Hush

The secure texting, calling, and media sharing mobile app

Caroline El Jazmi

Section 1 - Project Description

1.1 Project

Hush: The secure texting, calling, and media sharing mobile app

1.2 Description

Approximately 84% of Americans feel as if they have little control over data collected about them by companies and the government (Auxier, 2020). User data is often compared to oil, it is shared and sold by companies and entities, often time to craft user personas for targeted marketing and push political agendas. Mobile phones are a target for data collection and not knowing what is done with the information being collected about users every single day is not only unethical but also an invasion of privacy.

Hush, the free mobile application, has a no-frills approach to secure texting, calling and media sharing. Over the last few years, Hush has become one of the most popular mobile communication apps around. Chosen by tech-savvy, cybersecurity experts, government officials, and many more, Hush provides users with a range of mobile phone communication tools. The app allows users to connect with other Hush users privately by preventing data from being collected, stored or intercepted by the Hush app, phone companies, or the government.

Contents

Section 2 - Overview

2.1 Project Goals & Objectives

- Identify customer segment’s needs, wants, pain points and how to address them to improve customer experience
- Design a secure mobile communication software system that prevents data from being stored, collected, and intercepted when using the app.
- Design a user-friendly UI, built for easy navigation during everyday use.

2.2 Key Customer Segments

- Hush app users who are very concerned about data privacy and user info leaks + Requires additional security features.
- Hush users who aren't overly concerned about data privacy and user info leaks. + Satisfied with general features
- Consumers who haven't downloaded the Hush application yet but want to communicate with a Hush app user.

2.3 Customer Personas

Peter Isherwell

"Innovation distinguishes between a leader and a follower."

Age 58
Location SF, California
Occupation CEO of leading tech. company BASH
Family Single
Income $402 million/year

Driven | Social | Creative

Bio

Peter is a tech billionaire CEO of BASH and is known to be involved into politics. All he is interested in is creating his own inventions for profit and maintaining his power. Peter wants an extremely secure day-to-day communication app that prevents unwanted third-party access especially discussing highly classified information.

Brand Influence

Goals

- Expand his influence beyond tech and over the rest of the economy.
- Maintaing BASH position as a tech. leading company in the insdustry.
- Develop new and unique innovations that customers want.

Preferred Channels

Reddit
Instagram
Facebook
Twitter
Guerilla

Smartphone Usage

Comfortability
Daily use: Text
Daily use: Call
Daily use: Media Sharing

Frustrations

- Uncontrolleable events (market voability/changes).
- Unable to access his data quickly and reliably.
- Leaked information being used against him.
- Poorly written patents that would make a product idea susceptible to theft.

Motivation

Privacy
Convenience
Social
Power
Flexibility

Kate Dibiaski

" It's like...people don't even care"

Age 25
Location Seattle, WA
Occupation PHD Student
Family Single
Income $ 30,000/year

Resourceful | Inquisitive | Passionate

Bio

Kate is a graduate student. During her shorts breaks between labs, she picks up her smartphone and scrolls through news-sources, often reading about invasive data collecting practices. She doesn't find her life remotely interesting enough to worry about leaked information and although she knows she can't stop her data from being collected, she wants to have more control over what is being shared.

Brand Influence

Goals

- Successfully complete PHD in astrophysics that will allow her to advance into a meanigful career.
- Conduct high-quality, original research.
- Find a balance between work, social life and sleep.

Preferred Channels

Reddit
Instagram
Facebook
Twitter
Guerilla

Smartphone Usage

Comfortability
Daily use: Text
Daily use: Call
Daily use: Media Sharing

Frustrations

- Being forced to take classes that don't seem to pertain to her research field.
- Outdated thinking in academia.
- Dissatisfaction with politicians and the state of the world
- Low pay as a PHD student.

