SUPER HEROES IN TRAINING

My Academic Pyramid

Technical Specifications Document

CECS 491A Sec 05

October 09, 2018

Team Leader: Krystal Leon, 013986607
Arturo Peña Contreras, 010914811
Luis Julian, 007472593
Hyunwoo Kim, 014392909
Victor Kim, 012016990
Trong Nguyen, 016208983

Table of Contents

Abstract

This document contains information on the technology we will use to create our web application. The document is organized by technology. Each technology has our preferred option along with two other competitors. Information on technologies is organized within tables listing the pros and cons of each along with our preferred option and reasons why we believe it is the best choice.

Browser

	Stable Google Chrome (latest)	**Mozilla FireFox**	**Microsoft Edge**
Pros	-Great Development and debugging tools -Able to map web resources such as JS and CSS -Provides many extensions - Provides better isolation from crashes by spinning up separate processes for different tabs -Highest compatibility with HTML5	-The pages are loaded fast -Provides the “Debugger” feature in Developers Tools -Extensibility with more privacy	-Provides the “Debugger” feature in Developers Tools -Highest JetStream

Cons	-The pages are loaded slow -Large memory use -not privacy friendly: all the user data will be saved -Doesn't provide the "Debugger" feature in Developers Tools	-uses more RAM	-Large memory use -Small number of extensions library
Determination	The client requires us to use Google Chrome. We are able to check the results of each feature of the web application visually with many extensions that Google Chrome provides. Additionally, it provides great debugging tools that will decrease the time and increase the efficiency for implementation. We will be using latest version of Google Chrome.		

IDE

	Visual Studio Community 15.8.2	**Eclipse Neo 4.6**	**Netbeans 8.2**
Pros	- Managing source control branches, work items, etc, fully integrated with Azure DevOps -SQL Management and Cloud resource management -Supports multiple languages like C#, Javascript and Typescript	- Good syntax highlighting. - Allows for plugins to change its behavior by a lot - Handles password management for servers well, as they are encrypted and saved in a format acceptable to most security standards	- Supports a large number of Java applications such as J2SE and EJB - Indicates edited code and files based on GIT commits

Cons	-Complex and has a lot of features. -Resource heavy application	- The actual management of plugins is not as powerful to implement as Visual Studio. UI feels outdated. - Importing project files is not friendly. - File management on a server is tedious	- Outdated framework support - Takes a lot of time to load and consumes a lot of memory
Determination	Visual Studio because it allows us to do both backend and frontend sides of our application. It supports SQL Management and allows us to configure Azure resources. We will be using version 15.8.2.		

JavaScript Front End FrameWork

	ReactJS 16.5	**AngularJS 1.7.4**	**Ember 3.4.1**
Pros	- Has a good tool chain - Rendering is on service side , Initial page would load faster - Great with code reuse - Supports multiple platforms	-Learning curve is high - Rendering is on client side. Support rich interactions, optimized for web application. -Two – way data bind -Full-Fledged framework -Suitable for small to medium projects - Great with code reuse (Angular libraries)	-Learning curve is high, friendly docs and API -Client-side rendering (Same as AngularJS) -Two-way data binding - Full-Fledged framework ((Same as AngularJS) -Suitable for big project (No dirty checking) -Faster boot times and inherent stability

Cons	-Learning curve is low -Not a framework , rather a library -Architecture keep changing -Rendering is on server side. Our web app have rich interactions, it may be slow	- Steep learning curve and complexity. -Dirty checking could make app load slower -Need to emphasize on security	-Lack of reuse of components -Steep learning curve -Not suitable for small project
Determination	AngularJS because it has rich documentation and rendering on client side which support many interactions. Our project is small , then dirty checking is not a big problem		

Server-Side Programming

	.NET Framework (C#) 4.7.2	**Django (Python)**	**Laravel (PHP)**
Pros	-Tight integration with visual studio and azure -Great community -Good documentation -Good 3rd party libraries -Fast and secure	-Fast and secure -Open source , great community -Great library and good documentation	-Fast and great for dynamic -Reusable code -Built in command line tool “Artisan” to create code skeleton
Cons	-Code run with this framework run slower than the native code -Vendor lock-in may be a drawback ; .NET still depend mostly on Microsoft’s decision.	-Autoreload restarts whole server -Too Monolithic -Everything is based on ORM , can’t turn if off even project does not access to database - Use regex to specify the URL , can	- New framework, less built-in support compared to django

		create complicated syntax -A process can only handle a single request at a time d	
Determination	.NET Framework because of its tight integration with Azure and the high amount of documentation for Web Application development.		

