Dynamic Resizable Media Item Player

Client Device GUI Generator

Electronic Device Network Data Store

Content Sharing Platform

Channel A Media Item Channel B Media Item Channel N Media Item

Publication Classification

(51) Int. Cl.
H04L 29/06 (2006.01)
G06F 3/0484 (2006.01)
G06F 3/0482 (2006.01)

(52) U.S. Cl.
CPC .............. H04L 65/60 (2013.01); G06F 3/0482 (2013.01); G06F 3/04842 (2013.01)

Abstract

Described herein is a method for managing media item playback. A method includes receiving a media request to play a media item via a first user interface (UI) on a mobile device. The method further includes causing a media player in a second UI to play the media item. The method also includes displaying a representation of the media item in a first portion of the first UI, the first portion of the first UI being associated with the media player in the second UI. The method includes receiving, via a second portion of the first UI, an activity request to perform an activity that is independent of the viewing of the media item. The method further includes presenting content associated with the requested activity in the first UI while continuing to display, in the first portion of the first UI, the representation of the media item played in the media player in the second UI.
FIGURE 1

Client Device

GUI Generator

Dynamic Media Player

Electronic Device

Network

Data Store

Content Sharing Platform

Channel A

Media Item 116a

Media Item 116b

Media Item 116c

Media Item 116d

Media Item 116e

Media Item 116f

Channel B

Media Item 116c

Media Item 116d

Media Item 116e

Channel N

Media Item 116e

Media Item 116f
FIGURE 4A

FIGURE 4B

FIGURE 4C
Start

Present a media viewer in a first portion of a user interface (UI), the media player to provide playback of a media item

Receive a request to perform an activity that is independent of the viewing of the media item

Present content associated with the requested activity while continuing to provide playback of the media item in the media player of the first portion of the UI

Receive, via the second portion, a request for video playback of a second media item associated with the activity

Present video playback of the second media item in the first portion of the UI

End

FIGURE 7
Receive a media request to play a media item via a first user interface (UI) of a content sharing platform on a mobile device

Cause a media player in a second UI to play the media item

Present a representation of the media item in a first portion of the first UI

Receive, via a second portion of the first UI, an activity request to perform an activity that is independent of the viewing of the media item

Present content associated with the requested activity in the first UI while the media player in the second UI provides playback of the media item

Receive a transport control request in the first UI

In response to the received transport control request, cause the media player in the second UI to update playback of the media item

End
Present a media player in a first portion of a user interface (UI), the media player to provide playback of a first media item

Receiving a user selection of a second media item from the list of media items, the user selection including a gesture to move the second media item from the second portion to the first portion

In response to the user selection, cause the media viewer to provide playback of the second media item when playback of the first media item is complete

Receive, via the second portion of the UI, a user request to perform an activity that is independent of the viewing of the first media item or the second media item

Present content associated with the requested activity while continuing to provide playback of either the first media item or the second in the media viewer of the first portion of the UI

End

FIGURE 9
Receive a request from a first device with a first user interface (UI) for a media item to play in a second UI.

Authorized?

Yes

Provide media item to first device, where the first device either plays the media item or provides the media item to a second device.

No

Provide the media item to the second device that includes the second UI.

End
DYNAMIC RESIZABLE MEDIA ITEM PLAYER

RELATED APPLICATION

[0001] This application is a continuation of U.S. patent application Ser. No. 14/457,005, filed Aug. 11, 2014, which claims the benefit of U.S. Provisional Application No. 61/865,086, filed Aug. 12, 2013, the entire contents of which are incorporated herein by reference.

TECHNICAL FIELD

[0002] Implementations of the present disclosure relate to a media item playback system, and more specifically, to a dynamic resizable media item player on a mobile device.

BACKGROUND

[0003] Online content sharing platforms typically allow users to upload, view, and share digital content such as media items. Media items may include audio clips, movie clips, TV clips, music videos, images, or other multimedia content. Users may use computing devices (such as smart phones, cellular phones, laptop computers, desktop computers, networked computers, television, set-top boxes) to use, play, and/or otherwise consume media items (e.g., watch digital videos, and/or listen to digital music).

SUMMARY

[0004] The below summary is a simplified summary of the disclosure in order to provide a basic understanding of some aspects of the disclosure. This summary is not an extensive overview of the disclosure. It is intended to neither identify key or critical elements of the disclosure, nor delineate any scope of the particular implementations of the disclosure or any scope of the claims. Its sole purpose is to present some concepts of the disclosure in a simplified form as a prelude to the more detailed description that is presented later.

[0005] In one implementation, a method includes presenting a media player in a first portion of a user interface (UI) of a content sharing platform on a mobile device, the media player to provide playback of a media item. The method further includes receiving, via a second portion of the UI, a user request to perform an activity that is independent of the viewing of the media item. The method further includes presenting content associated with the requested activity while continuing to provide playback of the media item in the media player of the first portion of the UI.

[0006] In some implementations, the method can include receiving, via the second portion, a request for video playback of a second media item associated with the activity. The method can further include presenting video playback of the second media item in the first portion of the UI. The media player can be presented as part of an application. The activity can be associated with the application. The content can be provided by the media platform. The activity can be a search for a second media item. The content can be provided by a search engine platform. The activity can be social networking. The content presented in the second portion can be provided by a social network application. The activity can be messaging. The content can include a plurality of messages from at least two different users of a message service. The activity can include an interaction with an electronic advertisement. The first portion of the UI can be presented semi-transparently. The second portion of the UI can be associated with a social network application.

[0007] In additional implementations, computing devices for performing the operations of the above described implementations are also disclosed. Additionally, in implementations of the disclosure, a computer readable storage media stores methods for performing the operations of the above described implementations. Further, in implementations of the disclosure, means for performing the operations of the above described implementations are also disclosed.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] The present disclosure will be understood more fully from the detailed description given below and from the accompanying drawings of various embodiments of the present disclosure, which, however, should not be taken to limit the present disclosure to the specific embodiments, but are for explanation and understanding only.

[0009] FIG. 1 illustrates an example system architecture in which embodiments of the present disclosure can be implemented.

