

## PROJECT-General Description

### PART #1

CS 586; Spring 2023

#### Deadlines:

**Part #1: MDA-EFSM (5 points): Monday, April 3, 2023**

Late submissions: 50% off

After **April 8** the MDA-EFSM will not be accepted.

This is an **individual** project, not a team project.

**Submission:** The MDA-EFSM assignment must be submitted on Blackboard. Your submission should be as a **single PDF-file** (otherwise, a 10% penalty will be applied). The hardcopy submissions will not be accepted.

**The detailed description of Part #2 of the project will be posted later.**

#### Goal:

The goal of this project is to design two different Vending Machine (VM) components using the Model-Driven Architecture (MDA) and then implement these Vending Machine components based on this design.

#### Description of the Project:

There are two Vending Machine components: VM-1 and VM-2.

**VM-1** component supports the following operations:

create(int p);	// starts a vending machine application, where $p$ is an initial price of a drink
coin(float v);	// a coin with value $v$ is inserted
sugar();	// sugar button is pressed
tea();	// tea button is pressed
latte();	// latte button is pressed
insert_cups(int n);	// $n$ cups are inserted into the vending machine
set_price(float p);	// new price of a cup of tea/latte is set to value $p$
cancel();	// cancel selection for a cup of tea or latte

**VM-2** component supports the following operations:

CREATE(float p);	// starts a vending machine application, where $p$ is an initial price of a drink
COIN(int);	// a coin with value $v$ is inserted
CARD(int x)	// credit card is swiped, where $x$ is an available fund
SUGAR();	// sugar button is pressed
CREAM();	// cream button is pressed
COFFEE();	// coffee button is pressed
InsertCups(int n);	// $n$ cups are inserted into the vending machine
SetPrice(int p);	// new price of a cup of coffee is set to value $p$
CANCEL();	// cancel selection for a cup of coffee

Both Vending Machine components are state-based components that control simple vending machines. These vending machines dispose of a cup of tea/coffee/latte with or without additives (sugar and/or cream). The detailed behavior of both Vending Machine components is specified using EFSM. The EFSM of Figure 1 shows the detailed behavior of VM-1, and the EFSM of Figure 2 shows the detailed behavior of VM-2. Notice that there are several differences between both Vending Machines.

Aspects that vary between the two Vending Machine components:

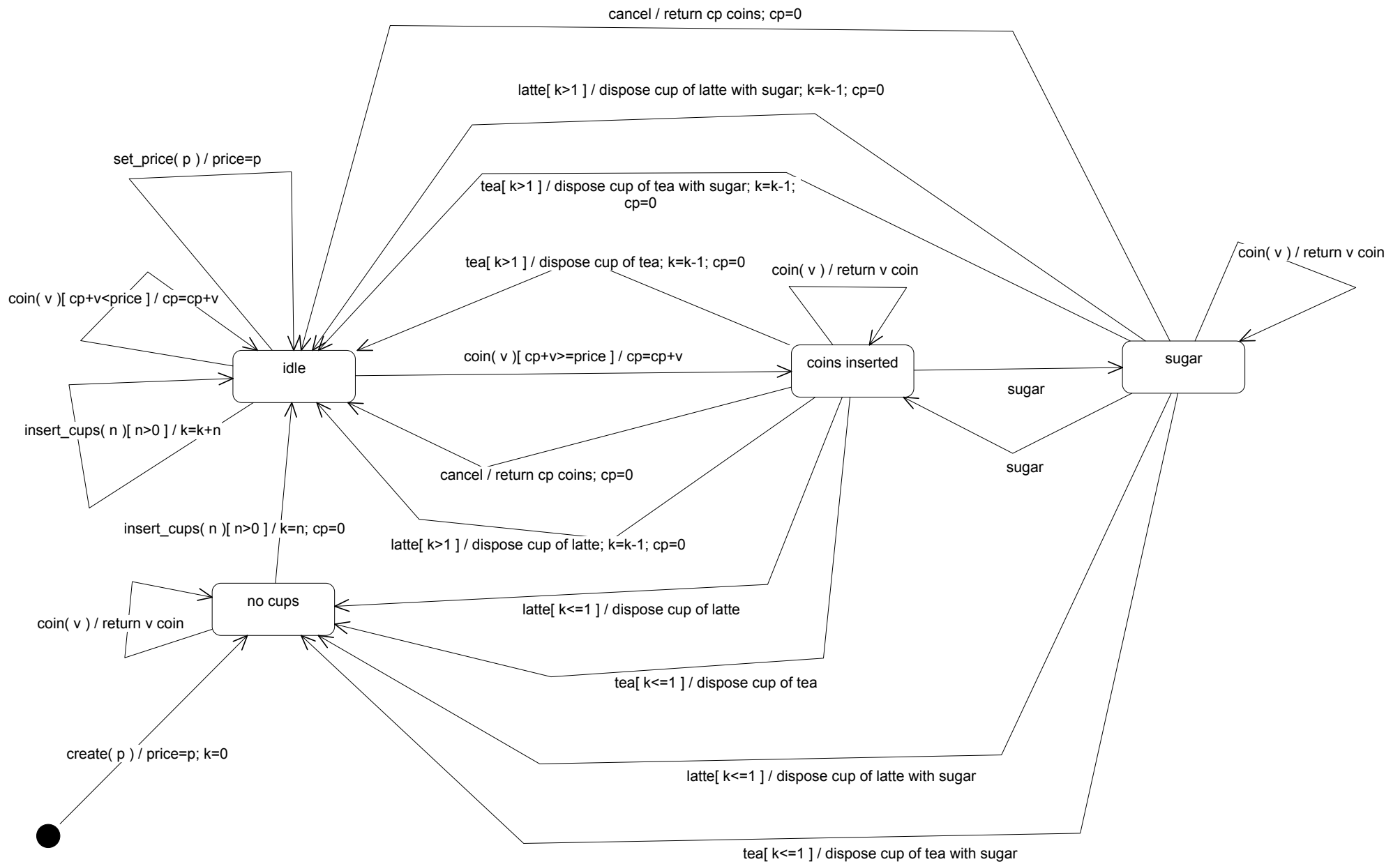
- a. Different methods of payment
- b. Different drinks disposed
- c. Different additives offered
- d. Different messages displayed
- e. Different operation names and signatures
- f. Different data types
- g. etc.