	C# v7.3	**Visual Basic 2017**
Pros	-Three members are learning it for CECS 475. -Native access to .NET Framework. -It’s recommended by the professor. -Active community with a lot of third party documentation. -Has more powerful features. -Primary .NET language.	-Native access to .NET Framework -Good readability -Friendly to beginners
Cons	-It has a relatively high learning curve and it’s constantly being updated which may lead to some of our code becoming out of date. -Not as readable	-Small community that isn’t very active. -No one in team is familiar with it. -Not as many features.
Determination	C# because of the amount of documentation it provides and our client recommends it for web application development.	

Database

	Microsoft SQL Server 13.0	MySQL 8.0	PostgreSQL 10.5
Pros	- Very fast and stable. - Works very well with other Microsoft products. - Allows us to choose from several editions of SQL Server.	- Offers a lot of functionality even for a free database engine. - Variety of user interfaces that can be implemented.	- Scalable and can handle terabytes of data. - Variety of predefined functions. - A number of interfaces are available.
Cons	- Can use up a lot of resources.	- Poor query optimization.	-Poor documentation.
Determination	MSSQL because it comes with the features that our web application will need. It will work well with our other Microsoft products. MSSQL works natively with .NET applications so this one would work better over the others. We will be using MSSQL Server 2016 version 13.0.2216.0. We will use the 2016 version over the 2014 version because it has more functionalities and has been improved, and over the 2017 version because it has been around longer and there is more documentation for it. The client recommended we use this MSSQL Server 2016.		

SQL Management

	SQL Server Management Studio 2017	DBeaver 5.2.2	Navicat Monitor 1.6.1
Pros	-Encryption at rest, transit, and in use by client -Supports Linux, Docker, Redhat, Suse, Ubuntu, and Windows -High-availability, disaster recovery solutions	-Free and open source -Supports any database that has JDBC Driver and is multiplatform -works with large variety of	-Real-time performance monitoring with colorful reports -Analyzes queries allowing to improve performance and efficiency of server -Supports Windows, Mac, Linux, and Docker.

		database managers	
Cons	-Doesn't support Mac OS. -Fully upgraded usage requires you to purchase the software.	-Doesn't support Ubuntu, Docker, and Redhat -bugs and errors pop up on certain scenarios.	-Requires to purchase as it only has a trial run. The price to use all the databases is too high.
Determination	SQL Server Management Studio is our choice because it supports more OS compared to other SQL Managements, encrypts data in any situation, and it constantly evolves, upgrading their disaster recovery solutions and long with their high-availability. Navicat Monitor was too expensive to use all of its features and DBeaver had many bugs and issues in its program. That is why SQL Server management studio is the program we will use for our project.		

Microsoft SQL Server 2016 Edition

	Developer	**Express**	**Standard**
Pros	-Free if you join the "Dev Essentials" program. -Has the same functions as the licensed Enterprise edition -Maximum relational database size : 524 PB -Basic security (Row-level security, data masking, basic auditing, separation of duties)	-Free -Entry-level database ideal for learning -Can be used in production environment	-Delivers basic data management and business intelligence database for small organizations. -Maximum relational database size : 524 PB -Basic security (Row-level security, data masking, basic auditing, separation of duties)

	-Advanced security (Transparent Data Encryption, Always Encrypted)		
Cons	-Uses a lot of disk space per instance. -Use rights are limited to test and development purposes only -Usage data is sent to Microsoft (performance, errors, feature use, IP addresses, device identifiers, and more)	-Extremely limited feature set -Maximum relational database size : 10 GB -No basic security (Row-level security, data masking, basic auditing, separation of duties) -No advance security (Transparent Data Encryption, Always Encrypted)	-$931 license -No advanced security (Transparent Data Encryption, Always Encrypted)
Determination	The client required us to use SQL Server 2016 Developer Edition. Developer edition meets our needs the best. As students our budget is limited and our main use is for development purposes. Because our database is aiming to support all students from multiple campuses, a greater sized database is beneficial to us.		