Motivation

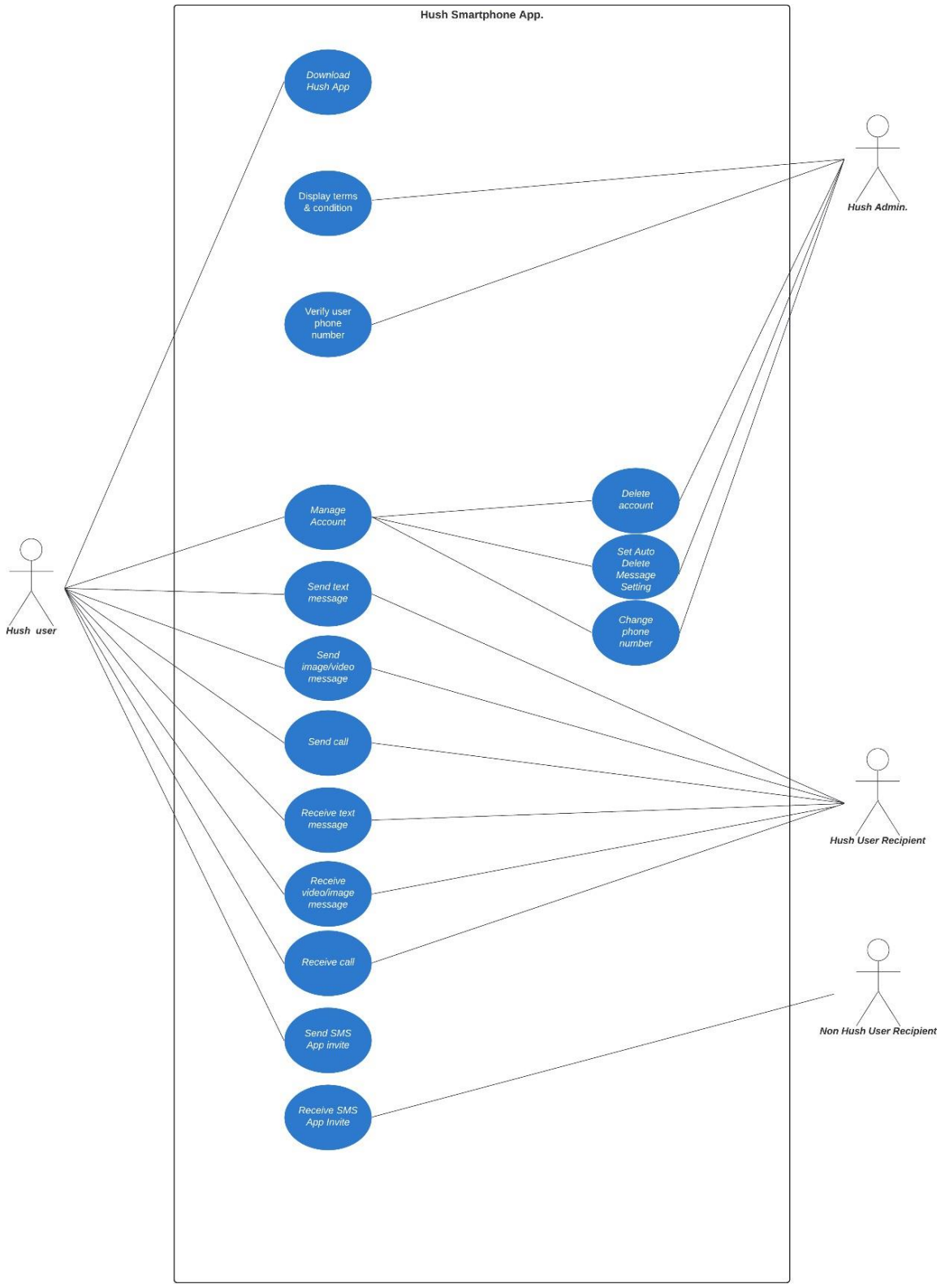
Privacy
Convenience
Social
Power
Flexibility

2.4 Customer Scenarios

Peter Isherwell is a billionaire CEO of BASH and is actively involved in politics to an extent that is currently unknown. He would label himself as a tech innovator, playing an important part in the impact of technology has made to make consumer's everyday life easier. Peter's tech intelligence, political affiliation and immense authority makes him a target. He can't afford to have the highly sensitive information he discusses with his peers to be leaked but still requires a way to communicate information from his smartphone.
< *magic happens* >
Peter discovers Hush, the mobile app, which provides him with a secure way to communicate with his peers through the app. He finds the app to be simple to use and efficient, therefore decides to make it his default calling, texting and media sharing channel when using his smartphone. Peter finds Hush to address his concerns in privacy breach and now makes it a requirement for anyone to connect with him over mobile devices, to download and use the Hush app.

Kate Dibiaski is a PHD student at UW. When she's not in a lab and doing research, she's hoping to catch a few hours of sleep and daydreaming for a better work-life balance. During her short breaks, she scrolls through her reddit feed, and listens to NPR podcasts on her smartphone. Major companies collecting data from users and using it to influence their behavior seem to be a common topic between her news outlets. Kate is aware that data is being collected from her through her smartphone and this makes her uncomfortable. Kate believes protecting all of her data is impossible but would like to start taking the steps into protecting what information is being shared about her from smartphone.
<*magic happens*>
Kate learns about the possibility of securing her texts, calls and media shares from being collected and shared by unwanted third-party actors. She discovers Hush, a simple and easy to use mobile application, commonly used by tech giants, journalists, government officials and many more. She finds out a few of her friends also use Hush and begins to use it as the main communication channel when getting in touch with those friends. Although not everyone in her social circle uses Hush actively, knowing that this type of privacy protecting application exists provides Kate with a peace of mind that she has some control over what user data she opts in to share.

Section 3 - Use Case Diagram



Section 4 - Requirements

4.1 Functional Requirements

4.1.1 User Registration

User Registration:

1) Users must be able to register for the application through a valid phone number. 2) On installing the application, users must be prompted to register their phone number. 3) If the user skips this step, the application should close.
4) The user's phone number will be the unique identifier of his/her account on the Hush app.
5) The Hush system must create a set of public and private keys for the user.
6) The Hush system must create rotating temporary public and private keys for each message/call made by the user in addition to the permanent keys to maintain high level security.

4.1.2 Adding New Contacts

Adding New Contacts:

7) The application should detect all contacts from the user's phone book.
8) If any of the contacts have user accounts with Hush, those contacts must automatically be added to the user's contact list on the Hush app.
9) If any of the contacts have not yet registered on Hush, the user should be provided with an invite option that sends those contacts a regular text message asking them to join Hush along with a link to the Hush application on the app store.

4.1.3 Send Messages

Send Messages:

10) Users should be able to send instant messages to any contact on his/her Hush app contact list. 11) Users should be notified when a message is successfully delivered to the recipient by displaying a tick sign next to the message sent.
12) The user should be able to turn on the auto delete message feature to erase encrypted messages from the user's device every time the user closes the app.
13) The application must delete all encrypted messages from the user's device local database if the auto delete message feature has been turned on by the user.

