



DataUp Web Application Programmer's Guide



Copyright © 2013 by The Regents of the University of California. All rights reserved.

This manual is licensed under the Creative Commons Attribution-Share Alike 3.0 United States License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/3.0/us/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

Version History

Version	Date	Notes
1.0	2/15/13	Initial Draft
1.1	2/25/13	First release
1.2	5/3/13	Minor edits

About the DataUp Project

In today's technologically advanced world, the data generated by researchers is increasingly born digital and subject to intensive transformation and analyses before publication. The various file formats, software, and hardware required to succeed in the modern research landscape can become daunting, especially since education about digital data management has not kept pace with these technological advancements. There is a significant gap between the data management skills needed by modern researchers and their current abilities; the gap is more noticeable given the current increase in funder requirements for data management plans and requests by journal publishers to make supporting data publicly available alongside traditional research articles.

The DataUp project was born out of this need for seamless integration of data management into the researchers' current workflows. We recognized that the large majority of Earth, environmental, and ecological scientists use spreadsheets in the course of their data collection and organization. Rather than requiring they learn a new program, we decided to meet them where they already work: Microsoft® Excel and related spreadsheet software.

The goals of the DataUp project were to facilitate data management, sharing, and archiving for scientists. The resulting tools from the DataUp project are part of the Investigator Toolkit for DataONE, an NSF DataNet project building cyber-infrastructure that links together existing archives of ecological and environmental scientific data. The OuterCurve Foundation holds the code's copyright and DataUp is part of their Research Accelerators Gallery.

The DataUp project is run through the University of California Curation Center (UC3) at the California Digital Library, a division of the University of California Office of the President. Project funders were the Gordon and Betty Moore Foundation and Microsoft Research Connections.

Contents

1	Introduction.....	3
2	Web.BizLogic Folders.....	3
3	Web.Client Folders.....	9
4	Web.Common Folders.....	12
5	Web.DataLayer Folders.....	14
6	Additional References.....	16

1 Introduction

The DataUp Excel web application is a Model-View-Controller (MVC) based program that emphasizes reusable code and a separation of concerns. As with all MVC applications, the DataUp web application is divided into three components: controller, model, and view.

2 Web.BizLogic Folders

This section provides information for files, functions, or classes that are used in code of the Web.BizLogic folder.

Table 1: Extensions Folder

Name	Description
DataDetailExtension	Extension for DataDetail object that represents an uploaded file.
DocumentDetailExtension	Extension for DocumentDetail object that represents a document.
DocumentStoreExtension	Extension methods to GET/SET values for DocumentStore.
GetCellValue	Get the value of a cell, given a file name, sheet name, and address name. Retrieve a reference to the worksheet part, and then use its Worksheet property to get a reference to the cell whose address matches the address you've supplied.
GetMetadataDetails	Gets the metadata details.
GetSheetName	Find the sheet with the supplied name and then use that sheet object to retrieve a reference to the appropriate worksheet.
GetZipFileStream	Gets the Zip file stream.

Name	Description
GeneratePartContent	Generates the content of the Worksheet part.
GenrateTableDefinitionPartContent	Generates content of tableDefinitionPart1.
InsertCellInWorksheet	Use regular expressions to get the row number and column name. If the parameter wasn't well formed, this code will fail.
InsertMetadataWorksheet	Add a blank WorksheetPart. Generate ParameterMetaData, Content. Get a unique ID for the new worksheet. Append the new worksheet and associate it with the workbook.
InsertSharedStringItem	Insert a value into the shared string table, creating the table if necessary. Insert the string if it's not already there. Return the index of the string.
InsertWorksheet	Add a blank worksheet. Get a unique ID for the new worksheet. Append new worksheet and associate it with the workbook.
SetCellValue	If the string exists in the shared string table, get its index. If the string doesn't exist in the shared string table, add it and get the next index. Opens the document for editing.
SpreadsheetDocumentExtension	Extension methods for an Excel spreadsheet. Given a workbook document, and a sheet name, return the worksheet corresponding to the supplied name. Finds the sheet with supplied name, and then use that sheet object to retrieve a reference to the appropriate worksheet.
UpdateDefaultProperties	Get Values from default properties.
WorkbookPartExtension	Given the main workbook part, and a text value, insert the text into the shared string table. Create the table if necessary. If the value already exists, return its index. If it doesn't exist, insert it and return its new index.
WorksheetExtension	Given a Worksheet and an address (like "AZ254"), either return a cell reference, or create the cell reference and then return it.