Caching System

	Redis 4.0.11	**Memcached 1.5.10**	**Ehcache 3.6**
Pros	-Open Source -Free -Advanced data structures (strings, lists, sets, sorted sets, hashes, bit arrays, hyperloglogs) -Data structures can store any aspect of the data natively, which	-Open Source -Free -Memory management is more efficient when caching small and static data: consumes comparatively less memory resources for metadata	-Open Source (Apache 2.0 license) -Free -Faster than Redis -Small footprint: keeps apps light -Offers multiple eviction policies and expiration policies.

	reduces serialization overhead -Can scale horizontally via clustering without loss of data. -Offers six different eviction policies to delete old data from memory -Key name limit: 512 MB -Supported Languages: C, C++, C#, Java, JavaScript(Node.js), PHP, and morel. -Azure supports it.	-multithreaded: easily scale up by giving it more computational resources -Efficient for caching small static data	
Cons	-Clustering is more complex to set up and operate	-Only supports strings -Loss of cached data when scaling up. -Uses only Least Recently Used algorithm to delete old data from memory -Key name limit: 250 bytes	-Supported Programming Language: Java
Determination	Redis because it has a wider range of capabilities than Memcached. It also has a wider range of supported languages than Ehcache does. We will primarily be using C# and Azure, so Redis is a better fit for our project. Redis 4.0.11 is the latest stable version.		

Server

	IIS 10	**Apache Server 2.4.35**	**Nginx 1.15.4**
Pros	-Free -Security is good -Good in-depth diagnostic tool -Strong support for Microsoft products .NET (framework) and ASPX -Integrates with Microsoft services	-Free -Support multiple platform -Open source -Quick support from community	-Free - High-performance http server and scale -Easy to configure -Open source

	-Reliable		
Cons	-Only run on windows -Not open source , then not able to customize much as open source web servers	**-**A process-based server .Each connection require a thread, can cause significant overhead	-There are not as many as modules as Apache -May have less community support than Apache
Determination	IIS because our team members use windows and Microsoft services. In order to optimize the performance , we use IIS for our local server. Then we will deploy our app using Microsoft Azure (see Web Server Host comparison). In addition, client would like to use that.		

Hardware

Client Side Hardware requirement	
Research method: In order to determine exact hardware requirement for our web application, we need to know traffic to our application such as number of visitor per day/month, maximum number of simultaneous visitors , size and complexity of our features, size of the database etc. However some of these statistic has not been known yet. The following hardware requirement is from other softwares which their complexity is more than us. In addition , our application is still in development and going to a demo stage, then this hardware capacity is sufficient We may scale the hardware requirement until we have actual statistics of traffic to our application after a public released	
Processor	2.0 Ghz and 4 cores of faster
RAM	4 GB or more

Messaging (Library)

	Discord	**LibraryH3lp**	**Slack**

Pros	-Allows users to text, video chat, and share screens -Capable of sharing images, videos, and links -Minimal usage of CPU compared to other products -Client-server architecture, keeping IP secure. -image and links are embedded when posted in chat. -TitanEmbeds allows Discord to appear in website.	-Allows to add chat services to the webpage. It can also be configurable and integrated with Facebook. -Highly Customizable -Can connect to phone, allowing users to send/receive messages outside of the website.	-channels can be built, where it can be divided into projects and teams -Connect tools such as Dropbox, Github, and Google Drive -Multiple security standards to protect data when signing in, transmitting data, and during rest
Cons	-Some operating systems cannot run the program. -Random people can join if they know the permanent server code.	-The product is free for 90 days, requiring to pay for full service. -Cost rises if more users and text messaging costs per number	-multiple channels can be hard to manage. -storage is short, as free tier only allows 10,000 messages. History gets deleted right after.
Determination	Discord meets the requirements needed for students to help each other as it has screen sharing and it uses minimal usage of CPU. TitanEmbeds will also allow Discord to be implemented into our website instead of creating our own chat system.		

