WASHINGTON STATE UNIVERSITY
VANCOUVER

Software Requirements Specification

For

Discrete Probability Calculator

Version <2.0>

Prepared by

Group Name: Team 2

Patrick Tsai	**011709316**	**patrick.tsai@wsu.edu**
Zhicheng Zhou	**011735296**	**zhichengzhou123@gmail.com**
Vuochlang Chang	**011706492**	**Vuochlang.chang@wsu.edu**

Date: **12/16/2020**

Contents

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Patrick Tsai Zhicheng Zhou Vuochlang Chang	First draft of the Software Requirements Specifications (SRS) for the Discrete Probability Calculator. This is our initial attempt at filling in all the required sections of the SRS	11/06/2020
2.0	Patrick Tsai Zhicheng Zhou Vuochlang Chang	• Reformatted document to IEEE style • Changed citation style • Changed Use-Case diagram • Changed the project diagram in section 2.1 • Reduced word length of section 2.1	12/06/2020

1 Introduction

Almost every decision humans' make relies on calculating a probability. We make quick probability calculations for simple everyday actions such as crossing the street, we may consider how far away the next car coming toward us is, how wide the street is, how fast we normally walk, we then quickly analyze these variables before we take our next step. Probability is an integral part of decision making, and a fundamental part of human life. However, most of us do not know the formal way of calculating probability, and therefore there is opportunity for all of us to make better informed decisions.

The Discrete Probability Calculator is a simple website-based application that user can use to calculate the probability of the discrete distribution. This section will cover the general view of the software including its purpose, scope, intended audiences, terms definitions, document conventions for this paper. The overall goals of the system are:

1. To provide easy access to the users to calculate discrete probabilities; Users can access DPC website through any browsers whether from their smartphones or computers
2. To provide as another option from a TI-84 calculator or other versions
3. To help users to understand each probability better with the Help sections
4. To improve our understanding of building a web application that interacts with user input.

The DPC system is intended for study purposes, its intended use is for student/academic questions. However, the DPC system is NOT intended to be used as a tool to make any real-life decisions, as that any probabilities derived from the program should not be used to make important life decisions.

1.1 Document Purpose

The Discrete Probability Calculator v1.0 purpose is to allow users to calculate discrete probabilities included: Binomial distribution, Hypergeometric distribution, Negative Binomial distribution, Poisson distribution, and Geometric distribution. Likewise, DPC also helps students and researchers to understand the meanings of different kinds of discrete distributions, and how to apply them appropriately into real life examples.

On the other hand, the purpose of this SRS document is to explain the usage of the system, layout all required functions, and outlines the server and client-side requirements needed in order for the program to operate.

1.2 Product Scope

The software itself will help the user to calculate the probabilities and learn more about each probability if needed.

The Discrete Probability Calculator (DPC) will be using the user's input to implement the data and display it back to the user. Besides, there will be a brief explanation for each distribution which will help users to understand better. Also, the benefit of this software is to

provide easy access to users to calculate discrete probabilities. Instead of carrying a graphing calculator, the user can access the DPC website with their smartphones or computers through any browsers.

1.3 Intended Audience and Document Overview

The intended audience for this Discrete Probability Calculator will be the client and the professor. This document will benefit developers, testers, program maintainers, clients, and professors to understand the all the requirements that will be needed to build this **DPC** software application. For developers, testers and program maintainers, this SRS document will benefit those who might need to adjust or increase efficiency of the DPC program.
For clients, this SRS document will help them to understand the functionality of the DPC system and uses of the DPC itself.

The rest of this **SRS** will cover over the Overall Description, Specific Requirements and Other Non-functional Requirements of the system. Each section will be included subsections that will describe the section thoroughly.

1.4 Definitions, Acronyms and Abbreviations

TERMS	DEFINITION
DPC	Discrete Probability Calculator
Input	The given data from the User
SRS	Software Requirement System
User	The person who is using the DPC software
Discrete Probability Distribution	Discrete Probability distribution describes the occurrences of every possible value of a random discrete variable, discrete means that the random variable can only take on some non-negative, non-decimal integer value such as: 0 -> positive infinity
Binomial Distribution	Binomial distribution counts the number of successes in some repeated independent experiments, each trial can only consist two possible outcomes: SCCUESS OR FAILURE. The parameters for Binomial distribution are: 1. The number of trials. 2. The number of successes. 3. The probability of a success on a given trial (the probability of a success may not vary)

Hypergeometric Distribution	Hypergeometric distribution is similar to Binomial distribution but with the variance in the probability of a success. In Hypergeometric distribution, the given probability changes for each trial due to the replacement of samples. The parameters for Hypergeometric distribution are: 1. The number of items in total. 2. The number of items that are classified as succuss in total items. 3. The number of items in the chosen sample. 4. The number of items that are classified as succuss in the chosen sample. 5. The number of success of failure that needs to be found.
Negative Binomial Distribution	Negative Binomial distribution is looking for the probability of the number of trials takes to produce n successes. The parameters for Negative Binomial distribution are: 1. The number of successful trials. 2. The kth trial where the number of successful trials happen on.
Poisson Distribution	Poisson distribution is counting the probability of some number of outcomes can occur during a given time interval (usually represented as μ) The parameters for Poisson distribution are: 1. The mean (expected value) of the random variable. 2. The number of outcomes.
Geometric Distribution	Geometric distribution is counting the probability of the number of trials on which the first succuss can happen. The parameters for Geometric distribution are: 1. The number of trials that takes for the first success to happen. 2. The probability of one success can happen.

