August 26. 2023

C964: Computer Science Capstone

Task 2 parts A, B, C, and D

[Part A: Project Proposal for Business Executives](#_heading=h.gjdgxs) 3

[Letter of Transmittal 3](#_heading=h.30j0zll)

[Project Recommendation](#_heading=h.1fob9te) 4

[Problem Summary](#_heading=h.3znysh7) 4

[Application Benefits](#_heading=h.2et92p0) 4

[Application Description](#_heading=h.tyjcwt) 5

[Data Description](#_heading=h.3dy6vkm) 5

[Objectives and Hypothesis](#_heading=h.1t3h5sf) 6

[Methodology](#_heading=h.4d34og8) 6

[Funding Requirements](#_heading=h.2s8eyo1) 7

[Data Precautions](#_heading=h.17dp8vu) 7

[Developer’s Expertise](#_heading=h.3rdcrjn) 7

[Part B: Project Proposal](#_heading=h.lnxbz9) 8

[Problem Statement](#_heading=h.35nkun2) 8

[Customer Summary](#_heading=h.1ksv4uv) 8

[Existing System Analysis](#_heading=h.44sinio) 9

[Data](#_heading=h.2jxsxqh) 9

[Project Methodology](#_heading=h.z337ya) 10

[Project Outcomes](#_heading=h.3j2qqm3) 10

[Implementation Plan](#_heading=h.1y810tw) 11

[Evaluation Plan](#_heading=h.4i7ojhp) 11

[Resources and Costs](#_heading=h.2xcytpi) 12

[Timeline and Milestones](#_heading=h.1ci93xb) 13

[Part C: Application](#_heading=h.3whwml4) 14

[Part D: Post-implementation Report](#_heading=h.2bn6wsx) 17

[A Business (or Organization) Vision](#_heading=h.qsh70q) 17

[Datasets](#_heading=h.3as4poj) 18

[Data Product Code](#_heading=h.1pxezwc) 19

[Objective (or Hypothesis) Verification](#_heading=h.49x2ik5) 20

[Effective Visualization and Reporting](#_heading=h.2p2csry) 20

[Accuracy Analysis](#_heading=h.147n2zr) 22

[Application Testing](#_heading=h.3o7alnk) 23

[Application Files](#_heading=h.23ckvvd) 23

[User Guide](#_heading=h.ihv636) 24

[Summation of Learning Experience](#_heading=h.32hioqz) 25

# **Part A: Project Proposal for Business Executives**

## Letter of Transmittal

Subject: Enhancing product decision-making with ESRB predictions

Date: 09/09/2023

Dear American Video Game Company’s leadership,

I am writing to you to address a problem and solution regarding our company’s ability to properly predict the ESRB ratings of our video games given their content. Without the ability to properly predict the rating of our product, we cannot make decisions about who our target demographic will be for the games we release, nor understand how the content will affect the final rating given by the Entertainment Software Rating Board. I believe to have a software solution which implements machine learning which will enhance our ability to predict the ESRB ratings for our products based on the content of the video game.

My solution is a data-driven tool named “ESRB Predictor”. This program uses machine learning technology to predict the ESRB rating of a video game based on its content attributes. The benefits of this data product for the organization include enhanced decision-making, better resource optimization, and a more competitive edge against our competitors. The program’s ability to predict the ESRB rating of our product provides insight in how we will market and distribute our product, ensuring it aligns with our intended target audience. Having the ability to evaluate the ESRB rating while a game is in development will also cut down on resources and reduce delays due to unexpected content-related issues. Using data-driven predictions will allow our games to more closely align with the preferences of our target audiences, giving us an advantage against other video game publishers.

The total cost estimate for this solution will include the cost for development, documentation for training purposes, and implementation with existing systems in the company. Depending on the amount of resources allocated for this project’s completion, the cost and timeline will vary, but can be estimated between $5,000 - $10,000.

My experience with this matter and my education from WGU make me an ideal candidate to develop this solution. I have a more than capable understanding of machine learning and the Python development language. Through my skills I will be able to develop this ESRB predictor and ensure its effectiveness in exceeding our business needs.

Sincerely, Cameron Woodard

Computer Science major

## Project Recommendation

### Problem Summary

I am proposing the development of an ESRB (Entertainment Software Rating Board) rating predictor for the video games developed by the company. The software will provide a user-friendly interface for allowing the company to predict the content rating of our games based on the content within them. Understanding the content rating our games will have while in development will be crucial in figuring out the target demographic, and better help us understand our marketing approach for the game. Providing insight into the potential ESRB rating will enhance our marketing strategies and ensure that the games we develop comply with content guidelines. Undergoing this project will deliver the company a software application which can be used to predict the rating of a game based on selected content on the user interface. This application also provides data visualizations such as the ESRB rating distribution of the dataset used to develop the program, a confusion matrix heatmap of the test data, and a pie chart showing the distribution of violence among the dataset. An achievement of this project is the informed decision-making that can be made in the marketing and distribution of our products. This project will also achieve better compliance with the ESRB and minimize the risk of unfavorable ratings that might limit our audience capture.

### Application Benefits

The project for a machine learning ESRB prediction program meets the needs of the business by providing accurate rating predictions, data-driven decision making, and better resource allocation. By employing a decision-tree algorithm to train our machine learning program, the model is able to predict the ESRB rating that a game is likely to receive based on its contents. This information allows senior leadership to understand how the content has an effect on the rating, and allows them to make better-informed decisions on the marketing and further development of our games. Being able to adjust the content used in the friendly user-interface allows leadership to see how choices can affect the ratings of our products. Resources can be better allocated from the development of this project, as the marketing team will have a better understanding of just who we should market our product in development towards. For example, a video game in development that will likely receive the rating “Mature” should be marketed towards adults and those over 17 years of age. This optimization of resources is also extended to the development team, which will tailor the game better to match the anticipated rating.

Benefits for the business upon implementing this solution include more informed decision making, better compliance assurance, and enhanced long-term planning. Allowing leadership to predict the ESRB rating for the games enhances their ability to make informed decisions going forward and create better marketing strategies for the intended audience. Understanding the rules of ESRB’s rating system ensures that the development and marketing of our products is in line with industry standards and regulations. Knowing the most likely ESRB rating also allows the company to plan ahead and deliver a variety of products that can be catered to audiences of all age groups.

### 

### Application Description

The proposed application for predicting the ESRB rating of the company’s video game products uses machine learning and a user-friendly graphical interface to make it easy for anybody to use. Data on video games within the industry was collected and used to train the decision-tree model in categorizing which ESRB rating should be given for a game based on its contents. Relevant content (ex: violence and crude humor) found in video games was converted into numerical boolean values that the decision tree classifier was able to understand. It uses this data to learn and discover patterns between the content itself and the ESRB ratings it was assigned. The decision tree’s performance was then evaluated using the test dataset to discover its accuracy and ability to perform the predictions. Other features of the program were fed both the main and test datasets in order to create data visualizations using the *matplotlib* Python library. These visualizations provide at-a-glance information for business leadership to better understand the technicalities of the machine learning model.

