

## CSCI 4140 – Tutorial 8

# Assignment 2 Overview (Server Side)

Matt YIU, Man Tung ([mtyiucse](mailto:mtyiucse@gmail.com))

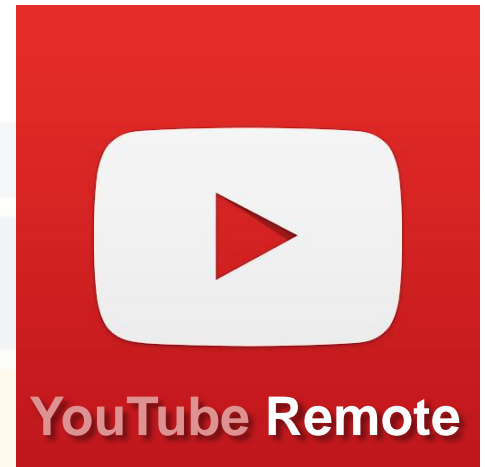
SHB 118

*Office Hour: Wednesday, 3-5 pm*

2016.03.17

# Outline

- Work flow
- Server side
  - Routing
  - Message forwarding

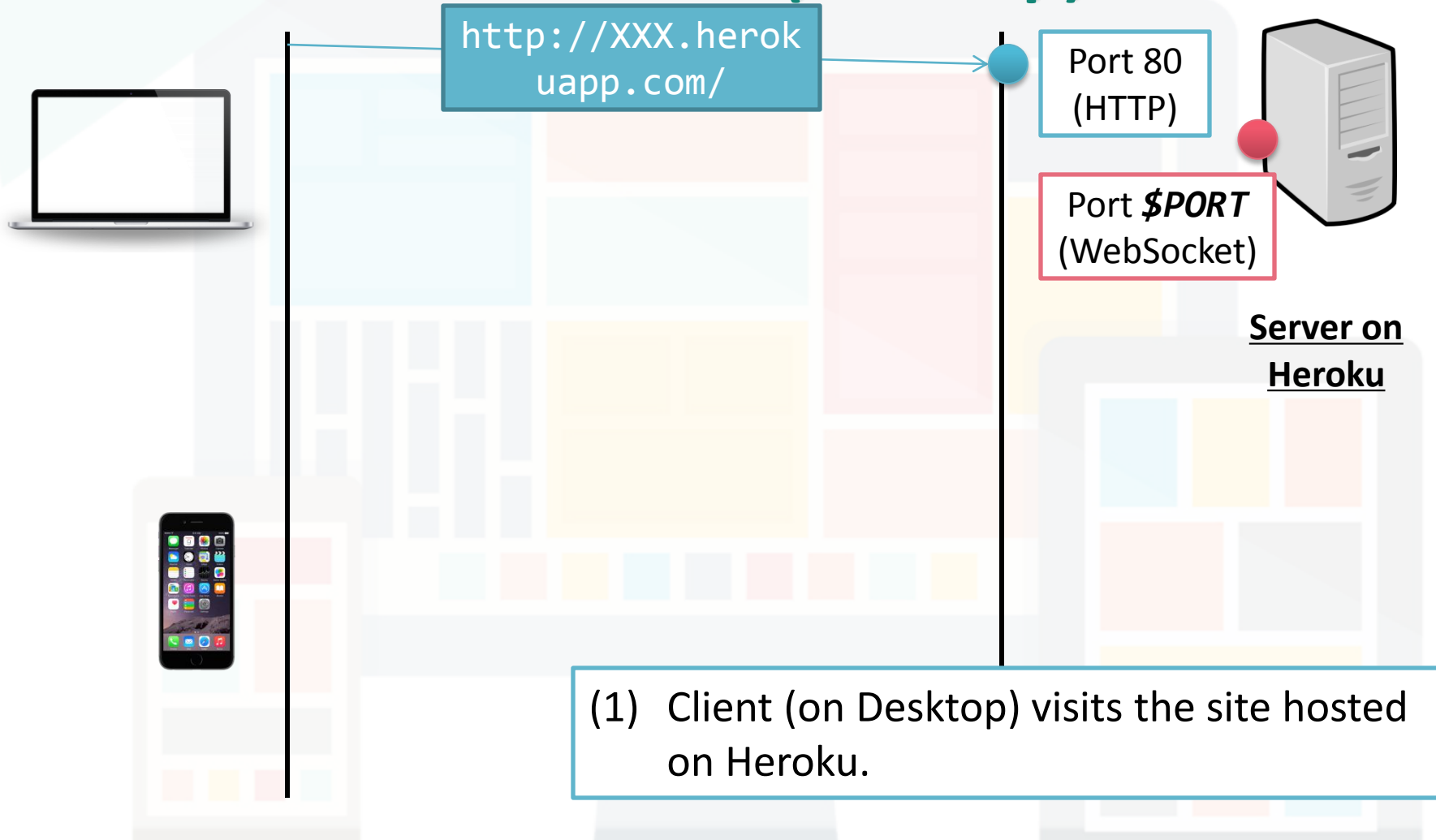




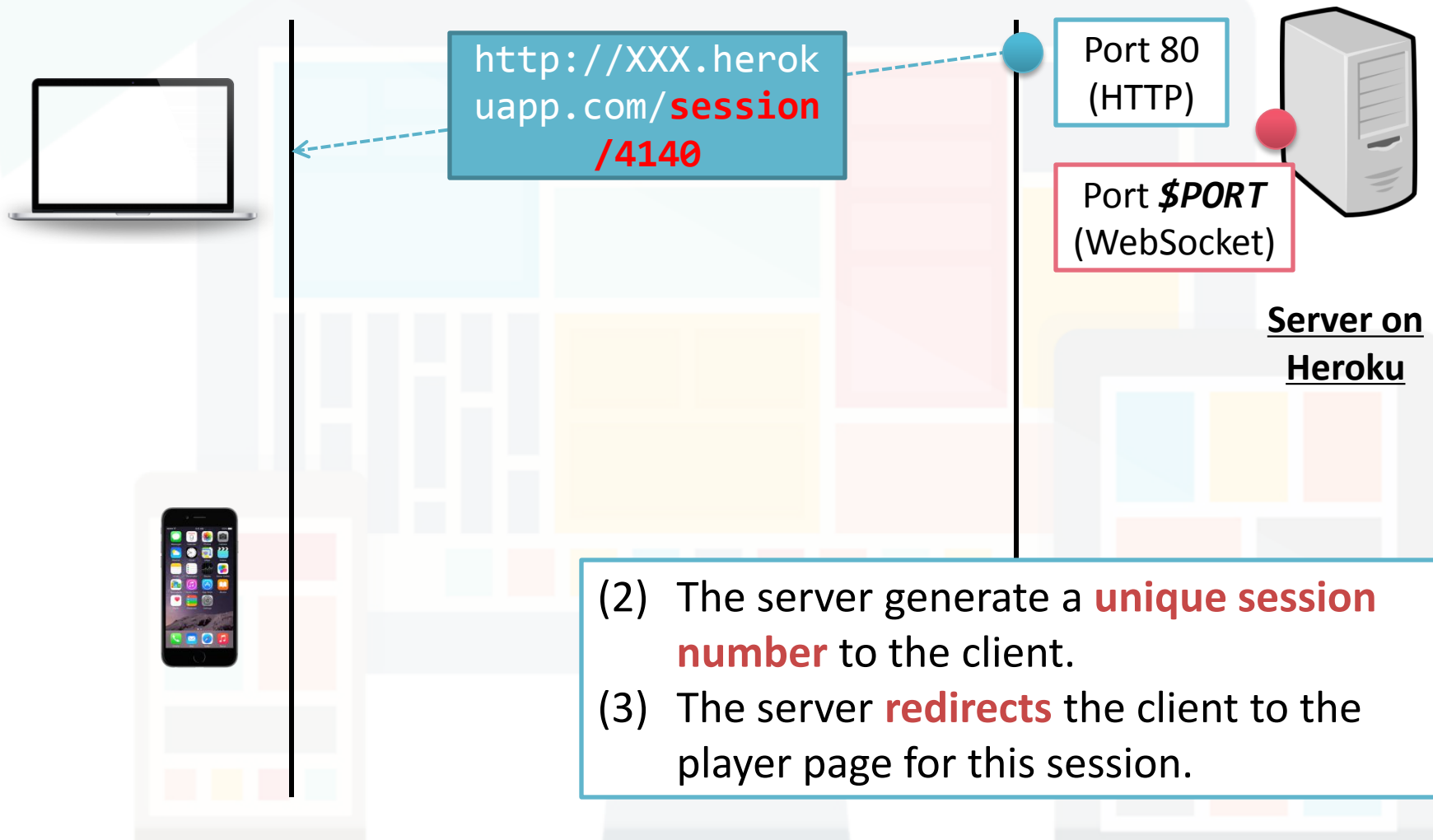
# Work flow

*Let's define what will happen between the clients and server...*

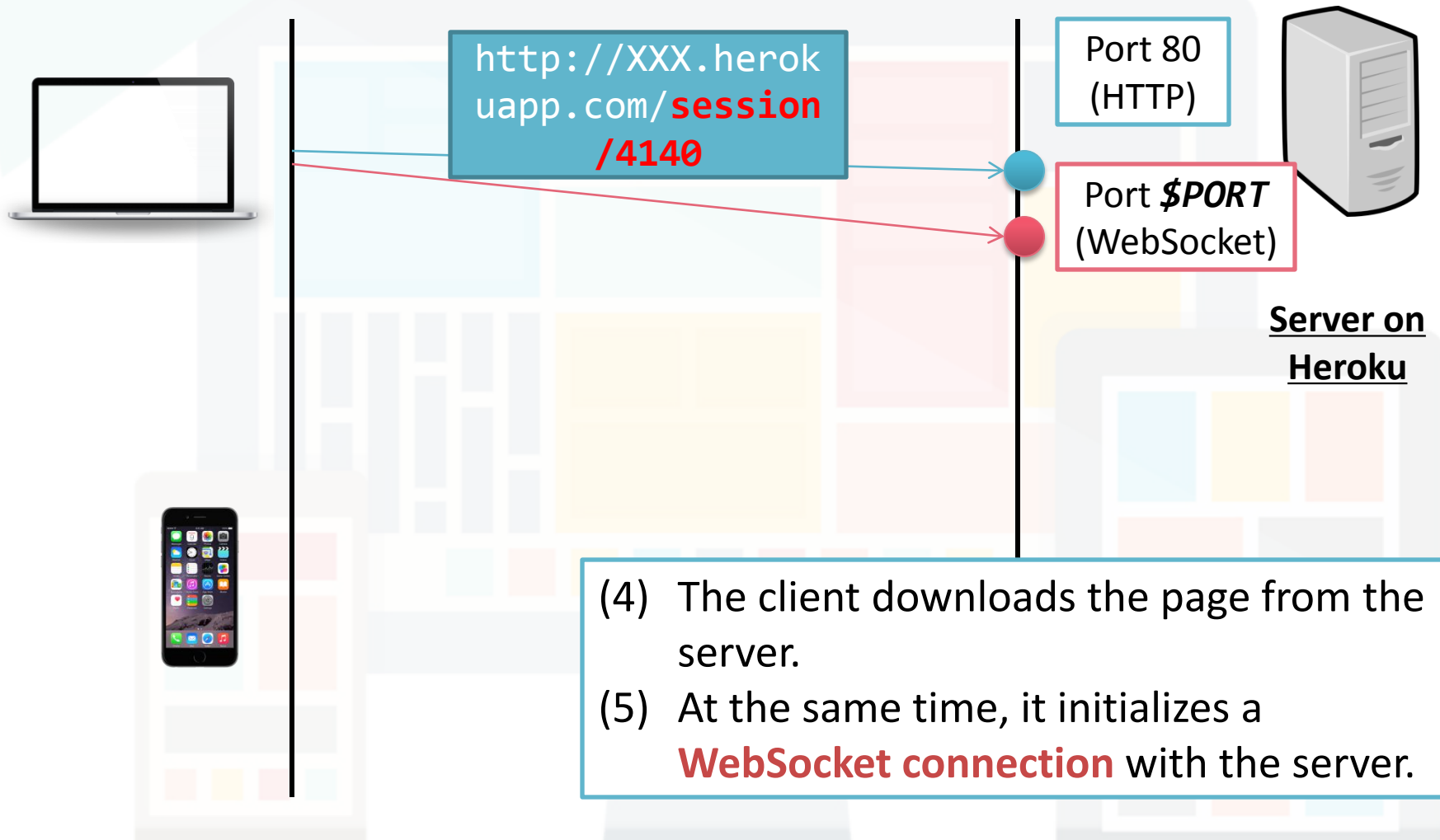
# Work flow: Initialization (Desktop)



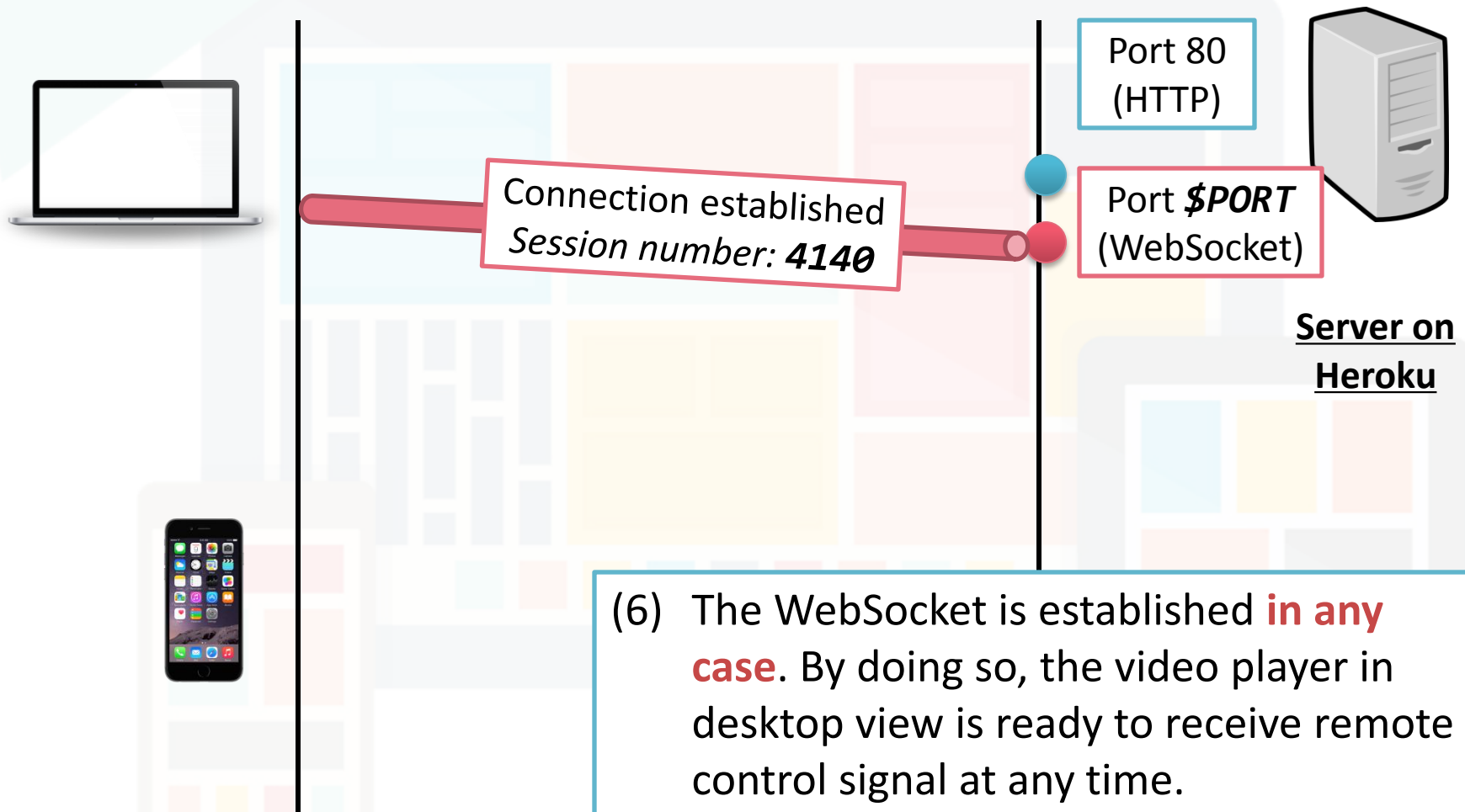
# Work flow: Initialization (Desktop)



