Intro to Flask

MVC Architecture

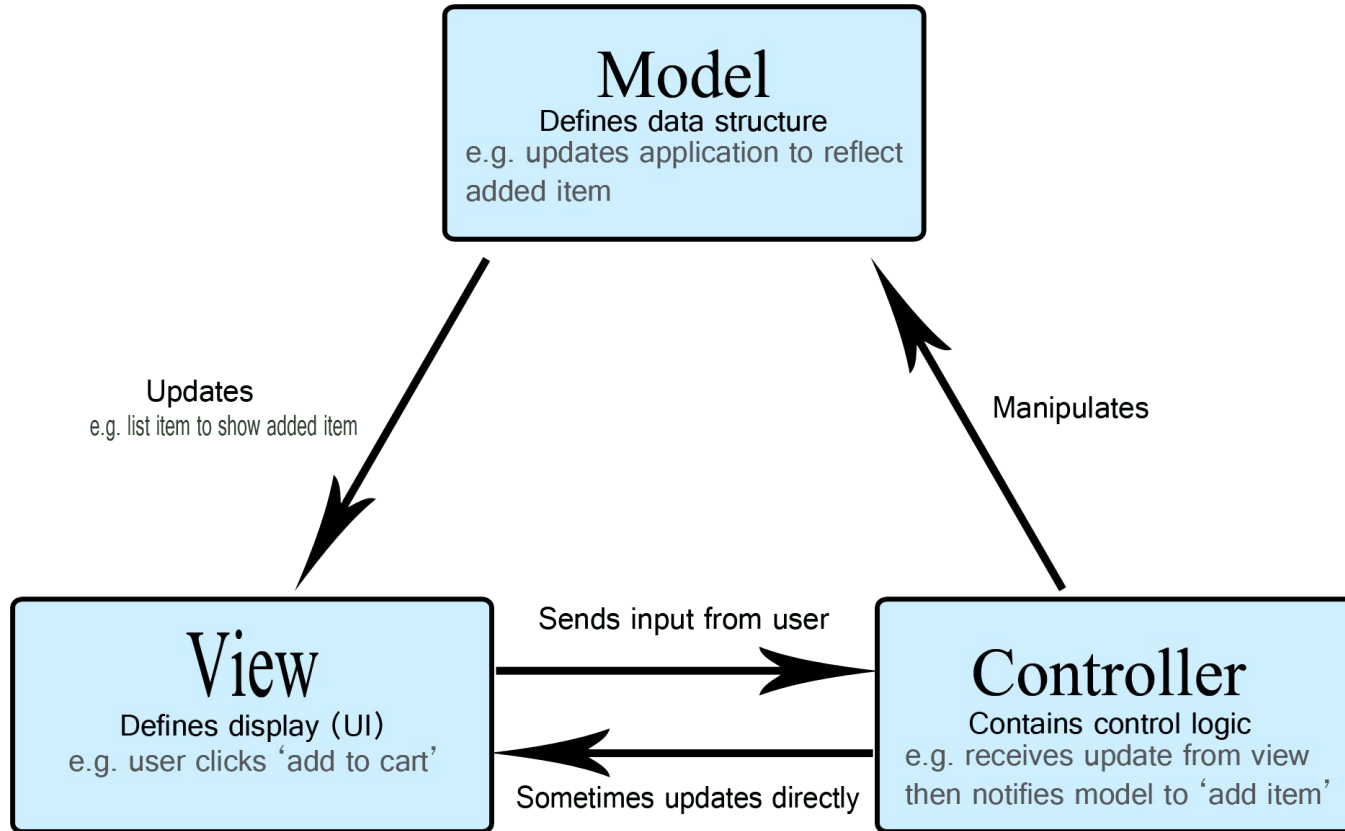Design pattern, that organises an application's logic into 3 components.

Traditionally used for desktop GUIs (Graphical User Interfaces).

Model (Business Logic): Data-related logic, talks to the database

View (UI Logic): Talks to the user, gets user inputs

Controller (Input Logic): Processes user inputs, core logic, connects view and model.

Examples: Flask, AngularJS, Django, ASP.NET, Ruby on Rails

Model
Defines data structure
e.g. updates application to reflect added item

View
Defines display (UI)
e.g. user clicks 'add to cart'

Controller
Contains control logic
e.g. receives update from view then notifies model to 'add item'

Updates
e.g. list item to show added item

Manipulates

Sends input from user

Sometimes updates directly

Flask

- Web Framework of Python (Framework: Platform for developing software applications)
- **Lightweight**: Minimal dependencies, Core functionality, quick setup
- **Flexible**: Customizable, Extensibility, Scalable
- **Easy to Learn**

Structure:

- app.py: Main python file where Flask app is created
- Templates: Has HTML templates for dynamic content
- Static: Static files like CSS, JavaScript, images, etc.

Demo