CalSAWS

California Statewide Automated Welfare System

Design Document

CA-234742

CMS Certification Recommendations (OCB/ORR) – e-Verifications Phase I

CalSAWS	DOCUMENT APPROVAL HISTORY	
	Prepared By	Maria Feliciano
	Reviewed By	Appalaraju Indala, Tisha Mutreja, Renee Gustafson, Maksim Volf, Jeyasundari Murugan, Geetha Ramalingam, Harish Nagendraprasad

DATE	DOCUMENT VERSION	REVISION DESCRIPTION	AUTHOR
3/4/2024	1.0	Draft design document	Maria Feliciano
4/22/2024	1.1	Reviewed with Dev Team	Maria Feliciano
4/24/2024	1.2	Reviewed with Medi-Cal and Batch/Interfaces Analysts	Maria Feliciano
5/15/2024	1.3	Reviewed with Medi-Cal Analysts	Maria Feliciano
5/17/2024	1.4	Updated per feedback from Medi-Cal Analysts to remove e-data approach and go with auto-create Vital Stats from Solicited DER/DER-U with Reviewed Status.	Maria Feliciano
05/20/2024	1.5	Added details to populate Vital Stats with 'Retrieve e-Verification' button. Updated eHIT and MC Rules recommendations. Added Data Change and journal recommendations.	Renee Gustafson
05/28/2024	1.6	Updated Page Mockups and document sections based on feedback received by BAs	Tisha Mutreja
6/5/2024	1.7	Reviewed with Medi-Cal/CMSP Committee	Maria Feliciano
6/6/2024	1.8	Updated document to delete Negative Action EDBC page from warning message recommendation 2.4	Maria Feliciano
6/6/2024	1.9	- Updated recommendation 2.4.2.2 - Added Recommendation 2.4.2.3 - Updated current design for warning messages - Updated Document Numbering and Formatting	Tisha Mutreja
6/10/2024	1.10	Updated Assumptions	Maria Feliciano
6/10/2024	1.11	- Removed Recommendation 2.4.2.3, and current design with respect to the recommendation - Added section 2.11 Automated	Tisha Mutreja

		Regression Test	
6/13/2024	1.12	Design Clarification for Recommendations 2.1.3.1, 2.2.3.5 and 2.10.2.a: Update language to include 'the oldest DER based on date received' Design Clarification for Recommendation 2.2.3.5 and 2.10.2.b: Deleted Transaction history recommendation as it is current functionality. Design Clarification for Recommendation 2.5.2.b and 2.10.2.b: Create the Vital Statistics and Verification records with data the same as Recommendation 2.2.3.5.a & 2.2.3.5.b, respectively.	Maria Feliciano
6/19/2024	1.13	Updated Current Design section for Validation message Design Clarification for Recommendations 2.2.3.5.a & 2.2.3.5.b. Design Clarification for Recommendation 2.4.2 to add update for Validation message criteria	Maria Feliciano
6/20/2024	1.14	Corrected 1.2.3 and corrected typo for Overview of recommendations 1.3.5.a. US Citizen Ind = Yes will be sent in eHIT if Vital Stats 3E exists. Updated Current Design for exception for Run EDBC and Manual EDBC page messages. Added mockup for Manual EDBC. Added note to 2.3.3.1. Clarified criteria in 2.4.2.2 for Run EDBC page. Added 2.4.2.3 for Manual EDBC and 2.4.2.4 for both Run EDBC and Manual EDBC.	Renee Gustafson, Tisha Mutreja
7/2/2024	1.15	Design Clarification: - Updated recommendation 2.4.2.2 and 2.4.2.3 to remove Identity verification from Message Trigger Conditions - Updated recommendation 2.9.2.1 to match validation criteria in 2.4.2.3	Tisha Mutreja

7/16/2024	1.16	Design Clarification: - Updated Journal criteria for eHIT and DCR	Tisha Mutreja

Table of Contents

1 OVERVIEW

The purpose of this SCR is to limit the amount of intervention required by a caseworker to ensure real time Modified Adjusted Gross Income (MAGI) applications are completed following electronic verification received via California Healthcare Eligibility, Enrollment, and Retention System (CalHEERS) Federal Data Services Hub.

1.1 Current Design

CalSAWS leverages CalHEERS for e-verification of attested MAGI application data and to provide a MAGI Medi-Cal eligibility determination.

The CalSAWS data collection pages of attested application data are shared among Non-MAGI Medi-Cal and other CalSAWS-managed programs; therefore, CalSAWS does not have a one-to-one mapping of attested MAGI application data to the CalSAWS data collection pages. This sometimes requires the user to interpret the attested MAGI application data to fit into the CalSAWS data collection pages or to request more information from the applicant/beneficiary. The lack of a one-to-one mapping can also cause the CalSAWS data collection to be mapped to something different in CalHEERS when the user sends a new Eligibility Determination Request (EDR) to CalHEERS for a new MAGI Medi-Cal eligibility determination.

The CalSAWS Medi-Cal EDBC rules use the verification status specified in the CalSAWS data collection pages to determine Non-MAGI Medi-Cal eligibility and to communicate Administrative Verifications to CalHEERS in the eHIT interface. The CalSAWS data collection pages do not have an option for the user to specify the data collection is e-Verified through CalHEERS, so the user sometimes requests the applicant to provide verification, or the user manually indicates the Verification Status as 'Verified'. If the user marks the Verification Status as 'Verified', the future EDRs will show this data collection as Administratively Verified by SAWS. Also, the CalSAWS EDBC rules do not consider a CalHEERS-provided e-Verification as 'Verified' for Non-MAGI Medi-Cal. With ACWDL 20-17, DHCS provided guidance that CalHEERS-provided e-Verifications can be used for Non-MAGI Medi-Cal eligibility determinations except for the e-verification for Income.

CalSAWS eHIT logic will send an individual in an EDR as "US Citizen Indicator = Yes" based on the following:

- Individual Demographics Birth Country is either US or US territory, or
- Individual Demographics Birth Country is not US or US Territory, and their Citizenship Immigration document is for a Naturalized or Derived US Citizen

Otherwise, send "US Citizen Indicator = No"

The CalSAWS eHIT logic does not consider the Vital Stats records for determining the US Citizen Indicator value in an EDR.

Per CalHEERS Interpretation of MEDS Interface Definition Document (IDD), CalHEERS sends 3E 'U.S. Citizenship verified via CalHEERS Federal Data Services Hub (FDSH) Social Security Administration (SSA) Composite Service to MEDS in the data elements for US Citizen Verification (DE 2081) and Identity Verification (DE 2081) when an individual is e-verified as a US Citizen through the Federal Services Hub: Social Security Administration.

There is no Birth Country element in eHIT so there is no e-data, and this leaves the user to have to determine the Individual Demographics, 'Birth Country' by other means.

CalSAWS eHIT logic will send an individual as "US Citizenship Admin Verification = Yes" based on the following:

- Individual Demographics 'Birth Country' is either US or US territory and the Birth Country verification status is 'Verified', or
- Individual Demographics 'Birth Country' is not US or US Territory, and their Citizenship Immigration document is for a Naturalized or Derived US Citizen, and the Citizenship Immigration record status is 'Verified'

CalSAWS eHIT logic will send an individual as "US Citizenship Admin Verification = No" based on the following:

- Individual Demographics 'Birth Country' is either US or US territory and the Birth Country verification status is 'Refused', or
- Individual Demographics 'Birth Country' is not US or US Territory, and either
    - their Citizenship Immigration document effective for the EDR benefit month is for a Naturalized or Derived US Citizen, and the Citizenship Status Detail Verification status is 'Refused', or
    - their Citizenship Status Detail record has 'Pending' Verification status AND the 'Medi-Cal Reasonable Opportunity Period' "Expired" value is 'Yes', or
    - there is no Citizenship Status Detail record effective for the EDR benefit month

If neither of the above are met, then eHIT logic will not send anything for US Citizenship Admin Verification.

CalSAWS MEDS logic will send the Values for Data Elements US Citizen Verification (DE 2086), Identity Verification (DE 2081) and Citizen/Alien Indicator (DE 2009) first based on Citizenship records, then based on Vital Statistics, then based on Birth Country in Individual Demographics.

CalSAWS Non-MAGI Medi-Cal EDBC rules will determine an individual as a US Citizen first based on the Individual Demographics 'Birth Country'. If the Birth Country is US or US Territory, then the rules check the Vital Stats US Citizenship Document Type and Verification Status. If the Individual Demographics 'Birth Country' is not US or US Territory, then the rules check the Citizenship Immigration

Document Type, Section Code and other factors in the Citizenship Immigration record itself; Vital Stats is not used when the Birth Country is not US or US Territory.

CalSAWS EDBC rules for all other EDBC-managed programs also use the Individual Demographics 'Birth Country' and Citizenship Immigration Document information to determine a US Citizen. No other EDBC program uses Vital Stats in their EDBC rules.

Individual Demographics has Birth Country option of 'Unknown' which is not considered as US or US Territory in all EDBC programs.

The Run EDBC and Manual EDBC pages display a message alerting the user that the Country of Birth/High-dated Immigration record is missing for a person and therefore Medi-Cal EDBC will process in MAGI Only mode when 'Birth Country' is not in 'US or US Territory' and there is no Citizenship record.

Currently, if any individual on the Medi-Cal program has Birth Country as 'Non-US or US Territory' with No Citizenship record and has Vital Statistics record for US Citizenship and Identity as 'Exempt from Verification', then all individuals with Birth Country as 'Non-US or US Territory' with No Citizenship record must also have Vital Statistics record for US Citizenship and Identity as 'Exempt from Verification' or else Run EDBC page and Manual EDBC page will trigger below messages -

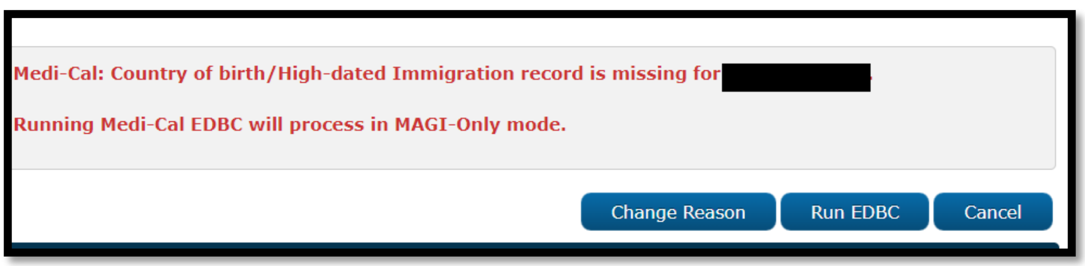


Figure 1-Run EDBC page warning message for Birth Country

Figure 2-Manual EDBC page validation message for Birth Country

CalSAWS will skip a program from Batch EDBC processing if an open Medi-Cal individual is missing a Birth Country in Individual Demographics except when there is a Citizenship record.

1.2 Requests

The variance in attested MAGI data collection and the use of MAGI e-Verifications creates an opportunity for user error when processing the case, potential issues with timeliness, incorrect eligibility determinations, and limits the amount of real-time or near real-time eligibility determinations. Users should not have to manually gather or verify data when an e-verification has taken place.

CalHEERS can electronically verify a person's US Citizenship and Identity from SSA via the Federal Services Hub.

1. Update CalSAWS to automatically create a Vital Statistics record for US Citizenship and Identity when an individual is e-Verified by CalHEERS Federal Data Services Hub (FedHub) Social Security Administration (SSA).
2. Update CalSAWS Non-MAGI Medi-Cal EDBC rules to use the new Vital Statistics record for US Citizens.
3. Update eHIT logic to consider the Vital Statistics US Citizenship Verification record for '3E' to send US Citizen Indicator = Y; there are no changes to eHIT logic when determining an Admin Verification of US Citizenship in an EDR.
4. Update e-Data for the Individual Demographics page to populate Birth Country 'Unknown' for Unsolicited DERs.

1.3 Overview of Recommendations

1. Update the Vital Statistics List page to display the CalHEERS Verifications received for US Citizenship.
2. Update the Vital Statistics page to document when Identity and US Citizenship is e-Verified from CalHEERS via FedHub SSA by adding a new value: '3E U.S. Citizenship verified via CalHEERS FDSH: SSA'.
3. Update the e-data functionality for the Individual Demographics page to default 'Birth Country' as 'Unknown' from an unsolicited DER from CalHEERS.
4. Update the Medi-Cal EDBC rules to consider Vital Statistics for US Citizenship check when the Individual Demographics Birth Country is not 'US' or 'US Territory'.
5. Update eHIT logic to create Vital Statistics records when a DER with US Citizenship e-verified by source of FDSH: SSA is received.
    a. Update logic to send the US Citizen indicator = Yes when there is a Vital Statistics record with '3E U.S. Citizenship verified via CalHEERS FDSH: SSA'.
6. Regression test MEDS transactions to ensure new value of 3E is populated correctly in outbound files for DE 2081, DE2086 and DE 2009.
7. Update the Batch EDBC skip logic to add an exception if the program is Medi-Cal and there is a Vital Statistics '3E' record.

8. With a one-time data change, create Vital Statistics records for individuals who do not already have a Vital Statistics record and CalSAWS has received a DER with an e-verification for US Citizenship from CalHEERS FDSH: SSA.

1.4 Assumptions

1. CalSAWS will continue to use the CalSAWS attested data collection and Verification Status in Medi-Cal EDBC.
2. There will be no e-Data changes for BenefitsCal or ICT.
3. Fields not mentioned to be modified within the online pages 'Description of Changes' will retain their current functionality.
4. Updates to utilize Birth Country ‘Unknown’ apply to the Medi-Cal Program only. Where other programs utilize birth country, the user is expected to determine the value and update the Individual Demographics page accordingly.
5. There will be no changes to existing Reports.
6. There will be no changes to existing CalSAWS eHIT logic to send US Citizenship Admin Verification to CalHEERS.
7. This SCR does not affect Foster Care, Adoption Assistance Program or KinGAP.

2 RECOMMENDATIONS

2.1 Vital Statistics List Page

2.1.1 Overview

Update the Vital Statistics List page to display the CalHEERS Verifications received for US Citizenship.

2.1.2 Vital Statistics List Page Mockup

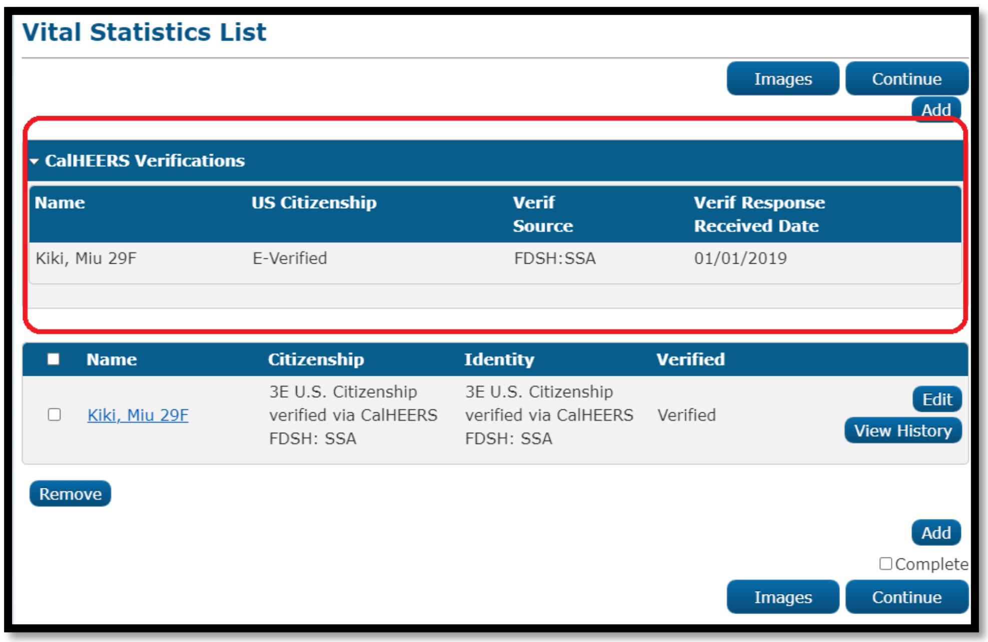


Figure 2.1.2.1 – Vital Statistics List Page with E-Verified record

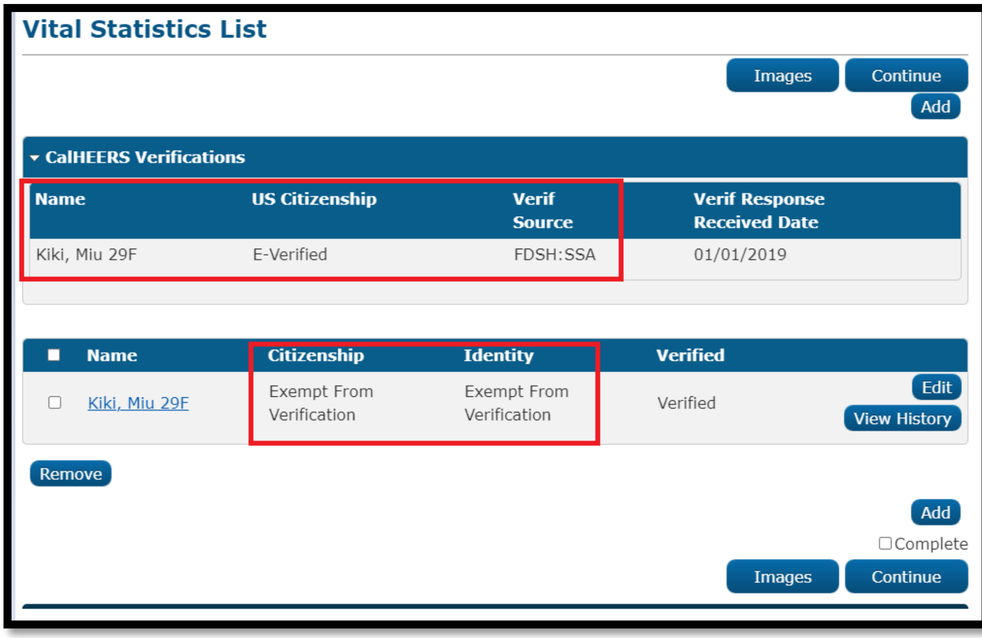


Figure 2.1.2.2 – Vital Statistics List Page with existing record

2.1.3 Description of Changes

1. Add a 'CalHEERS Verifications' expandable section to the Vital Statistics List page to display 'US Citizenship' e-verification information from CalHEERS.

   If the individual is e-verified as a US Citizen by FDSH: SSA, then display the values based on the oldest DER based on date received where this individual is e-verified as a US Citizen by FDSH: SSA. There will be no hyperlink to the specific DER.

Field Name	Description
Name	Displays the standard CalSAWS name format "Last Name, First Name <age in years> and the Gender (F/M)
US Citizenship	the e-verification 'Verif Status' of US Citizenship from the DER
Verif Source	the e-verification 'Verif Source' from the DER
Verif Response Received Date	'Verif Response Received Date' from the DER

   If the individual is not e-verified as a US Citizen by FDSH: SSA, then do not display the individual in the section.

Note: The CalHEERS Verification section only displays e-verifications from FDSH:SSA and may not match the actual Vital Statistics record for an individual if the Individual has Vital Statistics with other values.

2.1.4 Page Location

- **Global: Eligibility**
- **Local: Customer Information**
- **Task: Vital Statistics**

2.1.5 Security Updates

No Updates

2.1.6 Page Mapping

Update to include the new fields added to the page.

2.1.7 Accessibility

No Updates

2.1.8 Page Usage/Data Volume Impacts

No Updates

2.2 Vital Statistics Detail Page

2.2.1 Overview

Update the Vital Statistics page to document when Identity and US Citizenship is e-Verified from CalHEERS via FedHub SSA by adding a new value: '3E U.S. Citizenship verified via CalHEERS FDSH: SSA'.

2.2.2 Vital Statistics Detail Page Mockup

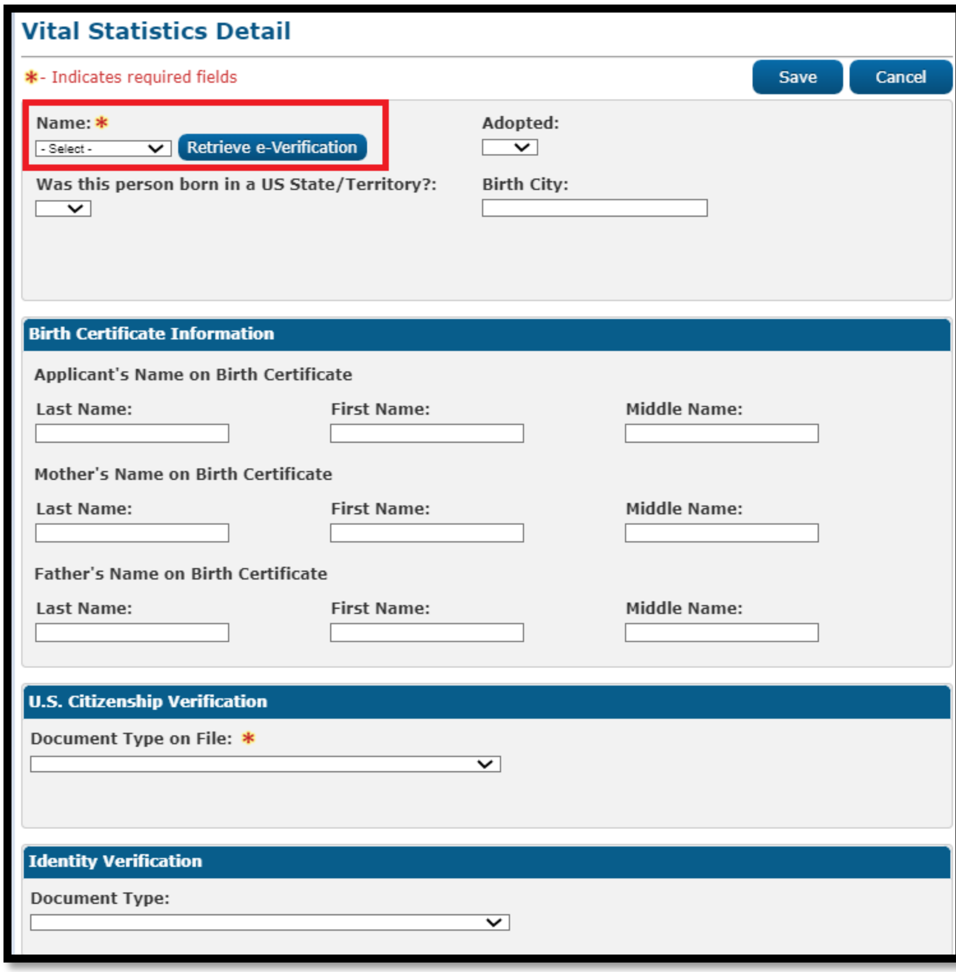


Figure 2.2.2.1 – Vital Statistics Detail Page with Retrieve e-Verification Functionality

