**A – NON-TECHNICAL QUESTIONS:**

1. Introduce yourself and your working experiences?

I am Thach. I worked on FE about 4 years. My main stack are angular and reactjs. Sometime I also work in Vuejs project. I can work with many FE library or FE framework because I see that they have some the same point such as the component, data binding, state, props, life circle and the way to passing data. They are different a little bit of syntax and some other things

1. Which angular version did you work with

I worked with angular 8, 9 and the last time is 13. I had to upgrade from angular 5 to angular 13

1. What is angular, what exactly do you understand Angular?

Angular is a framework lead by angular team at google. It make the web application easy to understand, easy to scale and maintain

1. Did you meet any challenges in your last project? What did you do to solve it?

I thinks it have no big issues except that the communication is quite hardly because we worked by online and we have the different working time. To resolve it, we have to defined some simple checklist and follow them and have to tried to make clear the requiments or the issues in each meeting

1. Did you join a project in the working process of Agile/ Scrum?

Yes of course, most of my project working with Scrum, We break tasks enough for a sprint, and focus to complete each sprints. We have a sprint planning meeting in head of each sprints and have daily meeting every day, we also have retrospective and refinement meeting but it is not frequently

**B - TECHNICAL QUESTIONS**

1. What are the key components of it?

The key components of angular are: components, modules, services, templates, databinding, directives, …

1. Components?

The components are fundamental units that make up an angular application and manage HTML views

Every components have 3 parts, template or templateUrl, styleUrls, and a Class which handle business logic for the components

1. Modules

An angular module is a group of the framework’s fundamental building parts such as components, services, directives and pipes.

It had 2 types: root module and feature module

There’re only one root module but can have one or more feature modules

1. Services

They are used to build parts that may be used by the entire application.

1. Directives? How many types of directives are there?

A directive is guidelines for rendering a template. It allows us to attach attributes to the element and with this attribute we can handle the behavior of a DOM element

There are 3 type of directives: component directives, stuctural directives, attribute directives

1. Pipes

Angular pipes are usually used to transform the output before the data is displayed

1. How can you include a service in the root modules?

Push them into providers.

1. The providers provice dependencies in modules
2. We need to push new components, directives or pices to the Declaretions
3. We need to push some modules to imports array to use functionnality of those modules
4. Other module can use some service, directtive, pipe of that modules if we push it into export array
5. A class decorator tells Angular if a particular class is a component or a module.
6. Life circle
   1. Contructor: is first called
   2. ngOnChanges: gets called after contructor and also when any Input properties of the components changes
   3. ngOnInit: gets called after ngOnchanges and only once
   4. ngDoCheck
      1. …
   5. ngOnDestroy: gets called before Angular destroys the components
7. JIT (Just-in-time): it is for development enviroment and it is default option when we start with angular
8. AOT (Ahead-of-time): it is for production enviroment
9. How to share data between components in Angular?

There are 3 ways:

The first is use Input and Ouput to share data between parent and children components

The second is use the service. With service we can share data between any component even if they are not parent and children components

The third is use Redux, it’s the same service, we can share data between any components in an angular applycations. It is even easier to manage the data.

1. MVVM

Model View ViewModel

Model contans the struture of an entity

View is the visual layer of the application, it displays the data contaned inside the Model

ViewModel is a abtract layer of the application, it handles the logic of the application

1. Redux (NgRx)

Redux is a reactive state management . In simple term, it is a solutions to manage the data or state of the aplication

The library name is Redux in reactjs and it is NgRx in angular

1. Observables is used to handle asynchronous operations.

An onservable is not called until we subscribe to the observable

Emits multiple values

Can be cancelled by using the unsubrcribe() method

1. Dependency injection is an application dessign pattern which is implemented by angular