Table 2: Helper Folder

Name	Description
AddCellStyleErrors	Adds the cell style errors.
CsvHelper	Class containing methods used to parse CSV files.
FixExcel	Contains methods that remove compatibility errors from Excel documents.
GenerateFileList	Generates file list.
GetCelltype	Gets the cell type.
GetColumnName	Gets the name of the column.
GetColumnNameFromIndex	Gets the Excel column name from the column index.

Name	Description
GetRowIndex	Gets the index of the row.
GetWorkSheetBounds	Gets the work sheet bounds.
MetadataHelper	Helper to read metadata tags from file.
RemoveAllObjects	Removes all objects.
RetrieveCellErrors	Retrieves the cell errors.
RetrieveCellValueErrors	Retrieves the cell value errors.
RetrieveChartErrors	Retrieves the chart errors.
RetrieveCommentErrors	Retrieves the comment errors.
RetrieveMergedCellsError	Retrieves the merged cell errors.
RetrieveMixedDataTypeErrors	Retrieves the mixed data type errors.
RetreivePicErrors	Retrieves the pic errors.
RetrieveShapeErrors	Retrieves the shape errors.
XLRemoveChart	XLs the remove chart.
XLRemoveColorCell	XLs the remove color cell.
XLRemoveComment	XLs the remove comment.
XLRemoveMergeCell	XLs the remove merge cell.
XLRemovePictures	XLs the remove pictures.
XLRemoveShape	XLs the remove shape.
ValidateCsvData	Validates the CSV data.
ValidateExcel	Contains methods to detect compatibility errors in uploaded Excel/CSV files.
ValidateGraphics	Validates the graphics.
ZipFiles	Creates Zip file.
ZipFileHelper	Helper for Zip functionality.

Table 3: Model Folder

Name	Description
AzureId	Gets or sets file Azure ID.
CanFix	Gets or sets a value indicating whether the error is fixable.
Citation	Gets or sets citation.
ContentId	Gets or sets content ID of the data.
ControlName	Gets or sets the name of the control.
CreateById	Gets or sets created by ID.
CreatedDateTime	Gets or sets created date time.
CreateFailureStatus	Create a failure operation status message.
CreateSuccessStatus	Create a success operation status message.
CustomErrorMessage	Gets or sets a value indicating whether a custom error message is sent from service or not.
CustomReturnValues	Gets or sets a custom return value from the operation.
Data	Gets or sets the data.
DataDetail	Class representing the model for files which are uploaded.
DataStream	Gets or sets file data stream.
Description	Gets or sets description of the content or error.
ErrorAddress	Gets or sets the address / unique identifier of the

Name	Description
	error.
ErrorMessage	Gets or sets the custom error message.
Exception	Gets or sets the exception, if any occurred.
Extension	Gets the extension of the file.
FileName	Gets or sets content ID of the data.
Id	Gets or sets ID.
Identifier	Gets or sets Identifier.
IdentifierDetail	Class to represent Identifier.
Metadata	Gets or sets metadata.
MetadataDetail	Class that represents metadata.
MetadataDetailList	Represents the metadata details.
MimeType	Gets or sets the content MIME type of the file.
OperationStatus	Class representing the status of the operation performed by the service method.
ParameterMetadataDetail	Class that represents parameter metadata.
PostFileDetail	Class to represent the details of the file posted to repository.
Recommendation	Gets or sets the recommendation for the error.
Repository	Represents the repository.
SheetId	Gets or sets the ID of the sheet.
SheetName	Gets or sets the name of the sheet.
SheetNameOnly	Gets or sets a value indicating whether this is only sheet name.
Size	Gets or sets file size.
Status	Gets or sets the status of the file.
Succeeded	Gets or sets a value indicating whether the operations succeeded or not.
Title	Gets or sets the filename without extension.
UploadDetails	Represents the uploaded file details.
User	Class that represents logged in user.

Table 4: Properties Folder

Name	Description
AssemblyInfo.cs	General information about an assembly is controlled through the following attributes: AssemblyTitle, AssemblyDescription, AssemblyConfiguration, AssemblyCompany, AssemblyProduct, AssemblyCopyright, and CLSCompliant. Change these attribute values to modify the information associated with an assembly.
Culture	Overrides the current thread's CurrentUICulture property for all resource lookups using this strongly typed resource class.
Resource.Designer.cs	Use caution when modifying this file. The code was generated by a tool and changes to this file may cause incorrect behavior and will be lost if the code is regenerated. Runtime version 4.0.30319.296.