[0010] FIG. 2A illustrates an example graphical user interface (GUI) that includes a dynamic media player, in accordance with embodiments.

[0011] FIG. 2B illustrates an example GUI that includes an alternate view of a dynamic media player, in accordance with embodiments.

[0012] FIG. 2C illustrates an example GUI that includes an alternate view of a dynamic media player, in accordance with embodiments.

[0013] FIG. 3 illustrates an example GUI that includes a dynamic media player, in accordance with another embodiment.

[0014] FIG. 4A illustrates an example GUI that includes a dynamic media player, in accordance with another embodiment.

[0015] FIG. 4B illustrates an example GUI that includes an alternate view of a dynamic media player, in accordance with another embodiment.

[0016] FIG. 4C illustrates an example GUI that includes an alternate view of a dynamic media player, in accordance with another embodiment.

[0017] FIG. 5A illustrates an example GUI that includes a dynamic media player, in accordance with another embodiment.

[0018] FIG. 5B illustrates an example GUI that includes an alternate view of a dynamic media player, in accordance with another embodiment.

[0019] FIG. 6 illustrates an example system that includes a first user device with a first UI to cause a media item to play in a media player in a second UI on a first or second user device, in accordance with some embodiments.

[0020] FIG. 7 is a flow diagram illustrating a method of providing a dynamic resizable media item player, in accordance with some embodiments.

[0021] FIG. 8 is a flow diagram illustrating a method of providing a media item player in a first UI to control media item playback in a second UI, in accordance embodiments.

[0022] FIG. 9 is a flow diagram illustrating a method of providing a dynamic resizable media item player, in accordance with some embodiments.

[0023] FIG. 10 is a flow diagram illustrating a method for receiving a request for a media item via a first UI and provid-
ing the requested media item for playback in a second UI, in accordance with some embodiments.

FIG. 11 illustrates block diagram of an example computing device that may perform one or more of the operations described herein.

DETAILED DESCRIPTION

Conventionally, media applications on mobile devices typically have two mutually exclusive modes of operation—discovery or playback. Because of this, a user may not be able to simultaneously watch a media item and do other online activities (e.g., searches for other media items). For example, users may have to stop video playback to search for another video to watch. This is particularly problematic with short-form video (e.g., short video clips), when a larger number of video discovery decisions may be made in a relatively short amount of time as compared to long-form video (e.g., movies). This may inhibit users from exploring and searching for more content and may lead to less video consumption. Further, conventional media applications on mobile devices may not permit a user to perform other operations outside of the media application (e.g., browsing content, surfing the web, social networking, checking email) while consuming a media item at the same time. Under these conventional systems, video playback is typically stopped when a user starts doing any other type of activity.

Implementations of the present disclosure address these and other shortcomings by providing playback of a media item while allowing a user to perform other online or offline activities within the interface. For example, a graphical user interface (GUI) may be presented with a dynamic media player that can play a video and be persistently visible when a user browses the web at the same time. The dynamic media player provides a “watch-while” experience. The “watch-while” experience allows a user of a content sharing platform or service to minimize the dynamic media player into a smaller sub-viewer in a GUI. The GUI may be provided, for example, by a mobile application (e.g., an “app” provided by a content sharing service) running on a user mobile device, by an operating system of the user mobile device or by a web browser running on the user mobile device (e.g., when the user accesses the website of the content sharing service). With the video player minimized in a first portion of the GUI, the user can then engage in other activities, such as browse or search for media items to decide what to watch next in a second portion of the GUI without interrupting video playback by the video player in the first portion of the GUI.

Although the description herein refers to video as an example type of media item, implementations of the disclosure may apply to other types of media items such as images, audio, and other multi-media.

FIG. 1 illustrates an example system architecture 100 for providing a dynamic media player 110 in accordance with implementations. The system architecture 100 includes one or more client devices 102, a network 104, a data store 112, and a content sharing platform 106. In one implementation, the network 104 may include a public network (e.g., the Internet), a private network (e.g., a local area network (LAN) or wide area network (WAN)), a wired network (e.g., Ethernet network), a wireless network (e.g., an 802.11 network or a Wi-Fi network), a cellular network (e.g., a Long Term Evolution (LTE) network), routers, hubs, switches, server computers, and/or a combination thereof. In one implementation, the data store 112 may be a memory (e.g., random access memory), a cache, a drive (e.g., a hard drive), a flash drive, a database system, or another type of component or device capable of storing data. The data store 112 may also include multiple storage components (e.g., multiple drives or multiple databases) that may also span multiple computing devices (e.g., multiple server computers).

The client device 102 may include computing devices such as personal computers (PCs), laptops, mobile phones, smart phones, tablet computers, netbook computers etc. In some implementations, the client device 102 may also be referred to as “user device” or “user mobile device.” Each client device 102 includes a dynamic media player 110, as described herein.

In one implementation, the content sharing platform 106 may be one or more computing devices (such as a rack-mount server, a router, a server computer, a personal computer, a mainframe computer, a laptop computer, a tablet computer, a desktop computer, etc.), data stores (e.g., hard disks, memories, databases), networks, software components, and/or hardware components that may be used to provide a user with access to media items and/or provide the media items to the user. For example, the content sharing platform 106 may allow a user to consume, upload, search for, approve of (“like”), dislike, and/or comment on media items. The content sharing platform 106 may also include a website (e.g., a webpage) that may be used to provide a user with access to the media items.

A media item may be consumed via the Internet, the content sharing platform 106 and/or via a mobile device application. In one implementation, the content sharing platform 106 may store the media items using the data store. The content sharing platform 106 may also store “playlists” of media items. A playlist may be a list of media items that can be played (e.g., streamed) in a sequential or shuffled order. For example, when the playback of one media item from the playlist is finished, the playback of the next media item in the playlist can automatically begin.