### Data Description

The data used to train the model was sourced from Kaggle, an online data science competition platform and community. The uploader set up the dataset with 1895 video game titles and 30 ESRB content rating variables to be used for the project. The dataset uses both nominal and quantitative data types: nominal for the ESRB rating itself (E for everyone, ET for everyone 10+, T for teen, M for mature) and quantitative for the boolean variables (0/1) indicating whether a game contains that specific content feature. The data is organized in a comma-separated format where each attribute’s value is separated by a comma. The dependent variable of the data is the ESRB rating of the video game itself, which depends on the content of the video game. The game’s content is the independent variable, and is different for each video game.

Since the data consists primarily of binary indicators of zero or one, the presence of any outliers in the data is limited. This limitation to binary values does not take into consideration the genre or release time for a specific video game, which limits the predictive ability of the program to solely the content itself. Another limitation of the data could be the imbalance of ratings. For example, the dataset may contain an abnormally larger amount of “Teen” games in comparison to other categories, and thus those categories might not receive enough input to fully train the model. The data does not also account for changes in ESRB rating criteria over time; certain themes may be rated differently in context to when the game was released.

### Objectives and Hypothesis

The desired outcome of this project is to develop a program with accurate ESRB rating predictions. It also seeks to empower leadership with insights to how different content elements affect the ESRB ratings of our products. Understanding the rating allows for better compliance and a better understanding of the target audience for each individual release. A hypothesis for the project could be as follows: “The presence or absence of specific content features in a video game is correlated with its ESRB rating. Analyzing these content indicators will allow us to predict the ESRB rating for our products by using machine learning.”

The desired prediction accuracy of the program would be at least 80%. This represents a strong correlation between the content features and the ESRB rating it is assigned. A prediction this certain means we can use the machine learning model to make confident business decisions. The accuracy of the model depends on the dataset itself and the machine learning model’s capabilities.

### Methodology

Since the requirements of the project are known upfront, I believe the waterfall approach to develop the project is most appropriate. The benefits of using waterfall over agile are that it will provide a clear roadmap for the project, as well as a predictable timeline for estimating the project’s completion date. First, requirements will be gathered for the application including features such as data visualizations and user interface functions. In the design phase, the GUI will be developed and the structure of the machine learning model will be planned accordingly. Implementation and development of the GUI will be done using the *tkinter* Python library, and the machine learning model will be created using *scikit-learn*. Unit testing will be done when both the UI and machine learning model are created, ensuring the ESRB rating prediction system works against the test dataset. The final project will then be packaged in the deployment phase for distribution. Documentation will then be created, including a user guide so that non-technical leadership are able to use the program. Additionally, training can be given to end-users and maintenance can be done on the project if any issues or bugs arise.

### Funding Requirements

The project’s funding requirements include paying for the development environment, the personnel’s paychecks, licensing fees, and development tools. The development environment includes paying for suitable hardware for the developers, including computers that are capable of managing the processing and memory needed for the software IDE and testing the program. The project itself requires skilled developers for designing and implementing the application, including a user experience designer and quality assurance testers. Licenses for the software must be allocated in the budget, so the software can be used by the business. Tools for the project that must be budgeted include the development software, machine learning libraries, and any version control system used to develop the project. For consistency’s sake, we can assign a budget for each of these to $1,500 and adjust them as the project is further realized, bringing the base amount of funding needed to $5,000.

### Data Precautions

The data used to train the machine learning model is made available to the public through Kaggle.com, and no protection requirements are needed. Protection of the business’s data and the development of the project must be kept secret from competing businesses, however.

### Developer’s Expertise

My expertise in the computer science field makes me a capable candidate for the development of this machine learning program. My years of coding and education through Western Governors University has given me the knowledge to work with Python and develop a program that uses machine learning to solve business problems. What I have learned in GUI development using the *tkinter* library will allow me to create a user-friendly interface that senior leadership can use to navigate the program with ease. Experience using *scikit-learn* will allow me to develop a machine learning model using a large dataset that will be able to accurately predict the ESRB rating for a video game given its content. I have learned the best practices in coding, and understand that documentation is necessary in creating a robust and maintainable program. My project management courses have taught me skills to oversee the project and manage the resources necessary to meet the milestones in the project. These skills and qualifications mean I can lead the production of this project, on time and on budget.

# **Part B: Project Proposal**

## Problem Statement

In the competitive and growing video game industry, it is important to evaluate the influence of the ESRB rating for a new product. Understanding this rating is critical to making informed decisions on how to move forward on these products, and establish better marketing strategies that will enhance the market share of the company. Given the many factors that go into the ESRB rating assigned to a video game, the process of predicting the rating can be rather complex and requires an accurate solution. This project proposal aims to develop a tool leveraging machine learning to predict the ESRB rating of our products based on selected content features. This will empower leadership to make more informed decisions for marketing and understanding our products.

## Customer Summary

The client in the context of this project is the leadership team of the organization, including professionals responsible for making critical decisions in the video game’s development and marketing. They are tasked with understanding the dynamic gaming industry, and ensuring that the company’s products are in line with market demands and comply with ESRB regulations. My application aims to resolve the problem stated above by utilizing data-driven decision making, a user-friendly interface, and data visualizations. By utilizing the power of machine learning and data analysis, the application delivered by the project will provide accurate predictions of ESRB ratings. This allows leadership to make informed decisions based on data rather than plain intuition. Understanding the diversity of expertise within leadership is what will lead to the development of an intuitive graphical user interface (GUI). This will empower even non-technical members the ability to interact with the program easily, and obtain ESRB rating predictions. Data will be made visualized in order to help leadership understand what data was used to develop the model, as well as provide insights in how certain content affects the rating of their video game products.

## Existing System Analysis

At present, our organization’s leadership relies on their expertise and industry knowledge to predict the ESRB ratings of our products. The technological environment involves manual evaluation and discussions based on content elements and industry trends. While these methods have proved useful, there are some limitations to their prediction accuracy. These shortcomings include subjectivity, limited data utilization, and lack of scalability. Currently, estimating the ESRB rating of a game has room for interpretation and bias among the leadership team, leading to inconsistencies on what rating the currently developed game will obtain. Without a data-driven machine learning tool, the company is not capitalizing on the value of historical data that is present in the gaming industry for predicting the ESRB ratings. Finally, as more games are developed by the company, the demand for establishing the maturity rating of our games increases. It is resource-intensive to manually have employees study and evaluate our products in order to provide an estimation of who the target market is going to be for all the games currently in development by the company. Automating this process with a program that can take advantage of a large dataset to accurately predict the ratings of our products will give our business an upper-hand in the video game industry.

## Data

The raw dataset contains rows of over 1,800 video game products, with each column providing information about specific content features related to the listed title. The first column contains the name of the game, and subsequent columns are in binary form indicating the presence of certain content elements included in the game. The final column contains the rating given to the game by the ESRB. The data was collected from Kaggle.com, an online data science platform. During the development phase, the data will be processed by converting it into a *.csv*  format that will be better understood by the application. The data will then be split into training and testing datasets to be used in the decision tree machine learning model. As addressed earlier, outliers in the data are not an issue due to the specific nature of the dataset (the ESRB rates games based on specific categories). If data anomalies should become present though, they will be handled accordingly.

