Utilizing Software Engineering and Cloud Computing Principles for the

Development of the Revised Self-Report Assessment of Functional

Visual Performance (R-SRAFVP) application

Kirk Hedlich

A Project Submitted to

GRAND VALLEY STATE UNIVERSITY

In Partial Fulfillment of the Requirements

For the Degree of

Master of Science in Computer Information Systems

School of Computing and Information Systems

December 2021



The signatures of the individuals below indicate that they have read and approved the project of Kirk Hedlich in partial fulfillment of the requirements for the degree of Master of Science in Computer Information Systems.

Dr. Byron DeVries, Project Advisor Date

Dr. Paul Leidig, Graduate Program Director Date

Dr. Robert Adams, ACS Program Chair Date

[**Documentation Standards to be Used**](#_1fz60cys86u5) **4**

[**Abstract**](#_o77i17gycuvu) **4**

[**Introduction**](#_fg4rbidrlhba) **5**

[**Project Management (software project)**](#_tyndimqj1xnd) **5**

[**Organization (software project)**](#_61bmbaoq5q5y) **5**

[**Reflection (software project)**](#_258vhfjb3fdx) **6**

[**Conclusions (software project)**](#_zccxhrju58iz) **6**

[**Appendices (software project)**](#_4hq8x1rs47p4) **6**

[**Bibliography (research project)**](#_75x63ywuutfs) **7**

# Documentation Standards to be Used

- *Every major division, chapters, references, etc., begin on a new page*

- *Use a single 12-point font (body text, footnotes, page numbers, etc.): Arial, Calibri, Courier, Garamond, and Times New Roman*

- *A different font for scientific notations and tables may be used as needed.*

- *All general text is double-spaced.*

- *Body of the manuscript must adhere to 1 inch margins all around*

- *Center page numbers one-half (1/2) inch from the bottom of the page*

- *Page number position should be the same on all pages where page numbers appear.*

- *Justify the left margins but do not justify the right margins.*

- *More information can be found at the* [*GVSU Graduate School website*](https://www.gvsu.edu/gs/project-guidelines-104.htm)*.*

# 

# Abstract

* *No more than 350 words. Double spaced.*
* *What is the question to be answered?*
* *What is the purpose of the research?*
* *What is the flow of the paper?*
* *KWH: The abstract of your project.*

# 

# Introduction

*In this section describe the deliverable created and provide a justification of the project. Include a discussion of how your project demonstrates the application of knowledge gained in the program, as well as new knowledge gained during project development. What is your project about and what have you done? What was your motivation? Why is this problem you've worked on important? What were your goals and objectives? What are you trying to do and why?*

*Project overview. What is your project about and what have you done? What was your motivation? Why is this problem you've worked on important? What were your goals and objectives? What are you trying to do and why?*

* Focus: Using software engineering and cloud computing principles to create the Revised Self-Report Assessment of Functional Visual Performance (R-SRAFVP) application for Occupational Therapists
* What is the Revised Self-Report Assessment of Functional Visual Performance (R-SRAFVP)?
* What is the user base for the application
* What electronic versions exist
  + Spreadsheet
  + App converting spreadsheet into an app
* Why is a new version needed
  + True mobile application instead of an application wrapping the spreadsheet
  + User experience improvements (requirements for what users what/need)
  + Why is an application better than existing tools (including electronic versions)
* What principles from software engineering and cloud computing will be applied
  + Software Engineering process: Define / Design / Develop / Debug / Deploy / Maintain (DevOps light)
  + Cloud computing: move portions into the cloud to reduce client side work
  + Security: Due to HIPAA requirements, incorporate security to prevent HIPAA violations
  + Metrics: Obtain assessment data that is patient agnostic (help to improve the assessment for future versions)

# 

# Project Management (software project)

*Include a discussion of the project management approach and include relevant artifacts (e.g., a project backlog, and burn-down chart indicating iterations).*

*KWH: Include a discussion of the project management approach and include relevant artifacts (e.g., a project backlog, and burn-down chart indicating iterations).*

* Provide for the different phases of software engineering:
* Define
  + Plan the project approach with timeline
  + Define how system will defined:
    - Use cases
    - Requirements format
    - Models for further requirements extraction
    - Etc.
  + Define how system definition will be vetted with users
  + Define how requirements will be tracing of requirements to testing established
* Design
  + System architecture
  + Control flow
* Development
  + Tools needed/used
  + Development process implemented
  + Deployment method implemented (environments established)
  + Version control implemented
  + Coding with phased deployments to production
  + Possible A/B testing for particular functionality
* Debug
  + Deployment to Test env used
  + Testing at System level
  + Testing at integration level
  + Testing at code level
  + Traceability completed
* Deploy
  + Deployment to Prod env used
* Maintain: production support/metrics
  + Feedback for issues established
  + Metrics established/reviewed

# 

# Organization (software project)

*For example, describe the architecture of the major components/services, their location (if distributed architecture is used), and a description of the interfaces. If a standalone application was built, describe the internal architecture of the major components and interfaces.*

***KWH: Describe the architecture of the solution, location, and a description of the interfaces. If a standalone application was built, describe the internal architecture of the major components and interfaces.***

Original Revised Self-Report Assessment of Functional Visual Performance (R-SRAFVP) information:

* Site
* Zip package
* Contents review

Review previously completed class work

* Requirements
* Diagrams
* Architecture
* Development
* Status of previous release

Review expectations of new work

* Processes to be used (what type of methodology will be followed)
* Solution architecture
* Location of project
* Description of interfaces
* Requirements
  + Type of format used
  + How were requirements vetted
  + How feedback was obtained
* Design
  + Traceability to requirements
  + Architecture
  + Tooling
  + Control flow diagrams
  + User Experience diagrams
  + Technical design
* Coding (version control)
  + Possible code review or static analysis
* Testing
  + Which levels of testing will be covered
  + Test cases with traceability to requirements
  + Coverage verification
* Deployment
* Production metrics

Reflection (software project)

*Reflect on both on the project’s progress towards initial objectives and on the new learning. Address the adequacy, efficiency, and effectiveness of your project. Communicate visually when possible (i.e. graphs/charts). Be honest in the evaluation of your work. Describe any weaknesses and false starts; highlight strengths and accomplishments. These items help with your professional growth and enhance your independent learning ability.*

* Discuss and contrast:
  + Processes used
  + Metrics from processes used via artifact production
  + Describe any weaknesses and false starts
  + Highlight strengths and accomplishments.
* Depending on solution release, **possibly provide early usage metrics**

# Conclusions (software project)

*Summarize accomplishments. Outline open issues/directions for future work.*

# Acknowledgements

Beth, Mary, Chris … etc.

# Appendices (software project)

*If applicable. Examples include a simple user manual, reference to repository, etc.*

# Bibliography (research project)

Credit others for their work. Cite location of all tools used.