

[MM2-NEH] App Technical Specifications v.4.0 July 2014

Center for Digital Archaeology - map2app

Document Purpose

Mukurtu Mobile Releases: 1.5 and 2.0

Final Product Technical Specifications

iPhone/iPad development

Android Smartphone/Tablet development

Login and Connection to Server

Content Upload

Media Handling: Images

Media Handling: Video (2.0)

Media Handling: Audio (2.0)

Document Purpose

This document is aimed at describing an explicit set of requirements to be satisfied by the Mukurtu Mobile app in its iPhone, iPad, Android(smartphone) and Android(tablet) versions, and relatively to its 1.5 and 2.0 Releases.

Although this is to be considered a working draft and edits can be proposed by either CoDA or map2app, the present document is to be considered by both parties as the most complete documentation of the product delivered to CoDA, with detailed reference to adopted technologies and differences amongst devices or OS.

Mukurtu Mobile Releases: 1.5 and 2.0

Mukurtu Mobile 1.5 is a minor update of MM 1.0 with respect to features. Mukurtu Mobile 1.0 was released in October 2012 and currently (Jan 2014) still [available on the Apple Store](#).

MM 1.5, the first release developed thanks to the NEH Grant (expected date: end of March 2014) will function exactly like the 1.0 version, except for minor adjusts to workflow and interface, and necessary bug fixes. MM 1.5 will be extended to Android and tablets including iPads.

The second release of the app for the MM@-NEH Project (Mukurtu Mobile 2.0, expected date: end of June 2014) will comprise a complete feature update of the app and workflow adjusts.

Final Product Technical Specifications

iPhone/iPad development

- The 1.5 and 2.0 iPhone releases of the app will be a native iOS update of the current (1.0) application with an iPad version.

A hybrid solution consisting in a HTML5 web app wrapped into a native container was evaluated and ruled out during the research phase in November 2013 because local resources such as the phone camera could not be reached through the wrapper.

- The 1.5 and 2.0 version will be available for any iPhone and iPad installing iOS7 and following.

Supported iPhone versions will include	Supported iPad versions will include
<ul style="list-style-type: none">• iPhone 4 (distributed since June 2010)• iPhone 4S• iPhone 5• iPhone 5C• iPhone 5S (or newest)	<ul style="list-style-type: none">• iPad 2 (distributed since March 2011)• iPad with Retina Display• iPad Air• iPad Mini• iPad Mini with Retina Display (or newest)

Android Smartphone/Tablet development

- The 1.5 and 2.0 Android releases of the app will be a native developed application that suits all screen ratios for phones and tablets. It will be developed to work exactly like the iPhone version.

Supported Android OS versions will include
<ul style="list-style-type: none">• Android 2.3.6 (distributed since September 2011) and all following versions on phone• Android 3.0 (distributed since February 2011) and following versions on tablet

- As of March 2014, versions of Android previous to 2.3.3 are estimated to cover about 1.3% of the Android market (source: developers.android.com). Android versions 2.3.3 - 2.3.7 (same SDK) alone cover 19% of the market.

Login and Connection to Server

Release 1.5

- Mukurtu Mobile needs to establish communication with a Mukurtu CMS powered site (server) under a working internet connection in order to allow retrieving of content (metadata) and content upload (stories created on mobile device).
- While the user is logged out, only the Settings view (plus Login Help content) is available. Settings preferences below the “Login” button will be greyed out.
- Content created and not uploaded is deleted when the user logs out. The user will receive a warning when explicitly logging off the app.
- ~~If user deselects the “Keep me logged in” flag in Settings login will only last until the app quits. A warning will be displayed on logout whenever possible.~~
- Settings for a user will be stored locally within the app and retrieved in case of a new login to the same account.
- Logout resets all settings to the default value.
- “Login”/“logout” status is stored locally. “Login” status will be maintained if during a valid login session:
 - phone turns off due to insufficient battery
 - phone OS crashes
 - app crashes
 - user quits the application
 - connection to server fails or user is working off-line

- sync/upload fail

Release 2.0

- Starting with the 2.0 release, the application will also manage user authentication and authorization to the external web service Youtube for video storage and rendering.
- Youtube API 3.0 uses OAuth2.0 to authenticate user requests. First login will take the user to a Google webview that will require login to the chosen Youtube channel.
- Logout resets all settings to the default value, including the Youtube login form, which is reset to logout mode.
- “Login”/”logout” status is stored locally. “Login” status will be maintained if during a valid login session:
 - phone turns off due to insufficient battery
 - phone OS crashes
 - app crashes
 - user quits the application
 - connection to server fails or user is working off-line
 - sync/upload/Youtube authentication fail

Content Upload

- Content Upload will happen through POST / GET queries posted to a REST web services that will use Drupal’s module Services. (Detailed description of functioning for MM1.0 in this document: [Mukurtu Mobile 1.0 | Communication to Mukurtu CMS Server](#))
- The Mukurtu CMS content types that Mukurtu Mobile is currently allowed to create are Digital Heritage Items and Files/Media items, plus keywords (‘Keywords’ taxonomy terms) that can be added directly by the user through the content creation interface.
- Warnings in case of failure or errors during upload are included in this document: [\[MM2-NEH\] Mukurtu Mobile Verbiage v.3.0 January 2014](#)
- Content that could not be uploaded due to invalid values in the metadata will be marked with a warning icon and will have to be edited before they can be uploaded.