## Project Methodology

The waterfall methodology I plan to use for development of my application is a well-established approach involving sequential progression through defined phases. Only once one phase completes may the project progress into the next one. During the requirements phase, the project’s scope and objectives will be obtained in order to build a blueprint for the entire project. In the design phase, the user interface of the program as well as the machine learning component will be planned and designed based on the requirements gathered. Afterwards, the implementation phase will begin and development of the application will be done, including the GUI and machine learning components. The testing phase will be used to verify that the application runs correctly, and the program is able to predict the ESRB ratings based on the content features of the video games. The deployment of the project involves providing leadership the application through the organization’s internal web services. Through maintenance and support, leadership will be trained on how to use the program and feedback will be obtained to make improvements on the program. This structured approach ensures that each phase is completed before moving on to the next, and ensures that all requirements will be met by the project’s completion.

## Project Outcomes

The main deliverable created over the course of this project is the finished application which allows leadership to input the contents of a video game and receive a prediction of its future ESRB rating. A user-friendly Graphical User Interface (GUI) will allow for easy interaction with the machine learning program. Other features of the program include three data visualizations that can be used to better understand the dataset used to train the machine learning model.

A user guide will be provided to assist leadership and other users in using the application effectively. This includes step-by-step instructions to navigate the program, select features of the game, and interpret the ESRB predictions made by the program. The guide will also feature troubleshooting tips in the instance that the program does not function as it should.

Another deliverable given by this project is the ongoing support of the program, and training sessions which may be given so that leadership is made proficient in the use of the application. Timely response will be given to user inquiries and help tickets, so that leadership can get back to using the program effectively. Demonstrations will be given on how to install and interact with the program on the user’s devices.

## Implementation Plan

The implementation of the ESRB rating prediction application will combine software development best practices with the Waterfall strategy. Our goal is to deliver a user-friendly and effective application that will be able to predict the ESRB rating of a game based on selected content features. Strategies that will be used in this project include rigorous requirements analysis, thorough design and planning, and development and testing phases to bring the project into fruition. Phases of the rollout include the Waterfall phases of requirements gathering, design, implementation, testing, deployment, and maintenance and support. Dependencies for the project include the sourcing of the data as well as the technical resources used to develop the program. The data must be made available to be collected, processed, and integrated into the machine learning program. The technical resources include skilled developers as well as the actual machines that the program will be developed on. Testing of the program will include unit testing on individual components of the program, as well as user acceptance testing which will be completed with leadership at the company. A deployment plan will be developed in order to distribute the program amongst the users, and access control will be implemented to ensure only authorized users have access to the application.

## Evaluation Plan

Before the project begins, requirements analysis on the project will be completed with leadership and other stakeholders to ensure that requirements are complete and well-defined. In the design phase, the proposed design will be validated with stakeholders to ensure that the design of the program meets the project objectives. During implementation, the code will be reviewed to ensure that it adheres to coding best-practices and fits the design specification. Unit testing will also be done to ensure that every individual unit works correctly and fits with the whole. In testing, the application’s features will be checked that they function correctly and pass the user acceptance testing. This is the last time to correct any defects, before the project moves on to deployment. Once deployed, leaderships and user's machines will be checked to ensure that the program works correctly in their operating environments.

Once completed, the project will receive a validation process to ensure that the application meets its intended objectives and fulfills the goal of aiding leadership to predict the ESRB rating of our products. By having leadership actively engage with the program, we can conduct user acceptance testing to ensure that the program aligns with their requirements and fulfills their needs. The scalability and responsiveness of the program will be tested in a series of performance tests, ensuring it will continue to run effectively under different scenarios. Documentation review will be conducted on the technical documents of the program, and training materials will be reviewed to make sure that they can be effectively used to train users on how to use the ESRB predictor program.

## Resources and Costs

| **Item** | **Description** | **Cost** |
| --- | --- | --- |
| Development hardware | System capable of developing the program | $800 - 1,000 |
| Server | Server to host downloads of the program | $1,000 - 1,500 |
| IDE software license | A license for the IDE to be used to develop the program | $250 - 500 |
| Operating system licenses | Fees for the operating system of both the development hardware and server | $100 - 200 |
| Software engineers for development | Labor paid for the development of the program | $1050 - 2000 |
| QA/Testing team paychecks | Labor paid to those testing the program to ensure it works correctly and meets requirements | $500 - 700 |
| Deployment and hosting | Costs for deploying the application to user’s computers and hosting the download on the server | $400 - 600 |
| Maintenance and support | Cost to maintain the program and answer help tickets | $400 - 500 |
| Security measures | Cost to maintain security of the program and control user access to data | $500 - 700 |
|  |  | **Total cost:** $5,000 - 7,700 |

