


Extended Reality (XR) Immersive Visualisation: Identifying AI project member ‘needs’ in order to design for a more effective Low-Code Machine Learning Model Development experience.

R. Wheeler and F. Carroll 

Cardiff School of Technologies, Cardiff Met University, Wales

Abstract

With organisations making machine learning projects more of a priority, issues have been found regarding the presentation of these types of projects and in particular, in explaining how the models that are produced work, not only internally but also to the final user. The following paper discusses the design and development of a novel Extended Reality (XR) solution that enables rapid development, experimentation and clear presentation of complex machine learning models using eXplainable AI (XAI) principles. The paper documents the findings from a short initial feasibility questionnaire study which probed participant’s opinions around their current use of XR environments, low-code development platforms, and their experience of working on machine learning model development projects. The findings of that study showed that the proposed solution could be deemed novel especially regarding its use of extended reality, as none of the participants had used this technology for machine learning development productivity or collaboration. The aim of the paper is to highlight the development of a system that uses a low-code development platform for the development of machine learning models and then uses an extended reality environment to not only enable collaboration within development teams but also as a system for presenting a model’s output. This paper documents the early phases of the research process (i.e. identifying the need) whilst also sharing ideas on how the issue can be solved.

1. Introduction

Extended reality (XR) immersive visualisations have huge potential to enhance the understanding and/or data experiences for many different sectors (i.e. health, education, engineering, manufacturing, tourism etc.) As Andrews et al. [ASSS19, p.1] points out ‘XR devices offer many advantages for medical applications including realistic 3D visualization and touch-free interfaces that can be used in sterile environments’. In the Architecture, Engineering, and Construction (AEC) industry, it is believed that XR technologies can be a tremendous help in the visualisation of all stages of a project [AHH20]. Furthermore, many feel that XR will reshape the experience of tourism [KK21]. In terms of education, it has been found that XR is an excellent technology to facilitate the teaching processes allowing learners to more actively control their learning strategies and support the interactivity and connectivity that students and faculties experience [KMS*21]. Indeed, the immersive visual nature of XR has the potential to provide an increased engagement and high knowledge retention compared to the more traditional modes of visualisation.

Moreover, many of the organisations within these sectors are making machine learning projects more of a priority. However, issues have been found regarding the presentation of these types of

projects and particularly, in explaining how the models that are produced work, not only internally but also to the final user. The main contribution of the paper lies in the documented development of a XR immersive visualisation solution. The following sections discuss a short initial feasibility study which probed people’s opinions around their current use of XR environments, low-code development platforms, and their experience of working on machine learning model development projects. It is the authors’ aim to firstly highlight the need/ rationale for a XR immersive visualisation solution. Secondly, to demonstrate how the development of an XR immersive visualisations needs to be approached in order to attempt to improve a project team’s understanding of machine learning development.

2. Background

The following research discusses the use of extended reality (XR) for both visualisation and as a tool for collaboration and how this has become more important for enterprise in the wake of the Covid-19 pandemic. It also investigates how "low-code" development platforms have also become more prevalent to enable rapid development and how this type of platform could be used for the development of machine learning models.

2.1. Extended Reality Immersive Visualisation

Extended Reality is an umbrella term used to describe immersive technologies that can merge the physical and virtual worlds [Mar19]. It is a new immersive form of visualisation which opens up possibilities for more insightful experiences and presentations of data. Rao and Dawarwadikar [RD20] describe immersive visualizations as a novel way of representing data and information through spatial computing technologies such as augmented and virtual reality. They highlight that it enhances the user perspective with enhanced insights from data and information; they believe that it can provide better visualizations over traditional methods resulting in better decision making, cost savings for organizations, enhanced experiences for consumers, and many more [RD20]. Moreover, Extended Reality can facilitate remote collaboration which, as discussed in studies by both Affendy and Wanis [AW19] and Ens et al. [ELT*19], is potentially achievable within any of the XR environment types, Virtual Reality (VR), Augmented Reality (AR), or Mixed Reality (MR):

- Virtual Reality - immerses the user in a 360-degree digital environment requiring the use of a headset [Mar19].
- Augmented Reality – the overlaying of digital objects in the real world experienced through AR glasses or a mobile device that has access to a camera such as smartphones and tablets [Mar19].
- Mixed Reality – also known as ‘Hybrid Reality’, allows the user to operate in an environment where both digital and real-world objects can interact with each other. As with VR technology this type of environment requires the user to wear a headset [Mar19].

