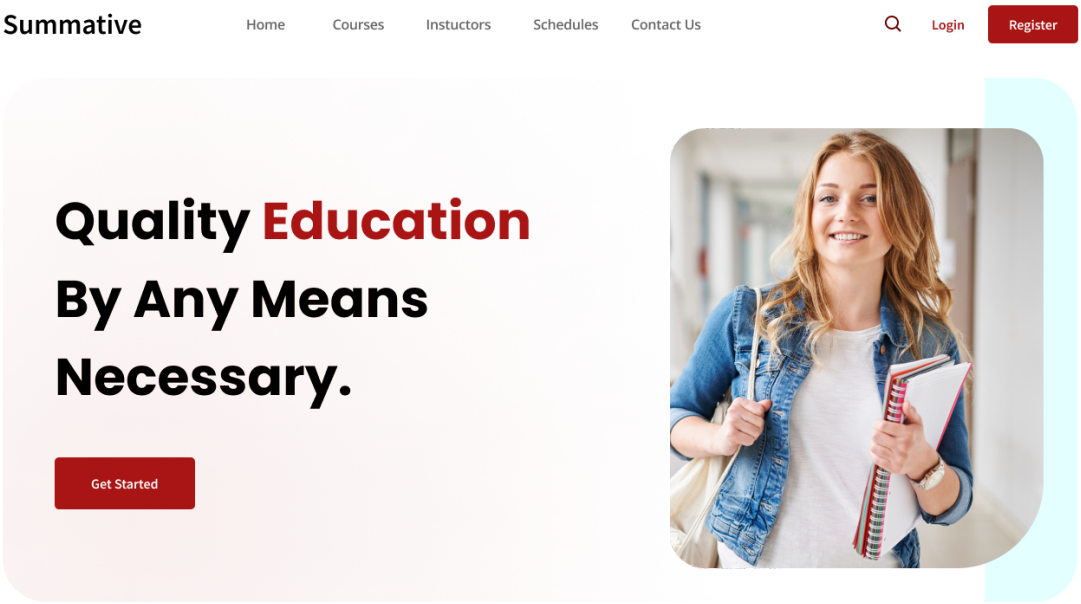
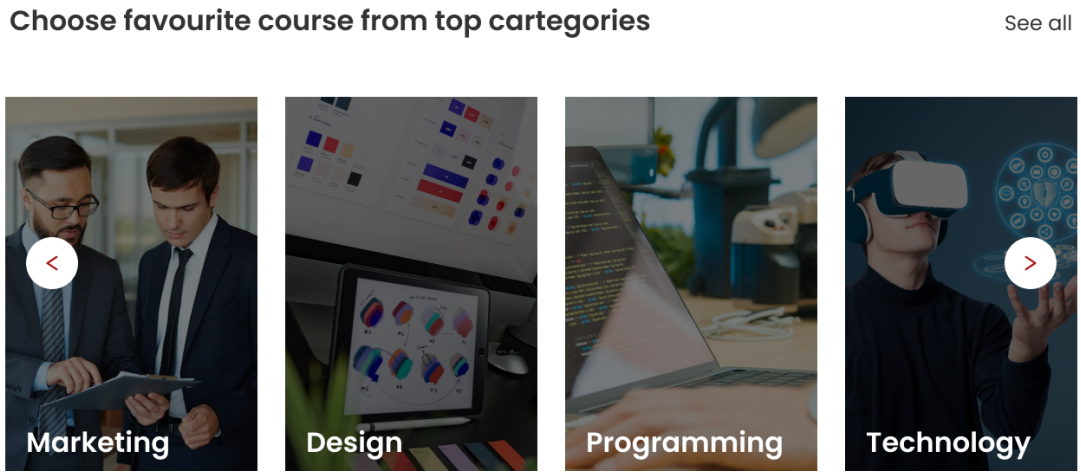
