

Chat room API -team dragonBall

11/20/2019

<http://chatapp-team-dragonball.herokuapp.com/>

Team members

Jiangguo Zhang (team lead)

Enze Zhong (dev)

Can Sun (dev)

Yifan Wang (tech lead)

Sibo Wang (doc lead)

Tingting Zhou (dev)

Shunda Huang (dev)

Contents



Using protocol



Design pattern

UML diagram



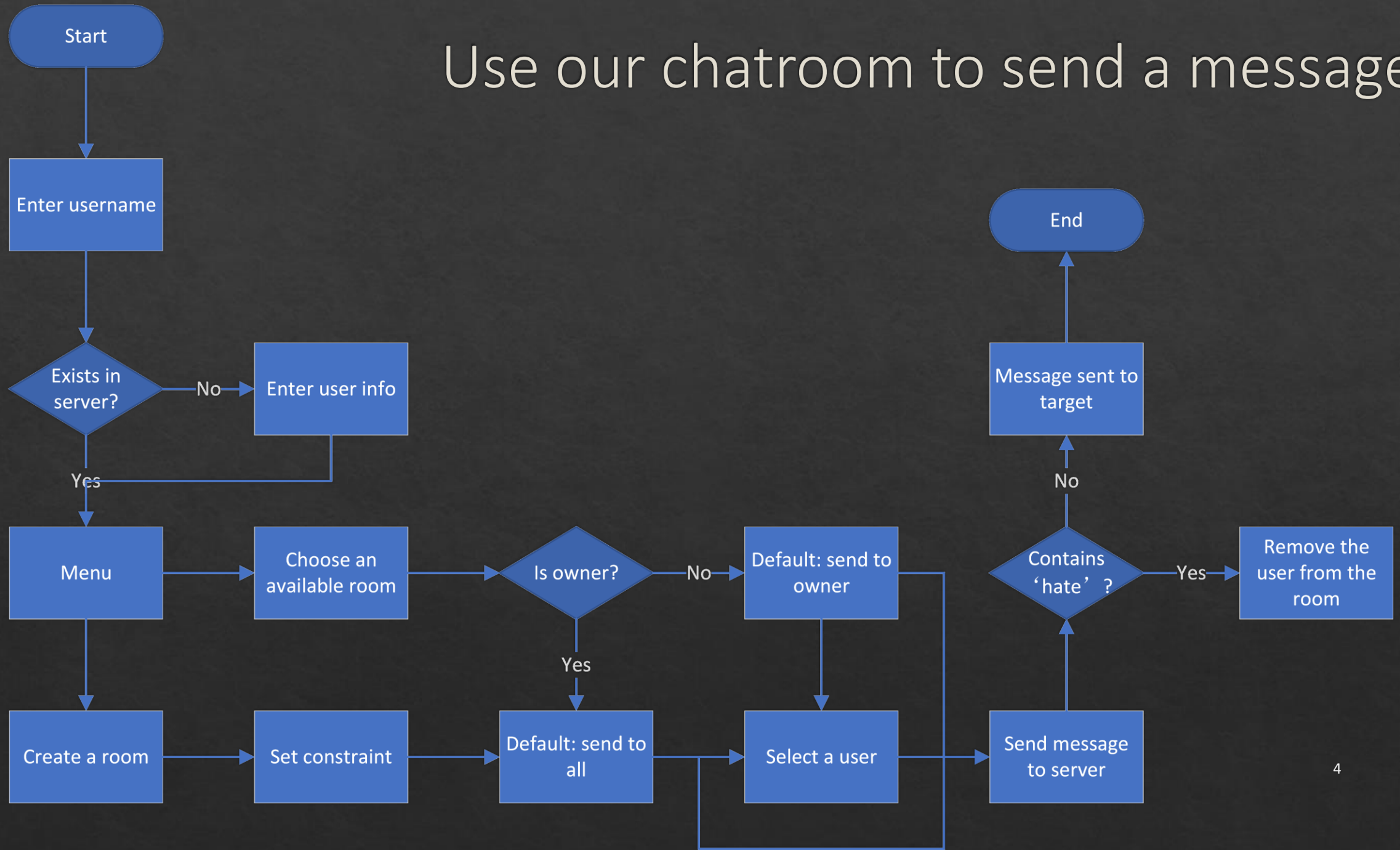
API

Interfaces/abstract classes

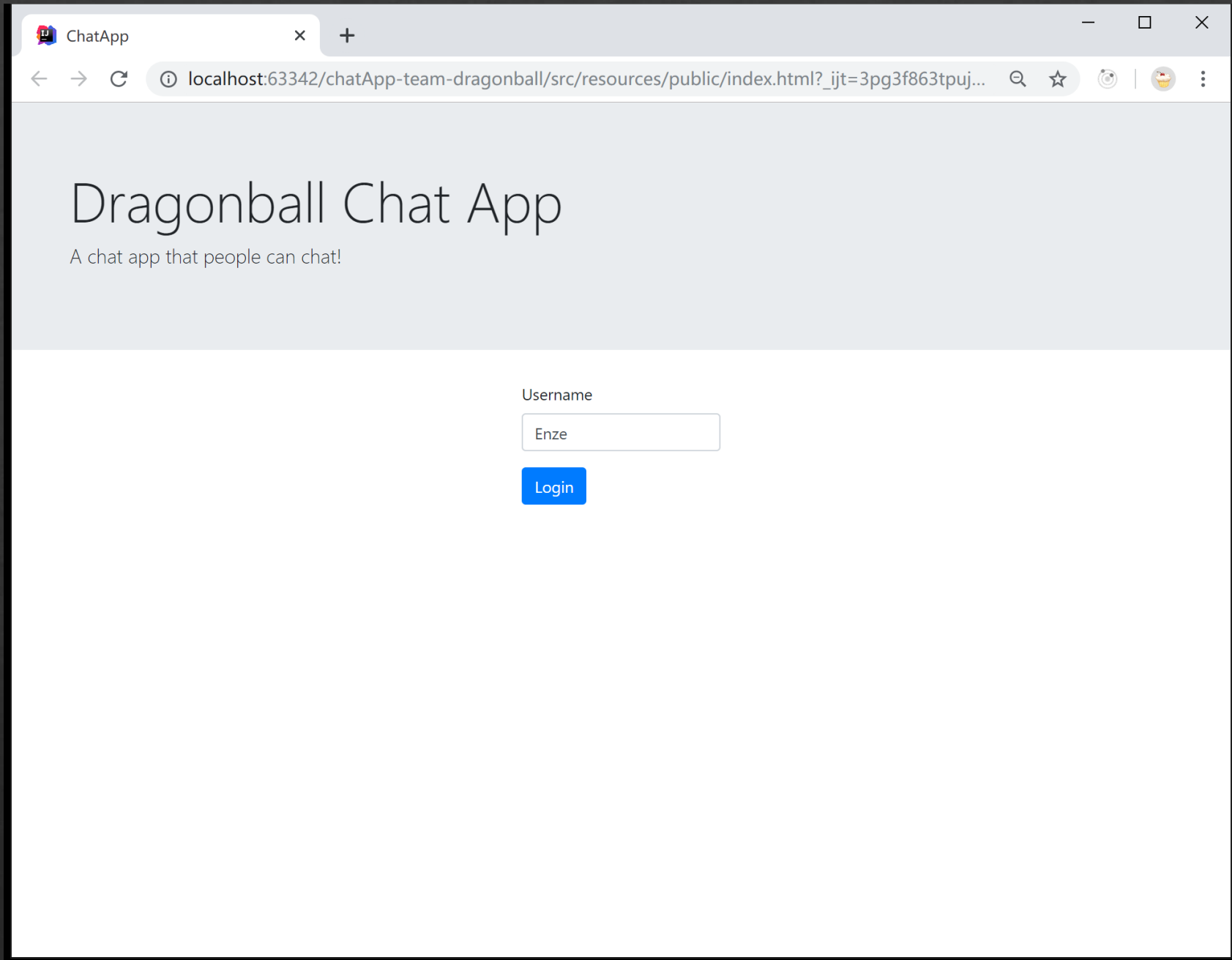
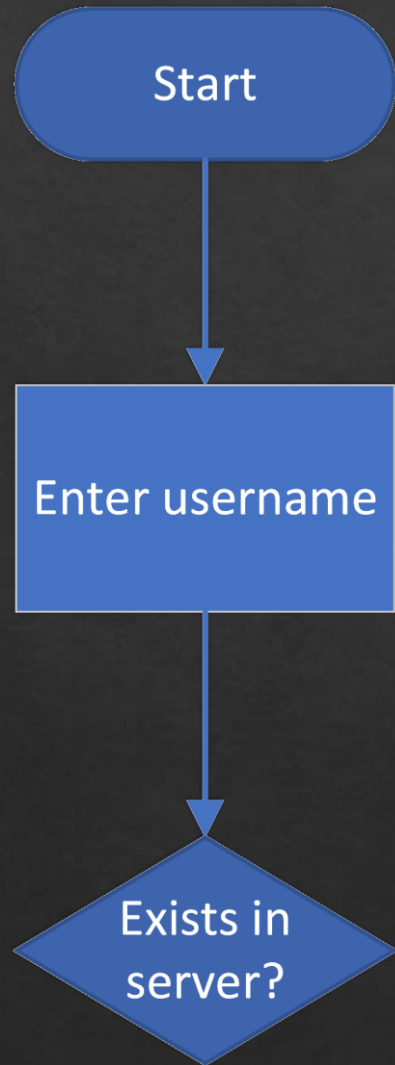