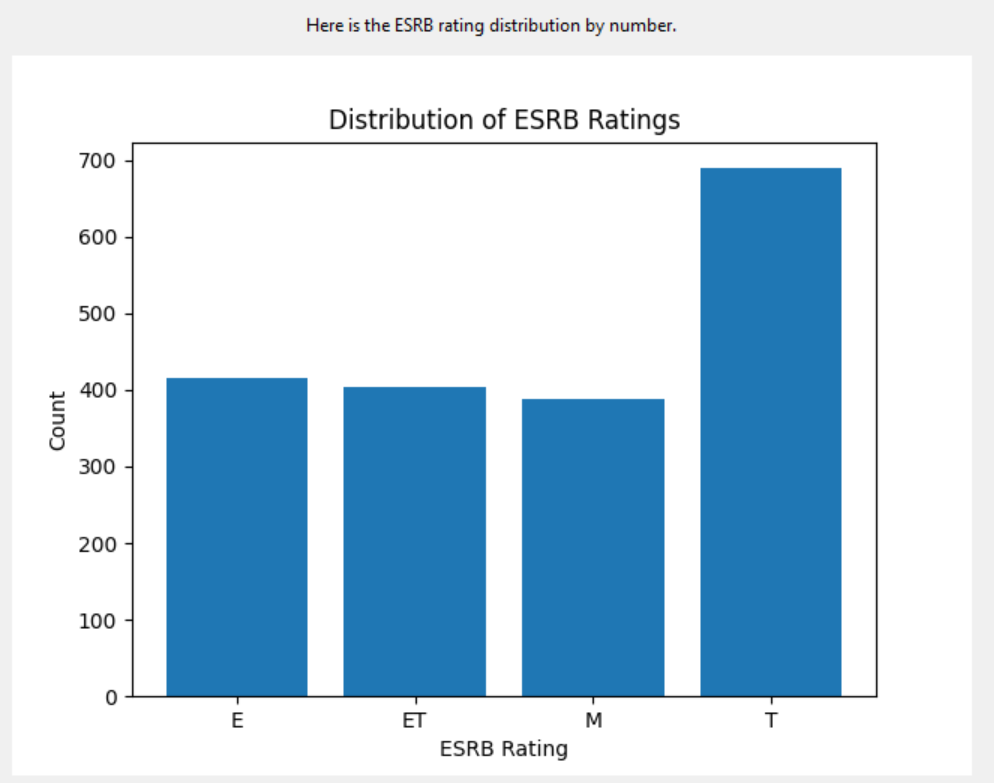
## Timeline and Milestones

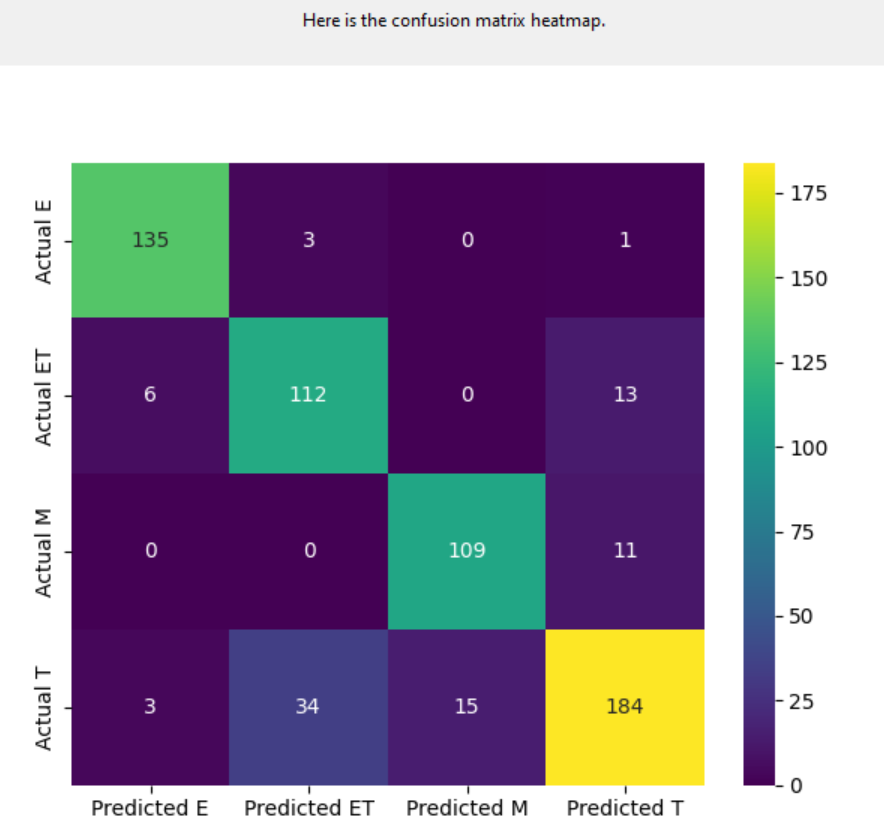
The following dates are only approximate, and will vary depending on when each phase of the waterfall method has been completed for the project. This timeline assumes that the project will begin in January of 2024, and the expected month of completion is in June of that year.

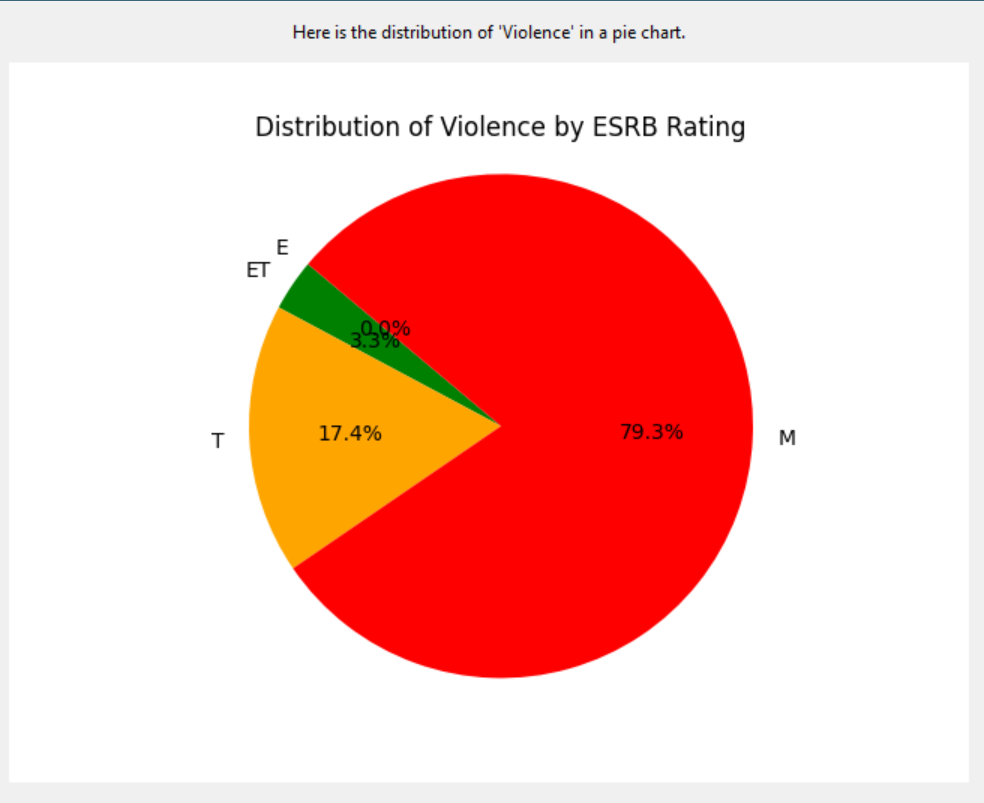
| **Milestone** | **Start date** | **End date** |
| --- | --- | --- |
| Requirements gathering & analysis | 01/01/24 | 01/25/24 |
| Design of the program | 01/26/24 | 02/26/24 |
| Implementation | 02/27/24 | 05/01/24 |
| Unit and integration testing | 05/02/24 | 05/26/24 |
| Deployment | 05/24/24 | 05/31/24 |
| User training | 06/01/24 | 06/14/24 |
| Maintenance and support | 06/15/24 | Ongoing |