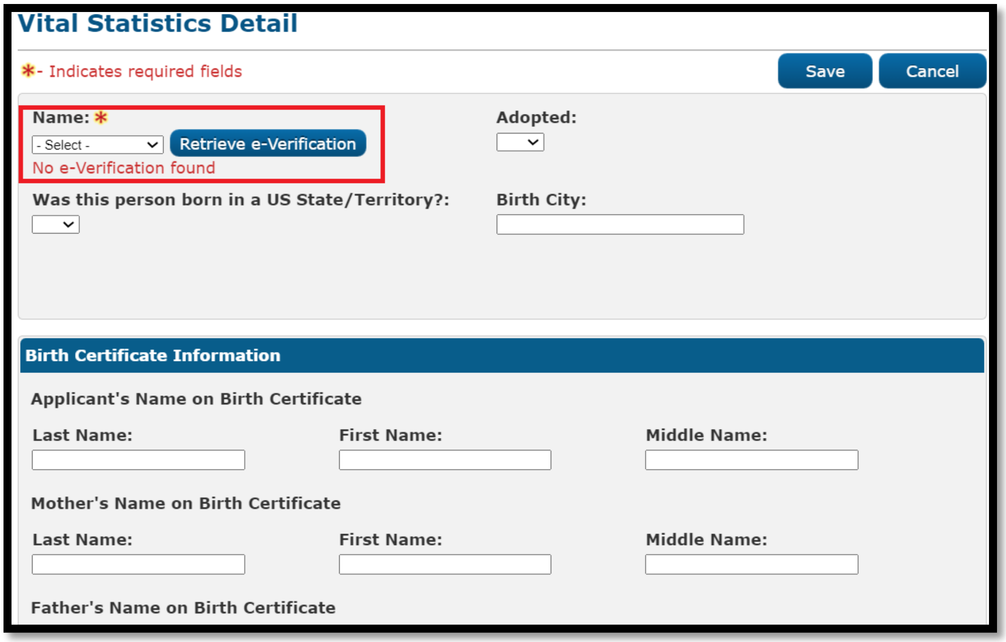


Figure 2.2.2.2 – Vital Statistics Detail Page with Retrieve e-Verification when No e-Verification is found

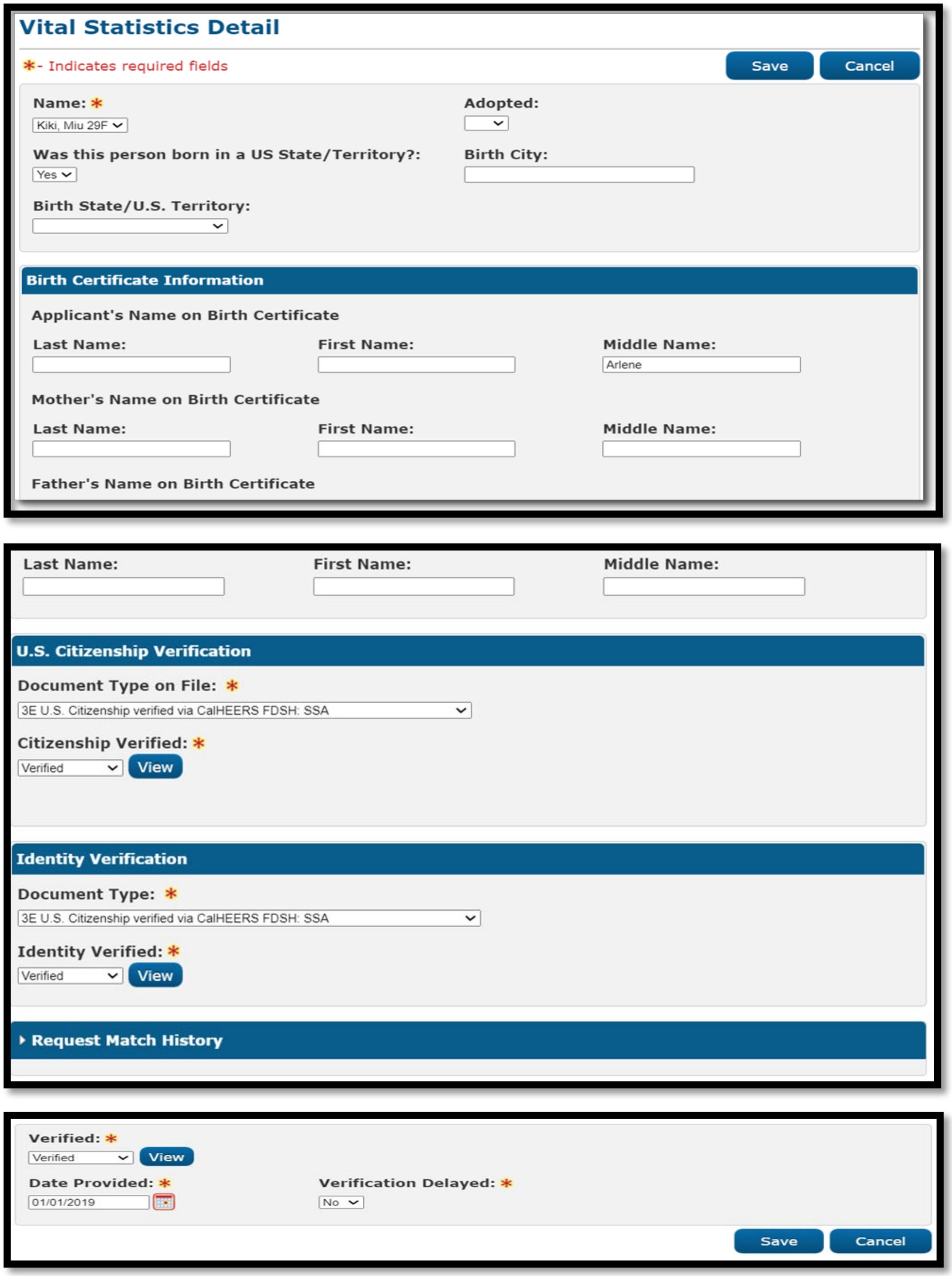


Figure 2.2.2.3 – Vital Statistics Detail Page with New U.S. Citizenship Verification and Identity Verification Type option for '3E U.S. Citizenship verified via CalHEERS FDSH: SSA'

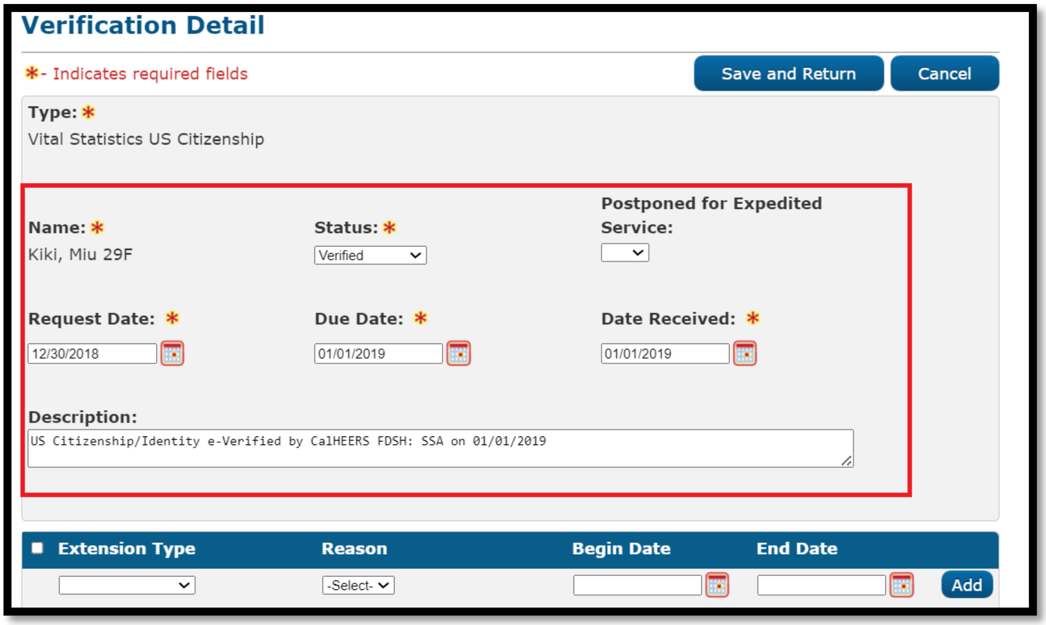


Figure 2.2.2.4 – Verification Detail Page with New U.S. Citizenship Verification

2.2.3 Description of Changes

1. Add a new Document Type option for both US Citizenship Verification and Identity Verification on the Vital Statistics page as follows (CT_2400):

Field	Value
Code Num Identif	TBD
Short Decode Name	U.S. Citizenship verified via CalHEERS FDSH: SSA
Long Decode Name	U.S. Citizenship verified via CalHEERS Federal Data Services Hub Social Security Administration Composite Service
MEDS Codes	3E
US Citizenship	Y
Identity	Y
Citizenship Tier	3
Doc Num Identif Required	N
MEDS Birth Certificate Req	N
EICT Doc Type	<Blank>

Note: The current Vital Statistics page functionality will display the new Document Type as '3E U.S. Citizenship verified via CalHEERS FDSH: SSA' by prepending the MEDS Code to the Short Decode Name.

2. Update the Vital Statistics page to display the U.S. Citizenship Verification 'Document Type on File' and Identity Verification 'Document Type' drop list sorted by the MEDS code, then by alpha if there is no MEDS code.
3. Update the Vital Statistics page to not require Birth Certificate Information, nor a Document Number for either the US Citizenship Verification or Identity Verification when the new '3E U.S. Citizenship verified via CalHEERS FDSH: SSA' Document Type is selected in the respective section.
4. Update the Vital Statistics page to default Identity Verification Document Type to '3E U.S. Citizenship verified via CalHEERS FDSH: SSA' when '3E U.S. Citizenship verified via CalHEERS FDSH: SSA' is selected as the US Citizenship Verification Document Type.
    a. Do not grey out the Identity Document Type and allow other Identity Document Types to be specified if the user chooses to change it.
5. For Vital Statistics in create mode, add a new button named, 'Retrieve e-Verification' to the right of the 'Name:' dropdown.
    a. **Vital Statistics**: After the user selects the individual's name from the dropdown and clicks the 'Retrieve e-Verification' button, retrieve the oldest DER based on date received where the individual has US Citizenship e-verification from Source 'CalHEERS FDSH: SSA' and populate the page as follows:

Field Name	Value
Was this person born in a US State/Territory?	Yes
US Citizenship Verification Document Type on File:	3E U.S. Citizenship verified via CalHEERS FDSH: SSA
Citizenship Verified:	'Verified'
Identity Verification Document Type:	3E U.S. Citizenship verified via CalHEERS FDSH: SSA
Identity Verified:	'Verified'
Verified	'Verified'
County of Record	County of the CalSAWS case in the DER

Field Name	Value
Date Provided:	'Verif Response Received Date' in the DER. For older DERs, the 'Verif Response Received Date' may be Null. If so, use the 'Initiated Date' of the DER instead. This is equivalent to the BRE Run Date in CalHEERS.
Verification Delayed:	'No'

All other fields that display on the Vital Statistics page that are not specified in the table above will not be populated from the 'Retrieve e-Verification' functionality.

b. **Verification Detail:** Create a Verification Detail record to support the 'Verified' Status for each of the 'Vital Statistics US Citizenship' and 'Vital Statistics Identity' Verification Types and as a whole 'Vital Statistics' verification when the Vital Statistics record is created.

Section Header	Description
Type	'Vital Statistics US Citizenship' (CT170_61), 'Vital Statistics Identity' (CT170_62), and 'Vital Statistics' (CT170_60)
Status	'Verified'
Request Date	'Verif Response Received Date' in the DER For older DERs, the 'Verif Response Received Date' may be Null. If so, use the 'Initiated Date' of the DER instead. This is equivalent to the BRE Run Date in CalHEERS.
Due Date	10 days after Request Date based on current functionality
Date Received	'Verif Response Received Date' in the DER
Description	US Citizenship/Identity e-Verified by CalHEERS FDSH: SSA on <Verif Response Received Date>

All other fields that display on the Verification Detail page that are not specified in the table above will not be populated from the 'Retrieve e-Verification' functionality.

c. After the user selects the individual's name from the dropdown and clicks the 'Retrieve e-Verification' button, if there is no DER with the person e-verified as US Citizen via FDSH: SSA, display 'No e-verification found' message below to the button.

2.2.4 Page Location

- **Global: Eligibility**
- **Local: Customer Information**
- **Task: Vital Statistics**

2.2.5 Security Updates

No Updates

2.2.6 Page Mapping

No Updates

2.2.7 Accessibility

No Updates

2.2.8 Page Usage/Data Volume Impacts

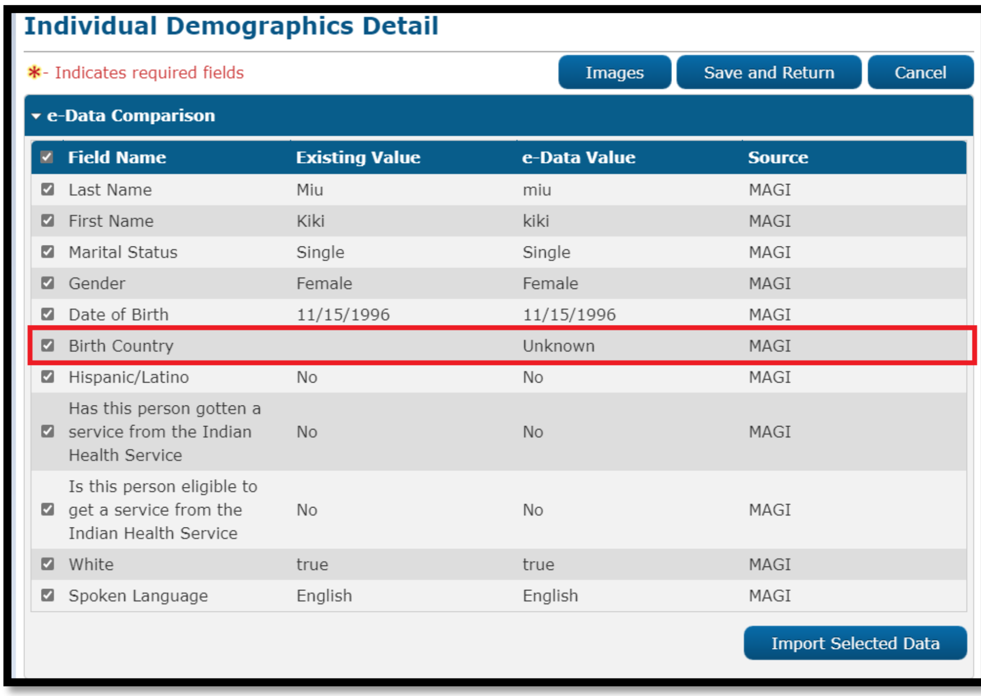
No Updates

2.3 Individual Demographics Page

2.3.1 Overview

Update the e-data functionality for the Individual Demographics page to default 'Birth Country' as 'Unknown' from an unsolicited DER from CalHEERS.

2.3.2 Individual Demographics Detail Page Mockup



Individual Demographics Detail

*- Indicates required fields

Images | Save and Return | Cancel

▾ e-Data Comparison

Field Name	Existing Value	e-Data Value	Source
Last Name	Miu	miu	MAGI
First Name	Kiki	kiki	MAGI
Marital Status	Single	Single	MAGI
Gender	Female	Female	MAGI
Date of Birth	11/15/1996	11/15/1996	MAGI
Birth Country		Unknown	MAGI
Hispanic/Latino	No	No	MAGI
Has this person gotten a service from the Indian Health Service	No	No	MAGI
Is this person eligible to get a service from the Indian Health Service	No	No	MAGI
White	true	true	MAGI
Spoken Language	English	English	MAGI

Import Selected Data

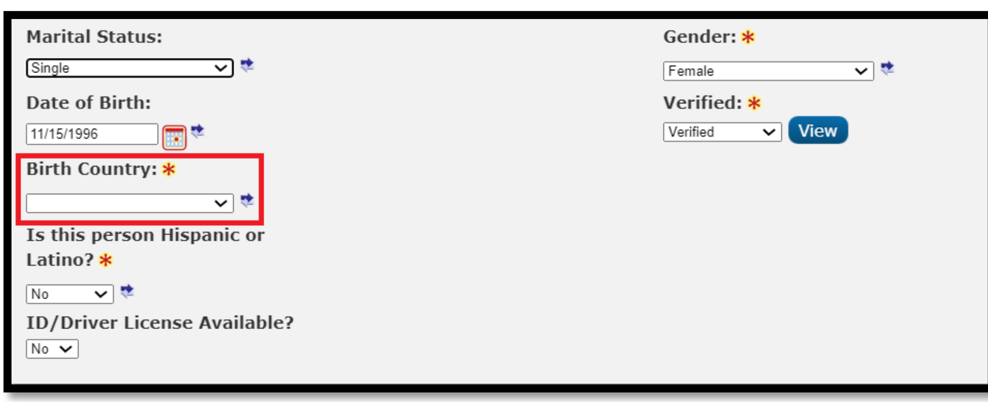


Figure 2.3.2 – Individual Demographics Detail Page

2.3.3 Description of Changes

1. Update existing e-data functionality on Individual Demographics Detail Page to default ‘Birth Country’ to ‘Unknown’ for an Unsolicited DER when the individual is e-verified as a US Citizen by FDSH: SSA.

   Note: The e-data value will not automatically populate in the “Birth Country” field on the Individual Demographics Detail page; the user can choose to import the ‘Unknown’ birth country or select it from the e-data arrow as displayed in the screenshot above (Figure 2.3.2 – Individual Demographics Detail Page)

2.3.4 Page Location

- **Global: Eligibility**
- **Local: Customer Information**
- **Task: Individual Demographics Detail**

2.3.5 Security Updates

No Updates

2.3.6 Page Mapping

Update to include the new fields added to the page.

2.3.7 Accessibility

No Updates

2.3.8 Page Usage/Data Volume Impacts

No Updates

2.4 Eligibility Rules Updates

2.4.1 Overview

Update the Medi-Cal EDBC rules to consider Vital Statistics for US Citizenship check when the Individual Demographics Birth Country is not 'US' or 'US Territory'. Update the existing criteria for displaying the warning messages in the Run EDBC page, and Manual EDBC page.

2.4.2 Description of Changes

1. Update Non-MAGI Medi-Cal EDBC rules to consider an individual as US Citizen if the individual has below:
    a. 'Birth Country' on the Individual Demographics page `is not US or US Territory, including when 'Birth Country' is set to 'Unknown' or 'Undeclared', and
    b. Vital Statistics "U.S. Citizenship Verification" Document Type on File as '3E U.S. Citizenship verified via CalHEERS FDSH: SSA' with "Citizenship Verified" as verified, otherwise use the existing Medi-Cal Citizenship criteria.
    Note: US Citizenship verification is not considered a mandatory verification for Medi-Cal.
2. Add to the existing exception criteria for the Warning message: "Medi-Cal: Country of birth/High-dated Immigration record is missing for <Person Name> Running Medi-Cal EDBC will process in MAGI-Only mode." to not display on Run EDBC page for an individual that meets all of the below criteria:
    a. Is an Active/Pending MEM
    b. 'Birth Country' on the Individual Demographics page is not 'US or US Territory', including when 'Birth Country' is set to 'Unknown' or 'Undeclared', and
    c. No Citizenship record exists on the Citizenship page
    d. Vital Statistics record has below:
        i. "U.S. Citizenship Verification" Document Type on File as '3E U.S. Citizenship verified via CalHEERS FDSH: SSA' with "Citizenship Verified" as verified
3. Add to the existing exception criteria for the validation message: "Medi-Cal: Country of birth/High-dated Immigration record is missing for <Person Name> EDBC cannot be run for this program." to not display on Manual EDBC page for an individual that meets all of the below criteria:
    a. Is an Active/Pending MEM
    b. 'Birth Country' on the Individual Demographics page is not 'US or US Territory', including when 'Birth Country' is set to 'Unknown' or 'Undeclared', and
    c. No Citizenship record exists on the Citizenship page
    d. Vital Statistics record has below:

i. "U.S. Citizenship Verification" Document Type on File as '3E U.S. Citizenship verified via CalHEERS FDSH: SSA' with "Citizenship Verified" as verified

4. Update messages in recommendations 2.4.2.2 and 2.4.2.3 to no longer require all individuals on the Medi-Cal program with Birth Country as 'Non-US or US Territory', and with No Citizenship record to have Vital Statistics record for US Citizenship and Identity as 'Exempt from Verification', if at least one individual on the same Medi-Cal program has Birth Country as 'Non-US or US Territory' with No Citizenship record and has Vital Statistics record for US Citizenship and Identity as 'Exempt from Verification'; the messages will trigger per individual.

2.4.3 Programs Impacted

Medi-Cal

2.4.4 Performance Impacts

N/A

2.5 eHIT

2.5.1 Overview

Update eHIT logic to create Vital Statistics records when a DER with US Citizenship e-verified by source of FDSH: SSA is received. Update logic to send the US Citizenship indicator when there is a Vital Statistics record with US Citizenship e-verified by source of FDSH: SSA.

2.5.2 Description of Change

1. Update eHIT logic to create a new Vital Statistics record for an individual on the CalSAWS case as follows:
   a. **Criteria:**
      i. the individual does not have a Vital Statistics record, and
      either of the following are true:
         1. a Solicited DER is received after the implementation of this SCR with the individual's US Citizenship e-verified by Source = "FDSH:SSA" (CT10592_FDS), or
         2. an Unsolicited DER is received with the individual's US Citizenship e-verified by Source = "FDSH:SSA" (CT10592_FDS) and after the implementation of this SCR the user updates and saves the DER status from 'In Process' to 'Reviewed'

b. **Data**: Create the Vital Statistics and Verification records with data the same as Recommendation 2.2.3.5.a & 2.2.3.5.b, respectively.

c. **Journal:**
Create one journal per case with an open Medi-Cal program when a Vital Statistics record is created by the CalHEERS WebService with the following:
i. Do not create Journal for hidden individuals

Journal Category: Interfaces
Journal Type: Activity
Short Description: US Citizenship/Identity e-Verified by FDSH:SSA
Long Description: A Vital Statistics record for US Citizenship/Identity was created based on the electronic verification from CalHEERS via FDSH: SSA for the following individuals:

< Last Name, First Name <age in years> and the Gender (F/M)>
< Last Name, First Name <age in years> and the Gender (F/M)>

CalHEERS Case: <CalHEERS Case Number>
CalSAWS Case: <CalSAWS Case Number>
County: <County>

DER ID Reference: <DER ID>
DER Initiated Date: <Initiated Date>

2. Update CalSAWS eHIT logic that sends an individual in an EDR as "US Citizen Indicator = Yes" when there is an e-verified Vital Statistics US Citizenship 'Document Type on File' = '3E' with 'Vital Statistics US Citizenship' Verification Status = Verified.
Technical Note: CalSAWS existing eHIT logic to send US Citizenship Admin Verification 'Yes/No' to CalHEERS will not change when Vital Statistics e- verified record with US Citizenship 'Document Type on File' = '3E' and 'Vital Statistics US Citizenship' Verification Status = Verified exists.

2.5.3 Interface Partner

CalHEERS

2.5.4 eHIT Schema Version

eHIT Schema Version 20

2.6 Batch - Regression test MEDS outbound Data Element (DE) 2081

2.6.1 Overview

CalSAWS sends DE 2081 (CITIZENSHIP_DOC_TYPE) in the outbound file to MEDS.

The overview of the current logic to populate DE 2081 is as follows.

1. If the Vital Statistics Citizenship Document Type is 'Exempt' or 'Unable to obtain', look up the Vital Statistics Citizenship Document Type Reasons (CT2401) and send the MEDS Code in CT 2401 - refer table 1.
2. If the Vital Statistics Citizenship Document Type is other than above condition, look up the Vital Statistics Citizenship Document Type (CT2400) and send the MEDS Code in CT 2400 - refer table 1.
    a. If the look up value is 2Z, do not send the value.

2.6.2 Description of Change

1. The logic will populate DE 2081 of 3E for the new document type. The logic will automatically use the Code detail value in CT 2400.
   DE 2081 is in the following MEDS transactions:
    a. MEDS AP19

2.6.3 Execution Frequency

N/A.

