Software Requirements Specification for S3PP version 1.0

Prepared by:

Ahmed Ehab Hamouda

Aly Moussa

Marwan Eid

Mohammed Abuelwafa

Group 7

Submitted on April 14, 2020

Table of Contents

Revision History

Name	Date	Reason for Changes	Version
S3PP SRS	April 14, 2020	Initial Draft	1.0

1. Introduction

1.1 Purpose

safe social systems plus plus (S3PP) is a software system that provides the social network for the next generation. In addition to the basic social media interactions represented in chatting, maintaining blogs, maintaining photo albums, tracking physical locations, lookups and recommendations, surveys and votings, and video and audio streaming, it aims to provide extra functionalities in order to promote their clients' privacy and security, since these are some of the most pressing problems in today's world. Hence, two main systems are to be implemented. One is the escalated identity revelation system (EIRS) and the other is the trusted shared storage system (TSS). S3PP is mainly a mobile application system that has a web-based application as well. However, the current initial focus is on developing the mobile application version. In addition to those main features of the system, other features are to be discussed in more detail throughout the document.
This SRS, software requirements specification, document is to demonstrate all possible system features in a detailed manner including various aspects such as functional and non-functional requirements as well as system features and interface requirements.

1.2 Document Conventions

- ❏ We use bold formatting for Headings.
  - ❏ Font size = 18 -> for Heading 1: used for the headings of the five main pillars of the SRS Documents (Introduction, Overall Description, External interface Requirements, System Features and Other Non-Functional Requirements)
  - ❏ Font size = 14 -> for Heading 2: used for the headings of each subcomponent within each pillar in the SRS Document
  - ❏ Font size = 12 -> for Heading 3: used for further formatting within the subsections of the subcomponents.
- ❏ The font used in the SRS documents is Times New Roman, 12 pt.
- ❏ Any important subject/term is written in **Bold** with font size 12 pt.
- ❏ The page number is in the top right of the page.
- ❏ A bolded asterisk followed by a bold sentence is used to represent a note to be considered.
- ❏ References are in APA format. Written in Times New Roman style with 12 pt font.
- ❏ System Features and Use Cases are represented within a tabulated form

- ❑ All Diagrams and Graphs are attached in the appendices to give the reader a better understanding of the document.

1.3 Intended Audience and Reading Suggestions

The Document is intended to various stakeholders that will be mainly categorized to two groups:

1. Management side of the project: This includes S3PP owners and managers as well as project managers. This group will be mainly concerned with sections 1, 2, 4, and 5 of the document. They are required to verify the product's purpose as well as its functional and non-functional requirements.
2. Technical side of the project: This includes the developers and testers that would be mainly concerned with sections 2, 3, 4, and 5 of the SRS document, since they will be required to design the system in such a way that meets the requirements, constraints, assumptions, and dependencies described in section 2, the functional requirements described in section 4, and the non-functional requirements described in section 5 using the proposed software, hardware, and communication interfaces introduced in section 3.

1.4 Product Scope

S3PP is an application on which all typical features of a social media application are provided, but in a more secure and private manner. Privacy is one crucial and significant aspect of this app as a brand-new feature of being able to have multiple identities which could possibly be either transient -fictitious identities which only last for a single session only, fictitious identities which will last until the user deletes them, partial identities which represents only parts of the real user identity or finally user's real identity which is the full real registered identity of the user. This application is risk free as it does not only allow having multiple identities yet it helps users keep track of their own identities by providing an identity tracking facility (ITF).

Since one of the top priorities of S3PP is privacy, the ITF feature is offered to help the users organize their many identities and keep track of all contacts who know them as a different identity, so that no one from the same list of friends of one of the identities will get to bump into another different identity of the same user in case the user has more than one identity. Not only privacy is a fundamental feature in S3PP, security is another predominant trademark in the application, so in order to prevent abusing the system, S3PP keeps track of all interactions and links them to the real users. In cases of abuse, S3PP would be able to reveal the real identity to the
authorities.

Another main goal of S3PP is to make users massively satisfied and this would not be possible if only typical and basic features of most of the social media platforms were provided even if it was with more privacy. Hence, in order to provide the users with an enhanced user experience, S3PP provides them with integrating their identities with bots to provide different functionalities impersonating them such as *silentlySummarizeandRecordBot* which can be used to join forums

and provides its users with a summary of conversations and alert the user to important announcements; the *Announcement Bot* which can be used to store and announce events to certain forums and the *weather bot* that can be used to announce weather forecasts. Moreover, third parties can provide their own bots and the users have the freedom to integrate with them.

Finally, S3PP app provides a trusted shared storage, TSS, system which enables users to store and share files in a secure manner. Files created under a certain identity (called file creation identity, or FCD) can only be read, written, or shared when the user assumes the same FCD. Moreover, the content creators have the freedom to charge the users for viewing their content and thus make profits through sharing such items.

1.5 References

"SRS Template ." IEEE , Arthur , web.cs.dal.ca/~hawkey/3130/srs_template-ieee.doc.

2. Overall Description

2.1 Product Perspective

S3PP is a new, self-contained social networking product that provides the usual set of products and services of social networking while keeping its clients' privacy and security very high priorities. S3PP provides its clients with the standard social networking features such as adding friends, chatting, maintaining personal blogs, and sharing free and paid content. What is special about S3PP is allowing the users to create multiple identities and offering the users the choice and level of their privacy. S3PP also allows users to share and store files privately such that accessing a file is only possible for the same user identity. Thus, S3PP offers its clients a powerful social networking platform and prioritizes their privacy and security.

2.2 Product Functions

S3PP should allow the users do the following:

- Create and validate a user account.
- Choose the account to be free or premium.
- Create a new identity from the four types of identities.
- Delete an identity.
- Add to or remove users from the ring of a given identity.
- Get warned if he/she bumps into a user that might know him/her under another alias.
- Chat with another user or with a group of users, with text, audio, or video.
- Add to or remove posts from personal blogs.
- Add to or remove photos or videos from photo albums.

- Allow one or more users to track his/her physical location.
- Ask for recommendations for real world services.
- Post surveys and votings.
- Video and audio stream.
- Interact with other users' different types of posts.
- Create new files.
- Read, edit, and share existing files, assuming the same FCD.
- Charge for the content they publish and restrict access to certain users.
- Purchase a ticket to access a specific item.
- Select a bot for a certain identity either from existing bots or from third-party providers.

2.3 User Classes and Characteristics

S3PP shall serve the following categories of users:

- General users:
    - Free users:
      This category of users shall have access to all of the system features except the additional support services and the advertisement-free experience. It is expected that the majority of users fall under this category.
    - Premium users:
      This category of users shall have access to all of the system features including the additional support services and the advertisement-free experience.
- Advertisers:
  This is a subcategory of general users. Advertisers are to pay S3PP in return for showing their advertisements.
- Celebrities:
  This is a subcategory of premium users. In addition to all the features celebrities will enjoy, S3PP will pay celebrities for their participation.
- Administrators:
  Administrators shall have access to all the features of S3PP; yet, they have more privileges than general users. Mainly, they should be able to respond to and take action against reports; this includes taking files offline, deleting content, warning users, and reporting to authorities when necessary.

2.4 Operating Environment

S3PP shall be working on various devices, typically PCs, smartphones, and tablets. For the web version, S3PP shall operate on most browsers that support HTML5 and CSS3 or later versions. Android operating system versions that shall support S3PP are Version 4 (Ice Cream Sandwich) and above. Apple operating system versions that shall support S3PP are iOS 8 and above.

2.5 Design and Implementation Constraints

The developers might be limited with the following constraints:

- A user's private information may have to be compromised had the user committed an illegal act that calls for reporting the user's data to authorities.
- S3PP shall warn the user if he/she bumps into another user that knows them under a different identity. However, it is up solely to the user to choose or not to do so, and developers have no privilege of preventing them in such a situation.

2.6 User Documentation

User documentation shall include the following components:

- A user manual that includes instructions for the main features of the application. The user manual shall appear the first time a user uses the application; the user can also access it later.
- Tutorials for many features of the application.
- On-line help that includes a FAQ page and options to contact the application help center.

2.7 Assumptions and Dependencies

The following factors are to be assumed:

- Third-party providers are assumed to provide free and premium bots to the application users.
- Local payment service providers like Fawry and other mobile operators are to be relied on to allow users to purchase tickets and to keep track of e-payment services.
- Users' and celebrities' identification are to be verified by authorities at first-time sign-up.
- A list of the common domains of the disposable email addresses are to be used to verify that the emails used at registration are indeed functional and not disposable emails.
- Legality of usage of cryptocurrency in each country is to be checked first before allowing its usage, since it is banned and restricted in some countries, so that should be accounted for.