1.5 Document Conventions

- Formatting Conventions:
  - Font: Arial
  - Size: 12
  - Single spaced
  - 1” margins
- Naming conventions:
  - Section: Arial font, size 18, Bold
  - Subsection: Arial font, size 14, Bold
  - Acronyms: Arial font, size 12, Bold

1.6 References and Acknowledgments

[1] *"How to Cite References: IEEE Documentation Style", IEEEDataPort, Available:* https://ieee-dataport.org/sites/default/files/analysis/27/IEEE%20Citation%20Guidelines.pdf. [Accessed Oct 27, 2020].

[2] "Usage share of operating systems", Wikipedia, Available: https://en.wikipedia.org/wiki/Usage_share_of_operating_systems. [Accessed10/24/020].

[3] "Desktop Browser Market Share Worldwide Sept 2019 - Sept 2020" Available: https://gs.statcounter.com/browser-market-share/desktop/worldwide [Accessed 10/24/2020].

[4] "Usage share of operating systems", Wikipedia, Available: https://en.wikipedia.org/wiki/Usage_share_of_operating_systems [Accessed 10/24/2020]

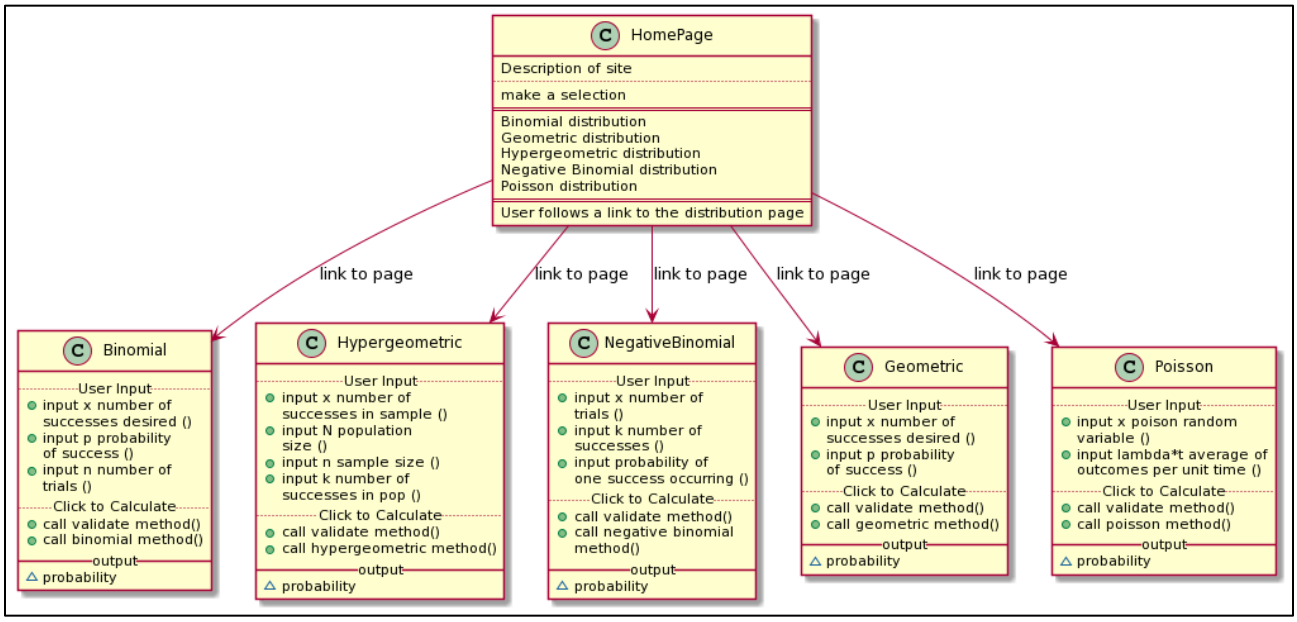
[5] "Mobile Browser Market Share Worldwide Sept 2019 - Sept 2020" Available: https://gs.statcounter.com/browser-market-share/mobile/worldwide [Accessed 10/24/2020]

2 Overall Description

2.1 Product Perspective

The Discrete Probability Calculator v1.0 purpose is two-fold. First it provides a freely accessible tool that allows anyone to calculate a discrete probability. Second, since the application is freely accessible, we hope formal probability calculations will become common practice and will lead to better informed decision making for the general public.

