User and Hierarchy Import Guide

Standard CSV Update Tool

02.00, 27 Apr 2012

Contents

Introduction 2
Essential files 2
Optional files 2
Producing CSV files 3
File types 3
User files 3
Hierarchy files 3
Relationship files 3
File relationships 4
CSV Initial Import Set Checklist 5
User and hierarchy import 6
CSV file specifications 10
User Files 10
Students.csv 10
Teachers.csv 11
Parents.csv (optional) 12
Hierarchy Files 13
Levels.csv 13
Classes.csv 14
Groups.csv (optional) 14
Relationship files 15
Class_Students.csv 15
Class_Teachers.csv 16
Level_Classes.csv 17
Parent_Students.csv (optional) 18
Student_Groups.csv (optional) 18
Teacher_Groups.csv (optional) 19
Parent_Groups.csv (optional) 20
Level_Groups.csv (optional) 21
Future imports and updates 22
General 22
Move students from one class to another 22
Add new classes 22
Add new groups 22
Add new students 22
Advanced update options 23
Overwriting existing data 23

Introduction

This document describes importing users and a hierarchy into a Studywiz deployment using Comma Separated Value (CSV) files. The first section contains the import procedures and the second describes the structure of the CSV files. The third section describes future imports and updates of an existing deployment using CSV files.

The CSV specification in this document relates specifically to Studywiz systems using the Standard CSV Update tool. If you are creating a CSV set for the Consortium (Multi-School) CSV Update tool then please contact Studywiz Support for the *User and Hierarchy Import – Consortium CSV Update Tool* document (email support@studywiz.com).

There are seven essential files and up to seven optional files.

Essential files

The seven files listed below are essential. They are used to set up Students, Teachers, Levels and Classes and contain the minimum information needed for the correct operation of Studywiz.

User Files	Hierarchy Files	Relationship Files
Students.csv	**Levels.csv**	**Class_Students.csv**
Teachers.csv	**Classes.csv**	**Class_Teachers.csv**
		Level_Classes.csv

Optional files

If your Studywiz system contains parents and groups, the optional files below are used to set them up.

Parents

The ***Parents.csv*** file sets the parents' details, and the ***Parent_Students.csv*** file creates the relationships between the Parents and the Students.

User Files	Hierarchy Files	Relationship Files
Parents.csv	*not required for Parents*	**Parent_Students.csv**

Groups

The ***Groups.csv***, ***Student_Groups.csv***, ***Teacher_Groups.csv***, ***Parent_Groups.csv*** and ***Level_Groups.csv*** files are used to set up group names and membership.

User Files	Hierarchy Files	Relationship Files
not required for Groups	**Groups.csv**	**Student_Groups.csv**
		Teacher_Groups.csv
		Parent_Groups.csv
		Level_Groups.csv

Producing CSV files

The best way to produce the CSV files required for your Studywiz installation is to export them from your school's records management system. You could also construct them using a spreadsheet program such as Microsoft Excel or by using a combination of both methods. Whatever method you use to create the CSV files, it is essential that they adhere to the specifications in this document.

File types

User files

The user files define the users of the type specified in the file name:

- Students.csv
- Teachers.csv
- Parents.csv.

Hierarchy files

The hierarchy files define the levels, classes and groups:

- Levels.csv
- Classes.csv
- Groups.csv.

Relationship files

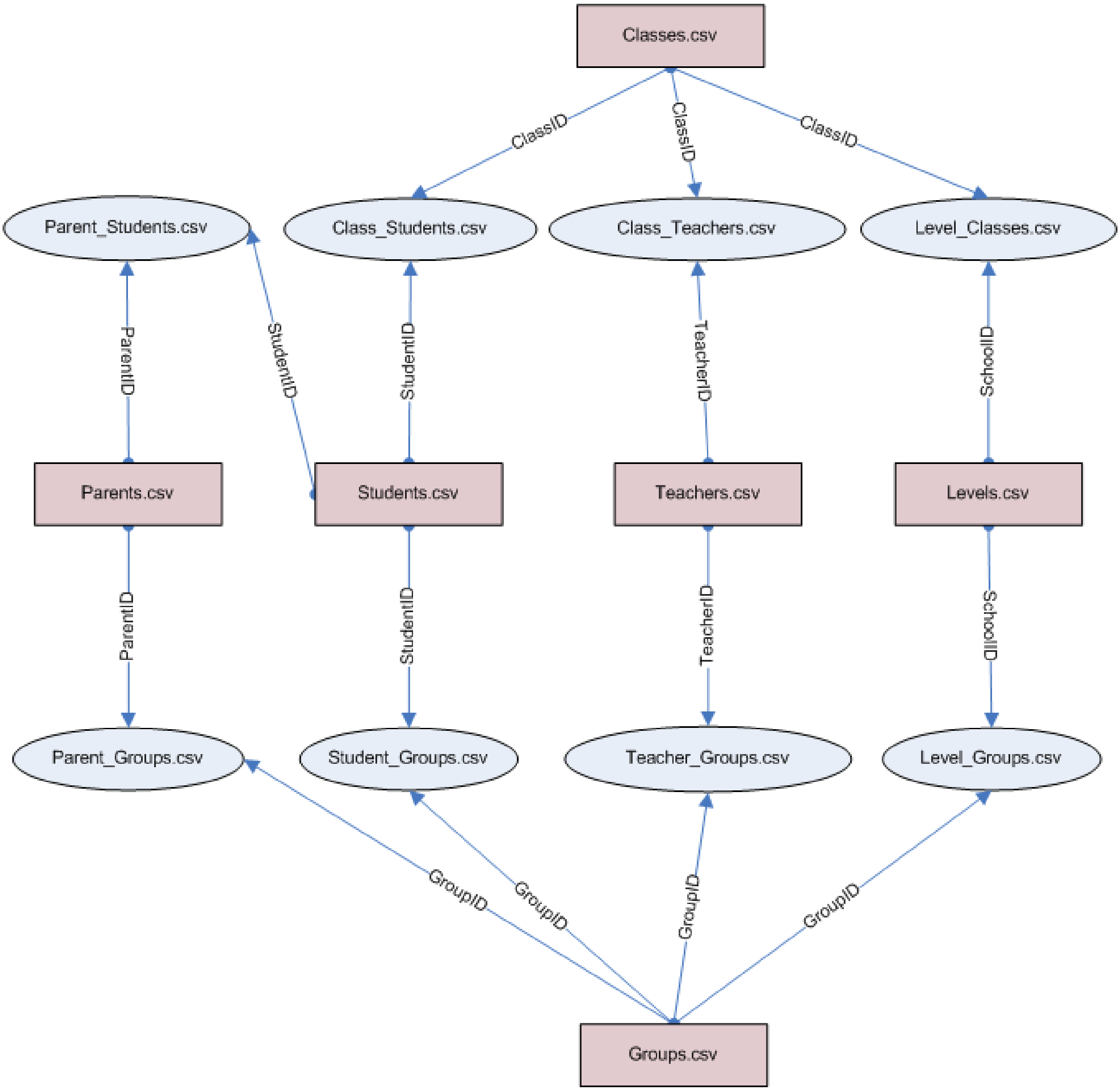
The relationship files (those with file names that are two words separated by an underscore) set up relationships between the two entities named in the file name:

- Class_Students.csv
- Class_Teachers.csv
- Level_Classes.csv
- Level_Groups.csv
- Parent_Students.csv
- Student_Groups.csv
- Teacher_Groups.csv
- Parent_Groups.csv.