3. External Interface Requirements

3.1 User Interfaces

3.1.1 Home Interface

Figure 1

3.1.2 Login/Sign Up Interface

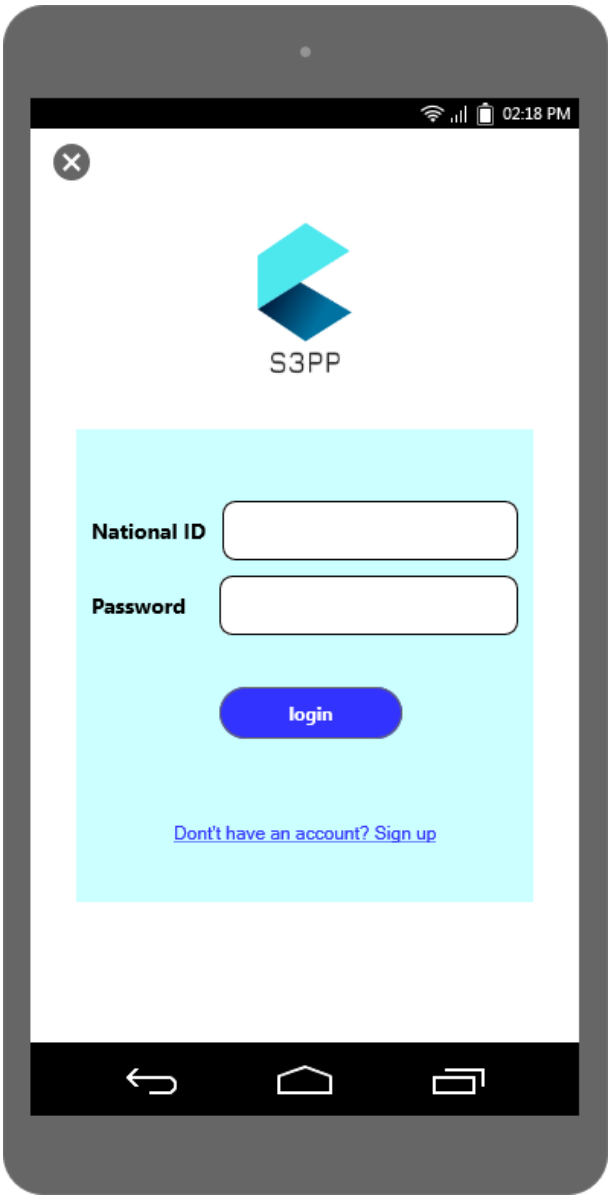

Figure 2

3.1.3 ITF Interfaces

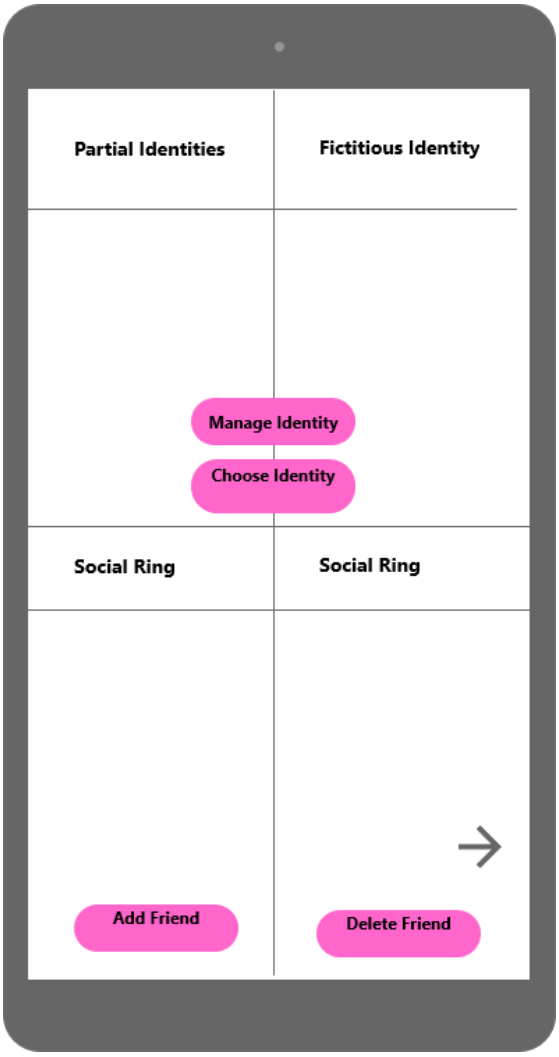

Figure 3

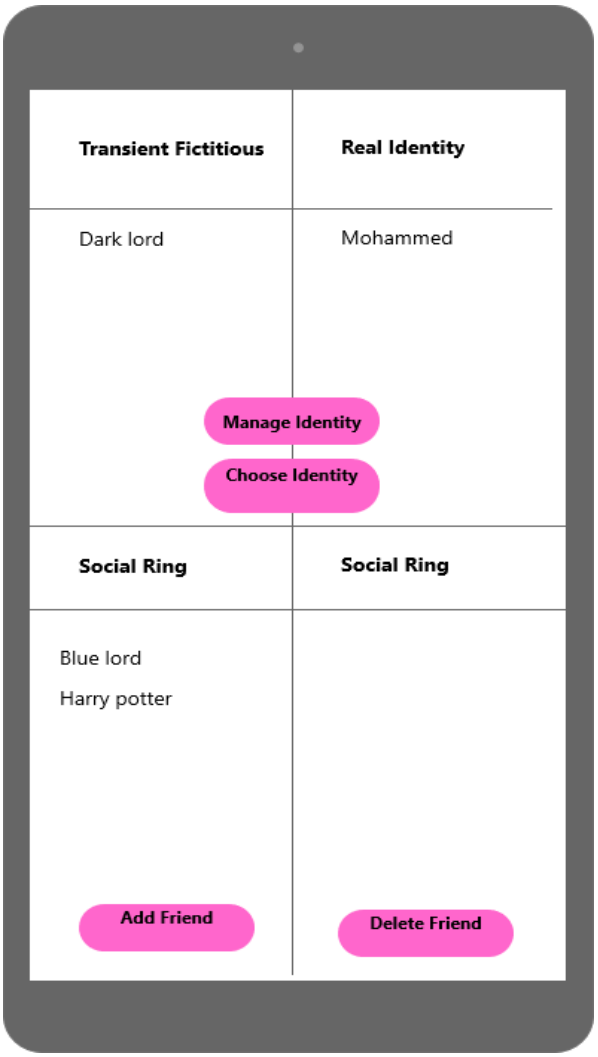

Figure 4

3.1.4 Newsfeed Page

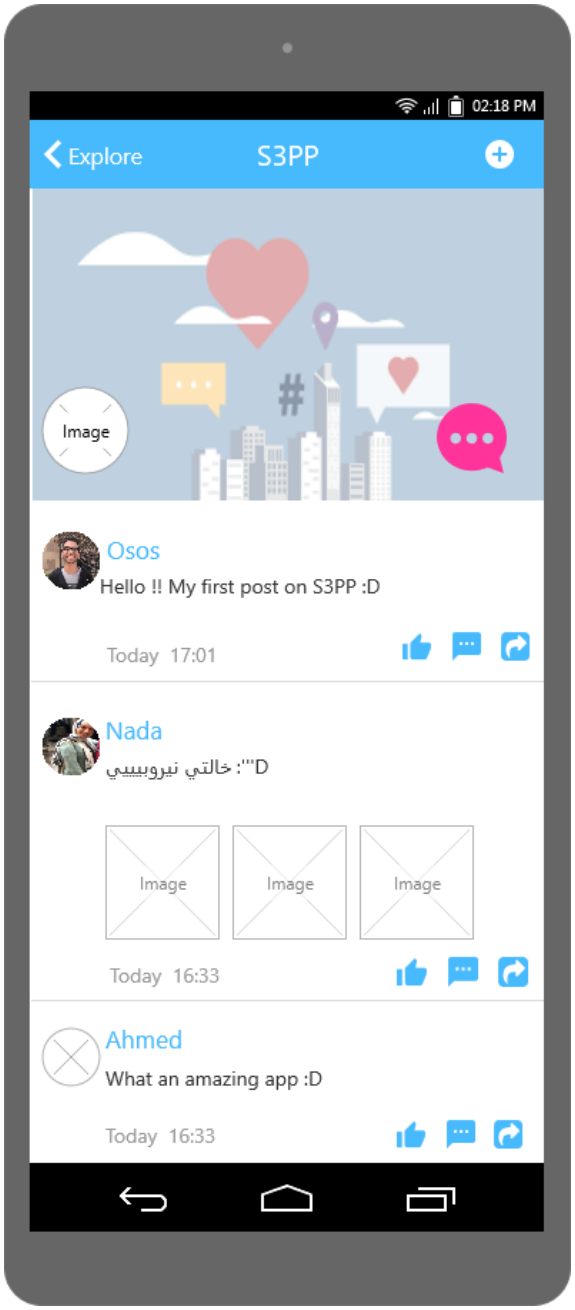

Figure 5

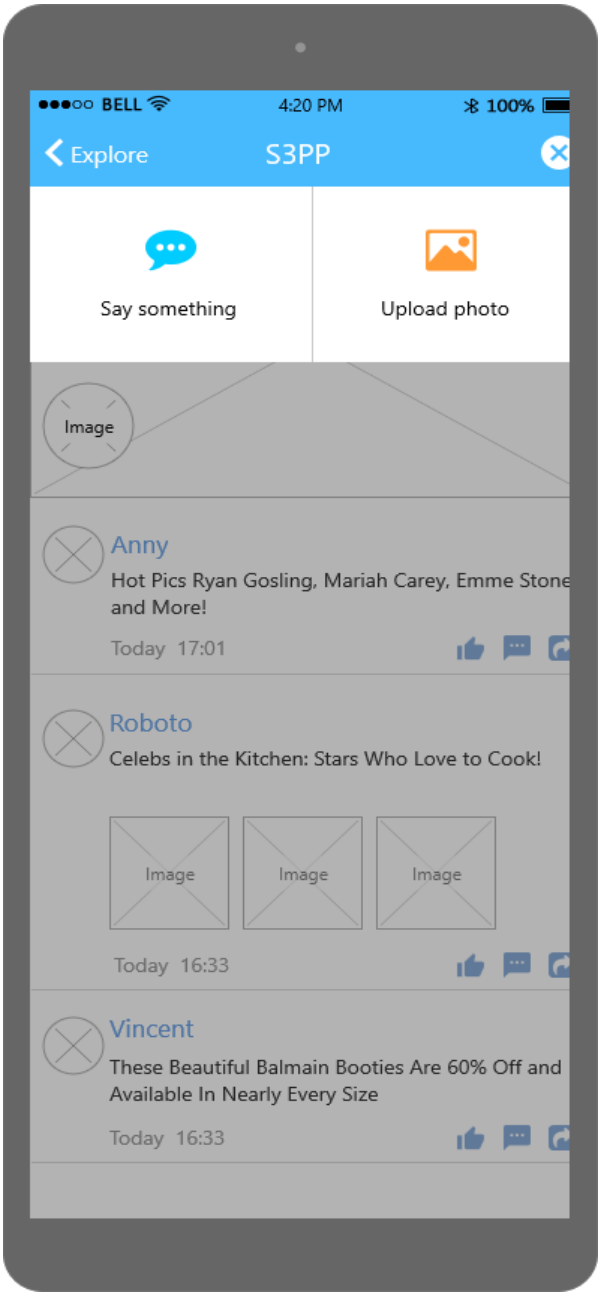

Figure 6

3.1.5 Chat Interface

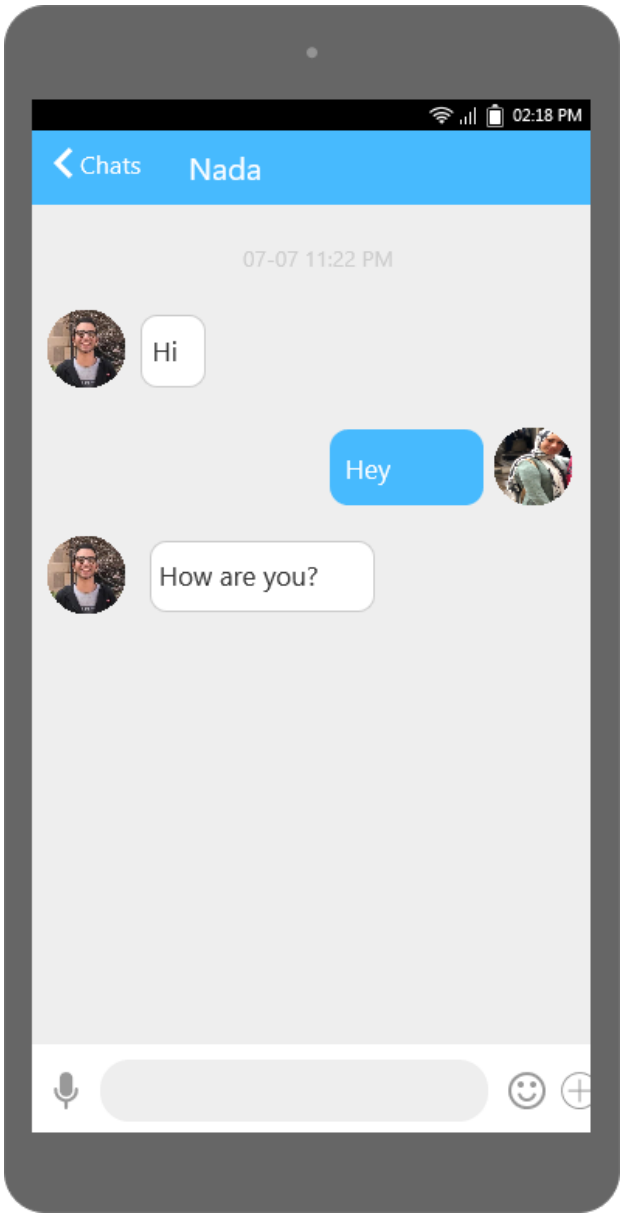

Figure 7

3.1.6 Manage Content Interface

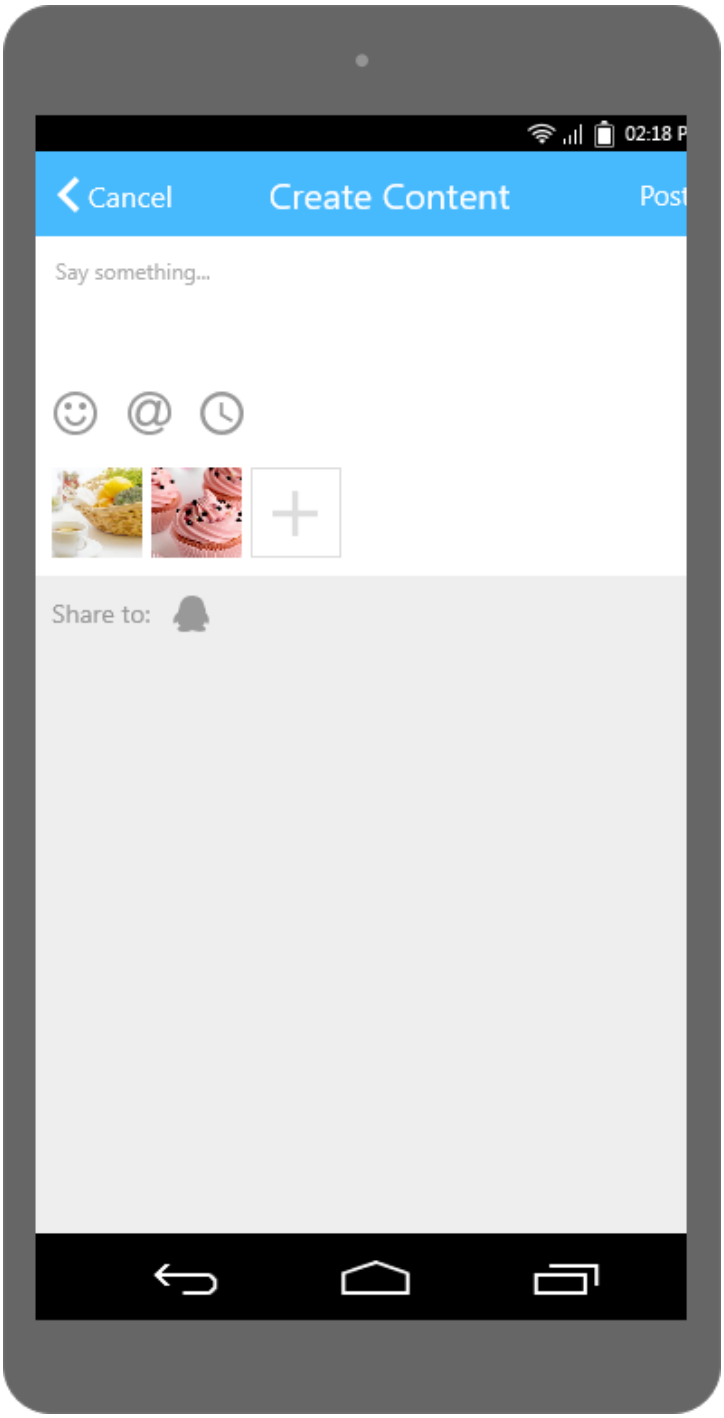

Figure 8

3.1.7 Bots Management Interface

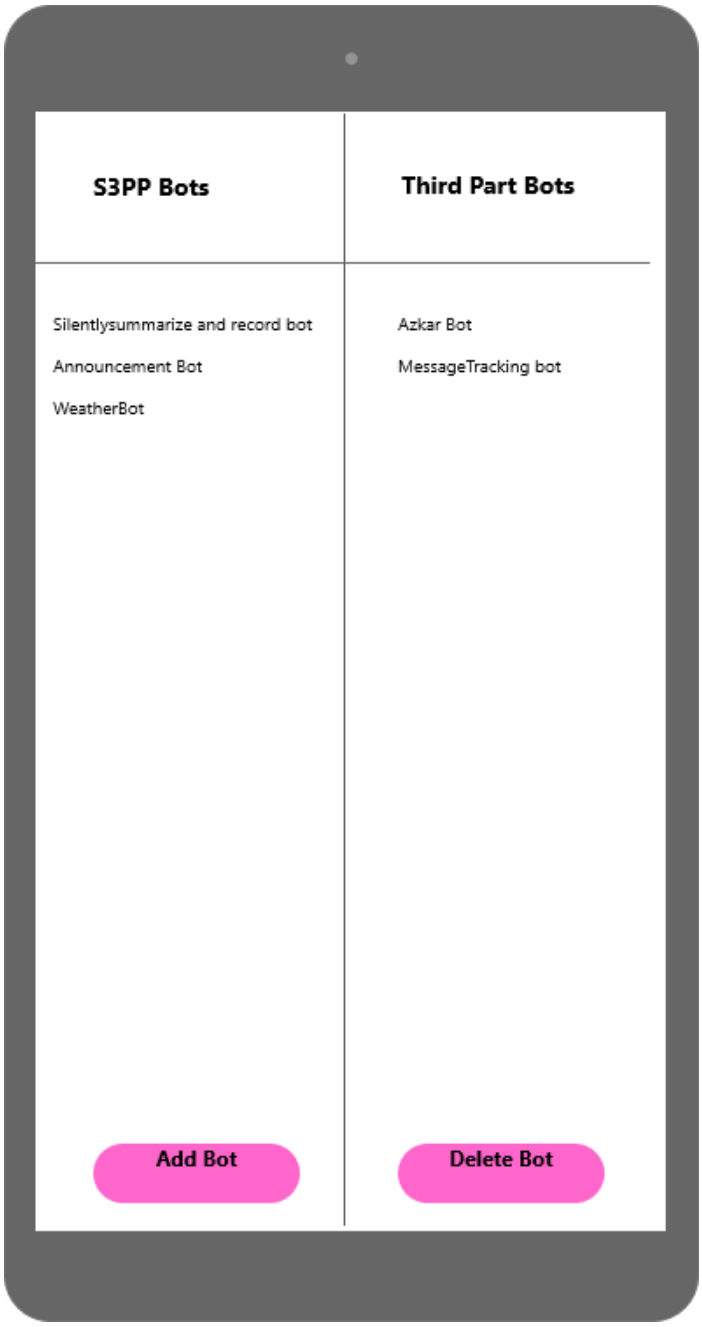

Figure 10

3.2 Hardware Interfaces

The S3PP system has two aspects; the first one is a mobile based application that needs a device to operate on such as an android device (mobile or tablet) or an IOS device (iPad, iPhone, iPod). The second aspect is the web-based application that would be the focus after implementing the mobile application completely and that would only need a device with a browser. However, we may need a server to store the messages and required data of the chatting system, another server to host S3PP data and a server to host the payment services. However, we are going to depend on available clouds to host our servers such as Microsoft azure as an example and for the payments, S3PP is going to depend on external payment providers such as Fawry and other mobile operators such as Orange, Etisalat and Vodafone. hence, S3PP does not need to manage those hardware Interfaces since they are going to be managed externally and S3PP would be charged for it.

3.3 Software Interfaces

3.3.1 APIs and Local Payment Interfaces

