July 20, 2023

Game Sales Document

Part A: Project Proposal for Business Executives 3

[Letter of Transmittal 3](#_Toc1044087116)

[Project Recommendation 4](#_Toc1721750458)

[Problem Summary 4](#_Toc205640471)

[Application Benefits 4](#_Toc739883765)

[Application Description 4](#_Toc1967641075)

[Data Description 4](#_Toc1580263866)

[Objectives and Hypothesis 5](#_Toc1084915425)

[Methodology 5](#_Toc1455093807)

[Funding Requirements 5](#_Toc1716359054)

[Data Precautions 5](#_Toc821607173)

[Developer’s Expertise 6](#_Toc742297278)

[Part B: Project Proposal 7](#_Toc1516064891)

[Problem Statement 7](#_Toc602847551)

[Customer Summary 7](#_Toc411678477)

[Existing System Analysis 7](#_Toc1668713332)

[Data 7](#_Toc1911399030)

[Project Methodology 8](#_Toc1307313575)

[Project Outcomes 8](#_Toc1967172832)

[Implementation Plan 8](#_Toc306674583)

[Evaluation Plan 8](#_Toc849271423)

[Resources and Costs 9](#_Toc1695791095)

[Timeline and Milestones 9](#_Toc787992355)

[Part C: Application 10](#_Toc1767968421)

[Part D: Post-implementation Report 11](#_Toc22579049)

[A Business (or Organization) Vision 11](#_Toc1496854501)

[Datasets 12](#_Toc645781746)

[Data Product Code 12](#_Toc157285634)

[Objective (or Hypothesis) Verification 13](#_Toc1315546441)

[Effective Visualization and Reporting 13](#_Toc656892821)

[Accuracy Analysis 13](#_Toc1956352798)

[Application Testing 14](#_Toc871280264)

[Application Files 15](#_Toc1639588139)

[User Guide 15](#_Toc1125023771)

[Summation of Learning Experience 15](#_Toc1702287338)

# Part A: Project Proposal for Business Executives

## Letter of Transmittal

## July 20, 2023

## Bob Dylan, CEO

## Games Co.

## 123 Games Ln

## Boston, Massachusetts

## Dear Mr. Dylan,

I’ve noticed the company is having trouble drafting the next game to make. It’s unknown what kind of video game would be best for the company at the moment, and I have an idea for a data product that could solve this problem. I understand the concerns about the profitability of the next game because the last couple of games haven’t been as successful as we’d projected, so we’re falling behind the curve. The data product I have in mind will pull data from the majority of video game sales from 2000-2016, and display trends, recommendations, as well as predictions.

The data product will be beneficial because it will give the company more guidance on what type of games the current market is interested in. We keep creating games for small, niche audiences, and it isn’t sustainable. By having a clear idea of what games excel the most, we can have an easier time coming up with ideas without the constant worry of it being a waste of resources. The total cost of the project will end up being around $11,100, and an additional $1,080 for every year onwards for maintenance. I’ll be developing the project alone as a senior machine learning engineer since the project scope is suitable for one developer. I’m capable of creating this project alone due to my past education and experience, having a bachelor’s in computer science, as well as 3 years in developing software. However, a quality assurance company will need to be hired for the project.

Thank you for considering this project as I truly think it will push the company in the right direction. Reach out to me with any questions you may have about the project.

Sincerely,

Nicholas Pantuso

## 

## Project Recommendation

### Problem Summary

The project will take a majority of video game sales (Over 16,000 games) from 2000-2016, and display sales trends, connections, recommendations, and calculate an estimation of whether a certain game will outsell another. This project is needed by Games Co. because the company is having trouble drafting the next game to develop. The past couple of projects haven’t been as profitable as projected, and the company is falling behind. The project will grant the company some guidance on what’s most successful in today’s market to increase the probability of meeting goals.

A data product will be delivered that will visualize and predict the outcome of video game sales, and an educated decision on the next game for the company will be achieved.

### Application Benefits

The project will meet the business’s current needs by offering a way to learn about the current gaming market and predict what could be a successful game. The company is at risk of going out of business if the next game is a flop, and this project will be able to offer ideas and trends that are commonly successful and profitable to avoid failure.

The application will benefit the company by creating tools to gain a better understanding of the market, predict the success of ideas before resources are allocated, and reduce the risk of game failure once released.

