

## 97. Refining Login Flow

The screenshot shows the Xcode interface with the project navigation bar at the top. Below it is the file browser showing the project structure under 'Smack'. The main editor area displays a Swift file named 'ChannelVC.swift'. A red box highlights a section of code that handles button presses:

```
    nil)

    SocketService.instance.getChannel { (success) in
        if success {
            self.tableView.reloadData()
        }
    }

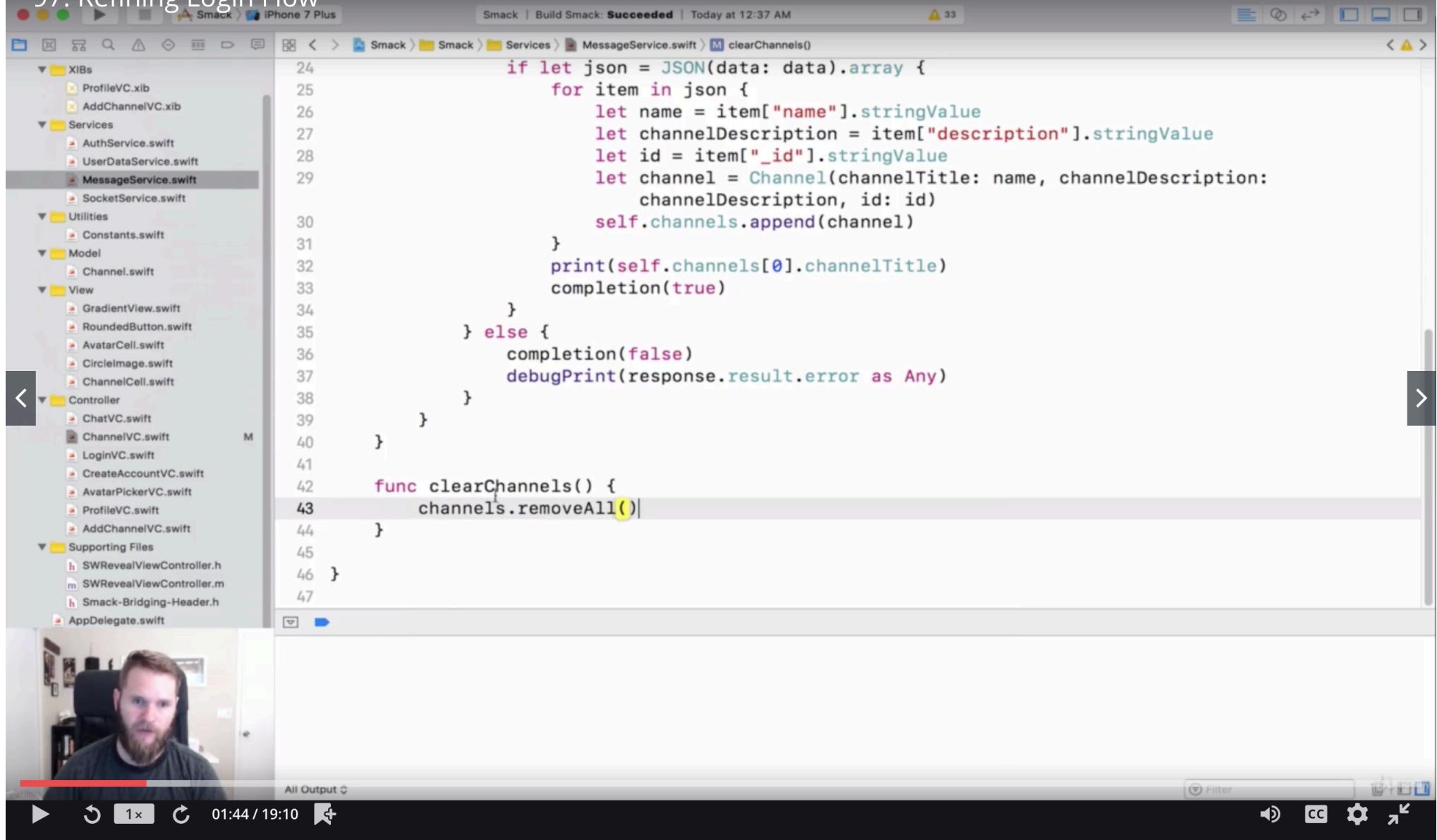
    override func viewDidAppear(_ animated: Bool) {
        setupUserInfo()
    }

    @IBAction func addChannelPressed(_ sender: Any) {
        if AuthService.instance.isLoggedIn {
            let addChannel = AddChannelVC()
            addChannel.modalPresentationStyle = .custom
            present(addChannel, animated: true, completion: nil)
        }
    }

    @IBAction func loginBtnPressed(_ sender: Any) {
        if AuthService.instance.isLoggedIn {
            let profile = ProfileVC()
            profile.modalPresentationStyle = .custom
        }
    }
}
```

The bottom of the screen shows a video feed of a person with a beard, indicating a live video call or recording. The Xcode status bar at the bottom includes icons for play/pause, zoom, and time.

## 97. Refining Login Flow



The screenshot shows the Xcode interface with the following details:

- Project Navigator:** Shows the project structure with files like XIBs, Services, Utilities, Model, View, Controller, and Supporting Files.
- Editor:** Displays the `MessageService.swift` file content. The code handles JSON parsing and channel management.
- Output Navigator:** Shows the output "Build Smack: Succeeded" and the date "Today at 12:37 AM".
- Bottom Bar:** Includes standard Xcode controls like play/pause, zoom, and a timestamp "01:44 / 19:10".

```
24     if let json = JSON(data: data).array {
25         for item in json {
26             let name = item["name"].stringValue
27             let channelDescription = item["description"].stringValue
28             let id = item["_id"].stringValue
29             let channel = Channel(channelTitle: name, channelDescription:
30                                   channelDescription, id: id)
31             self.channels.append(channel)
32         }
33         print(self.channels[0].channelTitle)
34         completion(true)
35     } else {
36         completion(false)
37         debugPrint(response.result.error as Any)
38     }
39 }
40 }
41
42 func clearChannels() {
43     channels.removeAll()
44 }
45
46 }
47
```

## 97. Refining Login Flow



A screenshot of an Xcode workspace titled "Smack". The project structure on the left shows files like ProfileVC.xib, AddChannelVC.xib, AuthService.swift, UserDataService.swift, MessageService.swift, SocketService.swift, Constants.swift, Channel.swift, GradientView.swift, RoundedButton.swift, AvatarCell.swift, CircleImage.swift, ChannelCell.swift, ChatVC.swift, ChannelVC.swift, LoginVC.swift, CreateAccountVC.swift, AvatarPickerVC.swift, ProfileVC.swift, and AddChannelVC.swift. The "MessageService.swift" file is open in the editor. The code starts with a multi-line comment and imports Foundation, Alamofire, and SwiftyJSON. It defines a class "MessageService" with a static let instance. A red box highlights the declaration of "var selectedChannel : Channel?". A red banner at the top right of the code area states: "Class 'MessageService' has no initializers" and "MessageService' cannot be constructed because it has no accessible...". The bottom of the screen shows a video player with a play button, volume controls, and a timestamp of 02:13 / 19:10.

```
1 // MessageService.swift
2 // Smack
3 //
4 //
5 // Created by Jonny B on 7/18/17.
6 // Copyright © 2017 Jonny B. All rights reserved.
7 //
8
9 import Foundation
10 import Alamofire
11 import SwiftyJSON
12
13 class MessageService {
14
15     static let instance = MessageService() // 'MessageService' cannot be constructed because it has no accessible...
16
17     var channels = [Channel]()
18     var selectedChannel : Channel?
19
20     func findAllChannel(completion: @escaping CompletionHandler) {
21         Alamofire.request(URL_GET_CHANNELS, method: .get, parameters: nil, encoding: JSONEncoding.default, headers: BEARER_HEADER).responseJSON { (response) in
22
23             if response.result.error == nil {
24                 guard let data = response.data else { return }
```

