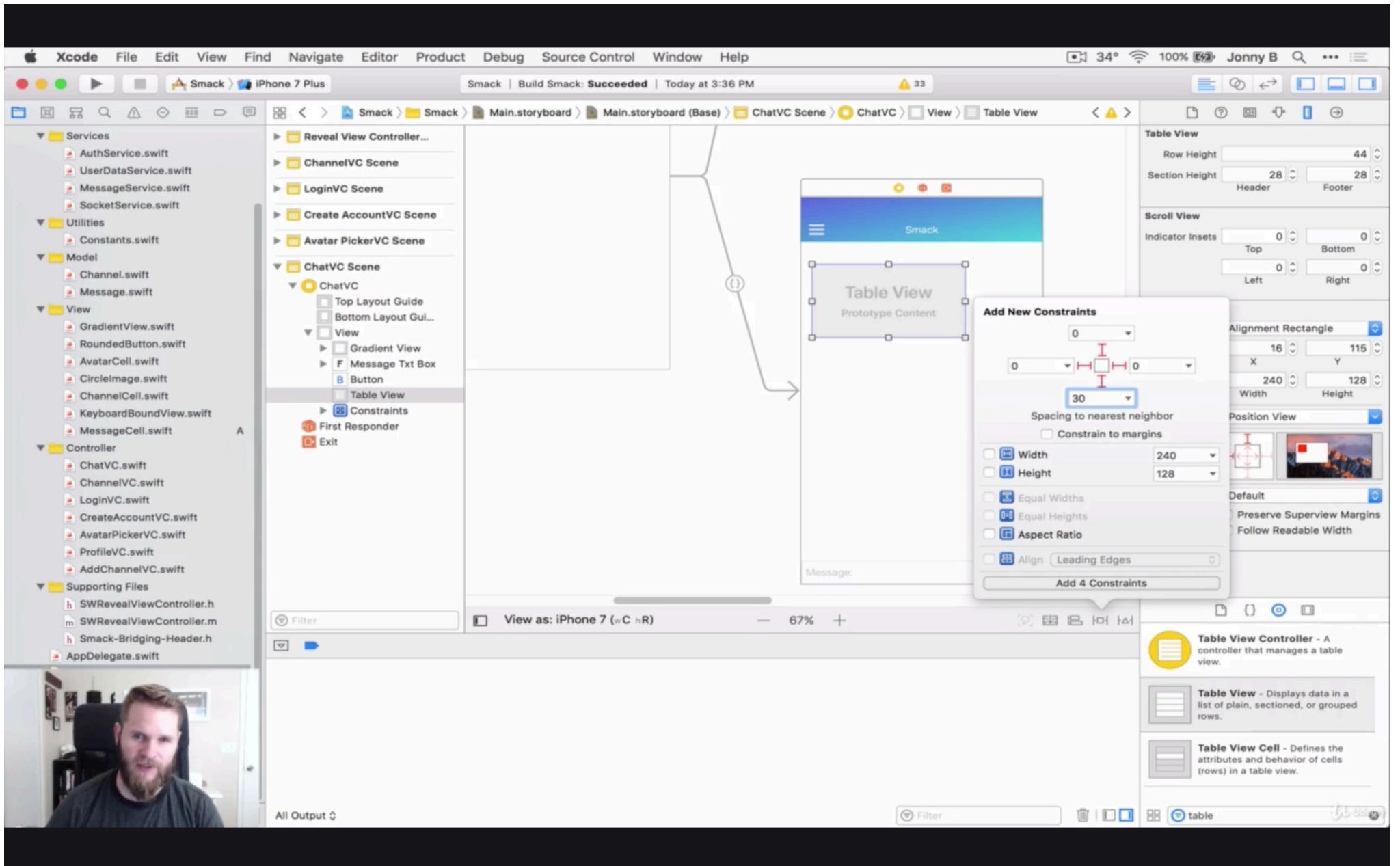




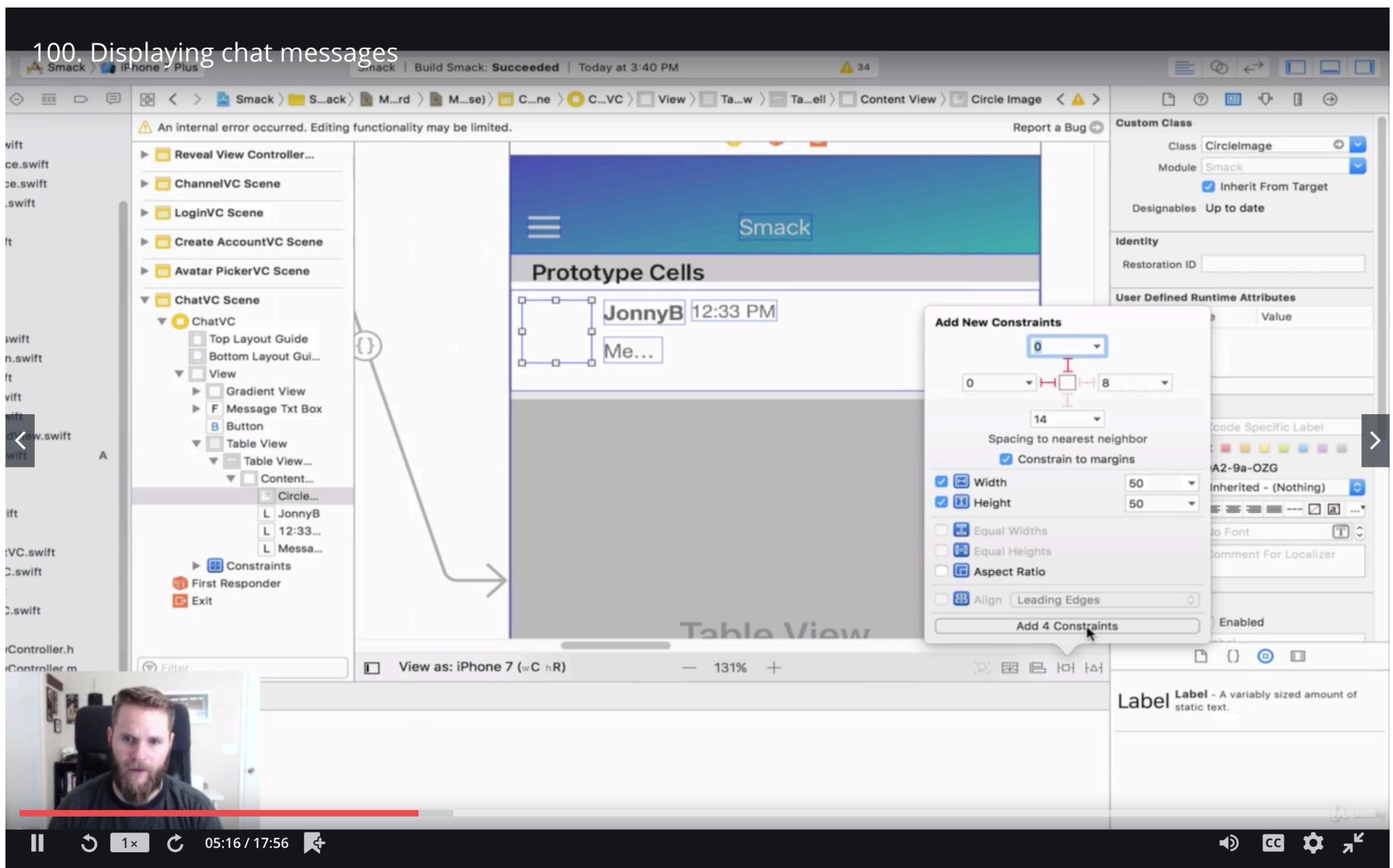
A screenshot of the Xcode IDE interface. The top menu bar includes Xcode, File, Edit, View, Find, Navigate, Editor, Product, Debug, Source Control, Window, and Help. The status bar shows the device as iPhone 7 Plus, the project name Smack, the build status Succeeded, the date Today at 2:08 PM, and battery level 34%. The main window shows the file structure of the Smack project. The left sidebar lists XIBs (ProfileVC.xib, AddChannelVC.xib), Services (AuthService.swift, UserDataService.swift, MessageService.swift, SocketService.swift), Utilities (Constants.swift), Model (Channel.swift, Message.swift), View (GradientView.swift, RoundedButton.swift, AvatarCell.swift, CircleImage.swift, ChannelCell.swift, KeyboardBoundView.swift, MessageCell.swift), Controller (ChatVC.swift, ChannelVC.swift, LoginVC.swift, CreateAccountVC.swift, AvatarPickerVC.swift, ProfileVC.swift, AddChannelVC.swift), and Supporting Files (SWRevealViewController.h, SWRevealViewController.m). The right pane displays the code for MessageCell.swift:

```
1 //  
2 // MessageCell.swift  
3 // Smack  
4 //  
5 // Created by Jonny B on 7/20/17.  
6 // Copyright © 2017 Jonny B. All rights reserved.  
7 //  
8  
9 import UIKit  
10  
11 class MessageCell: UITableViewCell {  
12  
13     override func awakeFromNib() {  
14         super.awakeFromNib()  
15         // Initialization code  
16     }  
17  
18     override func setSelected(_ selected: Bool, animated: Bool) {  
19         super.setSelected(selected, animated: animated)  
20  
21         // Configure the view for the selected state  
22     }  
23  
24 }
```

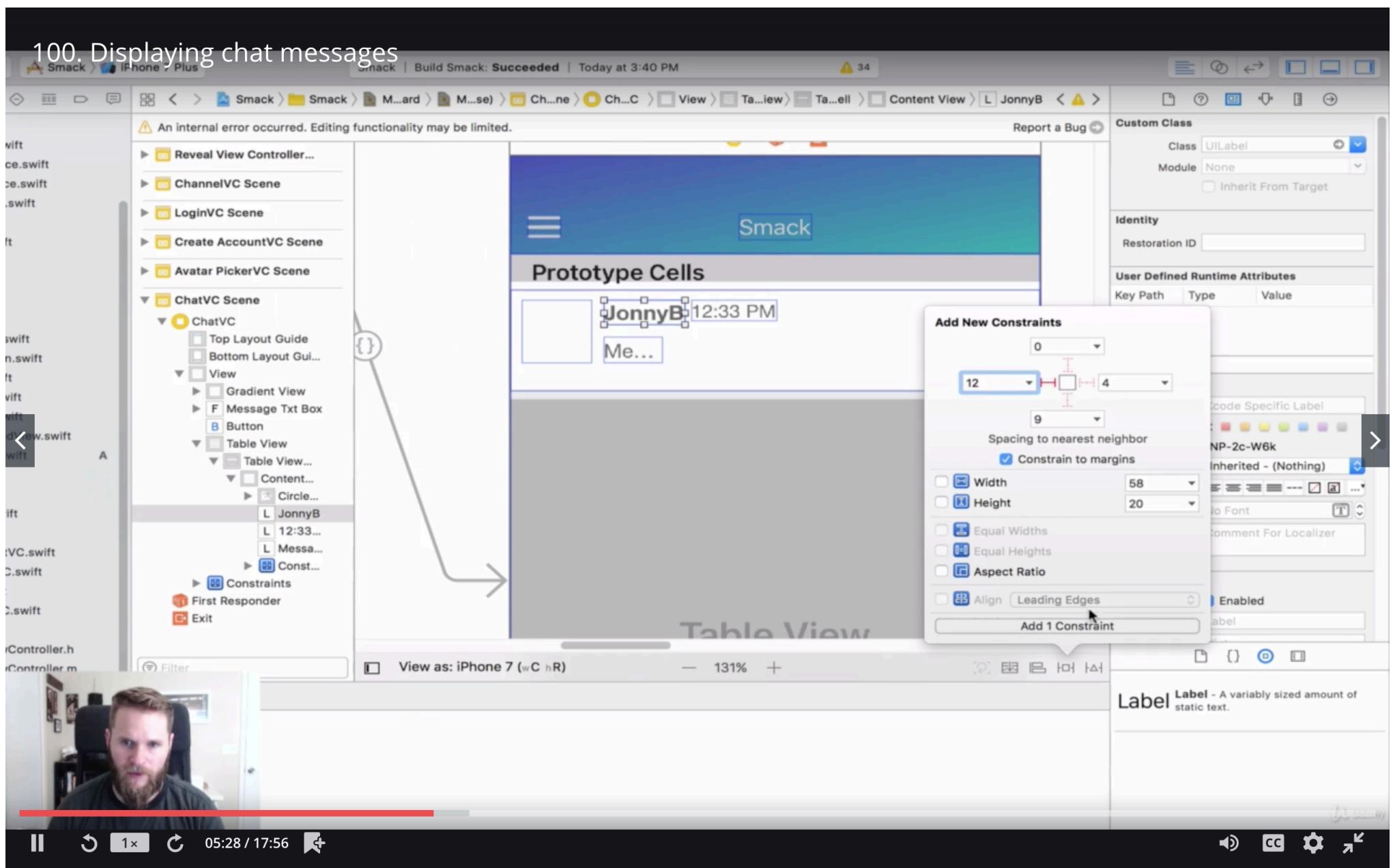
The bottom of the screen shows the "All Output" tab and a "Filter" search bar.



