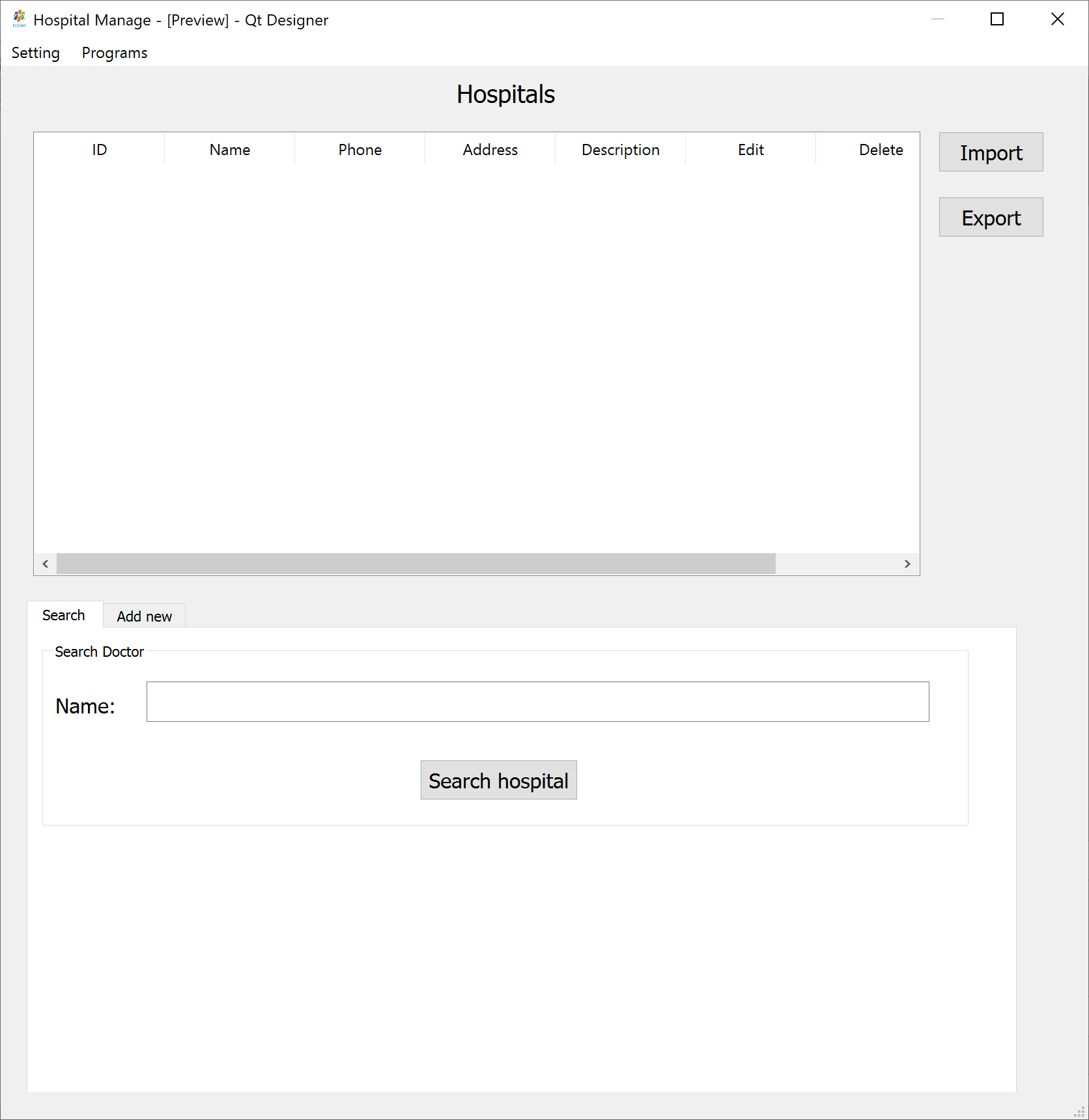


Figure 2: Hospital-1

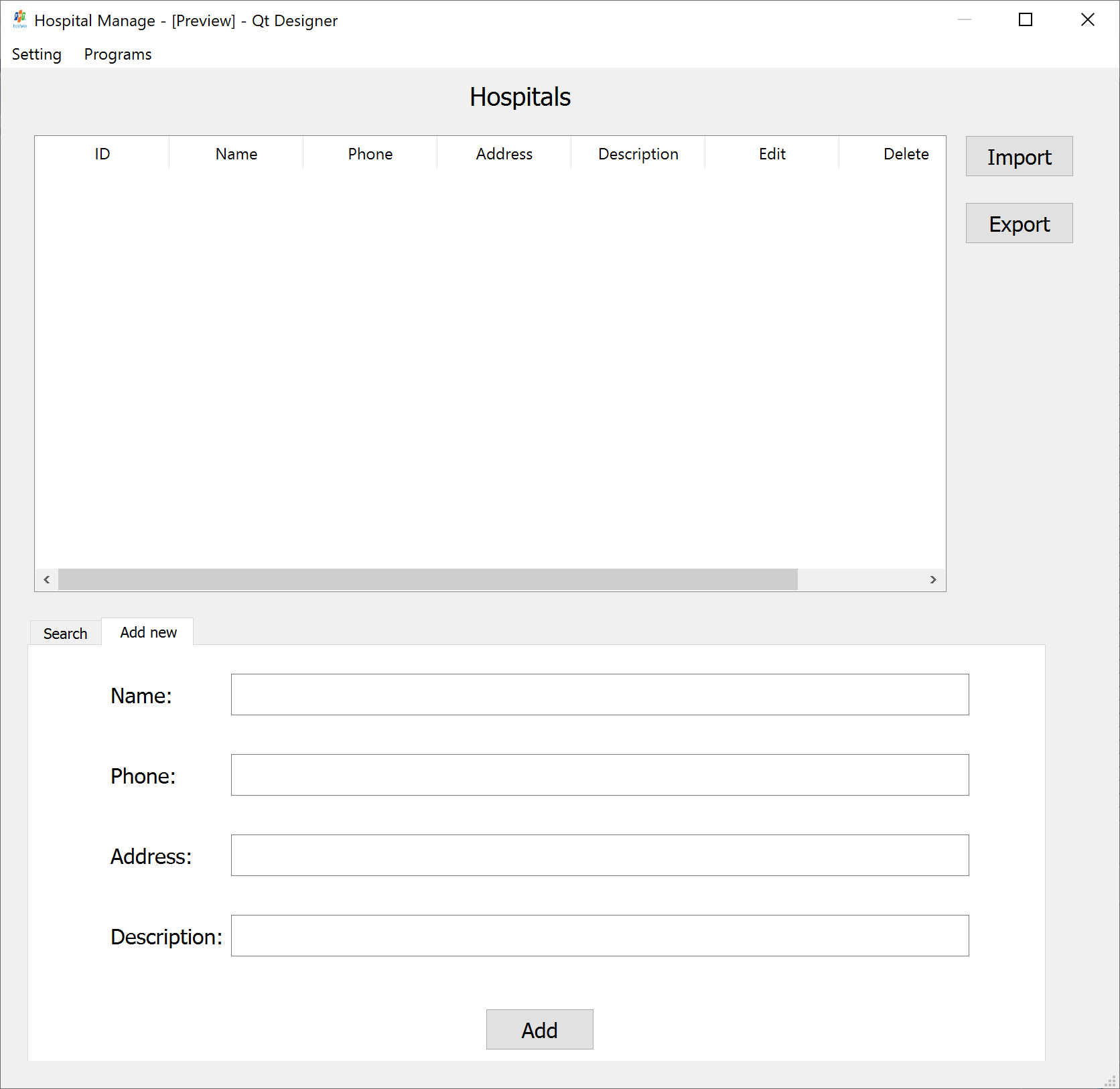


Figure 3: Hospital-2

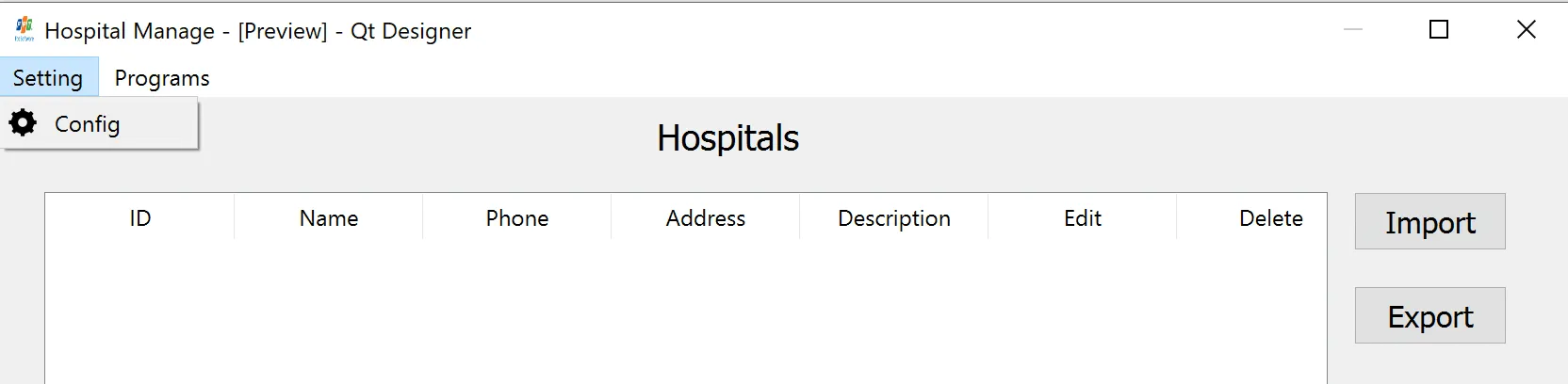


