STORIFY

1. GENERAL

a. Project Description

This is a web based application where authors are provided with a platform to post their work online and can get recognised by publishers. Amateur authors can make use of our auto story-completion service (trained using a generative model) in times when they are stuck with no ideas. A user friendly editor is provided to the authors, which enables saving of partially completed work, along with other editing options. When the articles are posted, it will be automatically tagged if the authors don't provide the genre. Apart from that a user friendly interface is provided to readers, who fancy literature. Also we provide a recommendation system for the readers based on their activities, such as upvoting, frequently reading articles of a particular genre.

i. Background

Amateur authors are not provided with a proper platform to leverage their talents even though there is an abundance of creative work in today's world. Unlike the times where authors spend hours and hours together to come up with new ideas, they can now use our platform. From the end-user perspective, it is rare to find a platform where it is possible to get personalised feed, automatic genre classification and their favorite short-stories all at one place. It is important to use the high-end technology available today to push the limits of one of the most beautiful human art - writing, higher. This is the reason we opted to build this project.

ii. Purpose

- To provide readers with good, interesting and personalized content.
- To provide authors with a platform to show their work
- To help authors in the creative aspect of story-telling.

Iii. Assumptions and Constraints

The following are the assumptions and constraints:

- GPU is available for model training.
- Open-ai allows use of its code and model weights.
- Authors agree to the terms and conditions related to the copyright transfer.
- The model available in the website after deployment is subject to changes depending on the SOTA techniques available in that time period.
- The programming language(python) used in the development of website has been quite dynamic and flexible to changes in the rapidly progressing technological needs of the world making it more probable for future advancements.
- Copyrights of the work created by authors, inspired by the services provided in the website are deemed by site owners and this is made clear in the initial agreement during account creation to avoid any legal issues.

- Readers don’t violate copyright.
- Authors don’t indulge in plagiarism.
- The site is initially made compatible for few standard browsers and the services are later extended to work on several other alternatives.
- The platform is not used to spread rumors under the disguise of story.

iii. Interfaces to External Systems

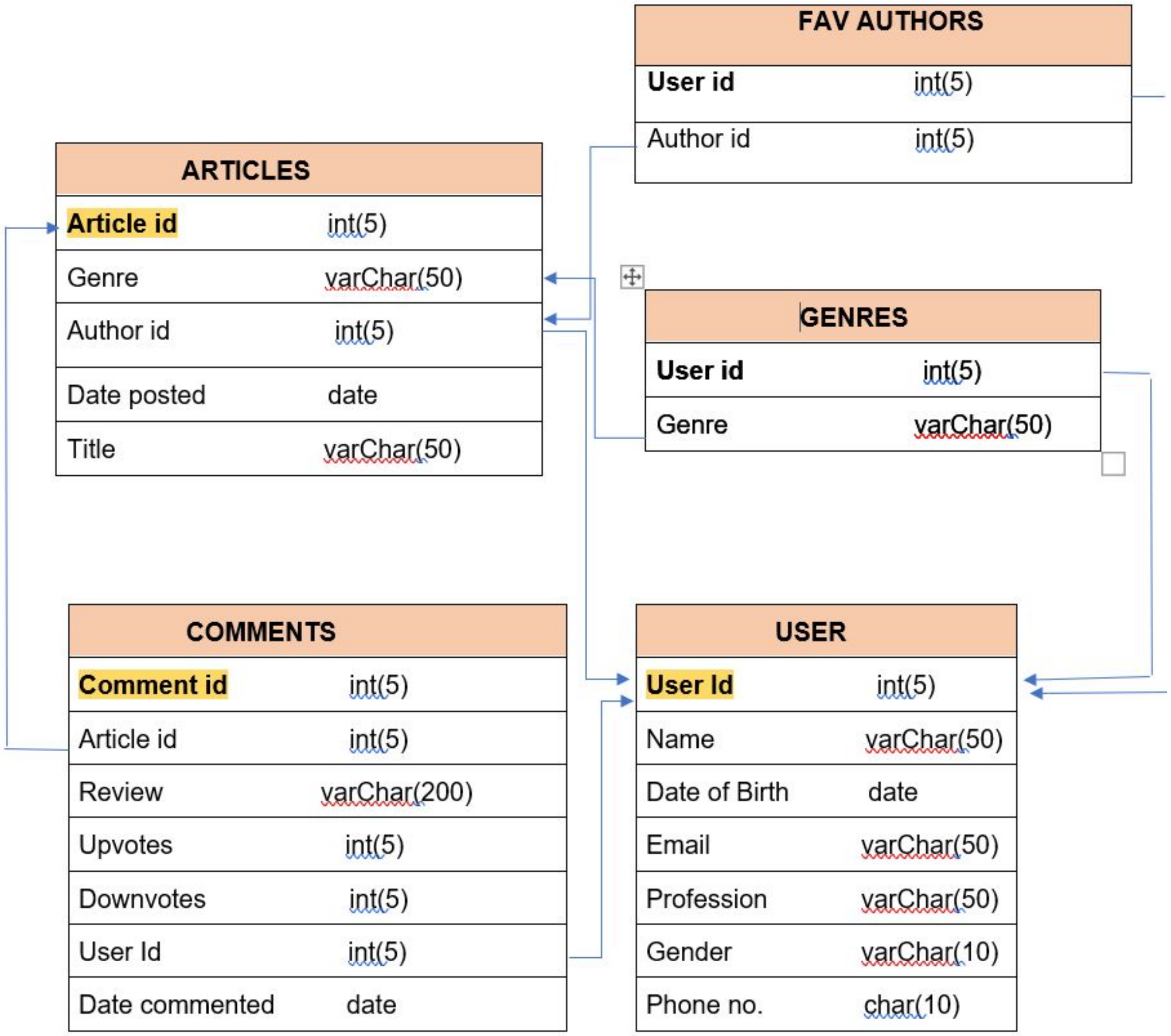
1)Medium : It will interface with our platform to get genre based stories, which will be provided as a service.
2) other writing platforms: Other writing platforms will access the story complete feature which will be provided as a service.

