

# Drink Machine

Finite State Machine

Vendredi 4 Dec 2020

**Keyan FANG**

Yao Lu

Chargé de TD  
Julien Deantoni



<b>Introduction</b>	<b>2</b>
Presentation of sujet	2
<b>Presentation of solutions</b>	<b>2</b>
state chart	2
Ordering section	4
Payment	4
Preparation Section One	4
Preparation Section Two	4
Bar	5
Cup Detection	5
<b>LTSA and V&amp;V</b>	<b>5</b>
<b>What would we have done differently</b>	<b>6</b>
<b>Auto-évaluation</b>	<b>6</b>

# Introduction

## Presentation of subject

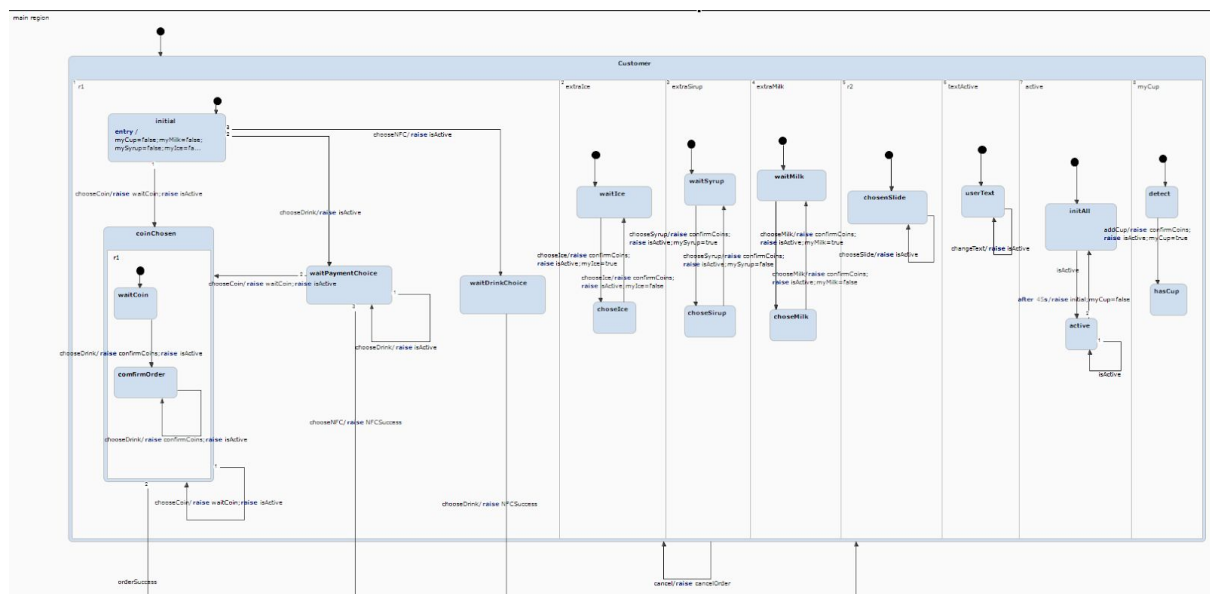
The goal of this subject is to simulate a drinking machine based on given code to the complete version that can offer people various kinds of drinks including tea, coffee and espresso and allows people to pay either by coins or NFC. After success payment the machine will automatically prepare the chosen drink and give it to the customer and after the drink was taken the machine will self-clean and be prepared for the next client.