Each relationship file contains the unique identifiers, as defined in the User and Hierarchy files, for each of the entities specified in the file name.

File relationships

The diagram below shows the User and Hierarchy files as red rectangles, and the Relationship files as blue ovals. The arrows show the unique identifiers from the user and hierarchy files that are used in each relationship file.



CSV Initial Import Set Checklist

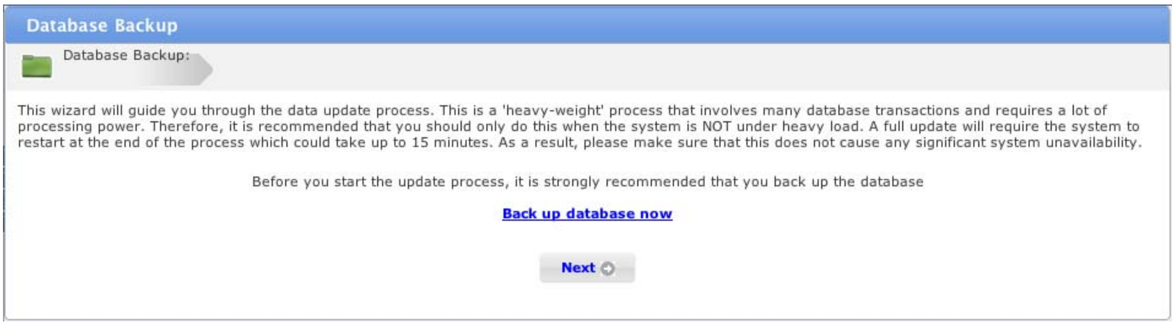
The table below lists the required and optional files for an initial CSV import:

CSV File Checklist	Required	Optional
Students.csv	✓	
Teachers.csv	✓	
Levels.csv	✓	
Classes.csv	✓	
Class_Students.csv	✓	
Class_Teachers.csv	✓	
Level_Classes.csv	✓	
Parents.csv		✓
Parent_Students.csv	required if *Parents.csv* exists.	✓
Groups.csv	required if any of the following exist: *Student_Groups.csv,* *Teacher_Groups.csv*, *Parent_Groups.csv*, or *Level_Groups.csv*.	✓
Student_Groups.csv		✓
Teacher_Groups.csv		✓
Parent_Groups.csv		✓
Level_Groups.csv		✓

User and hierarchy import

Follow the instructions in this section to use the Studywiz Standard CSV Updater to import CSV data files into Studywiz.

1. Ensure that you have CSV files that have been created according to the *CSV file specifications*, below.
2. Place the CSV files in a single ZIP file (with no sub-directories).
3. Log in to Studywiz as an Administrator.
4. From the **Admin Utilities** menu select **Update**. The Database Backup page appears.



Database Backup

Database Backup:

This wizard will guide you through the data update process. This is a 'heavy-weight' process that involves many database transactions and requires a lot of processing power. Therefore, it is recommended that you should only do this when the system is NOT under heavy load. A full update will require the system to restart at the end of the process which could take up to 15 minutes. As a result, please make sure that this does not cause any significant system unavailability.

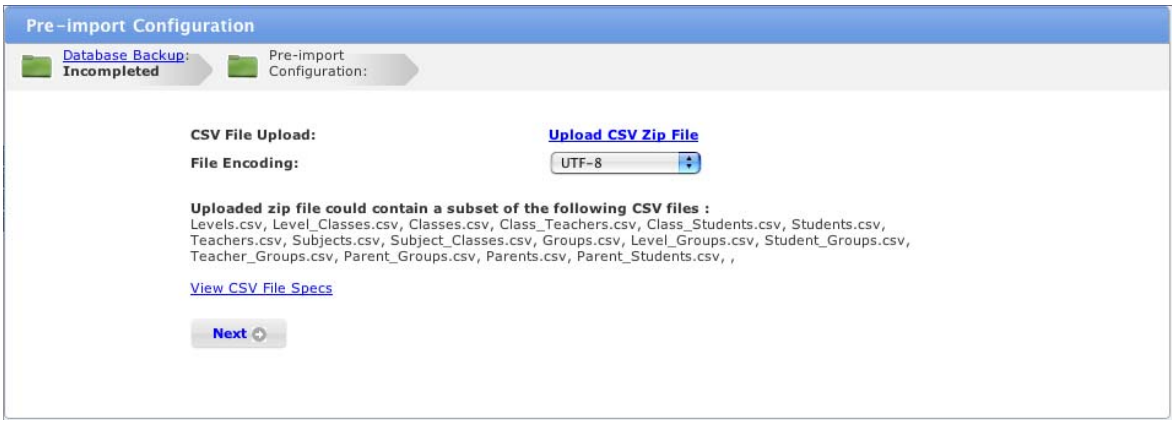
Before you start the update process, it is strongly recommended that you back up the database

5. Click **Back up database now**.

This step is very important as it will ensure that data can be restored should the update process make any undesired changes.

When the database backup is complete, the message ‘Database has been successfully backed up’ appears.

