

Table of Contents

[**Introduction 1**](#_heading=h.gjdgxs)

[**Description Model 1**](#_heading=h.30j0zll)

[**Class Diagram 1**](#_heading=h.1fob9te)

[**Use Case Diagram 1**](#_heading=h.3znysh7)

[**Use Case Scenarios 1**](#_heading=h.2et92p0)

[**System Sequence Charts 1**](#_heading=h.tyjcwt)

# **Introduction**

Describe the purpose of this requirements document and outline what it contains.

*Twitter + will be an improvement upon X(twitter). It will add new features to the app that would hope to fix and improve upon the site. The features that will be added to Twitter + will be download videos, live-stream, profile customization, customizable feeds, content communities, updated business features and a new dating feature. These features will help to fix/improve some of the problems with X but some will also be to brand new features to the app and website. It will be a whole new rebrand of twitter.*

*The document will be all of the diagrams laying out Twitter+ completely. The document will include a description model, which will describe the functions and requirements of Twitter+. A class diagram, which will display Twitter+ along with its attributes and methods. A use case diagram, which shows the uses between Twitter+ and everyone on it, and a use case scenario to describe each use case. Finally a system sequence diagram to pull the class diagram, use case diagram, and use case scenario together.*

# **Description Model**

Using text, describe the requirements for your system. Expand on the function section from your project plan. Include requirements for the following categories: Output, Input, Processes, Performance and Security.

* *Download Videos - For downloading videos, the user will need to be able to connect to Twitter+. Connecting to Twitter+ requires an internet connection and logging into a Twitter+ account. Whichever video they would like to download, they need to bring up the menu, either right click for desktop or press and hold for mobile, and download the video to wherever they decide. Depending on the size of the file will determine how quickly it will transfer.* 
  + *Input - URL, provided from user, pointing to video to be downloaded.*
  + *Output - Function to generate video file, specifies format and resolution from video.*
  + *Processes - Validation of the provided video URL. Download/retrieve the video content from the URL. Saved as local file to location user desires.*
  + *Performance - Download speeds, to be determined from size of file and capability of device. Concurrent, or multiple at the same time, downloads may affect performance as well.*
  + *Security - Virus protection from embedded malware.*
* *Live-stream Videos - Similar to downloading videos, live-streaming, or accessing live-streams, requires a connection to Twitter+ and the internet. The live-stream itself will need a video source, coming from camera, computer recorded, etc. The live-stream will be run through Twitter+’s servers to distribute the live feed to viewers. Since Twitter+ is used to stream through, the live-stream will be cast through to the Twitter+ platform. A chat will be available to send a message that will appear live for others to see and interact with or respond to.* 
  + *Input - A live video source will need to be provided. Chat messages.*
  + *Output - A live video feed is available through Twitter+ for users to watch. Chat messages from any user, under moderated conditions, will be displayed in the live-streams chat.*
  + *Processes - Feed of video source must be encoded to the proper format. The video source is streamed through Twitter+’s servers, and then onto Twitter+. Joined viewers can interact with chat and send messages.*
  + *Performance - Quality dependent on live-streamer settings, and capability of the user. Latency is determined by live-streamer, but can also be affected by user device. Live-streamer internet upload connection can affect streams, along with user experience can be affected by their internet download connection.*
  + *Security - Location anonymity, and moderated streams.*
* *Profile Customization - Users must be logged into their Twitter+ account to customize their profile. Within customization, the user will be able to set privacy preferences for anyone viewing their profile. Intuitive user interface for customization of profile. Data of profile changes must be saved and managed securely.* 
  + *Input - Users must provide account information. Data, such as profile picture, username, bio, etc, to be provided by users. Privacy preferences to be set by users.*
  + *Output - Updated profiles from users with the changes they have made. If blocked out, from privacy settings, nothing will be viewable.*
  + *Processes - Login authentication. Customizable changes to profile. Privacy preferences. Storage of data from profile changes. User interface accessibility for customization changes.*
  + *Performance - Ease of use for quick customization if wanted. Update response time for profile changes hastened, along with data stored through Twitter+ servers.*
  + *Security - Privacy preferences to block unwanted users from accessing the account.*
* *Customizable Feeds - Users have the ability to customize the content appearing in their feed based on preferences. User friendly feed. Filters for content to fill feed with specific types of content. Recommendations based on preferences. Feeds influenced based on who the user is following. Content moderation preferences if wanted.* 
  + *Input - User preferences based on filters, recommendations, moderation, followers, topics.*
  + *Output - Customized feed based on user preferences.*
  + *Processes - User selection for preferences wanted. Content recommendations provided. Moderation settings, if set, implemented. Filters set. Followers feed and interests implemented to feed.*
  + *Performance - Feed customization updated and data changes saved through Twitter+. Efficiently provides nearly endless feed depending on specifics of filtering.*
  + *Security - Moderation settings for what the user may want. Unrestricted, child friendly, etc.*
* *Content Communities - Users must be logged into Twitter+ to be a part of, and interact with content communities. Users can create content communities of their own, or join pre-existing communities. Community moderators to be set along with Twitter+ moderation. Users have the ability to post content within communities, or interact with others who have posted content.* 
  + *Input - User provided information upon creation of or when managing a content community. Name, description, privacy settings. Content created by users and sent to the community.*
  + *Output - Community’s information, name, description, member details. Content created and posted by users.*
  + *Processes - Creation of content community, name, description, privacy settings required. Managing and editing community. Moderator permissions through creator. Creation, or posting, of content within the community. Interaction and engagement from users, likes, comments, etc.*
  + *Performance - Community updates live for everyone with zero latency. Moderators equipped with tools to moderate communities to the liking of the community creator and the moderators.*
  + *Security - Moderators to be determined and set by creators of content communities. Moderators will have the ability to block and ban content or users if they are breaching community rules, even if those rules are not enforced by Twitter+ and only through the community. Twitter+’s moderation will also be prevalent throughout the communities.*

# **Class Diagram**

Create a class diagram. The Class Diagram should contain all of the system objects, their attributes, and any known methods. This diagram may be included as a separate file – it does not need to be inserted into this Word document.

# **Use Case Diagram**

Create a Use Case Diagram for all of the "uses" of your system. This diagram may be included as a separate file – it does not need to be inserted into this Word document.

# **Use Case Scenarios**

Create a full description Use Case Scenario (detailed descriptions) for each use case of the system. This full scenario should include an enumerated list of steps involved in the activity as well as any exception conditions.

# **System Sequence Charts**

For each Use Case Scenario, provide a sequence diagram. Use your class diagram, use case diagram and scenarios to create the corresponding System Sequence Diagram.