Demo

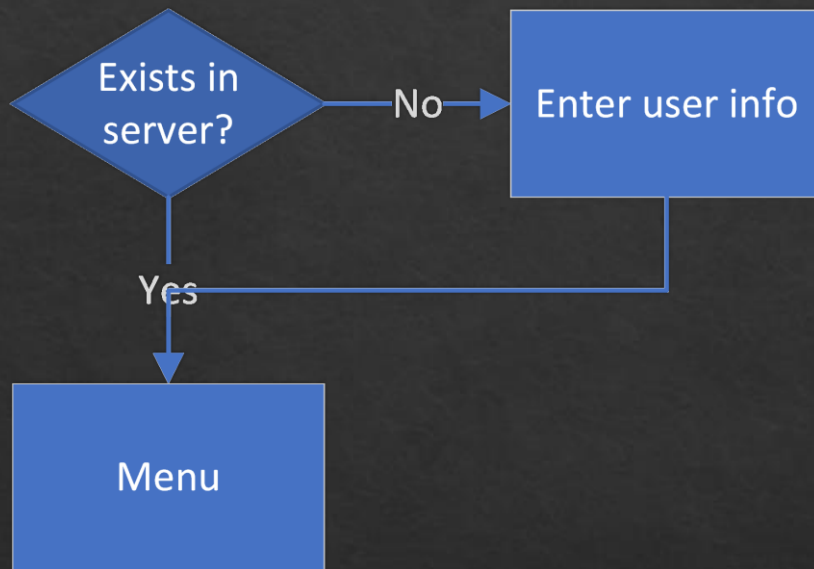
Use our chatroom to send a message



Login



User profile



ChatApp

localhost:63342/chatApp-team-dragonball/src/resources/public/index.html?_ijt=3pg3f863tpuj...

Dragonball Chat App

A chat app that people can chat!

Username

Age

Area

School

Login

6

Room profile

ChatApp

localhost:63342/chatApp-team-dragonball/src/resources/public/main.html

My rooms

room 1

room 2

room 3

new edit leave leave all

Available rooms

room 4

room 5

room 6

Rice Room

Room Name

Rice Room

Minimum Age

9

Maximum Age

99

Areas

☐ North America

☐ South America

☒ Asia

☒ Rice University

☐ Peking University

☒ UESTC

☐ Harvard

☐ Duke

☐ Nanjing

☐ Wuhan

Close

Save changes

Users

Mack

Risa

Can

Menu

Choose an available room

Is owner?