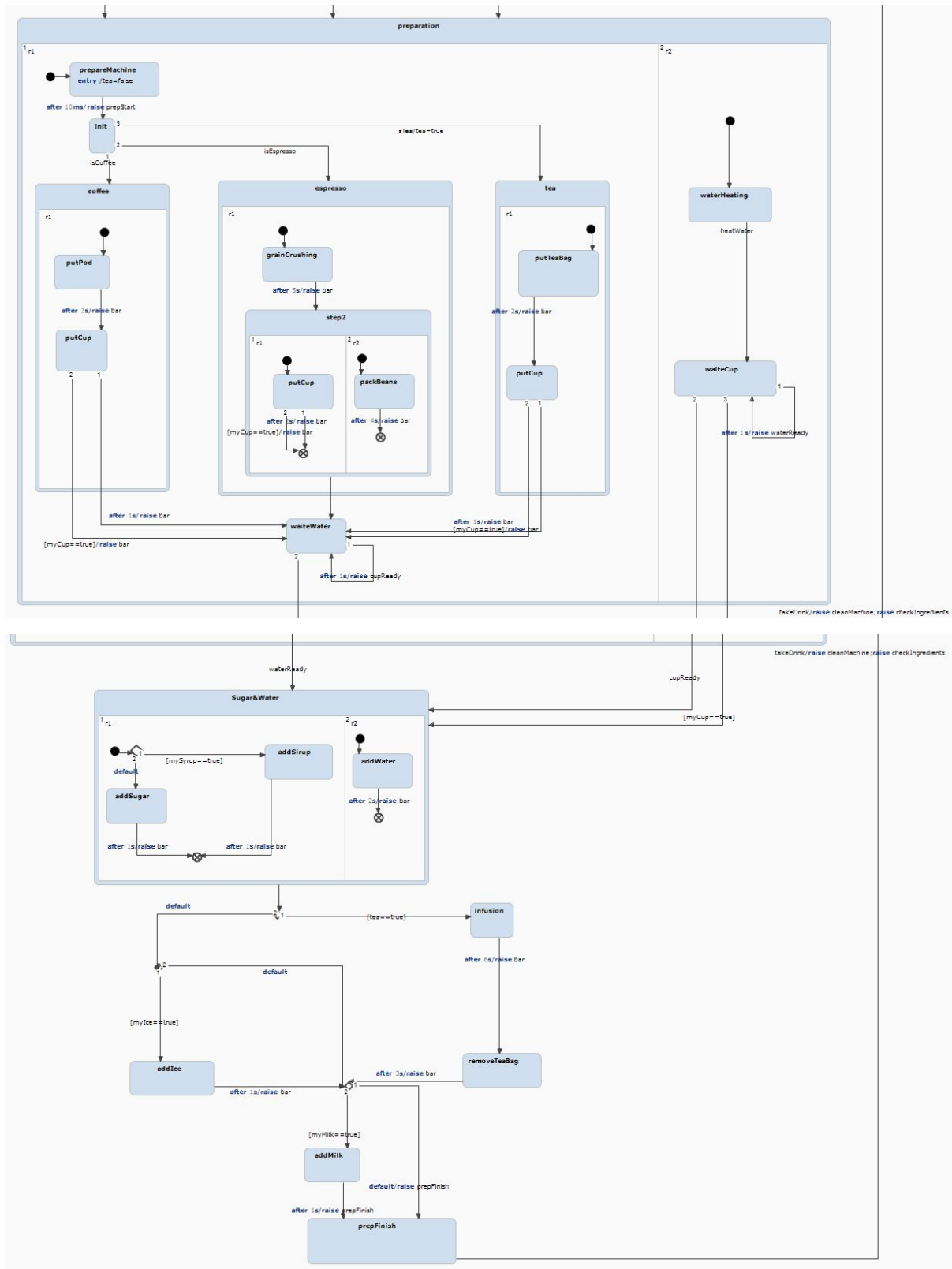
For our project, we have implemented all tasks needed for the MVP and we have chosen to realize the following extension tasks:

1. Gestion de l'avancement de la préparation
2. Détection des gobelets

## Presentations of solutions

## state chart





The statechart of our drinking machine is composed of two main parts: the ordering part where the customer chooses the drink along with the extra adding and pays for it; and the preparation part where the machine prepares and serves the drink to the customer. And the

preparation part can also be divided into two sections: 1,the water and cup are not ready  
2,water and cup are both ready

## Ordering Section

The ordering section is a composite state which has many parallel states inside. We choose to draw the statechart into what it is now because according to the rules, some events should take place at the same time and some have effects on others.

There are eight regions in total and all these regions must be paralleled and end at the same time. For example a customer can choose sliders and pay for the drink at the same time. And the re-initial state guarantees that as long as there's no interaction from the customer for 45 seconds everything is set back to the initial state and it always detects this anytime before the preparation stage starts. So these eight regions are in parallel together, and after the payment is successful, the composite state ends and enters the preparation state.