The use of extended reality as a tool for remote working and collaboration has seen rapid growth recently. During the COVID-19 pandemic, it was reported that organisations had been pushing what were proof-of-concept XR projects, into production to assist workers in remote working [Eve20]. The global market share for XR technologies is currently estimated at 31 billion U.S. dollars, and due, in part, to this acceleration in its use, is expected to rise by more than 900% over the next three years [Als22]. An example of this change is highlighted in an interview with Julie Sweet, CEO of Accenture, who discussed how the company has been sending out tens of thousands of VR headsets to new employees during the pandemic as part of the on-boarding process to “give new employees a virtual shared experience as they started their new job” [Kid21]. Examples of such applications are currently available allowing users to take part in virtual meetings using avatars as a representation of themselves and also include functionality to enable access to their computer’s desktop, allow users to upload and display files or 3D objects or use virtual whiteboards for ideation.

2.2. Low-Code Development Platforms for Machine Learning Model Development

A method that may be considered for both accelerating and managing machine learning development is the use of a Low Code Development Platform (LCDP). As described by Marvin [Mar18] the term “Low-Code” was first coined within a Forrester’s report from 2014 in which it was described as a platform for accelerated application development via a significant reduction in hand-coding [RRM*14]. LCDPs allow users, including those with little or no

programming background to develop and deploy fully functional applications via a drag and drop graphical user interface using pre-configured data connectors, templates, widgets, etc. [SIRP20]. Since 2014 there has been a rapid increase in the development of LCDPs, in 2019 Gartner published a report stating that by 2024 over 65% of all application development will be completed using this type of platform [VIL*21]. From an evaluation of existing “Low-Code” tools used for machine learning model development including RapidMiner Studio [Rap22] and DataRobot [Dat22] as well as platforms used for managing these types of project such as the Domino Data Science Platform [Dom22], Algorithmia [Alg22] and MLFlow [Pro22] the following functionality was found to be integral to this type of platform:

- Simultaneous training and validation of multiple machine learning algorithms that can then be selected for further analysis by the user
- Functionality for recording experiment results during model development
- Version control of the models developed and any related assets

3. Feasibility Study

The feasibility questionnaire was designed to investigate participants’ use of both extended reality environments and low-code development platforms (LCDP). It also focused upon collaborative applications as well as participants’ experiences of working on machine learning model development projects.

3.1. Method

The final questionnaire consisted of a combination of both open and closed questions regarding the participant’s current use of XR environments, low-code development platforms, and their experience of working on machine learning model development projects.

Regarding the use of extended reality environments, participants were asked general questions regarding their experiences using these platforms. This included the issues/ challenges that they may have faced, and the types of applications that they have used previously within this type of environment. The questionnaire then followed up with questions, asking participants if they had ever used XR applications that enabled collaboration, data visualisation, or for productivity applications, and if so, what their experience was like. The low-code development section of the questionnaire asked participants about their experience of using this type of platform including their use cases, any challenges encountered, their experience of collaboration within the platform, and if they had any experience of this within an extended reality environment. The final section focused on machine learning model development asking questions about the importance of various aspects of model development and the challenges of collaboration on this type of project. Moreover, it probed their opinions on whether allowing teams to interact and evaluate the output/ visualisations from machine learning model development without requiring them to have any programming knowledge would be useful within a project.

3.2. Procedure

The questionnaire was distributed via the author's LinkedIn profile as well as through university channels. The aim was to get as many responses as possible and so, was made available from the week commencing 13th September until the 19th of November 2021. During this time, and in response to an initial low uptake, the questionnaire was restructured by splitting it into three separate smaller questionnaires. A new questionnaire was created for each topic using the questions from the relevant block of the full questionnaire. This change was made to attract more participants with a smaller questionnaire that was topic-specific and would take less time to complete. Again, these were distributed through the same channels as the full questionnaire.

3.3. Analysis

The following analyses participant's demographics as well as the responses regarding each of the three main topics, extended reality, low-code development platforms, and machine learning model development.

3.3.1. Participants

In total the questionnaire, including the individual topic versions, had nineteen responses. Of those participants that gave demographic information, their ages ranged between eighteen and fifty-four, with 40% of participants aged between twenty-five and thirty-four years. Only one participant identified themselves as being female.

3.3.2. Findings: Extended Reality

When asked questions regarding their experience of using extended reality environments 42.8% of participants stated that they had never used this technology. Of those that had, 33% had never used virtual or mixed reality environments but all of them had used augmented reality. All three extended reality environment types had been used by participants for gaming with augmented reality also having been used for product visualisation/walkthroughs. One other use case was mentioned in which the participant had used virtual reality to help them get over their fear of heights. There were no participants that had used extended reality for any type of productivity or collaborative applications (See figure 1).

Regarding issues that users had faced when using these technologies, for virtual reality, the comments were focused on the lack of spatial awareness and problems with nausea. For the use of augmented reality, the issues encountered were mainly due to the use of this technology on mobile phones pointing out problems with responsiveness and scale as well as the potential danger of focusing on a device and not on the surrounding area during use.

