FINAL PROJECT - DATA IMPORT

Week 5

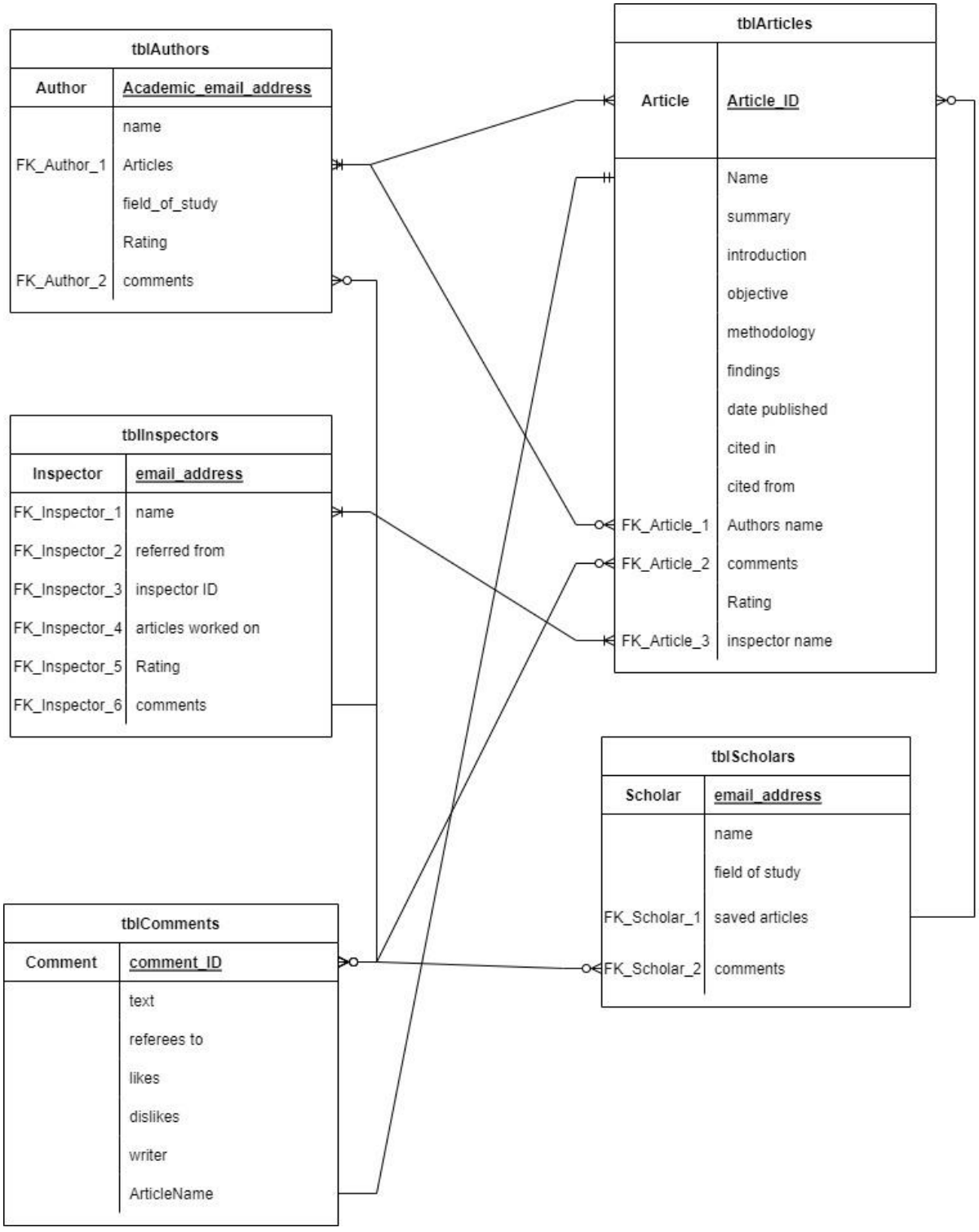
Abstract

In this assignment I explain the Architect of my project and my sql codes

Mohammad Hossein Movahedi

Movahedi.m@northeastern.edu

My Project ER Diagram showed below.



