# Software Requirements

Version 2, last updated by Zoey at 2021-04-19

# Software Requirements Specification (SRS)

Revision History:

|  |  |  |
| --- | --- | --- |
| Date | Author | Description |
| 4/14/2021 | Paloma | Adding/Editing Use Cases |
| 4/16/2021 | Paloma | System Inputs and Outputs |
| 4/16/2021 | Paloma | Drawing the Use Case Chart |
| 4/17/2021 | Paloma | Detailed Output Behavior, Quality Requirements |
| 4/18/2021 | Paloma | Con Ops, System Context, System Capabilities, Expected Subsets. Updating the Use Case Chart. |
| 4/19/2021 | Zoey | Proofreading |

Contents

[Software Requirements 1](#_Toc32569)

[Software Requirements Specification (SRS) 1](#_Toc17593)

[1.  Introduction 4](#_Toc17157)

[1.1    Intended Audience and Purpose 4](#_Toc8362)

[1.2    How to use the document 4](#_Toc28493)

[2.  Concept of Operations 5](#_Toc14627)

[2.1    System Context 5](#_Toc13586)

[2.2 System capabilities 5](#_Toc1245)

[3.  Use Cases 6](#_Toc6889)

[Case 1: User Wants to Quit the Mini Program 6](#_Toc6125)

[Case 2: User Wants to Open the Mini Program 7](#_Toc13593)

[Case 3: User Wants to Register an Account and Log In 7](#_Toc27618)

[Case 4: Patient wants to consult related information before the visit 7](#_Toc31692)

[Case 5: Patient Wants to Upload an X-ray Film and Have an Early Detection 8](#_Toc20908)

[Case 6: Patient Wants to Consult Doctor Questions 8](#_Toc6113)

[Case 7: Patient Wants to Check the Consultation Record 9](#_Toc9141)

[Case 8: Patient Wants to Check the Upload Record 9](#_Toc787)

[Case 9: Patient Wants to Check the Line Chart of Disease 9](#_Toc16858)

[Case 10: Doctor Wants to Check the Consultation List 9](#_Toc32403)

[Case 11: Doctor Wants to Reply to the Patient's Message 10](#_Toc28203)

[Case 12: Administrator Wants to Operate on the User Information 10](#_Toc15182)

[Case 13: Administrator Wants to Operate on the Treatment Record 11](#_Toc6289)

[Case 14: Administrator Wants to Manage the Algorithm 11](#_Toc10606)

[Case 15: Administrator wants to backing up information or restore backup 12](#_Toc22741)

[3.    Behavioral Requirements 12](#_Toc20697)

[3.1 System Inputs and Outputs 12](#_Toc4500)

[3.2 Detailed Output Behavior 14](#_Toc25277)

[4.2   Quality Requirements 15](#_Toc19495)

[5.    Expected Subsets 16](#_Toc29802)

[6.    Fundamental Assumptions 16](#_Toc7311)

[The applet can be run on the latest version of WeChat. 16](#_Toc5638)

[Software updates are launched by developers. 16](#_Toc5593)

[7.    Expected Changes 16](#_Toc19430)

[8.    Appendices 16](#_Toc23200)

[8.1    Definitions and acronyms 16](#_Toc26501)

[8.2    References 17](#_Toc29326)

## 1.  Introduction

### 1.1    Intended Audience and Purpose

This document is intended to provided information guiding the installation and development process, ensuring that all system requirements are met. The following entities may find the document useful:  
Primary Customer - This page will detail all of the application requirements as understood by the production team. The customer should be able to determine that their requirements will be correctly reflected in the final product through the information found on this page.  
User - A prospective user will be able to use this document to identify the main functionality included in the application. Furthermore, the application will have a set of system requirements before the application can be run. Details regarding these requirements can be found here.  
Development Team - Details of specific requirements that the final software build must include will be located here. Developers can use this document to ensure the software addresses each of these requirements.  
QA Team - By developing testing procedures founded in the system requirements, the QA Team can create a comprehensive testing regimen that will guarantee requirements are met.

