

# SOFTWARE DESIGN DESCRIPTION

*for*

## ION-CHAT SYSTEM

*Version 2.0*

*Prepared By*

**DeepBlue**

May 3<sup>rd</sup>, 2002

METU-ANKARA

## **Change History**

This document is the second version (version 2.0) of ION-CHAT SYSTEM Software Design Descriptions that was released on April 29<sup>th</sup>, 2002. This document is done according to the IEEE Std. 1016-1998, Recommended Practice for Software Design Descriptions [1]. The first version of this document was released on April 29<sup>th</sup>, 2002. The subsequent changes will this be mentioned in part of the SDD.

## **Preface**

The document contains the Software Design Description (SDD) of ION-CHAT SYSTEM. The mission of the project is to develop a web-based chat system for METU Informatics Institute Online IS Master Program.

ION-CHAT SYSTEM SDD consists of decomposition description, dependency description, interface description and the detailed design of the ION-CHAT SYSTEM.

The goal of the SDD is to describe the software design descriptions of the ION-CHAT SYSTEM. This document is prepared according to IEEE standard [1].

# Table of Contents

<b>CHANGE HISTORY .....</b>	<b>I</b>
<b>PREFACE .....</b>	<b>II</b>
<b>TABLE OF CONTENTS .....</b>	<b>III</b>
<b>LIST OF TABLES.....</b>	<b>VI</b>
<b>LIST OF FIGURES.....</b>	<b>VII</b>
<b>1 INTRODUCTION (SECTION 1 OF THE SDD).....</b>	<b>1</b>
1.1 PURPOSE.....	1
1.2 SCOPE.....	1
1.3 DEFINITIONS, ACRONYMS AND ABBREVIATIONS.....	2
<b>2 REFERENCES .....</b>	<b>4</b>
<b>3 DECOMPOSITION DESCRIPTION.....</b>	<b>5</b>
3.1 MODULE DECOMPOSITION .....	5
3.1.1 <i>Module 1 Chat Client</i> .....	5
3.1.1.1 Module 1.2 Client Site Root Functions.....	5
3.1.2 <i>Module 2 Chat Server</i> .....	6
3.1.2.1 Module 2.2 Server Site Root Functions .....	6
3.2 CONCURRENT PROCESS DECOMPOSITION.....	7
3.3 DATA DECOMPOSITION .....	7
3.3.1 <i>Data Entity 1 User Information File</i> .....	7
3.3.2 <i>Data Entity 2 Room Information File</i> .....	7
<b>4 DEPENDENCY DESCRIPTION.....</b>	<b>9</b>
4.1 INTERMODULE DEPENDENCIES .....	9
4.1.1 <i>Module 1 Chat Client</i> .....	9
4.1.1.1 Module 1.1 Client Site User Authentication .....	9
4.1.1.2 Module 1.2 Client Site Root Functions.....	9
4.1.1.2.1 Module 1.2.1 Client Site Root Console .....	9
4.1.1.2.2 Module 1.2.2 Client Site User Addition.....	10
4.1.1.2.3 Module 1.2.3 Client Site User Deletion .....	10
4.1.1.2.4 Module 1.2.4 Client Site User Update .....	11
4.1.1.2.5 Module 1.2.5 Client Site Room Addition .....	11
4.1.1.2.6 Module 1.2.6 Client Site Room Deletion.....	12
4.1.1.2.7 Module 1.2.7 Client Site Room Update .....	12
4.1.1.2.8 Module 1.2.8 Client Site Chat Group Addition .....	13
4.1.1.2.9 Module 1.2.9 Client Site Chat Group Deletion .....	13
4.1.1.2.10 Module 1.2.10 Client Site Chat Group Update .....	14
4.1.1.3 Module 1.3 Client Site Chat .....	14
4.1.2 <i>Module 2 Chat Server</i> .....	15
4.1.2.1 Module 2.1 Server Site User Authentication .....	15
4.1.2.2 Module 2.2 Server Site Root Functions .....	15
4.1.2.2.1 Module 2.2.1 Server Site Root Console.....	15
4.1.2.2.2 Module 2.2.2 Server Site User Addition .....	16
4.1.2.2.3 Module 2.2.3 Server Site User Deletion .....	16
4.1.2.2.4 Module 2.2.4 Server Site User Update.....	17

4.1.2.2.5	Module 2.2.5 Server Site Room Addition.....	17
4.1.2.2.6	Module 2.2.6 Server Site Room Deletion .....	17
4.1.2.2.7	Module 2.2.7 Server Site Room Update .....	18
4.1.2.2.8	Module 2.2.8 Server Site Chat Group Addition.....	18
4.1.2.2.9	Module 2.2.9 Server Site Chat Group Deletion .....	19
4.1.2.2.10	Module 2.2.10 Server Site Chat Group Update.....	19
4.1.2.3	Module 2.3 Server Site Chat.....	19
4.2	INTERPROCESS DEPENDENCIES.....	20
4.3	DATA DEPENDENCIES.....	20
4.3.1	<i>Data Entity 1 User Information File.....</i>	20
4.3.2	<i>Data Entity 2 Room Information File.....</i>	20
<b>5</b>	<b>INTERFACE DESCRIPTION.....</b>	<b>21</b>
5.1	MODULE INTERFACE.....	21
5.1.1	<i>Module 1 Chat Client.....</i>	21
5.1.1.1	Module 1.1 Client Site User Authentication.....	21
5.1.1.2	Module 1.2 Client Site Root Functions.....	23
5.1.1.2.1	Module 1.2.1 Client Site Root Console .....	23
5.1.1.2.2	Module 1.2.2 Client Site User Addition.....	25
5.1.1.2.3	Module 1.2.3 Client Site User Deletion .....	26
5.1.1.2.4	Module 1.2.4 Client Site User Update .....	27
5.1.1.2.5	Module 1.2.5 Client Site Room Addition .....	28
5.1.1.2.6	Module 1.2.6 Client Site Room Deletion.....	29
5.1.1.2.7	Module 1.2.7 Client Site Room Update .....	30
5.1.1.2.8	Module 1.2.8 Client Site Chat Group Addition .....	31
5.1.1.2.9	Module 1.2.9 Client Site Chat Group Deletion .....	32
5.1.1.2.10	Module 1.2.10 Client Site Chat Group Update .....	33
5.1.1.3	Module 1.3 Client Site Chat .....	34
5.1.2	<i>Module 2 Chat Server.....</i>	37
5.1.2.1	Module 2.1 Server Site User Authentication .....	37
5.1.2.2	Module 2.2 Server Site Root Functions .....	38
5.1.2.2.1	Module 2.2.1 Server Site Root Console.....	39
5.1.2.2.2	Module 2.2.2 Server Site User Addition .....	41
5.1.2.2.3	Module 2.2.3 Server Site User Deletion .....	42
5.1.2.2.4	Module 2.2.4 Server Site User Update.....	42
5.1.2.2.5	Module 2.2.5 Server Site Room Addition.....	43
5.1.2.2.6	Module 2.2.6 Server Site Room Deletion .....	44
5.1.2.2.7	Module 2.2.7 Server Site Room Update .....	45
5.1.2.2.8	Module 2.2.8 Server Site Chat Group Addition .....	46
5.1.2.2.9	Module 2.2.9 Server Site Chat Group Deletion .....	46
5.1.2.2.10	Module 2.2.10 Server Site Chat Group Update.....	47
5.1.2.3	Module 2.3 Server Site Chat.....	48
<b>6</b>	<b>DETAILED DESIGN.....</b>	<b>51</b>
6.1	MODULE DETAILED DESIGN.....	51
6.1.1	<i>Module 1 Chat Client.....</i>	51
6.1.1.1	Module 1.1 Client Site User Authentication.....	51
6.1.1.2	Module 1.2 Client Site Root Functions.....	51
6.1.1.2.1	Module 1.2.1 Client Site Root Console .....	51
6.1.1.2.2	Module 1.2.2 Client Site User Addition.....	51
6.1.1.2.3	Module 1.2.3 Client Site User Deletion .....	51
6.1.1.2.4	Module 1.2.4 Client Site User Update .....	52
6.1.1.2.5	Module 1.2.5 Client Site Room Addition .....	52
6.1.1.2.6	Module 1.2.6 Client Site Room Deletion.....	52
6.1.1.2.7	Module 1.2.7 Client Site Room Update .....	52
6.1.1.2.8	Module 1.2.8 Client Site Chat Group Addition .....	52

6.1.1.2.9	Module 1.2.9 Client Site Chat Group Deletion .....	52
6.1.1.2.10	Module 1.2.10 Client Site Chat Group Update .....	53
6.1.1.3	Module 1.3 Client Site Chat .....	53
6.1.2	<i>Module 2 Chat Server</i> .....	53
6.1.2.1	Module 2.1 Server Site User Authentication .....	53
6.1.2.2	Module 2.2 Server Site Root Functions .....	53
6.1.2.2.1	Module 2.2.1 Server Site Root Console.....	53
6.1.2.2.2	Module 2.2.2 Server Site User Addition .....	53
6.1.2.2.3	Module 2.2.3 Server Site User Deletion .....	54
6.1.2.2.4	Module 2.2.4 Server Site User Update.....	54
6.1.2.2.5	Module 2.2.5 Server Site Room Addition.....	54
6.1.2.2.6	Module 2.2.6 Server Site Room Deletion .....	54
6.1.2.2.7	Module 2.2.7 Server Site Room Update .....	54
6.1.2.2.8	Module 2.2.8 Server Site Chat Group Addition .....	54
6.1.2.2.9	Module 2.2.9 Server Site Chat Group Deletion .....	55
6.1.2.2.10	Module 2.2.10 Server Site Chat Group Update.....	55
6.1.2.3	Module 2.3 Server Site Chat.....	55
6.2	DATA DETAILED DESIGN.....	55
6.2.1	<i>Data Entity 1 User Information File</i> .....	55
6.2.2	<i>Data Entity 2 Room Information File</i> .....	55
<b>APPENDIX-A .....</b>		<b>56</b>
<b>APPENDIX-B .....</b>		<b>60</b>
<b>APPENDIX-C .....</b>		<b>86</b>
<b>APPENDIX-D .....</b>		<b>87</b>
<b>APPENDIX-E .....</b>		<b>88</b>
<b>APPENDIX-F.....</b>		<b>98</b>
<b>APPENDIX-G .....</b>		<b>125</b>
<b>APPENDIX-H.....</b>		<b>128</b>
<b>APPDENIX-I .....</b>		<b>129</b>

## List of Tables

Table 1: Description of common data elements .....	87
Table 2: Code Table .....	128
Table 3: Method Table .....	129

## List of Figures

Figure 1: Module 1.1 Client_Site_User_Authentication .....	60
Figure 2: Module 1.2.1 Client_Site_Root_Console.....	61
Figure 3: Module 1.2.2 Client_Site_User_Addition.....	62
Figure 4: Module 1.2.3 Client_Site_User_Deletion .....	63
Figure 5: Module 1.2.4 Client_Site_User_Update .....	64
Figure 6: Module 1.2.5 Client_Site_Room_Addition.....	65
Figure 7: Module 1.2.6 Client_Site_Room_Deletion .....	66
Figure 8: Module 1.2.7 Client_Site_Room_Update .....	67
Figure 9: Module 1.2.8 Client_Site_Chat_Group_Addition.....	68
Figure 10: Module 1.2.9 Client_Site_Chat_Group_Deletion .....	69
Figure 11: Module 1.2.10 Client_Site_Chat_Group_Update .....	70
Figure 12: Module 1.3 Client_Site_Chat .....	71
Figure 13: Module 2.1 Server_Site_User_Authentication.....	72
Figure 14: 14: Module 2.2.1 Server_Site_Root_Console .....	73
Figure 15: Module 2.2.2 Server_Site_User_Addition .....	74
Figure 16: Module 2.2.3 Server_Site_User_Deletion.....	75
Figure 17: Module 2.2.4 Server_Site_User_Update.....	76
Figure 18: Module 2.2.5 Server_Site_Room_Addition .....	77
Figure 19: Module 2.2.6 Server_Site_Room_Deletion .....	78
Figure 20: Module 2.2.7 Server_Site_Room_Update.....	79
Figure 21: Module 2.2.8 Server_Site_Chat_Group_Addition .....	80
Figure 22: Module 2.2.9 Server_Site_Chat_Group_Deletion .....	81
Figure 23: Module 2.2.10 Server_Site_Chat_Group_Update.....	82
Figure 24: Module 2.3 Server_Site_Chat .....	83
Figure 25: User Information File.....	84
Figure 26: Room Information File.....	85
Figure 27:Flow Chart of Authentication for Client software.....	98
Figure 28: Flow Chart of User Addition for Client software.....	99
Figure 29: Flow Chart of User Deletion for Client software .....	100
Figure 30: Flow Chart of User Update for Client software .....	101
Figure 31: Flow Chart of Room Addition for Client Software.....	102
Figure 32: Flow Chart of Room Deletion for Client software .....	103
Figure 33: Flow Chart of Room Update for Client software .....	104
Figure 34: Flow Chart of Chat Group Addition for Client Software.....	105
Figure 35:Flow Chart of Chat Group Deletion for Client software .....	106
Figure 36:Flow Chart of Chat Group Update for Client software .....	107
Figure 37:Flow Chart of Chat for Client software.....	109
Figure 38:Flow Chart of Authentication for Server software .....	110
Figure 39:Flow Chart of User Addition for Server software .....	111
Figure 40:Flow Chart of User Deletion for Server software .....	112
Figure 41:Flow Chart of User Update for Server software.....	113
Figure 42:Flow Chart of Room Addition for Server software.....	114
Figure 43:Flow Chart of Room Deletion for Server software .....	115
Figure 44:Flow Chart of Room Update for Server software .....	116
Figure 45:Flow Chart of Chat Group Addition for Server software .....	117
Figure 46:Flow Chart of Chat Group Deletion for Server software .....	118
Figure 47:Flow Chart of Chat Group Update for Server software .....	119
Figure 48:Flow Chart of Chat for Server software .....	121
Figure 49: Flow Chart of Root Console for Client software .....	123
Figure 50:Flow Chart of Root Console for Server software .....	124



Figure 51: Structure Chart of Chat Client.....	125
Figure 52: Structure Chart of Chat Server .....	126
Figure 53:Structure Chart of Information Files .....	127

# **1 INTRODUCTION (Section 1 of the SDD)**

## **1.1 Purpose**

This document provides a representation and model of ION-CHAT SYSTEM within the limitations indicated in the SRS document of ION-CHAT SYSTEM Version 1.2 [2]

SDD document of ION-CHAT SYSTEM has an important role in the development and maintenance of the ION-CHAT SYSTEM. Therefore, all the design information needed by project managers, quality assurance staff, configuration managers, software designers, programmers, tester, and maintainers are contained in the SDD document of the ION-CHAT SYSTEM.

## **1.2 Scope**

The SDD document of the ION-CHAT SYSTEM will define the product functions, user characteristics, constraints, and specific requirements of the system. Moreover, it also prepares a basis for the coding. In other words, the SDD document of the ION-CHAT SYSTEM scope is to design all the design information needed.

The ION-CHAT SYSTEM is composed of two modules; Client Chat and Server Chat.

The SDD document of the ION-CHAT SYSTEM contains a structural overview of all modules, interfaces, and data. It also contains a detailed design of each module. The SDD document of the ION-CHAT SYSTEM provides customers and instructor detailed descriptions of the software processing definitions. This volume provides the overall software architecture and describes the design methods used for each component of the software.

### 1.3 Definitions, Acronyms and Abbreviations

**Acquirer:** The customer, which specifies the requirements for the product and gets the final product and its documents.

**Basic Operation:** Add, delete and update operations for users, rooms and chat groups.

**CCmsg:** Client site communication message

**Chat group:** A group of users working on the same project either as a project developing team or as a reviewer team.

**Chat Function:** Function of making chat.

**Client Site:** The remote machine where the client software is to be executed.

**Client Software:** The part of the ION-CHAT SYSTEM that will be executed at the client site.

**DeepBlue:** Software development team of ION-CHAT SYSTEM.

**Emsg:** Error message

**IEEE:** Institute of Electrics & Electronics Engineering

**ION system:** Existing informatics online system.

**ION:** Informatics Online.

**ION-CHAT SYSTEM:** Informatics online chat system.

**IS:** Information System

**METU-II:** Middle East Technical University Informatics Institute.

**Reviewer Team:** Subcontractor of Reviewie Team, who are ION students.

**Reviewie Team:** Company that presents the project, who are ION students.

**Room:** The virtual place where users will chat with each other.

**Root:** Administrator account of ION-CHAT SYSTEM.

**SCmsg:** Server site communication message

**SDD:** Software Design Description

**Server Site:** The remote machine where the server software is to be executed.

**Server Software:** The part of the ION-CHAT SYSTEM that will be executed at the server site.

**Smsg:** Success message

**SRS:** Software requirements specification.

**System group:** A group of users, which have a common set of access restrictions to ION-CHAT SYSTEM.

**User Interface:** The interface that the users will see while using ION-CHAT SYSTEM.

**User:** The students, instructors, acquirers and the root.

**User\_id:** A set of characters that identifies the user.

## **2 References**

- [1] IEEE Std 1016-1998, Recommended Practice for Software Design Descriptions
- [2] ION-CHAT SYSTEM SRS, version 1.2, April 29<sup>th</sup>, 2002.

### **3 DECOMPOSITION DESCRIPTION**

A list of error messages [Emsg\_\*], success messages [Smsg\_\*] and communication messages [CCmsg\_\* and SCmsg\_\*] used on the DFD's can be seen in Appendix-A.

The structure charts of the ION-CHAT SYSTEM are shown in Appendix-G.

For an overall overview, Data Flow Diagrams of the ION-CHAT SYSTEM are shown in Appendix-E. A list of the codes used on the DFD's can be seen in Appendix-H.

#### **3.1 Module Decomposition**

##### **3.1.1 Module 1 Chat Client**

*Identification:* Module 1 Chat\_Client

*Type:* Packet

*Purpose:* To represent the collection of interfaces for users of the ION-CHAT SYSTEM.

*Function:* This module shall allow the users to enter the necessary inputs in order to login, to make basic operations about user, room or chat group and to chat in ION-CHAT SYSTEM, and to send the inputs entered to the server software.

*Subordinates:* Module 1.1 Client\_Site\_User\_Authentication

Module 1.2 Client\_Site\_Root\_Functions

Module 1.3 Client\_Site\_Chat

##### **3.1.1.1 Module 1.2 Client Site Root Functions**

*Identification:* Module 1.2 Client\_Site\_Root\_Functions

*Type:* Packet

*Purpose:* To represent the collection of interfaces for root in the ION-CHAT SYSTEM.

*Function:* This module shall allow the root to enter the necessary inputs in order to make basic operations about user, room or chat group in ION-CHAT SYSTEM, and to send the inputs entered to the server software.

*Subordinates:* Module 1.2.1 Client\_Site\_Root\_Console

Module 1.2.2 Client\_Site\_User\_Addition

Module 1.2.3 Client\_Site\_User\_Deletion

Module 1.2.4 Client\_Site\_User\_Update

Module 1.2.5 Client\_Site\_Room\_Addition

Module 1.2.6 Client\_Site\_Room\_Deletion

Module 1.2.7 Client\_Site\_Room\_Update

Module 1.2.8 Client\_Site\_Chat\_Group\_Addition

Module 1.2.9 Client\_Site\_Chat\_Group\_Deletion

Module 1.2.10 Client\_Site\_Chat\_Group\_Update

### **3.1.2 Module 2 Chat Server**

*Identification:* Module 2 Chat\_Server

*Type:* Packet

*Purpose:* To represent the collection of server site functions of the ION-CHAT SYSTEM.

*Function:* This module shall check the existence of the user, receive the inputs sent by the client software and finalize the basic operations about user, room or chat group, and send the message of clients to each other.

*Subordinates:* Module 2.1 Server\_Site\_User\_Authentication

Module 2.2 Server\_Site\_Root\_Functions

Module 2.3 Server\_Site\_Chat

#### **3.1.2.1 Module 2.2 Server Site Root Functions**

*Identification:* Module 2.2 Server\_Site\_Root\_Functions

*Type:* Packet

*Purpose:* To represent the collection of functions for root in the ION-CHAT SYSTEM.

*Function:* This module shall allow the root to receive the inputs sent by the client software and finalize the basic operations about user, room or chat group in ION-CHAT SYSTEM.

*Subordinates:* Module 2.2.1 Server\_Site\_Root\_Console

Module 2.2.2 Server \_Site\_User\_Addition

Module 2.2.3 Server \_Site\_User\_Deletion

Module 2.2.4 Server \_Site\_User\_Update

Module 2.2.5 Server \_Site\_Room\_Addition

Module 2.2.6 Server \_Site\_Room\_Deletion

Module 2.2.7 Server \_Site\_Room\_Update

Module 2.2.8 Server \_Site\_Chat\_Group\_Addition

Module 2.2.9 Server \_Site\_Chat\_Group\_Deletion

Module 2.2.10 Server \_Site\_Chat\_Group\_Update

## **3.2 Concurrent Process Decomposition**

This topic is not applicable for ION-CHAT SYSTEM SDD.

## **3.3 Data Decomposition**

### **3.3.1 Data Entity 1 User Information File**

*Identification:* Data Entity 1 User\_Information\_File

*Type:* Data store.

*Purpose:* To store the information regarding users.

*Function:* The structure of the entity for each individual entry shall be as follows:

user\_id:password:real\_name:system\_group:chat\_group:

*Subordinates:* None.

### **3.3.2 Data Entity 2 Room Information File**

*Identification:* Data Entity 2 Room\_Information\_File



*Type:* Data store.

*Purpose:* To store the information regarding rooms.

*Function:* The structure of the entity for each individual entry shall be as follows:

room\_name:reviewie\_group:reviewer\_group:customer\_user\_id:

*Subordinates:* None.

## **4 DEPENDENCY DESCRIPTION**

A list of error messages [Emsg\_\*], success messages [Smsg\_\*] and communication messages [CCmsg\_\* and SCmsg\_\*] used on the DFD's can be seen in Appendix-A.

A list of the codes used on the DFD's can be seen in Appendix-H.

### **4.1 Intermodule Dependencies**

#### **4.1.1 Module 1 Chat Client**

##### **4.1.1.1 Module 1.1 Client Site User Authentication**

*Identification:* Module 1.1 Client\_Site\_User\_Authentication

*Type:* Class

*Purpose:* To provide an interface, where the user shall enter the necessary inputs in order to login to the ION-CHAT SYSTEM, and to send the inputs entered to the server software.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-1 in Appendix-B.

*Resources:* java.applet.\*

java.awt.\*

java.awt.event.\*

java.net.\*

java.lang.\*

##### **4.1.1.2 Module 1.2 Client Site Root Functions**

###### **4.1.1.2.1 Module 1.2.1 Client Site Root Console**

*Identification:* Module 1.2.1 Client\_Site\_Root\_Console

*Type:* Class

*Purpose:* To provide an interface, where the root shall enter the necessary inputs in order to make basic operations about user, room or chat group of the ION-CHAT SYSTEM, and to send the inputs entered to the server software.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-2 in Appendix-B.

*Resources:* java.applet.\*

java.awt.\*

java.awt.event.\*

java.net.\*

java.lang.\*

#### **4.1.1.2.2 Module 1.2.2 Client Site User Addition**

*Identification:* Module 1.2.2 Client\_Site\_User\_Addition

*Type:* Class

*Purpose:* To provide an interface, where the root shall enter the necessary inputs in order to add a new user to the ION-CHAT SYSTEM, and to send the inputs entered to the server software.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-3 in Appendix-B.

*Resources:* java.applet.\*

java.awt.\*

java.awt.event.\*

java.net.\*

java.lang.\*

#### **4.1.1.2.3 Module 1.2.3 Client Site User Deletion**

*Identification:* Module 1.2.3 Client\_Site\_User\_Deletion

*Type:* Class

*Purpose:* To provide an interface, where the root shall approve deleting an existing user from the ION-CHAT SYSTEM, and to send the delete approval message to the server software.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-4 in Appendix-B.

*Resources*   java.applet.\*  
                  java.awt.\*  
                  java.awt.event.\*  
                  java.net.\*  
                  java.lang.\*

#### **4.1.1.2.4   Module 1.2.4 Client Site User Update**

*Identification:* Module 1.2.4 Client\_Site\_User\_Update

*Type:* Class

*Purpose:* To provide an interface, where the root shall enter the necessary inputs in order to update an existing user to the ION-CHAT SYSTEM, and to send the inputs entered to the server software.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-5 in Appendix-B.

*Resources:* java.applet.\*  
                  java.awt.\*  
                  java.awt.event.\*  
                  java.net.\*  
                  java.lang.\*

#### **4.1.1.2.5   Module 1.2.5 Client Site Room Addition**

*Identification:* Module 1.2.5 Client\_Site\_Room\_Addition

*Type:* Class

*Purpose:* To provide an interface, where the root shall enter the necessary inputs in order to add a new room to the ION-CHAT SYSTEM and to send the inputs entered to the server software.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-6 in Appendix-B.

*Resources:* java.applet.\*

java.awt.\*

java.awt.event.\*

java.net.\*

java.lang.\*

#### **4.1.1.2.6 Module 1.2.6 Client Site Room Deletion**

*Identification:* Module 1.2.6 Client\_Site\_Room\_Deletion

*Type:* Class

*Purpose:* To provide an interface, where the root shall approve deleting an existing room from the ION-CHAT SYSTEM, and to send the delete approval message to the server software.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-7 in Appendix-B.

*Resources:* java.applet.\*

java.awt.\*

java.awt.event.\*

java.net.\*

java.lang.\*

#### **4.1.1.2.7 Module 1.2.7 Client Site Room Update**

*Identification:* Module 1.2.7 Client\_Site\_Room\_Update

*Type:* Class

*Purpose:* To provide an interface, where the root shall enter the necessary inputs in order to update an existing Chat\_Group to the ION-CHAT SYSTEM, and to send the inputs entered to the server software.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-8 in Appendix-B.

*Resources:* java.applet.\*

java.awt.\*

java.awt.event.\*

java.net.\*

java.lang.\*

#### **4.1.1.2.8 Module 1.2.8 Client Site Chat Group Addition**

*Identification:* Module 1.2.8 Client\_Site\_Chat\_Group\_Addition

*Type:* Class

*Purpose:* To provide an interface, where the root shall enter the necessary inputs in order to add a new Chat\_Group to the ION-CHAT SYSTEM, and to send the inputs entered to the server software.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-9 in Appendix-B.

*Resources:* java.applet.\*

java.awt.\*

java.awt.event.\*

java.net.\*

java.lang.\*

#### **4.1.1.2.9 Module 1.2.9 Client Site Chat Group Deletion**

*Identification:* Module 1.2.9 Client\_Site\_Chat\_Group\_Deletion

*Type:* Class

*Purpose:* To provide an interface, where the root shall approve deleting an existing Chat\_Group from the ION-CHAT SYSTEM, and to send the delete approval message to the server software.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-10 in Appendix-B.

*Resources:* java.applet.\*

java.awt.\*

java.awt.event.\*

java.net.\*

java.lang.\*

#### **4.1.1.2.10 Module 1.2.10 Client Site Chat Group Update**

*Identification:* Module 1.2.10 Client\_Site\_Chat\_Group\_Update

*Type:* Class

*Purpose:* To provide an interface, where the root shall enter the necessary inputs in order to update an existing Chat\_Group to the ION-CHAT SYSTEM, and to send the inputs entered to the server software.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-11 in Appendix-B.

*Resources:* java.applet.\*

java.awt.\*

java.awt.event.\*

java.net.\*

java.lang.\*

#### **4.1.1.3 Module 1.3 Client Site Chat**

*Identification:* Module 1.3 Client\_Site\_Chat

*Type:* Class

*Purpose:* To provide an interface, where the user shall enter the necessary inputs in order to chat by using the ION-CHAT SYSTEM, and to send the inputs entered to the server software.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-12 in Appendix-B.

*Resources:* java.applet.\*

java.awt.\*

java.awt.event.\*

java.net.\*

java.lang.\*

## **4.1.2 Module 2 Chat Server**

### **4.1.2.1 Module 2.1 Server Site User Authentication**

*Identification:* Module 2.1 Server\_Site\_User\_Authentication

*Type:* Subroutine

*Purpose:* To check the existence of the user from the user information file of the ION-CHAT SYSTEM.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-13 in Appendix-B.

*Resources:* java.io.\*

java.net.\*

java.lang.\*

### **4.1.2.2 Module 2.2 Server Site Root Functions**

#### **4.1.2.2.1 Module 2.2.1 Server Site Root Console**

*Identification:* Module 2.2.1 Server\_Site\_Root\_Console

*Type:* Subroutine



*Purpose:* To receive the inputs sent by the client software and to finalize operation.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-14 in Appendix-B.

*Resources:* java.io.\*

java.net.\*

java.lang.\*

#### **4.1.2.2.2 Module 2.2.2 Server Site User Addition**

*Identification:* Module 2.2.2 Server\_Site\_User\_Addition

*Type:* Subroutine

*Purpose:* To receive the inputs sent by the client software and to finalize the user addition operation.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-15 in Appendix-B.

*Resources:* java.io.\*

java.net.\*

java.lang.\*

#### **4.1.2.2.3 Module 2.2.3 Server Site User Deletion**

*Identification:* Module 2.2.3 Server\_Site\_User\_Deletion

*Type:* Subroutine

*Purpose:* To receive the inputs sent by the client software and to finalize the user deletion operation.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-16 in Appendix-B.

*Resources:* java.io.\*

java.net.\*

java.lang.\*

#### **4.1.2.2.4 Module 2.2.4 Server Site User Update**

*Identification:* Module 2.2.4 Server\_Site\_User\_Update

*Type:* Subroutine

*Purpose:* To receive the inputs sent by the client software and to finalize the user update operation.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-17 in Appendix-B.

*Resources:* java.io.\*

java.net.\*

java.lang.\*

#### **4.1.2.2.5 Module 2.2.5 Server Site Room Addition**

*Identification:* Module 2.2.5 Server\_Site\_Room\_Addition

*Type:* Subroutine

*Purpose:* To receive the inputs sent by the client software and to finalize the room addition operation.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-18 in Appendix-B.

*Resources:* java.io.\*

java.net.\*

java.lang.\*

#### **4.1.2.2.6 Module 2.2.6 Server Site Room Deletion**

*Identification:* Module 2.2.6 Server\_Site\_Room\_Deletion

*Type:* Subroutine

*Purpose:* To receive the inputs sent by the client software and to finalize the room deletion operation.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-19 in Appendix-B.

*Resources:* java.io.\*

java.net.\*

java.lang.\*

#### **4.1.2.2.7 Module 2.2.7 Server Site Room Update**

*Identification:* Module 2.2.7 Server\_Site\_Room\_Update

*Type:* Subroutine

*Purpose:* To receive the inputs sent by the client software and to finalize the room update operation.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-20 in Appendix-B.

*Resources:* java.io.\*

java.net.\*

java.lang.\*

#### **4.1.2.2.8 Module 2.2.8 Server Site Chat Group Addition**

*Identification:* Module 2.2.8 Server\_Site\_Chat\_Group\_Addition

*Type:* Subroutine

*Purpose:* To receive the inputs sent by the client software and to finalize the chat group addition operation.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-21 in Appendix-B.

*Resources:* java.io.\*

java.net.\*

java.lang.\*

#### **4.1.2.2.9 Module 2.2.9 Server Site Chat Group Deletion**

*Identification:* Module 2.2.9 Server\_Site\_Chat\_Group\_Deletion

*Type:* Subroutine

*Purpose:* To receive the inputs sent by the client software and to finalize the chat group deletion operation.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-22 in Appendix-B.

*Resources:* java.io.\*

java.net.\*

java.lang.\*

#### **4.1.2.2.10 Module 2.2.10 Server Site Chat Group Update**

*Identification:* Module 2.2.10 Server\_Site\_Chat\_Group\_Update

*Type:* Subroutine

*Purpose:* To receive the inputs sent by the client software and to finalize the chat group update operation.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-23 in Appendix-B.

*Resources:* java.io.\*

java.net.\*

java.lang.\*

#### **4.1.2.3 Module 2.3 Server Site Chat**

*Identification:* Module 2.3 Server\_Site\_Chat

*Type:* Subroutine

*Purpose:* To send the message of clients to each other.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-24 in Appendix-B.

*Resources:* java.net.\*  
java.lang.\*

## **4.2 Interprocess Dependencies**

This topic is not applicable for ION-CHAT SYSTEM SDD.

## **4.3 Data Dependencies**

### **4.3.1 Data Entity 1 User Information File**

*Identification:* Data Entity 1 User\_Information\_File

*Type:* Data store.

*Purpose:* To store the information regarding users.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-25 in Appendix-B.

*Resources:* None.

### **4.3.2 Data Entity 2 Room Information File**

*Identification:* Data Entity 2 Room\_Information\_File

*Type:* Data store.

*Purpose:* To store the information regarding rooms.

*Dependencies:* The description of the data dependencies of this entity with other entities is given in Figure-26 in Appendix-B.

*Resources:* None

.

## 5 INTERFACE DESCRIPTION

A list of error messages [Emsg\_\*], success messages [Smsg\_\*] and communication messages [CCmsg\_\* and SCmsg\_\*] used on the DFD's can be seen in Appendix-A.

All of the communication messages are character strings. The communication of the messages shall be done by using Transmission Control Protocol (TCP) sockets.

Some of the details regarding the format of the communication messages are not specified in order to prevent duplication, these details can be found in Appendix-D

The complete description for the user interfaces is specified in subsection 3.1.1 of the SRS of ION-CHAT SYSTEM [2].

