ellucian™

Banner Student
Advising Student Profile Release Guide

Release 9.3
March 2016

Contents

Introduction

This release guide documents the Banner 9.3 release of the Banner® Student Advising Student Profile module. Release 9.3 delivers functional updates as well as resolutions for change requests. Information is included for the Student Advisor application and the Student Self-Service application.

New for this release:

- Release Holds enhancement

- Enhanced security control for the display of the student's alternate registration PIN

- Accessibility improvements

Details for releasing holds and using the PIN display security controls can be found in the *Banner Student Self-Service and Advising Student Profile Handbook*.

Requirements

Please refer to the *Banner Student Advising Student Profile Installation Guide 9.3* and the *Banner Student Self-Service Installation Guide 9.3* documents for information on the following hardware and software requirements, tested browser and tablet versions, and other installation prerequisites.

- Hardware

- Oracle Database

- CPU and memory

- Application server

- Application server and operating system combinations

- Banner software

- Other Ellucian software

- Supported browsers

- Java

- Tablets

- JAWS screen reader

- Database Extension Utility

Release tagging

Use the following tags for application source code in the git repository.

Application	Tag
Advising Self-Service	rel-advising-9.3
Student Self-Service	rel-student-9.3

Documentation

The Banner Student Advising Student Profile 9.3 release includes the following documentation:

- *Banner Student Advising Student Profile Release Guide*

- *Banner Student Self-Service and Advising Student Profile Handbook*

- *Banner Student Advisor Installation Guide*

The following documents are also used with this release but are delivered separately.

- *Banner Student Self-Service Installation Guide*

- *Banner Database Upgrade Guide*

Translation

The application supports the following languages:

- English

- English (Australian, Great Britain, Ireland, and India)

- Spanish

- Portuguese (Brazil)

- Arabic

- French

- French (Canada)

Single Sign On with CAS or SAML 2.0 SSO Protocols

Application Navigator and integrating Banner applications require CAS or SAML 2.0 SSO to navigate between applications. Please refer to the *CAS Single Sign On Handbook* or the *Setting Up Ellucian Identity Service* document, available for download from the Ellucian Support Center.

SAML 2.0 support

Banner Student Advising Student Profile now supports SAML 2.0 SSO with Ellucian Identity Service (EIS), in addition to CAS SSO.

SAML 2.0 is an XML-based protocol that uses security tokens containing assertions to pass information about a principal (usually an end user) between a SAML authority (an identity provider) and a SAML consumer (a service provider).

Please refer to the *Banner Student Advisor Installation Guide 9.3* and the *Banner Student Self-Service Installation Guide 9.3* for details on the SAML 2.0 enhancement implementation.

Releasing Holds - Functional

Authorized faculty members and advisors can release holds for students on the Student Profile page based on the hold type and the associated persona of faculty, advisor, or both. This allows the student to perform the process that was prevented by the hold.

For one example, if your institution requires that a student must meet with his/her advisor prior to registration, and a hold has been placed on the student's record that prevents registration, the advisor can release the hold at that time.

A single hold or multiple holds can be released. Also, an administrator can create a rule that requires the entry of a release reason or comment when the hold is released.

Advising Student Profile changes

The Student Profile page has been updated to allow the release of holds.

When an advisor or faculty member views a student's profile and is authorized to release a hold associated with that student, the holds list can be displayed by clicking on the number next to the **Holds** field. Holds are listed in two categories, Releasable Holds and View Only Holds. Each hold can be expanded to display more detail.

The user can select the check box for a releasable hold and then select the **Release** button. At this point a release comment can be entered in the Releasable Holds window. The user can then click **OK** to release the hold. The release hold comment can be viewed by the author in the Notes section of the Student Profile page.

- See the "Release student holds" topic in the "Set Up and Use the Applications" chapter of the *Banner Student Self-Service and Advising Student Profile Handbook* for setup and processing information.

- See the "Student Profile Page" topic in the "Student Self-Service Pages" chapter of the *Banner Student Self-Service and Advising Student Profile Handbook* for steps used to release holds.

A new **Advisor Holds** field/column has been added to the Advisee Listing page. An advisor can sort or filter on this column to find advisees with advisor holds. The column data can be exported to Excel with data from the other columns on the page.