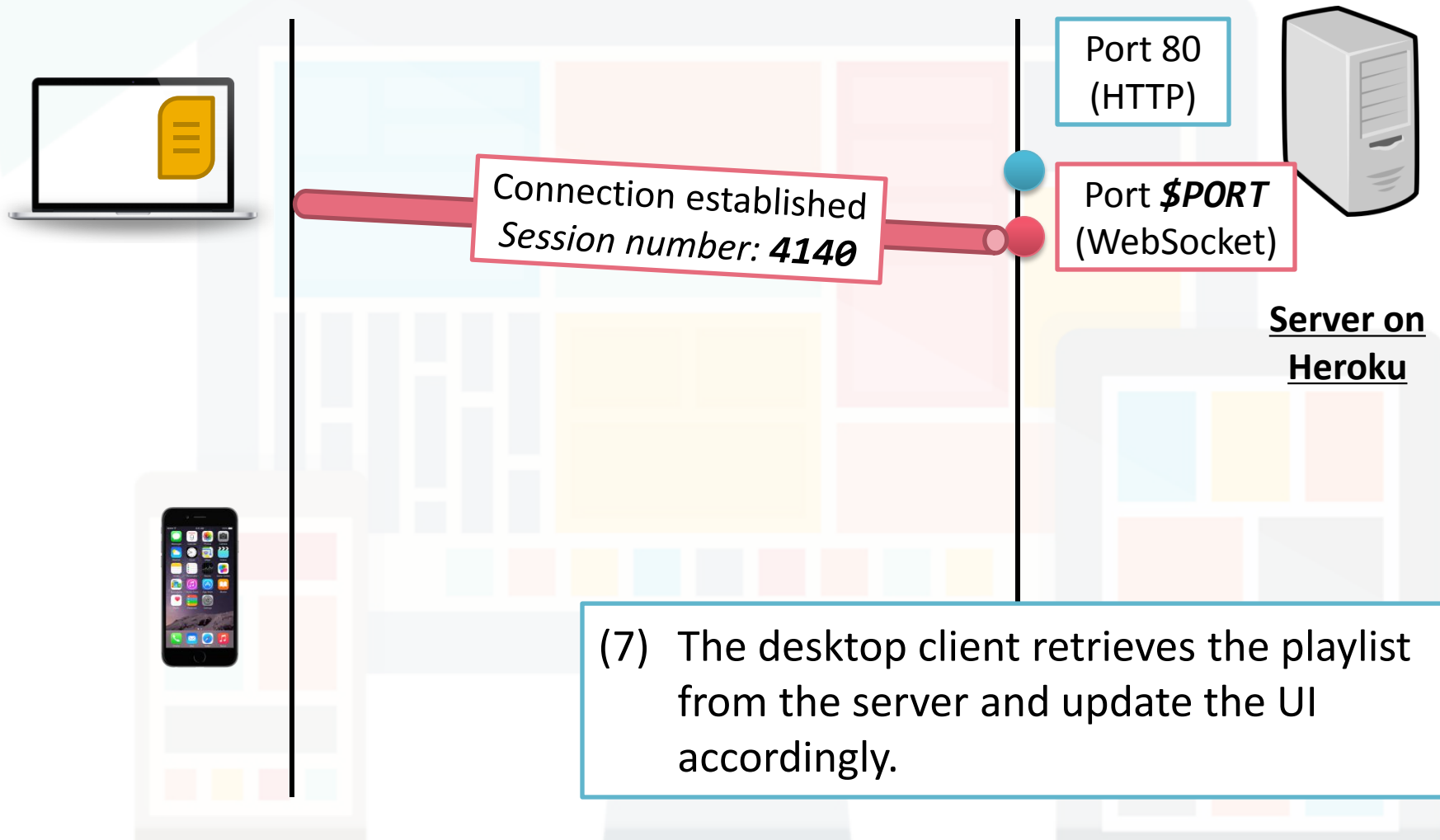
# Work flow: Initialization (Desktop)



