

# Phizi Project Requirements Document

## **Abstract**

This document is the first of three documents describing the final project in the software engineering program. Specifically, this document describes the requirements of the software project from the standpoint of the customer. Phizi company offers motor therapy for children in a virtual environment.

## Table of Contents

1. Introduction .....	3
1.1 The Problem Domain .....	3
1.2 Context .....	3
1.3 Vision .....	4
1.4 Stakeholders .....	5
1.5 Software Context .....	5
2. User Scenarios.....	7
2.1 User Profiles – The actors .....	7
2.2 Use-cases .....	7
3. Functional Requirements .....	9
4. Non-functional requirements.....	11
4.1 Implementation constraints.....	11
4.2 Platform constraints .....	11
4.2.1 SE Project constaraints .....	11
4.3 Special restrictions & limitations.....	11
5. Risk assessment & Plan for the proof of concept .....	133

# Chapter 1

## 1. Introduction

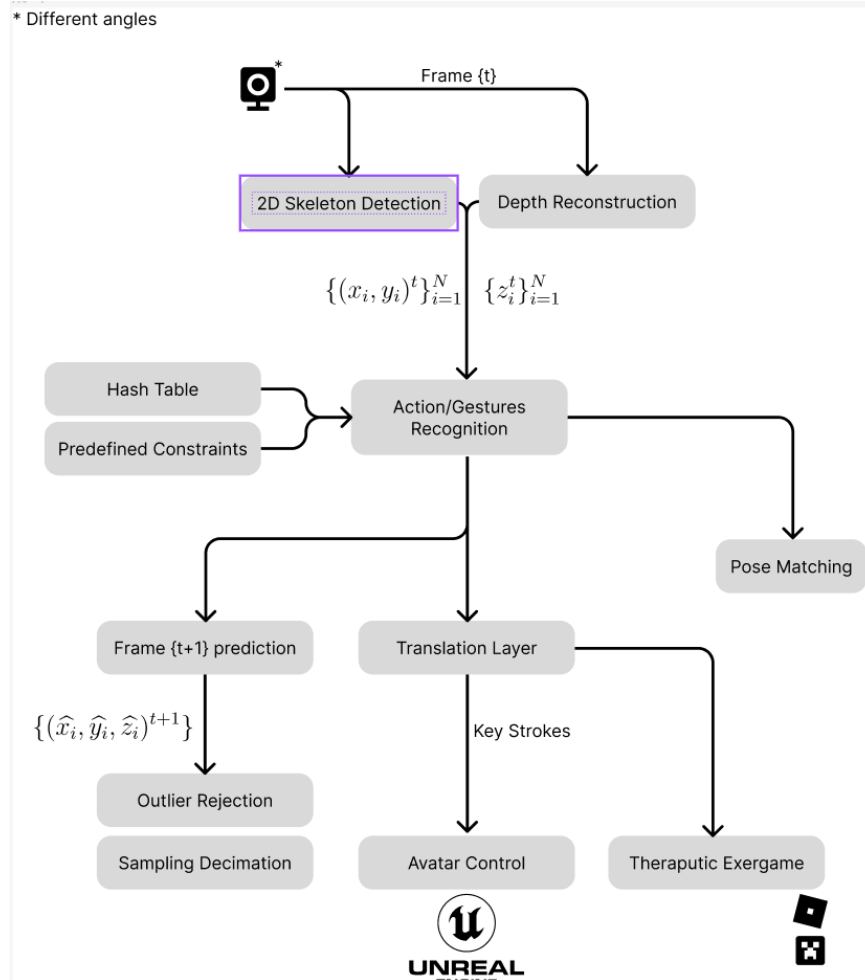
### 1.1 The Problem Domain

Our system provides tools that help diagnose developmental problems among children in the early stages, and in addition provides enjoyable care from a distance, without difficulty and frustration that supports both the children, the professional therapists, and the parents.

### 1.2 Context

During the project we will sample images and analyze the user's movements in order to allow the product to diagnose whether the user is in the correct position. The API we will write will be used as additional nodes in the system architecture.

It will replace the blocks of the skeleton identification and tracking algorithm and also the blocks that need the points that we will distribute in JSON format.



## 1.3 Vision

Creating an API that will be integrated into Phizi's overall system

**Input:** collection of points in JSON format or any other format

**Output:** Analytics, Pose matching, Error estimation, Features extraction.

- Additional goals of the project from our side:

1. To get exposed to advanced algorithms from the worlds of artificial intelligence and the use of visual signals.
2. Building an end-to-end architecture involving Client/Server and even Serverless functions.

## 1.4 Stakeholders

The product is mainly intended for children with Developmental Coordination Disorder (DCD) problems, their parents and their professional caregivers. Our games and tests are integrated into the virtual worlds that the children are already in (Roblox, Tiktok, etc.) and thus we allow them to improve without interfering with their daily lives, without taking them out of their social worlds. Most of the requirements will come from the therapists' side.

## 1.5 Software Context

Our system creates a three-dimensional scan of the patients and performs accurate analysis of his movements. The entire interface is based on computer vision from a standard RGB camera.

Project progress:

- Frame sample from an RGB camera on the end device (lean frontend UI/Client).
- Transferring the sample to the server (backend/cloud)
- Implementation of neural networks (familiar pre-trained architectures) that were pre-trained on the server with the aim of identifying the skeleton, tracking the points of the skeleton and performing depth reconstruction. When the product will be an identification skeleton in three axes.
- Building an API whose purpose is to receive a collection of points in JSON format as input, output: analytics, pose matching, error estimation, features extraction. When the analytics includes feature extraction, calculation of relationships between points, constraints, angles and their

derivatives and pose matching. Some of the surgeries will be performed subject to Basic gesture recognition.