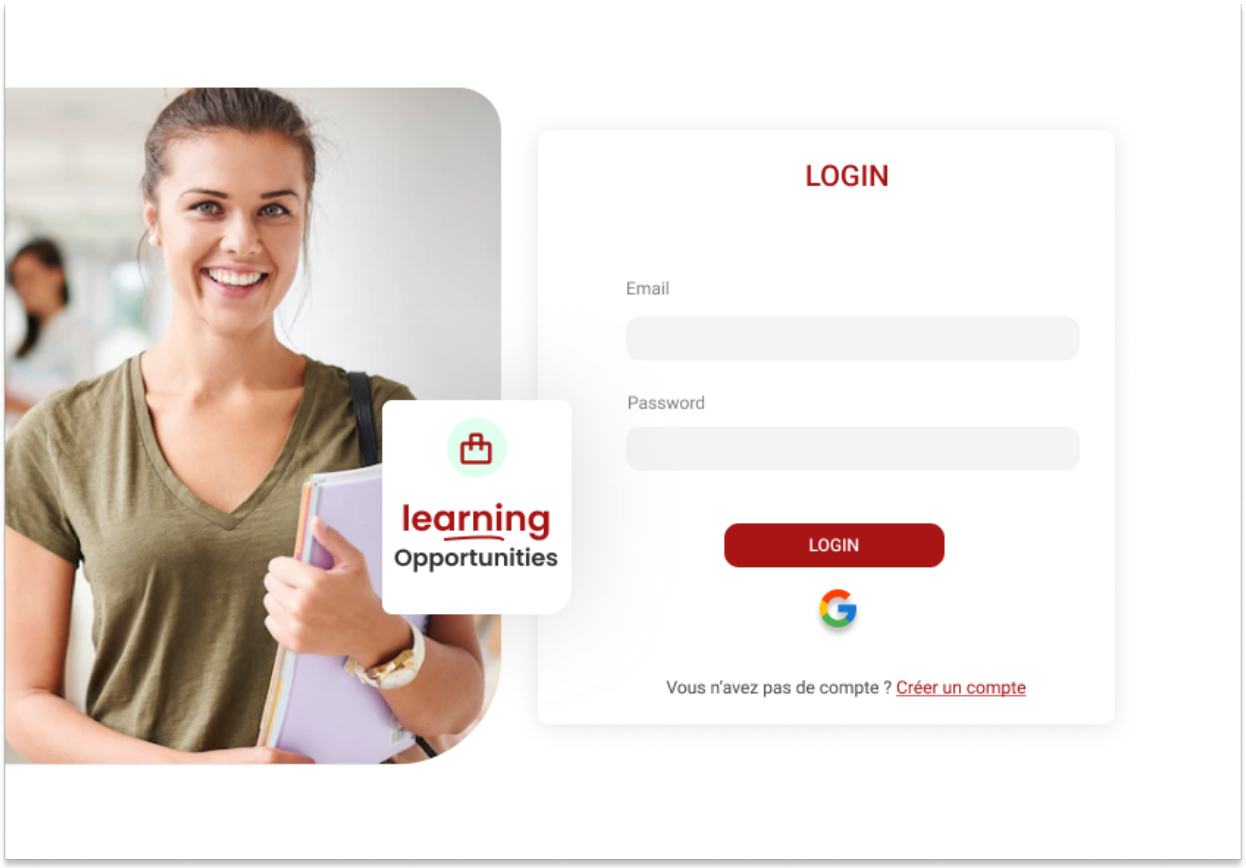
Question1