## 97. Refining Login Flow

The screenshot shows the Xcode interface with the project "Smack" open. The file structure on the left includes XIB files, Services (AuthService.swift, UserDataService.swift), Utilities (Constants.swift), Model (Channel.swift), View (GradientView.swift, RoundedButton.swift, AvatarCell.swift, CircleImage.swift, ChannelCell.swift), Controller (ChatVC.swift, ChannelVC.swift, LoginVC.swift, CreateAccountVC.swift, AvatarPickerVC.swift, ProfileVC.swift, AddChannelVC.swift), and Supporting Files (SWRevealViewController.h, SWRevealViewController.m, Smack-Bridging-Header.h, AppDelegate.swift). The current file is UserDataService.swift, specifically the logoutUser() function. The code is as follows:

```
61    }
62
63    func logoutUser() {
64        id = ""
65        avatarName = ""
66        avatarColor = ""
67        email = ""
68        name = ""
69        AuthService.instance.isLoggedIn = false
70        AuthService.instance.userEmail = ""
71        AuthService.instance.authToken = ""
72        MessageService.instance.clearChannels()
73    }
74
75
76
77
78
79
80
81
82
83
84
85
```

The line `MessageService.instance.clearChannels()` is highlighted with a red rectangle.

At the bottom of the screen, there is a video feed of a man with a beard, indicating a live video call or recording.

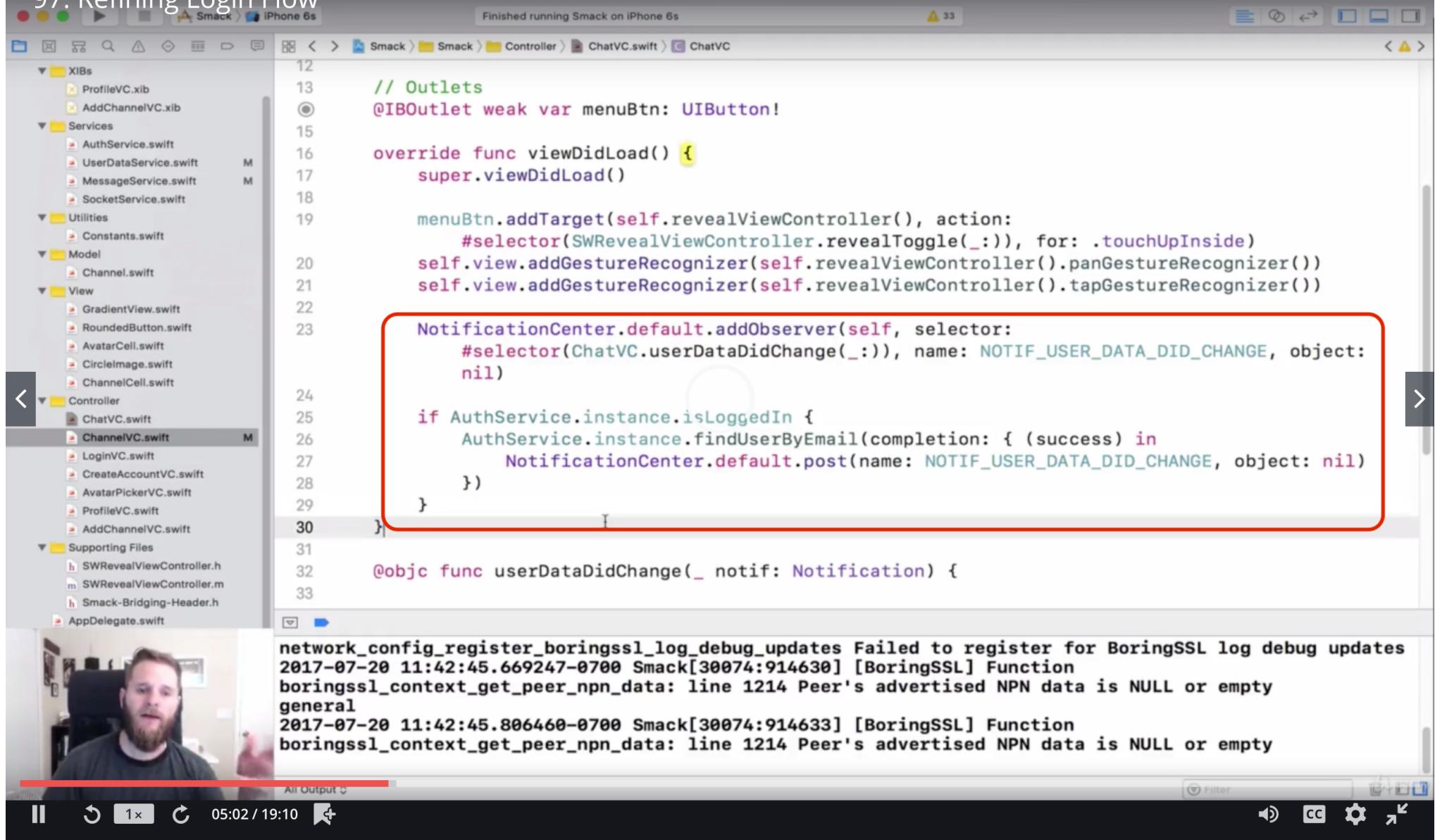
## 97. Refining Login Flow

The screenshot shows the Xcode interface with the project navigation bar at the top. The main area displays the `ChannelVC.swift` file under the `Controller` folder. The code is as follows:

```
49         present(profile, animated: true, completion: nil)
50     } else {
51         performSegue(withIdentifier: TO_LOGIN, sender: nil)
52     }
53 }
54
55 @objc func userDataDidChange(_ notif: Notification) {
56     setupUserInfo()
57 }
58
59 func setupUserInfo() {
60     if AuthService.instance.isLoggedIn {
61         loginBtn.setTitle(UserDataService.instance.name, for: .normal)
62         userImg.image = UIImage(named: UserDataService.instance.avatarName)
63         userImg.backgroundColor = UserDataService.instance.returnUIColor(components:
64             UserDataService.instance.avatarColor)
65     } else {
66         loginBtn.setTitle("Login", for: .normal)
67         userImg.image = UIImage(named: "menuProfileIcon")
68         userImg.backgroundColor = UIColor.clear
69         tableView.reloadData()
70     }
71 }
72
73 func tableView(_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell
```

A red box highlights the `tableView.reloadData()` call at line 68. In the bottom-left corner, there is a video feed of a person with a beard. The bottom of the screen shows the Xcode toolbar with various icons.

