Assignment 2: Report

Andrew Thom
Andrew Zurn
Chris Norby
Isaiah Schultz

September 21, 2012

Contents

1 Introduction & Purpose

The purpose of this project was to help the four members of this group become comfortable with writing technical reports for *CSCI 330: Software Engineering*, an upper-devision computer science course offered at *Saint John's University* in Minnesota.

To demonstrate this, the report that follows represents a hypothetical situation involving our team of experts giving the now quite popular *Netflix* recommendations for building the software architecture behind their product. As experts, we will be referencing architecture prerequisites from the course textbook "Code Complete." (See Section 2 for details).

Please note that while some of this report was compiled using information attained from good, old-fashioned research, the majority of it is based on the collective imaginative ingenuity of the team.

2 Materials

The following is a brief explanation of the resources used for this project.

Our main reference point from which we began was the course textbook "Code Complete" (Second Edition) by Steve McConnell. 10 fo the 20 architecture prerequisites were chosen from section 3.5 of the text for discussion in this report. The full list of prerequisites can be found on pages 45-53. Our personal in-class notes were also used to supplement this material.

For our research and composition of this report, each of our personal computers were used. They all had varying specifications, but the specific specs of the computers did not in any way affect the outcome of this project. Because of this, further details regarding them will be omitted from this report.

As a further learning tool, we chose to use LaTeXto compose this report. Primarily, the Eclipse plugin version of LaTeXwas used.

Supplied by Dr. Mike Heroux, instructor of the course, we used an online description as a guide for formatting this report[**?**].

3 Methods

Because the goal of this project was to become familiar with writing technical reports, we started by reading through the guidelines that provided. We discussed ways that we could combine any of the points outlined on the website so to ensure that each section had adequately substantial content.

For example, "Introduction," "Problem, background," and "Purpose, objectives, scope" were decidedly combined into our Section 1. As individual sections, we felt that they would contain a mere sentence or two which felt like a lot of wasted space. Therefore those sections were logically combined.

Our next step was to review the 20 architecture prerequisites mentioned in Section 2 of this report. As assigned by Dr. Heroux, we picked 10 of the most relevant prerequisites to discuss within the scope of our hypothetical situation:

1. Program Organization

2. Major Classes

3. Data Design

4. Performance
5. Business Rules
6. UI Design
7. Security
8. Scalability
9. Internationalization
10. Input/Ouput

With similar reasons for combining a number of suggested sections into one introductory section for this report, we decided to combine "Data Design" and "Performance" into one section as well. As can be seen in section 4.3, this decision made was logical for this situation.

We finally discussed how each prerequisite applied to Netflix's software product. Section 4 outlines the results of that discussion.

4 Results

4.1 Program Organization

Netflix can first be broken down into two major systems, one system that covers the online instant streaming, and the other one that covers the DVD-ship-to-home rental service. Both are comprised of a various major subsystems that are bulk of the current end-user service that is known to us as commercial users.

First we can break down the online streaming service. This service is comprised four subsystems. The first component of the online service is video storage, or how the videos that can be viewed by the Netflix user are stored. Our next component is the users profile management, which takes care of storing valuable data that expands from financial information to what is currently in the users queue. From there, there is another subsystem that is involved in managing the connection between the user interface and the database. Finally, there is the UI and service management, which controls the way content is viewed, whether that be through the website or through an application stored on the local system.

The DVD rental service is not nearly as involved as the online service is, as it is comprised of two major systems. One system, which manages the inventory currently in the warehouses, and the request and shipping service, which fulfills users requests for a DVD, and then handles the shipping of that DVD.

4.2 Major Classes

We can further break down the subsystems described above, mainly the online streaming service, into major object classes that take care a lot of the individual tasks and processes involved in the Netflix online system. There are three main classes within the Netflix system, those being of class Video, User, and DB Management. From those classes, the Video class can be broken down into TV Show and Movie subclasses. User is a super class for subclasses Customer and Admin, and Database Manager is a class that does a lot of backend work for maintaining and serving different content.

4.3 Data Design & Performance

As Netflix has a high traffic rate, and because it provides services in large file streaming, the data design and performance of the software is state-of-the-art, in order to maintain customer satisfaction and the quality of their service. As such, the service response time is very minimal, because the company various sites across the world to retrieve and send information to its customers. Because Netflix insures the backend of its services is well supported, the only bottle neck in performance that can be found is the users Internet speed. Netflix tries to condition its media using the latest video formats and codecs, in order to help ease the problematic bottleneck that it can do little in addressing.

4.4 Business Rules

Many of the business rules specified in this section will have to come not from us, the technical experts, but from the business department of Netflix. We can, however, take a guess at some of the things they might say.