Landing Page

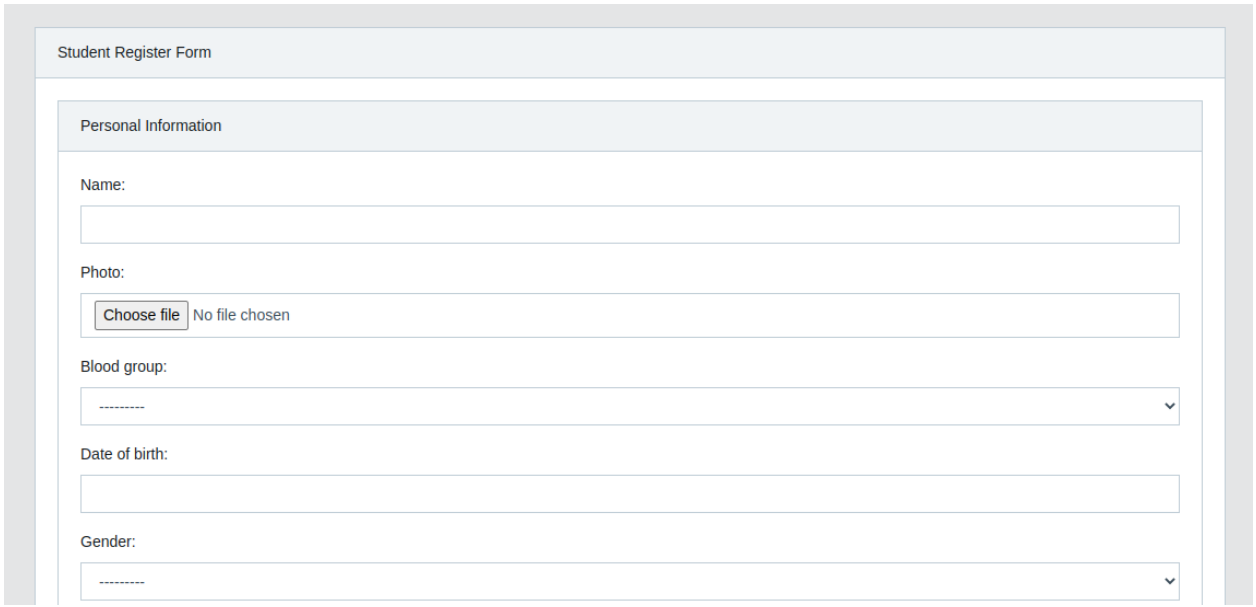




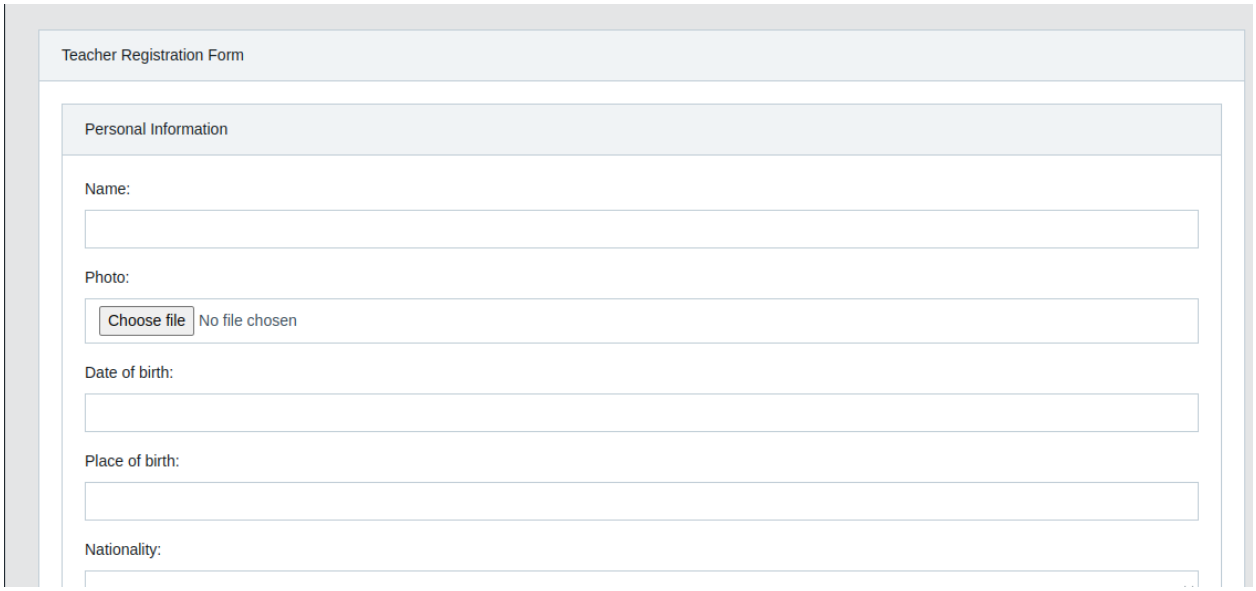
Login page



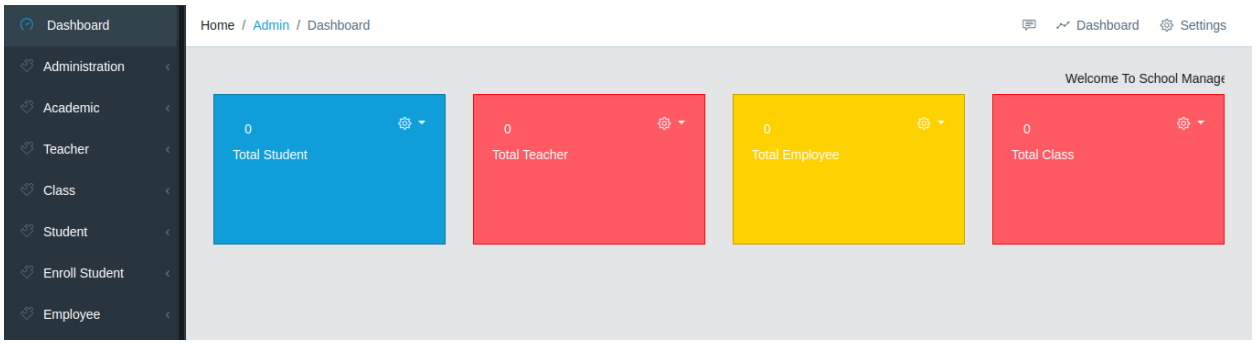
Student view



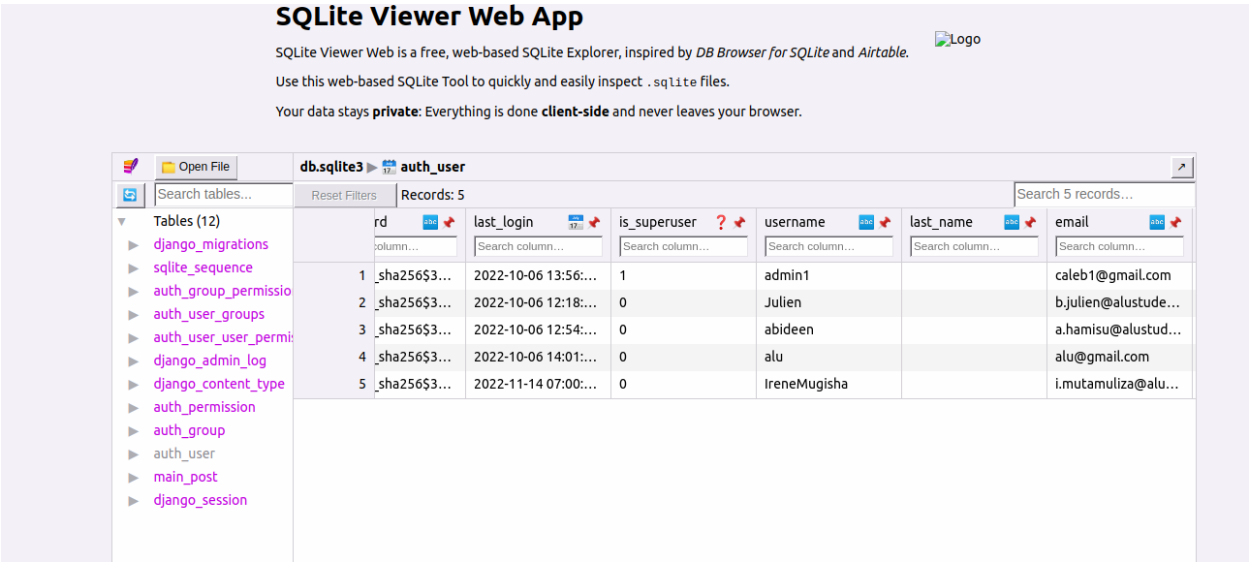
Teacher view



Team Lead view



Database view



What strategy will be used to create strong passwords?
A minimum of 8 characters, having at least one uppercase character, a symbol and a number. By default, Django uses the PBKDF2 algorithm with a SHA256 hash, a password stretching mechanism recommended by NIST. This should be sufficient for most users: it's quite secure, requiring massive amounts of computing time to break.
How will you ensure every user types only strong password while creating identity
To help mitigate this problem, Django offers pluggable password validation. You can configure multiple password validators at the same time.
Which protocols will you use to ensure that integrity is observed while data is exchanged between the browser and the server (which is remote).
Transport Layer Security (TLS)

Consider your database to be SQL based, what will you do to ensure that SQL-injection attacks will fail while attempted on the developed app.