4.1.4 Send Attachments

Send Attachments:

14) Users should be able to send video and images as attachments.
15) Video formats that the application should support: avi mp4 flv gif
16) Image formats that the application should support jpg png
17) File size must not exceed 2MB per message.

4.1.5 Message Status

Message Status:

18) User must be able to get information on whether the message sent has been read by the intended recipient
19) If the recipient reads the message, 2 ticks must appear next to the message read

20) The application should notify the user if a change has been successfully applied to the user account
21) The application must be notified if a voice call is not able to be transferred to a recipient user.

4.2 Non-Functional Requirements/Software Attributes

4.2.1 Scalability

Scalability:
1) Hush should be able to provide instant messaging services to 1 billion users at any given time. 2) Hush should be able to work on any android 2.1 and up version and any iPhone OS 3.1 and up version.

4.2.2 Confidentiality

Confidentiality:
3) Messages shared between users must be encrypted from any third party actors, Hush app included, to maintain user privacy/security.

4.2.3 Performance

Performance:

4) Communication functions between user and recipient must be fast regardless of a local server being down.
5) Hush must be able to perform in adverse conditions, extremely slow connections and low battery on the device.

4.2.4 Reliability

Reliability:
6) Hush app must be reliable to ensure users are able to perform critical/secure communication with recipients.
7) Hush code must be reliable and easily maintainable to ensure user privacy is not at risk. **4.2.5**

Transparency

Transparency:
8) Hush code should be transparent to users in order to convince users of trustworthiness. 9) Hush code should be transparent to developers in order to convey the intentions and guarantees that Hush offers.

4.2.6 Usability

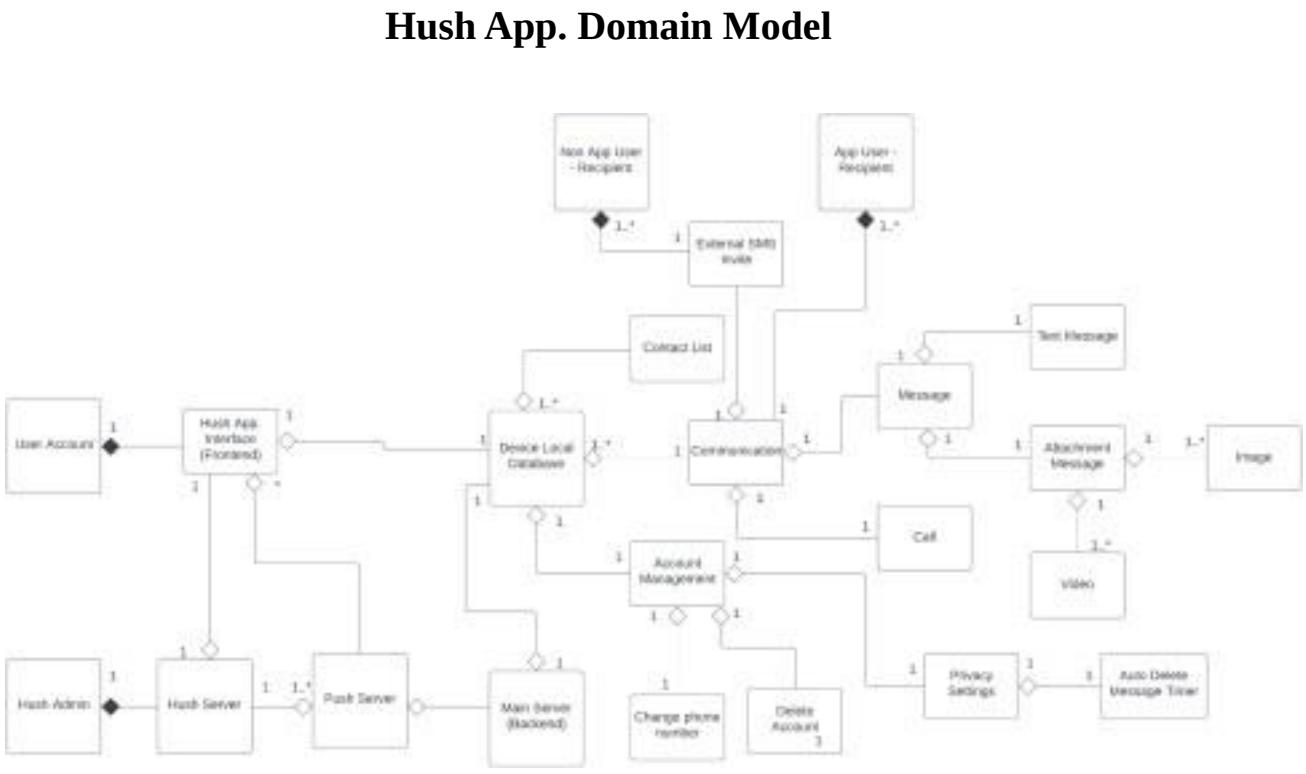
Usability:
10) Hush app should be easy to use, free from cluttered features with wide array of options.
11) Hush app interface should be like their smartphones messaging and calling features to increase user learnability.

Section 5 - 4+1 Architecture Diagram

5.1 Use Case View

Peter Isherwell, the tech CEO and **Kate Dibiaski**, the graduate student both want a secure way to communicate with their peers. Their relationship with Hush should be effortless, simple enough for everyday use. Peter is security conscious and requires higher level security features beyond what the general features offered by the Hush app. Because of Peter's role in the tech industry and political affiliation, Hush would become his default message calling app and wants to make sure that everyone he communicates with also utilizes the Hush app. Kate on the other hand isn't too concerned about what information could be leaked from her mobile communications but wants to feel like she has some control over what data is being collected from her.

