C🍪🍪KIE-APP

Convergent Capstone Design 2
Week 12 - Progress Report

Group 6 Park JongBeum & Baek SeungHeon, November 20th 2023

Progress

Maps

Maps Tab

- "MapViewModel" is too complex
- so we separate between Logics and Variables
    - Variables => MapViewModel
    - Business Logics => MapService

Maps Tab
MapViewModel

```dart
class MapViewModel with ChangeNotifier {
  // mapController
  late GoogleMapController mapController;

  // mapLog
  List<MarkerInfo> _mapLog = [];
  List<MarkerInfo> get mapLog => _mapLog;
  set mapLog(List<MarkerInfo> value) {
    _mapLog = value;
    notifyListeners();
  }
```

Maps Tab

MapService

```
void moveCamera(LatLng location) {
  context
      .read<MapViewModel>()
      .mapController
      .animateCamera(CameraUpdate.newLatLngZoom(location, 16.0));
}
```

Maps Tab

Image Caching & handling

- The profile image uses a "NetworkImage" type.
- This may result in exception situations and a waste of resources
- So we created logics that could solve both problems
  - Image exception handling
  - Image caching

Image Exception Handling

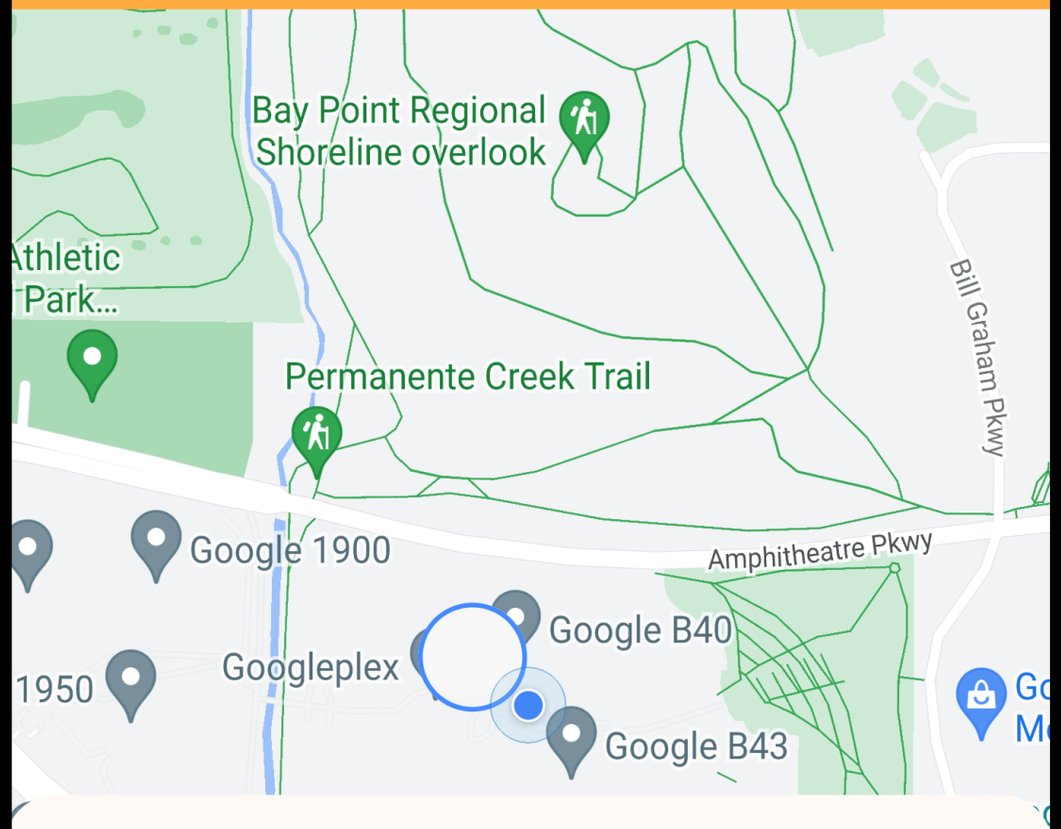
```
GET /uploads/testid2/testid2.profile.1695459635937.jpeg
11.226 ms - 1139
GET /uploads/testid3/testid3.profile.1695459682540.jpeg
10.580 ms - 1139
GET /uploads/testid4/testid4.profile.1695459707258.jpeg
10.493 ms - 1139
```