Figure 4: Menubar-1

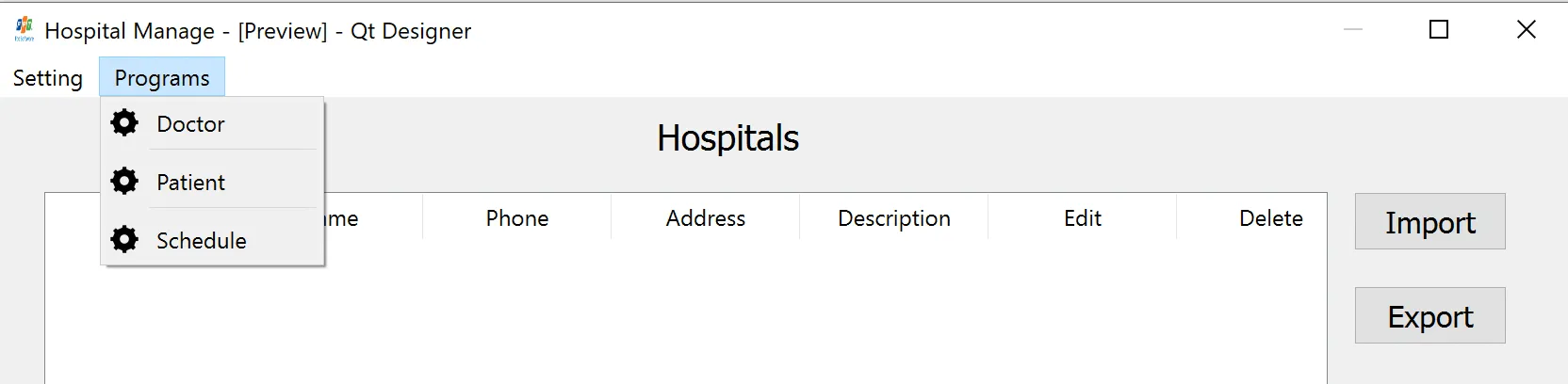


Figure 5: Menubar-2

1. Doctor Manage

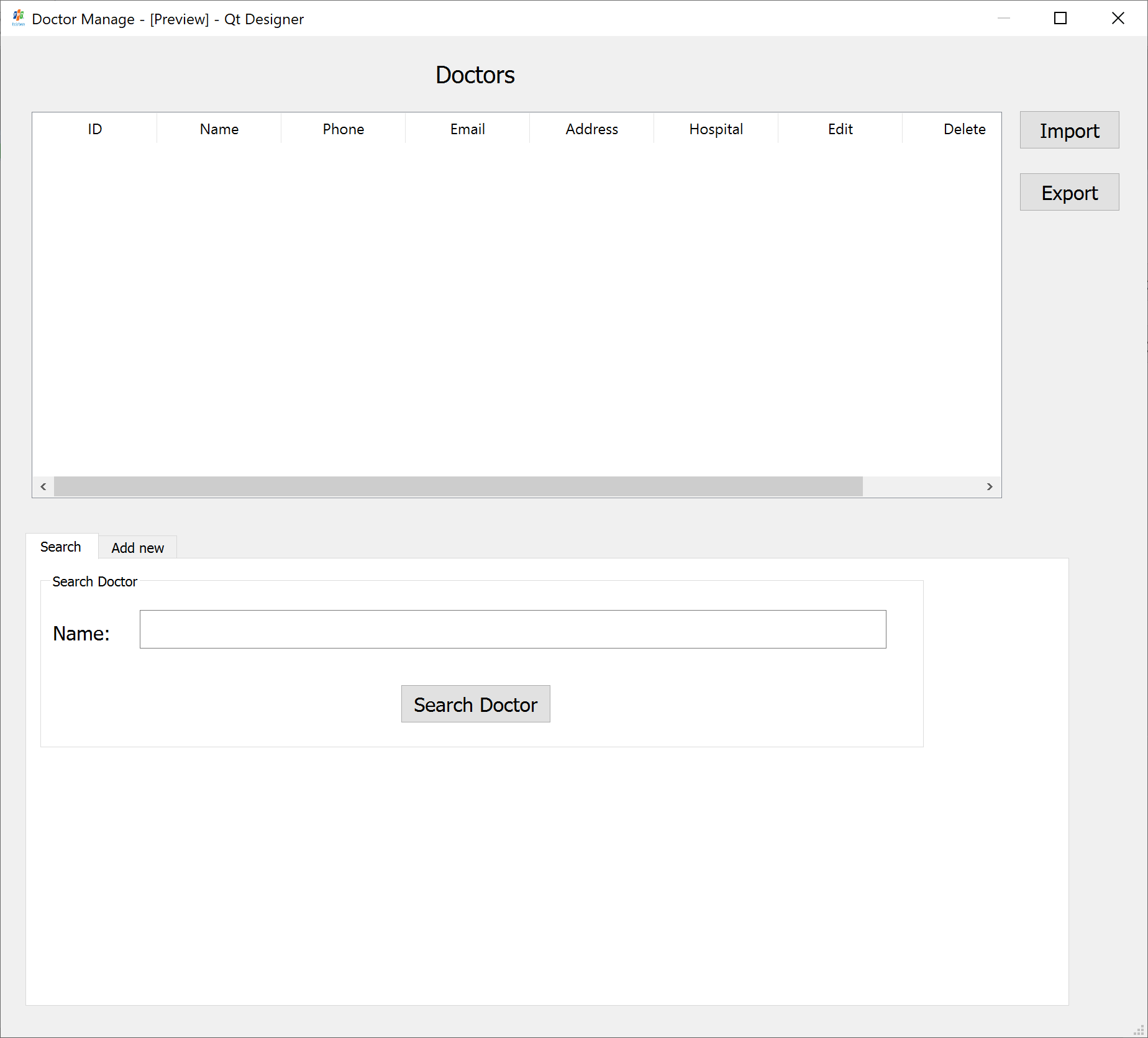