Name	Description
ResourceManager	Returns the cached ResourceManager instance used by this class.
Resources	A strongly-typed resource class, for looking up localized strings, etc. This class was auto-generated by the StronglyTypedResourceBuilder class via a tool like ResGen or Visual Studio. To add or remove a member, edit the .ResX file then rerun ResGen with the str option, or rebuild the VS project.
UnknownErrorMessage	Looks up a localized string similar to " <i>Unable to perform the operation. Please try again later.</i> "

Table 5: Service References Folder

Name	Description
Reference.cs	Use caution when modifying this file. The code was generated by a tool and changes to this file may cause incorrect behavior and will be lost if the code is regenerated. Runtime version 4.0.30319.296.
RepositoryDataContext	Initialize a new RepositoryDataContext object.
ResolveNameFromType	Since the namespace configured for this service reference in Visual Studio is different from the one indicated in the server schema, use type-mappers to map between the two.
ResolveTypeFromName	Since the namespace configured for this service reference in Visual Studio is different from the one indicated in the server schema, use type-mappers to map between the two.

Table 6: Services Folder

Name	Description
BlobService	Initializes a new instance of the BlobService class.
GetFile	Gets the files from Azure which is identified by file ID.
CheckIf Exists	Checks the file from Azure which is identified by file name.
CreateCSVFromExcel	Creates the CSV file from Excel.
DownloadDocument	Downloads the document.
DownloadDocumentForPost	Downloads the document for post.
DocumentService	Class that provides functionality for uploaded documents.
GetCSVFileStream	Creates a CSV File Stream from uploaded Excel file stream.
GetDocuments	Gets the documents.
GetFileStream	Gets the file stream from the Document Store.

Name	Description
IBlobService	Interface representing the blob service methods. Also, needed for adding unit test cases.
IncrementColRef	Returns the next Column Name as it exists in Excel Sheet
RemoveAll	Use to fix all the errors.
RemoveSelected	Use to fix selected error type.
UploadFile	Uploads the file to the database/Azure.
ValidateCsv	Use to check errors found in CSV file.

3 Web.Client Folders

This section provides information for files, functions, or classes that are used in code of the Web.Client folder.

Table 7: Web.Client Folder

Name	Description
Dispose	Disposes of the object.
EndRequestHandler	End request handler. Note: This is a workaround since subscribing to <code>HttpContext.Current.ApplicationInstance.EndRequest</code> from <code>HttpContext.Current.ApplicationInstance.BeginRequest</code> does not work.
GetLogFilePath	Method to get the log file path. Gets the base directory, creates a file, searches the file below the current directory and if the file exists, returns the path.
GetService	Gets the resolved service from the unity container.
GetServices	Gets one or more resolved services for the unity container.
GetValue	Gets the value of the key from the store.
Init	Application initialization.
LogErrorToFile	Method to write the log file to the base directory.
ParentUnityContainer	Gets the parent unity container.
PerRequestLifetimeManager	When you register an object in the unity container, you will need to specify the life time of the object unless you want the default lifetime to be assigned.
RemoveValue	Removes the key from the store.
RegisterRepositories	Registers the required repositories, which is required for all the controllers.
RegisterServices	Registers the required services, which is required for all the controllers.
RegisterUnityContainer	Creates an instance of <code>UnityContainer</code> and registers the instances which needs to be injected to <code>Controllers/Views/Services</code> , etc.
SetValue	Stores the value of the key.
UnityContainer	Instance of unity container.
UnityDependencyResolver	Class representing the unity dependency resolver used for dependency injection. Initializes a new instance of the <code>UnityDependencyResolver</code> class.

Table 8: Controllers Folder

Name	Description
AddPostedFile	Adds the posted file.
AuthenticationController	Controller used for authentication.
CheckDuplicate	Checks the duplicate.

Name	Description
DemoController	Controller for the demo functionality.
Detail	Retrieves document details for the specified ID.
DocumentController	Controller that handles all the document functionality.
Download	Downloads document with the specified id.
FetchMetadata	Fetches the metadata.
General	Action handling the general error messages.
GetAgreement	Gets the user agreement for the repository.
GetIdentifier	Gets the identifier from the repository.
Index	Action handling on the run time errors.
List	Retrieves a list of documents.
RemoveAll	Removes all the compatibility errors from the document.
RemoveSelected	Removes the selected error type from document.
RepositoryController	Controller that provides functionality to access the repository services.
Post	Posts the specified document detail.
SaveDetail	Saves the document detail.

