המכללה האקדמית להנדסה בכרמיאל

בראודה

Capstone Project Phase B

Virtual classroom analysis for ADHD screening.

PROJECT CODE: 23-1-D-21

Supervisor:

Dr. Anat Dahan

Authors:

Shimon Elkobi, ID: 203333398

Shoham1994@gmail.com

Meitar Sasson, ID: 318365970

Meitar27797@gmail.com

Table of contents:

Abstract

The awareness of children and students with ADHD during the daily routine is increasing. Despite the growing awareness in society, the diagnosis process is still long, and the information about the nature of the disorder for the specific child is limited. There is a fundamental need to create a screening tool that can be used to expand existing knowledge and provide tools adapted to the treatment of different ADHD behaviors. The existing computerized tests (such as Tova and MOXO) are artificial and do not simulate the daily challenges of a real classroom for parents and children. Our idea for the project stemmed from the problem mentioned here. We would like to present an initial screening tool for analyzing the diagnostic results of ADHD in virtual reality glasses. This project uses a virtual classroom environment designed in a previous project. This project analyzes the different types of data collected in virtual classroom settings. Our system analyzes and gives an initial characterization of whether the child has ADHD and how he reacts to disturbances in attention measures. In addition, we can make this data available to the relevant authorities (Diagnosis approved by the relevant authorities). This way they will receive the relevant information in order to take care of the child. The authorities also receive the test data on the subject and will analyze him effectively.

Keywords: *Machine Learning, Virtual classroom, ADHD, Data Analysis, Data Visualization, VR technologies, Oculus technology.*

1. Introduction

Attention Deficit Hyperactivity Disorder (ADHD) is a neurodevelopmental disorder whose core symptoms are attention deficits, hyperactivity, and impulsivity. ADHD is one of the most common neurodevelopmental disorders of childhood. It is usually first diagnosed in childhood and often lasts into adulthood. Children with ADHD may have trouble paying attention, controlling impulsive behaviors (may act without thinking about the result), or be overly active. Today ADHD is a common phenomenon among 6% to 10% of the population (according to a study by the Hebrew University). The awareness of ADHD has been increasing over the years, and as a result, many studies on this subject are being published worldwide by many researchers. Many studies in the world show that the earlier we diagnose the phenomenon and treat it effectively and appropriately for the child, the better the ability in the child's behavior. From a study conducted in 2022, it was discovered that ADHD treatment does not change the child, the adult who suffers from the disorder, but comes to help him and improve his ability to regulate his feelings to develop in the best way to reach awareness and personal growth. The coupling of the rise in awareness of ADHD with the ever-increasing percentage of undiagnosed children has increased the need for creative and accessible diagnostic tools. Today there are several diagnoses for identifying ADHD. For example, the Tova test, and the MOXO test. The diagnoses are made by measuring indicators while the participant is required to perform a task while ignoring visual and auditory distractions. Tests have a credibility problem because they are artificial and not sufficiently like real-life challenges. The ability of Virtual reality (VR) technology may improve the attention of the child and the adult during the diagnosis. Virtual reality allows us to create an ecologically valid environment that resembles the actual classroom where the child faces difficulties. Our development aims to expand the available tools for diagnosing ADHD. Our project aims to create an initial screening before contacting professionals. The system assigns the user to the group to which he belongs, according to the exam results. The results indicate whether the subject has ADHD. The system allows the ADHD diagnostics expert to analyze individually and in groups different measurements using visualizations. These measurements have been found to provide information about different performances in people with ADHD. The system provides a tool for the expert to analyze the subject's behavior test so that he can learn and draw conclusions about the various performances and behaviors of people diagnosed with ADHD.

2. Project Review and Process Description

In order to best define the expected achievements of this project, we met several times with two students who had completed the project (ADHD Screening using Virtual Classroom) we used to collect the dataset. They referred us to their project book and instructed us about how to operate the system as a user, and how to connect the VR technology to our home computer. They showed us how to use the website and the system they developed and clarified the uses of the outputs created from it. The dataset we used was the test results that the system provided. The test results that their system provides do not show a summary of test results, so this is where the need for our system comes from. Using this dataset, we are able to train a machine learning model that provides an initial diagnosis that shows the presence of ADHD in a subject. In the meetings that we held, we presented the system in order to hear their opinion, they were curious to hear more about the system and what technologies we used. The success criteria for this project is a system that will be used by ADHD diagnostics experts and researchers on the subject of ADHD, who will be able to delve deeper into the field and get more indications for people with ADHD through visualizations based on the test results.

Main supporting modules:

ADHD Screening using Virtual Classroom- is a system comprised of an Oculus Quest 2 VR headset running on the Android operating system and a web interface for managing the screening process. The web interface allows for the registration of therapists and patients and the presentation of the final report. The goal of this project is to create an innovative screening tool that can effectively estimate ADHD traits, while also providing a realistic simulation of a classroom environment for patients. The emphasis is on creating a realistic experience for the patient with disturbances that may be present in a real-world classroom. ADHD screening system using virtual classroom combines Virtual Reality technology with a web application interface, intended for use by professional therapists. Therapists first register the patient if he is not yet registered and can adjust the session settings before starting the screening by pressing the "Start Screening" button. The virtual environment simulates a real classroom, and the patient is required to perform a task during two separate sessions - one without interruptions and one with interruptions. The task is to press the letter "X" when it appears after the letter "A". The session with interruptions includes auditory and visual interference. Data from both sessions are collected, including times of necessary and unnecessary button presses, as well as head movement tracking, to measure the patient's level of concentration and how disturbances affected them. The report is then sent to the website where it is displayed, stored in the patient's screening history, and can also be exported to an Excel file if desired.

XGBoost Algorithm- is an open-source Python library that provides a gradient-boosting framework. It helps in producing a highly efficient and flexible model.

2.1 Product diagram

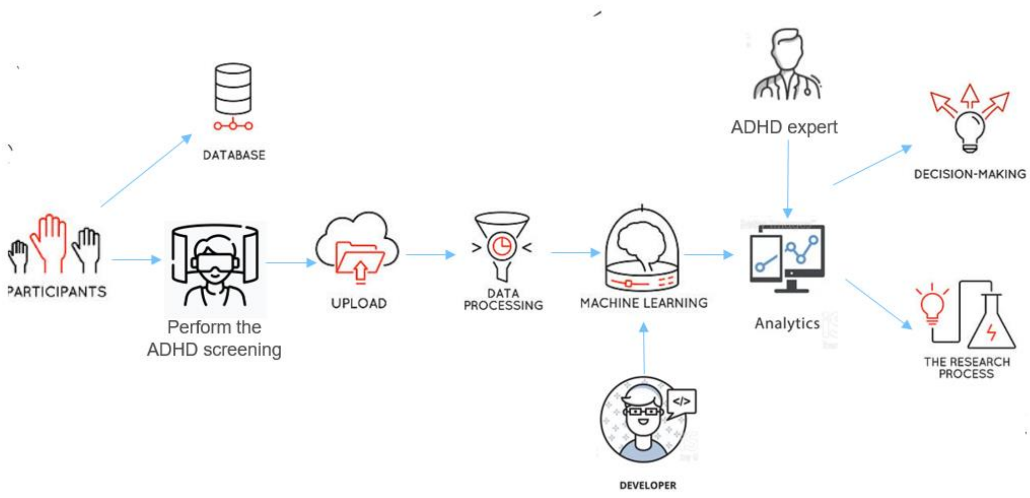


Figure 1: Product diagram

The experiment participants perform a test on the external system, from which we export the result files. We upload the result files to our system, analyze the data, select the appropriate ones and process them, train a classification type machine learning model and finally present the data with the help of graphs and analytics that can be used by the experts in the ADHD field for research in the field or for providing an initial diagnosis to the subject who performed the test. The three types of system users are: the test subjects, who can register in our system only if they have taken the test in the external system, and will be able to view their results. The software developer, that he can add more user results in the future, train the model and refine it. And the ADHD expert.

2.1 Architecture Design

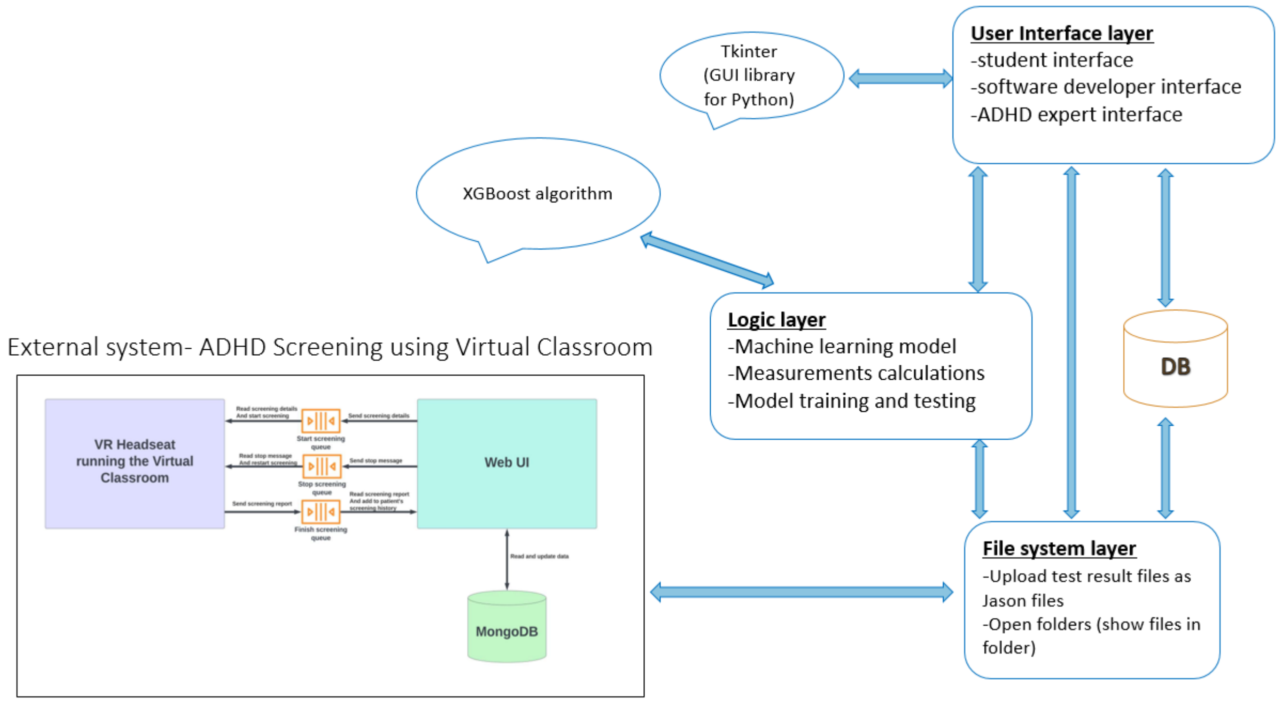


Figure 2: Architecture Design- Virtual classroom analysis for ADHD screening.

