Design Document for User Story 7109

As a user I want to commit the note to TIU and once successfully committed, close the note so that it can no longer be edited.

# ASP .NET Pages

### tbipat\_summary.aspx.cs

**Code Type**: C# Codebehind Class for ASP .NET Page

**Uses Classes**: CCPA, CMDWSOps, CPWEvent, CTemplate, CDataConverter, CInstrumentResponse, StringBuilder, XslCompiledTransform, List<object>

**Uses PL/SQL Procedures**: N/A

**Uses MDWS Calls**: N/A

**Lines of Code**: 421

protected void LoadNote(string strPatPWEventID)

* load the note

protected void OnEventNoteCommit(object sender, DirectEventArgs e)

* commit the note to tiu

protected void OnEventNoteSave(object sender, DirectEventArgs e)

* save note progress, used while the note is being written but before it is commited

protected void Page\_Load(object sender, EventArgs e)

* page load of the patient summary screen

# ASP .NET User Controls

N/A

# ASP .NET Classes

### CPWEvent.cs

**Code Type**: C# Class

**Uses Classes**: CDataParameterList, CDataSet

**Uses PL/SQL Procedures**: PCK\_PWEVENT.WriteNote, PCK\_PWEVENT.SaveNoteProgress, PCK\_PWEVENT.GetNextPatPWEventRS, PCK\_PWEVENT.GetOutcomeMeasuresRS, PCK\_PWEVENT.GetEventEntryRS, PCK\_PWEVENT.GetIntakeScoreRS, PCK\_PWEVENT.GetEventEntryTypeRS, PCK\_PWEVENT.GetEventStatusRS, PCK\_PWEVENT.GetEventSeverityRS, PCK\_PWEVENT.InsertPWEventEntry, PCK\_PWEVENT.UpdatePWEventEntry, PCK\_PWEVENT.UpdatePWEventPlan, PCK\_PWEVENT.DeletePWEventEntry

**Uses MDWS Calls**: N/A

**Lines of Code**: 342

public bool WriteNote(BaseMaster BaseMstr,

long lPatPWEventID,

string strDFN,

string strDUZ,

string strEncounterID,

string strConsultIEN,

string strCosignerDUZ,

string strNoteTitleIEN,

string strTIUNoteID,

string strTIUNote,

long lClinicID,

long lEncounterType,

string strNoteTitle,

string strLocation,

string strCosigner,

string strConsult,

string strTBINote)

* write the tiu note to TBI after written to MDWS

public bool SaveNoteProgress(BaseMaster BaseMstr,

long lPatPWEventID,

long lEncounterType,

string strNoteTitle,

string strLocation,

string strCosigner,

string strConsult,

string strNote)

* save note progress, used while the note is being written but before it is commited

# C++ Classes

N/A

# Oracle PL/SQL

### pck\_pwevent.bdy

**Lines of Code**: 811

procedure WriteNote (

pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

pi\_vPatientID in varchar2,

pi\_nPatPWEventID in number,

pi\_vDFN in varchar2,

pi\_vDUZ in varchar2,

pi\_vEncounterID in varchar2,

pi\_vConsultIEN in varchar2,

pi\_vCosignerDUZ in varchar2,

pi\_vNoteTitleIEN in varchar2,

pi\_vTIUNoteID in varchar2,

pi\_vTIUNote in clob,

pi\_nClinicID in number,

pi\_nEncounterType in number,

pi\_vNoteTitle in varchar2,

pi\_vLocation in varchar2,

pi\_vCosigner in varchar2,

pi\_vConsult in varchar2,

pi\_vTBINote in clob,

po\_nStatusCode out number,

po\_vStatusComment out varchar2

)

* write note details to tbi cds after commited to MDWS

procedure SaveNoteProgress (

pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

pi\_vPatientID in varchar2,

pi\_nPatPWEventID in number,

pi\_nEncounterType in number,

pi\_vNoteTitle in varchar2,

pi\_vLocation in varchar2,

pi\_vCosigner in varchar2,

pi\_vConsult in varchar2,

pi\_vNote in clob,

po\_nStatusCode out number,

po\_vStatusComment out varchar2

)