There must be a minimum acceptable quality (bitrate) for all videos stored in the database. The verbiage and terminology across the system must be consistent. For example, if we refer to a 'movie' on one page of the website, the same object must not be referred to as a 'video' elsewhere. This will avoid confusion by users as well as programmers. The styling and color scheme across the website as well as any mobile apps must be consistent. This will help the branding of the company. Also, a consistent header and footer must appear on every page.

Each of these things must be specified in business rules files, as opposed to being written directly into the code. This will make it easier to change any of these values.

4.5 UI Design

As mentioned in the business rules section, the User Interface must be consistent across every page with regard to colors, styling, fonts, etc. Additionally, the files that specify the UI must not contain any logic. Doing so would violate encapsulation and make any changes to either the UI or the logic very difficult.

There will be aspects of the UI that will be used many times across different pages of the website, such as headers, footers, menu bars, and frames for movie embedding. It is vital that these common parts are only written once, and then linked to each time thay are used. If the header, for example, is copy-pasted to the top of each UI file and a change of the header is required, the change will have to happen in many different places, which will lead to mistakes and inconsistencies.

4.6 Security

The videos for Netflix will be using Microsoft Silverlight. This will aid us in keeping the copyrighted content on the website secure, due to some built-in security functionalities. For example, users will not be able to directly download the movies, but only stream them to the website.

Usernames and passwords must be standardized, with a given number of characters as well as possibly numbers, capital letters, and special characters. Also, the master database that holds the movies must be fully encrypted, with all standard security features implemented. All queries to the database must be done with prepared statements so as to avoid SQL injection by Dr. Rahal.

4.7 Scalability

As Netflix has grown from a novelty service for those who wanted the convenience of having DVDs delivered right to their mailbox, the company has had to adapt their business model to meet increasing demand. Also, with the wide adoption of smartphones such as iPhone and Android devices, and with video game consoles becoming more like a home media center device, Netflix has had to create ways to not only deliver DVDs to a mailbox, but also to instantly deliver them to ones living room or a device in their pocket.

In doing that, Netflix has created mobile applications that run on iOS (both iPhone and iPad) and Android, and has also worked with Microsoft, Sony, and Nintendo to integrate the Netflix service into their respective video game consoles. This has allowed Netflix users to have access to their Netflix account anywhere at any time.

Also, Netflix has used a recommendation engine in order to help its users find more movies that they may like based upon the films they have watched, and the ratings that they have given to other movies. While the engine was producing good results, Netflix wanted it to become more accurate. They wanted teams of programmers to take their sample data, and use data mining techniques to come up with better algorithms that would predict movies better than its system, called *Cinematch* . The contest, called the *Netflix Prize*, ended up with one team that had successfully improved over the *Cinematch* algorithm by more than 10%. That team, called BellKor's Pragmatic Chaos, received a prize of $1,000,000.

4.8 Internationalization

As Netflix's Instant Streaming service gained popularity in the United States, many other people around the world expressed interest in the service. Netflix's first international venture began on September 22, 2010, when they launched instant streaming in Canada. By September 2011, Netflix instant streaming was introduced into the Caribbean, Mexican and South American markets. In January 2012, the service was also launched in the United Kingdom and Ireland, marking the first time that the service has gone overseas. Netflix has also announced that they will be launching the service in the Scandinavian countries of Norway, Denmark, Sweden, and Finland before the end of 2012, but as of September 21, 2012, that service has yet to launch. Currently, all of Netflix's operations outside of the United States include *only* the instant streaming service, not the DVD-by-mail service.

Because of the contracts that Netflix has negotiated with the content distributors, the service is not accessible outside of the countries where it has been officially launched. Netflix uses IP address geolocation to determine from where a user is accessing the service. If the user is in a location not supported by Netflix, the instant streaming service is not available.

4.9 Input/Output

There are many different inputs and outputs to the Netflix service. The movies must be added to the Netflix database, which is done manually. Also, the users must use the Netflix web interface to sign up for the service, enter their shipping and billing information, and also determine what movies they want to add to their DVD and Instant Queues. The information on how this is done is handled in subsection 4.5.

The most important output of the Netflix service is, obviously, the movies that are instant streamed to its users' computers, smartphones, video game consoles, or other devices that support Netflix playback. When playing back on the computer, Netflix uses the Microsoft Silverlight technology to stream the video while encoding it to remain secure. Also, the web interface displays information about the videos, including a description of the film, the cast, and its current rating. Administrative tasks are also output via the web interface. These tasks include handling the user's profile, updating personal information, and adding or changing payment details.

5 Conclusions

This concludes our report. While we are sad to relinquish our roles as experts for Netflix, we take solace in knowing that we have completed our first technical report as a team. While the content may not be as perfect as content provided by full-time experts for Netflix, we have a very good grasp on how to apply the structure of this report to real, practical applications for the *Software Engineering* course this semester. As a team, we would like to say, "Assignment 2: Mission accomplished!"