## Chapter 2

## 2. User Scenarios

### 2.1 User Profiles – The actors

- Children - 17% Of children worldwide are diagnosed with a development disability. A significant percentage of them are still diagnosed late after suffering a lot of frustration and almost in sporadic manner. The majority of them suffer from musculoskeletal disorder and could benefit from continuous occupational therapy (OT) and physical therapy (PT).

### 2.2 Use-cases

Use-case name		Pose matching
Actors		Server, Client, User
Pre conditions		The customer has a camera installed on his device
Normal flow	Description	The client takes a video of himself and sends it to the server frame by frame. The server checks whether the user's position is the correct one and will send feedback to the user.
	Post conditions	Feedback about the movement is sent to the client.
Alternative flows		<ul style="list-style-type: none"><li>• An error occurs when the server tries to detect the skeleton.</li><li>• Communication error with the client.</li><li>• Cannot provide feedback about the pose.</li></ul>

Use-case name		User management
Actors		Server, Client, User
Pre conditions		The user has access to Phizi's website
Normal flow	Description	The user registers/logs-in/ logs-out from the system.
	Post conditions	The action was completed successfully.
Alternative flows		<ul style="list-style-type: none"> <li>• Communication error with the client.</li> </ul>

Use-case name		Start Session
Actors		Server, Client, User
Pre conditions		The user is logged in and has a camera installed on his device
Normal flow	Description	The user clicks on "start session" button.
	Post conditions	A series of poses are sequentially presented on the screen for the user to perform.
Alternative flows		<ul style="list-style-type: none"> <li>• Communication error with the client.</li> </ul>



## Chapter 3

### 3. Functional Requirements

	Client / Server	Description	Priority	Risk
1	Client	Sampling the client's webcam	MH	Low
2	Client	Extract skeleton (x, y, z for 33 landmarks) points from the sampled video	MH	High
3	Client	Draw a 2D skeleton on a user's video input in real time	MH	Medium
4	Client	Calculate angles for specific joints in 3D space	MH	Low
5	Client	Calculate angles for specific joints in 3D space	MH	Low
6	Client	Calculate speeds for joints in 3D space	MH	Low
7	Client	Send each frame to the server as a JSON	MH	Medium
8	Client	Display 3D Skeleton in real time and according to the user movements for testing purposes	NTH	Low
9	Server	Store all the user's information (last recordings, login info, treatment progress...)	MH	Low
10	Server	Provide Client with all the exercises needed for a specific user using existing data and data received by the user (start session)	MH	Medium
11	Server	Progress report, provides the client with his progress so far and with various statistics regarding his performance	NTH	Low
12	Server	Collect various statistic information about clients behavior for internal use and Improvement (success rates, playing time, recurrent users, etc)	NTH	Low
13		Registration to the system	MH	Low

14		Login into the system as admin\user	MH	Low
15		Calibration of user data in first use	MH	Low
16		Logout from the system	MH	Low
17		Adding poses as admin	MH	Low
18		Admin can view details on the clients	MH	Low
19		Admin can assign a goal to clients	MH	Low
20		Admin can add a new session	MH	Low

## Chapter 4

# 4. Non-functional requirements

## 4.1 Implementation constraints

**Performance** - Get as close as possible to Real Time at a minimum of 10FPS

**Reliability** - Error in tracking the Skeleton - the pixel error won't exceed 100 pixels of the error in reality

**Usability** - The system should be suitable for use by children and therefore be as easy to use and understandable as possible

**Format** - The points output will be exported as a file in JSON format

## 4.2 Platform constraints

We chose to use Python to run the skeleton detection code because in other languages the skeleton detection algorithm requires an http object

### 4.2.1 SE Project constraints

In our system we will not need special hardware components because we want to sample any camera. In addition we will sample real data from our computer camera or any other camera and not from random data.

## 4.3 Special restrictions & limitations

We will need fast processors and fast communication between the server and the client in order to meet the frame sampling speed requirement.



## Chapter 5

# 5. Risk assessment & Plan for the proof of concept

## 5.1 Risk assessment

The purpose of this risk assessment is to identify and evaluate potential risks that could impact the success of our Phizi project.

<b>Risk</b>	<b>Likelihood</b>	<b>Impact</b>	<b>Response</b>
Dependence on third-party software could lead to failures if any issue arises with the software	Low	High	Using multiple third-party dependencies in case one of them fails
Too many calculations on the client side can lead to fps drops and failure	Moderate	Moderate	Testing the client's performance for each calculation that is being added, and if needed move some of the calculation to server side

## 5.2 Proof of concept

1. For the proof of concept we will develop: 1. Client-side app that runs on the browser, sampling the user's webcam running it through 'Media Pipe – Blaze Pose' and extract the 3D skeleton. This is the main part of the project and the most risky one, extracting an accurate 3D skeleton from a user's webcam.
2. Display, on the user's video, a 2D skeleton using the browser. Displaying the 2D skeleton will allow us to estimate the quality of the Blaze Pose output.
3. Analyze the 3D skeleton, calculate the lengths of body parts and the angle of the joints. This is done on the client side and passed to the server, there in the 3D skeleton we will display the information and see if this is accurate. This calculation, if accurate will allow us to do more complicate calculations in the future.
4. Send the data to the server in real time using a WebSocket. One of the main issues is keeping our data in real time, using a WebSocket might be good enough but if not we can change to a different protocol such as UDP.