### Application Description

The product will display information in multiple ways to encourage an educational decision on the next game. For example, there will be a bar graph showing total sales for each genre which will represent what genres are the most successful. Scatter plots will compare sales between the three big regions (NA, EU, JP), and visualize trends between said regions. The decision of whether to focus on a certain region will be easier to make, rather than it being shot in the dark. A linear regression graph will be created showing the correlation between genre, platform, and overall sales. Finally, there will be a prediction model which will enable users to input a genre and a platform, where the model will estimate the total sales of that genre/platform combination. The combined effect of these functions will ultimately lead the company to a successful game idea.

### Data Description

The data used for the project is from vgchartz.com and scraped by Smith (2016), who made it into a dataset on Kaggle.com. The information for each video game in the dataset includes:

* Rank (Rank among the rest of the video games regarding global sales, integer)
* Name (Name of the video game, text)
* Platform (Platform the game was released on, text)
* Year (Year the game was released, integer)
* Genre (Genre of the game, text)
* Publisher (Publisher of the game, text)
* NA Sales (Sales in North America, decimal)
* EU Sales (Sales in Europe, decimal)
* JP Sales (Sales in Japan, decimal)
* Other Sales (Sales in rest of world, decimal)
* Global Sales (Sales worldwide, decimal)

The independent variables are the name, platform, year, genre, and publisher. The dependent variables are rank, NA sales, EU sales, JP sales, other sales, and global sales. The dataset contains video games before 2000 and after 2016, but they will be removed from the dataset before implementation. The dataset does not update and is static, if the data needs to be dynamically updated, the cost of the project will increase.

### Objectives and Hypothesis

The main objective of the project is to give the company the tools it needs to come up with a successful game idea. Some secondary objectives consist of being easy to read and understand, easy to use and navigate, and cost-effective.

The hypothesis for this project is that creating a way to analyze video game sales and predict the sales of potential games will help the company make an educational decision on what type of game to develop next. If this proves to be a great help in decision-making, the data can be updated, and the project can be expanded to include even more ways to predict and compare data to further assist the company.

### Methodology

Since the project is relatively small and there isn’t a large team associated with it, the waterfall development methodology will be used. The waterfall methodology will also allow the project to be completed in a timely and cost-efficient manner because there won’t be iterative updates and scope changes during development. The following steps will be taken during the development method:

1. Requirements – Outline the project from beginning to end, create a timeline, and acquire resources.
2. System Design – The hardware and software being used to develop the project are specified. The bare bones and how the project will be put together are decided.
3. Implementation – Software is developed concerning the outline and requirements.
4. Testing – The project is sent to the quality assurance team for testing. Bugs are fixed.
5. Deployment – Software is deployed, and users will be able to access it.
6. Maintenance – New bugs are squashed, updates are created, and software is monitored.

### Funding Requirements

The software and data are free, but the cost of expertise and equipment is still present. The project is estimated to take around 90 working hours with a rate of $90 per hour. The equipment and tools required are approximately $3,000 and combined with salary add up to $11,100. Monitoring and maintenance of the software will cost 12 working hours at a rate of $90 per hour, per year, for a total of $1,080 per year.

### Data Precautions

The dataset contains no sensitive or protected data to be concerned about and is all public information. The dataset can be found with a simple search on Kaggle.com and downloaded for free by anybody. All information in the dataset is only related to video games and how much they sold.

### Developer’s Expertise

My qualifications for the project include a bachelor’s in computer science as well as 3 years in developing a wide variety of software. Since the machine learning in this project isn’t too complicated, I’m more than qualified for the development of this project. My education and experience combined will guarantee a functional and intuitive product delivered on time and in scope.

# Part B: Project Proposal

## Problem Statement

## The company is facing an issue with the decision of the next game to develop. The past few projects haven’t met sales goals and we’re falling behind competitors. There needs to be a way to analyze the current market to ensure the next game made is a success and is targeted at the right audience. The data product will take 16,000 video games from 2000-2016 and compare and contrast their sales relative to their genres, platform, and region. There will also be a prediction model where users can input a genre and a platform, and the model will output an estimate of how well the game will sell. This application will allow users to learn about the market, find what’s successful, connect the dots, and create their own idea based on what the general market is buying right now.

## Customer Summary

## The client for this project is specifically the designers of Game Co., but it could also be used by any game designer wanting to learn about the gaming market. Someone who wants to know why some games are excelling, and others aren’t; someone who wants to analyze what the general public buys vs what the more niche groups buy.

