

**IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
WACO DIVISION**

**TOUCHSTREAM TECHNOLOGIES,
INC.**

Plaintiff,

v.

GOOGLE LLC

Defendant.

Civil Action No. 6:21-cv-569

JURY TRIAL DEMANDED

ORIGINAL COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff Touchstream Technologies, Inc., hereby files this Original Complaint for Patent Infringement against Google LLC and alleges, upon information and belief, as follows:

THE PARTIES

1. Plaintiff Touchstream Technologies, Inc., d/b/a Shodogg (“Touchstream” or “Plaintiff”) is a New York corporation with its principal place of business in New York, New York.

2. Defendant Google LLC (“Google” or “Defendant”) is a Delaware limited liability corporation with its principal place of business at 1600 Amphitheatre Parkway, Mountain View, CA 94043. Google maintains a permanent physical presence within the Western District of Texas, conducting business from its offices at 500 West 2nd Street, Austin, Texas, 78701. Google may be served with process through its registered agent CSC – Lawyers Incorporating Service Company, at 211 E. 7th Street, Suite 620, Austin, Texas 78701. Google is currently registered to do business in the State of Texas, and has been since at least November 17, 2006.

NATURE OF THE ACTION

3. This is a civil action against Google for patent infringement arising under the patent statutes of the United States, 35 U.S.C. § 271 *et seq.* for the infringement of United States Patent Nos. 8,356,251 (the “’251 patent”), 8,782,528 (the “’528 patent”), and 8,904,289 (the “’289 patent”) (collectively, “the Touchstream Patents”). True and correct copies of the ’251 patent, the ’528 patent, and the ’289 patent are attached as Exhibits 1, 2, and 3, respectively, to this Complaint.

JURISDICTION AND VENUE

4. This action arises under the patent laws of the United States, Title 35 of the United States Code. This Court has subject matter jurisdiction over the subject matter of this action under 28 U.S.C. §§ 1331 and 1338(a).

5. This Court has personal jurisdiction over Google in this action because Google has committed acts within the Western District of Texas giving rise to this action and has established minimum contacts with this forum such that the exercise of jurisdiction over Google would not offend traditional notions of fair play and substantial justice. Google has engaged in continuous, systematic, and substantial activities within this State, including substantial marketing and sales of products—including the Chromecast products¹ that are used by Google in connection with performing the accused Chromecast functionalities²—within this State and this District. Furthermore, Google—directly and/or through subsidiaries or intermediaries—has committed and continues to commit acts of infringement in this District by, among other things, using and providing the Chromecast service.

6. Venue is proper in this District pursuant to 28 U.S.C. §§ 1391(b) and (c) and/or 1400(b). As discussed above, Google currently has a regular and established place of business in

¹ The term “the Chromecast products” is defined at ¶ 48, *infra*.

² The term “accused Chromecast functionalities” is defined at ¶ 48, *infra*.

this District, and has committed and continues to commit acts of patent infringement in this District.

7. Google maintains a regular and established place of business within the Western District of Texas, conducting business from its offices at 500 West 2nd Street, Austin, Texas, 78701.

8. Public reporting indicates that Google has maintained an office in Austin since 2007.³

9. Additionally, reporting indicates that Google currently owns or leases over 500,000 square feet of office space in downtown Austin across at least three locations: 100 Congress Ave., 901 E. Fifth St., and 500 W. Second St. Google has also indicated that it is scheduled to open another 750,000 square feet of office space in Austin, at 601 W. Second St.⁴

10. Google currently employs over 1,100 persons in Austin and—as of June 2019—Austin was Google’s seventh-largest office outside of the San Francisco Bay Area.⁵ As of June 1, 2021, Google had job postings for 219 positions in Austin, Texas.⁶

11. Google directly and/or indirectly develops, designs, tests, distributes, markets, offers to sell, sells, and/or utilizes the Chromecast products that Google uses to perform the accused Chromecast functionalities in the Western District of Texas, and otherwise purposefully directs infringing activities to this District in connection with its Chromecast products.

³ See, e.g., <https://www.builtinaustin.com/2019/06/17/google-new-offices-austin-data-center-midlothian>.

⁴ See, e.g., <https://www.kvue.com/article/money/economy/boomtown-2040/google-austin-texas-real-estate-report/269-2ce6e60e-e8c3-46f5-aca6-864175e67950>.

⁵ See, e.g., <https://austonia.com/technology/google-austin-office>, citing <https://www.statesman.com/news/20190614/google-says-it-plans-significant-expansion-in-austin>.

⁶ Search conducted via <https://careers.google.com/locations/austin/?src=Online%2FHouse%20Ads%2FAdSitelinks>.

12. On information and belief, the Chromecast products are sold at a variety of retailers in Austin and/or Waco, Texas—including Walmart, Target, Best Buy, Sam’s Club, Lowes, and Office Depot.

13. According to Google, its network infrastructure consists of three elements: data centers, Edge Points of Presence (“PoPs”), and Edge Nodes (“Google Global Cache,” or “CGC”).⁷

14. According to Google, its data centers “are the heart of Google content and services.”⁸ Google purportedly owns and operates a \$600 million data center in Midlothian, Texas, which is located in Ellis County, within the Northern District of Texas.⁹ According to Google, such data centers “keep all of Google’s products and services up and running around the clock and around the world.”¹⁰

15. According to Google, it operates “a large, global meshed network that connects our Edge PoPs to our data centers.”¹¹ Google purports to operate two points of presence (“PoPs”) in Dallas and/or Fort Worth, Texas.

16. According to Google, it operates Edge Nodes that are used by network operators to “deploy Google supplied servers” within their networks, and some nodes are used “to support the delivery of . . . Google services.”¹² According to Google, it currently operates three “network

⁷ See, e.g., <https://peering.google.com/#/infrastructure>.

⁸ *Id.*

⁹ See, e.g.,

<https://www.google.com/about/datacenters/locations/midlothian/#:~:text=Google%20is%20proud%20to%20call,data%20center%20in%20Midlothian,%20Texas>.

<https://www.kvue.com/article/money/economy/boomtown-2040/google-austin-texas-real-estate-report/269-2ce6e60e-e8c3-46f5-aca6-864175e67950>.

¹⁰ See, e.g., <https://www.google.com/about/datacenters/faq/>.

¹¹ See, e.g., <https://peering.google.com/#/infrastructure>.

¹² *Id.*

edge locations” in Dallas, Texas.¹³ According to Google, at least one Edge Node (GGC) is located in or close to Austin, Texas.¹⁴

17. On information and belief, the team lead or “director” responsible for the development and execution of Google’s initial go-to-market strategy, global retail sales strategy, and hardware distribution for Chromecast, as well as the development of partnerships for Chromecast, currently resides in or around Austin, Texas, and is currently employed by Google.

18. On information and belief, at least one person involved in the discussion between Touchstream and Google regarding a potential partnership related to the technology invented by Touchstream, as described further below, currently resides in Texas.

19. On information and belief, at least one Chromecast “partner” who produces video content that can be streamed via a Chromecast enabled application is located in Austin, Texas.

20. On information and belief, at least one Chromecast “partner” live-streams sports matches from Austin (as well as other locations) for a sports team located in Austin, Texas, via a Chromecast enabled application.

TOUCHSTREAM’S PATENTS

21. In 2010, David Strober, the inventor of the Touchstream Patents and the original founder of Touchstream, was working at Westchester Community College as a Program Manager and e-learning instructional designer. At this job Mr. Strober facilitated the development of online college courses, developing software as needed to support those efforts.

22. At least as early as mid-2010, Mr. Strober perceived the need to be able to take videos that could be viewed on a smaller device, like a smartphone, and “move” them to a larger screen, like a computer monitor or television. In working to bring his idea to fruition, Mr. Strober

¹³ See, e.g., <https://cloud.google.com/vpc/docs/edge-locations>.