Table 9: Extensions Folder

Name	Description
DocumentDetailExtension	Extension for DocumentDetail object that represents a document.
DocumentDetailViewModelExtension	Class that represents the Extension for the DocumentDetail ViewModel.
PostFileDetailExtensions	Extension for the PostFileDetail object which represents the file to be posted to repository.

Table 10: Infrastructure Folder

Name	Description
GetUrlWithQueryString	Gets the URL with its query string representing this request.
HrdRequest	Constructs HrdRequest. Represents a home real discover request.
ToQueryString	Creates query string using the name value collection.

Table 11: Properties Folder

Name	Description
AssemblyInfo.cs	General information about an assembly is controlled through the following attributes: AssemblyTitle, AssemblyDescription, AssemblyConfiguration, AssemblyCompany, AssemblyProduct, AssemblyCopyright, AssemblyTrademark, and AssemblyCulture. Change these attribute values to

Name	Description
	modify the information associated with an assembly.
Culture	Overrides the current thread's CurrentUICulture property for all resource lookups using this strongly typed resource class.
GeneralErrorMessage	Looks up a localized string similar to <i>"An unexpected error has occurred. Please use your browser Back button and try again."</i>
Resources.Designer.cs	Use caution when modifying this file. The code was generated by a tool and changes to this file may cause incorrect behavior and will be lost if the code is regenerated. Runtime version 4.0.30319.296.
Resources	A strongly-typed resource class, for looking up localized strings. This class was auto-generated by StronglyTypedResourceBuilder via a tool like ResGen or VisualStudio. To add or remove a member, edit the.ResX file then rerun ResGen with the /str option or rebuild your VS project.
ResourceManager	Returns the cached ResourceManager instance used by this class.

4 Web.Common Folders

This section provides information for files, functions, or classes that are used in code of the Excel.Addin folder.

Table 12: Web.Common Folder

Name	Description
CitationContent	Citation field name.
ConfigReader	Helper class for reading a value from Application Configuration file.
ContainerName	Gets Container name for files in Azure.
Constants	Class representing all the constants used across the applications.
DataUpWebServiceLink	Gets DataUpWebServiceLink.
Dateformat	Date format (YYYY-MM-DD).
DefaultRetryPolicy	Gets Retry policy for Azure connections.
DefaultMimeType	Default Mime type for blobs (binary large objects).
DocumentStatus	0 = Uploaded 1 = Posted
IdentifierKeyName	Identifier field name.
GetConfigSetting	Helper method to read a value from config file. Returns user specified default value in case any failure reading the config value.
GetString	Helper method to read a value from the application config or cloud config file.
MetaDataSheetName	Metadata range name.
MetadataTableStyleName	Metadata table style name.
MetadataXmlPath	Gets MetadataXmlPath.
RetryCount	Gets the number of times we need to retry for sending mails.
StorageSettingName	Config setting name for getting the Azure connection string.
Timeout	Timeout for Azure retry policy.
ParaMetadataRangeName	Parameter metadata range name.
PathSeparator	Path separator for Azure folders.
UploadToAzure	Gets indicating whether we need to upload files to Azure or not.

Table 13: Extensions Folder

Name	Description
CheckNotNull	Checks whether the given parameters are null or not. In case of null, throws argument null exception.
DecodeFrom64	The extension method can be used to decode the base

Name	Description
	64 encoded string.
DeserializeXML	Gets the decoded version of the string.
EncodeTo64	The extension method creates a Base64 encoded string from a normal string.
EnumExtensions	Class having the extension methods needed for Enums.
GetXmlStream	Serializes the XML content and returns the content as a memory stream.
ObjectExtensions	Checks whether the given parameters are null or not. In case of null, throws argument null exception.
SerializeObject	Serialize objects.
SerializeToString	Serializes the XML content and returns the content as a string.
SerializerExtensions	Class having the extension methods needed for serialization.
StreamExtensions	Class having extension methods needed for stream.
ToEnum	Converts a string or an integer value to respective Enum.

Table 14: Properties Folder

Name	Description
AssemblyInfo.cs	General information about an assembly is controlled through the following attributes: AssemblyTitle, AssemblyDescription, AssemblyConfiguration, AssemblyCompany, AssemblyProduct, AssemblyCopyright, AssemblyTrademark, and AssemblyCulture. Change these attribute values to modify the information associated with an assembly.