See the "Advisee Listing page" section in the "Student Advisor Pages" chapter of the *Banner Student Self-Service and Advising Student Profile Handbook* for more information.

Banner Student 8.x changes

The Persona Release Hold Type Form (SOAPHLD) and the associated Persona Release Hold Type Table (SORPHLD) were delivered in Banner Student 8.8 and are used to assign the hold types to the user roles/personas that can release holds. Hold types must exist on the Hold Type Code Validation Page (STVHLDD) in Banner 8.x and have the **Web Indicator** checked.

A process code and process rules are delivered for use with faculty security.

- The `RELEASEHOLDS` process code is used on the Process Control Code Validation Form (STVPROC).

- The `RELEASEHOLDS` process rules for `Advisor` and `Faculty` are used on the Faculty/Advisor Process Rules Form (SOAFACS).

See the "Delivered seed data" topic and the "Manually entered data" topic in the "Releasing Holds - Technical" section of this release guide for more information on new codes and rules used with this enhancement.

See the "Release student holds" topic in the "Set Up and Use the Applications" chapter of the *Banner Student Self-Service and Advising Student Profile Handbook* for setup and processing information.

New form

This administrative form is new for this enhancement. It was delivered in the Banner Student 8.8 release and is included in the *Banner Student Online Help 8.8* and *8.9*.

Persona Release Hold Type Form (SOAPHLD)

This form is used to assign hold types to user roles (personas) so that specific types of holds can be released for a student. Only one record per hold type is allowed.

> **Note:** This form is used with the Banner Student Advising Student Profile application. It is not used with Banner Student 8.x processing.

Fields	Descriptions
Hold Type	Release hold type code.
	Hold types used must have the **Web Indicator** checked on the Hold Type Code Validation Page (STVHLDD).
	List - Hold Type Code Validation Form (STVHLDD)

Fields	Descriptions
Description (untitled)	Hold type code description.
Persona	Persona code associated with the hold type. Supported role values are: FACULTYINSTRUCTOR, FACULTYADVISOR, FACULTYBOTH. List - Persona Codes (STVPERS)
Active	Checkbox used to indicate whether the persona is active for the release hold type. The default value is checked.
Activity Date	Date on which the record was last updated. Display only.
User	ID of the user who last updated the record. Display only.

Changed menu

The following menu has been modified for this enhancement. This change was delivered in the Banner Student 8.8 release.

Faculty Workload Control (*FACCONTROL)

The Persona Release Hold Type Form (SOAPHLD) has been added to this menu.

Releasing Holds - Technical

This section describes the technical updates for the release holds enhancement.

New table

The following table is used with this enhancement. It was delivered in the Banner Student 8.8 release and is used specifically with Banner Student Advising Student Profile 9.3.

Persona Release Hold Type Table (SORPHLD)

This table is used to control which hold types can be released and what personas can be used to release each hold type. Valid persona codes are `FACULTYADVISOR`, `FACULTYINSTRUCTOR`, or `FACULTYBOTH`. Hold types used must have the **Web Indicator** checked on the Hold Type Code Validation Page (STVHLDD).

Constraints are used to ensure that codes on STVPERS and STVHLDD cannot be deleted when they have been mapped in this table.

This table uses the following Primary Key:

```
PK_SORPHLD (SORPHLD_HLDD_CODE)
```

This table uses the following Foreign Keys:

```
FK1_SORPHLD_INV_STVHLDD_CODE
FOREIGN KEY (SORPHLD_HLDD_CODE)
REFERENCES STVHLDD (STVHLDD_CODE)

FK1_SORPHLD_INV_STVPERS_CODE
FOREIGN KEY (SORPHLD_PERS_CODE)
REFERENCES STVPERS (STVPERS_CODE)
```

The following columns are in this table.

Column	Attribute	Null	Comment
SORPHLD_HLDD_CODE	VARCHAR2(2)	No	HOLD CODE: The release hold type code.
SORPHLD_PERS_CODE	VARCHAR2(30)	No	PERSONA CODE: The persona code.