S3PP platform will count on Fawry and mobile operators' APIs and PayPal APIs for credit card payment. The users will be asked through the UI to enter their payment information, then the system will send this information to this API for allowing the transaction to take place. By this means, the money will be taken from the user balance and deposited into S3PP account. After that, this API will send a successful message for the system to indicate the success of the transition process or failed message otherwise. Moreover, the utilization of cryptocurrencies such as Bitcoins would be available as an option, but it depends on the legal requirements of each country and region.

3.3.2 Cloud Interface

S3PP platform will utilize externally operated clouds such as Microsoft Azure for storing and sharing files among the users. Microsoft Azure will act as a media to share those files after entering the confirmation code provided by the owner of this file. The idea is that the files will be encrypted and only the creator identity has the right to give access to the people they want either for free or by charging them. hence, only people who have access to view the content will be able to view it. Moreover, Azure would be responsible for storing all the other S3PP data as well.

3.3.3 Database Interface

The S3PP system will depend on an SQL server to store users' information/ bots/ photos. The data of the users will be inserted into the database whenever a user tries to sign up for the platform, add blog, get identities, or upload files. The admin of this platform will easily track the users' information by the help of phpMyAdmin/XAMPP software as they provide a visualized picture of the database. Figure 10 presents how the data would be shown on the database.

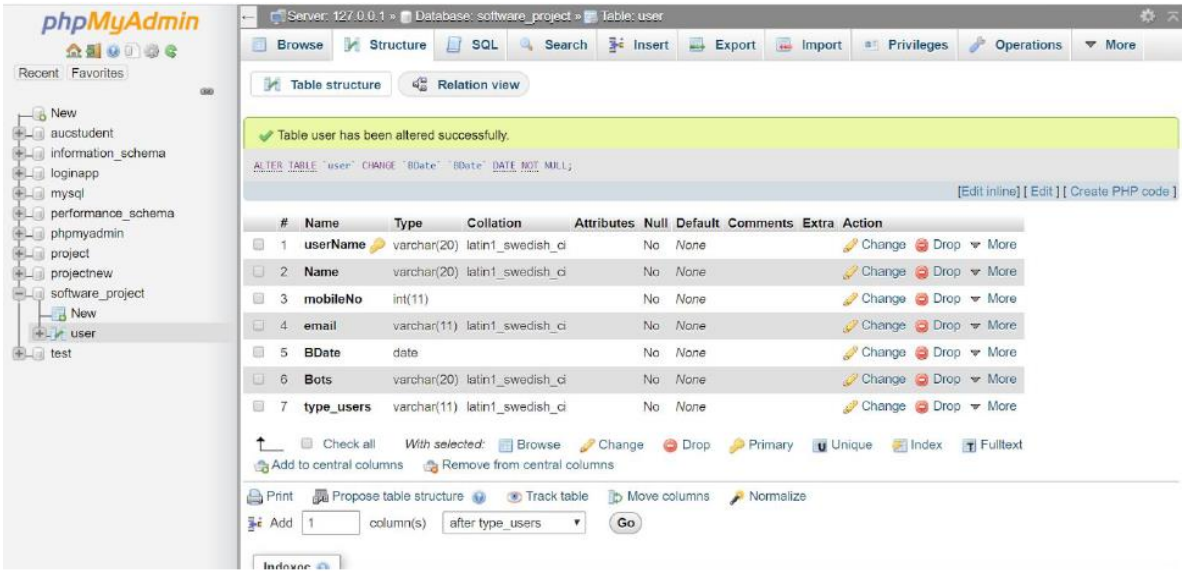

Figure 10

3.4 Communications Interfaces

- S3PP is supported by HTTPS protocol for the web application, so it should be used to communicate with the server.
- Gmail will be used to validate the user's identities by sending a confirmation mail whenever a new user wants to join the platform, to help the users whenever they want to subscribe in the paid service, and to allow the admin to send warning mails in case of violating any rules.
