**MSU Denver - Senior Technical Project**

FEV Tutor - Exploratory Consulting

**January 22, 2021**

# Executive Summary

Our team at MSU Denver would like to contribute to FEV Tutor’s platform as our senior technical project. We will be exploring the platform and identifying places where adjustments could be made. Areas of exploration include database structure, query optimization and breaking the main project into services which can scale with FEV Tutor’s growing client base.

# Problem/Opportunity

FEV Tutor’s platform has been growing and is reaching a point where future growth will become a challenge. The front-end website has most features dependent upon each other which is slowing down the application and makes development of new features cumbersome. When certain features break, the entire platform may go down impacting all clients. By decoupling these features downtime will be minimized and fewer clients affected. Database querying is expensive and needs to be optimized to best fit the cloud structure FEV Tutor is moving toward.

# 

# 

# 

# 

# Goals

1. **Identify Bottlenecks:** Analyze existing systems and identify where the most problems are occuring
2. **Optimize Database/ Cloud Infrastructure:** Look at existing queries and data structures to discover where changes can be made to make processes more inexpensive and efficient
3. **Modularize Project:** Identify pieces of the project which can become services, and develop automated processes to manage them

# Team Members and Roles

Austin Gailey - Team Lead / Main interface with FEV Tutor

Pedro Lima - Database Architect

James Coleman - Scrum Master

Emeka Asoluka - TBD

Michael Kerl - Automation Architect

# 

# 

# 

# 

# 

# Software Engineering

1. Life Cycle
   1. Two week sprints
   2. DevOps dev/deployment process
2. Project Management Tools
   1. Microsoft Teams
   2. Trello
   3. Toggl
3. Development Approach
   1. Agile
   2. TDD
   3. CI/CD
4. Development Technologies
   1. Containerization - Docker
   2. System/Service Automation - Kubernetes
   3. Build Automation - Jenkins
   4. Unit Tests - TBD \*Need more information\*

# Minimum Viable Product

Identify at least one bottleneck within the existing system. Produce an organized assessment of existing database structure and querying costs. Propose or build a platform upon which automated builds can be segmented into services managed by Kubernetes.

1. Stretch goals
   1. Successful integration of an existing feature as a service to the platform
   2. Optimized data structures presented/integrated
   3. Optimized DB architecture presented/integrated
   4. Assessment of impact if changes are not made

# 

# Project Deliverables and Timeline

## Existing System Analysis - Date: TBD \*Need more information\*

A general mapping and analysis of existing processes - including platform features, database architecture and querying speeds/costs.

## Microservice Oriented Automation Platform - Date: March 31st 2021

A basic template upon which additional services can be built. This will not include integration of the existing program. The next step would be integrating the existing application with this platform.

## Application launched on Microservice Platform - Date: May 1st 2021

The existing platform will be capable of being deployed from the microservice platform. Stretch goals include additional features/ functionality to application and database.

# Deployment Plans

Deployment will take place via Jenkins automation on Kubernetes managed pods. Each service will be developed and deployed independently.

# Form and Extent of Documentation

Clean code documents itself, but we also want to ensure future users of our project are able to navigate it easily. We will provide sufficient documentation to orient users to various services as well as outline each of the services and how they can be used.

# 

# 

# Risks and Constraints

We are currently working on getting access to the repositories containing the source code to the platform. We also need to figure out what data will be used to test the application in our own environment. The existing documentation for the application has no product release train or robust documentation to help navigate through the project. Finally, there is a chance the code was written in Hindi which may reduce readability/comprehension and make mapping the project more difficult.

1. SWOT analysis
   1. Strengths
      1. Highly motivated team
      2. Diversity of expertise
      3. Support of FEV Tutor’s leaders
      4. Team background with Agile practices
   2. Weaknesses
      1. Technologies are new to some members
      2. Some members are new to professional projects
      3. Team members have other priorities/jobs in addition to this project
   3. Opportunities
      1. Opportunity to learn new industry standard technologies
      2. Opportunity for everyone to build leadership/ communication skills
      3. Opportunity for members to dive into complicated projects
   4. Threats
      1. Global Labor Dev team may be resistant to help
      2. FEV Tutor does not have access to repositories - They are maintained by the Global Labor Dev team