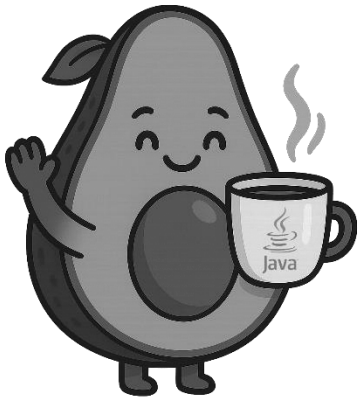
Javocado

Learning Java made Easier in one App

iLS for Mobile App Programming 1 and Computer/Web Programming 3

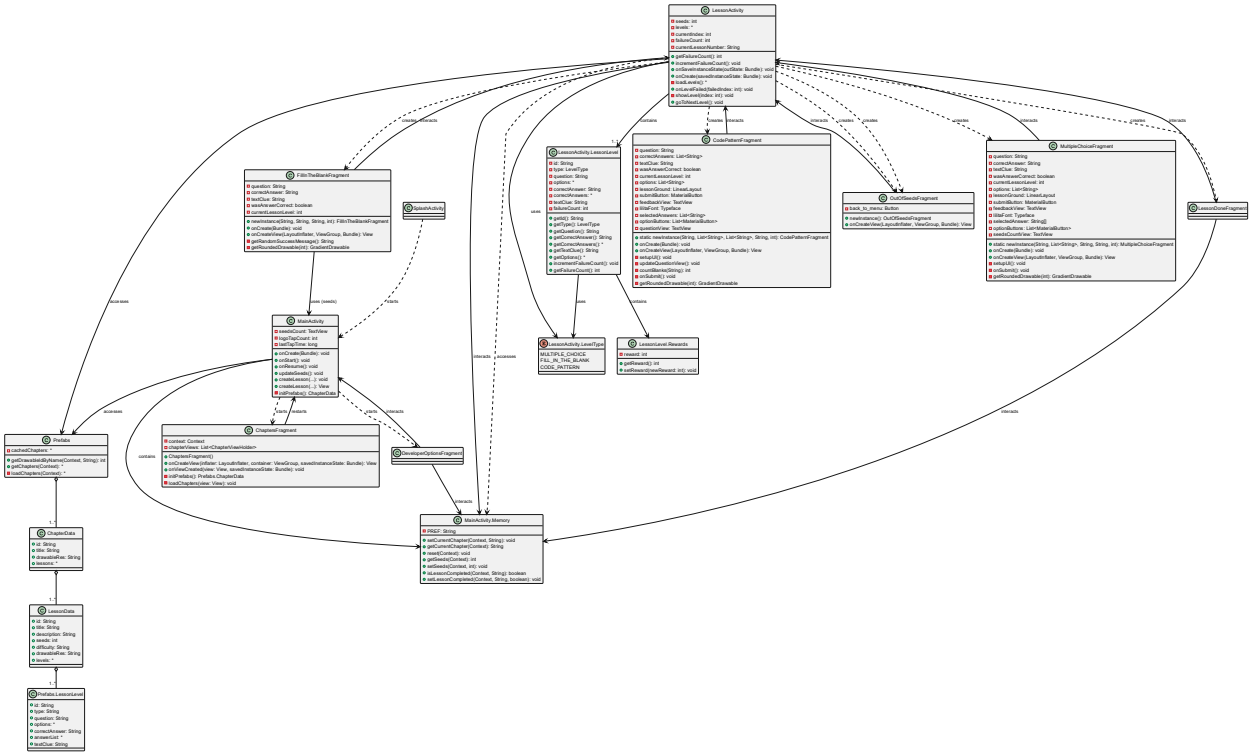
The Purpose of Javocado is to help Young and Inspired Programmers to Practice & Learn the Java Programming Language in a Gamified App.



2025

Javocado – Group 6
Documentation

Content:



1.MainActivity

- **Connections:**
  - Uses Prefabs to access chapter/lesson data.
  - Contains Memory as a static nested class.
  - Starts ChaptersFragment, DeveloperOptionsFragment, and is started by SplashActivity.
- **Description:**
  - Entry point of the app after splash screen.

- Handles UI setup, seed updates, and dynamic lesson creation.
- Maintains persistent app memory logic via nested Memory class.

2.MainActivity.Memory

Connections:

- Accessed by MainActivity, DeveloperOptionsFragment, LessonActivity, and LessonDoneFragment.
- Central storage interface for persistent data.

Description:

- Manages SharedPreferences: current chapter, seed count, lesson completion.

3.LessonActivity

Connections:

- Creates and interacts with all *Fragment classes related to lessons.
- Accesses Prefabs and Memory.
- Contains multiple LessonLevel instances and uses LevelType.

Description:

- Controls lesson flow: level loading, user progress, failure count.
- Navigates between question fragments and result fragments.