First Of all, I created tables.

```
CREATE TABLE "tblArticles" (

        "Article" INTEGER NOT NULL UNIQUE,

        "Name"    TEXT NOT NULL,

        "Summary "          TEXT NOT NULL,

        "Introduction"      TEXT NOT NULL,

        "Objective"         TEXT NOT NULL,

        "Methodology "      TEXT NOT NULL,

        "Findings"          TEXT NOT NULL,

        "DatePublished"     TEXT NOT NULL,

        "Citedin" TEXT NOT NULL,

        "CitedFrom"         TEXT NOT NULL,

        "AuthorsEmail"      TEXT NOT NULL,

        "Comments"          TEXT,

        "Rating"  NUMERIC NOT NULL,

        "InspectorID"       NUMERIC,

        CONSTRAINT "FK_tblArticles_tblComments" FOREIGN KEY("Comments") REFERENCES
"tblComments"("CommentID"),

        CONSTRAINT "FK_tblArticles_tblInspectors" FOREIGN KEY("InspectorID") REFERENCES
"tblInspectors"("InspectorID"),

        CONSTRAINT "FK_tblArticles_tblAuthors" FOREIGN KEY("AuthorsEmail") REFERENCES
"tblAuthors"("AcademicEmailAddress"),

        CONSTRAINT "PK_tblArticles" PRIMARY KEY("Article" AUTOINCREMENT)

);
```

```
CREATE TABLE "tblAuthors" (

        "AcademicEmailAddress"        TEXT NOT NULL UNIQUE,

        "Name"    TEXT NOT NULL,

        "Articles"          TEXT NOT NULL,

        "FieldOfStudy"      TEXT,

        "Rating"  NUMERIC,
```

```
          "Comments"          TEXT,

          CONSTRAINT "FK_tblAuthors_tblComments" FOREIGN KEY("Comments") REFERENCES
"tblComments"("CommentID"),

          CONSTRAINT "FK_tblAuthors_tblArticles" FOREIGN KEY("Articles") REFERENCES
"tblArticles"("Article"),

          CONSTRAINT "PK_Authors" PRIMARY KEY("AcademicEmailAddress")

);
```

```
 CREATE TABLE "tblComments" (

          "CommentID"         INTEGER NOT NULL UNIQUE,

          "Text"    TEXT NOT NULL,

          "likes"   INTEGER,

          "Dislikes"          INTEGER,

          "Writer"  TEXT,

          "ArticleID"         TEXT,

          CONSTRAINT "FK_tblComments_tblArticles" FOREIGN KEY("ArticleID") REFERENCES
"tblArticles"("Article"),

          CONSTRAINT "PK_tblComments" PRIMARY KEY("CommentID" AUTOINCREMENT)

);
```

```
 CREATE TABLE "tblInspectors" (

          "EmailAddress"      TEXT NOT NULL,

          "Name"    TEXT NOT NULL,

          "ReferredFrom"      TEXT NOT NULL,

          "InspectorID"       INTEGER NOT NULL UNIQUE,

          "ArticlesWorkedOn"  TEXT NOT NULL,

          "Rating"  NUMERIC,

          "Comments"          TEXT,

          CONSTRAINT "FK_tblInspectors_tblArticles" FOREIGN KEY("ArticlesWorkedOn") REFERENCES
"tblArticles"("Article"),

          CONSTRAINT "FK_tblInspectors_tblComments" FOREIGN KEY("Comments") REFERENCES
"tblComments"("CommentID"),

          CONSTRAINT "PK_tblInspectors" PRIMARY KEY("InspectorID" AUTOINCREMENT)

);
```

```
 CREATE TABLE "tblScholars" (

          "EmailAddress"      TEXT NOT NULL,

          "Name"    TEXT NOT NULL,

          "FieldOfStudy"      TEXT,

          "SavedArticles"     TEXT,

          "Comments"          TEXT,

          CONSTRAINT "FK_tblScholars_tblArticles" FOREIGN KEY("SavedArticles") REFERENCES
"tblArticles"("Article"),

          CONSTRAINT "FK_tblScholars_tblComments" FOREIGN KEY("Comments") REFERENCES
"tblComments"("CommentID"),

          CONSTRAINT "PK_tblScholars" PRIMARY KEY("EmailAddress")

);
```

