

COMP1531 WEEK 8

GOING THRU IT

TIMELINE

Week 8

- Iteration 2 Demo
- Iteration 3 Release
- Lab08

Week 9

- Lab09

Week 10

- Iteration 3 leaderboard
- Iteration 3 Due

Week 11

- Take home exam
- Thursday

COVERAGE – WHY?

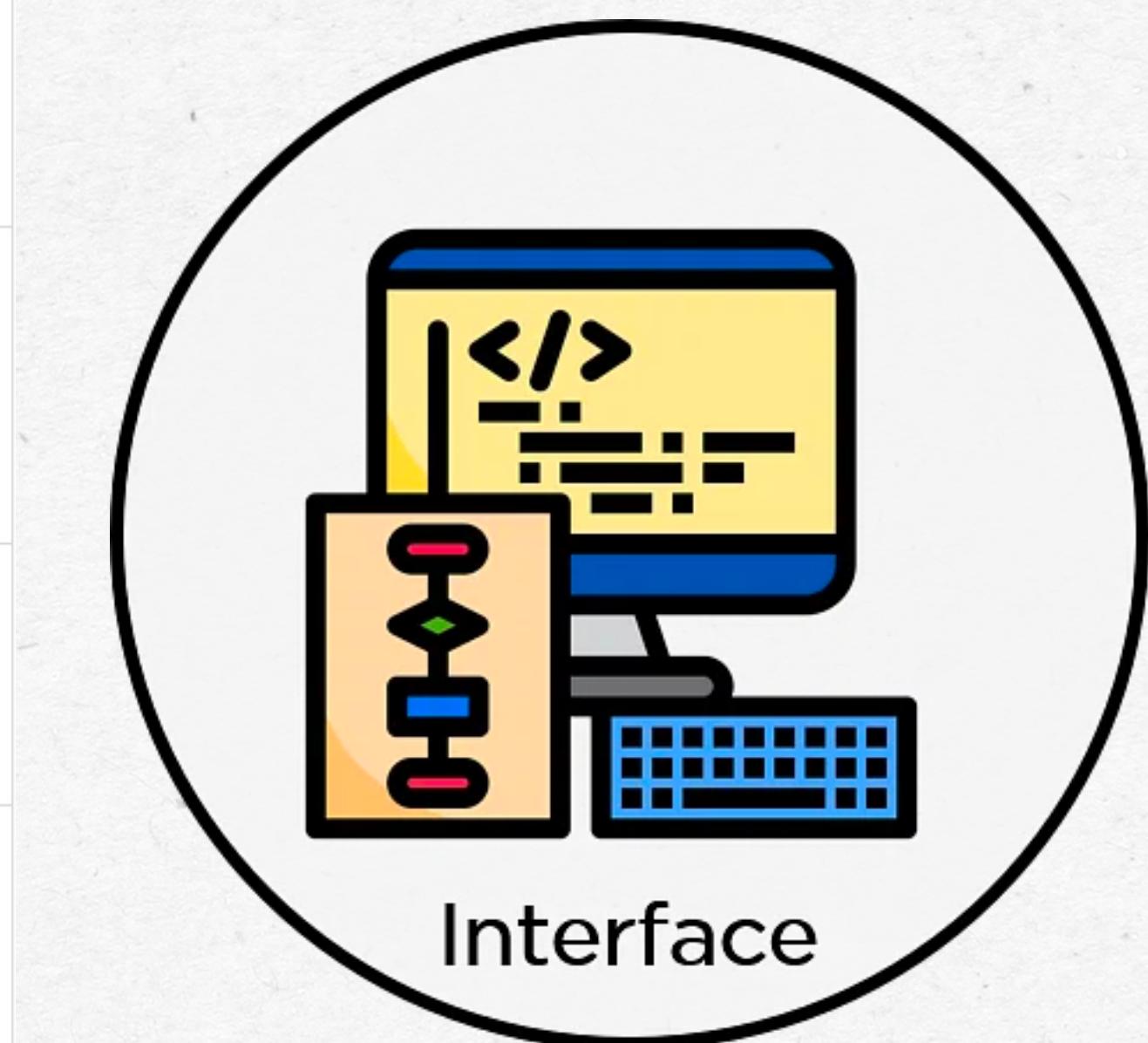
```
const ORIGIN_YEAR = 1970;

const isLeap = (y: number) => new Date(y, 1, 29).getDate() === 29;

export const dayToYear = (days: number) => {
  let year = ORIGIN_YEAR;
  while (days > 365) {
    if (isLeap(year)) {
      if (days > 366) {
        days -= 366;
        year += 1;
      } else {
        continue;
      }
    } else {
      days -= 365;
      year += 1;
    }
  }
  return year;
};
```