### 5.1 Module Interface

#### 5.1.1 Module 1 Chat Client

##### 5.1.1.1 Module 1.1 Client Site User Authentication

*Identification:* Module 1.1 Client\_Site\_User\_Authentication

*Function:* This module shall allow the user to enter the User\_id and Password. These inputs are checked according to the requirements specified in subclauses “Data format” and “Valid range” in section 3.1.1.1.a of [2]. If the check succeeds then inputs (CCmsg\_1) shall be sent to the module 2.1, else Emsg\_1 shall be displayed. Then the server response (SCmsg\_1 or SCmsg\_2) shall be waited. If the login failed Emsg\_11 shall be displayed.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_1	To send user_id and password of the user to the server software to authenticate the user.	None	SCmsg_1,

			SCmsg_2
SCmsg_1a	To authenticate the user to enter to the ION-CHAT SYSTEM for whom the system group is root	CCmsg_1	None
SCmsg_1b	To authenticate the user to enter to the ION-CHAT SYSTEM for whom the system group is instructor	CCmsg_1	None
SCmsg_1c	To authenticate the user to enter to the ION-CHAT SYSTEM for whom the system group is customer. Also, to specify the room that the customer is allowed to enter	CCmsg_1	None
SCmsg_1d	To authenticate the user to enter to the ION-CHAT SYSTEM for whom the system group is student. Also, to specify the rooms that the student is allowed to enter.	CCmsg_1	None
SCmsg_2	To indicate the failed authentication of the user to enter to the ION-CHAT SYSTEM	CCmsg_1	None

Code	Format
CCmsg_1	login:user_id:password:
SCmsg_1a	login:root:
SCmsg_1b	login:instructor:
SCmsg_1c	login:customer:room_name:
SCmsg_1d	login:student:room_name_as_reviewie_group: room_name_as_reviewer_group:
SCmsg_2	login:failed:

Code	Format Detail
CCmsg_1	login: static string
SCmsg_1a	login: static string root: static string
SCmsg_1b	login: static string instructor: static string
SCmsg_1c	login: static string customer: static string
SCmsg_1d	login: static string student: static string
SCmsg_2	login: static string failed: static string

### 5.1.1.2 Module 1.2 Client Site Root Functions

#### 5.1.1.2.1 Module 1.2.1 Client Site Root Console

*Identification:* Module 1.2.1 Client\_Site\_Root\_Console

*Function:* This module shall allow the root to select a basic operation and to enter User\_id, Room\_name, or Chat\_group\_name. This input is checked according to the requirements specified in subclauses “Data format” and “Valid range” in section 3.1.1.2.a of [2]. If check succeeds then inputs can be the following:

- CCmsg\_2 shall be sent to the module 2.2.1, else Emsg\_1 shall be displayed. Then the server response (SCmsg\_3 or SCmsg\_4) shall be waited for basic operations of *user*. Then if succeeds
  - User\_id is sent to module 1.2.2, else Emsg\_13 shall be displayed for *user addition*.
  - User\_id Password, Real\_name and System\_group are sent to module 1.2.3, else Emsg\_14 shall be displayed for *user deletion*.
  - User\_id Password, Real\_name and System\_group are sent to module 1.2.4, else Emsg\_14 shall be displayed for *user update*.
- CCmsg\_3 shall be sent to the module 2.2.1, else Emsg\_1 shall be displayed. Then the server response (SCmsg\_5 or SCmsg\_6) shall be waited for basic operations of *room*. Then if succeeds
  - Room\_name is sent to module 1.2.5, else Emsg\_15 shall be displayed for *room addition*.
  - Room\_name, Reviewie Group, Reviewer Group and User\_id of Customer are sent to module 1.2.6, else Emsg\_16 shall be displayed for *room deletion*.
  - Room\_name, Reviewie Group, Reviewer Group and User\_id of Customer are sent to module 1.2.7, else Emsg\_16 shall be displayed for *room update*.
- CCmsg\_4 shall be sent to the module 2.2.1, else Emsg\_1 shall be displayed. Then the server response (SCmsg\_7 or SCmsg\_8) shall be waited for basic operations of *chat group*. Then if succeeds



- Chat\_group\_name is sent to module 1.2.8, else Emsg\_17 shall be displayed for *chat group addition*.
- Chat\_group\_name and User\_id\_list are sent to module 1.2.9, else Emsg\_18 shall be displayed for *chat group deletion*.
- Chat\_group\_name and User\_id\_list are sent to module 1.2.10, else Emsg\_18 shall be displayed for *chat group update*.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_2	To query the existence of a username	None	SCmsg_3, SCmsg_4
SCmsg_3	To send the information (specified in format of SCmsg_3) about the user whose user_id is taken from the client software	CCmsg_2	None
SCmsg_4	To indicate the non-existence of the user.	CCmsg_2	None
CCmsg_3	To query the existence of a room name	None	SCmsg_5, SCmsg_6
SCmsg_5	To send the information (specified in format of SCmsg_5) about the room whose room_name is taken from the client software	CCmsg_3	None
SCmsg_6	To indicate the non-existence of the room.	CCmsg_3	None
CCmsg_4	To query the existence of a chat group	None	SCmsg_7, SCmsg_8
SCmsg_7	To send the information (specified in format of SCmsg_7) about the chat group whose chat_group is taken from the client software	CCmsg_4	None
SCmsg_8	To indicate the non-existence of the chat group.	CCmsg_4	None

Code	Format
CCmsg_2	askuser:user_id:
SCmsg_3	askuser:user_id:password:real_name:system_group:
SCmsg_4	askuser:failed:
CCmsg_3	askroom:room_name:
SCmsg_5	askroom:room_name:reviewie_group:reviewer_group:

	customer_user_id:
SCmsg_6	askroom:failed:
CCmsg_4	askchatgroup:chat_group:
SCmsg_7	askchatgroup:chat_group:user_id_list:
SCmsg_8	askchatgroup:failed:

Code	Format Detail
CCmsg_2	askuser: static string
SCmsg_3	askuser: static string
SCmsg_4	askuser: static string failed: static string
CCmsg_3	askroom: static string
SCmsg_5	askroom: static string
SCmsg_6	askroom: static string failed: static string
CCmsg_4	askchatgroup: static string
SCmsg_7	askchatgroup: static string user_id: semicolon seperated user_id list
SCmsg_8	askchatgroup: static string failed: static string

#### 5.1.1.2.2 Module 1.2.2 Client Site User Addition

*Identification:* Module 1.2.2 Client\_Site\_User\_Addition

*Function:* This module shall allow the root to enter the User\_id, Password, Confirm\_password, Real\_name, System\_group. These inputs are checked according to the requirements specified in subclauses “Data format” and “Valid range” in section 3.1.1.2.b of [2]. If the check succeeds then inputs (CCmsg\_5) shall be sent to the module 2.2.2 else Emsg\_1 shall be displayed. Then the server response (SCmsg\_9 or SCmsg\_10) shall be waited. If the add succeeds Smsg\_3, else Emsg\_4 shall be displayed.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_5	To get the necessary inputs (specified in format of CCmsg_5) for addition of a new user from the root and send them to server software.	None	SCmsg_9, SCmsg_10
SCmsg_9	To indicate the successful addition of the user.	CCmsg_5	None

SCmsg_10	To indicate the failed addition of the user.	CCmsg_5	None
----------	--	---------	------

Code	Format
CCmsg_5	adduser:user_id:password:real_name:system_group:
SCmsg_9	adduser:successful:
SCmsg_10	adduser:failed:

Code	Format Detail
CCmsg_5	adduser: static string
SCmsg_9	adduser: static string successful: static string
SCmsg_10	adduser: static string failed: static string

Code of Method	Method Name	Method Interface	Method Belongs To
f2	method2	method2(String user_id)	Module 1.2.2

#### 5.1.1.2.3 Module 1.2.3 Client Site User Deletion

*Identification:* Module 1.2.3 Client\_Site\_User\_Deletion

*Function:* This module shall allow the root to see the information related to the user to be deleted. The user information shall be composed of User\_id, Password, Confirm\_password, Real\_name and System\_group parts. The approval message (CCmsg\_7) shall be sent to the module 2.2.3. Then server response (SCmsg\_13 or SCmsg\_14) shall be waited. If the delete succeeds Smsg\_1, else Emsg\_2 shall be displayed.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_7	To get the user_id for deletion of an existing user from the root and send it to server software.	None	SCmsg_13, SCmsg_14
SCmsg_13	To indicate the successful deletion of the user.	CCmsg_7	None
SCmsg_14	To indicate the failed deletion of the user.	CCmsg_7	None

Code	Format
CCmsg_7	deleteuser:user_id:
SCmsg_13	deleteuser:successful:
SCmsg_14	deleteuser:failed:

Code	Format Detail
CCmsg_7	deleteuser: static string
SCmsg_13	deleteuser: static string successful: static string
SCmsg_14	deleteuser: static string failed: static string

Code of Method	Method Name	Method Interface	Method Belongs To
f3	method3	method3(String user_id, String password, String real_name, String system_group)	Module 1.2.3

#### 5.1.1.2.4 Module 1.2.4 Client Site User Update

*Identification:* Module 1.2.4 Client\_Site\_User\_Update

*Function:* This module shall allow the root to enter the new Password, Confirm\_password, Real\_name, System\_group. These inputs are checked according to the requirements specified in subclauses “Data format” and “Valid range” in section 3.1.1.2.d of [2]. If the check succeeds then inputs (CCmsg\_6) shall be sent to the module 2.2.4, else Emsg\_1 shall be displayed. Then the server response (SCmsg\_11 or SCmsg\_12) shall be waited. If the add succeeds Smsg\_2, else Emsg\_3 shall be displayed.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_6	To get the necessary inputs (specified in format of CCmsg_6) for update of an existing user from the root and send them to server software.	None	SCmsg_11, SCmsg_12
SCmsg_11	To indicate the successful update of the user.	CCmsg_6	None

SCmsg_12	To indicate the failed update of the user.	CCmsg_6	None
----------	--	---------	------

Code	Format
CCmsg_6	updateuser:user_id:password:real_name:system_group:
SCmsg_11	updateuser:successful:
SCmsg_12	updateuser:failed:

Code	Format Detail
CCmsg_6	updateuser: static string
SCmsg_11	updateuser: static string successful: static string
SCmsg_12	updateuser: static string failed: static string

Code of Method	Method Name	Method Interface	Method Belongs To
f4	method4	method4(String user_id, String password, String real_name, String system_group)	Module 1.2.4

#### 5.1.1.2.5 Module 1.2.5 Client Site Room Addition

*Identification:* Module 1.2.5 Client\_Site\_Room\_Addition

*Function:* This module shall allow the root to enter the Room\_name, Reviewie Group, Reviewer Group and User\_id of Customer. These inputs are checked according to the requirements specified in subclauses “Data format” and “Valid range” in section 3.1.1.2.e of [2]. If the check succeeds then inputs (CCmsg\_8) shall be sent to the module 2.2.5 else Emsg\_1 shall be displayed. Then the server response (SCmsg\_15 or SCmsg\_16) shall be waited. If the add succeeds Smsg\_6, else Emsg\_7 shall be displayed

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_8	To get the necessary inputs (specified in format of CCmsg_8) for addition of a new room from the root and send them to server software.	None	SCmsg_15, SCmsg_16

SCmsg_15	To indicate the successful addition of the room.	CCmsg_8	None
SCmsg_16	To indicate the failed addition of the room.	CCmsg_8	None

Code	Format
CCmsg_8	addroom:room_name:reviewie_group:reviewer_group: customer_user_id:
SCmsg_15	addroom:successful:
SCmsg_16	addroom:failed:

Code	Format Detail
CCmsg_8	addroom: static string
SCmsg_15	addroom: static string successful: static string
SCmsg_16	addroom: static string failed: static string

Code of Method	Method Name	Method Interface	Method Belongs To
f5	method5	method5(String room_name)	Module 1.2.5

#### 5.1.1.2.6 Module 1.2.6 Client Site Room Deletion

*Identification:* Module 1.2.6 Client\_Site\_Room\_Deletion

*Function:* This module shall allow the root to see the information related to the room to be deleted. The room information shall be composed of Room\_name, Review Group, Reviewer Group, User\_id of Customer parts. The approval message (CCmsg\_10) shall be sent to the module 2.2.6 and the server response (SCmsg\_19 or SCmsg\_20) shall be waited. If the delete succeeds Smsg\_4, else Emsg\_5 shall be displayed.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_10	To get the room_name for deletion of an existing room from the root and send it to server software.	None	SCmsg_19, SCmsg_20
SCmsg_19	To indicate the successful deletion of the room.	CCmsg_10	None

SCmsg_20	To indicate the failed deletion of the room.	CCmsg_10	None
----------	--	----------	------

Code	Format
CCmsg_10	deleteroom:room_name:
SCmsg_19	deleteroom:successful:
SCmsg_20	deleteroom:failed:

Code	Format Detail
CCmsg_10	deleteroom: static string
SCmsg_19	deleteroom: static string successful: static string
SCmsg_20	deleteroom: static string failed: static string

Code of Method	Method Name	Method Interface	Method Belongs To
f6	method6	method6(String room_name, String reviewie_group, String reviewer_group, String user_id_of_customer)	Module 1.2.6

#### 5.1.1.2.7 Module 1.2.7 Client Site Room Update

*Identification:* Module 1.2.7 Client\_Site\_Room\_Update

*Function:* This module shall allow the root to enter the new Review Group, Reviewer Group, User\_id of Customer. These inputs are checked according to the requirements specified in subclauses “Data format” and “Valid range” in section 3.1.1.2.g of [2]. If the check succeeds then inputs (CCmsg\_9) shall be sent to the module 2.2.7, else Emsg\_1 shall be displayed. Then the server response (SCmsg\_17 or SCmsg\_18) shall be waited. If the add succeeds Smsg\_5, else Emsg\_6 shall be displayed.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_9	To get the necessary inputs (specified in format of CCmsg_9) for update of an existing room from the root and send them to server software.	None	SCmsg_17, SCmsg_18

SCmsg_17	To indicate the successful update of the room.	CCmsg_9	None
SCmsg_18	To indicate the failed update of the room.	CCmsg_9	None

Code	Format
CCmsg_9	updateroom:room_name:reviewie_group: reviewer_group:customer_user_id:
SCmsg_17	updateroom:successful:
SCmsg_18	updateroom:failed:

Code	Format Detail
CCmsg_9	updateroom: static string
SCmsg_17	updateroom: static string successful: static string
SCmsg_18	updateroom: static string failed: static string

Code of Method	Method Name	Method Interface	Method Belongs To
f7	method7	method7(String room_name, String reviewie_group, String reviewer_group, String user_id_of_customer)	Module 1.2.7

#### 5.1.1.2.8 Module 1.2.8 Client Site Chat Group Addition

*Identification:* Module 1.2.8 Client\_Site\_Chat\_Group\_Addition

*Purpose:* To provide an interface, where the root shall enter the necessary inputs in order to add a new Chat\_Group to the ION-CHAT SYSTEM, and to send the inputs entered to the server software.

*Function:* This module shall allow the root to enter the Chat Group Name and User\_id List. These inputs are checked according to the requirements specified in subclauses “Data format” and “Valid range” in section 3.1.1.2.h of [2]. If the check succeeds then inputs (CCmsg\_11) shall be sent to the module 2.2.8 else Emsg\_1 shall be displayed. Then the server response (SCmsg\_21 or SCmsg\_22) shall be waited. If the add succeeds Smsg\_9, else Emsg\_10 shall be displayed.



*Interface:*

Code	Use	Predecessor	Successor
CCmsg_11	To get the necessary inputs (specified in format of CCmsg_11) for addition of a new chat group from the root and send them to server software.	None	SCmsg_21, SCmsg_22
SCmsg_21	To indicate the successful addition of the chat group.	CCmsg_11	None
SCmsg_22	To indicate the failed addition of the chat group.	CCmsg_11	None

Code	Format
CCmsg_11	addchatgroup:chat_group:user_id_list:
SCmsg_21	addchatgroup:successful:
SCmsg_22	addchatgroup:failed:

Code	Format Detail
CCmsg_11	addchatgroup: static string user_id: semicolon seperated user_id list
SCmsg_21	addchatgroup: static string successful: static string

Code of Method	Method Name	Method Interface	Method Belongs To
f8	method8	method8(String chat_group_name)	Module 1.2.8

#### 5.1.1.2.9 Module 1.2.9 Client Site Chat Group Deletion

*Identification:* Module 1.2.9 Client\_Site\_Chat\_Group\_Deletion

*Purpose:* To provide an interface, where the root shall approve deleting an existing Chat\_Group from the ION-CHAT SYSTEM, and to send the delete approval message to the server software.

*Function:* This module shall allow the root to see the information related to the Chat\_Group to be deleted. The Chat\_Group information shall be composed of Chat Group Name and User\_id List. The approval message (CCmsg\_13) shall be sent to the

module 2.2.9 and the server response (SCmsg\_25 or SCmsg\_26) shall be waited. If the delete succeeds Smsg\_7, else Emsg\_8 shall be displayed.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_13	To get the chat_group for deletion of an existing chat group from the root and send it to server software.	None	SCmsg_25, SCmsg_26
SCmsg_25	To indicate the successful deletion of the chat group.	CCmsg_13	None
SCmsg_26	To indicate the failed deletion of the chat group.	CCmsg_13	None

Code	Format
CCmsg_13	deletechatgroup:chat_group:
SCmsg_25	deletechatgroup:successful:
SCmsg_26	deletechatgroup:failed:

Code	Format Detail
CCmsg_13	deletechatgroup: static string
SCmsg_25	deletechatgroup: static string successful: static string
SCmsg_26	deletechatgroup: static string failed: static string

Code of Method	Method Name	Method Interface	Method Belongs To
f9	method9	method9(String chat_group_name, String user_id_list)	Module 1.2.9

#### 5.1.1.2.10 Module 1.2.10 Client Site Chat Group Update

*Identification:* Module 1.2.10 Client\_Site\_Chat\_Group\_Update

*Function:* This module shall allow the root to enter the new User\_id List. This input is checked according to the requirements specified in subclauses “Data format” and “Valid range” in section 3.1.1.2.j of [2]. If the check succeeds then inputs (CCmsg\_12) shall be sent to the module 2.2.10, else Emsg\_1shall be displayed. Then the server response

(SCmsg\_23 or SCmsg\_24) shall be waited. If the add succeeds Smsg\_8, else Emsg\_9 shall be displayed.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_12	To get the necessary inputs (specified in format of CCmsg_12) for update of an existing chat group from the root and send them to server software.	None	SCmsg_23, SCmsg_24
SCmsg_23	To indicate the successful update of the chat group.	CCmsg_12	None
SCmsg_24	To indicate the failed update of the chat group.	CCmsg_12	None

Code	Format
CCmsg_12	updatechatgroup:chat_group:user_id_list:
SCmsg_23	updatechatgroup:successful:
SCmsg_24	updatechatgroup:failed:

Code	Format Detail
CCmsg_12	updatechatgroup: static string user_id: semicolon seperated user_id list
SCmsg_23	updatechatgroup: static string successful: static string
SCmsg_24	updatechatgroup: static string failed: static string

Code of Method	Method Name	Method Interface	Method Belongs To
f10	method10	method10(String chat_group_name, String user_id_list)	Module 1.2.10

### 5.1.1.3 Module 1.3 Client Site Chat

*Identification:* Module 1.3 Client\_Site\_Chat

*Function:* This module shall allow the user to enter the message to the User\_area text box. This input is checked according to the requirements specified in subclauses “Data

format” and “Valid range” in section 3.1.1.3.a of [2]. If the check succeeds then inputs can be the following four option

- CCmsg\_14 shall be sent to the module 2.3, else Emsg\_1 shall be displayed. Then the server response (SCmsg\_27) shall be waited. Then if succeeds the message of the user shall be shown in the General\_area text box.
- CCmsg\_15, shall be sent to the module 2.3, else Emsg\_1 shall be displayed. Then the server response (SCmsg\_28) shall be waited. Then if succeeds the message of the user shall be shown in the General\_area text box.
- CCmsg\_16, shall be sent to the module 2.3, else Emsg\_1 shall be displayed. Then the server response (SCmsg\_29 and SCmsg\_30 or SCmsg\_31) shall be waited. Then if succeeds the message of the user shall be shown in the General\_area text box.
- CCmsg\_17) shall be sent to the module 2.3, else Emsg\_1 shall be displayed. Then the server response (SCmsg\_32 or SCmsg\_33) shall be waited. Then if succeeds the message of the user shall be shown in the General\_area text box.

When user clicks to the Logout\_button to logout from the ION-CHAT SYSTEM and if succeeds the Smsg\_10, else Emsg\_12 shall be displayed.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_14	To get the message from the user and send it to the server software	None	SCmsg_27
SCmsg_27	To get the message from the client software and send it to the all client software which are in the same room.	CCmsg_14	None
CCmsg_15	To query the room list of ION-CHAT SYSTEM	None	SCmsg_28
SCmsg_28	To send the room_list to the client software.	CCmsg_15	None
CCmsg_16	To query the user list of ION-CHAT SYSTEM for a specific room name.	None	SCmsg_29, SCmsg_31
SCmsg_29	To send the user_list of a specific room to the client software.	CCmsg_16	None
SCmsg_30	To send the active user_list of a specific room to the client software when a new	None	None

	user enter to the room.		
SCmsg_31	To indicate the failed query of the user_list of a specific room.	CCmsg_16	None
CCmsg_17	To send user_id of the user to the server software to logout the user.	None	SCmsg_32, SCmsg_33
SCmsg_32	To indicate the successful logout of the user from the ION-CHAT SYSTEM	CCmsg_17	None
SCmsg_33	To indicate the failed logout of the user from the ION-CHAT SYSTEM	CCmsg_17	None

Code	Format
CCmsg_14	chatmsg:chat_group:real_name:chat_message
SCmsg_27	chatmsg:chat_group:real_name:chat_message
CCmsg_15	askroomlist:
SCmsg_28	askroomlist:room_name_list:
CCmsg_16	askalluserlist:room_name:
SCmsg_29	askalluserlist:user_id/chat_group/real_name:user_id/ chat_group/real_name:user_id/customer/real_name: user_id/instructor/real_name:
SCmsg_30	askactiveuserlist:user_id_list
SCmsg_31	askalluserlist:failed:
CCmsg_17	logout:user_id:
SCmsg_32	logout:successful:
SCmsg_33	logout:failed:

Code	Format Detail
CCmsg_14	chatmsg: static string
SCmsg_27	chatmsg: static string
CCmsg_15	askroomlist: static string
SCmsg_28	askroomlist: static string room_name_list: semicolon seperated room_name list
CCmsg_16	askalluserlist: static string
SCmsg_29	askalluserlist: static string user_id/chat_group/real_name: semicolon seperated list for reviewie group

	<p>members</p> <p>user_id/chat_group/real_name: semicolon seperated list for reviewier group members</p> <p>user_id/customer/real_name: for the customer assigned to the room</p> <p>user_id/instructor/real_name: semicolon seperated list for all instructors</p>
SCmsg_30	<p>askactiveuserlist: static string</p> <p>user_id: semicolon seperated user_id list</p>
SCmsg_31	<p>askalluserlist: static string</p> <p>failed: static string</p>
CCmsg_17	logout: static string
SCmsg_32	<p>logout: static string</p> <p>successful: static string</p>
SCmsg_33	<p>logout: static string</p> <p>failed: static string</p>

Code of Method	Method Name	Method Interface	Method Belongs To
f1	method1	method1(String user_id, String reviewie_group_room_name, String reviewer_group_room_name, String system_group)	Module 1.3

### 5.1.2 Module 2 Chat Server

#### 5.1.2.1 Module 2.1 Server Site User Authentication

*Identification:* Module 2.1 Server\_Site\_User\_Authentication

*Function:* This module shall get the inputs (CCmsg\_1) from the module 1.1 to check the existence of the user form the User Information File. If the check succeeds then SCmsg\_1 else SCmsg\_2 shall be sent to the module 1.1.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_1	To send user_id and password of the user to the server software to authenticate the user.	None	SCmsg_1, SCmsg_2
SCmsg_1a	To authenticate the user to enter to the ION-CHAT SYSTEM for whom the system group is root	CCmsg_1	None

SCmsg_1b	To authenticate the user to enter to the ION-CHAT SYSTEM for whom the system group is instructor	CCmsg_1	None
SCmsg_1c	To authenticate the user to enter to the ION-CHAT SYSTEM for whom the system group is customer. Also, to specify the room that the customer is allowed to enter	CCmsg_1	None
SCmsg_1d	To authenticate the user to enter to the ION-CHAT SYSTEM for whom the system group is student. Also, to specify the rooms that the student is allowed to enter.	CCmsg_1	None
SCmsg_2	To indicate the failed authentication of the user to enter to the ION-CHAT SYSTEM	CCmsg_1	None

Code	Format
CCmsg_1	login:user_id:password:
SCmsg_1a	login:root:
SCmsg_1b	login:instructor:
SCmsg_1c	login:customer:room_name:
SCmsg_1d	login:student:room_name_as_reviewie_group: room_name_as_reviewer_group:
SCmsg_2	login:failed:

Code	Format Detail
CCmsg_1	login: static string
SCmsg_1a	login: static string root: static string
SCmsg_1b	login: static string instructor: static string
SCmsg_1c	login: static string customer: static string
SCmsg_1d	login: static string student: static string
SCmsg_2	login: static string failed: static string

### 5.1.2.2 Module 2.2 Server Site Root Functions

#### 5.1.2.2.1 Module 2.2.1 Server Site Root Console

*Identification:* Module 2.2.1 Server\_Site\_Root\_Console

*Function:* This module shall receive

- CCmsg\_2 from the module 1.2.1. User\_id\_list shall be taken from the User Information File to check the existence of the user. If the user exists SCmsg\_3 else SCmsg\_4 shall be sent to the module 1.2.1 for *user addition*.
- CCmsg\_2 from the module 1.2.1. User\_id, Password, Real\_name and System\_group shall be taken from the User Information File according to the specification given in subsection 3.3.1. If the operation succeeds SCmsg\_3 else SCmsg\_4 shall be sent to the module 1.2.1 for *user deletion and user update*.
- CCmsg\_3 from the module 1.2.1. Room\_name\_list shall be taken from the Room Information File to check the existence of the room. If the room exists SCmsg\_5 else SCmsg\_6 shall be sent to the module 1.2.1 for *room addition*.
- CCmsg\_3 from the module 1.2.1. Room\_name, Reviewie Group, Reviewer Group and User\_id of Customer shall be taken from the Room Information File according to the specification given in subsection 3.3.2. If the operation succeeds SCmsg\_5 else SCmsg\_6 shall be sent to the module 1.2.1 for *room deletion and room update*.
- CCmsg\_4 from the module 1.2.1. Chat\_group\_name\_list shall be taken from the User Information File to check the existence of the chat group. If the chat group exists SCmsg\_7 else SCmsg\_8 shall be sent to the module 1.2.1 for *chat group addition*.
- CCmsg\_4 from the module 1.2.1. Chat\_group\_name and User\_id\_list shall be taken from the User Information File according to the specification given in subsection 3.3.1. If the operation succeeds SCmsg\_7 else SCmsg\_8 shall be sent to the module 1.2.1 for *chat group deletion and chat group update*.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_2	To query the existence of a username	None	SCmsg_3, SCmsg_4



SCmsg_3	To send the information (specified in format of SCmsg_3) about the user whose user_id is taken from the client software	CCmsg_2	None
SCmsg_4	To indicate the non-existence of the user.	CCmsg_2	None
CCmsg_3	To query the existence of a room name	None	SCmsg_5, SCmsg_6
SCmsg_5	To send the information (specified in format of SCmsg_5) about the room whose room_name is taken from the client software	CCmsg_3	None
SCmsg_6	To indicate the non-existence of the room.	CCmsg_3	None
CCmsg_4	To query the existence of a chat group	None	SCmsg_7, SCmsg_8
SCmsg_7	To send the information (specified in format of SCmsg_7) about the chat group whose chat_group is taken from the client software	CCmsg_4	None
SCmsg_8	To indicate the non-existence of the chat group.	CCmsg_4	None

Code	Format
CCmsg_2	askuser:user_id:
SCmsg_3	askuser:user_id:password:real_name:system_group:
SCmsg_4	askuser:failed:
CCmsg_3	askroom:room_name:
SCmsg_5	askroom:room_name:reviewie_group:reviewer_group: customer_user_id:
SCmsg_6	askroom:failed:
CCmsg_4	askchatgroup:chat_group:
SCmsg_7	askchatgroup:chat_group:user_id_list:
SCmsg_8	askchatgroup:failed:

Code	Format Detail
CCmsg_2	askuser: static string
SCmsg_3	askuser: static string
SCmsg_4	askuser: static string failed: static string
CCmsg_3	askroom: static string

SCmsg_5	askroom: static string
SCmsg_6	askroom: static string failed: static string
CCmsg_4	askchatgroup: static string
SCmsg_7	askchatgroup: static string user_id: semicolon seperated user_id list
SCmsg_8	askchatgroup: static string failed: static string

#### 5.1.2.2.2 Module 2.2.2 Server Site User Addition

*Identification:* Module 2.2.2 Server\_Site\_User\_Addition

*Function:* This module shall receive the inputs (CCmsg\_5) from the module 1.2.2. The User Information File shall be modified by reformatting User\_id, Password, Real\_name and System Group according to the specification given in subsection 3.3.1. If the operation succeeds SCmsg\_9. else SCmsg\_10 shall be sent to the module 1.2.2.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_5	To get the necessary inputs (specified in format of CCmsg_5) for addition of a new user from the root and send them to server software.	None	SCmsg_9, SCmsg_10
SCmsg_9	To indicate the successful addition of the user.	CCmsg_5	None
SCmsg_10	To indicate the failed addition of the user.	CCmsg_5	None

Code	Format
CCmsg_5	adduser:user_id:password:real_name:system_group:
SCmsg_9	adduser:successful:
SCmsg_10	adduser:failed:

Code	Format Detail
CCmsg_5	adduser: static string
SCmsg_9	adduser: static string successful: static string
SCmsg_10	adduser: static string failed: static string

### 5.1.2.2.3 Module 2.2.3 Server Site User Deletion

*Identification:* Module 2.2.3 Server\_Site\_User\_Deletion

*Function:* This module shall receive the inputs (CCmsg\_7) from the module 1.2.3. The User Information File shall be modified by deleting the user from the User Information File. If the operation succeeds SCmsg\_13 else SCmsg\_14 shall be sent to the module 1.2.3.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_7	To get the user_id for deletion of an existing user from the root and send it to server software.	None	SCmsg_13, SCmsg_14
SCmsg_13	To indicate the successful deletion of the user.	CCmsg_7	None
SCmsg_14	To indicate the failed deletion of the user.	CCmsg_7	None

Code	Format
CCmsg_7	deleteuser:user_id:
SCmsg_13	deleteuser:successful:
SCmsg_14	deleteuser:failed:

Code	Format Detail
CCmsg_7	deleteuser: static string
SCmsg_13	deleteuser: static string successful: static string
SCmsg_14	deleteuser: static string failed: static string

### 5.1.2.2.4 Module 2.2.4 Server Site User Update

*Identification:* Module 2.2.4 Server\_Site\_User\_Update

*Function:* This module shall receive the inputs (CCmsg\_6) from the module 1.2.4. The User Information File shall be modified by reformatting Password, Real\_name and

System\_group according to the specification given in subsection 3.3.1. If the operation succeeds SCmsg\_11, else SCmsg\_12 shall be sent to the module 1.2.4.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_6	To get the necessary inputs (specified in format of CCmsg_6) for update of an existing user from the root and send them to server software.	None	SCmsg_11, SCmsg_12
SCmsg_11	To indicate the successful update of the user.	CCmsg_6	None
SCmsg_12	To indicate the failed update of the user.	CCmsg_6	None

Code	Format
CCmsg_6	updateuser:user_id:password:real_name:system_group:
SCmsg_11	updateuser:successful:
SCmsg_12	updateuser:failed:

Code	Format Detail
CCmsg_6	updateuser: static string
SCmsg_11	updateuser: static string successful: static string
SCmsg_12	updateuser: static string failed: static string

#### 5.1.2.2.5 Module 2.2.5 Server Site Room Addition

*Identification:* Module 2.2.5 Server\_Site\_Room\_Addition

*Function:* This module shall receive the inputs (CCmsg\_8) from the module 1.2.5. The Room Information File shall be modified by reformatting Room\_name, Reviewie Group, Reviewer Group and User\_id of Customer according to the specification given in subsection 3.3.2. If the operation succeeds SCmsg\_15. else SCmsg\_16 shall be sent to the module 1.2.5.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_8	To get the necessary inputs (specified in format of CCmsg_8) for addition of a new room from	None	SCmsg_15,

	the root and send them to server software.		SCmsg_16
SCmsg_15	To indicate the successful addition of the room.	CCmsg_8	None
SCmsg_16	To indicate the failed addition of the room.	CCmsg_8	None

Code	Format
CCmsg_8	addroom:room_name:reviewie_group:reviewer_group: customer_user_id:
SCmsg_15	addroom:successful:
SCmsg_16	addroom:failed:

Code	Format Detail
CCmsg_8	addroom: static string
SCmsg_15	addroom: static string successful: static string
SCmsg_16	addroom: static string failed: static string

#### 5.1.2.2.6 Module 2.2.6 Server Site Room Deletion

*Identification:* Module 2.2.6 Server\_Site\_Room\_Deletion

*Function:* This module shall receive the inputs (CCmsg\_10) from the module 1.2.5. The User Information File shall be modified by deleting the user from Room Information File. If the operation succeeds SCmsg\_19, else SCmsg\_20 shall be sent to the module 1.2.5.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_10	To get the room_name for deletion of an existing room from the root and send it to server software.	None	SCmsg_19, SCmsg_20
SCmsg_19	To indicate the successful deletion of the room.	CCmsg_10	None
SCmsg_20	To indicate the failed deletion of the room.	CCmsg_10	None

Code	Format
CCmsg_10	deleteroom:room_name:

SCmsg_19	deleteroom:successful:
SCmsg_20	deleteroom:failed:

Code	Format Detail
CCmsg_10	deleteroom: static string
SCmsg_19	deleteroom: static string successful: static string
SCmsg_20	deleteroom: static string failed: static string

#### 5.1.2.2.7 Module 2.2.7 Server Site Room Update

*Identification:* Module 2.2.7 Server\_Site\_Room\_Update

*Function:* This module shall receive the inputs (CCmsg\_9) from the module 1.2.7. The Room Information File shall be modified by reformatting Reviewie Group, Reviewer Group and User\_id of Customer according to the specification given in subsection 3.3.2. If the operation succeeds SCmsg\_17, else SCmsg\_18 shall be sent to the module 1.2.7.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_9	To get the necessary inputs (specified in format of CCmsg_9) for update of an existing room from the root and send them to server software.	None	SCmsg_17, SCmsg_18
SCmsg_17	To indicate the successful update of the room.	CCmsg_9	None
SCmsg_18	To indicate the failed update of the room.	CCmsg_9	None

Code	Format
CCmsg_9	updateroom:room_name:reviewie_group: reviewer_group:customer_user_id:
SCmsg_17	updateroom:successful:
SCmsg_18	updateroom:failed:

Code	Format Detail
CCmsg_9	updateroom: static string
SCmsg_17	updateroom: static string successful: static string

SCmsg_18	updateroom: static string failed: static string
----------	--

#### 5.1.2.2.8 Module 2.2.8 Server Site Chat Group Addition

*Identification:* Module 2.2.8 Server\_Site\_Chat\_Group\_Addition

*Function:* This module shall receive the inputs (CCmsg\_11) from the module 1.2.8. The User Information File shall be modified by reformatting Chat\_group\_name User\_id\_list according to the specification given in subsection 3.3.1. If the operation succeeds SCmsg\_21, else SCmsg\_22 shall be sent to the module 1.2.8.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_11	To get the necessary inputs (specified in format of CCmsg_11) for addition of a new chat group from the root and send them to server software.	None	SCmsg_21, SCmsg_22
SCmsg_21	To indicate the successful addition of the chat group.	CCmsg_11	None
SCmsg_22	To indicate the failed addition of the chat group.	CCmsg_11	None

Code	Format
CCmsg_11	addchatgroup:chat_group:user_id_list:
SCmsg_21	addchatgroup:successful:
SCmsg_22	addchatgroup:failed:

Code	Format Detail
CCmsg_11	addchatgroup: static string user_id: semicolon seperated user_id list
SCmsg_21	addchatgroup: static string successful: static string
SCmsg_22	addchatgroup: static string failed: static string

#### 5.1.2.2.9 Module 2.2.9 Server Site Chat Group Deletion

*Identification:* Module 2.2.9 Server\_Site\_Chat\_Group\_Deletion

*Function:* This module shall receive the inputs (CCmsg\_13) from the module 1.2.9. The User Information File shall be modified by deleting the Chat\_group\_name from all the entries. If the operation succeeds SCmsg\_25, else SCmsg\_26 shall be sent to the module 1.2.9.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_13	To get the chat_group for deletion of an existing chat group from the root and send it to server software.	None	SCmsg_25, SCmsg_26
SCmsg_25	To indicate the successful deletion of the chat group.	CCmsg_13	None
SCmsg_26	To indicate the failed deletion of the chat group.	CCmsg_13	None

Code	Format
CCmsg_13	deletechatgroup:chat_group:
SCmsg_25	deletechatgroup:successful:
SCmsg_26	deletechatgroup:failed:

Code	Format Detail
CCmsg_13	deletechatgroup: static string
SCmsg_25	deletechatgroup: static string successful: static string
SCmsg_26	deletechatgroup: static string failed: static string

#### 5.1.2.2.10 Module 2.2.10 Server Site Chat Group Update

*Identification:* Module 2.2.10 Server\_Site\_Chat\_Group\_Update

*Function:* This module shall receive the inputs (CCmsg\_12) from the module 1.2.10. The User Information File shall be modified by reformatting User\_id\_list according to the specification given in subsection 3.3.1. If the operation succeeds SCmsg\_23, else SCmsg\_24 shall be sent to the module 1.2.10.

*Interface:*



Code	Use	Predecessor	Successor
CCmsg_12	To get the necessary inputs (specified in format of CCmsg_12) for update of an existing chat group from the root and send them to server software.	None	SCmsg_23, SCmsg_24
SCmsg_23	To indicate the successful update of the chat group.	CCmsg_12	None
SCmsg_24	To indicate the failed update of the chat group.	CCmsg_12	None

Code	Format
CCmsg_12	updatechatgroup:chat_group:user_id_list:
SCmsg_23	updatechatgroup:successful:
SCmsg_24	updatechatgroup:failed:

Code	Format Detail
CCmsg_12	updatechatgroup: static string user_id: semicolon seperated user_id list
SCmsg_23	updatechatgroup: static string successful: static string
SCmsg_24	updatechatgroup: static string failed: static string

### 5.1.2.3 Module 2.3 Server Site Chat

*Identification:* Module 2.3 Server\_Site\_Chat

*Function:* This module shall receive following four the inputs

- CCmsg\_14 form the module 1.3, If the operation succeeds SCmsg\_27 shall be sent to the module 1.3.
- CCmsg\_15 form the module 1.3, If the operation succeeds SCmsg\_28 shall be sent to the module 1.3.
- CCmsg\_16, form the module 1.3, If the operation succeeds SCmsg\_29 and SCmsg\_30 or SCmsg\_31shall be sent to the module 1.3.
- CCmsg\_17 form the module 1.3, If the operation succeeds SCmsg\_32 or SCmsg\_33 shall be sent to the module 1.3.

*Interface:*

Code	Use	Predecessor	Successor
CCmsg_14	To get the message from the user and send it to the server software	None	SCmsg_27
SCmsg_27	To get the message from the client software and send it to the all client software which are in the same room.	CCmsg_14	None
CCmsg_15	To query the room list of ION-CHAT SYSTEM	None	SCmsg_28
SCmsg_28	To send the room_list to the client software.	CCmsg_15	None
CCmsg_16	To query the user list of ION-CHAT SYSTEM for a specific room name.	None	SCmsg_29, SCmsg_31
SCmsg_29	To send the user_list of a specific room to the client software.	CCmsg_16	None
SCmsg_30	To send the active user_list of a specific room to the client software when a new user enter to the room.	None	None
SCmsg_31	To indicate the failed query of the user_list of a specific room.	CCmsg_16	None
CCmsg_17	To send user_id of the user to the server software to logout the user.	None	SCmsg_32, SCmsg_33
SCmsg_32	To indicate the successful logout of the user from the ION-CHAT SYSTEM	CCmsg_17	None
SCmsg_33	To indicate the failed logout of the user from the ION-CHAT SYSTEM	CCmsg_17	None

Code	Format
CCmsg_14	chatmsg:chat_group:real_name:chat_message
SCmsg_27	chatmsg:chat_group:real_name:chat_message
CCmsg_15	askroomlist:
SCmsg_28	askroomlist:room_name_list:
CCmsg_16	askalluserlist:room_name:
SCmsg_29	askalluserlist:user_id/chat_group/real_name:user_id/ chat_group/real_name:user_id/customer/real_name: user_id/instructor/real_name:
SCmsg_30	askactiveuserlist:user_id_list
SCmsg_31	askalluserlist:failed:

CCmsg_17	logout:user_id:
SCmsg_32	logout:successful:
SCmsg_33	logout:failed:

Code	Format Detail
CCmsg_14	chatmsg: static string
SCmsg_27	chatmsg: static string
CCmsg_15	askroomlist: static string
SCmsg_28	askroomlist: static string room_name_list: semicolon seperated room_name list
CCmsg_16	askalluserlist: static string
SCmsg_29	askalluserlist: static string user_id/chat_group/real_name: semicolon seperated list for reviewie group members user_id/chat_group/real_name: semicolon seperated list for reviewier group members user_id/customer/real_name: for the customer assigned to the room user_id/instructor/real_name: semicolon seperated list for all instructors
SCmsg_30	askactiveuserlist: static string user_id: semicolon seperated user_id list
SCmsg_31	askalluserlist: static string failed: static string
CCmsg_17	logout: static string
SCmsg_32	logout: static string successful: static string
SCmsg_33	logout: static string failed: static string

## **6 DETAILED DESIGN**

A list of error messages [Emsg\_\*], success messages [Smsg\_\*] and communication messages [CCmsg\_\* and SCmsg\_\*] used on the DFD's can be seen in Appendix-A.

The description of common data elements used both in communication messages and in algorithms of modules are given in Appendix-D.

A list of the methods used on the flowcharts' can be seen in Appendix-I.

### **6.1 Module Detailed Design**

#### **6.1.1 Module 1 Chat Client**

##### **6.1.1.1 Module 1.1 Client Site User Authentication**

*Identification:* Module 1.1 Client\_Site\_User\_Authentication

*Processing:* The internal details of this module is given in Figure-27 in Appendix-F.