5.2 Logical View - Domain Model

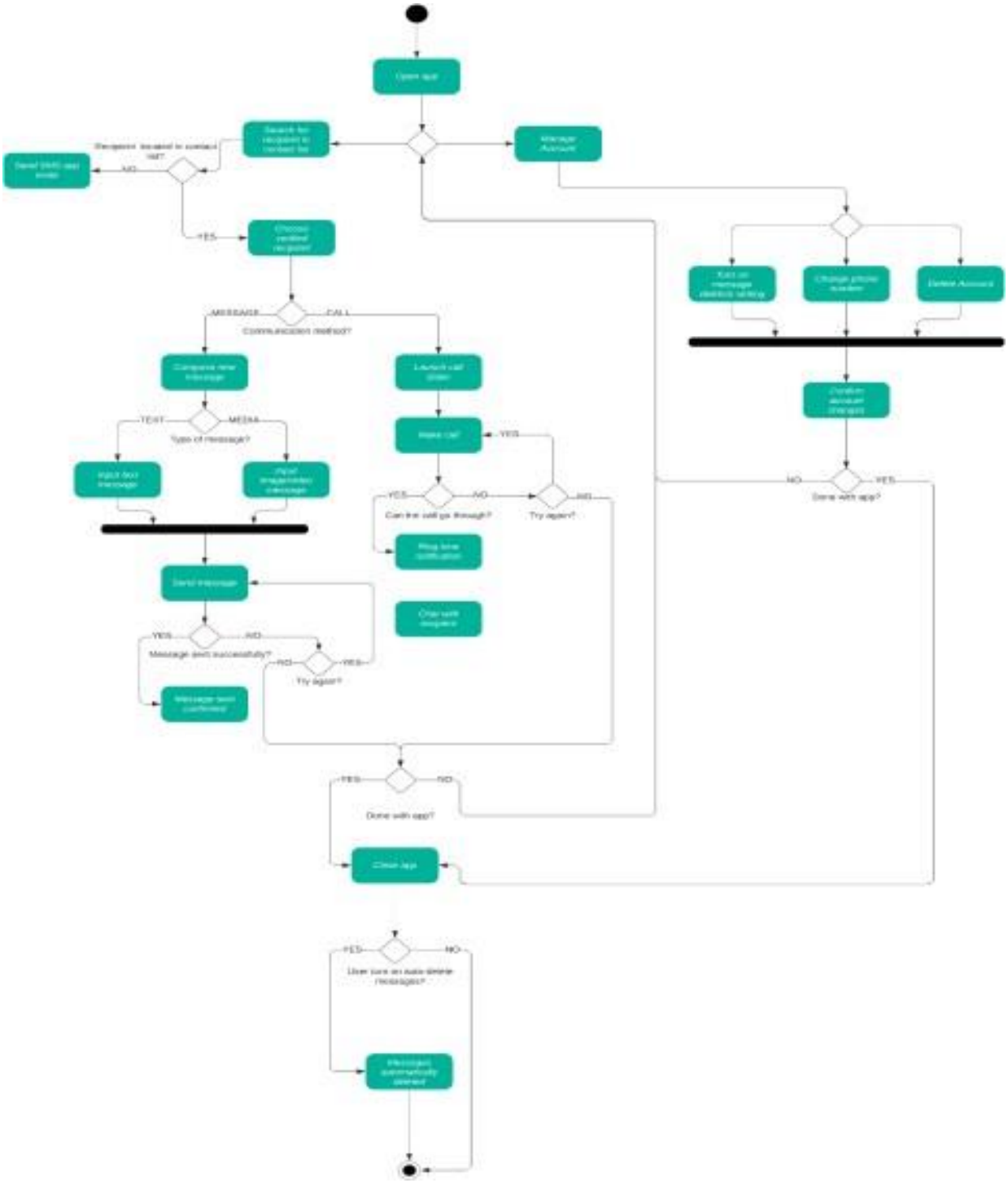


The domain model represents the Hush smartphone application, a secure privacy-central alternative to existing messaging and calling functions provided by android and iOS smartphones. The app delivers secure end-to-end encrypted communication to the app users when using the Hush app. Hush also gives app users the option to invite non-Hush users to download the app. The diagram displays Hush's client-server interactions and feature options that an app user would find useful and/or necessary. Through the application the app user is able to manage their account and/or enhance their privacy settings by turning on auto delete messages which as a result deletes the user's encrypted messages from the user's local device storage.

User Concerns:	**Decisions/Rationales:**
Communication with a non-Hush user	Invite non-Hush users to download the app via a regular SMS text message.
Sufficient Account Management	Through the smartphone application, the app user is able to manage their account. The application allows for deletion of the user account and changing of the user's phone number. These features require phone number verification from the Hush admin. As a security protocol.
Secure Communications	Secure end-to-end encrypted messaging and voice call, managed by Hush servers and achieved with encryption keys stored on the user'slocal database.
User messages accessed by third party	To avoid the potential of leaked messages, external or data transfers, Hush provides app users with the option to auto delete all messages when the user closes the app. This feature decreases the chances of having the app users' messages accessed by an unauthorized third party through stealing and by-passing the app user's secure app. password.
Simple communication features with Hush User	The diagram demonstrates an easy-to-use application, free from cluttered features with a wide array of options. Similar to a general smartphone texting/calling feature, Hush would indicate new messages and calls via push notifications.

