

Blog Database Repository (BDR)

Samantha Saxton-Getty

916198535

saxgetty

Milestone / Version	Date
M2V2	11.02.21
M2V1 (New ERD)	10.26.21
M1V1	10.14.21

Table of Contents

Section I: Project Description	3
Section II: Use Cases	4
Section III: Database Requirements (Business Rules)	6
Section IV: Detailed List of Main Entities, Attributes and Keys	9
Section V: Entity Relationship Diagram (ERD)	15
Section VI: Testing Table	16
Section VII: Database Model/EER	20
Section VIII: Forward Engineering	23
Section IX: Inserting Data	23
Section X: Testing	23
Section XI: Testing Table	24

Section I: Project Description

Blog Database Repository (BDR) is a new blog database management system that streamlines and simplifies data sharing through accounts that can create blogs, posts, and blog posts. This product is run on several types of technology, including phones, computers, and tablets. Due to the new design of the software, the company only has over 20 employees. These employees are all administrators that manage the Blog Database Repository (BDR) product. They create and develop a database where users are required to create an account using their information. The types of accounts they can create are general free user, general paid user, blog free user, or blog paid user.

Users can freely comment or like on any media on Blog Database Repository (BDR). This can be done independently of blogs, posts, or blog posts. Having an account allows users to interact with friends through commenting and liking as well. Depending on the permissions associated with the user's account, the user is given access to different Blog Database Repository (BDR) utility.

The Blog Database Repository (BDR) has a look up system that stores searched data for business trend purposes. This data can be used to target specific trends in searches and either store them as categories or tags. Categories are typically more broad such as “animals”, whereas tags are a more specific niche such as “cat”, “dog”, or “horse”.

Independent of the type of account a user has, in the Blog Database Repository (BDR), anyone can view news. However, having an account provides more specific news tailored to your frequent post, blog, and look up data. In addition to this, options are independent as well and allow you to edit the theme and product preferences. Similar to the news, having an account provides a more tailored options experience.

We here at Blog Database Repository (BDR) can't wait to offer a unique and diverse experience as opposed to your typical blog. We invite you to try our new product and all it has to offer!

Section II: Use Cases

Use Case Title:	Account Creation & Perks
Actors:	<ol style="list-style-type: none">1. Account2. Users3. Permission4. Friends
Description:	Sami wants to make a new blog on Blog Database Repository (BDR), but first she needs to create an account. Sami can pick between a general free user, general paid user, blog free user, or blog paid user account. Since Sami is one user, she can only create one account. Creating this account provides one account specific permissions. In addition, this account provides zero or many friends.

Use Case Title:	Making Blogs & Posts
Actors:	<ol style="list-style-type: none">5. Like6. Comment7. Post8. Blog
Description:	Andrew wants to create a blog on Blog Database Repository (BDR). He can create either no blogs or as many blogs as he wants. In addition, Andrew can create either no posts or many posts. Together, Andrew can use this to make blog posts. On blogs, posts, blog posts, as well as any social interaction on BDR, Andrew can not like or like as many as he wants. He can also do this with comments.

Use Case Title:	Product & Technology Management
Actors:	9. Product 10. Technology 11. Administrator
Description:	Maxine uses the Blog Database Repository (BDR) product. This product is run on technology. The types of technology Maxine can run this product on are phones, computers, or tablets. This product is managed by one or many administrators that ensure the product runs smoothly.

Use Case Title:	Storing Data & Seeing Trends
Actors:	12. Look Up 13. Tag 14. Category
Description:	When Matteo uses Blog Database Repository (BDR) and uses the search function, look up stores the associated key(s) looked up either as categories or tags. These look ups can be used to help the company keep track of frequently hit categories and tags. This can help the company increase specific content depending on what users are searching and actively trending.

Use Case Title:	Additional Account Features
Actors:	15. News 16. Options
Description:	Amber can access the news on Blog Database Repository (BDR) either independently of her account otherwise her account will provide access to it anyways. Options are usable on the website to edit the product theme and preferences. These options are usable with or without an account.

