

## WISEdata Ed-Fi Integration – SIS Certification Test Plan Details

In order for Wisconsin DPI to completely validate SIS vendors:

- Vendors will record data/property values that will be needed to validate.
- Vendors will be required to take screenshots or a video clips when applicable using tools like Jing (<https://www.techsmith.com/jing.html>).
- Procedure for video / screen-shot capture as well as parameters

### AUTHENTICATION AND AUTH TESTS

1. Provide a JING video for each test and provide the test case # in the video name. Naming convention for videos **SISvendor+testcasenumber+date** (ccyyymmdd). Example: SISName3621520151021

### DESCRIPTOR AND TYPE TESTS

1. Provide screen shot of values appearing in SIS for selection. Provide the test case # in the screen shot. **Same naming convention as used for videos.**

### RESOURCE TESTS

1. Provide JING video of one GET and one POST for each resource below that correlates to a certification test. Provide the test case # in the video name.
2. For each test record the parameters used for the resource as follows:

/student: provide the studentUniqueId

/studentSchoolAssociation: provide the studentUniqueId, schoolId and entryDate

/studentSchoolAssociationWisconsinExt: provide the studentUniqueId, schoolId and entryDate

/studentAcademicRecord: provide the studentUniqueId, educationOrganizationId, schoolYear, termDescriptor

- Upload all test documents, videos and screen shots to our Accellion upload site (you'll be issued an account following your pre-certification meeting).
- Once Wisconsin DPI staff finishes validating results, the SIS vendor will be notified.
  - If an error occurs, or inaccurate information lands in the database, WDC staff will work with the vendor to resolve the issue and continue on with the certification.
  - If validation finds no issues, the SIS vendor will be flagged as eligible for production credentials.

## Test plan **37901: SIS\_Cert**

SIS Certification test week of 11/2/15.

Status: Active | Active dates: 11/2/2015 12:00:00 AM - 11/6/2015 12:00:00 AM | Area: Collections | Iteration: Collections | Build: <Not assigned> | Build definition: <Not assigned> | Build quality: <Not assigned>

### Available Configurations (1)

Config 35: SIS

### Test Settings

### Suite Hierarchy

- SIS\_Cert (0)
  - Authentication & Auth Tests (0)
    - Agency / SIS Relationship (2)
    - Resource authorization (10)
  - Descriptors and Types Tests (16)
  - Resources Tests (0)
    - /student (17)
      - /studentAcademicRecord (6)
      - /studentSchoolAssociation (18)
      - /studentSchoolAssociationWisconsinExtension (14)

## Suite Details

You can group your test cases together by organizing test cases into a test suite hierarchy in your test plan. By grouping your test cases together, when you want to run all these tests, you can select this test suite.

### Suite 37993: Agency / SIS Relationship

State: In progress

#### Test Cases (2)

#### Test Case **37911: SIS with no credentials for district**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

##### Test Steps (1)

#	Title	Expected Value
1	Attempt to do a GET for a /student resource not associated to the district your credentials have been issued for	Should see a 403 Forbidden message returned with message = 'The request can not be authorized because the education organizations identifiers assigned to the claim did not exist in the underlying ODS.'

#### Test Case **37912: SIS with credentials for district**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

##### Test Steps (1)

#	Title	Expected Value
1	Attempt to do a GET for a /student resource associated with the district your credentials have been issued for.	Should see a 200 OK message returned with all properties in the /student resource.

### Suite 37994: Resource authorization


State: In progress

## Test Cases (10)

### Test Case [37913](#): SIS attempt to READ a resource not authorized

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


#### Test Steps (1)

#	Title	Expected Value
 1	Attempt a GET on resource = /gradingPeriods	403 response code; "Access to the resource could not be authorized. Are you missing a claim? This resource can be authorized by: . ClaimsPrincipal has the following claims: http://ed-fi.org/ods/identity/claims/domains/edFiTypes, http://ed-fi.org/ods/identity/claims/domains/systemDescriptors, http://ed-fi.org/ods/identity/claims/domains/managedDescriptors, http://ed-fi.org/ods/identity/claims/domains/identity, http://ed-fi.org/ods/identity/claims/assessmentFamily, http://ed-fi.org/ods/identity/claims/educationContent, http://ed-fi.org/ods/identity/claims/studentAcademicRecord, http://ed-fi.org/ods/identity/claims/studentSchoolAssociationWisconsinExt, http://ed-fi.org/ods/identity/claims/student, http://ed-fi.org/ods/identity/claims/studentSchoolAssociation, http://ed-fi.org/ods/identity/claims/school, http://ed-fi.org/ods/identity/claims/localEducationAgency, http://ed-fi.org/ods/identity/claims/namespacePrefix"

### Test Case [37914](#): SIS attempt to POST /PUT to a resource not authorized

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


#### Test Steps (1)

#	Title	Expected Value
 1	Attempt a POST or PUT to /classPeriod	403 response code; "Access to the resource could not be authorized. Are you missing a claim? This resource can be authorized by: . ClaimsPrincipal has the following claims: http://ed-fi.org/ods/identity/claims/domains/edFiTypes, http://ed-fi.org/ods/identity/claims/domains/systemDescriptors, http://ed-fi.org/ods/identity/claims/domains/managedDescriptors, http://ed-fi.org/ods/identity/claims/domains/identity, http://ed-fi.org/ods/identity/claims/assessmentFamily, http://ed-fi.org/ods/identity/claims/educationContent, http://ed-fi.org/ods/identity/claims/studentAcademicRecord, http://ed-fi.org/ods/identity/claims/studentSchoolAssociationWisconsinExt, http://ed-fi.org/ods/identity/claims/student, http://ed-fi.org/ods/identity/claims/studentSchoolAssociation, http://ed-fi.org/ods/identity/claims/school, http://ed-fi.org/ods/identity/claims/localEducationAgency, http://ed-fi.org/ods/identity/claims/namespacePrefix" }

### Test Case [37915](#): SIS attempt to DELETE a resource not authorized

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


#### Test Steps (1)

#	Title	Expected Value
 1	Attempt a DELETE on a /school resource record	403 response code; "Access to the resource could not be authorized. Are you missing a claim? This resource can be authorized by: . ClaimsPrincipal has the following claims: http://ed-fi.org/ods/identity/claims/domains/edFiTypes, http://ed-fi.org/ods/identity/claims/domains/systemDescriptors, http://ed-fi.org/ods/identity/claims/domains/managedDescriptors, http://ed-fi.org/ods/identity/claims/domains/identity, http://ed-fi.org/ods/identity/claims/assessmentFamily, http://ed-fi.org/ods/identity/claims/educationContent, http://ed-fi.org/ods/identity/claims/studentAcademicRecord, http://ed-fi.org/ods/identity/claims/studentSchoolAssociationWisconsinExt, http://ed-fi.org/ods/identity/claims/student, http://ed-fi.org/ods/identity/claims/studentSchoolAssociation, http://ed-fi.org/ods/identity/claims/school, http://ed-fi.org/ods/identity/claims/localEducationAgency, http://ed-fi.org/ods/identity/claims/namespacePrefix" }