# Work flow: Initialization (Desktop)

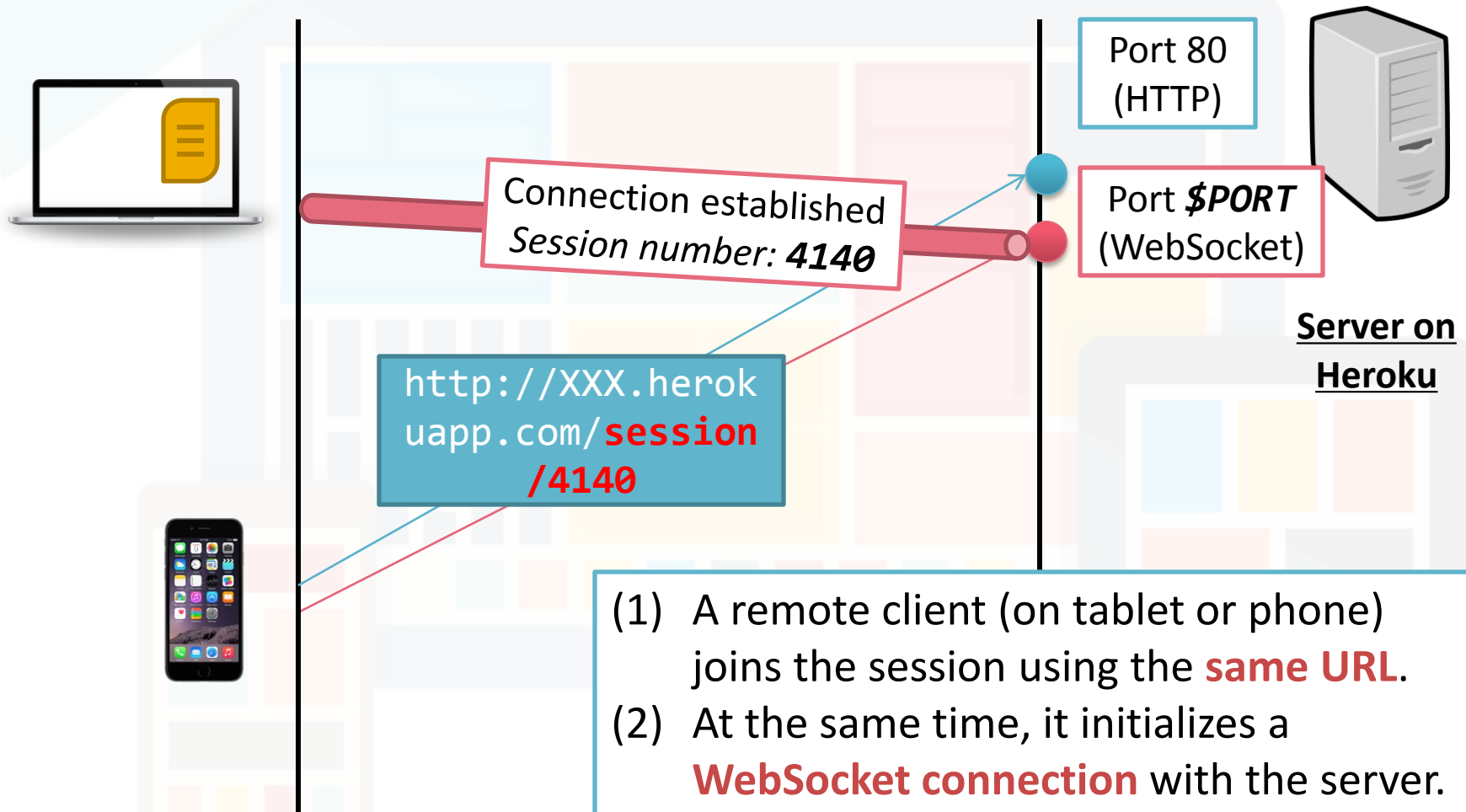


# Work flow: Initialization (Desktop)

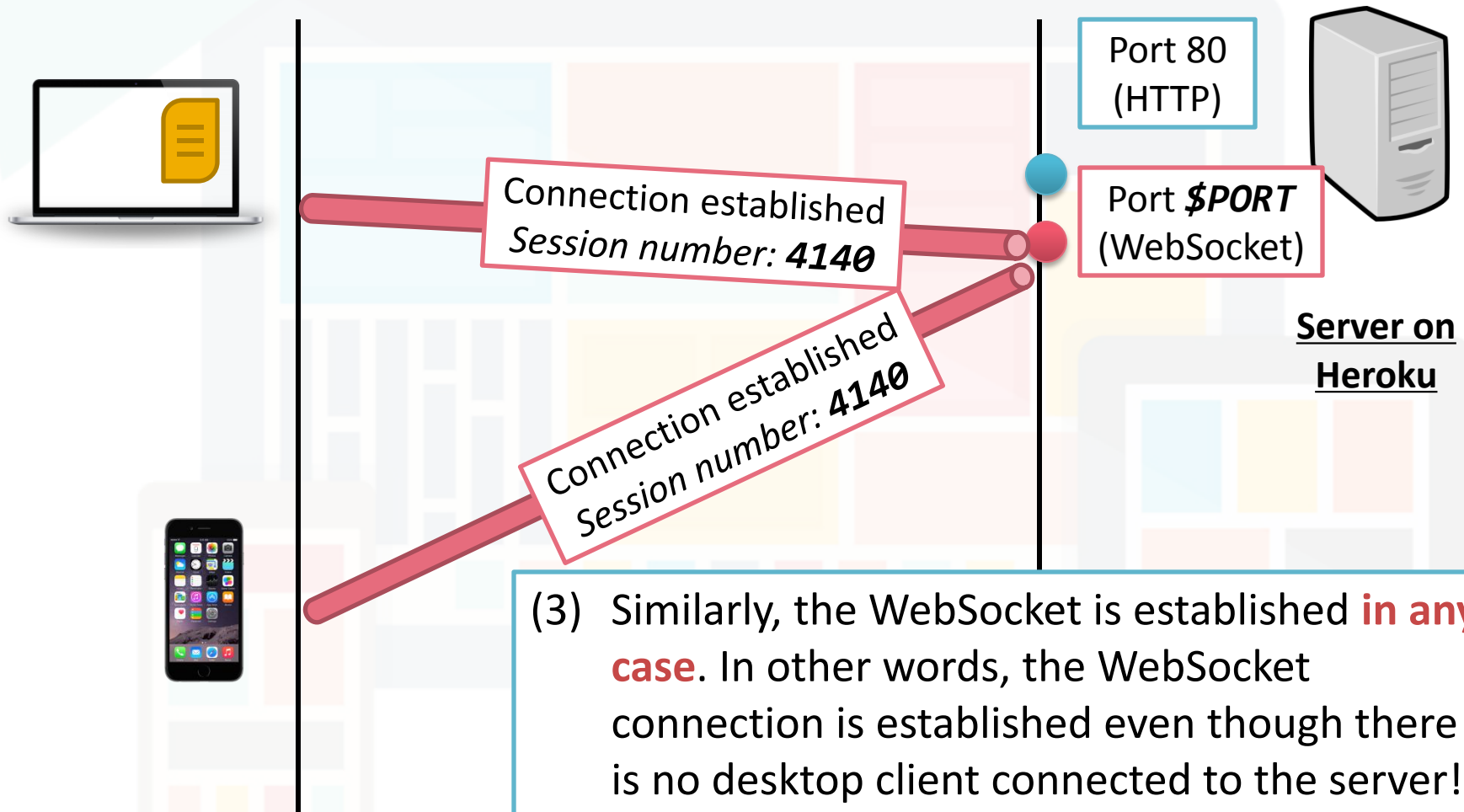




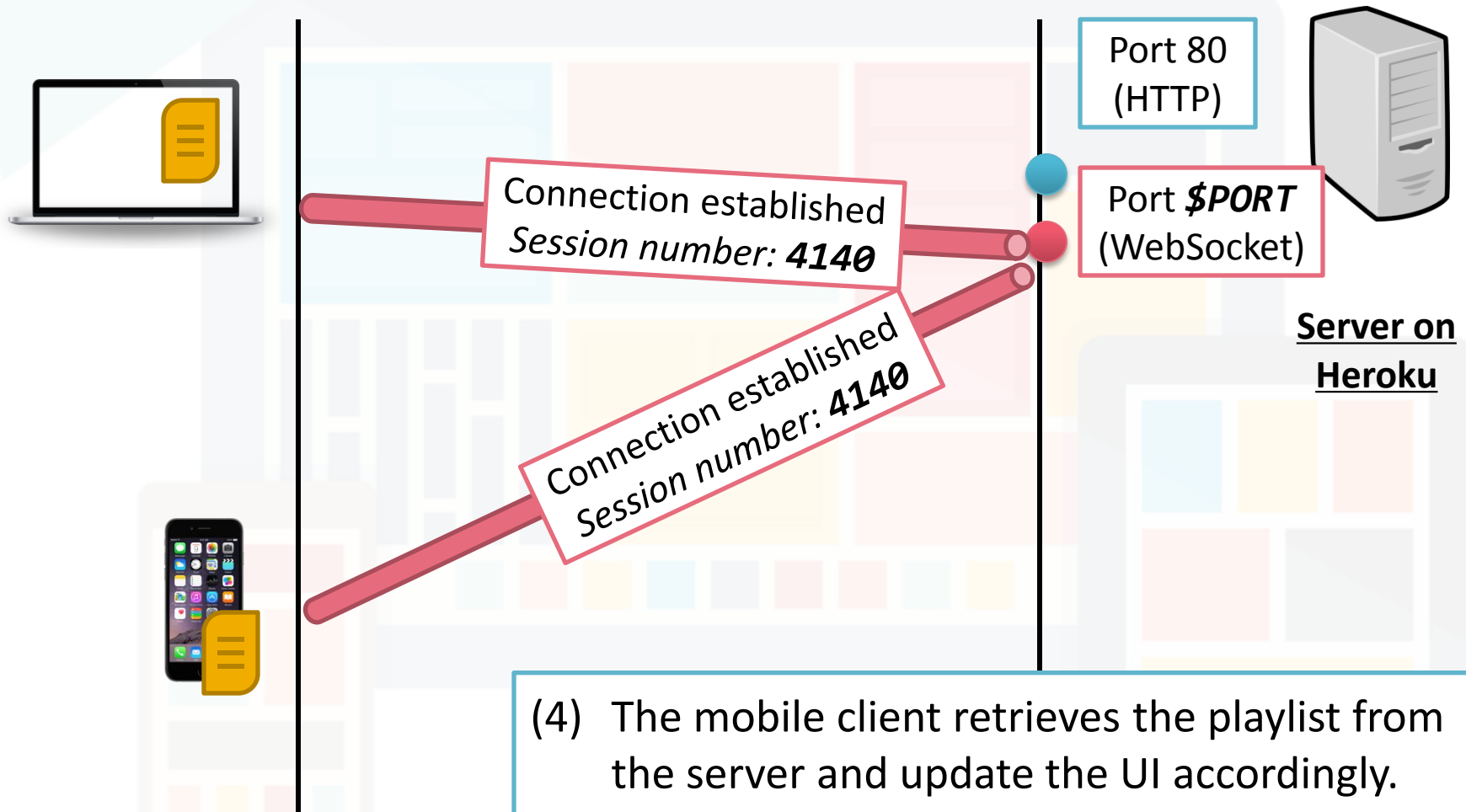
# Work flow: Initialization (Tablet or phone)