3.3.3. Findings: Low-Code Development Platforms (LCDP)

There was a total of six responses from both the full questionnaire and the standalone version that stated they had previously used a low-code development platform. The most used use case for this

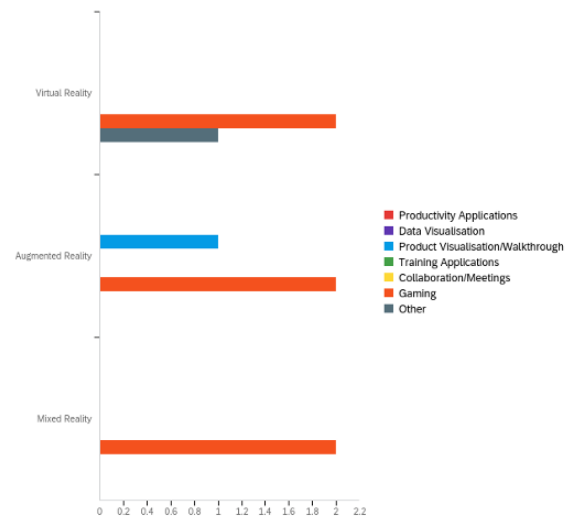


Figure 1: Which of the following types of applications/use cases have you used an Extended Reality environment for?

type of application was for web development with 75% of the participants stating they had used a LCDP for this, followed by automation and application development both with 50% of participants (See figure 2). Other uses for LCDP platforms that were noted included business application and mobile game development.

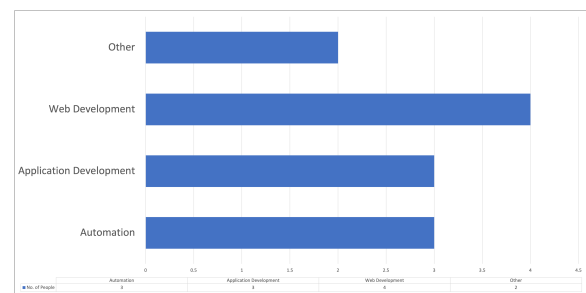


Figure 2: Which of the following use cases have you used a low-code/no-code development platform?

The participants also rated their experience of using this type of development platform, with 58.3% rating it as either extremely or very effective. There were no participants that had used this type of development platform within an extended reality environment. Three users had used collaborative functionality within LCDP's stating their ability to share between team members had made it easier for people in multiple locations to work together.

3.3.4. Findings: Machine Learning Model Development

There were only two responses to this section of the questionnaire, one from a participant stating their role within a machine learning model development project as a developer the other from a chief technical officer. As may be expected, they both had varying opinions regarding the importance of the different aspects of

model development. The aspects of machine learning model development where they both agreed upon the importance, were those of exploratory data analysis and data preparation, evaluating model performance, and internally explaining/ interpreting the model. All other aspects of development were deemed as being either extremely, very, or moderately important for both participants. The challenges raised for collaboration within teams for this type of project were regarding source control and the sharing of knowledge, with participants stating that the functionality required to improve collaboration should include:

- Something that helps to explain the choices made as a developer to those without a technical background
- A standardised output of model progress and results

When asked if an 'application that allows teams to interact and evaluate the output/ visualisations from machine learning model development without requiring them to have any programming knowledge would be useful within a project' both participants said yes. They continued by stating that it would help to explain decisions made during development. Especially, as models are often driven by business requirements from users, without the technical knowledge to understand the underlying modelling, that the proposed functionality could be used to bridge that gap.

3.3.5. Discussion

The results of the feasibility questionnaire showed that a solution that integrates extended reality with a "low-code" development platform that enables rapid development and collaboration would be deemed as a novel solution, as none of the participants had used this combination of technologies previously. This type of use case has only recently become more popular and so did not come as a surprise to the authors. The challenges related to the use of extended reality, as stated by the participants, all seem to focus on areas related to usability. Whether that's the effects of nausea and spatial awareness for VR, or the dangers of focusing on a single device such as a mobile phone instead of the surrounding area during use for AR. These issues should be considered when developing for either of these XR environment types. The use of LCDP's is popular with the participants that have used this type of platform before, with it being deemed as a very effective way of developing all types of applications. The responses from the machine learning model development section of the questionnaire went on to help define the requirements of the application in areas such as:

- Functionality for exploratory data analysis and preparation, model performance, and model interpretation
- Standardisation of the solutions output

The responses also confirmed one of the main requirements of the proposed solution in which the user, no matter their experience or technical skills, should be able to easily interact with the application and its output.