¹⁴ See, e.g., <https://peering.google.com/#/infrastructure>.

expanded his work by using a device like a smartphone to cause a video to play on a second screen, even if that video resided elsewhere (like the public internet). Near the end of 2010, Mr. Strober had developed a working prototype that demonstrated his groundbreaking concept. Recognizing that that his invention could revolutionize how people located, viewed, and shared media, Mr. Strober filed his first patent application in April 2011.

23. Each of the Touchstream Patents, which are each entitled “Play Control of Content on a Display Device,” claims priority to U.S. Provisional Patent Application No. 61/477,998 (filed on April 21, 2011).

24. On January 15, 2013, the U.S. Patent and Trademark Office duly and legally issued the ’251 patent to inventor David Strober.

25. Touchstream is the owner, by assignment, of all rights, title, and interest in the ’251 patent.

26. On July 15, 2014, the U.S. Patent and Trademark Office duly and legally issued the ’528 patent to inventor David Strober.

27. Touchstream is the owner, by assignment, of all rights, title, and interest in the ’528 patent.

28. On December 2, 2014, the U.S. Patent and Trademark Office duly and legally issued the ’289 patent to inventor David Strober.

29. Touchstream is the owner, by assignment, of all rights, title, and interest in the ’289 patent.

BACKGROUND OF THE DISPUTE

TOUCHSTREAM REVOLUTIONIZES VIDEO STREAMING

30. In 2011, inventor David Strober officially incorporated Touchstream to share his inventions with the world.

31. In the following years, Touchstream raised millions of dollars in investments.

32. Since 2011, Touchstream, d/b/a “Shodogg,” developed software that enables content to be wirelessly cast (e.g., accessed, displayed, and controlled) from a mobile device to a second display screen (e.g., TV, computer, tablet, etc.). Touchstream has been a leader in developing casting technology and has received numerous awards and recognition.

33. Unfortunately, the efforts of Touchstream and Touchstream’s partners to appropriately monetize Mr. David Strober’s inventions were significantly hindered by infringement of the Touchstream Patents, including by Google. The timing and scope of Google’s infringement is discussed in more detail below.

GOOGLE MEETS WITH TOUCHSTREAM IN 2011-2012 TO LEARN ABOUT THE PATENTED TOUCHSTREAM TECHNOLOGY

34. Since at least December 14, 2011, Touchstream has made clear that its revolutionary product offerings were “patent-pending.”¹⁵

35. Just days after the first Touchstream Patent issued on January 15, 2013, Touchstream issued a press release announcing this patent award.¹⁶

¹⁵ See e.g., Sean Ludwig, *Shodogg will let you pause and restart video from any device (exclusive)*, VentureBeat (Dec. 14, 2011 7:00 AM), <https://venturebeat.com/2011/12/14/shodogg-video-sharing-phones-tvs-exclusive/>; Shodogg, *Shodogg Launches at CES and Transforms Streaming Video Delivery by Fueling Industry Expansion with Content Providers*, Cision PR Newswire (Jan. 10, 2012, 9:43 ET), <https://www.prnewswire.com/news-releases/shodogg-launches-at-ces-and-transforms-streaming-video-delivery-by-fueling-industry-expansion-with-content-providers-137010098.html>; see also <https://web.archive.org/web/20111003131546/http://shodogg.com/> (archived snapshot of Shodogg website from October 3, 2011) (“Shodogg is a patent-pending technology that allows viewers to access online streaming content from any smartphone and display it to any larger connected screen, such as a laptop, tablet, or TV.”).

¹⁶ Shodogg, *Shodogg announces the release of ScreenDirect a business-to-business solution enabling companies to seamlessly direct digital content across screens*, Cision PR Newswire (Jan. 17, 2013 9:15 ET) <https://www.prnewswire.com/news-releases/shodogg-announces-the-release-of-screendirect-a-business-to-business-solution-enabling-companies-to-seamlessly-direct-digital-content-across-screens-187284641.html>; See also, e.g., *Meet Shodogg Who Won this Year’s Techweek NYC Launch Competition*, AlleyWatch (12/2014),

36. In December 2011, Touchstream and Google began discussing a potential partnership concerning the technology invented by Touchstream.

37. Touchstream signed Google's Non-Disclosure Agreement, which provided that neither company would get rights to the other company's intellectual property as a result of the meeting.

38. Google and Touchstream held a video conference by Skype on December 22, 2011.

39. At this meeting, the attendees for Touchstream informed the attendees for Google that Touchstream had filed for a patent on the Touchstream technology being discussed and demonstrated in the meeting.

40. In February of 2012, Google informed Touchstream that it did not want to pursue any business partnership with Touchstream in connection with the presented technology.

41. One or more of the Touchstream Patents (or their underlying applications) were cited to or by Google in connection with the following patents and patent applications filed by Google:

(i) U.S. Patent No. 9,841,939B2, filed December 18, 2014, titled "Methods, systems, and media for presenting requested content on public display devices" ('251 patent cited on face of the issued Google patent);

(ii) U.S. Patent No. 9,916,122B2, filed December 18, 2014, titled "Methods, systems, and media for launching a mobile application using a public display device" ('251 patent cited on face of the issued Google patent);

<https://www.alleywatch.com/2014/12/meet-shodogg-who-won-this-years-techweek-nyc-launch-competition/>.

(iii) U.S. Patent No. 9,967,320B2, filed December 18, 2014, titled “Methods, systems, and media for controlling information used to present content on a public display device” (the ‘251 patent is cited on face of the issued Google patent);

(iv) U.S. Patent No. 9,367,144B2, filed March 13, 2013, titled “Methods, systems, and media for providing a remote control interface for a media playback device” (Published Application No. 2012/0272148, which later issued as the ‘251 patent, is cited on face of the issued Google patent);

(v) U.S. Patent Application Pub. No. 2015/0046812 A1, filed August 11, 2014, titled “Dynamic resizable media item player” (Published Application No. 2012/0027214[8], which later issued as the ‘251 patent, was one of three references identified by Google on December 4, 2017) (this patent application was later abandoned by Google);

(vi) U.S. Patent No. 10,873,616B1, filed December 10, 2013, titled “Providing content to co-located devices with enhanced presentation characteristics” (Published Application No. 2012/0272147, which later issued as the ‘289 patent, and Published Application No. 2013/0124759, which later issued as the ‘528 patent, is cited on the face of the issued Google patent);

(vii) U.S. Patent No. 10,412,143B2, filed June 24, 2015, titled “Methods, systems, and media for presenting content based on user preferences of multiple users in the presence of a media presentation device” (Published Application No. 2012/0272147, which later issued as the ‘289 patent, is cited on the face of the issued Google patent);

(viii) U.S. Patent No. 10,306,323B2, filed December 7, 2016, titled “Fast television channel change initiated from a second screen device” (the ‘528 patent is cited on the face of the issued Google patent);

(ix) U.S. Patent No. 9,712,776B2, filed March 15, 2013, titled “Interfacing a television with a second device” (the ‘251 patent is cited on the face of the issued Google patent);

(x) U.S. Patent No. 9,992,307B2, filed February 3, 2015, titled “Interoperability of discovery and connection protocols between client devices and first screen devices” (the ‘251 patent and Published Application No. 2012/0272148, which later issued as the ‘251 patent, are both cited on face of the issued Google patent);

(xi) U.S. Patent No. 10,659,518B2, filed March 18, 2019, titled “Contextual Remote Control” (Published Application No. 2012/0272148, which later issued as the ‘251 patent, is cited on face of the issued Google patent);

(xii) U.S. Patent No. 10,969,950B2, filed May 19, 2015, titled “Dynamic Resizable Media Item Player” (Published Application No. 2012/0272148, which later issued as the ‘251 patent, is cited on face of the issued Google patent)

(xiii) International Application Publication No. WO 2015/200531, filed June 24, 2015, titled “Methods, Systems and Media for Presenting Content Based on User Preferences of Multiple Users in the Presence of a Media Presentation Device” (Published Application No. 2012/0272147, which later issued as the ‘289 patent, was cited to Google in a September 25, 2015 International Search Report);

(xiv) U.S. Patent Application Pub. No. 2016/0149982 A1, filed May 19, 2015, titled “Dynamic resizable media item player” (Published Application No. 2012/00272148, which later issued as the ‘251 patent, was identified by Google on July 17, 2018) (this patent application was later abandoned by Google).