# **Part C: Application**

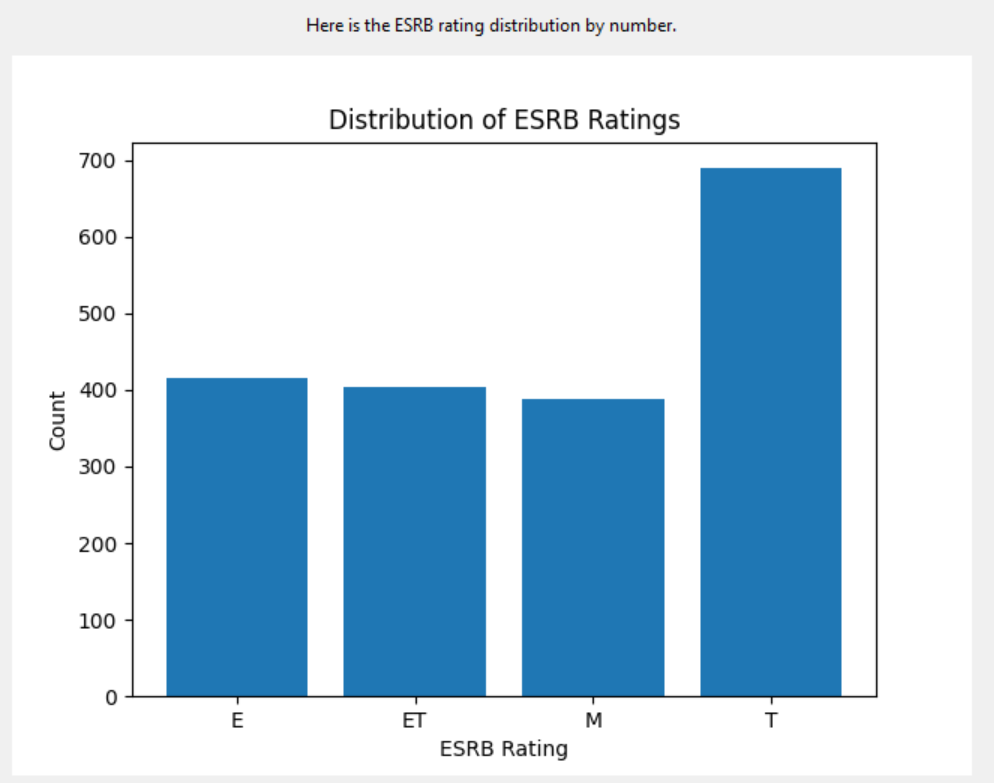
* Three visualizations (images). Static images are permissible:



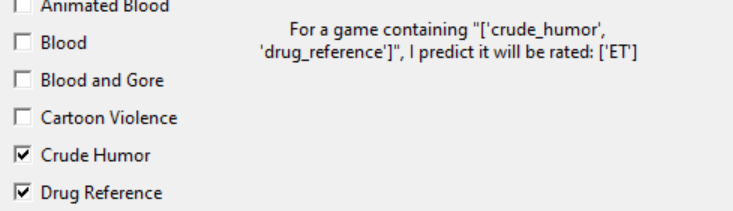




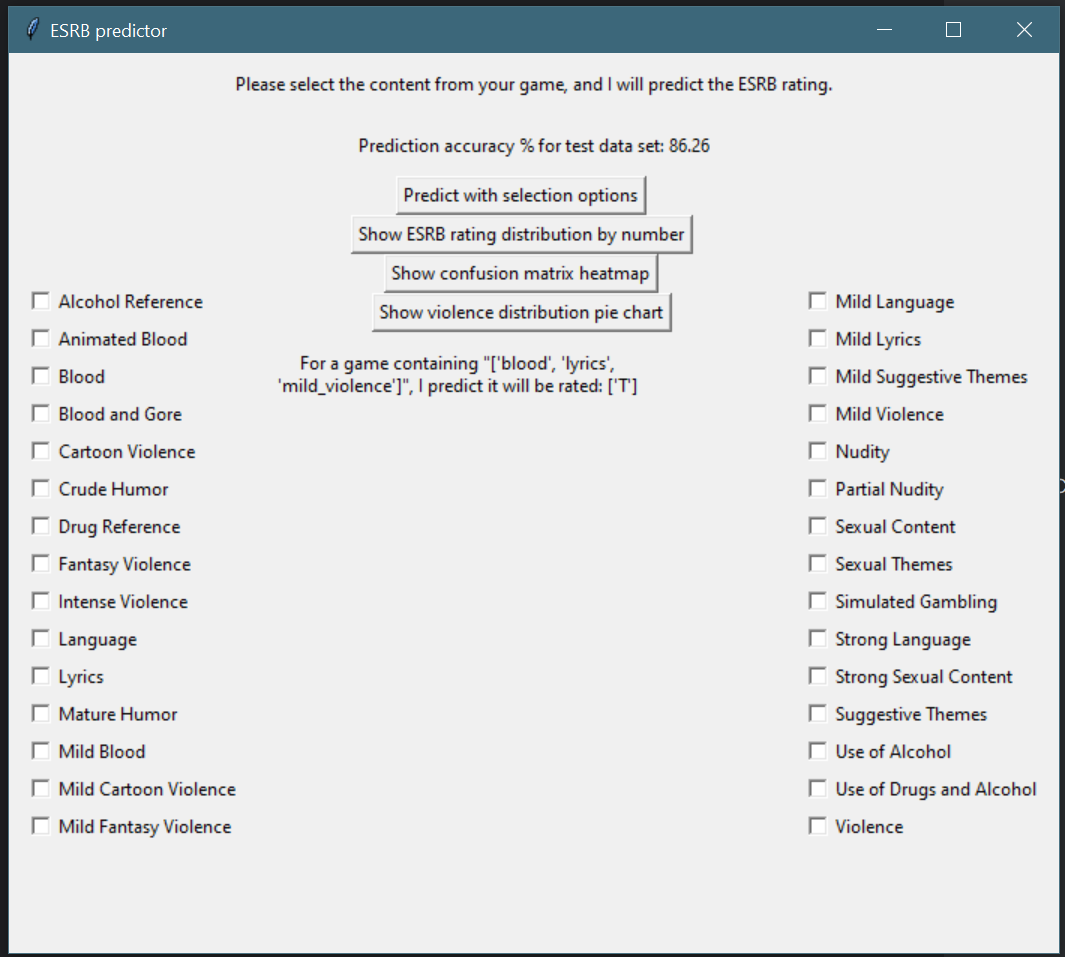
* A *Descriptive method* = anything that describes the data:



* A *Non-descriptive method* = anything that infers from the data, i.e., makes predictions or prescriptions:



* An application of “machine learning” in the non-descriptive OR descriptive method (most data analysis algorithms are acceptable -including regression), an interactive “dashboard, and a “user-friendly” interface:



* Security appropriate to your application’s needs.