2. FUNCTIONAL REQUIREMENTS

Data Requirements

1) User details - name, mobile, age etc
2) Author details - genre specialization, preferences
3) Payment details - card details , transaction history

Tables in Database:



FAV AUTHORS	
User id	int(5)
Author id	int(5)

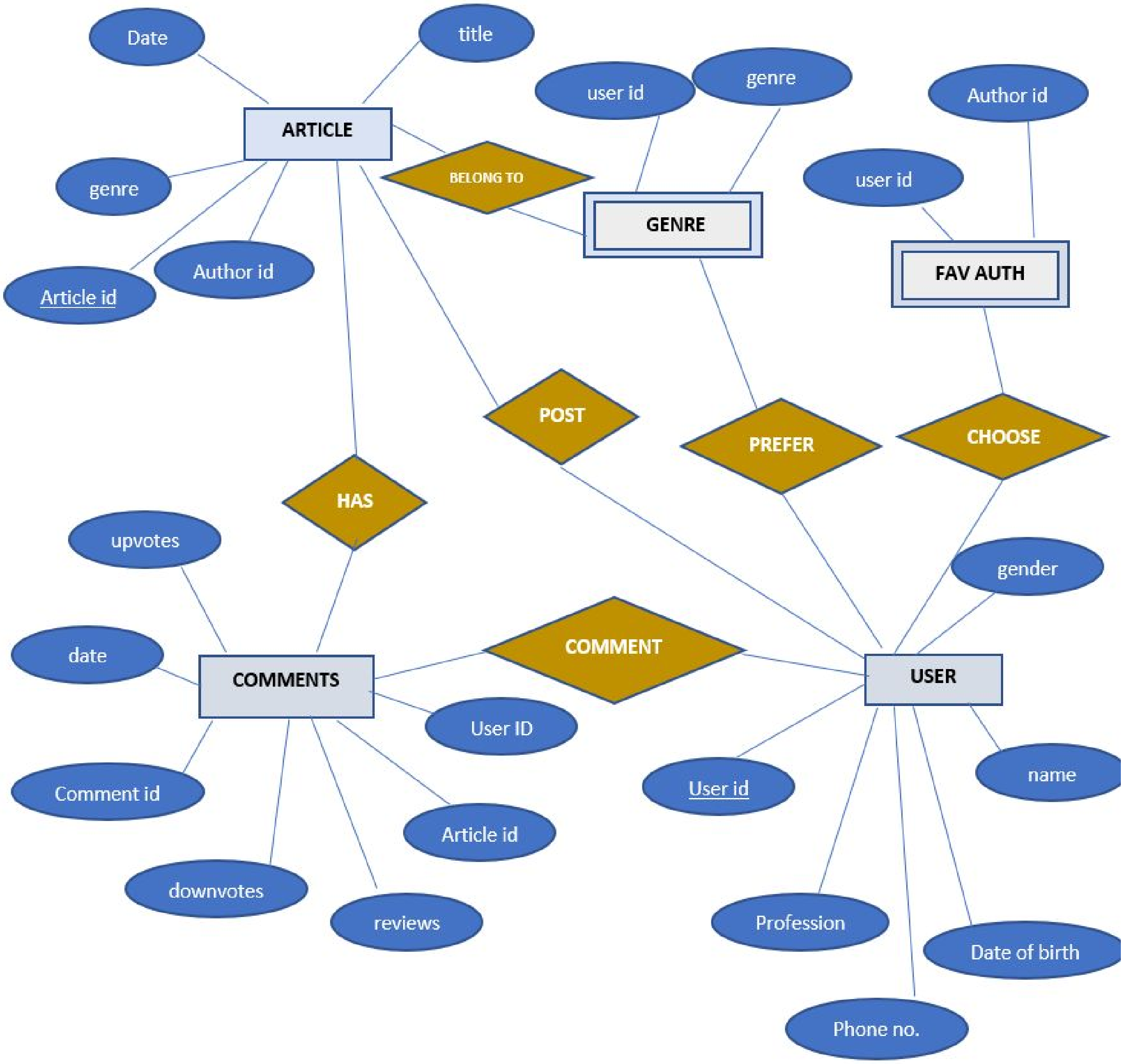
ARTICLES	
Article id	int(5)
Genre	varChar(50)
Author id	int(5)
Date posted	date
Title	varChar(50)

GENRES	
User id	int(5)
Genre	varChar(50)

COMMENTS	
Comment id	int(5)
Article id	int(5)
Review	varChar(200)
Upvotes	int(5)
Downvotes	int(5)
User Id	int(5)
Date commented	date

USER	
User Id	int(5)
Name	varChar(50)
Date of Birth	date
Email	varChar(50)
Profession	varChar(50)
Gender	varChar(10)
Phone no.	char(10)

ER DIAGRAM:



1. **Entities:**

   a. **Articles:**
      i. **Article id (primary key)**
      ii. **Title**
      iii. **Author id (foreign key)**
      iv. **Genre**
      v. **Date posted**

   b. **Comments:**
      i. **Comment id (primary key)**

ii. **User id (foreign key)**
iii. **Review**
iv. **Upvotes**
v. **Downvotes**
vi. **Date**
vii. **Article id (foreign key)**

c. **Users:**

i. **User id (primary key)**
ii. **Name**
iii. **Gender**
iv. **Profession**
v. **Date of birth**
vi. **Phone no.**
vii. **Email**
viii. **Fav author id(foreign key)**

d. **Genres:**

i. **User id (foreign key)**
ii. **Genre**

a. Functional Process Requirements

- Users can register by creating a profile.
- Users can login after entering the correct credentials.
- Users can upload their profile picture.
- The Users must be able to logout after they are finished using the services.
- System will be able to verify the information provided by the user.
- A search filter is provided for browsing authors and genres.
- Personalized user management profile for every user registered.
- Users can upvote and downvote articles, which helps us to determine the quality of the articles posted.
- Authors can edit stories online and offline.
- Authors can use story autocomplete feature.
- Users can mark authors as ‘favorite’.
- Authors can download manuscripts.
- Users can reset password.
- Authors can pay using gateway.
- Other aggregators can get genre based stories on api call.
- The transaction history can be viewed.
- Users can block authors.
- Users can delete their account.
- Authors can delete the stories posted by them.