- SMS will be used for payment notifications. All notifications will pop as a part of the application itself like most of the applications.
- Notifications shall be sent to users in the app as well as via email.
- Google maps APIs and GPS will be used for detecting the user location.
- All user information shall be encrypted in the system using end-to-end encryption.

4.1 System Feature

4.1 Identity Tracking Facility

4.1.1 Description and Priority

The identity tracking facility is designed to help the user track his all identities, manage them and their corresponding social rings in addition to managing the bots attached to each identity. hence, this system feature is of Highest priority.

4.1.2 Stimulus / Response Sequences

This system Feature is triggered when the user clicks on the button that opens the ITF dialog. The response would be opening the interface to let the user choose, manage an identity and its corresponding social ring.

4.1.3 Functional Requirements

Use Case ID	#1
Use case Name	Create Identity
Description	This function is responsible for creating a new user identity
Actor	S3PP user
Preconditions	This requires the user to be logged in and have a valid real identity already setted up.
Postconditions	a new identity is created with its social ring.
Normal Flow	1. A logged in user with a valid real identity accesses the ITF system 2. He chooses to create an identity of one of the following: a- Partial identity b- fictitious identity c- Transient-fictitious identity

Use case Functional Requirements:

- CreatePartial()
- CreateFictitious()
- CreateTransientFictitious()

Use Case ID	#2
Use case Name	Setup Identity
Description	This function is responsible for configuring a new user identity
Actor	No actor.
Preconditions	Having an identity created.
Postconditions	The alias and age for this identity are set up.
Normal Flow	The user sets up his identity by setting the name (alias) and age after creating it.