The Discrete Probability Calculator v1.0 web-based software is a new, self-contained product. To a certain degree, we believe it will replace graphing calculators, such as the TI-84, which are currently used to calculate discrete probabilities. The Discrete Probability Calculator v1.0 places focus entirely on discrete probability computations, and therefore the functions are easy to find, use and understand. Users accessing the Discrete Probability Calculator website are presented the option to calculate 5 different Discrete probabilities (Binomial, Geometric, Hypergeometric, Negative Binomial and Poisson). Once a user selects an option, they are taken to a page dedicated to the function they chose. Users will be presented with the probability equation of their choice, helpful information about the equation, a short description about usage. Users will also be presented with form inputs.
The inputs will be associated with the equation variables they must enter for the program to run their probability distribution and output an answer. After entering all the necessary variable inputs and selecting the “Result” button, the user will be presented with the program’s calculated probability according to the User’s desired probability distribution. User’s may return to the home page or any of the other probability distribution pages from the site menu located on the left of all pages. *(See the following UML image for site usage flow chart)*



2.2 Product Functionality

1. User will access the website landing page (functionality dependent on server accessibility).
2. User will choose which distribution to use by selecting a clickable link (functionality based on html and ability to link to separate webpage):
    - *Binomial distribution*
    - Hypergeometric distribution
    - Negative Binomial distribution
    - Poisson distribution
    - Geometric distribution
3. User must input the required numeric data for variables into the system (functionality dependent on html web form and transferring the inputted data to a JavaScript program):
    a. *Binomial distribution*
        i. *Variables: x, p, number of trials*
    b. Hypergeometric distribution
        i. Variables: x, N, n, k
    c. Negative Binomial distribution
        i. Variables: x, p, number of trials
    d. Poisson distribution
        i. Variables: lambda, x
    e. Geometric distribution
        i. *Variables: probability, x*
4. Software will calculate the probability from the inputted data and return the probability result to the user (functionality is based on JavaScript functions associated with each probability calculation).
5. The User may return to the home page or any other probability distribution page from the site menu located on the left side of all sub-pages.

2.3 Users and Characteristics

The Discrete Probability Calculator v1.0 calculates probabilities for 5 different discrete probability distributions. The calculator has one access level, “guest”, anyone can use the website at any time (except during maintenance), for any reason. Initially, the intended users were classified as academics, students, researchers and educators. However, there are potentially other types of users.

Probabilities may be used for predictions, and therefore we can think of many other uses and users. Gambling uses probability in many instances. Professionals, such as meteorologists, use probability to predict weather patterns. Due to the availability of smart phones, the Discrete Probability Calculator v1.0 may become a tool of gamblers, tradesmen, and news agencies trying to predict outcomes in local political races, weather patterns, card games etc.

Currently, the purpose for the Discrete Probability Calculator v1.0 is to help students and researchers in academic fields. Therefore, they are our most important user group. However, interests may change (i.e., generating income from the website may become more important) and in such cases other types of users may become most important for advancing any changed interests.

List of users - (most important to least):

Academic
- Student: STEM students
  - Uses program for homework.
- Research: Mathematics, Biology, Chemistry, Sociology
  - Uses program to answer questions around populations (Biology, Sociology)
  - Uses program to answer questions about atomic half-life (Chemistry)

Government
- Policy Researchers
  - Uses program to promote policy proposals
- Oversight committees
  - Uses program to assess the viability of current policy

Professional
- Data analytics
  - Uses program to assess future trends
- Meteorology
  - Uses program to determine validity of predicted weather patterns
- Sportscasters
  - Uses program to assess the viability of teams in competition

Gambling
- Card games, Dice games, Lottery, Sports Betting
  - Uses program to understand and determine which games played have the best odds of winning

2.4 Operating Environment

We have determined two main environments that users will access the Discrete Probability Calculator v1.0. The first is on a desktop computer. Current statistics show that, “in the area of desktop and laptop computers, Microsoft Windows is the most commonly installed OS...”

Desktop Operating System	**Market Share (approximately)**
Microsoft Windows	77 – 87.8%
Apple macOS	9.6 - 13%
Google Chrome OS	6%
Linux distros	2%

https://en.wikipedia.org/wiki/Usage_share_of_operating_systems Accessed 10/24/2020

Additionally, current market share statistics on desktop browsers show Chrome account for largest percentage of market-share:

Desktop Browser	**Market Share**
Chrome	69.66%
Safari	8.74%
Firefox	8.17%
Edge	5.54%
Internet Explorer	2.51%
Opera	2.3%
https://gs.statcounter.com/browser-market-share/desktop/worldwide Accessed 10/24/2020	

We would like to provide our program to the greatest number of people possible, therefore our target desktop audience uses Windows OS with Google Chrome web-browser.

Additionally, we believe users will also access the Discrete Probability Calculator v1.0 on mobile devices, specifically smartphone browsers. Currently, Android leads market share.

Smartphone OS	**Market-Share**
Android	54.16%
iOS	12.37%
Windows	11.79%
All others	21.66%
https://en.wikipedia.org/wiki/Usage_share_of_operating_systems Accessed 10/24/2020	

Mobile browser market share breaks down as follows:

Mobile Browser	**Market Share**
Chrome	64.63%
Safari	22.7%
Samsung Internet	6.44%
UC Browser	2.34%
Opera	1.91%
Firefox	0.47%
https://gs.statcounter.com/browser-market-share/mobile/worldwide Accessed 10/24/2020	

Therefore, we are targeting a mobile device audience by catering to users on Android devices using the Google Chrome web-browser.