42. As such, Google knew about the patent applications leading to the Touchstream Patents by no later than December 2011.

43. Google also knew, or at the very least should have known, of the issued Touchstream Patents on or shortly after the date each such patent was issued, beginning with the ‘251 Patent that issued in January 2013.

44. At no point in 2011, 2012, or 2013 did Google reach out to Touchstream about potentially acquiring a license to Touchstream’s pending or awarded patents, and to this day Google has not requested or received a license to any of the Touchstream Patents.

GOOGLE UNVEILS ITS INFRINGING CHROMECAST SERVICES

45. Google unveiled its Chromecast product—which performs the infringing Chromecast functionalities—in July 2013 at a price of \$35 per unit.¹⁷

46. As of October 4, 2017, Google announced it had sold 55 million Chromecast units.¹⁸

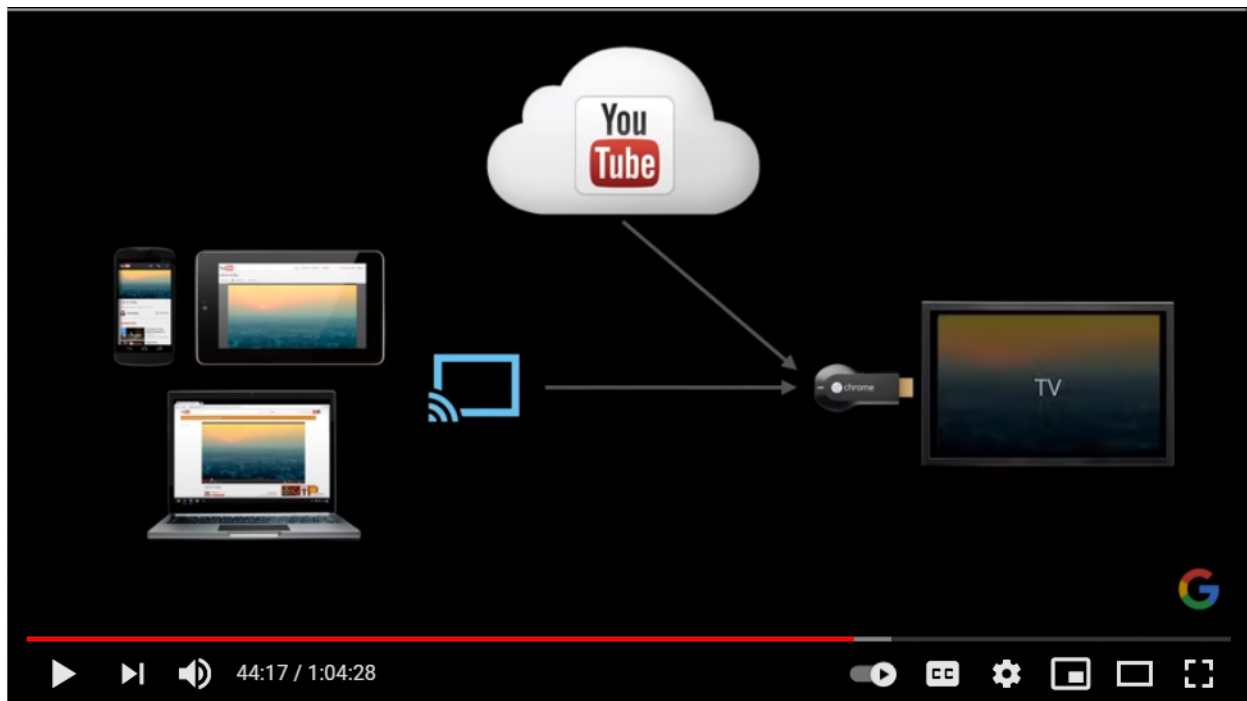
¹⁷ Matt Burns, *Google Launches The \$35 Chromecast Streaming Device To Bring Chrome To The Living Room*, Tech Crunch (July 24, 2013, 11:51 AM CDT) https://techcrunch.com/2013/07/24/google-chromecast/?_ga=2.60674024.478449141.1620755451-2045144417.1620755450.

¹⁸ Emil Protalinski, *Google has sold 55 million Chromecasts, up from 30 million in July 2016*, VentureBeat (Oct. 4, 2017) <https://venturebeat.com/2017/10/04/google-has-sold-55-million-chromecasts-up-from-30-million-in-july-2016/>.

47. Google’s financial benefit from its Chromecast product is not, however, limited to revenue from the sale Chromecast units. For instance, the accused Chromecast functionalities also increase ad revenue to Google and support the collection of data that is valuable to Google.

THE ACCUSED CHROMECAST FUNCTIONALITIES

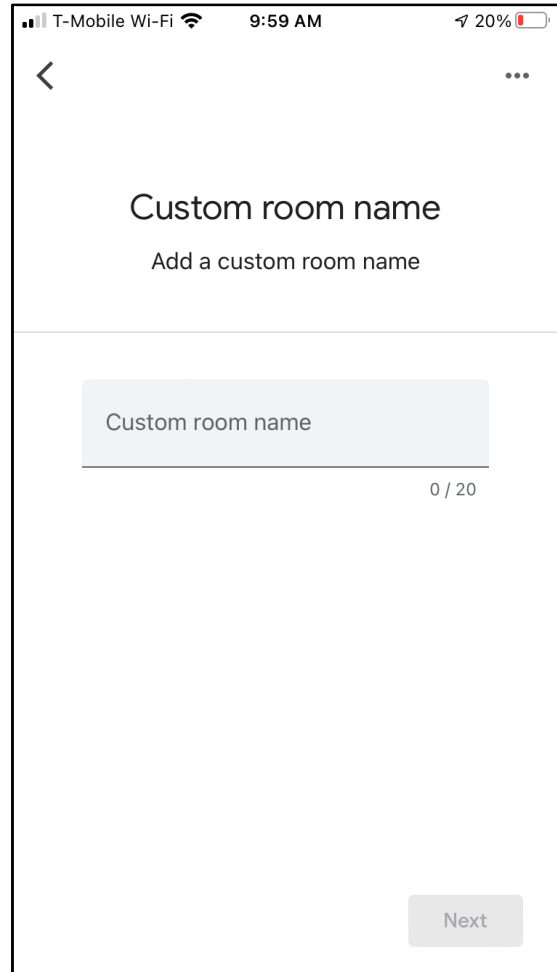
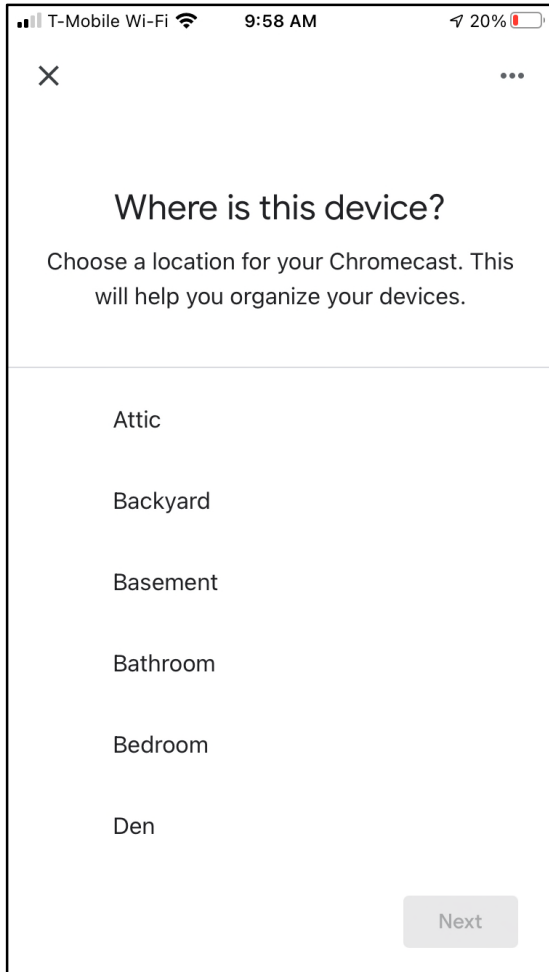
48. The accused Chromecast functionalities comprise the methods performed through operation of at least the standalone Chromecast devices (e.g., the Chromecast 1st Generation, Chromecast 2nd Generation, Chromecast 3rd Generation, Chromecast Ultra, and Chromecast with Google TV), as well as devices implementing Chromecast built-in (collectively, “Chromecast” or “the Chromecast products”) described in ¶¶ 49-62, *infra*. The Chromecast products did provide in the past, and continue to provide, functionality and structure that facilitates the controlling of presentation content, such as audio and/or video content, on a content presentation device that loads any one of a plurality of different media players (YouTube, e.g.), for instance as illustrated below.