2.6.4 Key Scheduling Dependencies

N/A.

2.6.5 Counties Impacted

All CalSAWS Counties.

2.6.6 Category

Core-Off Prime.

2.6.7 Data Volume/Performance

N/A.

2.6.8 Failure Procedure/Operational Instructions

Batch Support Operations staff will evaluate transmission errors and failures and determine the appropriate resolution (i.e., manually retrieving the file from the directory and contacting the external partner if there is an account or password issue, etc…)

2.7 Batch - Regression test MEDS outbound DE 2086

2.7.1 Overview

CalSAWS sends DE 2086 (IDENTITY_DOC_TYPE) in the outbound file to MEDS.

The overview of the current logic to populate DE 2086 is as follows.

1. If the Vital Statistics Identity Document Type is 'Exempt' or 'Unable to Obtain', look up the Vital Statistics Identity Document Type Reason (CT2401) and send the MEDS Code in CT 2401 - refer table 1.
2. If the Vital Statistics Identity Document Type is other than above, look up the Vital Statistics Identity Document Type (CT2400) and send the MEDS Code in CT 2400 - refer table 1.
    a. If the look up value is 2Z, do not send the value.

2.7.2 Description of Change

1. The logic will populate DE 2086 of 3E for the new document type. The logic will automatically use the Code detail value in CT 2400.
   DE 2086 is in the following MEDS transactions:
    a. MEDS AP19

2.7.3 Execution Frequency

N/A.

2.7.4 Key Scheduling Dependencies

N/A.

2.7.5 Counties Impacted

All CalSAWS Counties.

2.7.6 Category

Core-Off Prime.

2.7.7 Data Volume/Performance

N/A.

2.7.8 Failure Procedure/Operational Instructions

Batch Support Operations staff will evaluate transmission errors and failures and determine the appropriate resolution (i.e., manually retrieving the file from the directory and contacting the external partner if there is an account or password issue, etc…)

2.8 Batch - Regression test MEDS outbound DE 2009

2.8.1 Overview

CalSAWS sends DE 2009 (CITIZEN_ALIEN_INDICATOR) in the outbound file to MEDS.

The overview of the current logic to populate DE 2009 when the information is not entered on the Citizenship page and the data is in Vital Statistic page.

a. If the person has both Citizenship document type (CT 2400 – REFER TABLE 2 = Y) and Identity document type (CT 2400 – REFER TABLE 3 = Y), send a value of 'A' (both documents are verified) or 'B' (one or both document is/are not in Verified status).
b. If the document type is UNABLE TO OBTAIN DOCUMENT, send a value of 'B'.

2.8.2 Description of Change

1. Regression tests the DE 2009 logic with the new document type 3E. There is no change on the logic to populate DE 2009.

   DE 2009 is in the following MEDS transactions:
   a. MEDS AP18
   b. MEDS AP19
   c. MEDS AP34
   d. MEDS EW05
   e. MEDS EW12
   f. MEDS EW20

2.8.3 Execution Frequency

N/A.

2.8.4 Key Scheduling Dependencies

N/A.

2.8.5 Counties Impacted

All CalSAWS Counties.

2.8.6 Category

Core-Off Prime.

2.8.7 Data Volume/Performance

N/A.

2.8.8 Failure Procedure/Operational Instructions

Batch Support Operations staff will evaluate transmission errors and failures and determine the appropriate resolution (i.e., manually retrieving the file from the directory and contacting the external partner if there is an account or password issue, etc…)

2.9 Batch - Update Batch EDBC Skip Logic

2.9.1 Overview

CalSAWS will skip a program from Batch EDBC processing if an open Medi-Cal individual is missing a Birth Country in Individual Demographics except when there is a Citizenship record. Update the skip logic to add an exception if the program is Medi-Cal and there is a Vital Statistics '3E' record.

2.9.2 Description of Change

1. Update the Batch EDBC skip logic to add an exception for an individual that meets all of the below criteria:
    a. Is an Active/Pending MEM
    b. 'Birth Country' on the Individual Demographics page is not 'US or US Territory', including when 'Birth Country' is set to 'Unknown' or 'Undeclared', and
    c. No Citizenship record exists on the Citizenship page
    d. Vital Statistics record has below:
    "U.S. Citizenship Verification" Document Type on File as '3E U.S. Citizenship verified via CalHEERS FDSH: SSA' with "Citizenship Verified" as verified

2.9.3 Execution Frequency

N/A

2.9.4 Key Scheduling Dependencies

N/A

2.9.5 Counties Impacted

All CalSAWS Counties

2.9.6 Category

Core-On Prime

2.9.7 Data Volume/Performance

N/A

2.9.8 Failure Procedure/Operational Instructions

Batch Support Operations staff will evaluate transmission errors and failures and determine the appropriate resolution (i.e., manually retrieving the file from the directory and contacting the external partner if there is an account or password issue, etc…)

2.10 Data Change

2.10.1 Overview

Create Vital Statistics records for individuals who do not already have a Vital Statistics record and CalSAWS has received a DER with an e-verification for US Citizenship from CalHEERS FDSH: SSA.

2.10.2 Description of Change

1. With a one-time data change, create Vital Statistics records for individuals that meet the following criteria:
    a. **Criteria:**
        i. the individual is on a CalSAWS case with an open Medi-Cal program
        ii. the individual does not have a Vital Statistics record, and
        iii. the individual is in a DER:
            1. received prior to the implementation of this SCR, and
            2. the individual's US Citizenship is e-verified in the DER by Source = "FDSH:SSA" (CT10592_FDS)

For each identified individual, use the oldest DER based on date received that meets all the criteria above to populate their Vital Statistics record.

b. **Data:** Create the Vital Statistics and Verification records with data the same as Recommendation 2.2.3.5.a & 2.2.3.5.b, respectively.

c. **Journal:**

Create one journal per case per applicable e-verified DER ID with an open Medi-Cal program when a Vital Statistics record is created by the CalHEERS WebService with the following:
i. Do not create Journal for hidden individuals

Journal Category: Interfaces
Journal Type: Activity
Short Description: US Citizenship/Identity e-Verified by FDSH:SSA
Long Description: A Vital Statistics record for US Citizenship/Identity was created based on the electronic verification from CalHEERS via FDSH: SSA for the following individuals:

< Last Name, First Name <age in years> and the Gender (F/M)>
< Last Name, First Name <age in years> and the Gender (F/M)>

CalHEERS Case: <CalHEERS Case Number>
CalSAWS Case: <CalSAWS Case Number>
County: <County>

DER ID Reference: <DER ID>
DER Initiated Date: <Initiated Date>

2.10.3 Estimated Number of Records Impacted/Performance

~150K Vital Statistics records will be created

2.11 Automated Regression Test

2.11.1 Overview

Create new automated regression test scripts to verify U.S. Citizenship and Identity Verification document type ‘3E U.S. Citizenship verified via CalHEERS FDSH: SSA’ is available for selection on the Vital Statistics Detail page, and that this same value displays on the Vital Statistics List page upon save.

2.11.2 Description of Change

1. Create regression scripts to verify that the ‘3E U.S. Citizenship verified via CalHEERS FDSH: SSA’ value:
    a. Is selectable from the ‘Document Type on File’ dropdown in the U.S. Citizenship Verification section of the Vital Statistics Detail page in create mode.
    b. Is selectable from the ‘Document Type’ dropdown in the Identity Verification section of the Vital Statistics Detail page in create mode.
    c. Displays in the ‘Citizenship’ column of the Vital Statistics List page for a detail record with this value selected.
    d. Displays in the ‘Identity’ column of the Vital Statistics List page for a detail record with this value selected.

3 REQUIREMENTS

3.1 Project Requirements

REQ #	REQUIREMENT TEXT	How Requirement Met
2.20.1.9	The LRS Shall display summary and detailed interface LRS Data that has been received from external systems, as specified by COUNTY.	CalSAWS will send, receive, and display MAGI case, application, and person information via the CalHEERS-SAWS Interface with CalHEERS.

4 APPENDIX

Term	Definition
EDR	Eligibility Determination Request
DER	Determination Eligibility Response
Solicited DER	A Determination Eligibility Response received after sending an Eligibility Determination Request (EDR)from CalSAWS
Unsolicited DER (DER-U)	A Determination Eligibility Response received/originated from CalHEERS without initial contact from CalSAWS
Current US Territories	Virgin Island (US) GUAM American Samoa Puerto Rico

CalSAWS

California Statewide Automated Welfare System

Design Document

CA-248713

Conditional CAPI

CalSAWS	DOCUMENT APPROVAL HISTORY	
	Prepared By	Tom Lazio, Connor O'Donnell
	Reviewed By	

DATE	DOCUMENT VERSION	REVISION DESCRIPTION	AUTHOR
10/12/2023	1.0	Initial Draft	T. Lazio, C. O'Donnell
12/20/2023	2.0	- Updated Mockups 2.3.1 and 2.3.2 - Updated Section 2.3.3 (items 3 and 4) - Outreach Section 5.1 – removed 'Case Flag Title Description' from Additional Columns.	T. Lazio, C. O'Donnell
01/25/2024	3.0	- Updated verbiage in Overview and Section 1.4. - Removed Assumption # 6 since it is covered in 2.4.2.e. - Updated 2.3.3.3 to include Disability Type and Refused SSIAP Services as hidden fields on SSIAP detail page.	T. Lazio
02/01/2024	4.0	- Updated Mockup 2.3.2 - Added Section 2.3.3.3 and 2.3.3.5.b -Updated Section 2.4.2.1.e	T. Lazio C. O'Donnell
05/08/2024	5.0	Content Revision 1: • Added #5 and #6 to Section 1.3. • Updated Recommendation 2.2.3 to clarify when the Conditional CAPI field will appear. • Added Recommendations 2.4.2.3 and 2.4.2.4. • Added clarification to the Note portion of 2.5.3 and 2.6.3.	T. Lazio C. O'Donnell
05/23/2024	6.0	Content Revision 2:	T. Lazio

		• Added the following Accessibility sections:
		  ○ 2.5.6 (CAPI Manual EDBC)
		  ○ 2.6.6 (CAPI Program Configuration Override List)	

Table of Contents

1 OVERVIEW

Per ACL 08-03, individuals who become naturalized citizens while receiving Cash Assistance Program for Immigrants (CAPI) benefits can continue receiving payments on a conditional basis if they sign a Request for Conditional CAPI form (SOC 830), apply for SSI in a timely manner, and fully cooperate in the application and appeal process of the Social Security Administration (SSA).

The purpose of this SCR is to update existing CAPI EDBC functionality to allow the continuation of CAPI benefit payments for naturalized citizens who signs the conditional CAPI (SOC 830) and are in the process of SSI application approval and/or appeal.

1.1 Current Design

When a CAPI recipient becomes a 'Naturalized US Citizen', CAPI benefit payments are discontinued.

1.2 Requests

Allow CAPI benefits to continue for 'naturalized' citizens who signs conditional CAPI (SOC 830) and are in the process of SSI application approval and/or appeal.

1.3 Overview of Recommendations

1. Add 'Conditional CAPI' field to Case Summary and CAPI Detail pages.
2. Add data collection fields and required field logic to SSIAP (Supplemental Security Income Advocacy Program) Detail Page.
3. Update CAPI EDBC functionality to continue benefits for naturalized citizens with 'pending' SSI applications.
4. Update CAPI EDBC logic to set 'Conditional CAPI' field to 'Yes' on CAPI program case summary page when conditional CAPI criteria is met.
5. Display 'Conditional CAPI' field in the CAPI EDBC Summary Page.
6. Add page mapping for the CAPI EDBC Summary Page.
7. Add 'Conditional CAPI' field to CAPI EDBC (Manual) Detail Page.
8. Add 'Conditional CAPI' field to Program Configuration Override List Detail for CAPI EDBC override.
9. Data change to update Conditional CAPI field default value for existing cases.
10. Generate case listing for CAPI cases with 'Conditional CAPI' case flag.

1.4 Assumptions

1. Fields not mentioned to be modified within the description of changes will retain their current functionality and logic.

2. The user will be responsible for running EDBC to discontinue conditional CAPI for 'Naturalized Citizen' if SSI application is approved or rejected and/or the SSI appeal is approved or denied.
3. CAPI Interim Assistance Reimbursement (IAR) payments are collected under GR/CAPI recoverable accounts created automatically through batch job PBXXF207 (GRCAPIRecoveryBatch) for counties that are opted in.
4. SOC 830 (Request for Conditional CAPI After Naturalization Pending SSI/SSP Eligibility Determination) form is available in the correspondence template repository.
5. While receiving conditional CAPI all other CAPI eligibility rules will apply. A naturalized citizen receiving conditional CAPI cannot be reinstated into CAPI once CAPI benefits have been terminated.
6. The user will be responsible for setting up recovery accounts for conditional CAPI overpayments due to applicant failure to comply with the SSI application and/or appeal process.
7. Existing CAPI discontinuance NOAs will be used for CAPI participants who are discontinued from conditional CAPI or they don't meet the requirements for conditional CAPI.
8. The user will be responsible for rescinding the CAPI program that is discontinued for 'US Citizen' when a SOC 830 form is signed after the discontinuance but before the 30-day application period has expired.
9. No report changes with this SCR.

2 RECOMMENDATIONS

2.1 Case Summary Page

2.1.1 Overview

The Case Summary page needs to be updated to include an indicator as to whether the CAPI program has qualified for Conditional CAPI.

2.1.2 Case Summary Mockup

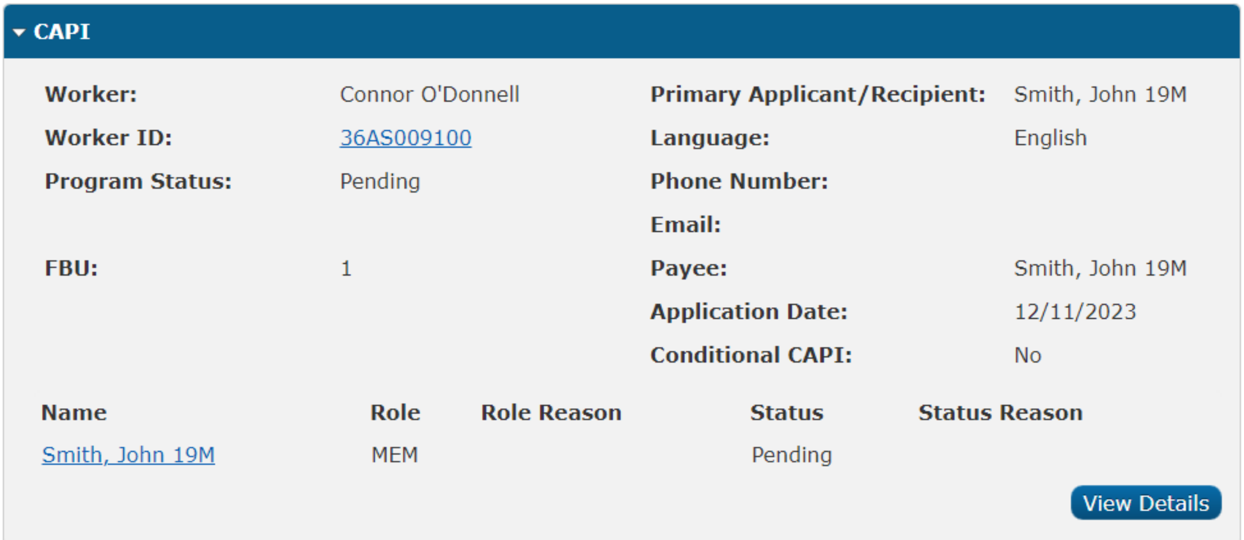


Figure 2.1.1 – Case Summary Mockup

2.1.3 Description of Changes

1. Add a new field to the Case Summary page titled 'Conditional CAPI' directly below the 'Application Date' field. This field will default to 'No' upon case creation. It can be set to 'Yes' via EDBC as described in Section 2.4.2.2. This field is the same value that is being added to the CAPI Detail page described in Section 2.2.3.

2.1.4 Page Location

- **Global: Case Info**
- **Local: Case Summary**
- **Task: Case Summary**

2.1.5 Security Updates

N/A

2.1.6 Page Mapping

Update page mapping for the new field.

2.1.7 Accessibility

No issues found.

2.1.8 Page Usage/Data Volume Impacts

N/A

2.2 CAPI Detail Page

2.2.1 Overview

The CAPI Detail page needs to be updated to include an indicator as to whether the CAPI program has qualified for Conditional CAPI.

2.2.2 CAPI Detail Mockup

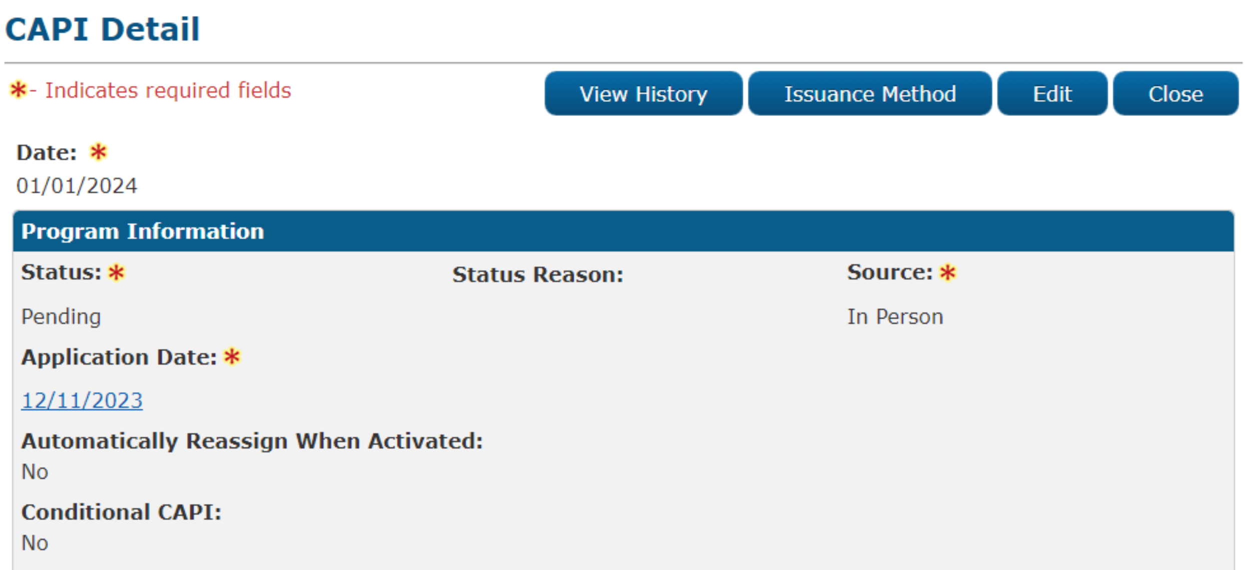


Figure 2.2.1 – CAPI Detail Mockup

2.2.3 Description of Changes

1. Add a new field to the CAPI Detail page titled 'Conditional CAPI' directly below the 'Automatically Reassign When Activated' field. This field will default to 'No' upon case creation. It can be set to 'Yes' via EDBC as described in Section 2.4.2.2. This field will appear as a read-only value on the page in both View Mode and Edit Mode. This field is the same value that is being added to the Case Summary page described in Section 2.1.3.

2.2.4 Page Location

- **Global: Case Info**
- **Local: Case Summary**

- **Task: CAPI Detail**

2.2.5 Security Updates

N/A

2.2.6 Page Mapping

Update page mapping for the new field.

2.2.7 Accessibility

No issues found.

2.2.8 Page Usage/Data Volume Impacts

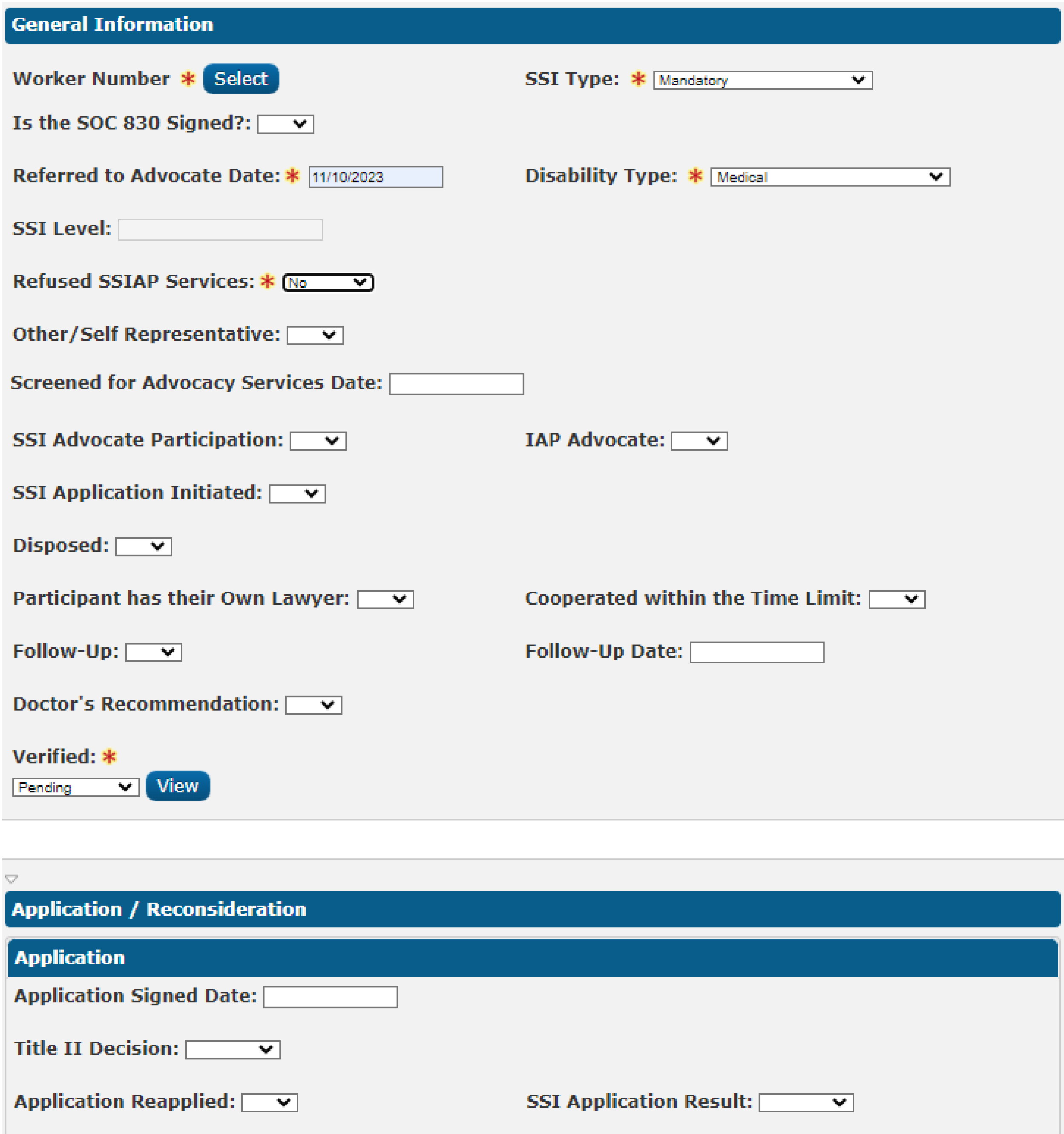
N/A

2.3 SSIAP Detail Page

2.3.1 Overview

The SSIAP Detail page needs to be updated to indicate whether the individual applying for aid has signed the SOC 830 and applied for SSI in a timely manner.

