# GEORGIA SUPERIOR COURT CLERKS' COOPERATIVE AUTHORITY

# Civil Case eFiling Portal

## County Implementation Guide

**11/01/2018**

This document contains information that allows an interested vendor or county IT department to access and integrate with the GSCCCA Civil Case eFiling portal

# Recent Changes

**December 27, 2010:**

- Initial Version

**November 1, 2018:**

- Removed references to civil case augmentation related to civil data transmission

## Purpose

This technical document is intended for information technology personnel (programmers) who service those who wish to receive and process electronically submitted civil case filings from the Georgia Superior Court Clerks Cooperative Authority eFiling portal.  The document describes the interface for automated retrieval and processing of such electronic filings to Clerks of Superior Court through the GSCCCA Civil Case eFiling Portal.  This document does not address eFiling through the use of individual web browsers, nor the interface for filers to submit documents to the GSCCCA; such specifications are provided in other documents.

## Assumptions

This document assumes the reader has a working understanding of NIEM compliant XML documents and schemas.  More information about NIEM and the standard can be found at http://niem.gov/. .Furthermore, it is assumed that the reader has experience developing applications that can produce well-formed and schema compliant XML documents, parse XML documents, and consume web services based on a published WSDL.

## Account Creation

All counties and vendor systems who will retrieve and process filings from the GSCCCA web portal must first create a filer account.  Accounts can be created online by visiting http://efile.gsccca.org and selecting the "Register" link in the "Account Management Menu".  General information about the eFiling website and account creation is available in the GSCCCA Filer guide located at http://efile.gsccca.org/flrguide.aspx.  The filer account must be granted special permissions to allow certain actions such as accepting and rejecting filings.  To request elevated account permissions, please send a request to help@gsccca.org.

## Web Service URL

The GSCCCA Real Estate web service portal can be accessed at
https://eFile.gsccca.org/eFileWebService/Civil.asmx

The formal service description (WSDL) is available at:
https://eFile.gsccca.org/eFileWebService/Civil.asmx?WSDL

## Test Portal Web Service

The GSCCCA maintains a separate environment for processing test filings.  The system is a mirror of the production system but uses a separate database.  For this reason, users of the test portal must have a separate account created on the testing website.  The test server for account creation is located at

http://efiledev.gsccca.org.

The web service for submissions is located at:
http://efiledev.gsccca.org/eFileWebService/Civil.asmx.

The formal service description (WSDL) is available at:
http://eFiledev.gsccca.org/eFileWebService/Civil.asmx?WSDL

**PLEASE NOTE**: While the production system uses valid SSL certificates for security and encryption, the test system does not have a certificate issued by a Trusted CA. For this reason, access to the test portal is available via both HTTP and HTTPS while submissions to the production server must use HTTPS. Please be advised that many common web access APIs (such as those provided in .net) throw an exception by default if a HTTPS connection is attempted and the certificate cannot be validated.

## High Level Overview of the GSCCCA Civil Filing Schema

The GSCCCA has adopted a NEIM compliant XML schema to represent civil case filings. The current version of the GSCCCA schema is 1.0 and can be downloaded at:
http://efile.gsccca.org/downloads/gsccca-spec-1-0.zip. Schema compliant XML examples can be found at http://efile.gsccca.org/downloads/civilcaseexamples.zip. The schema provides for substantial reuse of existing NIEM structures and data types. Generally speaking, the schema contains annotations that should be helpful in determining the logical layout of the data. However, to help clarify certain aspects of the schema it will be useful to discuss some additional detail. The discussion that follows is intended to provide a high level overview of certain aspects, but is not intended to be an exhaustive study of the topic.

### Civil Case Type

The primary data type for conveying a civil case is the CivilCaseType complex type. The CivilCaseType is derived from nc:CaseType, which in turn derives from nc:ActivityType. The elements defined by nc:CaseType are largely reused, though most of the information inherited from nc:ActivityType is not currently used. A notable exception are the nc:ActivityDate and nc:ActivityDescriptionText elements, which are used to convey information about an initial court date under certain circumstances. The value of the nc:CaseTrackingID element is considered the unique ID for each Civil Case, and is the value that shoud be used to uniquely identify a case when performing actions against the GSCCCA portal.

### Filer Information

A CivilCase element will contain a Filer element of type FilingPartyType. This element will be used to communicate information about the individual who submitted the document to be filed through the GSCCCA portal. Every element of type FilingPartyType will contain a GSCCCAAccountGUID element that contains a guid that can be used to positively (and uniquely) identify a filer in the GSCCCA filer database. Other information about the filer is provided through inheritance of the nc:EntityType type . Additional information is also provided through a nc:ContactType element.