INTERFACE

Name & Description	HTTP Method	Data Types	Errors	Variables	Type
/movie/add Adds a movie to the data store.	POST	Body Parameters: {title, director} Return Object: {movieId}	{error} when any of: <ul style="list-style-type: none"> • title is an empty string, "" • director is an empty string, "" 	is exactly error	string, with the value being a relevant error message of your choice
/movie/:movieid Edits a movie in the data store.	PUT	Body Parameters: {movieId, title, director} Return Object: {}	{error} when any of: <ul style="list-style-type: none"> • movieId does not refer to an existing movie • title is an empty string, "" • director is an empty string, "" 	contains suffix Id	number, specifically integer
/movies/list Lists all movies in the data store.	GET	Query Parameters: {}	N/A	is exactly title	string
/clear Delete all movie data and return an empty object.	DELETE	Query Parameters: {}	N/A	is exactly director	string
				is exactly movies	Array of objects, where each object has type {movieId, title, director}



MEASURING COVERAGE SERVER-SIDE

JEST

`jest --coverage` only tracks the lines of code covered by the execution of the tests, which works for imported functions.

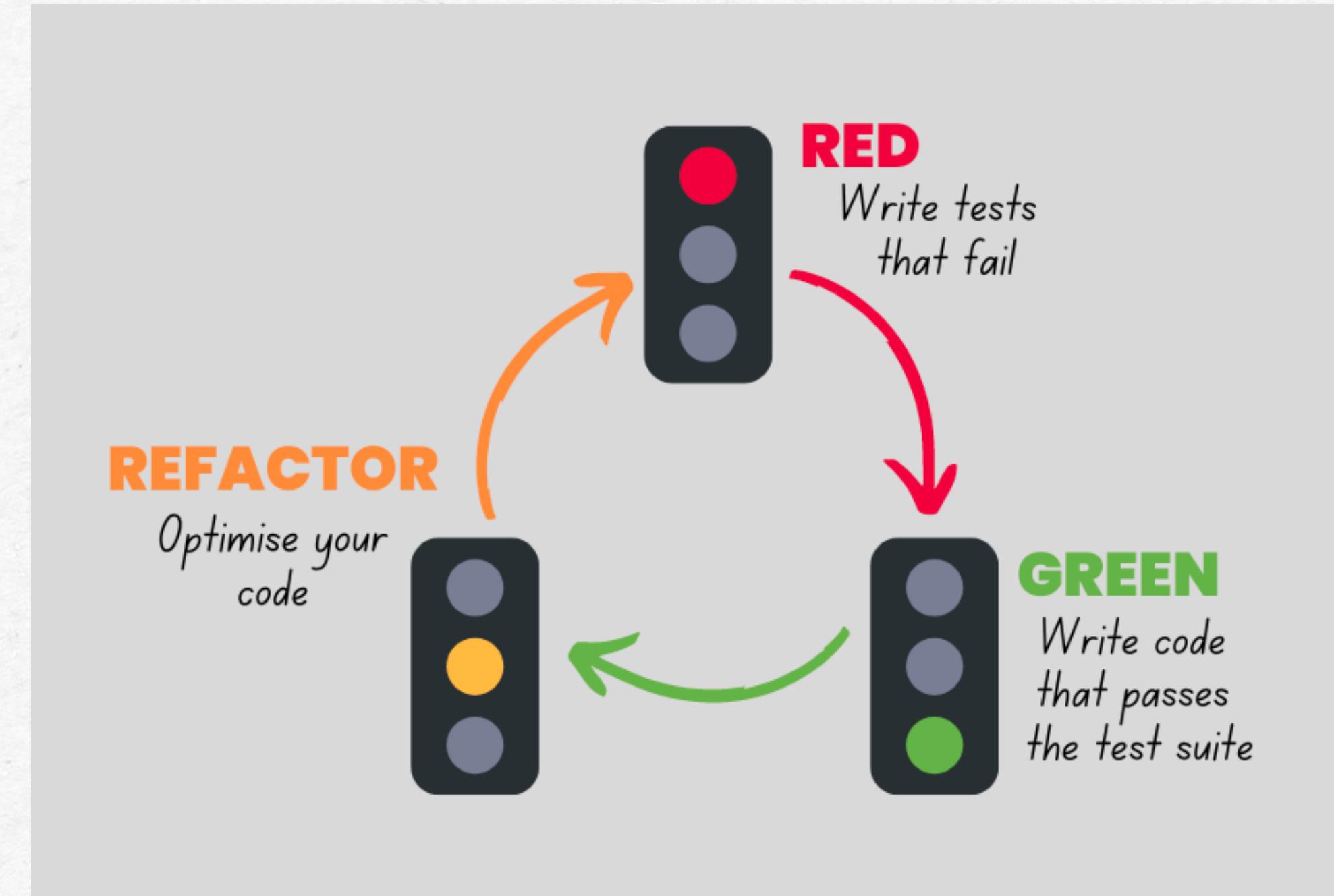
SERVER

However, our backend, the server and tests are running on two separate NodeJS instances. There is no way for Jest to track coverage on the server through sending http requests.