The experiment participants perform a test in the external system, from which we export the result files. We upload the result files to our system as JASON files, in addition we allow the software developer to select files in his user interface, from the file explorer. In the logical layer we process the data from the files. And when the student registers in the system with his details, the details should match the details in his results file. In the logical layer, the model is trained, by calling the XGBoost machine learning algorithm, in addition, calculations and preparation of the features are performed for the training. The training results can also be seen in the interface of the software developer. In the database we save the details of the subjects, when the subjects register in our system. We made the user interfaces with the help of the Tkinter library which is a GUI library for Python. The system displays the test results of the subjects through graphs in the interface of the ADHD expert, the results as they are in the original file are raw, therefore the results that are really displayed are the results after the calculations and processing of the features calculated in the logical layer.

2.2 Activity Design

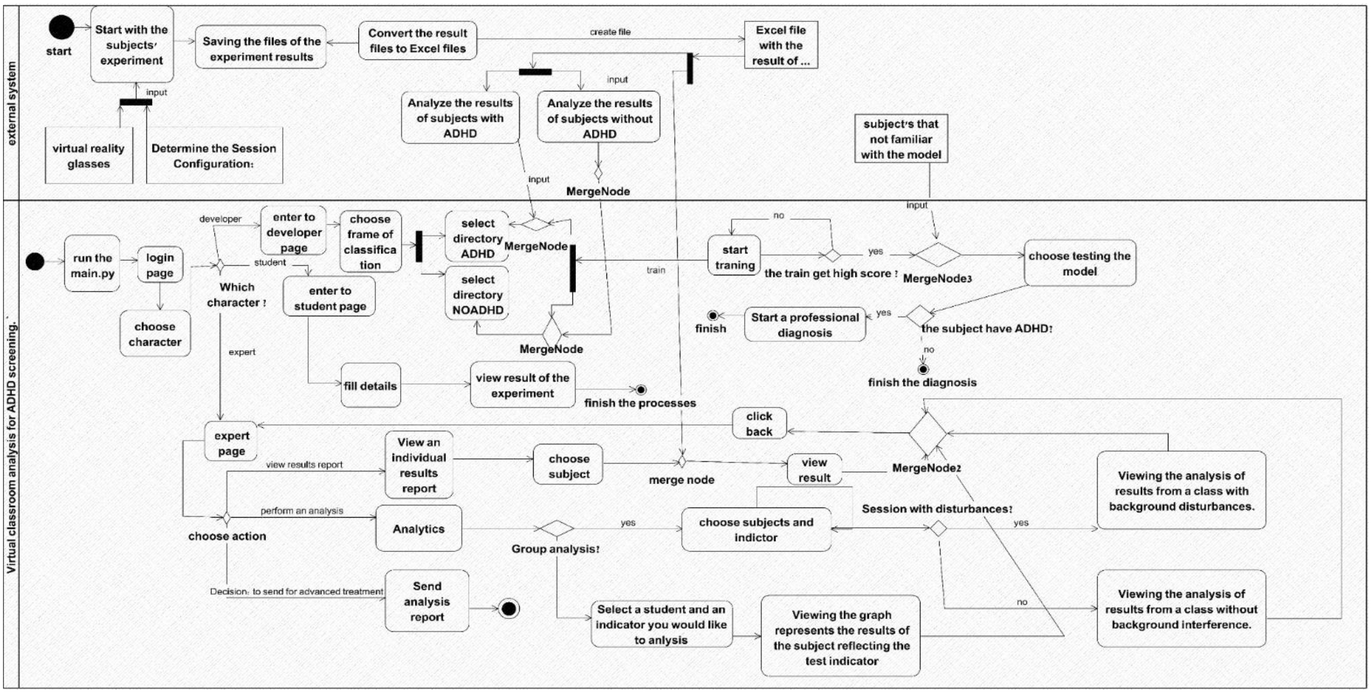


Figure 3: Activity Design

The Activity diagram describes a system for analysis ADHD results. There is an external system that provides us with the results of the subjects, some of whom have ADHD and some who do not. The role of the external system is to provide us with enough data that our system should know how to learn with the help of features and an algorithm used for machine learning. Our system starts with three types of users: Subject, ADHD expert, and developer. Each user has several functions in the system, as described in the diagram in front. For example, the software developer is in charge of the entire professional stage to set up the system from the technical side. He performs all the model training and prediction results using a machine learning algorithm called XGBoost and known machine learning methods. Finally, the system's function is to provide a reasonable suspicion on the subject. Does the subject indeed have attention deficit hyperactivity disorder? Moreover, the one who makes the decision or the expert who performs an in-depth analysis on the subject using graphs presented later in the book has enough experience to decide whether the subject has ADHD.

2.3 Product

Virtual classroom analysis for ADHD screening is a system used to analyze the results of the subjects and learns the critical characteristics with the help of a machine learning algorithm. This provides an initial diagnosis of whether the subject suffers from attention deficit disorder. The system allows the ADHD diagnostics expert to analyze different measurements individually and in groups using visualizations to learn and draw conclusions about the various performances and behaviors of people diagnosed with ADHD.

The system supports three types of users:

1. The subject performs a test in a virtual reality simulation system. The system is a single scene consisting of two sessions, with the aim of assessing the patient's performance on tasks in a realistic environment, including some disturbances that may occur in a real-world classroom. Once the patient clicks the button, the first session will begin. Random letters will appear on the board at regular intervals, and the patient has to press the button when the letter 'X' appears after the letter 'A'. After the first session, there will be a short break and then the second session will start with the same task and with interruptions. The disturbances in the virtual classroom are divided into two categories: visual and auditory. Visual disturbances: paper airplane, flying bird, dim light, and standing child. Hearing disturbances: children chatting, dog barking, phone ringing, bus, messages, birdsong, children's breaks, and gardening. After completing both sessions, the test ends. An Excel report is produced, which includes the patient's performance during the two sessions.
   He can then view the results of his report through our system.
   In addition, the subject will be able to go to a specialist who will provide him with preliminary information on whether he has a suspicion of ADHD and to understand more deeply the results of the test he performed.
2. The software developer conducts research and collects 10 belonging to 2 groups. Students who were previously professionally diagnosed with ADHD and students who were not. The developer asks all students to perform the test in the virtual reality simulation system. In the end, the system receives the results reports for all students. The developer continues to study the different and varied data and concludes that there is more relevant data that can indicate suspicion of ADHD according to the tests and collects them all. To get a clear picture, the developer trains the machine learning model using a guided machine learning algorithm called XGBoost on 80 percent of the test subjects. Performs a model test on the data of the remaining 20 percent that the model did not train. Saves the training results. The developer can choose to train the model with other data and perform a test accordingly. The developer can choose between a list of files, these files are the students' result files, which he retrieved from the system that simulates the virtual classroom.
3. ADHD diagnostics expert - first, the expert can access the results of the subjects. Second, the expert can perform an analysis of the test results with the help of the system. Accordingly, the expert will decide whether the subject has abnormal results. The specialist will choose a group from the two - diagnosed with ADHD or undiagnosed. The expert will choose a single student if he wants to analyze individually or he will choose a number of students if he wants to analyze a number of students compared to the other subjects. The expert will also choose an indicator and choose the session he wants to analyze - a test with or without distractions. Depending on the choices, a bar graph will be displayed. The graph will contain a column with the average of that indicator for all the subjects and another column that will contain the average of the indicator of that selected person. The expert will be able to see the student's situation compared to the other subjects according to the

results of the indicators. The indicators are
- Omission errors
- Commission errors
-Response accuracy
-Response time average
-Head rotation vector size Average.
In addition, the expert will analyze the results with the help of the system and will decide whether to send him for advanced treatment. The expert will select a student for whom he wants to produce a report from the system. The expert will choose to produce a report. The report will include all the analytics of the selected student.

2.4. Development Process

In this study, we performed several steps to arrive at a desktop application system that could make an initial diagnosis of Attention Deficit Hyperactivity Disorder (ADHD). Moreover, with the help of the analysis, we can create a suitable report for the parent/diagnoser to give initial speculation for an initial diagnosis of ADHD. The steps we took in order to reach the desired result:

1. **Learning the systems and technologies**- in the beginning, we met several times with two students who completed the project (screening for ADHD using a virtual classroom) which we used to collect the data set. They referred us to their project book and instructed us on how to operate the system as a user, and how to connect the VR technology to our home computer. They showed us how to use the site and the system they developed and clarified the uses of the outputs created from it. We explored the system that simulates a typical virtual classroom environment and the course of the exam.