6. Click the **Next** button. The Pre-Import Configuration page appears.



Pre-import Configuration

Database Backup: Incompleted

Pre-import Configuration:

CSV File Upload: Upload CSV Zip File

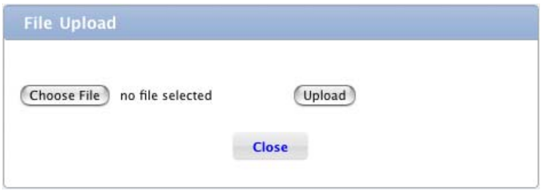
File Encoding: UTF-8

Uploaded zip file could contain a subset of the following CSV files :
Levels.csv, Level_Classes.csv, Classes.csv, Class_Teachers.csv, Class_Students.csv, Students.csv, Teachers.csv, Subjects.csv, Subject_Classes.csv, Groups.csv, Level_Groups.csv, Student_Groups.csv, Teacher_Groups.csv, Parent_Groups.csv, Parents.csv, Parent_Students.csv, ,

7. Click **Upload CSV Zip File**. The File Upload dialog box appears.

The file upload behaviour depends on the operating system you are using. Macintosh and PC upload procedures are shown below:

Macintosh

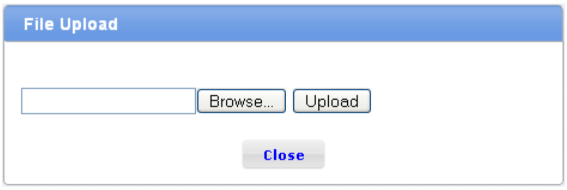


File Upload

no file selected

a. Click the **Choose File** button. The Finder

PC

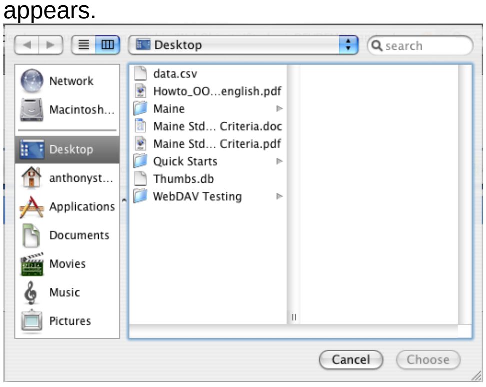


File Upload

a. Click the **Browse** button. The File Upload

Macintosh

appears.



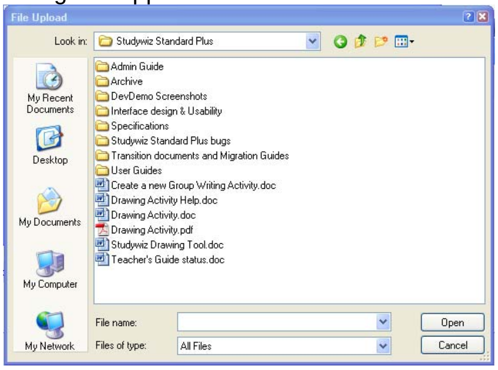
b. Navigate to the appropriate directory.

c. Select the ZIP file that contains the CSV data files.

d. Click the **Choose** button. The dialog box closes and the Select A File To Upload screen reappears.

PC

dialog box appears.

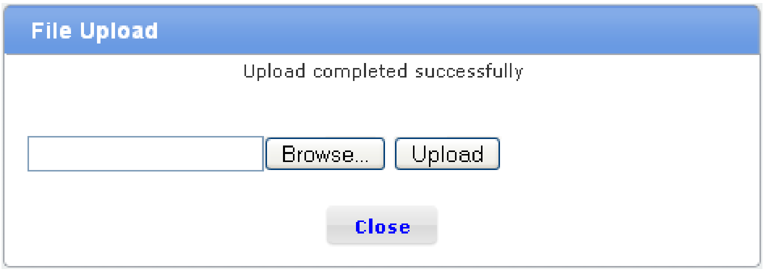


b. Navigate to the appropriate directory.

c. Select the ZIP file that contains the CSV data files.

d. Click the **Open** button. The dialog box closes and the Select A File To Upload screen reappears.

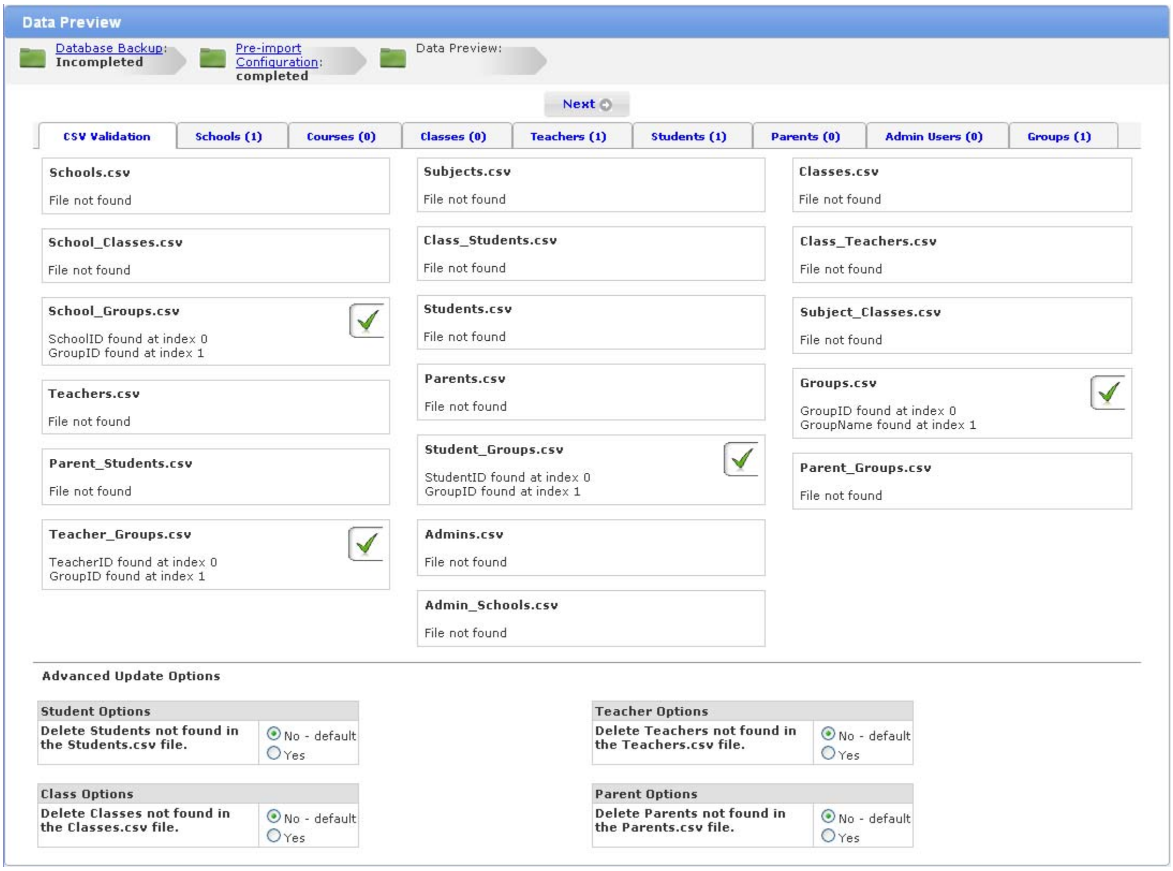
8. Click the **Upload** button. The file upload occurs and then the message ‘Upload completed successfully’ is displayed.