3.4. Summary

From the secondary research and feasibility study findings, the following conclusions were made that would guide the development of the solution. Regarding machine learning model development,

the solution should contain functionality for data analysis including the visualisation and preparation of the data to be used, Model training and validation, hyperparameter tuning of the model, and presentation of the final model. Other elements to be considered include explanations of the solutions functionality, data visualisations, model performance visualisations, as well as visual interpretations of a model's predictions. The visualisation elements, as well as being displayed within the solution should also be saved as part of a project's assets so that they can be referred to later if required. The Machine Learning development environment which would be developed as a 'low-code' platform, would not only assist in facilitating the required explanations of functionality but also make the solution easier to learn and allow for rapid experimentation and development of machine learning models. Regarding which Extended Reality environment would be used, from the research it was decided that a virtual reality environment would be beneficial due to the ability to create an immersive 'world'. This would allow multiple users to simultaneously interact with the environment and view the various assets that would be created as part of the machine learning model development process, but also to collaborate in both a models development and it's presentation to facilitate explainable AI (XAI). In its simplest form, the purpose of an XAI system is to make AI and ML model behaviour more accessible to its users by providing explanations for what it has done, what it is doing, what happens next and the information it is working from [GSC*19].

4. Design and Development of XR solution

The following looks at the requirements defined for the solution, it's initial design as a high-fidelity prototype and then it's development in the form of both a low-code development platform and a VR environment that can be used to present the stages of a machine learning models development and explanations on the output of a model based upon it's input.

4.1. Solution Requirements

From the research undertaken as well as the findings from the feasibility study questionnaire, the following requirements were defined for the proposed solution:

- The extended reality environment type to be used would be Virtual Reality (VR). Functionality for a desktop version of the environment should be considered for users that are uncomfortable within VR environments.
- All processing of the data and model development would take place in a cloud-based application that could be mirrored within the virtual reality environment.
- The application would be developed as a low-code development platform.
- The application would allow the user to develop either regression or classification models.
- The main functionality for machine learning model development would be split into three sections:
 - Data analysis/preparation.
 - Model development.
 - Explainable AI.
- The elements required for Explainable AI should include:

- Explanations of all functionality within the application
 - Explanatory visualisations of the data used for modelling
 - Visualisations used for model performance analysis
 - Visualisation methods used for interpretation of a models output
- Functionality should be available to the user for saving the output of the application such as visualisations, datasets, and the final model as well as any other project files.
 - The application should log all the user's actions within the application.

It should be noted that the solution does not include functionality for the development of machine learning models using artificial neural networks, also known as deep learning, at this time.

4.2. High-Fidelity Prototype

One of the stages in the solutions design was the development of a high-fidelity prototype of the VR environment. This was used to define how a project would be displayed and how the user would interact with the various elements required to develop, analyse and explain a machine learning model. The prototype was developed using Adobe XD, a User Interface (UI)/ User Experience (UX) design application. Within the prototype the environment would be broken up into five distinct sections:

- Project Detail
- Data Analysis
- Modelling
- Explainability (Explainable AI)
- Whiteboard

4.2.1. Project Detail

Accessing the 'Project Detail' panel would display general information regarding the dataset being used and the machine learning model that had been developed. This includes basic dataset information such as file type, file name, number of records/rows in the dataset, and number of attributes. The panel also showed model development information including the model type, target attribute used from the dataset, the name of the best performing algorithm, and if that model had been hyper-parameter tuned (See figure 3).

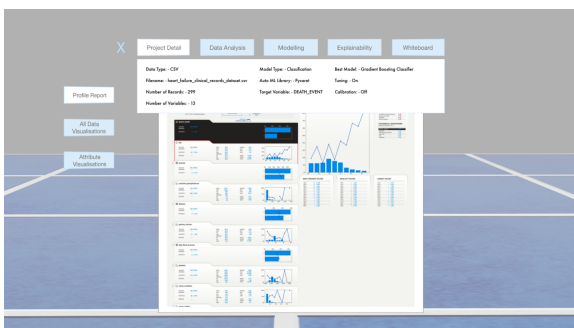


Figure 3: High fidelity prototype 'Project Detail' panel

4.2.2. Data Analysis

The 'Data Analysis' section would include options for three views, profile reports, all data visualisations, and attribute visualisations. At this stage of the design process. The 'Profile Report' section would contain data reports generated from the original dataset. It was decided to have two different reports that would be generated that visualise the dataset in different ways giving the user the option to select which they prefer depending on their requirements. The 'All Data Visualisations' section would display all of the visualisations created for the given dataset which can be individually selected by the user (See figure 4). These would include:

- Scatter plots
- Pair-wise scatter plots
- Normal distribution plots
- Probability plots
- Box plots
- A correlation heat-map
- Bar plots

The 'Attribute Visualisation' section would display visualisations related to user-specified attributes of the dataset. Using a dropdown menu, the user could select the attribute(s) that they are interested in, which would start the generation of a number of recommended visualisations based upon that selection.

