**Submitting: Ayoub nasrallah, Maya shema and Tal griman**

Hi,

“If you don’t know where you’re going, any road will get you there” – Lewis Carroll

1. Project

In order for us to start working on a new project, we first need to decide which project and what is the scope of it.

You can either select one of the following projects or decide on your own.

* Multiple users shopping list
  + Create and share shopping lists​
  + Add/update entries​
  + Mark an Item or strike-through ​
  + Copy a shopping list​
  + Add a picture of an Item​
  + Bonus: Add chat abilities​
* A shared paint program​
  + Multiple users draw simultaneously on the same canvas​
  + Create and share new canvas​
  + Copy canvas​
  + Bonus: Add chat abilities​
* An online game board for checkers or chess​
  + Let multiple people play together​
  + Create and share a game​
  + Handle move time limit timer​
  + Bonus: Add chat abilities​
  + Bonus: Add a computer player (learn about Minmax algorithm)​
* A Poll maker
  + Create and share Polls​
  + Present Poll results​

Once you’ve selected a project, you need to describe it in terms of requirements, that means what is the software going to do.

* Do not overcommit, be realistic
* Write down the requirements and email them to me, once they’re confirmed you may proceed to the next stage.
* Split requirements into “required” and “optional”

General guidelines

* You must have a database, it doesn’t matter which one as long as its suitable for your requirements
* The application must support multiple users, either by selecting a user in the ui or allowing username/password login
* You must have a service that handles the real work
* You must have a UI, one that most likely will call the service and/or other cloud services.
* Business logic must be tested, business logic is everything that is not a simple call this or that method.
* You must work with a partner
* Devops is optional but you must provide a deployment script.
  + Bonus points for CI
  + Extra bonus for CD

1. Choose a management tool

Since the project will need to be managed, you should practice on a reasonable tool, my recommendation is using either github or azure devops since they both can handle project management as well as source control.

Since managing a project on each tool is very different I recommend a tutorial:

<https://github.com/features/issues>

<https://www.youtube.com/watch?v=SbFKi6Hflc0>

Create a new project in your desired tool and insert 2-3 user stories for practice.

1. Read on AAA

<https://auth0.com/blog/id-token-access-token-what-is-the-difference/>

Answer the following questions:

1. What is an ID Token?

An ID token is an artifact that proves that **the user has been authenticated**. It was introduced by [OpenID Connect](https://openid.net/connect/) (OIDC), an open standard for authentication used by many identity providers such as Google, Facebook, and, of course, Auth0.

1. What is an access Token?

access Token is when a **client application** wants to access a **resource**, e.g., an API or anything else which is protected from unauthorized access. The other two elements in that diagram are the user, which is the **owner** of the resource, and the **authorization server**. In this scenario, the access token is the artifact that **allows the client application to access the user's resource**. It is issued by the authorization server after successfully authenticating the user and obtaining their consent.

1. What is JWT token?

JSON Web Token (JWT) is an open standard that defines a compact and self-contained way for securely transmitting information between parties as a JSON object. This information can be verified and trusted because it is digitally signed. JWTs can be signed using a secret (with the **HMAC** algorithm) or a public/private key pair using **RSA** or **ECDSA**