The Discrete Probability Calculator v1.0 source code will be hosted on a cloud server. We have chosen a GoDaddy web-server host plan. The source code will need to be compatible with the current server configuration:

Apache Version	2.4.43
Architecture	x86_64
Operating System	linux

Shared IP Address	107.180.119.112
Kernel Version	2.6.32-954.3.5.lve1.4.79.1.el6x86_64
CPU	1
Disk Space	60 GB
Bandwidth	unlimited
Physical memory	1 GB
Iops	4,096 bytes

Should the source code be hosted on a different server, the server will need the following minimum requirements: 30 GB disk space, 512 MB RAM, 1 CPU, using the server specifications from the table above.

To summarize, our product development will focus on functionality and maintainability for Windows OS/Google Chrome, and Android OS/Google Chrome Users. The source code will be hosted on a Linux based Apache webserver.

2.5 Design and Implementation Constraints

The Discrete Probability Calculator v1.0 functionality is written in JavaScript, and its webpages will be written in html and CSS. User's will have to enable JavaScript in their browsers if it is not already enabled.

Calculation response time will vary depending on the complexity of the problem. However, it is important for the Discrete Probability Calculator v1.0 to complete calculations as efficiently as possible. Our target is processing calculations requiring 10 trials or less at the same speed it takes the webserver to render a single page request. Subsequent time requirements are based on this measure. (i.e. a 20-trial calculation to take twice as long as one page to be rendered by the webserver.)

Our current server specifications are as follows (same table appears in section 2.4):

Apache Version	2.4.43
Architecture	x86_64
Operating System	linux
Shared IP Address	107.180.119.112
Kernel Version	2.6.32-954.3.5.lve1.4.79.1.el6x86_64
CPU	1
Disk Space	60 GB
Bandwidth	unlimited
Physical memory	1 GB
Iops	4,096 bytes

The Discrete Probability Calculator v1.0 does not require a database or login for users. The source code files will be hosted on a GoDaddy shared server. Therefore, security will only be applicable to the GoDaddy webserver. The webserver hosts html, CSS and JavaScript files. Two factor authentication has been enabled on the GoDaddy server login. The current team of developers will be the only person(s) with access to the Discrete Probability Calculator v1.0 webserver, and the site's functionality will be the responsibility of the development team, or until a future agreement is reached with an interested party.

Text on the website is written in English. This will constrain our audience regionally or will depend on the user's use of translation application.

As stated in section 2.4 Operating Environment, functionality and maintainability will be focused on Windows OS/Google Chrome and Android/Google Chrome user experiences. Hence, users not on these platforms may find limitations in operation.

2.6 User Documentation

There are two types of documentation needed for the Discrete Probability Calculator v1.0.

- User centered use documentation
    - This will take the form of help pages associated with each probability function. The help pages will further explain the selected probability function and which variables need to be inputted in order for the program to calculate the probability.
- Developer centered programming documentation
    - SRS document
        - Tech specs of server
        - Diagram of pages, how they are connected and their dependencies
        - List of classes, methods and their inputs