Section III: Database Requirements (Business Rules)

Strong Entities:

1. Account
 - a. An account shall be created by only one user.
 - b. An account is a general free user, general paid user, blog free user, or blog paid user.
 - c. An account has zero or many options.
 - d. An account has zero or many permissions.
2. Administrator
 - a. One or many administrators manage one product.
3. Blog
 - a. Zero or many blogs can be created by one or many users.
 - b. Zero or many blogs can blog post zero or many posts.
 - c. Zero or many blogs contain one or many categories.
 - d. Zero or many blogs contain one or many tags.
 - e. One or many blogs hold zero or many comments.
 - f. One or many blogs hold zero or many likes.
4. Category
 - a. One or many categories contain zero or many blogs.
 - b. One or many categories contain zero or many posts.
 - c. Zero or many categories are stored in zero or many look ups.
5. Comment
 - a. Zero or many comments can be created by one or many users.
 - b. Zero or many comments are held by one or many posts.
 - c. Zero or many comments are held by one or many blogs.
6. Friends
 - a. Zero or many friends have one user.
 - b. Zero or many friends are provided by one account.
7. Like
 - a. Zero or many likes can be created by one or many users.
 - b. Zero or many likes can be held by one or many posts.
 - c. Zero or many likes can be held by one or many blogs.
8. Look Up
 - a. One or many look ups store zero or many categories.
 - b. One or many look ups store zero or many tags.

9. News

- a. One or many news is provided by one product.

10. Permission

- a. Zero or many permissions are provided by one account.
- b. Zero or many permissions have one user.

11. Post

- a. Zero or many posts can be created by one or many users.
- b. Zero or many posts can blog post zero or many blogs.
- c. Zero or many posts contain one or many categories.
- d. Zero or many posts contain one or many tags.
- e. One or many posts hold zero or many comments.
- f. One or many posts hold zero or many likes.

12. Product

- a. Zero or one product runs on zero or many technologies.
- b. Zero or one product is managed by zero or many administrators.
- c. One product has zero or many administrators.
- d. One product provides zero or many news.

13. Options

- a. Zero or many options are for one account.

14. Tag

- a. One or many tags contain zero or many blogs.
- b. One or many tags contain zero or many posts.
- c. Zero or many tags are stored in one or many look ups.

15. Technology

- a. One technology runs one product.
- b. One technology is a phone, computer, or tablet.

16. Users

- a. A user shall create one account.
- b. Zero or one user has zero or many friends.
- c. Zero or one user can create zero or many blogs.
- d. Zero or one user can create zero or many posts.
- e. Zero or one user can create zero or many comments.
- f. Zero or one user can create zero or many likes.
- g. Zero or many users have one product.

Weak Entities:

1. Blog Free User
 - a. A blog free user is an account.
2. Blog Paid User
 - a. A blog paid user is an account.
3. Computer
 - a. A computer is a technology.
4. General Free User
 - a. A general free user is an account.
5. General Paid User
 - a. A general paid user is an account.
6. Phone
 - a. A phone is a technology.
7. Tablet
 - a. A tablet is a technology.

Section IV: Detailed List of Main Entities, Attributes and Keys

Strong Entities:

17. Account (Strong)
 - a. account_id: key, numeric
 - b. username: alphanumeric
 - c. password: alphanumeric
18. Administrator (Strong)
 - a. admin_id: key, numeric
 - b. fullname: composite, multivalued, alphanumeric
 - c. tier: numeric
19. Blog (Strong)
 - a. blog_id: key, numeric
 - b. name: alphanumeric
 - c. date_created: composite, date
20. Category (Strong)
 - a. category_id: key, numeric
 - b. type: multivalued, alphanumeric
 - c. count: numeric
21. Comment (Strong)
 - a. comment_id: key, numeric
 - b. date_created: composite, date
 - c. content: alphanumeric
22. Friends (Strong)
 - a. friends_id: key, numeric
 - b. name: alphanumeric
 - c. date_of_birth: composite, date
23. Like (Strong)
 - a. like_id: key, numeric
 - b. count: numeric
 - c. type: multivalued, alphanumeric
24. Look Up (Strong)