## The data product will resolve the problem regarding the game decision because it houses a variety of tools that visualize and predict the gaming market. The designers will be presented will graphs and estimations of what does and doesn’t work, which will ultimately assist them in a design decision for the better.

## Existing System Analysis

## The designers may or may not use the internet for market research, but other than that, they have nothing to go off of. They can look up top-selling games but won’t be given information as specific as what’s shown in the data product. Also, those top-selling games will only be as of the current year, where this solution will take a vast majority of games all the way back from 2000 and combine them into one big dataset with current best sellers. Various gaming websites will also not be comparing and contrasting game sales in scatter plots and bar graphs because that’s not what gaming websites are for. The data product will take over 16,000 games, connect them, visualize them, and make predictions from them.

## Data

## The raw dataset is on Kaggle.com and was scraped from vgchartz.com. The dataset includes the game’s rank (based on global sales), name, platform, year, genre, publisher, NA sales, EU sales, JP sales, other sales, and global sales. Only numerical and nominal values are used across the fields. The data is collected from Kaggle.com, sent to a Jupyter Notebook, and managed in the code; the data doesn’t require cleaning outside of removing games before 2000 and after 2016. The data is saved as a file inside the notebook, so the data and code are not separate things that need to be connected. During maintenance of the application, any changes that need to be made will be easy and quick. The data contains no anomalies and requires no handling other than the exclusion of some years.

## 

## Project Methodology

## The methodology to be used for the application is the waterfall method. Since I’m the only developer for the project and the project is relatively small, the simplicity of the waterfall method will be the best choice for development. The method will consist of six phases. First, is the requirements phase; the entire project will be outlined, a timeline made, and resources acquired. This phase will set the project up for success and set expectations. In the design phase, the hardware and software for the project will be specified, and the foundation of the project defined. During the implementation phase, code is written, and software is developed. The testing phase will allow the QA team to double-check my work, bugs will be fixed, and the next phase, deployment, will occur soon after. The software will be available to access, and the final maintenance phase will be in action, where I update the software if needed.

## Project Outcomes

## At the end of development, the data product will have four visualizations of the data, as well as an interactive prediction model. The top 5 selling games will be displayed with all of their fields present, a bar graph showing the top-selling genres (globally), various scatter plots will compare NA, EU, and JP sales, and a linear regression plot will show the correlation between genre, platform, and how much money the game makes. The prediction model will allow the user to enter a platform and a genre, where the model will then estimate how much money that game may make. A user guide will be included to ensure the application is used correctly and to its full potential.

## Implementation Plan

## First, the data will be extracted from Kaggle and uploaded to the Jupyter Notebook, where the Python (Pyodide) language will be used to code. The code will be written one module at a time, where each module is made fully functioning before moving on to the next. The first step is cleaning the dataset and outputting the head to ensure implementation works. Then, the data will be applied to the various plots for the user to analyze. A linear regression algorithm is then implemented and trained, which is what the prediction model will use to create its estimations. The buttons and input fields for the prediction model will then be made, and the project will be ready for testing. The application will be sent to the QA testers for testing and quality assurance. The project rollout will be extremely easy, and no hardware/software changes are necessary on user computers. The project is simply an online link that will take the user to the data product, and the only thing required of the user is an updated browser. If any maintenance updates are required, a new link will be provided with update notes.

## Evaluation Plan

## Since the waterfall methodology is being used, verification and validation happen once and at the same stage, the testing phase. The QA testers will verify the program does as required, and anything not meeting requirements will be fixed. The software will be validated with the game designers, we’ll have a meeting showcasing the software after verification and project completion. If all is well, the project will be deployed and signed off.

## 

## Resources and Costs

## The main resources for the project are labor and physical equipment. Labor is estimated to be around 90 hours, with a rate of $90 per hour. The equipment consists of computer hardware and office supplies/space, which total $3,000. Yearly maintenance is also estimated to be 12 working hours a year at the same rate of $90 per hour. Labor and equipment total $11,100, whereas maintenance totals $1,080 every year. The application is hosted on a free website, so no software, hosting, or server fees are required.