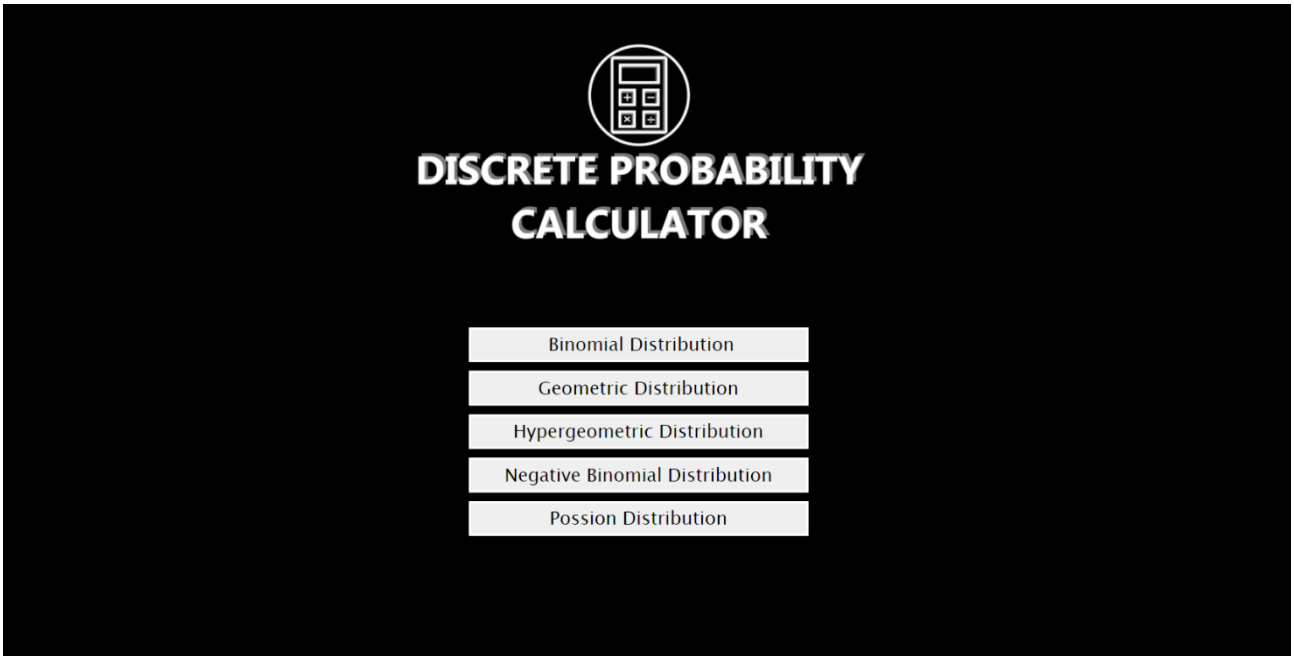
2.7 Assumptions and Dependencies

- JavaScript functionality will not deprecate the functionality we are using
- GoDaddy hosting will not change from Linux-Apache architecture
- The tags and class styles used in html and CSS files will not be deprecated
- html will continue to render JavaScript output
- User's will be accessing site from:
    - Windows OS / Google Chrome browser
    - Android OS / Google Chrome browser
- User's will have JavaScript enable on browser

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces



1. The initial interface will be a page that displays a logo of a calculator along with different distributions which the user can choose from, once the user has clicked on the desired distribution, a corresponding page that allows the user to enter input will show up along with a paragraph of what each distribution means and what the meanings of each inputs are.
2. Now that user can enter inputs for the distributions that want to calculate, there will be some restrictions on the range of the inputs depending on what each input means, if the user has entered some irrational inputs, the calculator will display a warning saying there are something with the entered inputs, until all the inputs are entered correctly within the reasonable range the calculator won't allow user to proceed.
3. If all the inputs are entered correctly, then the user can click on the "calculate" button to see the results of the calculation based on the given inputs.

3.1.2 Hardware Interfaces

1. This web application will be supported on Windows, Macintosh, IOS and Android.
2. This web application will require user to use mouse to make clicks to interact and the keyboard for typing in inputs for distribution calculation. No other hardware is required.
3. This web application will also require monitor to display all the reactions between the user and the application itself.

3.1.3 Software Interfaces

1. This application will be supported on most the popular browsers, for example: Safari, Chrome, Firefox, IE etc.
2. This application will be supported on Windows, Mac OS, Linux, etc.
3. JavaScript will be used to handle the calculations, interactions between functions/classes and any back-end process. Html and CSS will also be used for designing the appearance and formatting the outputs/messages.
4. This application will identify data shared across software components:
   - Inputted data from user
   - Message from system for invalid data / warnings
   - Data from program to the monitor

3.1.4 Communications Interfaces

1. This application will be supported on most the popular browsers, for example: Safari, Chrome, Firefox, IE etc.
2. This application will use cable or WIFI to send, store and process data that use would like to calculate.
3. This application will use the standard application layer protocols HTTP which defines the how the application communicates through the browser.

