Adapt Authoring Tool

Requirements

13.03.2020

prepared by Tom Taylor

Contents

Core

These requirements refer to the ‘core’ application bundle, and as such cover considerably more functionality that some of the other areas. The ‘core’ covers any ‘essential’ functionality that is used elsewhere in the application..

ID	Description	Priority	GitHub Issue
AAT001	Able to customise which modules are installed	M	
AAT002	Should be warned of any incompatibilities amongst installed modules (e.g. missing dependencies)	S	
AAT003	Able to easily write and include custom features/modules	M	
AAT004	Able to update the installed modules to include the latest changes	M	
AAT005	Able to automatically generate documentation from doc comments.	S	
AAT006	Able to generate documentation/guides from included files (preferably markdown format)	S	
AAT007	Able to run checks to verify that code conforms to the Adapt project's conventions	M	
AAT008	Able to run included unit tests	M	
AAT009	Developers should be able to write unit tests to be included in application test suite	M	
AAT010	Developers should be able to write performance tests for their code to be included in application test suite	S	
AAT011	Able to specify which APIs are essential for running the application, and require that these are implemented	M	
AAT012	Should create an abstract base to be used by other modules	S	

AAT013	Developers should be able to interrupt the application's execution at specific points (i.e. hooks)	S	
AAT014	Able to run an automated script to install the application	M	
AAT015	Able to run an automated script to update the application	M	

Assets

ID	Description	Priority	GitHub Issue
AM001	Able to upload a single asset file (should at least support image/audio/video mime types)	M	
AM002	Able to edit asset metadata when uploading asset file	M	
AM003	Able to query assets via API (to include filters & sorts)	M	
AM004	Able to update asset metadata via API	M	
AM005	Able to replace the file linked to an asset via API	S	
AM006	Able to permanently delete an asset via an API	M	
AM007	Able to query which courses use a specific asset via an API	M	
AM008	Able to query assets not used by any content	M	
AM009	Able to upload composite assets as a zip file	S	
AM010	Able to upload a zip file containing multiple assets	C	
AM011	Able to select folder/catalogue when uploading asset/s	C	
AM012	Able to catalogue assets using folders/collections (could, needs further defining and breaking down)	C	
AM013	Able to generate a small thumbnail image for a single asset (include videos?)	M	
AM014	Able to retrieve a single asset file via API	M	
AM015	Able to retrieve a single asset thumbnail file via API	M	
AM016	Able to view assets stored in multiple third-party locations (e.g. Google Drive, Dropbox, SVN)	S	

AM017	Developers should be able to modify assets before they're stored	S	
AM018	Developers should be able to modify assets before they're retrieved	S	

Auth

ID	Description	Priority	GitHub Issue
AUTH001	Able to authenticate users via extendable API	M	
AUTH002	Able to support multiple authentication mechanisms	M	
AUTH003	Able to restrict API endpoints according to a user's permissions	M	
AUTH004	Able to unrestrict API endpoints	M	
AUTH005	Able to restrict access to data according to user	M	
AUTH006	Able to apply specific permissions to users	M	
AUTH007	Able to apply predefined sets of permissions to users	M	
AUTH008	Able to organise users into access groups to share content	S	
AUTH009	Able to specify which content is accessible to each access group	S	
AUTH010	Able to share content with specific users	M	
AUTH011	Able to register new users	M	
AUTH012	Able to retrieve existing users	M	
AUTH013	Able to update existing users	M	
AUTH014	Able to remove existing users	M	
AUTH015	Able to log-in with username/password	M	
AUTH016	Passwords should be stored securely in the DB	M	
AUTH017	Able to reset passwords via email	M	

AUTH018	Able to give temporary access to content	S	
AUTH019	Able to temporarily disable existing users from accessing the system	M	
AUTH020	Able to authenticate with the API via token	M	
AUTH021	Able to generate auth tokens	M	
AUTH022	Able to verify auth tokens	M	
AUTH023	Able to revoke auth tokens	M	