Use case Functional Requirements:

- ChooseName()
- ChooseAge()

Use Case ID	#3
Use case Name	Attach to real Identity
Description	The function is responsible for tracking all the created identity to the user's real identity.
Actor	No Actors.
Preconditions	Having an identity created.
Postconditions	The created identity is linked to the user's real identity.
Normal Flow	The user creates an identity and chooses the type of it and then the attach to real identity function would automatically link it to the real identity.

Use Case ID	#4
Use case Name	Manage Identity
Description	The function is responsible for managing a user identity.
Actor	S3PP user
Preconditions	Having a pre-created identity to manage.
Postconditions	The identity is either edited in terms of name or age and it may be deleted as well based on the user's decision.
Normal Flow	The user opens the ITF facility and chooses an identity to manage, then he has the option of modifying the name, age or deleting the identity.

Use case Functional Requirements:

- ChangeName()
- ChangeAge()
- DeleteIdentity()

Use Case ID	#5
Use case Name	Choose Identity
Description	The function is responsible for Making the user able to choose a specific identity and managing the corresponding social ring and bots.
Actor	S3PP user
Preconditions	Having a pre-created identity to choose from.
Postconditions	The user impersonates the chosen identity and interacts using it. Moreover, he can manage the bots related to this identity and the social ring.
Normal Flow	The user opens the ITF facility and chooses the identity to interact with from a dialog, then he has the option of managing the bots by adding or deleting a bot or managing the social ring by adding or removing a friend.

Use Case ID	#6
Use case Name	Manage ring

Description	The function is responsible for Making the user able to manage the social ring related to a specific identity
Actor	S3PP user
Preconditions	choosing a specific identity to manage its social ring.
Postconditions	A friend is either added or removed from the social ring depending on the user's interaction.
Normal Flow	The user opens the ITF facility and chooses the identity to modify its social ring, then he chooses either to add a friend by his name or deleting a friend.

Use case Functional Requirements:

- AddFriend()
- delete Friend()
- Check If Existed()
- Create Warning()

Use Case ID	#7
Use case Name	Manage Bots
Description	The function is responsible for Making the user able to manage the Bots related to a specific identity
Actor	S3PP user
Preconditions	choosing a specific identity to manage its bots.
Postconditions	A Bot is either added or removed from the identity's bot collection depending on the user's interaction.
Normal Flow	The user opens the ITF facility and chooses the identity to modify its bots, then he chooses either to add a bot by its name or deleting it.

Use case Functional Requirements:

- AddBot()
- deleteBot()

4.2 Payment System

4.2.1 Description and priority

The payment system is designed to handle all payments made by any user for advertisement purposes, for participation or from buying content from other users. It is essential that the system first verifies the payment method the user has provided, and that this information is not revealed to anybody and finally gives the user a verifiable receipt to help ensure that it is a secure payment.

4.2.2 Functional Requirements

Use Case ID	#8
Use Case Name	Make Order Request.
Description	Responsible for initiating the payment is triggered when a user wants the right for content from another user.
Actor	S3pp User
Preconditions	Having a real identity.
Postconditions	An order starts to take place where the user gets the prices as well as subscription types and a verification process is initiated.
Normal Flow	A logged in user tries to access content that needs payment.

Use case functional requirements:

- Verify payment method()
- Check user agreement()

Use Case ID	#9
Use Case Name	Verify payment method.
Description	Process for checking funds.
Actor	Service providers, user wallet or cryptocurrency.
Preconditions	A payment has been requested.
Postconditions	Success or failure of the payment determines if the user will have access

	to the service requested.
Normal Flow	A payment is in process and is waiting for verification.

Use Case ID	#10
Use Case Name	Type of subscription
Description	If the owner of that content has a subscription type the user has the choice between different variants.
Actor	S3pp User.
Preconditions	Purchase is needed
Postconditions	A user can choose between different types of subscriptions.
Normal Flow	A list of subscriptions is seen by the user and is prompted to choose one.

Use case functional requirements:

- Choose Subscription()
- Ability to reshare()
- Length of subscription()

Use Case ID	#11
Use Case Name	Payment history
Description	A log for payments made by each user
Actor	S3pp User
Preconditions	A user needs to have made a payment for his log to be created, users with no record have not made a payment yet.
Postconditions	The user can take a look at previous payments made and subscriptions for later renewal.
Normal Flow	The user opens his payment history tab and is redirected to his log.

Use case functional requirements:

- Clear history()

4.3 Trusted Shared Storage

4.3.1 Description and Priority

The trusted shared storage system is a system feature that allows users to create, edit, read, and share files in a secure manner. It is thus a feature of highest priority.

4.3.2 Stimulus / Response Sequences

This system Feature is triggered when the user wants to create a new file or manage an existing one. Managing a file includes reading, editing and saving, sharing, or deleting the file.

4.3.3 Functional Requirements

Use Case ID	#12
Use case Name	Create File
Description	This function is responsible for creating a new file.
Actor	S3PP user
Preconditions	This requires the user to be logged in and to be using a certain identity.
Postconditions	A new file is created under the same user identity.
Normal Flow	A logged-in user with a certain identity accesses the TSS system. The user chooses to create a new file associated with this certain identity, File Creation Identity, (FCD).

Use case Functional Requirements:

- StoreFile()

Use Case ID	#13
Use case Name	Manage Existing File
Description	This function is responsible for managing an existing file.
Actor	S3PP user
Preconditions	This requires the user to be logged in and to assume the same FCD as that of the file to be managed.
Postconditions	If the user assumes the same FCD, he/she can read, edit, store, or share the file.

Normal Flow	1. A logged-in user with a certain identity requests access to manage an existing file. 2. The TSS system allows so if the user sume the same FCD as of the file. The user is free to read, edit, or share the file..

Use case Functional Requirements:

- ReadFile()
- EditFile()
- ShareFile()
- DeleteFile()

Use Case ID	#14
Use case Name	Encrypt File
Description	This function is responsible for encrypting a stored file.
Actor	The TSS system
Preconditions	This requires that the user stores a file.
Postconditions	If the user stores a file, TSS encrypts the file such that its content can only be accessed by assuming the same FCD.
Normal Flow	1. A logged-in user with a certain identity saves a file. 2. The TSS system encrypts the file such that decrypting it can only be done by the same user identity.

Use case Functional Requirements:

- EncryptFile()

4.4 Social System:

4.4.1 Description and Priority

This system is mainly designed to help and motivate interaction between users. It allows the user to post new content, chat and search. Moreover, it gives the user some recommendation for restaurants or pages.

4.4.2 Stimulus and Response

This system is invoked when the user tries to interact or lookup content. It is also called when the user tries to socialize with other users.

Use Case ID	#15
Use case Name	Post.
Description	This function is responsible for posting new content.
Actor	S3PP user
Preconditions	This requires the user to be logged in and to be using a certain identity.
Postconditions	A new post is created under the same user identity.
Normal Flow	A logged-in user with a certain identity accesses the Social system. The user chooses to create a new content, whether it is a blog, media, voting, survey….etc.

- PostBlog()
- PostVoting()
- PostPhoto/Video()
- StreamAudio()
- PostSurvey()

Use Case ID	#16
Use case Name	Send a Message.
Description	This function is responsible for sending new messages.
Actor	S3PP user
Preconditions	This requires the sender to be logged in and have the receiver in his ring.
Postconditions	A new encrypted message is sent.
Normal Flow	A logged-in user with a certain identity accesses the Social system. The user chooses to send a new message to one or more. The message can contain a text, photo, audio recording or a video.

- SendPhotos()
- SendText()
- SendAudio()

- SendVideo()

Use Case ID	#17
Use case Name	Open Device Files.
Description	This function is responsible for navigating the filesystem.
Actor	S3PP Algorithms
Preconditions	A Function that needs to access the device files.
Postconditions	A file should be of a supported format.
Normal Flow	Opens the selected file.

Use Case ID	#18
Use case Name	Recommend
Description	This function is responsible for recommending new pages to like or restaurants.
Actor	S3PP API
Preconditions	GPS and tracking the like and search history.
Postconditions	Displays and notifies the user of the recommendation.
Normal Flow	It accesses the databases and starts recommending pages and it uses the GPS to find nearby restaurants.

- RecommendRestaurant()
- RecommendPage()

Use Case ID	#19
Use case Name	Request Access
Description	This function is responsible for requesting access to use a piece of hardware from the device.