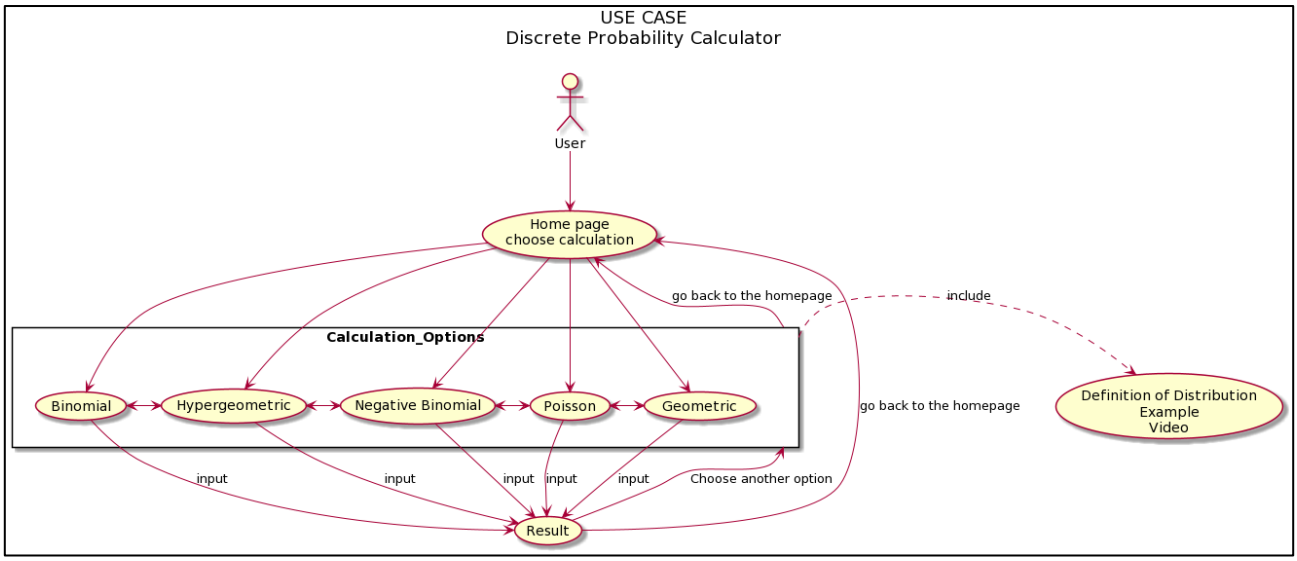
3.2 Functional Requirements

1. User will first access the website landing page, the purpose of this is for the user to choose which distribution to they want to work with by clicking on it, it then will direct the user to a new page that corresponding to the chosen distribution, a list of calculations that this calculator can do is:
   - Binomial distribution
   - Hypergeometric distribution
   - Negative Binomial distribution
   - Poisson distribution
   - Geometric distribution
2. To prevent any bad inputs that can result in error, this calculator will ask user the user to enter correct data within the reasonable rage into the system (there will be introduction on the correct range of each input and what they mean).
   The correct input range and explanations are as the following:
   - Binomial distribution
     - Random Variables: x, number of trials: n, probability of the event on a given trial: p
   - Hypergeometric distribution
     - Random variables: x, number of trials: N, number of success: k, number of items in the sample: n, number of successes in the sample: m,
   - Negative Binomial distribution
     - Random Variables: x, number of successful trials: n
   - Poisson distribution

    - The kth trial where the number of successful trials happen on: x, number of occurrences during given time interval: lambda
  - Geometric distribution
    - Number of trials take for the first success to occur: x, probability of success in each trial: p
3. The calculator will then read the data and calculate different distributions accordingly. The type of calculator will be known to the calculator before the calculation starts.
4. The calculator will identify the bad inputs that are out of range or in inappropriate format I.e., fractions, if a bad input is typed by the user, the user will be prevented from proceeding until the inputs are entered correctly.
5. The “learn more” button will be shown after each successful calculation, it will direct the user to some external web pages that explains the distribution more thoroughly.
6. In the site menu, the user may also choose the return to the home page or do another calculation (this will bring the user back to the choosing type of distribution stage)

3.3 Behavior Requirements

3.3.1 Use Case View



4 Other Non-functional Requirements

4.1 Performance Requirements

1. A 10-trial probability calculation should render a result in same amount of time it takes the webserver to render a page
    a. Calculation times should follow linearly, i.e. a 20-trial calculation should take no more time than twice the time it takes the webserver to render a page.
2. There will be a limit to the size of integers used in the inputs. (n < 100,000,000)
3. No calculation should exceed 7 seconds
    a. This is to keep server processing low
4. Pages that do not require probability calculations should be rendered in less than 1 second.
    a. In order to maintain acceptable user experience

4.2 Safety and Security Requirements

The Discrete Probability Calculator v1.0 holds no obligation to, or responsibility for persons and actions taken due to use. Probabilities are a measure of likelihood of outcomes and are not to be misinterpreted as a guarantee of any particular outcome. Any action taken by any person(s) who have used of the Discrete Probability Calculator v1.0 are the responsibility of the person(s) alone.

No user information is stored on the website, and no user information will be sold to third party. However, the Discrete Probability Calculator v1.0 retains the right to display advertisements on the site if we so choose.

Security is limited to access to the cloud-based webserver. Therefore, all login security will be handled by the cloud-based server host (GoDaddy). The account will at all times have two-factor authentication enabled.

4.3 Software Quality Attributes

The Discrete Probability Calculator v1.0 contains several characteristics pertaining to adaptability, availability correctness, flexibility, maintainability, portability, reliability, reusability, robustness, testability, and usability.

4.3.1 Reliability / Availability / Reliability / Maintainability / Usability

- DPC has a dedicated maintenance time windows scheduled
    - Maintenance will only occur between 12:01 am to 2:59 am on the first Wednesday of each month.
    - Unless a zero-day update is needed, then it is contingent on the immediacy of the update.
- DPC has a target time for algorithm execution

- As stated in section 2.5, "… calculations requiring 10 trials or less at the same speed it takes the webserver to render a single page request. Subsequent time requirements are based on this measure. (i.e. a 20-trial calculation to take twice as long as one page to be rendered by the webserver.)
- The average load time for a webpage, in order to reduce bounce rate, is 1 -3 seconds, and is our measure for calculation time. Use of server's AWstats allows us to verify time constraints are met.
- DPC has a 15-digit max size on user keyboard input and calculated result data, this increases efficiency of the calculation, however it potentially decreases accuracy.

- DPC uses an open source code base – JavaScript, html, css
  - The use of open source code base is important in terms of maintainability. Developers have open access to the environments in which the DPC was/is coded. Linux based Apache webserver, JavaScript, html, css codebase.
  - Testing and production sites should always use the latest stable release of Linux, JavaScript, html, css
    - Verifiable by version checks on each codebase and server
- DPC uses JavaScript functions to handle the calculations
  - The developers have written functions to handle factorial and combination calculations.
  - These functions have thresholds for accuracy, and depend on user inputted data to be within a certain range
    - The DPC developers have built in functions that check user inputs to make sure they are within these thresholds.
    - If $x <= 5$ then $n < 100$.