##### **6.1.1.2 Module 1.2 Client Site Root Functions**

###### **6.1.1.2.1 Module 1.2.1 Client Site Root Console**

*Identification:* Module 1.2.1 Client\_Site\_Root\_Console

*Processing:* The internal details of this module is given in Figure-49 in Appendix-F.

###### **6.1.1.2.2 Module 1.2.2 Client Site User Addition**

*Identification:* Module 1.2.2 Client\_Site\_User\_Addition

*Processing:* The internal details of this module is given in Figure-28 in Appendix-F.

###### **6.1.1.2.3 Module 1.2.3 Client Site User Deletion**

*Identification:* Module 1.2.3 Client\_Site\_User\_Deletion

*Processing:* The internal details of this module is given in Figure-29 in Appendix-F.

#### **6.1.1.2.4 Module 1.2.4 Client Site User Update**

*Identification:* Module 1.2.4 Client\_Site\_User\_Update

*Processing:* The internal details of this module is given in Figure-30 in Appendix-F.

#### **6.1.1.2.5 Module 1.2.5 Client Site Room Addition**

*Identification:* Module 1.2.5 Client\_Site\_Room\_Addition

*Processing:* The internal details of this module is given in Figure-31 in Appendix-F.

#### **6.1.1.2.6 Module 1.2.6 Client Site Room Deletion**

*Identification:* Module 1.2.6 Client\_Site\_Room\_Deletion

*Processing:* The internal details of this module is given in Figure-32 in Appendix-F.

#### **6.1.1.2.7 Module 1.2.7 Client Site Room Update**

*Identification:* Module 1.2.7 Client\_Site\_Room\_Update

*Processing:* The internal details of this module is given in Figure-33 in Appendix-F.

#### **6.1.1.2.8 Module 1.2.8 Client Site Chat Group Addition**

*Identification:* Module 1.2.8 Client\_Site\_Chat\_Group\_Addition

*Processing:* The internal details of this module is given in Figure-34 in Appendix-F.

#### **6.1.1.2.9 Module 1.2.9 Client Site Chat Group Deletion**

*Identification:* Module 1.2.9 Client\_Site\_Chat\_Group\_Deletion

*Processing:* The internal details of this module is given in Figure-35 in Appendix-F.

#### **6.1.1.2.10 Module 1.2.10 Client Site Chat Group Update**

*Identification:* Module 1.2.10 Client\_Site\_Chat\_Group\_Update

*Processing:* The internal details of this module is given in Figure-36 in Appendix-F.

#### **6.1.1.3 Module 1.3 Client Site Chat**

*Identification:* Module 1.3 Client\_Site\_Chat

*Processing:* The internal details of this module is given in Figure-37 in Appendix-F.

### **6.1.2 Module 2 Chat Server**

#### **6.1.2.1 Module 2.1 Server Site User Authentication**

*Identification:* Module 2.1 Server\_Site\_User\_Authentication

*Processing:* The internal details of this module is given in Figure-38 in Appendix-F.

#### **6.1.2.2 Module 2.2 Server Site Root Functions**

##### **6.1.2.2.1 Module 2.2.1 Server Site Root Console**

*Identification:* Module 2.2.1 Server\_Site\_Root\_Console

*Processing:* The internal details of this module is given in Figure-50 in Appendix-F.

##### **6.1.2.2.2 Module 2.2.2 Server Site User Addition**

*Identification:* Module 2.2.2 Server\_Site\_User\_Addition

*Processing:* The internal details of this module is given in Figure-39 in Appendix-F.

#### **6.1.2.2.3 Module 2.2.3 Server Site User Deletion**

*Identification:* Module 2.2.3 Server\_Site\_User\_Deletion

*Processing:* The internal details of this module is given in Figure-40 in Appendix-F.

#### **6.1.2.2.4 Module 2.2.4 Server Site User Update**

*Identification:* Module 2.2.4 Server\_Site\_User\_Update

*Processing:* The internal details of this module is given in Figure-41 in Appendix-F.

#### **6.1.2.2.5 Module 2.2.5 Server Site Room Addition**

*Identification:* Module 2.2.5 Server\_Site\_Room\_Addition

*Processing:* The internal details of this module is given in Figure-42 in Appendix-F.

#### **6.1.2.2.6 Module 2.2.6 Server Site Room Deletion**

*Identification:* Module 2.2.6 Server\_Site\_Room\_Deletion

*Processing:* The internal details of this module is given in Figure-43 in Appendix-F.

#### **6.1.2.2.7 Module 2.2.7 Server Site Room Update**

*Identification:* Module 2.2.7 Server\_Site\_Room\_Update

*Processing:* The internal details of this module is given in Figure-44 in Appendix-F.

#### **6.1.2.2.8 Module 2.2.8 Server Site Chat Group Addition**

*Identification:* Module 2.2.8 Server\_Site\_Chat\_Group\_Addition

*Processing:* The internal details of this module is given in Figure-45 in Appendix-F.

#### **6.1.2.2.9 Module 2.2.9 Server Site Chat Group Deletion**

*Identification:* Module 2.2.9 Server\_Site\_Chat\_Group\_Deletion

*Processing:* The internal details of this module is given in Figure-46 in Appendix-F.

#### **6.1.2.2.10 Module 2.2.10 Server Site Chat Group Update**

*Identification:* Module 2.2.10 Server\_Site\_Chat\_Group\_Update

*Processing:* The internal details of this module is given in Figure-47 in Appendix-F.

#### **6.1.2.3 Module 2.3 Server Site Chat**

*Identification:* Module 2.3 Server\_Site\_Chat

*Processing:* The internal details of this module is given in Figure-48 in Appendix-F.

### **6.2 Data Detailed Design**

#### **6.2.1 Data Entity 1 User Information File**

*Identification:* Data Entity 1 User\_Information\_File

*Data:* The data must be represented by the following way;

room\_name:reviewie\_group:reviewer\_group:customer\_user\_id:

The detailed description of data elements used in this data entity is given in Appendix-D.

#### **6.2.2 Data Entity 2 Room Information File**

*Identification:* Data Entity 2 Room\_Information\_File

*Data:* The data must be represented by the following way;

room\_name:reviewie\_group:reviewer\_group:customer\_user\_id:

The detailed description of data elements used in this data entity is given in Appendix-D.



## **APPENDIX-A**

### **Error messages:**

1. Emsg\_1: "Error: Invalid range or invalid data format for input(s)"
2. Emsg\_2: "Error: User deletion failed"
3. Emsg\_3: "Error: User update failed"
4. Emsg\_4: "Error: User addition failed"
5. Emsg\_5: "Error: Room deletion failed"
6. Emsg\_6: "Error: Room update failed"
7. Emsg\_7: "Error: Room addition failed"
8. Emsg\_8: "Error: Chat\_Group deletion failed"
9. Emsg\_9: "Error: Chat\_Group update failed"
10. Emsg\_10: "Error: Chat\_Group addition failed"
11. Emsg\_11: "Error: Login failed"
12. Emsg\_12: "Error: Logout failed"
13. Emsg\_13: "Error: Existing user"
14. Emsg\_14: "Error: Non-existing user"
15. Emsg\_15: "Error: Existing room"
16. Emsg\_16: "Error: Non-existing room"
17. Emsg\_17: "Error: Existing chat\_group"
18. Emsg\_18: "Error: Non-existing chat\_group"

### **Success messages:**

1. Smsg\_1: "User deletion successful"
2. Smsg\_2: "User update successful"
3. Smsg\_3: "User addition successful"
4. Smsg\_4: "Room deletion successful"
5. Smsg\_5: "Room update successful"
6. Smsg\_6: "Room addition successful"

7. Smsg\_7: “Chat\_Group deletion successful”
8. Smsg\_8: “Chat\_Group update successful”
9. Smsg\_9: “Chat\_Group addition successful”
10. Smsg\_10: “Logout successful”

### Communication Messages:

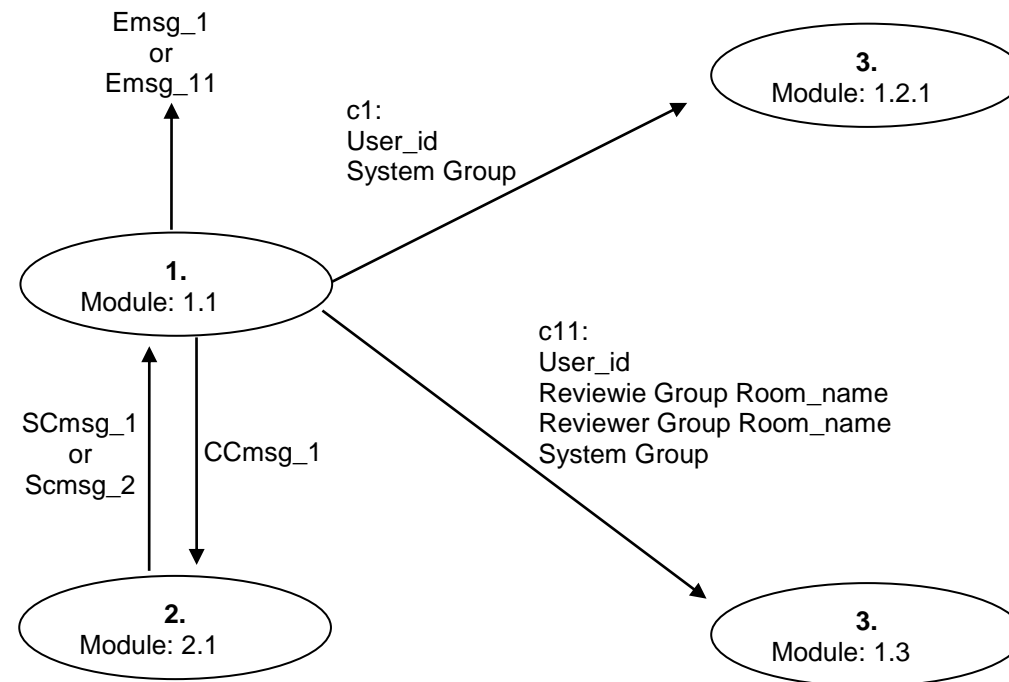
Code	Format
CCmsg_1	login:user_id:password:
SCmsg_1a	login:root:
SCmsg_1b	login:instructor:
SCmsg_1c	login:customer:room_name:
SCmsg_1d	login:student:room_name_as_reviewie_group: room_name_as_reviewer_group:
SCmsg_2	login:failed:
CCmsg_2	askuser:user_id:
SCmsg_3	askuser:user_id:password:real_name:system_group:
SCmsg_4	askuser:failed:
CCmsg_3	askroom:room_name:
SCmsg_5	askroom:room_name:reviewie_group:reviewer_group: customer_user_id:
SCmsg_6	askroom:failed:
CCmsg_4	askchatgroup:chat_group:
SCmsg_7	askchatgroup:chat_group:user_id_list:
SCmsg_8	askchatgroup:failed:
CCmsg_5	adduser:user_id:password:real_name:system_group:
SCmsg_9	adduser:successful:
SCmsg_10	adduser:failed:
CCmsg_6	updateuser:user_id:password:real_name:system_group:
SCmsg_11	updateuser:successful:

SCmsg_12	updateuser:failed:
CCmsg_7	deleteuser:user_id:
SCmsg_13	deleteuser:successful:
SCmsg_14	deleteuser:failed:
CCmsg_8	addroom:room_name:reviewie_group:reviewer_group: customer_user_id:
SCmsg_15	addroom:successful:
SCmsg_16	addroom:failed:
CCmsg_9	updateroom:room_name:reviewie_group: reviewer_group:customer_user_id:
SCmsg_17	updateroom:successful:
SCmsg_18	updateroom:failed:
CCmsg_10	deleteroom:room_name:
SCmsg_19	deleteroom:successful:
SCmsg_20	deleteroom:failed:
CCmsg_11	addchatgroup:chat_group:user_id_list:
SCmsg_21	addchatgroup:successful:
SCmsg_22	addchatgroup:failed:
CCmsg_12	updatechatgroup:chat_group:user_id_list:
SCmsg_23	updatechatgroup:successful:
SCmsg_24	updatechatgroup:failed:
CCmsg_13	deletechatgroup:chat_group:
SCmsg_25	deletechatgroup:successful:
SCmsg_26	deletechatgroup:failed:
CCmsg_14	chatmsg:chat_group:real_name:chat_message
SCmsg_27	chatmsg:chat_group:real_name:chat_message
CCmsg_15	askroomlist:
SCmsg_28	askroomlist:room_name_list:
CCmsg_16	askalluserlist:room_name:
SCmsg_29	askalluserlist:user_id/chat_group/real_name:user_id/

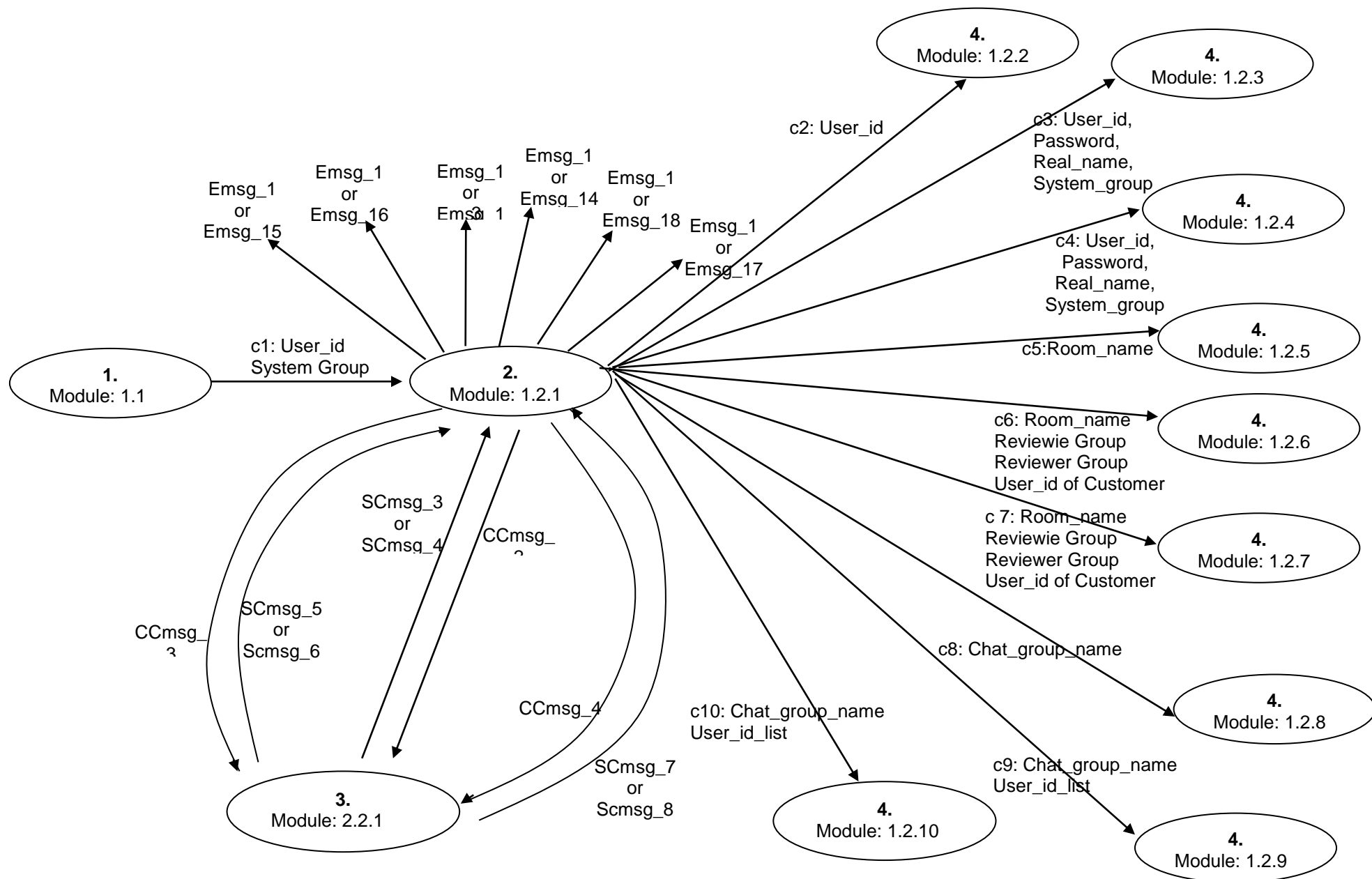
	chat_group/real_name:user_id/customer/real_name: user_id/instructor/real_name:
SCmsg_30	askactiveuserlist:user_id_list
SCmsg_31	askalluserlist:failed:
CCmsg_17	logout:user_id:
SCmsg_32	logout:successful:
SCmsg_33	logout:failed:

## APPENDIX-B

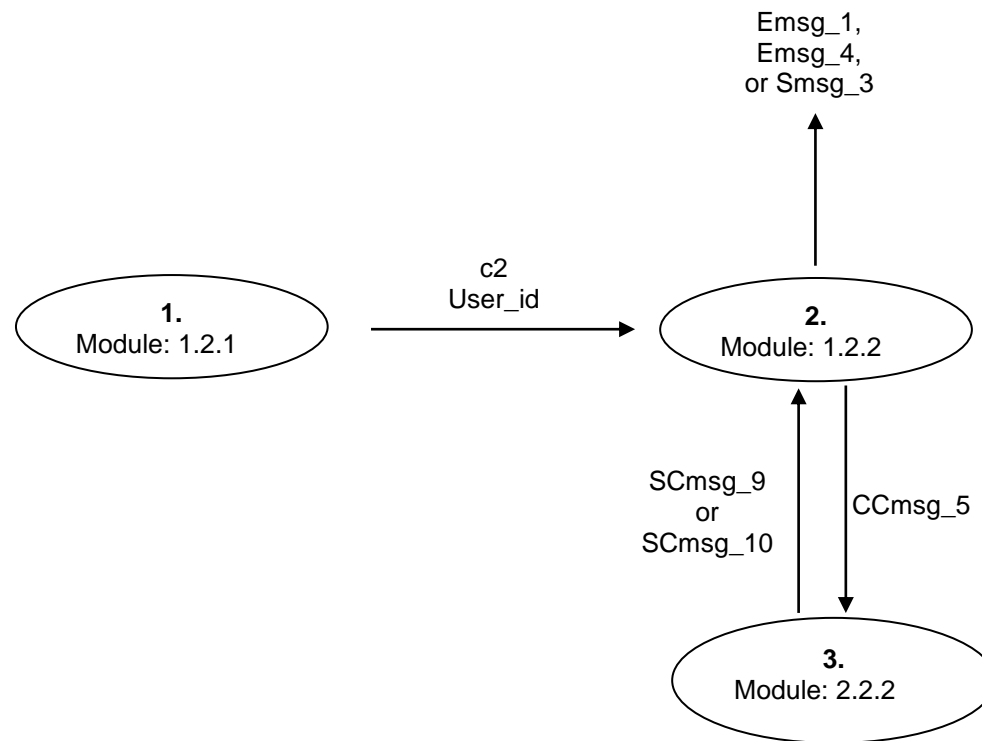
### Intermodule Dependencies:



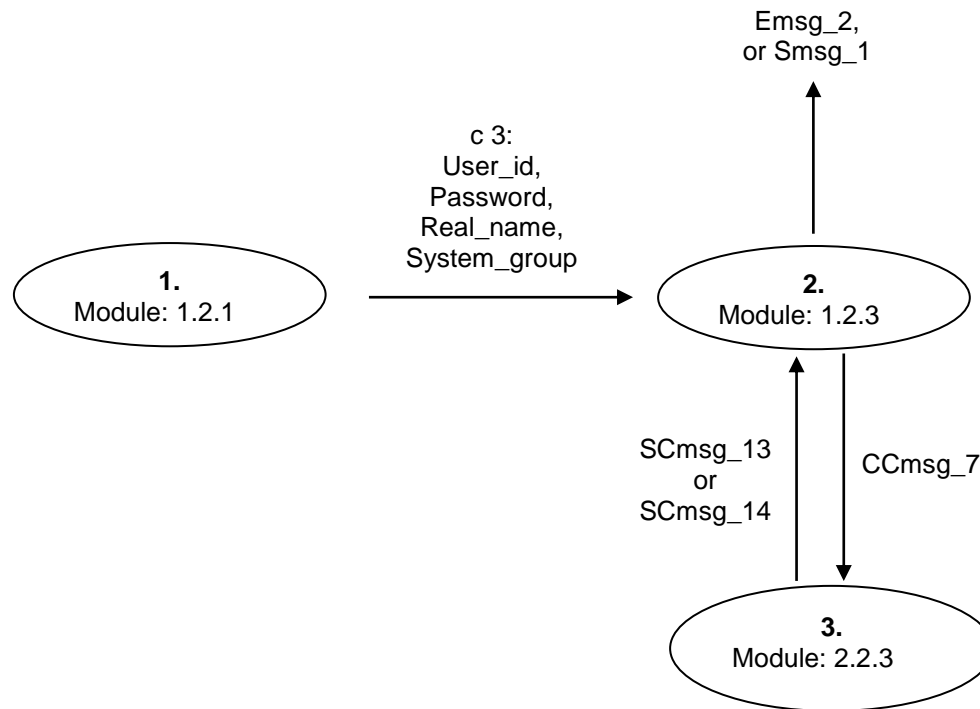
**Figure 1:** Module 1.1 Client\_Site\_User\_Authentication



**Figure 2:** Module 1.2.1 Client\_Site\_Root\_Console

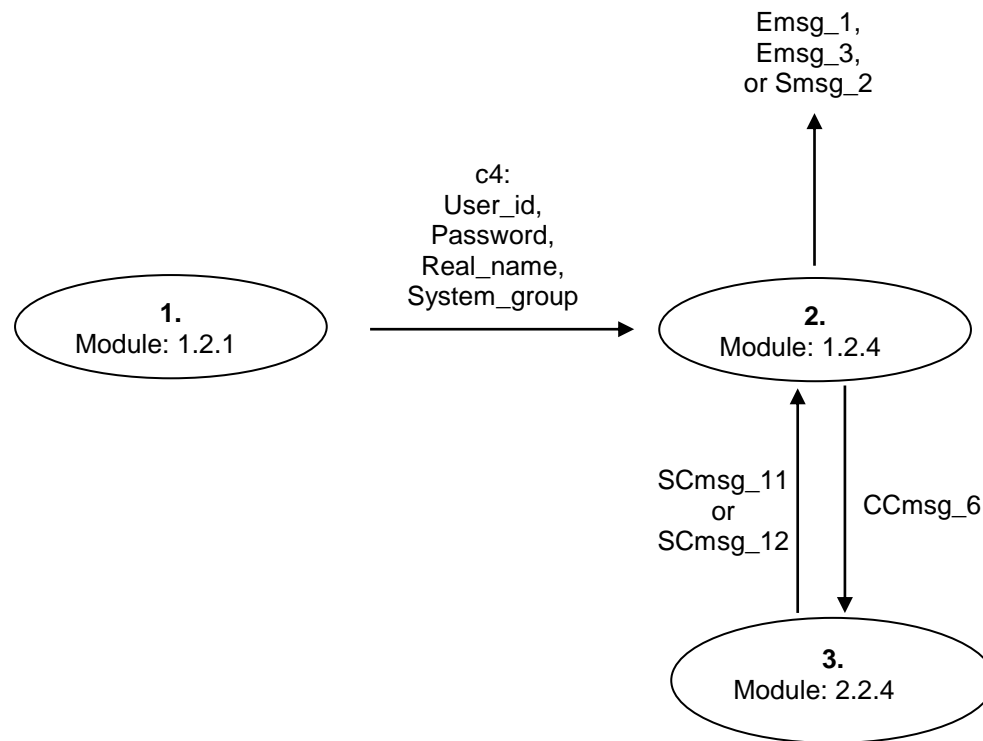


**Figure 3:** Module 1.2.2 Client\_Site\_User\_Addition

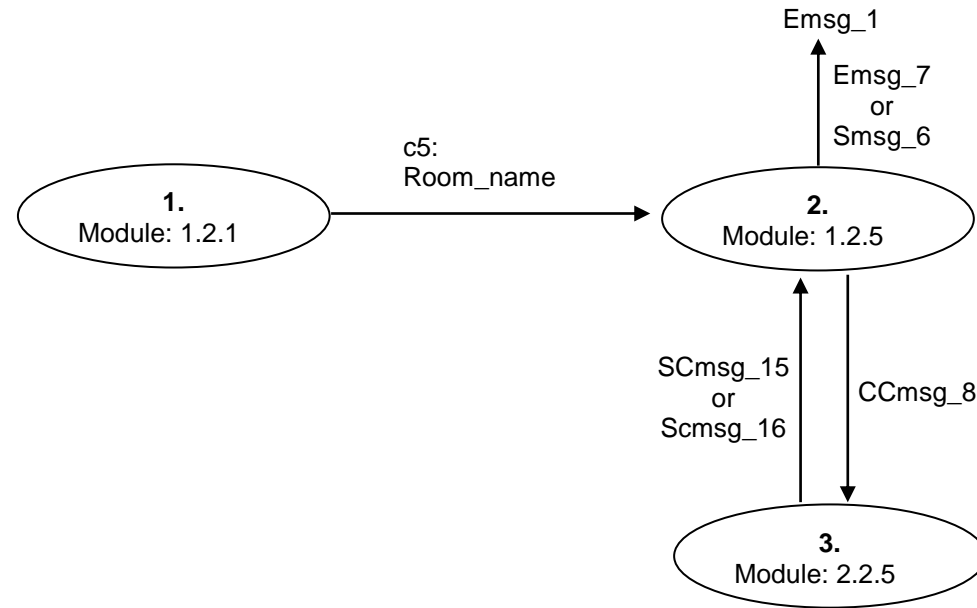


**Figure 4:** Module 1.2.3 Client\_Site\_User\_Deletion

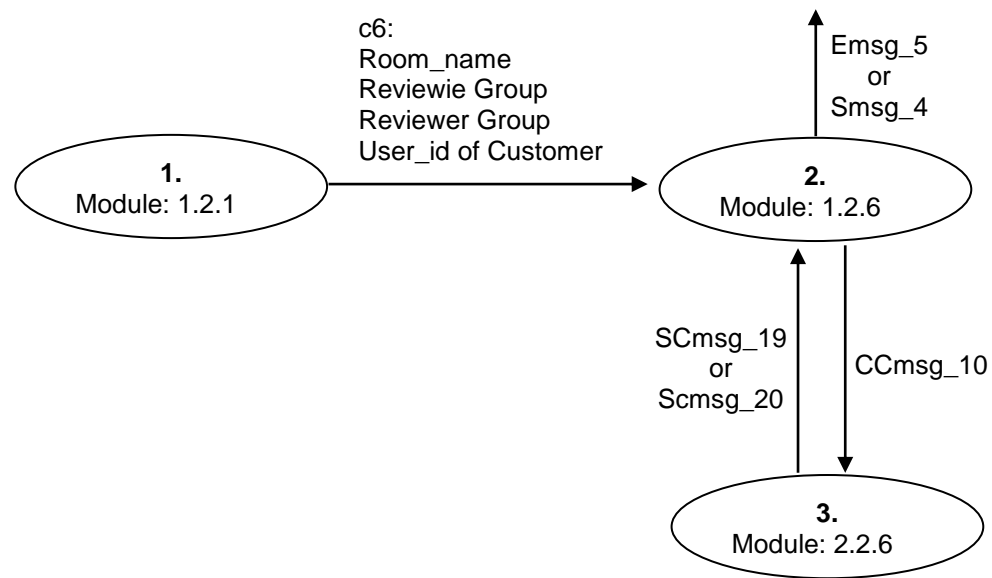




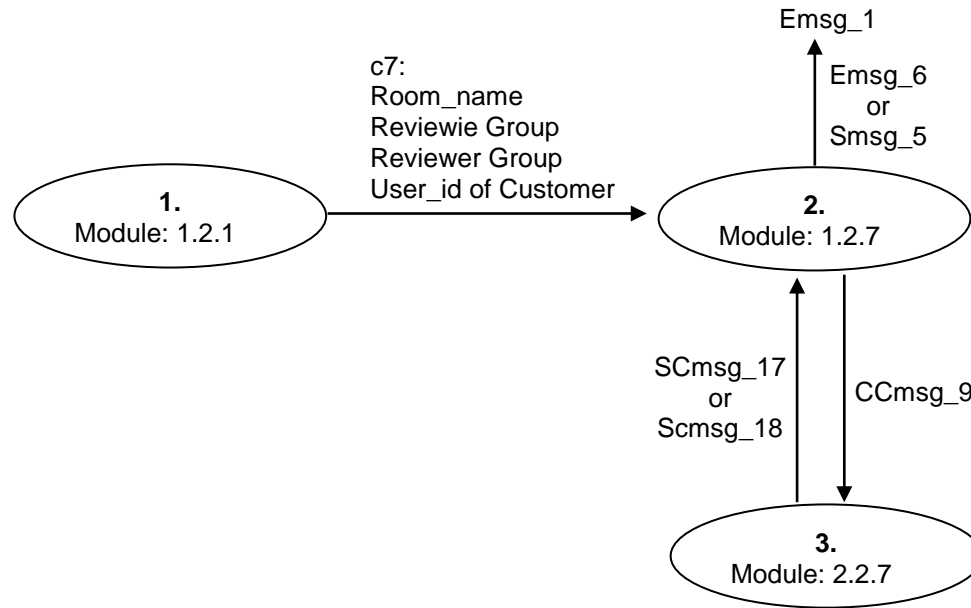
**Figure 5:** Module 1.2.4 Client\_Site\_User\_Update



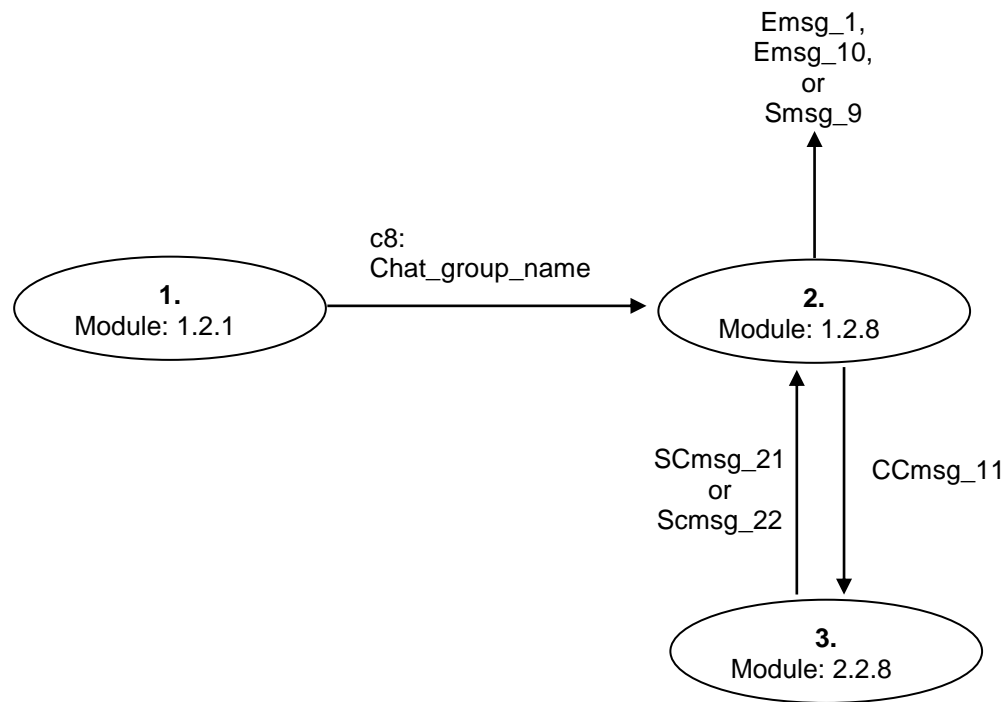
**Figure 6:** Module 1.2.5 Client\_Site\_Room\_Addition



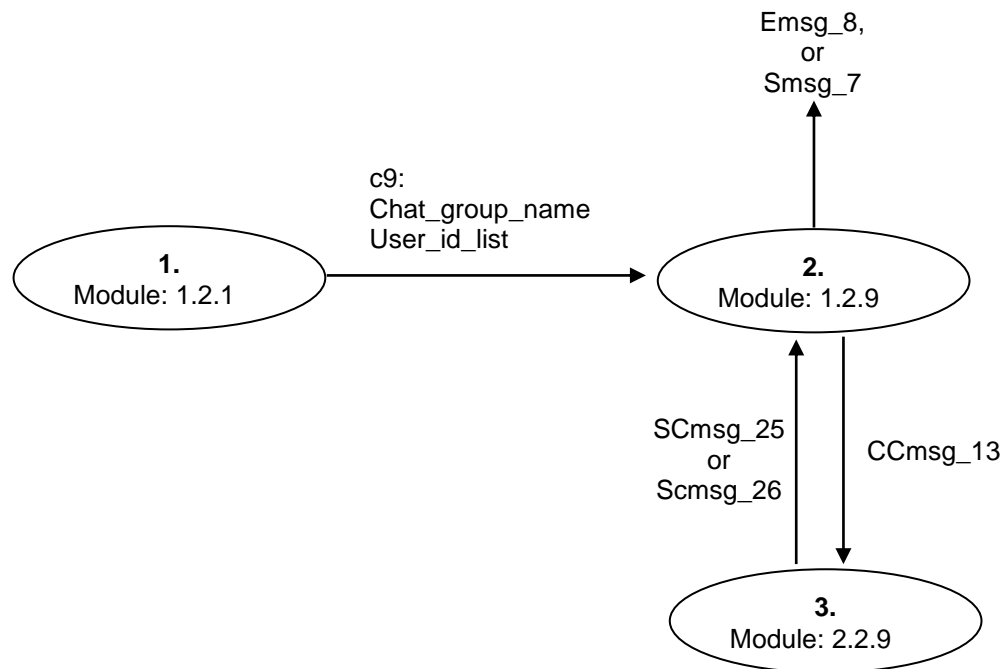
**Figure 7:** Module 1.2.6 Client\_Site\_Room\_Deletion



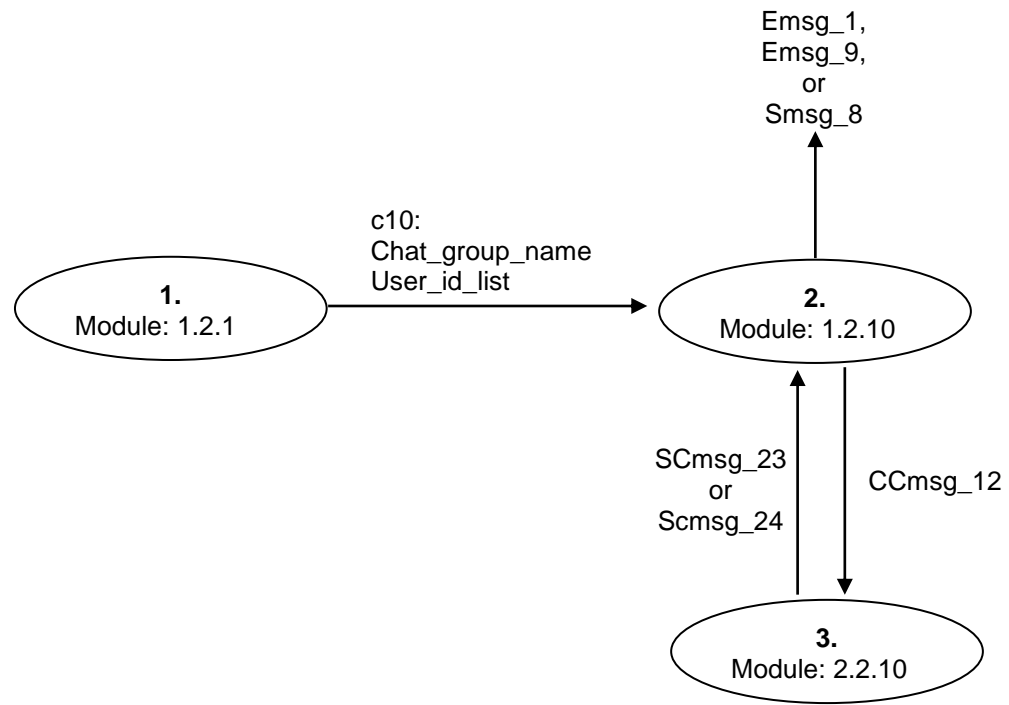
**Figure 8:** Module 1.2.7 Client\_Site\_Room\_Update



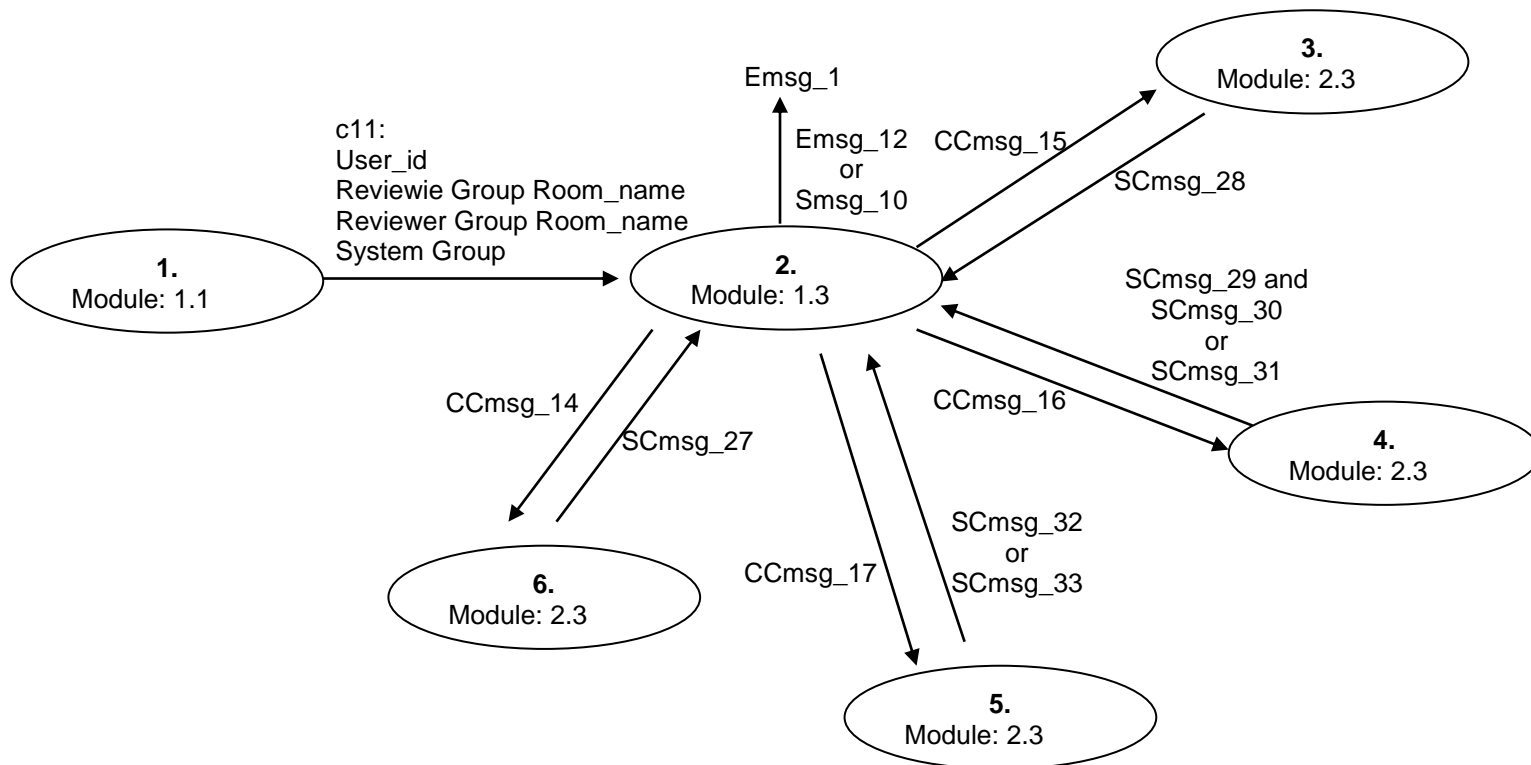
**Figure 9:** Module 1.2.8 Client\_Site\_Chat\_Group\_Addition