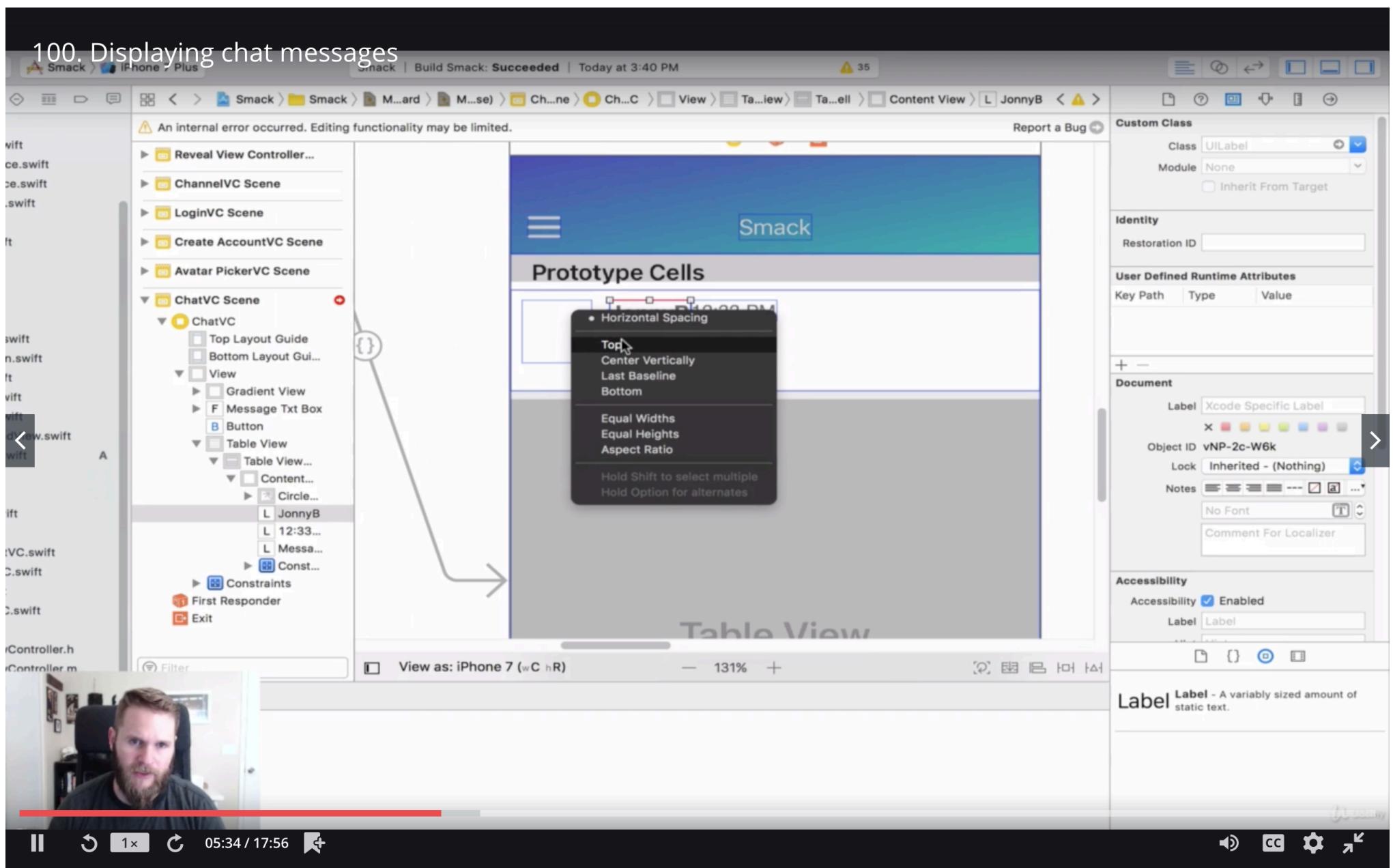
100. Displaying chat messages



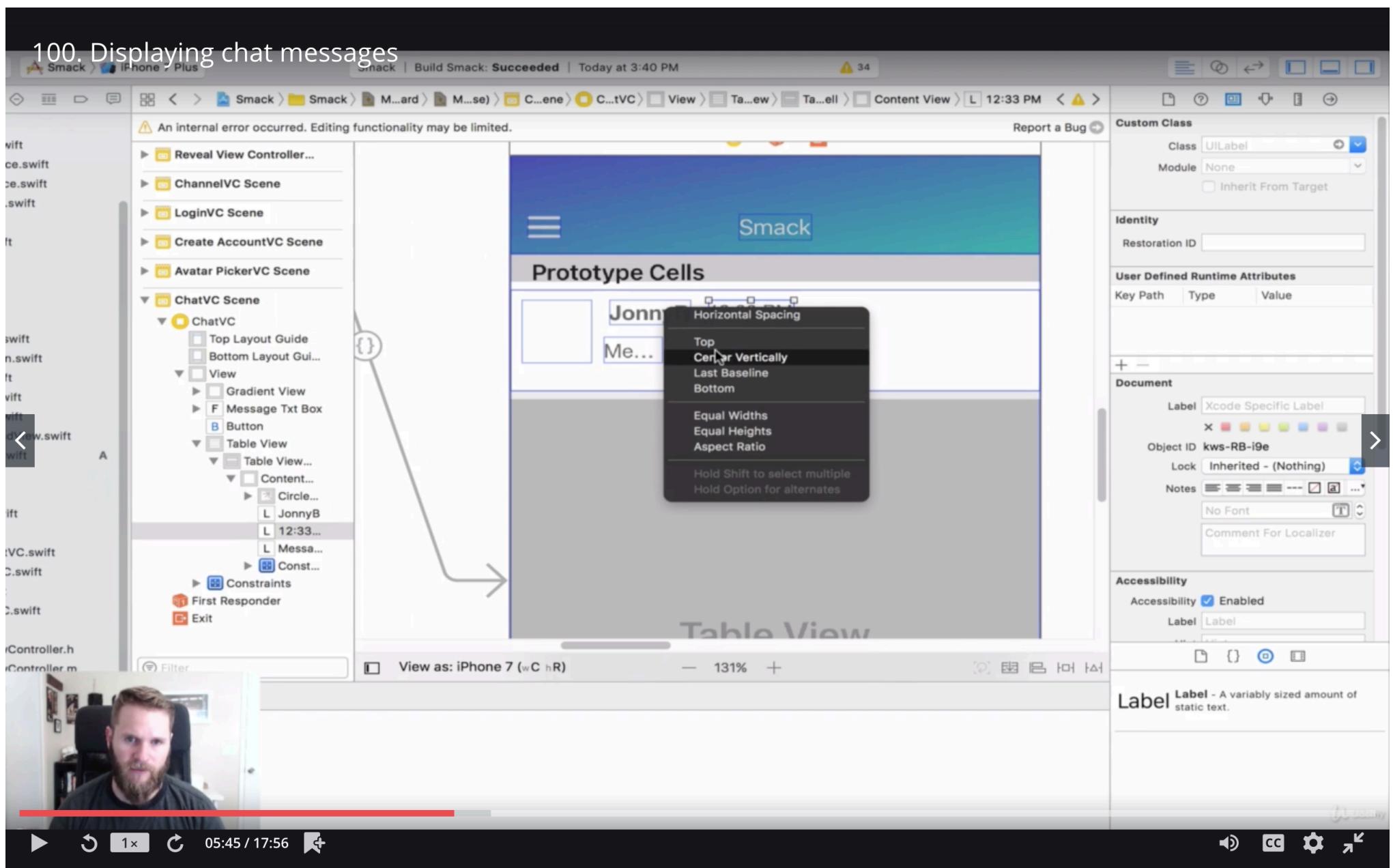
100. Displaying chat messages



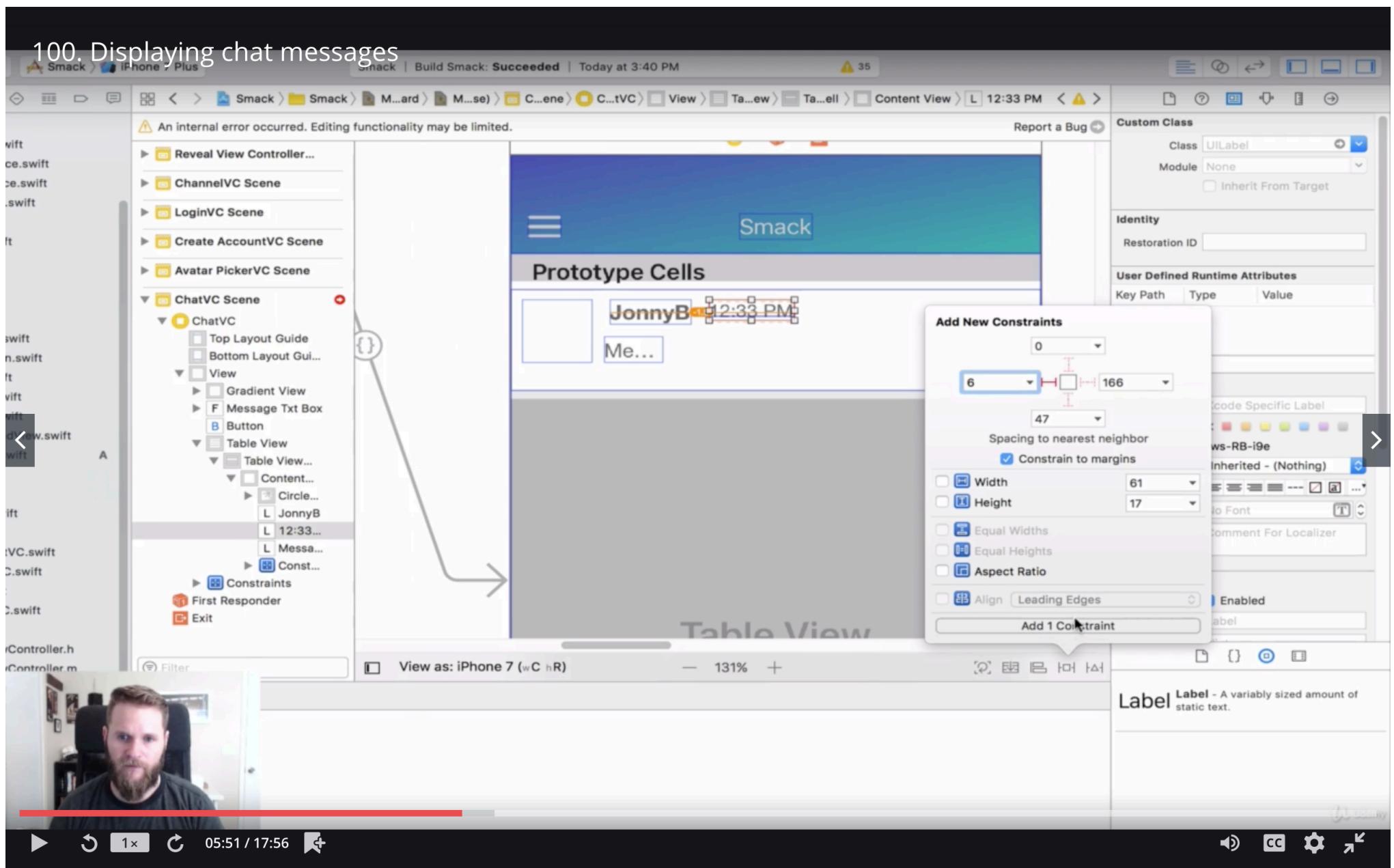
100. Displaying chat messages



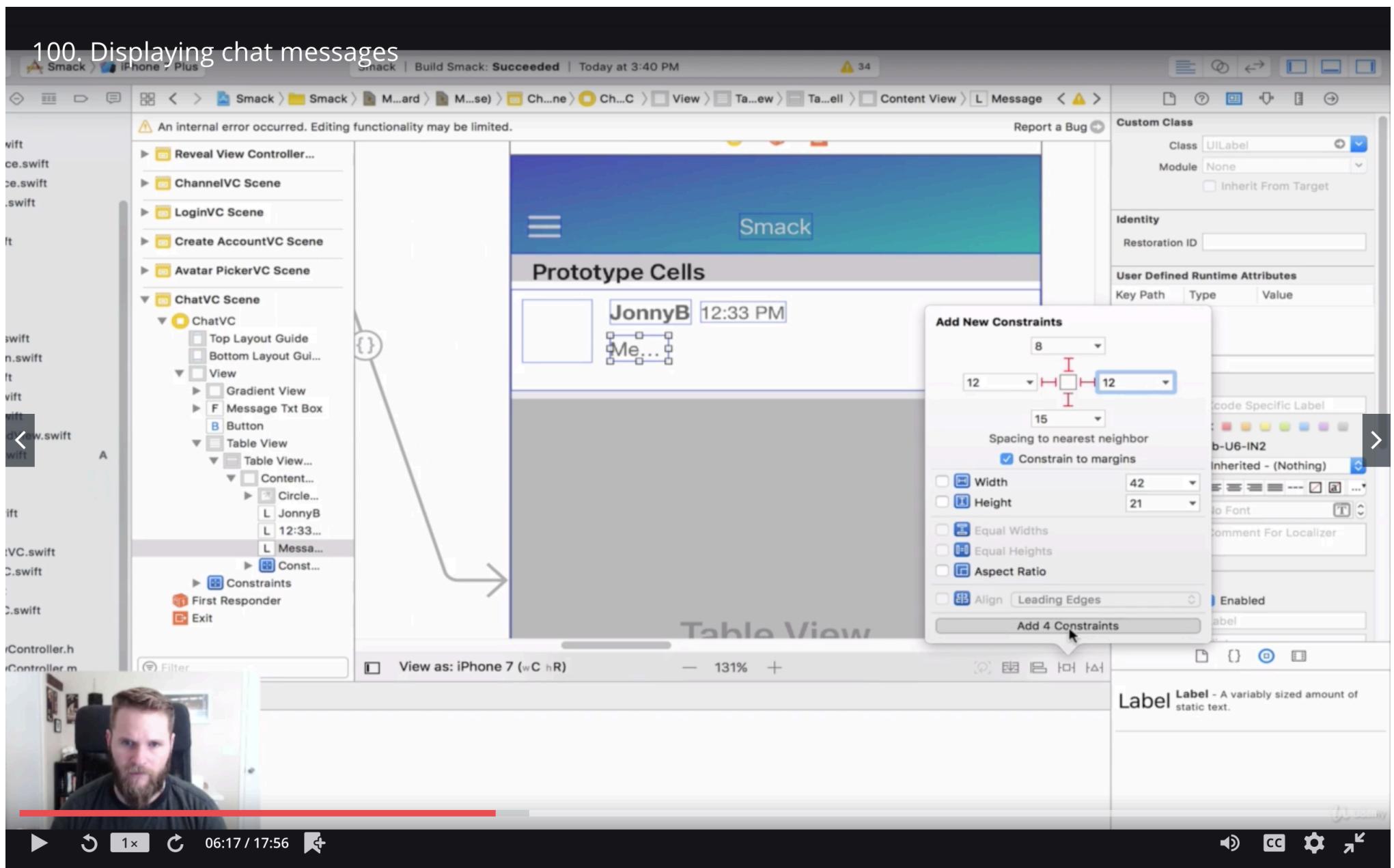
100. Displaying chat messages



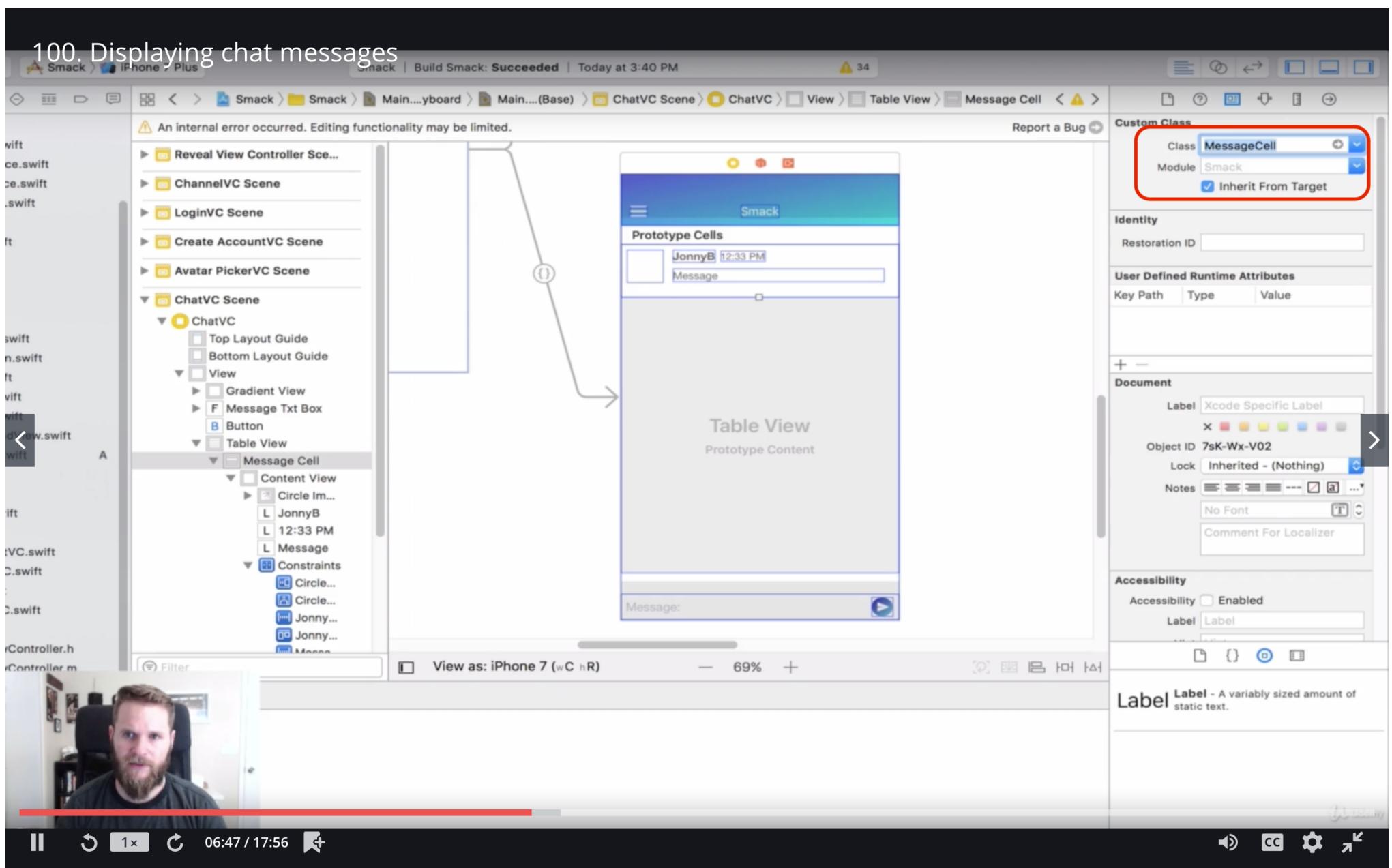
100. Displaying chat messages



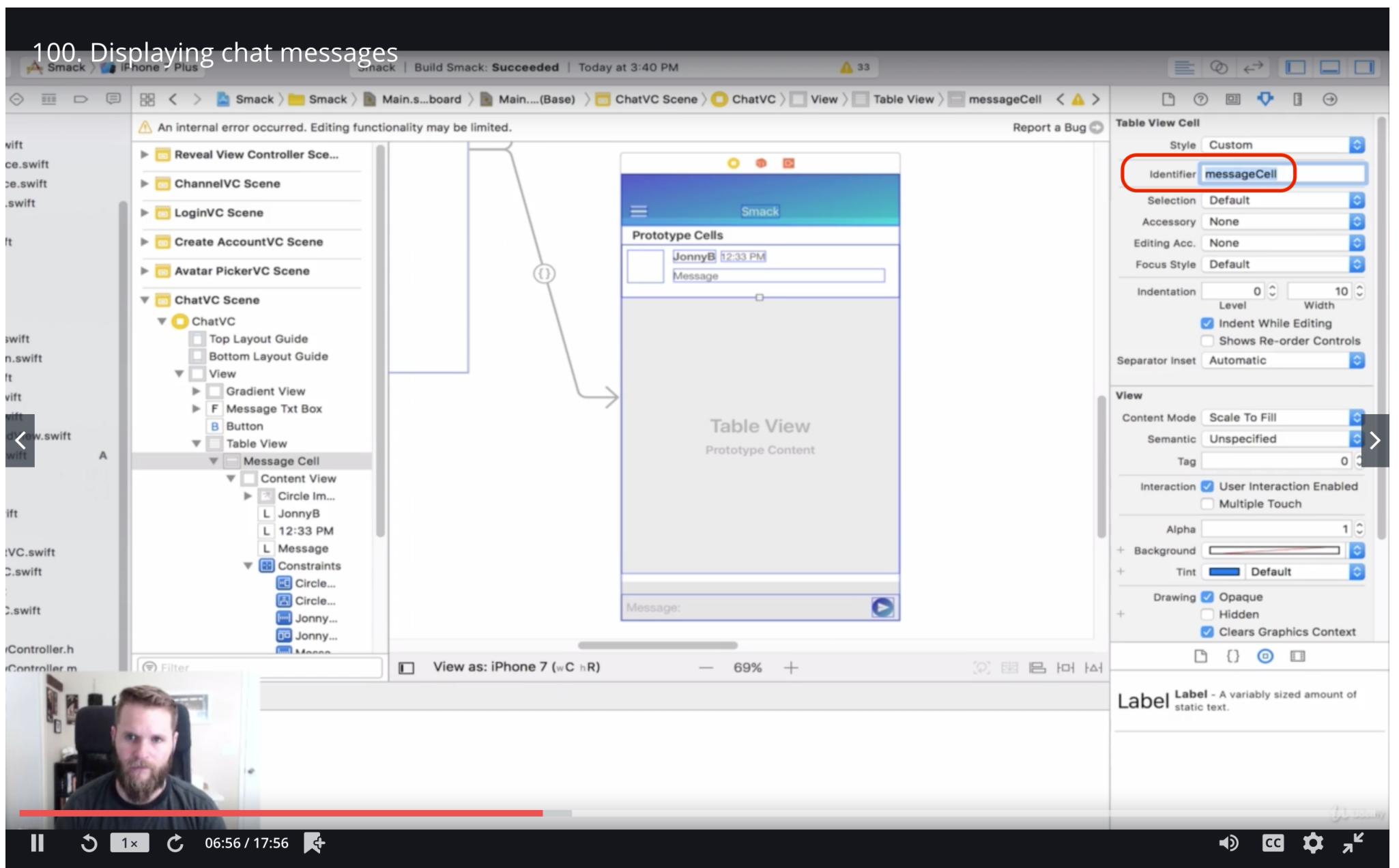
100. Displaying chat messages



100. Displaying chat messages



100. Displaying chat messages



100. Displaying chat messages

The screenshot shows the Xcode interface with the following details:

- Project Navigator:** Shows the project structure with files like AuthService.swift, UserDataService.swift, etc., under Services; Constants.swift under Utilities; Channel.swift and Message.swift under Model; and various View-related files like GradientView.swift under View.
