



## IIT Madras BSc Degree

### Copyright and terms of use

**IIT Madras is the sole owner of the content available in this portal - [onlinedegree.iitm.ac.in](https://onlinedegree.iitm.ac.in) and the content is copyrighted to IIT Madras.**

- Learners may download copyrighted material for their use for the purpose of the online program only.
- Except as otherwise expressly permitted under copyright law, no use other than for the purpose of the online program is permitted.
- No copying, redistribution, retransmission, publication or exploitation, commercial or otherwise of material will be permitted without the express permission of IIT Madras.
- Learner acknowledges that he/she does not acquire any ownership rights by downloading copyrighted material.
- Learners may not modify, publish, transmit, participate in the transfer or sale, create derivative works, or in any way exploit, any of the content, in whole or in part.

Frontend implementation

# What is the frontend?

- User-facing part of app
  - User Interface (UI) and User Experience (UX)

# What is the frontend?

- User-facing part of app
  - User Interface (UI) and User Experience (UX)
- Requirements
  - Avoid complex logic - application logic should be in backend
  - No data storage
  - Work with stateless nature of HTTP

# What is the frontend?

- User-facing part of app
  - User Interface (UI) and User Experience (UX)
- Requirements
  - Avoid complex logic - application logic should be in backend
  - No data storage
  - Work with stateless nature of HTTP
- Desirable
  - Aesthetically pleasing
  - Responsive - no lag / latency
  - Adaptive - different screens

# Programming Styles

- Imperative: sequence of actions to achieve final result
  - Draw boxes for navigation, main text, fill in text, wait for clicks etc.
  - Functions for each step, composition of functions
- Declarative: specify desired result
  - Compiler / Interpreter knows how to achieve result
  - Function integration automated

**UI = f( state )**

The layout  
on the screen

Your  
build  
methods

The application state

Credit: Flutter documentation “Start thinking declaratively”

# State?

- Internal details of the system: memory
- Reproducibility
  - Given a “system state”, the system should always respond the same way to input
- Complexity
  - Any non-trivial application needs internal state



## System State

- Complete database of amazon.in, flipkart.com
  - Stocks of available items, prices, logged in/registered users etc.
- All news articles ever published on toi.com, thehindu.com, bbc.com
- All students, courses, marks, certificates etc. for NPTEL

Typically huge, but comprehensive

- Completely independent of the user interface / frontend!

# Application State

Application:

- System as seen by an individual user / session
- Includes interactivity, session management

Examples:

- Shopping cart, user preferences, theme
- Followed news items, recommendations
- Dashboard displays

# UI State (Ephemeral State)

## UI

- Part of application actually seen / interacted with
- Ephemeral - “lasting for a very short time” (term used by Flutter)

## Examples:

- Loading icons
- Currently selected tab in multi-tab document / page

# Application and UI management

- HTTP is stateless
- How to convey state between client and server?
  - Client maintains state - sends requests to server for specific items
  - Server maintains state - only specific requests allowed to client

## Example: Tic-Tac-Toe

- What to display on screen?
- Who determines the display?
- How should user input be collected and processed?