Messaging (Personal)

	Discord	**Facebook Messenger**	**Hangouts**

Pros	-Allows users to text, video chat, and share screens -Capable of sharing images, videos, and links -Mobile app and online available.	-Allows users to text and video chat, with no fee -Mobile app and online available. -Location, payments, and group chats available	-150 participants in group chats -Available on mobile and online, with conversations being able to sync across devices. -Video conference can be seen if you have the URL for the meeting, requiring no google account.
Cons	-Some operating systems cannot run the program. -Random people can join if they know the permanent server code.	-Privacy is an issue as they can record audio without permission or call phone numbers without intervention.	-Short Message Service rates still apply -Less organized as it focuses more on time-sensitive communications.
Determination	Discord has a safer privacy system compared to Messenger and its services are not charged compared to Google Hangouts. It's interface allows us to read messages and know which channels have important messages based on the notifications system. Screen sharing also helps us when we want to see the same document for our projects.		

Automated Testing

	Fiddler 5.0.20182	**Postman 6.3.0**	**Assertible**

Pros	-Free -Supported Browsers: Mozilla Firefox, Chrome, Microsoft Edge, Safari, Opera, Konqueror -Supported Systems: Any Windows-based PC and mobile devices -Supported Platforms and Languages: Ruby, Python, Java, PHP, Microsoft .NET Framework -logging and capturing services for HTTP and HTTPS based traffic. -offers decryption of HTTPS traffic. -Isolate performance bottlenecks with rules feature. -Compose your own HTTP requests to run.	-Free -Supported Browsers: Chrome -Supported Systems: Mac, Windows, Linux -Write and run tests for each request using JavaScript -Tests are run in sandboxed environment (separate from the execution environment) -Free version aimed toward individuals and small projects. -Unlimited Postman collections, variables, environments, & collection runs -Has Run, Test, Document, and Monitoring functions -Allows users to easily share experience and knowledge with other team members, as it allows you to pack all requests and expected answers and send them to colleagues	-Free -Open Source -Provides automatic testing of the API at every stage of the process of integration and delivery of software -Provides support for current API tests after application deployment and integrates them with familiar tools such as GitHub. -Specifically designed for web service test automation. -Codeless web service testing -Free automated testing
Cons	-Supported Systems: Mac (Beta), or Linux system (Beta)	-All testing must be done using custom Javascript code snippets -Automated testing features require Postman Pro account	-Limited web services and tests for free plan.
Determination	Fiddler meets our needs best because it is free, it provides features that we need, and it supports the other technologies we are using such as Chrome, Microsoft .NET Framework, and Windows PC.		

UI Design Software

	Adobe XD 12.0.12.10	**Invision (Newest)**	**Moqups (Newest)**
Pros	-Free -Supported Platforms: Windows, Mac, iOS, Android -Desktop App -Clean and intuitive interface -Drag and Drop design method -Prototype Live Preview: no extra plugins or services -Repeat grid: repeating objects -Opens fast and files load quickly -Design specs feature that makes it easy to pass work on to a developer -Shared link for a prototype can be iteratively updated -Transition animations between screens -2GB cloud storage	-Free -Sharing and commenting system for feedback -Web display prototype -Advanced prototyping features such as hotspot templates (to link screens together), on hover hotspots and fixed headers/footers -History mode showing all of the past iterations of a mockup -Colour coded comments which allows better distinction -Screen statuses showing what stage of the process the mockup has reached (needs review, approved etc.)	-Free -Extensive list of keyboard shortcuts -Prototype interactions -Export to Google Drive -comprehensive library of widgets, smart-shapes, and integrated stencil kits for web design -Built-in library with thousands of Icon Sets - Hundreds of font choices with integrated Google Fonts.
Cons	-1 active shared prototype -Lack of layer panel: difficult to reorganize existing objects	-No animation or transitions -Free for only 1 prototype -Online application which may be	-Free package: 1 active project, 300 objects, and 5MB image storage -Online Application

	-Forces the user to link multiple screens to show simple functionality -Linking elements on the page for prototyping can get messy since you have to drag every link from one element to a page	unreliable based on your internet connection -The user interface is not intuitive and has a learning curve	
Determination	Adobe XD because it is a desktop application and does not depend on the Internet. It offers several features focused on not just UI design but prototyping as well. There is no real-time editing between team members, but there is a shared link between us that can be iteratively updated.		

