pastebin-django

Introduction

A pastebin web application programmed in Python using Django web framework. The web application allows both visitors and registered users to upload and view pastes. Registered users can also add comments to pastes, delete their own pastes and mark certain pastes as favorites.

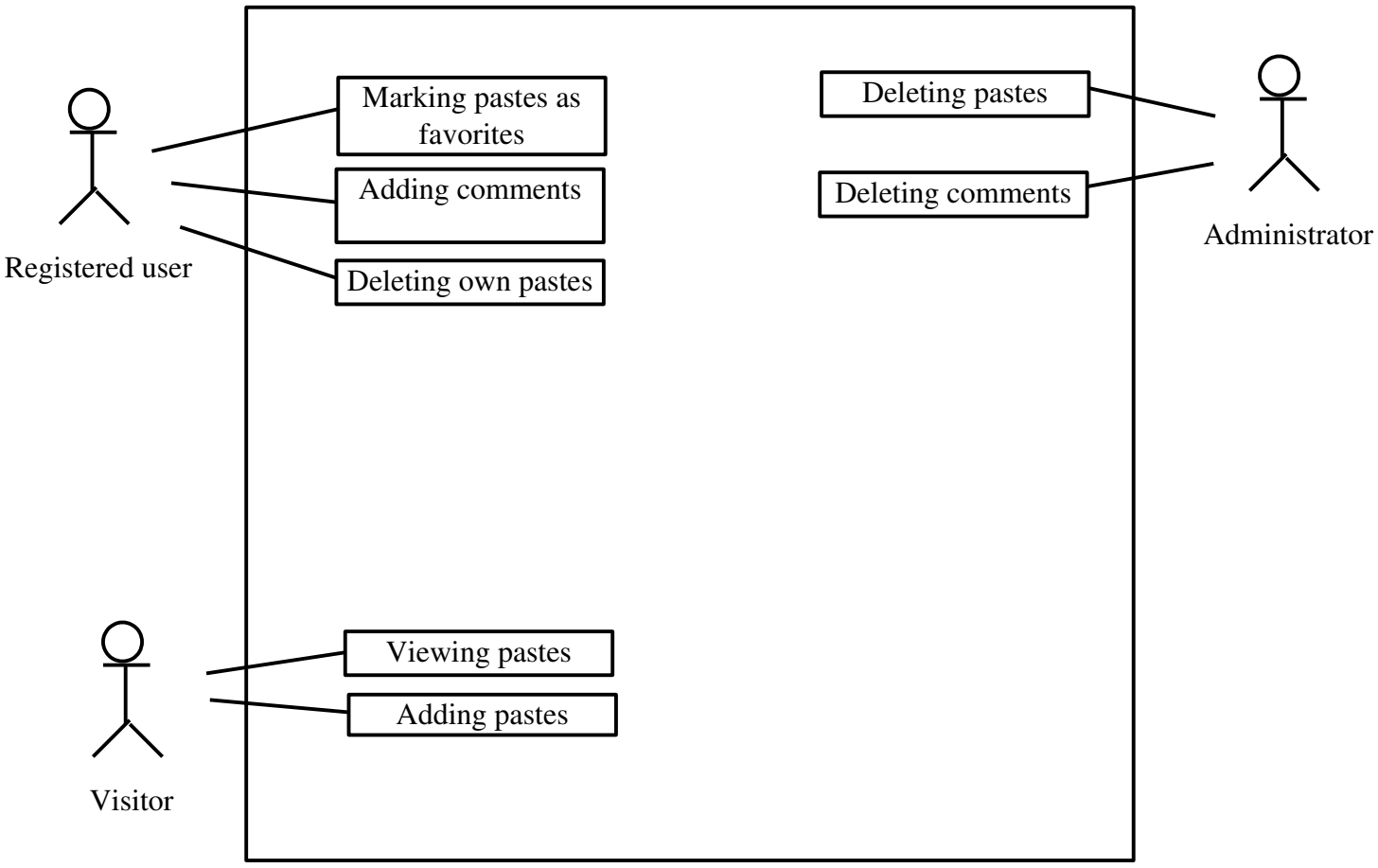
Database

Uses PostgreSQL as the relational database. Data is retrieved, added, updated, etc. using raw SQL queries instead of Django's database-agnostic ORM model.

