

VIA Information Services Application Coding Requirements

Frontend:

Html/CSS

- HTML5 standards
- https://www.w3schools.com/html/html5_syntax.asp
- CSS3 Guidelines
- <https://www.w3schools.com/css/default.asp>
- Use SASS or LESS if possible

Javascript

- Languages
- JQuery (1st choice)
- Use when you need a secure front end on top of MVC
- Use for data heavy applications
- Angular/React/VueJS (2nd choice)
- Use for non-secure applications only
- Use only for small projects with small data sets

Styling

- Bootstrap (Latest)
- Prefer use of MDBootstrap

Icons

- Maintain consistency in application
- Font Awesome (latest)
- Icons8 (latest)

Implementation

- Break JS files apart from HTML pages if possible
- Write once, use everywhere concept
- Minimize all JS and CSS files
- Minimize HTML files if possible
- All inputs require unobtrusive validation
- MVC
- Use partial views as much as possible for view components
- Business logic **does not** reside in JS

Logging

- Only use logging for debugging
- Writing to the console should be deactivated for production use

Backend:

Structures

- a. MVC (preferred with concept of moving to micro-services)
- b. Microservices
- c. Approved Languages
 - a) .Net Standard (Latest)
 - b) .Net Core (Latest) (preferred)
 - c) Golang (data processing)
 - d) Python (data processing)
 - e) Don't hardcode sql queries into code unless necessary
- d. Use Linq and Entity Framework instead
- e. Parse all inputs for sql injection
- f. Prevent all CORS requests
- g. Force HTTPS Redirects
- h. Require authorization/Authentication for all sensitive data requests
 - a) Use OAuth 2.0 against VIA AD
 - b) Use Dependency Injection for all reusable patterns
 - c) Business Logic

1.1.1.1.1.1. Should reside in Controllers

1.1.1.1.1.1.1. May reside in Model but preference is in Controller

Logging:

1. Logs should be written to central logging structure
2. Logging should be async
3. Email notifications should be sent based on threshold (This should be handled outside the application by a monitoring program such as DataDog)
 - i. Never send emails from the application itself unless part of the application function

Database:

Allowed DB's

Relational

- i. MSSQL (preferred)

NoSQL

- i. MongoDB (preferred for non-transient data)
- ii. Redis (preferred for transient data)

Table Structure:

Naming Conventions

- Plural
- Do not put table or any form/contraction of tbl in name
- Avoid using numbers
- Avoid dashes unless necessary
- Capitalcase
- Should reside in dbo namespace

Do not create backup tables in production server

- Backups should reside on a different server

Use Foreign Key constraints and mapping tables to avoid duplicating data

Always use Primary key (Identity) column indexing if possible

- If not possible, create multi-column constraint
- All tables should have a primary key, yes even mapping tables

Business logic ***should not*** reside in database

Leave SQL defaults as the default values

1. Server time should be set to UTC time
2. Log all user actions

Use Functions for modifying columns when necessary for BI Reporting, etc.

Reporting:

1. Tools
 - a) Power Bi(preferred)
 - b) Tableau
2. Use Power queries and R programming
3. Data modeling should be in DB
4. Reports should be automated with DB procedures and jobs

Documentation:

Please provide some form of basic coding documentation.