Config

ID	Description	Priority	GitHub Issue
CFG001	Should be able to specify config files in a separate file	M	
CFG002	Config file should allow dynamic values	S	
CFG003	Developers should be able to specify expected config values	S	
CFG004	Developers should be able to specify default config values	S	
CFG005	Should validate actual config values against expected values and warn appropriately	S	
CFG006	Should be able to generate a new config file with default/required values	S	
CFG007	Should be able to specify the config file to be used at runtime	M	
CFG008	Config should absorb system environment variables for use in the app	M	

Content

ID	Description	Priority	GitHub Issue
CON001	Able to create new courses/course content	M	
CON002	Able to modify existing courses/course content	M	
CON003	Able to remove existing courses/course content	M	
CON004	Able to set the theme for an existing course	M	
CON005	Able to set the menu for an existing course	M	
CON006	Able to add an extension to an existing course	M	
CON007	Able to remove an extension from an existing course	M	
CON008	Able to view which plugins are installed	M	
CON009	Able to install new plugins to the application	M	
CON010	Able to remove installed plugins from the application	M	
CON011	Able to update installed plugins in the application	M	
CON012	Able to disable plugins in the application	M	
CON013	Able to add an installed plugin to every new course	M	
CON014	Able to generate a preview of a course	M	
CON015	Able to create a 'runnable' course package	M	
CON016	Able to download a 'runnable' course package	M	
CON017	Able to export the source code for a course in a format recognised by the framework tools	M	
CON018	Able to download a course export	M	

CON019	Developers should be able to modify course data during course build	S	
CON020	Developers should be able to modify course files during course build	S	

Lang

ID	Description	Priority	GitHub Issue
LAN001	Able to define language files with all string translations for a single language	M	
LAN002	Able to load appropriate language file based on server language	M	
LAN003	API able to send appropriate language strings based on user (browser) language	S	
LAN004	Able to provide localised user output/error messages	M	
LAN005	Developer should be able to specify which language strings are required by their code	M	
LAN006	Able to check for missing language strings	C	
LAN007	Developer able to provide data to localisation functions to provide more customised messages	M	

Logger

ID	Description	Priority	GitHub Issue
LOG001	Able to log message to the console output	M	
LOG002	Able to specify the category/severity of individual log messages	M	
LOG003	Able to specify an identifier for the location/caller of each message to aid with debugging	S	
LOG004	Able to show the timestamp of log message	M	
LOG005	Developers should be able to write extensions to log messages to alternative persistence mechanisms (e.g. DB)	S	
LOG006	Able to customise which messages are displayed based on category/severity	S	

MongoDB

ID	**Description**	**Priority**	**GitHub Issue**
MDB001	Able to connect to a MongoDB instance	M	
MDB002	Able to connect to multiple MongoDB instances	C	
MDB003	Able to query a MongoDB instance	M	
MDB004	Able to sort query results	M	
MDB005	Able to filter query results	M	
MDB006	Able to create new documents in a MongoDB instance	M	
MDB007	Able to update existing documents in a MongoDB instance	M	
MDB008	Able to remove existing documents from a MongoDB instance	M	
MDB009	Developers should be able to access the MongoDB API	C	

Server

ID	Description	Priority	GitHub Issue
SVR001	Able to create an HTTP server to listen for incoming requests on a specified port	M	
SVR002	Able to receive incoming requests	M	
SVR003	Able to send HTTP responses	M	
SVR004	Able to add request handlers to specific endpoints	M	
SVR005	Able to call arbitrary code prior to handling a request	M	
SVR006	Able to handle request errors appropriately	M	
SVR007	Able to limit the request rate	S	

Validation

ID	Description	Priority	GitHub Issue
VAL001	Able to define data expectations in a standard format	M	
VAL002	Able to verify a piece of data complies with the defined expectations	M	
VAL003	Able to provide suitable output/feedback for the user	M	
VAL004	Able to define arbitrary custom data definitions/schemas	M	
VAL005	Able to override existing data definitions/schemas	M	
VAL006	Able to extend existing data definitions/schemas	M	
VAL007	Able to modify data definitions on the fly	M	
VAL008	Able to populate validated data with any defined defaults	M	
VAL009	Able to migrate from old schema format to new format	C	