## Case Participants

Each person or organization that is associated with a case will be included in the xml in a CaseParticipant element.  Each case participant will be assigned an s:id Attribute that will be used for reference from other XML elements.  These ID's are unique only to the XML document and cannot be used for any other identification purposes.  Each CaseParticipant has a mandatory CaseParticipantRoleCode element that will contain a text identify that describes the role of the participant in the case.  Currently, the following roles are defined:

- PLAINTIFF
- DEFENDANT
- ATTORNEY
- WITNESS
- JUDGE

It should be noted that the Justice Domain within NIEM defines a j:CaseAugmentationType data Type that contains some strongly typed case parties.  A decision was made to exclude these strongly typed data structures because the names associated with the elements were not well suited to civil cases.  Also, the j:CaseAugmentationType defines a significant amount of other data fields that are not necessarily relevant for civil cases.  However, the use of a generic "CaseParticipant" as defined in the GSCCCA standard does not violate any NIEM constraints and is allowed as the underlying data structures are reused from structures defined elsewhere in NIEM.

## Case Officials

For every attorney participant in a case, the CivilCaseAugmentation element will also contain a CaseOfficial element.  The CaseOfficial element collects additional information for attorneys, such of their Bar number and state of issuance.   Each CaseOfficial element has a nc:RoleOfPersonReference element that is used to map the Case Official to a specific Case Participant.  The value of the s:ref

attribute of this element will be a s:id Reference, matching the s:id attribute value of a case participant. Additionally, each attorney may represent one or more parties, all of which are Case Participants.  For each Case Participant represented by a case official, there should be a CaseRepresentedPartyReference element, which the s:ref attribute value set to the s:id value of the associated case participant.

### Civil Documents

Each CivilCase element can contain zero or more CivilDocument elements, each of type CivilDocumentType.   The CivilDocumentType reuses many structures from the Niem-core namespace, though the specific implementation of each data structure may be unique for GSCCCA use.  Notably, each document must have an associated document binary, so the GSCCCA has elected to use the CivilDocument/nc:DocumentApplicationName/nc:applicationVersionText attribute to convey the mime type of the associated binary.  Other useful fields of the CivilDocument element are documented in the annotations.

## Web Services Provided

The following web methods are provided for the retrieval and processing of electronically submitted civil case filings.

- GetPendingFilings()
- GetPendingFiling()
- AcceptFiling()
- RejectFiling()

All methods return an XML string that represents a well-formed FilingResponse messages as defined in the GSCCCA Civil Case Filing schema.  Each FilingResponse will contain a status element that communicates the outcome of a web service invocation.  Status/Code element will contain a value of "OK" is the requested action was successfully performed.  However, if there are any issues that prevent an action from being performed, the Status/Code element will have a value of "Error", with additional information provided in the Status/Description Element.

The AcceptFiling() and RejectFiling() methods will return FilingResponse messages that will not contain CivilCase nodes.  The GetPendingFilings() method will return a FilingResponse message with one or more CivilCase nodes assuming the request was successfully performed.  However, the CivilCase elements will contain a subset of the data associated with the case, specifically only the CaseTrackingID, which can be used with the GetPendingFiling() method to retrieve the full data associated with a civil case filing.  For this reason, a successfully invocation of the GetPendingFiling() method will return a FilingResponse element with exactly one CivilCase element.

## GetPendingFilings METHOD

## Syntax

```
public string GetPendingFilings(string username, string password)
```

### Parameters

```
username
        Type: string
        The previously established userID of the eFiling User

password
        Type: string
        The previously established password of the eFiling User
```

### Return Value

```
Type: string
A XML string that represents a well-formed GSCCCA FilingResponse including zero or more
CivilCase documents.
```

## Remarks

1) The filer is identified by the user id and password provided with the method invocation.  The method invocation will retrieve the list of pending civil filings for the county associated with the logged in user.
2) The response will contain a CivilCase element for each filing that is pending.  The CivilCase element will be empty except for the nc:CaseTrackingID element which will contain the GSCCCA assigned ID for the filing.
3) Subsequent actions on the filing such as retrieving the package, accepting and or rejecting will require the GSCCCA ID supplied in the nc:CaseTrackingID element.

