Andrew McLain
CSC 3150
System Specification

Executive Summary

This system specifications document outlines how LingoJive will achieve the objectives previously described in the system proposal. Through the system proposal, the development team and investors have determined that LingoJive is a favorable project in terms of feasibility and that we should proceed to the next stages of development. This document begins by reintroducing what LingoJive is as a product, and then details the structural model, architectural design, user interface, and security aspects of the system.

I. Introduction

For further details on any of the information contained in this introduction, please refer to the system proposal.

A. Problem Statement / Project Vision

LingoJive is a "language exchange" application targeted towards individuals looking to improve their speaking proficiency in a foreign language. There are other language exchange apps on the market, but they often come with problems such as unnecessary features and misuse of the platform/inappropriate behavior from users. LingoJive hopes to address these issues and provide a safe, structured, and productive method for enthusiastic language learners to improve their proficiency.

B. System Services

Refer to section 4.2 of the system proposal for more details on this section. Use case IDs are in parentheses after the use case description.

1. Language learner functional requirements.

- Create account to chat with and send messages to other users, add friends, and rate other users. (1, 15, 19, 12, 9)
- Pay for lessons with private instructors and rate instructors. (6, 8, 15, 19)
- Upgrade to a premium account for additional features. (3)
- Adjust privacy settings and report/block users (4)

2. Language instructor functional requirements.

- Create teaching account to instruct students and receive payment (1, 2, 8)

3. Developer functional requirements.

- Resolve payment disputes between students and instructors. (10)
- Disable accounts for inappropriate behavior or misuse of the system. (5)
- Fix bugs. (13)

3. Advertiser functional requirements

- Pay for advertising space on LingoJive and place ads. (16)

C. Nonfunctional Requirements and Design Constraints

Refer to section 4.4 of the system proposal for more details on the information contained in this section.

Non-functional Requirements:

- Support customers using a variety of different platforms (operating systems, web browsers), services, and payment options.
- User experience must be streamlined, simple, convenient, rewarding, and safe.
- LingoJive's system must be secure.
- LingoJive must be able to accommodate a culturally diverse, global user base.

Constraints:

Early prototypes must be released by the end of this summer, smaller-scale testing must begin by the end of this year, and a full-scale launch must be achieved by next summer.

D. System Evolution

In Version 1 of LingoJive, we plan to implement the features outlined in the functional, nonfunctional, and data requirements as well as the models in this document. Broadly, this includes chat and messaging, adding friends, paying for lessons with teachers, and rating other users and teachers. Version 1 will support advertisements as well.

Version 2 will include the option to upgrade to a premium account as well as the features that come along with that. This includes features such as searchable profiles, filtering partners by rating, removing ads, and seeing who has viewed your profile. Version 2 will also include the ability for users to build flash card decks and unlock achievement badges.

Version 3 will incorporate more of the gamification aspects of LingoJive. This will include further upgrades to the badge system, as well as chat games and guided conversation features that will enrich the language exchange experience.

E. Document Outline

Structural Model – A class diagram representing the major interactions between objects in the LingoJive system along with their associated metadata and class descriptions.

Architecture Design – Includes two deployment diagrams (architecture overview and nodes and artifacts) as well as a table summarizing the security plan for LingoJive.

User Interface – Includes a window navigation diagram and simple mockups of the most frequently visited screens.

Appendices – Includes the bibliography

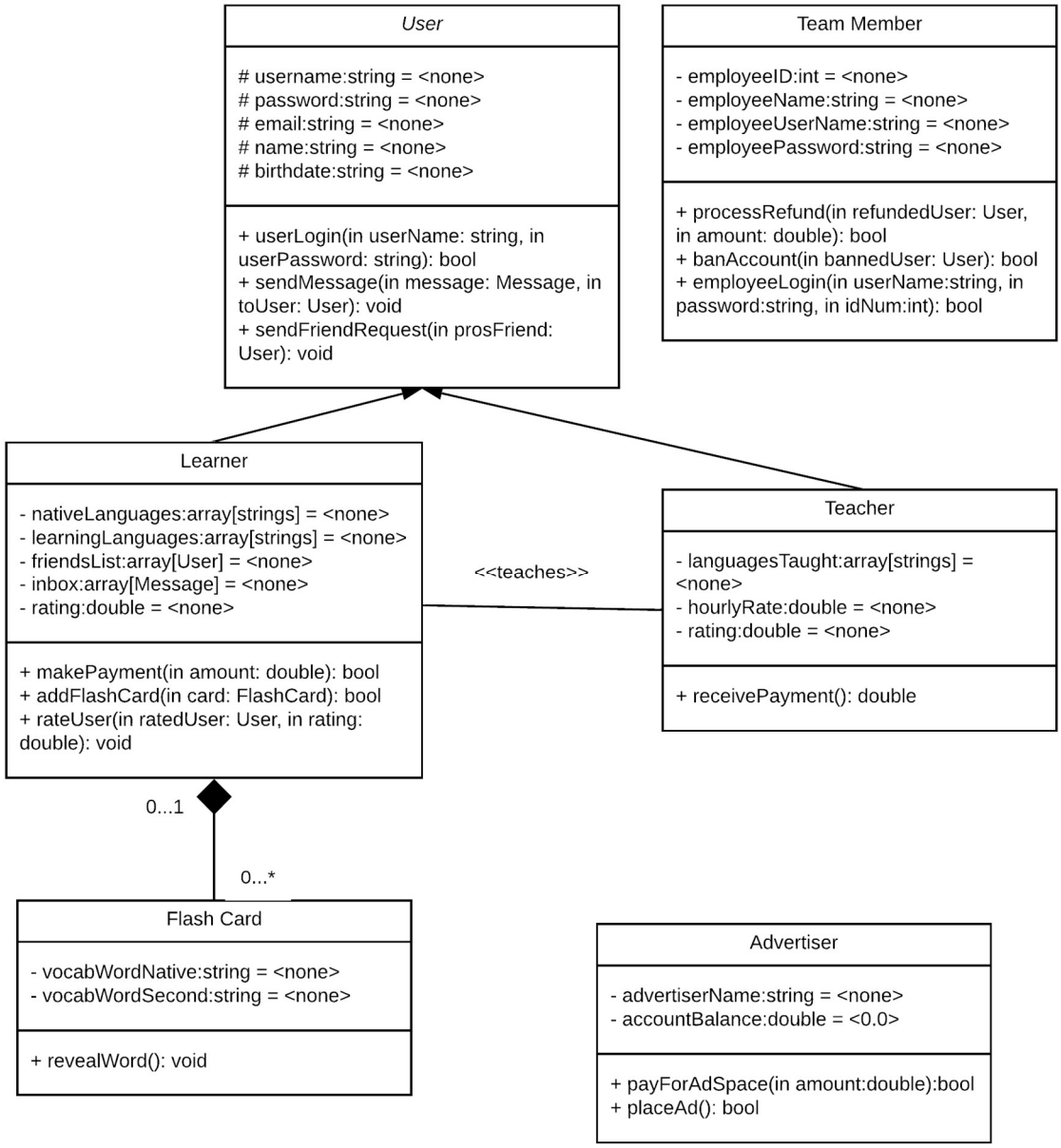
Andrew McLain
CSC 3150
System Specification

