



# Web Applications in OutSystems

---

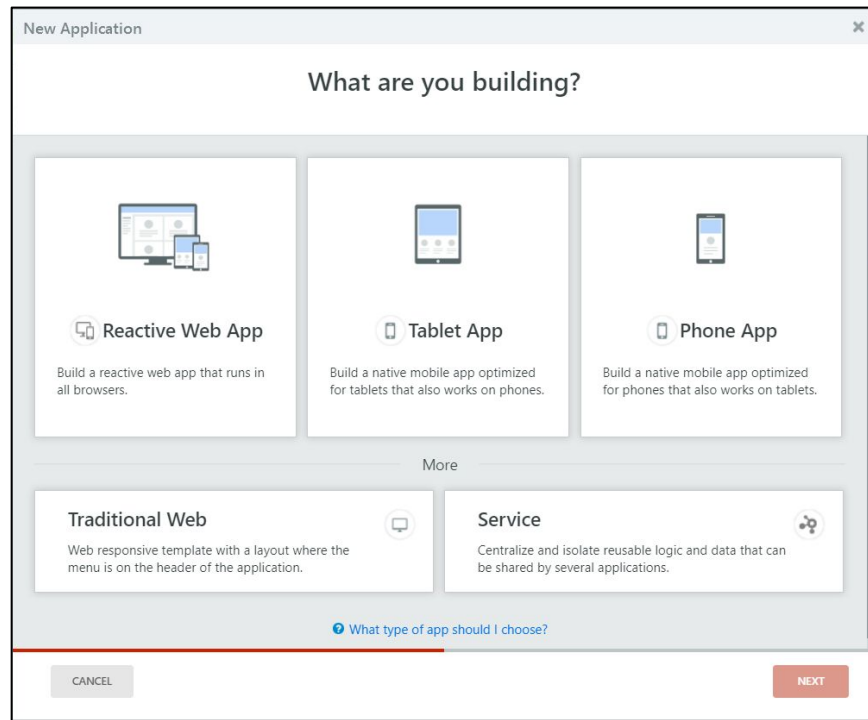


# Topics

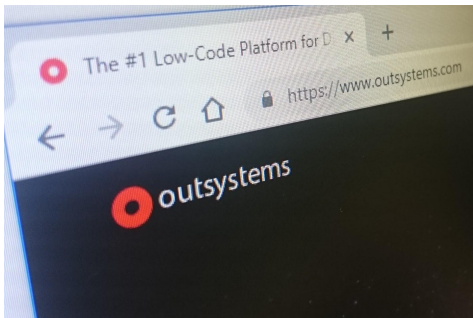
- Web Applications in OutSystems
  - Creating a Traditional Web application
  - Application modules
- Modular Programming with OutSystems
  - Consumer and producer modules

# Application types

- Applications can be...
  - Web apps (Reactive)
  - Traditional Web Apps
  - Tablet Apps and Phone Apps
  - Service apps
- Web apps, Traditional apps, and Mobile apps have different programming models
- Applications are the deployment unit in OutSystems
  - Apps can be versioned & tagged for easier management