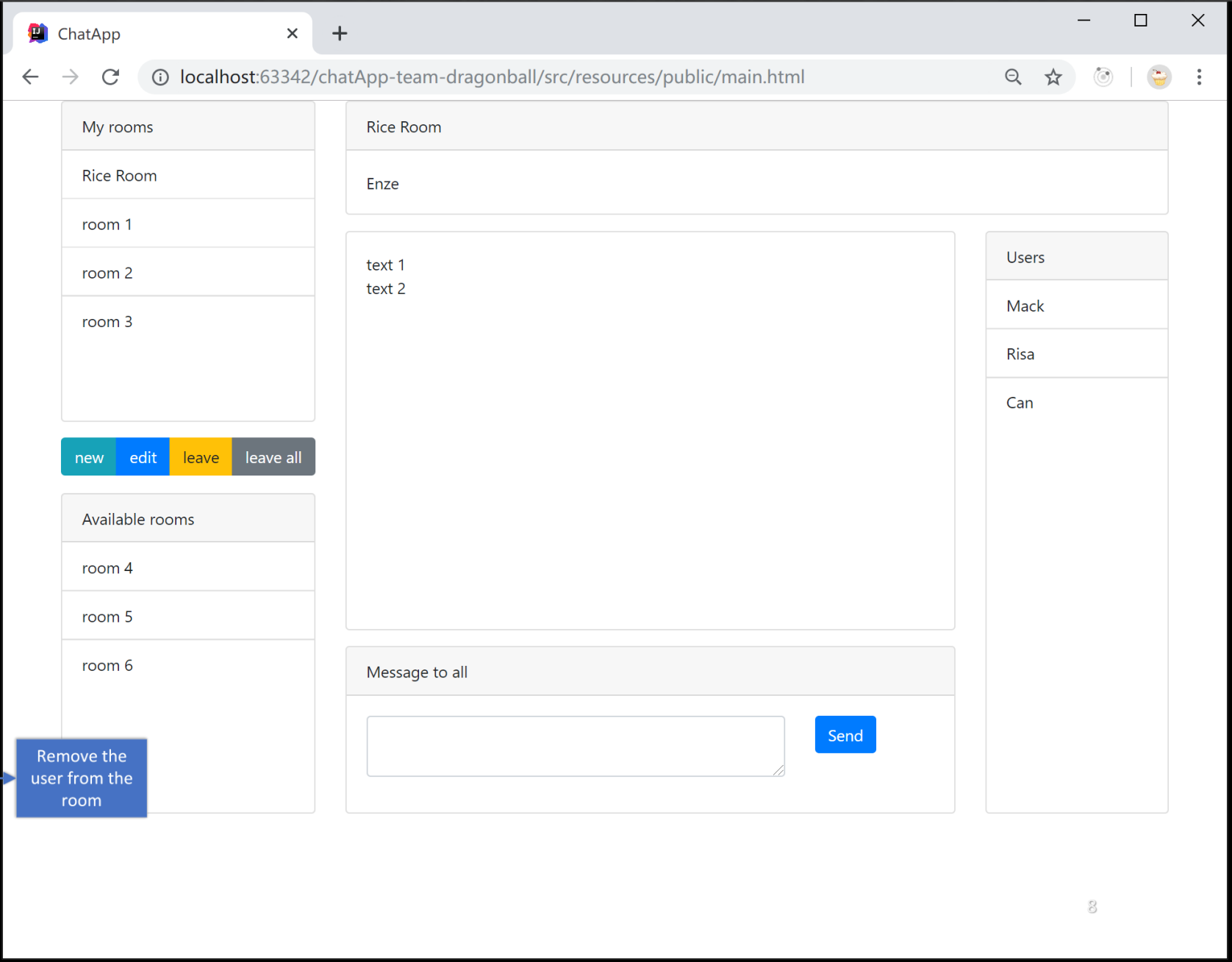
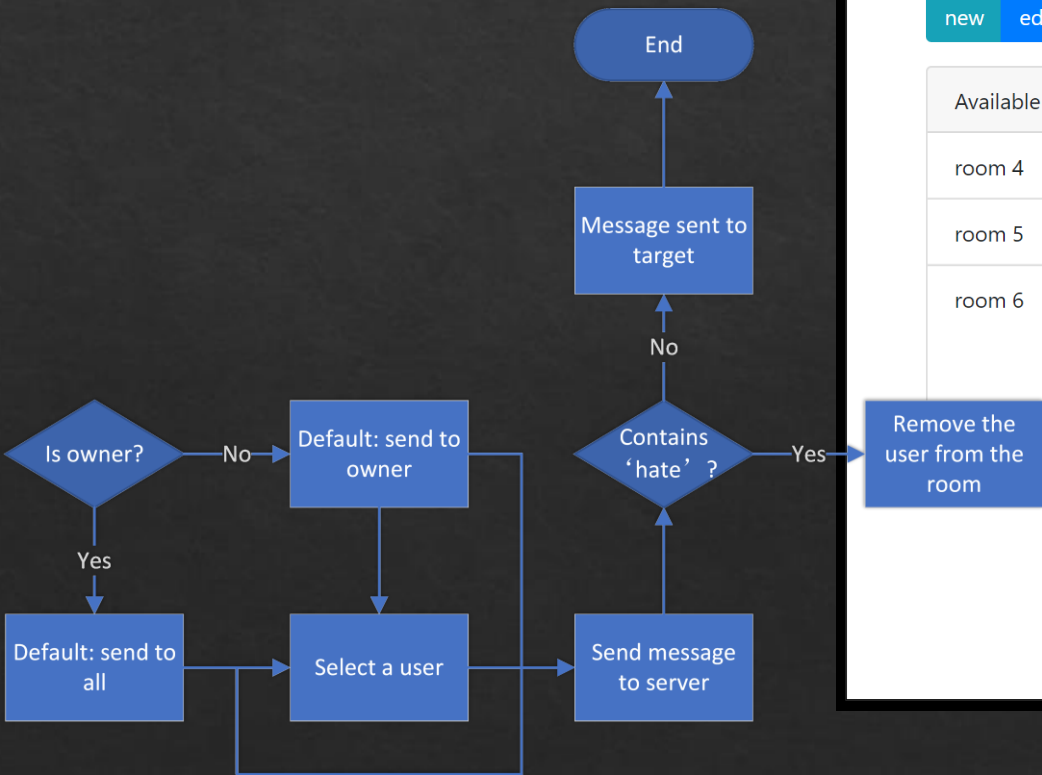
Default: send to owner

