Sprint 1 Planning Document

Team 16 - xpac

Anunai Ishan, Amol Moses Jha, Adit Kumar, Parth Shelgaonkar

Sprint Overview

For our first sprint, we plan to implement the basic architecture of the server and the client, which includes implementation of the basic protocol brewed in-house, the client and server classes being able to effectively communicate and exchange files over the network through the channels set by us. We also aim to implement the basic functionality of our packaging system which involves developing the metadata classes and ensuring compatibility with our protocols.

In conclusion, we aim to achieve:
1. Protocol for communication between the server and the client
2. Basic functionality of communication between the server and the client
3. Metadata definition of packages

Scrum Master: Adit Kumar

Meeting Plan: Sunday (5:30 pm to 7:30 pm)

Risks and Challenges: A significant number of user stories need to be completed in this sprint and a major portion of the server's codebase needs to be implemented for effective communication with the client, especially the ability to transfer the files requested by the client.

Additionally, the client needs to possess the basic functionality of a command line application. This requires the necessary utilities for a command line application to be fully functional, including parsing the command line and managing root permissions, which are essential to package managers.

We will be trying to implement the basic functionality of being able to transfer a single binary package over the network and install it, hence, both the client and the server need to be in sync when communicating over the network.

Current Sprint Detail

We plan to achieve the following user stories during the course of this sprint:

User Story #1:

As a user, I would like to have the ability to connect to the server which hosts repositories.

Task Details:

#	Task Description	Estimated Time	Team	Owner
1	Set up client class with basic functionality	**2 hrs**	**Client**	**Amol & Parth**
2	Set up server class with basic functionality	**4 hrs**	**Server**	**Anunai & Adit**
3	Set up server socket on the server side to accept client connections	**3 hrs**	**Server**	**Anunai & Adit**
4	Set sockets on the client side to set connection to a given server on a given IP address and port	**4 hrs**	**Client**	**Amol & Parth**
5	Define the protocol for the server and the client to interface over the network	**3 hrs**	**Dev**	**Amol & Parth**

Acceptance Criteria:

➤ Given that the client-end is able to ping the server and the server is able to parse the request sent by the client.

➤ Given that the server is able to accept a new socket connection with the client and able to parse the requested command based on the protocol

User Story #2:
As a user, I would like the server to establish multiple connections with clients, concurrently.

Task Details:

#	Task Description	Estimated Time	Team	Owner
1	Set up communication protocol to handle multiple clients	3 hrs	Dev	Amol & Parth
2	Set up a multi forked server that is able to manage connections with multiple clients	4 hrs	Server	Anunai & Adit
3	Set up the server to spawn new child processes to do requested file fetching tasks for each new socket connection	5 hrs	Server	Anunai & Adit

Acceptance Criteria:
➤ Given that the server, after parsing the request from the client, is able to spawn a new child process by forking from the command parsing module
➤ Given that the server is able to handle the rest of the file fetching tasks in the new process and still is able to accept new connections from multiple clients
➤ Given that the cloned process is able to terminate after execution of its task.

User Story #3:
As a user, I wish to be able to send and receive files to and from the repository.

Task Details:

#	Task Description	Estimated Time	Team	Owner
1	Set up client to connect to the server, interfacing with the defined protocol	4 hrs	Client	Amol & Parth
2	Set up server to connect to multiple clients, interfacing with the defined protocol	5 hrs	Server	Anunai & Adit
3	Set up the client to receive multiple files over the given server socket	3 hrs	Server	Anunai & Adit
4	Set up the client to install the given binaries in the given user's system	5 hrs	Client	Amol & Parth

Acceptance Criteria
➤ Given that the server and the client can communicate effectively and efficiently without any errors, a given client should be able to connect to a given server and request a given list of files, and the server should be able to parse the request and act on it.
➤ Given that the server is able to efficient access the requested file from its Software Repository and flush it to the client.
➤ Given that the client is able to receive the file(s) sent by the server.

User Story #4:
As a user, I want xpac to be able to install packages on my system.

Task Details:

#	Task Description	Estimated Time	Team	Owner
1	Check whether the given package is already installed or not on the user's system	3 hrs	Client	Amol & Parth
2	Set up client to check the authenticity and integrity of the package received by the client	3 hrs	Server	Anunai & Adit
3	Set up client to manage root access by the user in order to install packages	4 hrs	Server	Anunai & Adit
4	Set up client to install packages sent by server to the user's systems	3 hrs	Client	Amol & Parth
5	Set up the client to add the package to a list of already installed packages	2 hrs	Client	Amol & Parth

Acceptance Criteria:
➤ Given that the client is able to maintain a set to store the list of already installed packages and is able to poll this data structure before requesting the server for a given package
➤ Given that the client is able to request for the package to be installed to the server and receive the binary file
Given that the client is able to manage root access given by the user and place the received binary file into the specified path environment variable.
➤ Given a request of a list of package(s) by a given client connection, the server should be able to parse the request and send (flush) the requested files over the given server socket.
➤ Given the client receives a given list of binaries, the client should be able to place them in correct directories in the system to effectively install them on the users system
➤ The client should also look for and remove any inconsistencies resulting from any of the above.

User Story #5:
As a user I would like xpac to efficiently manage my packages.