### 1.2    How to use the document

Table of Contents:  
  
1. Introduction  
2. Concept of Operations - broad description of the purpose of the application  
  2.1 System Context - details any specific system requirements the application will require to run  
  2.2 System Capabilities - description in prose of all capabilities available to the user in the address book  
  2.3 Use cases - A detailed look at each functional requirement, describing the application context both before and after an action is taken  
3. Behavioral Requirements - How the application will interact with a user  
  3.1 Input and output requirements - A description of allowed inputs and generated outputs  
    3.1.1 Input - Describes any restrictions that will be placed on allowed input  
    3.1.2 Output - Describes the range of outputs that can be generated  
  3.2 Detailed Output Behavior - Output descriptions in prose  
4. Quality Requirements - Requirements not pertaining to the function of the application will be listed here  
5. Expected Subsets - Expected levels of functionality at checkpoints during development  
6. Fundamental Assumptions - Some specifics about input, output, or behavior upon which other requirements are founded will be listed here  
7. Expected Changes - Future features and directions the project is expected to take  
8. Appendices - Details aiding the understanding of this document  
  8.1 Definitions and acronyms - Any technical terms or abbreviations will be spelled out here for ease of use of the document  
    8.1 Definitions - Definitions of technical or unusual terminology  
    8.1.2 Acronyms and Abbreviations - Any abbreviated terms will be expanded here  
  8.2 References - any external references necessary or helpful to understanding this document will be listed here

## 2.  Concept of Operations

The goal is to create a user-friendly WeChat applet to help patients with scoliosis to have self-examination. It will allow its users to upload X-rays for self-examination, communicate with doctors, and Help them to observe the changes of scoliosis better. Users must install the latest version of WeChat on their mobile phones. For more details on the usage and capabilities of the application, please read the section, [System Capabilities](https://uocis.assembla.com/spaces/cis422w18-team2/wiki/Software_Requirements" \l "System%20Capabilities).

### 2.1    System Context

**System Requirements:**

WeChat

* WeChat Version 7.0.0 and above.

IDE:

* WeChat Developer Tools Version1.0 and above

### 2.2 System capabilities

This application is a mini program based on WeChat, which can be used without downloading. After it launches, it will display the home page. If users want to operate, they will be prompted to log in. If users do not have an account, they should register an account.

For patients:

After logging in, the user can upload the X-ray film to get the detection result of the algorithm. Each detection result will be automatically recorded and generated a chart reflecting the changes of scoliosis. If there is any doubt about the condition, the user can first consult the guide, and then refer to the advice of the guide to choose the appropriate doctor to consult. For detection records, users can select the time to search. For consultation records, users can search by inputting the doctor's name or selecting the time.

For doctors:

After logging in, the user can view the consultation list, check the patient's situation and reply to the patient's questions. For consultation records, users can search by inputting the patient's name or selecting the time.

For administrators:

After logging in, users can view the information of all users and operate on the user information. The allowed operations include: adding a new account, deleting an existing account, editing the information of an existing account and searching for accounts. If users want to add new accounts, they will be asked to provide relevant information. As long as the user provides the necessary information, the account will be added to the existing account list. Users can edit any part of an account as long as their final edit includes the fields needed to create the account. If users delete an account, they will be prompted to confirm the deletion. If they do, the account is deleted.

The user's operation of the visit information is similar to that of the user information.

The user can change the algorithm used by the algorithm detection. If the user wants to change the algorithm, the system will prompt the user to confirm the change. If they do, the algorithm changes.

## 3.  Use Cases

### Case 1: User Wants to Quit the Mini Program

**Players:**End User.  
**Goals:**The end user wants to close the application.  
**Preconditions:**The application is open and running.  
**Case:**  
1.1 The end user selects the "Quit" option.  
1.2 The mini program closes all On-going mission.

1.3 The mini program terminates itself.  
**Alternate Flows:**

**Exception Flows:**

1.2.1 The conversation with the Doctor is not over yet.

The user is warned about quitting before ending the conversation.

      1.2.1.1 The user decides to hold the conversation.  
                  The mini program closes and the conversation reserved.  
      1.2.1.2 The user decides to end the conversation.

                  This conversation is over. The mini program terminates.  
1.2.2 The algorithm is processing the image after the user has uploaded it.

The user is warned about that processing is not yet complete.

1.2.2.1 The user decides to continue to wait for the result of processing.  
                  The mini program does not terminate and continues processing.  
      1.2.2.2 The user decides to continue with the termination.

                  Terminate the process. The mini program terminates.