- a. look_up_id: key, numeric
- b. type: multivalue, alphanumeric
- c. count: numeric

25. News (Strong)

- a. news_id: key, numeric
- b. title: alphanumeric
- c. body: alphanumeric

26. Permission (Strong)

- a. permission_id: key, numeric
- b. tier: multivalue, numeric
- c. count: numeric

27. Post (Strong)

- a. post_id: key, numeric
- b. subject: alphanumeric
- c. body: alphanumeric

28. Product (Strong)

- a. product_id: key, numeric
- b. name: alphanumeric
- c. date_created: composite, date

29. Options (Strong)

- a. options_id: key, numeric
- b. mode: multivalue, numeric
- c. theme: multivalue, numeric

30. Tag (Strong)

- a. tag_id: key, numeric
- b. handle: alphanumeric
- c. content: alphanumeric

31. Technology (Strong)

- a. tech_id: key, numeric
- b. type: multivalue, alphanumeric
- c. age: numeric

32. Users (Strong)

- a. user_id: key, numeric
- b. first_name: multivalued, alphanumeric
- c. last_name: multivalued, alphanumeric

Weak Entities:

8. Blog Free User (Weak)

- a. blog_free_id: key, numeric
- b. subscription_length: numeric
- c. num_months_subscribed: numeric

9. Blog Paid User (Weak)

- a. blog_paid_id: key, numeric
- b. subscription_length: numeric
- c. num_months_subscribed: numeric

10. Computer (Weak)

- a. computer_id: key, numeric
- b. make: composite, alphanumeric
- c. model: composite, alphanumeric

11. General Free User (Weak)

- a. blog_paid_id: key, numeric
- b. subscription_length: numeric
- c. num_months_subscribed:

12. General Paid User (Weak)

- a. blog_paid_id: key, numeric
- b. subscription_length: numeric
- c. num_months_subscribed: numeric

13. Phone (Weak)

- a. computer_id: key, numeric
- b. make: composite, alphanumeric
- c. model: composite, alphanumeric

14. Tablet (Weak)

- a. tablet_id: key, numeric
- b. make: composite, alphanumeric
- c. model: composite, alphanumeric

Relationships becoming Entities:

1. Blog Post

- a. blog_post_id: key, numeric
- b. blog_id: key, numeric
- c. post_id: key, numeric

2. Containing 1

- a. container_id: key, numeric
- b. category_id: key, numeric
- c. blog_id: key, numeric

3. Containing 2

- a. container_id: key, numeric
- b. category_id: key, numeric
- c. post_id: key, numeric

4. Containing 3

- a. container_id: key, numeric
- b. tag_id: key, numeric
- c. blog_id: key, numeric

5. Containing 4

- a. container_id: key, numeric
- b. tag_id: key, numeric
- c. post_id: key, numeric

6. Creates 1

- a. creation_id: key, numeric
- b. users_id: key, numeric
- c. blog_id: key, numeric

7. Creates 2

- a. creation_id: key, numeric
- b. users_id: key, numeric
- c. post_id: key, numeric

8. Files

- a.

9. Holds 1

- a. holding_id: key, numeric
- b. blog_id: key, numeric
- c. comment_id: key, numeric

10. Holds 2

- a. holding_id: key, numeric
- b. blog_id: key, numeric
- c. like_id: key, numeric

11. Holds 3

- a. holding_id: key, numeric
- b. post_id: key, numeric
- c. comment_id: key, numeric

12. Holds 4

- a. holding_id: key, numeric
- b. post_id: key, numeric
- c. like_id: key, numeric

13. Host

14. Is Ran On

- a. run_id: key, numeric
- b. tech_id: key, numeric
- c. product_id: key, numeric

15. Manages

- a. manage_id: key, numeric
- b. admin_id: key, numeric
- c. product_id: key, numeric

16. Provides 2

- a. provision_id: key, numeric
- b. Account_id: key, numeric
- c. permission_id: key, numeric

17. Provides 3

- a. provision_id: key, numeric
- b. product_id: key, numeric
- c. news_id: key, numeric