Column	Attribute	Null	Comment
SORPHLD_ACTIVE_IND	VARCHAR2(1)	No	ACTIVE INDICATOR: Indicator for whether the persona release hold is active or inactive.
SORPHLD_ACTIVITY_DATE	DATE	No	ACTIVITY DATE: The date that information in this record was entered or last updated.
SORPHLD_USER_ID	VARCHAR2(30)	No	USER ID: The user ID of the person who inserted or last updated this record.
SORPHLD_SURROGATE_ID	NUMBER(19)	Yes	SURROGATE ID: The surrogate ID for this record.
SORPHLD_VERSION	NUMBER(19)	Yes	VERSION: The optimistic lock token for this record.
SORPHLD_DATA_ORIGIN	VARCHAR2(30)	Yes	DATA ORIGIN: Source system that created or updated the data.
SORPHLD_VPDI_CODE	VARCHAR2(6)	Yes	VPDI CODE: The code representing the entity to which this record belongs for Multi-Entity Processing support.

Changed table

The following table has been changed for this enhancement. This change is delivered in the Banner General 8.8.3 release. Please refer to the *Banner General Release Guide 8.8.3* for more information.

Person Related Holds Repeating Table (SPRHOLD)

Changes were made to the gb_hold Business API for SPRHOLD processing.

The user ID of the last person who changed an SPRHOLD hold record is now saved and is stored in the SPRHOLD_USER_ID field. This information can be used for audit purposes. The ID of the original creator of the hold record is still stored in the SPRHOLD_USER field and is displayed on the SOAHOLD page.

These changes apply to hold records entered on SOAHOLD or when holds are released by an advisor using Banner Student Advising Student Profile 9.3.

Delivered seed data

The following seed data is used with this enhancement.

Student Note Category Validation Page (STVNTCG)

The following seed data is used with releasing holds and associated comments.

Note category code	Description
9999	Release Hold Comment

Process Control Code Validation Page (STVPROC)

The following seed data is used with faculty security and releasing holds.

Process	Description	Check Order	Enforce Check Order	PIN Control Allowed	Sys Req
RELEASEHOLDS	Release Student Holds	Both	No	No	Yes

Faculty/Advisor Process Rules Page (SOAFACS)

The following seed data is used with faculty security and releasing holds.

Process	Desc	Check Order	Process Available on Self-Service	All Access	PIN Control	Relation ship	Primary	Attribute Type Checking	Sys Req
RELEASEHOLDS	Release Student Holds	Advisor	Yes	No	No	No	No	No	Yes
RELEASEHOLDS	Release Student Holds	Faculty	Yes	No	No	No	No	No	Yes

Information Text Editor Page (GUAINFO)

Seed data is used for the warning message displayed with releasing holds. This message can be changed at your institution.

Page Name	Message Label	Source Ind	Audience	Locale	Seq No	Text Type	Message
STUDENTPROFILE	student.profile.release. holds.warning	Baseline	Faculty	en_US	1	Panel	This action cannot be undone unless you contact the Student Records Office. Select Cancel to leave the hold(s) in place.

Manually entered data

The following pages use data that is manually entered at your institution. This data is required if you wish to require the entry of release hold comments.

- Set up the `STUDENT_PROFILE` proxy system code on the Proxy Access Systems Code Validation Page (GTVSYST).

- Set up the `REQUIRE_RELEASE_HOLD_COMMENT` option type code for the `STUDENT_PROFILE` system code on the Proxy Access Systems and Options Type Rule Page (GTVOTYP) when you wish to require the user to enter a release hold comment.

See the "Require a release hold" section under the "Release student holds" topic in the "Set Up and Use the Applications" chapter of the *Banner Student Self-Service and Advising Student Profile Handbook* for more information.

New scripts

The following scripts are used with this enhancement. The SORPHLD table scripts were delivered with the Banner Student 8.8 release.

Note: Unless otherwise noted, new scripts are run as part of the upgrade process for a release.

Script	Result
sorphld_080800_01.sql	Creates table
sorphld_080800_02.sql	Adds Primary Key
sorphld_080800_03.sql	Adds Foreign Key for STVHLDD
sorphld_080800_04.sql	Adds Foreign Key for STVPERS
sorphld_080800_05.sql	Adds column comments
sgurinfoi.sql	Adds seed data to GUAINFO
ssppsorfacsi.sql	Adds seed data to SOAFACS
ssppstvproci.sql	Adds seed data to STVPROC
sstvstvntcgi.sql	Adds seed data to STVNTCG

Configuration file changes for holds

The `bannerStudentAdvisorUI_configuration.example` file has been updated for this release.