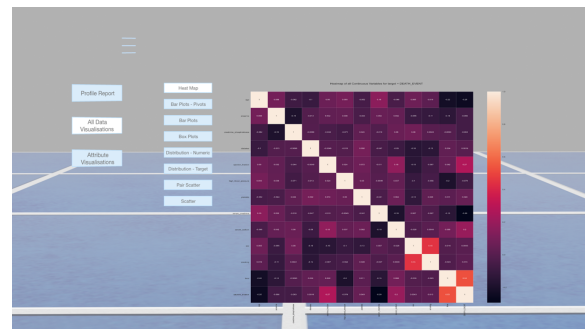


Figure 4: High fidelity prototype 'Data Analysis' visualisations

4.2.3. Modelling

Earlier in the design process it was decided to use an Automated Machine Learning (AutoML) framework to enable model development within the solution. PyCaret [Pyc22], an open-source python library, was selected for use in this project and it's output would be used to define the layout within this section. This would be split into four views enabling the user to set up the AutoML environment, analyse the data used for training the model, select and analyse the performance of various model algorithms, and review predictions made with the final model. The 'Environment' section displays the setup for the PyCaret autoML library including information about the training and validation datasets created after pre-processing has been completed. The user can also make changes to those settings from within this section or reset them back to the default. The 'Training Data' section is used to display a data report generated from the data used for training. The 'Model Selection' section would be split into three further views, Compare, Tuned

Model and Evaluation. The ‘Compare’ section displays the leaderboard generated during the automated model algorithm training and validation process. The table shows each algorithm that has been trained and its related performance metrics, depending on the type of model being developed.

- Classification – Accuracy, Area Under Curve (AUC), Recall, F1, Kappa, Matthews Correlation Coefficient (MCC)
- Regression – Mean Absolute Error (MAE), Mean Square Error (MSE), Root Mean Square Error (RMSE), R2, Root Mean Squared Log Error (RMSLE), Mean Absolute Percentage Error (MAPE)

From this section the user could also change the model algorithm to be taken forward in the development process, although it should be noted that this would take a significant period to be processed depending on the size of the dataset so users may wish to leave the VR environment and return once completed. The ‘Tuned Model’ section displays a table showing the performance metric scores from each iteration of the hyper-parameter tuning process of the final model. The ‘Evaluation’ section displays all the model performance analysis visualisations generated from the final model. The final modelling section, ‘Predict’ displays tables showing the predictions made by the final model using both the modelling test data and data that has not been previously used during model development. For classification models, a confusion matrix is also displayed showing the user the number of correctly predicted values as well as the number of false positives/negatives.

4.2.4. Explainability: Explainable AI

The ‘Explainability’ section displays visualisations generated using SHAP [Lun18]. SHAP or SHapley Additive exPlanations, is a game-theoretic approach that can be used to explain the predictions made by any machine learning model. SHAP works by assigning an importance value to each attribute used within a prediction made by a model which can then be used to visualise that prediction. Within this section, the user can view a Summary Plot as well as select a Force Plot related to the predictions made using data that has not been used during model development. This enables the user to analyse the effects of attributes on the model including in instances where the model has made incorrect predictions (See figure 5).

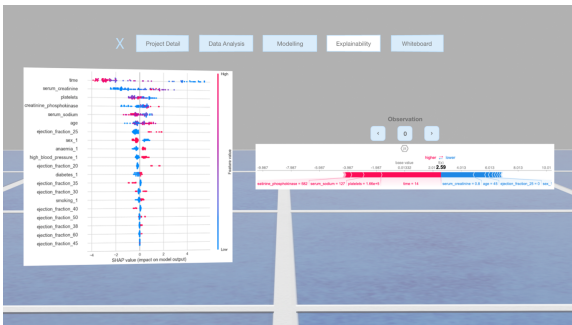


Figure 5: High fidelity prototype ‘Explainability’

4.2.5. Whiteboard

Within the prototype there is also a view for the ‘Whiteboard’ section. This was to be a meeting area that allowed the participants to make notes and discuss the project. This was not defined further at this stage and would be considered as part of the future development of the VR environment.

