University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: ___2_____

Date: _____3.1.2023______

Group Number: ___28_______

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Steve Wang	46916821	v7d7o	privacyalwaysfirst@proton.me
Ali Javed	96043765	e5h0i	ubc.ali.javed@gmail.com
Riley Zhang	91008524	s5p3i	ryyizg8@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

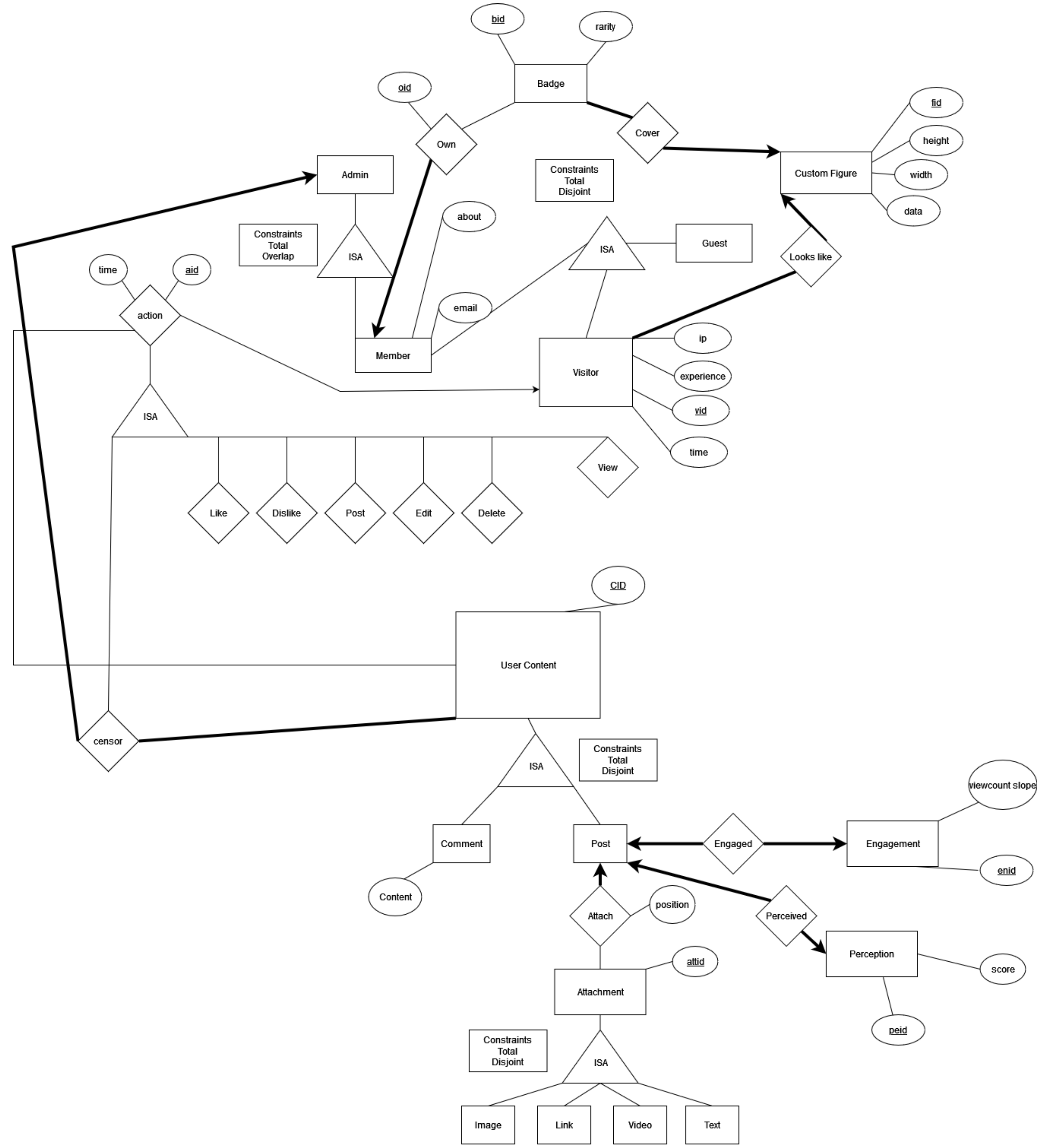
In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Project Description

The project is a forum-style website with the purpose of communication between like-minded individuals about various topics. The application domain is multi-purpose social media, social interaction, and news collections, where users will interact with each other through post-comment and comment-comment relationships. The more a member uses the application, the greater amount of influence they gain within the application and the more they can influence the popularity of other user content.