Media Handling: Images

- The original file is always stored in the device’s gallery with the highest available quality (digital original) and EXIF data completed with geo-location when supported on the device. The derivative filename structure makes it easy to track back to the original file.
- Files created within the app are stored in a separate /Mukurtu/ folder in Android (iOS won’t allow this) and on both platforms they can be retrieved whether or not upload was successful.

- Full-res image is used for upload unless the user has unchecked this option in Settings. Otherwise a lower resolution file is created based on the chosen option and stored locally for upload to server.
- All available EXIF data are maintained (more correct to say: re-embedded) into the image that Mukurtu uploads. Some of them are displayed [on the destination site](#) through Mukurtu CMS, others can be viewed downloading the original file. Mukurtu's stories provide the user with narrative metadata fields such as keywords, description, category, and a title to be associated to a media collection.
- Such metadata refer to the single media node in Drupal, regardless of the digital heritage item(s) they are associated with.
- Uploaded image files are stored directly within the destination Mukurtu CMS server.
- Filenames of media imported from the gallery and processed through Mukurtu Mobile start with the app name and media type, followed by the original filename, which usually includes a timestamp of the file creation date and time:

[Mukurtu_Image_IMG_20140703_191643.jpg](#)

Files created within the Mukurtu app are named following the same naming convention, with a timestamp referring to the media creation date and time.

[Mukurtu_Audio_20140730_184300.mp4](#)

[Mukurtu_Image_20140816_032836.jpg](#)

NOTE: Filenames have to be assigned to all files from iOS devices, since the system **doesn't automatically assign a filename to each file**. Something like IMG_0001.jpg will be generated when you sync photos from your device on to your computer or a storage service - but the lack of a local friendly filename is the reason for all those photo.jpg, photo-1.jpg and so on that you get anytime you email images or upload them to your browser. Android devices also can have **alternative rules for assigning file names**, so that for example a picture is named 2014-09-06 16.09.48.jpg when you export or share it, but the same file becomes IMG_20140906_160948.jpg if you import it to your computer via USB.

Media Handling: Video (2.0)

- If user doesn't provide a valid Youtube account, selection from the gallery will only allow access to images; video recording option should be grayed out while using the app internal camera.
- Full-resolution video file is maintained in the device's gallery (MM 2.0 only supports importing videos from gallery, not capturing with the in-app camera).
- A web optimized version of the video is produced within the app as the video is associated to a DH item and stored locally for upload to Youtube.
- Uploaded video files are stored in the Youtube account provided by the user.
- Settings are available for the user to make the video unlisted (default) or public.
- A title is associated to the uploaded video based on DH item title and consecutive numbers (e.g. 'Historic China Camp Village 01').

Media Handling: Audio (2.0)

- Audio files are recorded using an internal component of the app.
- The original file is always stored in the device's gallery with the highest available quality (digital original) and EXIF data completed with geo-location when supported on the device. The derivative filename structure makes it easy to track back to the original file.
- Files created within the app are stored in a separate /Mukurtu/ folder in Android (iOS won't allow this) and on both platforms they can be retrieved whether or not upload was successful. Retrieving audio files from the device is not trivial (ie often they are not available for browsing like images) and could require specific software (such as iTunes for iOS or Android File Transfer for Android devices)
- Size/length limits are not set for the file.
- Audio files are stored locally within the app
- The following metadata are maintained in association with the audio file
 - Original filename (timestamp?)
 - Original creation date-time
 - Creator (mukurtu username)
 - Device info (model-make)
 - Geolocation
- Such metadata refer to the single media node in Drupal, regardless of the digital heritage item(s) they are associated with.
- Uploaded audio files are stored directly within the destination Mukurtu CMS server.
- Filenames of media imported from the gallery and processed through Mukurtu Mobile start with the app name and media type, followed by the original filename, which usually includes a timestamp of the file creation date and time:

[Mukurtu_Image_IMG_20140703_191643.jpg](#)

Files created within the Mukurtu app are named following the same naming convention, with a timestamp referring to the media creation date and time.

[Mukurtu_Audio_20140730_184300.mp4](#)

[Mukurtu_Image_20140816_032836.jpg](#)