

Web Application Dictionary

Part I — User Interface (GUI)

Chapter 1. Basic Layout

- Header
- Footer
- Sidebar
- Toolbar
- Navigation Bar
- Content Area
- Panel
- Pane
- Card
- Container
- View
- Viewport
- Region

Each term would include:

Panel

Definition

A rectangular area that groups related controls.

Example

A panel containing all search filters.

When to use

When multiple controls belong together.

Related terms

Container

Card

Sidebar

Dialog

HTML example

```
<div class="panel">
```

Part II — User Interaction

- Button
- Icon Button
- Floating Button
- Dropdown
- Accordion
- Tabs
- Tree View
- Data Grid
- Modal
- Drawer
- Tooltip
- Toast
- Notification
- Wizard

Part III — Icons

- Chevron

- Caret
 - Hamburger
 - Kebab
 - Meatballs
 - Plus
 - Minus
 - Gear
 - Avatar
-

Part IV — Frontend Architecture

```
Browser
|
├─ HTML
├─ CSS
├─ JavaScript
├─ Framework
├─ Components
├─ Router
├─ State
└─ API Client
```

Terms:

- Component
- State
- Props
- Store
- Router
- Route
- View
- Hook

- Directive
 - Virtual DOM
 - Hydration
 - Rendering
-

Part V — Backend Architecture

HTTP Request

|

Controller

|

Service

|

Repository

|

Database

Terms:

- Controller
 - Action
 - Middleware
 - Service
 - Repository
 - Model
 - Entity
 - DTO
 - Factory
 - Dependency Injection
-

Part VI — Data

- Database
 - Table
 - Row
 - Record
 - Column
 - Field
 - Primary Key
 - Foreign Key
 - Index
 - View
 - Trigger
 - Stored Procedure
 - Transaction
 - Migration
 - Seed
-

Part VII — HTTP

```
Browser
  |
GET /students/5
  |
Server
  |
200 OK
```

Terms:

- Request
- Response
- Header

- Body
 - Status Code
 - Endpoint
 - Route
 - Resource
 - REST
 - GraphQL
-

Part VIII — Authentication

- Login
 - Authentication
 - Authorization
 - Session
 - Cookie
 - JWT
 - OAuth
 - Permission
 - Role
 - ACL
-

Part IX — Security

- HTTPS
- TLS
- Hashing
- Encryption
- Salt

- CSRF
 - XSS
 - SQL Injection
 - CORS
 - CSP
-

Part X — Performance

- Cache
 - CDN
 - Compression
 - Minification
 - Bundling
 - Lazy Loading
 - Code Splitting
 - Tree Shaking
-

Part XI — Deployment

```
Developer
  |
Git
  |
CI
  |
Build
  |
Docker
  |
Production Server
```

Terms:

- Build
 - Deploy
 - Pipeline
 - Docker
 - Container
 - Reverse Proxy
 - Nginx
 - Environment
 - Production
 - Staging
-

Part XII — Design Patterns

Not just definitions, but diagrams.

For example:

MVC

User

|

▼

Controller

|

▼

Model

|

▼

Database

|

▲

View

Similarly for:

- MVC
- MVVM
- Repository
- Observer
- Factory
- Singleton
- Dependency Injection
- Strategy
- Adapter
- Facade
- Decorator

Part XIII — Complete Application Walkthrough

A complete example application showing how all the pieces fit together.

Browser

Student List Page

Search Box

Data Grid

Pagination

↓

JavaScript

↓

GET /students

↓

Controller

↓

StudentService

↓

StudentRepository

↓

Database

↓

JSON Response

↓

Browser updates Data Grid

Then expand this with topics such as:

- Logging
- Caching
- Validation
- Authentication
- Authorization
- Error handling
- Background jobs
- Email sending
- File uploads

- Notifications
 - Audit logs
-

Consistent structure for every term

Every entry would follow the same template:

- **Definition** – a concise explanation.
- **Purpose** – why the concept exists.
- **When to use** – typical scenarios.
- **Common mistakes** – frequent misconceptions.
- **Related terms** – links to connected concepts.
- **Example** – a practical code or UI example.
- **Diagram** – where the concept fits in the overall architecture.

This approach turns the glossary into a practical handbook rather than a simple list of definitions.

Given your background as a software development teacher, I'd also tailor the examples to a familiar stack—HTML, CSS, JavaScript, PHP (Yii2), SQL, REST APIs, and Bootstrap—so students can immediately relate the terminology to code they are writing.

Revision #1

Created 2026-08-06 19:59:21 UTC by Max

Updated 2026-08-06 20:00:05 UTC by Max