



# Capstone Project 1

CMU-SE 450

## Code Standard

Version 1.0

Date: 13/03/2023

### LinguaSnap for Travelers

Submitted by

Dat, Nguyen Thanh  
Truong, Vu Dinh  
Long, Pham Ba Hoang  
Kha, Nguyen Ngoc

Approved by Nguyen Duc Man

**Proposal Review Panel Representative:**

Name

Signature

Date

**Capstone Project 1- Mentor:**

Name

Signature

Date

## PROJECT INFORMATION

<b>Project acronym</b>	LiS		
<b>Project Title</b>	LinguaSnap for Travelers		
<b>Start Date</b>	25 Feb 2023	<b>End Date</b>	31 May 2023
<b>Lead Institution</b>	International School, Duy Tan University		
<b>Project Mentor</b>	Man, Nguyen Duc, Ph.D		
<b>Scrum master / Project Leader &amp; contact details</b>	Nguyen Thanh Dat Email: ntdata1232001@gmail.com Tel: 0972.530.969 Student ID: 25211217197		
<b>Partner Organization</b>			
<b>Project Web URL</b>			
<b>Team members</b>	<b>Name</b>	<b>Email</b>	<b>Tel</b>
25211217197	Nguyen Thanh Dat	ntdata1232001@gmail.com	0972530969
25211201774	Long, Pham Ba Hoang	longphambahoang@gmail.com	0793310221
25211217275	Truong ,Vu Dinh	jonnyvu2210@gmail.com	0905223611
25211210026	Kha, Nguyen Ngoc	winkha14567@gmail.com	0945721427

## REVISION HISTORY

Version	Date	Comments	Author	Approval
1.0	13/03/2023	Initial Release	All members	

## 1. Introduction

### 1.1 Purpose

- This Coding Standard requires certain practices for developing programs in the Java language. The objective of this coding standard is to have a positive effect on
  - Avoidance of errors/bugs, especially the hard-to-find ones.
  - Maintainability, by promoting some proven design principles

### 1.2 Scope

- This standard pertains to the use of the Java language.

## 2. Code Standards

### 2.1 Variables

- Using **camelCase** for identifier names (variables and functions).
- All names start with a **letter**.
- Constants (like PI) written in **UPPERCASE**
- No unused variables.
- Avoid modifying variables of class declarations.
- No re-declaring variables.
- Avoid assigning a variable to itself.
- Avoid comparing a variable to itself.
- Restricted names should not be shadowed.

### 2.2 Spaces Around Operators

- Always put spaces around operators ( = + - \* / ), and after commas.

### 2.3 Statement Rules

- Put the opening bracket at the end of the first line.
- Use one space before the opening bracket
- Put the closing bracket on a new line, without leading spaces.
- Keep else statements on the next line after their curly braces.

### 2.4 Object Rules

- Place the opening bracket on the same line as the object name.
- Use colon plus one space between each property and its value
- Do not add a comma after the last property-value pair.
- Place the closing bracket on a new line, without leading spaces.
- Maintain consistency of newlines between object properties.
- Always end an object definition with a semicolon.

### 2.5 Line Length

- For readability, avoid lines longer than 80 characters

## **2.6 Spaces**

- Use 2 spaces for indentation.
- Add a space after keywords.
- Add a space before a function declaration's parentheses
- Commas should have a space after them.
- Add spaces inside single line blocks.
- No space between function identifiers and their invocations.
- Add space between colon and value in key value pairs.

## **2.6 Quotes**

- Use single quotes for strings except to avoid escaping.

## **2.7 Dot location**

- Dot should be on the same line as property.

## **2.8 Array**

- Use array literals instead of array constructors

## **2.9 Class**

- Use a single import statement per Class.
- Renaming import, export, and destructuring assignments to the same name are not allowed.

## **2.10 Functions**

- Avoid unnecessary function binding.
- No unnecessary parentheses around function expressions.
- No function declarations in nested blocks.

## **2.11 String**

- Regular strings must not contain template literal placeholders.
- No octal escape sequences in string literals.
- No multiline strings.
- No spacing in template strings.

## **2.12 Error catching**

- Only throw an Error object.

## **2.13 Files**

- Files must end with a newline.

## **2.15 Others**

- Semicolons must have a space after and no space before.
- Never start a line with (, [, `, or a handful of other unlikely possibilities.