**Figure 10:** Module 1.2.9 Client\_Site\_Chat\_Group\_Deletion

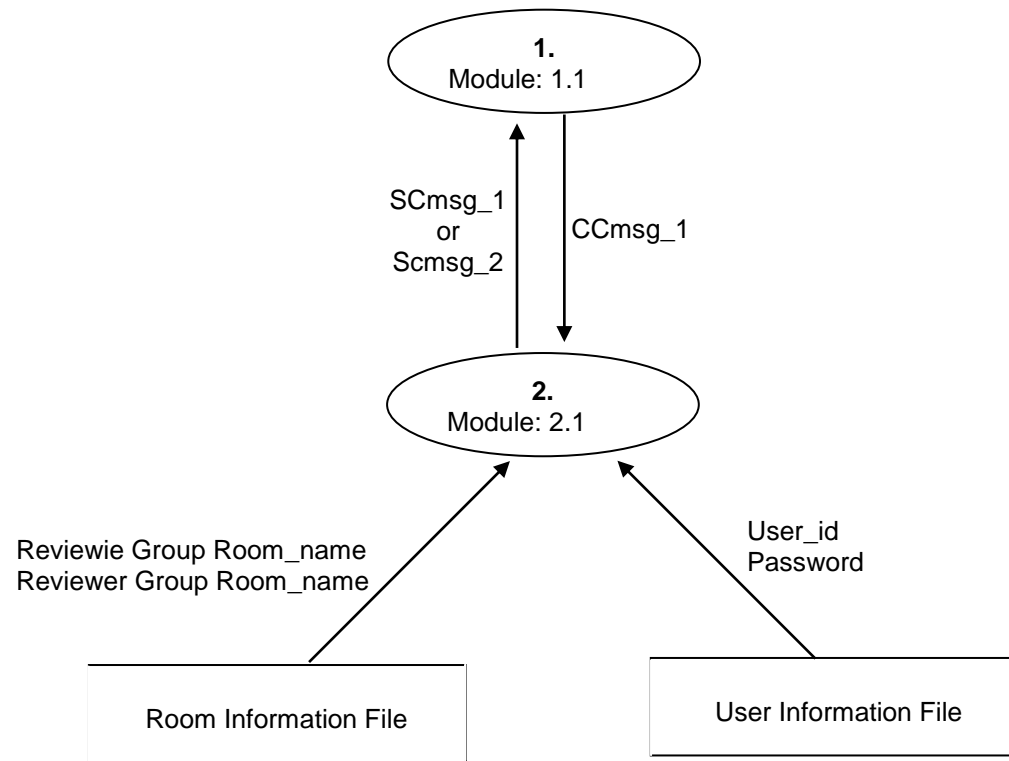


**Figure 11:** Module 1.2.10 Client\_Site\_Chat\_Group\_Update

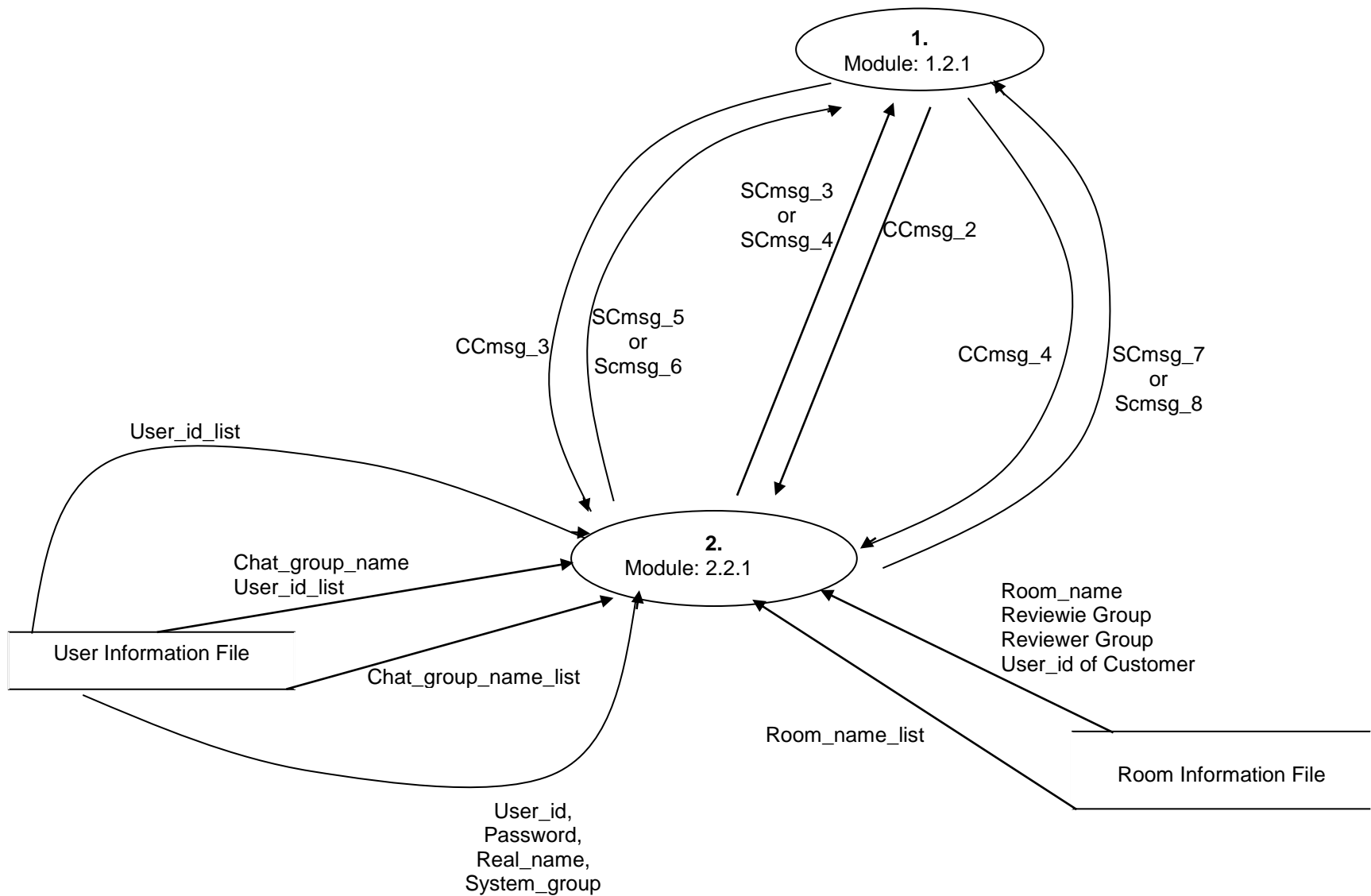


**Figure 12:** Module 1.3 Client\_Site\_Chat

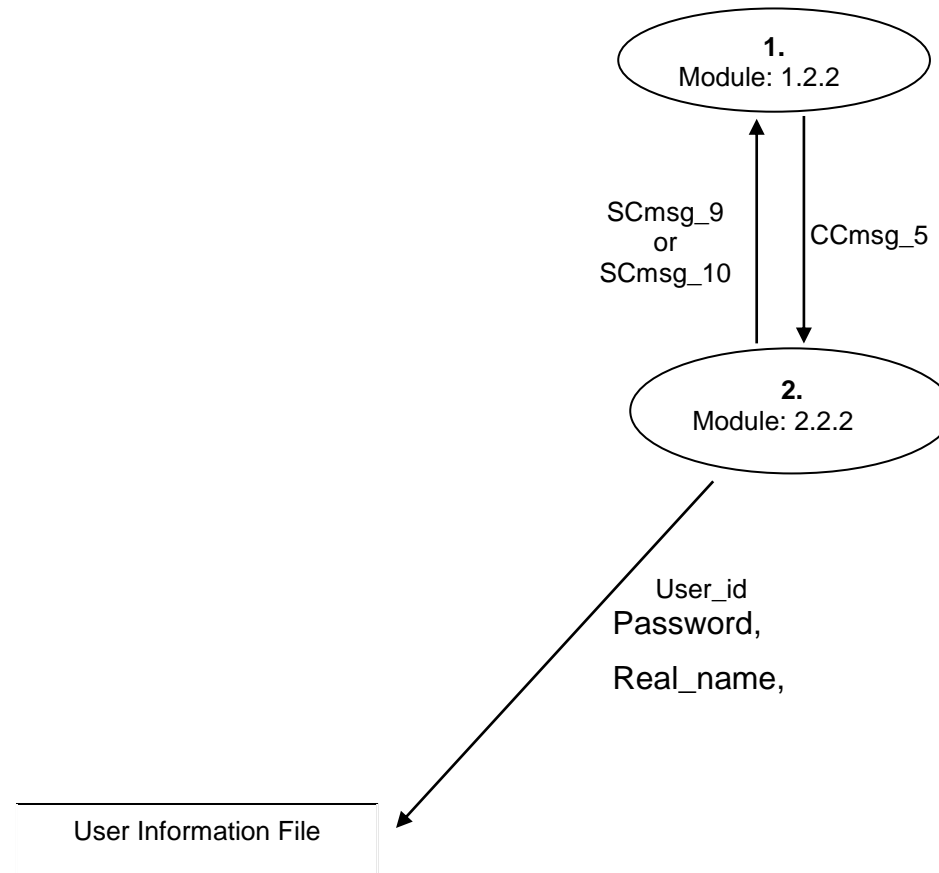




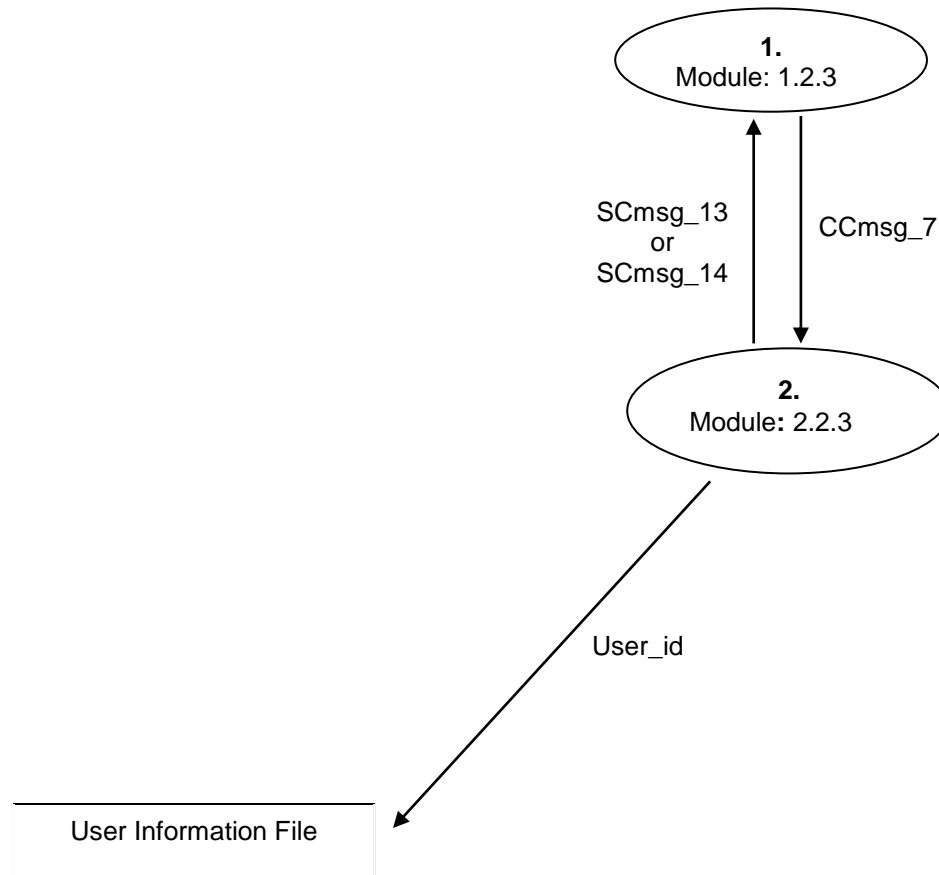
**Figure 13:** Module 2.1 Server\_Site\_User\_Authentication



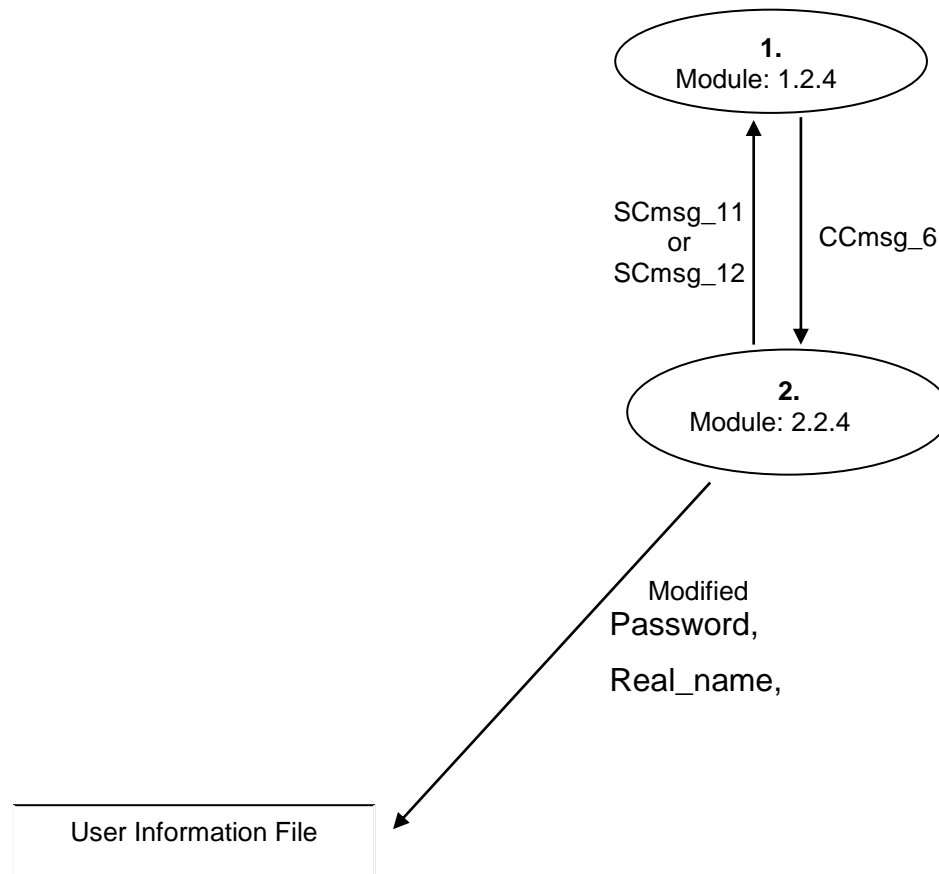
**Figure 14:** 14: Module 2.2.1 Server\_Site\_Root\_Console



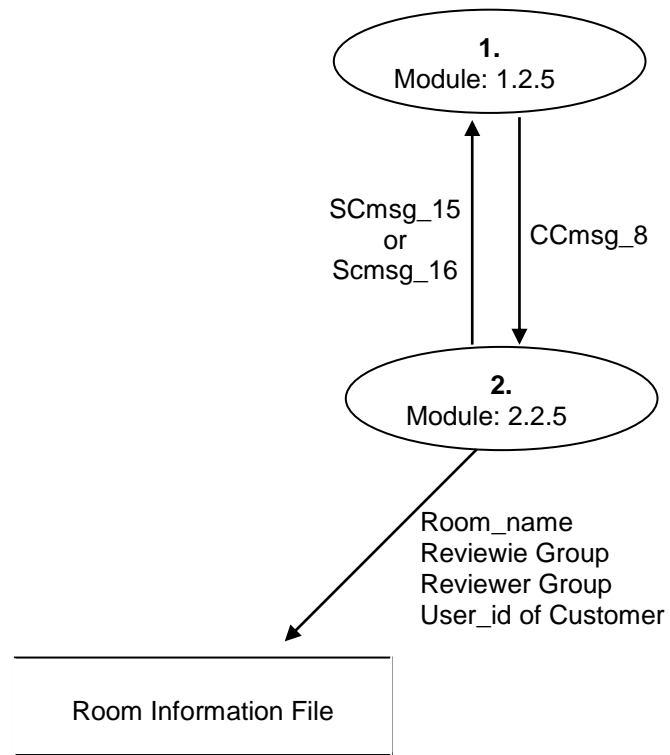
**Figure 15:** Module 2.2.2 Server\_Site\_User\_Addition



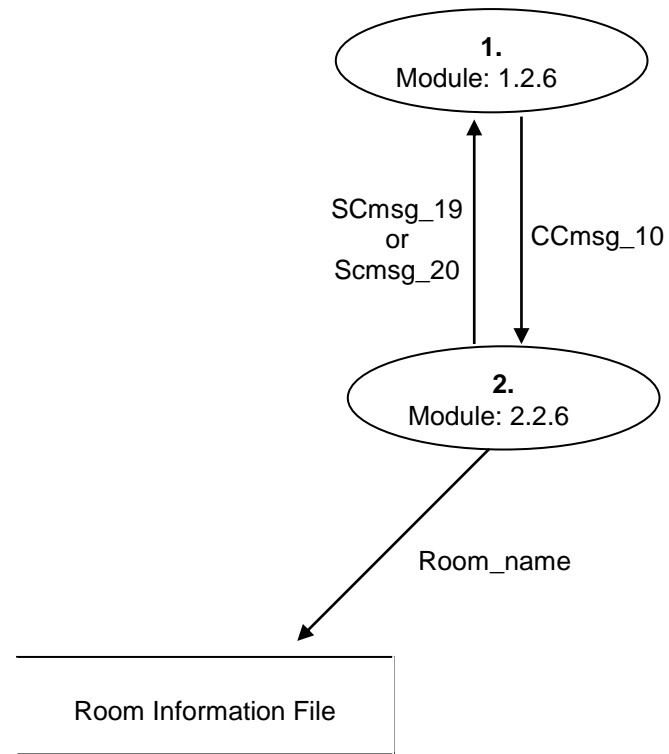
**Figure 16:** Module 2.2.3 Server\_Site\_User\_Deletion



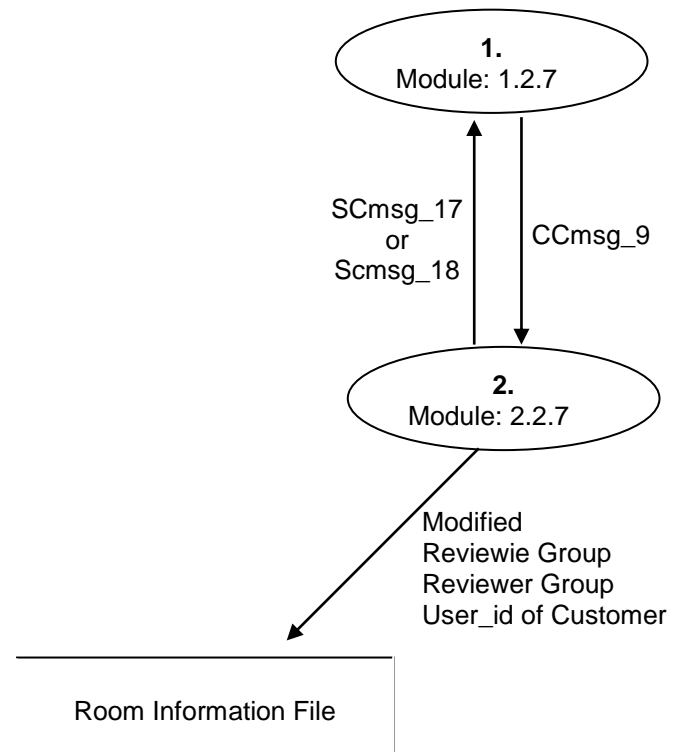
**Figure 17:** Module 2.2.4 Server\_Site\_User\_Update



**Figure 18:** Module 2.2.5 Server\_Site\_Room\_Addition

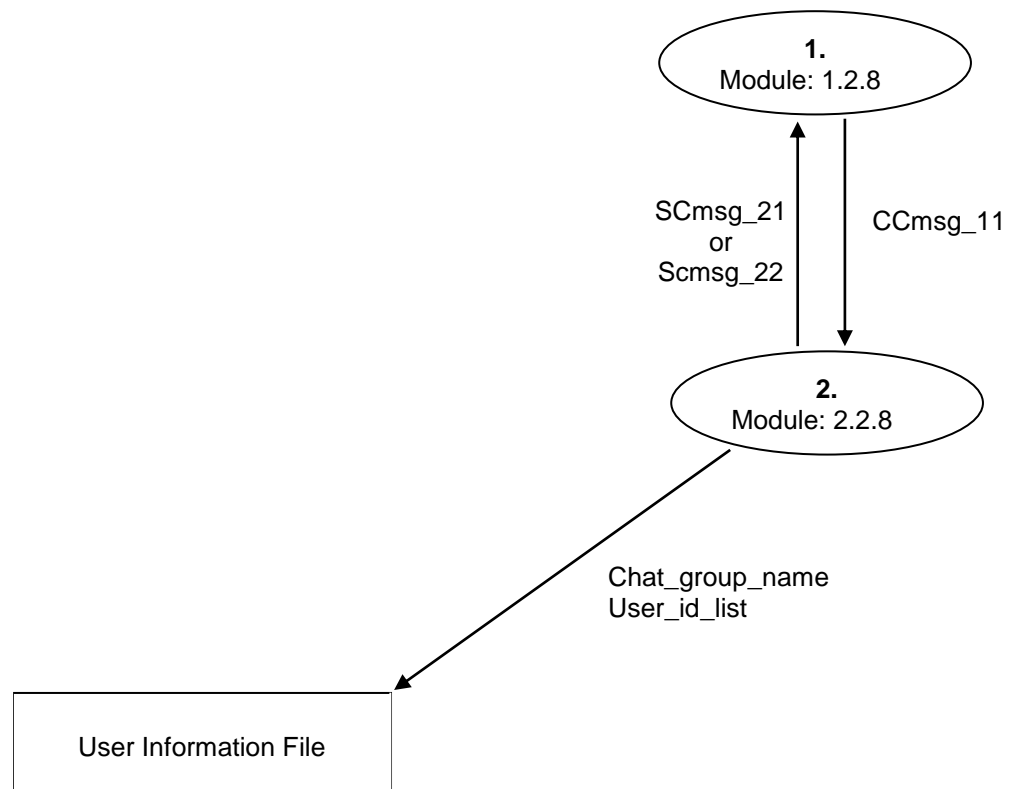


**Figure 19:** Module 2.2.6 Server\_Site\_Room\_Deletion

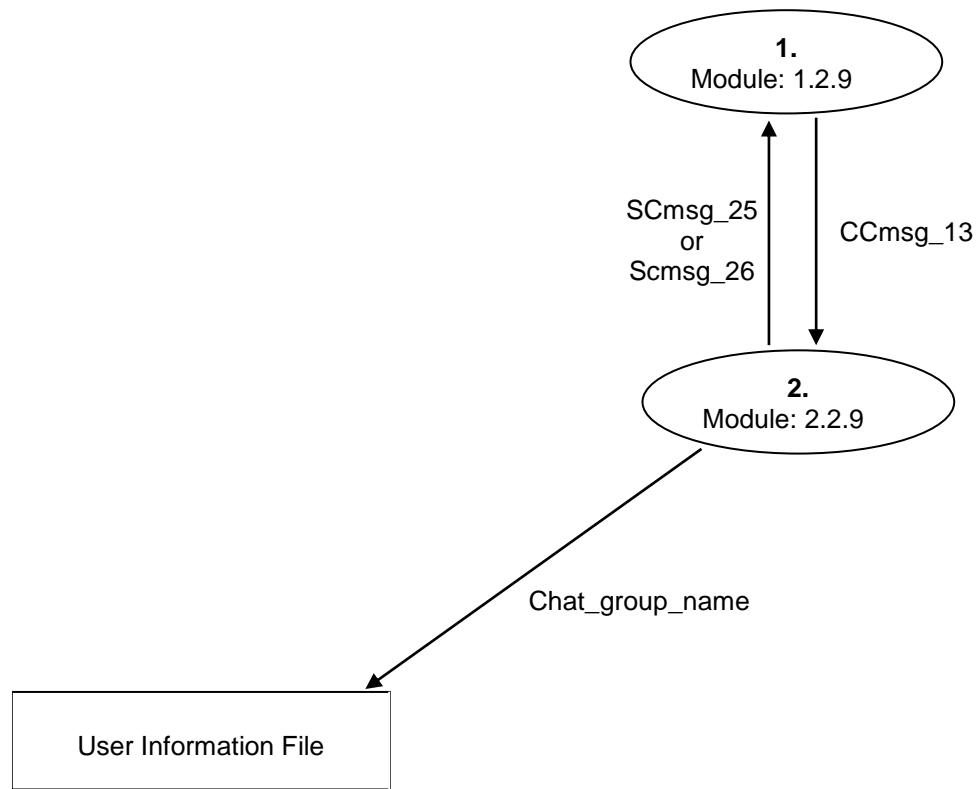


**Figure 20:** Module 2.2.7 Server\_Site\_Room\_Update

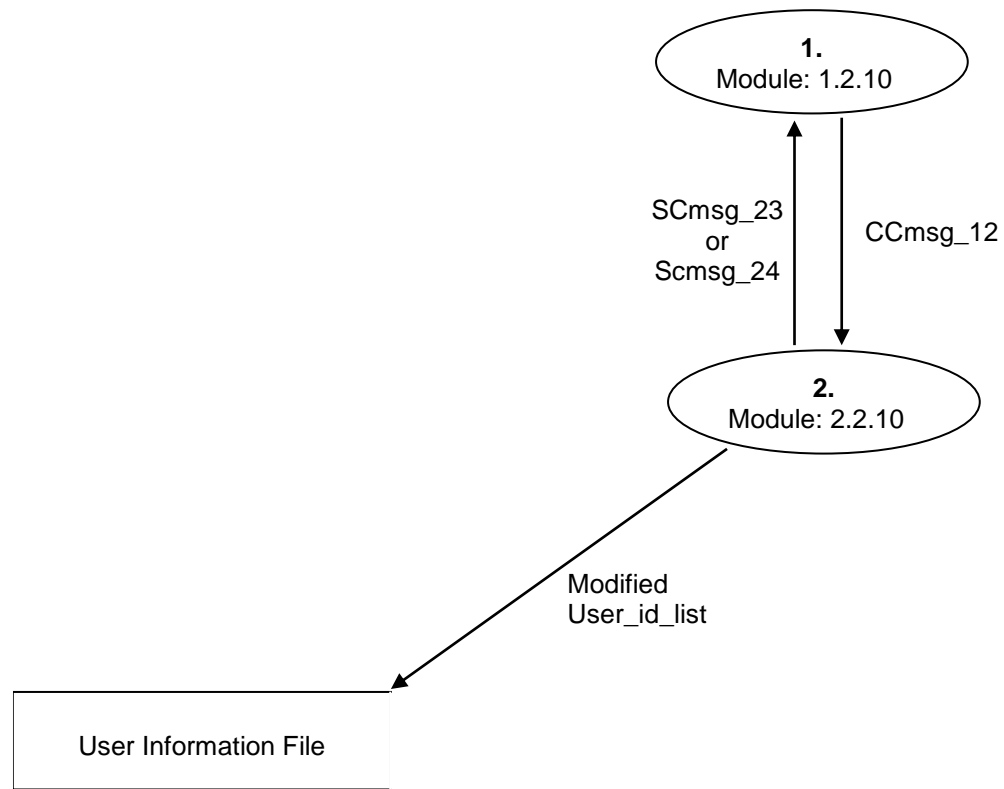




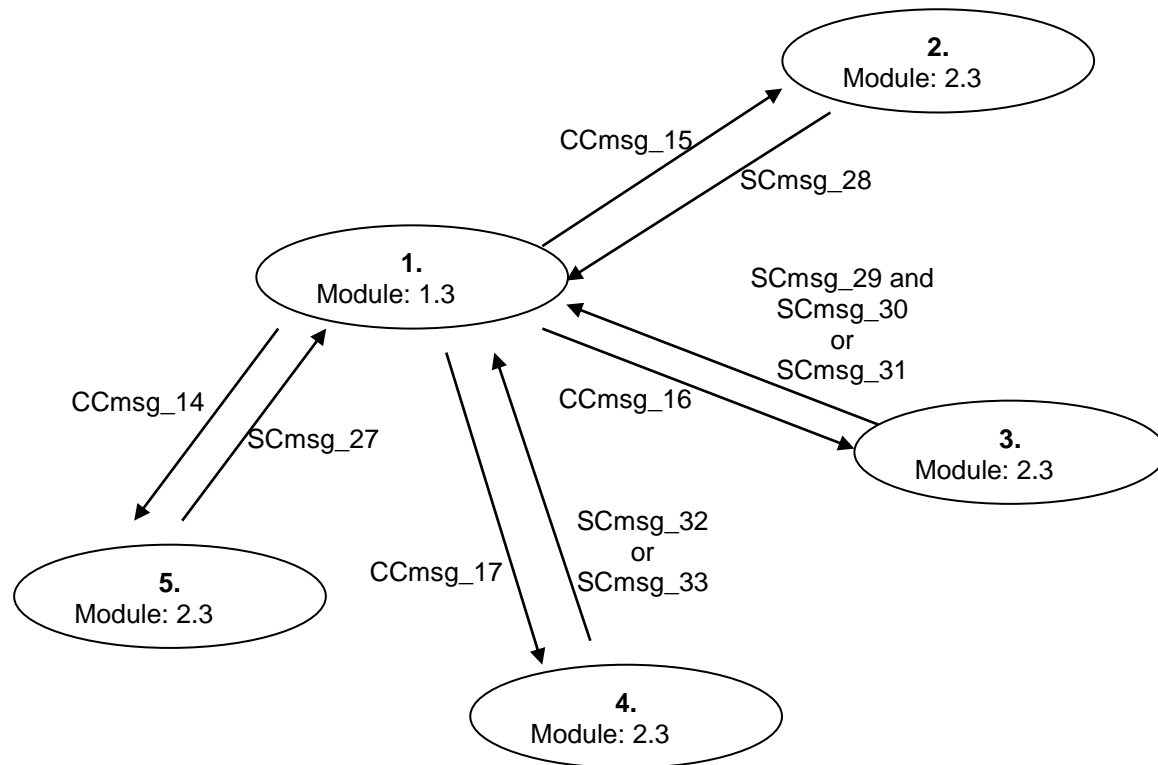
**Figure 21:** Module 2.2.8 Server\_Site\_Chat\_Group\_Addition



**Figure 22:** Module 2.2.9 Server\_Site\_Chat\_Group\_Deletion

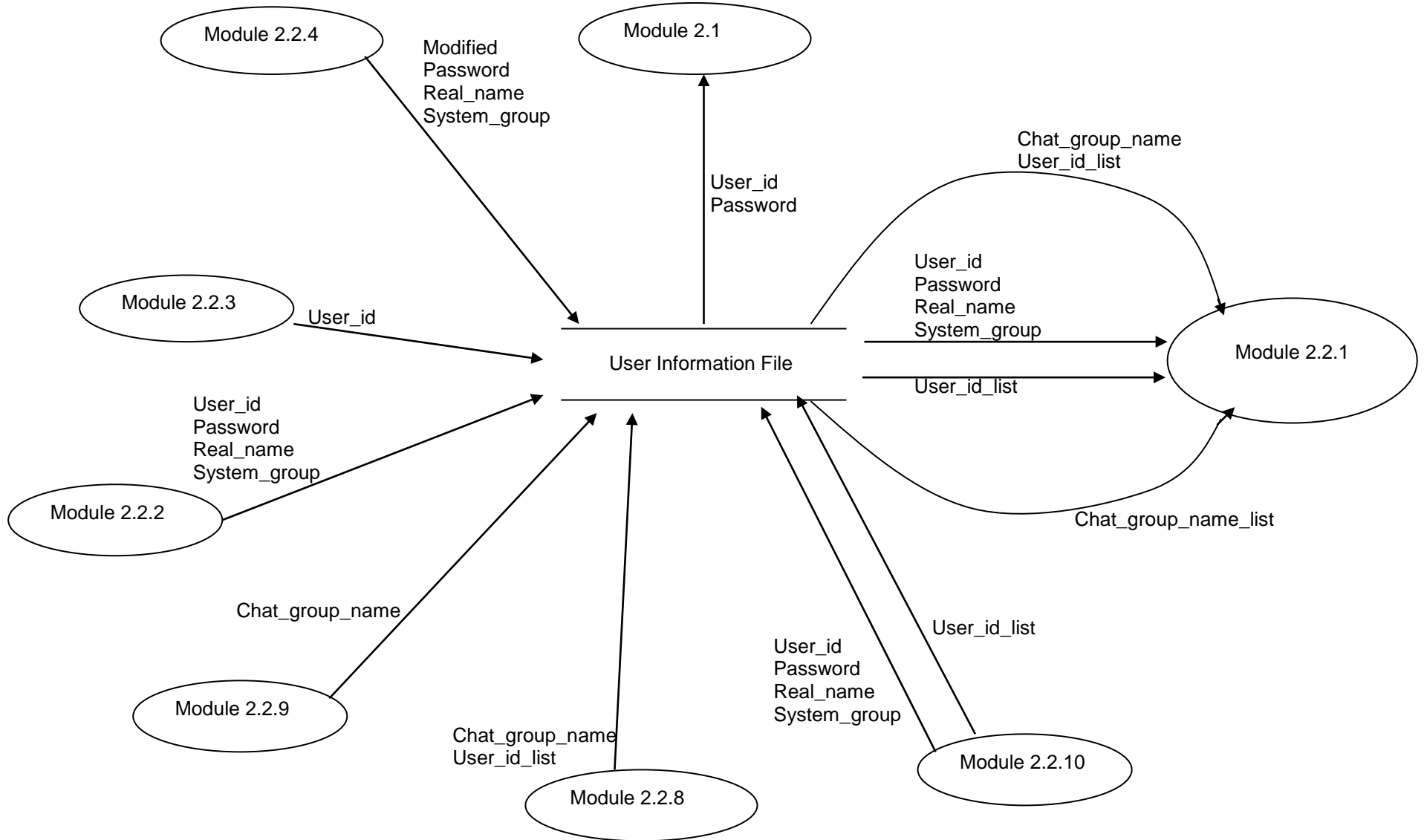


**Figure 23:** Module 2.2.10 Server\_Site\_Chat\_Group\_Update

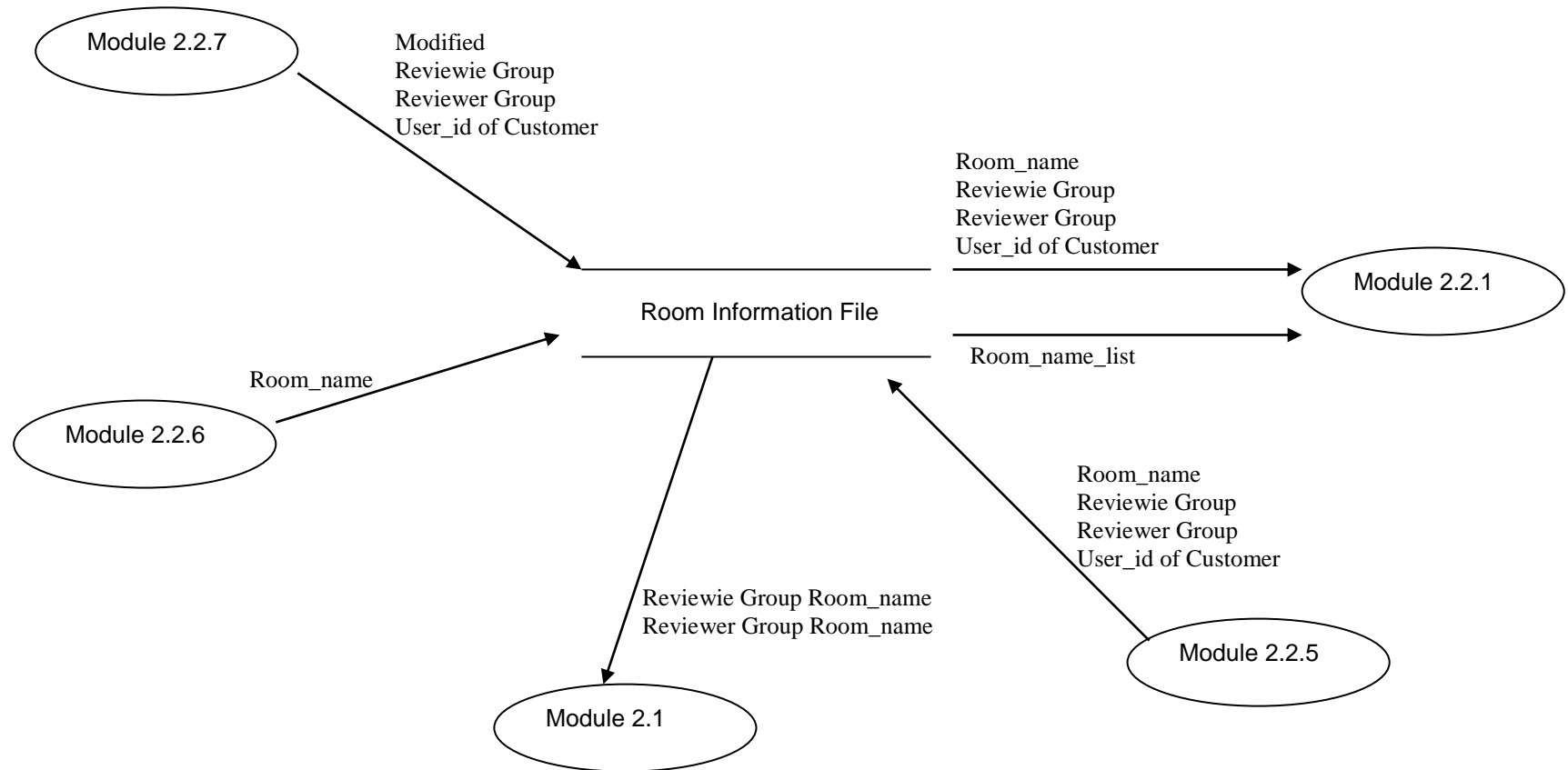


**Figure 24:** Module 2.3 Server\_Site\_Chat

### Data Dependencies:



**Figure 25:** User Information File



**Figure 26:** Room Information File

## **APPENDIX-C**

### **Module List:**

Module 1.1 Client Site User Authentication

Module 1.2.1 Client Site Root Console

Module 1.2.2 Client Site User Addition

Module 1.2.3 Client Site User Deletion

Module 1.2.4 Client Site User Update

Module 1.2.5 Client Site Room Addition

Module 1.2.6 Client Site Room Deletion

Module 1.2.7 Client Site Room Update

Module 1.2.8 Client Site Chat Group Addition

Module 1.2.9 Client Site Chat Group Deletion

Module 1.2.10 Client Site Chat Group Update

Module 1.3 Client Site Chat.