2.3.2 SSIAP Detail Mockup



General Information

Worker Number * Select

SSI Type: * Mandatory

Is the SOC 830 Signed?:

Referred to Advocate Date: * 11/10/2023

Disability Type: * Medical

SSI Level:

Refused SSIAP Services: * No

Other/Self Representative:

Screened for Advocacy Services Date:

SSI Advocate Participation:

IAP Advocate:

SSI Application Initiated:

Disposed:

Participant has their Own Lawyer:

Cooperated within the Time Limit:

Follow-Up:

Follow-Up Date:

Doctor's Recommendation:

Verified: *

Pending View

Application / Reconsideration

Application

Application Signed Date:

Title II Decision:

Application Reapplied:

SSI Application Result:

Figure 2.3.1 – SSIAP Detail Mockup (‘Is the SOC 830 Signed’ not set to ‘Yes’)

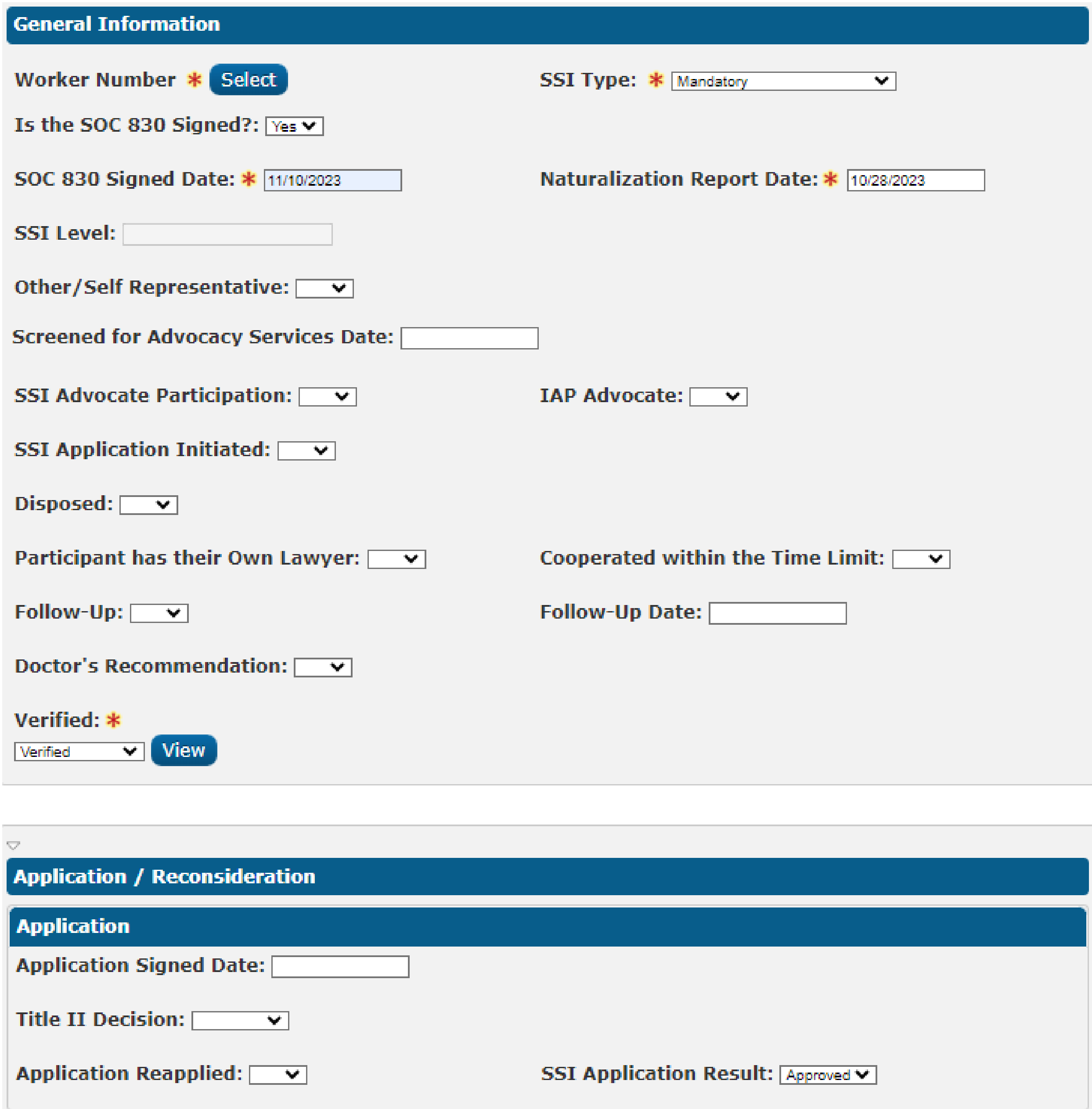


Figure 2.3.2 – SSIAP Detail Mockup ('Is the SOC 830 Signed' set to 'Yes')

2.3.3 Description of Changes

1. Add a new select field titled 'Is the SOC 830 Signed' directly above the 'Referred to Advocate Date' field. The field will have the following options:
    a. Blank space
    b. 'No'
    c. 'Yes'

2. Add a new date field titled 'SOC 830 Signed Date' directly below the 'Is the SOC 830 Signed' field. This field will dynamically appear and be required only when 'Is the SOC 830 Signed' is set to 'Yes'.
3. Add a new date field titled 'Naturalization Report Date' next to the 'SOC 830 Signed Date' field. This field will dynamically appear and be required only when 'Is the SOC 830 Signed' is set to 'Yes'.
4. Update the logic of the following fields to be hidden and not required when 'Is the SOC 830 Signed' is set to 'Yes'. The fields will otherwise remain visible and required:
    a. Referred to Advocate Date
    b. Disability Type
    c. Refused SSIAP Services
5. Add the following validation:
    a. SSI Application Result Validation
        i. Condition: When 'Is the SOC 830 Signed' is set to 'Yes', the 'SSI Application Result' field must have a value selected.
        ii. Message: "SSI Application Result – When 'Is the SOC 830 Signed' has a value of 'Yes', 'SSI Application Result' must have a value selected".
    b. SOC 830 Signed Date Validation
        i. Condition: The 'SOC 830 Signed Date' date entry must be on or after the 'Naturalization Report Date' date entry.
        ii. Message: "SOC 830 Signed Date – The 'SOC 830 Signed Date' date entry must be on or after the 'Naturalization Report Date' date entry".

2.3.4 Page Location

- **Global: Eligibility**
- **Local: Customer Information**
- **Task: SSIAP**

2.3.5 Security Updates

N/A

2.3.6 Page Mapping

Update page mapping for new fields.

2.3.7 Accessibility

No issues found.

2.3.8 Page Usage/Data Volume Impacts

N/A

2.4 Update CAPI EDBC Functionality

2.4.1 Overview

When a CAPI recipient becomes a naturalized citizen, EDBC automatically discontinues benefits. However, per policy a naturalized citizen can request continuation of CAPI benefits by signing a request form for conditional CAPI (SOC 830) while their SSI application is still pending or while their appeal of an SSI eligibility rejection determination is still pending.

This SCR will update existing CAPI EDBC functionality to allow CAPI benefits to continue for naturalized citizens who have a signed SOC 830 and have a pending SSI application and/or appeal.

2.4.2 Description of Changes

1. Update existing CAPI EDBC logic to set 'Conditional CAPI' field to 'Yes' on CAPI program Case Summary and CAPI Detail pages when all the following are true:
    a. Participant received CAPI benefits in the prior month.
    b. There exists a SSIAP record for the participant that is effective any date within the benefit month.
    c. Participant's Citizenship type is 'Naturalized US Citizen'. See Appendix.
    d. 'Is the SOC 830 signed?' = 'Yes' on SSIAP detail page.
    e. The 'SOC 830 Signed Date' must be within 30 days of the 'Naturalization Report Date' on the SSIAP detail page.

    \- AND -

    One of the following is true:

    f. SSI Application Result is 'Pending' on SSIAP detail page.

    \- OR –

    g. SSI Appeals Council Decision on SSIAP detail page equals one of the following:
        i. Pending
        ii. Pending Other
        iii. Pending Self

    Note: The 'Conditional CAPI' field default value will be 'No'.

2. Update CAPI EDBC logic to grant CAPI benefits to recipients who meet the criteria from recommendation 1 above.

3. The 'Conditional CAPI' field should be displayed in CAPI EDBC Summary Page as shown below:

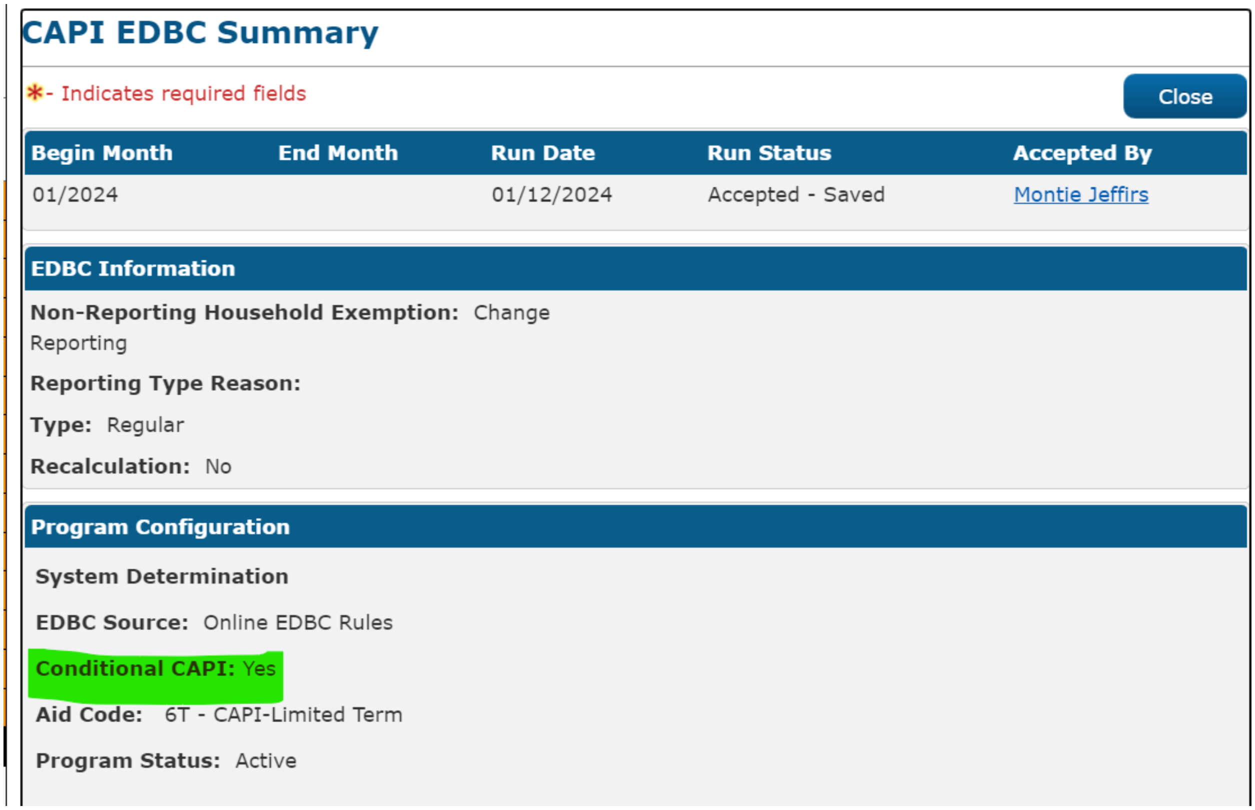


Figure 2.4.3.1 – CAPI EDBC Summary Page After SCR is Implemented

4. Add page mapping for the CAPI EDBC Summary Page.

2.4.3 Programs Impacted

CAPI

2.5 Update CAPI EDBC (Manual) Detail Page

2.5.1 Overview

The 'Conditional CAPI' option should be available for the user to choose on the CAPI EDBC (Manual) Detail page when performing a Manual CAPI EDBC.

2.5.2 CAPI EDBC (Manual) Detail Page Example

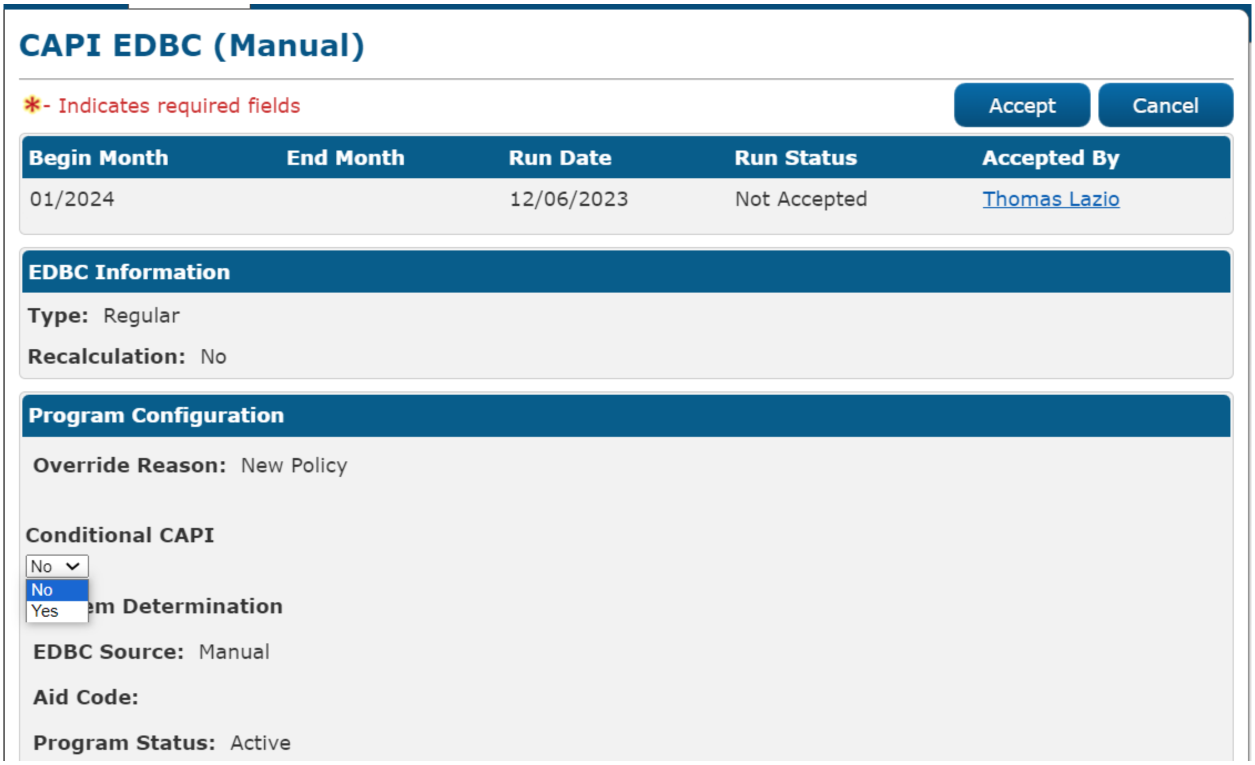


Figure 2.5.2.1 – CAPI EDBC (Manual) Detail Page After SCR is Implemented

2.5.3 Description of Changes

1. Update the CAPI EDBC (Manual) Detail page with 'Conditional CAPI' dropdown field with the following options:
    i. No
    ii. Yes

Note: Default value will be set to 'No' in the 'Program Configuration' section where the user can update the dropdown field value. The 'Conditional CAPI' dropdown field will be displayed in the CAPI EDBC (Manual) page as shown in **Figure 2.5.2.1** above.

2.5.4 Programs Impacted

CAPI

2.5.5 Page Mapping

Update page mapping for new fields.

2.5.6 Accessibility

The following Accessibility enhancements have been identified:

- Certain ARIA roles must contain particular children.

2.6 Update Override CAPI EDBC Page

2.6.1 Overview

The 'Conditional CAPI' option should be available for the user to choose on the Program Configuration Override List Detail page when overriding a CAPI EDBC.

2.6.2 Program Configuration Override List Detail Page Example (CAPI EDBC)

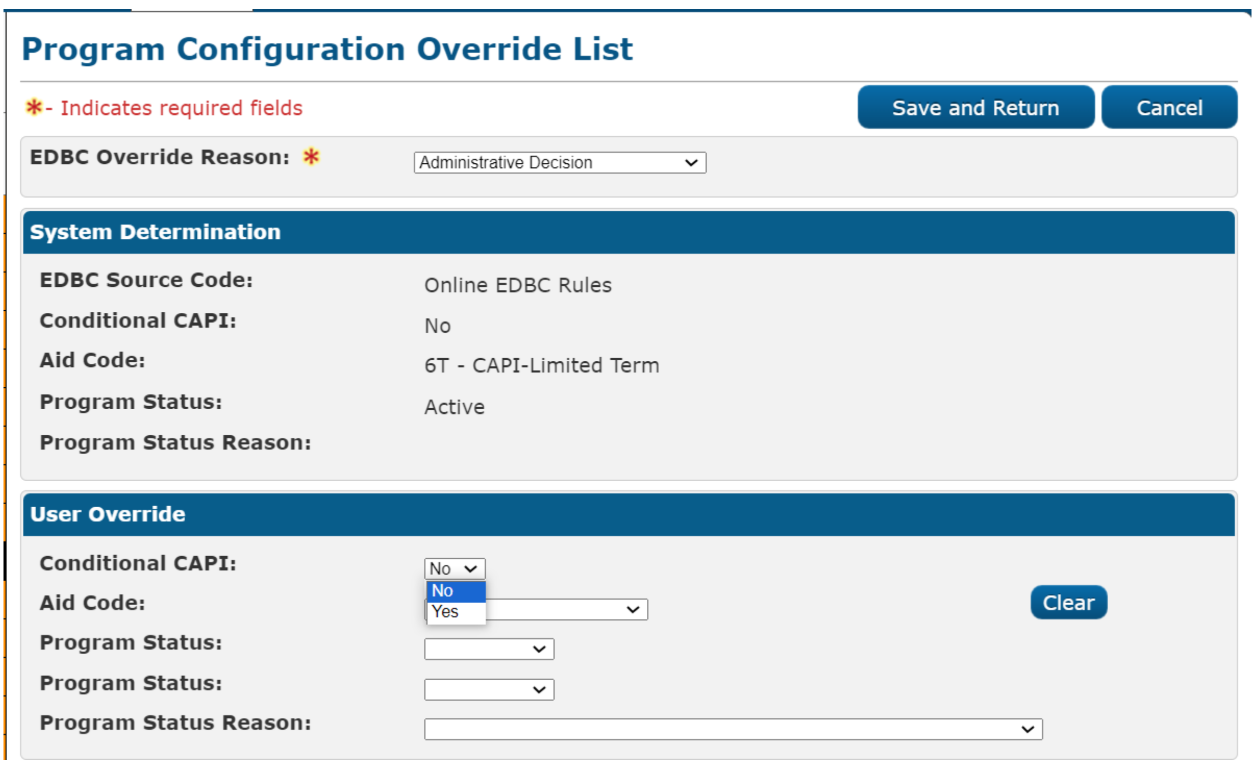


Figure 2.6.2.1 – Program Configuration Override List Detail Page (CAPI EDBC) After SCR is Implemented

2.6.3 Description of Changes

1. Update the Program Configuration Override List Detail page (CAPI EDBC) with 'Conditional CAPI' dropdown field with the following options:

i. No
ii. Yes

Note: Default value will be set to 'No' in the 'User Override' section where the user can update the dropdown field value. The 'Conditional CAPI' dropdown field will be displayed in the Program Configuration Override List Detail page as shown in **Figure 2.6.2.1** above.

2.6.4 Programs Impacted

CAPI

2.6.5 Page Mapping

Update page mapping for new fields.

2.6.6 Accessibility

The following Accessibility enhancements have been identified:

- Certain ARIA roles must contain particular children.
- Table header text should not be empty.

2.7 Data Change – Add Conditional CAPI

2.7.1 Overview

With the addition of the Conditional CAPI field, existing CAPI cases need to be updated to have the default value of 'No'.

2.7.2 Description of Change

1. Update all existing CAPI cases to have their Conditional CAPI value (added in Section 2.2) set to 'No'.

2.7.3 Estimated Number of Records Impacted/Performance

79,619

2.8 Automated Regression Test

2.8.1 Overview

Create new automated regression test scripts to verify the 'Conditional CAPI' options when running EDBC for the CAPI program, options on the SSIAP Detail page, and display on the Case Summary and CAPI Detail pages.

2.8.2 Description of Change

Create new regression scripts to provide the following coverage:

1. Verify the ‘Conditional CAPI’ field displays on the Case Summary page, with the following values:
    a. ‘No’ for a new CAPI program in ‘Pending’ status
    b. ‘No’ for a CAPI program in ‘Active’ status that was not determined to be conditionally eligible as outlined in Section 2.4.2
    c. ‘Yes’ for a CAPI program in ‘Active’ status that was determined to be conditionally eligible as outlined in Section 2.4.2
    d. ‘Yes’ for a CAPI program in 'Discontinued' status that was previously determined to be conditionally eligible as outlined in Section 2.4.2
2. Verify the ‘Conditional CAPI’ field displays on the CAPI Detail page, with the same value as the same field displayed on the Case Summary page in each scenario outlined in the previous recommendation.
3. Verify the following on the SSIAP Detail page:
    a. The ‘Is the SOC 830 Signed’ field displays in create and edit modes, with the following values available for selection:
        i. -Select-
        ii. No
        iii. Yes
    b. The ‘SOC 830 Signed Date’ field displays in create and edit modes only when ‘Yes’ is selected from the 'Is the SOC 830 Signed' field.
    c. The ‘Referred to Advocate Date’ field is not required when ‘Yes’ is selected from the ‘Is the SOC 830 Signed’ field.
    d. The ‘SSI Application Result’ field is required when ‘Yes’ is selected from the ‘Is the SOC 830 Signed’ field.
4. Verify the ‘Conditional CAPI’ field is available on the CAPI EDBC (Manual) Detail page in create mode, with ‘No’ and ‘Yes’ values available for selection. Verify that EDBC results can be accepted and saved with each value selected.
5. Verify the ‘Conditional CAPI’ field is available on the Program Configuration List page when overriding a CAPI EDBC result, with ‘No’ and ‘Yes’ values available for selection. Verify that EDBC results can be accepted and saved with each value selected.

3 REQUIREMENTS

3.1 Project Requirements

REQ #	REQUIREMENT TEXT	How Requirement Met
2.8.2.14	The LRS shall determine which individuals must be included in the eligibility	This SCR will update CAPI EDBC rules to allow naturalized

	determination and take appropriate action(s), based on program rules.	citizens who apply for conditional CAPI to continue benefits while SSI application is still pending.