Image Exception Handling

```dart
Future<String> getNetworkImage(String url) async {
  bool isValidUrl = Uri.tryParse(url)?.isAbsolute ?? false;
  if (isValidUrl) {
    return url;
  } else {
    return '''
    https://img.freepik.com/free-photo/
    abstract-surface-and-textures-of-white-concrete
    -stone-wall_74190-8189.jpg?w=740&t=st=1699954803
    ~exp=1699955403~hmac=b06fdddf22160522cb93f9a15950
    4e331c773055cf433fa5d5156348b4c49782
    ''';
  }
}
```

1:38

C🍪🍪KIE



현재 접속 중인 친구

test1 43 m

Image Caching

```dart
Future<File> getCachedImage(String url) async {
  File imageFile = await DefaultCacheManager().getSingleFile(url);
  return imageFile;
}
```

```dart
String imageUrl = await getNetworkImage(user.profile.image.toString());
File imageFile = await getCachedImage(imageUrl);
```

Maps Tab
Location Sharing

- Location sharing can be controlled via Dial Button
- The marker appears only when both friends share a location

4:55 LTE

C🍪🍪KIE

Amphitheatre Pkwy

Google B40

Googleplex
Google's large
global headquarters

친구 찾기

위치 공유

Google B43

E Charleston Rd

Charleston Rd

Balance - By
Albert Guibara

Alta Ave

Google Mountain
View 1098 Alta

C🍪🍪KIE

4:55

C🍪🍪KIE

Amphitheatre Pkwy

Google B40

Googleplex
Google's large
global headquarters

친구 찾기

공유 해제

Google B43

E Charleston Rd

Charleston Rd

Balance - By
Albert Guibara

Alta Ave

Google Mountain
View 1098 Alta

C🍪🍪KIE

5:06

C🍪🍪KIE

현재 접속 중인 친구

현재 접속 중인 친구가 없습니다.

