

Code Standards

Naming Conventions:

- Front-end
 - Classes
 - Describe what the class does
 - Use kebab case
 - ex) .text-align-center
 - IDs
 - Describe the element
 - Use camel case
 - ex) #heartReactButton
 - React components
 - Use Pascal case and be descriptive
 - ex) CommentSection
- Back-end
 - Private variables
 - Start with underscore and use camel case
 - ex) private bool _personalAccount
 - Public variables
 - Use Pascal case
 - ex) public class
 - Local variables
 - Use camel case
 - ex) string
 - Booleans
 - Name variables like a question
 - ex) isAdmin, hasSecurity
 - API endpoints
 - URLs should be kebab case
 - ex) /user-profile
 - Files for services, controllers, models, etc. should end with the word that describes the file
 - ex) PersonalAccountService, UserModel
- General
 - Avoid abbreviations – it may seem tedious, but spelling out variable names will reduce confusion on what elements we are referring to and not knowing acronyms

Code Formatting:

- Use tabs for indenting
- Delete console.log()/System.out.println() statements after testing
- Use a newline after each function or method

Code Standards

Exception Handling:

- The user should never see generic exception statements – build in descriptive and user-friendly error messages
- Use try/catch blocks in controllers and services

File Structure:

- Use separate folders to hold different kinds of files
 - ex) folders: Components, Models, Controllers, Stylesheets, etc.

Other:

- Functions and methods should only do one thing (single responsibility)
- Make components reusable when possible
 - ex) Post component – the same component can be used for any user posting any kind of post: check the user and type of post then hide/show elements like photo, hashtags, business account icon, etc.