5 Web.DataLayer Folders

This section provides information for files, functions, or classes that are used in code of the Web.DataLayer folder.

Table 15:Web.DataLayer Folder

Name	Description
Add	Adds the given entity to the DataUp database.
AddUser	Add a user to repository.
BlobDataRepository	Class representing the Blob Data Repository having methods for adding/retrieving files from Azure blob storage.
BlobProperties	Gets the content from Azure as a stream. Returns the name of the blob and the blob properties.
CheckIfExists	Checks a file in Azure.
CloudBlobClient	Gets the CloudBlobClient instance from lazyclient.
CloudBlobContainer	Used to retrieve the container reference identified by the container name.
ContainerURL	Gets the container URL.
DataUpDbContext	Gets or sets the DataUp database context
Delete	Deletes the given entity from the DataUp database.
DeleteBlob	Deletes the specified file pointed by the blob.
DeleteFile	Deletes a file from Azure.
DocumentStoreRepository	Entity representing document store repository.
GetAll	Gets all the Entities from the entities collection.
GetBlob	Gets the content of the blob from specified container.
GetBlobContent	Gets the blob content from Azure as a stream. Returns the name of the blob and the blob details.
GetItem	Gets the Entity which satisfies the given condition from the collection of entities and also loads the navigation property mentioned in Include.
GetItems	Gets the item count satisfying the given condition in the current database table. Gets multiple Entities which satisfies the given condition from the collection of entities.
GetItemsCount	Gets the items count satisfying the given condition in the current database table.
GetUser	Fetch user from repository.
Entities1.Designer.cs	Use caution when modifying this file. The code was generated by a tool and changes to this file may cause incorrect behavior and will be lost if the code is regenerated.
ExistsBlobContent	Checks if the blob (binary large object) content is present in Azure or not.
MoveBlob	Moves blob from source to destination.
ObjectSet	Gets the DbSet instance from the DataUp database

Name	Description
	context.
RepositoryBase	Abstract base repository class having the implementation for methods which are common to all the db entities.
SaveChanges	Saves the changes made in the data models to the DataUp database.
Update	Updates the given Entity in the DataUp database.
UploadBlobContent	Upload the blob content to Azure.
UploadFile	Uploads a file to Azure as a blob.
UserRepository	Class representing user repository

Table 16: Interfaces Folder

Name	Description
Add	Adds the given entity to the Layerscape database.
AddUser	Add user to repository.
CheckIfExists	Checks a file in Azure.
Delete	Deletes the given entity from the Layerscape database.
DeleteFile	Deletes a file from Azure.
GetAll	Gets all the entities from the entities collection.
GetBlobContent	Gets the blob content from Azure as a stream.
GetItem	Gets the entity that satisfies the given condition from the collection of entities and also loads the navigation property mentioned in Include.
GetItems	Gets multiple entities which satisfies the given condition from the collection of entities.
GetItemsCount	Gets the items count satisfying the given condition in the current database table.
GetUser	Fetch user from repository.
IBlobDataRepository	Interface representing the blob data repository methods. Also, needed for adding unit test cases.
IDocumentStoreRepository	Interface stating methods to be implemented for document store repository.
IRepositoryBase	Base repository interface stating the methods to be implemented by the domain object repositories.
IUserRepository	Base user repository stating methods which are to be implemented by domain classes.
Update	Updates the given Entity in the Layerscape database.
UploadFile	Uploads a file to Azure as a blob.
SaveChange	Saves the changes made in the data models to the Layerscape database.

Table 17: Models Folder

Name	Description
BlobDetails	Details of the blob.
BlobID	Gets or sets the blob id.
Data	Gets or sets the data the blob as a stream.
MimeType	Gets or sets the content MIME type

Table 18: Properties Folder

Name	Description
AssemblyInfo.cs	General information about an assembly is controlled through the following attributes: AssemblyTitle, AssemblyDescription, AssemblyConfiguration, AssemblyCompany, AssemblyProduct, AssemblyCopyright, AssemblyTrademark and AssemblyCulture. Change these attribute values to modify the information associated with an assembly.

6 Additional References

C# Coding Conventions (C# Programming Guide) -
[http://msdn.microsoft.com/en-us/library/ff926074\(v=VS.100\).aspx](http://msdn.microsoft.com/en-us/library/ff926074(v=VS.100).aspx)

Design Guidelines for Developing Class Libraries -
<http://msdn.microsoft.com/en-us/library/ms229042.aspx>