The content sharing platform 106 may include multiple channels 114a-c (e.g., channels A through N). A channel 114 can be data content available from a common source or data content having a common topic or theme. The data content can be digital content chosen by a user, digital content made available by a user, digital content uploaded by a user, digital content chosen by a content provider, digital content chosen by a broadcaster, etc. For example, a channel X can include videos Y and Z. A channel 114 can be associated with an owner, who is a user that can perform actions on the channel. Different activities can be associated with the channel 114 based on the owner’s actions, such as the owner making digital content available on the channel, the owner selecting (e.g., liking) digital content associated with another channel, the owner commenting on digital content associated with another channel, etc. The activities associated with the channel 114 can be collected into an activity feed for the channel. Users, other than the owner of the channel, can subscribe to one or more channels in which they are interested. Once a user subscribes to a channel 114, the user can be presented with information from the channel’s activity feed. If a user subscribes to multiple channels, the activity feed for each channel to which the user is subscribed can be combined into a syndicated activity feed. Information from the syndicated activity feed can be presented to the user.
Each channel 114 may include one or more media items 116. Examples of a media item can include, and are not limited to, digital video, digital movies, digital photos, digital music, website content, social media updates, electronic books (ebooks), electronic magazines, digital newspapers, digital audio books, electronic journals, web blogs, real simple syndication (RSS) feeds, electronic comic books, software applications, etc. As used herein, “media,” “media item,” “online media item,” “digital media,” “digital media item,” “content,” and “content item” can include an electronic file that can be executed or loaded using software, firmware or hardware configured to present the digital media item to an entity.

Each client device 102 includes a GUI generator 108 that may present a GUI including a dynamic media player 110 to provide playback of a media item. The GUI generator 108 may be, for example, part of a web browser, a mobile app (e.g., a standalone application that allows users to view digital media items 114 such as digital videos, digital images, electronic books, etc.), or an operating system that can manage the dynamic media player 110 and may send media items to the dynamic media player 110 for playback. Some example GUIs provided by the GUI generator 108 include control elements in the form of a button (e.g., a button for subscribing to a channel, playing a media item, minimizing a media player, etc.). Alternatively or in addition, various other control elements can be used for selection by a user such as a check box, a link, etc. Example GUIs may include a search tool (e.g., to search for a channel or media item of interest), an upload tool (e.g., to upload a new media item to an internet-based content platform), a menu (e.g., to navigate to different GUIs of the internet-based content platform), a user identifier, a settings tool (e.g., to configure settings of GUIs of the internet-based content platform), a mini-player tool (e.g., to minimize a dynamic media player that presents media items while other content is being displayed on a screen of the client device), a navigation tool for navigating to other platforms (e.g., an email platform, a social platform, a storage platform, an image platform, a calendar platform, among others), etc.

The dynamic media player 110 included in the GUI may render, display, and/or present the content (e.g., a web page, media items) to a user. The dynamic media player 110 may be provided to the client device 102 by a server and/or content sharing platform 106. For example, the dynamic media player 110 may be an embedded media player that is embedded in web pages provided by the content sharing platform 106.

The GUI may include a first portion to present the dynamic media player 110 providing media item playback and a second portion to present other content provided by the same application (e.g., a web browser or an application of the content sharing platform 106) or a different application (e.g., a media discovery application, an email application, a social network application, a calendar application, etc.). The media player providing media item playback can be visible to a user while the user engages in other activities via the second portion of the GUI. In implementations, the dynamic media player 110 can appear to “float” on top of other content provided by the same or different applications. For example, the dynamic media player 110 can appear to float on top content of a media discovery application, a content hosting application, a media item consumption platform, a social networking application, a browser, or the like.

A user can minimize or summon the dynamic media player 110 by using gestures on a graphical user interface (GUI). For example, a user can touch a static media player (not shown) and, while in contact with a touch screen, drag it to another position in the GUI, causing the dynamic media player 110 to appear in that other position in the GUI. The dynamic media player 110 can be smaller than a static media player, thus allowing other content to be displayed within the same GUI. As discussed above, the dynamic media player 110 can be managed by an application (e.g., a mobile app or a web browser), and while a media item provided by the application (e.g., a media item requested from the content sharing platform 106) is being played by the dynamic media player 110 managed by the application, the user can browse content or perform other activities within the same application. In another example, the dynamic media player 110 can be managed by an operating system of the client device 102, and while a media item provided by one application is being played by the dynamic media player 110 managed by the operating system, the user can switch to another application and browse content or perform other activities within the other application. In either example, the dynamic media player 110 can be used to permit simultaneous media consumption along with other activities enabled by the same or different applications.

In implementations, the dynamic media player 110 is enabled by default and can change position based on a user’s touch input (e.g., gestures) to enable a “watch-while” experience. The dynamic media player can be minimized, summoned, resized and dismissed using gestures (e.g., pinch to expand or contract, swipe up to expand, swipe down to collapse). The dynamic media player 110 may be implemented in any of a variety of sizes and is not limited to a specific position within the GUI of the content sharing platform application. In some embodiments, the dynamic media player 110 may be referred to as a “mini player.” The dynamic media player may also be moved (e.g., dragged) within the GUI to suit the user’s preferences.

For example, the dynamic media player 110 can be minimized or maximized by swiping vertically, and can be dismissed by swiping horizontally. The dynamic media player 110 can also be dismissed via a click target (e.g., an “x” button). In other implementations, when the user starts to drag the dynamic media player 110, a drag target (e.g., a trash can) appears on the right hand side of the screen. The user can drag the dynamic media player 110 on top of the drag target to dismiss it. In another example, tapping on the dynamic media player 110 expands the dynamic media player 110. The dynamic media player 110 can have “magnetic-like” attachment to a specific region of a GUI presented on the client device. When using a gesture to move the dynamic media player 110, it can snap into place and become docked. Such attachment to a specific region of the screen can persist regardless of other content within the GUI. For example, the docking positions remain even when a user is checking email in a different application.

In some implementations, the dynamic media player 110 may be introduced as a smaller format player and persist in this size throughout the user’s viewing experience. In further implementations, a transparency level of the dynamic media player 110 can be variable and can be based on user gestures. For example, a user may prefer to see both a video playing in the dynamic media player as well as text behind the dynamic media player. To accomplish this, the
dynamic media player 110 is semi-transparent. In implementations, the dynamic media player transparency can be adjustable by gestures. The user’s interaction with the GUI may be passed through to the content below the semi-transparent player, and the user could browse while watching a media item in the dynamic media player 110. In some implementations, the dynamic media player 110 may remain in its original location, while the “browsing” content is presented on top of the dynamic media player 110 (in some cases, the dynamic media player 110 may be partially obscured).