Module 2.1 Server Site User Authentication

Module 2.2.1 Server Site Root Console

Module 2.2.2 Server Site User Addition

Module 2.2.3 Server Site User Deletion

Module 2.2.4 Server Site User Update

Module 2.2.5 Server Site Room Addition

Module 2.2.6 Server Site Room Deletion

Module 2.2.7 Server Site Room Update

Module 2.2.8 Server Site Chat Group Addition

Module 2.2.9 Server Site Chat Group Deletion

Module 2.2.10 Server Site Chat Group Update

Module 2.3 Server Site Chat

## APPENDIX-D

**The Description of Common Data Elements Used Both in Communication Messages  
and in Algorithms of Modules:**

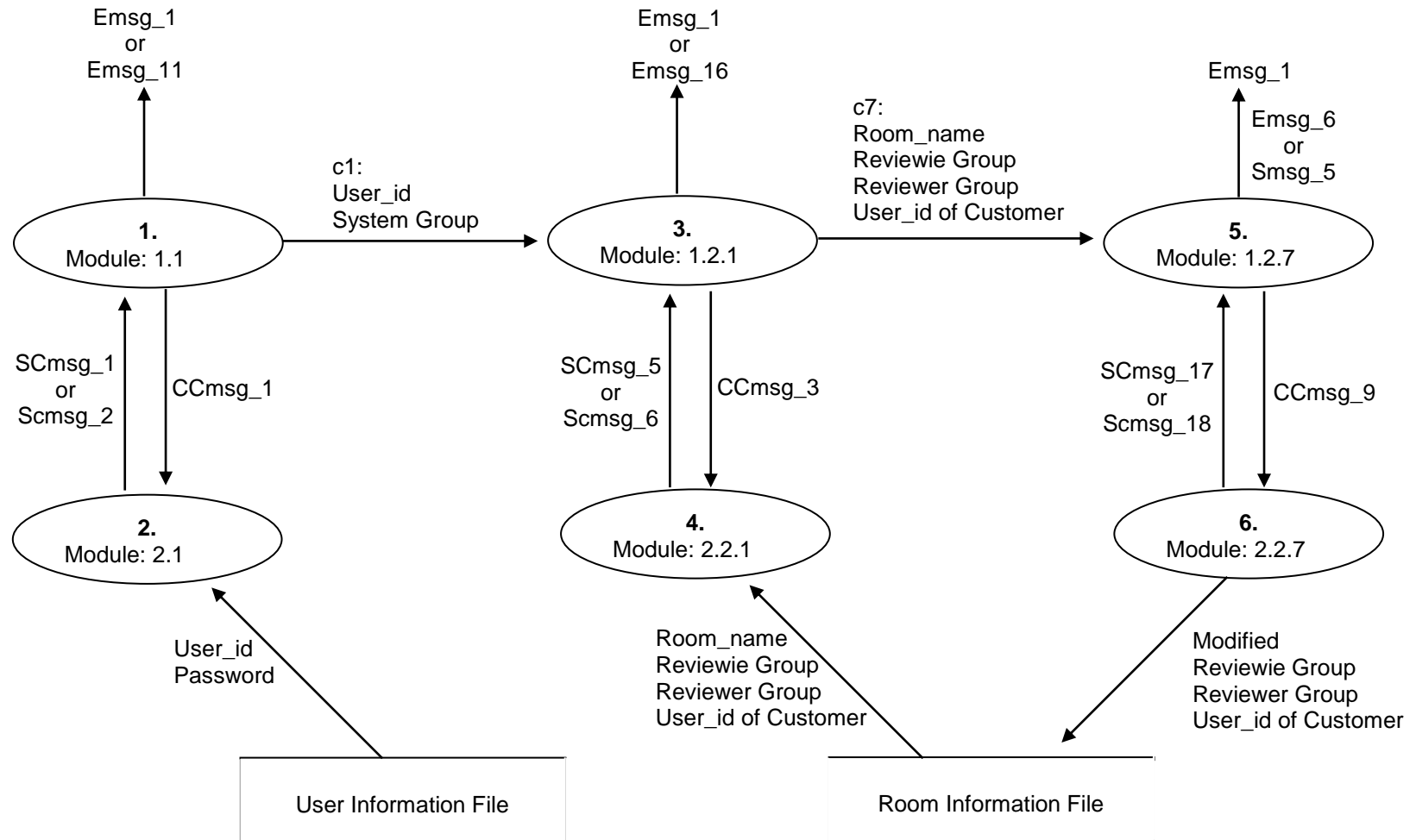
Name	Format	Allowed Range	Semantics
user_id	String	Alphanumeric characters	Username of the user
password	String	Alphanumeric characters	Password of the user
real_name	String	Alphanumeric characters	Real name of the user
system_group	String	Alphanumeric characters	System group name of the user
chat_group	String	Alphanumeric characters	Chat group name of the user
room_name	String	Alphanumeric characters	Name of the room
reviewie_group	String	Alphanumeric characters	Chat group assigned to the room as reviewie group
reviewer_group	String	Alphanumeric characters	Chat group assigned to the room as reviwer group
customer_user_id	String	Alphanumeric characters	User assigned to the room as customer
room_name_as_reviewie_group	String	Alphanumeric characters	Name of the room for which the specified chat goup is assigned to the room as the reviewie group
room_name_as_reviewer_group	String	Alphanumeric characters	Name of the room for which the specified chat goup is assigned to the room as the reviewer group
chat_message	String	All characters	Message of the user

**Table 1:** Description of common data elements

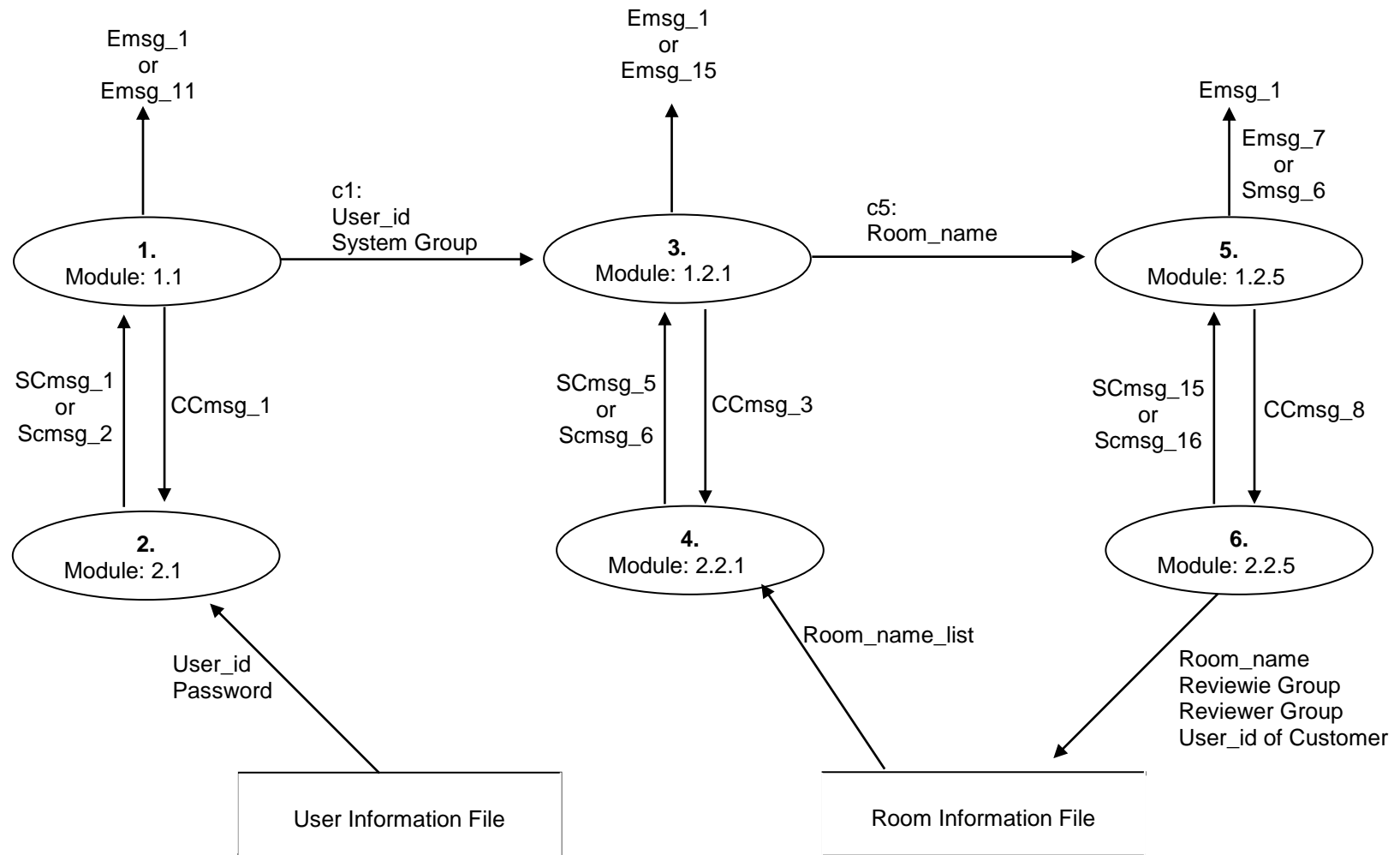


## APPENDIX-E

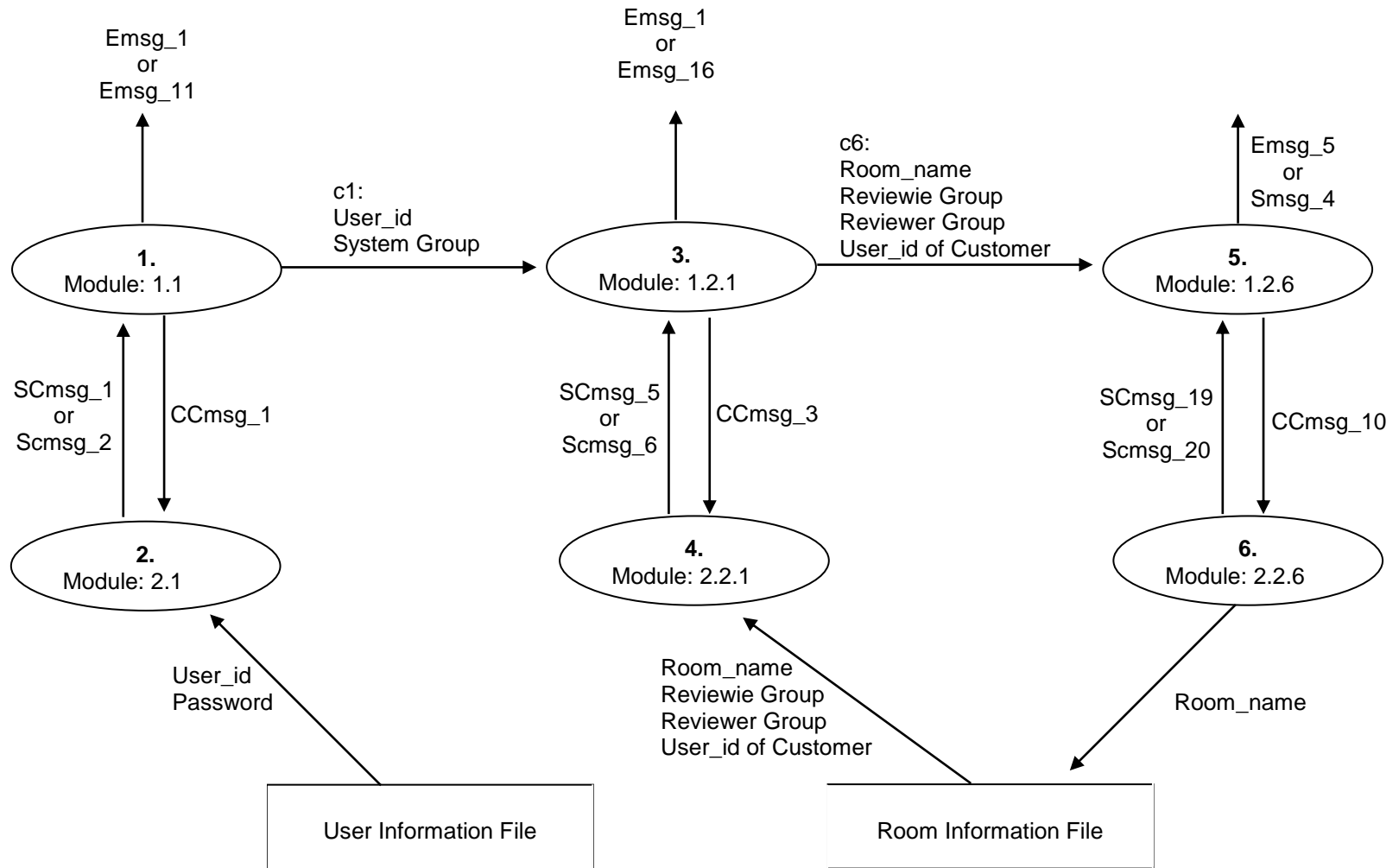
### Room Update:



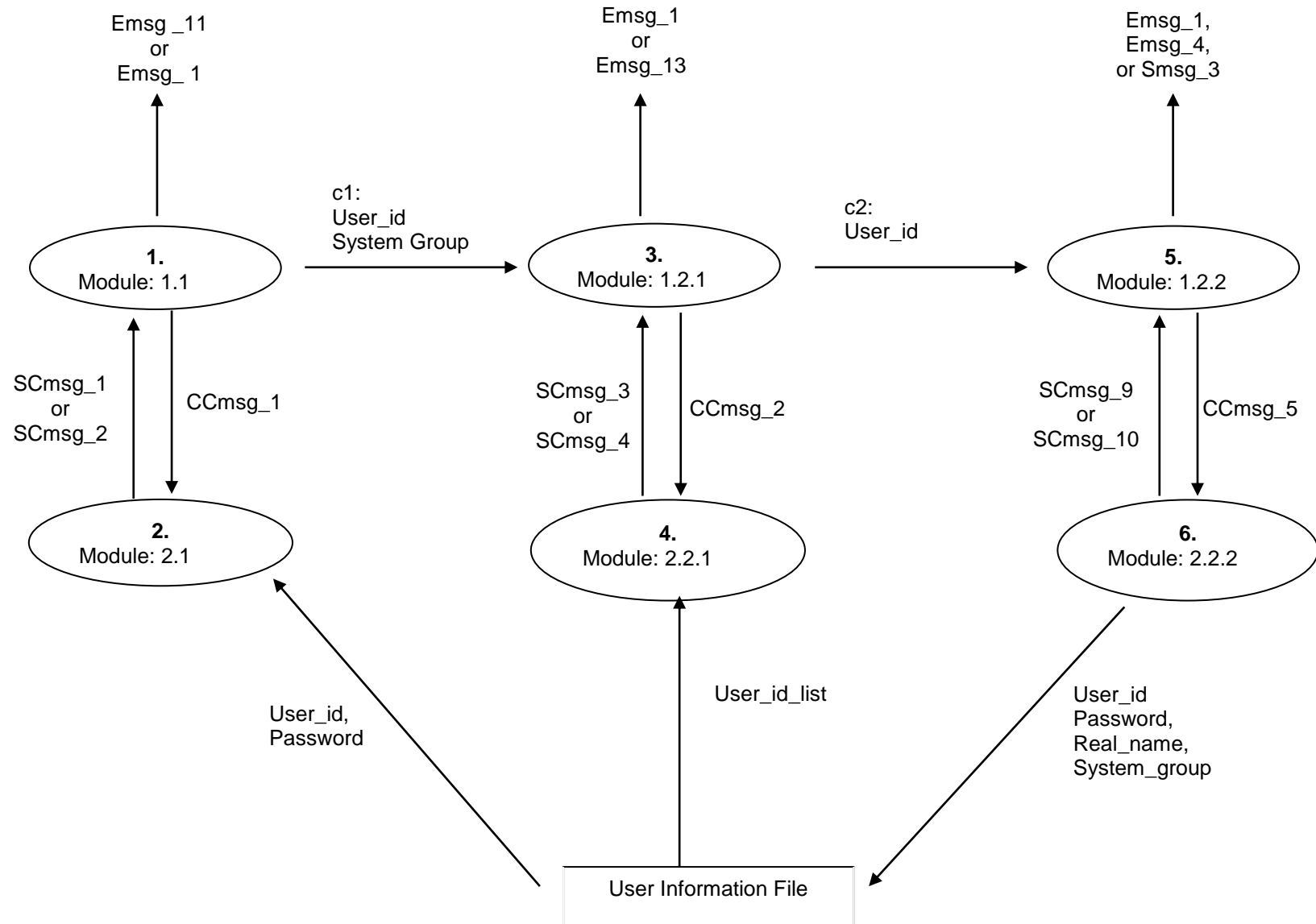
## Room Addition:



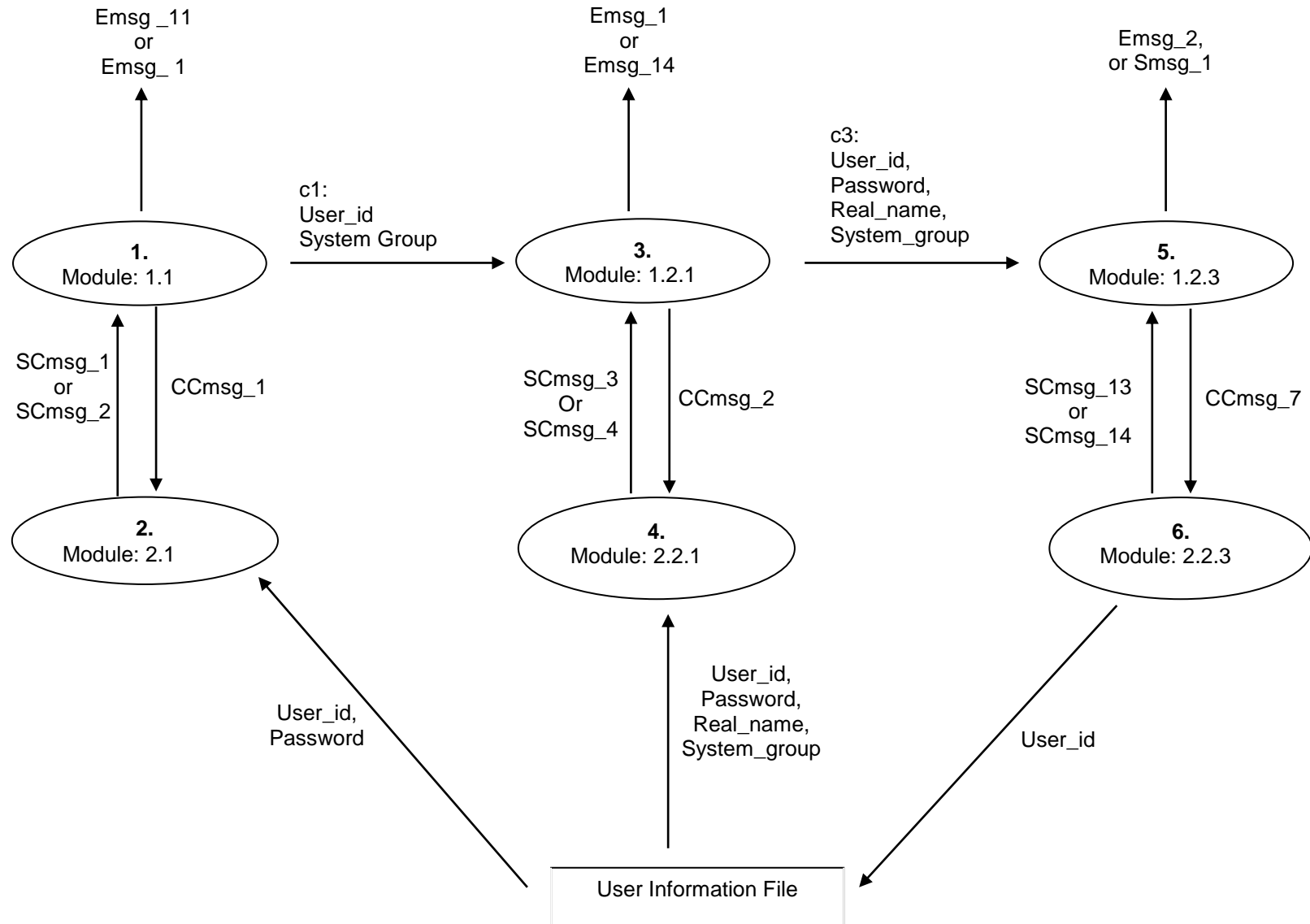
## Room Deletion:



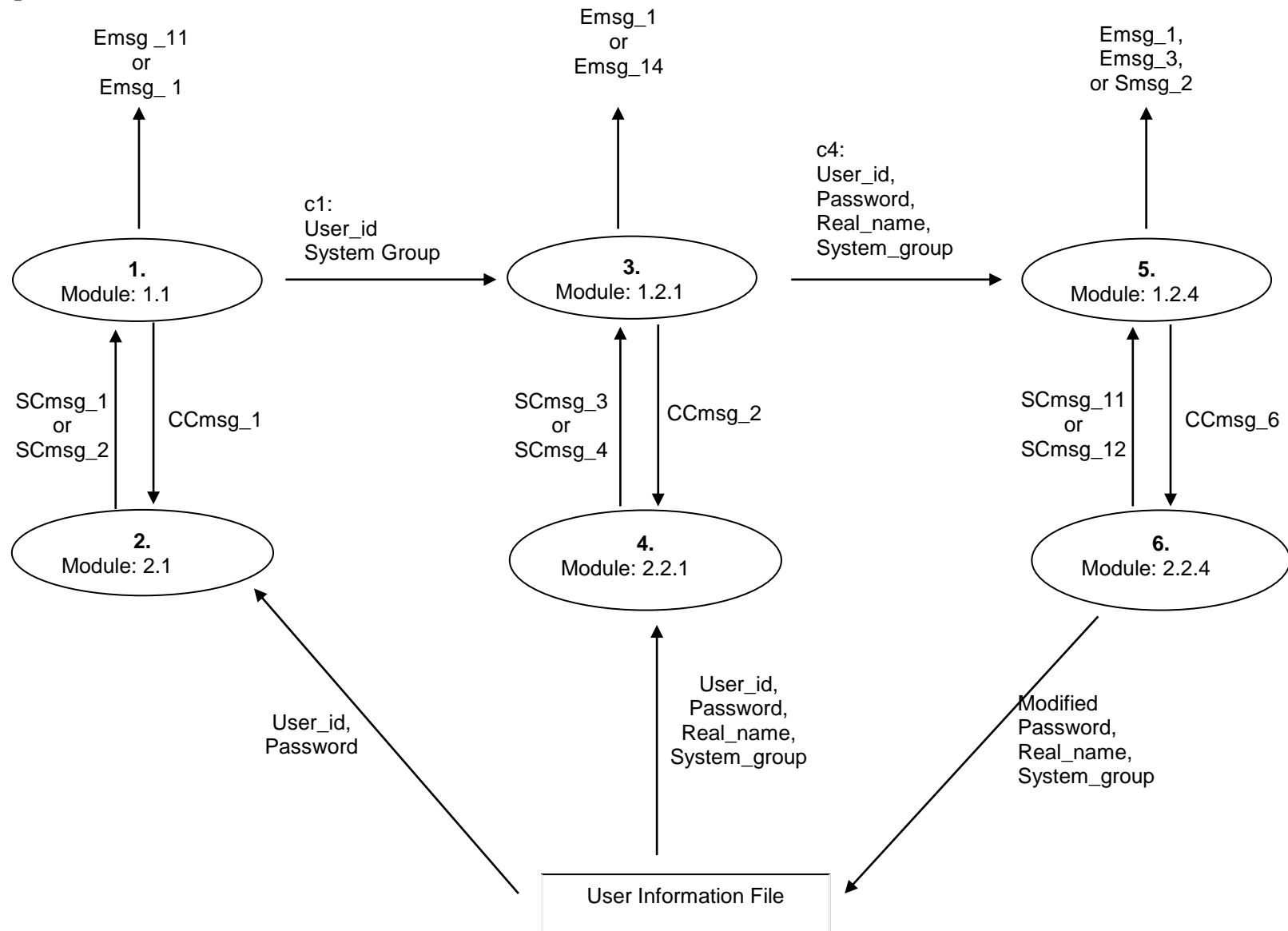
## User Addition:



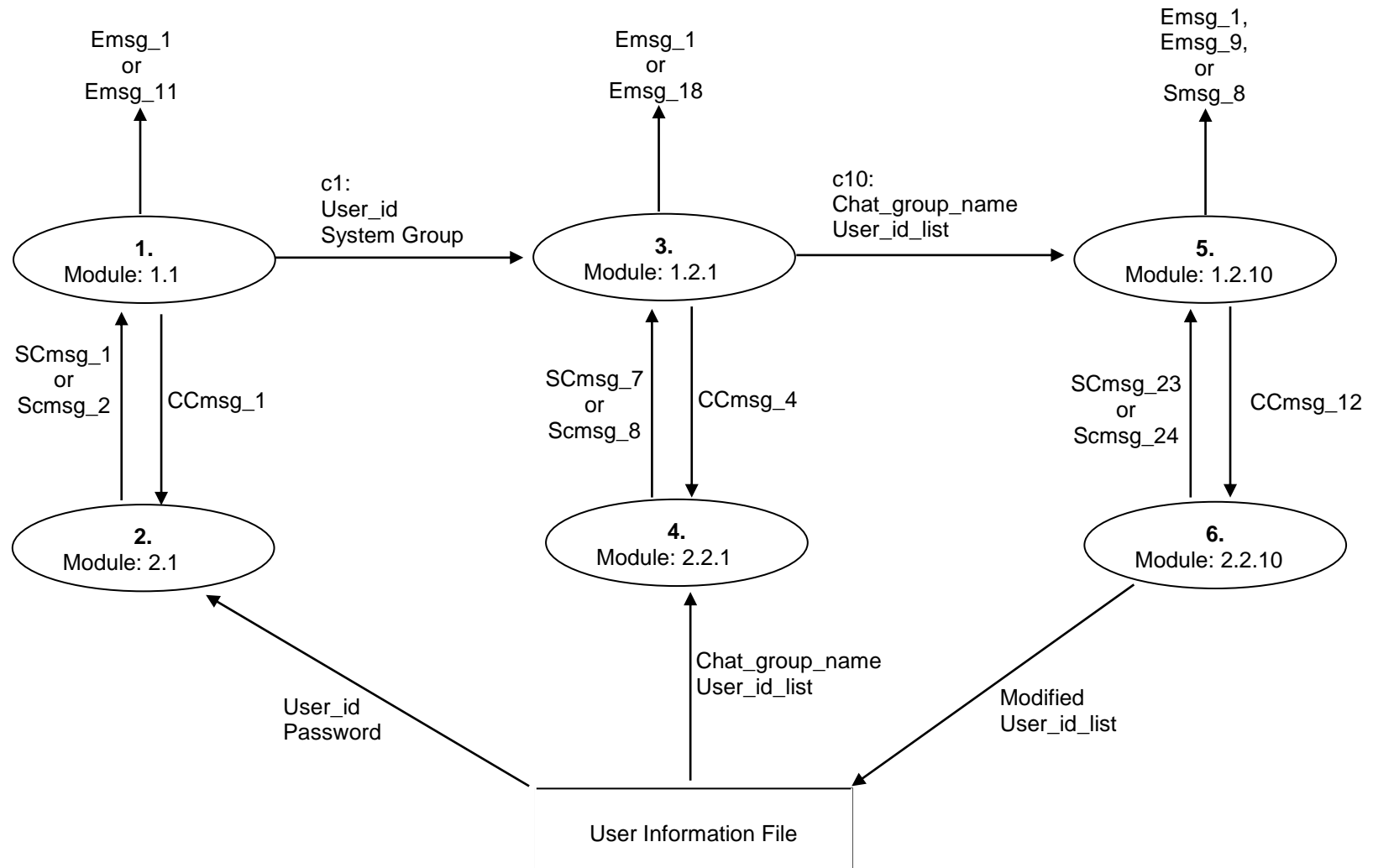
## User Deletion:



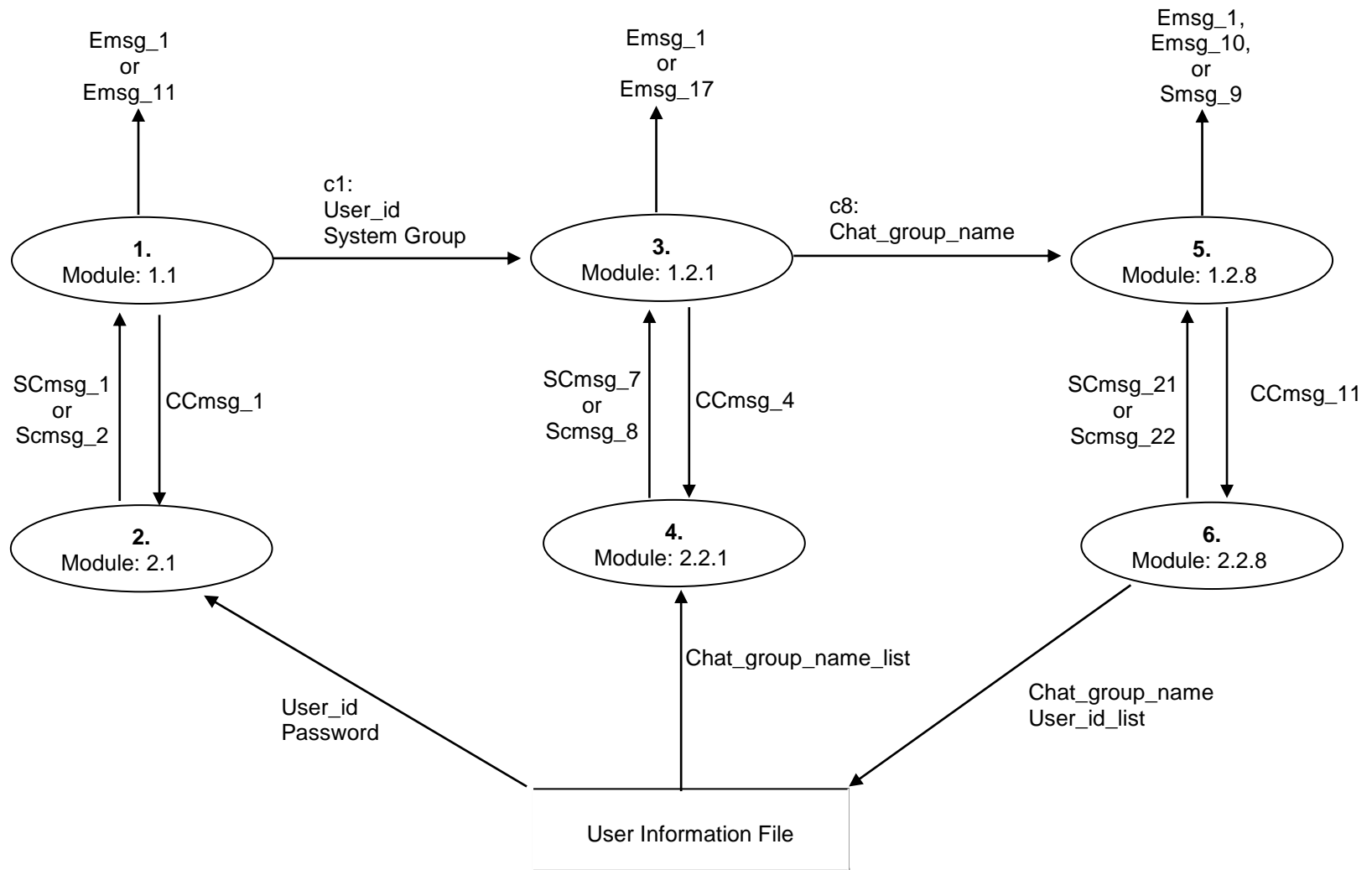
## User Update:



## Chat Group Update:

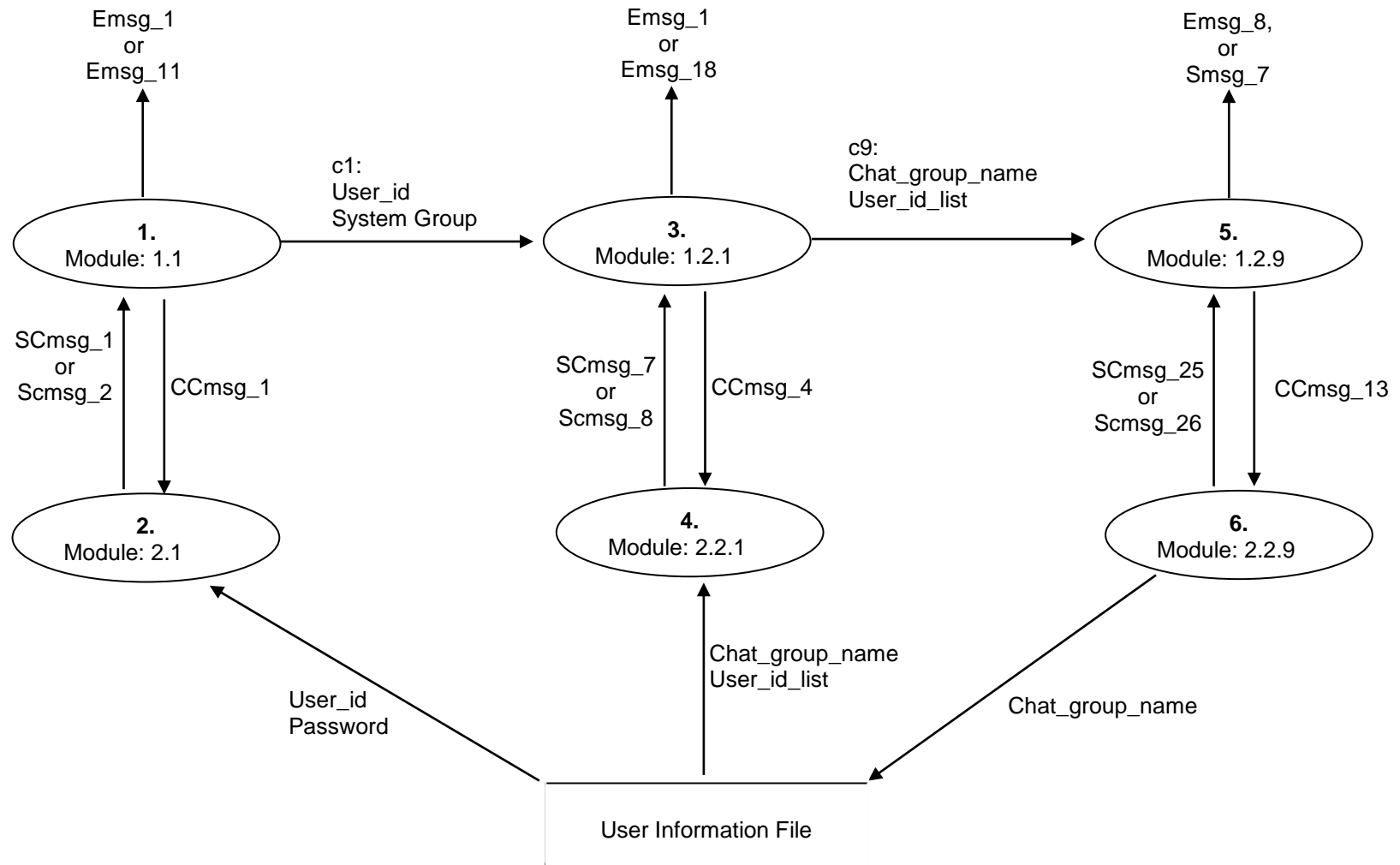


## Chat Group Addition:

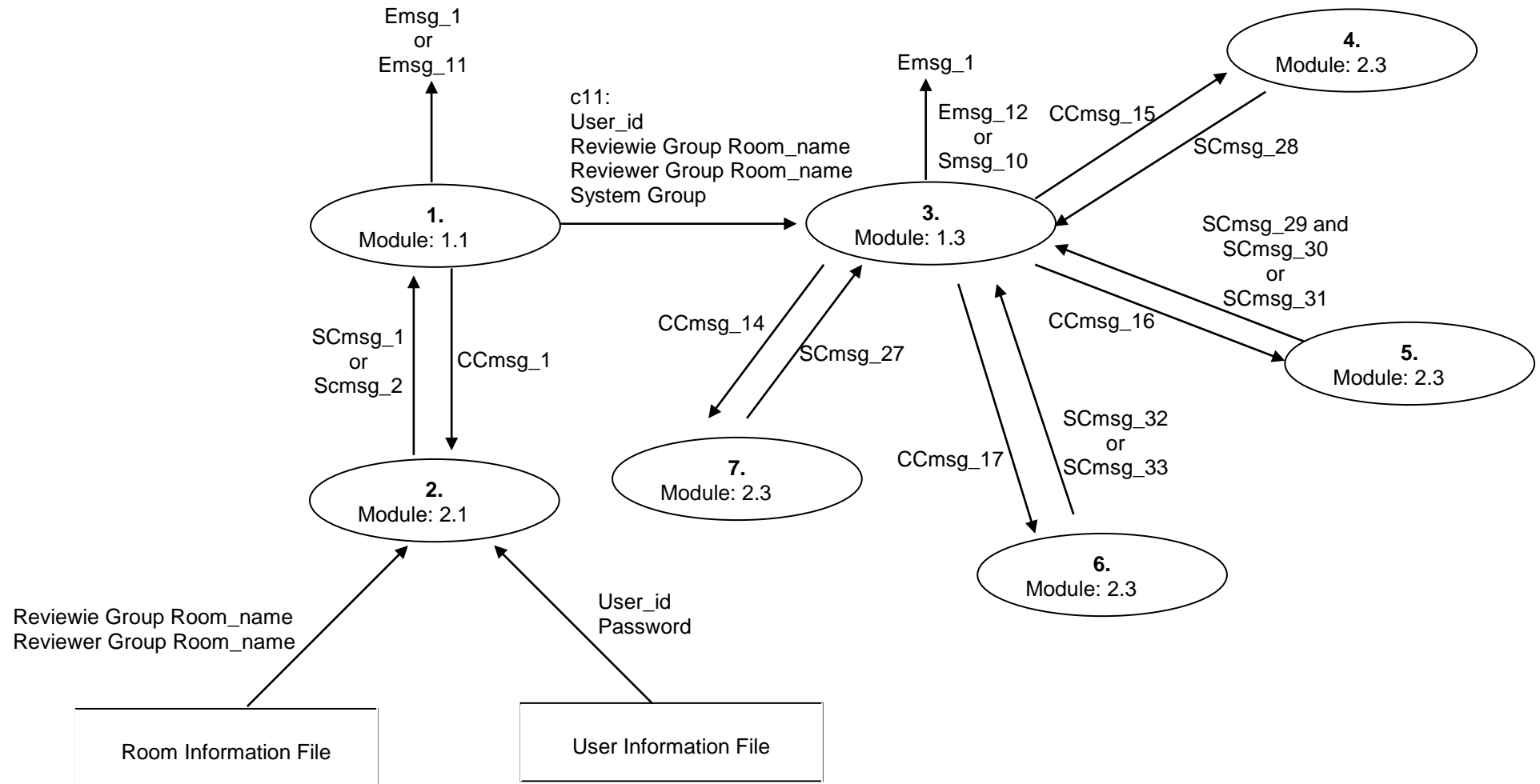




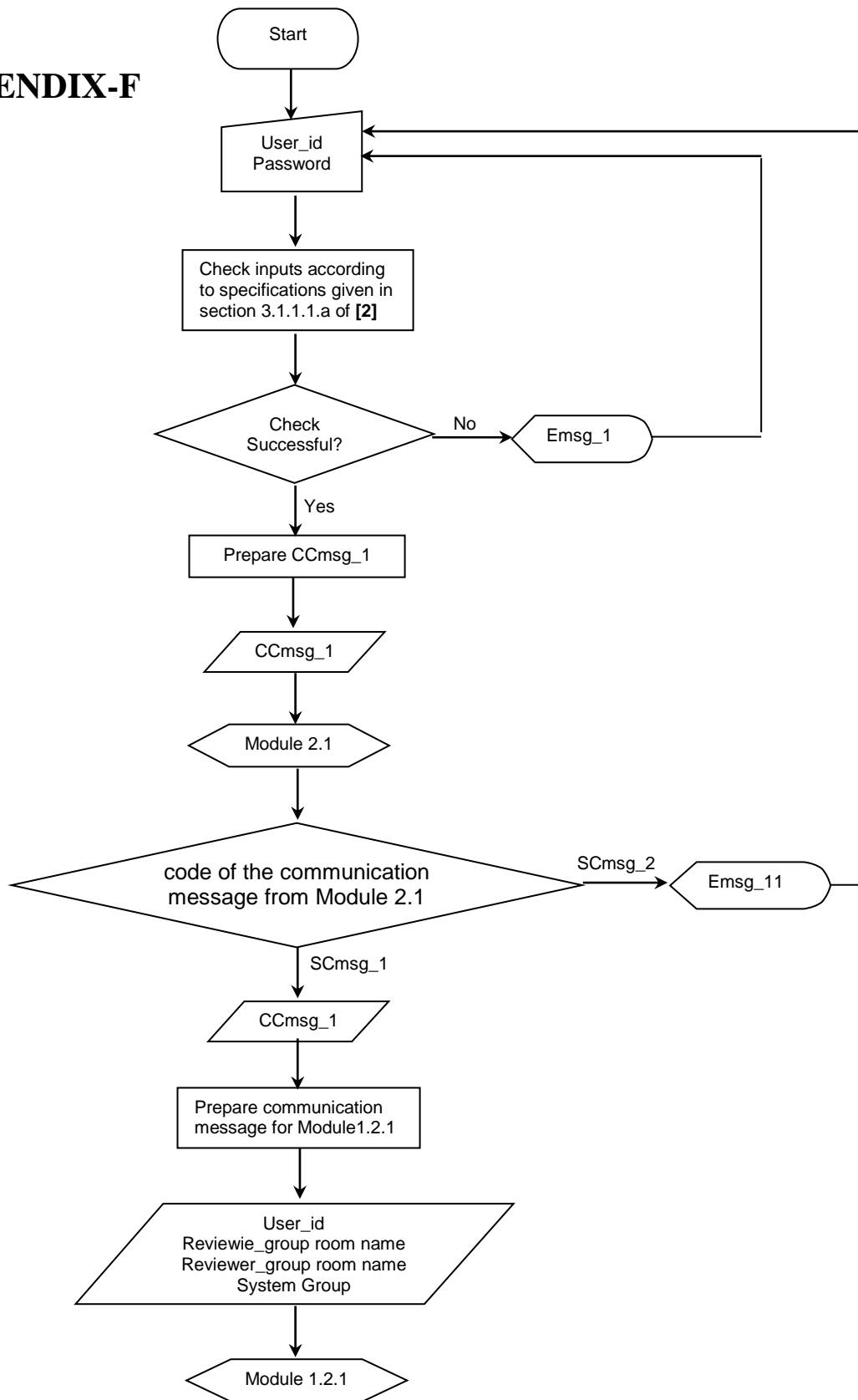
## Chat Group Deletion:



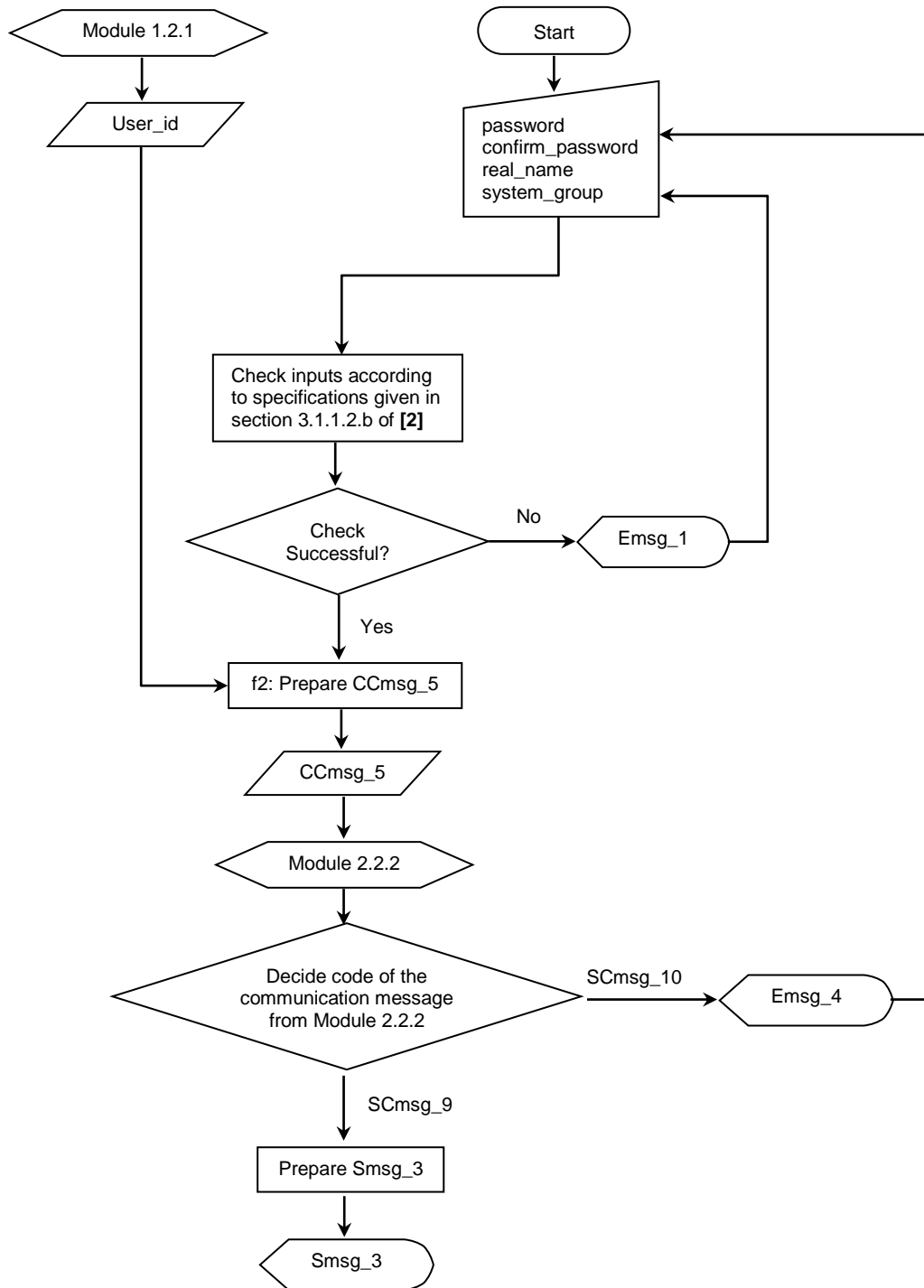
## Chat:



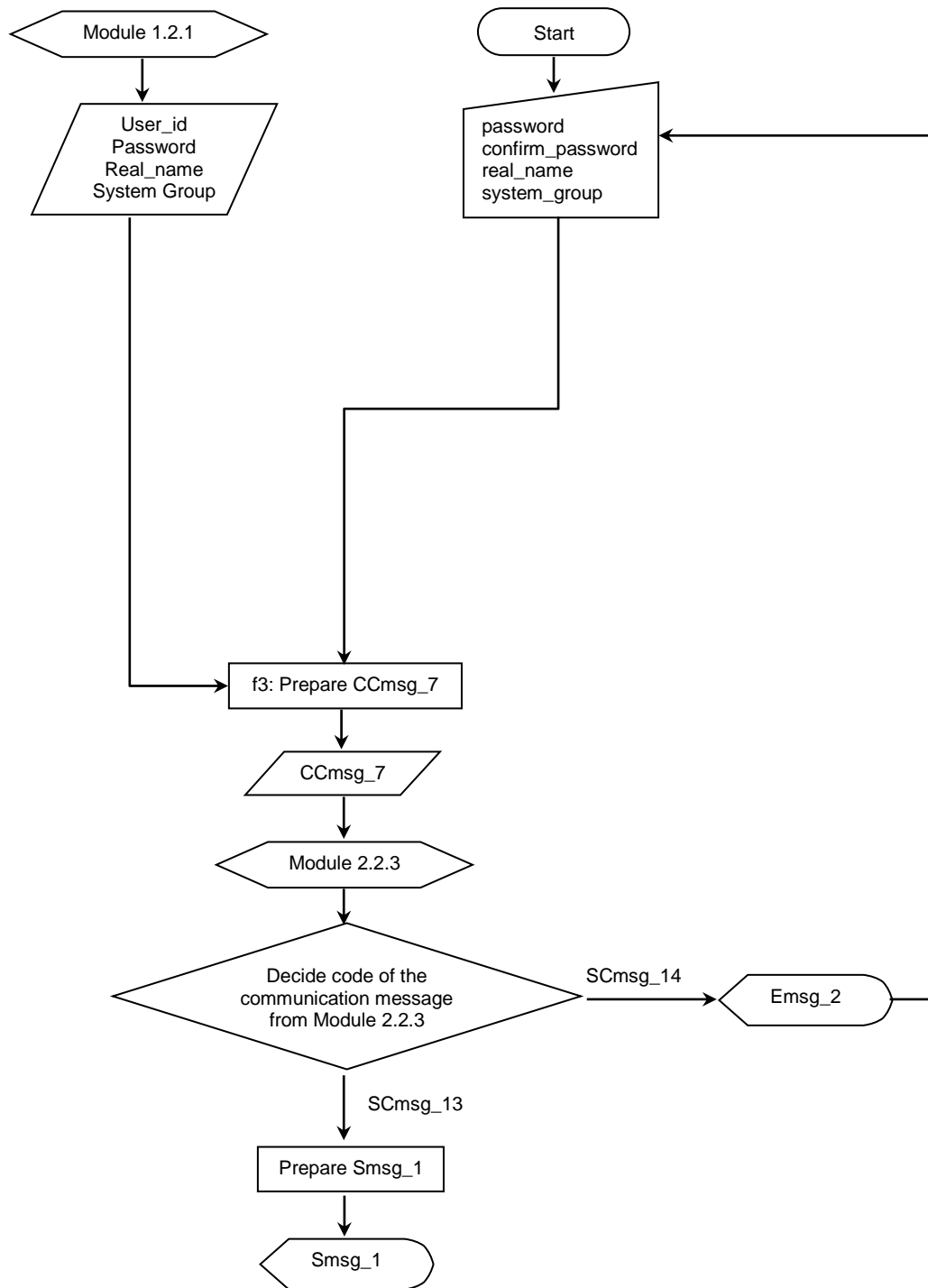
## APPENDIX-F



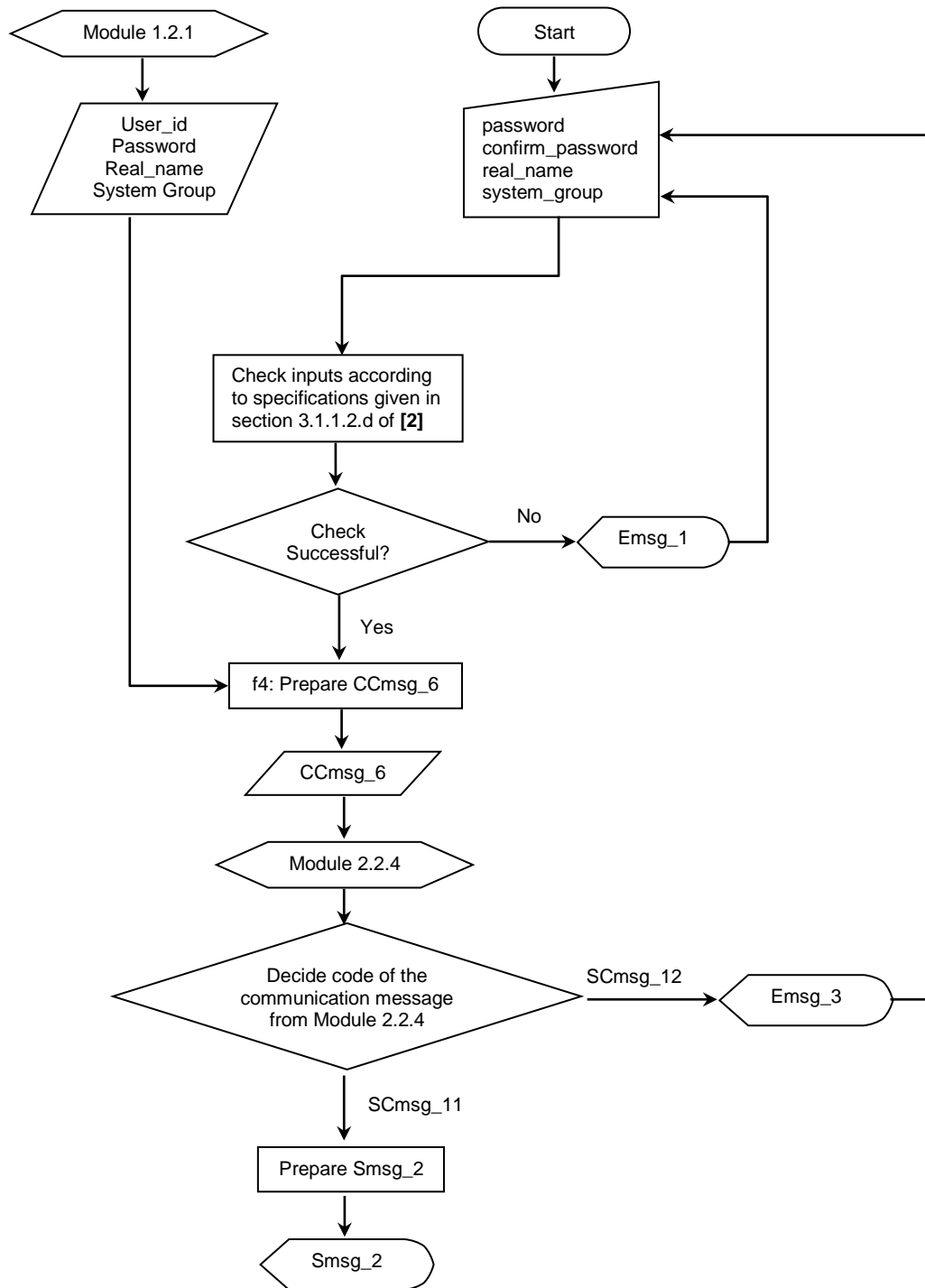
**Figure 27:**Flow Chart of Authentication for Client software



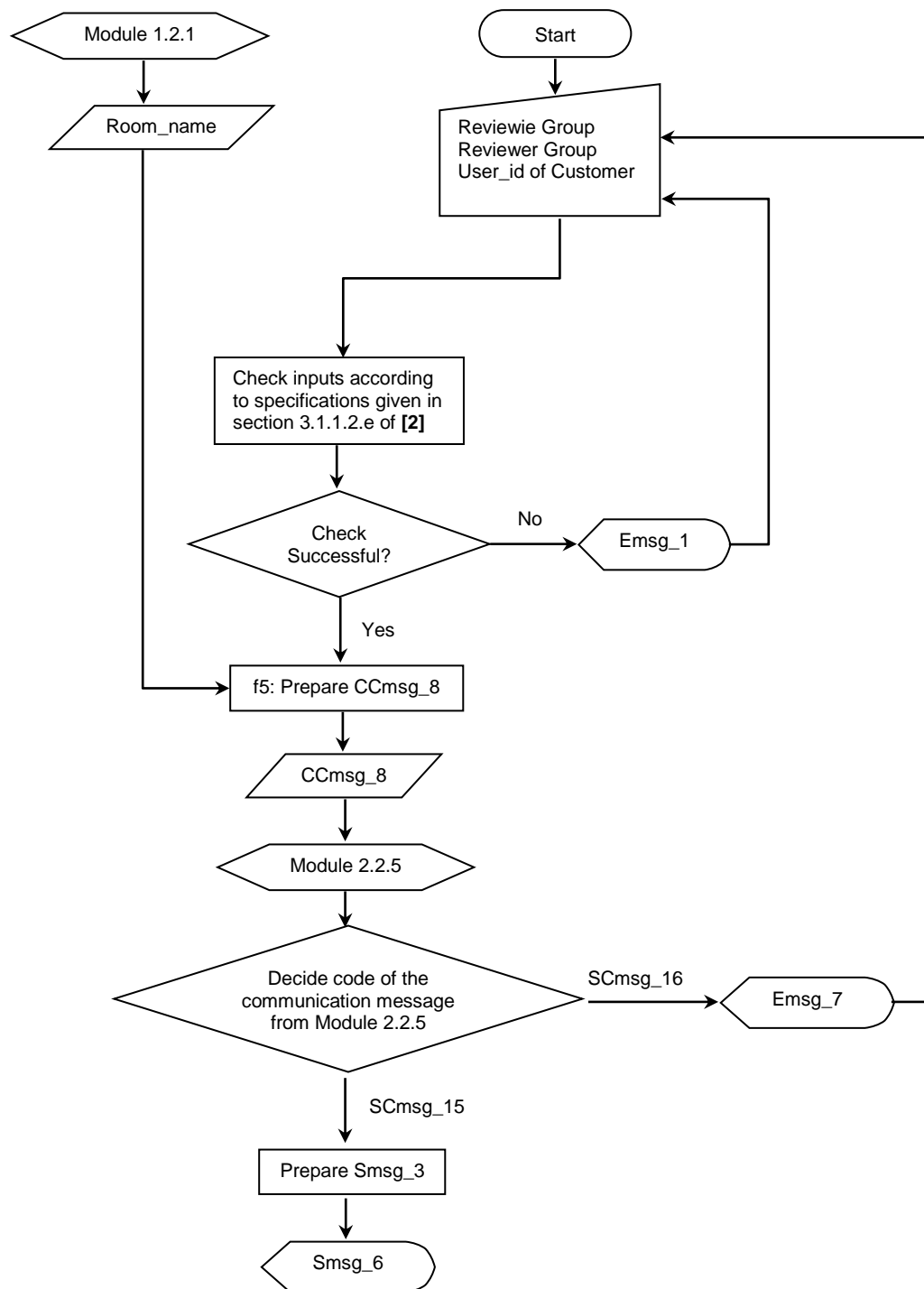
**Figure 28:** Flow Chart of User Addition for Client software



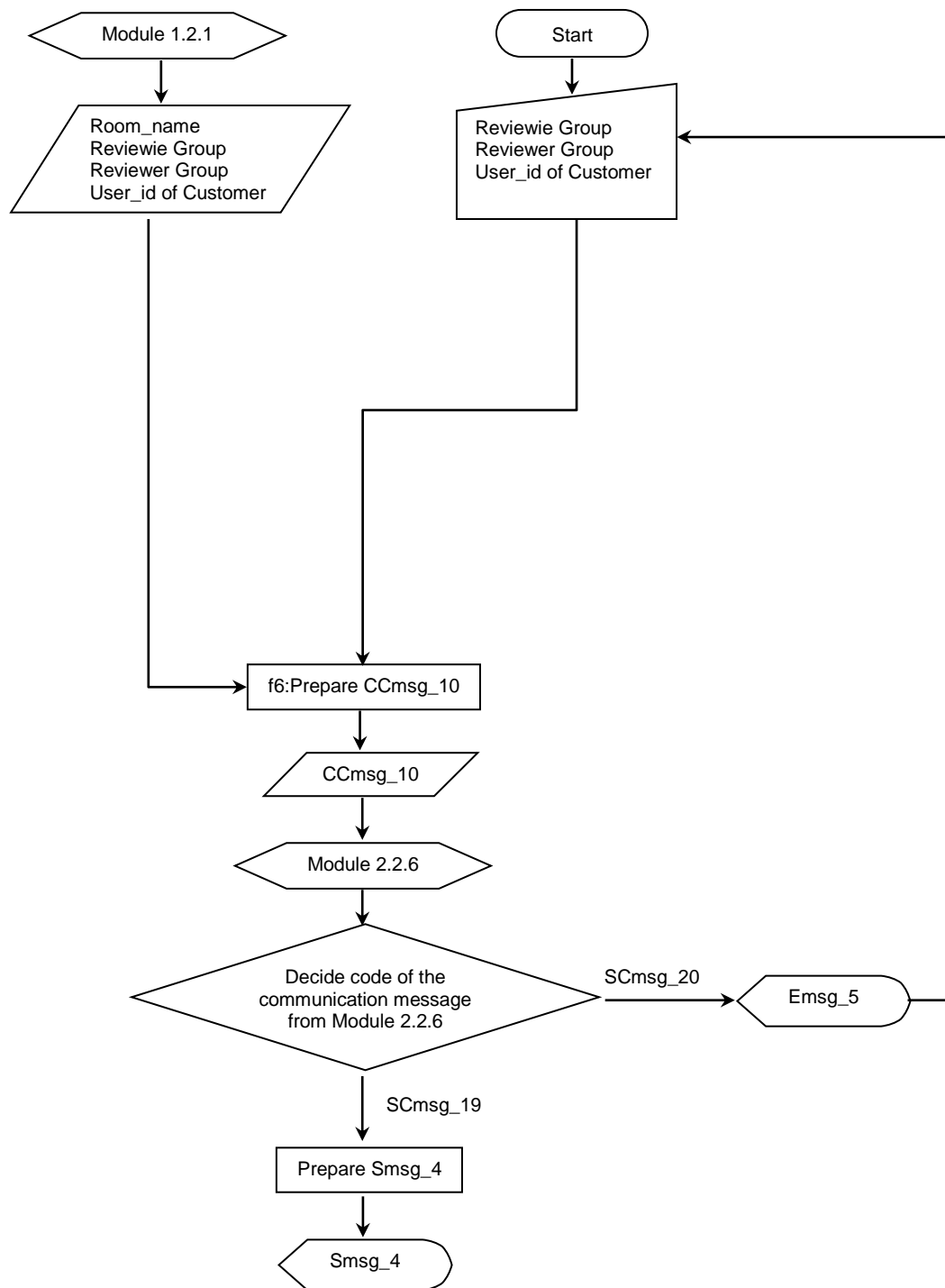
**Figure 29:** Flow Chart of User Deletion for Client software



**Figure 30:** Flow Chart of User Update for Client software

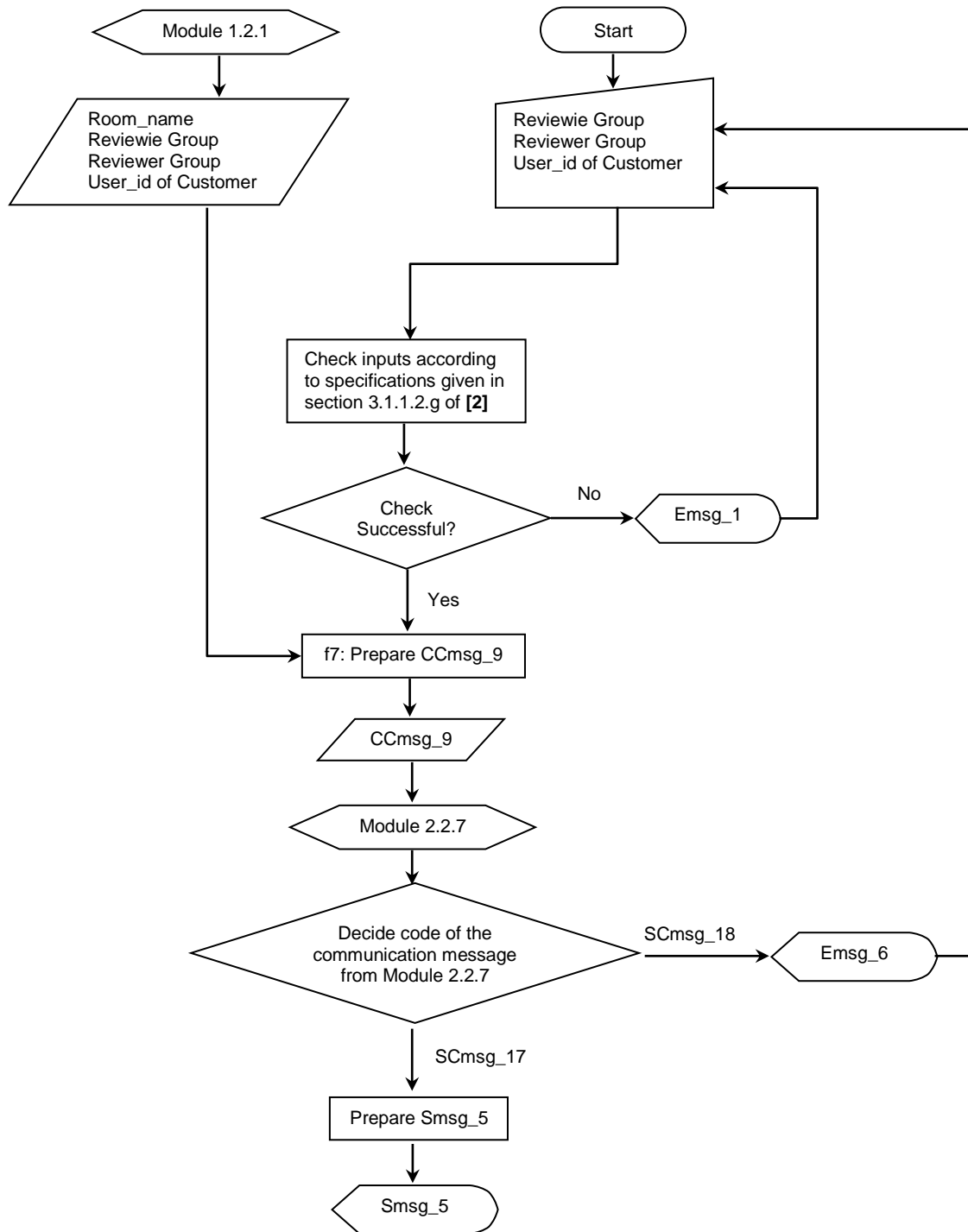


**Figure 31:** Flow Chart of Room Addition for Client Software

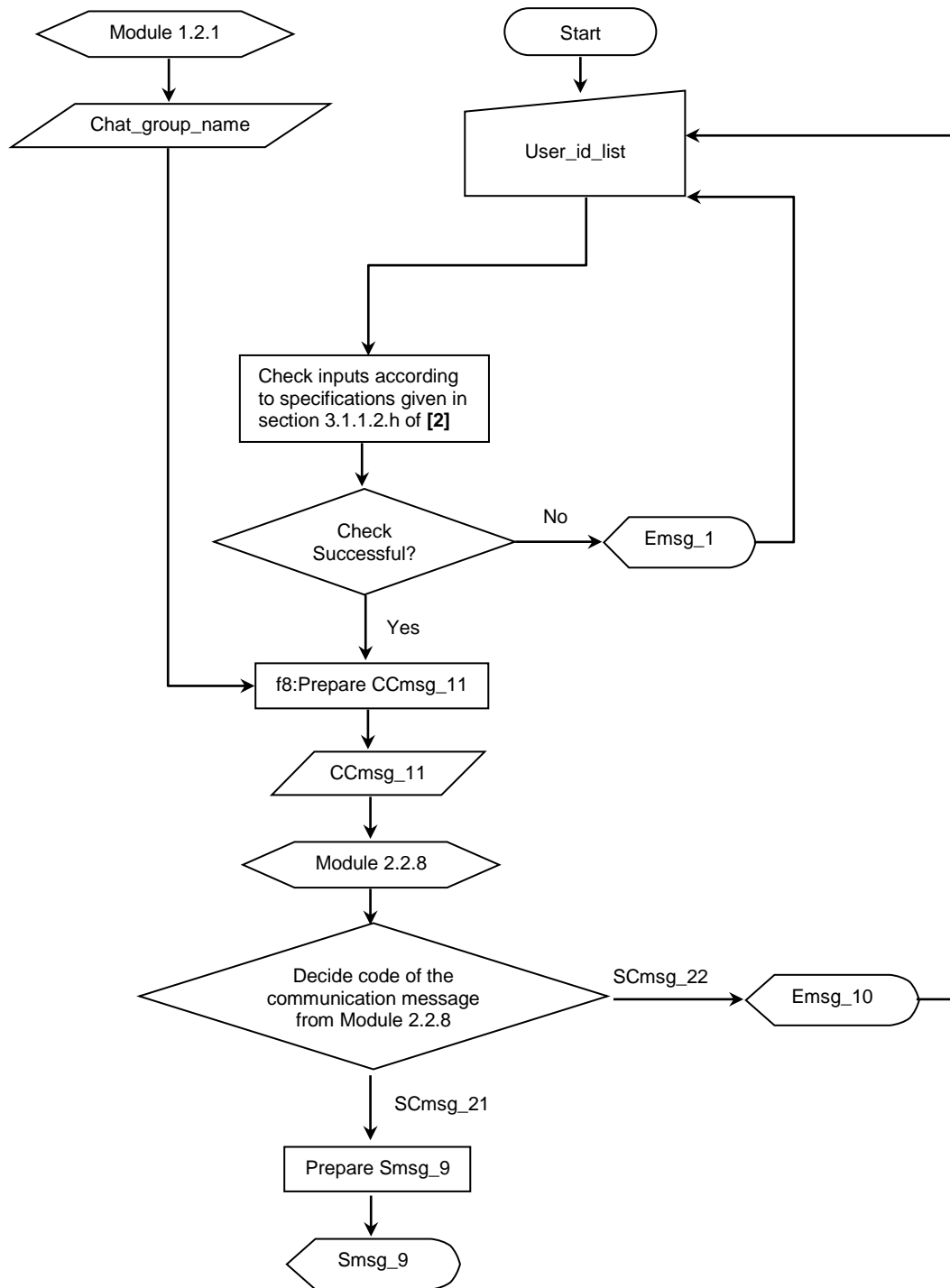


**Figure 32:** Flow Chart of Room Deletion for Client software

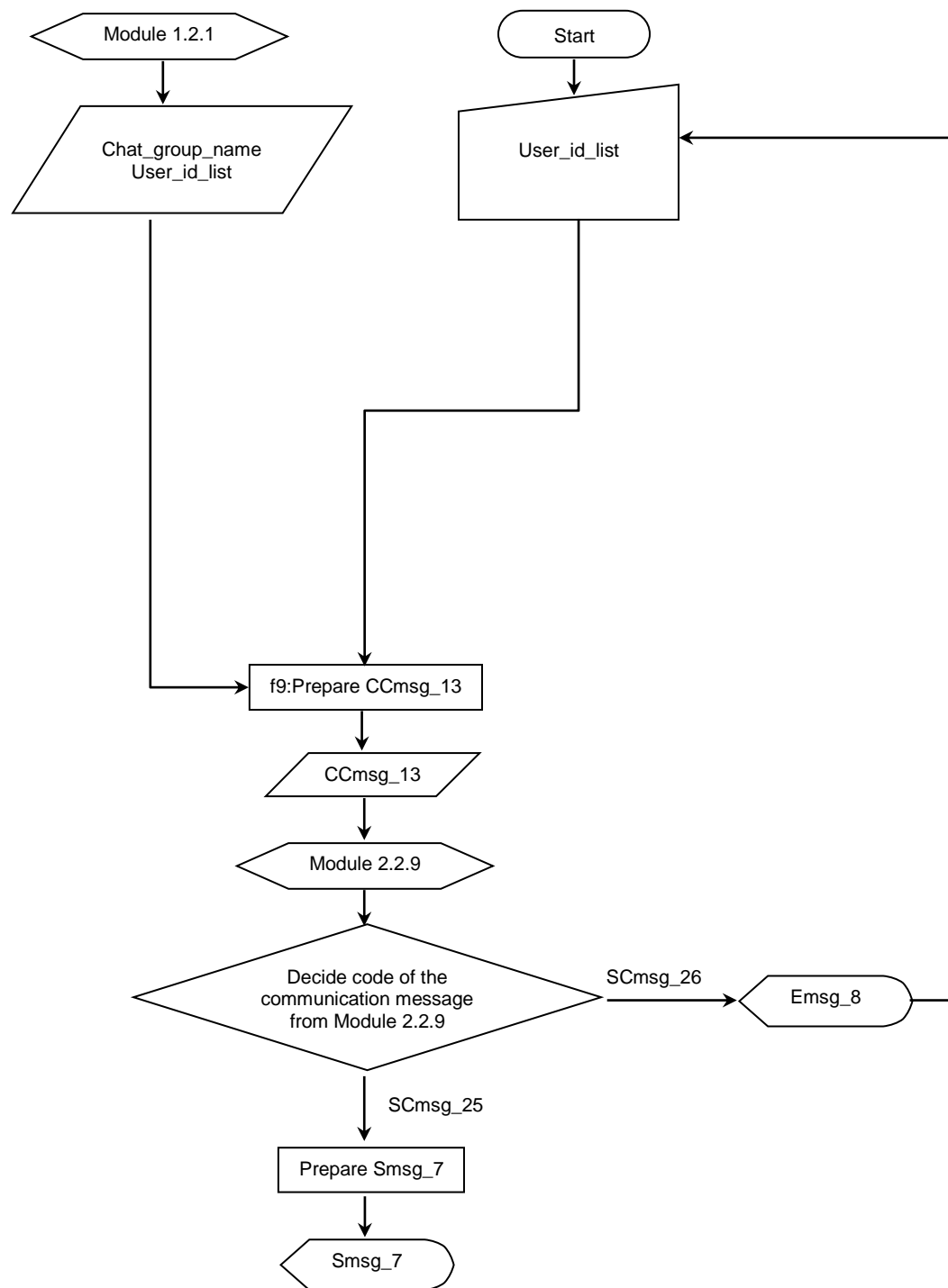




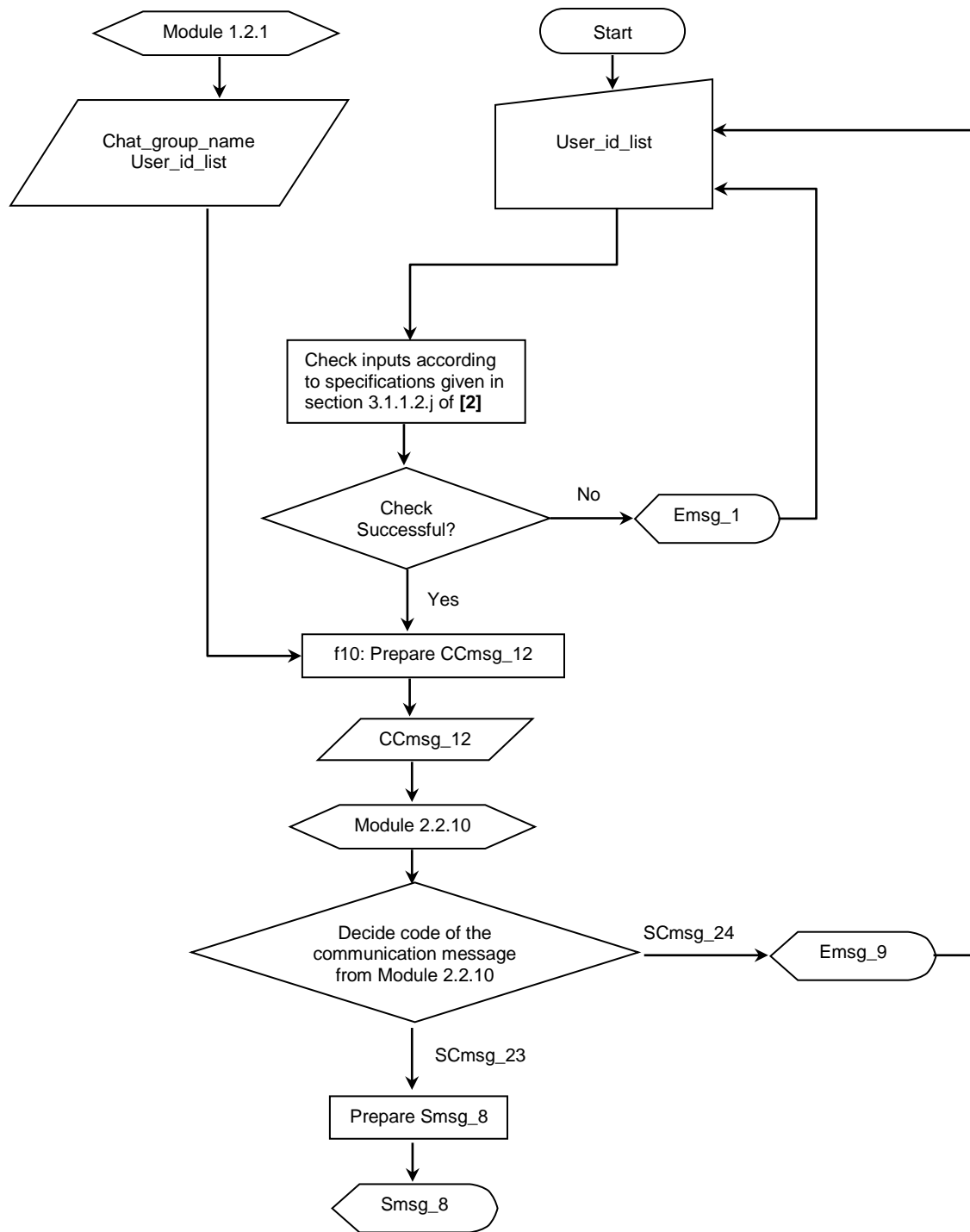
**Figure 33:** Flow Chart of Room Update for Client software