Other details

Uses Django's in-built authentication system for user registration and login, as this reduces the amount of boilerplate code and is likely more secure than a custom-built solution.

A pastebin web application developed in Python using Django



Uploading a paste

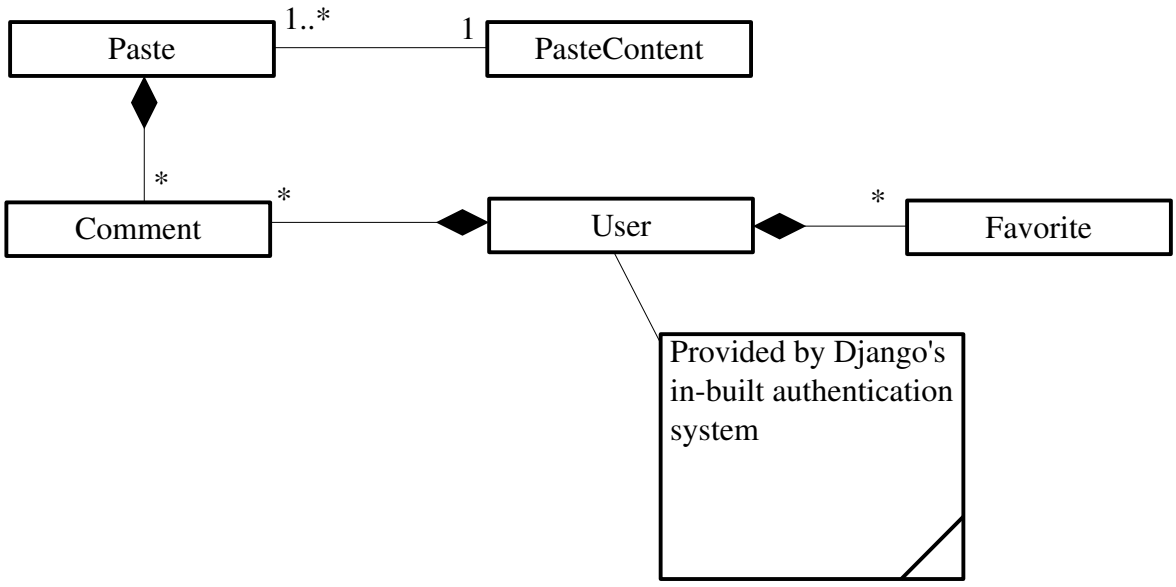
Goal: user uploads a paste which can then be viewed
Users: guest/registered user

1. user fills the form on the front page
2. user submits the form containing the paste's content
    - if a title isn't provided, it is changed automatically to “Untitled”
    - if any text isn't provided, user is returned to the form
3. paste is uploaded and the user is redirected to it

Adding a comment

Goal: user adds a comment to a paste
Preconditions: user is logged in
Users: registered user

1. user fills the comment form
2. user submits the form containing the comment
    - if any text isn't provided, user is returned to the form
3. comment is added and can be viewed by others


Paste
1..*
1
PasteContent
*
Comment
*
User
*
Favorite
Provided by Django's in-built authentication system

Paste

A paste submitted by a guest or a registered user. Text is retrieved based on its hash. Paste may be hidden in which case the only way to view it is to visit the URL directly. Paste may also have an expiration date, after which it can't be viewed at all.

Attribute	Type	Description
id	INTEGER	Primary key of the paste
char_id	CHAR(8)	8-character ID that is used in the URL (eg. G45aaBxy)
submit_user	INTEGER	Foreign key of the user who uploaded the paste, can be NULL if the paste was uploaded by a guest
title	VARCHAR(128)	Title of the paste, defaults to “Untitled” if nothing is provided by the uploader
hash	CHAR(64)	Hash used to identify the paste, which saves space as pastes with duplicate content don't need to be stored twice
expiration_date	TIMESTAMP	A timestamp after which the paste is considered expired. Is NULL if the paste doesn't have an expiration date
hidden	BOOLEAN	Is paste hidden. If the paste is hidden, it won't show up in the Recent pastes list and can only be reached by its URL
submitted	TIMESTAMP	When the paste was uploaded

PasteContent

Paste text is stored as an entry based on its hash, meaning duplicate entries don't take more space.