18. Stores 1

- a. stores_id: key, numeric
- b. look_up_id: key, numeric
- c. category_id: key, numeric

19. Stores 2

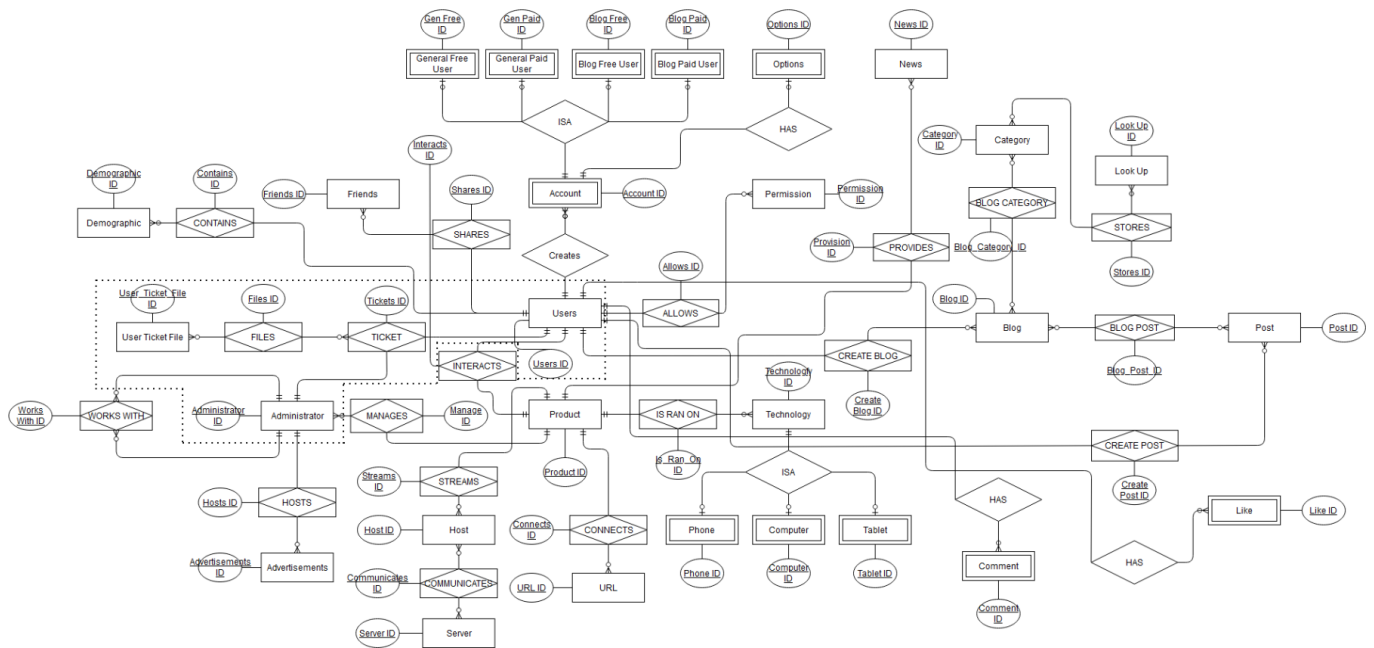
- a. stores_id: key, numeric
- b. look_up_id: key, numeric
- c. tag_id: key, numeric

20. Ticket

21. Works With

Section V: Entity Relationship Diagram (ERD)

UPDATED FOR RE-GRADE



[Direct ERD Link](#)

Section VI: Testing Table

Green = Strong

Blue = Weak

Rule	Entity A	Relation	Entity B	Cardinality	Pass/Fail	Error Description
1	Account	Creates	Users	M-to-N	Fail	One account should be made by one user
2	Account	Provides	Friends	M-to-N	Fail	One account provides zero or many friends
3	Account	Provides	Permission	M-to-N	Fail	One account provides zero or many permissions
4	Account	ISA	GFU, GPU, BFU, or BPU	1-to-M	Fail	Account ISA GFU, GPU, BFU, or BPU
5	Account	Has	Options	M-to-N	Fail	One account has zero or many options
6	Administrator	Has	Product	M-to-1	Pass	None
7	Administrator	Manages	Product	M-to-1	Pass	None
8	Blog	Created By	Users	M-to-N	Pass	None
9	Blog	Blog Post	Post	M-to-N	Pass	None
10	Blog	Contain	Category	M-to-N	Fail	I'm a little unsure how to fix this one. Perhaps