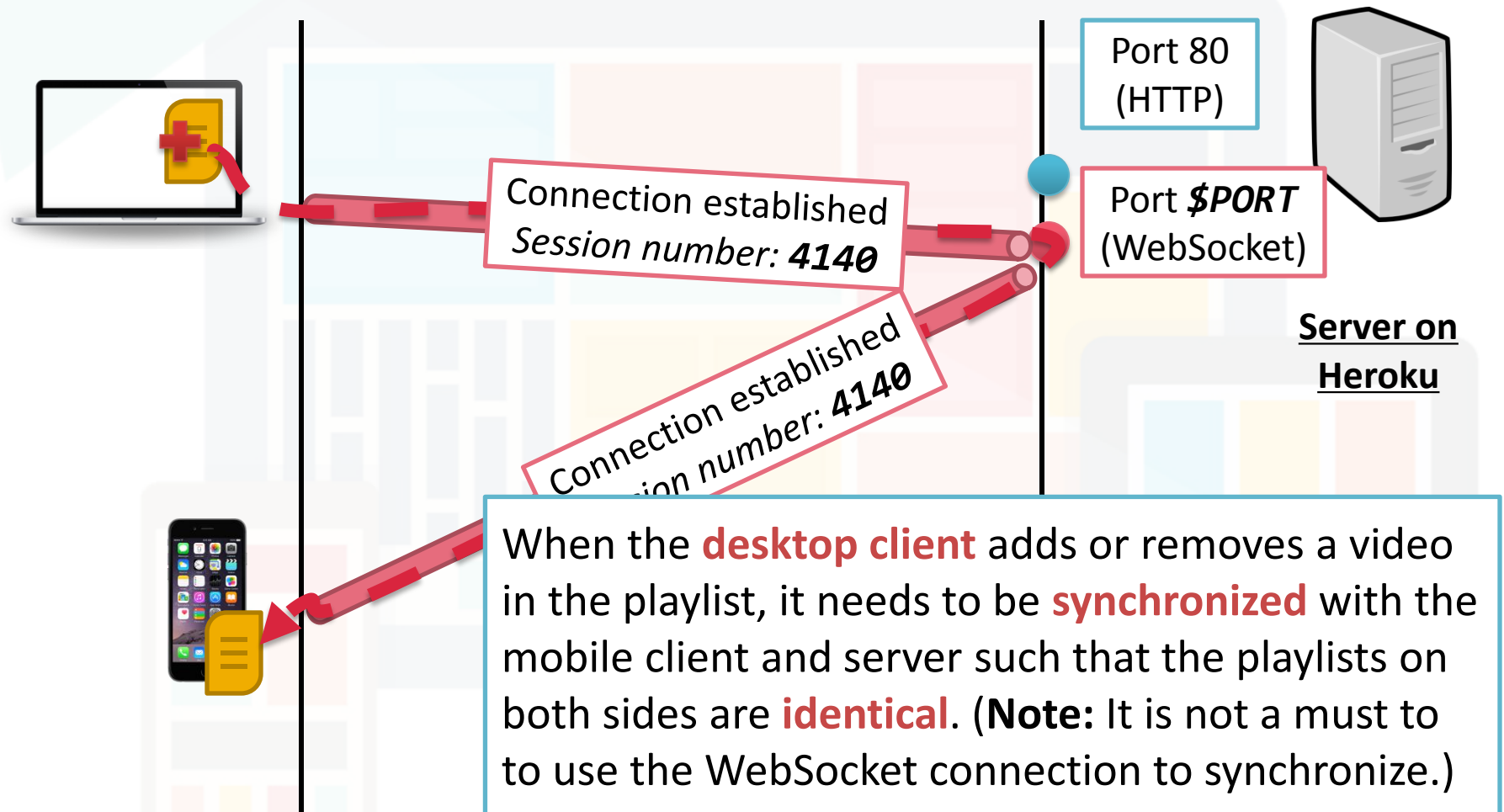
# Work flow: Initialization (Tablet or phone)



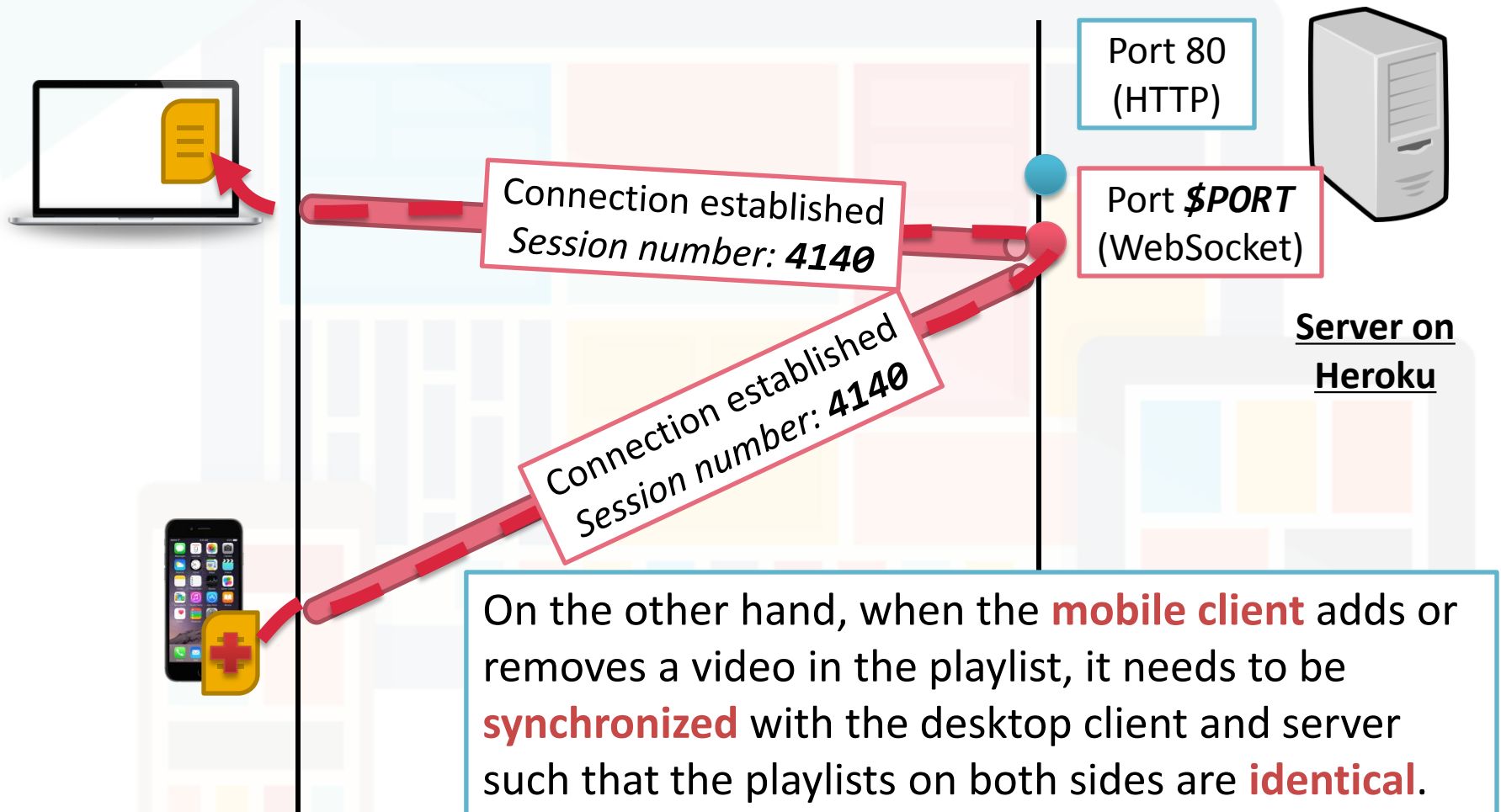
# Work flow: Initialization (Tablet or phone)