Figure 6: Doctor-1

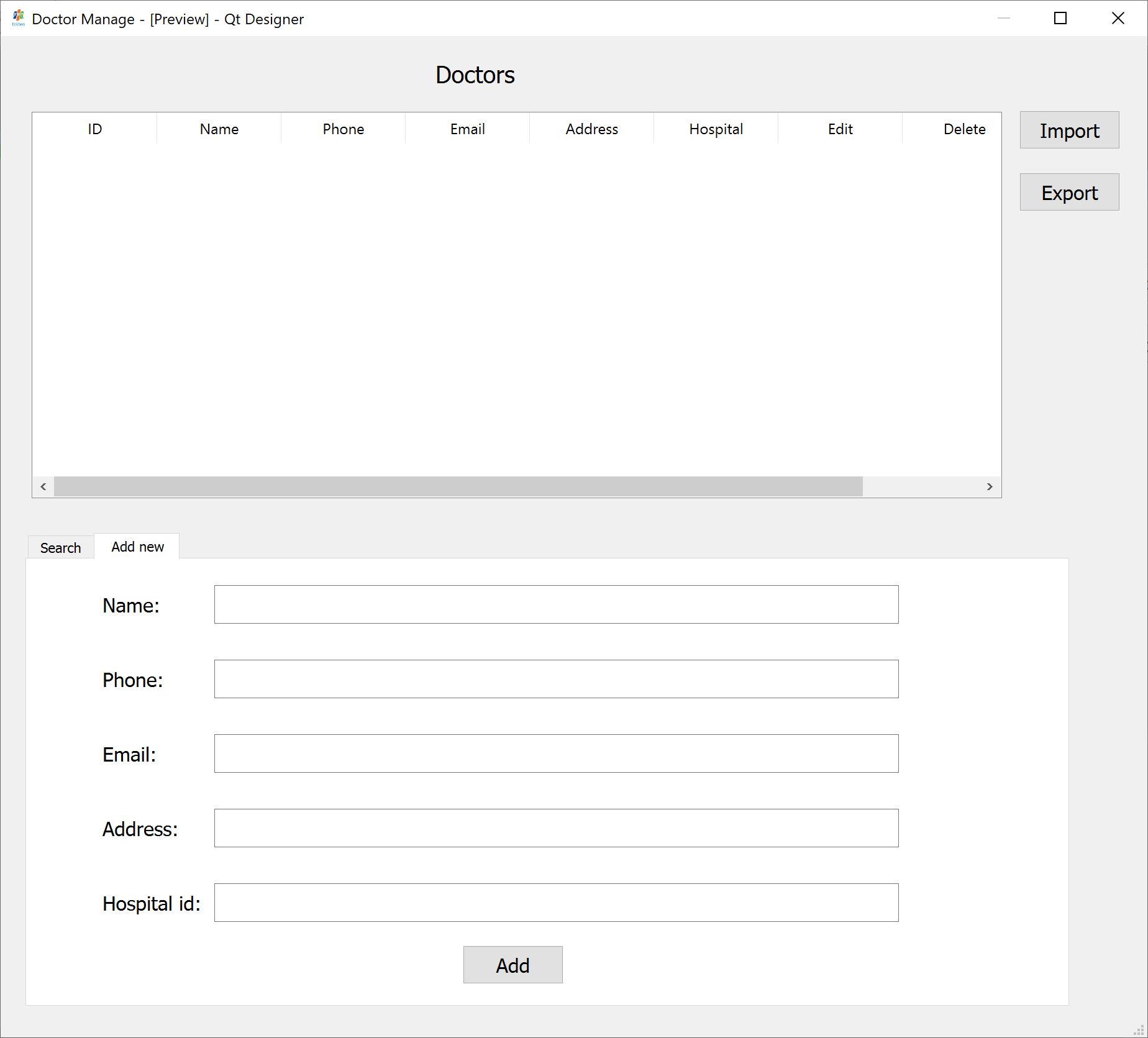


Figure 6: Doctor-2

1. Patient Manage

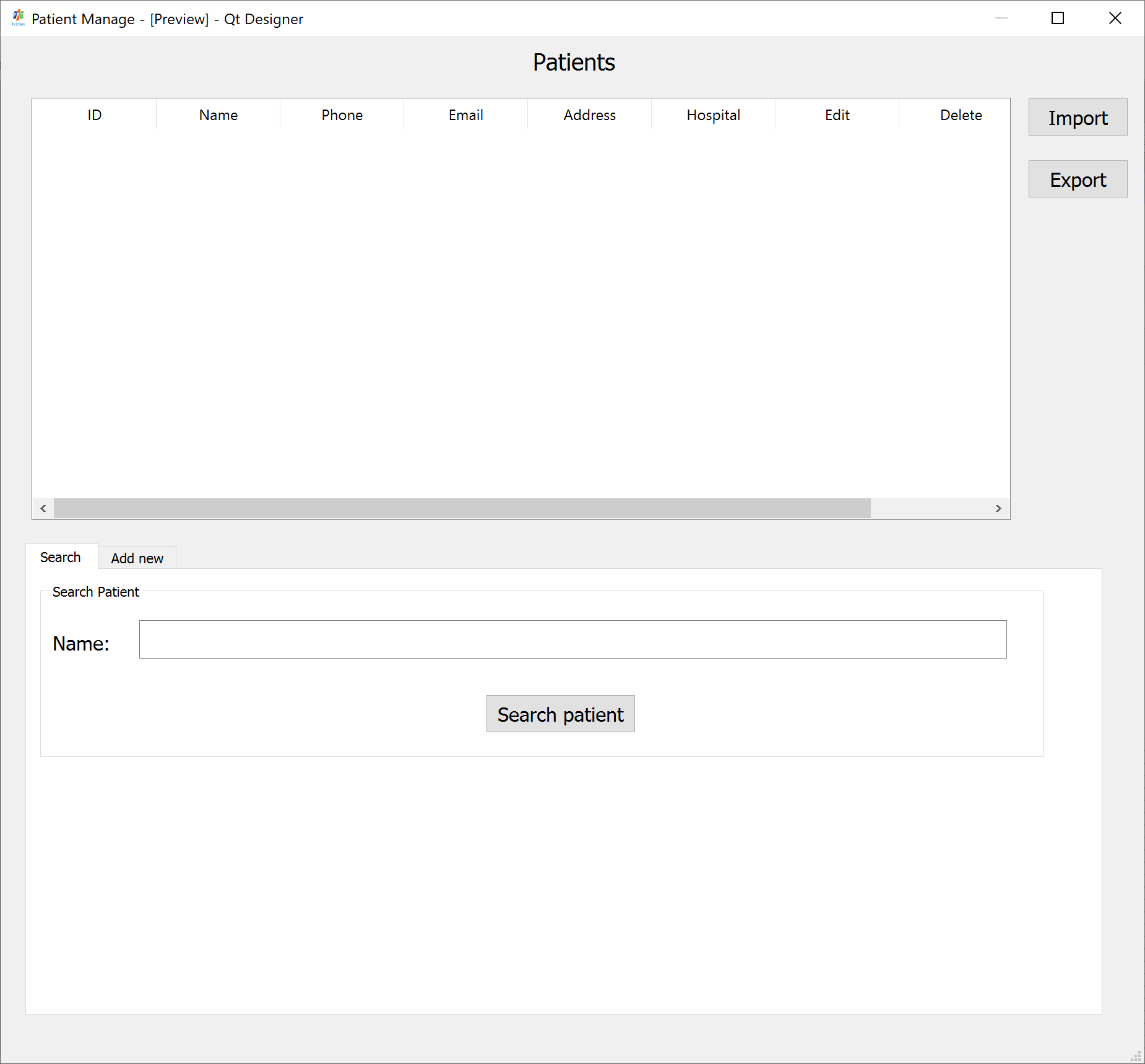