- A new example has been added under the Advisee Listing page information.

```
all.advisee.column.list.FacultyAdvisor=studentStatus,
studentType, hasReleasableHolds, isPrimaryAdvisor,
advisorType, program, major, campus, studyLevel, admitType,
admitTerm, catalogTerm, minor, concentration, college
```

The `hasReleasable Holds` item is used to enable if and where the Advisor Holds column should be displayed on the Advisee Listing page.

- A new comment has been added to the Holds information section.

The Release Holds feature also has a dependency on this property. In addition to the SOAFACS security rules for `DISPLAYHOLDS` and `RELEASEHOLDS`, the role must have access to view holds per this property record.

Configuration File Details

This section lists configuration file changes by application for release 9.3. Some changes from release 9.2 may be included.

> **Note:** If you are upgrading, the *.example changes should be merged/ verified against the existing *.groovy or *.properties files.

Student Self-Service application

The following changes have been made for the Student Self-Service application.

banner_configuration.example

The following has been added or updated.

Email batch size

```
/**
*************************************************************************
*              The Mail Server's recipient limit.
*
*************************************************************************/
email.batch.size = 600
```

Photo path

```
/**
*************************************************************************
*              Administrative User DataSource Configuration
*
*************************************************************************/
banner {
    picturesPath = System.getProperty('base.dir') + '/test/images'

    // defaultPhoto = <Fully qualified path to the photo to use if no user
photo is available.>
}
```

bannerStudentAdvisorUI_configuration.example

The following changes were made in this file.

- "FacultyOverride without AdvisorOverride is not supported." was added at line 21.

  # FacultyOverride - user is a faculty and override is selected in SIAINST - applies to Student Profile page only. **FacultyOverride without AdvisorOverride is not supported.**

- "hasReleasableHolds" was added at lines 30, 44, 46, 47, and 48.

  - #<columns> possible values: studentStatus, studentType, **hasReleasableHolds**, isPrimaryAdvisor, advisorType, program, major, standing, campus, studyLevel, admitType, admitTerm, catalogTerm,

  - #all.advisee.column.list.FacultyAdvisor=studentStatus, studentType, **hasReleasableHolds**, isPrimaryAdvisor, advisorType, program, major, campus, studyLevel, admitType, admitTerm, catalogTerm, minor, concentration, college

  - all.advisee.column.list.Advisor=program, major, standing, studentStatus, studentType, **hasReleasableHolds**, isPrimaryAdvisor, advisorType, campus, studyLevel, admitType, admitTerm, catalogTerm, minor, concentration, college, preferredEmailAddress, nonPreferredEmailAddress

  - all.advisee.column.list.FacultyAdvisor=program, major, standing, studentStatus, studentType, **hasReleasableHolds**, isPrimaryAdvisor, advisorType, campus, studyLevel, admitType, admitTerm, catalogTerm, minor, concentration, college, preferredEmailAddress, nonPreferredEmailAddress

  - all.advisee.column.list.AdvisorOverride=program, major, standing, studentStatus, studentType, **hasReleasableHolds**, isPrimaryAdvisor, advisorType, campus, studyLevel, admitType, admitTerm, catalogTerm, minor, concentration, college, preferredEmailAddress, nonPreferredEmailAddress

- The following was added at line 304.

  - # If Notes are enabled the Release Hold comment entered by the person releasing a hold is displayed in the Notes section to the author of the comment.

  - # Use the following property to control if in addition the roles AdvisorOverride and FacultyOverride can view the Release Hold Comment.

  - # You can add role of FacultyOverride in addition to AdvisorOverride or disable the Property.

    all.studentnotes.releaseholdnote.view=Profile
    all.studentnotes.releaseholdnote.roles=AdvisorOverride

- The following was added at line 396.

#The Release Holds feature also has a dependency on this property, in addition to SOAFACS security rules for DISPLAYHOLDS and RELEASEHOLDS the role must have access to view holds per this property record.

