Design Document for User Story 6942

As a user I want the ability to review each outcome measure and enter status, severity and comments.

# ASP .NET Pages

N/A

# ASP .NET User Controls

### ucEventEntry.ascx.cs

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

**Uses Classes:** CPWEvent, CDataConverter

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

**Uses MDWS Calls:** N/A

**Lines of Code:** 230

public string EventEntryTypeID

* Event entry type id property

public string EventEntryID

* event entry id

public string FXUserID

* user id who wrote this entry

public string PatPWID

* Patient pathway id property

public string PatPWEventID

* Pathway event id property

public string PWEventModuleID

* Pathway event module id property

public string PWEventEntryEditMode

* edit mode: UPDATE or INSERT

public void OnNewEntry(long lEventEntryTypeID,

string strPatientID,

long lPatPWID,

long lPatPWEventID,

long lPWEventModuleID,

long lStatusID,

long lSeverityID)

* show the event entry popup for a new entry

public void OnSave()

* user clicked the save button

public void OnCancel()

* user clicked the cancel button

public void OnEditEntry(string strJsonRecord)

* edit an entry

public void OnDeleteEntry(string strJsonRecord)

* delete an entry

public bool InitializeControl(long lPatPWEventID)

* initialize the control

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

* page load

### ucOutcomeMeasures.ascx.cs

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

**Uses Classes:** CPWEvent, }

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

**Uses MDWS Calls:** N/A

**Lines of Code:** 141

public void OnReviewCommand(string strJsonRecord)

* review an entry from the entry popup

# ASP .NET Classes

### CPWEvent.cs

**Code Type:** C# Class

**Uses Classes:** CDataParameterList, CDataSet

**Uses PL/SQL Procedures:** 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.DeletePWEventEntry

**Uses MDWS Calls:** N/A

**Lines of Code:** 199

public DataSet GetEventEntryDS(BaseMaster BaseMstr,

long lPatPWEventID)

* Get list of event entries for an event

public DataSet GetEventEntryTypeDS(BaseMaster BaseMstr)

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

public DataSet GetEventStatusDS(BaseMaster BaseMstr)

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

public DataSet GetEventSeverityDS(BaseMaster BaseMstr)

* get list of event severities unk, mild, moderate, severe

public long InsertPWEventEntry(BaseMaster BaseMstr,

long lPatPWEventID,

long lPWEventModuleID,

long lEventEntryTypeID,

DateTime dtEntryDateTime,

string strEventEntryComment,

long lEventStatusID,

long lEventSeverityID)

* insert an event entry

public bool UpdatePWEventEntry(BaseMaster BaseMstr,

long lPatPWEventEntryID,

long lPWEventModuleID,

long lEventEntryTypeID,

DateTime dtEventDate,

string strEventEntryComment,

long lEventStatusID,

long lEventSeverityID)

* update an event entry

public bool DeletePWEventEntry(BaseMaster BaseMstr,

long lPatPWEventEntryID)

* delete an event entry

# C++ Classes

N/A

# Oracle PL/SQL

### pck\_pwevent.bdy

**Lines of Code:** 488

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 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 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