Create a room

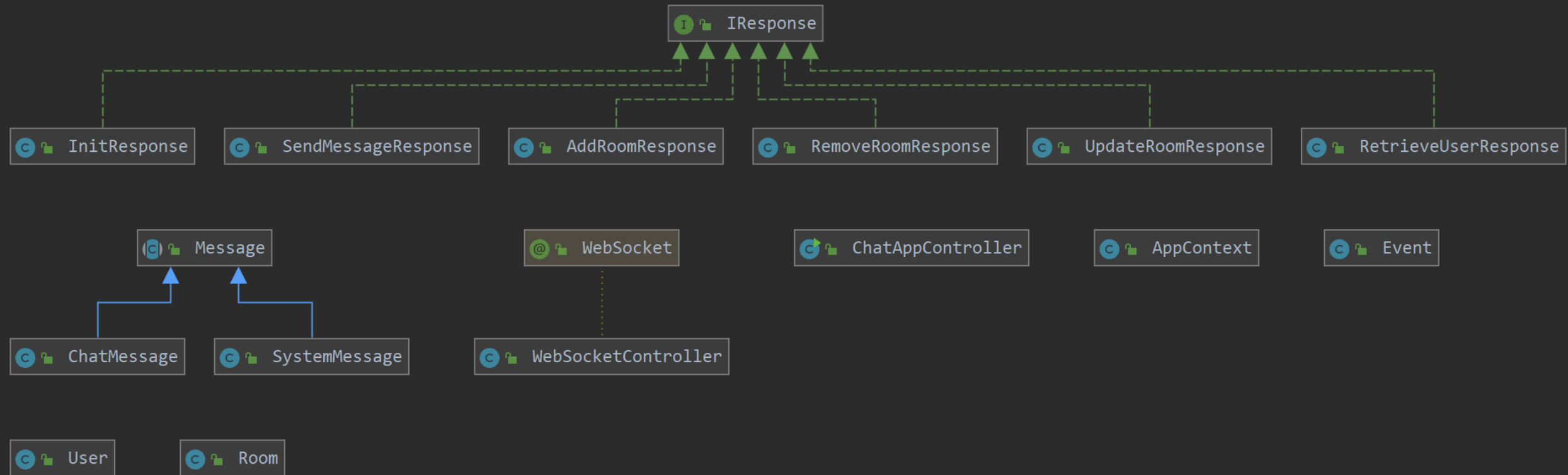
Set constraint

Default: send to all

Chat room



UML diagram



Design Patterns



MVC design pattern



Union design pattern



Strategy design pattern



Singleton design pattern



Model

AppContext, Responses, etc.