4 APPENDIX

As of 11/14/2023, the list below shows the Document Types for the Citizenship Status Detail page that result in the CAPI participant being a 'Naturalized US Citizen':

- Certificate of Citizenship-N-550, N-570, N-571 or N-578
- I-179 or I-197 - US Citizen Identification Card
- US Passport

5 OUTREACH

5.1 Lists

List Name: CAPI Cases With 'Conditional CAPI' Case Flags
List Criteria: All CAPI cases that meet the following criteria:

- CAPI program exists on the case.
- There exists a case flag record with one of the following title descriptions:
    - Conditional CAPI
    - Conditional CAPI Flag
    - Conditional CAPI-Naturalized
    - _zzzCAPI-Conditional CAPI

Standard Columns:

- Case Name
- Case Number
- County
- Unit
- Unit Name
- Office Name
- Worker ID

Additional Column(s):

- Program Status (**Note:** As of current system date)
- Program Status Reason (**Note:** Can be null)
- Case Flag Begin Date
- Case Flag End Date (**Note:** Can be null)

Frequency: One-Time

The list will be posted to the following location: CalSAWS Web Portal>System Changes>SCR and SIR Lists>2024>CA-248713

CalSAWS

California Statewide Automated Welfare System

Design Document

CA-264181

Allow Deactivation of Task Banks

CalSAWS	DOCUMENT APPROVAL HISTORY	
	Prepared By	Vallari Bathala, Justin Dobbs, Jennifer Muna, Soundarya Ramesh
	Reviewed By	Justin Dobbs, Sarah Rich, Carlos Albances, Dymas Pena, Connie Buzbee, Anthony Cheung, Nue Nelson

DATE	DOCUMENT VERSION	REVISION DESCRIPTION	AUTHOR
01/29/2024	1.0	Initial Revision	Vallari Bathala, Justin Dobbs, Jennifer Muna, Soundarya Ramesh
07/17/2024	1.1	Removed assumption #4 Enhancement CA-272460 is implemented in the same release or sooner than this enhancement from section 1.4 Assumptions. Added additional validation for Document Routing Rule Detail Page Included Accessibility sub-section in section 2.6 Additional Page Modifications. Updated section 2.17 Automated Regression Test to include test scenarios for Bank Deactivation and FDS API changes.	William Baretsky, Vallari Bathala

Table of Contents

1 OVERVIEW

This design outlines modifications to Task Management functionality within the CalSAWS System to allow Users to deactivate the Task Bank after being created.

1.1 Current Design

Task Banks are repositories within the county that can hold Tasks. The Bank Search and Bank Detail pages within the CalSAWS System allow Task Banks to be created and configured. Once a Task Bank is created it cannot be removed or deactivated.

Tasks can be created and assigned to Task Banks manually via worker transactions, by automated action trigger events, certain Self Service Portal actions and processing via the Task API (Application Programming Interface).

1.2 Requests

1. Add functionality to support the deactivation of Task Banks.
2. Update FDS (Functional Design Specification) Task API functionality to modify the existing Task Bank logic assignment to check if Task Bank Id has an Active status for the County prior to updating/creating a task.
3. Update BenefitsCal Task Assignment processing which evaluates for a Task Bank to consider the Bank status.

1.3 Overview of Recommendations

1. Add a Status field to Task Banks allowing a Task Bank to be deactivated.
2. Update Task Management online pages and processing which utilizes Task Banks to consider the Status of a Task Bank.
3. Update the FDS Task API update and create endpoints in both versions v1 and v2 to modify the existing Task Bank logic assignment logic to check if Task Bank has an Active status for the County prior to updating/creating a task.
4. Update BenefitsCal API Task Bank Assignment logic to check the Task Bank status prior to assigning a task.

1.4 Assumptions

1. An Inactive Task Bank cannot hold an open task.
2. There will be no front-end changes to the Task Reassignment functionality.
3. There are no modifications to the front-end and processing of QA/QC Task Sample logic. There is no impact to this functionality because a QA/QC Task Sample is based on Completed/Voided Tasks, which can reside in a Bank that has since become Inactive.
4. Existing Task Management assignment functionality is to only create a Task when a valid assignment exists. If an existing Task trigger such as an Automated

Action is configured to assign to a Task Bank that has become Inactive, a Task will not be created if the Automated Action is invoked due to lack of a valid assignment (Active Task Bank).

5. All existing Self-Service Portal API functionalities will remain unchanged that are not mentioned in this SCR. There is no update to API Swagger Documentation.
6. Release of Information API is currently disabled in CalSAWS. The ROI Automated Action update will be addressed in CA-258275 upon the re-enablement of the ROI API.
7. CA-264180 implemented enhancement to Application Transfer API to utilize the Automated Action framework.
8. No changes are required for the Self-Service Portal (BenefitsCal); however, end-to-end testing support will be needed. CSPM-73908 has been logged for the end-to-end testing support.

2 RECOMMENDATIONS

This section will outline recommendations to introduce the Bank deactivation functionality to the CalSAWS System.

2.1 Bank Search Page

2.1.1 Overview

This section describes recommendations to include a Status field in the Bank Search page.

2.1.2 Bank Search Page Mockup

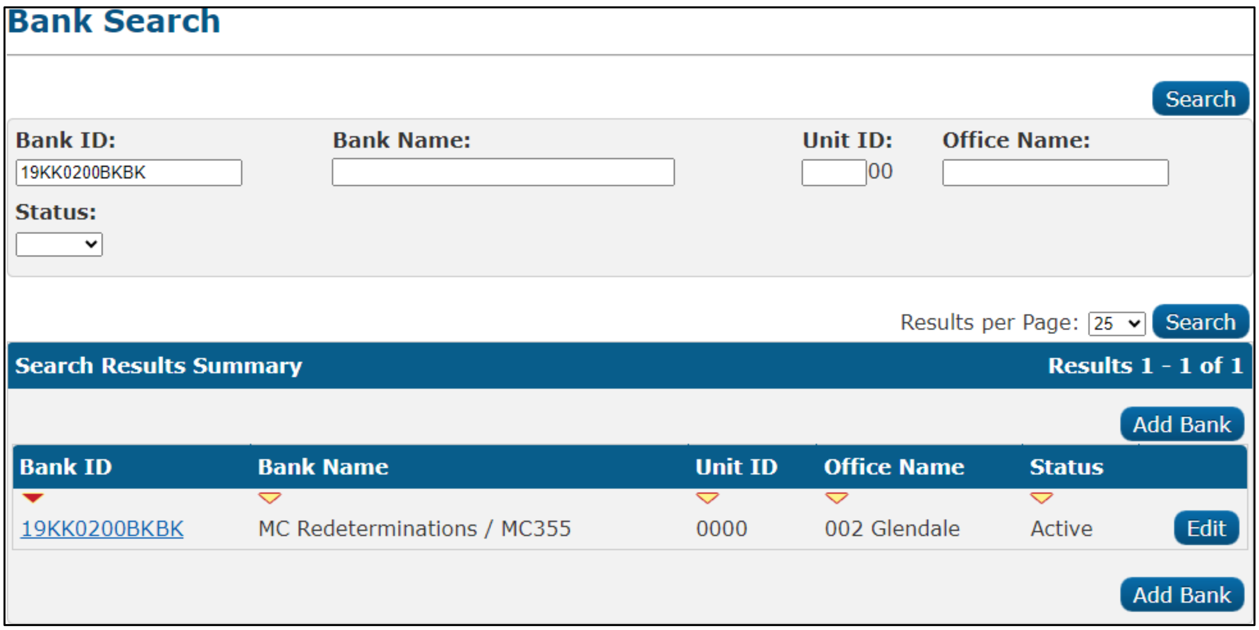


Figure 2.1.2 – Bank Search Page

2.1.3 Description of Changes

1. Search Parameters
    a. Status – Add a dropdown field which will filter the Banks based on the Status. Options included are:
        i. BLANK
        ii. Active (Default)
        iii. Inactive

    The Default value of "Active" will only display the Banks which are active at the time of the search. The value of 'BLANK' allows all Task Banks (Active or Inactive) to be displayed.
2. Search Results Summary
    a. Status – Add a sortable column to display the current status of the Bank.

2.1.4 Page Location

- **Global:** Admin Tools
- **Local:** Office Admin
- **Task:** Bank

  The Task Navigation will display if the user profile contains the "BankSearchView" security right.

2.1.5 Security Updates

N/A

2.1.6 Page Mapping

Update page mapping for the new Status field.

2.1.7 Accessibility

The following accessibility enhancements have been identified:

- iFrames must have an accessible name.

2.1.8 Page Usage/Data Volume Impacts

N/A

2.2 Bank Detail Page

2.2.1 Overview

This section describes recommendations to include a Status field in the Bank Detail page.

2.2.2 Bank Detail Page Mockup

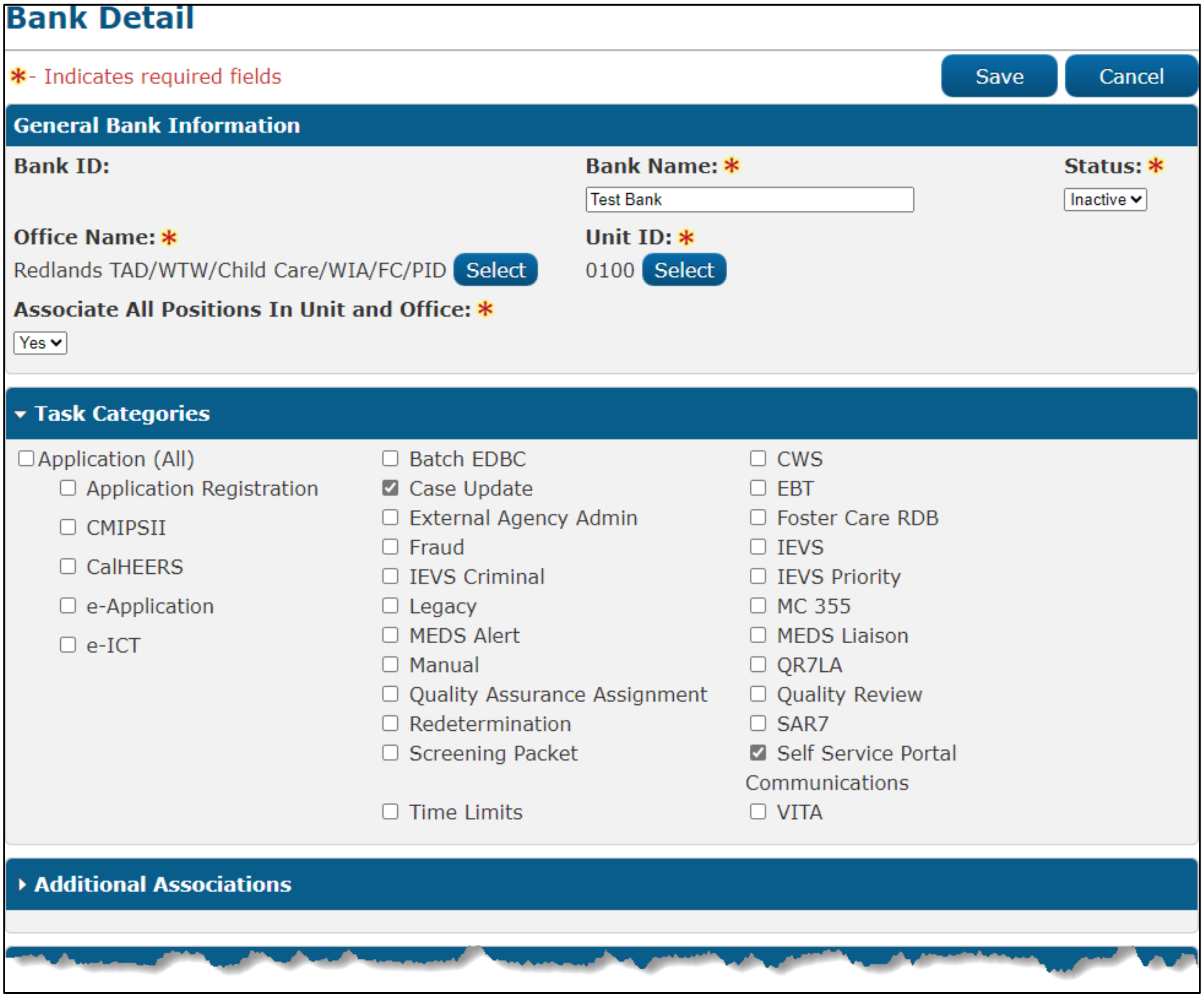


Figure 2.2.2-1 – Bank Detail Page – Create Mode

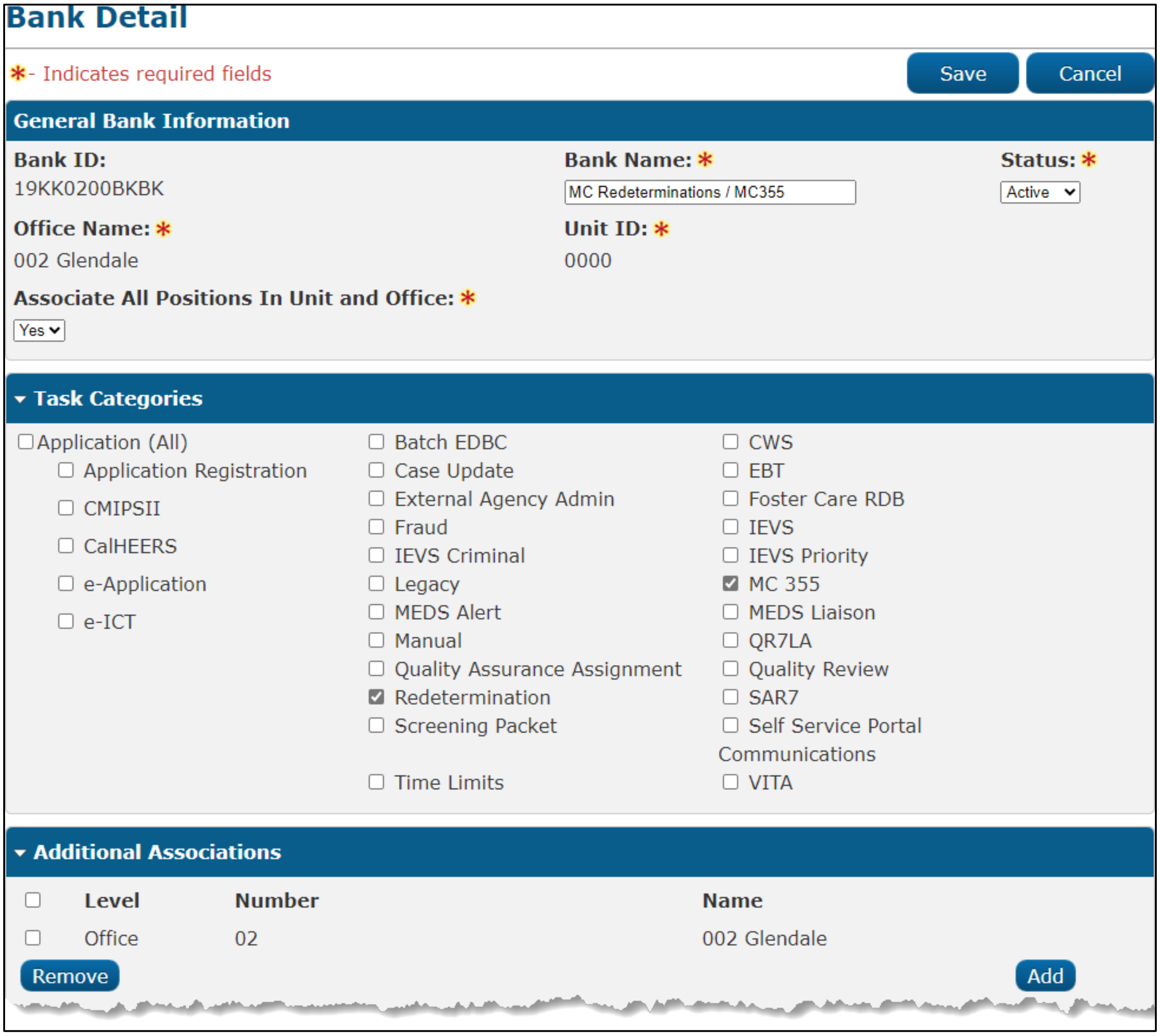


Figure 2.2.2-2 – Bank Detail Page – Edit Mode

2.2.3 Description of Changes

1. <u>General Bank Information</u>
    a. Status **(required)** – Add new a dropdown menu, in Create and Edit modes, that indicates the status of the Bank. Options included are:
        i. Active (Default)
        ii. Inactive

2.2.4 Page Validations

Add page validations to the Bank Detail page when the Status is set to Inactive and one or more of the following scenarios are true:

1. The Bank is associated to one or more open Tasks in ‘Assigned’ or ‘In Process’ status, display the following validation:
    a. “Status - The Bank is currently configured to one or more open Tasks.”

Any and all applicable validations will display at once at the top of the Bank Detail page.

2.2.5 Page Location

- **Global:** Admin Tools
- **Local:** Office Admin
- **Task:** Bank

Click on a hyperlink of the desired result displayed in the Bank Search page or the “Add Bank” button to navigate to the Bank Detail page.
The Task Navigation will display if the user profile contains the “BankSearchView” security right.

2.2.6 Security Updates

N/A

2.2.7 Page Mapping

Update page mapping for the new Status field.

2.2.8 Accessibility

The following accessibility enhancements have been identified:

- iFrames must have an accessible name.

2.2.9 Page Usage/Data Volume Impacts

N/A

2.3 Select Bank Page

2.3.1 Overview

This section describes recommendations to include a Status field in the Select Bank page.

2.3.2 Select Bank Page Mockup

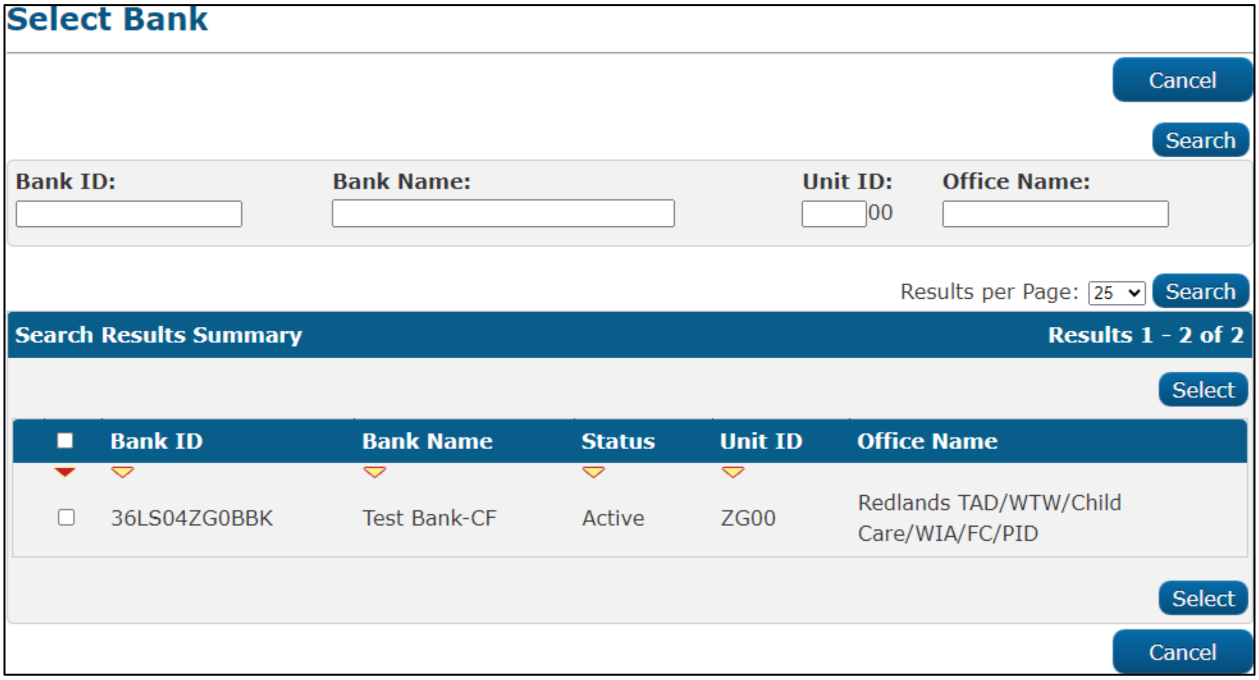


Figure 2.3.2-1 – Select Bank Page – Active Banks

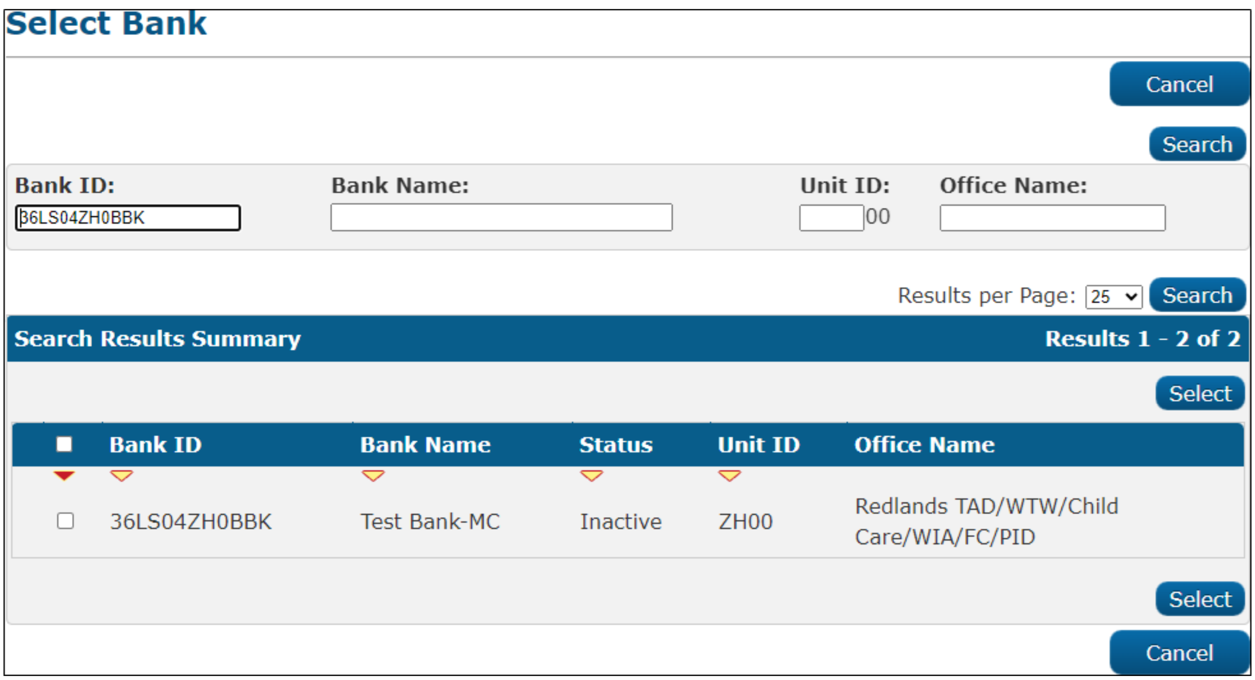


Figure 2.3.2-2 – Select Bank Page – Inactive Banks

2.3.3 Description of Changes

Functional Note: Inactive Banks will be included and searchable on the Select Bank page to allow for selection of an Inactive Bank from pages such as Task Search. Historical Tasks that were closed while assigned to the Bank when it was Active will still be searchable after the Bank becomes Inactive. Including Inactive banks in this page will support this.

1. Search Results Summary
    a. Status – Add a column to display the status of the Bank.

Note: These recommendations apply to the Select Bank page which allows selection of a single Task Bank and the Select Bank page which allows for multi select (i.e., QA/QC Task Sample).

2.3.4 Page Location

- **Global:** N/A
- **Local:** N/A
- **Task:** N/A

Select Bank Page is discoverable through multiple pages.

2.3.5 Security Updates

N/A

2.3.6 Page Mapping

Update page mapping for the new Status field.

2.3.7 Accessibility

The following accessibility enhancements have been identified:

- iFrames must have an accessible name.

2.3.8 Page Usage/Data Volume Impacts

N/A

2.4 Task Search Page

2.4.1 Overview

This section outlines the modifications required to display validation messages when an action attempts to assign a Task to an Inactive Task Bank.

