TELL ME ABOUT YOUR CURRENT PROJECT :

**Engage Employer:** the engage employer portal is for payer organizations, which will offer additional or better health plan management support.

The Employer provides easy-to-use self-service tools for employers who wish to perform the following transactions on behalf of members:

* New member enrollment
* Open membership enrollment
* Member messaging

The solution also allows self-management of billing processes. Bills can be evaluated and transactions performed quickly and efficiently at times convenient to the user. Employers and brokers can view the bill, make roster adjustments based on their current invoice, view the estimated impact of any adjustments and make a single payment or establish routine payments.

**Project Title: Engage Consumer**

EngageConsumer is a Third-party Administrators (TPAs), flexible enrollment and billing application that integrates with TriZetto Healthcare Products core administration systems that provides a digital environment to manage shopping, enrollment, administration. Individual employees and consumers can use this portal to shop, compare, review health insurance plans and enroll in benefit products.

This is a single page application based on Angular 5, HTML 5, SCSS, Bootstrap 4, YAML, TypeScript, Spring Boot, MongoDB.

**What is TPA?**

<https://economictimes.indiatimes.com/wealth/insure/what-is-a-third-party-administrator-in-health-insurance/articleshow/62483965.cms>

**Project Title: Engage Provider**

It is a portal to help physicians, hospitals and health systems simplify business processes and get insurance claims quickly. In this portal Provider organizations such as Organization of medical doctors, hospitals can raise claims against the services provided to the consumers and track the raised claims.

This is a single page application based on Angular 4, HTML 5, SCSS, Bootstrap 4, YAML, TypeScript, Spring Boot, MongoDB.

**How to improve performance?**

<https://www.keycdn.com/blog/front-end-optimization>

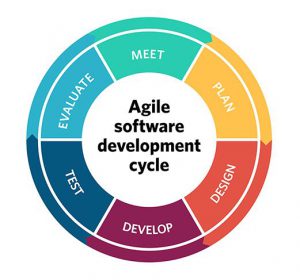
**GIT:**

Git is a versioning tool where we keep our source code. Git is used for coordinating work among programmers. Programmers collaborates and share their code in GIT during software development.

**What is agile ?**

Agile software development is an approach to software development under which requirements and solutions evolve through the collaborative effort of self-organizing and cross-functional teams and their customer(s)/end user(s).[1] It advocates adaptive planning, evolutionary development, empirical knowledge, and continual improvement, and it encourages rapid and flexible response to change

In simple words, Agile methodology is a practice that helps continuous iteration of development and testing in the SDLC(software development life cycle) process. Agile breaks the product development into smaller builds.



Scrum is an agile process that allows us to focus on delivering the business value in the shortest time. It rapidly and repeatedly inspects actual working software.

Sprint is a repeatable time frame of fixed days in which we should provide the delivery of highest possible value.

Scrum Sprint is a repeatable fixed time-box during which a deliverable("Done" product) of the highest possible value is created.

<https://searchsoftwarequality.techtarget.com/definition/Scrum-sprint>

<https://www.seguetech.com/waterfall-vs-agile-methodology/>

**Version control:**

In software engineering, version control is a class of systems responsible for managing changes to computer programs,

**What is Deveops?**

A compound of development (Dev) and operations (Ops), DevOps is the union of people, process, and technology to continually provide value to customers.

What does DevOps mean for teams? DevOps enables formerly siloed roles—development, IT operations, quality engineering, and security—to coordinate and collaborate to produce better, more reliable products.

* Explain CI/CD

<https://docs.microsoft.com/en-us/azure/architecture/solution-ideas/articles/azure-devops-continuous-integration-and-continuous-deployment-for-azure-web-apps>

A CI/CD pipeline is used to automate the process of continuous integration and continuous deployment. The pipeline facilitates the software delivery process via stages like Build, Test, Merge, and Deploy.

Steps:

* New pipeline
* It will provide us two options we have to choose either to go with yaml or classic editor
* Where is your code -> Azure repos
* After selecting the place where our code is it will ask us to select the repository
* After this It will provide us various configuring options like which kind of app it is

If app is built on asp.net core app, Nodejs or some other framework app.

* Then we need to create tasks like Restore, build, test, publish  
  and for each task we need to provide project path like ( “\*\*/\*.csproj”).

We need to choose a pool option.

Releases:

* Click new release pipeline
* Here we will get two options

Add artifact

Add stages

* When we add artifact -> We need to select the source type (i.e build)-> we need to provide the build name that we created earlier in pipeline as source
* Then In the stages we need to select App service deployment
* After this we should configure tasks in stages

It will automatically create deployment task but we need to configure by providing the subscription name , if subscription name is not present We should create service connection then only subscription will be available and we should provide app service name here.

New Agile question:

**TDD:** <https://www.browserstack.com/guide/what-is-test-driven-development>

**Kanban:** <https://www.digite.com/kanban/what-is-kanban/>



**All developers - Are you happy working in a test driven development (TDD) style? How much time do you typically spend writing tests?**

Yes, Almost every sprint I have to write test cases in Jasmine framework for Angular and make the code coverage more than 95%.

**In one sentence please can you describe what agile means to you?**

Agile is an approach in which we break our project into several features, and we do continuous iteration of Plan, Design, Development, Testing and Evaluate in software development life cycle.

**Which agile framework(s) / methodologies are you familiar with?**

Currently, I am using Azure boards. In my last organization I used Jira.

**Please can you name the Scrum ceremonies or the main practices of Kanban?**

Work in progress limits

Transparency

Visualization of work

Team capacity and Availablity

**What is it about Maersk that interested you enough to apply for a role?**

The job opening at your organization suits my skills and It is a very good opportunity to work in reputed organization like Maersk.

**How long have you worked in an agile environment / project or Scrum teams?**

3.5 years

**What do you see as the main benefit of working in an agile environment / framework / way?**

Continueous improvement in work done by us based on the result of Sprint Retrospective

Full transparency is there, Everyone will always be knowing How much work we have completed

We able to add some bussines value in every sprint

While planning only, we try to find if there are any risks invloved or any dependency on other team based on that we improvise our final plan.

**How should estimates be created and who is involved in an agile team?**

Whoever is involved in the story within the team should provide estimation in the same time independently.

All team member including Scrum master and Product owner

**Who should be responsible for quality in a agile delivery team?**

Our whole team should be responsible

**Can you explain what is meant by Definition of Done (DoD) and can you give any examples of things that might form part of your Definition of Done?**

DOD means when all defined acceptance criterias are met, For example: When we create our user stories that time we also define acceptance criterias and for the user story to be

completed the end result of work should meet the acceptance criteria.

**What is velocity and why it is used?**

Velocity is **a measure of the amount of work a Team can tackle during a single Sprint**

By looking at the amount of work your team completed in previous sprints, you should be able to estimate how much work they can do in future sprints.

**Agile Technical practices:**

For inly headings:

<https://hackerchick.com/just-do-it-quick-intro-to-agiles/>

**Pair programming** is an agile software development technique in which two programmers work together at one workstation. One, the driver, writes code while the other, the observer or navigator, reviews each line of code as it is typed in.