Software Project Management Plan

STV Services

October 4th, 2019

Team Members
Mazin Al Ismaili
Ciara Jones
Justin Balino
Chale Abad
Francisca Okojie

Document Control

Change History

Revision	Change Date	Description of changes
V1.0	10/04/2019	Initial release

Document Storage

This document is stored in the project's Github repository at:
https://github.com/umkc-cs-451-2019-fall/semester-group-assignment-group-8

Table of Contents

1 Overview

1.1 Purpose and Scope

Streaming services such as Hulu, Netflix, and more provide specific networks and shows. With the growing number of streaming services, it can be difficult to choose one that provides the entertainment you want.

The purpose of this project is to simplify the process of determining which streaming service is suitable for the user. Through STV Services, users can easily search and choose networks and/or shows that correspond with their entertainment preferences.

The following is what can be expected from the project:

- Users to login with a valid username and password. Incorrect attempts are rejected.
- Administrators to generate a report of login attempts, popular shows, recent purchases, and user reports/feedback.
- Users, administrators, and anons to see what packages or shows are available, although anons will not have access to price.
- Administrators can add, delete or edit streaming services, prices, networks or shows.
- Users to report a problem or request additional feedback.

The project will not include:

- Video-streaming or previews of the streaming services, networks, or shows.
- Purchase of packages or cancellation or purchases previously obtained elsewhere.
- One-on-one personalized training.

1.2 Goals and Objectives

Project Goals:

1. Give potential customers a simple way to compare available packages across various streaming services.
2. Create a web application that is user-friendly and easy to navigate.
3. Comply with safety and privacy concerns of users.

Project Objectives:

1. Create a database on streaming services and the variables they provide.
2. Create survey that provides information about streaming services' prices, packages, and variables
3. Create a feature that stores users' favorite shows and recommend variables based on the input.
4. Create an interface that users and administrators can use to enter and update streaming services data.

1.3 Project Deliverables

The following items will be delivered to the customer:

1. Source code for the system.
2. Requirements Specification
3. Technical Prototype
4. Architecture Document
5. User and System Guide
6. Test Plan

1.4 Assumptions and Constraints

Assumptions:

1. The team will meet at least once per week.
2. Each teammate will have access to a computer and Visual Studio.
3. A database, web application, and interface will all be worked on.
4. Customer and professor Kendall Bingham will be evaluating our process.

Constraints:

1. The software must run as a web application.
2. There must be communication between the database and web application.
3. The software must be ready by 12/02/2019.

1.5 Schedule and Budget Summary

Schedule Information (Major milestones and deliverables):

09/17/2019 - Iteration #1 Plan Complete
09/20/2019 - Project Charter Due
09/23/2019 - Requirements Document Baselined Due
09/30/2019 - Iteration #1 Complete
09/30/2019 - Overall Project Plan Due
10/14/2019 - Technical Prototype (Presentation and Document)
10/14/2019 - Iteration #2 Complete
10/28/2019 - Project Status (Report In Class)
10/28/2019 - Iteration #3 Complete
11/01/2019 - Architecture Document Due
11/11/2019 - Iteration #4 Complete
11/22/2019 - Test Plan Due
12/01/2019 - User and System Guide Due
12/01/2019 - Team Evaluation
12/01/2019 - Project Results Due
12/02/2019 - Iteration #5 Complete
12/08/2019 - Project Feedback

Budget Summary:
The team was instructed to give an estimate of how many hours they will spend on the project (minimum 3-5 hours/week). The total expense was estimated to be $3,750. The calculations can be seen at Section 3.4.

1.6 Success Criteria

- Teammates gain understanding of how the MVC process works under the ASP.NET framework.
- All high-priority use cases in the requirements specification are delivered before December 2nd.

1.7 Definitions

STV Services - stands for Streaming TV Services, the product or system being addressed within this document

Packages - refers to priced packages that a streaming service may offer, such as choosing a specific range of networks

Survey - refers to the questionnaire users can take to receive streaming service recommendations

Use case – describes a goal-oriented interaction between the system and an actor. A use case may define several variants called scenarios that result in different paths through the use case and usually different outcomes.

Variables - refers to networks, shows, and/or streaming services themselves. This is to summarize all the parts of a streaming service.

1.8 Evolution of the Project Plan

For the beginning of an iteration, tasks may be updated or added. At the end of an iteration, actual effort amounts will be documented.

Therefore, the Section 3.1 is subject to change. Section 5 will be updated once it is addressed in class.

2 Startup Plan

2.1 Team Organization

The assignment itself constrains the team to be classified as a Project Organizational Structure.

Project Manager:	(Chale) Responsible for planning team meetings, submissions, and tracking overall team motivation.
Project Admin.:	(Francisca) Assists PM in management responsibilities. Specifically, turns in other submissions as well and provides project control.
Programmers (5):	(Everyone) Programmers are primary responsible for coding and unit testing modules. They are also expected to take part in architecture planning and review meetings.
Tech Writers:	(Chale) The technical writer is responsible for providing documentation throughout the project regarding the learning and technical process. NOTE: All team members wrote the requirements document.
Database Manager:	(Mazin) The database manager is responsible for setting up, entering and maintaining streaming service data within the database.
Unit Tester:	(Francisca) Learns about, writes, and runs unit tests.

2.2 Project Communications

UMKC computer labs are available for us to use, although most of the team members have computers or laptops to work on. They will use Slack, Github, Trello, email, and/or text to communicate with each other. More specific expected meetings can be seen in Section 4.1.

