Software Development Plan

Dunham, Martinear, Richardson, and Traglia

September 2, 2016

Contents

1 Project Overview

1.1 Description

PoissonBarrel, is a statistical analysis software program which analyzes a set of data input by the user, makes calculations based on the statistical measure chosen by the user, and displays the results in a graph of the user's choice. For example, after entering the data, the user may choose a statistical measure for analysis (i.e. mean, median, mode, or standard deviation) and then choose a type of graph (i.e. vertical bar graph, X-Y Plot graph, table) to display the data. In short, PoissonBarrel functions similarly to Microsoft Excel.

PoissonBarrel is named after the French mathematician Simon Denis Poisson, a pioneer in modern probability theory and the namesake of the Poisson distribution. Poisson also means "fish" in French, and since this application is a "barrel of" statistical analyses.

1.2 Features

- Allows the user to enter data for analysis
- Provides a list of statistical measures to analysis the data
- Provides a list of graph types to represent the data
- Displays graphical representation of analysis results
- Creates a text file summarizing analysis results
- Allows user to load, save, edit and delete analysis results

1.3 Statistical Options

There are a number of supported statistical measure types. The types range from simple statistics, such as *mean*, to more complex statistics, such as *Chi Square*. These types are listed and described in the following table:

Statistical Measure Types	
Mean	Median
Mode	Standard Deviation
Variance	Coefficient of Variance
Standard Deviation	Percentiles
Probability Distribution	Binomial Distribution
Least Square Line	Chi Square
Correlation Coefficient	Sign Test
Rank Sum Test	Spearman Rank Correlation Coefficient

There are also multiple supported chart/diagram types. These types are listed and described in the following table:

Chart/Diagram Types
Horizontal/Vertical Bar Chart
Pie Chart
Normal Distribution Curve
X-Y Graph

2 Components

2.1 File I/O Components

There are three components that deal with file operations. These are listed and described in the following table:

File I/O Components	**Description**
Comma/Tab Delimited Files	The application contains a module that enable it to read, validate, and write CSV files that can be viewed in Microsoft Excel.
Summary Text Files	The application writes text files that summarize the statistical analysis performed by the user on the documents data.
JPEG Image Files	The application has a module that allows the user to create JPEG images of any chart or graph that the application generates.

2.2 Graphical User Interface

The largest component is the graphical interface. There are multiple sub-components that collectively make up the graphical interface. These components are listed and described in the following table:

Graphic User Interface Components	Description
Main Window	The application has a main window widget that allows the user to interact with the displayed data through mouse and keyboard input. Additionally, the main window contains a menu bar allowing the user to open, close, and save documents, as well as providing access to the Chart/Diagram Creation Wizard.
Spreadsheet Table	The application has a spreadsheet table widget contained within the main window's client area that allows the user to create, view, modify, and analyze statistical data.
Open/Save File Dialog Boxes	The application allows the user to open and save files through the modal dialog boxes that provide graphical user interface to the operating system's file management system.
Chart/Diagram Creation Wizard Dialog	The application provides a modeless dialog box that allows for user friendly creation of charts and diagrams.
Chart/Diagram Windows	The application generates and displays charts and diagrams in a window separate from that of the of the main window. Charts and diagrams are generated by the data contained within the spreadsheet table. The window shall have an option that allows the user to create a JPEG image file of the displayed chart.

3 Project Schedule

3.1 Deliverables

There are four project deliverables, as defined in the table below, that will be given to the customer. For each deliverable, there will be given a presentation of it to the customer.

Deliverable	Deadline
Project Requirements and Backlog	September 27, 2016
Architectural Design	October 18, 2016
User Interface Design	November 3, 2016
Product Delivery	November 29, 2016

3.2 Milestones

For this project, there are loosely defined milestones our team wishes to complete by certain deadlines. These milestones will encourage continuous development and mark certain stages of the project.