Figure 7: Patient-1

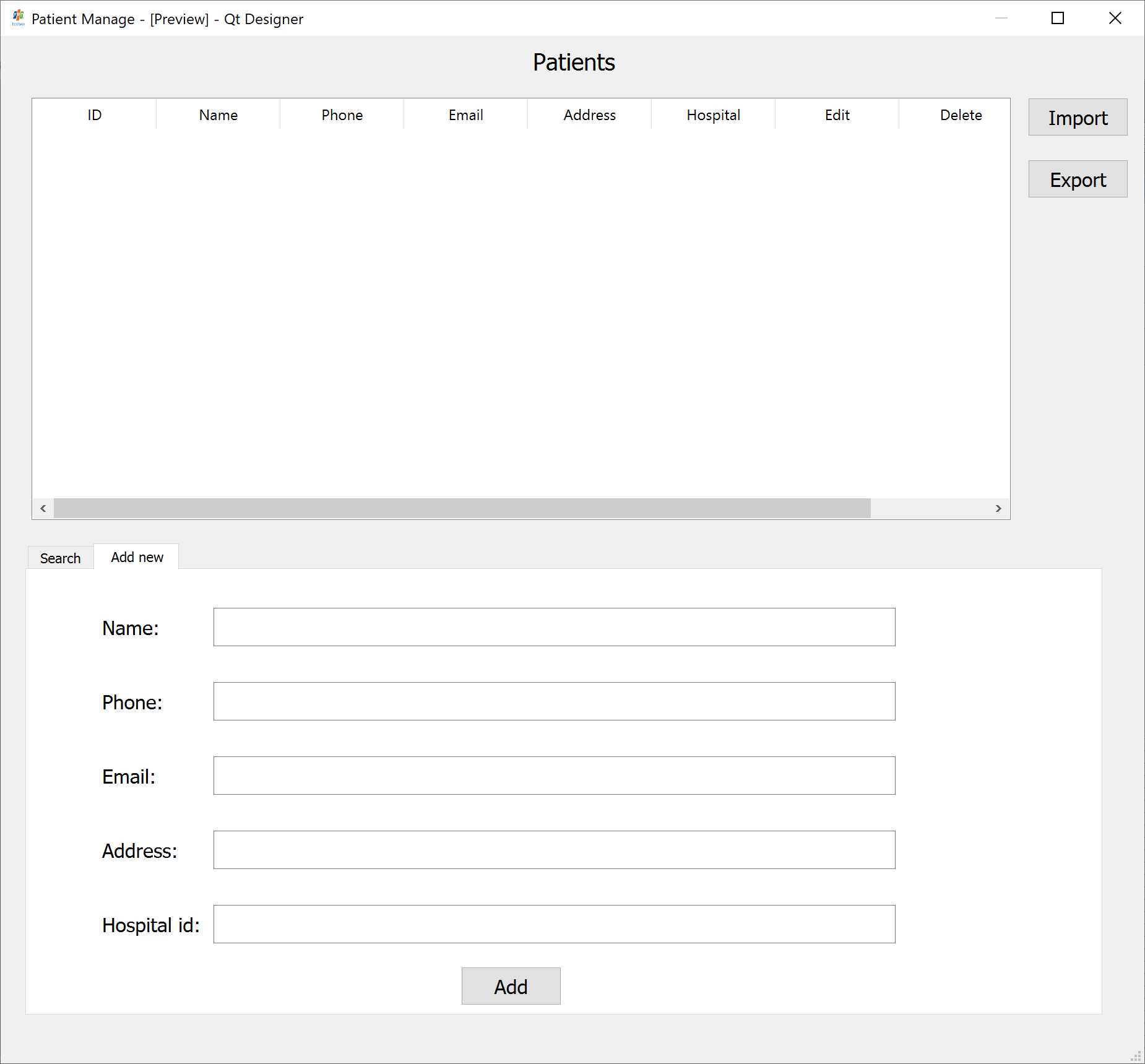


Figure 8: Patient-2

1. Schedule Manage

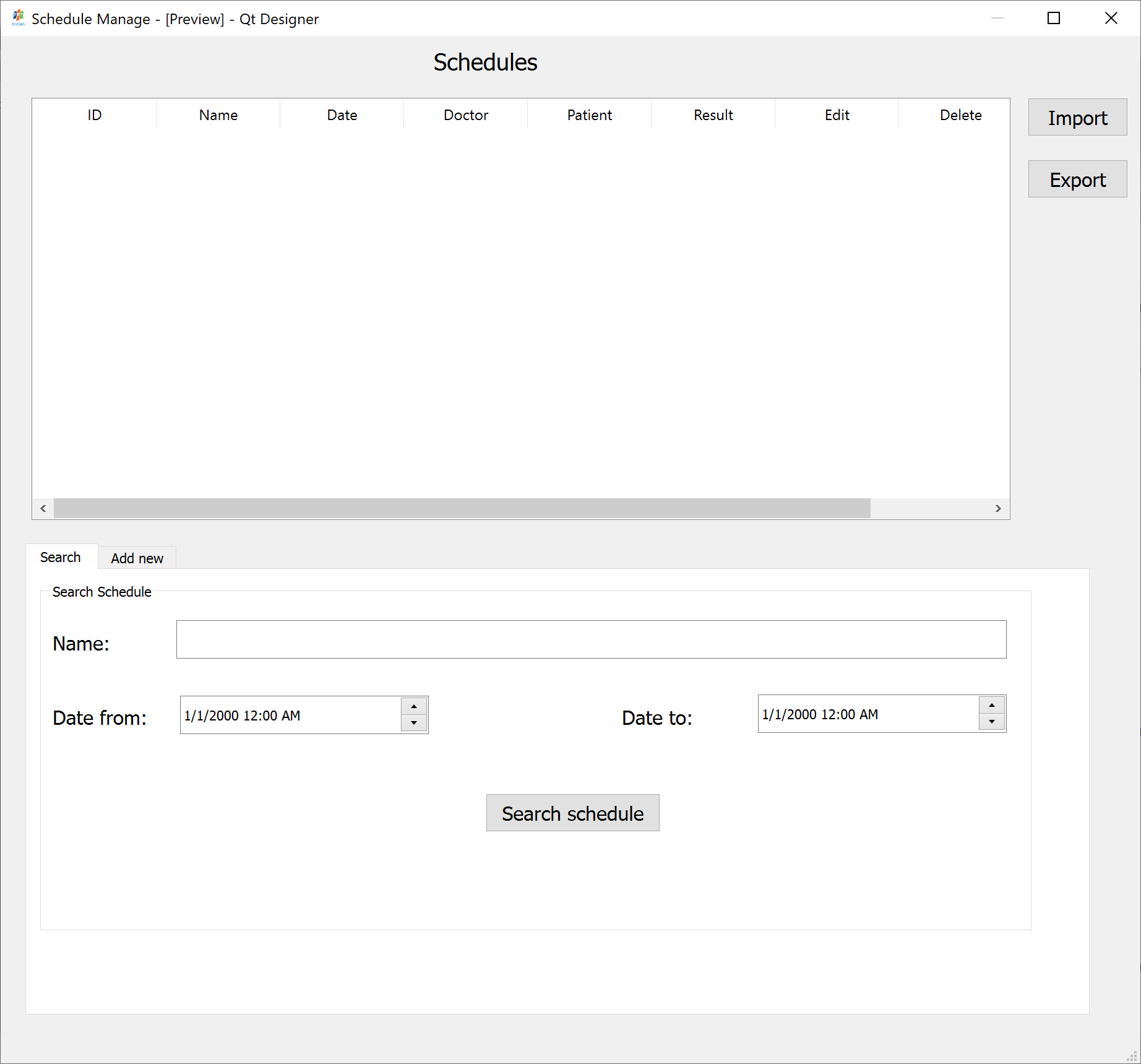