5.3 Process View - Activity Diagram

Hush App. Activity Diagram

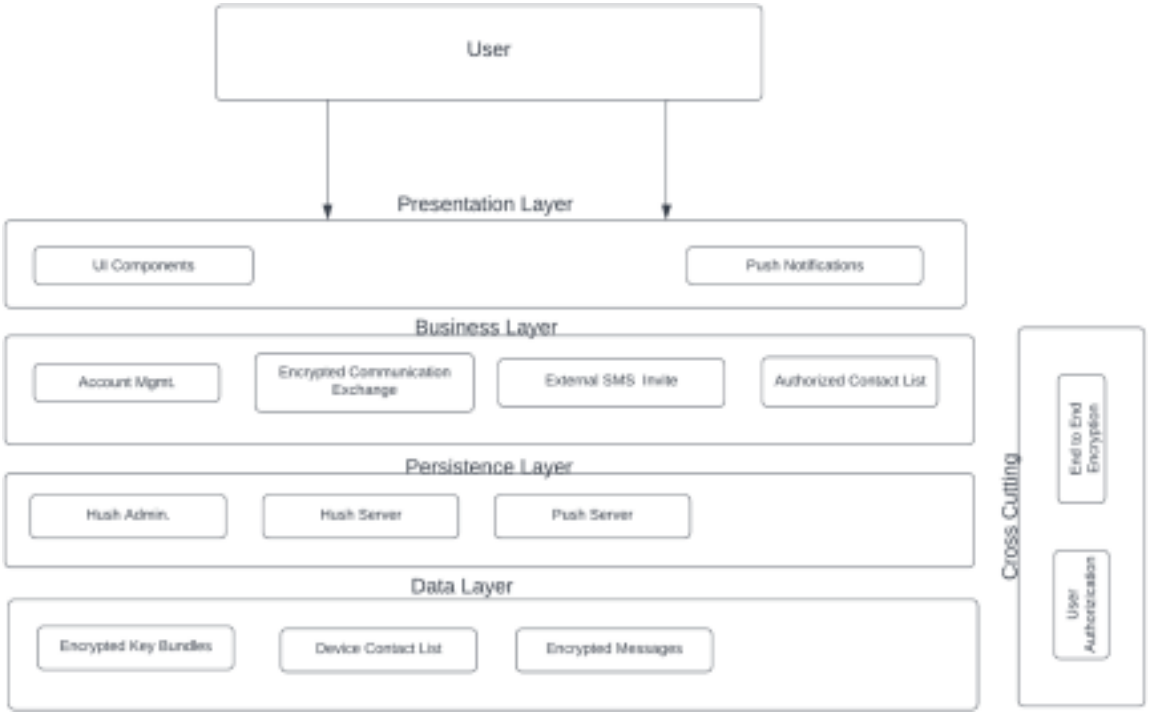


The activity diagram is designed as a series of actions available to a Hush app user when sending out a message, making a call or making changes to their user account. When using the communication features, the diagram displays a confirmation to the user if the action requested has been carried out successfully. The Hush app also displays a confirmation to the user when configurations are successfully made to the user's account.

User Concerns:	**Decisions/Rationales:**
Usability	The diagram demonstrates how the application leads users to each function. With fewer options, the Hush app is easier to use and easierfor engineers to maintain its software
Addition security features for more privacy conscious users	The diagram demonstrated the auto message delete feature that a user can enable through the account management option. Once the user turns on the feature, the change is confirmed by the app and applied after the app user closes the Hush app.
Confirmation that requested the app. actions/behaviors are performed successfully.	The diagram demonstrates the how the software system notifies app user once a call, message or account management has successfully done its task, reassuring the user that the app's secure features are functioning accordingly.