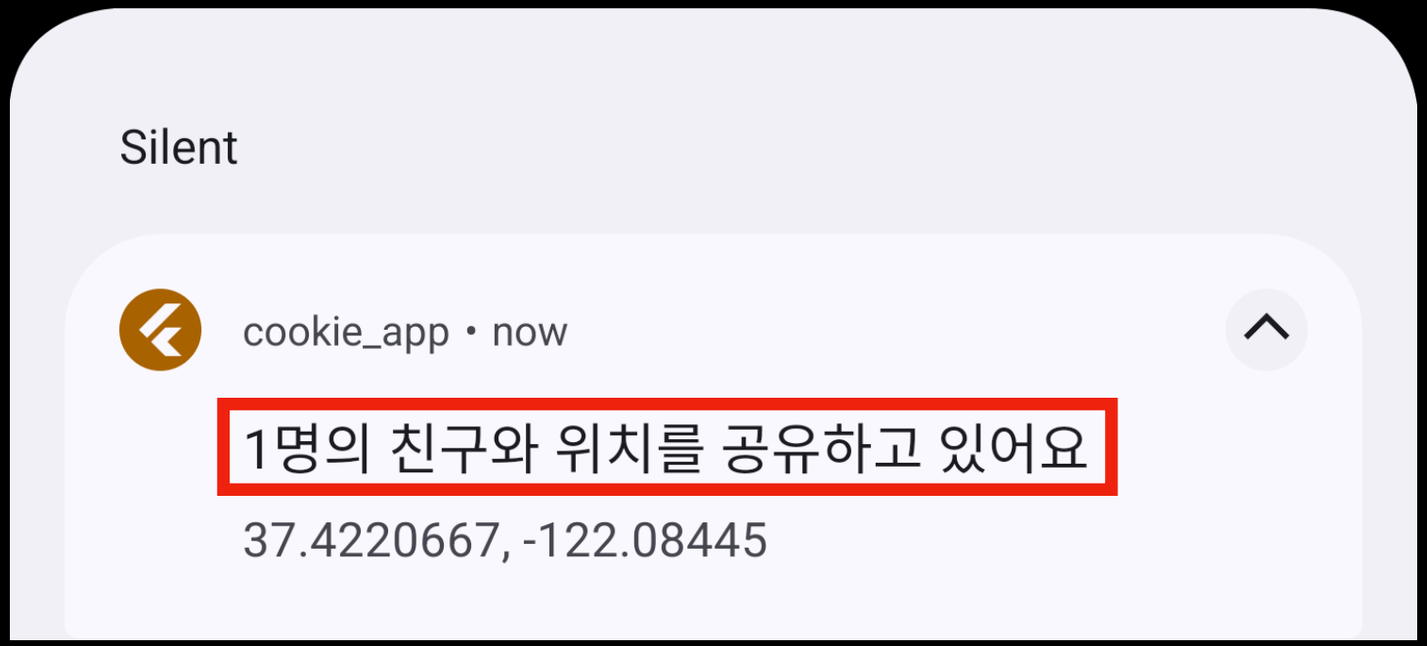
5:06

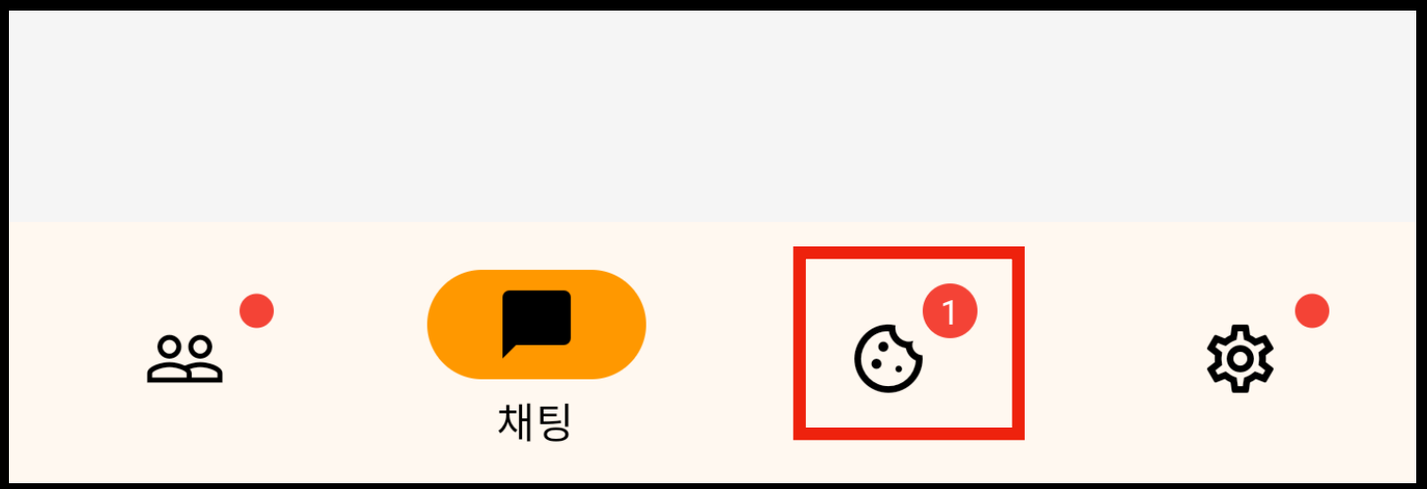
C🍪🍪KIE

현재 접속 중인 친구

test1 43 m

Maps Tab
Notification





State Management

MVVMS

Application Global Service

Auth Service, Account Service

- **Auth Service**
  - Automatically manages JWT Token
- **Account Service**
  - Provides Account Informations
    - such as friend’s id and profile
    - or my id, profile, phone number etc.

Application Global Service
SocketIO Based - Chat & Map Service

- **Chat Service**
  - Manages `Chat` namespace socket
  - Saves Information about all of the chatting rooms (as ViewModel)
  - Based on SocketIO Events
- **Map Service**
  - Manages `Map` namespace socket
  - Based on SocketIO Events

ChatService

```
Map<ChatRoomID (string), ChatRoomViewModel> {
  “12hfe”: ChatRoomViewModel
  “aiefd”: ChatRoomViewModel
  “ih4hj”: ChatRoomViewModel
  “sdfie”: ChatRoomViewModel
  “53he3”: ChatRoomViewModel
}
```