## 97. Refining Login Flow



The screenshot shows the Xcode interface with the project navigation bar at the top. Below it is the file browser showing the project structure, including XIB files, Services, Utilities, Model, View, and Controller folders. The ChatVC.swift file is selected in the controller folder. The main editor area contains Swift code for the ChatVC class. A red box highlights a specific block of code related to user data change notifications:

```
12 // Outlets
13 @IBOutlet weak var menuBtn: UIButton!
14
15 override func viewDidLoad() {
16     super.viewDidLoad()
17
18     menuBtn.addTarget(self.revealViewController(), action:
19         #selector(SWRevealViewController.revealToggle(_:)), for: .touchUpInside)
20     self.view.addGestureRecognizer(self.revealViewController().panGestureRecognizer())
21     self.view.addGestureRecognizer(self.revealViewController().tapGestureRecognizer())
22
23     NotificationCenter.default.addObserver(self, selector:
24         #selector(ChatVC.userDataDidChange(_:)), name: NOTIF_USER_DATA_DID_CHANGE, object:
25         nil)
26
27     if AuthService.instance.isLoggedIn {
28         AuthService.instance.findUserByEmail(completion: { (success) in
29             NotificationCenter.default.post(name: NOTIF_USER_DATA_DID_CHANGE, object: nil)
30         })
31     }
32
33     @objc func userDataDidChange(_ notif: Notification) {
34
35         network_config_register_boringssl_log_debug_updates Failed to register for BoringSSL log debug updates
36         2017-07-20 11:42:45.669247-0700 Smack[30074:914630] [BoringSSL] Function
37         boringssl_context_get_peer_npn_data: line 1214 Peer's advertised NPN data is NULL or empty
38         general
39         2017-07-20 11:42:45.806460-0700 Smack[30074:914633] [BoringSSL] Function
40         boringssl_context_get_peer_npn_data: line 1214 Peer's advertised NPN data is NULL or empty
41     }
42 }
```

The bottom output pane shows several error messages related to SSL/TLS configuration and BoringSSL.

## 97. Refining Login Flow

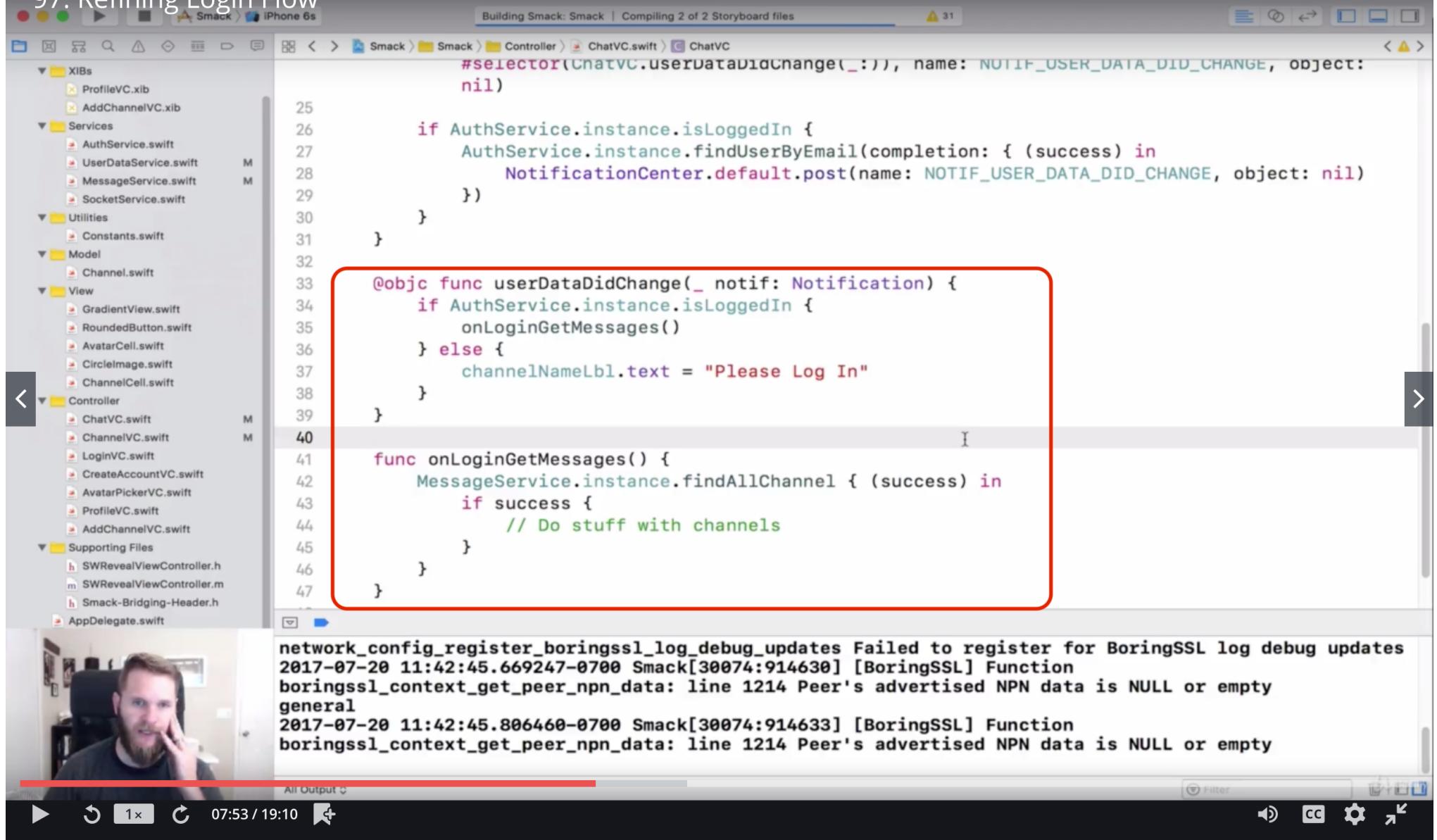
Building Smack: Socket.IO-Client-Swift | Compiling Swift source files 33

The screenshot shows the Xcode interface with the following components:

- Project Navigator:** Shows files like XIBs (ProfileVC.xib, AddChannelVC.xib), Services (AuthService.swift, UserDataService.swift, MessageService.swift, SocketService.swift), Utilities (Constants.swift), and Controller (ChatVC.swift, ChannelVC.swift, LoginVC.swift, CreateAccountVC.swift, AvatarPickerVC.swift, ProfileVC.swift, AddChannelVC.swift).
- Editor:** Displays the `ChatVC.swift` file with code related to UIKit and SWRevealViewController.
- Document Outline:** Shows the hierarchy of scenes and view controllers, including `ChannelVC Scene`, `LoginVC Scene`, `Create AccountVC S...`, `Avatar PickerVC Scene`, and `ChatVC Scene`.
- Storyboard Preview:** Shows a preview of the storyboard with a red arrow pointing to the top navigation bar of a simulated iPhone 7 screen.
- Output Window:** Shows log messages related to SSL configuration and BoringSSL.
- Bottom Bar:** Includes standard Xcode controls like play/pause, zoom, and search.

```
4 // Created by Jonny B on 7/14/17.
5 // Copyright © 2017 Jonny B. All rights reserved.
6
7
8
9 import UIKit
10
11 class ChatVC: UIViewController {
12
13     // Outlets
14     @IBOutlet weak var menuBtn: UIButton!
15
16     override func viewDidLoad() {
17         super.viewDidLoad()
18
19             menuBtn.addTarget(self.revealViewController(), action:
20                 #selector(SWRevealViewController.revealToggle(_:)),
21                 for: .touchUpInside)
22
23             self.view.addGestureRecognizer(self.revealViewController().panGesture
24
25
26             network_config_register_boringssl_log_debug_updates Failed to register for BoringSSL log debug updates
27 2017-07-20 11:42:45.669247-0700 Smack[30074:914630] [BoringSSL] Function
28 boringssl_context_get_peer_npn_data: line 1214 Peer's advertised NPN data is NULL or empty
29 general
30 2017-07-20 11:42:45.806460-0700 Smack[30074:914633] [BoringSSL] Function
31 boringssl_context_get_peer_npn_data: line 1214 Peer's advertised NPN data is NULL or empty
```