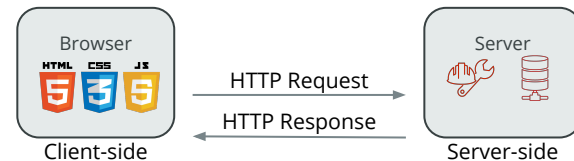
# Web Applications in OutSystems



**Run in a browser**

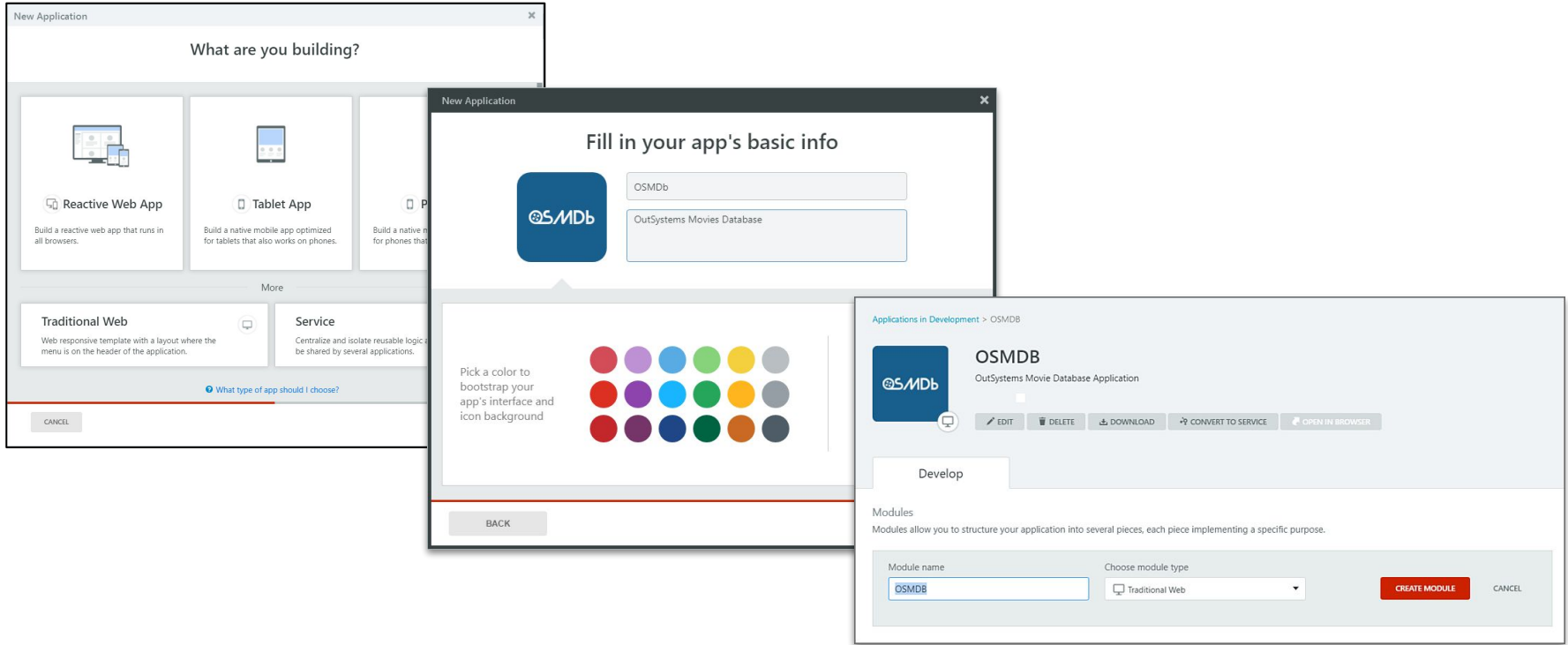


**Are Responsive:**  
adjust the layout depending  
on device type & screen size



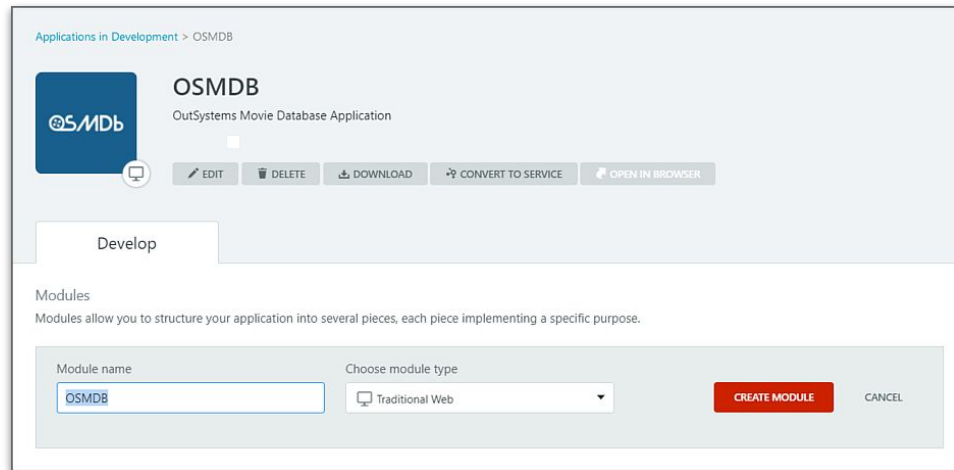
**Request - Response  
pattern**

# Creating a Traditional Web Application



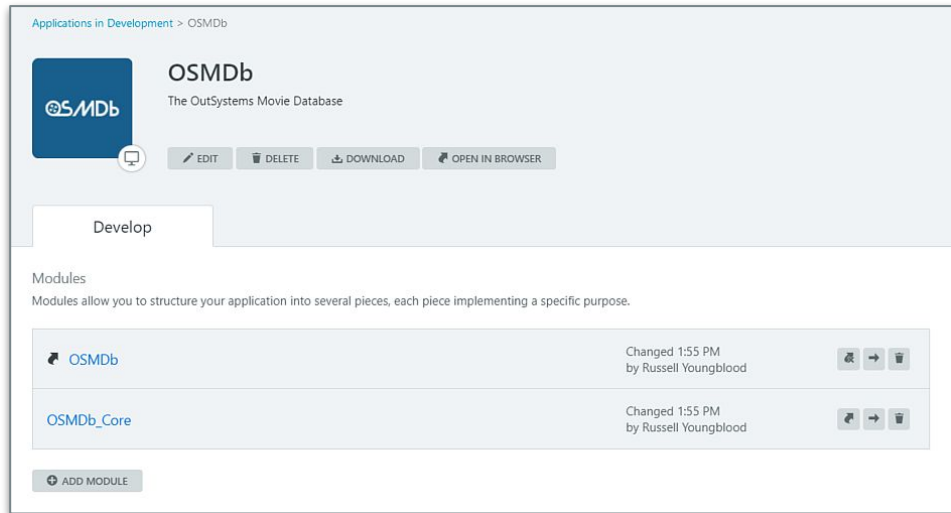
# Web Applications: Module types

- Applications should have at least one module
- Web application modules can be:
  - Traditional Web
  - Blank
  - Service
  - Extension



# Web Applications: Modules

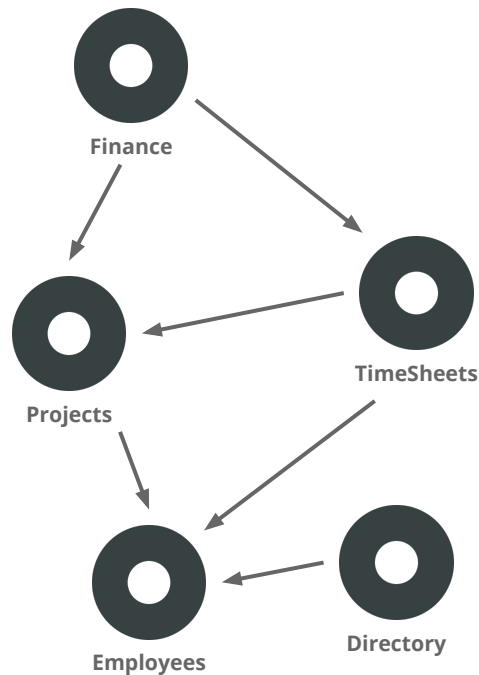
- Applications can have one or more modules
- Modules are where developers:
  - Create the data model
  - Define business logic
  - Build web pages



# Modular Programming

Software design technique that allows each module to...

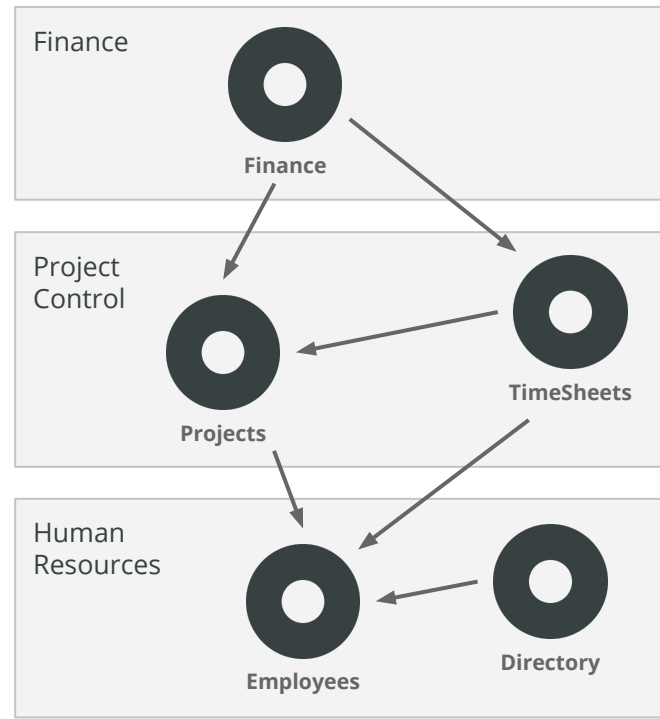
- **Encapsulate** everything necessary to execute **one aspect** of functionality
- Separate functionality by **independent** and (potentially) **replaceable** pieces of code





# Modular Programming in OutSystems

- An **Application** groups a set of related **modules**
- Modules can share elements with other modules
  - Modules that share features are called **Producers**
  - Modules that use features from others are called **Consumers**
- Producers and Consumers can be in different applications