Upload completed successfully

9. Click the **Close** button. The Pre-Import Configuration screen appears.

10. Click the **Next** button. The system analyses the data for errors or inconsistencies and then the Data Preview page appears.



Data Preview

Database Backup: Incompleted | Pre-import Configuration: completed | Data Preview:

Next

Schools.csv
File not found

School_Classes.csv
File not found

School_Groups.csv ✓
SchoolID found at index 0
GroupID found at index 1

Teachers.csv
File not found

Parent_Students.csv
File not found

Teacher_Groups.csv ✓
TeacherID found at index 0
GroupID found at index 1

Subjects.csv
File not found

Class_Students.csv
File not found

Students.csv
File not found

Parents.csv
File not found

Student_Groups.csv ✓
StudentID found at index 0
GroupID found at index 1

Admins.csv
File not found

Admin_Schools.csv
File not found

Classes.csv
File not found

Class_Teachers.csv
File not found

Subject_Classes.csv
File not found

Groups.csv ✓
GroupID found at index 0
GroupName found at index 1

Parent_Groups.csv
File not found

Advanced Update Options

Student Options
Delete Students not found in the Students.csv file. ● No - default ○ Yes

Teacher Options
Delete Teachers not found in the Teachers.csv file. ● No - default ○ Yes

Class Options
Delete Classes not found in the Classes.csv file. ● No - default ○ Yes

Parent Options
Delete Parents not found in the Parents.csv file. ● No - default ○ Yes

The Data Preview page shows the results of the data analysis.

a. Click the **CSV Validation** tab to see if there are problems with the data format. Any critical errors will appear in red. If there are critical errors, the importer cannot proceed and it will be necessary to correct the data in accordance with the error message displayed.

b. Click the other tabs to list the changes that will be made to the hierarchy or users.

If a tab is highlighted in brown (accompanied by an exclamation mark), the system has detected a possible logic error or data-entry error. For example, the system may warn you that a level that is about to be created appears to be the same as an existing level, but with a different External ID.

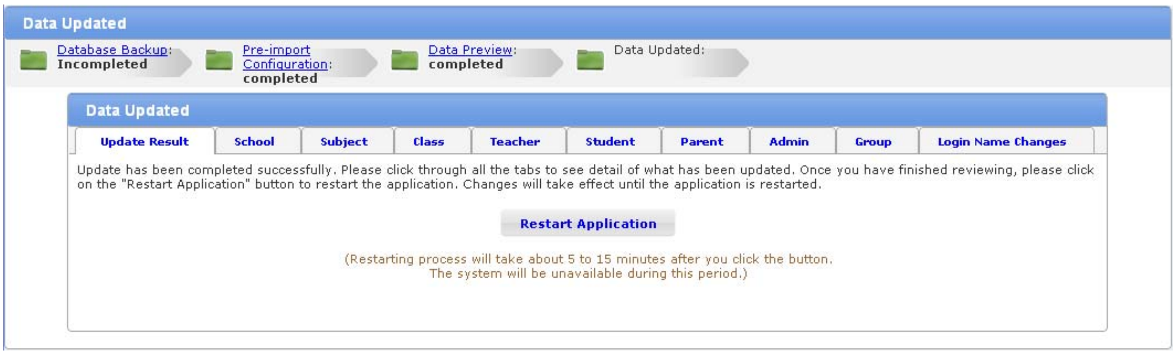
There are four **Advanced Update Options** available at the bottom of the Data Preview page. Select these options to delete Students, Teachers, Parents or Classes from the system if they are not defined in the CSV file.

If "Yes" is selected under an option, any entity of that type that is not contained in the selected CSV file will be deleted permanently from the system.

Use the advanced options with caution.

11. If there are no critical errors or warning messages, or if the warning messages are not applicable, click the **Next** button. The Update tool makes the required changes to the system and the update progress is displayed on the screen.

When the update is complete, the Data Updated screen appears.



Update has been completed successfully. Please click through all the tabs to see detail of what has been updated. Once you have finished reviewing, please click on the "Restart Application" button to restart the application. Changes will take effect until the application is restarted.

(Restarting process will take about 5 to 15 minutes after you click the button.
The system will be unavailable during this period.)

12. Click each of the tabs to see the system changes that have been made.

13. Click the **Restart Application** button. Studywiz restarts. This may take 10 to 15 minutes, during which time Studywiz will be unavailable. When Studywiz has restarted the changes will be visible.

CSV file specifications

User Files

This section describes the structure and contents of the three user files:

- ***Students.csv***
- ***Teachers.csv***
- ***Parents.csv*** (optional).

The user Files are used to create a profile for each user within the Studywiz system. Details of each file are provided in the sections below.

Students.csv

The **Students.csv** file contains the details of every Student in your Studywiz system.

File structure

The example below shows the first six columns of a **Students.csv** file for two students.

StudentID	FirstName	LastName	LoginName	Password	Email	…
S10002	John	Smith	John01	Smithy	john@email.com	…
S10003	Peter	Jones	Peter01	Jonesy	psmith@email.com	…

This file contains a **header row** and two **data rows**. A data row is included for each student in the system.

Header row

The first row of the file is called the **header row**. The header row has three **compulsory headers** and up to 12 **optional headers**.

The header row must contain the three compulsory headers; **StudentID**, **FirstName**, **LastName**.

The header row can contain up to 12 optional headers: **LoginName**, **Password**, **Email**, **DateOfBirth**, **WebsiteURL**, **FaxNumber**, **HomePhoneNumber**, **MobileNumber**, **WorkPhoneNumber**, **Address**, **Suburb**, **PostCode**.

- The header row must be present in each file for the import to function
- All headers used – both compulsory and optional – must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail
- Any of the optional headers can be left out of the files
- The headers may appear in any order.

⚠ If there are errors in the header rows, the data import will fail.

Data rows

The data rows contain student data. If you do not want to create any students, you can import a ***Students.csv*** file with no data rows; but the file **must** still contain the header row.

The **StudentID** entry for each student must be unique within the file. Use the school's ID system to fill in this column. If the school does not have an ID system then you will need to create a unique StudentID for each student.

Each student's StudentID should not change through their school life. This allows Studywiz to keep track of each user when updating data in such situations as a change in the school year or if a student changes classes. If it is not possible for unique identifiers to be maintained throughout a user's school life, contact Studywiz so that we can ensure that this is dealt with correctly.