## 97. Refining Login Flow



The screenshot shows the Xcode interface with the project "Smack" open. The file "ChatVC.swift" is selected in the navigation bar. The code editor displays the following Swift code:

```
Building Smack: Smack | Compiling 2 of 2 Storyboard files 31
...
#selector(ChatVC.userDataDidChange(_:)), name: NOTIF_USER_DATA_DID_CHANGE, object: nil)

if AuthService.instance.isLoggedIn {
    AuthService.instance.findUserByEmail(completion: { (success) in
        NotificationCenter.default.post(name: NOTIF_USER_DATA_DID_CHANGE, object: nil)
    })
}

@objc func userDataDidChange(_ notif: Notification) {
    if AuthService.instance.isLoggedIn {
        onLoginGetMessages()
    } else {
        channelNameLbl.text = "Please Log In"
    }
}

func onLoginGetMessages() {
    MessageService.instance.findAllChannel { (success) in
        if success {
            // Do stuff with channels
        }
    }
}

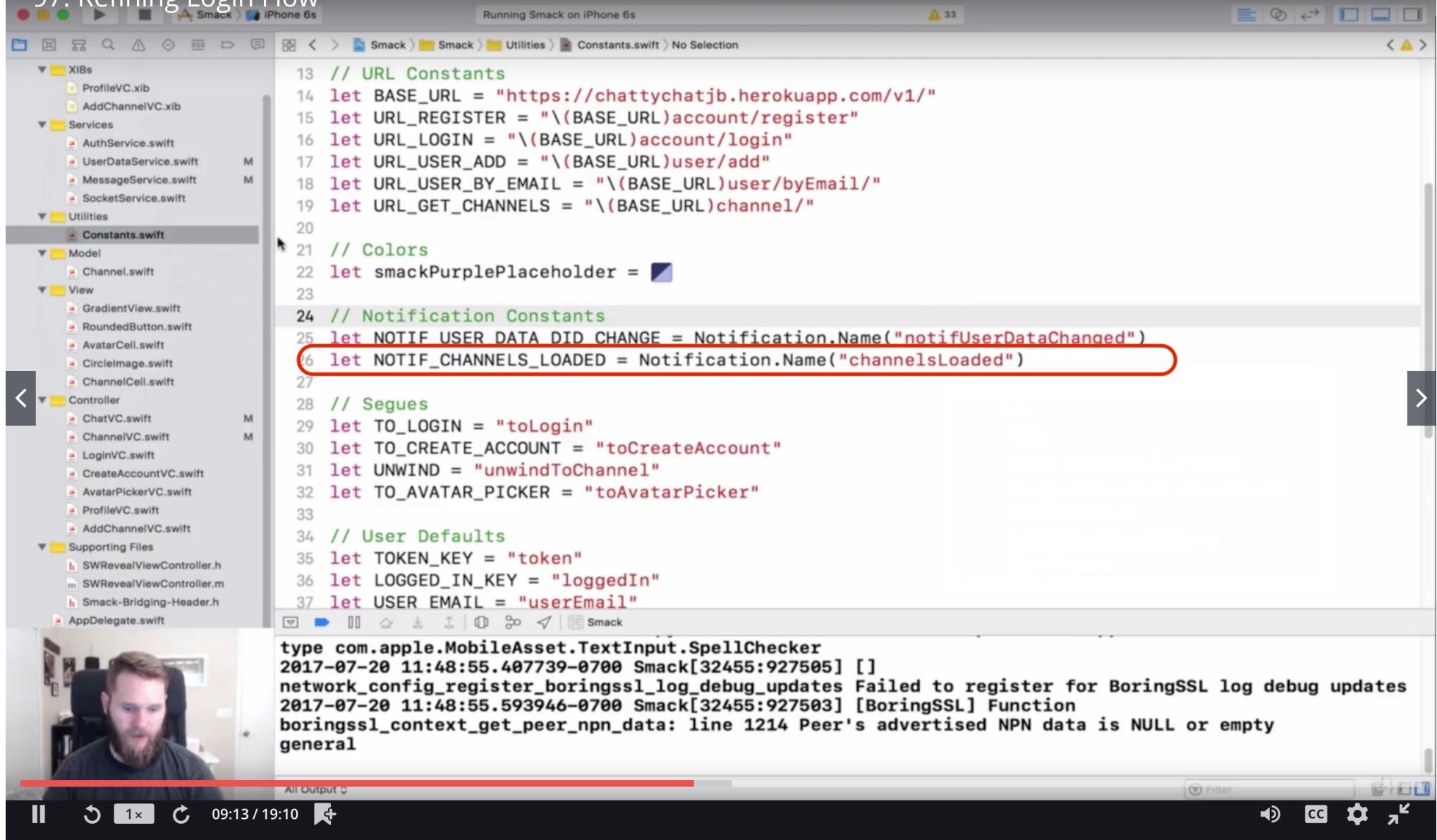
```

A red box highlights the `@objc func userDataDidChange(_ notif: Notification) {}` block. The bottom pane shows the "Output" tab with the following log messages:

```
network_config_register_boringssl_log_debug_updates Failed to register for BoringSSL log debug updates
2017-07-20 11:42:45.669247-0700 Smack[30074:914630] [BoringSSL] Function
boringssl_context_get_peer_npn_data: line 1214 Peer's advertised NPN data is NULL or empty
general
2017-07-20 11:42:45.806460-0700 Smack[30074:914633] [BoringSSL] Function
boringssl_context_get_peer_npn_data: line 1214 Peer's advertised NPN data is NULL or empty
```

The bottom left corner shows a video feed of a person speaking.

## 97. Refining Login Flow



The screenshot shows the Xcode interface with the project "Smack" running on an iPhone 6s. The left sidebar displays the project structure, and the main editor shows the `Constants.swift` file. Two specific lines of code are highlighted with a red oval:

```
13 // URL Constants
14 let BASE_URL = "https://chattychatjb.herokuapp.com/v1/"
15 let URL_REGISTER = "\(BASE_URL)account/register"
16 let URL_LOGIN = "\(BASE_URL)account/login"
17 let URL_USER_ADD = "\(BASE_URL)user/add"
18 let URL_USER_BY_EMAIL = "\(BASE_URL)user/byEmail/"
19 let URL_GET_CHANNELS = "\(BASE_URL)channel/"

20
21 // Colors
22 let smackPurplePlaceholder = □

23
24 // Notification Constants
25 let NOTIF_USER_DATA_DID_CHANGE = Notification.Name("notifUserDataChanged")
26 let NOTIF_CHANNELS_LOADED = Notification.Name("channelsLoaded")  
27
28 // Segues
29 let TO_LOGIN = "toLogin"
30 let TO_CREATE_ACCOUNT = "toCreateAccount"
31 let UNWIND = "unwindToChannel"
32 let TO_AVATAR_PICKER = "toAvatarPicker"
33
34 // User Defaults
35 let TOKEN_KEY = "token"
36 let LOGGED_IN_KEY = "loggedIn"
37 let USER_EMAIL = "userEmail"

type com.apple.MobileAsset.TextInput.SpellChecker
2017-07-20 11:48:55.407739-0700 Smack[32455:927505] []
network_config_register_boringssl_log_debug_updates Failed to register for BoringSSL log debug updates
2017-07-20 11:48:55.593946-0700 Smack[32455:927503] [BoringSSL] Function
boringssl_context_get_peer_npn_data: line 1214 Peer's advertised NPN data is NULL or empty
general
```

The bottom output window shows some log messages related to SSL/TLS configuration.