Version Control System

	Git 2.19.1	**SVN 1.10.2**
Pros	-Most popular version control system which means there’s a lot integration with third party dev tools. -Decentralized systems which prevents any single point of failure; Everyone has a copy of the full repository. -Better branch support. -All main features available offline -Github natively supports Git.	-Simpler for beginners to learn due to less features. -Good UI

Cons	-Complex due to many commands. -Constant pushes to server are required by members to prevent any major conflicts.	-Branching not as efficient. -Version control features lost while offline (although there are third party tools to offset its impact).
Determination	Git v2.19.1 because of its decentralized nature which means we can always work on project and it allows the client to track our work. We use this version because it’s most up to date.	

Project Code Hosting Platform

	GitHub	**Bitbucket**	**Gitlab**
Pros	-Email notifications -Graphs: pulse, contributors, commits, code frequency, members of it. -Largest repository host with more than 38+ million projects -has its own Wiki and issue tracking system -API integration for easy integration of 3rd-party tools, and integration with a large number of other tools and platforms	-Free unlimited number private repositories -provide Code search -Snippets that allow developers to share code segments or files -unlimited number of private repositories	-able to set the role for each people and modify the permission -Community Edition is free and open sourced -Git import -Commit graph and reporting tools

Cons	-private one is paid and has size limitations	-not open source. -limited products for integration with 3rd-party tools.	-Can't pull requests with code review and comments -relatively slow interface -Frequent technical problems with repositories
Determination	Our client required us to use GitHub. All of them provide the public repositories for free. However, GitHub provides API Integration for easy integration with 3rd-party tools such as AWS, Windows Azure, and Google Cloud.		

Cloud Service for Deployment

	AWS	**Microsoft Azure**	**Google Cloud**
Pros	- Has been perfect cloud computing since 2006 -More compute capacity -Multiple data center -AWS cloud ecosystem and products is a plus	-Cost effective -Flexible billing -Strong Platform-as-a-Service -Reliability and Scalability -Strong presence -.Net better compatibility -Have better tutorials , user guides and troubleshooting documents	- Flexible pricing - Great reputation in open-source community
Cons	-Catalog of features can be overwhelming and difficult to navigate for some users	-Not as much as functions compared to AWS -Setup to support Windows	-Lacks functions, still catching up with Microsoft and AWS -Only capable running small projects/

	-PaaS is not strong as Azure		
Determination	Microsoft Azure because it has enough features that built for our web app. Especially, we are still learning so good documentation and learning curve is a plus.		

Calendar Service API

	Google API v3	**Outlook Calendar API v2.0**
Pros	-Allows access to a tutor’s Google Calendar and add tutor sessions to it. -Great documentation in a variety of languages. -Supports .NET Framework and Node.JS -Google calendar is widely used so integrating to app will make it more competitive with rivals -Libraries that allow for rapid development of Calendar functionality	-Can be used to add sessions to Outlook calendar associated to tutor’s school email address if its a Microsoft email. -Supports .NET Framework and Node.JS
Cons	-May require another email address of the user: their Google email.	-Some schools may not provide Microsoft email. -Very technical documentation that’s not beginner friendly. -Outlook Calendar not as popular as Google Calendar
Determination	Google API v3 was chosen because of the official documentation written, the active community, and the fact that Google Calendar is widely used. Version 3 was chosen because that is the one Google is currently supporting. Version 2 was discontinued.	

Calendar Service API 2

	FullCalendar	tui.calendar	DHTMLX Scheduler
Pros	-Built in event creation features such as create and editing events. -Good documentation. -Most popular which means a lot of third party tutorials -Built-in plugin to connect to Google Calendar API	-Free -Supports rapid development of calendar	-Events and calendar are customizable which is needed to create sessions listing all registered students. -Good documentation and an active forum.
Cons	-No built in support to allow multiple users to be associated to an event.	-Little documentation -No community that provides help.	-Need to pay to use full features or to include in web application that's in production. -Uses AJAX instead of Fetch API
Determination	We will use FullCalendar as the documentation and community will allow us, with limited JavaScript, skills to implement a simple Calendar. It's Google Calendar plugin will also allow for less bug when connecting application to user's Google calendar.		

