MEet and You

Brent Nishioka (Leader)
Gideon Essel
Joshua Ramos
Raymond Guevara
Vivian Dinh
Team Pentaskilled
October 6th, 2021

Table of Contents

Purpose 3

Programming Languages 4

Software Requirements
- IDE 5
- Browser 6
- Web Server 7
- Database Management System 8
- Front-End Languages 9
- Back-End Languages 10
- Framework
  - Frontend 11
  - Backend 12
- Version Control Repository 13
- Version Control 14
- Diagrams/Modeling 15
- Security 16

Purpose

The purpose of this documentation is to provide an overview in regards to the technical aspects related to our MEet and You web application. The bulk of this document covers the potential tools as well technologies that will be utilised. Our technologies will include the accompanying programming languages, the development environment, and the frameworks(frontend and backend).

Programming Languages

C#

- C# is a general-purpose language that is used across a variety of tasks and objectives. It is used in conjunction with the Window .NET framework and can be applied to an open source platform. Due to being object oriented it is easily scalable and easy to maintain.

JavaScript

- JavaScript will be useful in helping us define the behavior of our content. JavaScript has no other alternatives since it is a standard of the Internet and is supported by all the major browsers.

HTML

- HTML lays the foundation for the structuring of our content and it's basic layout. HTML does not have any other alternatives and is also supported by the major web browsers. As a result compared to other markup languages it has more support and the other languages end up compiling to HTML in the end.

CSS

- CSS is how the presentation of our markup language is, more specifically the style and presentation of our content. Much like HTML and JavaScript it is also an Internet standard and has more support compared to other stylesheet languages.

SQL

- SQL is how we can define the data in our database and manipulate that data when needed. Many popular websites utilise SQL databases in order to pull user-specific data as people navigate the website. It is an industry standard as it is very flexible and scalable.

IDE

Name	**Intellij Idea: Free Community Edition**	**Eclipse: Free Community License**	**Microsoft Visual Studio 2019: Free Community Edition**
Version	2021.2.2	4.21.0	16.11
Pros	Intellij has live templates, is easily customizable and is good at auto-refactoring which is a plus for testing.	Eclipse is open source with a variety of various add-ons and great environmental tools	Most utilised IDE for team collaboration. Has a diversity of extensions and is updated regularly.
Cons	Utilises more memory without much flexibility in the community edition.	Does not have a lot of noticeable updates. Not much language support aside from Java	Hardware intensive, may cause computers to run slower.

Microsoft Visual Studio will be our first option due to its various plugins and support of various programming languages. Utilising the Github plugin with MVS allows for better collaboration when it comes to coding efficiently. It also supports all the currently popular front end and back end languages.

Browser

Name	**Google Chrome(Windows)**	**Firefox(Windows)**	**Microsoft Edge(Windows)**
Version	94.0.4606.61	92.0.1	94.0.992.31
Pros	Has a lot of addons, plugins, development tools and extensions. It is the most used browser in the world	Adequate addons for development. It tends to have better performance than Chrome.	Full hardware acceleration for graphics, texts, and videos.
Cons	Resource intensive since it uses a large portion of memory in order to run.	Smaller user base with limited amount of plugins.	Large memory utility as well as a small extensions library.

Google Chrome will be the best option for our suggested browser since we can test it for a better core audience and it generally has a variety of plugins. The benefit of developer features offered by Chrome outpaces its competitors and also has a better debugging tool for coding before deployment.

Web Server

Name	**Apache HTTP Server: Free**	**Microsoft Internet Information Service (IIS): Free**	**NGINX**
Version	2.4.49	10.0.17763.1	1.20.0
Pros	It runs in most OS and is a very popular web server with a lot of documentation. Various modules can be utilised along with it	Microsoft IIS is a very organized management system. Due to it being supported by Microsoft, it receives updates frequently.	NGINX is scalable and it works well with other web servers. It is lightweight and is capable of serving a large number of users using an event-driven asynchronous architecture.
Cons	It has a high ram consumption and a greater security risk since it is much easier to modify compared to the other servers.	It has limited extensions and is lacking customizability. It also has problems being configured to run with Node js.	It has a small number of extensions and requires third party web servers which can lead to complications.