# Producers

Make features **Public** to share them

The following elements can be made public:

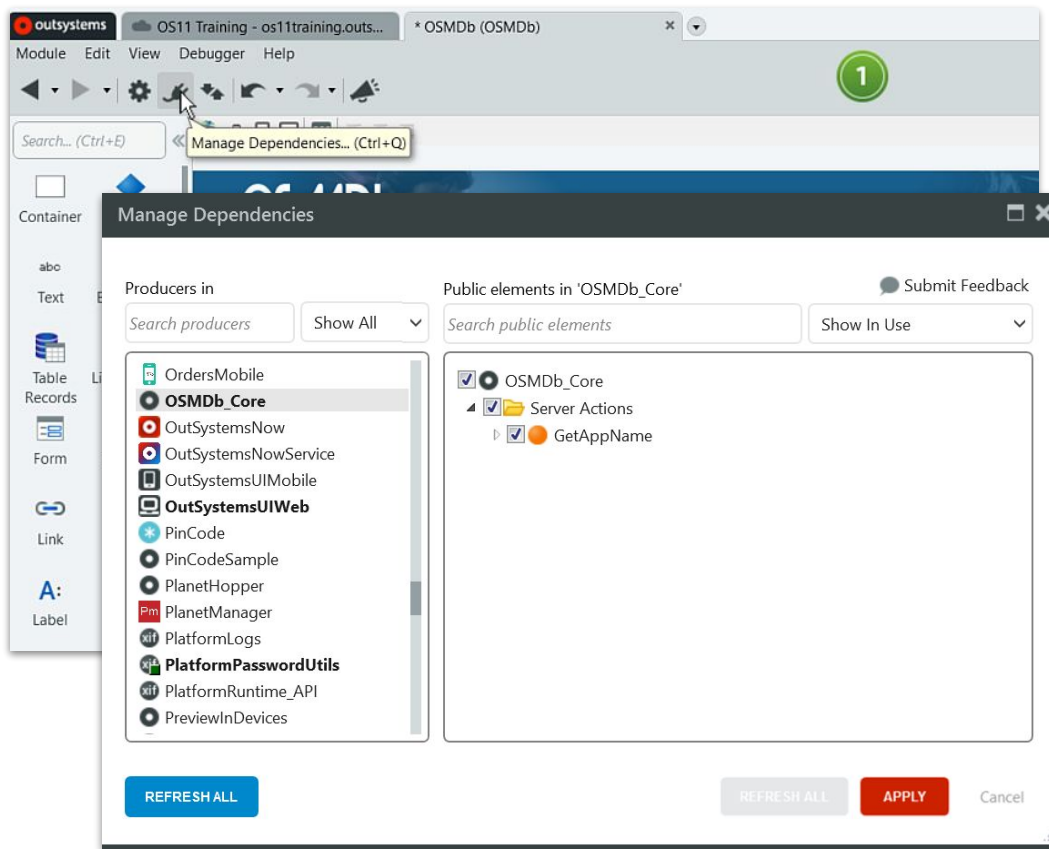
- Data
  - Entities
  - Structures
- Logic
  - Server Actions
  - Roles
- Interface
  - Web Blocks, Web Screens
  - Images
  - Themes
- Processes
  - (Business) Processes

The screenshot shows the OutSystems Studio interface. The top navigation bar includes 'Processes', 'Interface', 'Logic', and 'Data'. The 'Data' tab is active, showing a tree view of the 'OSMDB\_Core' project. The tree view includes 'Entity Diagrams' (containing 'MoviesDB Diagram') and 'Entities' (containing 'Database', 'Movie', 'MovieGenre', 'Person', 'PersonMovieRole', 'PersonRole', 'UserMovieComment', and 'UserMovieRating'). The 'Movie' entity is selected. Below the tree view, the 'Movie Entity' configuration is displayed. The 'Public' checkbox is checked, and the 'Expose Read...' dropdown is set to 'No'. The 'Indexes' and 'More...' options are also visible.

Name	Movie
Description	...
Public	Yes
Expose Read...	No
Indexes	...
More...	...

# Consumers

- Reuse public elements from producers
- Manage Dependencies
  - Modules with public features are displayed
  - Select the desired elements to be able to use them (consume)
- Selected elements become available in the module



# Summary

- Web Applications in OutSystems
  - Creating a Traditional Web application
  - Application modules
- Modular Programming with OutSystems
  - Consumer and producer modules



# **Web Applications in OutSystems**

## **Thank You!**