						I make the category zero or many
11	Blog	Contain	Tag	M-to-N	Fail	I'm a little unsure how to fix this one. Perhaps I make the tags zero or many
12	Blog	Hold	Comment	M-to-N	Pass	None
13	Blog	Hold	Like	M-to-N	Pass	None
14	Category	Is Stored	Look Up	M-to-N	Pass	None
15	Category	Contain	Post	M-to-N	Pass	None
16	Category	Contain	Blog	M-to-N	Pass	None
17	Comment	Created By	Users	M-to-N	Pass	None
18	Comment	Held By	Post	M-to-N	Pass	None
19	Comment	Held By	Blog	M-to-N	Pass	None
20	Friends	Has	Users	M-to-1	Pass	None
21	Friends	Provided By	Account	M-to-1	Pass	None
22	Like	Created By	Users	M-to-N	Pass	None
23	Like	Held By	Post	M-to-N	Pass	None
24	Like	Held By	Blog	M-to-N	Pass	None
25	Look Up	Store	Category	M-to-N	Pass	None
26	Look Up	Store	Tag	M-to-N	Pass	None
27	News	Provided By	Product	M-to-1	Pass	None
28	Permission	Provided By	Account	M-to-1	Pass	None

29	Permission	Has	Users	M-to-N	Fail	Zero or many permissions have one user
30	Post	Created By	Users	M-to-N	Pass	None
31	Post	Blog Post	Blog	M-to-N	Pass	None
32	Post	Contain	Category	M-to-N	Fail	I'm a little unsure how to fix this one. Perhaps I make the category zero or many, similar to blog
33	Post	Contain	Tag	M-to-N	Fail	I'm a little unsure how to fix this one. Perhaps I make the tags zero or many, similar to blog
34	Post	Hold	Comment	M-to-N	Pass	None
35	Post	Hold	Like	M-to-N	Pass	None
36	Product	Ran On	Technology	1-to-1	Pass	None
37	Product	Managed	Administrator	1-to-M	Pass	None
38	Product	Has	Administrator	1-to-M	Pass	None
39	Product	Provides	News	1-to-M	Pass	None
40	Options	Has	Account	M-to-1	Pass	None
41	Tag	Is Stored	Look Up	M-to-N	Pass	None
42	Tag	Contain	Post	M-to-N	Pass	None

43	Tag	Contain	Blog	M-to-N	Pass	None
44	Technology	Ran On	Product	1-to-1	Pass	None
45	Technology	ISA	Phone, Computer, or Tablet	1-to-M	Fail	Technology ISA phone, computer, or tablet
46	Users	Create	Account	1-to-1	Pass	None
47	Users	Has	Permission	1-to-M	Pass	None
48	Users	Has	Friends	1-to-M	Pass	None
49	Users	Create	Blog	M-to-N	Pass	None
50	Users	Create	Post	M-to-N	Pass	None
51	Users	Create	Comment	M-to-N	Pass	None
52	Users	Create	Like	M-to-N	Pass	None
53	Users	Has	Product	M-to-1	Pass	None
54	Blog Free User	ISA	Account	1-to-M	Fail	A BFU is an account
55	Blog Paid User	ISA	Account	1-to-M	Fail	A BPU is an account
56	General Free User	ISA	Account	1-to-M	Fail	A GFU is an account
57	General Paid User	ISA	Account	1-to-M	Fail	A GPU is an account
58	Computer	ISA	Account	1-to-M	Fail	A computer is a technology
59	Phone	ISA	Account	1-to-M	Fail	A phone is a technology
60	Tablet	ISA	Account	1-to-M	Fail	A tablet is a technology

