Dear candidate,

Thank you so much for taking your time to answer the following technical questions.

Please consider this a way that we can get familiar with your technical knowledge. We would like to know the cool things you have being done and how you do think about technology.

On the following board, feel free to provide us any additional info that you want related to the questions or anything else.

You can choose 1-3 technologies questions to answer. It is up to you.

We thank you in advance for doing this and wish you the best in your career.

Sincerely,

Team members

|  |  |
| --- | --- |
| Your Name | Priscilla Sewaybricker Rizzardo Guimarães |
| Date of testing | 09/05/2019 |
| Additional comments from candidate |  |

**ANGULAR**

1. **How to generate a class in Angular 7 CLI?**

Angular 7 CLI helps with many commands to create components, class, directive, service etc.

If you want to generate a class you just need type:

**ng generate class <name>**

**or**

**ng g class <name>**

1. **What is Angular Material?**

Angular Material it’s a UI library for Angular based on Material Design. It helps to constructing attractive components to web applications. There is many components available (layouts, buttons, forms, checkbox, etc)

1. **Could you explain services in Angular?**

Services allow you to define code that's accessible and reusable throughout multiple components. A common use case for services is when you need to communicate with a backend of some sort to send and receive data.

1. **How do Observables differ from Promises?**

Observable is lazy, you can define when you want to use it and reduce the time of process when the request is cancel, or retry when you had loose connection, so it could spend time to execute the operation and this could helps in some needs. To execute the observable you should use the method subscribe() at this way the operation will be executed and get the value.

Promises is eager, it represents actions to async codes. It has callback function (resolve and reject) to call the methods and return with a promise, and the methods then() and catch() to return with a promise and receive the value.

1. **What is Data Binding? Can you describe one?**

Data-binding in Angular apps is the automatic synchronization of data between the model and view components.

Property binding: [property]=”value”

With property binding, the value is passed from the component to the specified property, which can often be a simple html attribute:

<input type="email" [value]="user.email">

1. What is ngOnInit() and how to define it?

A lifecycle hook that is called after Angular has initialized all data-bound properties of a directive. Define an ngOnInit() method to handle any additional initialization tasks.

If you have a lot of processing to do when the component gets created, it’s good practice to do it in the ngOnInit hook rather than in the constructor.

1. What is SPA in Angular and how it works?

Single page application (SPA) is a web application that fits on a single page. All your code (Typescript/JS, HTML, CSS) is recovered with a single page stack. Further more, route between pages performed without invigorating the entire page.

**EXERCISE**

**Goal**

The purpose of the exercise is to evaluate your knowledge about Angular, Angular Material, Node.js, Express, and testing.

**Introduction**

Monsanto invests heavily in seed technology to maximize yield in any soil and weather conditions. Our commercial department works very closely with farmers. These farmers plant seeds with our technology and expect maximum yield.

**Description**

Following a product workshop, where business needs are identified and captured as high level stories, The development team then grooms all high level stories and breaks them down into small vertical slices. The following story is prioritized and moved from our backlog to the sprint:

***“AS Commercial User, I want to know the hectares planted divided per Lots from each farmer, So with that I can know how much the farmer needs to pay.”***

After analyzing the requirement, the team have sliced the story in small tasks. The following is one of the tasks that need to be tackled to complete the story.

* One webpage component, that could be reused, with the following:

**Angular 7, Angular Material**

* + Component to search farmers by document # or Name.

Technical Requirements

Website:

* Technologies:
  + Angular 7
  + Angular Material
* Pages:
  + Search form:   
    Farmer search form should match mockups below.

If the search has more than one result, return any result that match’s.  
  
Once the farmer is selected, the farmer’s details should be populated in the correct inputs.