The goal of this project is to design two VM components using the Model-Driven Architecture (MDA) covered in the course. An executable meta-model referred to as MDA-EFSM of VM components should capture the “generic behavior” of two VM components and should be de-coupled from data and implementation details. Notice that in your design there should be **ONLY** one MDA-EFSM for two VM components. The meta-model (MDA-EFSM) used in the Model-Driven architecture should be expressed as an EFSM (Extended Finite State Machine) model. Notice that the EFSMs shown in Figure 1 and Figure 2 are **not acceptable** as a meta-model (MDA-EFSM) for this model-driven architecture.

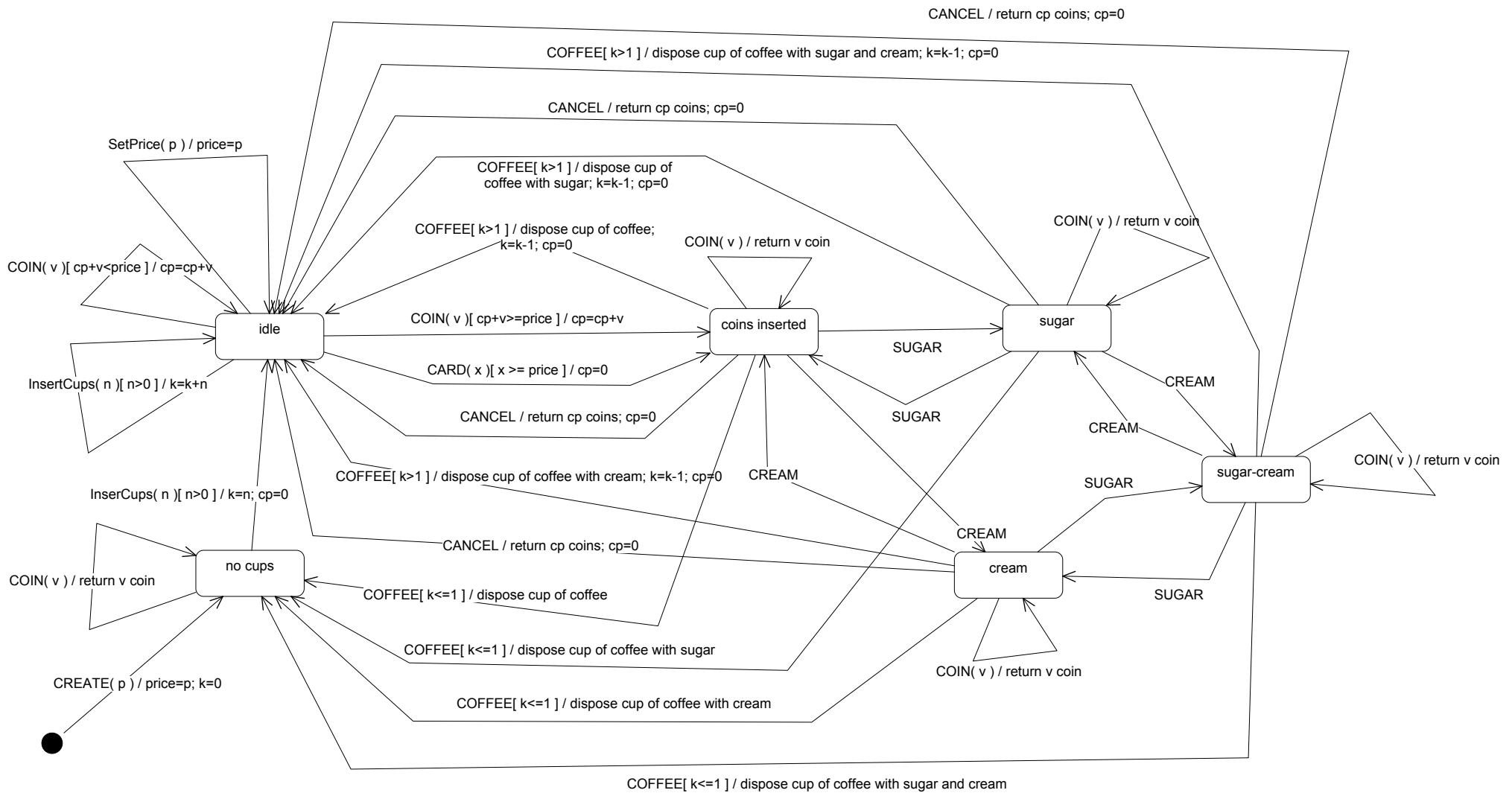
### **MDA-ESM REPORT SUBMISSION**

The MDA-EFSM model report for the *VM* components should contain:

- A class diagram
- A list of meta events for the MDA-EFSM
- A list of meta actions for the MDA-EFSM, where the responsibility of each action must be described
- A state diagram/model of the MDA-EFSM
- Pseudo-code of all operations of Input Processors of *VM-1* and *VM-2*



**Figure 1: EFSM of VENDING-MACHINE-1**



**Figure 2: EFSM of VENDING-MACHINE-2**