Source: <https://www.youtube.com/watch?v=GbXeZl6FoCY> (2013 Google presentation on Chromecast functionality).

49. Upon initial operation, the Chromecast is connected to a device having a screen (e.g., a television). The Chromecast may prompt the user, via the screen, to download and open the Google Home application (or “app”) on the user’s mobile device (e.g., a smartphone or tablet). Upon opening the Google Home application on the mobile device, the user is shown the option to “Set up Chromecast.” Upon choosing this option, the Google Home application searches for devices and locates the Chromecast. The application then prompts the user to set up the Chromecast device. The mobile device and the Chromecast may be linked via a display of matching alphanumeric codes on the mobile device and display screen. The Google Home application running on the mobile device prompts for confirmation that the alphanumeric codes match. Upon receiving confirmation, the Google Home app and Chromecast are linked.

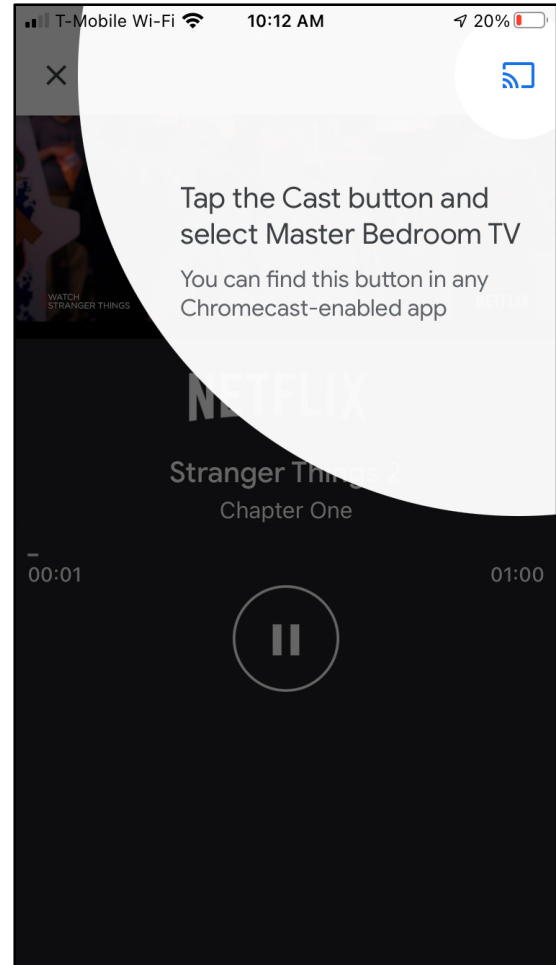
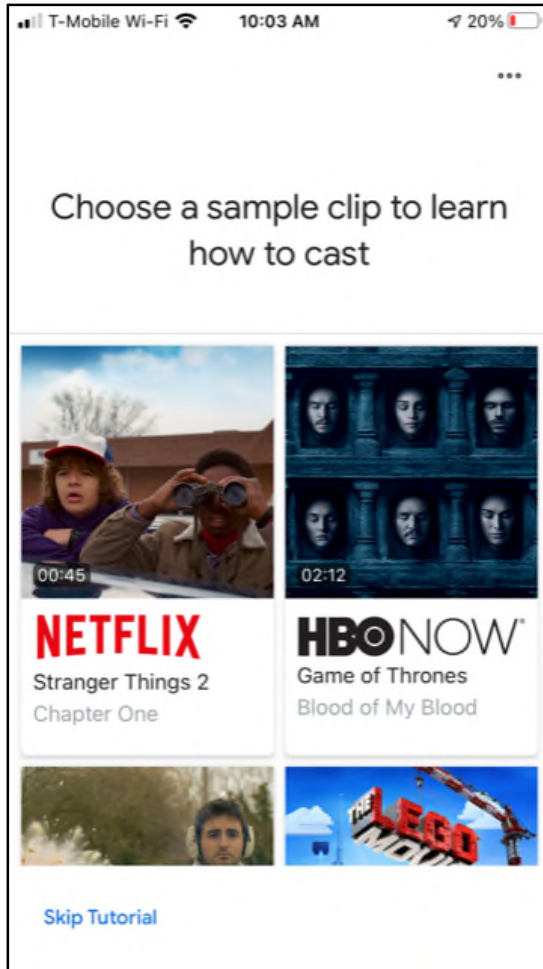
50. The Chromecast prompts the user to specify the Chromecast’s location (e.g., bedroom or living room) via the Google Home app. Upon receiving location information, the Chromecast further prompts for a “custom room name” (or “friendly name”) (e.g., “Master Bedroom” or “Living Room”), which the user provides. An example of this is provided below.



Exemplary assignment of custom room name on Chromecast 3rd Generation.

That friendly name is then assigned to the Chromecast device by Google for future reference. Next, the Chromecast prompts for internet connection information (such as the user's preferred Wi-Fi network) and connects to the internet.

51. Once connected to the internet, the Chromecast may present a tutorial screen from which the user can, for instance, choose a sample video clip and learn how to cast it for playback on the display screen via the Chromecast. When the user touches a video clip on the mobile device, a prompt appears on the mobile device screen instructing the user to tap the "Cast" button and select the Chromecast to be used for casting (e.g., "Master Bedroom"). An example of this is provided below.



Exemplary tutorial screens on Chromecast 3rd Generation

Upon touching cast, a casting session is established and the video clip begins playback on the display screen via the Chromecast.

52. The Chromecast is enabled to cast content from a mobile device to a display screen for thousands of both Android and iPhone applications, including video streaming services such as YouTube, Google Play, Netflix, Hulu, Prime Video, Disney Plus, as well as audio streaming services such as Spotify and Pandora, as indicated e.g. on the cover of the box in which the Chromecast 3rd Generation is packaged and sold:



By pressing the “Cast” button in any Chromecast-compatible app running on a user’s mobile device, the user can stream content from the app to a selected Chromecast device and view the content on a television or other display screen connected to the Chromecast.

53. Following initial setup, Google periodically pushes software and firmware updates for the Chromecast products, which are downloaded and installed automatically by the Chromecast when it is powered on and connected to the internet.¹⁹ Google also requires Chromecast users and developers to abide by Google’s terms of service.²⁰