4.3.2 Portability / Correctness / Flexibility / Reusability

- DPC uses an open source code base
  - The use of Linux based Apache webserver, JavaScript, html, css reduces obstacles for porting to different environments.
  - Folder structure also helps define structure when porting
    - JavaScript kept in js/ folder
    - css kept in css/ folder

- Reuse of JavaScript functions within different pages on website
  - The DPC developers have created a number of functions that check user input and are used across all distribution pages, they include:
    - Check that input is a number
    - Check that number of trials are greater than random variable
    - Check that all the user inputs are entered before making calculation

5 Other Requirements

N/A – no other requirements

Appendix A – Data Dictionary

TERMS	DEFINITION
DPC	Discrete Probability Calculator
Input	The given data from the User
SRS	Software Requirement System
User	The person who is using the DPC software
JavaScript	an object-oriented computer programming language which can be used when implementing a web browser
CSS	Cascading Style Sheets, a computer language use to editing the display and laying of web pages (HTML)
HTML	Hyper Text Markup Language, standard language use to create a webpage, commonly focus on the structure of web pages
GB	Gigabyte, a unit of storage in the computer term, 1GB = 1,073,741,824 bytes
MB	Megabyte, a unit of storage in the computer term, 1 MB = 1048576 bytes
Disk space	the available space on the disk to store data
RAM	Random access memory, the short term memory of a computer
CPU	Central processing unit, principle component of the computer system
Discrete Probability	Discrete Probability distribution describes the occurrences of every possible value of a random discrete variable, discrete means that the random variable can only take on some non-negative, non-decimal integer value such as: 0 -> positive infinity
Binomial distribution	Binomial distribution counts the number of successes in some repeated independent experiments, each trial can only consist two possible outcomes: SCCUESS OR FAILURE. The parameters for Binomial distribution are: 1. The number of trials. 2. The number of successes. 3. The probability of a success on a given trial (the probability of a success may not vary)

Hypergeometric distribution	Hypergeometric distribution is similar to Binomial distribution but with the variance in the probability of a success. In Hypergeometric distribution, the given probability changes for each trial due to the replacement of samples. The parameters for Hypergeometric distribution are: 1. The number of items in total. 2. The number of items that are classified as succuss in total items. 3. The number of items in the chosen sample. 4. The number of items that are classified as succuss in the chosen sample. The number of success of failure that needs to be found.
Negative Binomial distribution	Negative Binomial distribution is looking for the probability of the number of trials takes to produce n successes. The parameters for Negative Binomial distribution are: 1. The number of successful trials. The kth trial where the number of successful trials happen on.
Poisson distribution	Poisson distribution is counting the probability of some number of outcomes can occur during a given time interval (usually represented as μ) The parameters for Poisson distribution are: 1. The mean (expected value) of the random variable. The number of outcomes.
Geometric distribution	Geometric distribution is counting the probability of the number of trials on which the first succuss can happen. The parameters for Geometric distribution are: 1. The number of trials that takes for the first success to happen. 2. The probability of one success can happen.