**Test Case 37916: SIS attempt to POST / PUT to a descriptor**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
 1	Attempt to POST / PUT on gradeLevelDescriptor	403 response code; "Access to the resource could not be authorized. Are you missing a claim? This resource can be authorized by: . ClaimsPrincipal has the following claims: http://ed-fi.org/ods/identity/claims/domains/edFiTypes, http://ed-fi.org/ods/identity/claims/domains/systemDescriptors, http://ed-fi.org/ods/identity/claims/domains/managedDescriptors, http://ed-fi.org/ods/identity/claims/domains/identity, http://ed-fi.org/ods/identity/claims/assessmentFamily, http://ed-fi.org/ods/identity/claims/educationContent, http://ed-fi.org/ods/identity/claims/studentAcademicRecord, http://ed-fi.org/ods/identity/claims/studentSchoolAssociationWisconsinExt, http://ed-fi.org/ods/identity/claims/student, http://ed-fi.org/ods/identity/claims/studentSchoolAssociation, http://ed-fi.org/ods/identity/claims/school, http://ed-fi.org/ods/identity/claims/localEducationAgency, http://ed-fi.org/ods/identity/claims/namespacePrefix" }

**Test Case 37917: SIS attempt to DELETE a descriptor**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


Test Steps (1)

#	Title	Expected Value
 1	Attempt a DELETE on gradeLevelDescriptor	405 response code; "The requested resource does not support http method 'DELETE'."

**Test Case 37918: SIS attempt to POST to a type**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
 1	Attempt a POST / PUT on diplomaTypes	403 response code; "Access to the resource could not be authorized. Are you missing a claim? This resource can be authorized by: . ClaimsPrincipal has the following claims: http://ed-fi.org/ods/identity/claims/domains/edFiTypes, http://ed-fi.org/ods/identity/claims/domains/systemDescriptors, http://ed-fi.org/ods/identity/claims/domains/managedDescriptors, http://ed-fi.org/ods/identity/claims/domains/identity, http://ed-fi.org/ods/identity/claims/assessmentFamily, http://ed-fi.org/ods/identity/claims/educationContent, http://ed-fi.org/ods/identity/claims/studentAcademicRecord, http://ed-fi.org/ods/identity/claims/studentSchoolAssociationWisconsinExt, http://ed-fi.org/ods/identity/claims/student, http://ed-fi.org/ods/identity/claims/studentSchoolAssociation, http://ed-fi.org/ods/identity/claims/school, http://ed-fi.org/ods/identity/claims/localEducationAgency, http://ed-fi.org/ods/identity/claims/namespacePrefix" }

**Test Case 37919: SIS attempt to DELETE a type**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
 1	Attempt a DELETE on diplomaTypes	405 response code; "The requested resource does not support http method 'DELETE'."

**Test Case 37980: SIS attempt READ/GET on resource authorized**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
1	Attempt GET on resource = /schools	200 response code; returned with school data

**Test Case 37981: SIS attempt POST to a resource authorized**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
1	Attempt a POST to a /student already created and associated with the district associated with the credentials you are using. Update the firstName	200 response code; firstName is updated as expected (can do a GET to verify)

**Test Case 37982: SIS attempt to DELETE a resource authorized**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
1	Delete a /studentSchoolAssociation for a /student	200 response code; success

## **Suite 37991: Descriptors and Types Tests**

State: In progress

### **Test Cases (16)**

**Test Case 37920: Validate correct values for sexTypes**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (2)

#	Title	Expected Value
1	Execute a GET on on the Descriptor/Type and validate the values returned.	"codeValue": "F", "description": "Female", "shortDescription": "Female" "codeValue": "M", "description": "Male", "shortDescription": "Male" "codeValue": "O", "description": "Other", "shortDescription": "Other"
2	Compare values returned from API match selectable / mapped values in SIS	All values returned in API should be values selectable / mapped in SIS

**Test Case 37921: Validate correct values for schoolFoodServicesEligibilityDescriptor**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (2)



#	Title	Expected Value
1	Execute a GET on on the Descriptor/Type and validate the values returned.	Values should be as follows: [ { "codeValue": "01", "shortDescription": "Free", "description": "Free", "schoolFoodServicesEligibilityType": "Free", }, { "codeValue": "02", "shortDescription": "Full price", "description": "Full price", "schoolFoodServicesEligibilityType": "Full price", }, { "codeValue": "03", "shortDescription": "Reduced price", "description": "Reduced price", "schoolFoodServicesEligibilityType": "Reduced price", }, { "codeValue": "04", "shortDescription": "Unknown", "description": "Unknown", "schoolFoodServicesEligibilityType": "Unknown", } ]
2	Compare values returned from API match selectable / mapped values in SIS	All values returned in API should be values selectable / mapped in SIS

**Test Case 37922: Validate correct values for studentCharacteristicDescriptor**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (2)



#	Title	Expected Value
---	-------	----------------

	1	Execute a GET on on the Descriptor/Type and validate the values returned.	Only value returned should be codeValue = "SATA", shortDescription, description and studentCharacteristicType = "State Assessment Type - Alternate"
	2	Compare values returned from API match selectable / mapped values in SIS	Only value of SATA should be selectable for initial release.

### **Test Case 37923: Validate correct values for limitedEnglishProficiencyDescriptors**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:



#### Test Steps (2)

#	Title	Expected Value
	1	Execute a GET on on the Descriptor/Type and validate the values returned.
		"codeValue": "1", "shortDescription": "ELL/LEP Beginning Preproduction.", "description": "ELL/LEP Beginning Preproduction. (WIDA label: Entering)" "codeValue": "2", "shortDescription": "ELL/LEP Beginning Production.", "description": "ELL/LEP Beginning Production. (WIDA label: Beginning)" "codeValue": "3", "shortDescription": "ELL/LEP Intermediate.", "description": "ELL/LEP Intermediate. (WIDA label: Developing)" "codeValue": "4", "shortDescription": "ELL/LEP Advanced Intermediate.", "description": "ELL/LEP Advanced Intermediate. (WIDA label: Expanding)" "codeValue": "5", "shortDescription": "ELL/LEP Advanced.", "description": "ELL/LEP Advanced. (WIDA label: Bridging)" "codeValue": "6", "shortDescription": "Formerly ELL/LEP, now fully English proficient", "description": "Formerly ELL/LEP, now fully English proficient" "codeValue": "7", "shortDescription": "Fully English proficient, never ELL/LEP", "description": "Fully English proficient, never ELL/LEP"
	2	Compare values returned from API match selectable / mapped values in SIS
		All values returned in API should be values selectable / mapped in SIS

### **Test Case 37924: Validate correct values for raceTypes**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


#### Test Steps (2)

#	Title	Expected Value
	1	Execute a GET on on the Descriptor/Type and validate the values returned.
		[ "codeValue": "", "description": "American Indian - Alaskan Native", "shortDescription": "American Indian - Alaskan Native", }, { "codeValue": "", "description": "Asian", "shortDescription": "Asian", }, { "codeValue": "", "description": "Black - African American", "shortDescription": "Black - African American", }, { "codeValue": "", "description": "Native Hawaiian - Pacific Islander", "shortDescription": "Native Hawaiian - Pacific Islander", }, { "codeValue": "", "description": "White", "shortDescription": "White", } ]
	2	Compare values returned from API match selectable / mapped values in SIS
		All values returned in API should be values selectable / mapped in SIS

### **Test Case 37925: Validate correct values for disabilityDescriptors**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (2)