ChatService
New Event!

```
Event: Chat
Data: {
  roomId: “12hfe”,
  sender: “testid2”,
  timestamp: “2023-11-20T16:30:00Z”,
  payload: Message
}
```

ChatService

```
Map<ChatRoomID (string), ChatRoomViewModel> {
  “12hfe”: ChatRoomViewModel
  “aiefd”: ChatRoomViewModel
  “ih4hj”: ChatRoomViewModel
  “sdfie”: ChatRoomViewModel
  “12he3”: ChatRoomViewModel
}

map[“12hfe”].addChat(message);
```

ChatRoomViewModel

```
this._model.id: “12hfe”
Messages getter
addChat:
  this._model.messages.push(message)
  // To Update the View
  notifyListers();
```

ChatTabs

```
List<ChatRoomViewModel> chatRooms = context.watch<ChatService>().rooms;
return ListView.separated(
  separato
      cons
  itemCount: context.watch<ChatService>().rooms.length,
  itemBuilder: (BuildContext context, int index) {
    ChatRoomViewModel chatRoom = chatRooms[index];

    return ChatRoomListEntry(
      name: chatRoom.name,
      image: chatRoom.image,
      message: chatRoom.messages.isNotEmpty
          ? chatRoom.lastMessage
          : "Empty Message",
      time: chatRoom.messages.isNotEmpty
          ? chatRoom.lastActive
          : chatRoom.createdAt,
      navigate: ChatPage(room: chatRoom),
    ); // ChatRoomListEntry
  },
); // ListView.separated
```

Keep Watches if the ChatService Rooms has changed

ChatPage

```
@override
Widget build(BuildContext context) => ChangeNotifierProvider.value(
      value: this._room,
      builder: (context, chiid) => Scaffold(
        appBar: AppBar(
          title: const Text('채팅'),
        ), // AppBar
        body: Chat(
          theme: DefaultChatTheme(
            // inputPadding: EdgeInsets.all(24),
            inputBackgroundColor: Colors.orangeAccent,
            inputTextStyle: const TextStyle(
              color: Colors.black,
              fontSize: 16,
            ), // TextStyle
            inputTextDecoration: InputDecoration(
              border: OutlineInputBorder(
                borderSide: const BorderSide(
                  color: Colors.orangeAccent,
                ), // BorderSide
                borderRadius: BorderRadius.circular(1),
              ), // OutlineInputBorder
            ), // InputDecoration
```

Keep Watches if the ChatViewModel messages has changed

```
          messages: context.watch<ChatRoomViewModel>().messages,
          onAttachmentPressed: _handleAttachmentPressed,
          onMessageTap: _handleMessageTap,
          onPreviewDataFetched: _handlePreviewDataFetched,
```