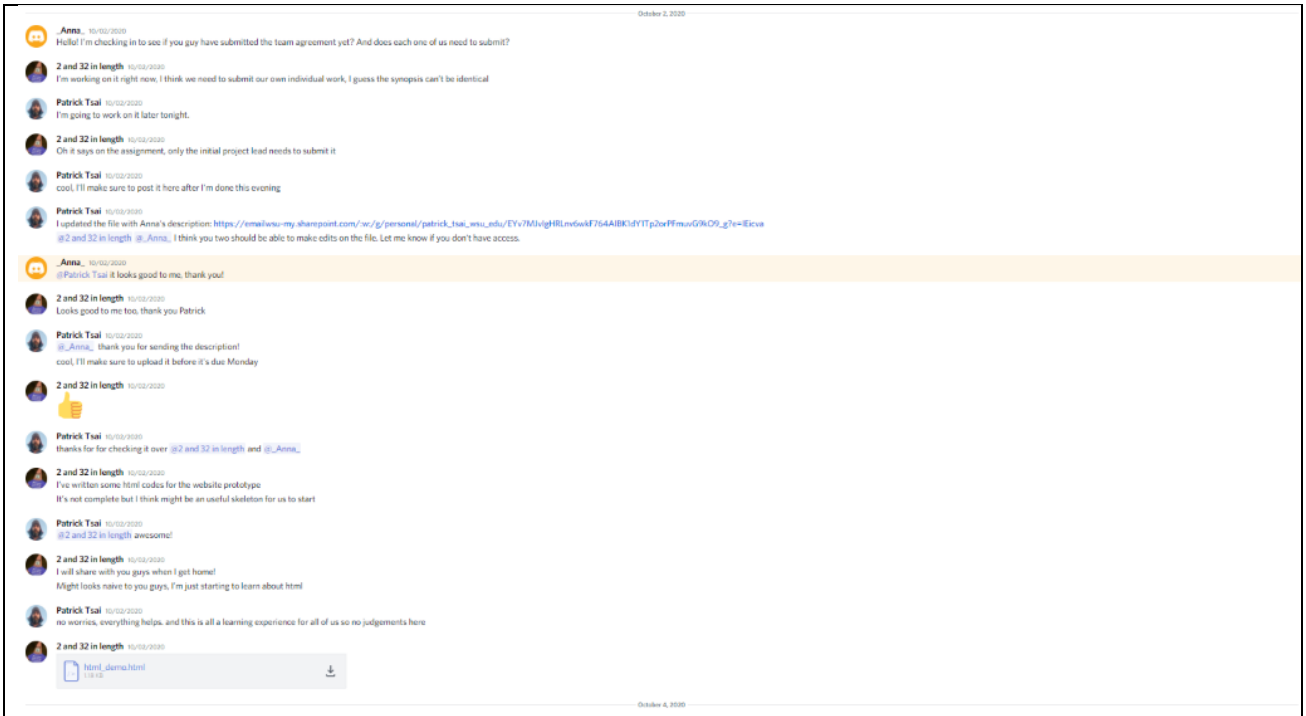
Appendix B - Group Log

CS320 – Meeting Notes

09/30/2020 – Wednesday, weekly zoom meeting

- Decided on a project / coding language / centralized version control system
  - Anna's idea of probability calculator
  - Javascript with html and css for display on web
  - Github
- Description of project
  - Web-based calculator, will calculate probability for the following functions:
    - Binomial distribution
    - Hypergeometric distribution
    - Negative Binomial distribution
    - Poisson distribution
    - Geometric distribution
  - Users will be given a list of all distribution options and explanation of the application on the website landing page.
  - Based on the user's distribution choice, they will be directed to a sub-page displaying the distribution formula and a number of inputs specific to the function.
  - The user will then input the required numerical data for the distribution function.
  - The code will generate a probability based on the data entered for the specific distribution.
  - We plan to break each distribution function into separate methods.
  - We'd like the design to be simple and intuitive.

10/02/2020 – Friday, weekly discord written update

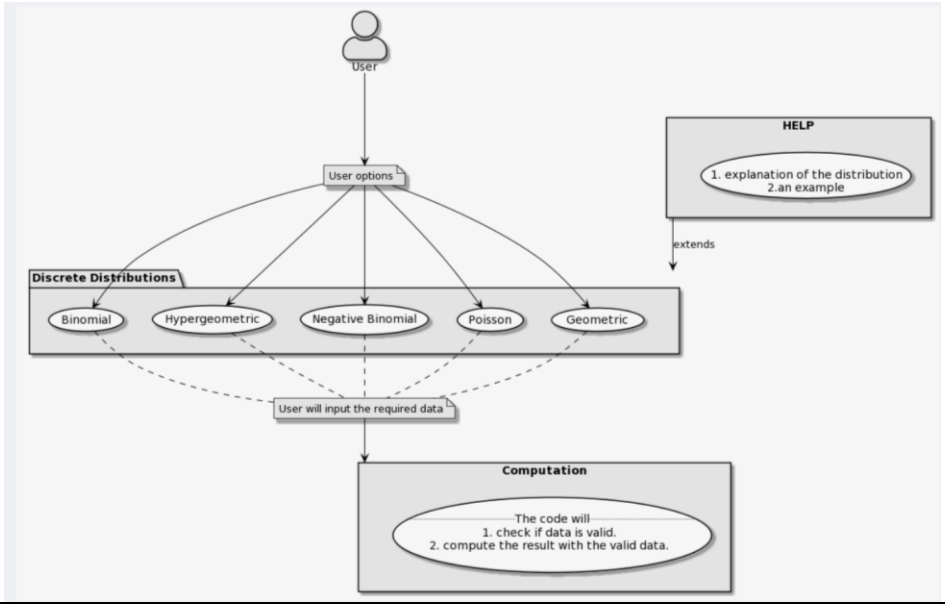


10/07/2020 – Wednesday, weekly zoom meeting

- Talked about if our project fulfills the requirements of the project
- Confirmation about the general idea of the project

10/21/2020 – Wednesday, weekly zoom meeting

- Confirmation that everyone has access to the project repository on GitHub
- Discussion over the SRS-template and plan to get it done by the end of next week
- Divided SRS sections section 1 - Anna, 2 - Patrick, 3 - Z
- Overview on the USE-CASE diagram that Anna has made-shared on GitHub



10/28/2020 – Wednesday, weekly zoom meeting
- Improve the SRS all together

10/30/2020 – Friday, zoom meeting
- Work on SRS, go through the SRS, and define what else to change
- Anna will start the first page of the website to include in section 3.1
- Initial codes have pushed to GitHub and the sample website is working to access with only the first page of the website.
- http://dpc.patricktsai.com/ - the website for DPC

11/4/2020 - Wednesday, weekly zoom meeting
- Work on SRS
- Anna went over web design
- Reviewed every section together

- Section 1 Anna
- Section 2 and 4 Patrick
- Section 3 Z

11/6/2020 - Friday, zoom meeting
- Final review on SRS
- Filled out remaining subsection 4.3
- Fixed layout
- Anna showed application functionality and design

12/16/2020 - Zoom meeting
- Dec. 16, 11am - 12:00pm
- Review SRS v2 an Final Report
- Patrick will submit them on BlackBoard