Enable the Chat Feature

Native App Development

Time: 60 mins

Introduction

In this class, the student/s will learn to enable the chat feature so that doctors and patients can send custom messages to each other. Student/s will learn to save and retrieve the messages to/from the database. They will learn how to display the chat messages on the chat window and style the messages sent and received chronically and in different styles.

New Commands Introduced

- firebase.database.servervalue.timestamp

  Gets the current time in milliseconds from firebase server

- limitToLast()

  Puts a limit to the number of items returned from a list beginning from the last one.

- orderByChild()

  Specifies the child key to order the results by.

- .on()

  Retrieves data from the Firebase database. Returns a snapshot of the data

- forEach()

  Calls a function for each element in an array while iterating through the array

- .va()

  Extracts a JavaScript value from a snapshot.

Vocabulary

- Timestamp: The current in seconds or milliseconds represented as a number. This is the number of seconds or milliseconds that has elapsed since the epoch time January 1, 1970 (midnight UTC/GMT). Timestamp is a 10, 13, 16 digit number based on whether seconds, milliseconds, and microseconds are represented.

Learning Objectives

Student/s should be able to:
- **Recall** how to add a chat window and send a message using code.
- **Demonstrate** how to send and save messages in the database
- **Explain** the concept of timestamp.
- **Demonstrate** displaying messages using a FlatList.

Activities

1. **Class Narrative:** *(2 mins)*
   - Brief the student/s about sending and receiving custom messages are an important part of the chat feature. Patients and doctors will be able to send messages to each other. The messages will be displayed within each chat window chronologically.

2. **Concept Introduction Activity:** *(5 mins)*
   - Let the student/s play the explore-activity and explore how to send and view messages both as a doctor and as a patient.
     Note: Before the class starts, create profiles of a doctor and a patient so that it is easy for students to explore the activity.
   - Using the slides, explain that the student/s will learn:
     - to send a message.
     - to receive and display messages.
     - to style the messages based on the signed-in user.

3. **Activity 1: Send Chat Messages:** *(12 mins)*

   Teacher Activity: *(6 mins)*.
   - Explain the process of sending messages and saving them in the database in the collection named "chats". The chatRoomId is the unique identifier to identify a particular chat thread and is obtained from fellowUser['connections'].
   - Explain to the student/s how to build a chat message using currentUserId, message, timestamp, and message type. The message type is always set to "text" in this class.
   - Explain the concept of timestamp and how the timestamp is obtained from the Firebase database using firebase.database.ServerValue.TIMESTAMP

   Student Activity: *(6 mins)*
   - Guide the student/s to send and save messages in the database by following similar steps as in the teacher activity.

4. **Activity 2: Receive Chat Messages:** *(12 mins)*

   Teacher Activity: *(6 mins)*
   - Explain to students how to retrieve chat messages from the collection "chats" using .ref() function and chatRoomId.
   - Explain how the messages can be arranged chronologically by ordering them by timestamp.
   - Describe the concept of limitToLast() using which a limit can be put to the number of messages retrieved at once.
     Note: Show the output of the teacher activity on a mobile device on which Snack Expo is installed.

   Student Activity: *(6 mins)*
   - Guide the student/s to display the home screen of the doctor when clicking on the Map Icon.
     Probing question: What is the use of the splice() function?
     Expected answer: Changes the items in an array by adding, deleting or replacing existing items.

5. **Activity 3: Style the Chat Messages:** *(8 mins)*

   Student Activity: *(8 mins)*

- Guide the student/s to pass the userType to the bottom tabs. Explain how a ternary operator can be used to check the user type and display the Home and Map tabs for a patient and the Calendar tab for a doctor.

6. **Introduce the Post class project:** *(2 min)*
   - Add a carousel and customize the bottom tabs for the Campus Security app..

7. **Test and Summarize the class learnings:** *(5 mins)*
   - Check for understanding through quizzes and summarize learning after respective missions.
   - Summarize the overall class learning towards the end of the class.

8. **Additional activities:**
   - Encourage the student/s to restrict displaying and sending empty messages.
   - Encourage the student/s to the date and time of a message on the chat window.

9. **State the Next Class Objective:** *(1 min)*
   - In the next class, student/s will learn to add features to send and play voice messages.

U.S. Standards:

CSTA: 2-AP-11, 2-AP-12, 2-AP-13, 2-AP-14, 2-AP-19

Links Table		
Activity	**Activity Name**	**Link**
Class Presentation	Enable the Chat Feature	https://s3-whjr-curriculum-uploads.whjr.online/228f0c03-a047-4c74-b654-f0922b58e6b4.html
Explore Activity	Enable the Chat Feature	https://snack.expo.dev/@procodingclass/wad-402-c63-sa3-style-the-chat-message--solution
Teacher Activity 1	Send Chat Message	https://snack.expo.dev/@procodingclass/wad-402-c63-ta1-send-chat-message?platform=web
Teacher Activity 1 Solution	Send Chat Message - Solution	https://snack.expo.dev/@procodingclass/wad-402-c63-ta1-send-chat-message--solution?platform=web
Student Activity 1	Send Chat Message	https://snack.expo.dev/@procodingclass/wad-402-c63-sa1-send-chat-message?platform=web
Teacher Reference: Student Activity 1 Solution	Send Chat Message - Solution	https://snack.expo.dev/@procodingclass/wad-402-c63-sa1-send-chat-message--solution
Teacher Activity 2	Receive Chat Messages	https://snack.expo.dev/@procodingclass/wad-402-c63-ta2-receive-chat-messages

Teacher Activity 2 Solution	Receive Chat Messages - Solution	https://snack.expo.dev/@procodingclass/wad-402-c63-ta2-receive-chat-messages--solution
Student Activity 2	Display Chat Messages	https://snack.expo.dev/@procodingclass/wad-402-c63-sa2-display-chat-messages
Teacher Reference: Student Activity 2 Solution	Display Chat Messages - Solution	https://snack.expo.dev/@procodingclass/wad-402-c63-sa2-display-chat-messages--solution
Student Activity 3	Style the Chat Messages	https://snack.expo.dev/@procodingclass/wad-402-c63-sa3-style-the-chat-message
Teacher Reference: Student Activity 3 Solution	Style the Chat Messages - Solution	https://snack.expo.dev/@procodingclass/wad-402-c63-sa3-style-the-chat-message--solution
Student's Additional Activity 1	Remove Blank Messages	https://snack.expo.dev/@procodingclass/wad-402-c63-aa1-remove-blank-messages
Teacher Reference: Student's Additional Activity 1 Solution	Remove Blank Messages - Solution	https://snack.expo.dev/@procodingclass/wad-402-c63-aa1-remove-blank-messages--solution
Student's Additional Activity 2	Display Date and Time	https://snack.expo.dev/@procodingclass/wad-402-c63-aa2-display-date-and-time
Teacher Reference: Student's Additional Activity 2 Solution	Display Date and Time - Solution	https://snack.expo.dev/@procodingclass/wad-402-c63-aa2-display-date-and-time-solution
Post Class Project	Enable Chat	https://snack.expo.dev/@procodingclass/wad-402-c63-pcp-enable-chat?platform=web
Teacher Reference: Post Class Project Solution	Enable Chat - Solution	https://snack.expo.dev/@procodingclass/wad-402-c63-pcp-enable-chat--solution?platform=web