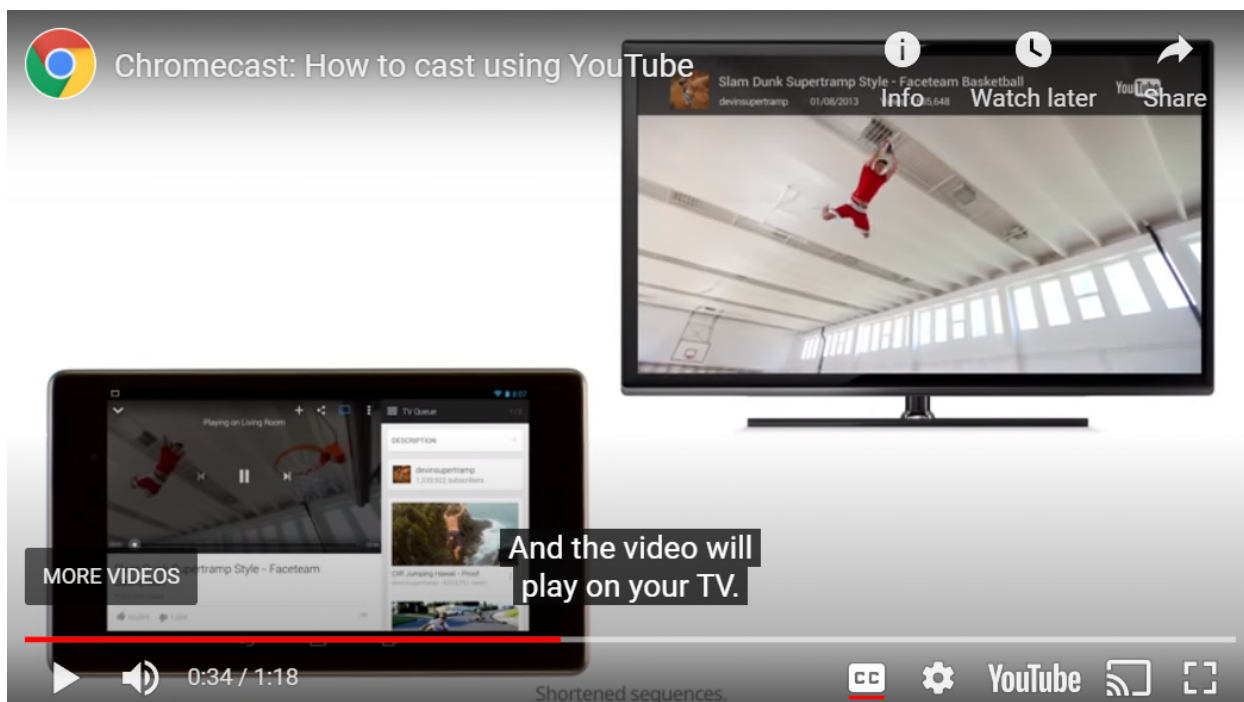
54. At the outset of a casting session, the Chromecast-compatible app running on the mobile device (e.g., YouTube or Netflix) causes messages to be sent from the mobile device and

¹⁹ See <https://support.google.com/chromecast/answer/6292664?hl=en>.

²⁰ See <https://support.google.com/cast-developer/answer/4513288?hl=en>; <https://developers.google.com/cast/docs/terms>; <https://developers.google.com/terms/>; <https://policies.google.com/terms?hl=en-US>;

received by a server system implemented by Google. These messages relate generally to authentication, selection of the device to cast to, and selection of the casting content.

55. The messages sent from the mobile device and received by the server system include a “networkAddress,” such as an IP address of the Chromecast device, which is assigned to and uniquely identifies the equipment. The messages also include the Chromecast’s user-assignable “friendlyName,” such as “Living Room” or “Master Bedroom.” The “networkAddress” and “friendlyName” are locally unique to the particular Chromecast within the user’s local access network. See example, below. On information and belief, the server system stores a record based on these messages associating the mobile device with the Chromecast, for instance so that an app running on the mobile device continually recognizes which Chromecast device (e.g., “Living Room”) is currently casting and can be controlled.



Source: <https://support.google.com/chromecast/answer/2995235?hl=en>.

56. The messages sent from the mobile device and received by the server system include messaging that specifies a media file (e.g., audio or video) to be acted upon. The media

file to be acted upon may be specified by providing, for instance, the uniform resource locator (URL) of the media file, for instance as specified by the `contentURL` property of the `GCKMediaInformationBuilder` class in the iOS Sender API,²¹ which class is further used to build `GCKMediaInformation` instances that describe a media item to load.²²

57. The messages sent from the mobile device and received by the server system further include messaging that identifies an application for playing content from the specified media file. The appropriate application may be identified by providing, for instance, information about the MIME type of the content to be played or by providing an app ID (“`applicationID`”) corresponding to a particular receiver application to be used to play back the content from the media file. Google uses the app ID to locate the appropriate receiver application, for example via a “resolved URL associated with the app ID.”²³ The receiver application may then be “downloaded from the network” and loaded onto the Chromecast via the resolved URL.²⁴ The receiver application “provides an interface to display the app’s content on the TV” or other display device, and “handles messages from the sender application to control content on the receiver device.”²⁵ The messages sent from the mobile device and received by the server system also specify the accompanying media codecs needed for playback of different types of content (audio, video, etc.) such as mp3, HE-AAC, h.264, or VP9, as described at <https://developers.google.com/cast/docs/media>. On information and belief, loading the receiver app includes loading one or more corresponding

²¹ *See*

https://developers.google.com/cast/docs/reference/ios/interface_g_c_k_media_information_builder.

²² https://developers.google.com/cast/docs/reference/ios/interface_g_c_k_remote_media_client

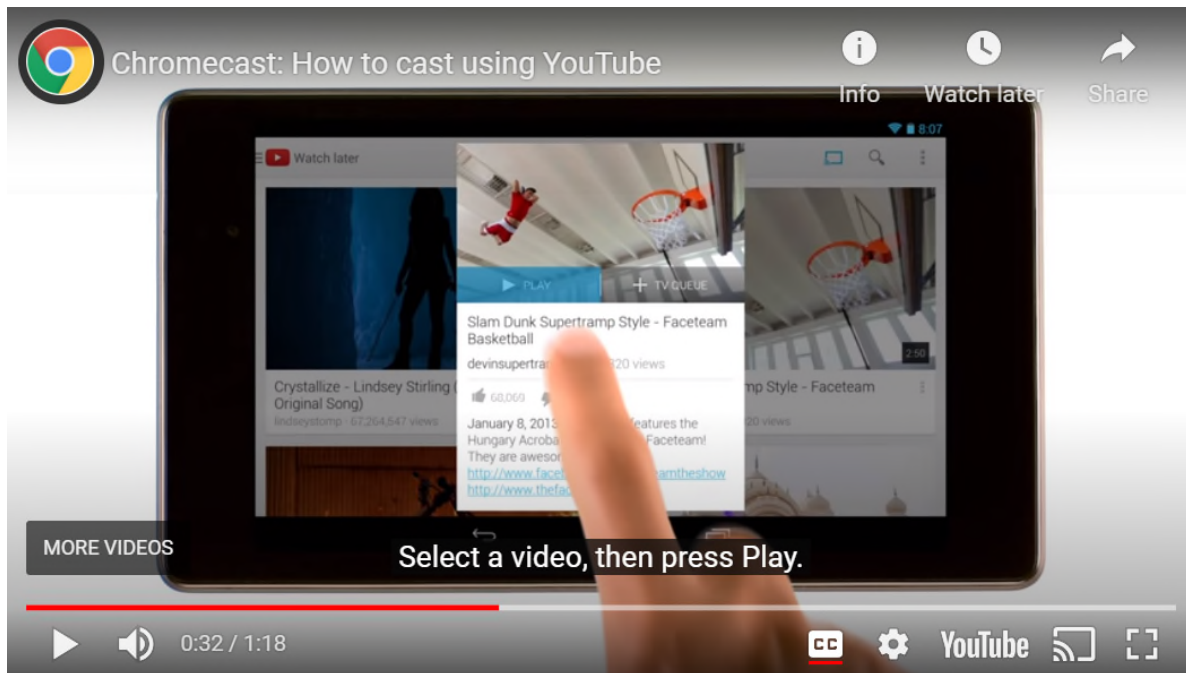
²³ *See* https://developers.google.com/cast/docs/caf_receiver/basic.

²⁴ *Id.*

²⁵ *See* https://developers.google.com/cast/docs/web_receiver.