2.3 Technical Process

The standard software development process will be the following:

1. At least one use case will be assigned to each programmer for each iteration.
2. Individual programmers will research and implement code according to the feature of their iteration.
    a. Proper comments and identification will be maintained by each programmer.
3. A review meeting before the iteration due date will be set.

2.4 Tools

- Programming Language – C#/HTML
- Version Control – source code and written artifacts will be stored in a Github repository.
- Defect tracking – defects and issues will be tracked under Visual Studio's error acknowledgements .
- Build tools – local and main builds will be done using Visual Studio Community.

3 Work Plan

3.1 Activities and Tasks

The following is a breakdown of tasks and their estimates. Some tasks, although ideally fixed, are subject to change with each iteration. Owners will be assigned for each iteration.

Task #/Name	Description	Effort Estimate (Hours)	Actual Effort (Hours)	Planned Start/Stop (Dates)	Actual Start/Stop (Dates)	Dependencies
1. Login Research	Find a proper way to store accounts and properly login.	2		10/1-10/14		Database/Variables holding account.
2. Implement Login	Implement login method.	1		10/1-10/14		Task 1
3. Incorrect Login Research	Find a way to handle (3) incorrect logins then lock account.	2		10/1-10/14		Task 1
4. Implement Incorrect Log	Implement incorrect login method.	1		10/1-10/14		Task 3
5. Create STV Database	Create database with streaming services with factors such as price, shows, etc and vice-versa.	3		10/1-10/14		Needed for multiple features.
6. Research Search and Results	Find an efficient way to implement search feature.	3		10/1-10/14		Task 5
7. Implement Search	Implement search method and results.	1		10/1-10/14		Task 6
8. Research Surveys	Find way to categorize variables of survey.	3		10/14-10/28		Needed for overall Survey feature.
9. Survey Algorithm	Create method to properly	2		10/14-10/28		Task 8

	survey users' interests.					
10. Implement survey	Implement survey method.	1		10/14-10/28		Task 8
11. Allow Admin Edits	Allow Admin to Add, Delete, and Delete streaming service variables and services themselves (edit database).	3		10/14-10/28		Needed for overall Admin Maintenance feature.
12. Report Problems	Allow users to input feedback/problems and write onto report.	2		10/28-11/11		~Admin Add, Security Dependency
13. Print Report	Allow Admin to view report of feedback/problems/statistics.	1		10/28-11/11		Needed for Admin Maintenance feature.
14. Favorites - Search	Allow users to add STV services or their variables from search/info page.	2		10/28-11/11		Needed for Favorites feature.
15. Favorites - Page	Allow users to view and maintain their Favorites.	1		10/28-11/11		~same use case as task 11 + search buttons
16. Clean Up UI	Check for grammar, consistency, and overall usability.	5		11/11-12/02		Assure usability.

3.2 Release Plan

- 09/30/2019 - Iteration #1 Complete
  - Research
- 10/14/2019 - Technical Prototype

- 10/14/2019 - Iteration #2 Complete
    - Use Case 1 and 6 - Login and Search
- 10/28/2019 - Iteration #3 Complete
    - Use Case 3 and 4 - Survey and Admin Add/Edit/Delete
- 11/01/2019 - Architecture Document Due
- 11/11/2019 - Iteration #4 Complete
    - Use Case 2, 5, and 7 - Report Problem, Print Report, and Favorites
- 11/22/2019 - Test Plan Due
- 12/01/2019 - User and System Guide Due
- 12/02/2019 - Iteration #5
    - Complete UI, Final Documentations, Fix Loose-Ends

3.3 Iteration Plans

- Iteration #1 - Research
- Iteration #2 - Login and Search
    - Tasks 1 - 7
- 10/28/2019 - Iteration #3 Complete
    - Use Case 3 and 4 - Survey and Admin Add/Edit/Delete
- 11/11/2019 - Iteration #4 Complete
    - Use Case 2, 5, and 7 - Report Problem, Print Report, and Favorites
- 12/02/2019 - Iteration #5
    - Complete UI, Final Documentations, Fix Loose-Ends

3.4 Budget

5 software engineers at 3-5 hours per week for 10 weeks	50 hours * $40/hour = $2000
Required documentation by technical writer, 8 documents	50 hours * $35/hour = $1750

4 Control Plan

4.1 Monitoring and Control

Event	Info	Audience/Deliverables	Frequency
Team Meeting	Updates and clarifications are provided.	All team members must attend. Trello comments will be updated to show more specific progress.	Twice Per Week

Project Prototype	Team presents prototypes and gains more understanding.	All team members must be present. A prototype must be shown to our class.	10/14/2019
Project Status	Team presents status of project.	All team members must present. This is an opportunity to track overall development.	10/28/2019
Submission Validation	Consent from all team members given to turn in submissions (iterations, docs, etc.)	All team members must agree to turn in submission or offer reachable changes.	Closes 1 hour before submission deadlines

4.2 Project Measurements

Phase	Measurement	Source
Iteration Planning	Record effort estimates for scheduled tasks. Update effort estimates for product features. Update estimated dates in release plan.	Team
Iteration Closeout	Record actual effort for scheduled tasks. Record actual effort for product features. Submitting iteration beginnings, closeouts, and documents.	Chale or Francisca
System Test	Record the rate at which errors are found.	Francisca
Maintain Database	Records variables in database.	Mazin
Ongoing	Record defects found from integration testing. Assign each defect to one of the following categories: blocker, critical, major, minor or trivial. Keep track of the state of each defect: open, assigned, fixed, closed.	Team