2. **The use of technologies**- we downloaded the Unity software to our computers. A system where the students who developed the experiment built the virtual classroom. We learned how to use this system and ran the experiment in the virtual classroom.

3. **Virtual Reality Application understanding**-

Figure 4: Virtual classroom application.

We look for the Application Instructions:
1. Put on the headset
2. The first session without disturbances starts.
3. Press the button whenever you are ready to start the session.
4. Press the button when the letter 'X' appears after the letter 'A'
5. 30-second break
6. The second session with disturbances starts.

We noticed that the system checks four parameters during the exam. Which appear in the test results report as follows:
Not pressed and should
Pressed and should not
Pressed and should
Head rotation

Study of the parameters:
<u>Not pressed and should</u>- Array of times the patient didn’t press the button and the letter 'X' appeared after the letter 'A'.
<u>Pressed and should not-</u> Array of times the patient presses the button and shouldn't press.
<u>Pressed and should</u>- Array of times the patient pressed after the letter 'X' appeared after the letter 'A'.
<u>Head rotation</u>- Array of vectors that tracks head movements.

We researched several task indicators that are relevant factors in the ADHD detection tools like MOXO and Tove tests. The indicators are: **Omission errors** are the subject's failures, who did not press at all when he was supposed to. **Commission errors** that occur by the subject when he overreacts. That is when he should not have responded at all. **Response accuracy.** In addition, we researched a biological indicator which is **head movements**. That can be caused by a variety of external stimuli, including sounds, sights, and other sensations. People with ADHD may exhibit a variety of movement patterns.

When **Omission errors** appear in the result report as ‘Not pressed and should’, **Commission errors** appear in the result report as ‘Pressed and should not’, **Response accuracy** appears in the result report as ‘Pressed and should’, and **head movements** appear in the result report as ‘Head rotation.

3. **Carrying out the experiment**- we invited 8 people to carry out the experiment, we explained to them what its purpose was and asked them to do the experiment in the best way they could. We explained to them about its progress and what they should do in it exactly according to the instructions for the user, which we received from the students who developed the system. We performed the experiment on eight people and also on the two of

us. We exported from the system an Excel file containing all the results of the experiment for each subject.

4. **Data Analysis**- we collected the different and varied results of the experiment. We had data on two groups of people - five diagnosed with ADHD and five undiagnosed. We expected the results between the groups to be significantly different. We researched the experimental indicators and the results in depth. We created an Excel file of all of these data and created several graphs that helped us understand the distinct difference between the two groups and between the types of tests they all took - a test without distractions and a test with distractions. After we studied the indicators and the results, we realized that these indicators derive two more indicators, which can take part in deciding our hypothesis - whether a person is suspected of ADHD. We researched these indicators and finally decided that there are five main indicators whose results will help in the process of training the model. Here is the process: We conducted an initial analysis in order to differentiate between the two studied groups, how significant the difference between them is, and will contribute to the training of the model. During the analysis, we noticed that there are other parameters that are derived from the given results and can even refine the model's ability to identify the group to which the patient belongs. First, we found that there are significant differences in the parameters tested during the exam, between the two groups, and they are:

- The difference between the results of a test performed without distractions and the results of a test performed with distractions for examinees diagnosed with ADHD.
- The difference between the results of a test performed without distractions and the results of a test performed with distractions for examinees diagnosed without ADHD.
- The difference between the results of a test performed without distractions for subjects with ADHD, and the results of a test performed without distractions for subjects without ADHD.
- The difference between the results of a test performed with distractions for subjects with ADHD, and the results of a test performed with distractions for subjects without ADHD.

To make a better diagnosis, we calculated the averages of the parameters tested during the test, which are response accuracy (amount of clicks), commission errors, and omission errors.

Participant:	Features that was tested during the test:	ADHD		Participant:	NO-ADHD	
		SessionWithoutDisturbances	SessionWithDisturbances		SessionWithoutDisturbances	SessionWithDisturbances
EA	Response accuracy(amout of clicks)	5	3	CS	8	7
	Commission errors	3	5		3	4
	Omission errors	5	7		2	3
ES	Response accuracy(amout of clicks)	5	3	DS	10	9
	Commission errors	5	6		0	2
	Omission errors	5	7		0	1
MS	Response accuracy(amout of clicks)	6	4	OA	8	7
	Commission errors	4	5		3	2
	Omission errors	4	6		2	3
OS	Response accuracy(amout of clicks)	6	4	OS	10	7
	Commission errors	5	6		2	3
	Omission errors	4	6		0	3
SE	Response accuracy(amout of clicks)	6	3	SS	8	7
	Commission errors	5	7		1	3
	Omission errors	4	7		2	3

Figures 5: The collected data, when response accuracy is the amount of corrects clicks, commission errors are the amount of clicks that the subject missed and omission errors are the amount of clicks when the subject pressed when he shouldn't have to.

To get an indication of the head movement vectors, we took the vectors and calculated the average of their sizes for each subject.
To get an indication of the response time of each subject, we checked in which second the subject answered, from the time he could answer after seeing the letter X after A. We added the seconds that passed until he clicked every time he needed to and divided them by the number of correct clicks.

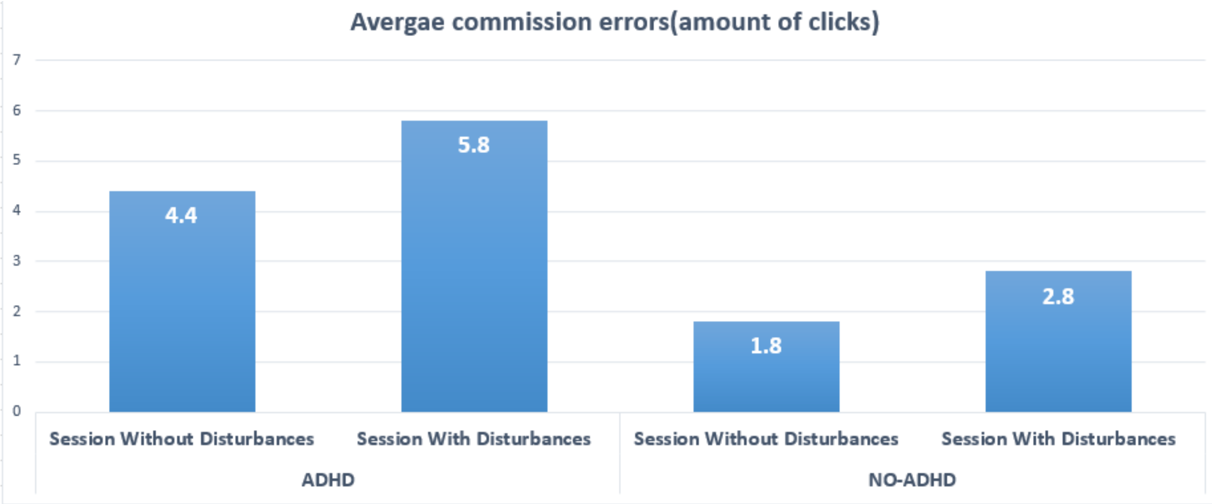
Participant:	features that drived from the features above:	ADHD		Participant:	NO-ADHD	
		SessionWithoutDisturbances	SessionWithDisturbances		SessionWithoutDisturbances	SessionWithDisturbances
EA	Response time avarage	3.2 sec	4.3 sec	CS	2.6 sec	2.7 sec
	Head rotation vector size Average	542.2781	622.5718		545.1526	554.2041
ES	Response time avarage	3.8 sec	4 sec	DS	2.4 sec	2.3 sec
	Head rotation vector size Average	595.2781	642.5718		585.2392	598.2521
MS	Response time avarage	2.8 sec	4.5 sec	OA	2.7 sec	3.7 sec
	Head rotation vector size Average	464.5425	572.4123		467.2591	472.5812
OS	Response time avarage	2.8 sec	3.75 sec	OS	2.7 sec	2.1 sec
	Head rotation vector size Average	545.2663	582.5718		480.6782	490.2471
SE	Response time avarage	3.5 sec	4.3 sec	SS	2.3 sec	2.2 sec
	Head rotation vector size Average	545.1816	605.2594		510.4205	560.2856

Figures 6: Two feature calculations derived from the main features, where average response time is calculated according to how many seconds the subject answered minus the time from the moment he could answer after seeing the letter X after A. We summed them all up and divided by the number of correct clicks. And head rotation vectors were calculated as the average of their magnitudes for each subject.