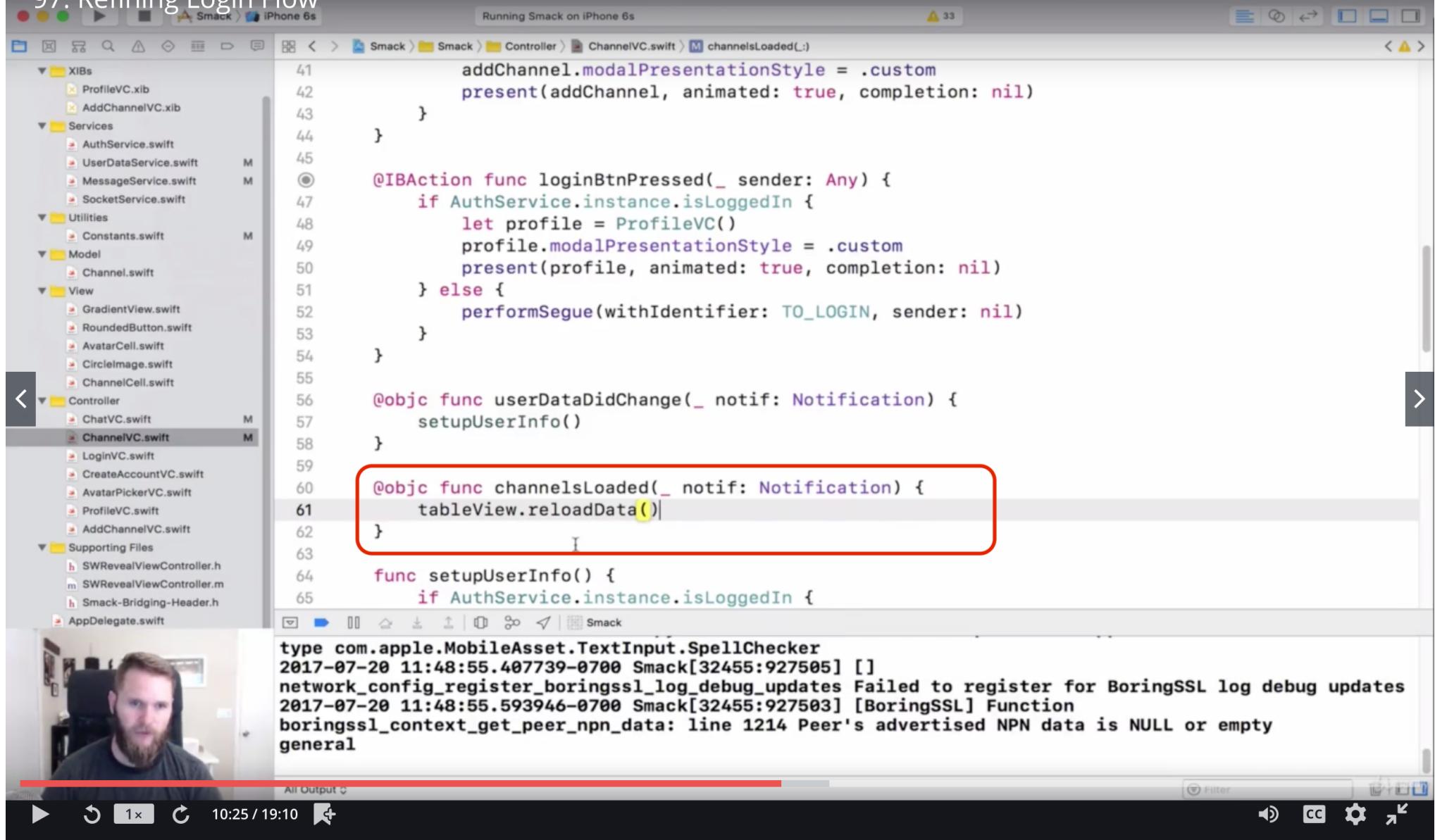
## 97. Refining Login Flow

The screenshot shows the Xcode IDE with the following details:

- Project Structure:** The left sidebar shows the project structure with files like XIBs, Services, Utilities, Model, View, Controller, and Supporting Files.
- Code Editor:** The main editor displays the `MessageService.swift` file. A specific line of code is highlighted with a red rectangle:

```
NotificationCenter.default.post(name: NOTIF_CHANNELS_LOADED, object: nil)
```
- Terminal Output:** Below the editor is a terminal window showing logs from the application's run. The logs include:type com.apple.MobileAsset.TextInput.SpellChecker  
2017-07-20 11:48:55.407739-0700 Smack[32455:927505] []  
network\_config\_register\_boringssl\_log\_debug\_updates Failed to register for BoringSSL log debug updates  
2017-07-20 11:48:55.593946-0700 Smack[32455:927503] [BoringSSL] Function  
boringssl\_context\_get\_peer\_npn\_data: line 1214 Peer's advertised NPN data is NULL or empty  
general
- Bottom Bar:** The Xcode bottom bar includes icons for play/pause, refresh, zoom, and a timestamp of 09:38 / 19:10.

## 97. Refining Login Flow



The screenshot shows the Xcode interface with the project navigation bar at the top. Below it is the file browser showing the project structure, including XIB files, Services, Utilities, Model, View, and Controller folders. The Controller folder contains several Swift files, with `ChannelVC.swift` currently selected and highlighted in grey. The main editor area displays the code for `ChannelVC.swift`. A red box highlights the `@objc func channelsLoaded(_ notif: Notification) { tableView.reloadData()}` line. The bottom of the screen shows the Xcode status bar with various icons and the current time.

```
Running Smack on iPhone 6s 33
Smack > Smack > Controller > ChannelVC.swift > channelsLoaded(_)

41     addChannel.modalPresentationStyle = .custom
42     present(addChannel, animated: true, completion: nil)
43 }
44 }

@IBAction func loginBtnPressed(_ sender: Any) {
46     if AuthService.instance.isLoggedIn {
47         let profile = ProfileVC()
48         profile.modalPresentationStyle = .custom
49         present(profile, animated: true, completion: nil)
50     } else {
51         performSegue(withIdentifier: TO_LOGIN, sender: nil)
52     }
53 }

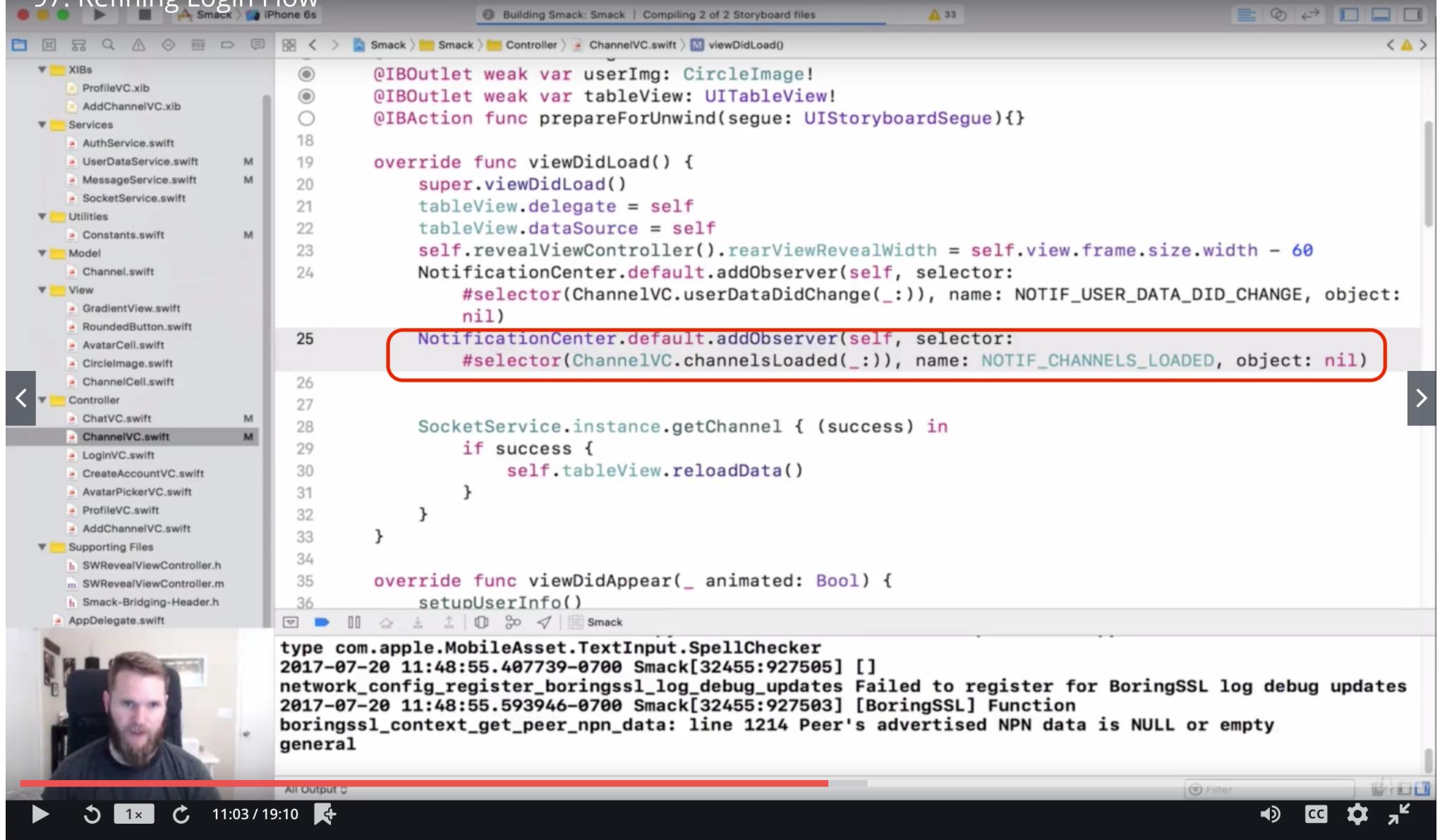
@objc func userDataDidChange(_ notif: Notification) {
55     setupUserInfo()
56 }

@objc func channelsLoaded(_ notif: Notification) {
58     tableView.reloadData()
59 }

func setupUserInfo() {
60     if AuthService.instance.isLoggedIn {
61
62
63
64
65
type com.apple.MobileAsset.TextInput.SpellChecker
2017-07-20 11:48:55.407739-0700 Smack[32455:927505] []
network_config_register_boringssl_log_debug_updates Failed to register for BoringSSL log debug updates
2017-07-20 11:48:55.593946-0700 Smack[32455:927503] [BoringSSL] Function
boringssl_context_get_peer_npn_data: line 1214 Peer's advertised NPN data is NULL or empty
general
```

