1. HTTP applications is a stateless protocol, this means that each response cycle is independent and the server does not remember previous interactions. To overcome this, it uses session management or authentication tracking. Session management, the server creates unique session IDs stored in cookies, that contains session data like user info and preference, this is saved on the server in cache systems. The authentication tracking is either session based or token based. In both methods, the server generates a session ID on the first visit, this is stored on the server side and sent to the browser as a cookie. In subsequent requests, the browser automatically includes a session cookie. The server processing reads the session ID and retrieves corresponding user data and maintains a continuous state.
2. Django migrations provide a version control system for database schema changes, this allows a smooth evolution database structure and preserves data integrity. To ensure this process, we start off by configuring the production database settings in Django, we specify the Maria DB database engine. We also provide connection parameters such as host, port, database username, username and password. The database must be pre-configured before applying migrations. The migration involves generating migration files locally using “python manage.py makemigrations”. For production safety, migrations should be performed during maintenance periods with database backups created beforehand. This should also be tested beforehand to ensure that it’s compatible with MySQL and ensure seamless operations.