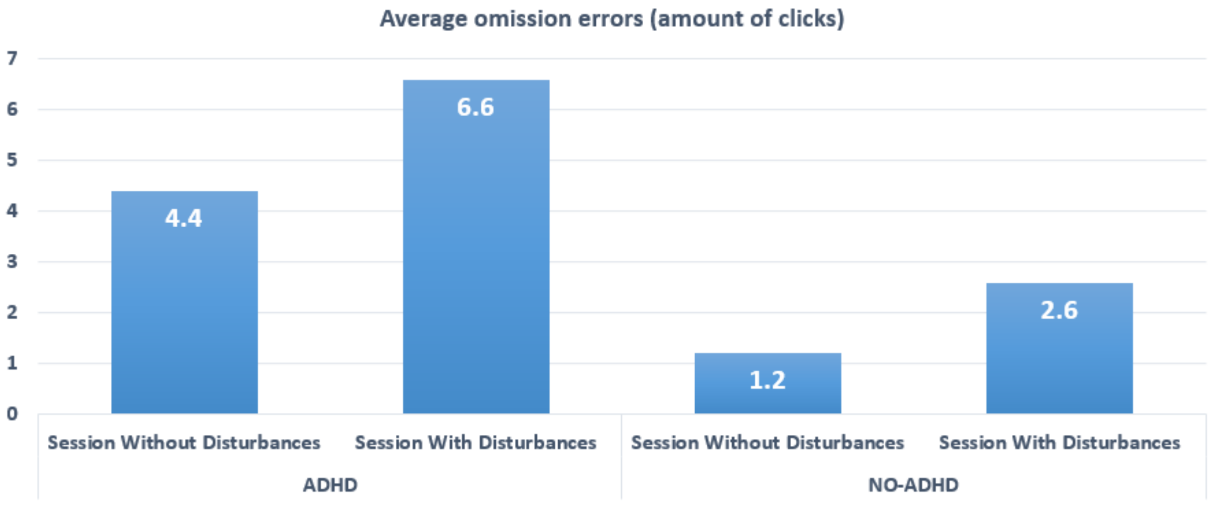
We performed more average calculations for each parameter, these calculations helped us to distinguish in depth between the two main groups and between the two types of tests:

	ADHD		NO-ADHD	
	SessionWithoutDisturbances	SessionWithDisturbances	SessionWithoutDisturbances	SessionWithDisturbances
Average response accuracy(amout of clicks)	5.6	3.4	8.8	7.4
Avergae commission errors	4.4	5.8	1.8	2.8
Average omission errors	4.4	6.6	1.2	2.6
Avarage average response time	3.2	4.2	2.5	2.6
Average average head rotation vector size	538.50932	605.07742	517.74992	535.11402

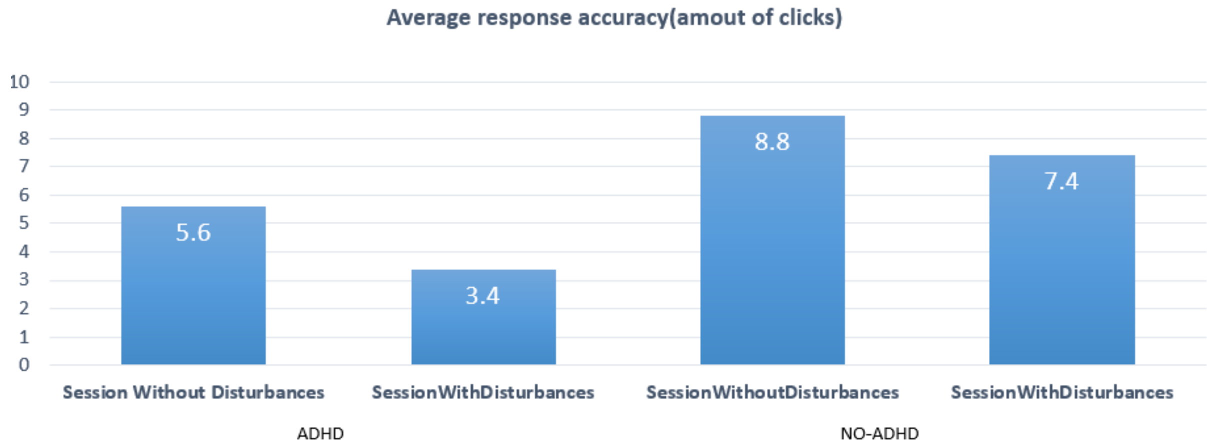
Figures 7: Data analysis- Averages calculation for the five various features indicating the difference in results between the tested groups and between the tests.



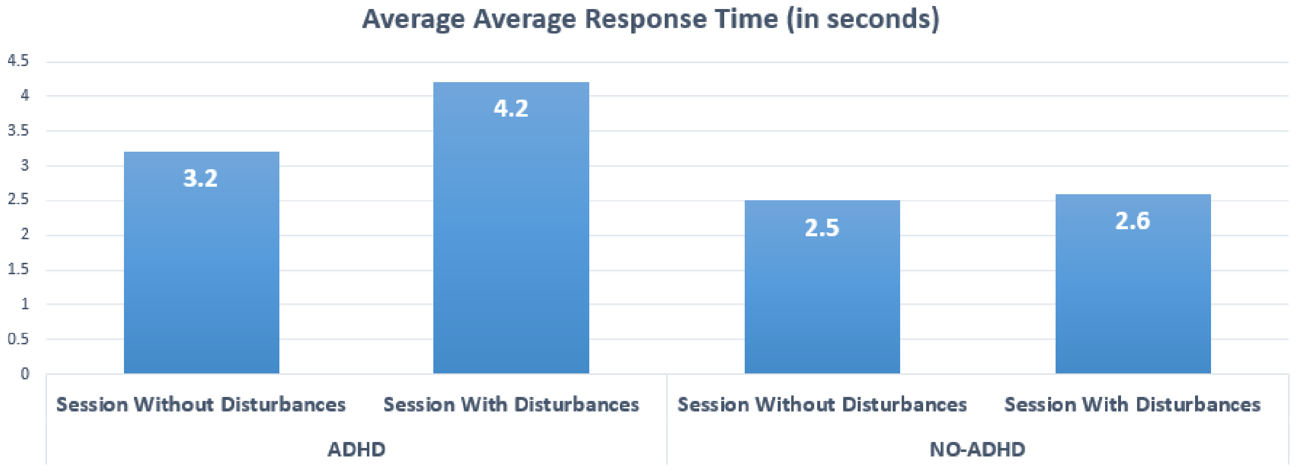
Figures 8: The differences between the two groups and the two tests where the Y-axis is the average commission errors of the subjects.



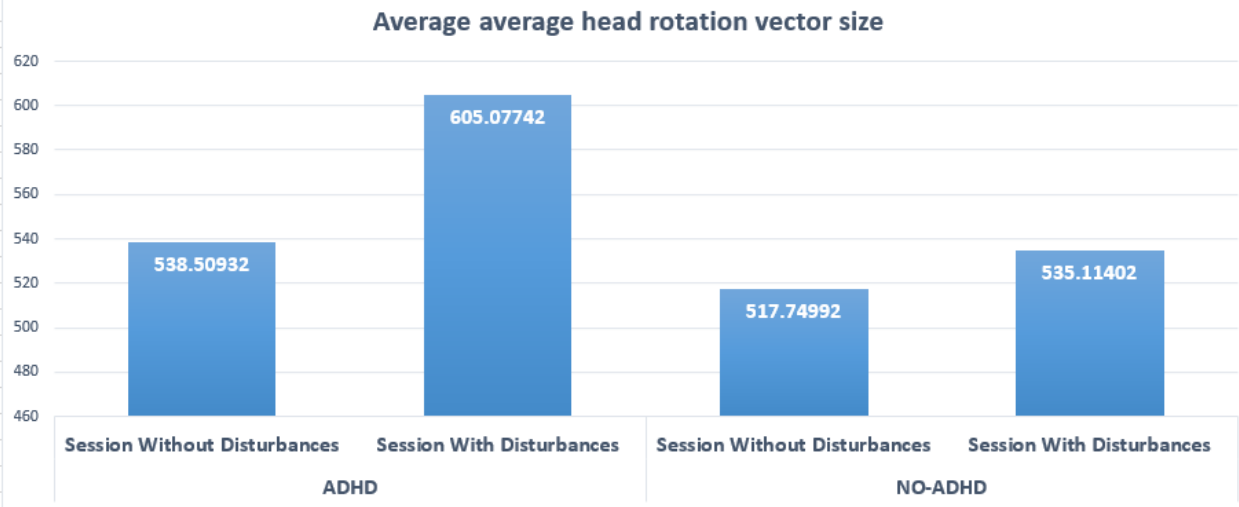
Figures 9: The differences between the two groups and the two tests where the Y-axis is the average of the omission errors of the subjects.



Figures 10: The differences between the two groups and the two tests, where the Y-axis is the average response accuracy of the subjects.



Figures 11: The differences between the two groups and the two tests where the Y-axis is the average of response time averages.



Figures 12: The differences between the two groups and the two tests where the Y-axis is the average of head rotation vector size averages.

According to the findings and the analysis we conducted, we noticed significant features that indicate the performance levels of the two groups we are studying. The following features will be used in model training:
-Omission errors
-Commission errors
-Response accuracy
-Response time average
-Head rotation vector size Average.

6. **Project creation**- after we have carried out the research of the indicators and the preparation for the construction of the machine learning model. We installed Python and

created a new project in Visual Studio code. We uploaded to the project the results of the experiments as Jason files for each subject.

7. **Jason files processing-** Because the test results in the Jason files were raw, we had to perform several calculations for each measure. For example, calculating the sum of the number of clicks the user pressed when he missed, calculating the sum of the number of clicks the user reacted impulsively, calculating the average reaction time, calculating the average of the vector sizes of the head movements and calculating the accuracy of the response.

8. **Machine learning model preparation**- we started to research and study the steps of building a machine learning model. The purpose of our model was to make a preliminary diagnosis for ADHD based on the test results of the subject who wants to know if he has a suspicion of it. First, we had to understand the model would be of a classification type. Classification in machine learning is a supervised learning task that involves assigning predefined labels or categories to input data based on their attributes. The goal is to build a model that can learn from labeled training data and accurately predict the class labels of unseen data points. In a classification problem, the input data consists of a set of features, and each data point is associated with a specific class or category. The model learns patterns and relationships in the training data to create a set of rules so that the classification model can then be used to predict the class labels of new instances based on their feature values. The output of a classification model is a discrete value, representing the predicted class label. We prepared the data for training by performing necessary preprocessing steps such as handling missing values, rescaling features, and coding categorical variables. We split the data into training and testing sets. We chose an XGBoost machine learning model, which is suitable for our problem and data. We imported the library of the algorithm into the project.