- MessageCell.swift:** The current file being edited is MessageCell.swift, which is a UITableViewCell subclass.
- Editor:** The code editor displays the following Swift code:

```
4 //  
5 // Created by Jonny B on 7/20/17.  
6 // Copyright © 2017 Jonny B. All  
rights reserved.  
7 //  
8  
9 import UIKit  
10  
11 class MessageCell: UITableViewCell  
{  
12  
    // Outlets  
    @IBOutlet weak var userImg: CircleImage!  
    @IBOutlet weak var userNameLbl: UILabel!  
    @IBOutlet weak var timeStampLbl: UILabel!  
    @IBOutlet weak var messageBodyLbl: UILabel!  
18  
19    override func awakeFromNib() {  
20        super.awakeFromNib()  
        // Initialization code  
21    }  
}
```
- Assistant Editor:** The right pane shows the storyboard representation of the MessageCell. It's a prototype cell within a UITableView. The cell contains a UIImageView (User Img), a UILabel (User Na...), a UILabel (Time Sta...), and a UILabel (Message...).
- Utilities:** The Utilities panel on the right shows the "Label" tool selected, with a tooltip: "Label - A variably sized amount of static text."
- Bottom Bar:** The Xcode bottom bar includes standard controls like play/pause, zoom, and a timestamp (08:06 / 17:56).

100. Displaying chat messages

The screenshot shows the Xcode interface with the following details:

- Project Navigator:** Shows the project structure with files like AuthService.swift, UserDataService.swift, MessageService.swift, SocketService.swift, Constants.swift, Channel.swift, Message.swift, GradientView.swift, RoundedButton.swift, AvatarCell.swift, CircleImage.swift, ChannelCell.swift, and KeyboardBoundView.swift.
- Code Editor:** Displays the `MessageCell.swift` file content:

```
8
9 import UIKit
10
11 class MessageCell: UITableViewCell {
12
13     // Outlets
14     @IBOutlet weak var userImg: CircleImage!
15     @IBOutlet weak var userNameLbl: UILabel!
16     @IBOutlet weak var timeStampLbl: UILabel!
17     @IBOutlet weak var messageBodyLbl: UILabel!
18
19     override func awakeFromNib() {
20         super.awakeFromNib()
21     }
22
23     func configureCell(message: Message) {
24         messageBodyLbl.text = message.message
25         userNameLbl.text = message.userName
26         userImg.image = UIImage(named: message.userAvatar)
27         userImg.backgroundColor =
28             UserDataService.instance.returnUIColor(components:
29                 message.userAvatarColor)
30     }
31 }
```
- Assistant View:** Shows the definition for `Label`: "Label - A variably sized amount of static text."
- Bottom Bar:** Includes icons for play, refresh, zoom, and other controls, along with the text "10:33 / 17:56".