4.3. The Low-Code Machine Learning Development Web Application

As defined in the requirements, the solution would be split into two parts. A web application that would be used to initiate projects and be responsible for all data processing, model training and validation, and a VR environment that would mirror the web applications output. The web application was developed using Streamlit [Str22], another open-source Python library, that enables the creation of custom machine learning and data science web applications that can be run from a local server or deployed to web application services such as AWS (Amazon Web Services), Heroku, Azure, Google App Engine, etc. Much of the web applications functionality was based upon the design from the high-fidelity prototype but now included additional explanations for each element. As the user was making selections that would enable the development of a machine learning model, visualisations and text would be generated describing what was happening as part of the explainable AI functionality of the application.

The final application would be split into sections that walks the user through the machine learning model development process including setting up a project, performing exploratory data analysis, development of a machine learning model and analysis of it’s performance and finally visually interpreting it’s output (See figure 6). Each step of the process would also be logged and all of the output saved, not only so that it could be visualised within the VR environment but also so that it could be used again not only to continue development within the application but also as part of future projects/development. the assets created by the application for each project include:

- The data visualisations and reports generated during ‘Data Analysis’
- Copies of the datasets used by the application, in CSV format, including:
 - The original uploaded dataset
 - A ‘cleaned’ version of the original dataset, removing attributes containing all unique values, empty and single-valued attributes and empty and duplicate rows.
 - After the ‘Data Analysis’ is completed the data is split, 90% to be used for training and validation of the model and the remaining 10% kept back as unseen data for model predictions, copies of both sets are saved
 - A copy of the unseen data including predictions made by the final model
 - A copy of the comparison table of all the models that have been trained and cross-validated during model development
- A comparison data report of the data used for training and testing of the model

- Model performance analysis visualisations including, but not limited to, Learning Curve, Feature Importance, Classification Reports, Residuals Plot, etc.
- The finalised model (as a .pkl file) that can then be used for further testing and integration into other applications
- All SHAP visualisations generated from the eXplainable AI section related to a models output
- The full project log file



Figure 6: Web Application - SHAP Force, Explainer and Summary Plots for a Classification model

4.4. Virtual Reality Environment

In the high-fidelity prototype, the VR experience was seen as an open space with a floating UI that displayed the output from the web application. Given one of the disadvantages of working within a virtual reality environment is motion sickness it was decided that having this open space environment without any recognizable boundaries may exacerbate the problem for some users. Motion sickness within a VR environment is caused by conflicting messages sent to the brain whereby the eyes are seeing the movement through the environment, but the inner ear does not sense the same motion [Gor21]. The effect can be mitigated by following defined VR “rules” [For13] including:

- Not changing the users view without their input
- Not changing the user’s field of view
- There should be no violent or unprovoked movements within the view
- The software should never ignore the users head movements

Also, by creating a recognizable environment that the users can interact with and move around, it was felt that this also improved the collaborative experience. To facilitate this change it was decided to create a ‘virtual lab’ consisting of two rooms containing four distinct working areas (See figure 7). Each of these working areas contain the assets generated by the web application for each stage of a model’s development, data analysis, model development, and explainable AI, as well as access to the web application so that users can make changes to a project as required.

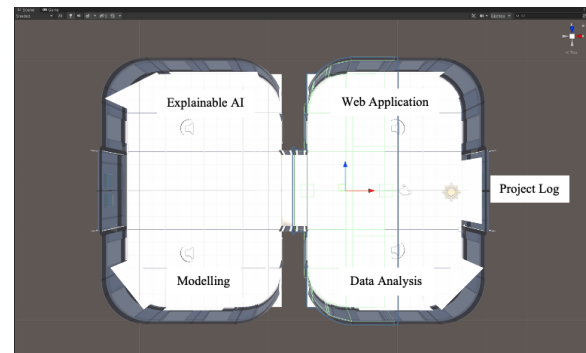


Figure 7: VR Environment Layout

As all the assets generated by the web application are either image or HTML files a method had to be found to integrate them into interactive displays for each of the working areas. This was achieved by developing HTML files that both managed content as well as including UI elements that are required to enable user interaction. A total of eleven HTML files were developed to manage the content to be displayed within the VR environment covering the three development stages. These files were developed using Bootstrap [Boo22], an open-source front-end development framework used for developing responsive websites and web applications which includes many pre-built JavaScript based components for user input and controlling layout. Although simple in their construction, each of these files included a description of what the user was viewing as well as user input controls such as drop-down menus or the Carousel component for image slideshows. These were included so that the user could interact with the content being displayed.

4.4.1. Developing the Virtual Lab

The first iteration in the development of the VR environment consisted of creating the virtual lab and setting up the web browser elements (See figure 8). Another consideration made during the environment’s development was that it should also be accessible as a desktop application for those that do find using VR difficult. To allow for this the environment would be the same in both versions of the application but changes to both the build configuration settings of the application and the camera setup would be required to enable first-person controls using a mouse and keyboard.