ER Diagram

We took the TA's advice and redrew the ER Diagram using drawing software instead of by hand. We also added ISA constraints according to the TA's advice.



Schema:

Visitor(ip:string,experience:int NOT NULL,vid:integer PK,time:integer NOT NULL)
Guest(gid:integer PK/FK(Visitor.vid))
Member(mid:integer PK/FK(Visitor.vid),email:string NOT NULL,about:string)
Admin(aid:integer PK/FK(Member.mid))

Badge(bid:integer PK,rarity:integer NOT NULL,name:string NOT NULL)

CustomFigure(fid:integer PK,height:integer NOT NULL,width:integer NOT NULL,data:array NOT NULL)

UserContent(cid:integer PK)
Comment(cid:integer PK/FK(UserContent.cid),content:string NOT NULL)
Post(pid:integer PK/FK(UserContent.cid))

Engagement(eid:integer PK,slope:float NOT NULL)
Perception(peid:integer PK,score:integer NOT NULL)

Attachment(attid:integer PK)
Image(iid:integer PK/FK(Attachment.attid),link:string NOT NULL)
Link(lid:integer PK/FK(Attachment.attid),link:string NOT NULL)
Video(viid:integer PK/FK(Attachment.attid),link:string NOT NULL)
Text(tid:integer PK/FK(Attachment.attid),text:string NOT NULL)

Action(aid:integer PK,time:integer,vid:integer FK(Visitor.vid),cid:integer FK(UserContent.cid))
Like(likeid:integer PK/FK(Action.aid))
Dislike(dislikeid:integer PK/FK(Action.aid))
Post(postid:integer PK/FK(Action.aid))
Edit(editid:integer PK/FK(Action.aid))
Delete(delid:integer PK/FK(Action.aid))
Censor(censorid:integer PK/FK(Action.aid))
View(viewid:integer PK/FK(Action.aid))

Own(oid:integer PK,mid:integer FK(Member.mid),bid:integer FK(Badge.bid))
LooksLike(vid:integer PK/FK(Visitor.vid),fid:integer FK(CustomFigure.fid))
Cover(bid:integer PK/FK(Badge.bid),fid:integer FK(CustomFigure.fid))

Attach(pid:integer PK/FK(Post.pid),attid:integer FK(Attachment.attid),position:integer NOT NULL)
Engage(pid:integer PK/FK(Post.pid),eid:integer FK(Engagement.eid))
Perceived(pid:integer PK/FK(Post.pid),peid:integer FK(Perception.peid))

Functional Dependencies:

vid → vid,experience,time,ip
vid → about, experience, email
aid → aid, time
bid → bid, rarity
fid → fid, height, width, data
cid → content
eid → slope
peid → score

attid → position

Normalization:

Action Table:
Action(aid, time, vid, cid), FD: aid → time

- aid is not a superkey, thus decompose
- Action1(aid, time), Action2(aid, vid, cid) Final answer

Attach Table:
Attach(pid, attid, position), FD: attid → pos

- attid is not a superkey, thus decompose
- Attach1(attid, pos), Attach2(pid, attid) Final Answer