The application will be distributed and limited to leadership’s working environment

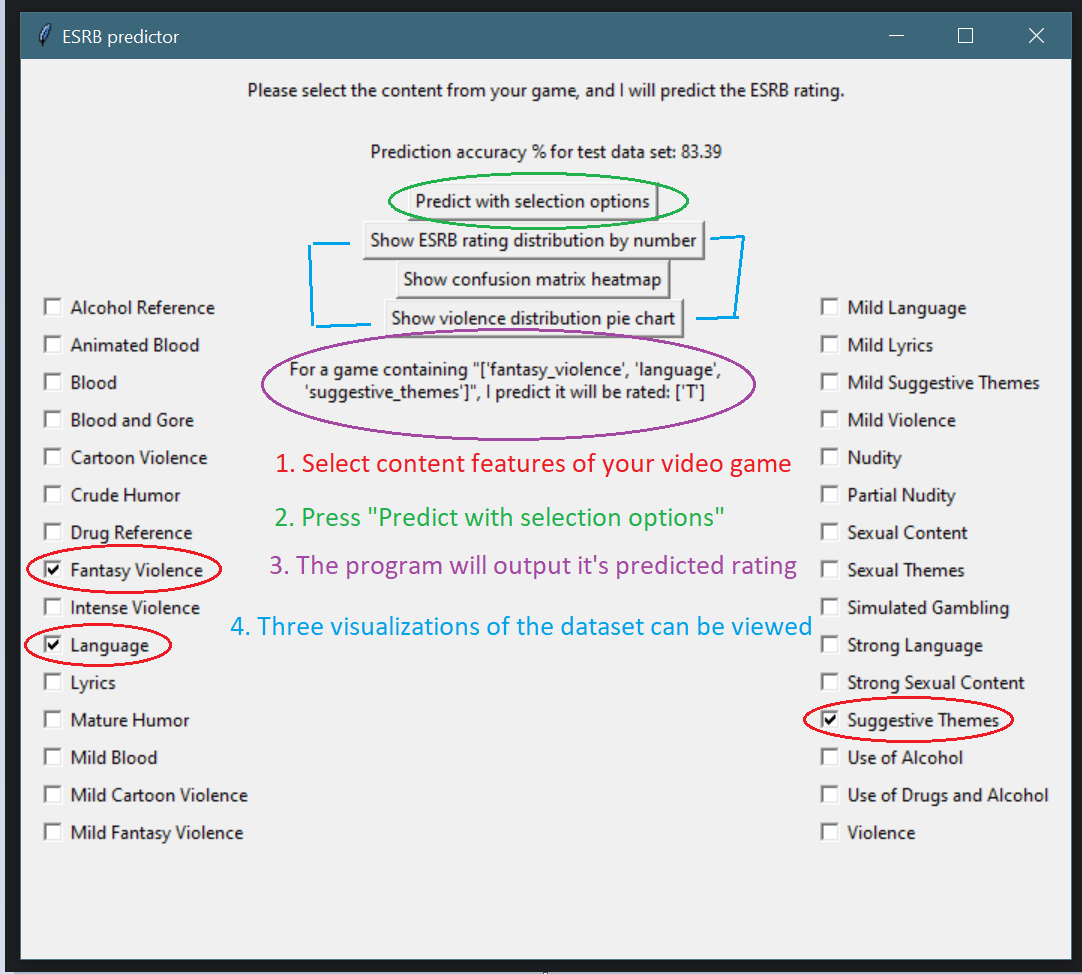
# **Part D: Post-implementation Report**

Create a post-implementation as outlined below. Provide sufficient detail so that a reader knowledgeable in computer science but unfamiliar with your project can understand what you have accomplished. Using examples and visualizations (including screenshots) beyond the three required is highly recommended. **Write everything in the past tense.**

## A Business (or Organization) Vision

In the video game industry, understanding and predicting a game’s Entertainment Software Rating Board (ESRB) rating is crucial in determining the game’s target audience and marketing strategy. Currently, the company is using intuition and industry expertise to predict the rating that their upcoming games will receive. This approach for predicting the ESRB rating is subject to biases and does not accurately reflect the rating which the games receive once the rating board has reviewed them on their completion. The business loses money when a prediction is wrong – for example, when a game was predicted to be “E” for everyone and was marketed to young children instead receives an “ET” rating for everyone ten or older.

In order to address these faults in the business, I have developed a program which can accurately predict the ESRB rating for a game given its content features. Using innovative machine-learning techniques and historical data and content analysis, the developed program is able to be used by leadership to accurately predict the ESRB ratings of our upcoming products. All the user has to do is run the program, input the content features of the video game of which they would like the prediction based on, and the algorithm will generate an ESRB rating based on the analysis of almost 2,000 video game titles in the program’s dataset. The program is enhanced with insights in the data used and trends found in recent video games by displaying to the user three visualizations of the data, which can be used to help further the decision-making capability of both the program and leadership itself.



## Datasets

The raw dataset used for this project contained information on over 1,800 video game titles. The data consisted of rows representing different video games, and columns providing details on specific content features related to the game. The content feature columns consisted of binary values (1 or 0) indicating whether or not that game contained that listed piece of content (crude humor, violence, ect.). The final column in the data contained the ESRB rating given for that specific video game.

The dataset was fairly complete and minimal processing was done in order to make it ready for use in this project. The original dataset contains columns not used for the project and had been removed prior to converting the dataset into a *.csv* format. This included the “console” and “no\_descriptors” columns because they are not needed for evaluating the content features related to the ESRB ratings. Once the dataset was cleaned and changed into *.csv* format, the data was ready to be used by the machine learning algorithm and so it was split into two subsets: a training dataset and a testing dataset. The training dataset trained the decision tree used in the project, and the testing dataset was used to evaluate the model’s predictive ability. This ensured the model was able to handle and predict new, unseen data.

Here is an example of how the raw data was formatted in an *.xls* file for use in Microsoft Excel:

| Title | Violence | Strong Language | Sexual Content | Blood | Rating |
| --- | --- | --- | --- | --- | --- |
| Assassin’s Creed | 1 | 1 | 0 | 1 | M |

Once the data was processed for use in the project, its features will delimited by a comma ‘,’ in *.csv* format:

Assassin’s Creed,1,1,0,1,M

The original *.xls* is provided along with this project. The converted and processed *.csv* is used in the project files.

## Data Product Code

The application did not need to process raw data, as I preprocessed the dataset and converted it to *.csv* format prior to beginning development of the project. The three descriptive method visualizations included a bar graph, confusion matrix heatmap, and a pie chart. The bar graph used the *Counter* function from the *collections* library of Python to count the distribution of the four ESRB ratings (E, ET, T, and M) in the original dataset. The *sns* function of the *seaborn* library in Python was used to display a heatmap visualizing the confusion matrix of the program as it trained with the test data. This shows at a glance how accurate the program was in predicting the ESRB rating for the test dataset. The final visualization also used the *Counter* function, but this time to count the instances of “violence” as a feature in the original dataset. This showed the visualization of how often the different ESRB ratings contained “violence”, but could be expanded to show other content feature distributions in future updates of the program.