## Timeline and Milestones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Milestone | Activity | Hours | Start | End |
| 1 | Approval of outline and requirements | 8 | 7/31/23 | 7/31/23 |
| 2 | Design skeleton of software | 8 | 8/1/23 | 8/2/23 |
| 3 | The dataset was accessed and cleaned | 6 | 8/2/23 | 8/3/23 |
| 4 | Data visualized into graphs | 12 | 8/3/23 | 8/7/23 |
| 5 | Machine learning model trained | 12 | 8/7/23 | 8/9/23 |
| 6 | Prediction model implemented | 6 | 8/9/23 | 8/11/23 |
| 7 | QA testing | 20 | 8/14/23 | 8/17/23 |
| 8 | Project verification/validation | 12 | 8/18/23 | 8/22/23 |
| 9 | Project deployment | 6 | 8/23/23 | 8/24/23 |

# Part C: Application

Data product link (please read the user guide before using):

<https://mybinder.org/v2/gh/NickPantuso/C964Capstone/HEAD>

# Part D: Post-implementation Report

## A Business (or Organization) Vision

## The problem was the lack of sales of recent video games made by Games Co., as well as poor development decisions leading to falling behind competitors. More knowledge of the gaming market was needed to educate the designers on what was seeing success and what wasn’t. The application solved the problem by visualizing and predicting sales data from over 16,000 video games between 2000 and 2016. The designers were able to analyze the market with the tools the application offered and made a logical decision on the next game to develop. The user can analyze graphs, make connections, and compare predictions.

## A graph with blue bars Description automatically generated

## For example, the above picture is one of the graphs created in the application. The user can see that action games are by far the best-selling of all genres and that they have a huge audience. On the flip side, strategy games have a very niche audience, and most gamers are looking to spend their time in action rather than tough and rigorous decision-making. This assists the user in better deciding what game they want to make that aligns with their goals.

## Another example is if the user wanted to use the prediction model. If the user inputs the platform ‘Wii,’ and the genre ‘Sports,’ the estimated sales are much higher than ‘XOne,’ and ‘Sports.’ This means sports games on the Wii sell better than sports games on the Xbox One, which is very useful information for developers when deciding on a platform to develop for.

## Datasets

## The raw data had over 16,000 video games from 1980-2017, and the fields consisted of rank, name, platform, year, genre, publisher, NA sales, EU sales, JP sales, other sales, and global sales. When moving the data to the application, rows with games before 2000 and after 2016 were removed, as well as any rows that had a non-numerical value in a numerical field. For the machine learning algorithm, the platform and genre fields were replaced with indices rather than their actual names, and all other fields were dropped to compare platform and genre against global sales.

## The top 5 rows of the raw data:

## A screenshot of a white grid Description automatically generated

## The top 5 rows of the processed data:

## A screenshot of a video game menu Description automatically generated

## Dataset used: <https://www.kaggle.com/datasets/gregorut/videogamesales>

## The dataset is also found in the references section.

## Data Product Code

## The raw data was processed in Python using a library called pandas. Any modifications to the dataset were made with pandas methods such as dropping or renaming fields. All visualizations and descriptive methods were done with the libraries pandas, matplotlib, and seaborn. The data is extracted and processed via pandas and made into a graph via either pandas, matplotlib, or seaborn.

## In the non-descriptive method, a linear regression model was created from the sklearn library and was trained by the processed game sales dataset. I used the game sales dataset to train the model because the model was going to predict sales based on that same dataset. The model helps solve the problem of ‘what game should be made’ by learning from the dataset and estimating how much a potential platform/genre combination could make in comparison to other combinations.

## I chose linear regression for the analytical method because its best use is predicting a value based on another value, which is exactly what I wanted to do. The method was tested with the same sklearn library that created the model, and it calculates an R-squared error that represents how much of the target variance can be explained by the model (the closer the R-squared error value is to 1, the more closely related the variables are).

## The small user interface was created using the library ipywidgets methods for dropdown menus and buttons. On button click, the linear regression model takes a platform and a genre of a game and predicts its global sales.

## I came up with the descriptive and non-descriptive methods by looking at the raw data and wondering where the connections were for a high-selling game. Is it platform, genre, or region? This made me compare global sales to genre, region to region, and the correlation between a platform/genre combination and global sales.

## Hypothesis Verification

The project hypothesis was that creating a way to analyze video game sales and predict the sales of potential games will help the company make an educational decision on what type of game to develop next. The hypothesis was correct, and the Games Co. designers decided on the type of new game to develop based on their findings in the information and analysis of the data product. The hypothesis could be made even stronger if the dataset contained the marketing budget of each game, as well as more specific genres.