Fortunately Django's querysets are protected from SQL injection since their queries are constructed using query parameterization, meaning that SQL injection are remotely impossible while using the django framework.

Question2

1. Create 4 database users

```
caleb@mugisha:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 26
Server version: 8.0.31-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE USER 'calebm'@'localhost' IDENTIFIED BY 'munana@12';
Query OK, 0 rows affected (0,35 sec)

mysql> CREATE USER 'diego'@'localhost' IDENTIFIED BY 'masimbi@12';
Query OK, 0 rows affected (0,16 sec)

mysql> CREATE USER 'shenge'@'localhost' IDENTIFIED BY 'rubangutsangabo@12';
Query OK, 0 rows affected (0,14 sec)

mysql> CREATE USER 'niyonkuru'@'localhost' IDENTIFIED BY 'paul@12';
Query OK, 0 rows affected (0,15 sec)

mysql>
```

2. Create 2 different roles (Admin and Developer)

```
mysql> CREATE ROLE "admin";
Query OK, 0 rows affected (0,24 sec)

mysql> CREATE ROLE "developer";
Query OK, 0 rows affected (0,16 sec)

mysql>
```

3. Add the first two users to Admin role and the other two remaining users to the Developer role

```
mysql> GRANT "admin" To "calebm"@"localhost", "diego"@"localhost";
Query OK, 0 rows affected (0,15 sec)

mysql> GRANT "developer" TO "shenge"@"localhost", "niyonkuru"@"localhost
";
Query OK, 0 rows affected (0,16 sec)
```

4. Grant any 5 privileges to Admin and grant 3 privileges (subset of Admin privileges) to the developer

```
mysql> GRANT CREATE, UPDATE, DELETE, SELECT, DROP ON * . * TO 'admin';
Query OK, 0 rows affected (0,14 sec)

mysql> GRANT CREATE, UPDATE, SELECT ON * . * TO 'developer';
Query OK, 0 rows affected (0,11 sec)
```

5. Create a view that combine data from 3 tables.
    a. Creating tables

```
mysql> use summatives;
Database changed
mysql> CREATE TABLE programmers ( programmerID int, LastName varchar(255
), Stack varchar(255));
Query OK, 0 rows affected (1,87 sec)

mysql> CREATE TABLE languages (Lname varchar(255), age int)
    -> ;
Query OK, 0 rows affected (1,26 sec)

mysql> CREATE TABLE machine (Mname varchar(255), RAM int);
Query OK, 0 rows affected (1,76 sec)

mysql>
```

