

Twitter Clone Overview

Problem Statement

This project will build a communications application similar to the popular application “Twitter.” This will be called “Twitter Clone” or *Twic* for short. Twic will allow interested persons to subscribe to download the application to their PC and/or mobile device and subscribe to the service.

The service allows a user to post short updates and subscribe to updates by specific persons. The list of subscribers will be searchable by name and subject.

Requirements

This document is NOT a full requirements document for the system. That will be developed by the student practicum team for approval by the instructor as part of the project. There will be a high level or general directions stated here, followed by constraints or design directions in the next section. The instructor will work with the team to formulate a full requirements document, and development cannot proceed until requirements are approved.

Feature Ideas

These are the initial ideas on how the system should work. These concepts will be refined into a final feature set as part of the requirements phase.

1. The subscriber must sign-in to the application to access the services. The subscriber’s account will maintain a set of data, such as username, real name, likes/dislikes, following, followers, and twic history.
2. The user can connect to other users by searching for their username or real name
3. The service will show users a list of "trends" using #hashtags)
4. Allow users to share and link directly to a tweet via a url link on each Twic (link displayed based on Twic time)
5. Allow a re-Twic.
6. Application can be downloaded to device or accessed via webpage.
7. Additional “stretch goals” will be determined if time permits. Some possible additions are:
 - a. Additional signup methods (oauth, google auth, facebook auth)
 - b. Blocking users
 - c. Email notifications

Design Considerations

The following considerations must be clearly defined in the requirements document before beginning design and code.

1. Applications will need to be built for PC and mobile device. Because of the short timeframe, the technologies need to be fairly standard, or quick to learn (a few weeks to come up to speed).

- a. Android SDK for Eclipse can be used. At least one student will need to have a real android phone.
 - b. A good GUI design and screen layout plus sequence will be needed as part of the requirements to design transitions, so the operation of the app is clear.
 - c. Javascript or JQuery can be used for the client-side.
2. Server-side development for the service, including a data store. This could be SQL, or something simpler such as JSON or XML. This needs to be built somewhere accessible. The franklinpracticum.com website can be used for hosting the service.
3. Code will be version controlled at franklin.unfuddle.com. Either SVN or GIT can be used.