1.2.3 The user is submitting appointment.

The user is warned about that if exit now, the information has been filled in will be lost.

1.2.3.1 The user decides to continue submitting the appointment.  
                  The mini program does not terminate and continues submitting.  
      1.2.3.2 The user decides to continue with the termination.

                  Abandon the submission. The mini program terminates.

**Postconditions: The mini program is closed.**

### Case 2: User Wants to Open the Mini Program

**Players:**End User.  
**Goals:**The end user would like to run the application so that they can consult their condition.

**Preconditions:**The application is loaded successfully.  
**Case:**  
2.1 The end user opens the mini program.  
2.2 The mini program opens with the home page.

**Alternate Flows:**

**Exception Flows:**  
**Postconditions:** The application is open and running, waiting for its next instruction from the end user.

### Case 3: User Wants to Register an Account and Log In

**Players:** End User.  
**Goal:** The user would like to create a new account.    
**Preconditions:** The application is open and running.  
**Case**:  
3.1 From the home page, the end user selects the "Me" option.

3.2 Jump to the personal page, the end user selects the " Log In" option.  
3.3 Jump to the login page, the end user selects the "If you don’t have an account, please sign up" option.

3.4 Choose one identity between “Patient” and “Doctor”.  
3.5 Fill in the information according to the prompts on the page.

3.6 Registered successfully, jump back to the login page and log in.  
**Alternate Flows:**

**Exception Flows:**

**3.5.1 The format of the user information inputted is incorrect.**

**3.5.2 The mobile phone number/email has been registered by others.**  
**Postconditions:**

**Account is registered successfully, jump to the login page.**

### Case 4: Patient wants to consult related information before the visit

**Players:** End User (Patient)  
**Goal:** The user would like to consult some information.  
**Preconditions:** The application is open and running. The user has logged in.

**Case**:

//chatbot

Relative information is still needed from the chatbot group. (Interfaces, format, etc.)

**Alternate Flows:**

**Exception Flows:**  
**Postconditions:**

### Case 5: Patient Wants to Upload an X-ray Film and Have an Early Detection

**Players:** End User (Patient).  
**Goal:** The user would like to have an early detection.   
**Preconditions:** The application is open and running. The user has logged in.  
**Case**:

5.1 From the home page, the end user selects the "Early Detection" option.

5.2 The end user selects the "Upload Picture" option.

5.3 The end user selects one picture from photo album and uploads it.

5.4 Wait for the algorithm to give the result.

5.5 Get the result and update to chart (line chart about changes of the results).

**Alternate Flows:**

**Exception Flows:**

5.4.1 The user decides to "Cancel" the process after deciding to have an early detection.

         The application returns to the previous interface

5.4.2 The image uploaded does not meet the requirements.

The user is prompted that “The photo does not meet the requirements, please try again”.

5.4.3 If the algorithm detects a timeout, please re-upload the film.

If the network is abnormal, please refresh the page.  
**Postconditions:**

**The algorithm gives the result.**

### Case 6: Patient Wants to Consult Doctor Questions

**Players:** End User (Patient)  
**Goal:** The user would like to make an appointment with doctor.    
**Preconditions:** The application is open and running. The user has logged in.  
**Case**:

6.1 From the home page, the end user selects the "Consult Doctors" option.

6.3 The user selects the doctor he or she wants.

6.4 The users leave messages to doctors describing their problems and can send pictures.  
**Alternate Flows:**

**Exception Flows:**

6.4.1 The user decides to "Cancel" the process after deciding to consult doctor questions.

          The application returns to the previous interface.

**Postconditions:**

A message was left for the doctor. The user can check it in the consultation record.

### Case 7: Patient Wants to Check the Consultation Record

**Players:** End User (Patient).   
**Goal:** The user would like to check the consultation record.  
**Preconditions:** The application is open and running. The user has logged in.  
**Case**:

7.1 From the personal page, the end user selects the "consultation record" option.

7.2 The mini program shows the consultation list in chronological order.

7.3 If the end user wants to query the consultation record with a doctor, just input the doctor's name in the search box above and confirm.

**Alternate Flows:**

**Exception Flows:**

**Postconditions:**

The consultation record is visible to the user.