#	Title	Expected Value
	1	Execute a GET on on the Descriptor/Type and validate the values returned.
		Values should be as follows: [ { "codeValue": "A", "shortDescription": "Autism", "description": "Autism", "disabilityCategoryType": "Individuals with Disabilities Education Act", "disabilityType": "Autism", }, { "codeValue": "DB", "shortDescription": "Deaf-Blind", "description": "Deaf-Blind", "disabilityCategoryType": "Individuals with Disabilities Education Act", "disabilityType": "Deaf-Blind", }, { "codeValue": "EBD", "shortDescription": "Emotional Behavioral Disability", "description": "Emotional Behavioral Disability", "disabilityCategoryType": "Individuals with Disabilities Education Act", "disabilityType": "Emotional Behavioral Disability", }, { "codeValue": "H", "shortDescription": "Hearing Impairment", "description": "Hearing Impairment", "disabilityCategoryType": "Individuals with Disabilities Education Act", "disabilityType": "Hearing Impairment", }, { "codeValue": "LD", "shortDescription": "Specific Learning Disability", "description": "Specific Learning Disability", "disabilityCategoryType": "Individuals with Disabilities Education Act", "disabilityType": "Specific Learning Disability", } ]

Education Act", "disabilityType": "Specific Learning Disability", }, {  
 "codeValue": "N", "shortDescription": "Not IDEA Eligible or No  
 Disability", "description": "Not IDEA Eligible or No Disability",  
 "disabilityCategoryType": "Individuals with Disabilities Education  
 Act", "disabilityType": "Not IDEA Eligible or No Disability", }, {  
 "codeValue": "OHI", "shortDescription": "Other Health  
 Impairment", "description": "Other Health Impairment",  
 "disabilityCategoryType": "Individuals with Disabilities Education  
 Act", "disabilityType": "Other Health Impairment", }, { "codeValue":  
 "OI", "shortDescription": "Orthopedic Impairment", "description":  
 "Orthopedic Impairment", "disabilityCategoryType": "Individuals  
 with Disabilities Education Act", "disabilityType": "Orthopedic  
 Impairment", }, { "codeValue": "SDD", "shortDescription":  
 "Significant Developmental Delay", "description": "Significant  
 Developmental Delay", "disabilityCategoryType": "Individuals with  
 Disabilities Education Act", "disabilityType": "Significant  
 Developmental Delay", }, { "codeValue": "SL", "shortDescription":  
 "Speech or Language Impairment", "description": "Speech or  
 Language Impairment", "disabilityCategoryType": "Individuals with  
 Disabilities Education Act", "disabilityType": "Speech or Language  
 Impairment", }, { "codeValue": "TBI", "shortDescription": "Traumatic  
 Brain Injury", "description": "Traumatic Brain Injury",  
 "disabilityCategoryType": "Individuals with Disabilities Education  
 Act", "disabilityType": "Traumatic Brain Injury", }, { "codeValue":  
 "V", "shortDescription": "Visual Impairment", "description": "Visual  
 Impairment", "disabilityCategoryType": "Individuals with  
 Disabilities Education Act", "disabilityType": "Visual Impairment", },  
 { "codeValue": "ID", "shortDescription": "Intellectual Disability",  
 "description": "Intellectual Disability", "disabilityCategoryType":  
 "Intellectual Disability", "disabilityType": "Intellectual Disability", } ]



2

Compare values returned from API match selectable / mapped values in SIS

All values returned in API should be values selectable / mapped in SIS

### Test Case **37926**: Validate correct values for entryTypeDescriptors

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (2)

#	Title	Expected Value
1	Execute a GET on on the Descriptor/Type and validate the values returned.	<pre>&lt;FlowDocument xmlns:space="preserve" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"&gt;&lt;Paragraph&gt;Values should be as follows: &lt;Run FontFamily="inherit"&gt;[&lt;/Run&gt;&lt;/Paragraph&gt;&lt;Section FontFamily="droid sans" FontSize="14.4px"&gt;&lt;Section FontFamily="anonymous pro" TextAlignment="Left" FontSize="0.9em" Margin="0,5px,0,0" Padding="10px" BorderThickness="1px"&gt;&lt;Paragraph FontFamily="anonymous pro" FontSize="0.9em"&gt;&lt;Span FontFamily="inherit"&gt;{ "&lt;Run FontFamily="inherit"&gt;codeValue&lt;/Run&gt;": &lt;Span FontFamily="inherit"&gt;&lt;Run FontFamily="inherit"&gt;"01"&lt;/Run&gt;&lt;/Span&gt;, "&lt;Run FontFamily="inherit"&gt;shortDescription&lt;/Run&gt;": &lt;Span FontFamily="inherit"&gt;&lt;Run FontFamily="inherit"&gt;"Transfer"&lt;/Run&gt;&lt;/Span&gt;, "&lt;Run FontFamily="inherit"&gt;description&lt;/Run&gt;": &lt;Span FontFamily="inherit"&gt;&lt;Run FontFamily="inherit"&gt;"Transfer"&lt;/Run&gt;&lt;/Span&gt;, "&lt;Run FontFamily="inherit"&gt;entryType&lt;/Run&gt;": &lt;Span FontFamily="inherit"&gt;&lt;Run FontFamily="inherit"&gt;"Transfer"&lt;/Run&gt;&lt;/Span&gt;, &lt;Run FontFamily="inherit"&gt;}&lt;/Run&gt;, { "&lt;Run FontFamily="inherit"&gt;codeValue&lt;/Run&gt;": &lt;Span FontFamily="inherit"&gt;&lt;Run FontFamily="inherit"&gt;"02"&lt;/Run&gt;&lt;/Span&gt;, "&lt;Run FontFamily="inherit"&gt;shortDescription&lt;/Run&gt;": &lt;Span FontFamily="inherit"&gt;&lt;Run FontFamily="inherit"&gt;"Re- entry"&lt;/Run&gt;&lt;/Span&gt;, "&lt;Run FontFamily="inherit"&gt;description&lt;/Run&gt;": &lt;Span FontFamily="inherit"&gt;&lt;Run FontFamily="inherit"&gt;"Re- entry"&lt;/Run&gt;&lt;/Span&gt;, "&lt;Run FontFamily="inherit"&gt;entryType&lt;/Run&gt;": &lt;Span FontFamily="inherit"&gt;&lt;Run FontFamily="inherit"&gt;"Re- entry"&lt;/Run&gt;&lt;/Span&gt;, &lt;Run FontFamily="inherit"&gt;}&lt;/Run&gt;, { "&lt;Run FontFamily="inherit"&gt;codeValue&lt;/Run&gt;": &lt;Span</pre>

```

FontFamily="inherit"> <Run
FontFamily="inherit"> "03" </Run> </Span>, " <Run
FontFamily="inherit"> shortDescription </Run>": <Span
FontFamily="inherit"> <Run FontFamily="inherit"> "Next year
school" </Run> </Span>, " <Run
FontFamily="inherit"> description </Run>": <Span
FontFamily="inherit"> <Run FontFamily="inherit"> "Next year
school" </Run> </Span>, " <Run
FontFamily="inherit"> entryType </Run>": <Span
FontFamily="inherit"> <Run FontFamily="inherit"> "Next year
school" </Run> </Span>, <Run FontFamily="inherit"> } </Run>, {
" <Run FontFamily="inherit"> codeValue </Run>": <Span
FontFamily="inherit"> <Run
FontFamily="inherit"> "04" </Run> </Span>, " <Run
FontFamily="inherit"> shortDescription </Run>": <Span
FontFamily="inherit"> <Run FontFamily="inherit"> "New to
education system" </Run> </Span>, " <Run
FontFamily="inherit"> description </Run>": <Span
FontFamily="inherit"> <Run FontFamily="inherit"> "New to
education system" </Run> </Span>, " <Run
FontFamily="inherit"> entryType </Run>": <Span
FontFamily="inherit"> <Run FontFamily="inherit"> "New to
education system" </Run> </Span>, <Run
FontFamily="inherit"> } </Run>, { " <Run
FontFamily="inherit"> codeValue </Run>": <Span
FontFamily="inherit"> <Run
FontFamily="inherit"> "05" </Run> </Span>, " <Run
FontFamily="inherit"> shortDescription </Run>": <Span
FontFamily="inherit"> <Run
FontFamily="inherit"> "Other" </Run> </Span>, " <Run
FontFamily="inherit"> description </Run>": <Span
FontFamily="inherit"> <Run
FontFamily="inherit"> "Other" </Run> </Span>, " <Run
FontFamily="inherit"> entryType </Run>": <Span
FontFamily="inherit"> <Run
FontFamily="inherit"> "Other" </Run> </Span>, <Run
FontFamily="inherit"> } </Run>
] </Span> </Paragraph> <Paragraph> <LineBreak
/> </Paragraph> </Section> </Section> <Paragraph>
</Paragraph> </FlowDocument>
All values returned in API should be values selectable / mapped in
SIS