Usage Analysis Dashboard Graphing

	Amplitude (Newest)	Mix Panel (Newest)	Google Analytic (Newest)
Pros	-Have free plan -Focus mostly tracking on action taken by users -User friendly	-Have free plan -Tracking visitor action -Able to track specific users	-Have Free Plan -Using JavaScript code snippets to track visitor information and traffic -Support funnel reporting -User friendly

	- More visualization of data compared to Mix Panel -Support A|B testing	-Offer event tracking in real time -Support A|B testing	-More features in free plan
Cons	-Super expensive one pass the free plan -Many important features are limited in free plan	-Require some knowledge about flows and event to begin with tracking -Only 60 days data history in free plan	-Data processing time is longer than Mix Panel - Hard to track web site events
Determination	Regarding to requirements of Usage Analysis Dashboard,Google Analytic is chosen because it provides sufficient resources and features to the project. Google Analytic is also generous with features in free plan		

FrameWork of CSS

	BootStrap 4.1.3	**Foundation 6**	**Semantic UI 2.4**
Pros	-Highest popularity -Many free and professional templates, themes and plugins - Many open source projects using Bootstraps -Consistent UI -Good documentation	-out of the box CSS design provides beautiful flat design that doesn’t require heavy customization. -Flexible grids by providing useful features and classes	-Provide the consistency -Able to load only the components that you need -Simplified debugging with performance logging -Support for third-party applications such as Angular or Wordpress. -Good documentation

Cons	-Restricted customization due to lots of style overriding or rewriting files -Large size of the file	-Less community support due to popularity -Lack of QA sites and forums for fixing problems	-Less community support due to popularity -Large package size -Lack of number of classes
Determination	Our client doesn't want us to use BootStrap due to large file size. Semantic UI is chosen, because it officially third-party applications such as Angular or Wordpress. Therefore, it will decrease the development time and enhance the quality of the code.		

FrameWork of CSS 2

	Bulma 0.7.1	**UIKit 3.0.0-rc.17**	**Pure 1.0**
Pros	-Free, open source on GitHub -Responsive grid -No JavaScript required, so it integrates in any JS environment -integrated Flexbox which allows users to build fancy designs.	-Good animation and user experience -Lightweight -No restriction on customization -Modular, so developers are able to choose the components to add to their stylesheet.	-Lightweight and provide responsive grid - No restriction on customization
Cons	-Runs slow on Internet Explorer -It's still in the development phase for version 1.0.	-Lack of community support due to popularity -UIKit Pro is not free, so some of features are not available in free version.	-Limited number of designs and templates -Lack of community support due to popularity
Determination	We chose Bulma, because it's free and modular. It provides the features and classes that we need without requiring a payment. We will be using latest version of Bulma.		

Command-Line Shell (For Git)

	Windows Powershell 5.1	**Git Bash 2.17**	**Windows Command Prompt (Windows 10)**
Pros	-Object Oriented Output -.Net Enabled: can access any .NET type available in C# -Can debug Powershell jobs -Autocomplete feature -Can develop using classes -Create your own commands and scripts using the C# programming language. -Advanced Features: like remote execution of tasks, background tasks, task automation, command piping, and more. -Can do everything you can in Command Prompt, and more.	-Cross Platform: Can be run in Linux, Unix, MAC OS, Microsoft Windows, and more. -Easy to bash from current directory.	-More simple than Powershell, so more sufficient for lighter use.
Cons	-Not Cross Platform: Windows Only -Steep learning curve.	-Text Based Output	-Not Cross Platform: Windows only -Text Based Output
Determination	Git Bash because even though Powershell has more advanced features and capabilities, Git Bash is more sufficient for the few git commands we intend to use. It is easy to bash from the current directory by		

	right-clicking on the directory, which is why we chose it over the Command Prompt.

Encryption Standard

	.NET .Cryptography	**BouncyCastle**	**SecurityDriven.Inferno**
Pros	- Provides cryptographic services, including secure encoding and decoding of data, as well as many other operations, such as hashing, random number generation, and message authentication. - Encryption algorithm classes autogenerates keys for ease of use and default properties are as safe and secure as possible.	- Lightweight cryptography API. - Free, open source.	- Free, open source (MIT license). -Developer-friendly, misuse-resistant API. - Does not re-implement crypto primitives.
Cons	- Large number of options can lead to possible misuse.	- Not well documented. - Not maintained frequently.	- Third party cryptography library.
Determination	We recommend using the cryptography library already available in .NET (System.Security.Cryptography). We do not need to use a third party product or create our own crypto API.		