- ", Faculty, FacultyOverride" was removed from line 412.

```
all.curriculum.holdsAffected.roles=Advisor,
AdvisorOverride, FacultyAdvisor, Student, Faculty,
FacultyOverride
```

StudentSSB_configuration.example

The following has been added for platform upgrade and SAML support.

```
/**
*************************************************************************
*                                                                     *
*                SAML configuration
*
*        Un-comment the below code when authentication mode is saml
*
*************************************************************************/
/*grails.plugins.springsecurity.auth.loginFormUrl = '/saml/login'

grails.plugins.springsecurity.saml.afterLogoutUrl ='/logout/customLogout'

banner.sso.authentication.saml.localLogout='false'   // To disable single
logout set this to true,default 'false'.

grails.plugins.springsecurity.saml.keyManager.storeFile =
'classpath:security/<KEY_NAME>.jks'  // for unix File based Example:-
'file:/home/u02/samlkeystore.jks'

grails.plugins.springsecurity.saml.keyManager.storePass = 'test1234'

grails.plugins.springsecurity.saml.keyManager.passwords = [ 'banner-
<short-appName>-sp': 'test1234' ]  // banner-<short-appName>-sp is the
value set in EIS Service provider setup

grails.plugins.springsecurity.saml.keyManager.defaultKey = 'banner-
<short-appName>-sp'                   // banner-<short-appName>-sp is the
value set in EIS Service provider setup

grails.plugins.springsecurity.saml.metadata.sp.file = 'security/banner-
<Application_Name>-sp.xml'    // for unix file based Example:-'/home/u02/
sp-local.xml'

grails.plugins.springsecurity.saml.metadata.providers = [adfs: 'security/
banner-<Application_Name>-idp.xml'] // for unix file based Example: '/
home/u02/idp-local.xml'

grails.plugins.springsecurity.saml.metadata.defaultIdp = 'adfs'

grails.plugins.springsecurity.saml.metadata.sp.defaults = [

        local: true,

        alias: 'banner-<short-appName>-sp',
// banner-<short-appName>-sp is the value set in EIS Service provider
setup

        securityProfile: 'metaiop',

        signingKey: 'banner-<short-appName>-sp',
// banner-<short-appName>-sp is the value set in EIS Service provider
setup

        encryptionKey: 'banner-<short-appName>-sp',
// banner-<short-appName>-sp is the value set in EIS Service provider
setup
```

```
        tlsKey: 'banner-<short-appName>-sp',
// banner-<short-appName>-sp is the value set in EIS Service provider
setup
        requireArtifactResolveSigned: false,
        requireLogoutRequestSigned: false,
        requireLogoutResponseSigned: false
]*/
```

Advisor Self-Service Application

The following changes have been made for the Advisor Self-Service application.

banner_configuration.example

The following has been added or updated.

Email batch size

```
/**
*************************************************************************
*                The Mail Server's recipient limit.
*
*************************************************************************/
email.batch.size = 600
```

Photo path

```
/**
*************************************************************************
*            Administrative User DataSource Configuration
*
*************************************************************************/
banner {
    picturesPath = System.getProperty('base.dir') + '/test/images'

   // defaultPhoto = <Fully qualified path to the photo to use if no user
photo is available.>
}
```

bannerStudentAdvisorUI_configuration.example

The following changes were made in this file.

- "FacultyOverride without AdvisorOverride is not supported." was added at line 21.

  # FacultyOverride - user is a faculty and override is selected in SIAINST - applies to

Student Profile page only. **FacultyOverride without AdvisorOverride is not supported.**

- "hasReleasableHolds" was added at lines 30, 44, 46, 47, and 48.

  - #<columns> possible values: studentStatus, studentType, **hasReleasableHolds**, isPrimaryAdvisor, advisorType, program, major, standing, campus, studyLevel, admitType, admitTerm, catalogTerm,

  - #all.advisee.column.list.FacultyAdvisor=studentStatus, studentType, **hasReleasableHolds**, isPrimaryAdvisor, advisorType, program, major, campus, studyLevel, admitType, admitTerm, catalogTerm, minor, concentration, college

  - all.advisee.column.list.Advisor=program, major, standing, studentStatus, studentType, **hasReleasableHolds**, isPrimaryAdvisor, advisorType, campus, studyLevel, admitType, admitTerm, catalogTerm, minor, concentration, college, preferredEmailAddress, nonPreferredEmailAddress

  - all.advisee.column.list.FacultyAdvisor=program, major, standing, studentStatus, studentType, **hasReleasableHolds**, isPrimaryAdvisor, advisorType, campus, studyLevel, admitType, admitTerm, catalogTerm, minor, concentration, college, preferredEmailAddress, nonPreferredEmailAddress

  - all.advisee.column.list.AdvisorOverride=program, major, standing, studentStatus, studentType, **hasReleasableHolds**, isPrimaryAdvisor, advisorType, campus, studyLevel, admitType, admitTerm, catalogTerm, minor, concentration, college, preferredEmailAddress, nonPreferredEmailAddress

- The following was added at line 304.