9. **Model training and evaluation**- we trained the selected model according to the training data. The model learned patterns and relationships within the data. We then performed an evaluation of the model's performance on the test data which was twenty percent of each labeled group which is four subjects data files.

10. **Users interfaces development**- we started working on the front end of the system, which included building the layout and implementing user interactions. It took us some time to figure out how to build the components correctly. We used the prototype we made the previous semester. For the ADHD diagnostics expert screens, we put a lot of thought into it. We tried to think which graph would best illustrate the essential difference between the different and varied data between the groups and between the types of tests that the subjects performed. which will finally be used by the ADHD specialist in the decision-making process regarding subjects who wish to know whether they have suspected ADHD.

2.4.2 Algorithm

XGBoost algorithm

We used the XGBoost classification algorithm used as a supervised machine-learning tool to classify our datasets into groups of suspected ADHD and non-suspected ADHD. Due to the nature of supervised machine learning, we created a labeled dataset that was used to train our classification model. The users performed a test on the external system, thus creating a labeled data set for training our model. After training our model we can classify the subjects. In order to understand the XGBoost algorithm, first we should understand two terms in gradient boosting:

- Loss function- The loss function in regression problems may be a squared error, and in classification may be a logarithmic loss. It is decided by the matter being resolved. The user can set their loss function.
- Weak Learner- They are weak models that do not perform well with our data. Therefore, do not fit too well. They perform better than the random model. In our case, decision trees are used as the weak learner. Trees are built greedily, choosing the best split points based on scores or to minimize loss function. Initially, we will use short decision trees that had a single split.

  The algorithm steps are:
  -Build an XGBoost tree with a single leaf.
  - Create a first prediction and calculate residuals. Calculate the target variable's average as a prediction, and then use the loss function to calculate the residuals. The residuals for subtrees come from prior tree predictions.
  -Compute the similarity score according the following:

$$Similarity\ Score = \frac{(Sum\ of\ Residuals)\hat{}2}{Number\ of\ Residual + \lambda}$$

  Where λ is regulation parameter, which control overfitting (when the model fits to its training set exactly) and reduces the prediction's sensitivity to individual observations.
  - Using the similarity score, choose the suitable node.
  - Gain is calculated using similarity score. Gain indicates how much homogeneity is produced by separating the node at a specific place by comparing the differences between old and new similarities. This formula is used to compute it:

$$Gain = Left(similarity) + Right(similarity) - Root(similarity)$$

  Positive gain means the splitting idea was good, otherwise – not.
  -Use the procedure described above to create the ideal length tree. Tuning with the regularization model parameters would be used for pruning and regularization.
  -Using the Decision Tree that created, predict the residual values.
  -The following formula is used to find the new set of residuals:

$$New\ Residuals = Old\ Residuals + \rho \sum Ppredicted\ Residuals$$

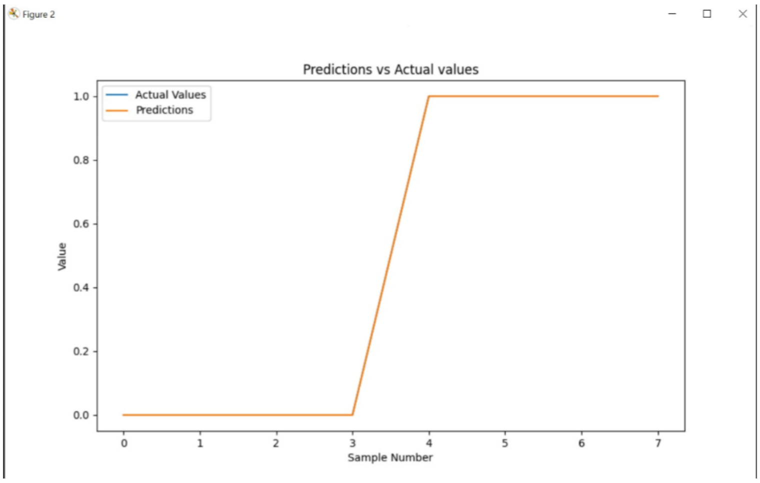
  Where ρ is the learning rate.

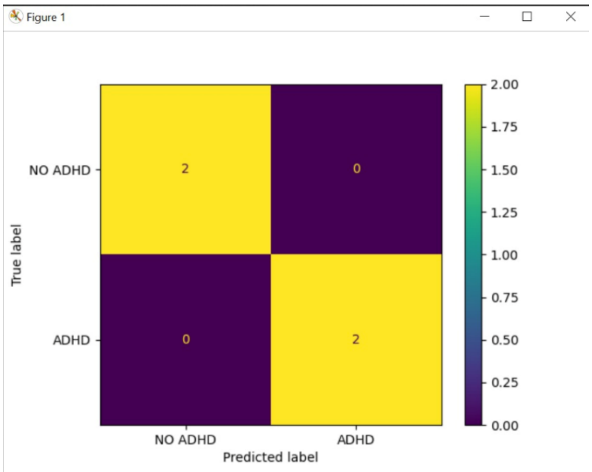
- Repeat the steps from the beginning, for all the trees in the model. Now we keep repeating the same procedure, creating a new tree, making predictions, and calculating residuals with each iteration. We continue doing this until the residuals are extremely low or the algorithm's maximum number of iterations has been reached [1].

Training results:

```
x_train shape: (12, 6), y_train shape: (12,)
|   iter    |  target   |   gamma   | learni... | max_depth |
-------------------------------------------------------------
| 1         | -0.8      | 0.5813    | 0.8567    | 5.481     |
| 2         | -0.8      | 0.3529    | 0.08974   | 5.549     |
| 3         | -0.8      | 0.6765    | 0.1043    | 5.506     |
| 4         | -0.8      | 0.4166    | 0.2114    | 7.41      |
| 5         | -0.8      | 0.5462    | 0.2902    | 5.165     |
| 6         | -0.8      | 0.04973   | 0.9643    | 9.988     |
| 7         | -0.8      | 0.518     | 0.5913    | 4.951     |
| 8         | -0.8      | 0.4528    | 0.8292    | 7.65      |
=============================================================
Model trained successfully.
```

```
Model trained successfully.
ADHD label:  0
ADHD label:  0
ADHD label:  1
ADHD label:  1
x_test shape: (8, 6), y_test shape: (8,)
x_test shape: (8, 6), y_test shape: (8,)
Accuracy: 100.00

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         4
           1       1.00      1.00      1.00         4

    accuracy                           1.00         8
   macro avg       1.00      1.00      1.00         8
weighted avg       1.00      1.00      1.00         8

y_pred: [0 0 0 0 1 1 1 1], y_test: [0 0 0 0 1 1 1 1]
```





Figures 13, 14: Training plots.

The following two images represent to us the results obtained by our model with the help of the algorithm. We have two groups of subjects: one group that represents people with ADHD and the other group that has people without ADHD. We trained the model on three files from each group and predicted the remaining two in each group. We can see that, indeed, in the confusion matrix, we got an accurate swing, and also, in the prediction graph, we realized that the graphs are converging, but this can also indicate overfitting. For a future development, we considered that more than the number of files would probably be needed for learning. It is essential to know that the results are not the most accurate because of this, so for future work, they will have to add many more subjects to get more accuracy in diagnosing ADHD.

2.5 Testing Process

Table 1: Functional Testing. The testing environment is manual tests.

#	Test ID	Description	Expected Result	Precondition	Comments	Result
1	Commission Error	When the subject did not press the button at all during the exam	In the test results we will see in the commission error column empty.	1. Performing the experiment without pressing the button at all throughout the experiment.	View the Excel file that shows the results of the test.	Success
2	Subject's details	Choose the student actor in choose actor box Go in the student page Don't fill all the fields	We will receive a pop-up message when the student forgot to fill in his details such as email or full name and press on the bottom register	1. Choose the student actor in choose actor box 2. Forget to fill one of the details and click of each bottom.	Check which label of information has not been filled with details.	Success
3	User register Successfully	In the student page the fill all the details to register to the system	A database user has been created	Filling in all the required details	Registration has been successfully completed	Success
3.1	Viewing the results report	Registration in the system with a name different from the test results	Throwing an exception that the name you registered with is different from the name that appears in the report	Filling in the details on the student page with a different name from the report in Jason	Registration has been successfully completed	Success
4	Choose to test the model when it has not yet been trained	In Software developer screen- Choose to test the model before train the model	It is not possible to perform a test when the model is not yet trained, results that do not correspond to reality.	In Software developer screen- Choose to test the model before train the model	"The model cannot be tested when it has not yet been trained".	Success
5.1	Check learning of the model	Overfitting, choose quite a few learning files	Overfitting - the system predicts subjects it has learned	When we want to select model training on the programmer's screen, the files in	See if there is a 100 percent match between the study and the prediction	Success

				training are the same as the files in prediction		
5.2			low learning rate - We will get a very low score in the feature matrix	When we want to select a training model on the programmer's screen, we select a small amount of training files	Check whether the prediction results are high or low.	Success
6	Choose actor	Before entering the system, the user must choose a character, a programmer, an expert, or a student.	A message will pop up for the user to select a character before pressing the continue button.	Enter to the system. Try to continue before choosing a character.	Check the combo box if any character was choosing.	Success
7	Choose action	On the expert's page, select an action from the action list	A message will pop up for the user: Please select action from dropdown menu	Enter to the expert user and try to do continue without choose action	Check the combo box if any character is chosen.	Success

2.6 Project Discussion

2.6.1 Result

We have managed to create the proposed system from phase A, which includes a classification model, which predicts whether the subject has suspected ADHD or not. By combining our system with actual data from people with ADHD, it can be used to validate simulation results from the virtual classroom. In addition, we conducted simulations on people with ADHD and without ADHD and created a system that shows ADHD diagnostics experts the significant differences between the results and thus provides them a tool for simulation results analysis, help in the process of decisions making and for research in the field. The system is based on machine learning and is designed to classify subjects who have made an initial diagnosis of ADHD. The system does not classify the subject with 100% accuracy, but can be used as a good basis for future development. Any software developer can train the model himself with the help of the system, with a larger data set after performing the experiment on more subjects, and reach better accuracy.