- If the optional **LoginName** and **Password** fields are not filled, Studywiz will use the user's StudentID in its place.

- It is best if the User Files do not contain any non-alphanumeric characters
- URLs and email addresses may contain the characters @ and **/**.
- Do not use underscores (_) in any part of a CSV file. Underscores in the CSV will cause the import to fail.
- No user, of any type, can have the same LoginName as any other user.

Dependent files

A student cannot exist within Studywiz without being assigned to at least one class.

Every StudentID listed in the ***Students.csv*** file **must** also be listed in the ***Class_Students.csv*** file.

StudentID is used in the following files:

- ***Class_Students.csv***
- ***Parent_Students.csv***
- ***Student_Groups.csv***

Teachers.csv

The ***Teachers.csv*** file contains the details of every Teacher in your Studywiz system.

File structure

The example below shows the first six columns of a **Teachers.csv** file for two teachers.

TeacherID	FirstName	LastName	LoginName	Password	Email	…
T20002	Paul	Brown	Paul01	Browny	paulB@email.com	…
T20003	Mike	Green	Mike01	Greeny	MikeG@email.com	…

This file contains a **header row** and two **data rows**. A data row is included for each student in the system.

Header row

The first row of the file is called the **header row**. The header row has three **compulsory headers** and up to 12 **optional headers**.

The header row must contain the three compulsory headers; **TeacherID**, **FirstName**, **LastName**.

Students will see teachers' names as they are entered in the CSV file. If you don't want students to be able to see teacher's first names, use their first initial instead.

The header row can contain up to 12 optional headers: **LoginName**, **Password**, **Email**, **DateOfBirth**, **WebsiteURL**, **FaxNumber**, **HomePhoneNumber**, **MobileNumber**, **WorkPhoneNumber**, **Address**, **Suburb**, **PostCode**.

- The header row must be present in each file for the import to function
- All headers used – both compulsory and optional – must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail
- Any of the optional headers can be left out of the files
- The headers may appear in any order.

If there are errors in the header rows, the data import will fail.

Data rows

The data rows contain teacher data. If you do not want to create any teachers, you can import a ***Teachers.csv*** file with no data rows; but the file **must** still contain the header row.

The **TeacherID** entry for each teacher must be unique within the file. Use the school's ID system to fill in this column. If the school does not have an ID system then you will need to create a unique TeacherID for each teacher.

Each teacher's TeacherID should not change. This allows Studywiz to keep track of each user when updating data in such situations as a change in the school year or if a teacher changes classes. If it is not possible for unique identifiers to be maintained throughout a user's school life, contact Studywiz so that we can ensure that this is dealt with correctly.

- If the optional **LoginName** and **Password** fields are not filled, Studywiz will use the user's TeacherID in its place.
- It is best if the User Files do not contain any non-alphanumeric characters
- URLs and email addresses may contain the characters @ and **/**.
- Do not use underscores (_) in any part of a CSV file. Underscores in the CSV will cause the import to fail.
- No user, of any type, can have the same LoginName as any other user within the same school.

Dependent files

A teacher cannot exist within Studywiz without being assigned to at least one class.

Every TeacherID listed in the ***Teachers.csv*** file must also be listed in the ***Class_Teachers.csv*** file.

TeacherID is used in the following files:

- ***Class_Teachers.csv***
- ***Teacher_Groups.csv***

Parents.csv (optional)

The ***Parents.csv*** file contains the details of every Parent in your Studywiz system. If you don't want to create any Parents, this file does not have to be included.

File structure

The example below shows the first six columns of a ***Parent.csv*** file for two parents.

ParentID	FirstName	LastName	LoginName	Password	Email	…
P30002	Sam	Smith	Sam01	Sammy	SamS@email.com	…
P30003	Mary	Jones	Mary01	MaryJo	MaryJ@email.com	…

This file contains a **header row** and two **data rows**. A data row is included for each Parent in the system.

Header row

The first row of the file is called the **header row**. The header row has three **compulsory headers** and up to 12 **optional headers**.

The header row must contain the three compulsory headers; **ParentID**, **FirstName**, **LastName**.

The header row can contain up to 12 optional headers: **LoginName**, **Password**, **Email**, **DateOfBirth**, **WebsiteURL**, **FaxNumber**, **HomePhoneNumber**, **MobileNumber**, **WorkPhoneNumber**, **Address**, **Suburb**, **PostCode**.

- The header row must be present in each file for the import to function
- All headers used – both compulsory and optional – must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail
- Any of the optional headers can be left out of the files
- The headers may appear in any order.

⚠ If there are errors in the header rows, the data import will fail.

Data rows

The data rows contain parent data.

The **ParentID** entry for each parent must be unique within the file. Use the school's ID system to fill in this column. If the school does not have an ID system that includes parents, you will need to create a unique ParentID for each parent.

Each parent's ParentID should not change. This allows Studywiz to keep track of each user when updating. If it is not possible for unique identifiers to be maintained throughout a user's school life, contact Studywiz so that we can ensure that this is dealt with correctly.

- If the optional **LoginName** and **Password** fields are not filled, Studywiz will use the user's ParentID in its place.
- It is best if the file does not contain any non-alphanumeric characters.
- URLs and email addresses may contain the characters @ and ***/***.
- Do not use underscores (_) in any part of a CSV file. Underscores in the CSV will cause the import to fail.
- No user, of any type, can have the same LoginName as any other user within the same school.

Dependent files

A parent cannot exist within Studywiz without being assigned to at least one student.

Every ParentID listed in the ***Parents.csv*** file must also be listed in the ***Parent_Students.csv*** file.

ParentID is used in the following files:

- ***Parent_Students.csv***
- ***Parent_Groups.csv*.**

Hierarchy Files

This section describes the structure and contents of the three Hierarchy files:

- ***Levels.csv***
- ***Classes.csv***
- ***Groups.csv*** (optional).

The Hierarchy Files define the levels, classes and groups in the Studywiz deployment, and establish the relationships between them.

Details of each file are provided in the sections below.

Levels.csv

The *Levels.csv* file contains an ID number (**LevelID**) and a name (**LevelName**) for each level in the Studywiz deployment.

File structure

The example below shows a ***Levels.csv*** file for two levels.

LevelID	LevelName
YEAR7	Year 7
YEAR8	Year 8

This file contains a header row and two data rows. A data row is included for each level in the system.

Header row

The first row of the file is called the **header row**. The header row has two **compulsory headers**: **LevelID** and **LevelName**.

- The header row must be present in the file for the import to function

- The headers must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail

Data rows

Each data row in the file defines the ID number and name for each level in the deployment.

⚠ Underscores in the CSV will cause the import to fail.