  - # If Notes are enabled the Release Hold comment entered by the person releasing a hold is displayed in the Notes section to the author of the comment.

  - # Use the following property to control if in addition the roles AdvisorOverride and FacultyOverride can view the Release Hold Comment.

  - # You can add role of FacultyOverride in addition to AdvisorOverride or disable the Property.

    all.studentnotes.releaseholdnote.view=Profile
    all.studentnotes.releaseholdnote.roles=AdvisorOverride

- The following was added at line 396.

  #The Release Holds feature also has a dependency on this property, in addition to SOAFACS security rules for DISPLAYHOLDS and RELEASEHOLDS the role must have access to view holds per this property record.

- ", Faculty, FacultyOverride" was removed from line 412.

```
all.curriculum.holdsAffected.roles=Advisor,
AdvisorOverride, FacultyAdvisor, Student, Faculty,
FacultyOverride
```

StudentAdvisorSSB_configuration.example

The following has been added for platform upgrade and SAML support.

Added at line191.

```
if(authenticationProvider != 'default') {
    grails.plugins.springsecurity.failureHandler.defaultFailureUrl
= '/login/error'
}
```

Added at line 221.

```
/**
***************************************************************************
*                                                                         *
*               SAML configuration
*
*        Un-comment the below code when authentication mode is saml
*
***************************************************************************/
/*grails.plugins.springsecurity.auth.loginFormUrl = '/saml/login'

grails.plugins.springsecurity.saml.afterLogoutUrl ='/logout/customLogout'

banner.sso.authentication.saml.localLogout='false'   // To disable single
logout set this to true,default 'false'.

grails.plugins.springsecurity.saml.keyManager.storeFile =
'classpath:security/<KEY_NAME>.jks'  // for unix File based Example:-
'file:/home/u02/samlkeystore.jks'

grails.plugins.springsecurity.saml.keyManager.storePass = 'test1234'

grails.plugins.springsecurity.saml.keyManager.passwords = [ 'banner-
<short-appName>-sp': 'test1234' ]  // banner-<short-appName>-sp is the
value set in EIS Service provider setup

grails.plugins.springsecurity.saml.keyManager.defaultKey = 'banner-
<short-appName>-sp'                  // banner-<short-appName>-sp is the
value set in EIS Service provider setup

grails.plugins.springsecurity.saml.metadata.sp.file = 'security/banner-
<Application_Name>-sp.xml'    // for unix file based Example:-'/home/u02/
sp-local.xml'

grails.plugins.springsecurity.saml.metadata.providers = [adfs: 'security/
banner-<Application_Name>-idp.xml'] // for unix file based Example: '/
home/u02/idp-local.xml'

grails.plugins.springsecurity.saml.metadata.defaultIdp = 'adfs'

grails.plugins.springsecurity.saml.metadata.sp.defaults = [

    local: true,

    alias: 'banner-<short-appName>-sp',
// banner-<short-appName>-sp is the value set in EIS Service provider
setup

    securityProfile: 'metaiop',
```

```
        signingKey: 'banner-<short-appName>-sp',
// banner-<short-appName>-sp is the value set in EIS Service provider
setup
        encryptionKey: 'banner-<short-appName>-sp',
// banner-<short-appName>-sp is the value set in EIS Service provider
setup
        tlsKey: 'banner-<short-appName>-sp',
// banner-<short-appName>-sp is the value set in EIS Service provider
setup
        requireArtifactResolveSigned: false,

        requireLogoutRequestSigned: false,

        requireLogoutResponseSigned: false

]*/
```

Added at line 258.

```
/**
**********************************************************************
*                                                                    *
*                      Homepage URL                                  *
*          Un-comment the below code when configuring a homepage URL
*
**********************************************************************/
//grails.plugins.springsecurity.homePageUrl='URL to homepage'
```

Miscellaneous Enhancements

This section contains the following:

- New security control `DISPLAYALTPIN` rule

- Update to SORNOTE table

- Accessibility improvements

Enhanced security for alternate PIN display

The display of the student's alternate registration PIN on the Student Profile page has been enhanced with the creation of an additional security control. A new STVPROC process code and SOAFACS process rule code have been delivered to further control which advisors or faculty members can view the student's alternate registration PIN in the Registration Notices section of the page. The new `DISPLAYALTPIN` rule allows advisors or faculty members with a specific relationship to the student to view the PIN.