All Output Filter 1x 10:25 / 19:10 CC

## 97. Refining Login Flow



The screenshot shows the Xcode interface with the project navigation bar at the top. Below it is the file browser showing the project structure, including XIB files, Services, Utilities, Model, View, and Controller folders. The Controller folder contains several Swift files, with `ChannelVC.swift` currently selected and highlighted.

The main editor area displays the `viewDidLoad()` method of `ChannelVC.swift`. A red box highlights the following code block:

```
NotificationCenter.default.addObserver(self, selector:  
    #selector(ChannelVC.channelsLoaded(_)), name: NOTIF_CHANNELS_LOADED, object: nil)
```

Below the code editor is the output pane, which shows the following log entries:

```
type com.apple.MobileAsset.TextInput.SpellChecker  
2017-07-20 11:48:55.407739-0700 Smack[32455:927505] []  
network_config_register_boringssl_log_debug_updates Failed to register for BoringSSL log debug updates  
2017-07-20 11:48:55.593946-0700 Smack[32455:927503] [BoringSSL] Function  
boringssl_context_get_peer_npn_data: line 1214 Peer's advertised NPN data is NULL or empty  
general
```

At the bottom of the screen, there is a video feed of a person with a beard, indicating a video conference or screen recording session.

## 97. Refining Login Flow

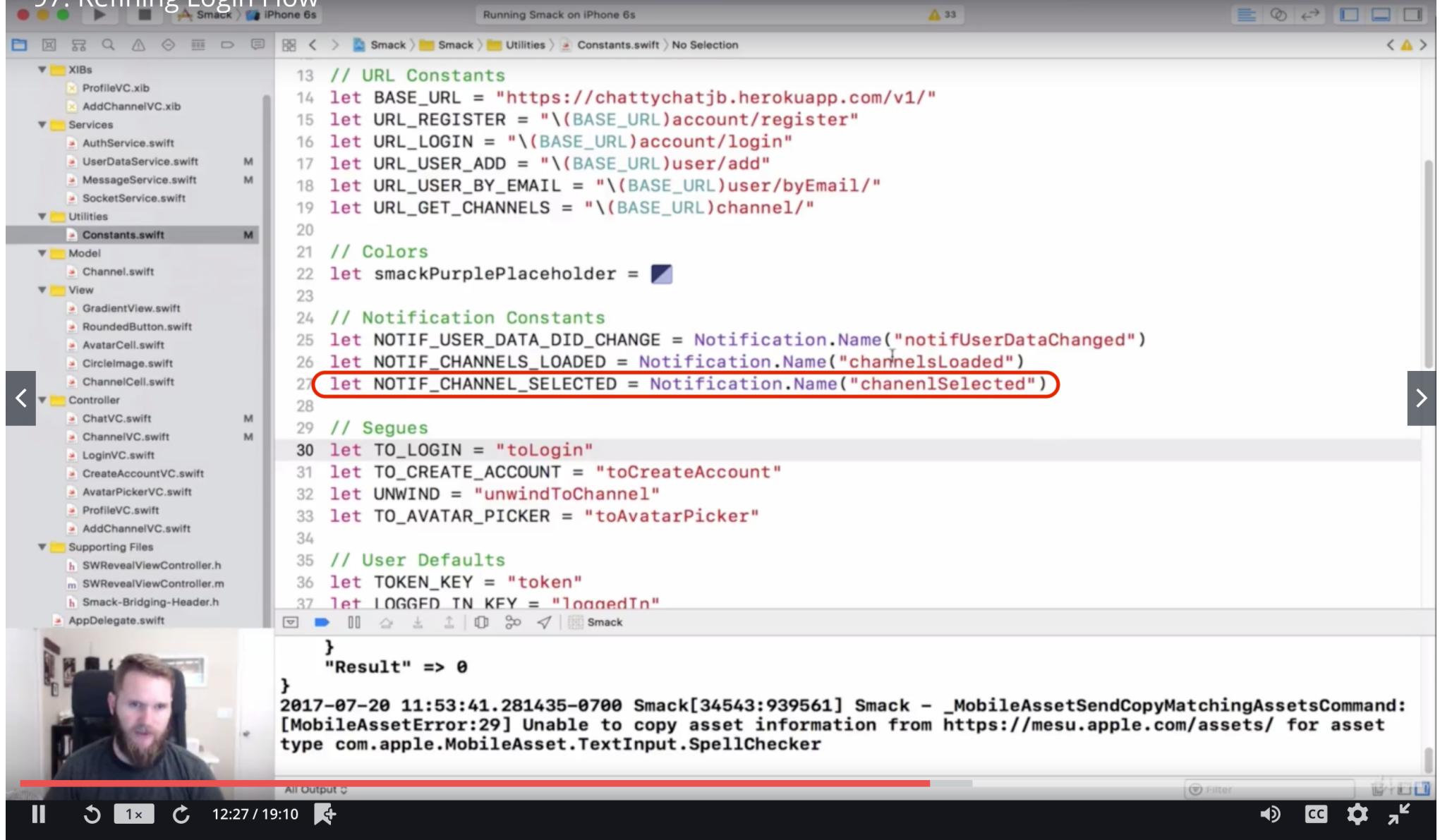
Running Smack on iPhone 6s

The screenshot shows the Xcode interface with the following details:

- Project Structure:** The left sidebar shows the project structure with files like ProfileVC.xib, AuthService.swift, and ChatVC.swift.
- Code Editor:** The main editor shows the `ChannelVC.swift` file with Swift code for managing channels and user profiles.
- Simulator:** An iPhone 6s simulator is running the application. The screen displays a list of channels: #general, #random, #dev, #Devslopes, #socket test, and #socket are bomb. A profile card for "Emma" is visible at the bottom.
- Output Log:** The bottom right shows the log output:

```
2017-07-20 11:53:41.281435-0700 Smack[MobileAssetError:29] Unable to copy asset type com.apple.MobileAsset.TextInput.Swift
```
- Bottom Bar:** The Xcode toolbar at the bottom includes icons for play, stop, zoom, and other development tools.