The LevelD entry for each level must be a unique number within the file. Use the school's ID system to fill in this column. If there is no current ID system in place then you will need to create a unique identifier for each level.

LevelID is used in the following files:

- ***Level_Classes.csv***
- ***Level_Groups.csv*.**

Classes.csv

The ***Classes.csv*** file contains an ID number (**ClassID**) and a name (**ClassName**) for each class in the Studywiz deployment.

File structure

The example below shows a ***Classes.csv*** file for two classes.

ClassID	ClassName
ENG101	English 101
ENG201	English 201

This file contains a header row and two data rows. A data row is included for each class in the system.

Header row

The first row of the file is called the **header row**. The header row has two **compulsory headers**: **ClassID** and **ClassName**.

- The header row must be present in the file for the import to function
- The headers must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail.

Data rows

Each data row in the file defines the ID number and name for each class in the deployment.

⚠ Underscores in the CSV will cause the import to fail.

The ClassID entry for each class must be a unique number within the file. Use the school's ID system to fill in this column. If there is no current ID system in place then you will need to create a unique identifier for each class.

ClassID is used in the following files:

- ***Level_Classes.csv***
- ***Class_Students.csv***
- ***Class_Teachers.csv*.**

Groups.csv (optional)

The ***Groups.csv*** file contains an ID number (**GroupID**) and a name (**GroupName**) for each group in the Studywiz deployment. If you do not need to set up groups for your deployment you can omit this file.

File structure

The example below shows a ***Groups.csv*** file for two groups.

GroupID	GroupName
GR1004	Basketball Group
GR1005	Chess Club

This file contains a header row and two data rows. A data row is included for each group in the system.

Header row

The first row of the file is called the **header row**. The header row has two **compulsory headers**: **GroupID** and **GroupName**.

- The header row must be present in the file for the import to function
- The headers must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail

Data rows

Each data row in the file defines the ID number and name for each group in the deployment.

⚠ Underscores in the CSV will cause the import to fail.

The GroupID entry for each group must be a unique number within the file. Use the school's ID system to fill in this column. If there is no current ID system in place then you will need to create a unique identifier for each group.

GroupID is used in the following files:

- ***Level_Groups.csv***
- ***Teacher_Groups.csv***
- ***Student_Groups.csv***
- ***Parent_Groups.csv*.**

Relationship files

This section describes the structure and contents of the eight Relationship files:

- ***Class_Students.csv***
- ***Class_Teachers.csv***
- ***Level_Classes.csv***
- ***Parent_Students.csv*** (optional)
- ***Student_Groups.csv*** (optional)
- ***Teacher_Groups.csv*** (optional)
- ***Parent_Groups.csv*** (optional)
- ***Level_Groups.csv*** (optional).

The Relationship Files are used to create the links between the users and their classes, groups and levels.

Class_Students.csv

This file links each Student to their Classes.

- Every ClassID in this file must also be in the ***Classes.csv*** file.
- Every StudentID in this file must also be in the ***Students.csv*** file.

File structure

There are two possible formats for the ***Class_Students.csv*** file. Use the format that is easier to create from your school's records system.

Format 1

The first format has one **StudentID** column and one **ClassID** column. Each StudentID is repeated as many times as necessary in the StudentID column to assign students to multiple classes. For example:

StudentID	ClassID
S10002	ENG101
S10002	GEO101
S10003	ENG101
S10003	GEO201
…	…

Format 2

The second format has one **StudentID** column and multiple **ClassID** columns. As many columns are used as are needed to assign the student to multiple classes. For example:

StudentID	ClassID	ClassID	…
S10002	ENG101	GEO101	…
S10003	ENG101	GEO201	…

Header row

The ***Class_Students.csv*** file must contain a header row. The header row must contain the headers: **StudentID** and **ClassID**. It can contain one or many ClassID headers, depending on the format chosen.

- The header row must be present in the file for the import to function
- The headers must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail

Data rows

The data rows contain the StudentID and one or many ClassIDs, depending on the format chosen.

Class_Teachers.csv

This file links each Teacher to their Classes.

- Every ClassID in this file must also be in the ***Classes.csv*** file.
- Every TeacherID in this file must also be in the ***Teachers.csv*** file.

File structure

There are two possible formats for the ***Class_Teachers.csv*** file. Use the format that is easier to create from your school's records system.

Format 1

The first format has one **TeacherID** column and one **ClassID** column. Each TeacherID is repeated as many times as necessary in the TeacherID column to assign teachers to multiple classes. For example:

TeacherID	ClassID
T20002	ENG101
T20002	ENG102
T20003	GEO101
T20003	GEO201

…	…

Format 2

The second format has one **TeacherID** column and multiple **ClassID** columns. As many columns are used as are needed to assign the teacher to multiple classes. For example:

TeacherID	**ClassID**	**ClassID**	**…**
T20002	ENG101	ENG102	…
T20003	GEO101	GEO201	…

Header row

The ***Class_Teachers.csv*** file must contain a header row. The header row must contain the headers: **TeacherID** and **ClassID**. It can contain one or many ClassID headers, depending on the format chosen.

- The header row must be present in the file for the import to function
- The headers must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail

Data rows

Each data row contains one TeacherID and one or many ClassIDs, depending on the format chosen.

Level_Classes.csv

This file links each Class to a Level.

- Every LevelID in this file must also be in the ***Levels.csv*** file.
- Every ClassID in this file must also be in the ***Classes.csv*** file.

File structure

There are two possible structures for the ***Level_Classes.csv*** file. Use the structure that is easier to create from your school's records system.

Structure 1

The first format has one **LevelID** column and one **ClassID** column. Each ClassID is repeated as many times as necessary in the ClassID column to assign multiple classes to the same level. For example:

LevelID	**ClassID**
YEAR7	ENG101
YEAR7	GEO101
YEAR8	ENG201
YEAR8	GEO201
…	…

Structure 2

The second format has one **LevelID** column and multiple **ClassID** columns. As many columns are used as are needed to assign the classes to the level. For example:

LevelID	**ClassID**	**ClassID**	**…**
YEAR7	ENG101	GEO101	…
YEAR8	ENG201	GEO201	…

Header row

The ***Level_Classes.csv*** file must contain a header row. The header row must contain the headers: **LevelID** and **ClassID**. It can contain one or many ClassID headers, depending on the format chosen.

- The header row must be present in the file for the import to function

- The headers must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail

Data rows

Each data row contains one LevelID and one or many ClassIDs, depending on the format chosen.

Parent_Students.csv (optional)

This file links each Parent to one or more Students.