B.creating view

```
mysql> CREATE VIEW combinee AS SELECT programmers.LastName, languages.ag
e, machine.RAM FROM programmers, languages, machine WHERE programmers.pr
ogrammerID = languages.age;
Query OK, 0 rows affected (0,21 sec)

mysql>
```

6. Create another role called viewer

```
mysql> CREATE ROLE 'viewer';
Query OK, 0 rows affected (0,19 sec)

mysql>
```

7. Create another user and add this user to the viewer role.

```
mysql> CREATE USER 'Nyemina'@'localhost' IDENTIFIED BY 'kigali@123';
Query OK, 0 rows affected (0,14 sec)

mysql> GRANT 'viewer' TO 'Nyemina'@'localhost';
Query OK, 0 rows affected (0,12 sec)

mysql>
```

8. The user under the viewer can only read data in a view (meaning that s/he has not access other tables and database itself)

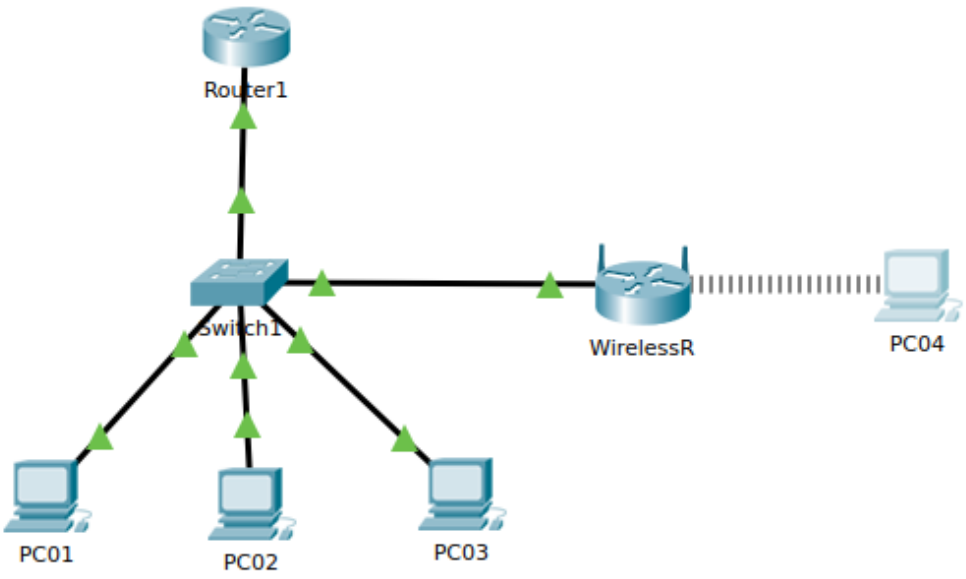
```
mysql> GRANT SHOW VIEW ON *.* TO 'viewer';
Query OK, 0 rows affected (0,12 sec)

mysql>
```