## Effective Visualization and Reporting

## When exploring the dataset, I wanted to be able to make a model that predicted the sales of a game based on their traits, and I found a game’s platform and genre to be an interesting combination to predict from. After looking at my bar graph of the genre to global sales, it was obvious that genre had something to do with total sales, so that increased my inspiration to use genre as an input for prediction. In my region vs region scatterplots, I found that NA and EU sell at a 4:3 ratio, and JP has little to no correlation to NA or EU. It seemed that games are usually either made specifically for Japan or the rest of the world. This finding didn’t contribute much to what I wanted to predict, but it displayed enough information about the main regions that I didn’t need to include anything regional in the prediction. In the final visualization, the linear regression graph displayed a very low correlation between the platform/genre combination and total sales. However, this didn’t stop me because I wanted to know if the results would still make sense. Turns out the prediction model still displayed logical results even with such a low correlation. For example, the model shows that sports games for the Xbox will sell for less than sports games for the Wii, which makes sense given the audience of the platform (Wii being more for kids).

## Accuracy Analysis

## The metric used to assess the model is known as R-squared error, which is a value from 0-1. The closer the value is to one, the better the regression model is at prediction, or the more closely correlated the variables are. As stated above, the correlation in my linear regression model is extremely low, at an R-squared error of 0.006, which is little to no association. However, regardless of the very low accuracy, it still displays results that make sense. The dollar value of the game it displays is worthless, but what does have worth is whether a game has a higher dollar value than another. The model is still useful if taken as “this will sell more” rather than “this will sell (dollar value) more.” For example, if you input ‘PS4’ and ‘Action,’ the dollar value shown is useless and won’t mean anything. However, if you then input ‘PS4’ and ‘Puzzle,’ the dollar value is less, therefore the action game will likely make more money than the puzzle game.

## Application Testing

## The application was tested by using it how it would normally be used. I thought, does the data make sense? Do the graphs show the right numbers and fields? Does the prediction model work every time? The application is simple and didn’t need to be unit tested or something of the like. Something that changed through testing was the regional scatter plots. Originally the user was supposed to be able to control which regions they wanted to see on one scatterplot, but I changed it to all three being shown at once. This way the user could compare all scatterplots at once rather than having to click back and forth. No other modifications were necessary due to the simplicity of the project and the low number of modules. I also made sure that what I wanted to do was feasible by looking up how what I wanted to do was accomplished to avoid surprises, which made development a smooth process.

## Here's an example of the scatter plots being shown next to each other as a result of testing:

## A comparison of a graph Description automatically generated

## Another way I was able to test the data was by looking at the head of the data and seeing if my changes affected anything. The following picture shows the rank of the data on the far left. The jumps in rank are indicative of the games I removed that were released before the year 2000.

## A screenshot of a video game menu Description automatically generated

## Application Files

## There are no application files, only a link to the online notebook. The source code is shown in the notebook.

## User Guide

## Copy the project link into your browser search bar and hit enter, wait for the page to load.

## On the left side of your screen, double click on “C964Task2.ipynb”

## At the top left of your screen, click ‘Kernel,’ then ‘Restart Kernel and Run All Cells…’ and wait for the modules to run (asterisks will appear on the left side of the modules, wait for them to turn into numbers).

## Scroll to view the visualizations and prediction model.

## Once at the bottom, you can use the prediction model to estimate a game’s sales. Click the dropdown menus next to ‘Platform:’ and ‘Genre:’ to choose the desired platform and genre, and then click ‘Estimate Sales’ to view the prediction.

## Summation of Learning Experience

## My previous experience made this project easy to understand and prepare for. My academic experience allowed me to properly sort through all instructions and find the best route to complete this project, while my programming experience made it easy for me to learn how to visualize and process data. Academic experience also helped me figure out what I wanted the project to be about, as I found that making things interesting to me greatly improves the speed and quality of my projects. The only significant thing I needed to learn was how to use the various Python libraries to build the linear regression model as well as the graphs.

## Developing this project has contributed to my concept of lifelong learning by showing that learning isn’t a chore if it’s about something you enjoy. Learning how to program and develop software has been fun because I’ve always liked working with computers, and it’s easy to tie in with other interests as well due to the flexibility and freedom of the job. Since software development is competitive *and* I enjoy it, I’m always wanting to learn more to enhance my skills and knowledge of the craft, which is required if I want to keep up with new technology and peers.

**References**

Smith, G. (2016) *Video Game Sales* (Version 2) [Data set]. Kaggle. https://www.kaggle.com/datasets/gregorut/videogamesales