- Every ParentID in this file must also be in the ***Parents.csv*** file.
- Every StudentID in this file must also be in the ***Students.csv*** file.

File structure

There are two possible structures for the ***Parent_Students.csv*** file. Use the structure that is easier to create from your school's records system.

Structure 1

The first format has one **ParentID** column and one **StudentID** column. Each ParentID is repeated as many times as necessary in the ParentID column to assign parents to multiple students. For example:

ParentID	StudentID
P30002	S10002
P30002	S10003
P30003	S10004
P30003	S10005
…	…

Structure 2

The second format has one **ParentID** column and multiple **StudentID** columns. As many columns are used as are needed to assign the parent to multiple students. For example:

ParentID	StudentID	StudentID	…
P30002	S10002	S10003	…
P30003	S10004	S10005	…

Header row

The ***Parent_Students.csv*** file must contain a header row. The header row must contain the headers: **ParentID** and **StudentID**. It can contain one or many **StudentID** headers, depending on the format chosen.

- The header row must be present in the file for the import to function
- The headers must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail

Data rows

Each data row contains a ParentID and one or many StudentIDs, depending on the format chosen.

Student_Groups.csv (optional)

This file links each Student to their Groups.

- Every StudentID in this file must also be in the ***Students.csv*** file.
- Every GroupID in this file must also be in the ***Groups.csv*** file.

File structure

There are two possible structures for the ***Student_Groups.csv*** file. Use the structure that is easier to create from your school's records system.

Structure 1

The first format has one **StudentID** column and one **GroupID** column. Each StudentID is repeated as many times as necessary in the StudentID column to assign Students to multiple Groups. For example:

StudentID	GroupID
S10002	GR1004
S10002	GR1005
S10003	GR1006
S10003	GR1007
…	…

Structure 2

The second format has one **StudentID** column and multiple **GroupID** columns. As many columns are used as are needed to assign the Student to multiple Groups. For example:

StudentID	GroupID	GroupID	…
S10002	GR1004	GR1005	…
S10003	GR1006	GR1007	…

Header row

The ***Student_Groups.csv*** file must contain a header row. The header row must contain the headers: **StudentID** and **GroupID**. It can contain one or many **GroupID** headers, depending on the format chosen.

- The header row must be present in the file for the import to function
- The headers must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail

Data rows

Each data row contains a StudentID and one or many GroupIDs, depending on the format chosen.

Teacher_Groups.csv (optional)

This file links each Teacher to their Groups.

- Every TeacherID in this file must also be in the ***Teachers.csv*** file.
- Every GroupID in this file must also be in the ***Groups.csv*** file.

File structure

There are two possible structures for the ***Teacher_Groups.csv*** file. Use the structure that is easier to create from your school’s records system.

Structure 1

The first format has one **TeacherID** column and one **GroupID** column. Each TeacherID is repeated as many times as necessary in the TeacherID column to assign Teachers to multiple Groups. For example:

TeacherID	GroupID
T20002	GR1004
T20002	GR1005
T20003	GR1006
T20003	GR1007
…	…

Structure 2

The second format has one **TeacherID** column and multiple **GroupID** columns. As many columns are used as are needed to assign the Teacher to multiple Groups. For example:

TeacherID	GroupID	GroupID	…
T20002	GR1004	GR1005	…
T20003	GR1006	GR1007	…

Header row

The ***Teacher_Groups.csv*** file must contain a header row. The header row must contain the headers: **TeacherID** and **GroupID**. It can contain one or many **GroupID** headers, depending on the format chosen.

- The header row must be present in the file for the import to function
- The headers must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail

Data rows

Each data row contains a TeacherID and one or many GroupIDs, depending on the format chosen.

Parent_Groups.csv (optional)

This file links each Parent to their Groups.

- Every ParentID in this file must also be in the ***Parents.csv*** file.
- Every GroupID in this file must also be in the ***Groups.csv*** file.

File structure

There are two possible structures for the ***Parent_Groups.csv*** file. Use the structure that is easier to create from your school's records system.

Structure 1

The first format has one **ParentID** column and one **GroupID** column. Each ParentID is repeated as many times as necessary in the ParentID column to assign Parents to multiple Groups. For example:

ParentID	GroupID
P30002	GR1004
P30002	GR1005
P30003	GR1006
P30003	GR1007
…	…

Structure 2

The second format has one **ParentID** column and multiple **GroupID** columns. As many columns are used as are needed to assign the Parent to multiple Groups. For example:

ParentID	GroupID	GroupID	…
P30002	GR1004	GR1005	…
P30003	GR1006	GR1007	…

Header row

The ***Parent_Groups.csv*** file must contain a header row. The header row must contain the headers: **ParentID** and **GroupID**. It can contain one or many **GroupID** headers, depending on the format chosen.

- The header row must be present in the file for the import to function

- The headers must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail

Data rows

Each data row contains a ParentID and one or many GroupIDs, depending on the format chosen.

Level_Groups.csv (optional)

This file links each Group to a Level.

- Every GroupID in this file must also be in the ***Groups.csv*** file.
- Every LevelID in this file must also be in the ***Levels.csv*** file.

File structure

There are two possible structures for the ***Level_Groups.csv*** file. Use the structure that is easier to create from your school's records system.

Structure 1

The first format has one **LevelID** column and one **GroupID** column. Each LevelID is repeated as many times as necessary in the LevelID column to assign each group to multiple levels. For example:

LevelID	GroupID
YEAR7	GR1004
YEAR7	GR1005
YEAR8	GR1006
YEAR8	GR1007
…	…

Structure 2

The second format has one **LevelID** column and multiple **GroupID** columns. As many columns are used as are needed to assign multiple groups to the level. For example:

LevelID	GroupID	GroupID	…
YEAR7	GR1004	GR1005	…
YEAR8	GR1006	GR1007	…

Header row

The ***Level_Groups.csv*** file must contain a header row. The header row must contain the headers: **LevelID** and **GroupID**. It can contain one or many GroupID headers, depending on the format chosen.

- The header row must be present in the file for the import to function
- The headers must be exactly as specified, including capital and lower case letters. Any variation from this scheme will cause the import to fail

Data rows

Each data row contains one LevelID and one or many GroupIDs, depending on the format chosen.

Future imports and updates

General

Once a Studywiz implementation has been set up, it is possible to update the information in Studywiz at any time using a selection of CSV files.

- An update need not use all of the files that are required for the initial setup
- Each file need only contain data for the things you are changing. For example if you need to change the information for two students, you need only import data for those two students. All other students will remain unchanged.

Many update scenarios are possible using imports of selected CSV files. Four update scenarios are outlined in the sections below.