In some implementations, the dynamic media player 110 may maintain the same quality of the presented content item (e.g., video) while the dynamic media player 110 is minimized. As a result, after the user “unminimizes” the dynamic media player 110, the quality remains as it was before the dynamic media player 110 was minimized. In contrast, conventional solutions typically downgrade the quality of a video when the video is presented in a smaller format because the video is originally downloaded with a higher resolution and then transcoded to a lower quality to suit the size of the player. As such, with conventional solutions, when the player is returned to a larger size, it continues to show a low quality until a next high-quality video chunk can be downloaded.

In some embodiments, the dynamic media player 110 may be part of video-chat (video messaging) component of a social network. For example, a user may view a video of a person that they are video-chatting with in a minimized view while simultaneously browsing the social network application. The user could then make the video-chatting window/player full-sized again at will.

The GUI generator 108 may also present a video that is being recorded by a camera of the client device 102 while the user browses other features in the content sharing platform application, for example, while sharing previously recorded videos, or changing video-recording settings on the fly.

In some implementations, a mapping application may show a live navigation map in a minimal view of a content item player, so that a user can change navigation settings and still see what is coming up in their navigation route. In other implementations, the mapping application may show a “street view” of a location (e.g., current location, selected location) of a mobile device while the rest of the screen shows a map or navigation information.

In some implementations, metadata can be presented via the interface that describes characteristics or information of a media item being presented in the dynamic media player 110. The metadata can be displayed over the top of the dynamic media player or can be displayed adjacent to the dynamic media player 110. In implementations, when a new media item is played in the dynamic media player 110, metadata of the new media item can be presented via the interface. Examples of the media item metadata can include a title, a length, an artist, an uploading entity, among others.

In implementations, transport controls can be presented in the GUI for the dynamic media player 110. The transport controls can include next, previous, pause, play, stop, scrub, and the like. The transport controls can be displayed over the dynamic media player 110 or adjacent to the dynamic media player 110. The transport controls can be toggled between a hidden and a displayed state. For example, the transport controls can be hidden and when a user taps the screen of the client device 102, the transport controls can be displayed in response.

The system 100 may also include an electronic device 118, which may include any type of computing devices such as personal computers (PCs), laptops, mobile phones, smart phones, tablet computers, netbook computers, televisions, set top boxes etc. As described in greater detail below, a user may select a media item 116 via a GUI presented on the client device 102, and the media item 116 may be played on the electronic device 118 in response to the user selection.

In general, functions described in one implementation as being performed by the content sharing platform can also be performed on the client device in other implementations if appropriate. In addition, the functionality attributed to a particular component can be performed by different or multiple components operating together. The content sharing platform can also be accessed as a service provided to other systems or devices through appropriate application programming interfaces, and thus is not limited to use in websites.

FIGS. 2A, 2B, 2C, 3A, 4A, 4B, 4C, 5A and 5B illustrate example graphical user interfaces (GUI) 200 in accordance with some embodiments. The example GUIs 200 may be presented by and/or displayed within a web browser (e.g., when a user accesses a content sharing platform via a web browser), or within an application (e.g., an app, an application, a program, a software module/component, etc.), that may be used to subscribe to channels and view, play, and/or consume media items of the channels. Some example GUIs include control elements in the form of a button (e.g., a button for subscribing to a channel). However, it should be noted that various other control elements can be used for selection by a user such as a check box, a link, or any other user interface elements.

FIG. 2A illustrates an example GUI 200a for minimizing a dynamic media player 202a in accordance with aspects of the present disclosure. The example GUI 200a includes an indicator 204 that the dynamic media player 202a may be minimized. The indicator 204 may be any type of button, selectable icon, etc. In an example, the indicator 204, when activated, can present the dynamic media player seemingly on top of the other elements of the GUI, as illustrated in FIG. 2B. Selecting (e.g., clicking or tapping) on the indicator 204 (e.g., the small triangle in the upper-left corner of the dynamic media player) may cause the dynamic media player 202a to transition to a minimized state 202b (FIG. 2B). In some embodiments, the indicator 204 is a drag animation that visually appears in the GUI 200 once a user attempts to minimize the dynamic media player 202a. In one implementation, the user can also make a selection anywhere in the dynamic media player 202a and drag downwards and the minimized dynamic media player 202b appears in the GUI, as in GUI 200b of FIG. 2B. In another implementation, a user can use gestures to minimize or summon the minimized dynamic media player 202b. An example user gesture is illustrated by the downward arrow. In this example, the user touches a screen of a client device over the dynamic media player 202a, and while maintaining contact with the screen of the client device, the user drags the dynamic media player 202a to another location on the GUI 200. FIG. 2B illustrates an example in which the dynamic media player 202b is “in transit” (while the user is still manipulating the position of the dynamic media player 202b using gestures). FIG. 2C illustr-
trates a docked position of the dynamic media player 202c. When minimized, the dynamic media player 202c may be presented in the lower right corner of the GUI 200c. In other implementations, the dynamic media player may reside in other portions of the GUI while “minimized.” While minimized, the dynamic media player 202c may continue to play a media item while the user performs other actions within the GUI 200.

[0051] FIG. 3 illustrates an example GUI 300 for adding a media item 302 to a queue, such as a playlist, from a search result 304 in accordance with aspects of the present disclosure. In an example, the user gestures are illustrated by the arrow 306. At the upper portion of the arrow 306, a circle illustrates the user’s initial touch on a screen of a client device. While maintaining contact with the screen, the user can drag the search result into the dynamic media player 308. The dynamic media player 308 can begin playing the media item 302 represented in the search result 304b. In other implementations, when a media item 302 is drag-and-dropped onto the dynamic media player 308, the media item 302 can be added to a playlist.