5.4 Development View - Layered Diagram

Hush App. System Layered Diagram



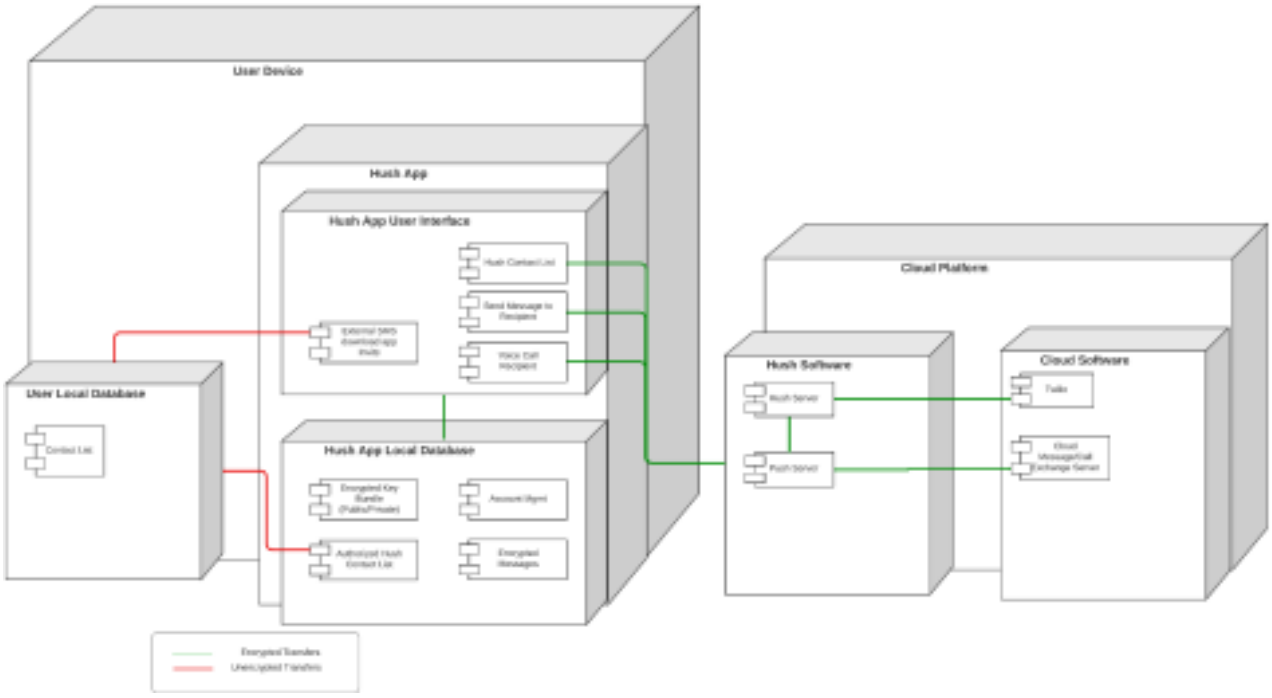
The layered diagram demonstrates how the Hush app. System interacts with the internal database stored on the user's device to ensure secure communication exchanges. . The persistence layer acts as a gateway between the data later and the functions of the app, transferring encrypted communication from one user to another.

User Concerns:	**Decisions/Rationales:**
Security between users during communication exchanges	The persistence layer manages the authentication of users and relays and temporarily stores the end-to-end encrypted messages/calls during transfer (communication exchange). The data layer stores the user's encrypted key bundles within the user's device, ensuring security from unauthorized third parties.

Incoming notifications of new messages and incoming calls from authorized Hush user's	Push notifications located in the presentation layer notifies users of incoming messages/voice calls. This layer requires the business and cross cutting layer's user authentication to accept notifications from authorized Hush users. Without this layer, the Hush app would have to be permanently connected to the Hush Server, reducing security and app users would have to manually check for new messages/calls by opening the app.

5.5 Physical View - Deployment Diagram

Hush App System Deployment Diagram



The deployment diagram demonstrates the hardware, software and cloud platform relationships required for secure communication from an app user device. The Hush app maintains secure communication exchanges by storing app data within an encrypted localized database on the user's device. The Hush app also implements end-to-end encryption of all communication and data transfers when migrating outside of the app.

User Concerns:	**Decisions/Rationales:**
Reliable communication exchanges between user and recipient at all times when using the app	Reliability of the code over time is achievable through deployment cycles that use beta testers to verify that important functionalities of the Hush app works successfully. Multiple cloud based Hush Servers are available, if a server outage were to happen, another Hush server would be used as backup.
User privacy maintained even if communication exchanges were accessed during transfer	The diagram demonstrates the Hush app's local data storage. The app stores private and public generated keys locally, encrypting messages being transmitted to and from the server. No one except the intended recipient has the key to decrypt the messages being transmitted.
Authenticating recipient of communication exchange to maintain privacy.	Hush app imports contact list from user's device, authenticating Hush app user's and authorizing communication exchanges via the Hush app.