4.4.2. Improving the User Experience

The second and final iteration of the development of the VR environment was concerned with improving the user experience (See

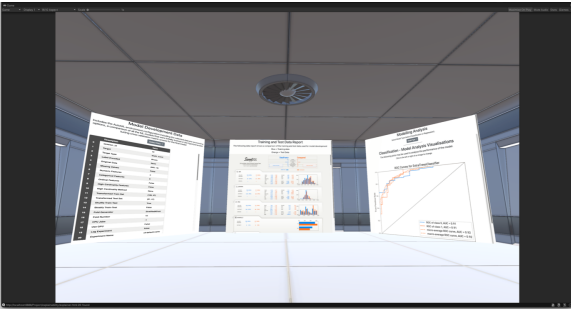


Figure 8: VR Environment - 'Modelling' Section First Iteration



Figure 10: VR Environment - 'Modelling' Section Second Iteration

figures 9 and 10). It was felt that in the first iteration that the environment was sterile in its appearance and lacked context, especially with the displays unrealistically floating in the air. A study by Hvass et al. [HLV*18] found that by increasing the visual realism of a VR environment users would report a stronger presence and would respond more realistically. To rectify this, the following changes were made to the environment:

- New lighting was added to the scene including both ceiling and wall lights
- The displays were added to 3D models of screens and moved to appear as if they are fixed to the walls
- Other decorative elements were added to the scene including plant pots, bookcases, and ornaments

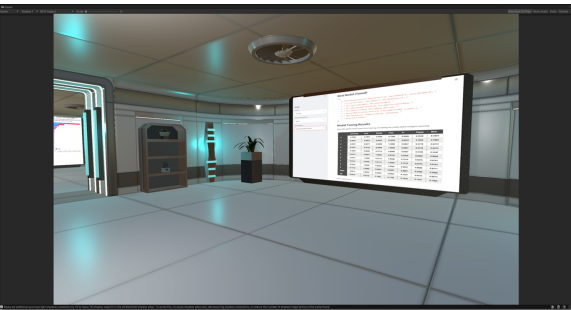


Figure 9: VR Environment - 'Web Application' Section Second Iteration

5. Future Development

There are areas in which the final solution could be improved if taken forward, including the following additional functionality to be considered for the web application:

- Development of additional model types such as anomaly detection, topic modelling, and time series analysis as well as the existing classification and regression.
- The integration of other autoML libraries such as H2O, as well as introducing neural network algorithms
- Code files should be generated as part of the project download

so that experiments can be recreated. For example, the code generated from the data analysis stage could be used to prepare new data for further testing

- Additional functionality to be added to the data analysis section to allow users to develop/run their own code for creating new attributes or new visualisations.
- More granular levels of project logging and version control of output assets to be broken down by experiment.
- Development of a database that would be used to store project settings and the files generated by the application.

Additional features that could be made to enhance the VR environment include:

- Functionality for improving collaboration such as creating avatars to represent users and speech-to-text for creating transcripts of the interactions within the environment
- Adding the whiteboard section, as seen in the high-fidelity prototype, a meeting area that allows participants to make notes and discuss the project.

6. Conclusion

It is the authors' opinion that the working XR solution (presented in this paper) holds huge potential for enhancing the understanding of the machine learning development process. As highlighted in the feasibility study, there is a need for new ways to improve the presentation / explanation of what is happening with AI algorithms (i.e. to open up the black box of machine learning). The next steps for the project team are to undertake a detailed evaluation study of the current XR solution. Particularly, to understand what additional features could be added for the data analysis section (i.e. to ensure that the user has more options for configuring data, creating new attributes and fully presenting what is happening). The aim is to achieve this within a VR environment that multiple users can use simultaneously to discuss and present the output from the development of a machine learning model. As the pace of machine learning accelerates, it will become more and more crucial for society to be able to maintain control and understanding of what it is doing. As the authors of this paper have shown, XR and its scope for immersive visualisations might just have some of the answers.