SOLUTION

This is why we use nyc to start the server and measure its coverage directly (on the server side, as opposed to client side).

REFACTORING WITH EXCEPTIONS



REMOVE THE ERROROBJECT FROM THE UNION RETURN TYPE OF THE MOVEADD AND MOVEEDIT FUNCTIONS

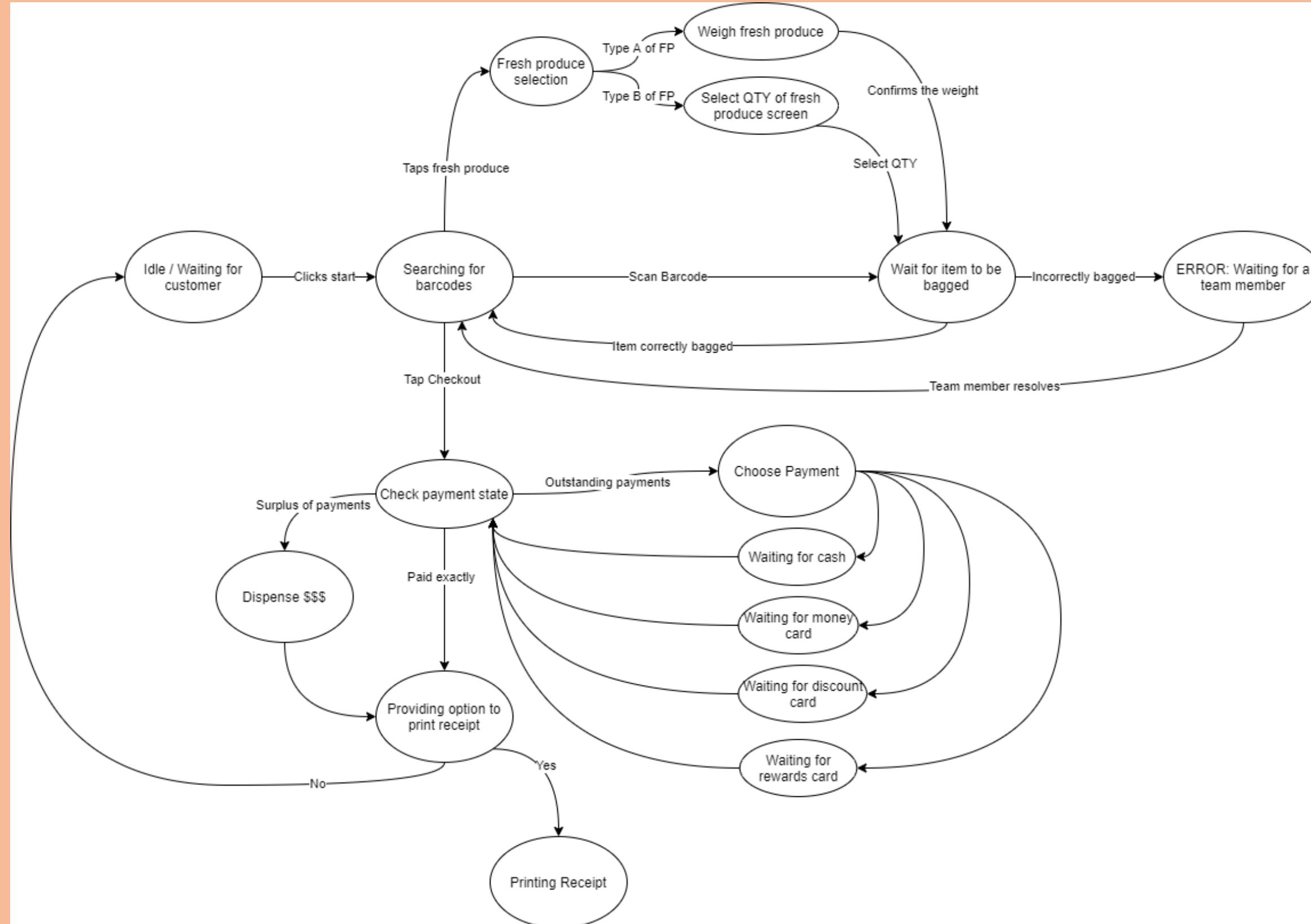
REPLACE RETURN { ERROR: '...' } WITH THROW HTTPERROR

SYSTEM MODEL – STATE DIAGRAM

Create a state diagram to describe the states and subsequent transitions that would occur for a grocery store checkout system, from the perspective of the user-machine interaction.



SAMPLE SOLUTION



QUESTIONING OUR MODEL

01

What are some
limitations of our
current model?

02

How can our model be
used in the designing
stages of software?

LAB TIME!