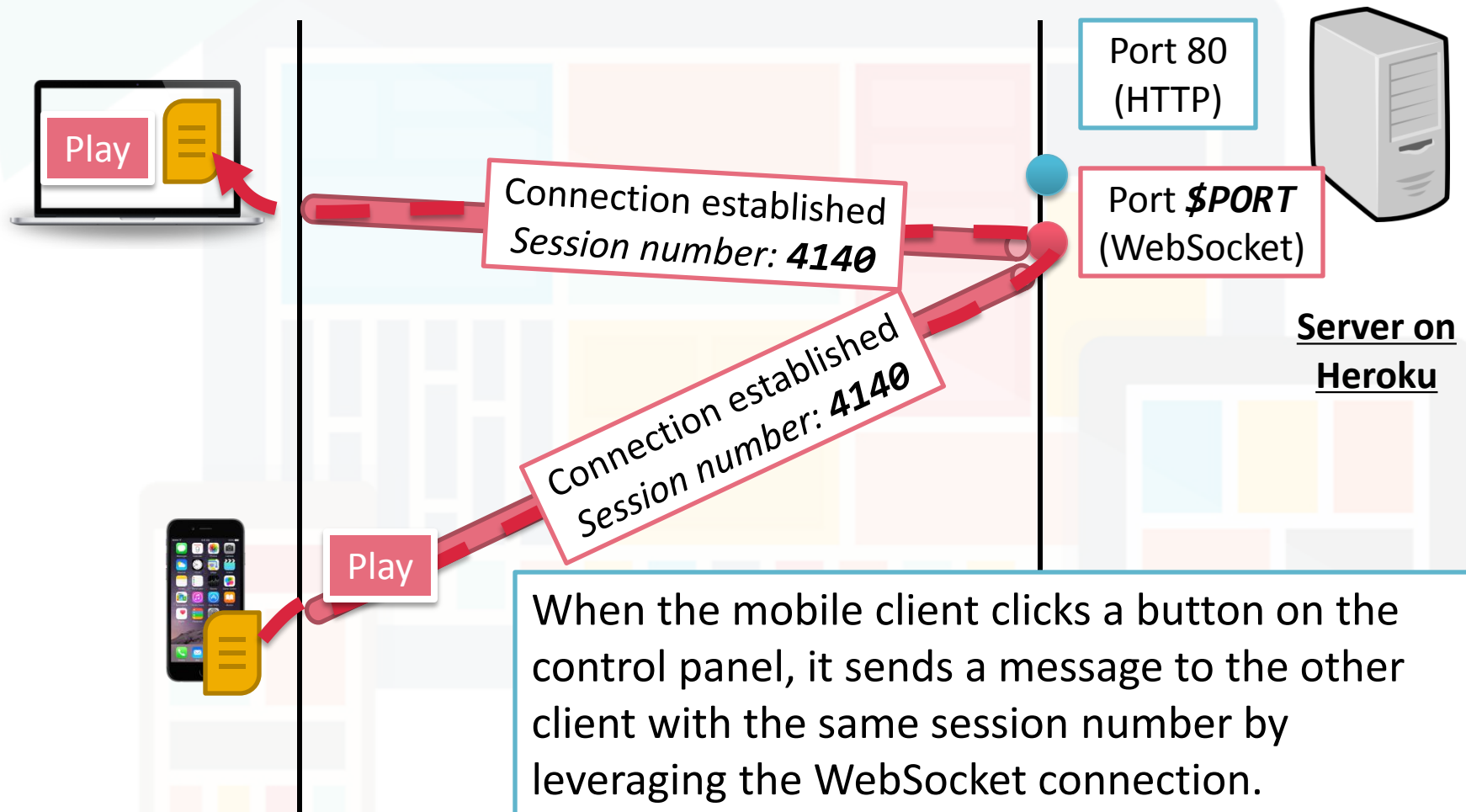
# Work flow: Playlist synchronization



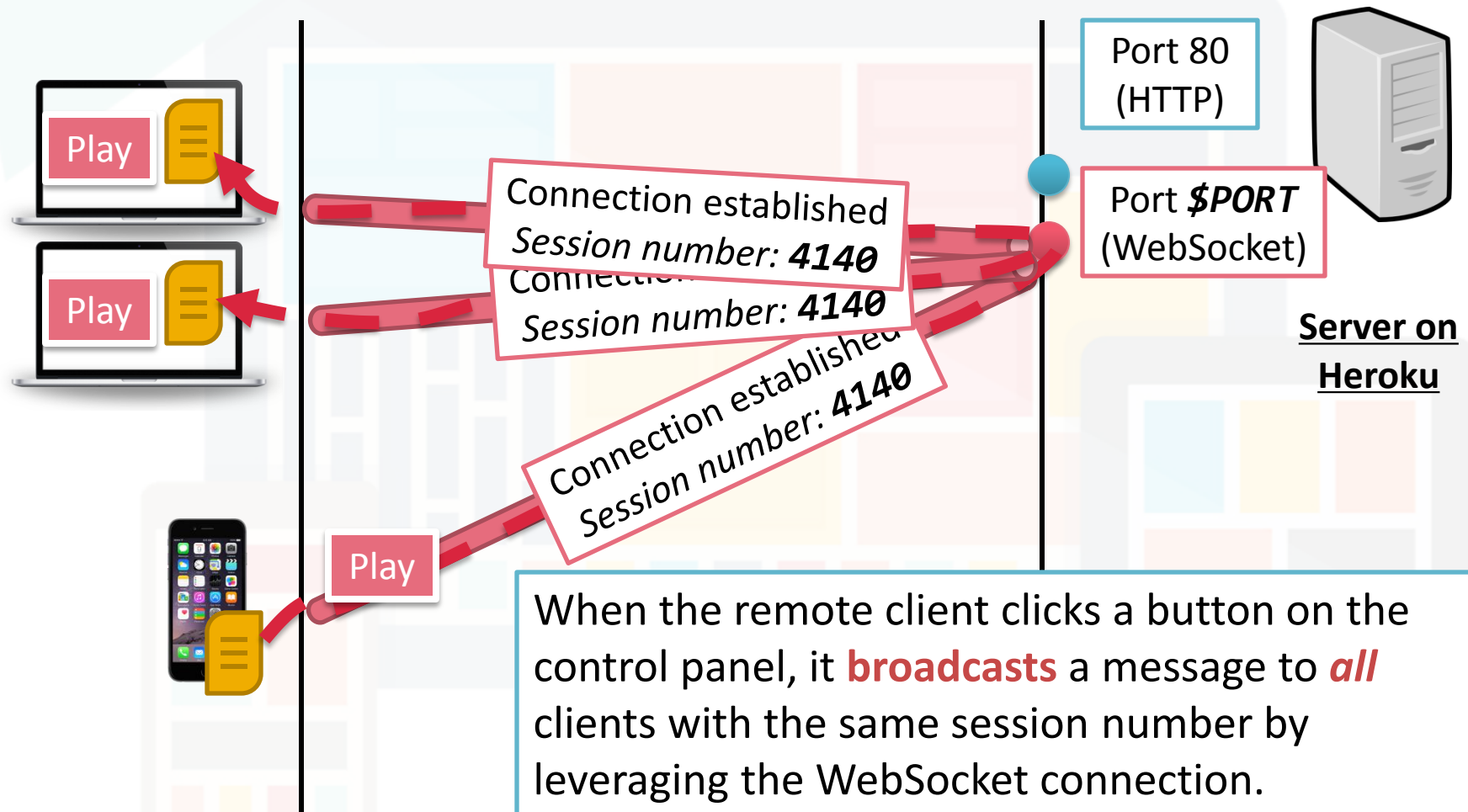
# Work flow: Playlist synchronization



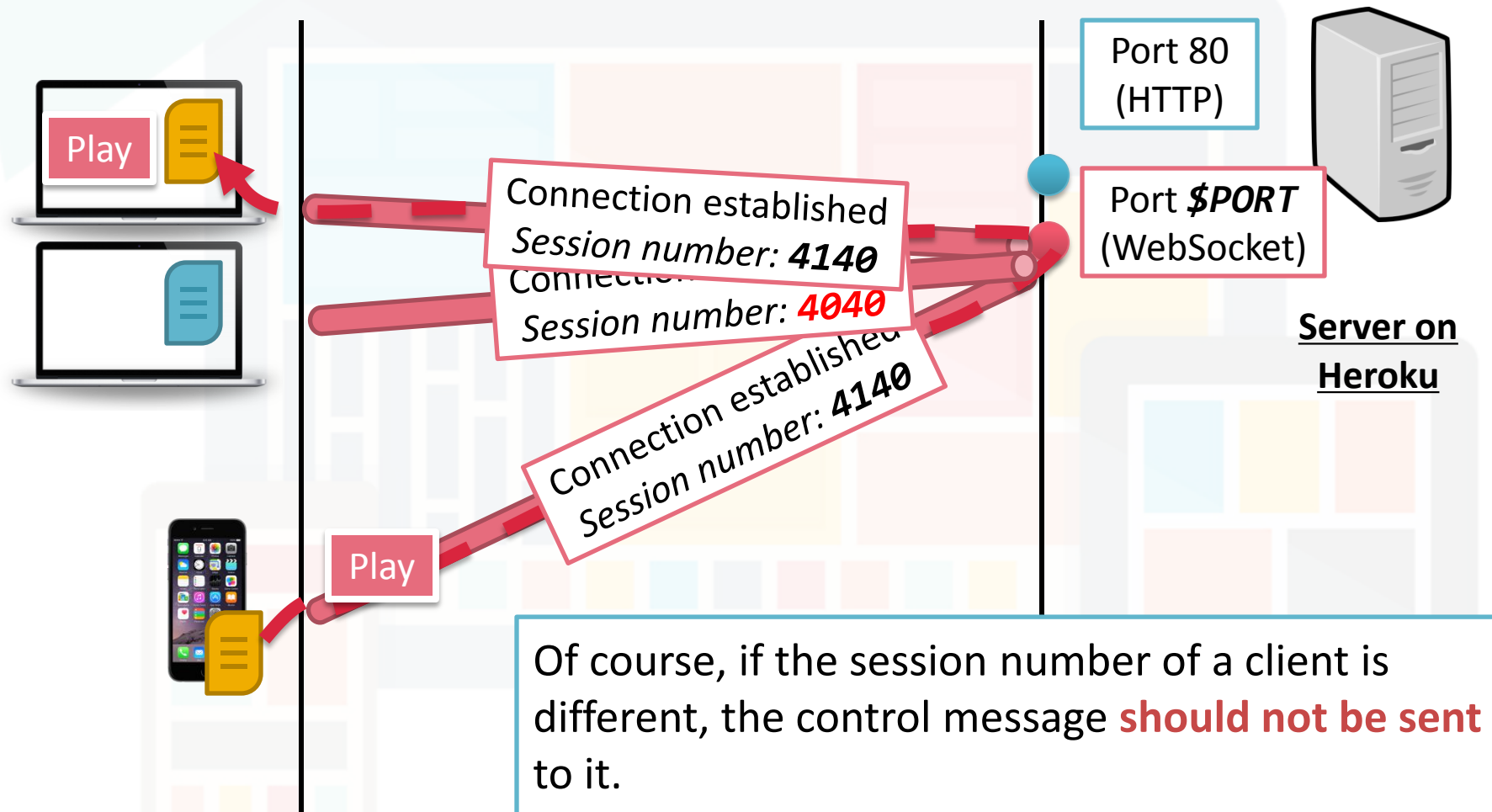
# Work flow: Remote control



# Work flow: Remote control

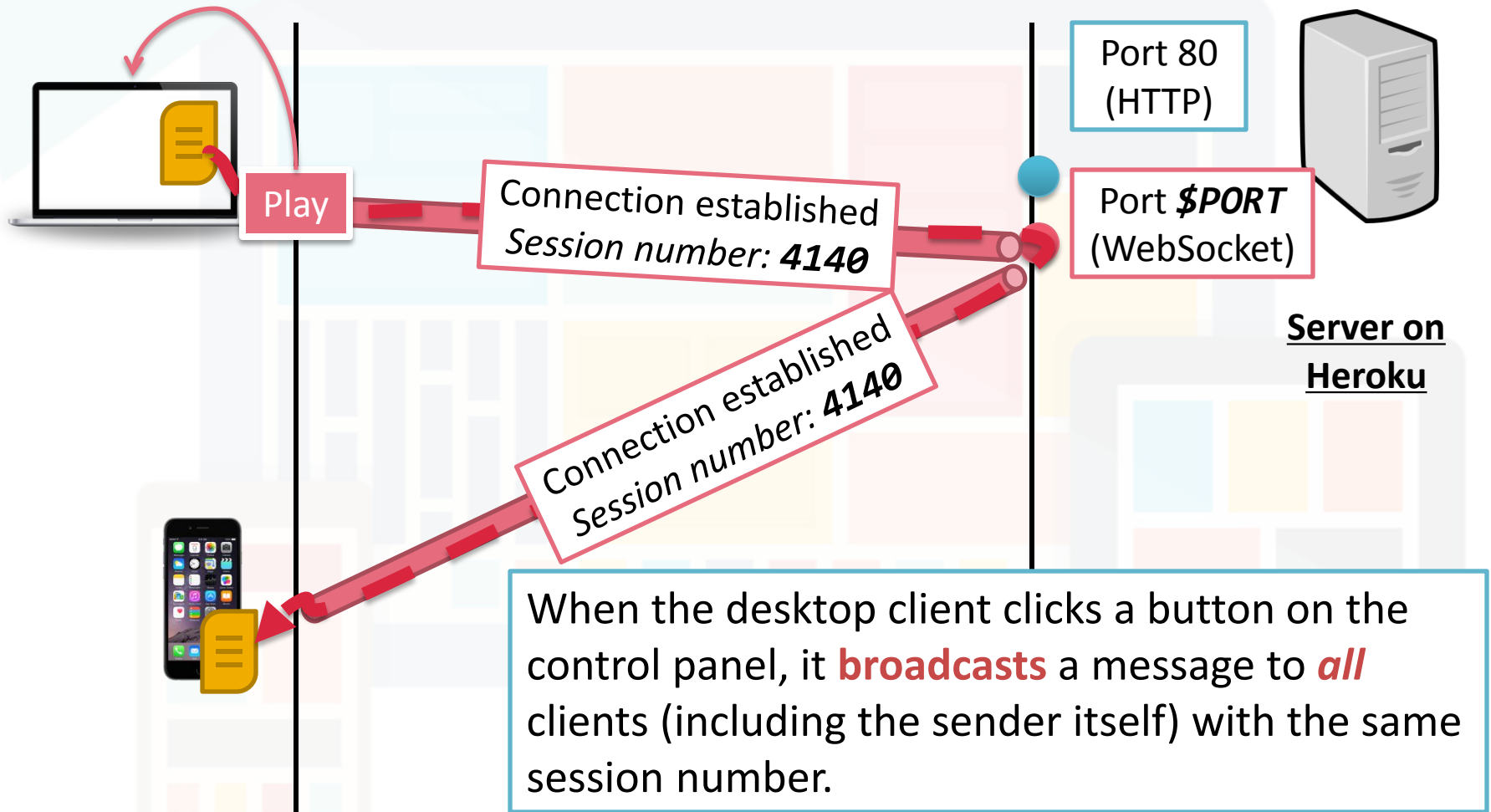


# Work flow: Remote control

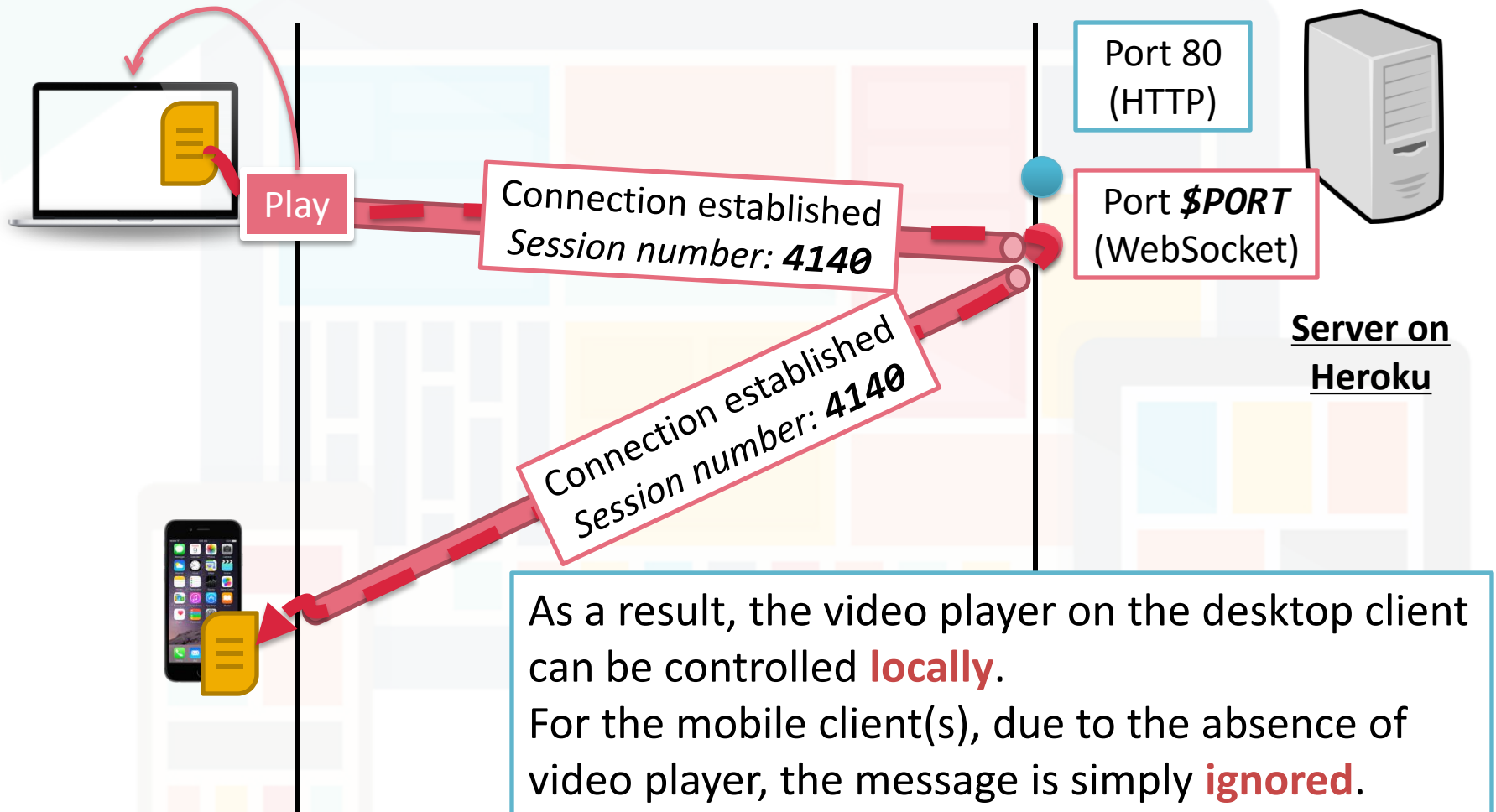




# Work flow: Remote control



# Work flow: Remote control





# Server side: Routing

Client enters the page without session number  