Section VII: Database Model/EER

TABLE	FK	ON DELETE	ON UPDATE	COMMENT
Provides	product_id	ON CASCADE	ON CASCADE	Changes should be updated
Provides	news_id	ON CASCADE	ON CASCADE	Changes should be updated
Connects	product_id	ON CASCADE	ON CASCADE	Changes should be updated
Connects	url_id	ON CASCADE	ON CASCADE	Changes should be updated
Streams	product_id	ON CASCADE	ON CASCADE	Changes should be updated
Streams	host_id	ON CASCADE	ON CASCADE	Changes should be updated
Communicates	server_id	ON CASCADE	ON CASCADE	Changes should be updated
Communicates	host_id	ON CASCADE	ON CASCADE	Changes should be updated
Is Ran On	product_id	ON CASCADE	ON CASCADE	Changes should be updated
Is Ran On	technology_id	ON CASCADE	ON CASCADE	Changes should be updated
Hosts	advertisements_id	ON CASCADE	ON CASCADE	Changes should be updated
Hosts	administrator_id	ON CASCADE	ON CASCADE	Changes should be updated
Works With	administrator_id	ON CASCADE	ON CASCADE	Changes should be updated
Works With	administrator_id	ON CASCADE	ON CASCADE	Changes should be updated
Interacts	product_id	ON CASCADE	ON CASCADE	Changes should be updated
Interacts	users_id	ON CASCADE	ON CASCADE	Changes should be updated

Allows	permission_id	ON CASCADE	ON CASCADE	Changes should be updated
Allows	users_id	ON CASCADE	ON CASCADE	Changes should be updated
Shares	friends_id	ON CASCADE	ON CASCADE	Changes should be updated
Shares	users_id	ON CASCADE	ON CASCADE	Changes should be updated
Computer	technology_id	ON CASCADE	ON CASCADE	Changes should be updated
Tablet	technology_id	ON CASCADE	ON CASCADE	Changes should be updated
Phone	technology_id	ON CASCADE	ON CASCADE	Changes should be updated
General Free User	account_id	ON CASCADE	ON CASCADE	Changes should be updated
General Paid User	account_id	ON CASCADE	ON CASCADE	Changes should be updated
Blog Free User	account_id	ON CASCADE	ON CASCADE	Changes should be updated
Blog Paid User	account_id	ON CASCADE	ON CASCADE	Changes should be updated
Account	users_id	ON CASCADE	ON CASCADE	Changes should be updated
Options	account_id	ON CASCADE	ON CASCADE	Changes should be updated
Contains	users_id	ON CASCADE	ON CASCADE	Changes should be updated
Contains	demographic_id	ON CASCADE	ON CASCADE	Changes should be updated
Like	users_id	ON CASCADE	ON CASCADE	Changes should be updated
Comment	users_id	ON CASCADE	ON CASCADE	Changes should be updated
Create Blog	users_id	ON	ON CASCADE	Changes should be

		CASCADE		updated
Create Blog	blog_id	ON CASCADE	ON CASCADE	Changes should be updated
Blog Post	blog_id	ON CASCADE	ON CASCADE	Changes should be updated
Blog Post	post_id	ON CASCADE	ON CASCADE	Changes should be updated
Create Post	users_id	ON CASCADE	ON CASCADE	Changes should be updated
Create Post	post_id	ON CASCADE	ON CASCADE	Changes should be updated
Stores	look_up_id	ON CASCADE	ON CASCADE	Changes should be updated
Stores	category_id	ON CASCADE	ON CASCADE	Changes should be updated
Blog Category	blog_id	ON CASCADE	ON CASCADE	Changes should be updated
Blog Category	category_id	ON CASCADE	ON CASCADE	Changes should be updated
Files	ticket_id	ON CASCADE	ON CASCADE	Changes should be updated
Files	user_ticket_file_id	ON CASCADE	ON CASCADE	Changes should be updated
Ticket	users_id	ON CASCADE	ON CASCADE	Changes should be updated
Ticket	administrator_id	ON CASCADE	ON CASCADE	Changes should be updated

Section VIII: Forward Engineering

This portion was implemented in the `dataschema.sql` provided on Github.