PIN display is configured using the existing `DISPALTTPIN` rule on GTVSDAX, existing settings in the `bannerStudentAdvisorUI_configuration .properties` file, and the new SOAFACS `DISPLAYALTPIN` process rule.

The existing GTVSDAX rule is as follows:

External Code	Internal Code	Internal Code Group	Description	Activity Date
Y/N	DISPALTPIN	FACWEB	Display Alt PIN for Advisee	Sysdate

The existing configuration file settings are:

```
all.registrationnotices.registrationPin.view=Profile
```

```
all.registrationnotices.registrationPin.roles=Advisor,
AdvisorOverride, FacultyAdvisor
```

New process code

The `DISPLAYALTPIN` process code is used on the Process Code Validation Page (STVPROC).

> **Note:** Data for STVPROC is delivered in a seed data script.

Process	Description	Check Order	Enforce Check Order	PIN Control Allowed	Sys Req
DISPLAYALTPIN	ALTPIN SP Registration Notices	Both	No	No	Yes

New process rule

The `DISPLAYALTPIN` process rule for Advisor and Faculty users is used on the Faculty/Advisor Process Rules Page (SOAFACS).

> **Note:** Data for SOAFACS is delivered in a seed data script.

Process	Desc	Check Order	Process Available on Self-Service	All Access	PIN Control	Relation ship	Primary	Attribute Type Checking	Sys Req
DISPLAYALTPIN	ALTPIN SP Registration Notices	Advisor	Yes	No	No	No	No	No	Yes
DISPLAYALTPIN	ALTPIN SP Registration Notices	Faculty	Yes	No	No	No	No	No	Yes

New scripts

The following scripts are used with this enhancement.

> **Note:** Unless otherwise noted, new scripts are run as part of the upgrade process for a release.

Script	Result
ssppsorfacsi.sql	Adds seed data to SOAFACS
ssppsstvproci.sql	Adds seed data to STVPROC

New table index

A Key Index has been added to the SORNOTE table to assist with performance.

    SORNOTE_KEY_INDEX2 (SORNOTE_PIDM)

Changes were delivered in the `sornote_080900_01.sql` script in the Banner Student 8.9 release.

Accessibility improvements

The following updates are delivered for accessibility.

- Titles have been added to all page headers.

- The Student Profile page contains enhanced headers for use with screen readers.

- The Advisee Search page contains improved navigation for access to radio buttons and tool tips.

- The Advisee Listing page grid has been modified to be a single table. This allows the use of keyboard navigation to move across the rows between the frozen and unfrozen columns.

  Users must toggle between navigation for read only/display mode and data entry/action mode in the grid. Each actionable cell now has only one associated actionable item.

  Use Tab to move through the grid and land on the actionable items. Use Escape to return to read only mode.

- The Advisee Listing page access to the Student Card and the Student Profile page has been updated.

  The Student Card is now only accessed by clicking on the student's photo. The link from the **Name and ID** field to the Student Card is no longer available.

  The Student Profile page is accessed by selecting the **View Profile** link under the **Name and ID** field. This field contains one link where it used to contain two links.

Resolutions

The *Banner Student Advising Student Profile Resolutions Report 9.3* is a companion to this release guide. The Resolutions Report provides summary information about the change requests that are resolved in Banner Student Advising Student Profile 9.3. Use the following steps to access the Resolutions Report on the Ellucian Support Center:

1. Go to the Ellucian Client Support web site: http://www.ellucian.com/Solutions/Ellucian-Client-Support.

2. Click the **Ellucian Hub** button to log in to the Ellucian Hub.

3. Choose the Ellucian Download Center application.

   or

   Choose the Ellucian Support Center application.