The component should have, at least, one input and one output:

@Input() farmerSearchAbstractProvider: FarmerSearchAbstractProvider;

@Output() onPartnerSelectedEvent: new EventEmitter();

This abstract provider should be implemented to allow us to reuse the same component for different source of farmers

**import** { Farmer } **from** '../model/Farmer;  
**export declare abstract class** FarmerSearchAbstractProvider {  
 **abstract** searchFarmers(params: SearchParams): Promise<Farmer[]>;  
}

Use this farmer model as sample:

**export interface** Farmer {

id: **string**;

document: Document;

name: **string**;

address: Address;

}

**export interface** Address {

street: **string**;

state: **string**;

address: **string**;

country: **string**;

}

**export interface** Document {

documentNumber: **string**;

documentType: **string**;

}

The component should looks like this:

<farmer-search-card

(onFarmerSelectedEvent)="mySelectedFarmer($event)"

[farmerSearchAbstractProvider]="myFarmerSearchProvider">

</farmer-search-card>

**Backend:**

* Technologies:
  + Node.js
  + Express
  + PostgreSQL
* Endpoints:
  + GET – Farmer search

Endpoint should accept one search parameter.

Endpoint should take parameter and search against farmer’s document # and name.

Endpoint should return list of search results.

Mockups

UX team collaborates with users and comes up with a mockup. This is the look and feel we need to follow. **This is a sample mockup, It is not mandatory to follow it as is in your solution and also it would need some extra CSS that is not required to follow for this exercise.**

Card for farmer search:

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

What is needed?

Delivering a implemented and working component.

The source data (Farmers) can be hardcoded. (not in the component but it could be in the backend side)

Using any open source library.

Include all tests (unit, integration tests is considered bonus)

Publish the solution to a public GitHub repository and share the URL as response at pre-interview form.

Include a README.md about how to install and run.

**JAVASCRIPT**

1. Can you name two programming paradigms for JavaScript?

JavaScript is a multi-paradigm language, supporting imperative/procedural programming along with OOP (Object-Oriented Programming) and functional programming.

### Could you name the 3 ways to define a variable in JavaScript and what the differences between those**?**

There is 3 ways to define a variable using var, let, const. Var is scoped to the nearest function block and let is scoped to the nearest enclosing block.

Technically a const (constant) isn’t a variable. The particularity of a constant is that you need to assign a value when declaring it and there is no way to reassign it. A const is limited to the scope of the enclosing block, like let. Constants should be used whenever a value must not change during the applications running time, as you’ll be notified by an error when trying to overwrite them.

### **How do you create an object in JavaScript, can you give us a code example?**

var person = {  
  firstName: "Priscilla",  
  lastName: "Guimarães",  
  age: 30   
};

### **What are the scopes of a variable in JavaScript?**

The scope of a variable is controlled by the location of the variable declaration, and defines the part of the program where a particular variable is accessible.

### **What is callback and callback hell?**

A callback is a simple function that’s passed as a value to another function, and will only be executed when the event happens.

### **What is closure? Give an example.**

A function can be created dynamically, copied to another variable or passed as an argument to another function and called from a totally different place later.

### **What would be the result of 2+5+”3″ in JavaScript?**

The result is 73, 5+3 = 7 and the 3 is a string so concate with the result.

### **What is the purpose of ‘this’ in JavaScript?**

This in JS it’s a little different than other languages. You can use it to define how the function can be called. To understand this keyword, only we need to know how, when and from where the function is called, does not matter how and where function is declared or defined.

## Why would you favor object composition over class inheritance?

## What is asynchronous programming, and why is it important in JavaScript? Can you give us a code example of asynchronous code?

## Asynchronous means that things can happen independently of the main program flow. JavaScript environments typically implement this style of programming using callbacks, functions that are called when the actions complete.

**JAVA**

1. What is the difference between class and object?
2. In a few words explain the Singleton pattern.
3. Sort the array {1,7,6,3,2,9} using any sort algorithm and tell the name of the one used.
4. What is the purpose of Serializable interface?
5. What is polymorphism? Does it work with Java language?
6. Given the following snippet:

**public** **static** **void** main(String[] args) {

String String = "Hello";

**if**(String == "Hello") {

System.***out***.println(String + " World");

} **else** {

System.***out***.println("World");

}

}

Does this code has a compile or execution failure or it prints something on the console? If fails why does it fail? If runs, what does it print and why?

1. What is the difference between List, Set and Map? When you use one over another?
2. What Java Frameworks have you used until now? Tell a little bit about it.