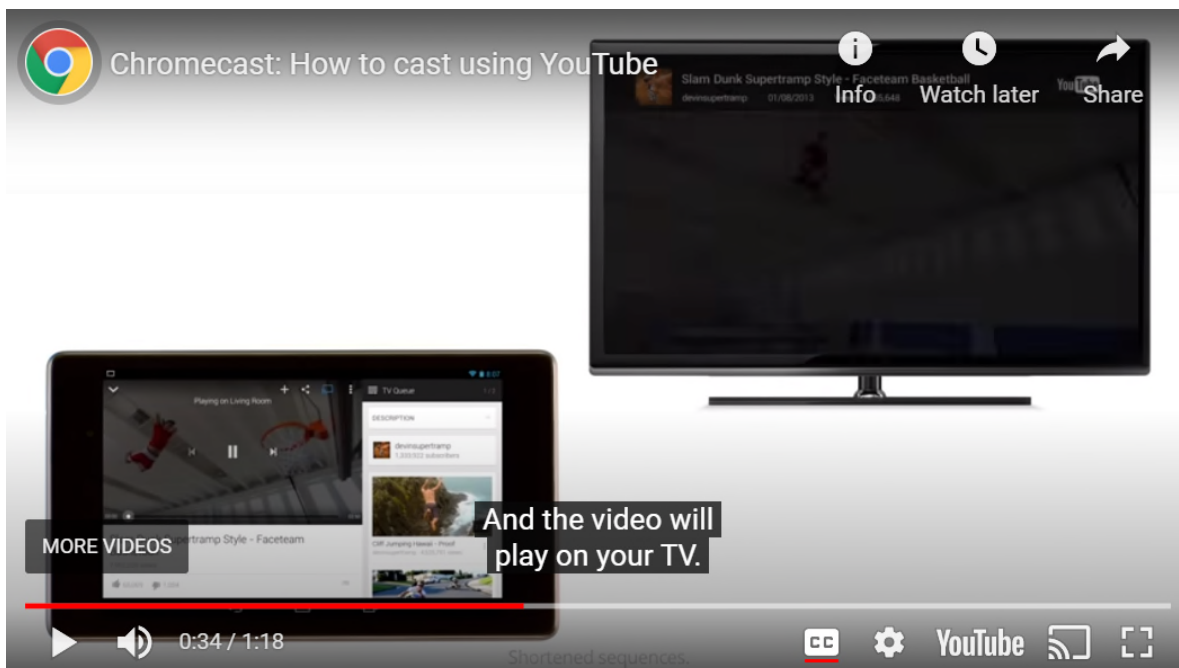
application programming interfaces (APIs) such as the “Web Receiver API” accessed by a reference in the app.²⁶

58. The Chromecast-compatible app running on the mobile device allows the user to touch “Play,” “Resume,” or equivalent on the mobile device screen so as to cast the media to the Chromecast and control playback. An example is shown below.

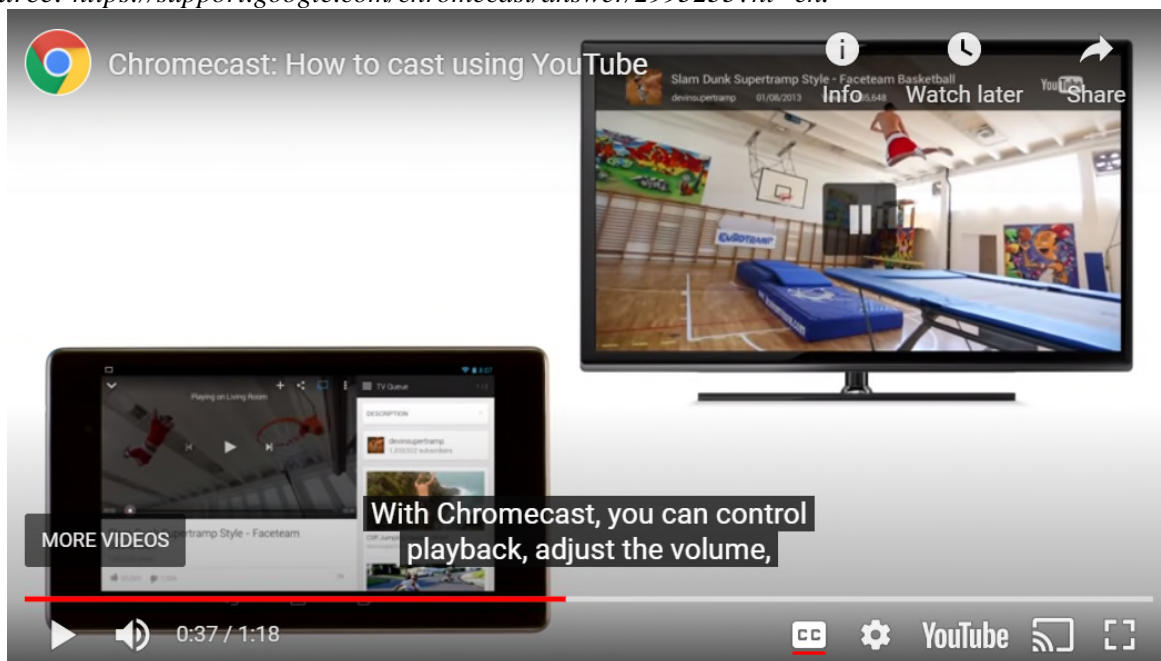


Source: <https://support.google.com/chromecast/answer/2995235?hl=en>.

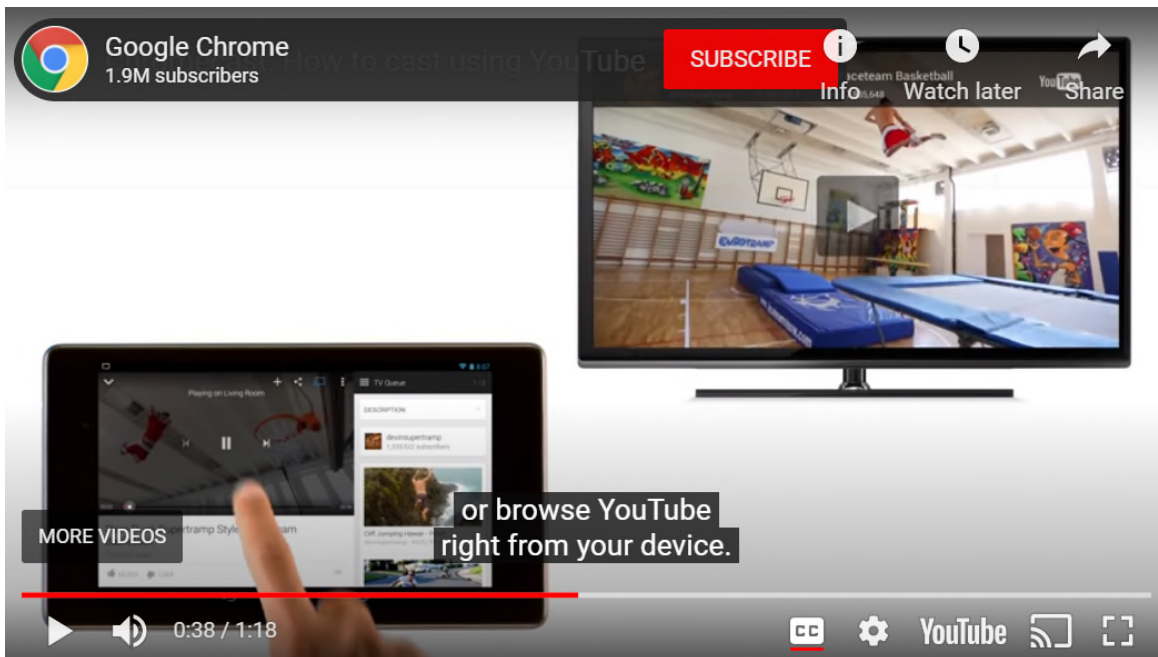
²⁶ See https://developers.google.com/cast/docs/web_receiver/basic (“Your Web Receiver app accesses the Web Receiver API with the following reference: `<script src="//www.gstatic.com/cast/sdk/libs/caf_receiver/v3/cast_receiver_framework.js"></script>`”); see also *id.* at discussion of “major classes” associated with the Web Receiver SDK framework, including `cast.framework.CastReceiverContext` which “manages overall framework and loads any necessary libraries.”)



Source: <https://support.google.com/chromecast/answer/2995235?hl=en>.



Source: <https://support.google.com/chromecast/answer/2995235?hl=en>.



Source: <https://support.google.com/chromecast/answer/2995235?hl=en>.