9. Revoke 1 privilege from admin and 1 privilege from developer.

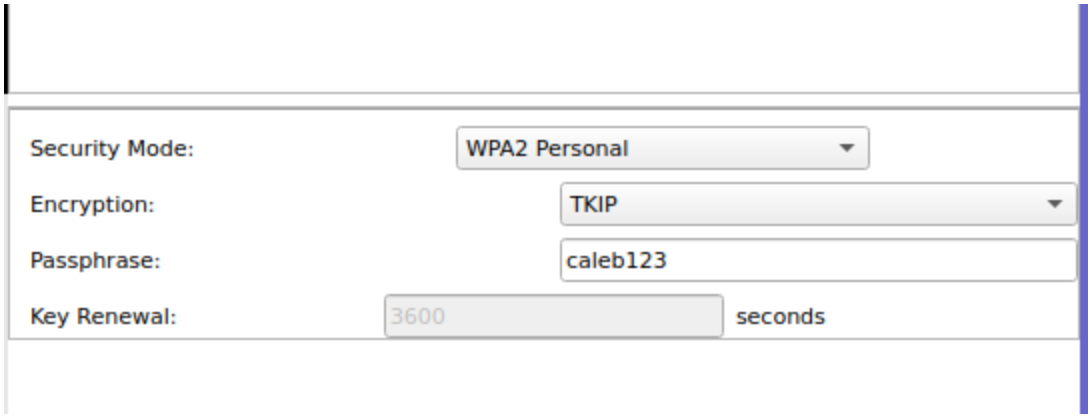
```
mysql> REVOKE UPDATE ON *.* FROM 'admin';
Query OK, 0 rows affected (0,14 sec)

mysql> REVOKE SELECT ON *.* FROM 'developer';
Query OK, 0 rows affected (0,11 sec)

mysql>
```

Question3

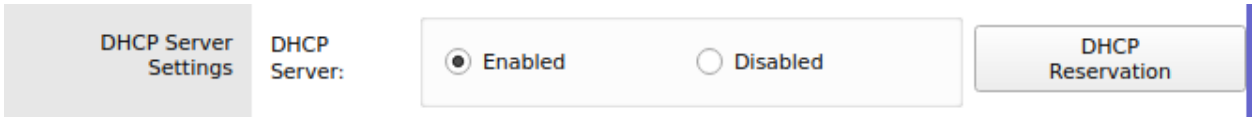
Network Topology



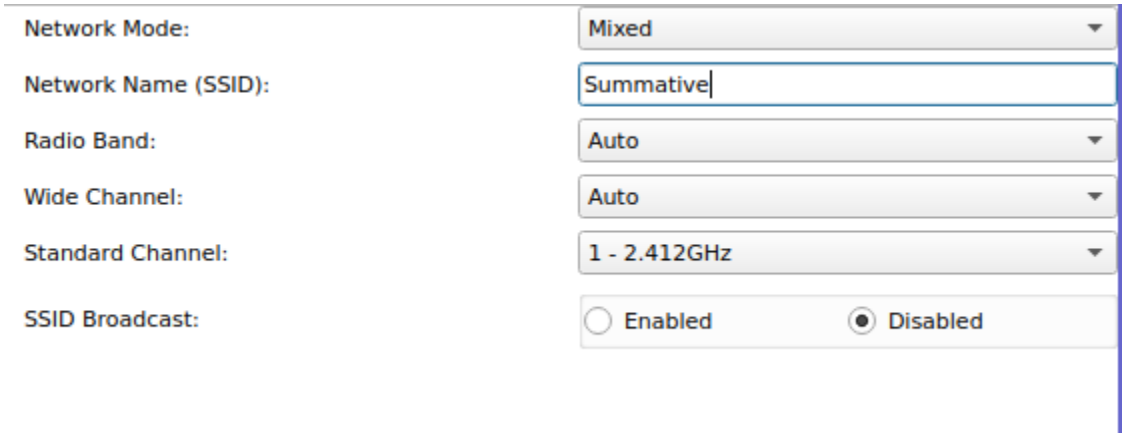
The router(WirelessR) uses WP2 Personal



Enabling DCHP



Disabling Broadcast



Why is WPA2 recommended over WEP?

WEP is less secure than WPA2 in the sense that it uses a much stronger encryption algorithm than WEP, making it harder to decode. WPA uses a passphrase to perform the authentication and generate the initial data encryption keys, then dynamically varies the encryption key, with it's improvement to WPA2, more security is assured.

What does WPS button serve on a wireless router how do hackers take advantage of it?

WPS(Wi-Fi Protected Setup) allows you to connect devices to your internet without requiring a password. This is not secure, because hackers can decode this supposed security PIN within a short period of time.