Section 6 - Data Flow Diagram and Trust Boundaries

Hush App System Data Flow Diagram and Trust Boundaries

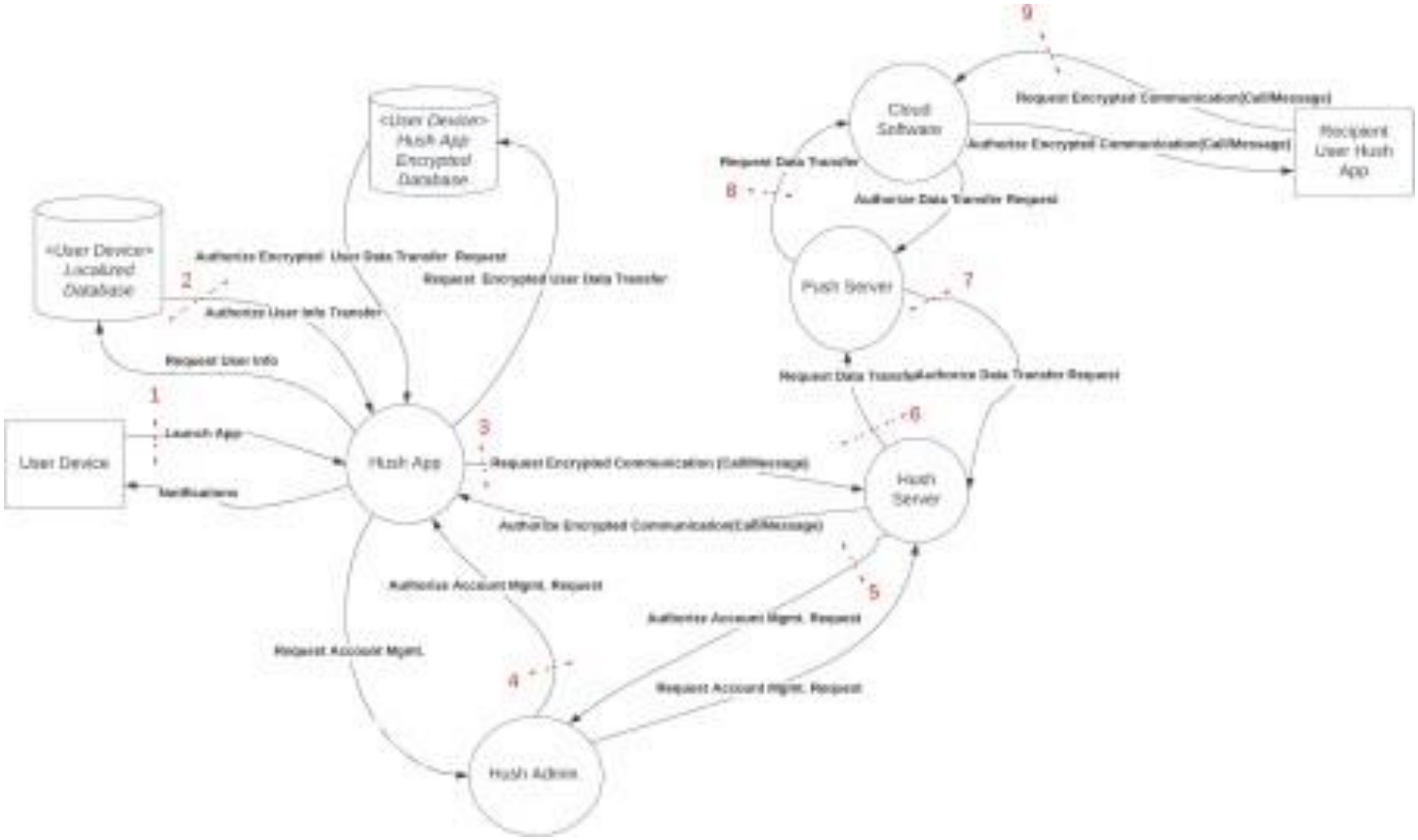


Figure 1 - Data Flow Diagram/Trust Boundaries

Section 7 - Threat Analysis

7.1 STRIDE

The Diagram (figure 6) demonstrated multiple trust boundaries. In the threat analysis below, I've targeted the threat boundaries most critical to maintaining user privacy and app security and addressed their security issues following the STRIDE method.

Data Flow: User Device to Hush App (Trust Boundary 1)

Threat Type	**Threat**	**Threat Description**
Spoofing Identity	(1) Unauthorized access to user's Hush app via unauthorized possession of user's device	Threat actor impersonates as user, gaining unauthorized access to the user's smartphones through physical acts (stealing device), breaking into the device to access the user's Hush app, jeopardizing the user's privacy
Tampering with Data	(2) Unauthorized access to user's Hush app account management properties	Threat actors gain unauthorized access to the user's Hush app and modify the user's account information, changing the app's communication address from the authorized user to the threat actors.
Repudiation	(3) Suspicious behavior from the user device while accessing the user's Hush app.	Lack of system monitoring from the user device to the Hush app. Hush app. Is unable to determine if the user device is acting suspiciously.
Information Disclosure	(4) Unauthorized access to sensitive user data components stored within the device local database. (5) Information leak due to message screenshot feature available on smartphones.	Lack of privacy-enhanced protocol if the threat actor accesses the Hush application, possibly accessing sensitive information beyond the contact list stored within the user's device local database. Recipient user screenshots and/or records user's message/voice call with the intent to share with unauthorized third parties
Denial of Service	N/A	N/A
Elevation of Privilege	(6) Data breach from undocumented backdoors within the Hush app softwaresystem.	Undocumented backdoors in the user's device, potentially leaving access into the user's Hush app making it vulnerable to data breach.

Data Flow: Hush Server to Push Server (Trust Boundary 6)

Threat Type	**Threat**	**Threat Description**
Spoofing Identity	(7) Impersonation of Hush Server and performing unauthorized communication exchanges outside of the Hush app.	Threat actor impersonates Hush App. gaining Hush Server and communicating to recipient user's, jeopardizing Hush user's privacy.
Tampering with Data	(8) Malicious attacks on server	Threat actor inserting malicious files to the server, changing the configuration of the network system and accessing user's encrypted key bundles and tampering log files to cover their tracks.
Repudiation	(9) Hush Server receiving requests from a source outside of the Hush app	Hush server receives continuous requests even though the user does not send it, with no security notification alerting the Hush server of suspicious behavior.
Information Disclosure	(10) Compromised Hush system	Hush server code is compromised, and a malicious code update is sent to the users by a threat actor, gaining access to the users encrypted keys and gaining access to their encrypted messages.
Denial of Service	(11) Cyber-attack affecting Hush app performance	Threat actors distribute a cyber-attack on Hush servers through repetitive requests, clogging the Hush server and preventing users from using the app's communication features.
Elevation of Privilege	N/A	N/A

Data Flow: User Device Localized Database to Hush App (Trust Boundary 2)

Threat Type	**Threat**	**Threat Description**
Spoofing Identity	(12) Impersonation of the Hush app to access User's contact list via the device localized database	Threat actor impersonates Hush app. gaining access to public/private encrypted key bundles stored within the localized database, jeopardizing Hush user's communication privacy and the integrity of the Hush app.
Tampering with Data	(13) Unauthorized to the user's account and user ID authentication	Threat actor inserting/modifying the user's info (contact list, encrypted key bundles, etc.) and tampering log files to cover their tracks.
Repudiation	(14) No notification alert of user device local database performing unauthorized requests from the Hush app	Localized databases receive continuous requests from the Hush app even though the user does not send it, with no security notification alerting the user of suspicious behavior
Information Disclosure	(15) Unauthorize access to the user's device local database and user's data components beyond contact list.	Hush app is compromised and accessed by a threat actor, gaining access to the users localized database and user info. Sensitive information such as contact list, social security number and banking info could be potentially jeopardized.
Denial of Service	(16) Continuous requests to the user's device local database, effecting Hush app performance	Hush app could send requests to the localized database too frequently, potentially overloading the system.
Elevation of Privilege	(17) Unauthorized modification of user's account ID address	Hush app accessed by a threat actor, gaining unauthorized use of the user's application and deleting/modifying user account information stored within the smartphones localized database.