2.6.2 Challenges

1. Harnessing the subjects for the experiment - we had to gather a group of people who had already been diagnosed with ADHD and a group that had not. It was challenging to coordinate a time and place that would suit everyone so we could perform the experiment simultaneously.
2. In the project we used a machine learning algorithm, so its goal is for the model to learn to predict subjects who have never been tested whether they have ADHD. Because collecting a large number of subjects was very difficult, we knew that there probably wouldn't be good enough data to get the best result.
3. The development of the model - at the beginning of the development we did not know where to start. After meeting with our supervisor, Dr. Anat Dahan about the project, we received advice from her that we should work with the iterative development method, and develop a little bit of each part of the project at each stage.
4. Developing the GUI - Python has a GUI library called Custom Tkinter, an extension of Tkinter. However, it is minimal in terms of presentation capabilities and design additions, so we had to compromise significantly in the design of the GUI in terms of designs or functionality.
5. Development of a system continuation of the system developed in the previous semester - in the beginning, we had difficulty coordinating Zoom meetings with the students who developed the system we used to collect data for the model. Dependence on them hindered us a little, but they helped us and we managed to achieve what was required.
6. Use of new systems - we have not worked with systems like Unity and VR before.
7. First we decided to train the model on raw data from the exam results, and then we noticed that this data does not contribute to the classification of the model. To overcome this, we decided that we perform data analysis. We created an Excel file in which we gathered all the data. For each indicator, we thought about what size would be significant for us. The sum of the number of clicks, for example, the difference between the clicks and the times in order to calculate the response time of the subject and more. This file helped us see the significant differences between the groups, in contrast to the raw data which did not contribute anything to us.
8. Developing a machine learning model - this is the first time we have developed a machine learning model, therefore learning the rules and methods was new to us and we had to learn a lot before we developed it and during the development process.
9. Timeliness - we are both students in the last semester of our degree and also working, so combining all our tasks and responsibilities at the same time was challenging for us.

2.6.3 Conclusions

The system we used to collect the data for our model simulates a typical classroom environment. Provides children with an opportunity to have a diagnosis that includes real-life challenges. Through the test, we learned that the VR simulation is distracting and requires a high level of concentration. This can be deduced through the visualizations we present through our system. With their help, the data collected from the simulations can be analyzed to reveal important insights about the child's level of concentration and how he is affected by distractions. It can be concluded that people diagnosed with ADHD will act more impulsively, less focused, and sometimes even slower.

3. User Guide

3.1 General System Description

Virtual classroom analysis for ADHD screening incorporates virtual reality Technology with a desktop application interface. The user must activate the virtual reality application and the experience. At the end of the experiment, the subject receives the results of the experiment, which are saved as an Excel file. Our system accepts all test results as input and analyzes them. The system uses a machine learning algorithm and extracts special features such as average head movements or average reaction time to understand the distribution of the data and thus learn to identify the subjects with ADHD and those without ADHD. The purpose of the system is to learn to predict whether new subjects are suspected of having ADHD.

3.2 Operating Instructions

Figure 15: Home page

On the home page, the user must choose a user type with which he wants to enter the system and perform the following actions. The system has three types of users: an ADHD expert, a subject (student), and a software developer.

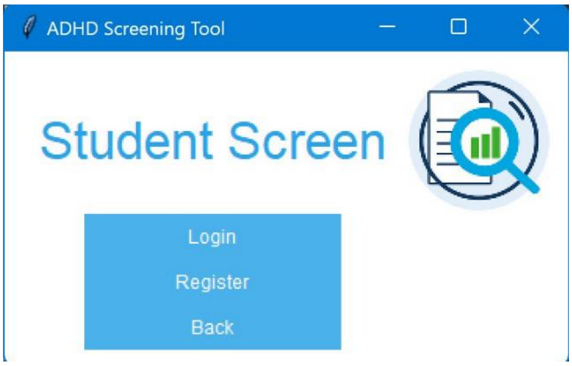


Figure 16: Student page

We can see the following screen when the user has decided that he has chosen the student's user type. There are three options on the student page:

- The first option is to register in the system by filling in the student's details and opening a user with which he can enter the system.
- The second option is to log in to the system using the user and password he registered with, and then he will be able to view the test results.
- The third option is return to the home page.

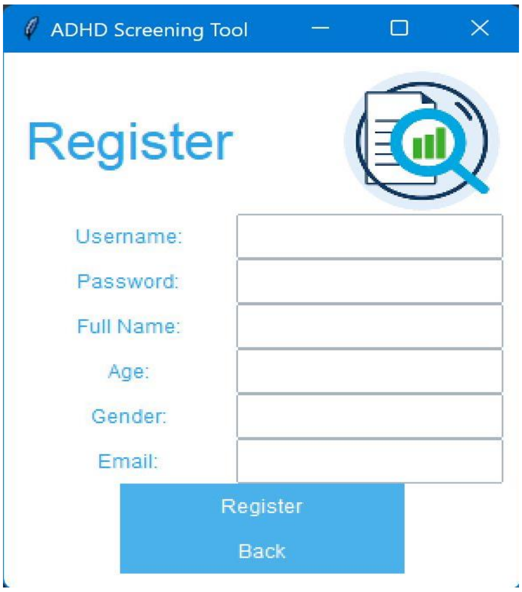


Figure 17: student Register

The following figure describes the first option, when the student has not yet registered in the system and wants to register for the first time in order to log in. The user needs to fill in his details and click ‘Register’.

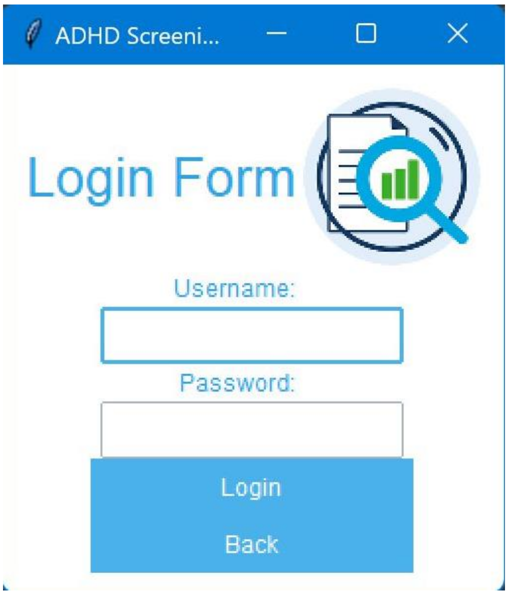


Figure 18: login to student

The following figure describes the second option when the student has registered in the system and wants to log in. The user needs to fill in his details and click ‘Login’.

Figure 19: Details of the subject's results

The following figure describes the option number two when the student has connected to the system. here he can see his details and his test results. Now, the student can choose to see his test result, the test that was done without interruptions as well as with interruptions.

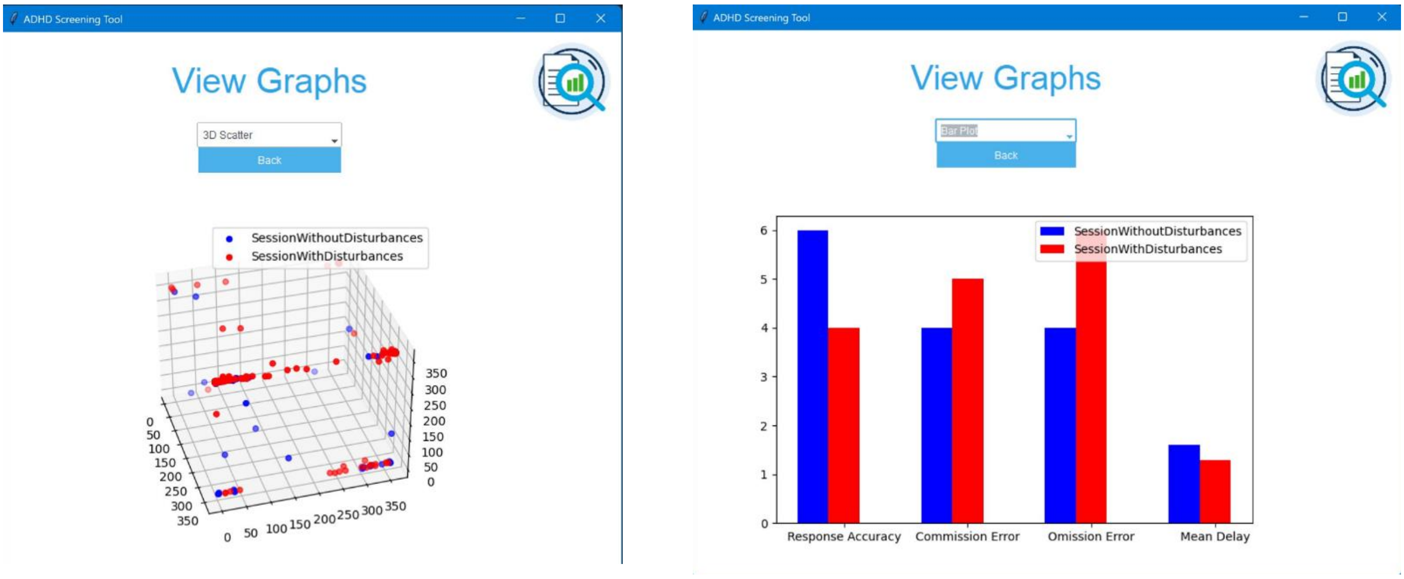


Figure 20: Viewing graphs

The student can also view his test results as graphs and may have the option to know how to estimate the data more and understand them more deeply. And this is done when the user clicks the "View Graphs" button.