```



2

Compare values returned from API match selectable / mapped values in SIS

### Test Case [37927](#): Validate correct values for exitWithdrawTypeDescriptors

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (2)

#	Title	Expected Value
1	Execute a GET on the Descriptor/Type and validate the values returned.	<p>Values should be as follows: [ { "codeValue": "PCC", "shortDescription": "Prior Completion Credential", "description": "Prior Completion Credential", "exitWithdrawType": "Prior Completion Credential", }, { "codeValue": "BCA", "shortDescription": "Below Compulsory Age", "description": "Below Compulsory Age", "exitWithdrawType": "Below Compulsory Age", }, { "codeValue": "TOS", "shortDescription": "Transfer Out-of-State.", "description": "Transfer Out-of-State. Known to be Continuing.", "exitWithdrawType": "Transfer Out-of-State.", }, { "codeValue": "TNC", "shortDescription": "Transfer to a School Not Covered by WSLs", "description": "Transfer to a School Not Covered by WSLs. Known to be Continuing.", "exitWithdrawType": "Transfer to a School Not Covered by WSLs", }, { "codeValue": "ETC", "shortDescription": "Expected Transfer to Another School Covered by WSLs.", "description": "Expected Transfer to Another School Covered by WSLs; Not Known to be Continuing", "exitWithdrawType": "Expected Transfer to Another School Covered by WSLs.", }, { "codeValue": "HSC", "shortDescription": "High School Completion", "description": "High School Completion", "exitWithdrawType": "High School Completion", }, { "codeValue": "DE", "shortDescription": "Death", "description": "Death", "exitWithdrawType": "Death", }, { "codeValue": "MA", "shortDescription": "Reached Maximum Age For Services", "description": "Reached Maximum Age For Services", "exitWithdrawType": "Reached Maximum Age For Services", }, { "codeValue": "ODO", "shortDescription": "Other Dropout or</p>



Possible Dropout", "description": "Other Dropout or Possible Dropout", "exitWithdrawType": "Other Dropout or Possible Dropout"}, {"codeValue": "TC", "shortDescription": "Transfer to Another WI School", "description": "Transfer to Another WI School. Known to be Continuing.", "exitWithdrawType": "Transfer to Another WI School"}, {"codeValue": "ISM", "shortDescription": "Interstate move. Move to another state within the country.", "description": "Interstate move. Move to another state within the country. Not known to be continuing", "exitWithdrawType": "Interstate move. Move to another state within the country."}, {"codeValue": "INM", "shortDescription": "International move.", "description": "International move. Not known to be continuing", "exitWithdrawType": "International move."}]

All values returned in API should be values selectable / mapped in SIS






2

Compare values returned from API match selectable / mapped values in SIS

### Test Case [37928](#): Validate correct values for schoolYearTypes

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (3)

#	Title	Expected Value
 1	Execute a GET on on the Descriptor/Type and validate the values returned.	Values should be as follows: schoolYear: 1989, schoolYearDescription: 1988-1989 and follow that pattern all the way to 2016. The only exception is 2015. It will have currentSchoolYear = true where as all other years will = false for the 2014/15 school year.
 2	Compare values returned from API match selectable / mapped values in SIS	All values returned in API should be values selectable / mapped in SIS
 3		

### Test Case [37929](#): Validate correct value for gradeLevelDescriptors

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (2)

#	Title	Expected Value
 1	Execute a GET on on the Descriptor/Type and validate the values returned.	Values should be as follows: [{"codeValue": "KG", "shortDescription": "5 year old Kindergarten", "description": "5 year old Kindergarten", "gradeLevelType": "5 year old Kindergarten"}, {"codeValue": "PK", "shortDescription": "Pre-Kindergarten", "description": "Pre-Kindergarten", "gradeLevelType": "Pre-Kindergarten"}, {"codeValue": "1", "shortDescription": "First Grade", "description": "First Grade", "gradeLevelType": "First Grade"}, {"codeValue": "2", "shortDescription": "Second Grade", "description": "Second Grade", "gradeLevelType": "Second Grade"}, {"codeValue": "3", "shortDescription": "Third Grade", "description": "Third Grade", "gradeLevelType": "Third Grade"}, {"codeValue": "4", "shortDescription": "Fourth Grade", "description": "Fourth Grade", "gradeLevelType": "Fourth Grade"}, {"codeValue": "5", "shortDescription": "Fifth Grade", "description": "Fifth Grade", "gradeLevelType": "Fifth Grade"}, {"codeValue": "6", "shortDescription": "Sixth Grade", "description": "Sixth Grade", "gradeLevelType": "Sixth Grade"}, {"codeValue": "7", "shortDescription": "Seventh Grade", "description": "Seventh Grade", "gradeLevelType": "Seventh Grade"}, {"codeValue": "8", "shortDescription": "Eighth Grade", "description": "Eighth Grade", "gradeLevelType": "Eighth Grade"}, {"codeValue": "9", "shortDescription": "Ninth Grade", "description": "Ninth Grade", "gradeLevelType": "Ninth Grade"}, {"codeValue": "10", "shortDescription": "Tenth Grade", "description": "Tenth Grade", "gradeLevelType": "Tenth Grade"}, {"codeValue": "11", "shortDescription": "Eleventh Grade", "description": "Eleventh Grade", "gradeLevelType": "Eleventh Grade"}, {"codeValue": "12", "shortDescription": "Twelfth Grade", "description": "Twelfth Grade", "gradeLevelType": "Twelfth Grade"}, {"codeValue": "K4", "shortDescription": "4-year-old Kindergarten", "description": "4-year-old Kindergarten", "gradeLevelType": "4-year-old Kindergarten"}, {"codeValue": "K3", "shortDescription": "3-year-old Kindergarten", "description": "3-year-old Kindergarten", "gradeLevelType": "3-year-old Kindergarten"}]