## 97. Refining Login Flow



The screenshot shows the Xcode interface with the project navigation bar at the top. Below it is the file browser showing the project structure. The main editor area contains the `Constants.swift` file. A red box highlights the line `let NOTIF_CHANNEL_SELECTED = Notification.Name("chanenlSelected")`. The bottom of the screen shows the output log with a recent error message.

```
13 // URL Constants
14 let BASE_URL = "https://chattychatjb.herokuapp.com/v1/"
15 let URL_REGISTER = "\(BASE_URL)account/register"
16 let URL_LOGIN = "\(BASE_URL)account/login"
17 let URL_USER_ADD = "\(BASE_URL)user/add"
18 let URL_USER_BY_EMAIL = "\(BASE_URL)user/byEmail/"
19 let URL_GET_CHANNELS = "\(BASE_URL)channel/"

20
21 // Colors
22 let smackPurplePlaceholder = □

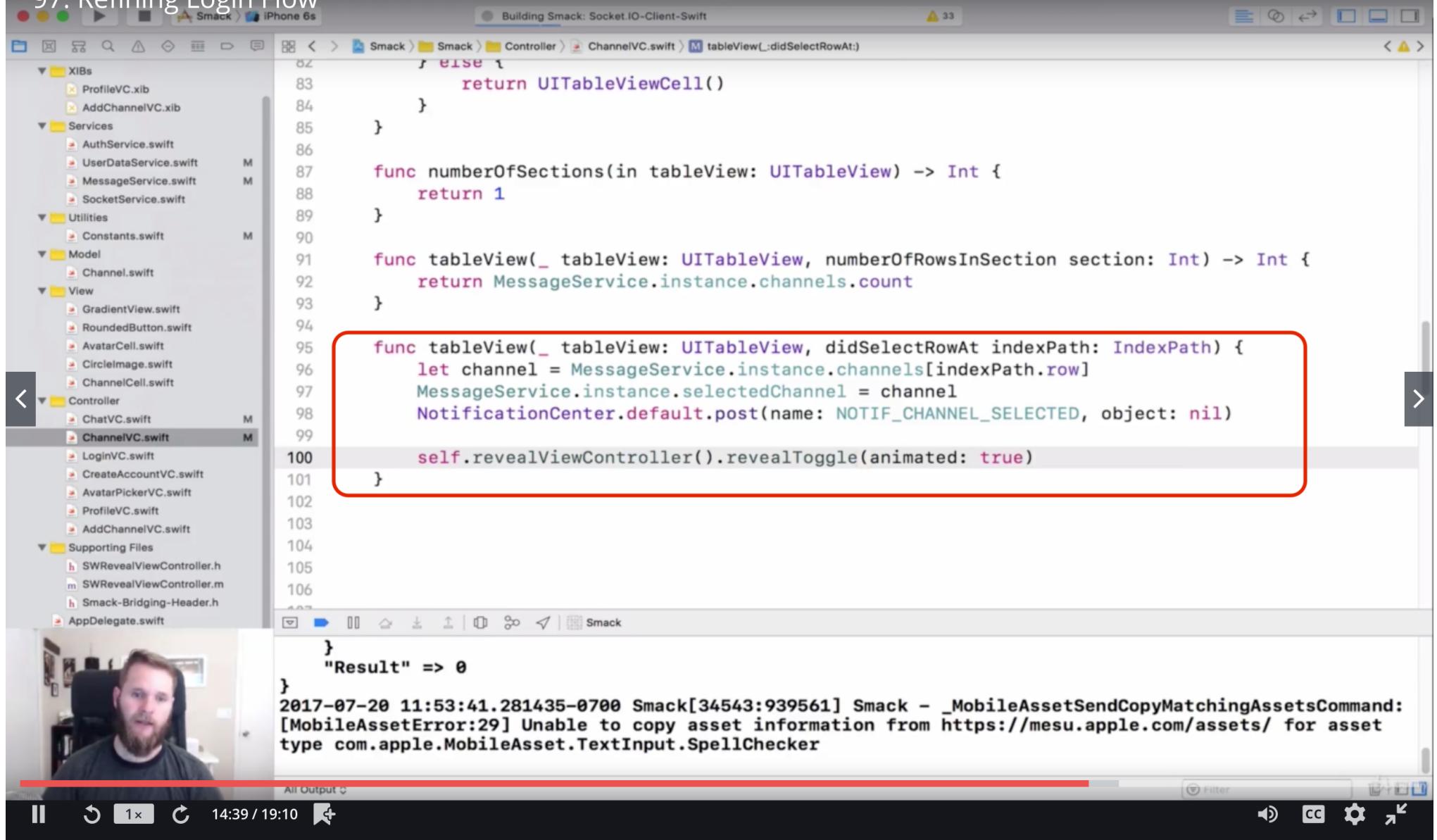
23
24 // Notification Constants
25 let NOTIF_USER_DATA_DID_CHANGE = Notification.Name("notifUserDataChanged")
26 let NOTIF_CHANNELS_LOADED = Notification.Name("channelsLoaded")
27 let NOTIF_CHANNEL_SELECTED = Notification.Name("chanenlSelected") (highlighted)
28
29 // Segues
30 let TO_LOGIN = "toLogin"
31 let TO_CREATE_ACCOUNT = "toCreateAccount"
32 let UNWIND = "unwindToChannel"
33 let TO_AVATAR_PICKER = "toAvatarPicker"
34
35 // User Defaults
36 let TOKEN_KEY = "token"
37 let LOGGED_IN_KEY = "loggedIn"

}
"Result" => 0
}

2017-07-20 11:53:41.281435-0700 Smack[34543:939561] Smack - _MobileAssetSendCopyMatchingAssetsCommand:
[MobileAssetError:29] Unable to copy asset information from https://mesu.apple.com/assets/ for asset
type com.apple.MobileAsset.TextInput.SpellChecker
```

The output log at the bottom shows a warning about an asset transfer failure.

## 97. Refining Login Flow



The screenshot shows the Xcode interface with the project "Smack" open. The left sidebar displays the project structure, including XIB files, Services, Utilities, Model, View, and Controller. The "Controller" folder contains several Swift files, with "ChannelVC.swift" selected and highlighted in blue. The main editor area shows the code for "ChannelVC.swift". A red box highlights the following code block:

```
func tableView(_ tableView: UITableView, didSelectRowAt indexPath: IndexPath) {
    let channel = MessageService.instance.channels[indexPath.row]
    MessageService.instance.selectedChannel = channel
    NotificationCenter.default.post(name: NOTIF_CHANNEL_SELECTED, object: nil)

    self.revealViewController().revealToggle(animated: true)
}
```

The code handles a table view selection event. It retrieves the selected channel from the MessageService, sets it as the selected channel, posts a notification, and then reveals the view controller using the SWRevealViewController API.