2.4.2 Task Search Page Mockup

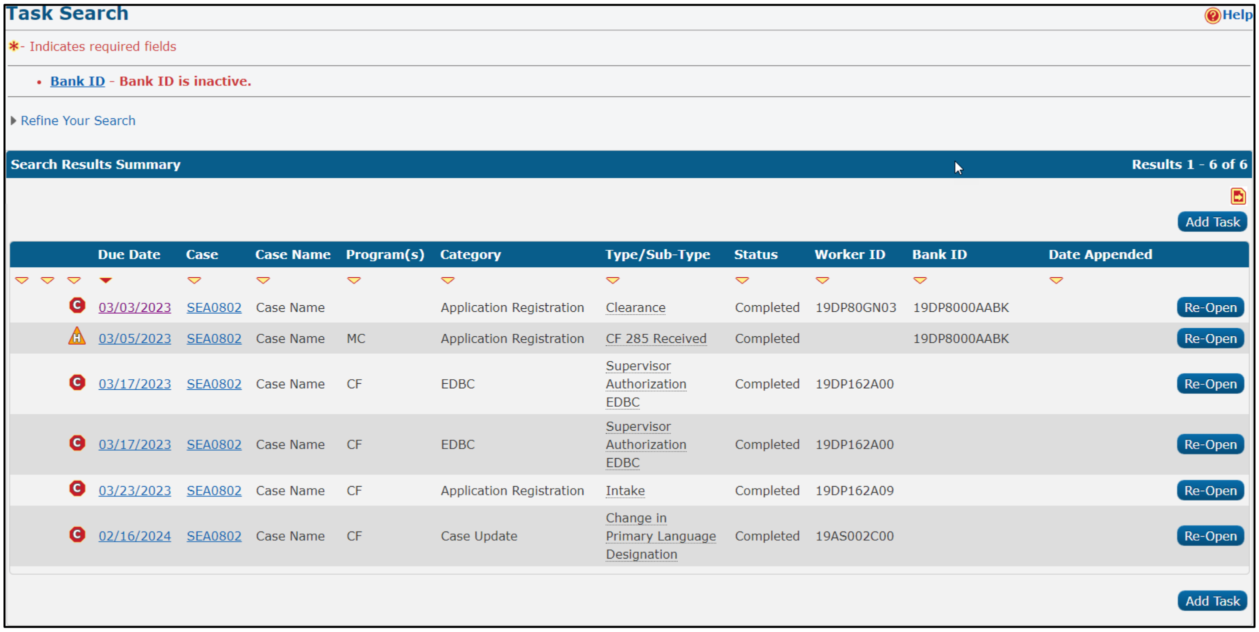


Figure 2.3.2 – Task Search Re-Open Validation

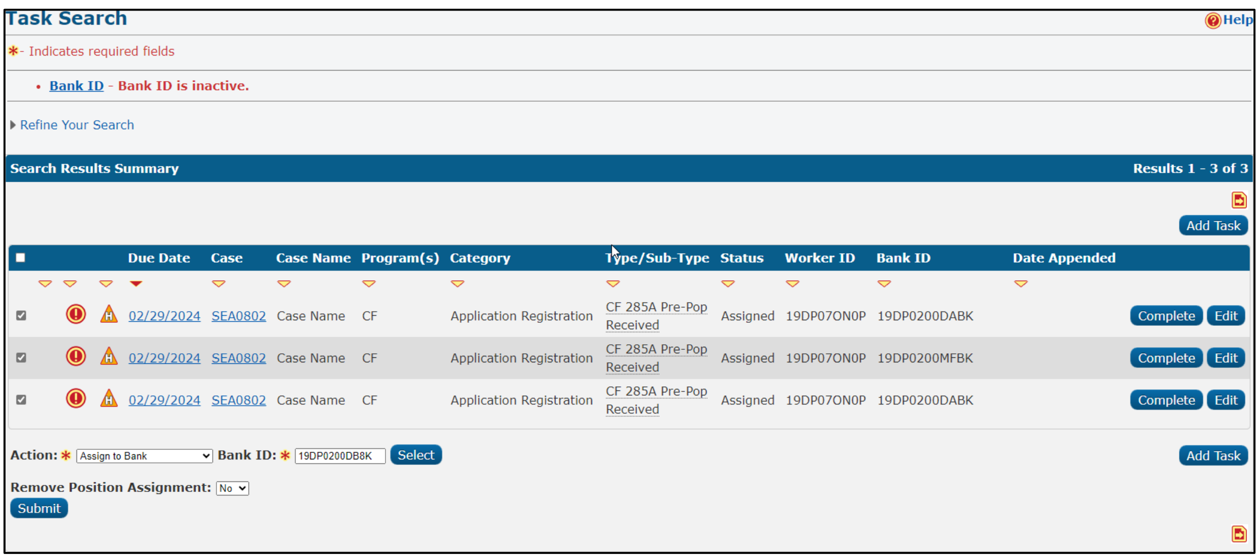


Figure 2.3.2 – Task Search Action Validation

2.4.3 Description of Changes

1. Implement validations listed in section 2.4.4 Page Validations.

   Functional Note: CA-214923 introduced the 'Re-Open' functionality which, upon the click of the 'Re-Open' button sets the status of the Task to "Assigned" and assigns the Task to the previous owner. If the previous assignee is a Task Bank and that Bank is now inactive, the Task cannot be re-opened.

2.4.4 Page Validations

Add page validations to Task Search page when a User attempts an action that will result in assigning a Task to an inactive Task Bank in the following scenarios:

1. A User manually enters an inactive Bank ID in the Bank ID field for Action of 'Assign to Bank' and clicks Submit:
   a. "Bank ID - Bank ID is inactive."
2. A User clicks the Re-Open button for a closed Task that is assigned to a now Inactive Bank:
   a. "Bank ID - Bank ID is inactive."

2.4.5 Page Location

- **Global:** N/A
- **Local:** N/A
- **Task:** N/A

  Task Search Page is accessible through Utility bar's Tasks Option.

2.4.6 Security Updates

N/A

2.4.7 Page Mapping

N/A

2.4.8 Accessibility

The following accessibility enhancements have been identified:

- iFrames must have an accessible name.

2.4.9 Page Usage/Data Volume Impacts

N/A

2.5 My Banks Page

2.5.1 Overview

This section outlines the modifications to the My Banks page to not display Inactive Task Banks.

2.5.2 Description of Changes

The My Banks page displays information of all Banks that the viewing worker is associated to as a paginated list.

1. Modify the My Banks page to not include Inactive Banks.

2.5.3 Page Validations

N/A

2.5.4 Page Location

- **Global:** N/A
- **Local:** N/A
- **Task:** N/A

  My Banks Page is accessible through Utility bar's Tasks Option.

2.5.5 Security Updates

N/A

2.5.6 Page Mapping

N/A

2.5.7 Accessibility

The following accessibility enhancements have been identified:

- iFrames must have an accessible name.

2.5.8 Page Usage/Data Volume Impacts

N/A

2.6 Additional Page Modifications

2.6.1 Overview

This section outlines additional modifications to Task Management functionality to only allow selection of 'Active' Task Banks in pages currently utilizing the Bank attribute.

2.6.2 Description of Changes

1. Add the following page validations to Task Management pages when a User selects or manually enters an 'Inactive' Task Bank:

Page(s)	Field(s)	Action Scenario	Validation(s)
Task Detail Page	Bank ID	Assigning an Assigned or In-Process Task to an Inactive Bank.	Bank ID - Bank ID is inactive.
Document Routing Rule Program Detail Page	Bank ID	Configuring a Document Routing Rule with Task assignment to an Inactive Bank.	Bank ID - Bank ID is inactive.
Document Routing Rule Program Detail Page	Bank ID	Configuring a Document Routing Program Based Rule to an Inactive Bank.	Bank ID - Bank ID is inactive.
MEDS Alert Admin Detail Page	Default Bank ID	Configuring the Task Information panel of the MEDS Alert Admin Detail page with Task assignment to an Inactive Bank.	Default Bank ID - Bank ID is inactive.
Automated Action Detail Page	Default Bank ID	Configuring the Task Information panel of the Automated Action Detail page with Task assignment to an Inactive Bank.	Default Bank ID - Bank ID is inactive.
Task Reassignment Detail Page	Task Sources(s)	Configuring a Task Reassignment with one or more Source Banks that are Inactive.	Task Sources(s) – One or more source Banks are inactive.
Task Reassignment Detail Page	Task Destination(s)	Configuring a Task Reassignment with one or more Destination Banks that are Inactive.	Task Destination(s) – One or more destination Banks are inactive.
Task Type Detail	Bank ID	Configuring the Resulting Task Information panel to assign to an Inactive Bank.	Bank ID - Bank ID is inactive.
Task Sub-Type Detail	Bank ID	Configuring the Resulting Task Information panel to assign to an Inactive Bank.	Bank ID - Bank ID is inactive.

2.6.3 Accessibility

The following accessibility enhancements have been identified:

- iFrames must have an accessible name.

2.7 Task Upload

2.7.1 Templates

1. Update the "Manual Assignment – Bank ID" column instructions in the Task Upload templates to specify that the Bank entries must have an "Active" status.

   The templates that require updates are listed in section 3. Supporting Documents.

Column	Required	Instructions
Manual Assignment – Bank ID	No	If manual Task assignment is applicable, enter a Bank ID to assign the Task to. This value must be a valid Bank ID for the county. The Bank must have an "Active" status. This field may be left blank if either the "Automated Assignment" or "Manual Assignment - Worker ID" field has a valid entry. An entry in this field with a simultaneous entry in the "Manual Assignment - Worker ID" field is acceptable. In this case, the Task will be assigned to both the Worker ID and Bank ID.

2.7.2 Preview Processing

1. Update the Closest Bank assignment logic to only evaluate Active Banks for all Task Upload types. The templates that require updates are listed in section 3. Supporting Documents.

2. If available, evaluate the Bank ID value entered in the Bank ID column within the Task Upload template and generate a Preview Error Message if the Bank ID is Inactive.

Field	Validation	Error Message
Manual Assignment - Bank ID	1. If the Bank ID is valid for the county of the Task Upload, confirm that the Bank ID Status is set to 'Active'.	Inactive Bank ID.

3. If a Task Upload preview has been generated and has Tasks queued up for Task creation to be assigned to a Task Bank that has been set to Inactive Status AFTER the preview has been generated, these Tasks will not be created if the preview is Approved.

   As a simplified example, if an uploaded template contains:

Case Number	Task Type	Manual Assignment - Bank ID
1234567	Case Review	99ABC1230BBK
0000011	Case Review	99DEF4560BBK

Both Banks are in the 'Active' Status at the time of Preview Processing. The Preview Processing successfully completes and moves to 'Preview Complete'. At this point the User sets Bank 99DEF4560BBK to 'Inactive' and then the User approves the generated preview.

Only the Tasks for Bank 99ABC1230BBK will be created.

2.7.3 Mass Closure Void Processing

For "Closure" Task Uploads that have been processed, if the Task Upload closed a Task that was assigned to a Task Bank that has since been deactivated; if the Task Upload is Voided, the Task(s) assigned to the Inactive Bank will not be reopened.

2.8 Processing Modifications

2.8.1 Overview

This section outlines additional modifications to Task Management processing functionality to respect the Task Bank Status.

2.8.2 Description of Changes

1. Modify the following processing functions to respect the Task Bank Status:

Functionality	Modification
Get Next	Modify Get Next processing logic to only consider Active Task Banks.
Office Distribution Assignment	Modify Office Distribution assignment logic to only consider Active Task Banks.
Closest Bank Assignment	Modify Closest Bank assignment logic to only consider Active Task Banks.
Task Reassignment Processing	Modify the Task Reassignment instruction processing to only consider Active Task Banks when distributing or reclaiming Tasks.
Task Reassignment Run Result Processing	Update the following Task Reassignment Run Result Detail reason "Task Destination(s) Do Not Have Active/Available Workers" to "Task Destination(s) Do Not Have Active/Available Workers or Active Banks". If a Task Reassignment is configured with Destination Workers and/or Destination Banks and processing cannot identify at least one valid Worker or Bank, the Run Result Reason is logged, and Tasks are not reassigned.

NOTE: If a Task Bank is set to Inactive status, and an Automated Task trigger such as an Automated Action, Document Routing Rule or MEDS Alert are configured to route directly into this Task Bank, these automated events will NOT create a Task if they are triggered.

For example, if a Document Routing Rule is configured to assign Tasks into Bank 123BK and Bank 123BK is set to Inactive status, this Document Routing Rule will no longer produce Tasks due to lack of a valid assignment location for the Tasks.

2.9 Self-Service Portal: Update EBT Theft API Task Bank Assignment Logic

2.9.1 Overview

The EBT Theft API is a RESTful webservice that will allow CalSAWS to do the following:

- Retrieve the EBT 2259 form information from the Self-Service Portal and store in the CalSAWS new EBT Theft pages.
- Create an automated Journal Entry upon receiving an electronic form of EBT 2259/EBT 2259A.
- Create an Automated Action that will create a task for the worker upon receiving an electronic form of EBT 2259/EBT 2259A.

This section outlines the necessary modifications in the EBT Theft API to update the Task Bank task assignment logic when attempting to assign the task to a Task Bank.

2.9.2 Description of Change

1. Update the EBT Theft API by modifying the existing Task Bank logic assignment to check if Task Bank Id has an Active status for the County prior to assigning a task. If a valid task assignment cannot be determined, the task will not be created. **Note**: No error response code will be returned to the Self-Service Portal, thus the data transfer from the Self-Service Portal will still occur regardless of the task generation outcome.

2.9.3 Execution Frequency

Real Time

2.9.4 Key Scheduling Dependencies

N/A

2.9.5 Counties Impacted

CalSAWS Counties

2.9.6 Category

Real Time API

2.9.7 Data Volume/Performance

N/A

2.9.8 Failure Procedure/Operational Instructions

Operations staff will evaluate transmission errors and failures and determine the appropriate resolution (i.e., manually retrieving the file from the directory and contacting the external partner if there is an account or password issue, etc.)

2.10 Self-Service Portal: Update Support Request API Task Bank Assignment Logic

2.10.1 Overview

The Support Request API is a RESTful webservice that is used to send a customer's Self-Service Portal request for additional services and/or support to CalSAWS, including the generation of an Automated Action task for the worker or bank assigned to the case.

This section outlines the necessary modifications in the Support Request API to update the Task Bank task assignment logic when attempting to assign the task to a Task Bank.

2.10.2 Description of Change

1. Update the Support Request API by modifying the existing Task Bank logic assignment to check if Task Bank Id has an Active status for the County prior to assigning a task.
    a. If a valid assignment is not found for the task, return a 422 error response with the following to the Self-Service Portal:
        i. benefitscal-00009: The worker or Task Bank cannot be found to create a task.

2.10.3 Execution Frequency

Real Time

2.10.4 Key Scheduling Dependencies

N/A

2.10.5 Counties Impacted

CalSAWS Counties

2.10.6 Category

Real Time API

2.10.7 Data Volume/Performance

N/A

2.10.8 Failure Procedure/Operational Instructions

Operations staff will evaluate transmission errors and failures and determine the appropriate resolution (i.e., manually retrieving the file from the directory and contacting the external partner if there is an account or password issue, etc.)

2.11 Self-Service Portal: Update Report a Change API Task Bank Assignment Logic

2.11.1 Overview

The Report a Change (RAC) is a RESTful webservice that creates a Report a Change task and journal entry in CalSAWS when a change is reported from the Self-Service Portal.

This section outlines the necessary modifications in the RAC API to update the Task Bank task assignment logic when attempting to assign the RAC task to a Task Bank.

2.11.2 Description of Change

1. Update the RAC API by modifying the existing Task Bank logic assignment to check if Task Bank Id has an Active status for the County prior to assigning a task.
    a. If a valid assignment is not found for the task, return existing 404 response code to the Self-Service Portal.
        i. 404 response code description: Could not find a valid assignment.

2.11.3 Execution Frequency

Real Time

2.11.4 Key Scheduling Dependencies

N/A

2.11.5 Counties Impacted

CalSAWS Counties

2.11.6 Category

Real Time API

2.11.7 Data Volume/Performance

N/A

2.11.8 Failure Procedure/Operational Instructions

Operations staff will evaluate transmission errors and failures and determine the appropriate resolution (i.e., manually retrieving the file from the directory and contacting the external partner if there is an account or password issue, etc.)

2.12 Self-Service Portal: Update Application Transfer API Task Bank Assignment Logic

2.12.1 Overview

The Application Transfer API is a RESTful webservice which purpose is to save an applicant's information into the CalSAWS system. Once the applicant information is successfully saved in CalSAWS, a task will generate to alert the worker of the received E-application.

This section outlines the necessary modifications in the Application Transfer API to update the Task Bank task assignment logic when attempting to assign the E-App task to a Task Bank.

2.12.2 Description of Change

1. Update the Application Transfer API by modifying the existing Task Bank logic assignment to check if Task Bank Id has an Active status for the County prior to assigning a task. If a valid task assignment cannot be determined, the task will not be created. **Note**: No error response code will be returned to the Self-Service Portal, thus the data transfer from the Self-Service Portal will still occur regardless of the task generation outcome.

2.12.3 Execution Frequency

Real Time

2.12.4 Key Scheduling Dependencies

N/A

2.12.5 Counties Impacted

CalSAWS Counties

2.12.6 Category

Real Time API

2.12.7 Data Volume/Performance

N/A

2.12.8 Failure Procedure/Operational Instructions

Operations staff will evaluate transmission errors and failures and determine the appropriate resolution (i.e., manually retrieving the file from the directory and contacting the external partner if there is an account or password issue, etc.)

2.13 Self-Service Portal: Update CBO Account API Task Bank Assignment Logic

2.13.1 Overview

The CBO Account API is a RESTful webservice that creates a task when a CBO request an account creation in CalSAWS.

This section outlines the necessary modifications in the Application Transfer API to update the Task Bank task assignment logic when attempting to assign the E-App task to a Task Bank.

2.13.2 Description of Change

1. Update the CBO Account API by modifying the existing Task Bank logic assignment to check if Task Bank Id has an Active status for the County prior to assigning a task.
   a. If a valid assignment is not found for the task, return the existing 404 response code to the Self-Service Portal.
      i. 404 response code description: Could not find a worker or a bank to create a task.

2.13.3 Execution Frequency

Real Time

2.13.4 Key Scheduling Dependencies

N/A

2.13.5 Counties Impacted

CalSAWS Counties

2.13.6 Category

Real Time API

2.13.7 Data Volume/Performance

N/A

2.13.8 Failure Procedure/Operational Instructions

Operations staff will evaluate transmission errors and failures and determine the appropriate resolution (i.e., manually retrieving the file from the directory and contacting the external partner if there is an account or password issue, etc.)

2.14 Self-Service Portal: Update Task API – Schedule, Reschedule Appointment Task Bank Assignment Logic

2.14.1 Overview

The Task API – Schedule, Reschedule Appointment API is a RESTful webservice that creates a task in CalSAWS when a customer sends an appointment request from Self-Service Portal.

This section outlines the necessary modifications in the Application Transfer API to update the Task Bank task assignment logic when attempting to assign the E-App task to a Task Bank.

2.14.2 Description of Change

1. Update the Task API – Schedule, Reschedule Appointment API by modifying the existing Task Bank logic assignment to check if Task Bank Id has an Active status for the County prior to assigning a task.
    a. If a valid assignment is not found for the task, return the existing 404 response code to the Self-Service Portal.
        i. 404 response code description: Could not find a worker or a bank to create a task.

2.14.3 Execution Frequency

Real Time

2.14.4 Key Scheduling Dependencies

N/A

2.14.5 Counties Impacted

CalSAWS Counties

2.14.6 Category

Real Time API

2.14.7 Data Volume/Performance

N/A

2.14.8 Failure Procedure/Operational Instructions

Operations staff will evaluate transmission errors and failures and determine the appropriate resolution (i.e., manually retrieving the file from the directory and contacting the external partner if there is an account or password issue, etc.)

2.15 FDS-API: Task API – Update Task (V1/V2)

2.15.1 Overview

This API will allow for the update of Task data in CalSAWS.

This section outlines the necessary modifications in the Task API to update the Task details if active Task Bank details are provided.

2.15.2 Description of Change

The Task API will include the following data elements. Please refer to the **tasks.html** document for the technical specifications and data element definitions.

Update the Task API update endpoint (v1 and v2) to modify the existing Task Bank logic assignment to check if Task Bank Id has an Active status for the County prior to updating a task.

2.15.3 Execution Frequency

Real Time

2.15.4 Key Scheduling Dependencies

N/A

2.15.5 Counties Impacted

CalSAWS Counties

2.15.6 Category

Real Time API

2.15.7 Data Volume/Performance

N/A

2.15.8 Failure Procedure/Operational Instructions

Operations staff will evaluate transmission errors and failures and determine the appropriate resolution (i.e., manually retrieving the file from the directory and contacting the external partner if there is an account or password issue, etc.)

2.16 FDS-API: Task API – Create Task (V1/V2)

2.16.1 Overview

This API will allow for the creation of Task data in CalSAWS.

This section outlines the necessary modifications in the Task API to create the Task details if active Task Bank details are provided.

2.16.2 Description of Change

The Task API will include the following data elements. Please refer to the **tasks.html** document for the technical specifications and data element definitions.

Update the Task API create endpoint (v1 and v2) to modify the existing Task Bank logic assignment to check if Task Bank Id has an Active status for the County prior to creating a task.

2.16.3 Execution Frequency

Real Time

2.16.4 Key Scheduling Dependencies

N/A

2.16.5 Counties Impacted

CalSAWS Counties

2.16.6 Category

Real Time API

2.16.7 Data Volume/Performance

N/A

2.16.8 Failure Procedure/Operational Instructions

Operations staff will evaluate transmission errors and failures and determine the appropriate resolution (i.e., manually retrieving the file from the directory and contacting the external partner if there is an account or password issue, etc.)

2.17 Automated Regression Test

2.17.1 Overview

Create new regression test scripts based on the system test scenarios for the permanent functional changes outlined above.

Note: The search result restriction of the My Banks page, Task Upload functionality, and new status-based page validations are out of scope.

2.17.2 Description of Change

Bank Search & Bank Detail:

1. Create regression scripts to verify the following:
    a. The Status of a Bank can be updated and saved on the Bank Detail page.
    b. Search results can be filtered by Status on the Bank Search page.
    c. The Search Results Summary section of the Bank Search page displays the Status of each matching record.
    d. The Search Results Summary section of the Select Bank page displays the Status of each matching record.
    e. Inactive Banks display in the search results of the Select Bank page and can be selected.

FDS-API: Task API

2. Update the expected results of any existing scripts that leverage the Task API create or update endpoints to exclude inactive Banks as outlined in sections 2.15 and 2.16.
   Technical Notes:
   a. No new scripts will be created to verify the updated task assignment functionality.
   b. No applicable automated regression test scripts existed at the time of design approval. This recommendation is only applicable to new scripts created between the approval and implementation of this SCR.

Self-Service Portal:

3. Evaluate each system test scenario for the potential of automation. Known exclusionary criteria:
   a. Temporary or one-time changes (ex., Data Change Requests, operational batch job execution)
   b. Technical limitations (ex., visual comparison of a static document against a template)
   c. Security restrictions (ex., access to an external service requiring Multi-Factor Authentication)
   d. Required manual intervention (ex., physical printing, document scanning, forced service outage)
4. For each scenario determined to be an automation candidate, modify the system test scenario to be executable as part of the Regression Test Suite. This may include the following:
   a. Repeatability: The script must be able to execute multiple times between data refreshes.
   b. Targetability: The script must fully and accurately verify the actual result against the expected result of the scenario.

