Digital twin monitoring system project charter

by budurovici Cosmin

Table of Contents

[1. Introduction 3](#_Toc176955057)

[2. Business Case 3](#_Toc176955058)

[3. Approach 3](#_Toc176955059)

[4. Scope 3](#_Toc176955060)

[4.1 In Scope 3](#_Toc176955061)

[4.2 Out of scope 3](#_Toc176955062)

[5. Deliverables 3](#_Toc176955063)

[6. Quality management 3](#_Toc176955064)

[7. Prerequisites 3](#_Toc176955065)

[8. Success criteria 3](#_Toc176955066)

[9. Work break down structure 3](#_Toc176955067)

# Introduction

This project was commissioned by Fontys Venlo more precisely the informatics department. The role of this system is to provide information about the how well Digital Twin is working at any given time so that the developers know when there is a problem, what impact will the problem have on operations. The project will be made using a combination of waterfall and agile and the quality will be mainly managed by the supervisor Gregory Scwake.

# Business Cases

For the digital twin project to be used effectively its users need to be able to know if the project is functioning at any given time. The users need to know when any part of the system has problems and what impact these problems have on the larger system so that they can fix them and take measures to mitigate any problems caused by the failure. While this information can be gathered by studying the system directly this is often slow and error prone.

The objective of the digital twin monitoring system is to provide a way to see the current state of digital twin by automatically collecting data about the system and displaying it in a visual interface in real time.

# Approach

This project will be done using a hybrid approach. In our case this means that the first part of the project (the requirements and design) will be done by using the traditional waterfall approach. We will the split the implementation into individual sprints using the agile approach. At the end of the project, we will do one single roll out.

# Scope

## In Scope

To accomplish its objectives, the monitoring system will:

* Provide information about what actions digital twin is doing at any given time
* Show when any part of the system fails and what consequences on the larger system that failure has
* Display this information through a visual interface in real time

## Out of scope

The monitoring system will not:

* send notifications to users when it detects a failure.
* have functionality that helps solve the problems it detects (it is solely a detection tool)

# Deliverables

The deliverables of this project are as follows:

* Class diagram
* SRS
* User stories
* High level overviews
* A demo monitoring application
* Documentation
* Unity tests

# Quality management

In order to manage the quality of this project the developer Budurovici Cosmin will be supervised by Schwake Gregor and give a midterm report/presentation. During development the quality will be maintained using unity test with at least 95% precent coverage and frequent reviews.

Another step that we will take to ensure quality is manage the following risks:

|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Chance | Impact | Mitigation |
| Part of the digital twin project is unable to be finished in a reasonable time | Low | High | I will make a demo of that part of digital twin to continue finetuning the monitoring system. |

# Prerequisites

The digital twin project is functional so that it can be monitored.

There is a machine capable of running the monitoring system in addition to the rest of digital twin.

# Success criteria

The project is considered a success if the following factors are true at the end:

* The project fulfils the minimum agreed upon functional requirements
* The project can run without any major bugs or loss of data.
* The project can run on the machine provided by the company without impeding other processes of digital twin.

A feature of the project is considered done once it has been reviewed and approved by the relevant parties.

# Work break down structure

1. Basic requirements
   1. Technical system
   2. Product backlog
   3. User stories
   4. SRS
2. Design
   1. Basic structure
   2. High level overview
   3. Class diagram
3. Sprints
   1. First use case
   2. Second use case
   3. …….
4. Rollout
   1. Launch
   2. Post launch