Milestone	**Deadline**
Create Proof-of-Concept GUI	September 13, 2016
Develop Statistical Algorithms and Spreadsheet UI	September 29, 2016
Begin Testing on Statistical Functionality And Development on Chart Generation	October 20, 2016

4 Team Members

The team consists of four members. Each team member is assigned one unique position, in addition to the following positions: *Requirements Analyst*, *Designer*, *Programmer*, *Test Designer*, and *Test*.

Richard Dunham ***Software Lead***

Please write me!

Chelsie Martinear ***Customer Liaison***

Chelsie is a computer science and mathematical sciences major who has an extensive background in programming, numerical computation, and technical writing. Additionally, she has over five years of experience in customer relations. As a valuable member of this team, she is dedicated to the development of PoissonBarrel and is determined to put her best effort into delivering and presenting a polished product to the customer.

Joseph Richardson ***Scrum Master***

Joseph is returning to computer science after completing bachelor's and master's degrees in history, with additional background in classical languages and English literature. His liberal arts roots bring versatility and unique perspectives and skills to his work in information technology. Some of his fortes in computers include regular expressions, data mining, databases, and scripting languages.

Justin Traglia ***Product Owner***

With over four years of software engineering experience, Justin is qualified to work on this project. He has experience working with small groups, agile programming development, version control, and documentation. With a completed minor in mathematics, Justin has

the ability to help develop many of the algorithms for this project, and should be a valuable member of the team.

5 Risk Management

As with all projects there are risks. Our team has conceived of eight possible risks. There will be a description of each risk, along with a plan of what to do if it were to happen.

Death or Dismemberment of Team Member(s)

Chance	*Minimal*
Severity	*Catastrophic*
Description	If any of the group members were to die or lose body parts essential to working on this project.
Contingency	Distribute the team member(s) share of the project to the rest of the team. If all team members were to become unable to work, this project would be the least of our concerns.

Negligence of Group Member(s)

Chance	*Possible*
Severity	*Marginal*
Description	If any of the team member(s) were to "slack off" and not do their part of the project.
Contingency	We would contact the customer and inform them of the issue. If the issue is not solved, we would fire the team member(s), distribute their share of the project to the rest of the team, and continue working on the project.

Unexpected Obligations Outside of Project

Chance	*Possible*
Severity	*Marginal*
Description	Since every member of the team is a full-time student, we all have obligations outside of this project. Obligations such as exams and other class projects might temporarily prevent a team member from working on their tasks.
Contingency	If this were to happen to a team member, their tasks would be temporarily assigned to team members without such obligations. Once free, the aforementioned tasks would be re-assigned back to the original team member.

Team Member(s) Unavailable for Meetings

Chance	*Very Possible*
Severity	*Marginal*
Description	Due to the team member's busy schedules, there might be instances when certain members are unable to attend group meetings. At times, the entire team may be unable to gather for a meeting.
Contingency	If the team is unable to meet in person, we will have online meetings using *Slack* and/or *Skype*.

Poor Organization/Lack of Leadership

Chance	*Minimal*
Severity	*Serious*
Description	If tasks are not properly broken up and reasonably assigned, the project may become unorganized and as a result become difficult to develop. Similarly, the team may lack direction if there is no leader.
Contingency	If the project becomes unorganized, we will temporarily stop development to re-organized the project. For direction, the team must work together to create new tasks and move the project forward.

Unable to Understand Math and Algorithms

Chance	*Possible*
Severity	*Serious*
Description	Some of the statistical measures are unknown to the team. If we are unable to understand these measures, we will not be able to program them.
Contingency	If after research we still do not understand anything, we will contact a mathematics professor and ask for help.

Challenging GUI Framework

Chance	*Very Possible*
Severity	*Critical*
Description	The project language, Python, does not have many good GUI frameworks. As a result, creating the graphical interface might be difficult.
Contingency	Spend more time researching how to use the GUI framework. Attempt to overcome it's faults.