* save note progress, used while the note is being written but before it is commited

procedure GetOutcomeMeasuresRS (

pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

pi\_vPatientID in varchar2,

pi\_nPatPWEventID in number,

po\_nStatusCode out number,

po\_vStatusComment out varchar2,

rs out RetRefCursor

)

* Get all outcome measures associated with an event

procedure GetIntakeScoreRS (

pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

pi\_vPatientID in varchar2,

pi\_nIntakeID in number,

pi\_nMID in number,

po\_nStatusCode out number,

po\_vStatusComment out varchar2,

rs out RetRefCursor

)

* Get all scores for an intake

procedure GetEventEntryRS (

pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

pi\_vPatientID in varchar2,

pi\_nPatPWEventID in number,

po\_nStatusCode out number,

po\_vStatusComment out varchar2,

rs out RetRefCursor

)

* get all event entries for an event

procedure GetEventEntryTypeRS (

pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

po\_nStatusCode out number,

po\_vStatusComment out varchar2,

rs out RetRefCursor

)

* get event entry types Subjective, Objective, Assessment, Plan etc...

procedure GetEventStatusRS (

pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

po\_nStatusCode out number,

po\_vStatusComment out varchar2,

rs out RetRefCursor

)

* get event statuses Better, Same, Worse etc...

procedure GetEventSeverityRS (

pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

po\_nStatusCode out number,

po\_vStatusComment out varchar2,

rs out RetRefCursor

)

* get event severities: mild, moderate, severe etc...

procedure InsertPWEvent ( pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

pi\_nRegionID in number,

pi\_nSiteID in number,

pi\_vPatientID in varchar2,

pi\_nPatPWID in number,

pi\_vEventTitle in varchar2,

pi\_dtEventDate in varchar2,

pi\_nRelatedEventID in number,

po\_nPatPWEventID out number,

po\_nStatusCode out number,

po\_vStatusComment out varchar2

)

* insert a new pathway event

procedure InsertPWEventEntry ( pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

pi\_vPatientID in varchar2,

pi\_nPatPWEventID in number,

pi\_nPWEventModuleID in number,

pi\_nEventEntryTypeID in number,

pi\_dtEventDate in date,

pi\_vEventEntryComment in varchar2,

pi\_nEventStatusID in number,

pi\_nEventSeverityID in number,

po\_nEventEntryID out number,

po\_nStatusCode out number,

po\_vStatusComment out varchar2

)

* insert an event entry

procedure UpdatePWEventEntry ( pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

pi\_nEventEntryID in number,

pi\_nPWEventModuleID in number,

pi\_dtEventDate in date,

pi\_vEventEntryComment in varchar2,

pi\_nEventStatusID in number,

pi\_nEventSeverityID in number,

po\_nStatusCode out number,

po\_vStatusComment out varchar2

)

* update an event entry

procedure UpdatePWEventPlan ( pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

pi\_nPatPWEventID in number,

pi\_nPatPWPlanEventID in number,

pi\_vPlanText in clob,

po\_nStatusCode out number,

po\_vStatusComment out varchar2

)

* update the plan portion of the event

procedure DeletePWEventEntry ( pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

pi\_nEventEntryID in number,

po\_nStatusCode out number,

po\_vStatusComment out varchar2

)

* delete an event entry

function GetPrevIntakeScore(pi\_nMID in number,

pi\_vPatientID in varchar2,

pi\_nIntakeID in number)

return number

* Gets the previous score value for a mid

function GetNextPatPWEventID(pi\_nPatPWEventID in number,

pi\_vPatientID in varchar2)

return number

* Gets the next event id after an event

procedure GetNextPatPWEventRS (

pi\_vSessionID in varchar2,

pi\_vSessionClientIP in varchar2,

pi\_nUserID in number,

pi\_vPatientID in varchar2,

pi\_nPatPWEventID in number,

po\_nStatusCode out number,

po\_vStatusComment out varchar2,

rs out RetRefCursor

)

* Gets next pathway event associated with a patient