Chat

Test 1
iOS 17.0

Test 2
iOS 17.0

Test 3
iOS 17.0

3:27

채팅

234
Test message
방금 전

asdf
Oaoaoa
6분 전

test
Empty Message
3시간 전

34
Empty Message
3시간 전

test
Empty Message
3시간 전

dfasdf
Empty Message
3시간 전

sdfasfe
Empty Message
3시간 전

asdf
Empty Message
3시간 전

0

채팅

3:28

채팅

234
Test message
방금 전

asdf
Oaoaoa
6분 전

test
Empty Message
3시간 전

dfasdf
Empty Message
3시간 전

sdfasfe
Empty Message
3시간 전

asdf
Empty Message
3시간 전

0

채팅

3:28

채팅

234
Test message
방금 전

asdf
Oaoaoa
6분 전

test
Empty Message
3시간 전

34
Empty Message
3시간 전

0

채팅

Test 1
iOS 17.0

Test 2
iOS 17.0

Test 3
iOS 17.0

3:44

채팅

15:32

Hello

test2
Hi

Ch8.Testing.pptx
531.47 kB

Message

3:44

채팅

15:32

test1
Hello

Hi

Ch8.Testing.pptx
531.47 kB

Message

3:44

채팅

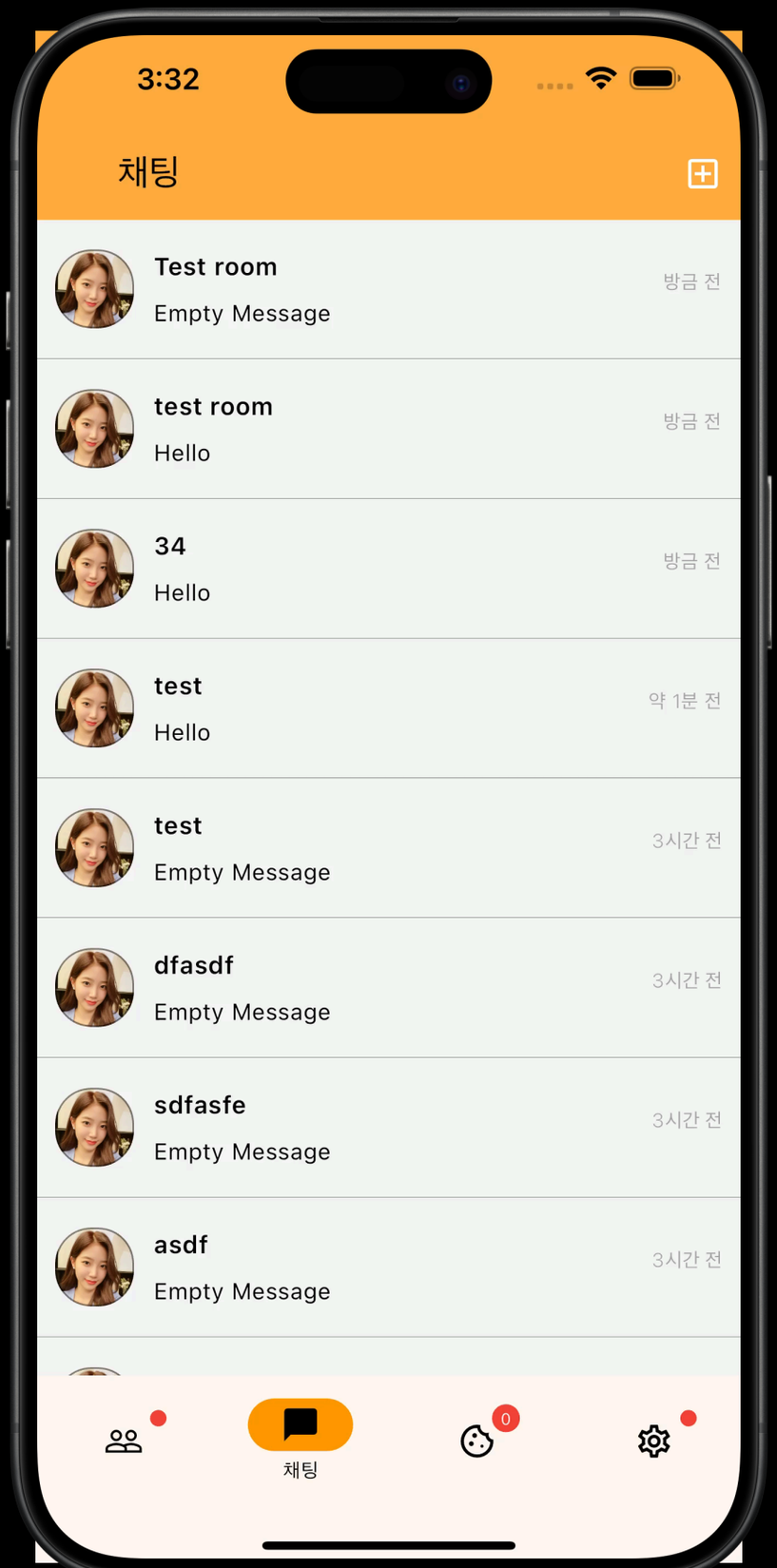
15:32

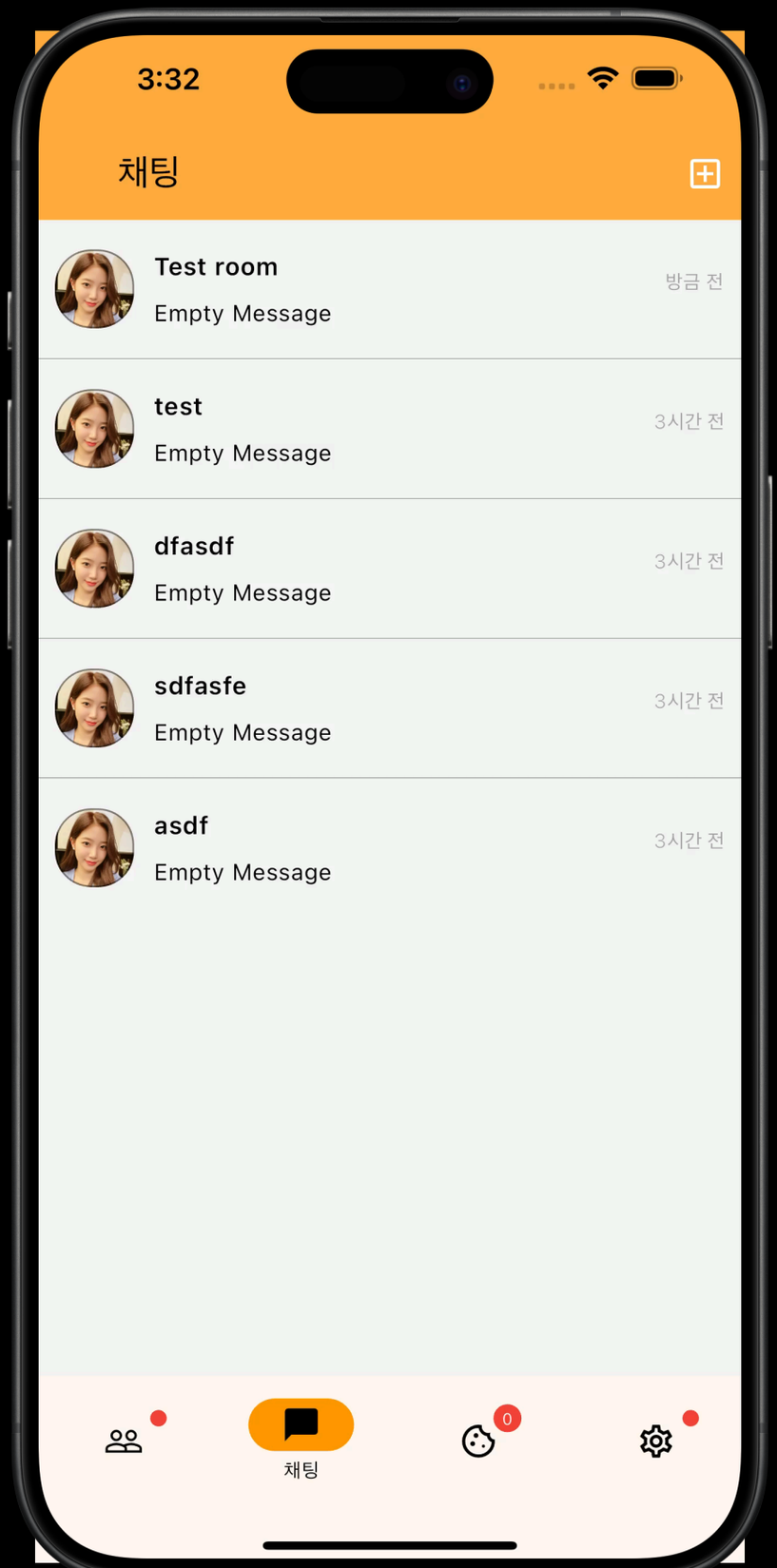
test1
Hello

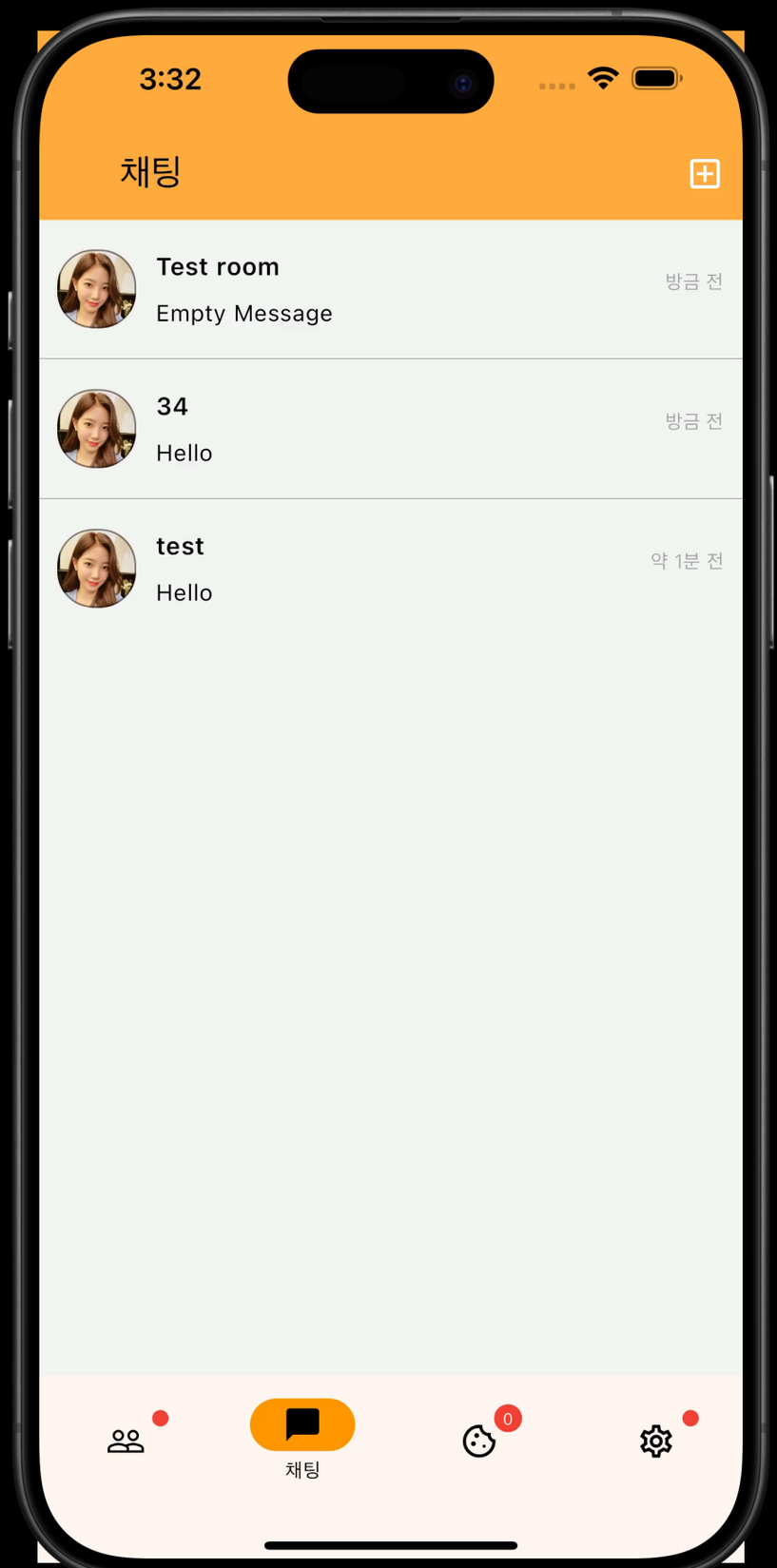
test2
Hi

Ch8.Testing.pptx
531.47 kB

Message


3:32
채팅
Test room
Empty Message
방금 전
test room
Hello
방금 전
34
Hello
방금 전
test
Hello
약 1분 전
test
Empty Message
3시간 전
dfasdf
Empty Message
3시간 전
sdfasfe
Empty Message
3시간 전
asdf
Empty Message
3시간 전
0
채팅


3:32
채팅
Test room
Empty Message
방금 전
test
Empty Message
3시간 전
dfasdf
Empty Message
3시간 전
sdfasfe
Empty Message
3시간 전
asdf
Empty Message
3시간 전
0
채팅


3:32
채팅
Test room
Empty Message
방금 전
34
Hello
방금 전
test
Hello
약 1분 전
0
채팅

Server

Debugging & Refactoring

- Feat: Express Global Error Handler
- Fix: socket.io admin panel xhr polling error
    - Caused by customized socket.io middleware
- Fix: chat message not saved to db
- Refactor: Folder Architecture

Thank You

Questions are welcome