[0052] FIGS. 4A-C illustrate example GUIs 400a-c for dismissing the dynamic media player 404 in accordance with aspects of the present disclosure. The dynamic media player 404 can be dismissed using gestures, such as when a user clicks or touches a “close” button or uses gestures. As illustrated in FIG. 4A, a user touches the dynamic media player 404 on a screen of a client device and, while maintaining contact with the screen, the user slides the dynamic media player 404 off of the screen, as illustrated by FIG. 4B. The dynamic media player 404 is dismissed when the user moves (e.g., swipes) the dynamic media player 404 off of the screen. In implementations, when a predetermined portion of the dynamic media player 404 (e.g., fifty percent, sixty percent, etc.) is no longer visible on the GUI 400, the dynamic media player 404 will continue to dismiss even if the user loses contact with the screen. For example, at FIG. 4C approximately sixty percent of the dynamic media player 404 is not visible on the GUI 400c. Should the user lose contact with the screen, the dynamic media player 404 may still dismiss from the visible area of the GUI.

[0053] In other implementations, while the dynamic media player 404 is being moved toward a dismissal point (e.g., edge of a screen), a media item being played by the dynamic media player 404 can change state, such as become more transparent. The closer the dynamic media player 404 is to the dismissal point, the closer dynamic media player 404 is to being transparent. The transparency level of the dynamic media player 404 can change depending on its proximity to the dismissal point. In further implementations, when a media item (e.g., video) is being played while the dynamic media player 404 is being dismissed, sound associated with the video can be attenuated. For example, as the dynamic media player 404 approaches the dismissal point, an audio portion of the video can decrease in volume. In other implementations, the dynamic media player 404 can undergo a combination of the transparency alteration and sound attenuation while it is being dismissed. Should a user decide to not dismiss the dynamic media player 404 after starting to move the dynamic media player 404 toward the dismissal point, the user can swipe in the opposite direction and any transparency alteration or sound attenuation can be reversed.

[0054] FIGS. 5A-B illustrate example GUIs 500 that demonstrate the dynamic nature of the dynamic media player in accordance with aspects of the present disclosure. The example GUIs include a menu 508 (e.g., a guide tray), a browse area 502 and a dynamic media player 506. The menu 508 and the browse area 502 can be presented as if they were one continuous surface. In FIG. 5A, the menu 508 is not viewable by the user in the GUI 500a. For reference purposes, the left side of the dynamic media player 506 is coincident with an invisible reference line 510. A user, for example, can use a gesture to move the browse area 502 to the right, which exposes the menu 508, as illustrated in FIG. 5B. When the menu 508 is moved onto the viewing area of the GUI 500b, the position of the dynamic media player 506 can dynamically change such that it does not overlap the menu 508. In FIG. 5B, the position of the dynamic media player 506 is further to the right when the menu 508 is visible as compared to the position of the dynamic media player 506 in GUI 500a.

[0055] The menu 508 and browse area 502 may work together as a single, continuous surface. Swiping to the right moves the browse area 502 off the screen to the right, while the menu 508 moves in from the left. The menu 508 and browse area 502 do not overlap each other, but form a single surface which slide together as a unit. The dynamic media player 506 can slide over the top of the display area of the screen independent of the menu 508 and browse area 502.

[0056] FIG. 6 illustrates an example system 600 that includes a first user device 602 with a first UI 605 to cause a media item 116 to play in a media player 612 in a second UI 611 on a first or second user device, in accordance with some embodiments. The first device 602 may be a client device 102, as described in conjunction with FIG. 1. The second device 610 may be a different client device 118, as described in conjunction with FIG. 1. In some embodiments, the first UI 605 and the second UI 611 are both presented on the first device 602. In other embodiments, the first UI 605 is presented on the first device 602 and the second UI 611 is presented on the second device 610. The first device 602 and the second device 610 may be in communication via a network 104, as described in conjunction with FIG. 1.

[0057] The first device 602 may include a display to present a UI 605 that includes content 604, as described herein. The content 604 may be any content (e.g., search results, social media, webpage) that includes or identifies at least one media item 116. The first device 602 may also present a mini player 606 in a first portion of the UI 605 and the content 604 in a second portion of the UI 605. The mini player 606 can be an interface tool that receives requests from a user, via the UI 605, that pertain to playback of the media item 116 in the second UI 611. For example, a user may provide input via the mini player 606 to play a media item 116 in the second UI. In a specific example, the user may drag and drop a media item 116 to the mini player 606. Upon receiving the dropped media item 116, the mini player 606 may cause the media item 116 to be played in the second UI 611.

[0058] A user of the first device 602 may provide a media request to play the media item 116 in the second device 610. For example, first device 602 may receive the media request from the user in the form of a gesture. The user may touch an area of the display of the first device 602 that corresponds to the media item 116 and, while keeping in contact with the display, the user may drag the media item 116 or its identifier (e.g., link) to the mini player 606, similar to what is described in conjunction with FIG. 3. The mini player 606 is associated
with the media player 612 and can cause media items to play, stop, pause, fast forward, rewind, etc. in the media player 612 in the second UI 611.

[0059] Upon receiving the media request to play the media item 116, the first device 602 may cause a media player in the second UI 611 to play the media item 116. In some embodiments, the first device 602 requests the media item 116 from a content platform (not shown), such as the content sharing platform 106 of FIG. 1. The content platform may determine whether the second device 610 is authorized to play the media item 116. If the second device 610 is authorized (e.g., based on a user account on the content platform), the content platform may provide the media item 116 to the second device 610. In some implementations, the content platform sends (e.g., streams) the media item 116 directly to the second device 610. In some implementations, the content platform sends the media item 116 to the first device 602 and the first device 602 then sends (e.g., streams) the media item 116 to the second device 610. In one example, the first device 602 may be a mobile device (e.g., a smart phone or a tablet) of a user, and the second device 610 may be a television of the user. Alternatively, the first device 602 may be a mobile device of a user, and the second device 610 may be a laptop computer of the user. Yet alternatively, the first device 602 may be a user device (e.g., a mobile device, a laptop, etc.) of a first user, and the second device 610 may be a user device (e.g., a mobile device, a laptop, a television, a smart TV, etc.) of a second user.