Actor	S3PP API
Preconditions	When the user attempts to open a camera or mic or the GPS.
Postconditions	Opens the desired hardware .
Normal Flow	When the user wants to open the camera or any other hardware. The API requests access to get permission of using this piece of hardware.

- RequestMicAccess()
- RequestCameraAccess()
- RequestGPSAccess()

Use Case ID	#20
Use case Name	Search
Description	This function helps the user to find the desired content.
Actor	S3PP user
Preconditions	Logging in and having an identity.
Postconditions	Find results that best match the search .
Normal Flow	When the user wants to find some content of any type, it shows the best and closest matches for what he entered

- SearchByLocation()
- SearchByName()

5. Other Nonfunctional Requirements

5.1 Performance Requirements

5.1.1 Lightweight

The application should be lightweight and work smoothly even on older systems, as it does not require High specs.

5.1.2 Response Time

The system should have an average response time of two seconds.

5.1.3 Power Efficient

The app will run efficiently in the background since it uses devices' batteries. hence, instead of running in the background for maximum performance, it only runs minimal processes to save the battery of the device it runs on (Mobile, tablet, or a laptop).

5.1.4 Low Memory Usage

The application should use the devices' memory efficiently to save resources on mobile phones and laptops to boost performance.

5.1.5 Low Internet Usage

The application doesn't require a fast internet connection as it uses minimum data. However, a stable internet connection is recommended to enhance user's experience. This program requires reliable internet connection of minimum five Megabytes per second.

5.1.6 Scalability

The system should be able to accommodate a large number of users (1,000,000 concurrent users on release date).

5.2 Safety Requirements

- The software guarantees the safety of all transactions.
- The app is free of any kind of malware (if downloaded from IOS store and google android).
- The software provides ways to report cyber bullying.
- The software tracks the real identities of each user in case of any violation.

5.3 Security Requirements

- This software is mainly designed to protect the privacy of the users, so it should be secure,
- The messaging system should be encrypted with end-to-end encryption.
- The website and app should be accessed by the HTTPS.
- The communication between the web server and site/app is protected using SSL certificate.
- Users' passwords should be of at least length 8 and must include a capital letter and a symbol.
- Third-party bots are from trusted vendors.

5.4 Software Quality Attributes

5.4.1 Availability

- The application should be available worldwide without any restrictions.
- The application should be available on different devices and operating systems.
- Different applications stores should provide the application.

5.4.2 Accessibility

- The user should be able to change the text fonts and colors to their preferable choice.
- The application should have a color-blind mode.
- Text to speech availability for accessing the software.
- Zooming in the application should be available. (triple finger touch)

5.4.3 Reliability

- The system should be maintained regularly, and updates should be available for the user and is easy to get the newest update.
- Downtime should not exceed 10 mins for updates.

5.4.4 Ease of Use

- Upon starting the application for the first time, the user should be prompted to enter a tutorial to help introduce the user to the different aspects of the application and how to get the best experience out of it and the tutorial should be available in settings to be played again if needed by the user.
- A detailed description of the software's functionalities and features.
- Users should be able to easily navigate the application after the tutorial is finished.
- Informative error messages should be output to the user when needed.
- The Graphical user interface should have accessible buttons for different tasks.

5.4.5 Maintainability

- The system should be regularly maintained and is designed in a way that is easy to update and add/remove features.
- Regular maintenance should be accounted for and updates are easy.

5.4.6 Robustness

- The application should not be vulnerable to any breaches.
- The application should crash on very rare occasions and should be up and running within 5 minutes on average and with no need for interference from the user.
- The program should not compromise the device's resources or space and should use the minimum.

5.4.7 Portability

- Files and information owned by any user can be accessed easily and from multiple devices at once.
- Activities, messages and logs can be backed up and restored on other devices.
- Identities can be accessed from devices that support the program.

5.5 Business Rules

- All advertisements should abide by our terms of service and apply our rules and regulations and any offences should be reported and/or removed if these terms are violated.
- Cryptocurrency use should align with certain countries and regions that accept it.
- Advertisements should not ask for any personal data and targeting a certain audience is done anonymously.

