DEADLINE: Friday, December 26, 2025

Yaşar University
Fall, 2025-2026
SE 3317 – Software Design and Architecture
Project

Notes:

- **Submit your Project to Moodle "SE3317 Project" assignment link. Do not send your project via email!**
- **Late submissions (submissions after the deadline) will not be evaluated.**
- **You can do your project individually or as a group. The group should include at most two students.**
- **The project requires the completion of two parts: 1- Implementation (Coding) and 2- Demo (Presentation).**

Moodle Submission Information:

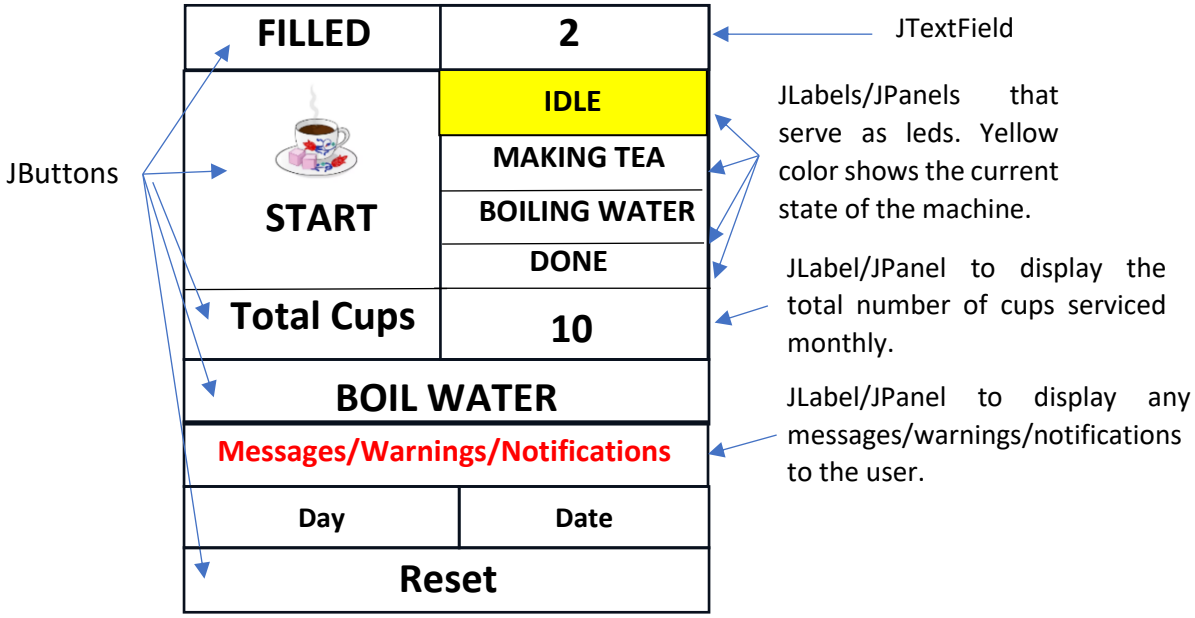
- Submit the Coding and Demo parts in a single zip/rar compressed folder to Moodle Assignment link "SE3317 Project".

1. **Code**
   - Requires the submission of all Java files and Database files.