The class GCKUIMediaController is “a controller for UI views that are used to control or display the status of media playback on a Cast receiver” such as the receiver application running on the Chromecast.²⁷ The controller “responds to touch events on the controls [e.g., on the mobile device] by issuing the appropriate media commands to the receiver.”²⁸ Among the controls associated with the controller are “a ‘play’ button” (playButton) and “a ‘pause’ button” (pauseButton).²⁹ Further, “Google Cast sender applications control the playback on the receiver device by sending messages in JSON format to the receiver application.”³⁰ Among the JSON messages are “commands from the sender that change the player state,” including “Play” (“Begins playback of the content that was loaded with the load call, playback is continued from the current time

²⁷ See

https://developers.google.com/cast/docs/reference/ios/interface_g_c_k_u_i_media_controller.

²⁸ *Id.*

²⁹ *Id.*

³⁰ See <https://developers.google.com/cast/docs/reference/messages>,

position), “Pause” (“Pauses playback of the current content”), “Seek” (“Sets the current position in the stream”), and “Stop” (“Stops playback of the current content”).³¹

59. According to the Google Cast SDK Additional Developer Terms of Service, “Cast media application[s] must use these messages as defined here”—e.g., “Play,” “Pause,” “Seek,” and “Stop”—to control media playback on the receiver application running on the Chromecast.³² “Doing so provides the media app with a consistent user experience across platforms and it ensures that a Cast application will support new and future use cases.”³³

60. On information and belief, the server system receives text-based JSON commands and converts them into code that can be executed by the Chromecast devices. The Chromecast then executes the retrieved programming code so as to control playback of the specified media file—that is, to play, pause, or stop, e.g., playback of the audio or video streaming on the Chromecast in response to user commands sent from the Chromecast-compatible app running on the mobile device.

61. Further, during a cast session, on information and belief, Google stores information relating to the source of the media file, the media file itself, the particular receiver app, and playback control commands for statistical purposes, so that playback can be resumed in the event of a crash, and for purposes of allowing another user to take over and control the content streaming on the Chromecast from a separate mobile device. The Cast receiver API development framework includes the class `PlayerData`, whose variables include “applicationData,” “media” (“current media information”), “metadata” (“media metadata”), “state” (“current player state”), and

³¹ *Id.*

³² *Id.*

³³ *Id.*

“supportedMediaCommands” (“the commands supported by this player”).³⁴ The Cast receiver API development framework also includes a state object, `cast.framework.messages.SessionState`, which “contain[s] all data to be stored in `StoreSession` and to be recovered in `ResumeSession`.”³⁵ On information and belief, a receiver application running on the Chromecast retrieves the information associated with these classes from a database to restore content when playback resumes.

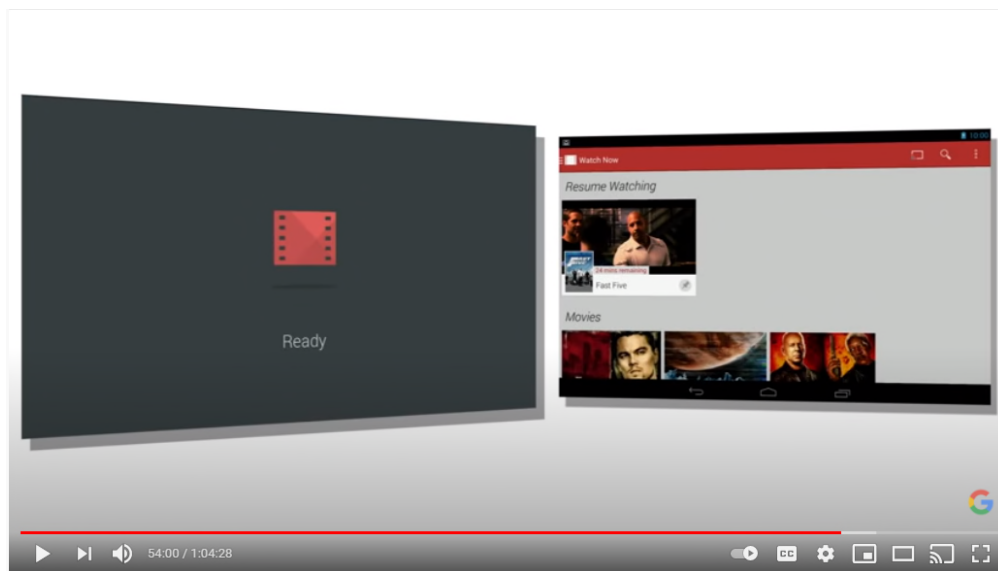
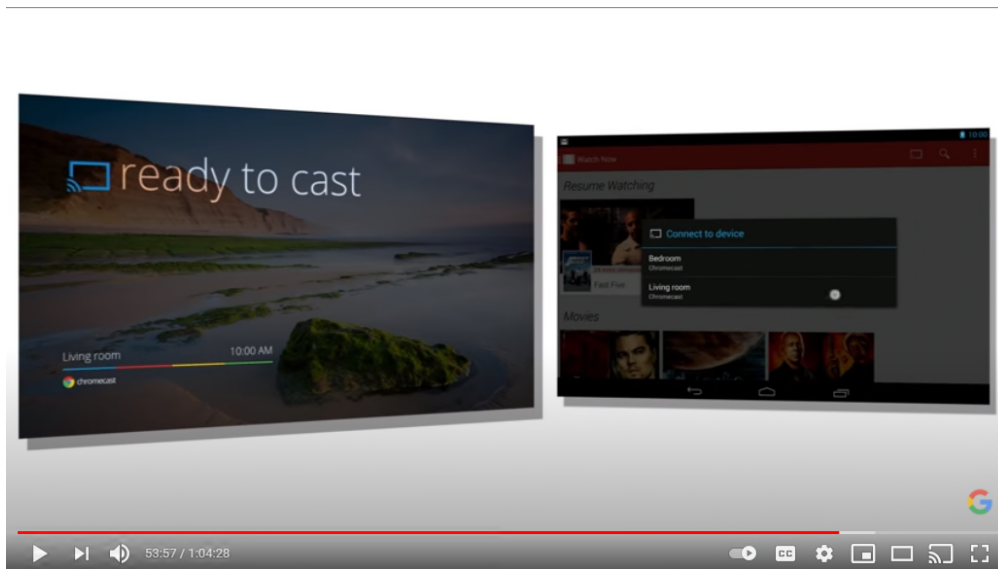
62. Through the managing of the Chromecast by the server, and through the Chromecast’s processing of messaging sent by the Chromecast-compatible app running on the mobile device—as described at ¶¶ 54-61, *supra*—the Chromecast allows the user to consume media content on a remote device, separate from the user’s mobile device, where the app for playing the media and the media content may further be downloaded from the user’s content delivery network (CDN) rather than from the mobile device itself. The user is therefore free to use his or her mobile device for other purposes during playback of the media on the remote device via the Chromecast. See example below.

³⁴ *See*

https://developers.google.com/cast/docs/reference/web_receiver/cast.framework.ui.PlayerData.

³⁵ *See*

https://developers.google.com/cast/docs/reference/web_receiver/cast.framework.messages.SessionState.





Source: <https://www.youtube.com/watch?v=GbXeZl6FoCY> (2013 Google demonstration of Chromecast functionality).

63. Each of the steps discussed above is either performed by or otherwise attributable to Google. To the extent another actor performs any of these steps, Google directs or controls that performance, conditioning participation in the activity or the receipt of a benefit upon performance of the patented method steps, and establishing the manner or timing of that performance. Additionally, Google profits from its infringement and has the right and ability to stop or limit the infringement. For instance, Google tests and demonstrates the accused functionality, including in

advertisements. Further, Google advertises and demonstrates to customers, and directs to Chromecast developers, that the infringing method steps will be performed, as shown above. Further, Google causes automatic updates to the Chromecast. As discussed below, the functionality advertised and directed by Google infringes the Touchstream Patents, and on information or belief, is known by Google to do so.