View

GUI

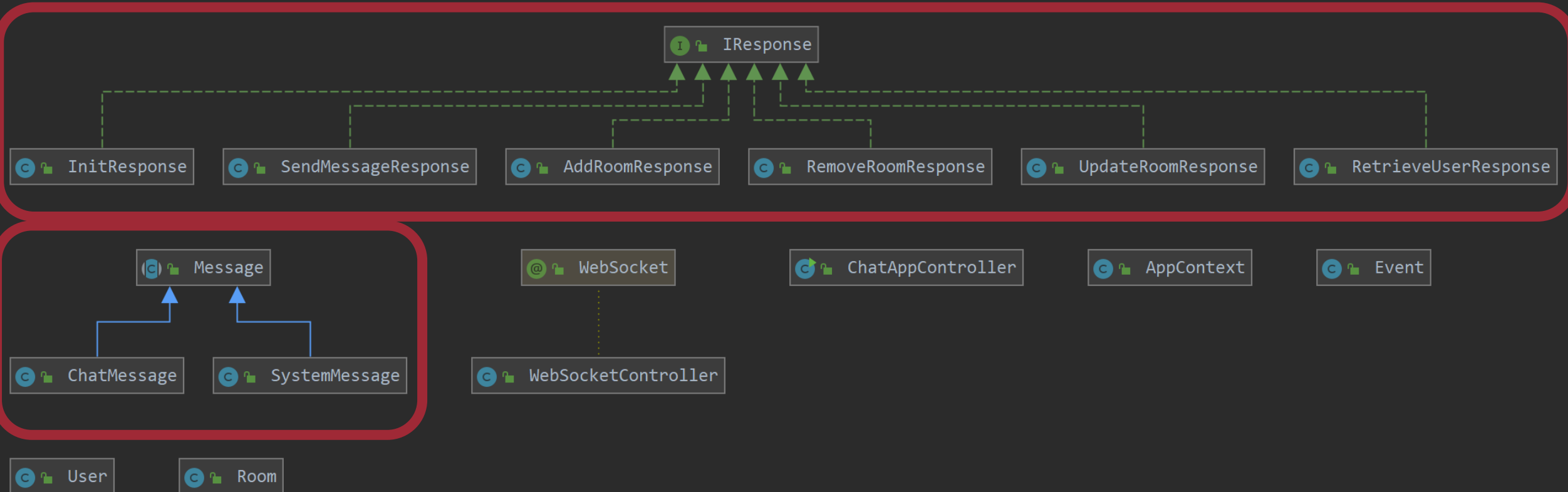


Controller

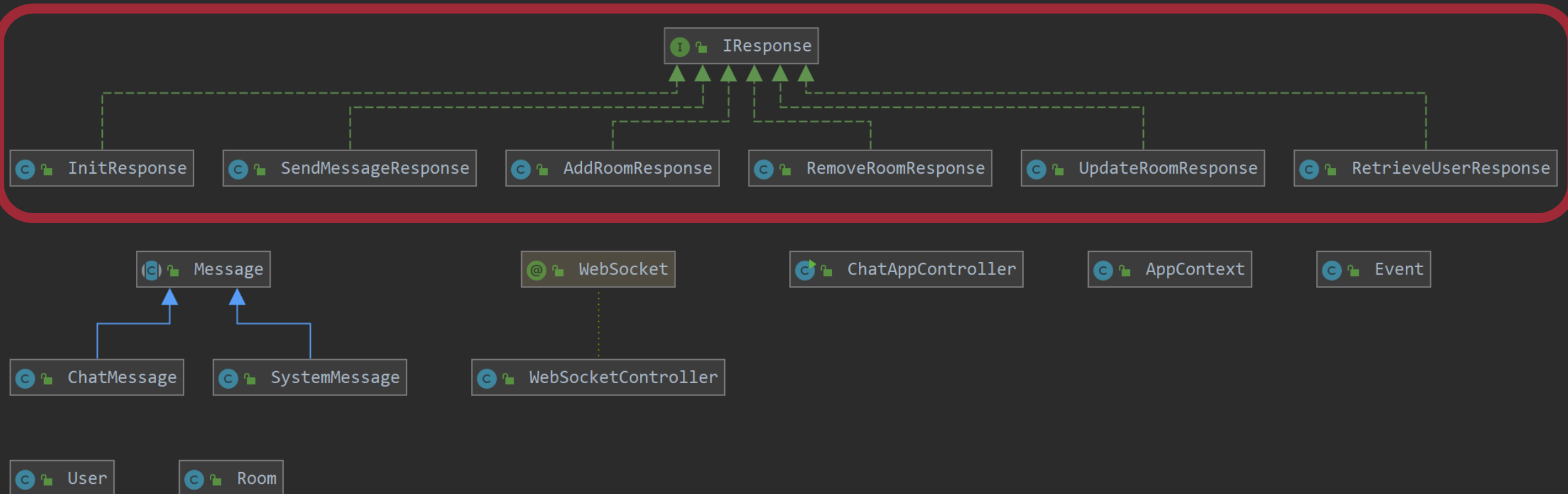
ChatAppController,
WebSocketController

MVC design pattern









































Union Design Pattern



Strategy Design Pattern



Singleton Design Pattern

 	AppContext	
 	only	AppContext
 	userMap	Map<String, User>
 	roomMap	Map<Integer, Room>
 	getOnly()	AppContext
 	addUser(User)	void
 	addRoom(Room)	void
 	removeUser(User)	void
 	removeRoom(Room)	void
 	joinRoom(int, String)	Room
 	sendMessage(int, String, String, String)	Room
 	leaveRoom(int, String)	Room
 	updateRoom(int, String, int, int, ArrayList<String>, ArrayList<String>)	Room
 	createRoom(String, int, int, ArrayList<String>, ArrayList<String>)	Room
 	getQualifiedUsers(int)	ArrayList<User>
 	detectHate(String, Room, User)	void
 	deleteRoomIfEmpty(Room)	void
 	sendSystemMessage(Room, User, Event)	void
 	kickUnqualifiedUsers(Room)	void
 	users	ArrayList<User>