[0060] In further implementations, the system 600 includes a set top box 620 that is coupled to the second device (e.g., a television) 610, such as via a cable or the network 104, and the set top box 620 receives the media item 116 from the first device 602 or the content platform, and sends the media item 116 to the media player 612. In some embodiments, the set top box 620 performs one or more encode or decode operations on the media item 116 before sending to the media player 612.

[0061] While the media player 612 is presenting the media item 116 to a user, the first device 602 may receive an activity request (such as via the first UI 605) from the user to perform an activity that is independent of the viewing of the media item 116. The first device 602 may present content 604 associated with the requested activity in the first UI 605 while the media player 612 in the second UI 611 provides playback of the media item 116. The first device 602 may also present, in the mini player 606, a representation (e.g., thumbnail, title) of the media item 116 being played in the media player 612. Such a representation may be displayed only if the playback of a respective media item was requested on the first device 602. Once the playback of the media item ends, the first device 602 is notified and the representation of the media item is no longer displayed in the mini player 606. Alternatively, a presentation of any media item played by the media player 612 is displayed in the mini player 606, including media items requested for playback on the second device 610. In some implementations, the mini player 606 maintains a list of media items played by the media player 612 that can be viewable by the user. The user can then use this list to create a playlist of media items, or such a playlist can be automatically created.

[0062] The first device 602 may also present one or more transport controls 624 for controlling playback of the media item 116 in the media player 612. The transport controls may also include a scrubber for changing the playback position of the media item 116. The transport control 624 may receive a request pertaining to playback of the media item in the media player 612 (e.g., to rewind, stop, pause, play and fast forward). In some embodiments, the first device 602 receives a selection of the transport control (e.g., play control) and, in response to the received user selection, the first device 602 instructs a content platform (not shown) to stream the media item for playback in the media player 612, as described herein.

[0063] FIGS. 7-10 are flow diagrams illustrating methods for performing various operations, in accordance with some embodiments of the present disclosure, including providing a dynamic resizable media item player. The methods may be performed by processing logic that comprises hardware (e.g., circuitry, dedicated logic, programmable logic, microcode, etc.), software (e.g., instructions run on a processor to perform hardware simulation), or a combination thereof. Processing logic can control or interact with one or more devices, applications or user interfaces, or a combination thereof, to perform operations described herein. When presenting, receiving or requesting information from a user, processing logic can cause the one or more devices, applications or user interfaces to present information to the user and to receive information from the user.

[0064] For simplicity of explanation, the methods of FIGS. 7-10 are depicted and described as a series of operations. However, acts in accordance with this disclosure can occur in various orders and/or concurrently and with other operations not presented and described herein. Further, not all illustrated operations may be required to implement the methods in accordance with the disclosed subject matter. In addition, those skilled in the art will understand and appreciate that the methods could alternatively be represented as a series of interrelated states via a state diagram or events.

[0065] FIG. 7 is a flow diagram illustrating a method 700 of providing a dynamic resizable media item player, in accordance with some embodiments of the present disclosure. In one implementation, method 700 may be performed by a GUI generator 108, as illustrated in FIG. 1.

[0066] Referring to FIG. 7, the method 700 begins at block 702 where processing logic presents a media player in a first portion of a user interface (UI) of a content sharing platform on a mobile device, the media player to provide playback of a media item.

[0067] At block 704, the processing logic receives, via a second portion of the UI, a user request to perform an activity that is independent of the viewing of the media item. The processing logic may present the media player as part of an application and the activity is associated with the application. The activity may be a search for a second media item, and wherein the content is provided by a search engine platform. In some implementations, the activity comprises an interaction with an electronic advertisement.

[0068] At block 706, the processing logic presents content associated with the requested activity while continuing to provide playback of the media item in the media player of the first portion of the UI. In some implementations, the activity is social networking, and the content presented in the second portion is provided by a social network application.

[0069] At block 708, the processing logic receives, via the second portion, a request for video playback of a second media item associated with the activity. For example, a user may discovery a video of interest and may request to watch
At block 710, the processing logic presents video playback of the second media item in the first portion of the UI.

At block 906, the processing logic causes the media player to automatically provide playback of the second media item when playback of the first media item is complete. In some implementations, causing the media player to provide playback of the second media item includes presenting, in the first portion of the UI, a transport control for the media player to advance to a next media item in the playlist when receiving a selection of the transport control, and in response to the selected transport control, causing the media player to provide playback of the second media item. In some implementations, the playlist includes at least two different types of media items. For example, the playlist may include a video and an audio track. In some embodiments, responsive to the user selection, the processing logic adds the second media item to a playlist.

At block 908, the processing logic receives, via the second portion of the UI, a user request to perform an activity that is independent of the viewing of the first media item or the second media item, as described herein.

At block 910, the processing logic presents content associated with the requested activity while continuing to provide playback of either the first media item or the second media item in the media player of the first portion of the UI.

FIG. 10 is a flow diagram illustrating a method 1000 for receiving a request for a media item via a first UI and providing the requested media item for playback in a second UI, in accordance with some embodiments. In one implementation, method 1000 may be performed by a content sharing platform 106, as illustrated in FIG. 1.

Referring to FIG. 10, the method 1000 begins at block 1002 where processing logic receives a request from a first device with a first user interface (UI) for a media item to play in a second UI. The first UI may be presented on a first device and the second UI may be presented either on the first device or on a second device. In some implementations, the request from the first device includes both the request for the media item and an identifier of the second UI. When the second UI is associated with a second device, the identifier of the second UI may include characteristics of the second device, such as a MAC address, IP address, a unique identifier assigned by a content hosting platform, etc.

At block 1004, the processing logic may perform a check as to whether the second UI is authorized to play the media item. In embodiments, the processing logic may use the identifier of the second UI to perform the check. The processing logic may query a database that includes a record of identifiers of authorized UIs and/or devices. When the processing logic locates the identifier from among the record of identifiers of authorized UIs and/or devices, the processing logic can proceed to block 1006 or 1008.