4.LessonActivity

Connections:

- Creates and interacts with all *Fragment classes related to lessons.
- Accesses Prefabs and Memory.
- Contains multiple LessonLevel instances and uses LevelType.

Description:

- Controls lesson flow: level loading, user progress, failure count.
- Navigates between question fragments and result fragments.

5.LessonActivity.LessonLevel

- **Connections:**
    - Contains Rewards, uses LevelType.
    - Accessed by all quiz-type fragments.
- **Description:**
    - Encapsulates data for individual levels: question, answers, clues, type.
    - Tracks user's performance (e.g., failure count).

6.LessonLevel.Rewards

- **Connections:**
    - Owned by LessonLevel.
- **Description:**
    - **Stores and provides access to the reward (e.g., seed count) of a level.**

7.ChaptersFragment

- **Connections:**

- Started by MainActivity, can restart it.
- Uses Prefabs to get ChapterData.

- **Description:**
  - Displays all chapters in a scrollable view.
  - Handles chapter click logic and progress visualization.

8.DeveloperOptionsFragment

- **Connections**:
  - Started by MainActivity, accesses Memory.
- **Description**:
  - Offers debug or developer settings.
  - May modify app state using Memory.

9.SplashActivity

- **Connections**:
  - Starts MainActivity.
- **Description**:
  - Initial screen shown to users before main app loads.

10.Prefabs

- **Connections**:
    - Provides chapter and lesson data to MainActivity, LessonActivity, and ChaptersFragment.
    - Owns multiple ChapterData.
- **Description**:
    - Central static utility for accessing predefined content structures.
    - Caches chapter data for reuse.

11. ChapterData

- **Connections**:
    - Owned by Prefabs, contains multiple LessonData.
- **Description**:
    - Represents a single chapter with metadata and associated lessons.

12. LessonData

- **Connections**:

- Belongs to ChapterData, contains multiple PrefabsLessonLevel.

- **Description**:
  - Encapsulates a lesson's metadata and list of levels.

13. Prefabs.LessonLevel

- **Connections**:
  - Used within LessonData as a prefab/template before being turned into runtime LessonActivity.LessonLevel.

- **Description**:
  - Raw level data from prefabs: includes answers, question, clue, and type.

14–18. Fragments for Levels

a. MultipleChoiceFragment

b. FillInTheBlankFragment

c. CodePatternFragment

- **Connections**:

- Created by LessonActivity, interacts with it and LessonLevel.

- **Description**:
  - Show specific quiz types, manage user input and answer validation.

d. OutOfSeedsFragment

- **Connections**:
  - Created by LessonActivity, interacts with it.
- **Description**:
  - Shown when the user runs out of seeds (energy).

e. LessonDoneFragment

- **Connections**:
  - Created by LessonActivity, accesses Memory.
- **Description**:
  - Final screen after lesson completion.

Advanced Concepts of Java:

- **Object-Oriented Programming (OOP):** Use of classes, inheritance, encapsulation, and composition (e.g., inner classes, fragments, and data models).

- **Advanced Methods:** Overriding methods like onCreate, onCreateView, onSaveInstanceState, etc.

- **Inheritance:** Passing data between fragments and activities using arguments and bundles.

- **Custom Data Models:** Use of custom classes for chapters, lessons, and levels, including enums and nested classes.

- **Shared Preferences/State Management:** Managing persistent state via static methods and shared preferences (e.g., MainActivity.Memory).

- **UI Component Manipulation:** Dynamic interaction with UI elements (e.g., updating TextView, handling Button clicks).
- **Intent and Navigation:** Using Intent for activity navigation and controlling the back stack.

- **Resource Management:** Accessing and managing resources (drawables, layouts, strings) programmatically.

- **Encapsulation and Access Modifiers:** Use of private fields and public methods for controlled access.

- **Collections and Generics:** Use of List<T> for managing groups of objects (e.g., lessons, options).

Javocado – Group 6

https://github.com/altxxr0/Javocado/

Member Contributions

1. ████████ **Faulve**
    - Majority of the Java Code, XML animations, Transitions to activity_splash.xml to home.xml, Dropdown & Lessons Capsule.
2. ████████ **Soleta**
    - Logo of Javocado, Color Pallete of Javocado, Levels Design in Javocado, Splash Screen Design, Javocado Level Complete Design.
3. ████████ **Fortaliza**
    - Dropdown Lesson Capsule Design, Home Layout Design, Side-Menu Design, Home Layout Gradient, Animations.
4. ██████ **Colobong**
    - Home Layout Toolbar Design, Chapter Capsule Design, Side-Menu Gradient Design, Level Design in Javocado, Javocado Fonts.
5. ███████ **Tejada**
    - Present /w Contributions (The Login/Signup [Old]) Until 2nd Quarter 2nd Term, Absent on the 1st of 2nd Quarter 2nd Term