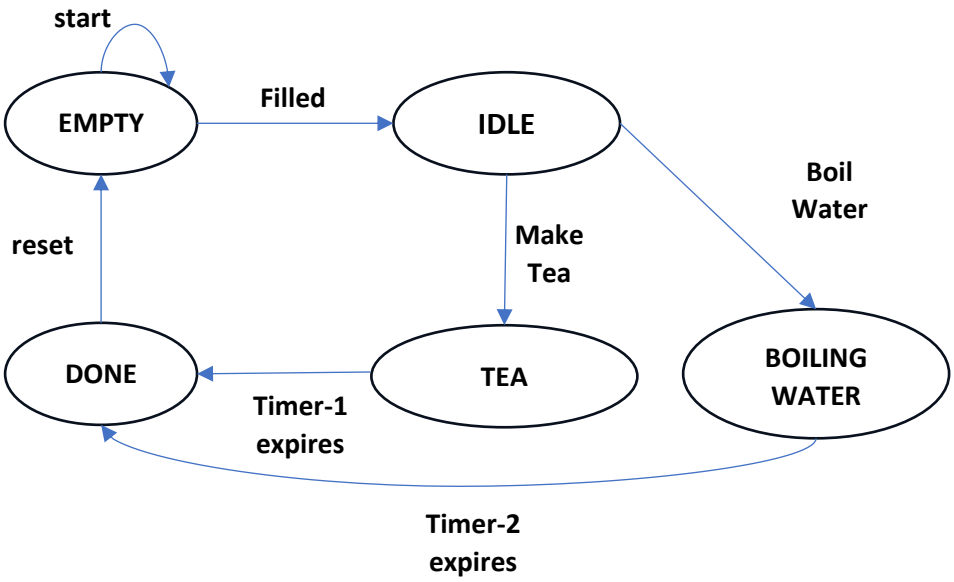
2. **Demostration (Demo)**
   - Prepare a Demo Video in English that demonstrates the execution of your application.
   - Each member should equally participate in the presentation.
   - Show the execution of each function in your application.
   - Show your code in the background when you are running your application.
   - Show the tables in your database.
   - Briefly explain the goal of each of your Java classes and which design patterns are used in which class(es).
   - **Duration:** 4-8 minutes.
   - **Format:** mp4
   - No powerpoint presentation is needed. The presentation on the IDE is enough.
   - Check to see if your demo video works before submitting it.
   - The project submission of only one group member to Moodle is enough. Put a **text document** in your project folder which includes the section number, name and surname, and student number of each group member.

INTRODUCTION

In this project, you are developing a "tea maker" application. The application will be designed as a **Desktop application with a Graphical User Interface (GUI)** using an **IDE** (NetBeans/IntelliJ) which will communicate with a **database** (MySql). The application should be developed with **Java** programming language and implemented using the **Model-View-Controller (MVC)** architectural design pattern. Along with the **Strategy, Observer, and Composite** design patterns that are used in this architecture, you should also use the **State and Decorator design patterns.** There will be a **single interface** as shown below that the user interacts with.



- **The state machine** of the application is as follows. The oval shapes represent the states while the arrows represent the state transitions.



❖ **State Machine Execution:**

- The initial state of the machine is “EMPTY”.
- When the user fills the machine with sufficient amount of tea and enters the number of cups that the machine is filled with and then presses the “FILLED” button, the state of the machine changes to “IDLE”. “IDLE” state shows that the machine is ready to go.
- If the user presses the “START” button when the machine is in “EMPTY” state, then a warning message should be displayed to the user.
- If the user presses start button when the machine is in “IDLE” state, then the state of the machine changes to “TEA” and a timer (Java Timer) is started.
- If the user presses “Boil Water” button when the machine is in “IDLE” state, then the state of the machine changes to “BOILING WATER” and a timer (Java Timer) is started.
- When the timer (timer-1 or timer-2) expires, the state of the machine is changed to “DONE”. In “DONE” state, the number of cups and the current date should be inserted in a database Table.
- Assuming that the user gets the prepared tea after it is done, when the user presses the “RESET” button, the state of the machine changes to “EMPTY” and the texts in the textfields are deleted.
- When the machine is in “TEA” or “BOILING WATER” states, disable the “Filled”, “Start”, and “Boil Water” buttons. Enable these buttons when the state of the machine is changed to “DONE”.
- When the current state of the machine changes, you should also change the background of color of the IDLE/MAKING TEA/BOILING/DONE JLabels/JPanels.

❖ “Total Cups” button is to display the total number of cups that are consumed by the user in the current month.

❖ Whenever the number of total cups in a day is above a threshold value, the machine will display a notification (e.g. “The total numbers of cups today has reached to 10!”).

Decorator Pattern: You should use the **Decorator Patten** in the part explained below:

- The **sections that are involved** for this part are: “Day”, “Date”, “Message/Warnings/Notifications”.
- The **basic message** to the user includes the **the day, date, and state of the machine** which are shown on Day, Date, and Messages/Warnings sections, respectively. The basic message can simply concatenate day, date, and state as a string which are separated with a delimiter such as comma, new line, etc.
- Whenever the number of total cups in a day is above a threshold value (e.g. 10 cups), the **basic message should be decorated** with a health warning message “The number of total cups today has reached to …”.