Task Details:

#	Task Description	Estimated Time	Team	Owner
1	Set up graph database of packages by setting up metadata class	3 hrs	Dev	Amol & Parth
2	Set up metadata class by defining the required metadata for each and every package	2 hrs	Dev	Amol & Parth
3	Set up functions to compress and decompress metadata by working on common compression algorithms	3 hrs	Dev	Amol & Parth
4	Set up functions to save and read metadata to disk in order for future access to the graph	4 hrs	Dev	Amol & Parth
5	Set up server repository schema to optimize package database schema	3 hrs	Dev	Amol & Parth

➤ Given that the client and the server communicate effectively, and can send and receive files over the given protocol, the client and server should have a good idea of the internal graph structure abstraction to represent the packages
➤ Given that the both the client and the server can use the metadata class to work with the packages, the server and the client should have the ability to effectively read and write the associated metadata to disk and also compress/decompress the metadata before reading and writing the same to disk in order to minimize the time wasted due to disk accesses.

User Story #6:
As a user, I wish to be able to list all the packages that have been installed by xpac.

Task Details:

#	Task Description	Estimated Time	Team	Owner
1	Set up client to store and list all packages available yet	**2 hrs**	**Dev**	**Amol & Parth**

Acceptance Criteria:
➤ Given that the client has a good idea of the package graph data structure, it should be able to maintain a list of all packages currently installed on the user's system.

User Story #7:
As a user, I would like to have access to proper software documentation, like man pages for xpac.

Task Details:

#	Task Description	Estimated Time	Team	Owner
1	Set up proper man pages and documentation for xpac	7 hrs	Dev	Anunai & Adit

Acceptance Criteria:
➤ Given that the client and the server are able to execute all of the above user stories, proper system-wide documentation, like man pages should be available to the users. This would greatly help the usability of xpac from the get go and can be expanded on as we add more features to xpac.

User Story #8:
As a user, I would like xpac to have an online presence to be able to to connect with the community of contributors and other users.

Task Details:

#	Task Description	Estimated Time	Team	Owner
1	Set up a landing page for the project.	**8 hrs**	**Website**	**Anunai & Adit**
2	Make the landing page clear and concise, so that it includes instructions on how to install and use xpac.	**6 hrs**	**Website**	**Anunai & Adit**

Acceptance Criteria:
➤ The website should be responsive and sleek. The design should be minimalist with a dash of personality. Users should be able to easily read and understand what xpac is, how to install it and how to use it for their daily use.

User Story #9:

As a user, I would like xpac to have a way to connect to the developers in order to submit valuable feedback and necessary critiques and comments about xpac.

Task Details:

#	Task Description	Estimated Time	Team	Owner
1	Set up an IRC server for the users to be able to connect to the developer community	**12 hrs**	**Website**	**Anunai & Adit**
2	Set up a feedback form where users can submit valuable feedback and bug reports.	**4 hrs**	**Website**	**Anunai & Adit**

Acceptance Criteria:

➤Given that the above user stories are completed, an IRC Server should be set up for xpac where users can connect with the developer and contributor's community.

➤Given that the above user stories are completed, a mechanism to allow users to submit feedback and bug reports.

User Story #10:
As a user, I would like xpac to be a stable and robust piece of software.

Task Details:

#	Task Description	Estimated Time	Team	Owner
1	Set up unit testing for server	8 hrs	Testing	Anunai & Adit
2	Set up unit testing for client	8 hrs	Testing	Anunai & Adit

Acceptance Criteria:
➤Given that the above user stories are completed, rigorous testing and quality control should be done on each component of xapc and it's functionality to ensure rock solid stability.

Remaining Backlog (10/25 Stories Complete)

1. ~~As a user, I would like to have the ability to connect to the server which hosts repositories.~~
2. ~~As a user, I would like the server to establish multiple connections with clients, concurrently.~~
3. ~~As a user, I wish to be able to send and receive files to and from the repository.~~
4. ~~As a user, I would like xpac to able to efficiently manage packages.~~
5. As a user, I would like to have the ability to know the packages hosted by the server.
6. As a user, I would like to have the ability to know the versions available of a particular package.
7. ~~As a user, I would like to have the ability to install packages.~~
8. ~~As a user, I would like to have the ability to list the packages already installed using xpac.~~
9. As a user, I would like to have the ability to update the software database.
10. As a user, I would like to view if there are any updates available for the packages installed.
11. As a user, I would like to have the ability to update packages.
12. As a user, I would like to have the ability to do a full system-wide update.
13. As a user, I would like to have the ability to delete packages.
14. As a user, I would like to have the ability to delete orphans and unneeded packages.
15. As a user, I would like to have the ability to compile from source from packages if needed.
16. As a user, I would like to have the ability to choose between a stable version, or a current bleeding-edge version package for installation.
17. As a user, I would like to be able to automate package management using xpac.
18. As a user, I would like to be able to tweak and customize xpac for a more personalized experience.
19. As a user, I would like to have the ability to use xpac to manage packages on different Unix based operating systems.
20. ~~As a user, I would like to have access to proper software documentation, like man pages for xpac.~~
21. ~~As a user, I would like xpac to have an online presence to be able to to connect with the community of contributors and other users.~~
22. ~~As a user, I would like xpac to have a way to connect to the developers in order to submit valuable feedback and necessary critiques and comments about xpac.~~
23. As a user, I would like xpac's website to responsive and elegant.
24. ~~As a user, I would like xpac to be a stable and robust piece of software.~~
25. As a user, I would like to have xpac to be easy to use.