Most Critical Trust Boundary:

Trust Boundary 1, User Device to Hush App
Hush's end-to-end encryption system prevents threat actors from interfering in-transit communications between a user and its recipients but does not prevent interference if the user's device (smartphone) is physically retrieved and then accessed or if a backdoor was discovered, allowing the user's device to access the Hush app. If this dataflow was compromised, the user's privacy is jeopardized as the threat actor would have access to the user's stored encrypted messages and contact list, potentially spoofing the user's identity to gain access to sensitive information. The integrity of the Hush app would be highly impacted by a failure in this trust boundary.

Trust Boundary 2, User Device Localized Database to Hush App
Hush app is authorized to access the user's device local database to retrieve contacts and then important Hush authorized users into the app's contact list. Tampering of this data flow is likely the result of tampering with the user device to Hush app data transfer or a threat actor impersonating the Hush app. Impersonating the Hush app and accessing the user device's local database and recovering user data such as banking into is a critical trust boundary due to the effects that banking and financial data tampering would have on a user.

Least Critical Trust Boundary:

Trust Boundary 6, Hush Server to Push Server
Communication exchanges are secured through end-to-end encryption. The servers act as a gateway between the user and the recipient. Although tampering of this data flow is critical to the performance of the Hush app, it does not jeopardize the privacy of the user.

7.2 DREAD

After identifying the Hush app system's most critical threats, the DREAD classification method assists in prioritizing each threat from most to least critical and what strategy method to implement to maintain user and system security and privacy.

Trust Boundary 1, User Device to Hush App

Threat ID Num.:	D	R	E	A	D	Average Score:	Action:
(1) Unauthorized Access to user's Hush app via unauthorized possession of user's device	7	10	8	3	8	**7**	**Mitigate**
(2) Unauthorized Access to user's Hush app account management properties	7	5	4	6	3	**5**	**Mitigate**

(3) Suspicious behavior from the user device while accessing the user's Hush app.	8	4	5	10	1	**6**	**Transfer**
(4) Unauthorized access to sensitive user data components stored within the device local database	10	2	3	19	3	**14**	**Avoid**
(5) Information leak due to message screenshot feature available on smartphones	5	7	10	3	5	**5**	**Accept**
(6) Data breach from undocumented backdoors within the Hush app software system.	7	4	6	10	6	**7**	**Mitigate**

Trust Boundary 2, User Device Localized Database to Hush App

Threat ID Num.	**D**	**R**	**E**	**A**	**D**	**Average Score:**	**Action:**
(12) Impersonation of the Hush app to access user's contact list via the device localized database	9	4	6	8	3	**6**	**Mitigate**
(13) Unauthorized modifications to the user's account settings	6	6	8	8	5	**7**	**Mitigate**

(14) No notification alert of user device local database performing	4	5	8	5	3	**5**	**Transfer**

unauthorized requests from the Hush app							
(15) Unauthorize access to the user's device local database and user's data components beyond contact list	10	5	5	10	8	**8**	**Avoid**
(16) Continuous requests to the user's device local database, effecting Hush app performance	4	8	19	5	6	**6**	**Accept**
(17) Unauthorized modification of user's account ID address without notification alert to the user.	10	6	5	10	9	**8**	**Avoid**

Section 8 - Glossary

Glossary

Term:	Definition:
End-to-end encryption(E2EE)	A method of secure communication that prevents third parties from accessing data while it's transferred from one end system or device to another
Public Key/Private Key/Key Bundle	Private Key and public key are a part of encryption that encodes the information. Both keys work in two encryption systems called symmetric and asymmetric.

Section 9 - Change Log

Change Log

Revision:	Revision Location	Description
NEW	Physical View, Deployment Diagram.	- Added Connections between the software and hardware components - Demonstrated encrypted/unencrypted data transfers - Modified the attributes nested inside the software and hardware components
	Development View, Layered Diagram	- Added cross cutting layer - Removed service layer - Removed external system layer - Modified components within the layers
	Logical View	- Added multiplicities
	Data Flow Diagram	- Implemented Data Flow Diagram and Trust Boundaries
	Threat Analysis	- Implemented threat analysis
	Use Case View	- Added details to use case view - Relocated customer scenario section to 4+1 architecture section.

Section 10 - References

Works Cited

Auxier, Brooke, et al. “Americans and Privacy: Concerned, Confused and Feeling Lack of Control over Their Personal Information.” *Pew Research Center: Internet, Science & Tech*, Pew Research Center, 17 Aug. 2020, https://www.pewresearch.org/internet/2019/11/15/americans-and-privacy-concerned-confused-and-feeling-lack-of-control-over-their-personal-information/.