2



Compare values returned from API match selectable / mapped values in SIS

All values returned in API should be values selectable / mapped in SIS

### **Test Case 37930: Validate correct values for residencyStatusDescriptors**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:



#### Test Steps (2)

#	Title	Expected Value
 1	Execute a GET on on the Descriptor/Type and validate the values returned.	Values should be as follows: [ { "codeValue": "01", "shortDescription": "Resident of administrative unit and usual school attendance area", "description": "Resident of administrative unit and usual school attendance area", "residencyStatusType": "Resident of administrative unit and usual school attendance area", }, { "codeValue": "02", "shortDescription": "Resident of administrative unit, but of other school attendance area", "description": "Resident of administrative unit, but of other school attendance area", "residencyStatusType": "Resident of administrative unit, but of other school attendance area", }, { "codeValue": "03", "shortDescription": "Resident of an administrative unit that crosses state boundaries", "description": "Resident of an administrative unit that crosses state boundaries", "residencyStatusType": "Resident of an administrative unit that crosses state boundaries", }, { "codeValue": "04", "shortDescription": "Resident of another state", "description": "Resident of another state", "residencyStatusType": "Resident of another state", }, { "codeValue": "05", "shortDescription": "Resident of this state, but not of this administrative unit", "description": "Resident of this state, but not of this administrative unit", "residencyStatusType": "Resident of this state, but not of this administrative unit", }, ]
 2	Compare values returned from API match selectable / mapped values in SIS	All values returned in API should be values selectable / mapped in SIS

### **Test Case 37931: Validate correct values for countDateNameTypes**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:



#### Test Steps (2)

#	Title	Expected Value
 1	Execute a GET on on the Descriptor/Type and validate the values returned.	Values should be as follows: [ { "codeValue": "3FS", "description": "3rd Friday of September", "shortDescription": "3rd Fri Sept", }, { "codeValue": "O1", "description": "October 1st", "shortDescription": "Oct 1st", }, ]
 2	Compare values returned from API match selectable / mapped values in SIS	All values returned in API should be values selectable / mapped in SIS

### **Test Case 37932: Validate correct values for countDateReceivingServicesTypes**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (2)

#	Title	Expected Value
 1	Execute a GET on on the Descriptor/Type and validate the values returned.	Values should be as follows: [ { "countDateReceivingServiceTypeId": 1, "codeValue": "P", "description": "Present, receiving services on the count date", "shortDescription": "Present", }, { "countDateReceivingServiceTypeId": 2, "codeValue": "A", "description": "Absent, receiving services on the count date", "shortDescription": "Absent", }, { "countDateReceivingServiceTypeId": 3, "codeValue": "N", "description": "Not receiving services", "shortDescription": "Not Receiving Services", }, ]
 2	Compare values returned from API match selectable / mapped values in SIS	All values returned in API should be values selectable / mapped in SIS

### **Test Case 37933: Validate correct values for termTypes**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (2)

#	Title	Expected Value
---	-------	----------------



1

Execute a GET on on the Descriptor/Type and validate the values returned.

Values should be as follows: [ { "codeValue": "10B", "description": "10th Block", "shortDescription": "10th Block", }, { "codeValue": "1B", "description": "1st Block", "shortDescription": "1st Block", }, { "codeValue": "1Q", "description": "1st Quarter", "shortDescription": "1st Quarter", }, { "codeValue": "1S", "description": "1st Semester", "shortDescription": "1st Semester", }, { "codeValue": "1T", "description": "1st Trimester", "shortDescription": "1st Trimester", }, { "codeValue": "2B", "description": "2nd Block", "shortDescription": "2nd Block", }, { "codeValue": "2Q", "description": "2nd Quarter", "shortDescription": "2nd Quarter", }, { "codeValue": "2S", "description": "2nd Semester", "shortDescription": "2nd Semester", }, { "codeValue": "2T", "description": "2nd Trimester", "shortDescription": "2nd Trimester", }, { "codeValue": "3B", "description": "3rd Block", "shortDescription": "3rd Block", }, { "codeValue": "3Q", "description": "3rd Quarter", "shortDescription": "3rd Quarter", }, { "codeValue": "3T", "description": "3rd Trimester", "shortDescription": "3rd Trimester", }, { "codeValue": "4B", "description": "4th Block", "shortDescription": "4th Block", }, { "codeValue": "4Q", "description": "4th Quarter", "shortDescription": "4th Quarter", }, { "codeValue": "5B", "description": "5th Block", "shortDescription": "5th Block", }, { "codeValue": "6B", "description": "6th Block", "shortDescription": "6th Block", }, { "codeValue": "7B", "description": "7th Block", "shortDescription": "7th Block", }, { "codeValue": "8B", "description": "8th Block", "shortDescription": "8th Block", }, { "codeValue": "9B", "description": "9th Block", "shortDescription": "9th Block", }, { "codeValue": "A", "description": "Annual", "shortDescription": "Annual", } ]



2

Compare values returned from API match selectable / mapped values in SIS

Only the value of 'A' (Annual) returned in API should be selectable for initial release.

### Test Case 37934: Validate correct values for diplomaTypes

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (2)

#	Title	Expected Value
1	Execute a GET on on the Descriptor/Type and validate the values returned.	Values should be as follows: [ { "codeValue": "Regular diploma", "description": "Regular diploma", "shortDescription": "Regular diploma", }, { "codeValue": "Endorsed/advanced diploma", "description": "Endorsed/advanced diploma", "shortDescription": "Endorsed/advanced diploma", }, { "codeValue": "Regents diploma", "description": "Regents diploma", "shortDescription": "Regents diploma", }, { "diplomaTypeId": 4, "codeValue": "International Baccalaureate", "description": "International Baccalaureate", "shortDescription": "International Baccalaureate", }, { "codeValue": "Modified diploma", "description": "Modified diploma", "shortDescription": "Modified diploma", }, { "codeValue": "Other diploma", "description": "Other diploma", "shortDescription": "Other diploma", }, { "codeValue": "Alternative credential", "description": "Alternative credential", "shortDescription": "Alternative credential", }, { "codeValue": "Certificate of attendance", "description": "Certificate of attendance", "shortDescription": "Certificate of attendance", }, { "codeValue": "Certificate of completion", "description": "Certificate of completion", "shortDescription": "Certificate of completion", }, { "codeValue": "High school equivalency credential, other than GED", "description": "High school equivalency credential, other than GED", "shortDescription": "High school equivalency credential, other than GED", }, { "codeValue": "General Educational Development (GED) credential", "description": "General Educational Development (GED) credential", "shortDescription": "General Educational Development (GED) credential", }, { "codeValue": "Post graduate certificate (grade 13)", "description": "Post graduate certificate (grade 13)", "shortDescription": "Post graduate certificate (grade 13)", }, { "codeValue": "Career and Technical Education certificate", "description": "Career and Technical Education certificate", "shortDescription": "Career and Technical Education certificate", }, { "codeValue": "Other", "description": "Other", "shortDescription": "Other", }, { "codeValue": "Occupational License", "description": "Occupational License", "shortDescription": "Occupational License", }, { "codeValue": "Industry-recognized Certification", "description": "Industry-recognized Certification", "shortDescription": "Industry-recognized Certification", }, { "codeValue": "Apprenticeship", "description": "Apprenticeship", "shortDescription": "Apprenticeship", } ]