3 SUPPORTING DOCUMENTS

Number	Functional Area	Description	Attachment
1	Task Management	Task Upload – Standard Template	Standard Template.xlsx
2	Task Management	Task Upload – County Death Match Template	County Death Match Template.xlsx
3	Task Management	Task Upload – Jail Incarceration Match Template	Jail Incarceration Match Template.xlsx
4	Task Management	Task Upload – Jail Release Referral Template	Jail Release Referral Template.xlsx

4 REQUIREMENTS

4.1 Project Requirements

REQ #	REQUIREMENT TEXT	How Requirement Met
2.19.1.4	The LRS shall generate alerts, reminders, and controls that may not directly affect eligibility.	This modification allows Banks to be deactivated as long as the Bank is not associated to any additional Task configurations.

5 MIGRATION IMPACTS

N/A

6 OUTREACH

N/A

7 APPENDIX

N/A

CalSAWS

California Statewide Automated Welfare System

Design Document

CA-265030

Revision Updates for the MC 221 LA and MC 222 LA in the Template Repository

CalSAWS	DOCUMENT APPROVAL HISTORY	
	Prepared By	Ramon Villarreal-Leal
	Reviewed By	[individual(s) from Build and Test teams that reviewed document]

DATE	DOCUMENT VERSION	REVISION DESCRIPTION	AUTHOR
5/9/2024	1.0	Initial Draft	Ramon Villarreal-Leal
7/29/24	2.0	Content Revision 1 – Add recommendation to truncate SSN for MC 222	Ramon Villarreal-Leal

Table of Contents

1 OVERVIEW

This SCR is to update the two forms MC 221 LA (06/07) and MC 222 (08/07) to the latest state versions (02/14) for both forms.

1.1 Current Design

Currently the forms MC 221 LA (06/07) and MC 222 LA (08/07) are in the template repository.

1.2 Requests

Update MC 221 LA and MC 222 LA to the latest state version (02/14). Due to jurisdictional changes, these forms are now used by all counties.

1.3 Overview of Recommendations

1. Update the MC 221 LA (06/07) to the latest state version (02/14)
2. Update the MC 222 LA (08/07) to the latest state version (02/14)

1.4 Assumptions

1. All existing functionality of these forms are not updated unless specified by the recommendation section of this design.
2. The MC 221 LA (02/14), and MC 222 LA (02/14), are only implemented in English.

2 RECOMMENDATIONS

2.1 Updates to the new MC 221 LA Form Recommendation

2.1.1 Overview

The MC 221 LA form is updated to the latest state version with this recommendation.

State Form: MC 221 (02/14)

Current Programs: Medi-Cal

Current Attached Form(s): N/A

Current Forms Category: Referrals

Current Template Repository Visibility: All Counties

Existing Languages: English

2.1.1 Form Verbiage

The MC 221 will have the following verbiage updates.

<u>Updated Form XDP</u>

Updated Languages: English

Updated Form Number and Version: MC 221 and (02/14)

Form Mockups/Examples: Please see supporting document #1.

Location	Existing Verbiage	Updated Verbiage
On the top right of page 1	Blank	***Please Print*** Retain Copy 4 (Send copies 1, 2, and 3 to DDSD) **DO NOT MAIL TO APPLICANT**
On the bottom left of page 1	Blank	SEE BACK OF COPY 4
On the bottom right of page 1	Blank	[Checkbox] Oakland [Checkbox] Los Angeles

2.1.2 Form Variable Population Logic

Form Body Variables:

Variable Name	Population	Formatting	Editable* / Field Type	Template Repository Population	Populates with Form Generation
Los Angeles Checkbox	This checkbox is always checked	Checkmark	Yes/Checkbox	Yes	No

* Note: The Editable column of the table above refers to if the variable will be editable when populated. When generating a Blank Form from Template Repository the field will be editable unless otherwise indicated.

Variables Requiring Translations: N/A

2.1.3 Form Generation Conditions

There are no changes to this section.

2.2 Updates to the new MC 222 (02/14) Form Recommendation

2.2.1 Overview

This section is to update the MC 222 to the latest state version (02/14).

State Form: MC 222 (02/14)

Current Programs: Medi-Cal

Current Attached Form(s): N/A

Current Forms Category: Forms

Current Template Repository Visibility: All Counties

Existing Languages: English

2.2.2 Form Verbiage

The MC 222 will have the following verbiage updates.

Updated Form XDP

Updated Languages: English

Updated Form Number and Version: MC 222 and (02/14)

Form Mockups/Examples: Please see supporting document #2.

The verbiage updates on this form are all cosmetic changes consisting of "—" and "(   )" for the Social Security and Phone Number.

2.2.3 Form Variable Population Logic

Variable Name	Population	Formatting	Editable* / Field Type	Template Repository Population	Populates with Form Generation
SSN	Prepopulates with the SSN of the person. SSN is truncated when displayed on the form and only shows the last 4 digits.	Arial Font Size 10	Yes – Text field	Yes	Yes

2.2.4 Form Generation Conditions

There are no changes to this section.

3 SUPPORTING DOCUMENTS

Number	Functional Area	Description	Attachment
1	CalSAWS	The new design mockup for form MC 221 LA (02/14)	MC221_EN.pdf
2	CalSAWS	The new design mock up for form MC 222 (02/14)	MC222_EN.pdf

4 REQUIREMENTS

4.1 Project Requirements

REQ #	REQUIREMENT TEXT	How Requirement Met
CAR - 1242	The LRS shall produce notices, NOAs, forms, letters, stuffers, and flyers, either generated by the LRS or initiated by COUNTY-specified Users, that may be sent to an applicant, participant, caregiver, sponsor, authorized representative, Vendor, landlord, and/or any other public or private individual or agency.	Updated both forms to latest state versions.

CalSAWS

California Statewide Automated Welfare System

Design Document

CA-266070

Archive Phase 2 - Journal

CalSAWS	DOCUMENT APPROVAL HISTORY	
	Prepared By	ZhaoA
	Reviewed By	LeS, KrishnamoorthyK, MunaJ

DATE	DOCUMENT VERSION	REVISION DESCRIPTION	AUTHOR
11/9/2023	1.0	Initial Draft	ZhaoA
3/4/2024	1.1	Added BenefitsCal API necessary modifications to use Journal API services	Jennifer Muna
6/3/2024	1.2	Add note on contact center changes in assumptions. Removed section 2.9, which is being realigned to a separate SCR. Response elements updated in section 2.4.4 and 2.5.4.	ZhaoA

Table of Contents

1 OVERVIEW

Currently, CalSAWS hosts all journal data in the primary Oracle database. This SCR will introduce a PostgreSQL database to store all journal tables and data. The tables comprise the following –

JOURNAL_ENTRY
JOURNAL_ENTRY_APPEND
RPTD_CASE_CHNG
JOURNAL_TEMPL
JOURNAL_TEMPL_GRP
JOURNAL_TEMPL_SECT
JOURNAL_ITEM
JOURNAL_SECT_ITEM
JOURNAL_SECT
JOURNAL_TEMPL_GRP_SECT

New lambda APIs will be introduced to manipulate the journal tables in the PostgreSQL DB based on existing access patterns. Online pages, batch jobs, and other APIs which currently access the journal tables will now interact with these tables by queuing requests for the new lambda APIs.

While this SCR is targeted for 24.07 baseline release, there will be a consortium validation period before the code is approved to be enabled in production. To accommodate this validation period, there are two considerations made –

1. The tables will exist in both the existing Oracle and new PostgreSQL DB until approval is granted. The tables will be kept in sync by new batch jobs. These batch jobs will be delivered to production ahead of this SCR via SCR CA-274767. Once approved for enablement, the tables will be dropped from the Oracle database.
2. Flags will be implemented to disable new code changes to online pages, batch jobs, and other APIs from queuing requests to new lambda APIs. While flagged, these components will continue to execute existing code which accesses the journal tables in Oracle database directly. Once approved for enablement, the flag will be removed to enable new code.

1.1 Current Design

Currently, CalSAWS hosts all journal data in the primary Oracle database. These tables are directly accessed by online pages, batch jobs, and APIs.

1.2 Requests

Prepare scripts to drop the journal tables from the Oracle database, which will be deployed to production upon approval. Develop lambda APIs to manipulate journal data in PostgreSQL database based on existing access patterns from online pages, batch jobs, and APIs. Update online, batch, and API code to queue requests to the lambda APIs wherever they currently access the journal tables. Implement flagging to disable the new online, batch, and API code upon production deployment, so that it can be enabled upon approval. Rework existing batch jobs which access the journal tables to access them in a centralized DAO to reduce efforts to update code to queue to new lambda APIs and simplify future work. Update queries used in online, batch, and API code which join tables in Oracle database with the journal tables so that the queries are split into two new components – one query which continues to query only tables in the Oracle database and one request to lambda API to retrieve journal data.

1.3 Overview of Recommendations

1. Prepare scripts to drop the journal tables from the Oracle database.
2. Develop lambda APIs to manipulate journal data in PostgreSQL database based on existing access patterns from online pages, batch jobs, and APIs.
3. Update online, batch, and API code to queue requests to the lambda APIs wherever they currently access the journal tables.
4. Implement flagging to disable the new online, batch, and API code upon production deployment.
5. Rework existing batch jobs which access the journal tables to access them in a centralized DAO.
6. Update queries used in online, batch, and API code which join tables in Oracle database with the journal tables so that the queries are split into two new components – one query which continues to query only tables in the Oracle database and one request to lambda API to retrieve journal data.

1.4 Assumptions

1. Data move will be handled in SCR CA-274767 along with creation of the production PostgreSQL database and scripts to create the journal tables in the new database.
2. Journals linked to cases considered inaction for six or more years will continue to be purged per existing requirements (reference SCR CA-229298), and workers would be able to see only the remaining shell case.
3. Unlike archive phase 1, all data for the identified journal tables will be moved to the new PostgreSQL database. The tables will be dropped from Oracle database upon approval after the validation period.
4. IVR and legacy APIs create journal entries. Updates to these APIs will be implemented in SCR CA-278760 to accommodate planned code changes for these APIs which involve moving code to a new repository and architectural changes.
5. Reports/analytics changes will be aligned to SCR CA-274618.

2 RECOMMENDATIONS

2.1 Journal Architecture

2.1.1 Overview

The update to journal tables and how to access them from online pages requires implementation of new lambda APIs to handle current access patterns and queues for requests to the new lambda APIs.

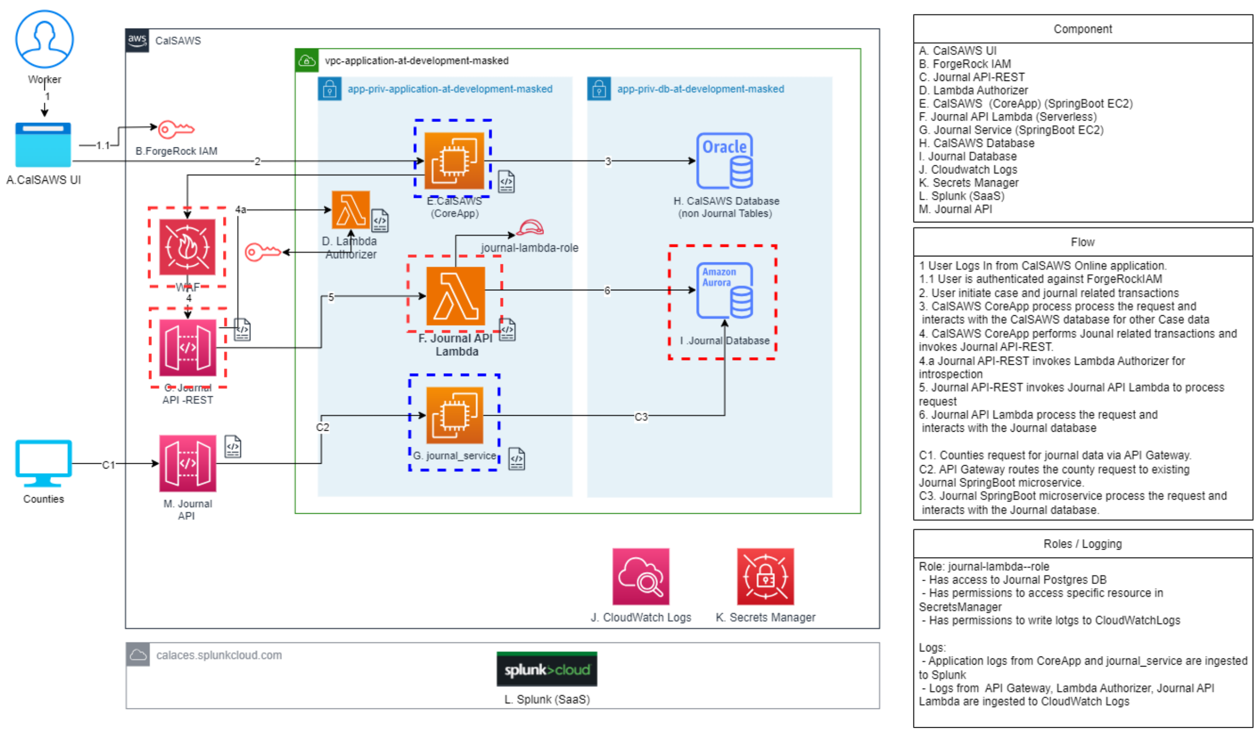


Figure 1 – Architecture Diagram for Online Access to Journal Tables

The same queue will be used to queue requests to the new lambda APIs from batch jobs.

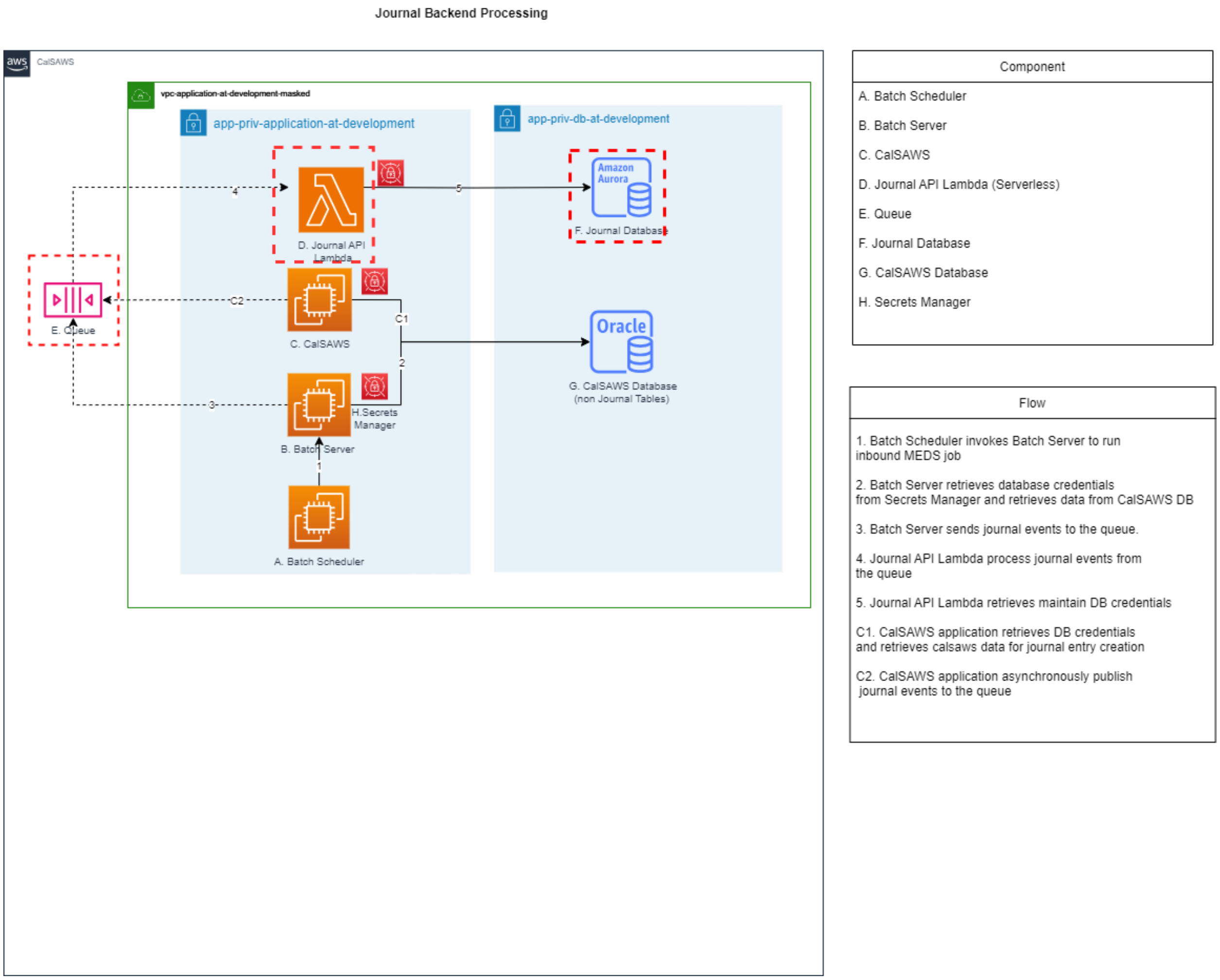


Figure 2 – Architecture Diagram for Batch Job Access to Journal Tables

2.1.2 Description of Change

1. Reference figure 1 for following descriptions of the online flow for journal database access.
    a. Online page – Worker accesses journal related page (i.e. page triggers journal creation, journal search, etc.)
    b. Queue – Journal related access request is pushed to queue from online page
    c. Journal Service – Journal service processes the events from the queue, accessing the Journal RDS database
2. Reference figure 2 for following descriptions of the batch flow for journal database access.
    a. Batch job – Job pushes journal event to queue
    b. Queue – Same queue as 1(b)
    c. Journal Service – Same as 1(c)

2.2 Journal API - Search

2.2.1 Overview

This API will allow for retrieval of journal entries satisfying search criteria that workers input in the journal search online page. This service is intended to populate the journal entry results for the journal search online page.

2.2.2 Description of Change

The Journal API will include the following data elements. Please refer to the journalsMS.yaml document for the technical specifications and data element definitions.

2.2.3 Request Body

The following data elements are included in the Journal API request for the search operation.


```
[
    {
        "startDate": "2024-02-29",
        "endDate": "2024-02-29",
        "category": "string",
        "journalType": "string",
        "countyCode": "string",
        "offsetParam": 0,
        "limitParam": 1,
        "sortParam": "string",
        "orderParam": "string"
    }
]
```


2.2.4 Response

The search operation will return the following data elements for each journal entry that meets the criteria specified in the request:

```
[
    {
        "journalId": 0,
        "orgId": 0,
        "caseNum": "string",
        "caseName": "string",
        "countyCode": "string",
        "typeCode": "string",
        "shortDescr": "string",
        "longDescr": "string",
        "workerNum": "string",
        "classCode": "string",
        "contactTypeCode": "string",
        "filterCode": "string",
        "createdOn": "2024-02-29",
        "initiatedBy": "string",
        "entryAppend": [{
            "appendId": "string",
            "appendDescr": "string",
            "createdOn": "2024-02-29",
            "createdBy": "string",
            "supprInd": "string",
            "workerNum": "string",
        }]
    }
]
```

2.2.5 Headers

The following headers are required and included in all operations by the application gateway and do not need to be included in the request.

X-County-Code: This header is added to all requests by the application gateway. The county code header is used to limit returned values to be specific to the county.

X-App-Id: This header is added to all requests by the application gateway. This is a unique App-id that is created for each county application during the onboarding process. This Id should be referenced against a mapping table with staff Ids (one App-Id per staff id) that will in turn be used for database audit fields (created_by, updated_by) when an application creates or updates data in the CalSAWS system.

2.3 Journal API - Find

2.3.1 Overview

This API will allow for retrieval of a journal entry given the journal entry ID.

2.3.2 Description of Change

The Journal API will include the following data elements. Please refer to the journalsMS.yaml document for the technical specifications and data element definitions.

2.3.3 Request Body

The following data element is included in the Journal API request for the search operation: journalId.

2.3.4 Response

The find operation will return the same data elements as the search operation (see section 2.2.4). In contrast to search, data elements will be returned for only one journal entry.

2.3.5 Headers

The following headers are required and included in all operations by the application gateway and do not need to be included in the request.

X-County-Code: This header is added to all requests by the application gateway. The county code header is used to limit returned values to be specific to the county.

X-App-Id: This header is added to all requests by the application gateway. This is a unique App-id that is created for each county application during the onboarding process. This Id should be referenced against a mapping table with staff Ids (one App-Id per staff id) that will in turn be used for database audit fields (created_by, updated_by) when an application creates or updates data in the CalSAWS system.

2.4 Journal API - Create

2.4.1 Overview

This API will allow for creation of a journal entry.

2.4.2 Description of Change

The Journal API will include the following data elements. Please refer to the journalsMS.yaml document for the technical specifications and data element definitions.

2.4.3 Request Body

The following data element is included in the Journal API request for the create operation:

```
[
    {
        "countyCode": "string",
        "caseNum": "string",
        "category": "string",
        "journalType": "string",
        "shortDescr": "string",
        "longDescr": "string",
        "changeTypes": ["string"],
        "methodOfContact": "string",
        "recoveryAcctId": 0,
        "ssaapId": 0,
        "ssiapId": 0
    }
]
```


2.4.4 Response

The create operation will return the journalEntryId.

2.4.5 Headers

The following headers are required and included in all operations by the application gateway and do not need to be included in the request.

X-County-Code: This header is added to all requests by the application gateway. The county code header is used to limit returned values to be specific to the county.

X-App-Id: This header is added to all requests by the application gateway. This is a unique App-id that is created for each county application during the onboarding process. This Id should be referenced against a mapping table with staff Ids (one App-Id per staff id) that will in turn be used for database audit fields (created_by, updated_by) when an application creates or updates data in the CalSAWS system.

2.5 Journal API - Update

2.5.1 Overview

This API will allow for appending of a journal entry.

2.5.2 Description of Change

The Journal API will include the following data elements. Please refer to the journalsMS.yaml document for the technical specifications and data element definitions.

2.5.3 Request Body

The following data element is included in the Journal API request for the update operation:

```
[
    {
        "journalId": "string",
        "appendDescription": "string",
        "suppressIndicator": "string",
        "classificationCode": "string",
        "creatorName": "string"
    }
]
```


2.5.4 Response

The update operation will not return any data elements.

2.5.5 Headers

The following headers are required and included in all operations by the application gateway and do not need to be included in the request.

X-County-Code: This header is added to all requests by the application gateway. The county code header is used to limit returned values to be specific to the county.

X-App-Id: This header is added to all requests by the application gateway. This is a unique App-id that is created for each county application during the onboarding process. This Id should be referenced against a mapping table with staff Ids (one App-Id per staff id) that will in turn be used for database audit fields (created_by, updated_by) when an application creates or updates data in the CalSAWS system.

2.6 Journal API - Delete

2.6.1 Overview

This API will allow for deletion of a journal entries for a case. This is intended to support existing case purge deletion logic.

2.6.2 Description of Change

The Journal API will include the following data elements. Please refer to the journalsMS.yaml document for the technical specifications and data element definitions.

2.6.3 Request Body

The following data element is included in the Journal API request for the delete operation: caseId.

2.6.4 Response

The delete operation will not return any data elements.

2.6.5 Headers

The following headers are required and included in all operations by the application gateway and do not need to be included in the request.

X-County-Code: This header is added to all requests by the application gateway. The county code header is used to limit returned values to be specific to the county.

X-App-Id: This header is added to all requests by the application gateway. This is a unique App-id that is created for each county application during the onboarding process. This Id should be referenced against a mapping table with staff Ids (one App-Id per staff id) that will in turn be used for database audit fields (created_by, updated_by) when an application creates or updates data in the CalSAWS system.