Appendix A: Use Case Diagrams

General System Structure

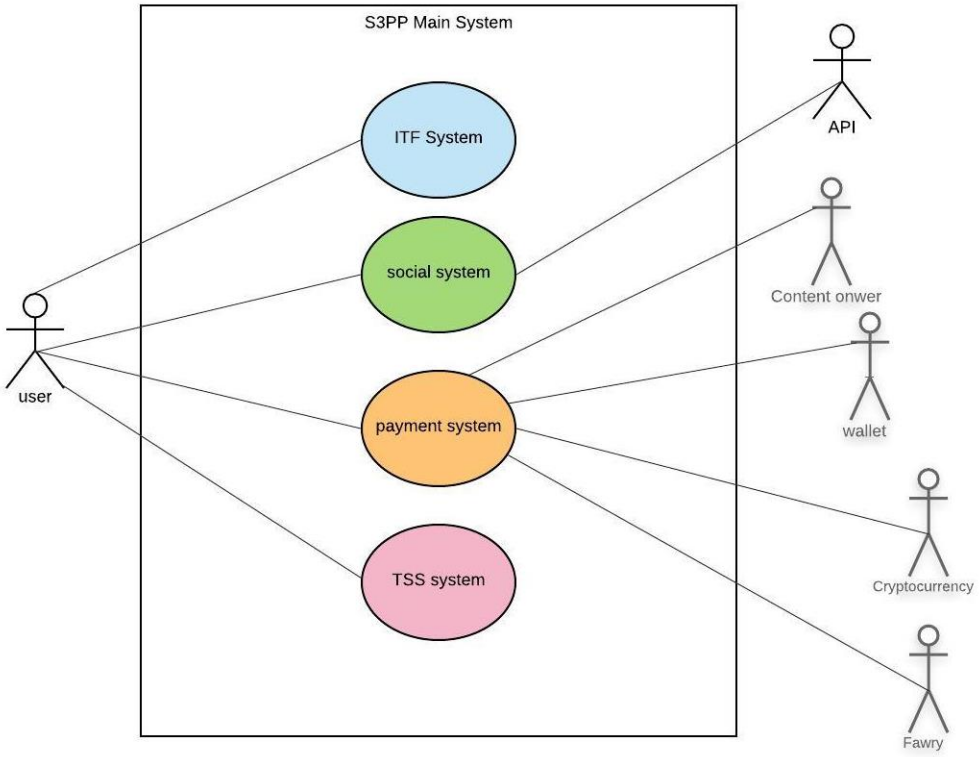

Figure 11

ITF Use Case Diagram

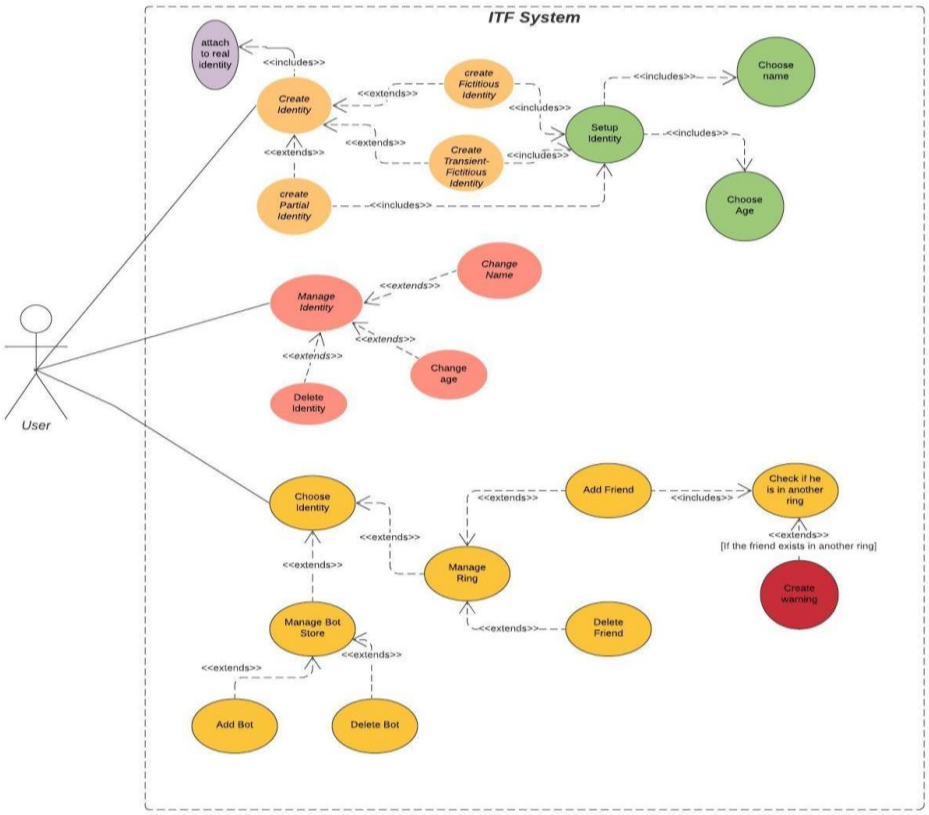

Figure 12

Payment System Use Case Diagram

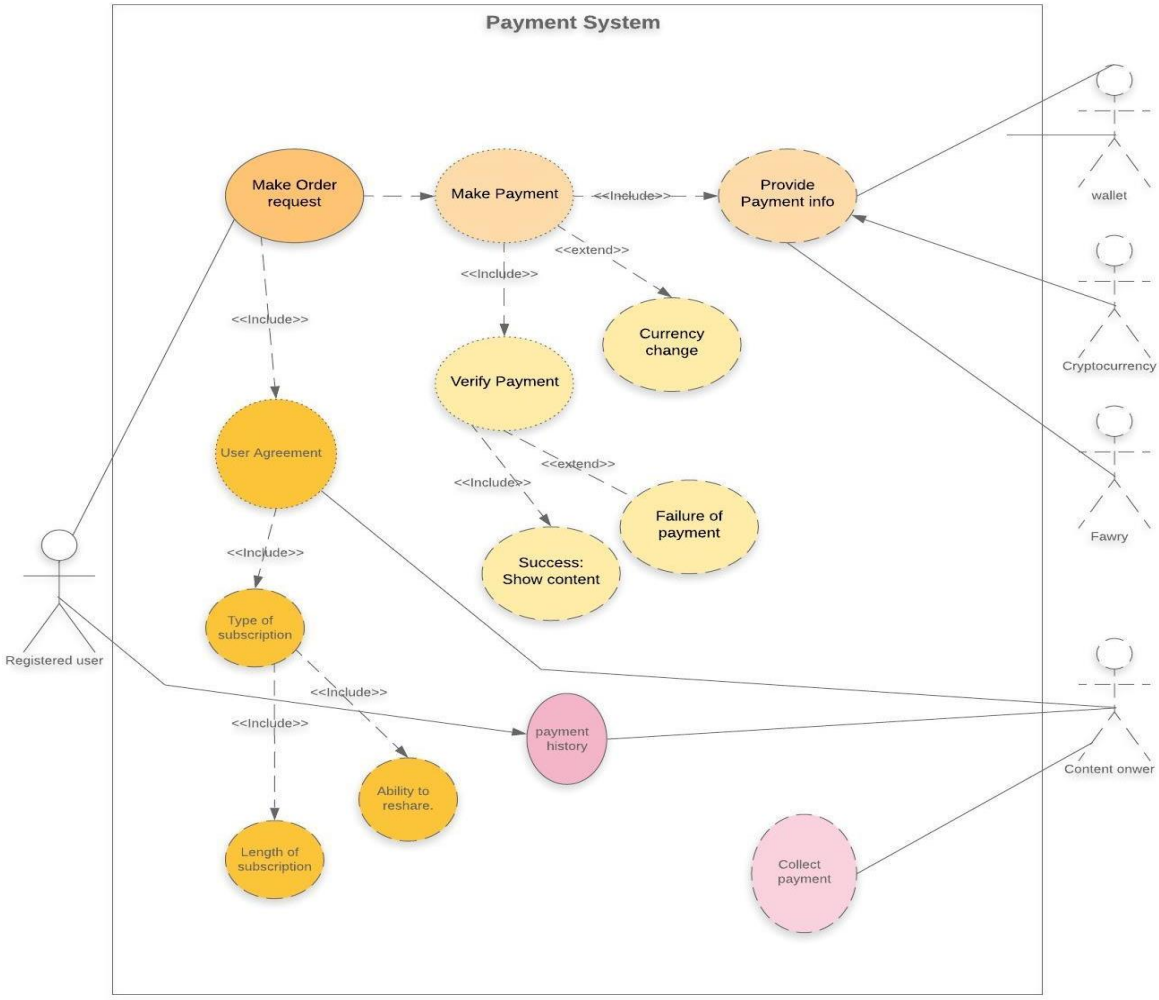

Figure 13

TSS Use Case Diagram

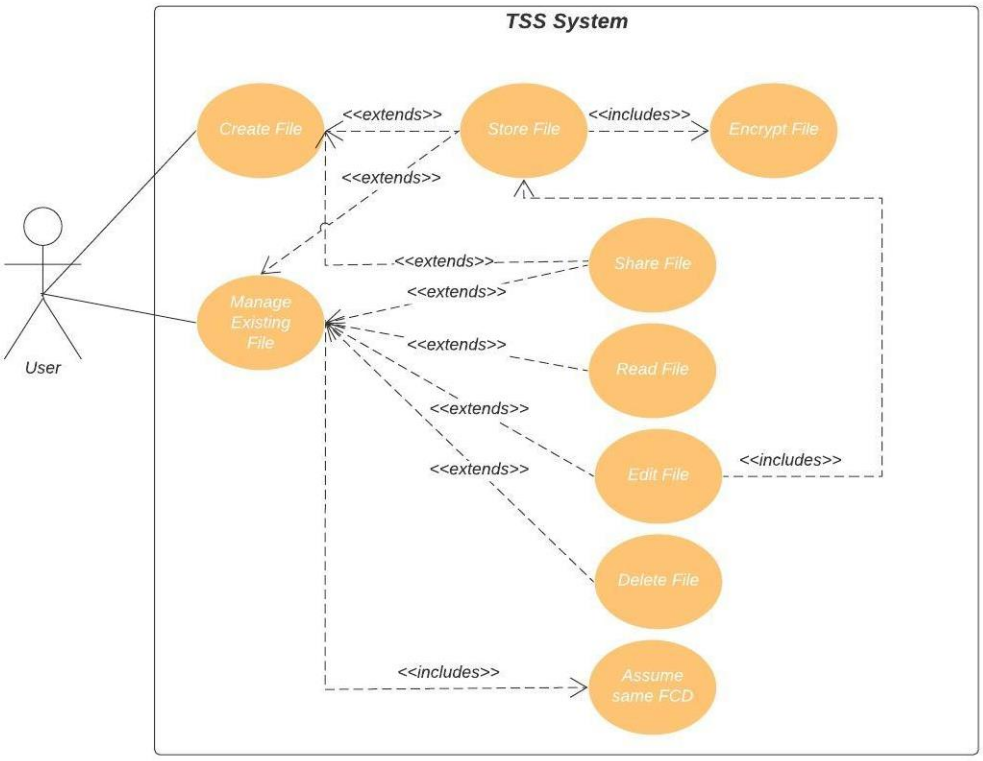

Figure 14

Social System Use Case Diagram

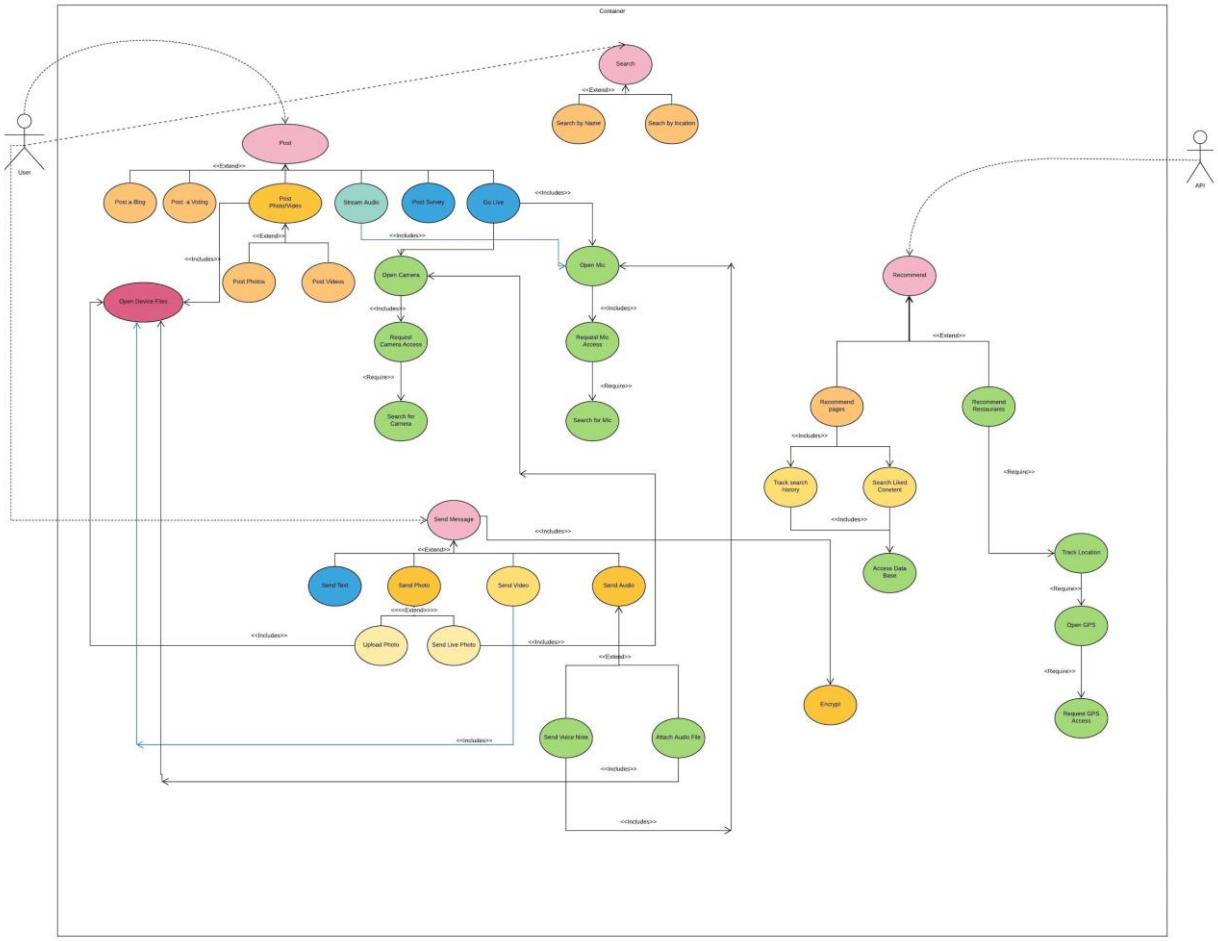

Figure 15

Appendix B: Activity Diagrams