At the bottom of the screen, the Xcode status bar shows the date and time (2017-07-20 11:53:41.281435-0700), the project name (Smack), and a log message: "2017-07-20 11:53:41.281435-0700 Smack[34543:939561] Smack - \_MobileAssetSendCopyMatchingAssetsCommand: [MobileAssetError:29] Unable to copy asset information from https://mesu.apple.com/assets/ for asset type com.apple.MobileAsset.TextInput.SpellChecker".

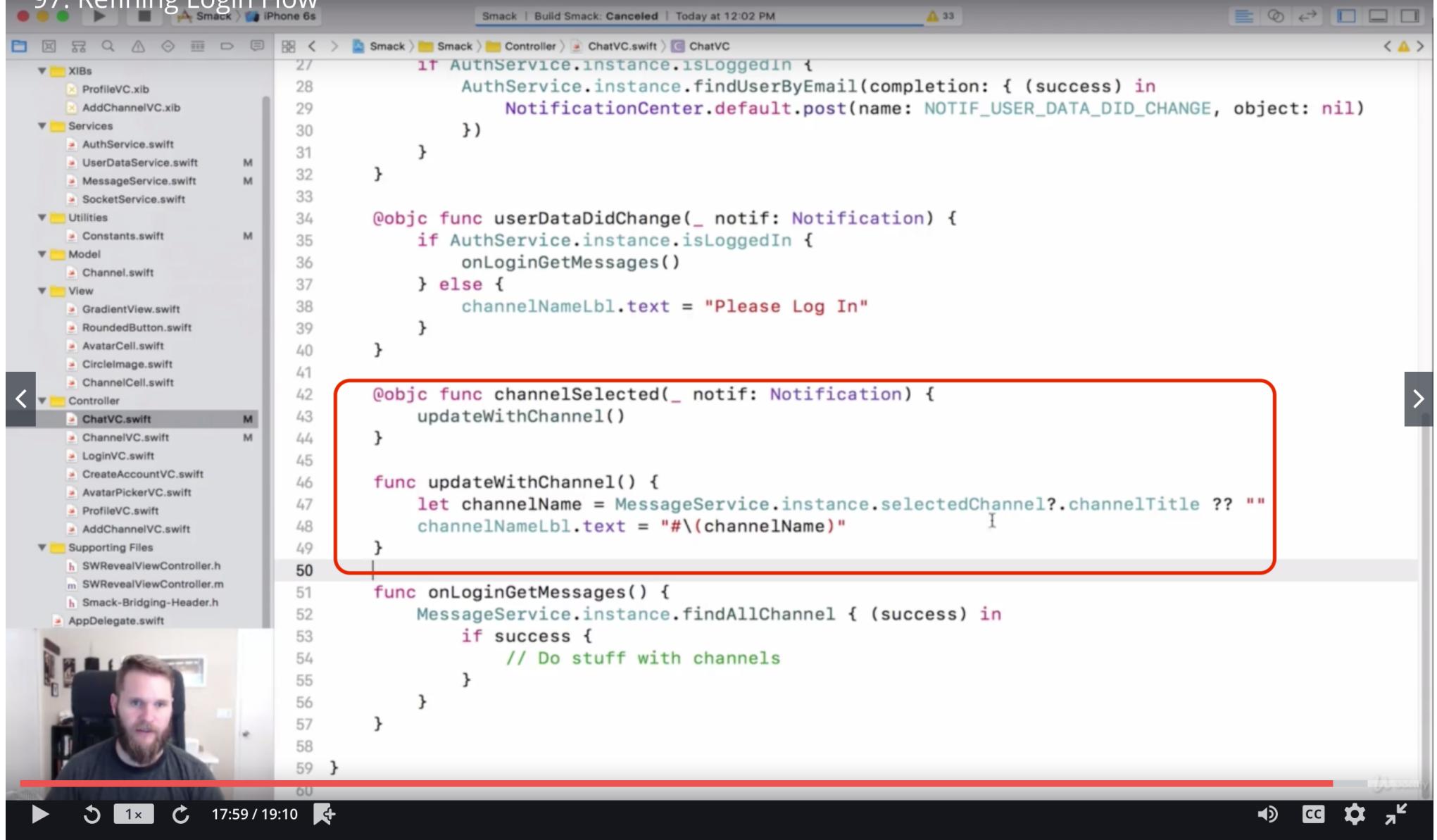
## 97. Refining Login Flow

The screenshot shows the Xcode interface with the project navigation bar at the top. The main area displays the `ChatVC.swift` file under the `Controller` folder. The code is annotated with line numbers and highlights:

```
21     #selector(SWRevealViewController.revealToggle(_:)), for: .touchUpInside)
22     self.view.addGestureRecognizer(self.revealViewController().panGestureRecognizer())
23     self.view.addGestureRecognizer(self.revealViewController().tapGestureRecognizer())
24
25     NotificationCenter.default.addObserver(self, selector:
26         #selector(ChatVC.userDataDidChange(_:)), name: NOTIF_USER_DATA_DID_CHANGE, object:
27             nil)
28    NotificationCenter.default.addObserver(self, selector:
29         #selector(ChatVC.channelSelected(_:)), name: NOTIF_CHANNEL_SELECTED, object: nil)
30
31     if AuthService.instance.isLoggedIn {
32         AuthService.instance.findUserByEmail(completion: { (success) in
33             NotificationCenter.default.post(name: NOTIF_USER_DATA_DID_CHANGE, object: nil)
34         })
35     }
36
37     @objc func userDataDidChange(_ notif: Notification) {
38         if AuthService.instance.isLoggedIn {
39             onLoginGetMessages()
40         } else {
41             channelNameLbl.text = "Please Log In"
42         }
43
44     @objc func channelSelected(_ notif: Notification) {
45
46     func onLoginGetMessages() {
47         MessageService.instance.findAllChannel { (success) in
48             if success {
49                 // Do stuff with channels
50             }
51         }
52     }
53
54     }
55
56     }
57
58     }
59
59 }
```

A red box highlights the line `NotificationCenter.default.addObserver(self, selector: #selector(ChatVC.channelSelected(_:)), name: NOTIF_CHANNEL_SELECTED, object: nil)`. The bottom of the screen shows a video feed of a person with a beard and a dark shirt, and the Xcode status bar at the bottom.

## 97. Refining Login Flow



The screenshot shows the Xcode interface with the project navigation bar at the top. The main area displays the code for `ChatVC.swift`. A red box highlights the `updateWithChannel()` function, which retrieves the selected channel name from `MessageService` and updates the `channelNameLbl` label.

```
27     if AuthService.instance.isLoggedIn {
28         AuthService.instance.findUserByEmail(completion: { (success) in
29             NotificationCenter.default.post(name: NOTIF_USER_DATA_DID_CHANGE, object: nil)
30         })
31     }
32 }
33
34 @objc func userDataDidChange(_ notif: Notification) {
35     if AuthService.instance.isLoggedIn {
36         onLoginGetMessages()
37     } else {
38         channelNameLbl.text = "Please Log In"
39     }
40 }
41
42 @objc func channelSelected(_ notif: Notification) {
43     updateWithChannel()
44 }
45
46 func updateWithChannel() {
47     let channelName = MessageService.instance.selectedChannel?.channelTitle ?? ""
48     channelNameLbl.text = "#\u{1}(channelName)"
49 }
50
51 func onLoginGetMessages() {
52     MessageService.instance.findAllChannel { (success) in
53         if success {
54             // Do stuff with channels
55         }
56     }
57 }
58
59 }
```