Our first option is Microsoft Internet Information Services since it has greater security compared to Apache. And with security being an aspect of our application, it would be best to select IIS since it has the best security compared to other web servers. Since IIS natively supports .NET that cinches our decision to go with it.

Database Management System

Name	**MySQL: Free Community License**	**Microsoft SQL Server: Free Community License**	**Netbeans**
Version	8.0.26	SQL Server 2019	12.5
Pros	Unlimited size for each database/table and is fully-manage SBaaS through Google or Azure	It is capped at 542,272 TB and allows us to store each data once without duplicates	It supports a large number of Java applications and is able to indicate edited code as well as files based on GIT commits.
Cons	It requires a third party in order to work with .NET. Data restoration also ends up being time consuming due to the need of multiple SQL statements	It supports only Linux and Windows platom and is also more costly than MySQL since it requires the purchase of licenses to use/run multiple databases.	It takes time to load and utilises a lot of memory. It also has outdated framework support.

Our first choice would be Microsoft SQL Server since it is the best ast managing larger volumes of data while remaining stable and fast. With it's frequent updates it remains an ideal option as our preferred database management system.

Front-End Languages

Name	**JavaScript**	**TypeScript**	**PHP**
Version	ECMAScript 2021	4.2.4	8.0.11
Pros	It allows for both client and the server side runtime. The client side can be fast as it can be run within the client-side browser. It is also ideal for dealing with JSON.	In terms of larger projects it will result in a more robust software and allows for better collaboration and static typing.	It's easier to maintain due to being loosely coupled with little repetition. It has extensive database support with SQL.
Cons	Its interpretation can differ depending on the specific browser. However it can be difficult to debug with more complex projects.	It requires compilation before providing an output and is not easily manipulated with a Content Management System.	It can be utilised with JSON and is more situational than other languages. However it does have slower execution.

Javascript is our go to language since it allows for the creation of highly responsive interfaces without having to wait for server reaction. It can also be used to load specific content a user needs without reloading the whole page.

Back-End Languages

Name	**C#**	**Java**	**Python**
Version	9.0	SE 17	3.9.7
Pros	It's object oriented programming and integrates well with Windows . It can also read and write files when the Silverlight version allows it.	It has excellent documentation to reference. Using multithreading we can switch efficiently between threads. It has a significant amount of libraries and APIs.	Cost efficient in terms of resources and takes much less time to build things compared to other languages. It has easy scalability and is dynamic enough to be used for the most simple to the most complex agile based programming.
Cons	Requires a specific IDE to code it with and also requires a plugin like Silverlight in order to run on a browser.	The code can become bloated and more complex in comparison to a language like Python.	It is not a useful choice for memory intensive tasks and has it's restrictions when it comes to database access.

C# would be our recommended language for backend due to how it has better error handling features and a higher performance. And it would also improve our ability to both check and find errors in our code.

Framework

- **Frontend**

Name	**React.js**	**Angular**	**Vue.js**
Version	17.0.2	12.2.0-rc.0	3.2.1
Pros	Compared to other frameworks, React has a simpler syntactical form as well as a nice UI. It is also flexible and highly responsive.	It constantly has newer features due to its frequent updates. It's detailed documentation helps in utilising and implementing the framework.	It can be upscaled with larger projects and is also lightweight as well as offering excellent performance.
Cons	React can often be bloated as a result of having too many options. It also lacks official documentation.	Contains complex syntax inherited from the first generation of Angular and comes with migration issues.	Due to being lightweight it lacks a few resources and it might be harder to upscale your projects as a result.