2.7 Update Online Pages and Batch Jobs to Queue Journal Events

2.7.1 Overview

Currently, all online pages and most batch jobs that access journal tables access them through the same DAO. There are 77 additional batch jobs which access the journal tables through custom DAOs. These additional jobs will be updated to use the centralized DAO to make journal access changes more maintainable.

The centralized DAO will be updated as follows:

a. Existing queries which join tables that will remain in the Oracle database to tables that will be moved to the Journal database will be reworked such that the query only selects from tables in the Oracle database and journal data will be manipulated through the Journal APIs.
b. Methods which access journal tables will instead pass parameters in a request pushed to a new queue for the Journal APIs to process.
c. Existing queries and code will be enabled via flagging until this functionality is approved for production. The flag will be stored in the database so that on approval, the flag can be removed via DCR.

2.7.2 Description of Change

1. The batch jobs and corresponding changes can be found in the following supplemental document "CentralizeBatchInventory.xlsx."

2. Update queries in JournalDaoImpl.java file and helper files so that they do not join tables that will remain in the Oracle database to tables that will be moved to the Journal database. Rework these queries so that they only select from tables in the Oracle database and journal data will be manipulated through the Journal APIs.
3. Access journal table by pushing requests to the new queue for Journal API.
4. Add a flag for journal in the DATA_GRWTH_PARAMTR table to indicate that the functionality is initially disabled.
5. Implement flagging in the JournalDaoImpl.java file so that while disabled, the online pages and batch jobs continue to access the journal tables from the Oracle database using their existing queries. Once flagging is removed, the online pages and batch jobs should access the journal tables via Journal APIs.

2.8 BenefitsCal – Update existing APIs to call Journal API when creating Journal Entry

2.8.1 Overview

CalSAWS has four APIs which creates an automatic Journal Entry based on a specific trigger from the Self-Service Portal which includes the following:

BenefitsCal API	Journal Entry Creation Trigger
Report A Change	When customer reports a change for their case from their Self Service Portal account.
Customer Contact	When customer makes and update from their Self Service Portal account. This includes e-mail address update, e-notification opt-in/opt-out, Phone number update, text notification opt-in.
EBT Theft	When a customer submits a EBT 2259 form from the Self-Service Portal and the data transfer into CalSAWS is successful.
Release of Information (ROI)	When a customer submits a ABCDM 229 form from the Self-Service Portal and the data transfer into CalSAWS is successful.

This section outlines the necessary modifications for the above APIs to begin creating Journal entries through the Journal API.

2.8.2 Description of Change

1. Update the Report a Change, Customer Contact, and EBT Theft API to call the Journal API service when triggered to create a Journal Entry. **Note**: All Journal details (i.e. Journal Category, Journal Short description, etc.) will remain unchanged for this implementation.

3 SUPPORTING DOCUMENTS

Number	Functional Area	Description	Attachment
1	Journal API	API specifications for new Journal lambdas	Refer to journalsMS.yaml attachment in JIRA

CalSAWS

California Statewide Automated Welfare System

Design Document

CA-277526

Update Juvenile Probation Interface Phase II

CalSAWS	DOCUMENT APPROVAL HISTORY	
	Prepared By	Shining Liu, Vallari Bathala
	Reviewed By	Sunitha Sampathkumar, William Baretsky, Chitra Barsagade, Edgars Reinholds, Tisha Mutreja, Justin Dobbs

DATE	DOCUMENT VERSION	REVISION DESCRIPTION	AUTHOR
8/17/2023	1.0	Initial draft	Gerald Limbrick, Shining Liu, Vallari Bathala
5/7/2024	1.1	Split from CA-212925	Shining Liu, Vallari Bathala
7/24/2024	1.2	Content revision – journals to be created on all cases a person is in	Shining Liu

Table of Contents

1 OVERVIEW

The Probation Interface with the Los Angeles Probation department will be enhanced with additional journal entries.

1.1 Current Design

CalSAWS receives a daily inbound file from the Los Angeles Probation department containing the individuals who have been incarcerated. This inbound file is a full draw of Probation Department's database which is information gathered from different sources. Upon receipt of the electronic data, CalSAWS identifies all minors who exist in CalSAWS to determine eligibility by taking the required action to stop cash benefits for matched minors. When a match occurs, a worker is tasked, CalWORKs, CalFresh, General Relief and Medi-Cal programs will receive a case review error because timely action is not taken by the caseworker when a match occurs.

For individuals in which a match is found and there is no end-date present, CalSAWS will update the Household Status List to Temporarily or Permanently Out of the Home. SCR CA-257096 modified the Institutional List and Institutional Detail pages to allow workers to update all end-date related updates so that intervention from the development team is not needed.

1.2 Requests

The Probation Interface will be enhanced with additional journal entries.

1.3 Overview of Recommendations

1. Modify the Probation Interface to do the following:
    a. Only modify a record if the record was last updated by the Probation Interface.
    b. Create new Journal when interface matches an individual.
    c. Create new Journal when interface modifies an individual's Household Status.
    d. Create new Journal when an individual is "Released".
    e. Create new Journal when an individual is "Deceased".
2. Create new Automated Action when the Probation Interface skips an update.

1.4 Assumptions

1. Matching criteria for the Probation Interface will not be updated in this SCR.
2. The Ethnicity field from the Probation Interface is not and will not be stored or displayed in CalSAWS in this SCR.
3. Los Angeles is the only county using the Probation Interface.

4. The following code(s) received as the Juvenile Hall Release Type, Camp Release Type, or Stop Placement Code indicates that an individual has been released.
    a. Released – Home (HOME)
    b. Placement Home (PLCH)
    c. DJJ Home (DHOME)
    d. Graduate Home (GRADHOME)
    e. Return Home (RETHOME)
5. The following code(s) received as the Juvenile Hall Release Type, Camp Release Type, or Stop Placement Code indicates that an individual is deceased.
    a. Released – Died While In Custody (EXP)
6. The Probation Update Skipped: Review Automated Action will not generate Tasks until the Automated Action is set to Active status.
7. If the PDJ number is edited through the Juvenile Detail page and the old PDJ number is received through the inbound Probation Interface, they will be treated independently per current functionality.
8. There are no changes to the outbound Probation Interface.
9. There are no changes to current functionality for CalWORKs, CalFresh, General Relief and Medi-Cal programs. When a minor is incarcerated and a match is found in CalSAWS, cash benefits are stopped and Medi-Cal benefits are suspended.
10. SCR CA-212925 will update the Juvenile Detail page to display and allow editing of the PDJ Number.

2 RECOMMENDATIONS

2.1 Inbound Probation Interface Updates

2.1.1 Overview

The inbound Probation interface has two modules. The first module will read the inbound file from the interface partner and look for a match within the CalSAWS system. The second module will modify the matched individuals' Household status and Living Arrangements.

2.1.2 Description of Change

1. Modify the inbound Probation Interface to update a record only if it was last updated by the Probation Interface. If there is a match and the record should be updated, but the record was not last updated by the Probation Interface, invoke the Automated Action that is detailed in Section 2.2. If the Case already has a Task in an Assigned or In Process Status, then do not invoke the Automated Action.
    a. Do not trigger any Journal Entry if the Automated Action detailed in Section 2.2 is invoked, or if there is already a Task in an Assigned or In Process Status.
2. Trigger a Journal Entry when the inbound Probation Interface finds a match. This journal should be triggered when a match is found for the first time (i.e. no record yet exists) and when the record was last updated by the Probation Interface.
    a. Create new Journal Entry -- Probation Demographic Match
        i. New/Update: New
        ii. Category: All
        iii. Journal Type: Probation Demographic Match
        iv. Short Description: Probation file match found
        v. Long Description: <FirstName LastName> - CIN <CIN#> has been matched to the probation file with run date <Run Date>.
            1. CIN is not provided in the inbound file. This value should come from CalSAWS since a match was found in the system.
            2. <Run Date> should have a format of MM/DD/YYYY.
        vi. Trigger Condition: Inbound Probation Interface has found a match within the CalSAWS system.
    b. If the matching person is on multiple cases, the journal should be generated for all cases the person is on.
3. Trigger a Journal Entry when the inbound Probation Interface modifies an individual's Household Status.
    a. Create new Journal Entry – Probation Household Status

        i. New/Update: New
        ii. Category: All
        iii. Journal Type: Probation Household Status
        iv. Short Description: Probation household status modified
        v. Long Description: Household status for <FirstName LastName> - CIN <CIN#> by probation interface with run date <Run Date>.
            1. CIN is not provided in the inbound file. This value should come from CalSAWS since a match was found in the system.
            2. <Run Date> should have a format of MM/DD/YYYY.
        vi. Trigger Condition: Upon inbound Probation Interface modifying a minor's household status.
    b. If the matching person is on multiple cases, the journal should be generated for all cases the person is on.

4. Trigger a Journal Entry when an individual is 'Released'.
    a. An individual is considered 'Released' when the Juvenile Hall Release Type, Camp Release Type, or Stop Placement Code is any one of the following in the inbound interface file:
        i. Released – Home (HOME)
        ii. Placement Home (PLCH)
        iii. DJJ Home (DHOME)
        iv. Graduate Home (GRADHOME)
        v. Return Home (RETHOME)
    b. This journal entry will only be triggered from the inbound Probation Interface and will not be triggered from Online by worker action.
    c. Create new Journal Entry -- Probation Released
        i. New/Update: New
        ii. Category: All
        iii. Journal Type: Probation Released
        iv. Short Description: CalSAWS received a release date
        v. Long Description: On <Run Date>, CalSAWS received a release date of <Release Date> from the Probation interface for the following <CIN#> and <DOB>.
            1. CIN is not provided in the inbound file. This value should come from CalSAWS since a match was found in the system.
            2. <Run Date>, ~~and~~ <Release Date>, and <DOB> should have a format of MM/DD/YYYY.
            3. Use existing functionality that determines the probation end date to populate <Release Date>.
        vi. Trigger Condition: Upon inbound Probation Interface processing a record for a minor whose release type or placement code within the file is any one of the following:
            1. Released – Home (HOME)

            2. Placement Home (PLCH)
            3. DJJ Home (DHOME)
            4. Graduate Home (GRADHOME)
            5. Return Home (RETHOME)
    d. If the matching person is on multiple cases, the journal should be generated for all cases the person is on.
5. Trigger a Journal Entry when an individual is 'Deceased'.
    a. An individual is considered 'Deceased' when the Juvenile Hall Release Type, Camp Release Type, or Stop Placement Code is 'Released – Died While In Custody' (EXP) in the inbound interface file.
    b. This journal entry will only be triggered from the inbound Probation Interface and will not be triggered from Online by worker action.
    c. Create new Journal Entry -- Probation Deceased
        i. New/Update: New
        ii. Category: All
        iii. Journal Type: Probation Deceased
        iv. Short Description: CalSAWS received a deceased date
        v. Long Description: On <Run Date> CalSAWS received a deceased date of <Deceased Date> from the Probation interface for the following <CIN#> and <DOB>.
            1. CIN is not provided in the inbound file. This value should come from CalSAWS since a match was found in the system.
            2. <Run Date>, ~~and~~ <Deceased Date>, and <DOB> should have a format of MM/DD/YYYY.
            3. Use existing functionality that determines the probation end date to populate <Deceased Date>.
        vi. Trigger Condition: Upon inbound Probation Interface processing a record for a minor whose release type or placement code within the file is 'Released – Died While In Custody' (EXP).
    d. If the matching person is on multiple cases, the journal should be generated for all cases the person is on.

2.1.3 Partner Integration Testing

No.

2.1.4 Execution Frequency

No change. Daily.

2.1.5 Key Scheduling Dependencies

No change.

2.1.6 Counties Impacted

Los Angeles only.

2.1.7 Category

No change.

2.1.8 Data Volume/Performance

N/A

2.1.9 Interface Partner

Los Angeles Probation Department.

2.1.10 Failure Procedure/Operational Instructions

Batch Support Operations staff will evaluate transmission errors and failures and determine the appropriate resolution (i.e., manually retrieving the file from the directory and contacting the external partner if there is an account or password issue, etc.).

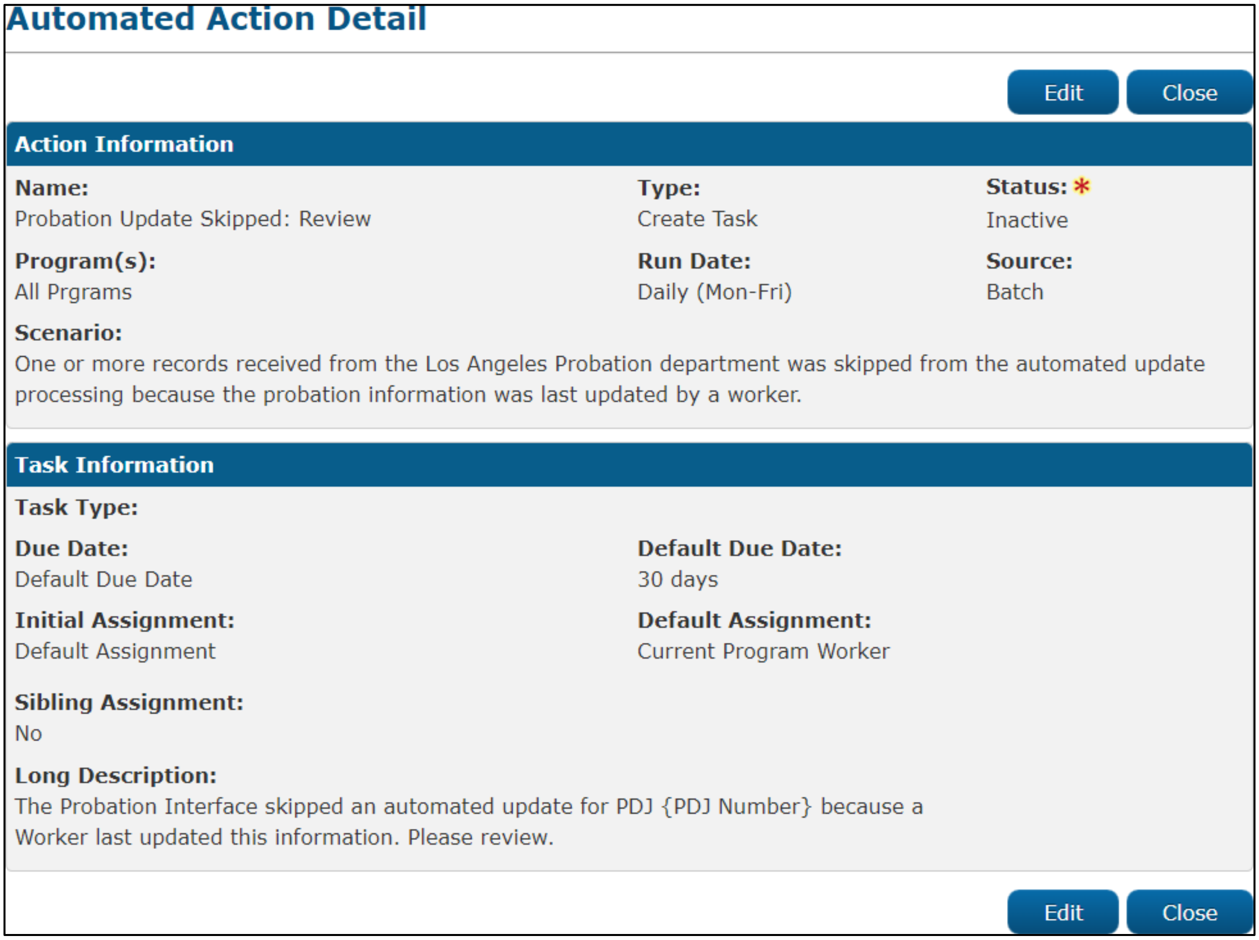
2.2 Probation Update Skipped: Review Automated Action

2.2.1 Overview

The Probation interface has two modules. The first module will read the inbound file from the interface partner and look for a match within the CalSAWS system. The second module will modify the matched individuals' Household status and Living Arrangements.

Modify the Probation Interface to update a record only if it was last updated by the Probation Interface. If there is a match but the record was not last updated by the Probation Interface, generate the task.

2.2.2 Automated Action Detail Mockup



Automated Action Detail

Edit | Close

Action Information

Name:
Probation Update Skipped: Review

Type:
Create Task

Status: *
Inactive

Program(s):
All Prgrams

Run Date:
Daily (Mon-Fri)

Source:
Batch

Scenario:
One or more records received from the Los Angeles Probation department was skipped from the automated update processing because the probation information was last updated by a worker.

Task Information

Task Type:

Due Date:
Default Due Date

Default Due Date:
30 days

Initial Assignment:
Default Assignment

Default Assignment:
Current Program Worker

Sibling Assignment:
No

Long Description:
The Probation Interface skipped an automated update for PDJ {PDJ Number} because a Worker last updated this information. Please review.

Edit | Close

Figure 2.2.2-1 – Automated Action Detail View Mode

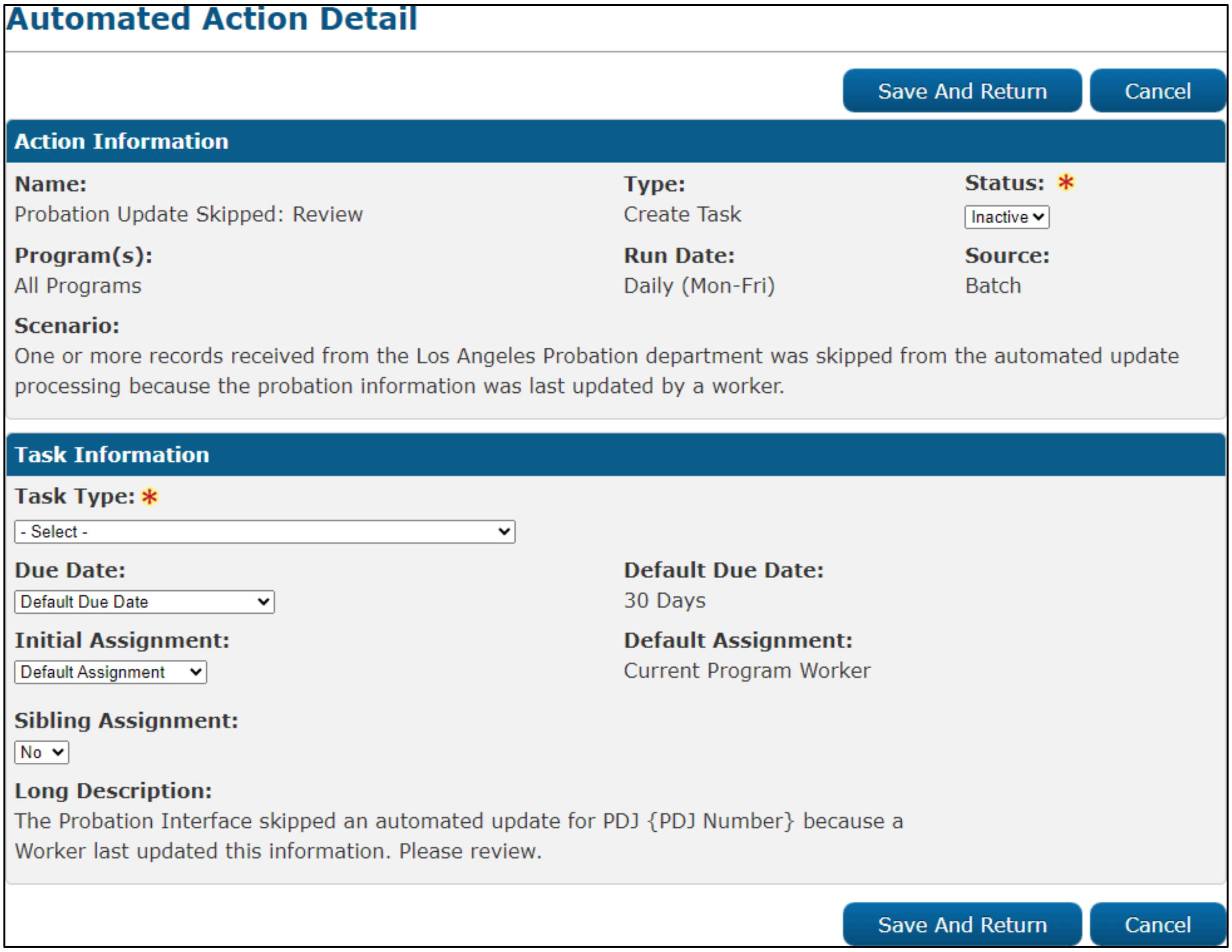


Figure 2.2.2-2 – Automated Action Detail Edit Mode

2.2.3 Description of Changes

Implement a Probation Update Skipped: Review Automated Action for Los Angeles County only.

The Automated Action will have an initial status of Inactive and a blank Task Type. If a county Activates an Automated Action, a Task Type will be required at that time.

If there exists a Medi-Cal program on the Case with a currently assigned Worker, the Default Assignment processing will route to the Medi-Cal worker and associate the Task to the Medi-Cal program. If the Case does not have a Medi-Cal program with a currently assigned Worker, the Default Assignment will route to the worker who is currently assigned to the highest priority program on the Case. The Task will also be associated to this highest priority program.

1. Probation Update Skipped: Review
    a. Action Information

i. Name: Probation Update Skipped: Review
ii. Type: Create Task
iii. Status: Inactive
iv. Program(s): All Programs
v. Run Date: Daily (Mon-Fri)
vi. Source: Batch
vii. Scenario: One or more records received from the Los Angeles Probation department was skipped from the automated update processing because the probation information was last updated by a worker.

b. Task Information
   i. Task Type: BLANK
   ii. Task Sub-Type: BLANK
   iii. Due Date: Default Due Date
   iv. Default Due Date: 30 days
   v. Initial Assignment: Default Assignment
   vi. Default Assignment: Current Program Worker
   vii. Sibling Assignment: No
   viii. Long Description: The Probation Interface skipped an automated update for PDJ {PDJ Number} because a Worker last updated this information. Please review.

2.2.4 Page Validations

N/A

2.2.5 Page Location

Automated Action Detail Page:

- **Global:** Tools
- **Local:** Admin
- **Task:** Automated Actions > Task Admin

Click on the hyperlink of the "Probation Update Skipped: Review" Automated Action result displayed in the Automated Actions Search to navigate to the Automated Action Detail page.

The Task Navigation will display if the user profile contains the "AutomatedActionsListView" security right.

2.2.6 Security Updates

N/A

2.2.7 Page Mapping

N/A

2.2.8 Accessibility

The following accessibility enhancements have been identified:

- IDs used in Accessible Rich Internet Applications (ARIA) and labels must be unique.

2.2.9 Page Usage/Data Volume Impacts

N/A

2.3 Automated Regression Test

2.3.1 Overview

Create new automated regression test scripts to verify the static details of the 'Probation Update Skipped: Review' automated action in the context of Los Angeles County, and that this Automated Action does not display for other counties.

2.3.2 Description of Changes

1. Create regression scripts to verify the static details of the 'Probation Update Skipped: Review' automated action when logged in as a Los Angeles County worker.
   Technical Note: Values that can be edited by a worker are out of scope for this verification.
2. Create regression scripts to verify that the 'Probation Update Skipped: Review' automated action does not display when logged in as a worker from a county other than Los Angeles.
   Technical Note: This scenario will be executed in the context of only one other county. The scenario will not be duplicated for all 57 applicable counties.

3 REQUIREMENTS

3.1 Project Requirements

REQ #	REQUIREMENT TEXT	How Requirement Met
2.20.1.8	The LRS shall alert and provide COUNTY-specified details to the worker of discrepancies between LRS Data received via external interfaces and existing applicant, participant, and/or case records, so that the worker can take the necessary action(s) to resolve the discrepancies.	Additional journal entries when matches are found through the Probation Interface should be generated.