(e.g., `http://XXX.herokuapp.com/`)

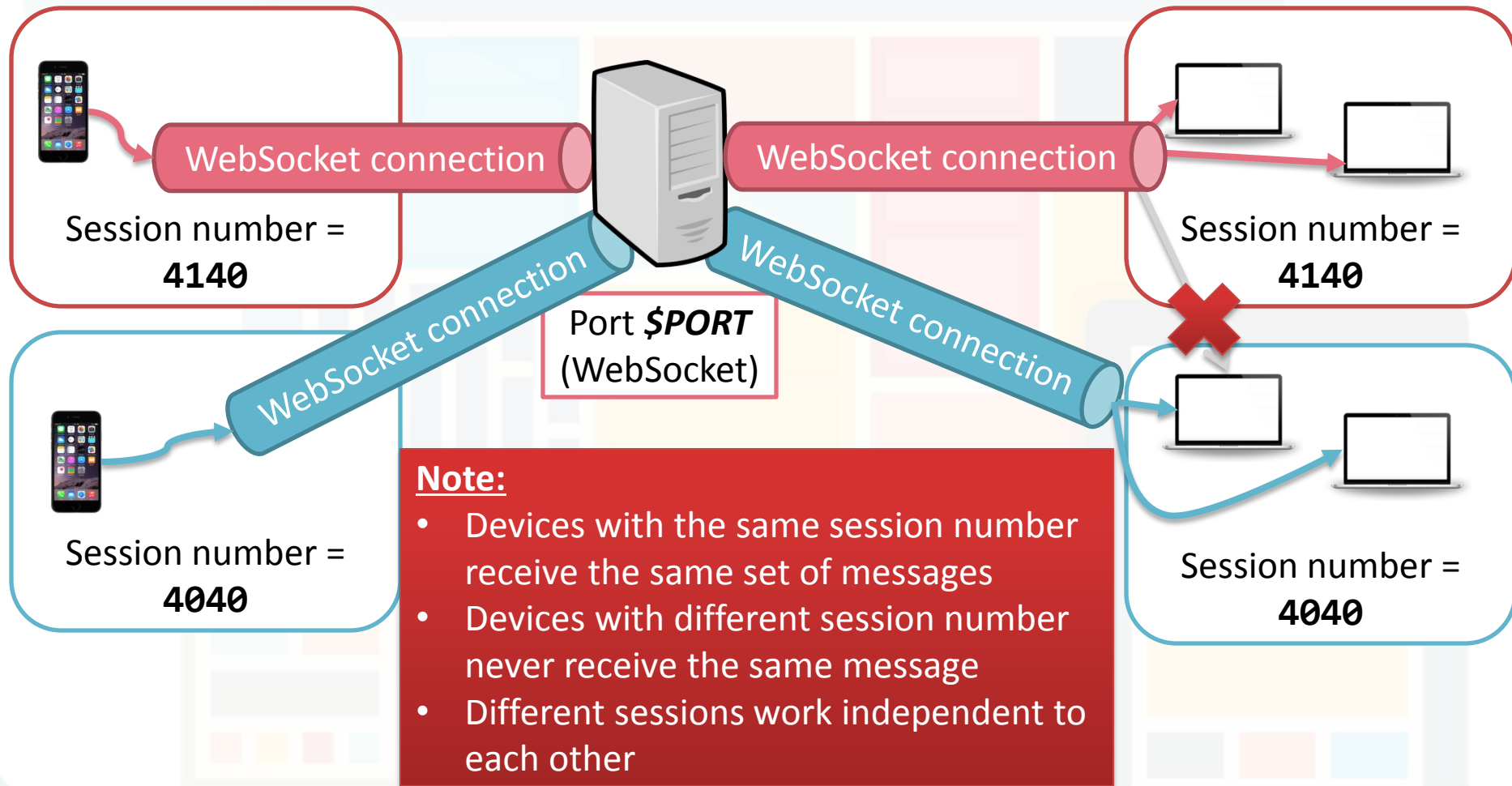


Server generates a unique session number



Server redirects the client to the page representing this session  
(e.g., `http://XXX.herokuapp.com/session/4140`)

# Server side: Message forwarding



# Server side: Implementation

- How to implement?
  - Routing: **Node.js** and **Express**
  - Message forwarding: **Socket.IO**
- How to store the playlists of each session?
  - **MongoDB** / **MySQL** database (provided in Heroku)
- How to retrieve the playlists of each session?
  - Leverage the **Socket.IO** connection; or
  - Use **RESTful APIs**!

# Reminder

- You should finish the **UI design, player** and **control logic**
- You should start implementing the **server side**
  - We will cover **Node.js** in this and next tutorial
  - We will cover **Socket.IO** in next tutorial
  - In other words, you can complete Assignment 2 **after next tutorial**
- Remember to deploy your Node.js application on **Heroku**

– End –