**Archival Database System - First Prototype**

**Team Members:**

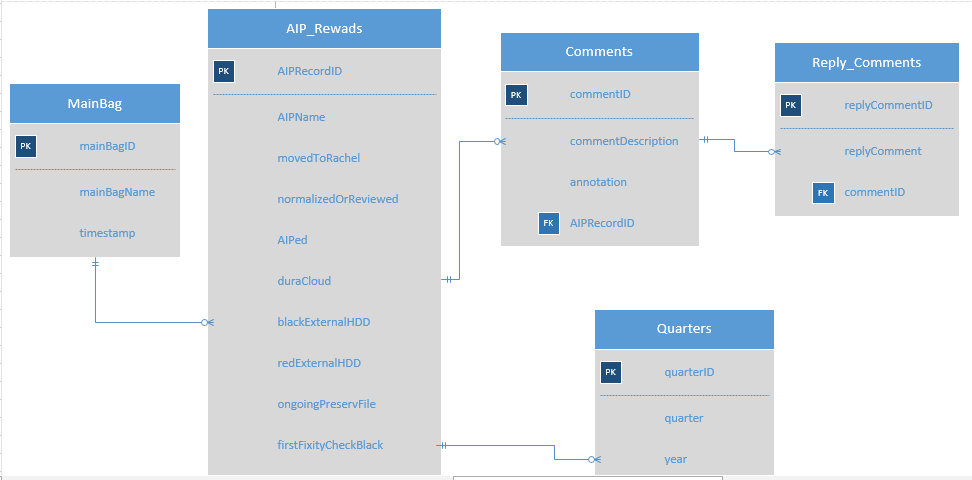
Durga Sparsha Bhagwat

Shafali Gondi

Pavan Jupally

Raihan Ahmed Mohammed

**Entity-Relationship Diagram:**

****

**API Documentation Methods:**

The following methods are used for database access:

* **createAIPRecord** **()**

**Syntax:** createAIPRecord(int aipRecordId, String aipName, int mainBagId, int commentID, int quarterID, bool movedToRachel, bool normalizedOrReviewed, bool aiped, bool duraCloud, bool blackExternalHdd, bool redExternalHdd, bool ongoingPreserveFile, bool firstFixityCheckBlack, bool firstFixityCheckRed)

**Description:** This method creates an AIP Record by taking in the boolean values form the user and aipName from the method getLatestFixityReportName.

* **editAIPRecord()**

**Syntax:** editAIPRecord (int aipRecordId, String aipName, int mainBagId, int commentID, int quarterID, bool movedToRachel, bool normalizedOrReviewed, bool aiped, bool duraCloud, bool blackExternalHdd, bool redExternalHdd, bool ongoingPreserveFile, bool firstFixityCheckBlack, bool firstFixityCheckRed)

**Description:** Edits the AIPRecord based on the the aipRecordId and updates all the other values.

* **deleteAIPRecord()**

**Syntax:** deleteAIPRecord (int aipRecordId)

**Description:** Deletes the AIP Record associated with the aipRecordId.

* **getAIPRecord()**

**Syntax:** getAIPRecord (int aipRecordId)

**Description:** Edits the AIP Record associated with the aipRecordId.

* **getLatestFixityReportName()**

**Syntax:** getLatestFixityReportName (Datetime timestamp)

**Description:** Based on given timestamp the method will check latest fixity report. The mainBagName is concatenated with timestamp and aipName.

* **removeLatestFixityReportName()**

**Syntax:** removeLatestFixityReportName (Datetime timestamp)

**Description:** Based on given timestamp the method will delete the fixity report.

* **getComments()**

**Syntax:** getComments (int aipRecordID)

**Description:** Returns comment based on aipRecordId

* **postComments()**

**Syntax:** postComments (int aipRecordID, String commentDescription, String annotation)

**Description:** It stores the actual commentDescription and the annotation of the comment to the database.

* **editComments()**

**Syntax:** editComments (int aipRecordID)

**Description:** Edits the comments based on the aipRecordId

* **removeComments()**

**Syntax:** removeComments (int aipRecordID)

**Description:** This method deletes the specified comment based on the aipRecordId.

* **getReply()**

**Syntax:** getReply (int replyCommentID)

**Description:** Returns reply of a comment based on the commentId

* **postReply()**

**Syntax:** postReply (int replyCommentID, String replyCommentDescription, int commentId )

**Description:** Saves the reply for a comment specified by the commentId

* **editReply()**

**Syntax:** editReply (int replyCommentID)

**Description:** Edits the reply comment for the replyCommentId

* **removeReply()**

**Syntax:** removeReply (int replyCommentID)

**Description:** Deletes the reply Comment associated with the replyCommentId.