References

- [AHH20] ALIZADEHSALEHI S., HADAVI A., HUANG J. C.: From bim to extended reality in aec industry. *Automation in Construction* 116 (2020). doi:10.1016/j.autcon.2020.103254. 1
- [Alg22] ALGORITHMIA: Product - algorithmia, 2022. URL: <https://algorithmia.com/product>. 2
- [Als22] ALSOP T.: Extended reality (xr): Ar, vr, and mr - statistics facts | statista, 2022. URL: <https://www.statista.com/topics/6072/extended-reality-xr/>. 2
- [ASSS19] ANDREWS C., SOUTHWORTH M. K., SILVA J. N., SILVA J. R.: Extended reality in medical practice, 2019. doi:10.1007/s11936-019-0722-7. 1
- [AW19] AFFENDY N. M. N., WANIS I. A.: A review on collaborative learning environment across virtual and augmented reality technology. vol. 551. doi:10.1088/1757-899X/551/1/012050. 2
- [Boo22] BOOTSTRAP: Bootstrap · the most popular html, css, and js library in the world., 2022. URL: <https://getbootstrap.com/>. 7
- [Dat22] DATAROBOT: Datarobot ai cloud - the next generation of ai, 2022. URL: <https://www.datarobot.com/>. 2
- [Dom22] DOMINO: Domino data lab | unleash data science at scale, 2022. URL: <https://www.dominodatalab.com/>. 2
- [ELT*19] ENS B., LANIR J., TANG A., BATEMAN S., LEE G., PIUMSOMBOON T., BILLINGHURST M.: Revisiting collaboration through mixed reality: The evolution of groupware. *International Journal of Human Computer Studies* 131 (2019). doi:10.1016/j.ijhcs.2019.05.011. 2
- [Eve20] EVERETT C.: Extended reality software gets pandemic lift, 2020. URL: <https://www.computerweekly.com/feature/Extended-Reality-software-gets-pandemic-lift>. 2
- [For13] FORSYTH T.: Vr sickness, the rift, and how game developers can help, 2013. URL: https://developer.oculus.com/blog/vr-sickness-the-rift-and-how-game-developers-can-help/?locale=en_GB. 7
- [Gor21] GORDON W.: How to reduce motion sickness in virtual reality | wired, 2021. URL: <https://www.wired.com/story/how-to-reduce-motion-sickness-virtual-reality/>. 7
- [GSC*19] GUNNING D., STEFIK M., CHOI J., MILLER T., STUMPF S., YANG G. Z.: Xai-explainable artificial intelligence. *Science Robotics* 4 (2019). doi:10.1126/scirobotics.aay7120. 4
- [HLV*18] HVASS J., LARSEN O., VENDELBO K., NILSSON N., NORDAHL R., SERAFIN S.: Visual realism and presence in a virtual reality game. vol. 2017-June. doi:10.1109/3DTV.2017.8280421. 8
- [Kid21] KIDWAI A.: Lessons from onboarding over 100,000 employees during the pandemic | fortune, 2021. URL: <https://fortune.com/2021/10/12/lessons-from-onboarding-over-100000-employees-during-the-pandemic/>. 2
- [KK21] KWOK A. O., KOH S. G.: Covid-19 and extended reality (xr), 2021. doi:10.1080/13683500.2020.1798896. 1
- [KMS*21] KULETO V., MILENA I. P., STANESCU M., RANKOVIĆ M., ŠEVIĆ N. P., PÄUN D., TEODORESCU S.: Extended reality in higher education, a responsible innovation approach for generation y and generation z. *Sustainability (Switzerland)* 13 (2021). doi:10.3390/su132111814. 1
- [Lun18] LUNDBERG S.: Welcome to the shap documentation — shap latest documentation, 2018. URL: <https://shap.readthedocs.io/en/latest/index.html>. 6
- [Mar18] MARVIN R.: The best low-code development platforms, 2018. URL: <https://uk.pcmag.com/migrated-46739-onlinecloud-backup-services/89789/the-best-low-code-development-platforms>. 2
- [Mar19] MARR B.: What is extended reality technology? a simple explanation for anyone, 2019. 2
- [Pro22] PROJECTS L.: Mlflow - a platform for the machine learning life-cycle | mlflow, 2022. URL: <https://mlflow.org/>. 2
- [PyC22] PYCARET: Home - pycaret, 2022. URL: <https://pycaret.org/>. 5
- [Rap22] RAPIDMINER: Predictive analytics software | rapidminer studio, 2022. URL: <https://rapidminer.com/products/studio/>. 2
- [RD20] RAO M., DAWARWADIKAR M.: Immersive visualizations using augmented reality and virtual reality, 2020. doi:10.1007/978-3-319-08234-9_395-1. 2
- [RRM*14] RICHARDSON C., RYMER J. R., MINES C., CULLEN A., WHITTAKER D.: New development platforms emerge for customer-facing applications. *Forrester: Cambridge, MA, USA* (2014). 2
- [SIRP20] SAHAY A., INDAMUTSA A., RUSCIO D. D., PIERANTONIO A.: Supporting the understanding and comparison of low-code development platforms. doi:10.1109/SEAA51224.2020.00036. 2
- [Str22] STREAMLIT: Streamlit • the fastest way to build and share data apps, 2022. URL: <https://streamlit.io/>. 6
- [VIL*21] VINCENT P., IJIMA K., LOEW A., AKISH J., WONG J.: Magic quadrant for enterprise low-code application platforms, 2021. 2