All of these make for good options since they would all work well with our recommendations for backend frameworks. However we decide to go with React since its purpose iS very straightforward and because of its simplicity would be a great option for us to utilise.

- **Backend**

Name	**ASP .NET Framework**	**Express.js**	**Flask**
Version	4.8	4.17.1	2.0.1
Pros	Integrates naturally with a SQL server. It is also a framework with huge extensibility and third-party components. It is a mainstream standard that has many developers actively working on it. This means it comes with a lot of support and documentation.	It supports many third party plugins and happens to be a very simple and fast framework with a large library. It happens to be one of the first frameworks that worked with node.js and maintained its popularity.	It has a very simple and clean implementation and is lightweight as well. It is a flexible framework that upscales with larger products.
Cons	Scalability of your projects becomes costly as they become larger.	There is no default method of arranging things and as a result it becomes harder to manage in the long term.	Has a smaller library in comparison to the others and has nothing in the form of a database.

ASP .NET Framework will be our first pick since it offers one of the highest web performances of our web application frameworks. It has fast performance and provides greater security functions.

Version Control Repository

Name	**Github**	**Sourceforge**	**Bitbucket**
Version	N/A	N/A	N/A
Pros	A free and open source repository that allows collaboration across the board. There is a great number of documentation in order to better utilise its features. Offers unlimited private repositories.	Allows users to make contributions to code within a repository and is effective in allowing groups to contribute together.	It offers free private repositories and also allows users to import existing repositories.
Cons	Private repositories do not necessarily mean that data is private. Its project management could be more robust.	Its free private repositories are limited to a group of five or less.	Its download installer attempts to install third-party software.

By recommendation of the client we plan on using Github. It will help in keeping track of modifications of code as well as the progress of said code. By utilising Github, our workflow can be better improved as various aspects can be safely implemented in branches of the code.

Version Control

Name	**Apache Subversion**	**Git**	**AWS CodeCommit**
Version	1.14.1	2.33.0	N/A
Pros	It is also another free repository that allows the creation of branches within code files.	It is an open source repository and allows the creation of branches in any existing code files.	As long as less than 5 people utilise it, then it is free to utilise it's capabilities such as creating branches in code files.
Cons	Merging code from one branch to another is difficult.	Difficult to integrate different repositories to one another.	Much smaller user base compared to its competitors. Users pay if the group limit is exceeded.

Git would be the best software to use in terms of VCS since it integrated well with GitHub. Git can be utilised to push updates to Github in a more efficient manner compared to the other alternatives.

Diagrams/Modeling

Name	**LucidCharts**	**Draw.io**	**Figma**
Version	N/A	N/A	N/A
Pros	It contains an online repository for both diagrams and for collaboration. It is also able to import and export different file types.	Has great accessibility to the specific features. The diagrams also revert with their pages	Allows for a simultaneous collaboration . Files can be accessed anywhere and anytime.
Cons	Certain features are locked behind a paywall and there is a lack of customizability when it comes to it's options.	It is only available as a web application. It is not well optimized in terms of team collaboration.	Requires online connections at all times to utilise collaboration features as well as some of its other features.

We recommend the utilisation of either LucidCharts or Figma for modeling and diagrams as they are both free with great accessibility and features. They both provide the capability to export their products in multiple formats as well as an inbuilt capability for collaboration.

Security

Transfer Protocol

Name	**HTTPS**	**HTTP/2**	**HTTP**
Version	N/A	N/A	N/A
Pros	Automatically understood by website users as being secure. Has the capability to encrypt the data being transferred between two systems and is an easy transfer from the HTTP system.	Performs the best in terms of header compression and has a faster load time compared to its competitors.	Utilised as the basic means for two systems to communicate with one another
Cons	An SSL certificate is required.	Not as well as established as the others or as supported	Data being transferred is not encrypted and can be intercepted.

Our recommendation would be for HTTPS since it is a standard to utilise HTTPS in order to keep communication between two systems secure.