Attribute	Type	Description
id	INTEGER	Primary key of the paste text
hash	CHAR(64)	Hash identifying the paste
text	VARCHAR(200 000)	The paste text as it was submitted by the user
formatted_text	VARCHAR(400 000)	The paste text formatted in HTML with syntax highlighting (not implemented yet)

Comment

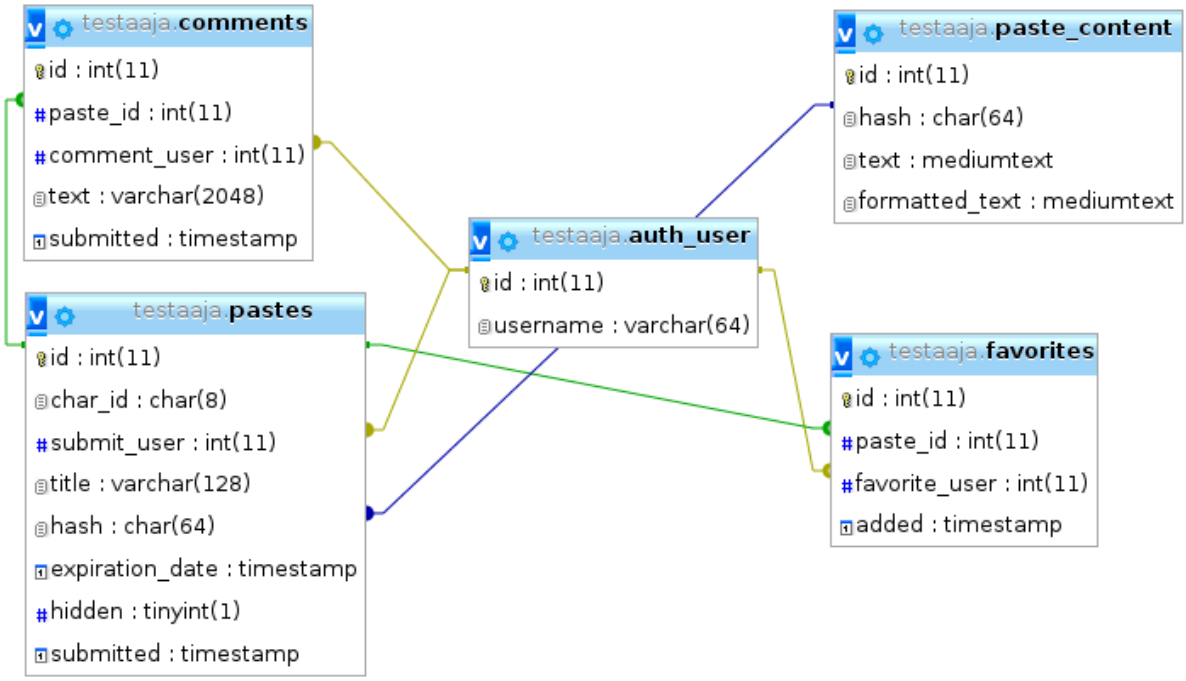
A comment submitted by an user to a certain paste.

Attribute	Type	Description
id	INTEGER	Primary key of the comment
paste_id	INTEGER	Foreign key of the paste this comment was submitted under
comment_user	INTEGER	Foreign key of the user who submitted the comment
text	VARCHAR(2048)	The comment text
submitted	TIMESTAMP	When the comment was submitted

Favorite

Registered users can mark pastes as favorites.

Attribute	Type	Description
id	INTEGER	Primary key of the favorite
paste_id	INTEGER	Foreign key of the paste that was added as a favorite
favorite_user	INTEGER	Foreign key of the user who added the paste as a favorite
added	TIMESTAMP	When the paste was added as a favorite


testaaja.comments
id : int(11)
paste_id : int(11)
comment_user : int(11)
text : varchar(2048)
submitted : timestamp
testaaja.paste_content
id : int(11)
hash : char(64)
text : mediumtext
formatted_text : mediumtext
testaaja.auth_user
id : int(11)
username : varchar(64)
testaaja.pastes
id : int(11)
char_id : char(8)
submit_user : int(11)
title : varchar(128)
hash : char(64)
expiration_date : timestamp
hidden : tinyint(1)
submitted : timestamp
testaaja.favorites
id : int(11)
paste_id : int(11)
favorite_user : int(11)
added : timestamp