Certificate", "description": "Apprenticeship Certificate",  
"shortDescription": "Apprenticeship Certificate", } ]



2

Compare values returned from API match selectable / mapped values in SIS

All values returned in API should be values selectable / mapped in SIS

### **Test Case 37935: Validate correct values for achievementCategoryDescriptor**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (2)

#	Title	Expected Value
1	Execute a GET on on the Descriptor/Type and validate the values returned.	Values should be as follows: [ { "codeValue": "Competency Mastered", "shortDescription": "Competency Mastered", "description": "Competency Mastered", "achievementCategoryType": "Competency Mastered", }, { "codeValue": "Certificate Earned", "shortDescription": "Certificate Earned", "description": "Certificate Earned", "achievementCategoryType": "Certificate Earned", }, { "codeValue": "Competency Retained", "shortDescription": "Competency Retained", "description": "Competency Retained", "achievementCategoryType": "Competency Retained", }, { "codeValue": "Course Completed", "shortDescription": "Course Completed", "description": "Course Completed", "achievementCategoryType": "Course Completed", }, { "codeValue": "Diploma Earned", "shortDescription": "Diploma Earned", "description": "Diploma Earned", "achievementCategoryType": "Diploma Earned", }, { "codeValue": "Level Completed", "shortDescription": "Level Completed", "description": "Level Completed", "achievementCategoryType": "Level Completed", }, { "codeValue": "License Earned", "shortDescription": "License Earned", "description": "License Earned", "achievementCategoryType": "License Earned", }, { "codeValue": "License Endorsement Earned", "shortDescription": "License Endorsement Earned", "description": "License Endorsement Earned", "achievementCategoryType": "License Endorsement Earned", }, { "codeValue": "Non-Academic Honor", "shortDescription": "Non-Academic Honor", "description": "Non-Academic Honor", "achievementCategoryType": "Non-Academic Honor", }, { "codeValue": "Participation", "shortDescription": "Participation", "description": "Participation", "achievementCategoryType": "Participation", }, { "codeValue": "Recognition", "shortDescription": "Recognition", "description": "Recognition", "achievementCategoryType": "Recognition", } ]
2	Compare values returned from API match selectable / mapped values in SIS	All values returned in API should be values selectable / mapped in SIS

## **Suite 37987: /student**

State: In progress

### Test Cases (17)

#### **Test Case 37903: Create STUDENT**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (13)

#	Title	Expected Value
1	Find a WISEid that hasn't been used (no student record exists)	WISEid found that hasn't been used to create a student record
2	Begin creating student record in SIS	Fill in WISEid from prior step. When JSON is built should populate studentUniqueId in JSON
3	Enter first name and last name	When JSON is built should populate firstName and lastName
4	Enter in gender	When JSON is built should populate sexType with a valid value from /sexTypes Type
5	Enter birth date	When JSON is built should populate birthDate
6	Enter date entered United States (if applicable)	When JSON is built should populate dateEnteredUS

	7	Enter economic disadvantage value (true or false)	When JSON is built should populate economicDisadvantaged with a value = true or false
	8	Enter school food services eligibility descriptor	When JSON is built should populate schoolFoodServicesEligibilityDescriptor with a valid codeValue from /schoolFoodServicesEligibilityTypes
	9	Enter limited english proficiency value	When JSON is built should populate limitedEnglishProficiencyDescriptor with a valid codeValue from /limitedEnglishProficiencyTypes
	10	Enter hispanic latino ethnicity	When JSON is built should populate hispanicLatinoEthnicity with a value = true or false
	11	Enter a minimum of two student characteristics	When JSON is built should populate characteristics property with two string values (e.g. descriptor: "Refugee", descriptor: "Section 504 Handicapped"). Valid values are codeValue from /studentCharacteristicTypes
	12	Enter a minimum of two races	When JSON is built should populate races property with two string values (e.g. raceType: "Asian", raceType: "White". Valid values are Description from /raceTypes
	13	Enter a minimum of two disabilities where one is flagged as primary and one is flagged as secondary	When JSON is built should populate disabilities property with two string values per disability (e.g. disabilityDescriptor: "A",orderOfDisability:1 AND disabilityDescriptor: "H",orderOfDisability:2). Valid values are codeValue from /disabilityTypes

### **Test Case 37904: Save / Submit STUDENT**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (1)

#	Title	Expected Value
	1 Save / Submit STUDENT to WI DPI Ed-Fi API. Capture JING video of this action and record URL of video.	Submission is successful to API and Response code = 200 is returned.

### **Test Case 37905: Attempt to create / submit a STUDENT with a hispanicLatinoEthnicity value = truthful**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (3)

#	Title	Expected Value
	1 Create a student with HispanicLatinoEthnicity other than true/ false/ null	Response should be a HTTP-500 which states "message": "An error has occurred." should be returned
	2 Create a student with HispanicLatinoEthnicity= true/ false	Student should be created with a HTTP-200(Success) code as response
	3 Create a student with HispanicLatinoEthnicity= null	Student should be created with a HTTP-200(Success) code as response and this record when queried should have HispanicLatinoEthnicity= false

### **Test Case 37906: Attempt to create / submit a STUDENT with an invalid date format for birthDate and dateEnteredUS**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (1)

#	Title	Expected Value
	1 Attempt to create / submit a STUDENT with an invalid date format for birthDate and dateEnteredUS	Observed response should be HTTP-400 (Bad Request)

### **Test Case 37907: Attempt to create / submit a STUDENT with a valid racetype**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (5)

#	Title	Expected Value
	1 Attempt to create / submit a STUDENT with a value of "White" for raceType	Student should be created with Races field displaying the value as 'White'
	2 Attempt to create / submit a STUDENT with a value of "American Indian - Alaskan Native" for raceType	Student should be created with Races field displaying the value as 'American Indian - Alaskan Native'
	3 Attempt to create / submit a STUDENT with a value of "Asian" for raceType	Student should be created with Races field displaying the value as 'Asian'
	4 Attempt to create / submit a STUDENT with a value of "Black - African American" for raceType	Student should be created with Races field displaying the value as 'Black - African American'
	5 Attempt to create / submit a STUDENT with a value of "Native Hawaiian - Pacific Islander" for raceType	Student should be created with Races field displaying the value as 'Native Hawaiian - Pacific Islander'

**Test Case 37908: Attempt to create / submit a STUDENT without sending birthDate required by EdFi**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
1	Create a student with no Date of Birth field.	Should return an error with response: BirthDate is required.

**Test Case 37909: Modify a STUDENT resource record with no StudentSchoolAssociation.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
1	Modify a STUDENT resource record with no StudentSchoolAssociation.	Resource shouldnot be updated. Response : "Authorization denied. The claim does not have any established relationships with the requested resource."

**Test Case 37910: Attempt to delete a STUDENT resource record that does have child resources tied to it (e.g. studentSchoolAssociation)**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
1	Attempt to delete a StudentUSI attached to child resources (e.g. studentSchoolAssociation)	"The resource (or a subordinate entity of the resource) cannot be deleted because it is a dependency of the 'studentUSI' value of the 'studentSchoolAssociation' entity."