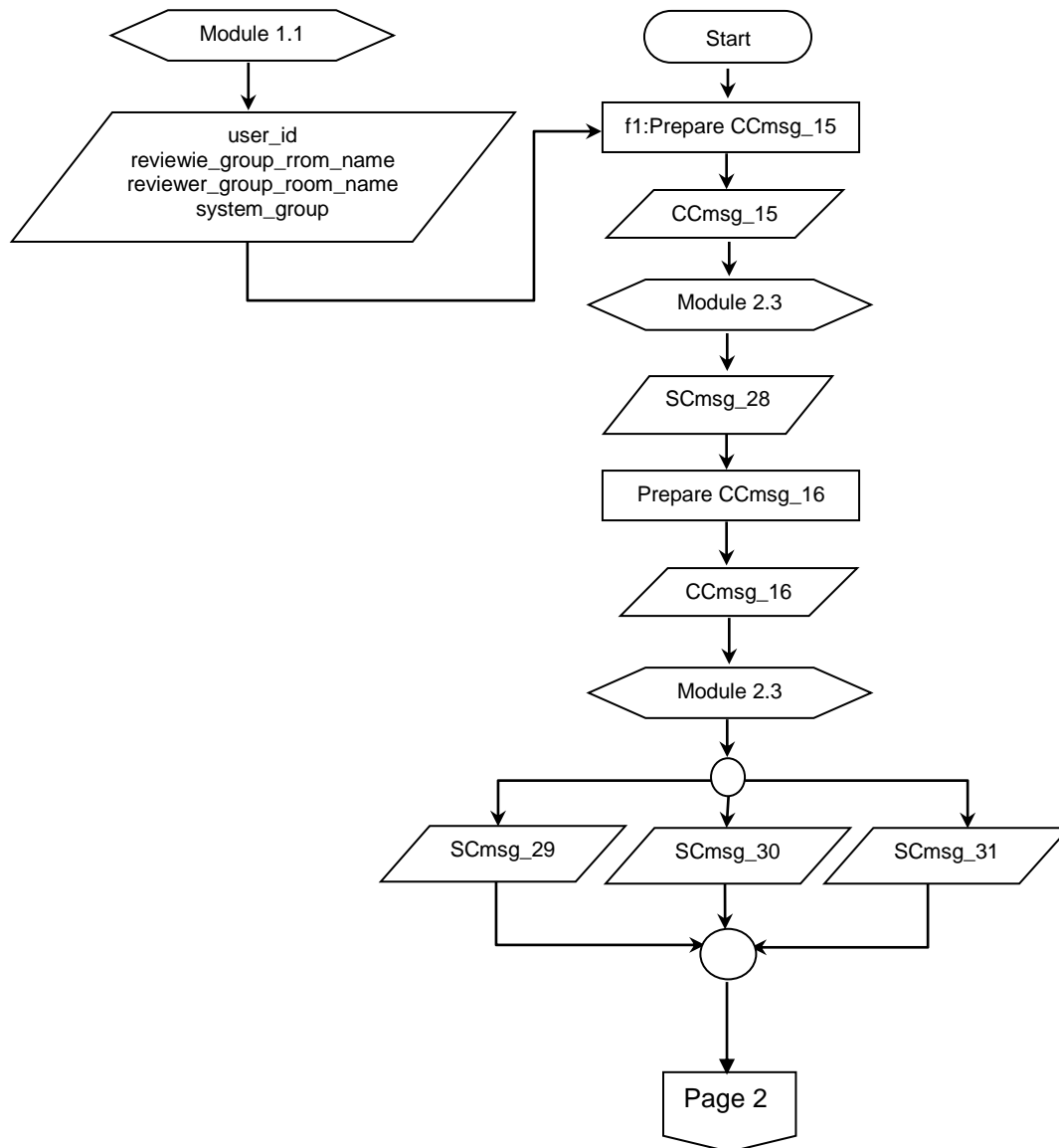
**Figure 34:** Flow Chart of Chat Group Addition for Client Software

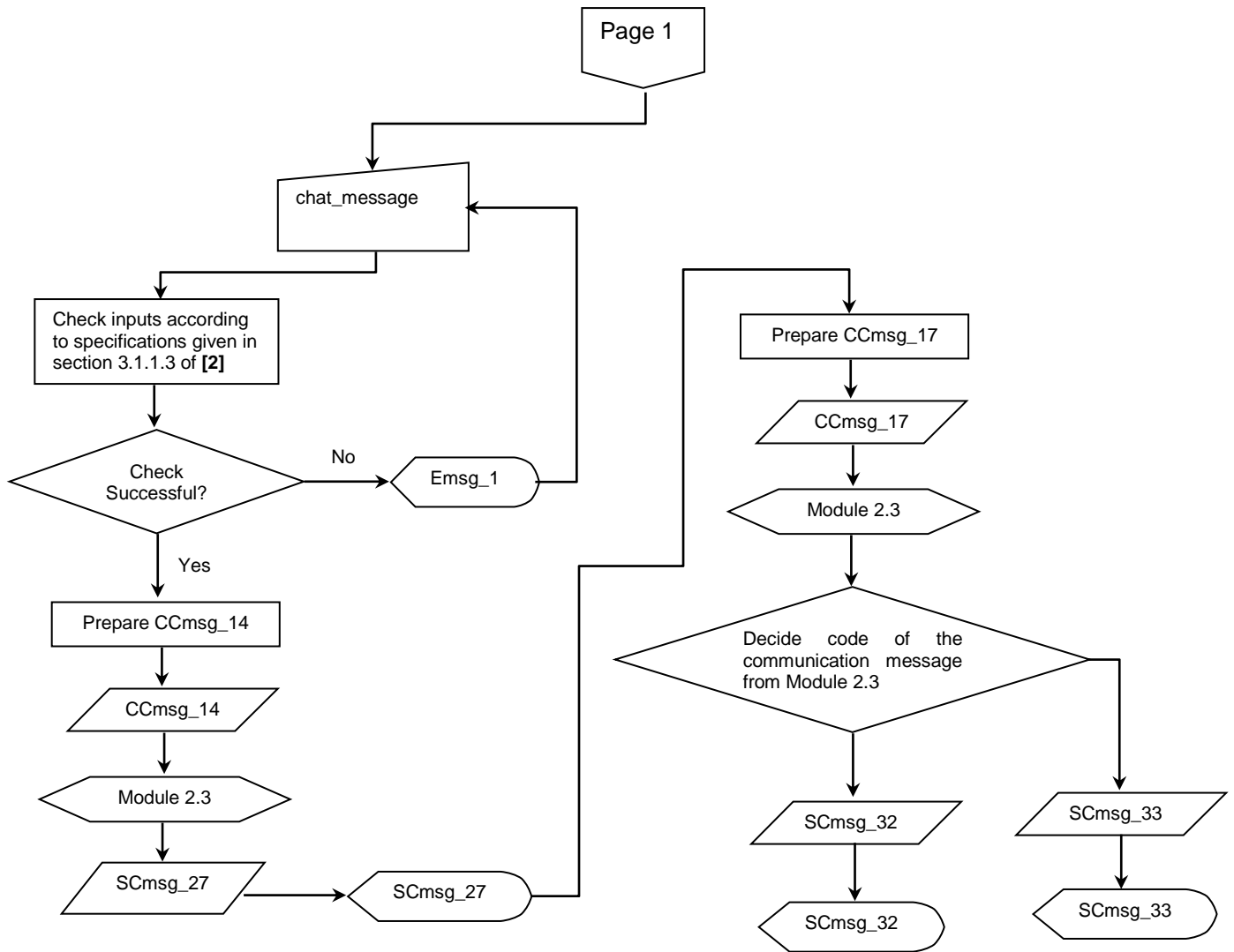


**Figure 35:**Flow Chart of Chat Group Deletion for Client software

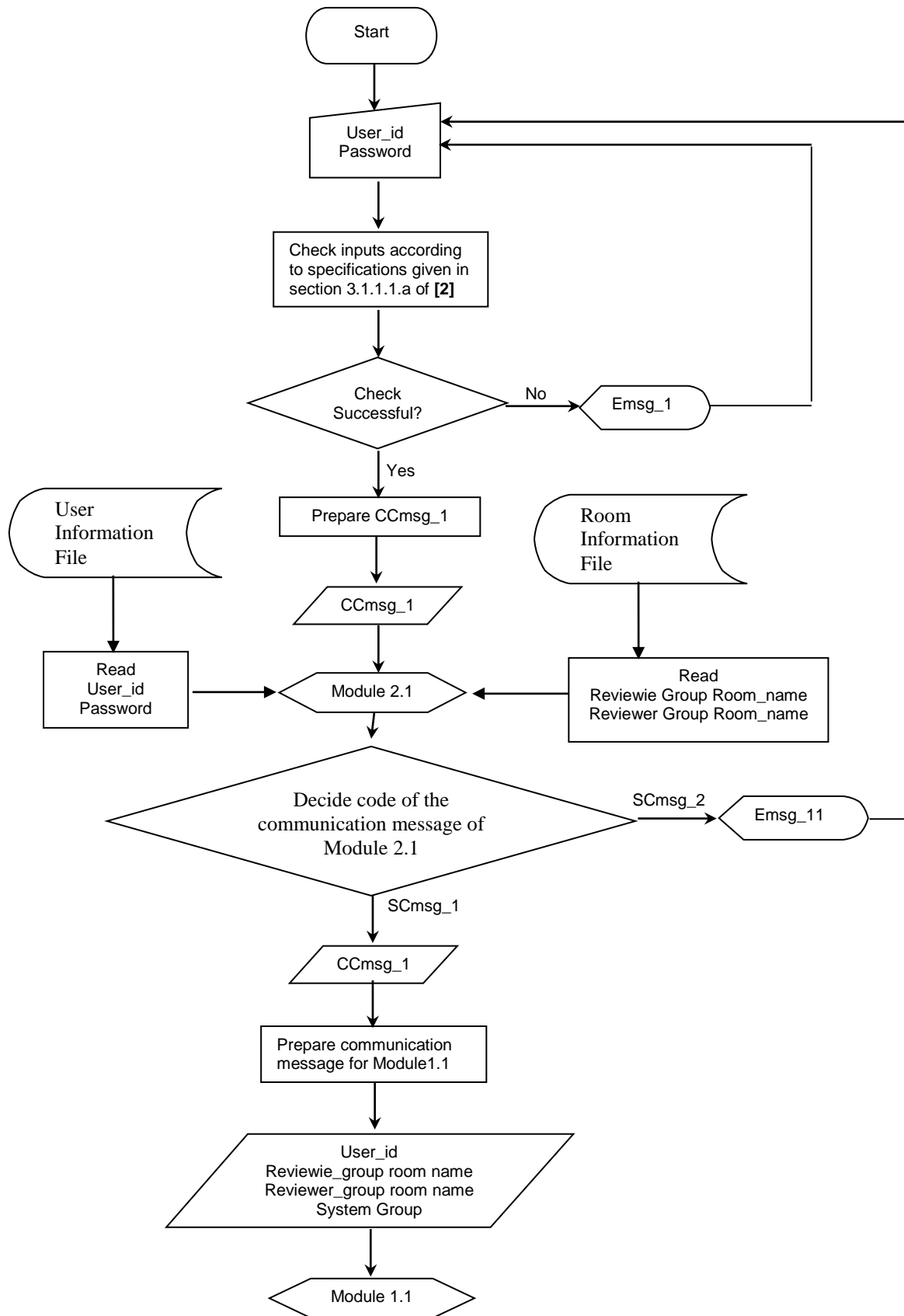


**Figure 36:**Flow Chart of Chat Group Update for Client software

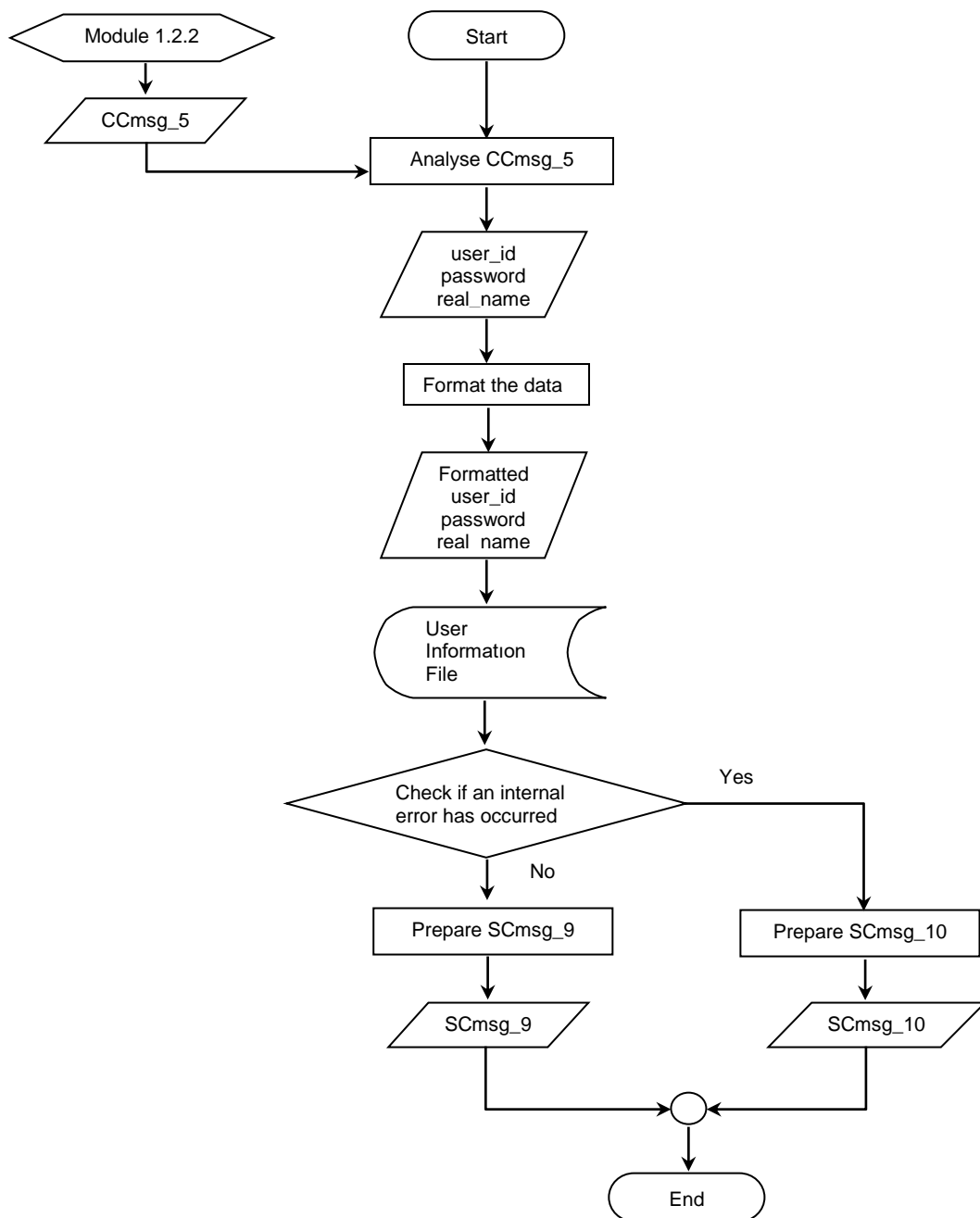




**Figure 37:**Flow Chart of Chat for Client software

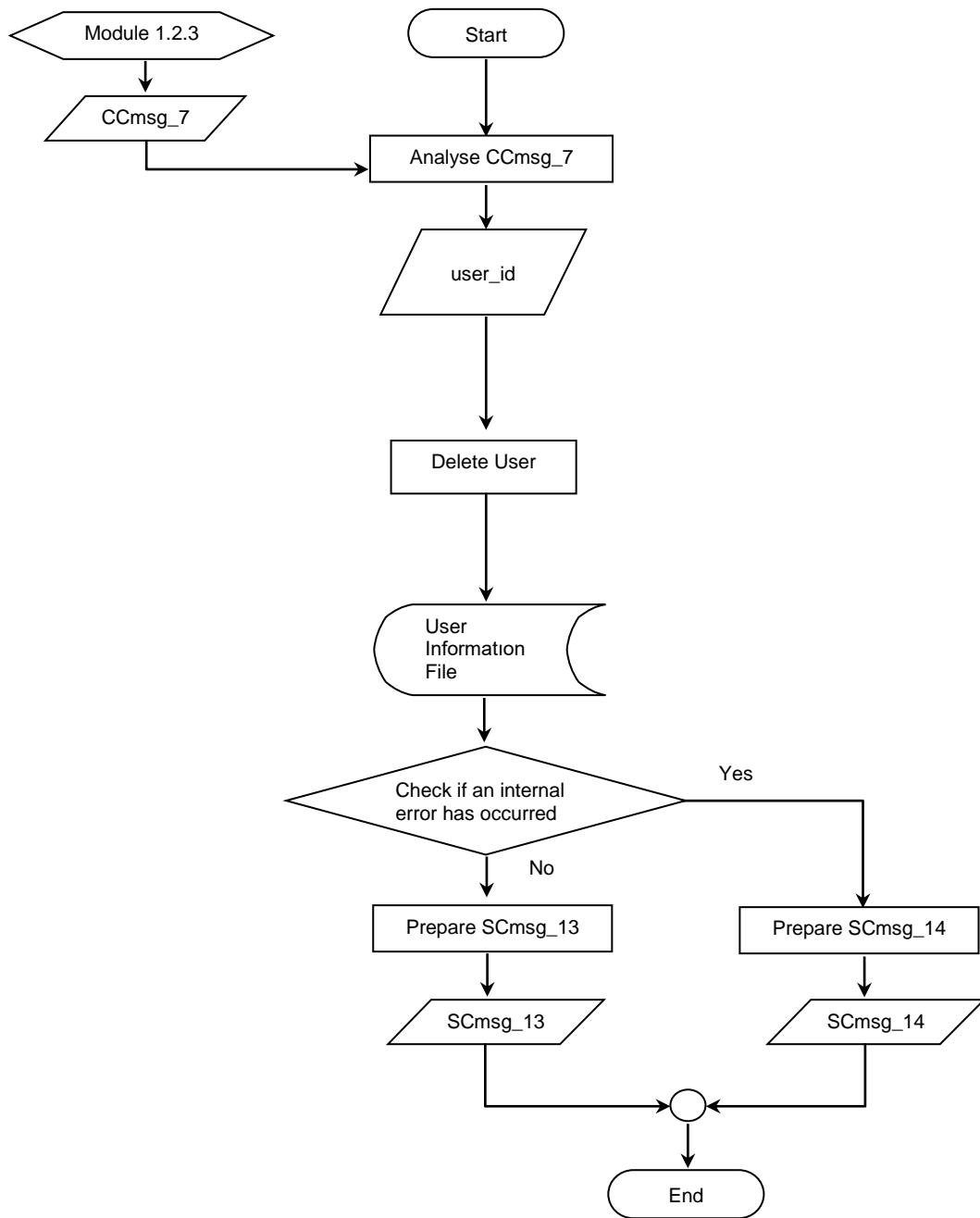


**Figure 38:**Flow Chart of Authentication for Server software

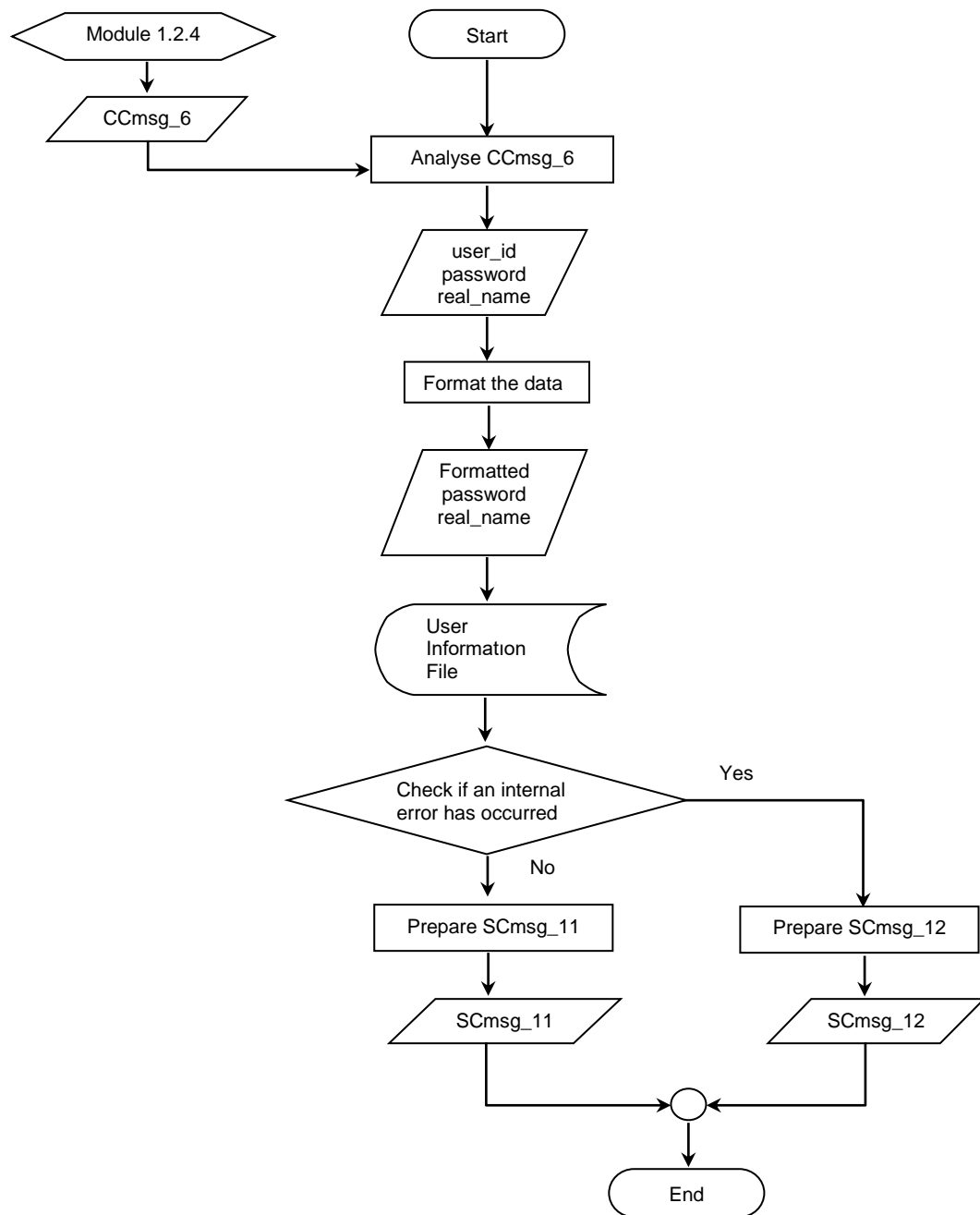


**Figure 39:**Flow Chart of User Addition for Server software

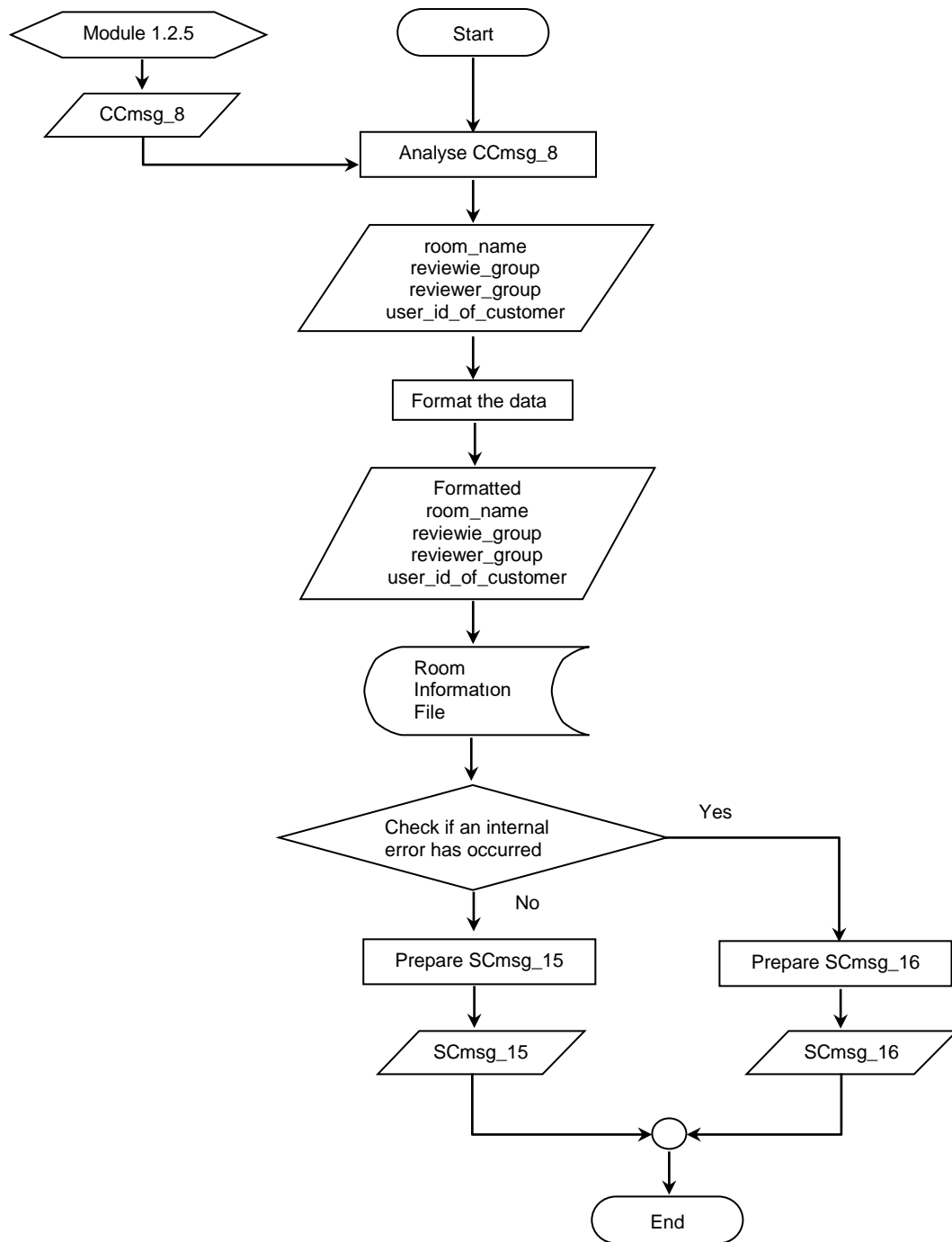




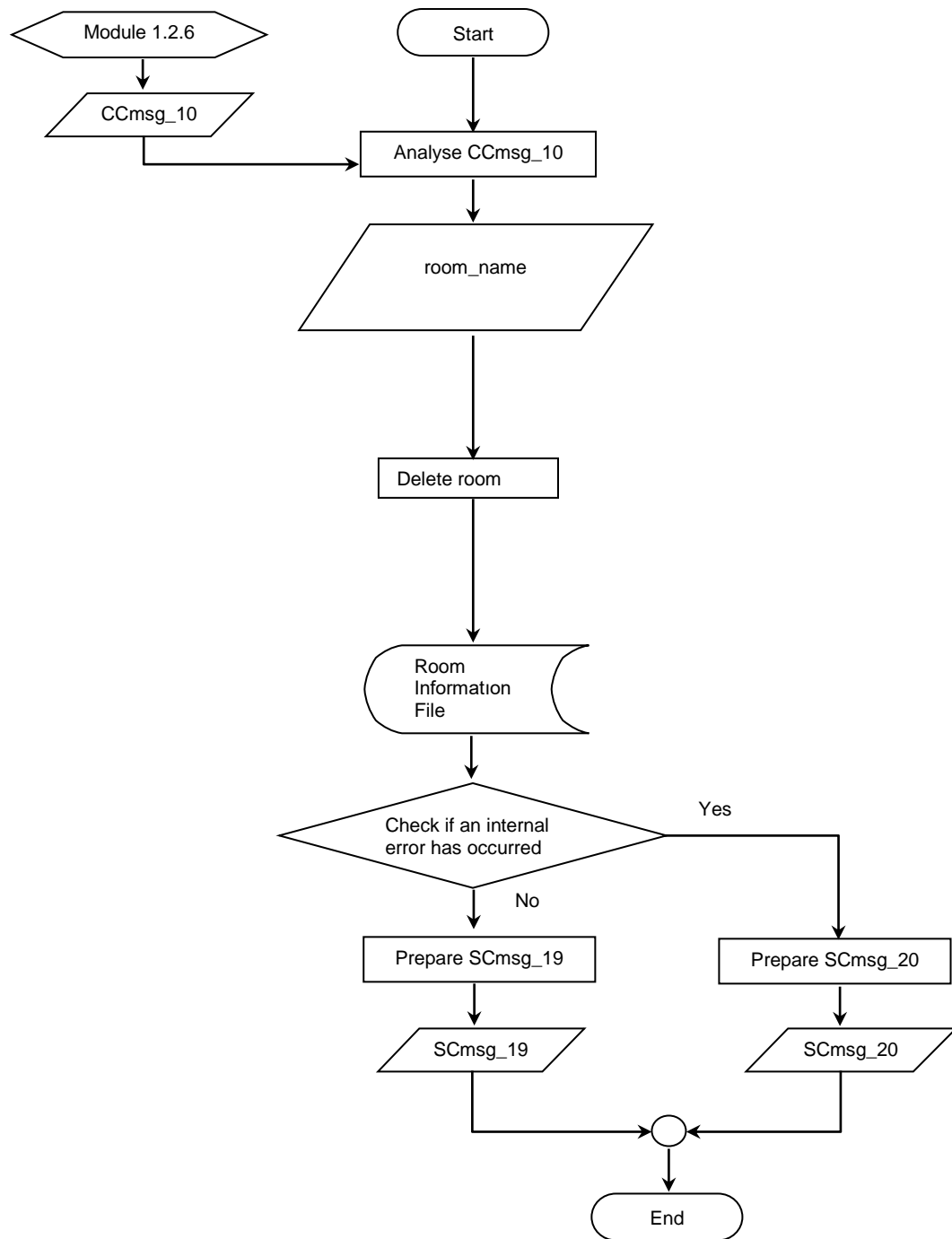
**Figure 40:**Flow Chart of User Deletion for Server software



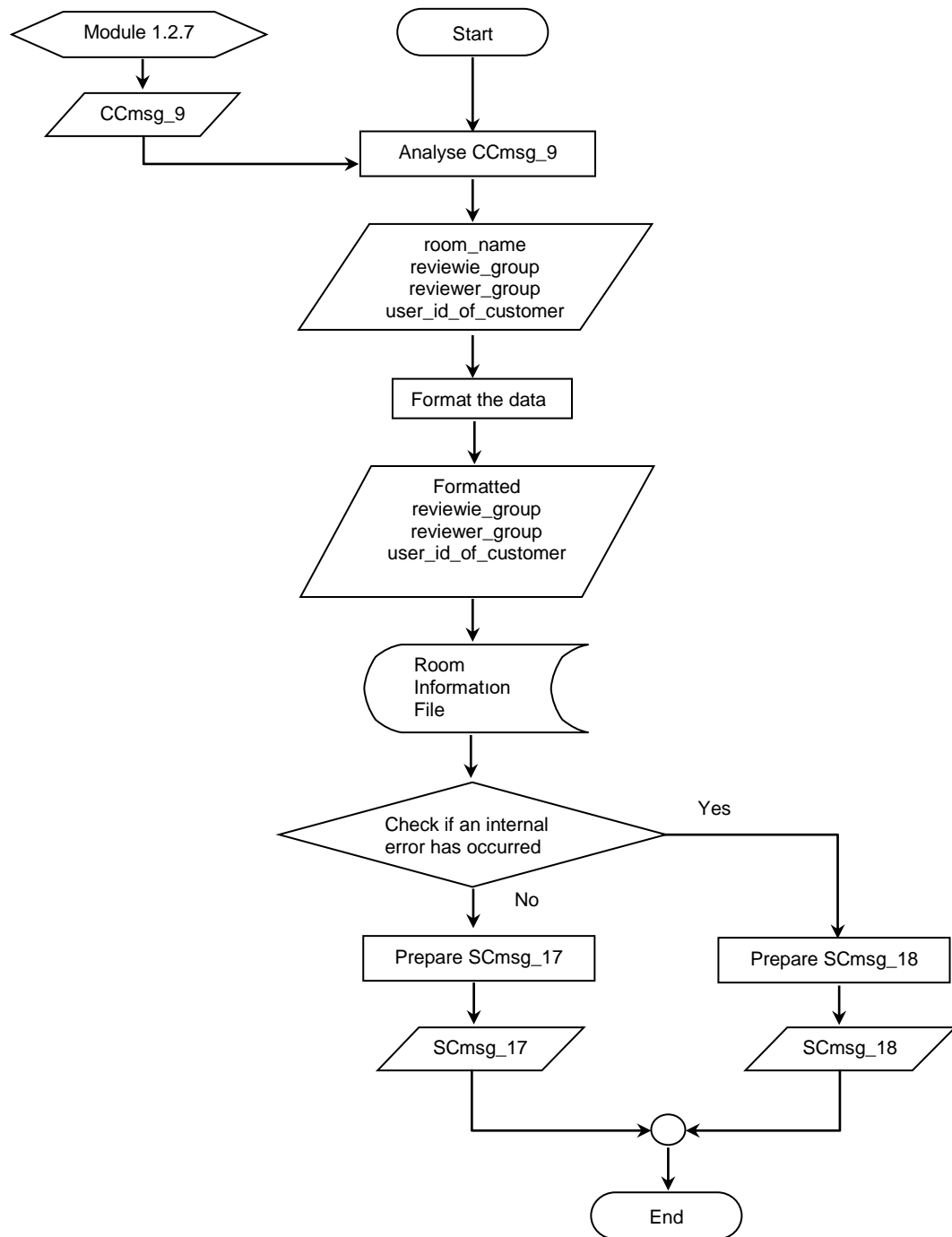
**Figure 41:**Flow Chart of User Update for Server software