In some embodiments, the first device and the second UI/device are associated with a user account. When the processing logic receives the request from the first device to play the media item on the second device, the processing logic can check for devices that have been associated with the user account. When both the first device and the second device are associated with the account, the processing logic can proceed to block 1008.

In some embodiments where the request does not include an identifier of the second UI or a second device, the processing logic can check a user account for authorized devices. Upon identifying an authorized device that includes the second UI, the processing logic may proceed to block 1008. For example, a user may associate a mobile device and...
a television with a user account. When the processing logic receives a request from the mobile device to play a media item, the processing logic can identify the television as being an authorized device and then proceed to block 1008.

[0088] In some embodiments, the first device is associated with a first user and the second device is associated with a second user. Upon receiving a request from the first user device to play the media item on the second user device, the processing logic may check whether the second user has authorized media items from the first user. For example, the second user may have previously provided input to always receive and allow media items from the first user. Alternatively, the processing logic can provide a prompt to the second user to confirm whether the second user would like to receive the media item. For example, when the first user requests to provide the media item to the second user, a GUI dialog box may appear on the second device that asks the second user if they would like to receive the media item. When the second user provides affirmative input (e.g., yes, accept), the processing logic may proceed to block 1008.

[0089] In some implementations, the processing logic may request user credentials from a requesting user before performing the authorization check. Upon receiving valid access credentials, the processing logic may proceed with the authorization check at block 1004.

[0090] At block 1006, the processing logic provides the media item to the first device. The processing logic can provide the media item directly, or the processing logic may provide instructions to a media server to provide the media item to the first device. When the first device includes the second UI, the first device may play the media item in the second UI. When a second device presents the second UI, the first device may provide (e.g., stream) the media item to the second device for playback in a media player in the second UI.

[0091] At block 1008, the processing logic can provide the media item directly to a second device when the second UI is associated with the second device.

[0092] Although implementations of the disclosure are discussed in terms of content sharing platforms and media consumption view the content sharing platform, implementations may also be generally applied to any type of social network providing connections between users. Implementations of the disclosure are not limited to content sharing platforms that provide channel subscriptions to users.

[0093] In situations in which the systems described here collect personal information about users, or may make use of personal information, the users may be provided with an opportunity to control whether the content sharing platform collects user information (e.g., information about a user's social network, social activities or activities, profession, a user's preferences, or a user's current location), or to control whether and/or how to receive content from the content server that may be more relevant to the user. In addition, certain data may be treated in one or more ways before it is stored or used, so that personally identifiable information is removed. For example, a user's identity may be treated so that no personally identifiable information can be determined for the user, or a user's geographic location may be generalized where location information is obtained (such as to a city, ZIP code, or state level), so that a particular location of a user cannot be determined. Thus, the user may have control over how information is collected about the user and used by the content sharing platform.
"computer-readable storage medium" can include a single medium or multiple media (e.g., a centralized or distributed database, and/or associated caches and servers) that store the sets of instructions 1126. The term "computer-readable storage medium" can include any medium that is capable of storing, encoding or carrying a set of instructions for execution by the machine and that cause the machine to perform any one or more of the methodologies of the present disclosure. The term "computer-readable storage medium" can include, but not be limited to, solid-state memories, optical media, and magnetic media.

0100 In the foregoing description, numerous details are set forth. It will be apparent, however, to one of ordinary skill in the art having the benefit of this disclosure, that the present disclosure may be practiced without these specific details. In some instances, well-known structures and devices are shown in block diagram form, rather than in detail, in order to avoid obscuring the present disclosure.

0101 The components and other features described herein can be implemented as discrete hardware components or integrated in the functionality of hardware components such as ASICs, FPGAs, DSPs or similar devices. In addition, these components can be implemented as firmware or functional circuitry within hardware devices. Further, these components can be implemented in any combination of hardware devices and software components.

0102 Some portions of the detailed description have been presented in terms of algorithms and symbolic representations of operations on data bits within a computer memory. These algorithmic descriptions and representations are the means used by those skilled in the art to most effectively convey the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of steps leading to a desired result. The steps are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, or the like.

0103 It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise as apparent from the following discussion, it is appreciated that throughout the description, discussions utilizing terms such as "capturing," "performing," "causing," "encoding," "storing," "receiving," "assigning," "generating," "presenting," "transmitting," "obtaining," "displaying," "forming," or the like, may refer to the actions and processes of a machine, a computer system, or similar electronic computing device, that manipulates and transforms data represented as physical (e.g., electronic) quantities within the machine's registers and memories into other data similarly represented as physical quantities within the machine memories or registers or other such information storage, transmission or display devices.

0104 The words "example" or "exemplary" are used herein to mean serving as an example, instance, or illustration. Any aspect or design described herein as "example" or "exemplary" is not necessarily to be construed as preferred or advantageous over other aspects or designs. Rather, use of the words "example" or "exemplary" is intended to present concepts in a concrete fashion. As used in this application, the term "or" is intended to mean an inclusive "or" rather than an exclusive "or." That is, unless specified otherwise, or clear from context, "X includes A or B" is intended to mean any of the natural inclusive permutations. That is, if X includes A; X includes B; or X includes both A and B, then "X includes A or B" is satisfied under any of the foregoing instances. In addition, the articles "a" and "an" as used in this application and the appended claims should generally be construed to mean "one or more" unless specified otherwise or clear from context to be directed to a singular form.

0105 For simplicity of explanation, the methods have been depicted and described as a series of acts. However, acts in accordance with this disclosure can occur in various orders and/or concurrently, and with other acts not presented and described herein. Furthermore, not all illustrated acts may be required to implement the methods in accordance with the disclosed subject matter. In addition, those skilled in the art will understand and appreciate that the methods could alternatively be represented as a series of interrelated states via a state diagram or events. Additionally, it should be appreciated that the methods disclosed in this specification are capable of being stored on an article of manufacture (e.g., a computer readable storage medium) to facilitate transporting and transferring such methods to computing devices. The term article of manufacture, as used herein, is intended to encompass a computer program accessible from any computer-readable device or storage media.