Then I import some sample data.

```
 PRAGMA foreign_keys = 0;

INSERT INTO tblComments (Text,likes,Dislikes,Writer)

VALUES ('The quick brown fox jumps over the lazy dog',10,15, 'Mark Dunn');

INSERT INTO tblComments (Text,likes,Dislikes,Writer,ArticleID)

VALUES ('so good ',100,11, 'Caesar Rodney','Declaration of Independence');

INSERT INTO tblScholars (EmailAddress,Name,FieldOfStudy,SavedArticles,Comments)

VALUES ('a@gmail.com','Mark','IT', 'Declaration of Independence',4);

INSERT INTO tblInspectors
(EmailAddress,Name,ReferredFrom,InspectorID,ArticlesWorkedOn,Rating,Comments)

VALUES ('b@gmail.com','Rex','Randy',1,'Declaration of Independence',5,'The unanimous Declaration
of the thirteen united States of America');

INSERT INTO tblAuthors (AcademicEmailAddress,Name,Articles,FieldOfStudy,Rating)

VALUES ('c@gmail.com','alexander hamilton','Declaration of Independenc','theater',5);

INSERT INTO tblArticles ("Name","Summary ","Introduction","Objective","Methodology
","Findings","DatePublished","Citedin","CitedFrom","AuthorsName","Comments","Rating","InspectorI
D")

VALUES ('Declaration of Independence','he unanimous Declaration of the thirteen united States of
America','We hold these truths to be self-evident','all men are created equal',' whenever any
Form of Government becomes destructive of these ends, it is the Right of the People to alter or
to abolish it','let Facts be submitted to a candid world','2020/2/2','the Rotunda at the
National Archives Museum','Wikipedia','c@gmail.com',3,5,1);
```

```
PRAGMA foreign_keys = 1;
```

The first select function selects the name of the author by the articles

```
 SELECT tblAuthors.Name ,AcademicEmailAddress



FROM tblAuthors



JOIN tblArticles

on tblArticles.AuthorsEmail = tblAuthors.AcademicEmailAddress



WHERE tblArticles.Name = 'Declaration of Independence'
```

The second select function selects the Scholars by their likes

```
 SELECT tblScholars.Name , tblScholars.EmailAddress



FROM tblScholars



JOIN tblArticles

on tblArticles.Name = tblScholars.SavedArticles



WHERE tblArticles.Name = 'Declaration of Independence'
```

The third select function selects the Inspectors by their Article

```
 SELECT tblInspectors.Name,tblInspectors.EmailAddress



FROM tblInspectors

JOIN tblArticles

ON tblArticles.InspectorID = tblInspectors.InspectorID



WHERE tblArticles.Name = 'Declaration of Independence'
```

The fourth select function selects the Article by their Comments

```
 SELECT tblAuthors.Name , tblAuthors.AcademicEmailAddress



FROM tblAuthors



JOIN tblArticles

on tblArticles.AuthorsEmail = tblAuthors.AcademicEmailAddress



JOIN tblComments

on tblComments.ArticleID = tblArticles.Name



WHERE tblComments.ArticleID = 'Declaration of Independence'
```

The fifth select function selects the Article by their Comments

```
 SELECT tblArticles.Name , tblArticles.Rating



FROM tblArticles

JOIN tblInspectors

ON tblArticles.InspectorID = tblInspectors.InspectorID



WHERE tblInspectors.Rating >= 3
```

My database architecture

Due to the nature of the website and the need for it to be expandable and reliable, the best choice for the hosting model would be the cloud-based solution.

The website will start small with just one university as a pilot and then scale up. Still, because the nature of the project is heavily data-centred, maybe we would be a need a medium-size space on a server for the database at the beginning.

The architect of the project is the trickiest part because I want my website to have the ability to use peer-to-peer architecture. At the same time, it stays efficient and straightforward because I don't want to make it always depend on a third party for required computation. Maybe a hybrid solution would be the most beneficial. We start with the server-host model, but we move to the P2P model as we scale up. If this becomes possible, we can have the best of two worlds.

Also, there should be a monthly backup, so if anything happened to the cloud server, there would be another copy of the articles in this way. The authors can trust us better.