100. Displaying chat messages

The screenshot shows the Xcode interface with the following details:

- Project Navigator:** Shows the project structure with files like AuthService.swift, UserDataService.swift, MessageService.swift, SocketService.swift, Constants.swift, Channel.swift, Message.swift, GradientView.swift, RoundedButton.swift, AvatarCell.swift, CircleImage.swift, ChannelCell.swift, KeyboardBoundView.swift, and MessageCell.swift.
- Code Editor:** The ChatVC.swift file is open, displaying Swift code for a ChatViewController. It includes imports for UIKit, defines a ChatVC class, and contains outlets for menuBtn, channelNameLbl, messageTxtBox, and a UITableView named tab. An override for viewDidLoad() is also present.
- Assistant Editor:** A popover shows the connection setup for the tab outlet, linking it to the UITableView in the storyboard.
- Document Outline:** Shows the hierarchy of the storyboard scene, including the ChatVC view, its subviews (Gradient View, Message Txt Box, Button, Table View), and the Table View's prototype cell (messageCell) which contains subviews for User, Time, and Message.
- Storyboard Preview:** The storyboard preview shows a login screen followed by a ChatVC screen. The ChatVC screen displays a single message cell with placeholder data.
- Attributes Inspector:** On the right, the attributes inspector is open for the Table View, showing settings for Content (Dynamic Prototypes), Style (Plain), and Separator (None). It also includes sections for Selection, Drag and Drop, and Section Index.
- Utilities Inspector:** The utilities inspector is also visible on the right side of the interface.
- Bottom Bar:** The Xcode bottom bar includes standard navigation icons (play, stop, refresh, zoom) and a status bar indicating the current time (10:50 / 17:56).

100. Displaying chat messages

The screenshot shows the Xcode interface with the project 'Smack' open. The file 'ChatVC.swift' is selected in the navigation bar, which also displays the status 'Build Smack: Failed' and the date 'Today at 3:47 PM'. The code editor shows the implementation of the `ChatVC` class, which conforms to `UIViewController`, `UITableViewDelegate`, and `UITableViewDataSource`. The `viewDidLoad()` method is overridden to set the `tableView.delegate` and `tableView.dataSource` properties to `self`. A red box highlights these two lines of code.

```
7 //  
8  
9 import UIKit  
10  
11 class ChatVC: UIViewController, UITableViewDelegate, UITableViewDataSource {  
12     {  
13         // Outlets  
14         @IBOutlet weak var menuBtn: UIButton!  
15         @IBOutlet weak var channelNameLbl: UILabel!  
16         @IBOutlet weak var messageTxtBox: UITextField!  
17         @IBOutlet weak var tableView: UITableView!  
18  
19     override func viewDidLoad() {  
20         super.viewDidLoad()  
21         view.bindToKeyboard()  
22         tableView.delegate = self  
23         tableView.dataSource = self  
24         let tap = UITapGestureRecognizer(target: self, action:  
25             #selector(ChatVC.handleTap))  
26         view.addGestureRecognizer(tap)  
menuBtn.addTarget(self.revealViewController(), action:  
    #selector(SWRevealViewController.revealToggle(_:)),  
    for: .touchUpInside)
```

The left sidebar shows the project structure with various Swift files in Services, Utilities, Model, View, Controller, and Supporting Files. The bottom left corner features a video feed of the developer. The bottom right corner contains a 'Label' section from the Xcode library.

100. Displaying chat messages

The screenshot shows a Xcode interface with the following details:

- Project Navigator:** Shows the project structure with files like AuthService.swift, UserDataService.swift, MessageService.swift, SocketService.swift, Constants.swift, Channel.swift, Message.swift, GradientView.swift, RoundedButton.swift, AvatarCell.swift, CircleImage.swift, ChannelCell.swift, KeyboardBoundView.swift, and MessageCell.swift.
- Editor:** Displays the `ChatVC.swift` file. The code is for a UITableView delegate method `tableView(_:numberOfRowsInSection:)`. The cursor is at the end of the line `return MessageService.instance.messages.cou`, and a completion dropdown is open, showing the following suggestions:
 - Int count
 - Int count
 - Int underestimatedCount
 - Int underestimatedCount
- Build Bar:** Shows "Smack | Build Smack: Failed | Today at 3:50 PM".
- Bottom Bar:** Includes standard Xcode navigation icons (play, stop, refresh, zoom) and a timestamp "13:59 / 17:56".

100. Displaying chat messages

The screenshot shows the Xcode IDE with the project "Smack" open. The file "ChatVC.swift" is selected, and the code editor displays the following Swift code:

```
Building Smack: Smack | Compiling Swift source files
Smack / Smack / Controller / ChatVC.swift / getMessages()

if success {
    if MessageService.instance.channels.count > 0 {
        MessageService.instance.selectedChannel =
            MessageService.instance.channels[0]
        self.updateWithChannel()
    } else {
        self.channelNameLbl.text = "No channels yet!"
    }
}

func getMessages() {
    guard let channelId = MessageService.instance.selectedChannel?.id
        else { return }
    MessageService.instance.findAllMessageForChannel(channelId:
        channelId) { (success) in
        if success {
            self.tableView.reloadData()
        }
    }
}

func tableView(_ tableView: UITableView, cellForRowAt indexPath:
```

A red box highlights the line `self.tableView.reloadData()`. The right sidebar shows the "Label" definition: "Label - A variably sized amount of static text." The bottom status bar shows the video feed of the speaker and the time 14:22 / 17:56.