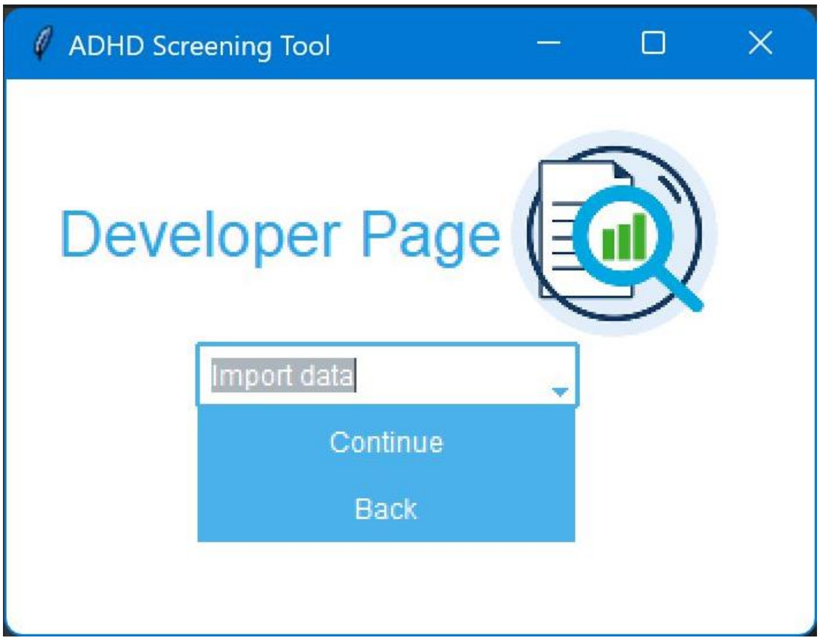


Figure 21: Developer page

After the software developer selects the "Software developer" user type, he arrives at this screen. The software developer is responsible for developing the model by applying the machine learning algorithm and extracting essential features from the subjects' pages so that the model will learn whether the subject has ADHD or not. Now we will present more of the functionality of the software developer in the system and how it is manifested.

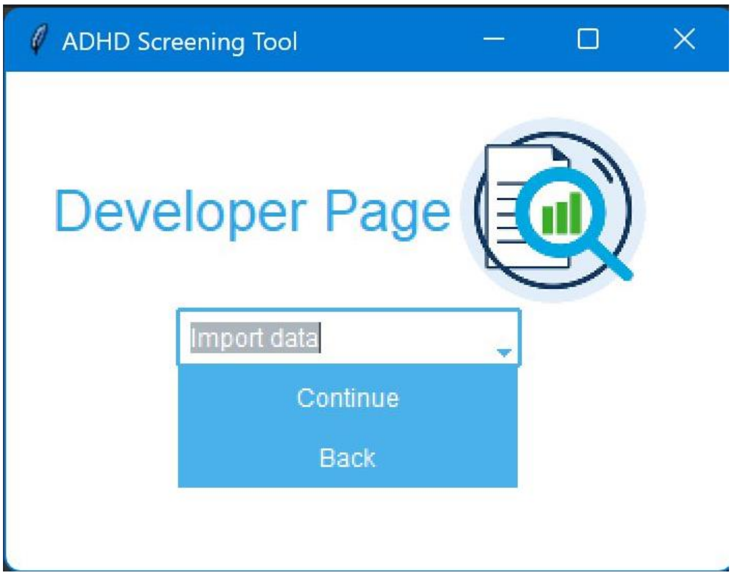


Figure 22: Importing the subjects' files

Adding data from the operating system to the database (JSON files), these files represent the results (input) from the external system after in-depth analysis, understanding them, and adding the data to the model. This way, the model can perform training on the data most efficiently.

It is important to note that the JSON files are called by these names and not by the names of the subjects in order to maintain the discretion of the subjects.

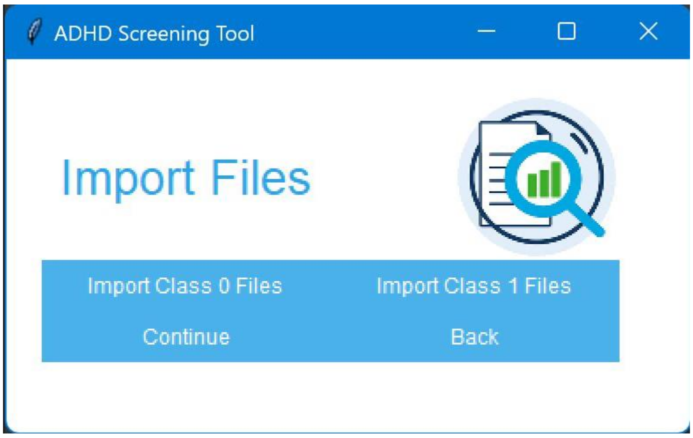
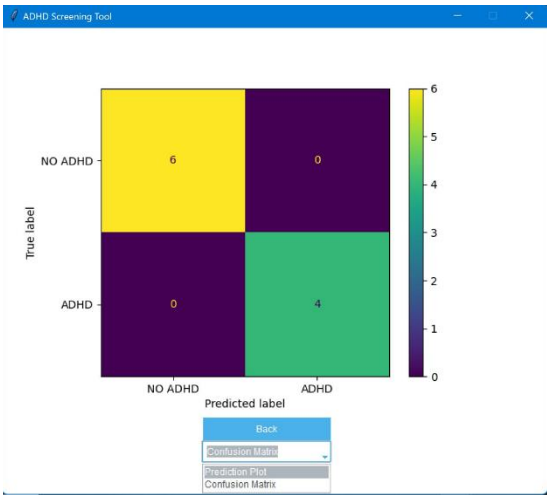


Figure 23: Import the files according to the classification of the subjects

Because the experiment began with a division of 2 groups of subjects, one group in which the subjects indeed suffer from ADHD and the other group that has never been diagnosed claims that they do not have ADHD. So, for training, we put a 0 or 1 label on the files when there is reasonable suspicion in the results of the subject.



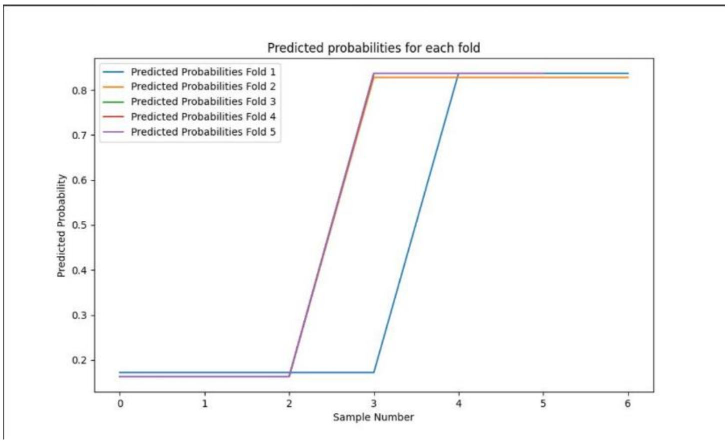


Figure 24: confusion matrix

figure 25: predict probabilities

When the software developer chooses to see the model's prediction, he can display it with a confusion matrix or graphically. The right graph shows for each K-fold its predictions (this is what divides the training set into training and validation for five iterations because we chose the number k=5).

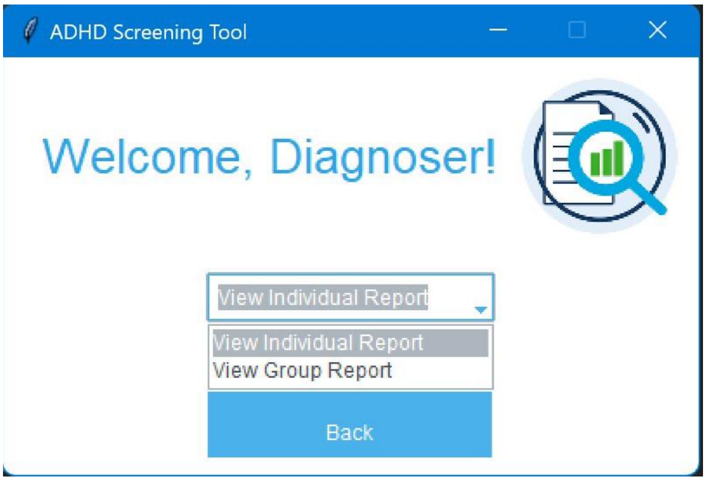


figure 26: ADHD expert's page

An ADHD diagnosis expert will click on View individual report when he wants to analyze data for a specific subject and make a decision - whether to send the subject to advanced treatment. The second option is to click on View group report to conduct a group analysis of several subjects to learn more about the performance.

Figure 27: Viewing an individual report.

The expert can view the personal reports of each subject, and the expert can select the file of the subject he wants to examine.