0106 It is to be understood that the above description is intended to be illustrative, and not restrictive. Many other implementations will be apparent to those of skill in the art upon reading and understanding the above description. The scope of the disclosure should, therefore, be determined with reference to the appended claims, along with the full scope of equivalents to which such claims are entitled.

0107 Although the operations of the methods herein are shown and described in a particular order, the order of the operations of each method may be altered so that certain operations may be performed in an inverse order or so that certain operations may be performed, at least in part, concurrently with other operations. In another implementation, instructions or sub-operations of distinct operations may be in an intermittent and/or alternating manner.

0108 Reference throughout this disclosure to "one implementation," or "an implementation" means that a particular feature, structure, or characteristic described in connection with the implementation or implementation is included in at least one implementation. Thus, the appearances of the phrase "in one implementation," or "in an implementation" in various places throughout this specification are not necessarily all referring to the same implementation. Furthermore, the particular features, structures, or characteristics may be combined in any suitable manner in one or more implementations.

0109 Implementations also relate to an apparatus for performing the operations herein. This apparatus may be specially constructed for the required purposes or it may comprise a general purpose computer selectively activated or reconfigured by a computer program stored in the computer. Such a computer program may be stored in a computer readable storage medium, such as, but not limited to, any type of disk including floppy disks, optical disks, CD-ROMs and magnetic-optical disks, read-only memories (ROMs), random access memories (RAMs), EPROMs, EEPROMs, magnetic or optical cards, flash memory devices including uni-
universal serial bus (USB) storage devices (e.g., USB key devices) or any type of media suitable for storing electronic instructions, each of which may be coupled to a computing system bus.

[0110] The methods and systems described herein are used in a wide variety of implementations, including as part of a mobile application (“app”), and can be part of photo or video-related software including a mobile operating system. Apps installed on the mobile device can access the systems and methods via one or more application programming interface (API).

[0111] The algorithms and displays presented herein are not inherently related to any particular computing system or other apparatus. Various general purpose systems may be used with programs in accordance with the teachings herein or it may prove convenient to construct more specialized apparatus to perform the required method steps. The required structure for a variety of these systems will be apparent from the description above. In addition, the present disclosure is not described with reference to any particular programming language. It will be appreciated that a variety of programming languages may be used to implement the teachings of the disclosure as described herein.

What is claimed is:

1. A method comprising:
   receiving a media request to play a media item via a first user interface (UI) on a mobile device;
   causing a media player in a second UI to play the media item;
   displaying, by a processing device, a representation of the media item in a first portion of the first UI, the first portion of the first UI being associated with the media player in the second UI;
   receiving, via a second portion of the first UI, an activity request to perform an activity that is independent of the viewing of the media item; and
   presenting, by the processing device in the second UI, the representation of the media item played in the media player in the second UI.

2. The method of claim 1 wherein the first UI comprises a transport control to receive requests pertaining to playback of the media item in the media player in the second UI.

3. The method of claim 2 wherein the transport control includes at least one of: play, fast-forward, rewind, stop, or pause.

4. The method of claim 2 further comprising:
   receiving a user selection of the transport control; and
   in response to the received user selection, instructing a server to stream the media item for playback in the media player in the second UI.

5. The method of claim 1 wherein the representation of the media item is a thumbnail of a video.

6. The method of claim 1 wherein the second UI is presented on a user device that is different from the mobile device.

7. The method of claim 1 wherein the activity is messaging, and the content includes a plurality of user messages.

8. An apparatus comprising:
   a display device;
   a memory communicably coupled to the display device; and
   a processing device communicably coupled to the memory, the processing device to execute instructions to:
   receive a media request to play a media item via a first user interface (UI) on a mobile device;
   cause a media player in a second UI to play the media item;
   display a representation of the media item in a first portion of the first UI, the first portion of the first UI being associated with the media player in the second UI;
   receive, via a second portion of the first UI, an activity request to perform an activity that is independent of the viewing of the media item; and
   present content associated with the requested activity in the first UI while continuing to display, in the first portion of the first UI, the representation of the media item played in the media player in the second UI.

9. The apparatus of claim 8 wherein the first UI comprises a transport control to receive requests pertaining to playback of the media item in the media player in the second UI.

10. The apparatus of claim 9 wherein the processing device is further to execute instructions to:
    receive a user selection of the transport control; and
    in response to the received user selection, instruct a server to stream the media item for playback in the media player in the second UI.

11. The apparatus of claim 8 wherein the representation of the media item is a thumbnail of a video.

12. The apparatus of claim 8 wherein the second UI is presented on a user device that is different from the mobile device.

13. The apparatus of claim 8 wherein the activity is messaging, and the content includes a plurality of user messages.

14. A non-transitory machine-readable storage medium storing instructions which, when executed, cause a processing device to perform operations comprising:
   receiving a media request to play a media item via a first user interface (UI) on a mobile device;
   causing a media player in a second UI to play the media item;
   displaying, by a processing device, a representation of the media item in a first portion of the first UI, the first portion of the first UI being associated with the media player in the second UI;
   receiving, via a second portion of the first UI, an activity request to perform an activity that is independent of the viewing of the media item; and
   presenting, by the processing device in the second UI, the representation of the media item played in the media player in the second UI.

15. The non-transitory machine-readable storage medium of claim 14 wherein the first UI comprises a transport control to receive requests pertaining to playback of the media item in the media player in the second UI.

16. The non-transitory machine-readable storage medium of claim 15 wherein the transport control includes at least one of: play, fast-forward, rewind, stop, or pause.

17. The non-transitory machine-readable storage medium of claim 16 wherein the operations further comprising:
   receiving a user selection of the transport control; and
   in response to the received user selection, instructing a server to stream the media item for playback in the media player in the second UI.
18. The non-transitory machine-readable storage medium of claim 14, wherein the representation of the media item is a thumbnail of a video.

19. The non-transitory machine-readable storage medium of claim 14, wherein the second UI is presented on a user device that is different from the mobile device.

20. The non-transitory machine-readable storage medium of claim 14, wherein the activity is messaging, and the content includes a plurality of user messages.

* * * * *