100. Displaying chat messages

The screenshot shows a developer's workspace in Xcode. On the left is the file browser with project files like Services, Utilities, Model, View, Controller, and Supporting Files. The main area shows the code for ChatVC.swift:

```
else { return }
MessageService.instance(channelId) { (success, error) in
    if success {
        self.tableView.reloadData()
    } else {
        self.showAlert("Error", message: error?.localizedDescription)
    }
}

func tableView(_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell {
    if let cell = tableView.dequeueReusableCell(withIdentifier: "messageCell", for: indexPath) as? MessageCell {
        let message = messages[indexPath.row]
        cell.configureCell(message)
        return cell
    } else {
        return UITableViewCell()
    }
}

func numberOfSections(in tableView: UITableView) -> Int {
    return 1
}
```

At the bottom, there are two log entries:

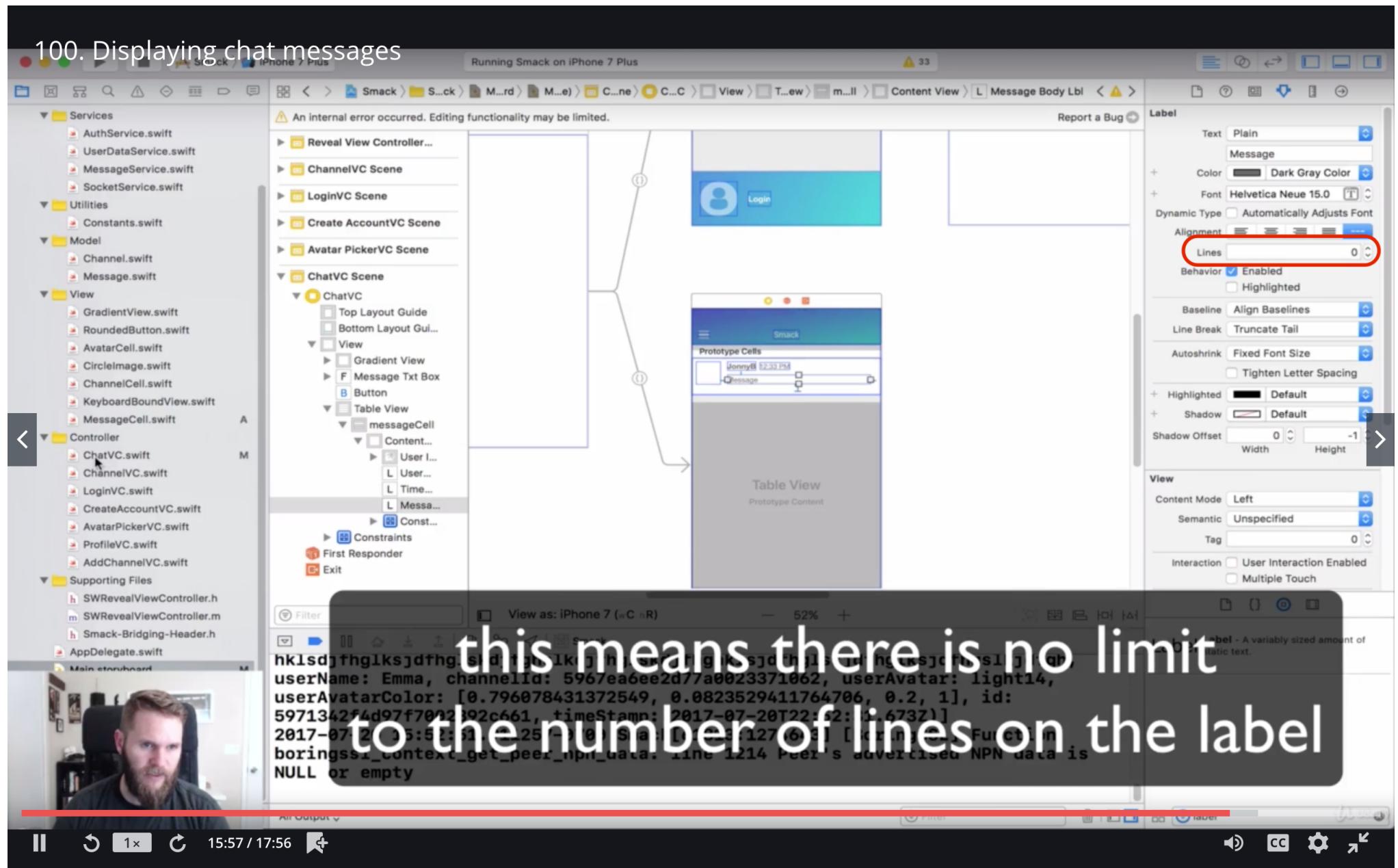
```
Smack.Message(message: , userName: , userAvatar: light14, userAvatarColor: 0.2, 1], id: 597115fa3118840023da
Smack.Message(message: sending our message, channelId: 5967ea6ee2d77a00233710 [0.796078431372549, 0.08235294117, timeStamp: 2017-07-20T21:07:35.62
```

The iPhone 7 Plus simulator in the center displays a chat interface titled '#general'. It shows messages from 'jonnyb' and 'Emma' at 12:33 PM. The messages are:

- jonnyb 12:33 PM Hey whats up everyone!
- Emma 12:33 PM Hey Jonny! What's going on?
- jonnyb 12:33 PM Oh you know! Just learning how to code with...
- Emma 12:33 PM that's what's up!
- Emma 12:33 PM Hello students!!
- Emma 12:33 PM hi
- Emma 12:33 PM Hey there everyone!
- Emma 12:33 PM

The Xcode interface also includes a 'Quick Help' panel on the right.

100. Displaying chat messages



100. Displaying chat messages

The screenshot shows the Xcode IDE with the project "Smack" open. The file "ChatVC.swift" is selected in the left sidebar under the "Controller" folder. The code editor displays the following Swift code:

```
8
9 import UIKit
10
11 class ChatVC: UIViewController, UITableViewDelegate, UITableViewDataSource {
12
13     // Outlets
14     @IBOutlet weak var menuBtn: UIButton!
15     @IBOutlet weak var channelNameLbl: UILabel!
16     @IBOutlet weak var messageTxtBox: UITextField!
17     @IBOutlet weak var tableView: UITableView!
18
19     override func viewDidLoad() {
20         super.viewDidLoad()
21         view.bindToKeyboard()
22         tableView.delegate = self
23         tableView.dataSource = self
24
25         tableView.estimatedRowHeight = 80
26         tableView.rowHeight = UITableViewAutomaticDimension
27
28         let tap = UITapGestureRecognizer(target: self, action:
29             #selector(ChatVC.handleTap))
30         view.addGestureRecognizer(tap)
31
32
33 }
```

A red box highlights the following two lines of code:

```
tableView.estimatedRowHeight = 80
tableView.rowHeight = UITableViewAutomaticDimension
```

The right side of the Xcode interface shows the "Quick Help" panel with the message "No Quick Help" and a "Search Documentation" button. The bottom of the screen shows the Xcode status bar with various icons and the text "All Output".

100. Displaying chat messages