### Case 8: Patient Wants to Check the Upload Record

**Players:** End User (Patient).  
**Goal:** The user would like to check the upload record.  
**Preconditions:** The application is open and running. The user has logged in.  
**Case**:

8.1 From the personal page, the end user selects the "Upload Record" option.

8.2 The mini program shows the upload list in chronological order.

**Alternate Flows:**

**Exception Flows:**  
**Postconditions:**

The upload record is visible to the user.

### Case 9: Patient Wants to Check the Line Chart of Disease

**Players:** End User (Patient).  
**Goal:** The user would like to check the line chart of disease  
**Preconditions:** The application is open and running. The user has logged in.  
**Case**:

9.1 From the personal page, the end user selects the "Line Chart of Disease" option.

9.2 The user selects the node on the chart to see the details.

**Alternate Flows:**

**Exception Flows:**  
**Postconditions:**

The line chart is visible to the user.

### Case 10: Doctor Wants to Check the Consultation List

**Players:** End User (Doctor).  
**Goal:** The user would like to check the consultation list.  
**Preconditions:** The application is open and running. The user has logged in.  
**Case**:

10.1 From the home page, the end user selects the "Consultation List" option.

10.2 The end user selects one consultation and can view part of the consultant's information.

10.3 The end user checks the message left by patient.

**Alternate Flows:**

**Exception Flows:**  
10.3.1 The user decides to "Cancel" the check process  
            The application returns to its initial state.  
**Postconditions:**

The consultation list is visible to the user.

### Case 11: Doctor Wants to Reply to the Patient's Message

**Players:** End User (Doctor).  
**Goal:** The user would like to get the patients informed.  
**Preconditions:** The application is open and running. The user has logged in.  
**Case**:

11.1 The end user can view some of the patient's information by selecting the patient's head portrait.

11.2 The end user replies to the unresponsive message according to the waiting list.

11.3 chatbot informs the patient? (available?)

//chatbot

**Alternate Flows:**

**Exception Flows:**  
**Postconditions:**

### Case 12: Administrator Wants to Operate on the User Information

**Players:** End User (Administrator).  
**Goal:** The user would like to operate on the user information.  
**Preconditions:** The application is open and running. The user has logged in.  
**Case**:

12.1 From the home interface, the end user selects the "User" option.

12.2 The mini program shows the information of users.

12.3 If the end user wants to register a new account, just selects “Add a new account”, and then fill in the information according to the prompts on the page.

12.4 If the end user wants to remove an account, just selects the account he/she wants to remove and selects “Delete this account”. After the pop-up window prompts "Are you sure you want to remove this account?", select "yes".

12.5 If the end user wants to find some information, he/she can determine the corresponding filter conditions and get the required information.

12.6 If the end user wants to modify some information about an account, just selects the account he/she wants to modify and selects “Modify”, then modifies the information and selects “Confirm”. After the pop-up window prompts "Are you sure you want to save this modification?", select "yes".

**Alternate Flows:**

**Exception Flows:**  
**Postconditions:** The user information list is visible to the user. User information is available for users to operate.

### Case 13: Administrator Wants to Operate on the Treatment Record

**Players:** End User (Administrator).  
**Goal:** The user would like to operate on the treatment record.  
**Preconditions:** The application is open and running. The user has logged in.  
**Case**:

13.1 From the home interface, the end user selects the "Treatment Record" option.

13.2 The mini program displays all treatment record.

13.3 If the end user wants to add a new record, just selects “add a new record”, and then fill in the information according to the prompts on the page.

13.4 If the end user wants to delete a record, just selects the record he/she wants to remove and selects “delete this account”. After the pop-up window prompts "Are you sure you want to delete this account?", select "yes".

13.5 If the end user wants to find some records, he/she can determine the corresponding filter conditions and get the required records.

13.6 If the end user wants to modify some information about a record, just selects the record he/she wants to modify and selects “Modify”, then modifies the information and selects “Confirm”. After the pop-up window prompts "Are you sure you want to save this modification?", select "yes".

**Alternate Flows:**

**Exception Flows:**  
**Postconditions:** The treatment record list is visible to the user. Treatment record is available for users to operate.