Figure 9: Schedule -1

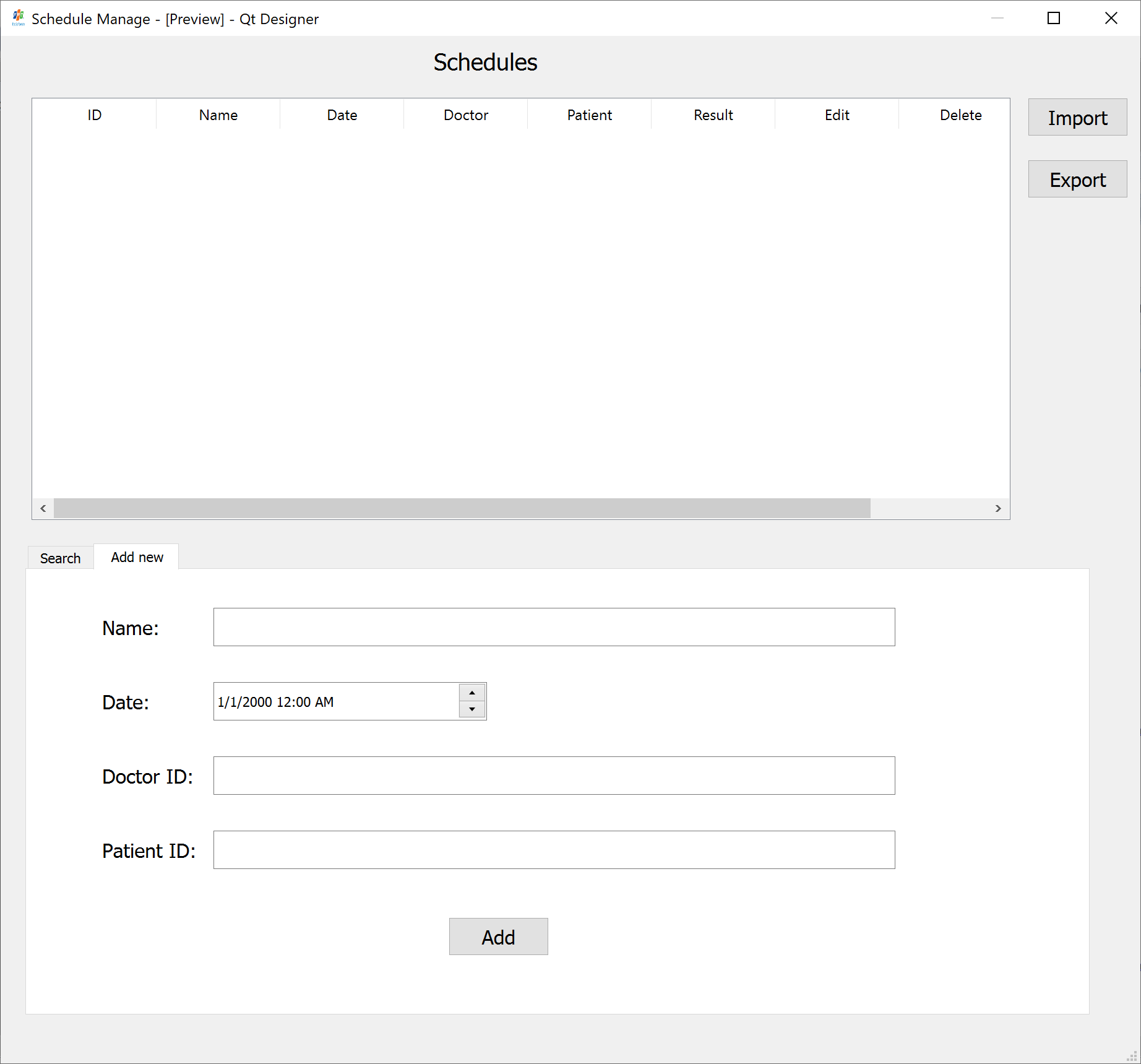


Figure 9: Schedule -2

1. Non Functional Requirements
2. Performance Requirements (optional)

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.>

1. Safety Requirements (optional)

<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied.>

1. Security Requirements (optional)

<Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. Refer to any external policies or regulations containing security issues that affect the product. Define any security or privacy certifications that must be satisfied.>

1. Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>