- Users can comment on articles.
- User can delete comments.
- User can set genre and author preferences.
- Authors can use automatic genre-tagging.
- Authors can manually tag genres for their stories.
- Other writing platforms can get story autocomplete using api calls.
- Authors can directly add auto completed stories to their manuscript.

3. OPERATIONAL REQUIREMENTS

a. Security

- A user will be able to access the content only after login.
- All the passwords should be stored securely and in an encrypted manner.
- All the data packets should be encrypted to avoid man-in-the-middle attack.
- The incomplete works of authors will only be accessible to the authors after login.
- All the creative work should be secured through copyright laws.
- No reader should have access to half-written manuscripts before release.
- Payment information should be secured and not leaked to hackers.
- Unaouthorised web crawlers should not be able to access stories.
- No user without admin privileges can delete an existing article.
- Write functionality is available only to authors.

Consequences of the following breaches of security in the subject application:

- Disclosure of Government secrets - not applicable
- Disclosure of privileged information about individuals - copyright violation

b. Audit Trail

- Payments
- Key strokes
- Likes and dislikes

c. Data Currency

- All stories which are at least 1 hour old must be available for retrieval.
- The training of the story complete model should be at most 1 day old.
- The likes and dislikes should be instantly updated.

d. Reliability

- What damage can result from this system’s failure?
  - o Loss of readership

- Loss of ad revenue.
- Loss of author‘s time, as they won’t be able to upload.

- What is the minimum acceptable level of reliability?

  Reading and searching must be supported at all times.

  Writing and editing must be supported at all times.

- Mean Time Between Failure = 10 day
- Mean Time To Repair = 1hour

e. Recoverability

- In the event the application is unavailable to users (down) because of a system failure, The system should be back up in less than 2 hours.
- The database must be able to restore back to the exact version before the corruption occurred. Getting back the stories is extremely important and hence data concurrency should be maximum.
- If the process site (hardware, data, and onsite backup) is destroyed the system should be back up in 6 hours.

f. System Availability

Our application is available to users for 24 hours everyday except on Monday. On Monday there will be maintenance from 3am-4am IST and the system will be available for the rest of the day.

The time when the usage of the application is expected to be at its peak is between 6:30 pm and 11:30pm IST. This is because most of the people are free in the evenings after finishing their daily chores. So during this time, we ensure the best performance of our system.

g. Fault Tolerance

In case of faults and system failure:

1) All the stories which have already been retrieved will be available.
2) New registrations will be withheld.
3) Authors will be able to edit their stories offline, but not upload.
4) Story autocomplete feature will not be available. It won’t be accessed as many times as search, and hence we can afford to not give this feature during system failures.
5) Search will be available in pre-loaded articles of a genre.

Fault tolerance is applicable for our website because, even when there is a system failure, we can still continue to show the data preloaded from the database. Hence, enabling users a smoother reading experience even at times of failure.

h. Performance

- Response time for autocomplete model should be in minutes.
- Response time for searching an author, story etc should be less than 10 seconds.
- We expect total 1000 searches and reads per second.
- We expect 100 new stories and 10000 new users daily.
- We expect at least 10 million users to search and read stories daily.
- At least 1000 authors will write every day.
- Story auto-complete will be accessed at least 100 times daily

i. Capacity

There are 4 cases in our application where data storage takes place:

1. User details: The system should have the capacity to store details like username, password, activity of 100 million users over a year, as readership is extremely important for success.
2. Articles: The system should support at least 10,000 new articles per genre per month.
3. Comments: All users should be able to comment on all the articles. All the comments should be forever stored in the database.
4. Every author should be able to work on 5 manuscripts at a time.

j. Data Retention

- The data collected from the users - articles, posts, reviews are all stored in such a way that they can be accessed by users anytime.
- To maintain the quality of the content, for every three months, we automatically remove articles which have been continuously downvoted.
- Manuscripts of authors are retained till 2 weeks after deletion.

4. REQUIREMENTS TRACE ABILITY MATRIX

Requirement description	Requirement ID	Verification Method	Priority	Completed (yes/no)
Story auto Complete	1	Judgment	Medium	no
Comment	2	Write a comment	Medium	no
Post	3	Post a completed story	High	no