COUNT I: INFRINGEMENT OF THE '251 PATENT

64. Touchstream adopts and incorporates by reference the allegations set forth in ¶¶ 1-63, *supra*.

65. Google directly infringes at least claim 1 of the '251 patent by performing the methods described in ¶¶ 48-63, *supra*.

66. For example, Google performs the machine-implemented method of controlling presentation of video content on a display device that loads any one of a plurality of different media players. *See, e.g.,* ¶¶ 54-62, *supra*. Google further assigns, by a server system, a synchronization code to the display device. *See, e.g.,* ¶ 50, *supra*. Google further receives, in the server system, a message from a personal computing device that is separate from the server system and separate from the display device, wherein the message includes the synchronization code. *See, e.g.,* ¶¶ 54-55, *supra*. Google further stores, by the server system, a record establishing an association between the personal computing device and the display device based on the synchronization code. *See, e.g.,* ¶ 55, *supra*. Google further receives, in the server system, one or more signals from the personal computing device, the one or more signals specifying a video file to be acted upon and identifying a particular media player for playing the video content, the one or more signals further including a universal playback control command for controlling playing of the video content on the display device by the particular media player. *See, e.g.,* ¶¶ 56-60, *supra*. Google further converts, by the server system, the universal playback control command into

corresponding programming code to control playing of the video content on the display device by the particular media player, wherein converting the universal playback control command includes selecting from among a plurality of specific commands, each of which represents a corresponding playback control command for a respective media player. *See, e.g.*, ¶ 60, *supra*. Google further stores, in a database associated with the server system, information for transmission to or retrieval by the display device, wherein the information specifies the video file to be acted upon, identifies the particular media player for playing the video content, and includes the corresponding programming code to control playing of the video content on the display device by the particular media player in accordance with the universal playback control command. *See, e.g.*, ¶ 61, *supra*.

67. Google's infringement of the '251 patent has been, is, and continues to be willful, including Google's infringement of at least claim 1, as described at ¶¶ 34-44, *supra*.

68. Touchstream has been and will continue to be irreparably harmed by Google's infringing acts, requiring the entry of a permanent injunction to prevent Google's further infringement of the '251 patent because Touchstream does not have another adequate remedy at law.

COUNT II: INFRINGEMENT OF THE '528 PATENT

69. Touchstream adopts and incorporates by reference the allegations set forth in ¶¶ 1-68, *supra*.

70. Google directly infringes at least claim 1 of the '528 patent by performing the methods described in ¶¶ 48-63, *supra*.

71. For example, Google performs the method of controlling presentation of content on a content presentation device that loads any one of a plurality of different media players. *See, e.g.*, ¶¶ 54-62, *supra*. Google further receives, in a server system, one or more messages from a personal computing device that is separate from the server system and separate from the content

presentation device, wherein the one or more messages, taken together, include information associated with a synchronization code assigned to the content presentation device, specify a file to be acted upon, identify a particular media player for playing content from the file, identify a location of the particular media player, and include an action control command for presentation of the content on the content presentation device by the particular media player, the action control command being independent of the particular media player. *See, e.g.*, ¶¶ 54-60, *supra*. Google further uses the information associated with the synchronization code to store a record establishing an association between the personal computing device and the content presentation device. *See, e.g.*, ¶ 55, *supra*. Google further identifies, by the server system, programming code corresponding to the action control command, wherein the programming code is for controlling presentation of the content by the content presentation device using the particular media player. *See, e.g.*, ¶ 60, *supra*. Google further obtains, by the content presentation device, the particular media player, wherein the particular media player is obtained over a network from a content provider. *See, e.g.*, ¶ 57, *supra*. Google further loads the particular media player in the content presentation device. *See, e.g.*, ¶ 57, *supra*. Google further uses the particular media player to execute the programming code with respect to the file. *See, e.g.*, ¶ 57, *supra*.

72. Google's infringement of the '528 patent has been, is, and continues to be willful, including Google's infringement of at least claim 1, as described at ¶¶ 34-44, *supra*.

73. Touchstream has been and will continue to be irreparably harmed by Google's infringing acts, requiring the entry of a permanent injunction to prevent Google's further infringement of the '528 patent because Touchstream does not have another adequate remedy at law.

COUNT III: INFRINGEMENT OF THE '289 PATENT

74. Touchstream adopts and incorporates by reference the allegations set forth in ¶¶ 1-73, *supra*.

75. Google directly infringes at least claims 1 and 6 of the '289 patent by performing the methods described in ¶¶ 48-63, *supra*.

76. For example, Google performs the method of controlling presentation of content on a content presentation device that loads any one of a plurality of different media players. *See, e.g.*, ¶¶ 54-62, *supra*. Google further receives, in a server system, one or more messages from a personal computing device that is separate from the server system and separate from the content presentation device, wherein the one or more messages, taken together, (i) include information associated with a unique identification code assigned to the content presentation device, (ii) specify a file to be acted upon, (iii) identify a particular media player for playing content from the specified file, wherein the media player is a computer application operable to present content and control presentation of the content, (iv) identify a location of the particular media player, and (v) include an action control command for presentation of the content on the content presentation device by the particular media player, the action control command being independent of the particular media player. *See, e.g.*, ¶¶ 54-60, *supra*. Google further uses the information associated with the unique identification code to store a record establishing an association between the personal computing device and the content presentation device. *See, e.g.*, ¶ 55, *supra*. Google further identifies, by the server system, programming code corresponding to the action control command, wherein the programming code is for controlling presentation of the content by the content presentation device using the particular media player, wherein, based on information received or retrieved from the server system, the content presentation device uses the particular media player to execute the programming code with respect to the file. *See, e.g.*, ¶ 60, *supra*. Google further loads, by the

server system, a set of protocols or application programming interfaces from a library based on the identity of the particular media player. *See, e.g.*, ¶ 57, *supra*. Google further identifies, based on the set of protocols or application programming interfaces, programming code corresponding to the action control command, wherein the programming code is for controlling presentation of the content by the content presentation device using the particular media player, wherein, based on information received or retrieved from the server system, the content presentation device uses the particular media player to execute the programming code with respect to the file. *See, e.g.*, ¶ 60, *supra*.

77. Google's infringement of the '289 patent has been, is, and continues to be willful, including Google's infringement of at least claims 1 and 6, as described at ¶¶ 34-44, *supra*.

78. Touchstream has been and will continue to be irreparably harmed by Google's infringing acts, requiring the entry of a permanent injunction to prevent Google's further infringement of the '289 patent because Touchstream does not have another adequate remedy at law.

JURY DEMAND

79. Touchstream demands a trial by jury on all issues.

PRAYER FOR RELIEF

WHEREFORE, Touchstream requests entry of a judgment in its favor and against Google as follows:

- a) Judgment that Google has directly infringed one or more claims of the Touchstream Patents;
- b) An award of damages to compensate for Google's infringement, including damages pursuant to 35 U.S.C. § 284, as well as prejudgment and post-judgment interest.

- c) An award of costs and expenses in this action, including an award of Touchstream's reasonable attorneys' fees pursuant to 35 U.S.C. § 285;
- d) A permanent injunction restraining and enjoining Google, and its respective officers, agents, servants, employees, attorneys, and those persons in active concert or participation with Google who receive actual notice of the order by personal service or otherwise, from any further sales or use of their infringing products and/or services and any other infringement of the Touchstream Patents;
- e) A finding that Google has willfully infringed and is willfully infringing one or more claims of one or more of the Touchstream Patents;
- f) A finding that this case is an exceptional case, and awarding treble damages due to Google's deliberate and willful conduct, and ordering Google to pay Touchstream's costs of suit and attorneys' fees; and
- g) For such other and further relief as the Court may deem just, proper, and equitable under the circumstances.

Dated: June 4, 2021

/s/ Fiona A. Bell

Fiona A. Bell (TX Bar No. 24052288)

SHOOK, HARDY & BACON L.L.P.

600 Travis Street, Suite 3400

Houston, TX 77002

(713) 227-2008

Fax: 713-227-9508

Email: fbell@shb.com

Counsel for Plaintiff

Touchstream Technologies, Inc.