API



Login

Retrieve a user

Create a user



Join room



Edit room

Create a room

Update a room



Send message

To target member

To all members

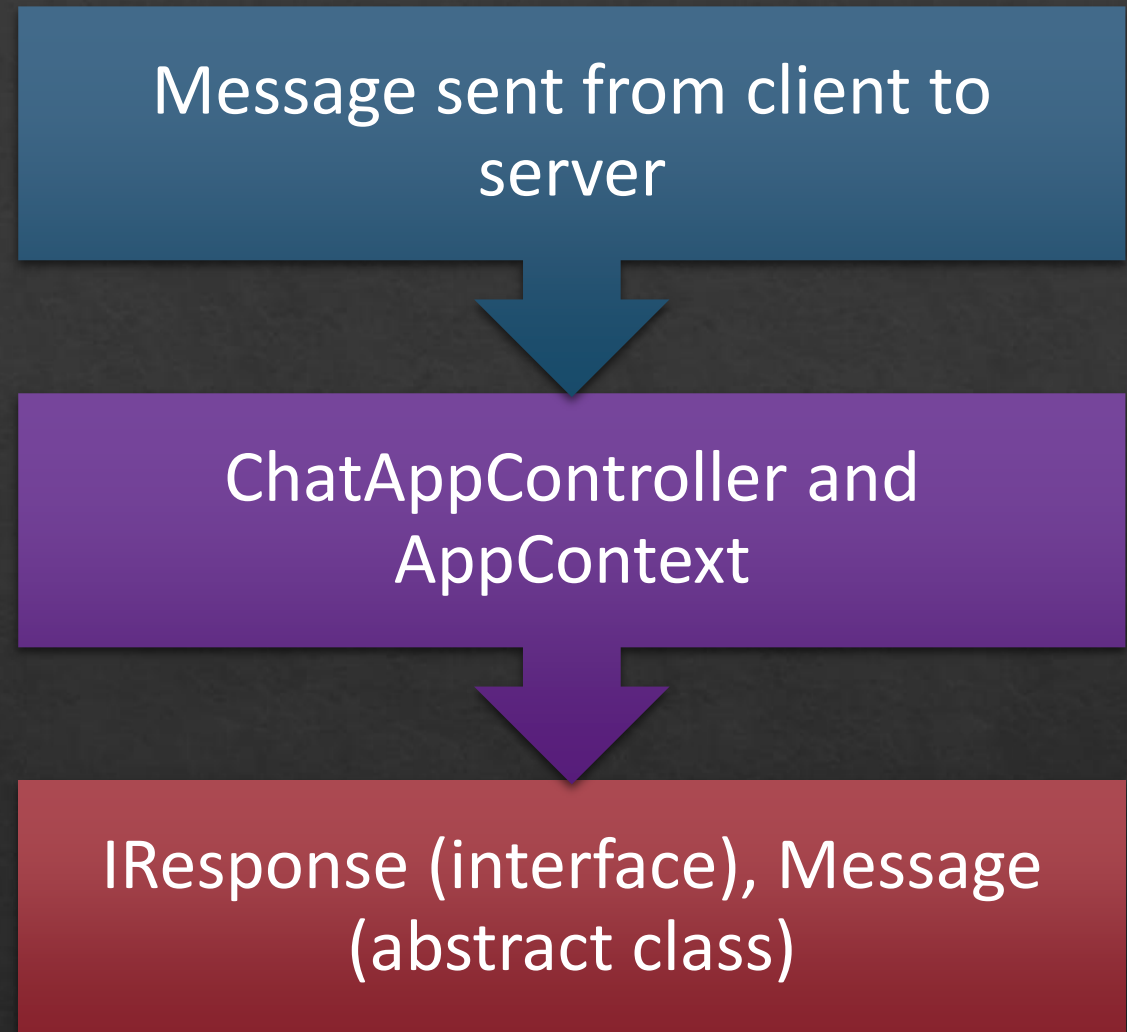


Leave room

Leave selected room

Leave all rooms

Legend



Login

RETRIEVE USER: userID



handleRetrieveUser



RetrieveUserResponse: user

CREATE USER: username, age,
area, school

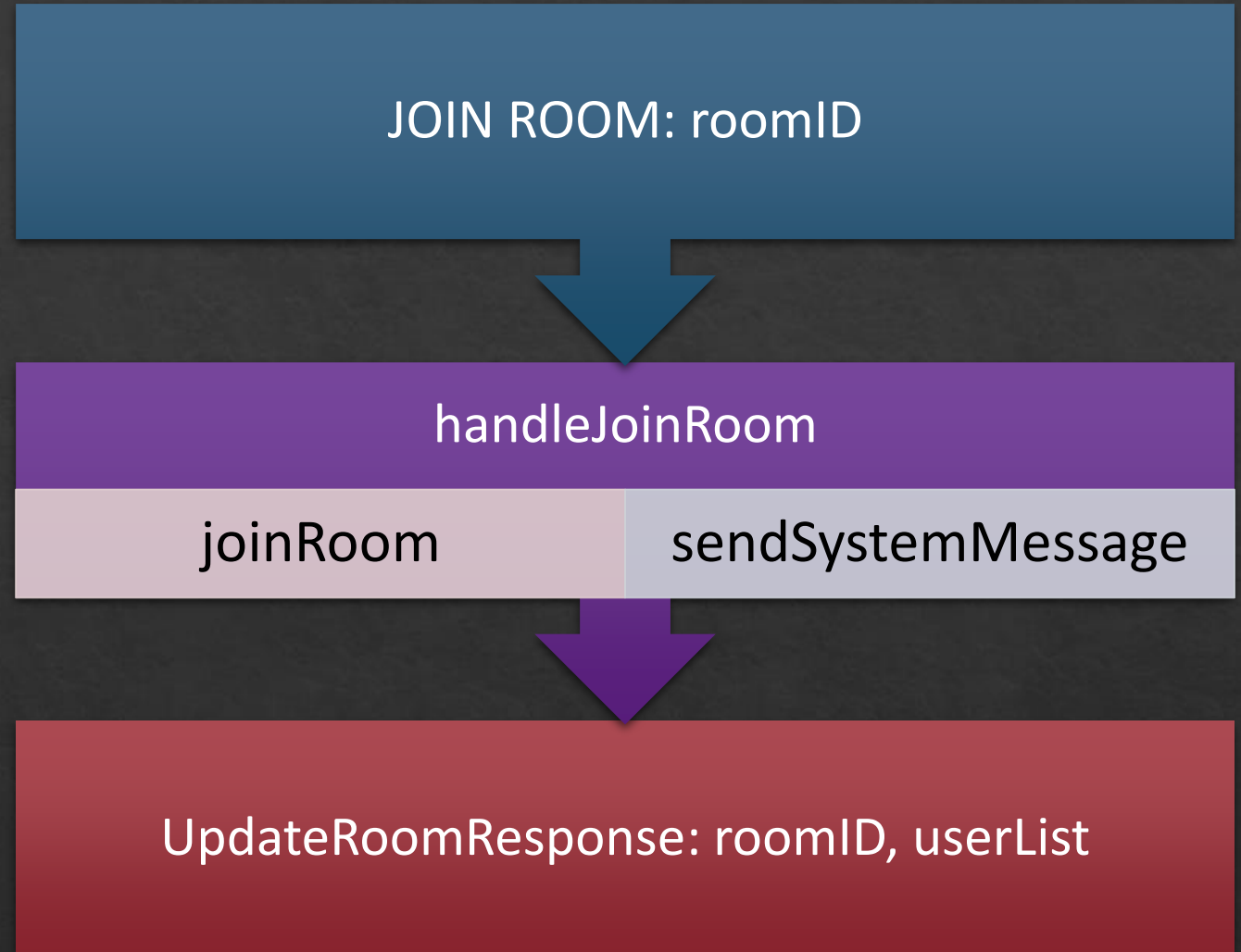


handleCreateUser



RetrieveUserResponse: user

Join Room



Edit Room

CREATE ROOM: ageMin, ageMax, areas,
schools



handleCreateRoom

createRoom

getQualifiedUsers



AddRoomResponse: room

UPDATE ROOM: roomId, ageMin, ageMax, areas,
schools



handleUpdateRoom

updateRoom

kickUnqualifiedUsers

sendSystemMessage



RemoveRoomResponse: roomIDs

Send Message

SEND MESSAGE: roomId, receiverID, text



handleSendMessage

detectHate

- handleLeaveRoom

sendMessage



SendMessageResponse: roomId, message

SEND MESSAGE TO ALL: roomId



handleSendMessageToAll

handleSendMessage



SendMessageResponse: roomId, message

Leave Room

LEAVE ROOM: roomId



handleLeaveRoom

leaveRoom

deleteRoomIfEmpty

sendSystemMessage



UpdateRoomResponse: roomId, userList

LEAVE ALL ROOMS



handleLeaveAllRooms

handleLeaveRoom



UpdateRoomResponse: roomId, userList

Dragonball Chat App

A chat app that people can chat!

Demo

<http://chatapp-team-dragonball.herokuapp.com/>

Username

Login



Thank you!