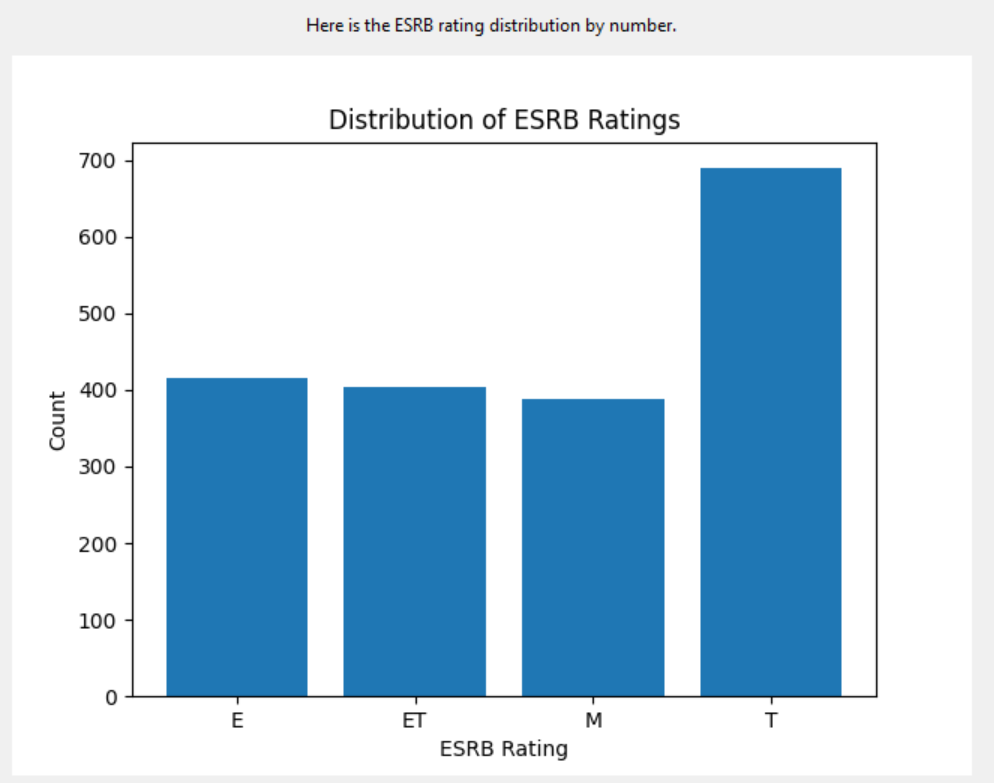
The non-descriptive method in the program is the decision tree algorithm used to make the ESRB rating predictions. The decision tree classifier used for the project is a supervised machine learning algorithm, suitable for predicting the ESRB ratings based on content features. A decision tree is able to automatically rank the importance of different content features. For example, certain features such as “nudity” might have more importance to a rating being “M for mature” than others. Decision trees are best used for classification problems, making it the ideal choice for this project.

## Objective (or Hypothesis) Verification

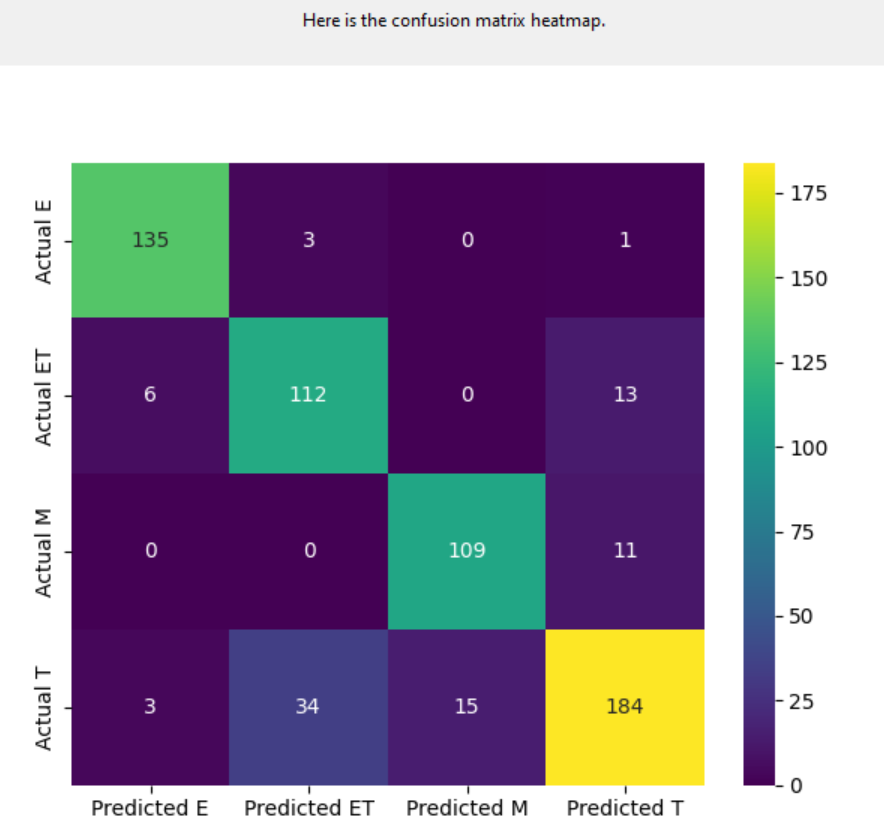
The project’s objective was to develop a machine learning tool for predicting the ESRB rating of video games based on the selected content features. The tool aims to provide our business’s leadership with a data-driven solution for predicting the ESRB ratings of our products in production, which will assist them in making informed decisions related to the game’s development and marketing. The objective of the project was met by implementing a machine learning model, providing a friendly-user interface, and complimenting the tool with three data visualizations. The decision tree classifier used to develop the project was trained with a large dataset that allowed the program to accurately predict the ESRB ratings of our products. The user-friendly interface makes it easy for leadership and other users to effectively interact with the program, without extensive technical knowledge. Finally, the visualizations of the data provide more insight on the data used to train the model, as well as being a tool for assisting in the decision-making process done by the leadership team.

## Effective Visualization and Reporting

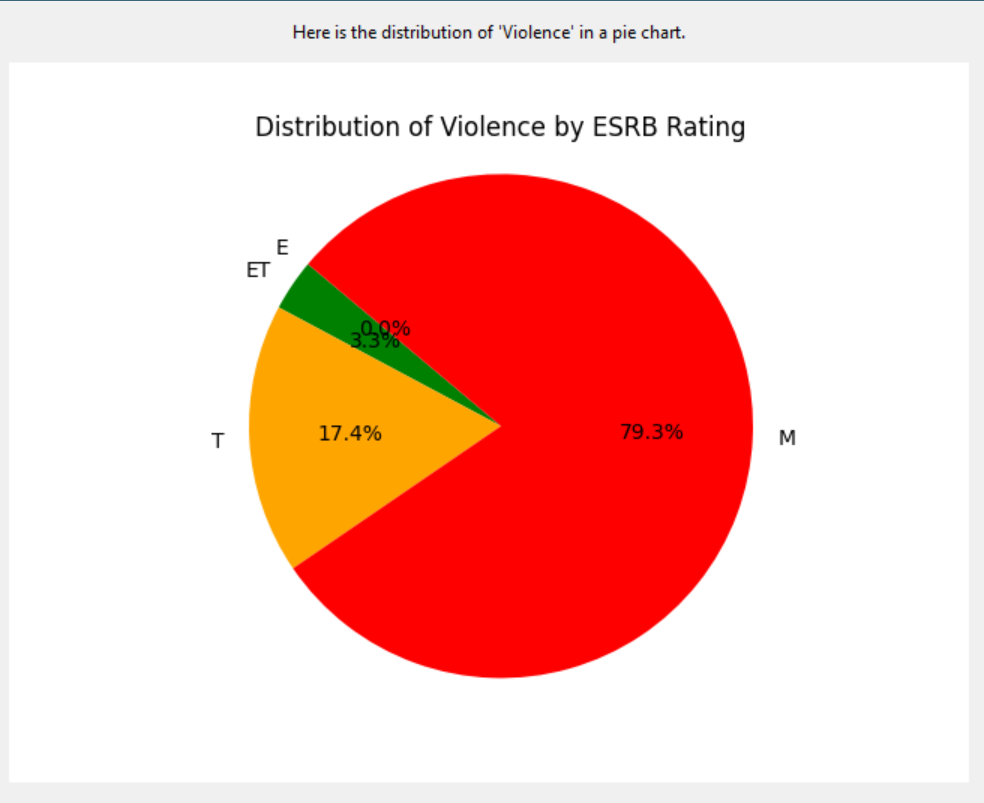
The descriptive bar chart visualization supported the non-descriptive decision tree by providing an overview of the distribution of ESRB ratings in the dataset. It aided in understanding the imbalance of classes – in this case, there were more “T” rated games than other ESRB ratings in the dataset. Games rated “T” occurred the most often, so the trained machine learning model might be more confident in its “T” ratings then the other ESRB ratings. The data also shows leadership in an intuitive way the distribution of ESRB ratings to grasp the dataset’s characteristics.