Move students from one class to another

If the students' details have not changed, you do not need to import a new ***Students.csv*** file.

1. Import a new ***Class_Students.csv***, containing only the information for the students whose classes have changed.

   This will remove the students from their old classes and add them to their new ones. It will not affect any other students or classes.

Add new classes

1. Import the following files:

   - ***Classes.csv*** containing only the new classes.
   - ***Level_Classes.csv*** containing information only for the classes that have changed.
   - ***Class_Students.csv*** containing information only for the new classes and their students.
   - ***Class_Teachers.csv*** containing information only for the new classes and their teachers

   Importing these files adds the new classes, assigns them to existing levels, and assigns existing teachers and students to the classes.

   This import will not affect existing classes (unless the **Delete Classes** option is set – see the *Advanced Update Options* section below for details).

Add new groups

1. Import the following files:

   - ***Groups.csv*** containing only the new groups.
   - ***Level_Groups.csv*** containing information only for the groups that have changed.
   - ***Student_Groups.csv*** containing information only for the groups and their students.
   - ***Teacher_Groups.csv*** containing information only for the groups and their teachers.
   - ***Parent_Groups.csv*** containing information only for the groups and their parents.

   Importing these files adds the new groups, assigns them to existing levels, and assigns existing teachers, students and parents to the groups.

   This import will not affect existing groups.

Add new students

1. Import the following files:

   - ***Students.csv*** containing only the new students
   - ***Class_Students.csv*** containing information only for the new students and their classes.
   - ***Student_Groups.csv*** containing information only for the new students and their groups.

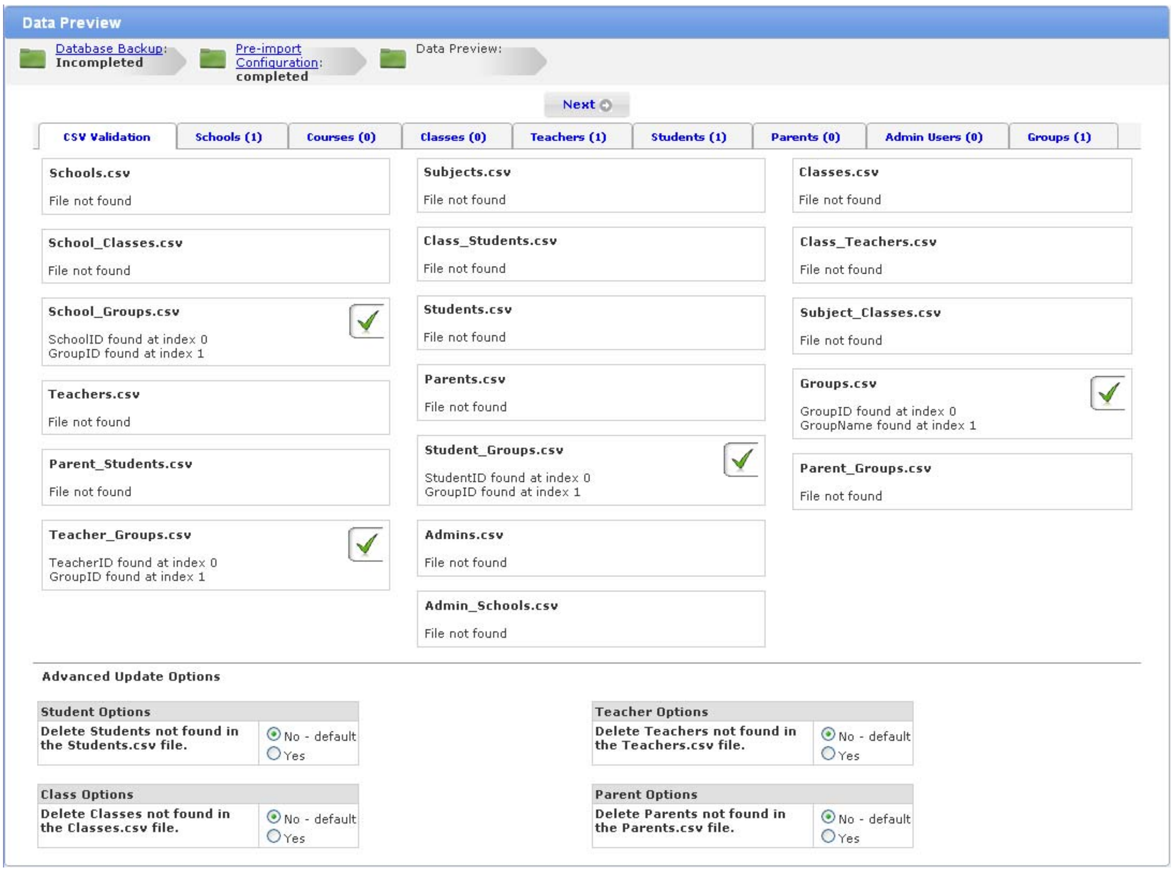
   Importing these files adds the new students to the system and assigns them to their classes and groups. This import does not alter existing students.

If the files include students who are already in the system, this import will overwrite those students' details.

If you select the **Delete Students not found in the Students.csv file** options, all other students will be overwritten.

Advanced update options

The **Advanced Update Options** allow you to have the importer replace data already in the system. The Advanced Update Options are at the bottom of the Data Preview page.



Data Preview

Database Backup: Incompleted

Pre-import Configuration: completed

Data Preview:

Next

Schools.csv
File not found

School_Classes.csv
File not found

School_Groups.csv ✓
SchoolID found at index 0
GroupID found at index 1

Teachers.csv
File not found

Parent_Students.csv
File not found

Teacher_Groups.csv ✓
TeacherID found at index 0
GroupID found at index 1

Subjects.csv
File not found

Class_Students.csv
File not found

Students.csv
File not found

Parents.csv
File not found

Student_Groups.csv ✓
StudentID found at index 0
GroupID found at index 1

Admins.csv
File not found

Admin_Schools.csv
File not found

Classes.csv
File not found

Class_Teachers.csv
File not found

Subject_Classes.csv
File not found

Groups.csv ✓
GroupID found at index 0
GroupName found at index 1

Parent_Groups.csv
File not found

Advanced Update Options

Student Options	
Delete Students not found in the Students.csv file.	● No - default ○ Yes

Teacher Options	
Delete Teachers not found in the Teachers.csv file.	● No - default ○ Yes

Class Options	
Delete Classes not found in the Classes.csv file.	● No - default ○ Yes

Parent Options	
Delete Parents not found in the Parents.csv file.	● No - default ○ Yes

Overwriting existing data

To completely erase student, class, teacher or parent data, and replace it with the data in the new files, select the appropriate option.