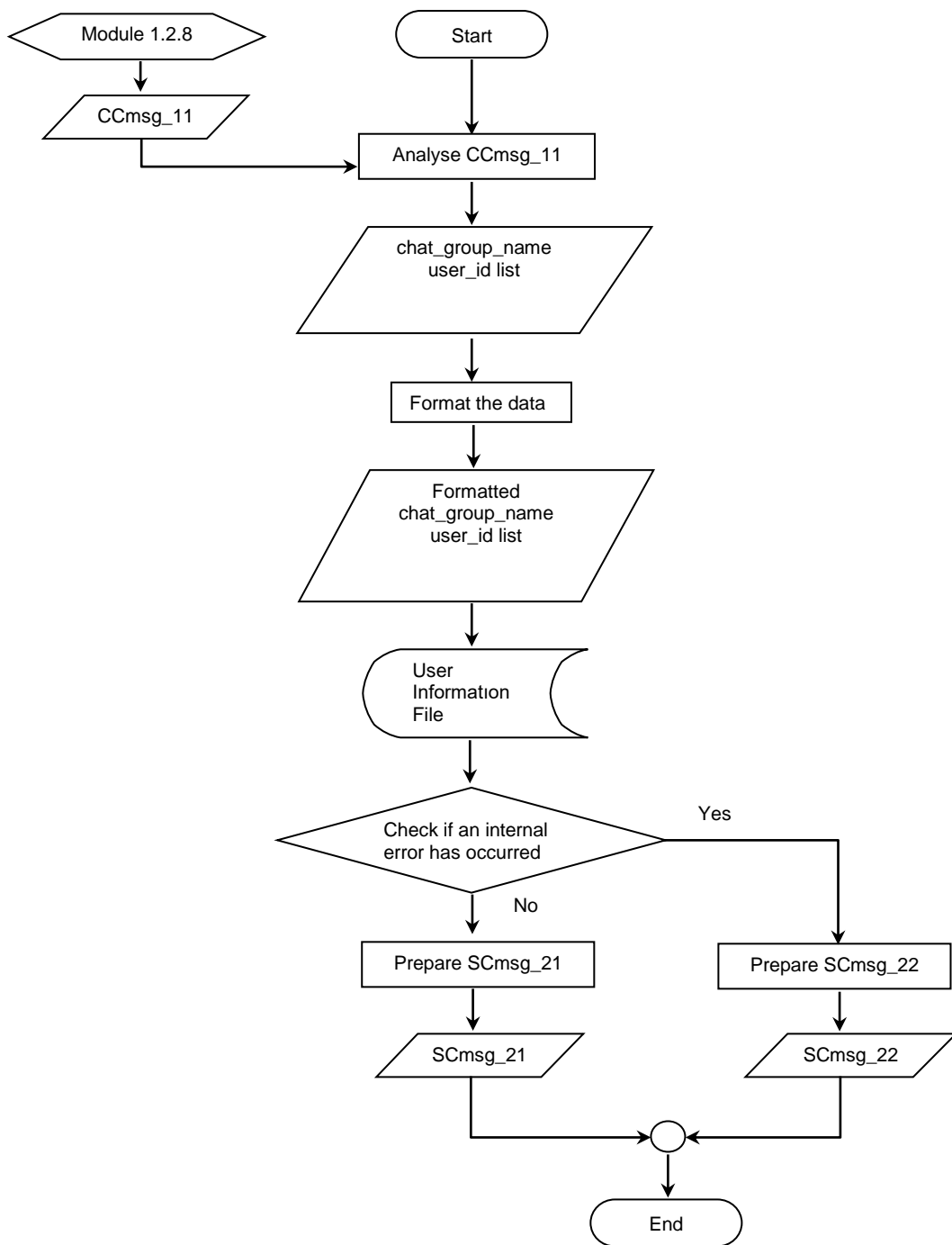
**Figure 42:**Flow Chart of Room Addition for Server software



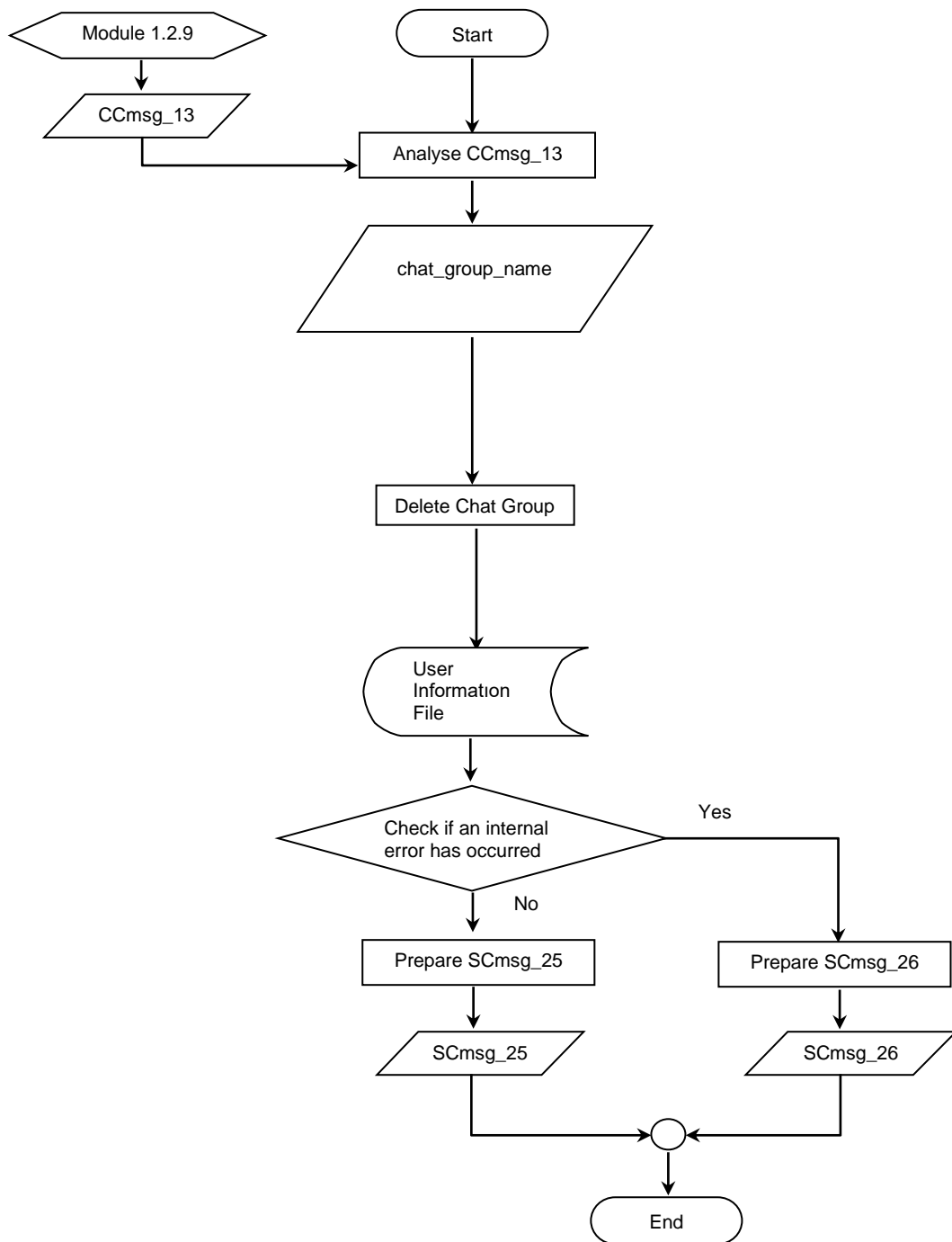
**Figure 43:**Flow Chart of Room Deletion for Server software



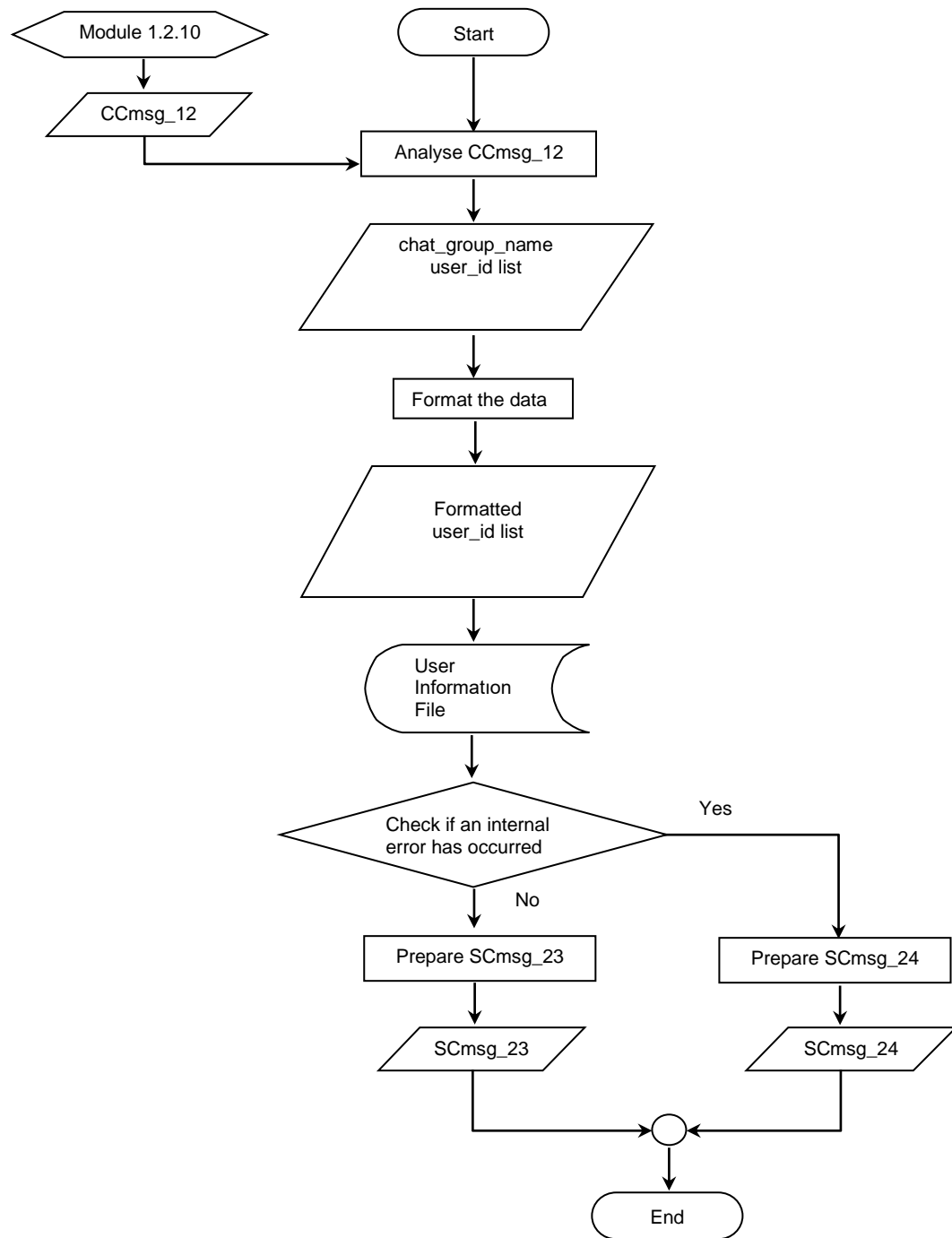
**Figure 44:**Flow Chart of Room Update for Server software



**Figure 45:**Flow Chart of Chat Group Addition for Server software

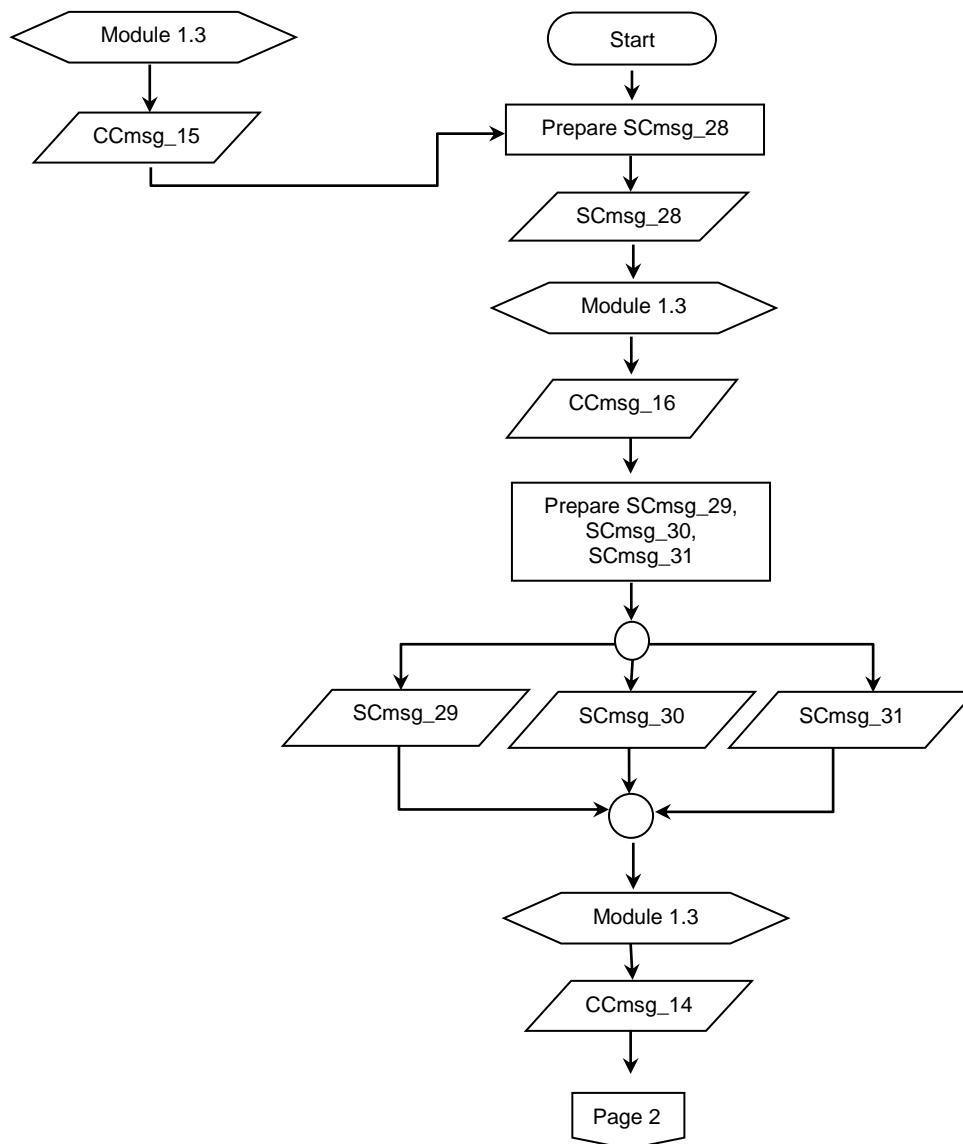


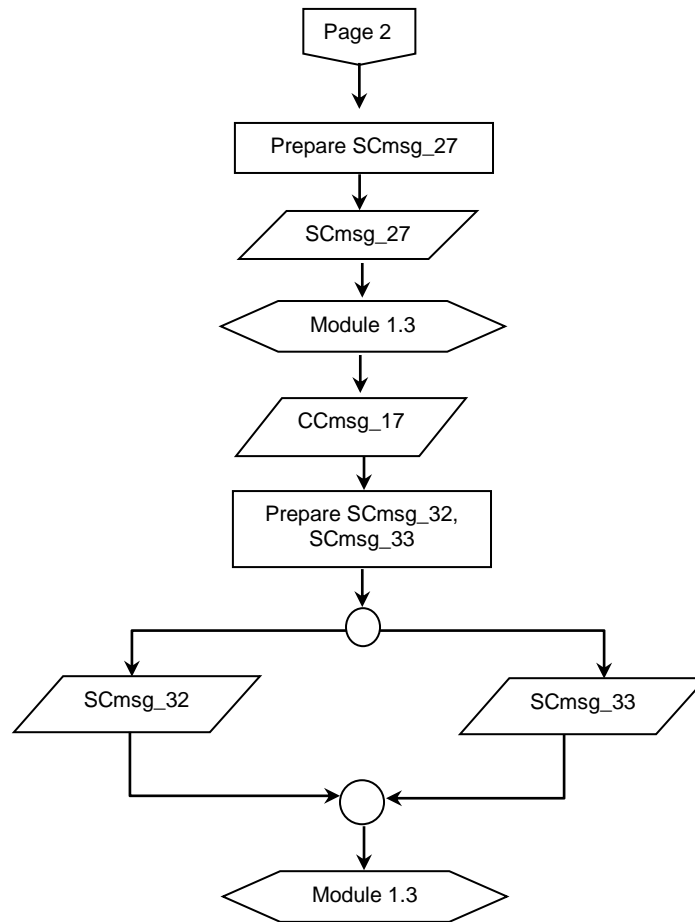
**Figure 46:**Flow Chart of Chat Group Deletion for Server software



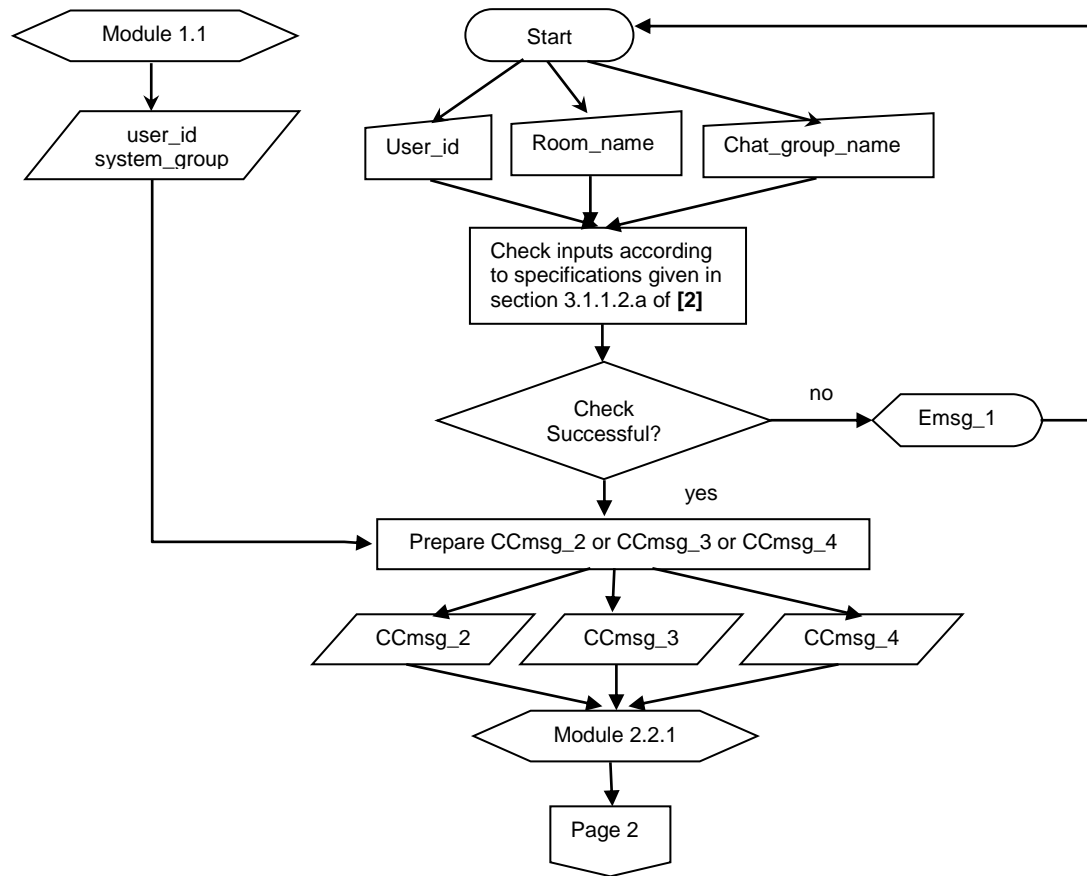
**Figure 47:**Flow Chart of Chat Group Update for Server software

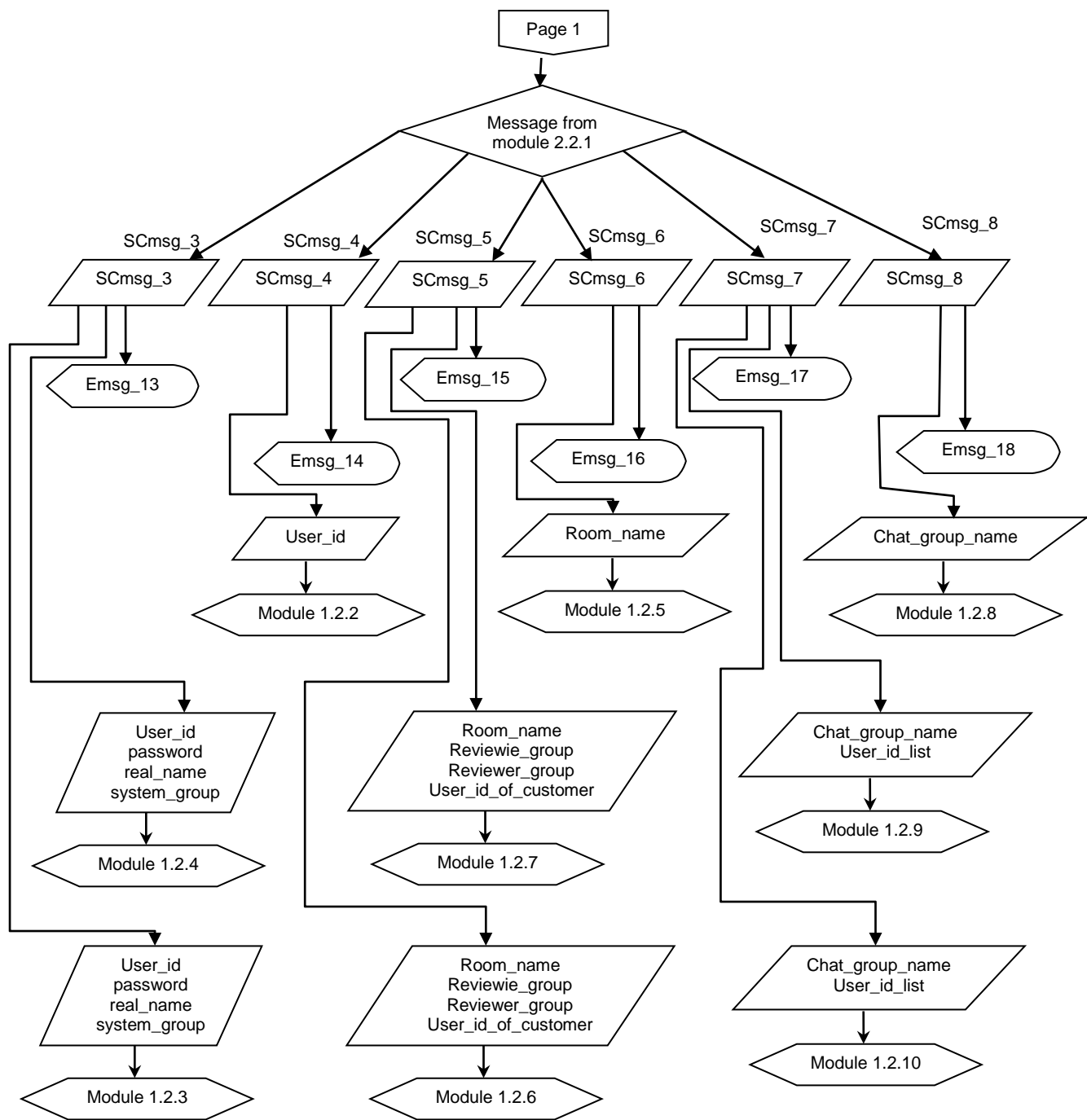




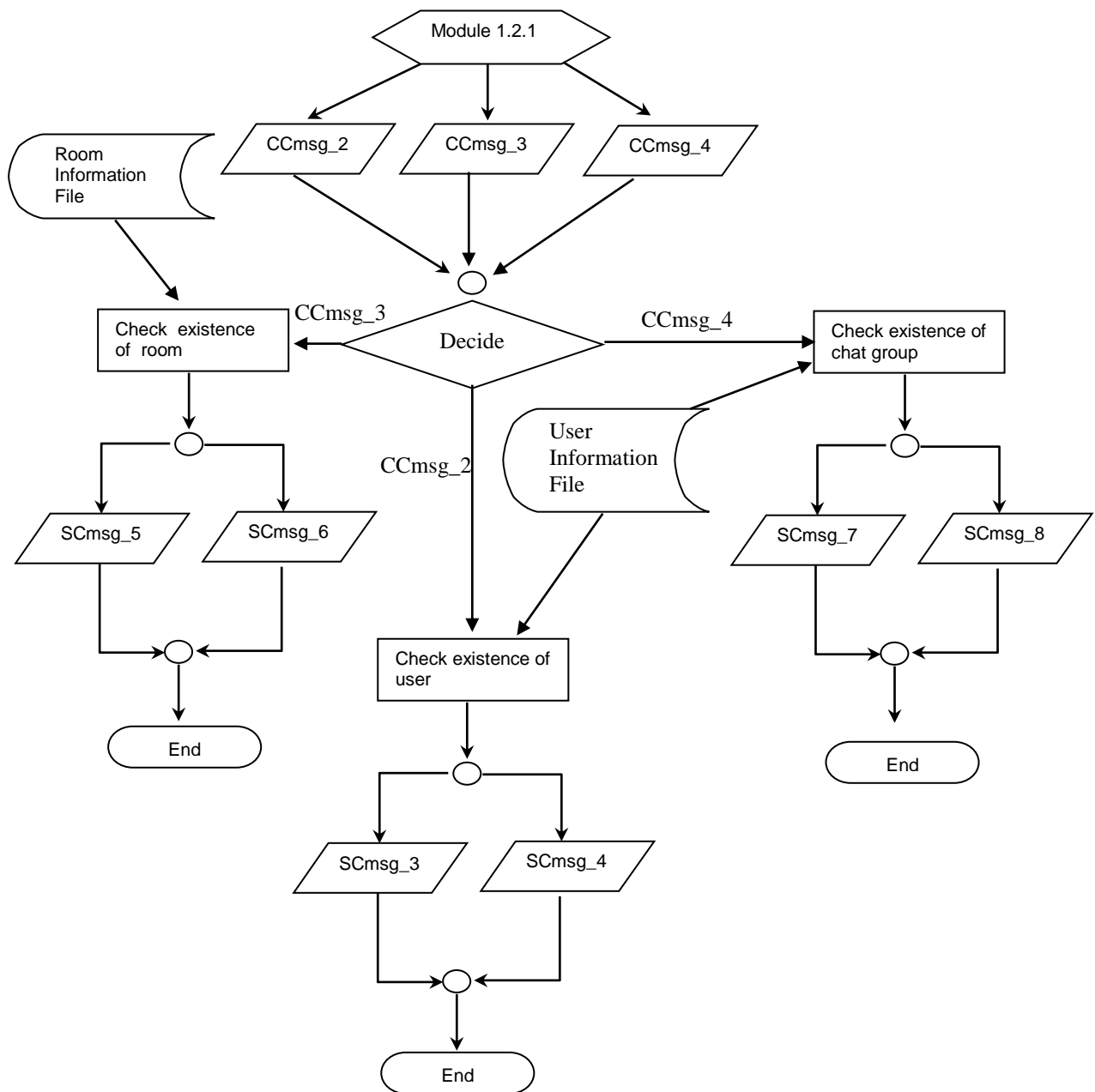


**Figure 48:**Flow Chart of Chat for Server software



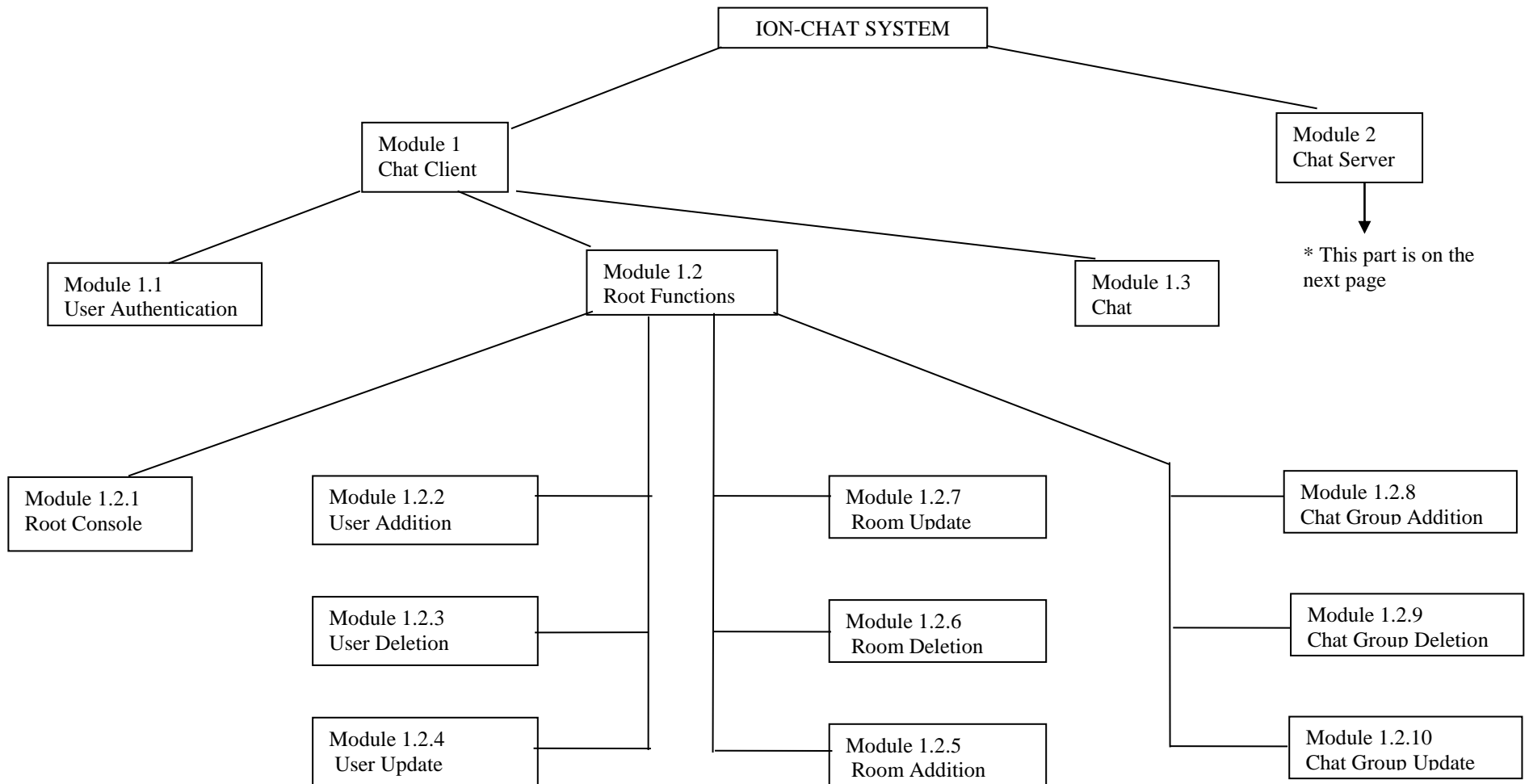


**Figure 49:** Flow Chart of Root Console for Client software

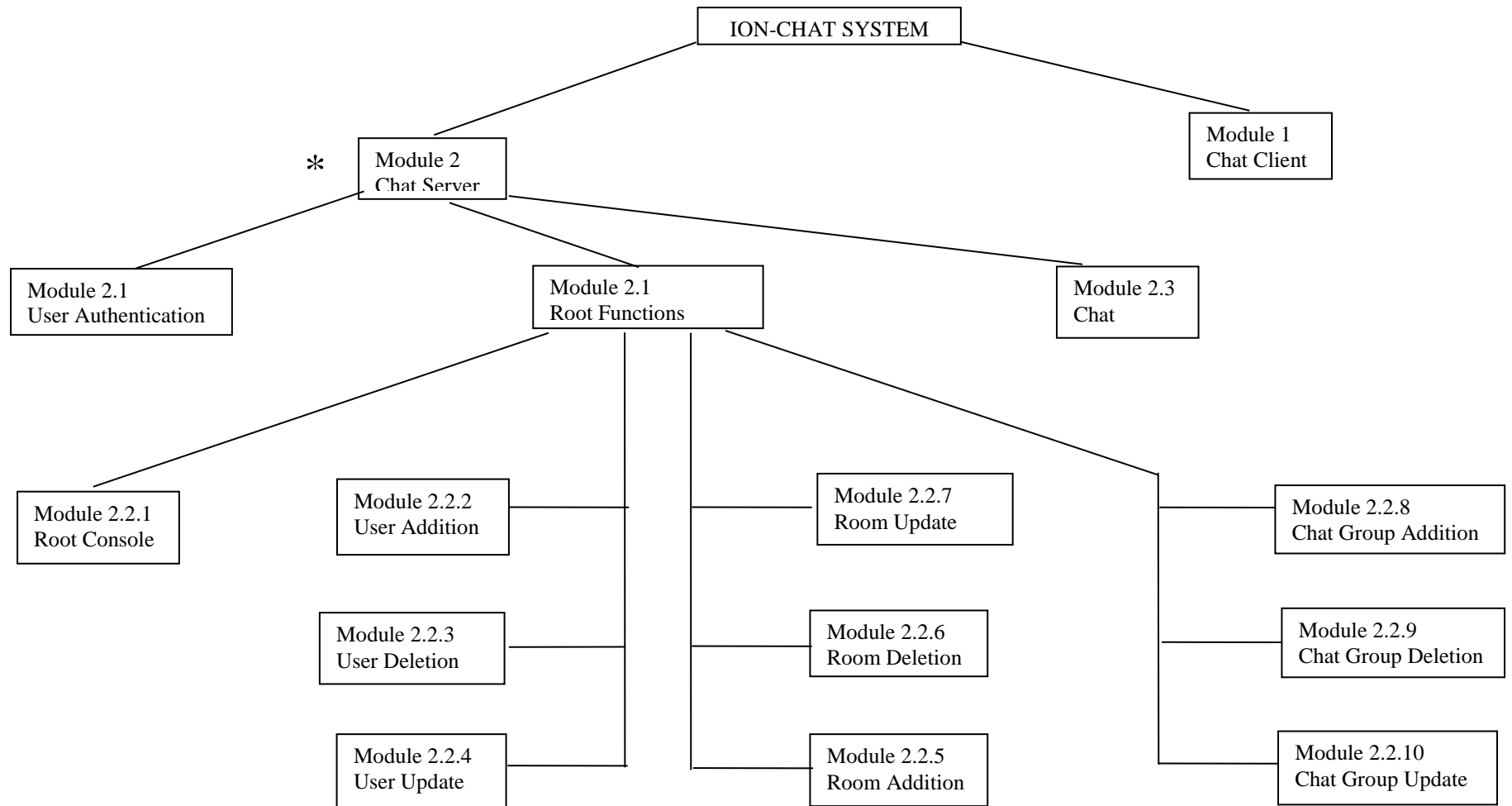


**Figure 50:**Flow Chart of Root Console for Server software

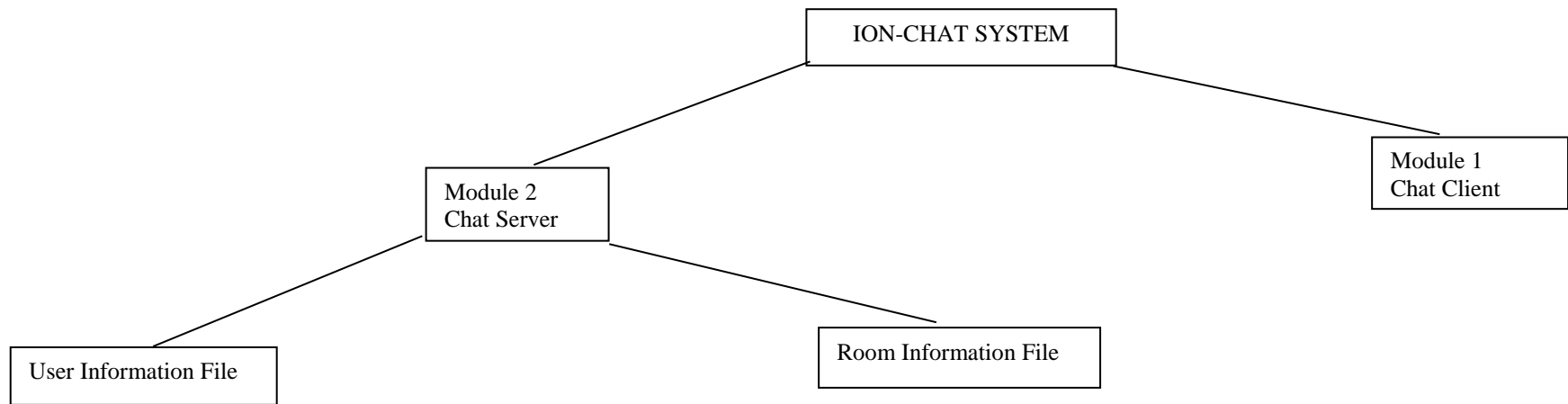
## APPENDIX-G



**Figure 51:** Structure Chart of Chat Client



**Figure 52:** Structure Chart of Chat Server



**Figure 53:**Structure Chart of Information Files



## APPENDIX-H

Code of Parameter Set	Parameter List
c1	String user_id String system_group
c2	String user_id
c3	String user_id String password String real_name String system_group
c4	String user_id String password String real_name String system_group
c5	String room_name
c6	String room_name String reviewie_group String reviewer_group String user_id_of_customer
c7	String room_name String reviewie_group String reviewer_group String user_id_of_customer
c8	String chat_group_name
c9	String chat_group_name String user_id_list
c10	String chat_group_name String user_id_list
c11	String user_id String reviewie_group_room_name String reviewer_group_room_name String system_group

**Table 2:** Code Table

## APPDENIX-I

Code of Method	Method Name	Method Interface	Method Belongs To
f1	method1	method1(String user_id, String reviewie_group_room_name, String reviewer_group_room_name, String system_group)	Module 1.3
f2	method2	method2(String user_id)	Module 1.2.2
f3	method3	method3(String user_id, String password, String real_name, String system_group)	Module 1.2.3
f4	method4	method4(String user_id, String password, String real_name, String system_group)	Module 1.2.4
f5	method5	method5(String room_name)	Module 1.2.5
f6	method6	method6(String room_name, String reviewie_group, String reviewer_group, String user_id_of_customer)	Module 1.2.6
f7	method7	method7(String room_name, String reviewie_group, String reviewer_group, String user_id_of_customer)	Module 1.2.7
f8	method8	method8(String chat_group_name)	Module 1.2.8
f9	method9	method9(String chat_group_name, String user_id_list)	Module 1.2.9
f10	method10	method10(String chat_group_name, String user_id_list)	Module 1.2.10

**Table 3: Method Table**