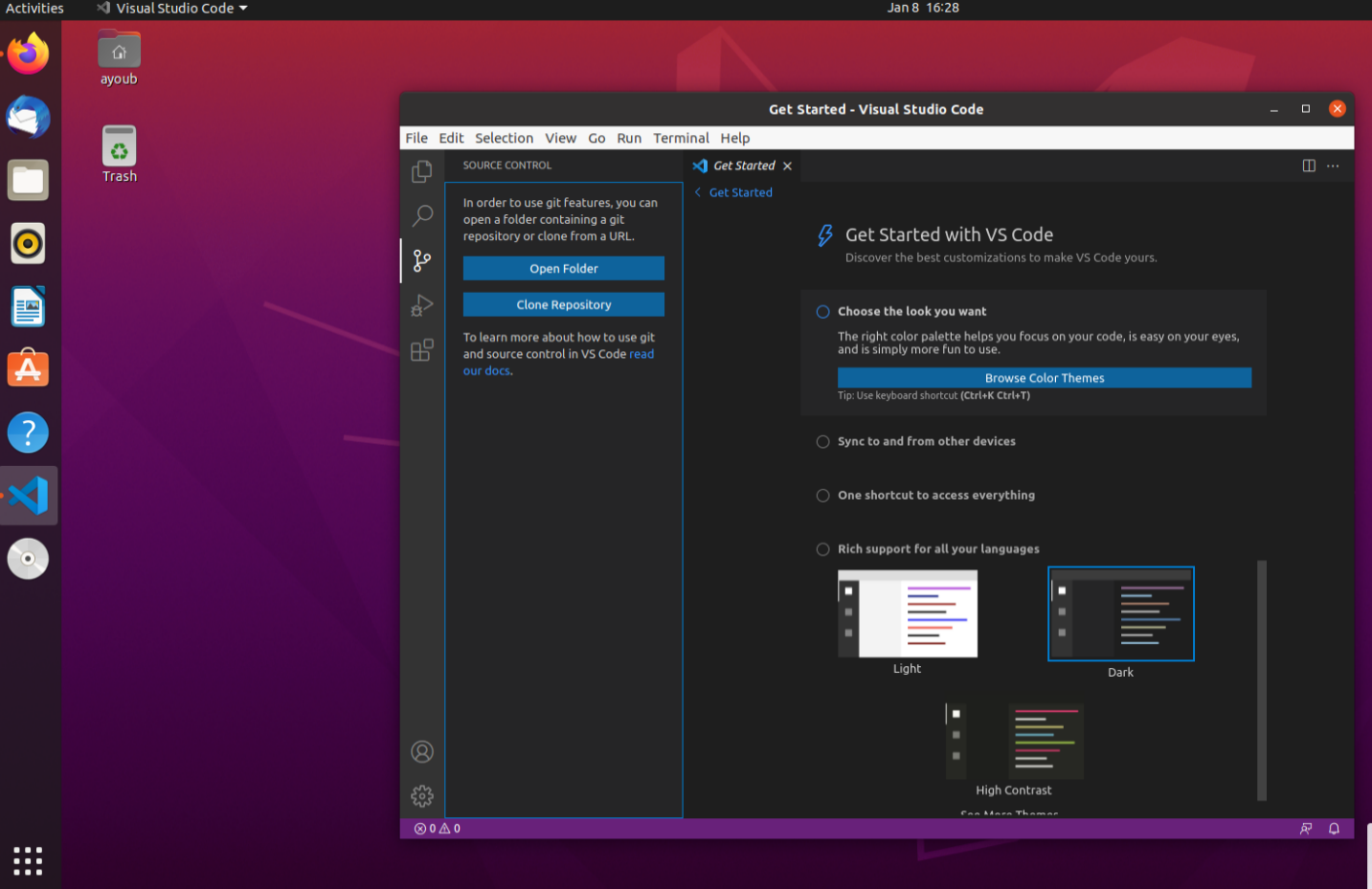
1. Which services are available for your programming languages?

4. Virtual Machines

* Install your choice of virtual machine, VirtualBox is my personal favorite, its free and fast.
* Install your favorite linux distribution, Ubuntu is my favorite.

Create a virtual machine with less than half of your computer’s resources:

* 1. if you have 4GB of RAM, create a VM with 2GB of RAM.
  2. Create a disk with at least 15GB space, you can use dynamic disks to prevent it from being allocated up-front
  3. Install GIT and VSCode
  4. Send me a screenshot of VSCode



5. Bonus: create a simple application that communicates

Please see reference material

* Create a remote calculator with client and a server program
* The client should send a simple math request, such as ADDing two numbers.
* The server should perform the calculation and return an answer
* The client should display the answer

Guidelines:

* You should use Web API with Swagger / OpenAPI
* It doesn’t matter which language you choose however, C# is probably going to be simpler than node js to implement the server with swagger.
* The client should be auto-generated from the Swagger documentation (the json file containing the API structure)
* Document your code and actions you performed - if you use git on each step it will be easier to track, create a readme with steps you performed.

References:

* If using C# swashbuckle will most likely be used - there are other options as well.

<https://docs.microsoft.com/en-us/aspnet/core/tutorials/getting-started-with-swashbuckle?view=aspnetcore-6.0&tabs=visual-studio>

* If using node js, you will need to read the following tutorials, they may help you get started.

<https://medium.com/bb-tutorials-and-thoughts/how-to-add-swagger-to-nodejs-rest-api-typescript-version-5a63953c993b>

<https://dev.to/po5i/how-to-generate-a-typescript-client-from-a-swagger-documented-api-14d8>

https://github.com/TalGriman/hw1-zionet