SQL DDL:

```
CREATE TABLE Visitor (
  ip string,
  experience int NOT NULL,
  vid int PRIMARY KEY,
  time int NOT NULL
);

CREATE TABLE Guest (
  gid int PRIMARY KEY REFERENCES Visitor(vid)
);

CREATE TABLE Member (
  mid int PRIMARY KEY REFERENCES Visitor(vid),
  email string NOT NULL,
  about string
);

CREATE TABLE Admin (
  aid int PRIMARY KEY REFERENCES Member(mid)
);

CREATE TABLE Badge (
  bid int PRIMARY KEY,
  rarity int NOT NULL,
  name string NOT NULL
);

CREATE TABLE CustomFigure (
  fid int PRIMARY KEY,
  height int NOT NULL,
  width int NOT NULL,
  data array NOT NULL
);

CREATE TABLE UserContent (
 cid int PRIMARY KEY
);

CREATE TABLE Comment (
 coid int PRIMARY KEY REFERENCES UserContent(cid),
 content string NOT NULL
);

CREATE TABLE Post (
 pid int PRIMARY KEY REFERENCES UserContent(cid)
);

CREATE TABLE Engagement (
 eid int PRIMARY KEY,
 slope float NOT NULL
);

CREATE TABLE Perception (
 peid int PRIMARY KEY,
 score int NOT NULL
);

CREATE TABLE Attachment (
 attid int PRIMARY KEY
);

CREATE TABLE Image (
 iid int PRIMARY KEY REFERENCES Attachment(attid),
 link string NOT NULL
);

CREATE TABLE Link (
 lid int PRIMARY KEY REFERENCES Attachment(attid),
 link string NOT NULL
);

CREATE TABLE Video (
 viid int PRIMARY KEY REFERENCES Attachment(attid),
 link string NOT NULL
);

CREATE TABLE Text (
 tid int PRIMARY KEY REFERENCES Attachment(attid),
 text string NOT NULL
);

CREATE TABLE Action1(
 aid integer primary key,
 time integer
);
CREATE TABLE Action2(
 aid integer primary key,
 vid integer references Visitor(vid),
 cid integer references UserContent(cid)
);
CREATE TABLE Like(
 likeid int PRIMARY KEY REFERENCES Action(aId)
);
CREATE TABLE Dislike(
 dislikeid int PRIMARY KEY REFERENCES Action(aId)
);
CREATE TABLE Makepost(
 postid int PRIMARY KEY REFERENCES Action(aId)
);
CREATE TABLE Edit(
 editid int PRIMARY KEY REFERENCESs Action(aId)
);
CREATE TABLE Delete(
 delid int PRIMARY KEY REFERENCES Action(aId)
);
CREATE TABLE Censor(
 censorid int PRIMARY KEY REFERENCES Action(aId)
);
CREATE TABLE View(
 viewid int PRIMARY KEY REFERENCES Action(aId)
);


CREATE TABLE Own (
  oid int PRIMARY KEY,
  mid int REFERENCES Member(mid),
  bid int REFERENCES Badge(bid)
);

CREATE TABLE LooksLike (
  vid int PRIMARY KEY REFERENCES Visitor(vid),
  fig int REFERENCES CustomFigure(fid)
);

CREATE TABLE Cover (
  bid int PRIMARY KEY REFERENCES Badge(bid),
  fid int REFERENCES CustomFigure(fid)
);

CREATE TABLE Attach1 (
  attid int PRIMARY KEY REFERENCES Attachment(attid),
  position int NOT NULL
);

CREATE TABLE Attach2 (
  pid int PRIMARY KEY REFERENCES Post(pid),
  attid int REFERENCES Attachment(attid),
);

CREATE TABLE Engage (
  pid int PRIMARY KEY REFERENCES Post(pid),
  eid int REFERENCES Engagement(eid)
);

CREATE TABLE Perceived (
  pid int PRIMARY KEY REFERENCES Post(pid),
  peid int REFERENCES Perception(peid)
);
```

INSERT:

Visitor Table:

```
INSERT INTO Visitor      VALUES(“192.010.303”, 50, 10302, 0821)
INSERT INTO Visitor      VALUES(“192.163.093”, 1, 20530, 1430)
INSERT INTO Visitor      VALUES(“192.156.525”, 1354, 778, 0120)
INSERT INTO Visitor      VALUES(“192.321.444”, 642, 1053, 0352)
INSERT INTO Visitor      VALUES(“192.111.593”, 123, 10111, 1252)
```