

Research Findings on HTTP Applications and State Preservation

HTTP is a stateless protocol, which means each request from the client to the server is treated independently. However, web applications often need to maintain state across multiple request-response cycles, particularly for user authentication and session management. This is typically achieved through the use of cookies, sessions, and tokens.

Cookies: Small pieces of data stored on the client side, allowing servers to remember user preferences and login states.

Sessions: Server-side storage of user data that persists for the duration of a user's visit. Each session is identified by a unique session ID, often stored in a cookie.

Tokens: Used in modern applications (especially with REST APIs), where the server issues a token upon login. This token is sent with each subsequent request, allowing the server to authenticate the user without needing to maintain a session.

These methods enable applications to create a seamless user experience while managing state effectively.

Database Migrations in Django with MariaDB

Django supports various relational databases, including MariaDB, which is a fork of MySQL. Database migrations in Django allow developers to manage changes to the database schema over time.

The process typically involves two main steps:

Creating Migrations: When changes are made to the models (such as adding fields or new models), developers run the command:

`Python manage.py makemigrations`

This command generates migration files that describe the changes.

Applying Migrations: To apply these changes to the database, the following command is used:

`Python manage.py migrate`

This command executes the generated migration files, updating the database schema accordingly.

Django's migration framework provides a robust way to track and apply changes, ensuring that the database structure remains in sync with the application code.