**Test Case 37936: Attempt to create / submit a STUDENT with a value of "Hispanic/ Latino" for raceType**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
1	Create a value with Hispanic/ Latino as raceType.	Should error out stating: Unable to resolve value 'Hispanic or Latino' to an existing 'RaceType' resource.

**Test Case 37937: Attempt to create / submit a STUDENT with invalid Gender details**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
1	Attempt to create/ submit a student where sexType = NA	Resource should not be created and the response should be : 'Unable to resolve value 'NA' to an existing 'SexType' resource.'

**Test Case 37964: Attempt to create / submit a STUDENT with invalid Student Characteristics.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (2)

#	Title	Expected Value
1	Attempt to create / submit a STUDENT with duplicate descriptors for Student Characteristics.	Resource should not be created. The response: "A duplicate StudentCharacteristicDescriptorId conflict occurred when attempting to create a new Student resource."
2	Attempt to create/ submit a STUDENT with no descriptor tag.	Resource should not be created. The response: "StudentCharacteristicDescriptorId is required."

**Test Case 37965: Attempt to create / submit a STUDENT with invalid Student Disabilities.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (2)

#	Title	Expected Value
---	-------	----------------

	1	Attempt to create / submit a STUDENT with duplicate disability descriptors	Resource should not be created. The Response: "An unexpected error occurred on the server."
	2	Attempt to create / submit a STUDENT with duplicate disability descriptors having multiple ordersofDisability.	Resource should not be created. The Response: "An unexpected error occurred on the server."

**Test Case 37966: Attempt to Create/ Submit a Student populating all the fields defined by Ed-Fi with valid properties.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
	1 Attempt to Create/ Submit a Student populating all the fields defined by Ed-Fi with valid properties.	Student record should be created

**Test Case 37971: Attempt to Create a Student resource with StudentCharacteristicDescriptor as 'Unaccompanied Youth'.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
	1 Attempt to Create a Student resource with StudentCharacteristicDescriptor as 'Unaccompanied Youth'.	Resource should not be created. The response: "Unable to resolve value 'Unaccompanied Youth' to an existing 'StudentCharacteristicDescriptor' resource."

**Test Case 37972: Attempt to Create/ Update the Student resource with LearningStyle values beyond the range.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
	1 Attempt to Create/ Update the Student resource with LearningStyle values beyond the range (eg. 10).	Resource should not be created. The number for the fields should be between -9.9999 to 9.9999

**Test Case 37973: Attempt to create/ submit Student with missing Disability Properties.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (2)

#	Title	Expected Value
	1 Attempt to create/ submit Student with no entry for DisabilityDescriptorId	Optional in EdFi model, so will be allowed.
	2 Attempt to create/ submit Student with no entries for orderofdisability, disabilityDiagnosis and disabilityDeterminationSourceType.	Resource should be created with missing fields displaying the values as NULL

**Test Case 37974: Attempt to create/ submit a Student with duplicate addressTypes.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
	1 Attempt to create/ submit a Student with duplicate addressTypes.	Resource should not be created. The Response: "A duplicate AddressTypeId conflict occurred when attempting to create a new Student resource."

## Suite 37990: /studentAcademicRecord

State: In progress

### Test Cases (6)

**Test Case 37946: Create a Student Academic Record with no Student School Association**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (2)

#	Title	Expected Value
---	-------	----------------



	1	Create a student	Student should be created with a Unique Student Id
	2	Attempt to generate Student Academic Records for this Student without any Student School Association.	Record should not be created. The Response: "Authorization denied. The claim does not have any established relationships with the requested resource."

**Test Case 37947: Create a Record with no Education Organization Reference**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (3)

#	Title	Expected Value	
	1	Create a Student	Student Record should be created with Unique Id
	2	Establish student school association for this Id	Student school association should be successful
	3	Attempt to create Student Academic Record with no Education Organization Reference	Resource should not be generated. The Response: EducationOrganizationId is required.

**Test Case 37948: Create a Record With no SchoolYearTypeReference.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (3)

#	Title	Expected Value	
	1	Create a Student	Student Record should be created with Unique Id
	2	Establish student school association for this Id	Student school association should be successful
	3	Attempt to create Student Academic Record with no SchoolYearTypeReference.	Resource should not be created. The Response: SchoolYear is required.

**Test Case 37949: Delete a Record**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value	
	1	Delete a Student Academic Record	Record should be deleted with a response code 204: Successfully deleted.

**Test Case 37975: Create a Student Academic Record for Student with Valid Student School Association.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (3)

#	Title	Expected Value	
	1	Create a Student.	Student Record should be created with Unique Id
	2	Provide a valid School Association for this Record	Student School Association should be created
	3	Attempt to create Academic Record for this Student	Record should be successfully created

**Test Case 37976: Attempt to create a Student Academic Record without TermDescriptor.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value	
	1	Attempt to create a Student Academic Record without TermDescriptor.	Resource should not be created. The Response: TermDescriptorId is required.

**Suite 37988: /studentSchoolAssociation**

State: In progress




## Test Cases (18)

### **Test Case 37944: Create a Student School Association record with no entry date.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (1)

#	Title	Expected Value
 1	Create a Student School Association record with no entry date.	Resource should not be created. The Response: EntryDate is required.

### **Test Case 37945: Create a Student School Association record with no entrylevelgradeddescriptor.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (1)

#	Title	Expected Value
 1	Create a Student School Association record with no entrylevelgradeddescriptor.	Resource should not be created. The Response: EntryGradeLevelDescriptorId is required.

### **Test Case 37950: Create a record with no school reference.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (1)

#	Title	Expected Value
 1	Create a record with no school reference.	Record shouldnot be created. Response: SchoolId is required.

### **Test Case 37951: Create a record with no student reference.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


#### Test Steps (1)

#	Title	Expected Value
 1	Create a record with no student reference.	Record shouldnot be created. Resonse: StudentUniqueId is required.

### **Test Case 37952: Create/ Update a record with no SchoolYearType reference.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


#### Test Steps (1)

#	Title	Expected Value
 1	Create/ Update a record with no SchoolYearType reference.	This is allowed in the EdFi model although will likely be an L2 validation error in the WISEdata Portal. Recommended for SIS to implement client side check.

### **Test Case 37953: Create/ Update the Student record with no entry type descriptor.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (1)

#	Title	Expected Value
 1	Create/ Update the Student record with no entry type descriptor.	This is allowed in the EdFi model although will likely be an L2 validation error in the WISEdata Portal

### **Test Case 37954: Create/ Update the Student record with no exit withdraw type descriptor.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


#### Test Steps (1)

#	Title	Expected Value
 1	Create/ Update the Student record with no exit withdraw type descriptor.	Record should be created.

### **Test Case 37955: Create/ Update the Student record with no exit withdraw date.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
 1	Create/ Update the Student record with no exit withdraw date.	Record should be created.

 **Test Case 37956: Create/ Update the Student record with no residency status descriptor.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


Test Steps (1)

#	Title	Expected Value
 1	Create/ Update the Student record with no residency status descriptor.	Record should be created.

 **Test Case 37957: Create/ Update the Student record with no primary school info.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


Test Steps (1)

#	Title	Expected Value
 1	Create/ Update the Student record with no primary school info.	This is allowed in the EdFi model although will likely be an L2 validation error in the WISEdata Portal. Recommended for SIS to implement client side check.