## GetPendingFiling METHOD

## Syntax

```
public string GetPendingFiling(string username, string password, string filingID)
```

### Parameters

```
username
        Type: string
        The previously established userID of the eFiling User
```

```
password
        Type: string
        The previously established password of the eFiling User

filingID
        Type: string
        The GSCCCA assigned filing identifier for the document being requested
```

*Return Value*

```
Type: string
A XML string that represents a well-formed GSCCCA FilingResponse including a single
CivilCase element, which may contain one or more CivilDocument elements.
```

# Remarks

1) The filer is identified by the user id and password provided with the method invocation.  The filing can only be obtained by  an account associated with the county in which the package was filed.
2) The Filing ID is the GSCCCA system generated identifier that was returned in the value of the nc:CaseTrackingID  element for a specific case filing.
3) The Filing ID must be a well formed GUID.  Supplying a nonconforming GUID will result in an error message being returned.
4) If the filing identified by the filing ID cannot be found, an error message is generated.
5) The requested document must be in the "Pending" state.  If the requested filing is not in the "Pending" state, an error message will be generated.
6) The response is guaranteed to contain a single CivilCase element if the request is successfully performed.

## AcceptFiling METHOD

# Syntax

```
public string AcceptFiling (string username, string password, string filingXML)
```

*Parameters*

```
username
        Type: string
        The previously established userID of the eFiling User
```

```
password
      Type: string
      The previously established password of the eFiling User

filingXML
      Type: string
      An xml string representing a well-formed CivilCase xml document.
```

*Return Value*

```
Type: string
A XML string that represents a well-formed GSCCCA FilingResponse including a single
Status element
```

# Remarks

1) The filer is identified by the user id and password provided with the method invocation.  The filing can only be Accepted by an account associated with the county in which the document was filed.
2) All calls to the AcceptFiling () method must  contain a well-formed, GSCCCA CivilCase xml document.
3) The CivilCase document submitted must contain a value in the nc:CaseTrackingID element, and this value must represent a well formed GUI value.
4) The value of the nc:CaseTrackingID must match a the case tracking ID of a previously submitted civil filing, and the filing must be in the "Pending" state (not already accepted or rejected).  If the civil filing referenced is not Pending, an error is generated.
5) The value of the nc:DocketID element must be set to the docket number assigned to the accepted case.
6) The CivilCase document submitted must contain at least one CivilDocument element or an error will be generated.
7) Generally, the binary images contained within each CivilDocument element should be the final "stamped" images of the original submitted documents.  The documents conveyed by an invocation to this method will be returned to the original filer.  No verification is performed by the GSCCCA to ensure a properly stamped, final document is contained in the CivilCase document submitted by the county.

## RejectFiling METHOD

## Syntax

```
public string RejectFiling(string username,
        string password, string filingGUID,
        string[] rejectionReasons)
```

### *Parameters*

```
username
        Type: string
        The previously established userID of the eFiling User

password
        Type: string
        The previously established password of the eFiling User

filingGUID
        Type: string
        The GSCCCA assigned filing identifier for the package being rejected

rejectionReasons
        Type: string[]
        A list of reasons the filing could not be recorded. At least one reason must be
        specified
```

### *Return Value*

```
Type: string
A XML string that represents a well-formed GSCCCA FilingResponse conveying the status of
the rejection request.
```

## Remarks

1) The filer is identified by the user id and password provided with the method invocation.  The filing can only be rejected by an account associated with the county in which the document was filed.
2) The Filing GUID is the GSCCCA system generated identifier that was returned in the value of the nc:CaseTrackingID  element when the case was retrieved with the GetPendingFiling() method.
3) The Filing GUID must be well formed.  Supplying a nonconforming GUID will result in an error message being returned.
4) If the filing identified by the filing GUID cannot be found, an error message is generated.
5) The requested filing must be in the "Pending" state.  If the requested filing is not in the "Pending" state, an error message will be generated.
6) At least one reason must be provided as to why the filing could not be recorded.

## Exception Handling and Service Interruptions

Care has been given in the implementation of all web methods to handle most predictable errors and system exceptions.  Generally speaking, business rule violations are intercepted and reported back as errors within the FilingResponse / Status element.  System errors on the GSCCCA server, such as temporary service interruption, are expected to be very rare.  However, system errors due tend to occur from time to time and where such errors can be anticipated, the web methods will intercept such exceptions and report the issue in the FilingResponse / Status element of the response.  If the system error prevented the successful execution of an action, the status message may indicate that a call to GSCCCA technical support will be necessary.  ALL ERRORS that can be anticipated are logged in the GSCCCA database when they occur and GSCCCA technical support staff is alerted.

It is unlikely that an exception will be raised from the web method to the consuming application unless there is a basic access with the method call itself.  However, each method does not include a catch-all error handler and it is possible, though unlikely, that an unhandled exception will be raised from the web method itself.  It is recommended that your application consider this in its design, and that the GSCCCA technical support department is notified whenever such errors occur.

Generally, if a system error occurs that prevents an action from being performed, it is safe to simply retry the transaction.