### Case 14: Administrator Wants to Manage the Algorithm

**Players:** End User (Administrator).  
**Goal:** The user would like to change the algorithm.  
**Preconditions:** The application is open and running. The user has logged in.  
**Case**:

14.1 From the home interface, the end user selects the "Algorithm" option.

14.2 The end user selects the algorithm he/she needs.

**Alternate Flows:**

**Exception Flows:**  
**Postconditions:**

**Available a**lgorithms are visible to the user and available for users to choose.

### Case 15: Administrator wants to backing up information or restore backup

**Players:** End User (Administrator).  
**Goal:** The user would like to backing up information or restore backup.  
**Preconditions:** The application is open and running. The user has logged in.  
**Case**:

15.1 From the home interface, the end user selects the "Backup" option.

15.2 The end user selects what kinds of information he/she wants to back up.

15.3 If the end user wants to retrieve the deleted information, he/she can restore the corresponding backup to achieve.

15.4 If the end user wants to delete a backup, just selects the backup he/she wants to remove and selects “delete this backup”. After the pop-up window prompts "Are you sure you want to delete this backup?", select "yes".

**Alternate Flows:**

**Exception Flows:**  
**Postconditions:**

The backup is visible to the user.

## 3.    Behavioral Requirements

### 3.1 System Inputs and Outputs

#### 3.1.1 Inputs

Inputs to the application come from the user.

   Inputs When Register an Account:

      \* Name: A contact can have a first name, or last name, or both. The first and last names will be separate fields. The system will accept any characters in the first and last name fields.

 \* Telephone Numbers: The only acceptable inputs will be numerical characters ranging 0-9.

      \* Email Address: Email addresses will be a single field. Acceptable inputs will be alphanumeric and must include '@' and '.' symbols. If the system does not detect '@' and '.', it is assumed that the email address is invalid.

\*The Type of Identity Documents: The user needs real-name authentication and must choose from one of the following types of identity documents: People's Republic of China Resident Identity Card, Mainland travel permit for Taiwan residents, Mainland Travel Permit for Hong Kong and Macao Residents, Passport Hong Kong Identity Card, Macao identity card, Residence permit for Taiwan residents of People's Republic of China, Residence permit for Hong Kong and Macao residents of People's Republic of China.

\*ID Number: The ID number must match the format of the identity document selected by the user.

\*Employee Number: (Only for Doctors) The employee number must match the format and be numerical characters ranging 0-9.

Uploading pictures during early detection:

\*General: Only x-ray images can be uploaded. Other images are not valid. Detailed requirements for the image will be displayed on the upload page. The picture must be（image recognition ability available for algorithm teams）

For the administrator, when managing user information, they can do this by interacting with buttons and other fields in the GUI: “Add a new account”, “Delete this account”, “Edit an entry”, “Modify”, “Sort” and “Search”. The same goes for managing treatment record.

#### 3.1.2 Outputs

The Patients:  
      Registering accounts:  
      \*If the system finds that the number inputted by the user has already been registered when registering for a new account, the system will remind the user that the account already exists. Users can either jump to the login interface and log in directly, or input a valid unregistered number.  
  
      Viewing Records:

  \*If our system flags a record's field as invalid, that field will be displayed in grey when viewing the contact. This grey record will persist.

      Searching:

\*If the record does not contain entries that meet the user's search requirements, the result list will be blank.

Uploading:

\*If the uploaded image does not meet the detection standard of the algorithm, the user will be informed that the uploaded image is not standardized.

  Sorting:  
      \*The list will be sorted in chronological order, and the results of user searching will also be sorted in chronological order.

The Doctors:

Viewing Consultation List:

\*If the message has not been viewed, the system will display a red dot at the end of the corresponding entry in the list. Click to view the message and the red dot will disappear.

The Administrator:

Adding:  
      \*If the system finds that the entry the user wants to add has already existed, the system will remind the user that the entry already exists.

The entry will be added to the record list if added successfully.

Deleting:

\*If users delete an entry, they will be prompted to confirm the deletion. If they do, the entry is deleted.

The entry will no longer be seen in the record list if deleted successfully

Modifying:

\*The user can edit any part of an entry as long as their final edit includes the fields needed to create the entry.

      Searching:

\*If the record list does not contain entries that meet the user's search requirements, the result list will be blank.

  Sorting:  
      \*The list will be sorted in chronological order, and the results of user searching will also be sorted in chronological order as well.