## Payment

There are three situations for payment in general: 1. Choose a drink after using NFC. 2. Use NFC after choosing a drink. 3. Coins are performed before and after the drink is selected.

Therefore, the payment process cannot be done in parallel. These three payment behaviors are incompatible with each other, and users can only perform one payment situation. But if the user chooses a drink first, then there will be two possibilities for payment with coins or NFC. If he uses NFC to pay, he can go directly to preparing drinks. If he uses coins to pay, he enters the third situation mentioned before.

## Preparation Section One

The first stage of the preparation of drinks is composed of two parallel parts: 1.the preparation of water 2.the preparation of ingredients and cup. We decided to make these two regions parallel, because all drinks have the same process of preparing water, and the time for preparing water will vary due to user choices. Therefore, it is more concise and clear to abstract the process of preparing water.

On the other hand, based on the type of drink chosen by the customer, after the initial stage the machine will go into three sets of states and make three sets of preparations before the infusion of water.

And we have to make sure that the cup is placed when the water is flowing, so in the last two states of the two regions, they wait for each other to finish, so as to ensure that both sides are ready then starting to pour water into the cup.

## Preparation Section Two

The second stage of the preparation of drinks has less parallel regions as most states have to wait for the former one to finish. And we have chosen to add more than one 'choice' here for the extras to be added correctly. For the adding of milk/syrup/ice, each comes in two different situations and the state chart determines the different following states. The states are put in the chronological order and as the sugar/syrup and water are added at the same time it's only possible to put them in a composite state and make them parallel.

At the end of the preparation stage the prepFinish is raised, and after the drink was taken by the customer the machine is cleaned and initialized. We choose to clean the machine after each drink because comparing to cleaning the machine before the preparation starts, this can shorten the waiting time of the customer and is easier for the machine to be cleaned properly (if there is a huge time gap between two customer it may be more difficult for the machine to be cleaned as stains become more sticky during time)

## Bar

And between each state during the preparation part the out event bar is raised to change the progress bar. The bar is meant to show each progress made inside the machine so it has to be raised each time there is a state change. And it can not be an individual state in a parallel region because states in a parallel region can be triggered anytime during the composite state but the bar must advance only after the finishing of each state. So making it an out event is the most accurate as for each state the progress bar needs to be changed to show the difference.

## Cup detection

The user should put the cup before the preparation of the drink, but the user puts the cup to affect whether the machine puts the plastic cup during the preparation of the drink, so I used the boolean variable to save this information.

## LTSA and V&V

According to the LTSA code, we can verify that this system works in the correct way and has no error in its programming. And it can serve customers the drink chosen and repeat this process without extra interference so it's validated to be a state machine which works in good condition.

The result of Check LTL property:

Compiled: Mutex

Compiled: Order

Compiled: Prep

Formula !Prop = (true U (initial & (false R !chooseNFC)))

...  
No LTL Property violations detected.

As the process goes when the user(customer)has chosen a drink and the machine demanding the payment will receive a payment so it has liveness property.

## What would we have done differently

### 1. Customer cup size detection and alert on unmatching sizes

Right now no matter what the cup size of the customer the machine will still offer drinks so this might lead to a cup too small for the drink size chosen by the customer. We could offer more than one pics of different cup of different size as an in event "pickCup" for the customer to choose from and an out event "alertSize" will alert customers that the cup chosen is smaller than the drink size chosen and the machine will not prepare the drink for the customer unless another drink size is chosen or a bigger cup is placed.

### 2.Smooth progress bar

We could use a parallel state. The parallel state is used to perform the operation of the progress bar forward action, and a smooth effect should be achieved by making this state self-circulate with "after 100ms". The "bar" out event that we wrote would be just to ensure that the progress bar reaches the corresponding value and tell the user which step is currently reached.

## Auto-évaluation

Name	Yao LU	Keyan FANG
<b>Finished work</b>	Payment code and state chart, Cup detection code and state chart, code and state chart for "Options" function	Progress Bar code and state chart, extra adding code(partial) and state chart,code of LTSA and V&V
<b>Points(200/200)</b>	100	100