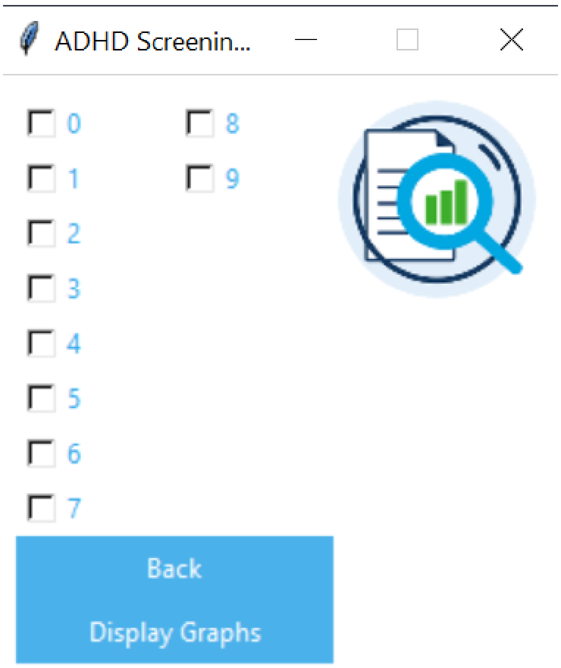


Figure 28: Viewing several reports of subjects at the same time.

The expert can view several reports of subjects at the same time to compare the subjects' performance in a graphical form so that he can estimate the subject in comparison to other subjects.

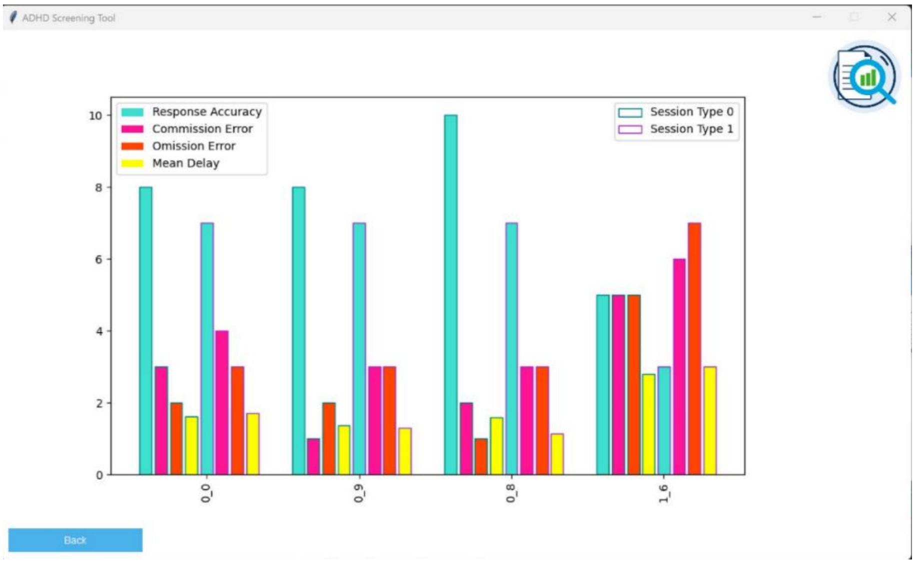


Figure 29: The global graph with the selected files.

After the expert has chosen to view several reports simultaneously and has chosen which files he wants to see, the expert can display the files he has selected in a graphical form to estimate the comparison between the subjects.

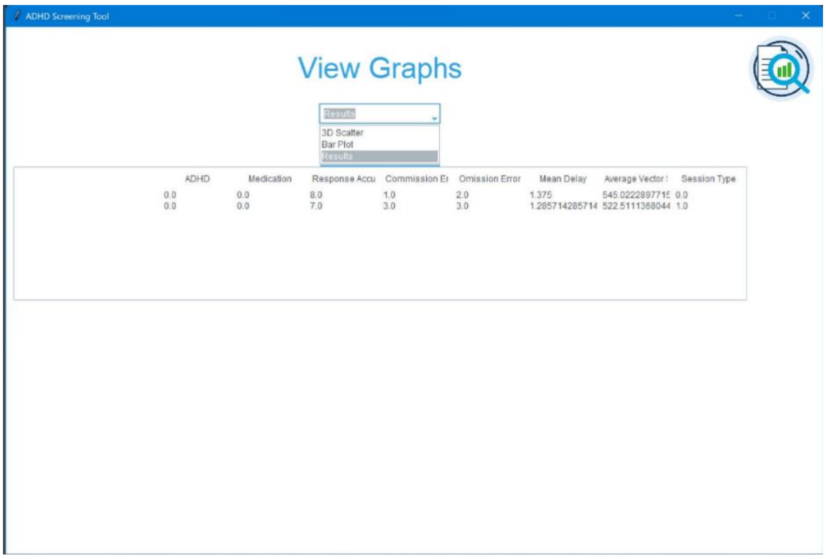


Figure 30: Final results of the subjects

Here, the final decision will be presented as to whether the subject is suspected of ADHD. As soon as a one appears in the ADHD column, yes, then there is suspicion; otherwise, 0 means no.

3.3 Maintenance Guide

3.3.1 Architecture Design

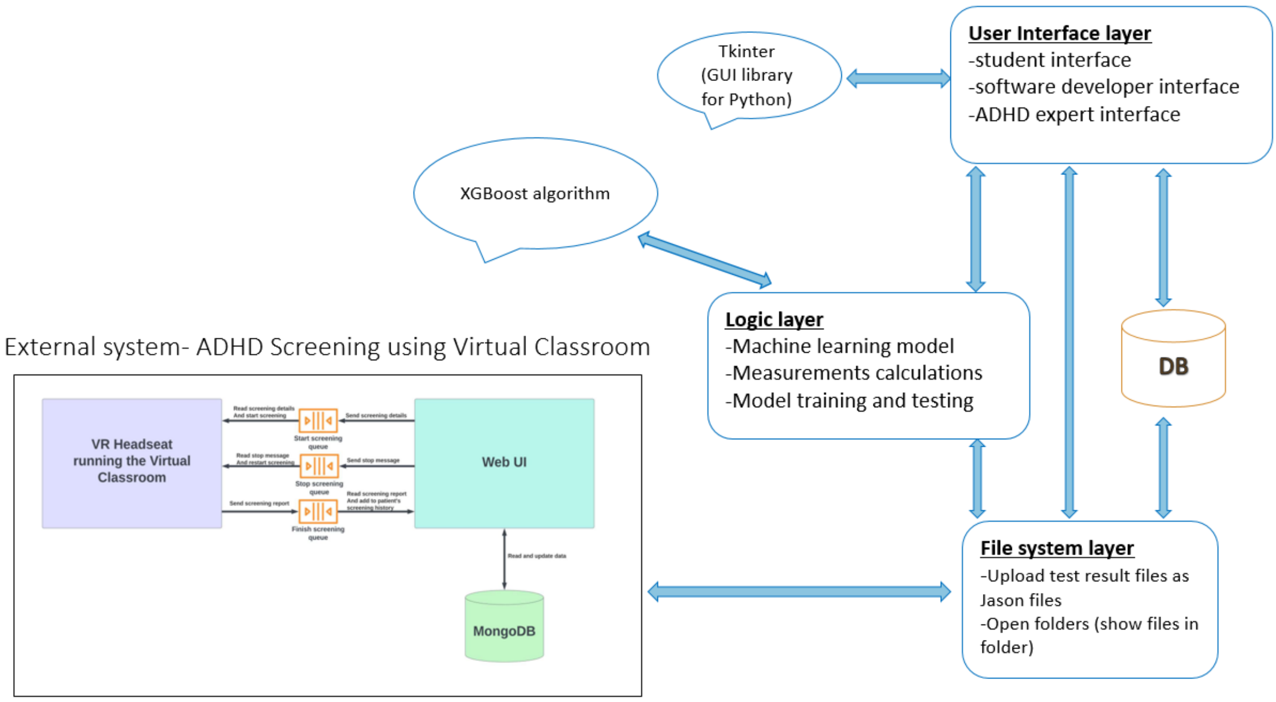


Figure 31: Architecture Design- Virtual classroom analysis for ADHD screening.

3.3.2 Operation environment

PyCharm Community Edition 2023.1.3- Download the software, open the project folder inside a new project in the software, make sure all the libraries are imported before running, run the project using the following button: 

CouchDB -Download the software to your computer, during the download you will be asked to enter a username and password - use the username 'admin' and password 1234. Then you will be asked to enter a cookie, use 1234. After the download is complete, go to the web browser and log in. Enter in the address bar -localhost:5984/_utils. Enter the username and password as specified.

3.3.3 Python libraries that needed to be installed in order to execute the code

```
bayesian_optimization
CouchDB
customtkinter
imbalanced_learn
imblearn
matplotlib
numpy
```

```
pandas
scikit_learn
tk
ttkbootstrap
ttkthemes
xgboost
```

3.3.4 Database description

Description of the contents of the database.
Our database consists of three main files:

- student_details: There will be an entire list of students who registered through the GUI system and registered on the student's page.
  It is essential to know that when a student registers in the system, then in the label of the full name: he must enter the name exactly as it appears in his test file in the JSON FILE, otherwise when he connects and tries to view the report, he will not be able to view it. He will receive an exception because the file does not exist.
- test_dataframes: In order to enter all the details of the subjects into the data frame and there the results of the subject will be displayed in terms of all the indices tested and whether he is suspected of attention and concentration disorders. Training should be done on the desired files in the system.
  That is to enter the character who is the software developer -> perform training -> select the desired files that we want to train both from the folder of those without ADHD and from those with ADHD -> and import the information from the files and only then will all the information of the files enter the data frame.
- test_results: All the tested JSON files will appear here. Each subject in the file has two results, one test without interference and one with interference. A test without disturbances is labeled as a 0 by us, and a test with disturbances is labeled as a one by us.

Reference:

[1]. https://www.csias.in/explain-the-step-by-step-implementation-of-xgboost-algorithm/

Git link:

https://github.com/MeitarSasson/Virtual_classroom_analysis_for_ADHD_screening/tree/master/src