### 3.2 Detailed Output Behavior

For the patient：

From the home page, select "Disease Detection", and on the page it jumps to, "Upload Picture" will pop up a prompt: select "Camera" or "Album" to take pictures or choose pictures to upload. After the image is uploaded successfully and the algorithm gives the result, select the “View the trend of disease changes”to jump to the trend chart page, on which displays a line chart showing the changes in the condition with the time of the detection on the horizontal axis and the Angle on the vertical axis.

From the home page, select "Guidance", a dialogue page will pop up. The user can input text in the input box at the bottom of the page and select “send”. Select plus icon on the right side of the input box to select pictures to send.

From the home page, select "Consult Doctor", and the page it jumps to will display a list of doctors available for consultation. Select a doctor and it will display basic information about that doctor. Select "Ask" below, input the question in the page it jumps to, and confirm sending it.

From the home page or personal page, select "Detection Record" to view the historical detection record.

From the home page or personal page, select "Consultation Record" to view the historical consultation record.

For doctors:

From the home page, select "Consultation List" to view the consultation to be replied to.

For administrators:

From the home page, the user can view the information of all users by selecting "user information".

"Add Account" brings up a window for users to provide inputs like telephone, name, email, certificate number and more. Once the user provides the necessary information, the new account appears in the user information list.

"Delete account" removes an account from the displayed account list. It is not deleted from the list until the user saves the deletion.

"Edit Account" brings up a window that allows the user to change their previous account information. Changes are reflected immediately in the GUI.

"Searching" accounts by keyword or time is also supported.

From the home page, the user can view the treatment record of all users by selecting "Treatment Record".

"Add Record" brings up a window for users to provide inputs like name of the patient, name of the doctor, time and more. Once the user provides the necessary information, the new record appears in the treatment record list.

"Delete Record" removes a record from the displayed record list. It is not deleted from the list until the user saves the deletion.

"Edit Record" brings up a window that allows the user to change their previous record. Changes are reflected immediately in the GUI.

"Searching" records by keyword or time is also supported.

### 4.2   Quality Requirements

The mini program must be competitive with mini programs in regards to performance, reliability, consistency, and scalability.  
  
Performance: Responsiveness to user input.  
      \* Standard actions on the mini program should not exceed 500ms execution time.

      \* Searching on the user information list and medical record list by administrator is a possible exception to the above standard, as search performance will depend on the size of the list.  
Reliability: Confidence that actions taken will not result in errors, and that changes made to the user information list and medical record list are persistent.  
      \* User input should not produce faults or errors that impact or hinder use of the mini program.  
      \* Any modifications administrators made to the user information list and medical record list should produce a lasting change that persists through any following series of actions taken by the user.  
Consistency: Persistent data in user information and medical record.  
      \* User information and medical record should be modifiable after being added in such a way that target fields can be changed without affecting data in other fields.  
Scalability: Ease of extending application capabilities  
      \* Mini program should be modularized such that adding/extending features and functions only require changes to a single component and the interface with that component, if applicable.  
  
Furthermore, the application should adhere to U.S. Postal standards, validating input against the appropriate specifications. Input that deviates from the U.S postal standards will not be rejected, but the user will be notified that the input deviates from standards.

## 5.    Expected Subsets

L0:  
-Basic GUI with the ability to send data to and recall data from the backend storage  
-Ability to add address to address book  
-Ability to view an entry in an address book  
  
L1:  
-Address book capable of: storing complete entries, sorting, editing entries, and saving  
-Complete GUI for access to address book  
-Ability to import and export address books  
-Open recent

## 6.    Fundamental Assumptions

## The applet can be run on the latest version of WeChat.

## Software updates are launched by developers.

## 7.    Expected Changes

   Features to Add:  
      Online real-time interaction between patient and physician

Send Voice Message

      Search Contacts By Photo

      Voice-To-Text Searching

      Appointment operation of doctor-patient communication

## 8.    Appendices

### 8.1    Definitions and acronyms

#### 8.1.1    Definitions

|  |  |
| --- | --- |
| **Keyword** | **Definitions** |
|  |  |
|  |  |
|  |  |
|  |  |

#### 8.1.2    Acronyms and abbreviations

|  |  |
| --- | --- |
| **Acronym or**  **Abbreviation** | **Definitions** |
| GUI | Graphical User Interface |
|  |  |
|  |  |

### 8.2    References

Comments are disabled for this space. In order to enable comments, Messages tool must be added to project.

You can add Messages tool from Tools section on the Admin tab.