Section IX: Inserting Data

This portion was implemented in the `inserts.sql` provided on Github.

Section X: Testing

This portion was implemented in the `test.sql` provided on Github.

Section XI: Testing Table

Entity	SQLQuery	Pass/Fail	Error Description	Possible Solution
Comment	Delete	Pass	None	None
Comment	Update	Pass	None	None
Like	Delete	Pass	None	None
Like	Update	Pass	None	None
Contains	Delete	Pass	None	None
Contains	Update	Pass	None	None
Demographic	Delete	Pass	None	None
Demographic	Update	Pass	None	None
Shares	Delete	Pass	None	None
Shares	Update	Pass	None	None
Friends	Delete	Pass	None	None
Friends	Update	Pass	None	None
Allows	Delete	Pass	None	None
Allows	Update	Pass	None	None
Permission	Delete	Pass	None	None
Permission	Update	Pass	None	None
Communicates	Delete	Pass	None	None
Communicates	Update	Pass	None	None
Server	Delete	Pass	None	None
Server	Update	Pass	None	None
Streams	Delete	Pass	None	None
Streams	Update	Pass	None	None
Host	Delete	Pass	None	None
Host	Update	Pass	None	None

Connects	Delete	Pass	None	None
Connects	Update	Pass	None	None
URL	Delete	Pass	None	None
URL	Update	Pass	None	None
General Free User	Delete	Pass	None	None
General Free User	Update	Pass	None	None
Blog Free User	Delete	Pass	None	None
Blog Free User	Update	Pass	None	None
Blog Paid User	Delete	Pass	None	None
Blog Paid User	Update	Pass	None	None
General Paid User	Delete	Pass	None	None
General Paid User	Update	Pass	None	None
Options	Delete	Pass	None	None
Options	Update	Pass	None	None
Account	Delete	Pass	None	None
Account	Update	Pass	None	None
Hosts	Delete	Pass	None	None
Hosts	Update	Pass	None	None
Works With	Delete	Pass	None	None
Works With	Update	Pass	None	None
Files	Delete	Pass	None	None
Files	Update	Pass	None	None
Ticket	Delete	Pass	None	None
Ticket	Update	Pass	None	None
User Ticket File	Delete	Pass	None	None
User Ticket File	Update	Pass	None	None
Advertisements	Delete	Pass	None	None

Advertisements	Update	Pass	None	None
Administrator	Delete	Pass	None	None
Administrator	Update	Pass	None	None
Phone	Delete	Pass	None	None
Phone	Update	Pass	None	None
Tablet	Delete	Pass	None	None
Tablet	Update	Pass	None	None
Computer	Delete	Pass	None	None
Computer	Update	Pass	None	None
Technology	Delete	Pass	None	None
Technology	Update	Pass	None	None
News	Delete	Pass	None	None
News	Update	Pass	None	None
Provides	Delete	Pass	None	None
Provides	Update	Pass	None	None
Create Post	Delete	Pass	None	None
Create Post	Update	Pass	None	None
Blog Post	Delete	Pass	None	None
Blog Post	Update	Pass	None	None
Blog Category	Delete	Pass	None	None
Blog Category	Update	Pass	None	None
Create Blog	Delete	Pass	None	None
Create Blog	Update	Pass	None	None
Stores	Delete	Pass	None	None
Stores	Update	Pass	None	None
Blog	Delete	Pass	None	None
Blog	Update	Pass	None	None

Post	Delete	Pass	None	None
Post	Update	Pass	None	None
Look Up	Delete	Pass	None	None
Look Up	Update	Pass	None	None
Category	Delete	Pass	None	None
Category	Update	Pass	None	None
Interacts	Delete	Pass	None	None
Interacts	Update	Pass	None	None
Users	Delete	Pass	None	None
Users	Update	Pass	None	None
Manages	Delete	Pass	None	None
Manages	Update	Pass	None	None
Is Ran On	Delete	Pass	None	None
Is Ran On	Update	Pass	None	None
Product	Delete	Pass	None	None
Product	Update	Pass	None	None