Activity Diagram I

Logging in, selecting a bot and trying to access a file

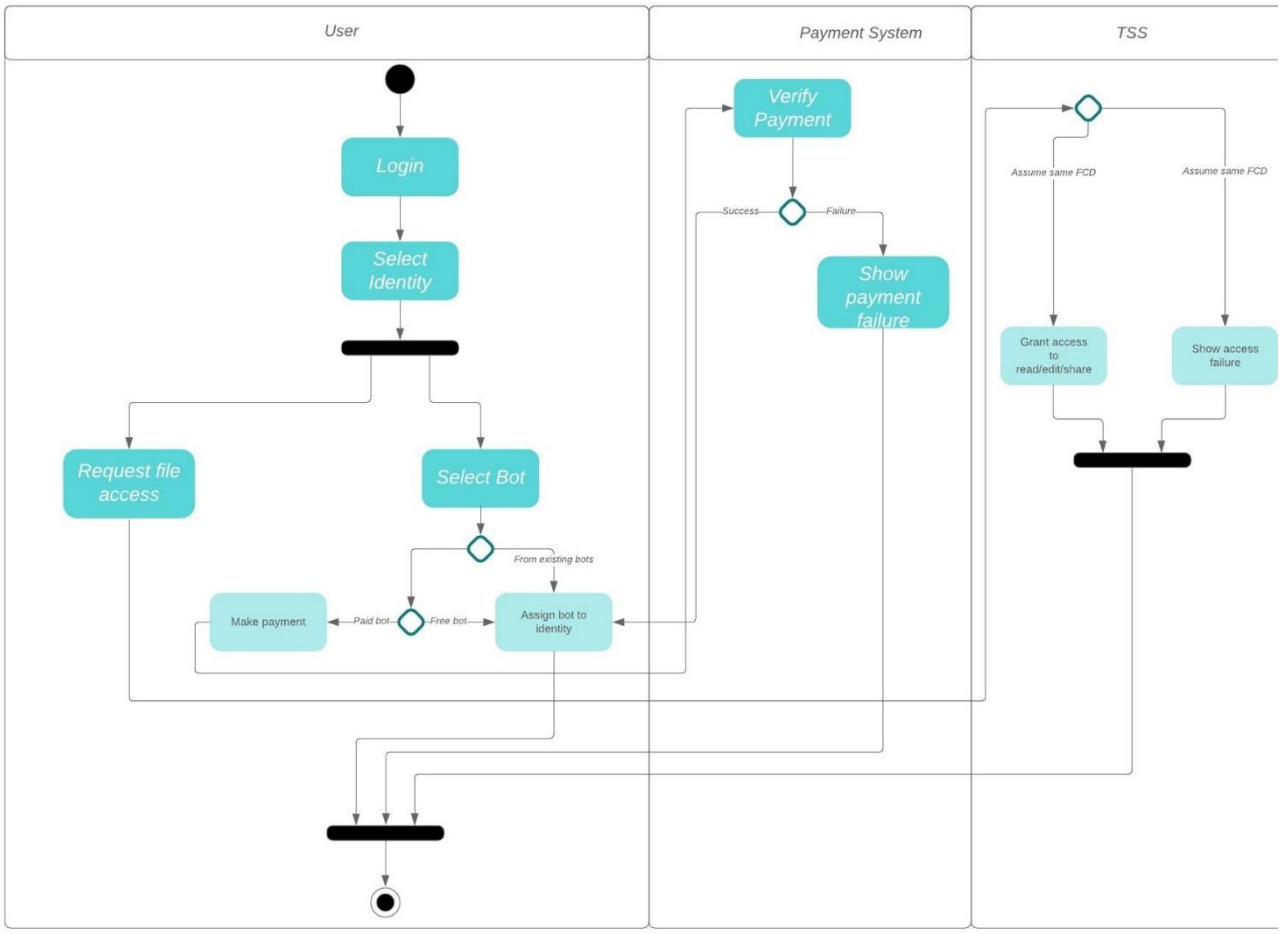

Figure 16

Activity Diagram II

Logging in, selecting an identity and using the features of the social system

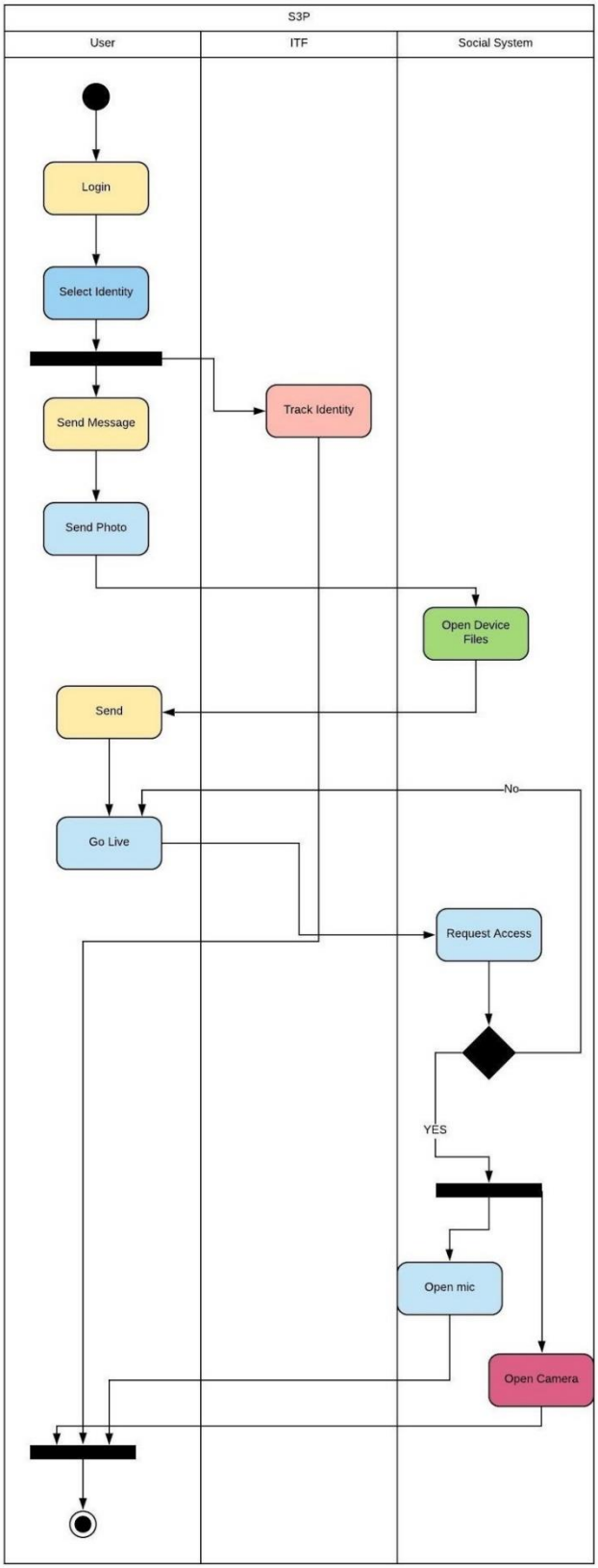
S3P
User
ITF
Social System
Login
Select Identity
Track Identity
Send Message
Send Photo
Open Device Files
Send
No
Go Live
Request Access
YES
Open mic
Open Camera

Figure 17