The confusion matrix heatmap supported the decision tree in evaluating its effectiveness with the training data. It provided insight into the model’s ability to classify the different ESRB ratings based on the content features of the video games. We can see from the visualization model that the decision tree had the most confusion in whether a game was rated “T for teen” or “ET for ages 10+”, with some confusion on whether or not a game was rated “M for mature” or “T for teen”. This insight can inform the tuning of the model to further improve its performance.



The pie chart visualization supports the decision tree and dataset analysis by exploring the presence of “violence” across the different ESRB ratings. This helps in understanding how substantiation the content feature “violence” is in the rating decisions of the ESRB. This chart could be switched out with other content features to show how each one might have an effect on the final rating of the video game and our products. From this visualization, we can see that “violence” is overwhelmingly more common in games that are rated “Mature”, showing a correlation between that content feature and that ESRB rating. Zero games in the dataset were given the most accessible rating “E for everyone” when there was violence present in the game.



These three descriptive visualizations provided a look into the data exploration, analysis, and summarization of the project. They provide insight into the dataset’s characteristics and the model’s performance, supporting the development of the non-descriptive decision tree. They also help stakeholders and leadership make informed decisions with the development of our video game’s and the strategies with which we market them.

## Accuracy Analysis

The metric used to assess the accuracy of the project’s non-descriptive method (decision tree) is the accuracy score. This is the percentage of the training data the machine learning algorithm was able to accurately predict. When the application is run, the accuracy score is put on the main page’s GUI and ranges from 80-90%. This means for a test dataset of 100 video game titles, the decision tree model correctly predicts the ESRB rating of 80-90 of those titles. This level of accuracy suggests that the decision tree model is correctly predicting the ESRB ratings for a large part of the test dataset, and makes it effective for making predictions. The prediction accuracy is given by the program on the main page to show leadership the effectiveness of the program at making ESRB predictions based on the content features of the game. The formula for calculating the accuracy score of a given decision tree is as follows:

Accuracy = (Number of correct predictions) / (Total number of predictions)

## Application Testing

In this project, testing was conducted using unit testing to verify the correctness of individual components (units) in the application. For example, when writing the code for the main prediction algorithm and GUI to work alongside it, I encountered an error with the program displaying the output on the GUI if a previous prediction was already made during that session. To remedy this, I took the prediction label outside of the function used when the user clicks “submit”, and instead reassigned it’s value every time the submit button is pressed. This ensured that when a new ESRB prediction was requested, it would correctly display the result onto the GUI. Unit testing while the project was being developed helped catch bugs early on, and ensure that the whole project would run correctly once development was finished. Thorough testing of the application contributes to the reliability of the program, ensuring it meets the project’s objectives and provides accurate ESRB predictions for the leadership team.

## Application Files

Libraries required to run the program: tkinter, matplotlib, decisionTree, collections, sklearn, classes, pandas, and seaborn

Below are the list of the files as they are ordered in the project submission, indentations indicating the item belongs in a lower subfolder:

* Project files (Open me) – Project folder
  + .idea – project specific settings files
  + \_\_pycache\_\_ – Python interpreter’s directory for module imports
  + classes – Holds the classes for the project
    - videogame.py – Creates the “videoGame” class for making predictions in the GUI
  + datasets – Contains the dataset used for the project
    - video\_games\_esrb\_rating.csv – the dataset for the project
  + GUI – Contains the four windows of the GUI in separate .py files
    - esrb\_dist.py – Contains the descriptive bar graph visualization for the GUI
    - main\_menu.py – Contains the main menu and non-descriptive predictive functions of the program/GUI
    - matrix.py – Contains the confusion matrix heatmap window of the GUI
    - violence.py – Contains the descriptive pie chart window for the GUI
  + decisionTree.py – contains the decision tree’s creation and training
  + main.py – main file to start the program

## User Guide

1. Download PyCharm from the official website (https://www.jetbrains.com/pycharm/download/?section=windows)

Default installation settings are fine

2. Download python 3.9.0 from the official website (https://www.python.org/downloads/release/python-390/)

3. Open PyCharm, agreeing to terms and conditions

4. Activate pyCharm, or start the free trial

Any email will work for starting the trial

5. On the main screen after signing-in and/or starting the trial, click "Open"

6. Navigate and open "Project files (Open me)"

7. Click "Trust Project"

8. If no Python interpreter is found, click on the warning "Configure Python Interpreter"

8A. Click "Add new interpreter"

8B. Click "Add local interpreter"

8C. An existing environment location may be detected under "venv". Just rename the new environment something else. ex: "venv2"

8D. If installed to default locations, Python 3.9.0 may be detected automatically. Otherwise, it may need to be manually located.

8E. Press "OK" once new location is made and python.exe is set as base interpreter

The Python interpreter is now installed.

9. Re-open the project so libraries can be installed. Go to "file -> Open -> Project files (Open me)"

10. Navigate to "matrix.py". Import all packages highlighted with a red underline by hovering over them and clicking "install package \_\_\_\_"

This includes matplotlib, seaborn, sklearn, and pandas.

11. Packages will take some time to install. Once this is done, "Main" should be able to run and the program will start.

## Summation of Learning Experience

My completion of the WGU Computer Science bachelor’s program served as the foundation for the completion of this project. The program gave me a comprehensive understanding of the algorithms, data structures, and crucial skills needed to develop this ESRB rating prediction program. In addition to this formal education, I independently researched Youtube videos demonstrating how to work with tkinter, sklearn, and matplotlib. This demonstrated my ability to research independently and continue my learning with the tools and technology I have on hand. This project has taught me that I still have a lot to learn in my path towards becoming a software developer, and that being open to new information is a key skill necessary in a field as dynamic as computer science.