 **Test Case 37958: Delete a Student School Association Record With attached 'studentSchoolAssociationWisconsinExt' entity.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


Test Steps (1)

#	Title	Expected Value
 1	Delete a Student School Association Record With attached 'studentSchoolAssociationWisconsinExt' entity.	Resources should not be deleted. The Response: "The resource (or a subordinate entity of the resource) cannot be deleted because it is a dependency of the 'studentSchoolAssociationWisconsinExt' entity."

 **Test Case 37967: Attempt to Create/ Modify student record with Grade Level Placement outside the School Range.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


Test Steps (1)

#	Title	Expected Value
 1	Attempt to Create/ Modify student record with Grade Level Placement outside the School Range.	This is allowed in the EdFi model although will likely be an L2 validation error in the WISEdata Portal. Recommended for SIS to implement client side check.

 **Test Case 37968: Create a student record with entry date before date of birth.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


Test Steps (1)

#	Title	Expected Value
 1	Create a student record with entry date before date of birth.	This is allowed in the EdFi model although will likely be an L2 validation error in the WISEdata Portal. Recommended for SIS to implement client side check.

 **Test Case 37969: Create/ Submit a student with multiple primary enrollment records with no exit dates.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
 1	Create/ Submit a student with multiple primary enrollment records with no exit dates.	This is allowed in the EdFi model although will likely be an L2 validation error in the WISEdata Portal. Recommended for SIS to implement client side check.

 **Test Case 37970: Create/ Modify a Student record with invalid exit dates.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


Test Steps (1)

#	Title	Expected Value
 1	Create/ Modify a Student record with invalid exit dates.	This is allowed in the EdFi model although will likely be an L2 validation error in the WISEdata Portal. Recommended for SIS to implement client side check.

 **Test Case 37977: Modify a Student School Association Record.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
 1	Modify a Student School Association Record by changing the primary keys.	Record should be successfully updated.

 **Test Case 37983: Delete a Student School Association Record Without attached 'studentSchoolAssociationWisconsinExt' entity.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:



Test Steps (1)

#	Title	Expected Value
 1	Delete a Student School Association Record Without attached 'studentSchoolAssociationWisconsinExt' entity.	Record should be successfully deleted.

 **Test Case 37985: Attempt to Create StudentSchoolAssociation with multiple entryGradeLevelDescriptors.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (2)

#	Title	Expected Value
 1	Attempt to Create StudentSchoolAssociation with multiple entryGradeLevelDescriptors.	Record should be created with entryGradeLevelDescriptor displaying the highest grade of the provided entries.
 2		

## Suite 37989: /studentSchoolAssociationWisconsinExtension


State: In progress

### Test Cases (14)

 **Test Case 37938: Create/ Update a record with invalid School Id.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


Test Steps (1)

#	Title	Expected Value
 1	Create/ Update a record with invalid School Id.	Record shouldnot be created. response: "The value supplied for the related 'studentSchoolAssociation' resource does not exist."

 **Test Case 37939: Create/ Update a record with Invalid EntryDate.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
 1	Create/ Update a record with Invalid EntryDate that doesn't match the entryDate for the /studentSchoolAssociation for the same student.	Record shouldnot be created. response: "The value supplied for the related 'studentSchoolAssociation' resource does not exist."

 **Test Case 37940: Create/ Update record with Invalid values for actualDaysAttendance**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
---	-------	----------------



1


Enter 1000.

Allowed. There will be an L2 validation that will appear in the WISEdata Portal depending on other data properties received. Recommended that SIS has a client edit to avoid invalid entry.

### **Test Case 37941: Create/ Update record with Invalid values for possibleDaysAttendance**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


#### Test Steps (1)

#	Title	Expected Value
 1	Enter 5000	Allowed. There will be an L2 validation that will appear in the WISEdata Portal depending on other data properties received. Recommended that SIS has a client edit to avoid invalid entry.

### **Test Case 37942: Create/ Update record with actualDaysAttendance higher than possibleDaysAttendance**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (1)

#	Title	Expected Value
 1	For possible enter 250, but for actual enter 300.	Allowed. There will be an L2 validation that will appear in the WISEdata Portal depending on other data properties received. Recommended that SIS has a client edit to avoid invalid entry.

### **Test Case 37943: Test for decimal values in actual/ possibleDaysAttendance.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


#### Test Steps (1)

#	Title	Expected Value
 1	Test for decimal values in actual/ possibleDaysAttendance.	Record should be created. The Response when queried should display the decimal values as provided.

### **Test Case 37959: Create/ update a record with no expectedTransferEducationalOrganizationReference as needed by DPI.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


#### Test Steps (1)

#	Title	Expected Value
 1	Create/ update a record with no expectedTransferEducationalOrganizationReference as needed by DPI.	Allowed. There will be an L2 validation that will appear in the WISEdata Portal depending on other data properties received. Recommended that SIS has a client edit to avoid invalid entry.

### **Test Case 37960: Create/ update a record with no expectedTransferSchoolReference as needed by DPI.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


#### Test Steps (1)

#	Title	Expected Value
 1	Create/ update a record with no expectedTransferSchoolReference as needed by DPI.	Allowed. There could be an L2 validation that will appear in the WISEdata Portal depending on other data properties received.

### **Test Case 37961: Validate the record for privateSchoolChoiceProgramParticipant.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

#### Test Steps (1)

#	Title	Expected Value
 1	Validate the record for privateSchoolChoiceProgramParticipant.. Pass in a value of true and then for the same student pass in false and then null.	All values are allowed.

### **Test Case 37962: Create a student with Invalid receiving services info.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


Test Steps (3)

#	Title	Expected Value
 1	Create a record with no "countDateNameType"	Record shouldn't be created. The Response: StudentSchoolAssociationWisconsinExtReceivingService[0]: CountDateNameTypeId is required.
 2	Create a record with no "countDateReceivingServiceType"	Record shouldn't be created. The Response: StudentSchoolAssociationWisconsinExtReceivingService[0]: CountDateReceivingServiceTypeId is required.
 3	Create a record with invalid "countDateNameType" and "countDateReceivingServiceType"	Record shouldn't be created. The Response: 'Unable to resolve value ' ' to an existing resource.'

 **Test Case 37963: Delete a WisconsinExtension record with Child entities like 'studentSchoolAssociationWisconsinExtReceivingService'.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
 1	Delete a WisconsinExtension record with Child entities like 'studentSchoolAssociationWisconsinExtReceivingService'.	Record shouldn't be deleted. The Response: "The resource (or a subordinate entity of the resource) cannot be deleted because it is a dependency of the 'studentSchoolAssociationWisconsinExtReceivingService' entity."

 **Test Case 37978: Modify a Wisconsin Extension Record.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:


Test Steps (1)

#	Title	Expected Value
 1	Modify a Wisconsin Extension Record.	Record should be updated accordingly.

 **Test Case 37979: Attempt to create Wisconsin Extension Record with multiple entry dates.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
 1	Attempt to create Wisconsin Extension Record with multiple entry dates.	Record shouldn't be created. The Response: "Aggregate was modified by another consumer."

 **Test Case 37984: Delete a WisconsinExtension record without Child entities.**

Owner: | Ready | Type: Manual | Collections | Iteration: Collections | Automated test: Not set | Assigned to:

Test Steps (1)

#	Title	Expected Value
 1	Delete a WisconsinExtension record without Child entities.	Record should be successfully deleted.