II. Structural Model

A. Introduction

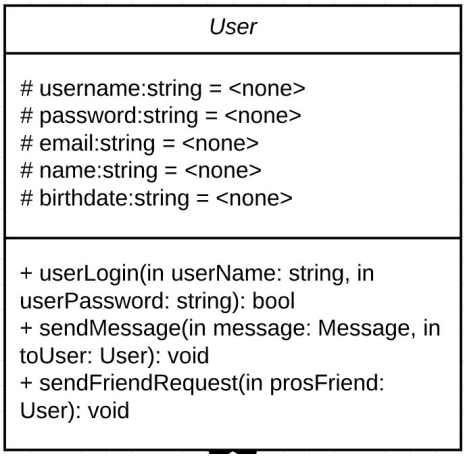
Section B is a class diagram representing how the learners, teachers, team members, and advertisers interact within the system. Section C contains a breakdown of each of these objects, their attributes, and their operations using a class description template.

B. Class Diagram



Andrew McLain
CSC 3150
System Specification

C. Metadata



Description: Represents a LingoJive user
Visibility: Public
Is Abstract: Yes

Attributes:

Name	Description	Data Type	Is Derived	Is Read Only	Visibility	Multiplicity	Default Value
username	Unique username.	string	No	No	protected	1	None
password	Secure password for login.	string	No	No	protected	1	None
email	Email address associated with account. Cannot be associated with another account.	string	No	No	protected	1	None
name	First and last name of user.	string	No	No	protected	1	None
birthdate	Birthdate (yy/mm/dd) of user.	string	No	No	protected	1	None

Andrew McLain
CSC 3150
System Specification

Operations:

Name	userLogin
Description	User logs in to their account to access LingoJive
Return Type	bool
Parameters	userName type: string direction: in default: none userPassword type: string direction: in default: none
Visibility	public
Scope	instance
Is Query	No

Processing outline:
User inputs username and password to log in.
If valid: Allow user to login
Else: Inform user of incorrect login credentials and prompt for correct login.

Name	sendMessage
Description	User sends message to friend in LingoJive
Return Type	void
Parameters	message type: Message direction: in default: none toUser type: User direction: in default: none
Visibility	public
Scope	instance
Is Query	No

Processing outline:
User sends message to another user from their friend's list

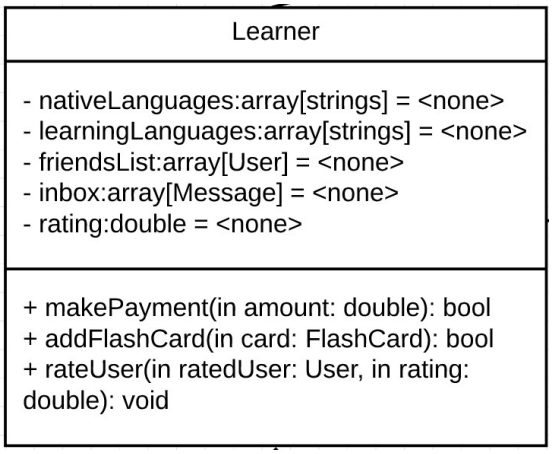
Andrew McLain
CSC 3150
System Specification

Name	sendFriendRequest
Description	User sends friend request to prospective friend in LingoJive
Return Type	void
Parameters	prosFriend type: User direction: in default: none
Visibility	public
Scope	instance
Is Query	No

Processing outline:

User sends friend request to another user on LingoJive. Other user can accept or decline the request.

Andrew McLain
CSC 3150
System Specification



Description: Represents a LingoJive learner, subclass of user.
Visibility: Public
Is Abstract: No

Attributes:

Name	Description	Data Type	Is Derived	Is Read Only	Visibility	Multiplicity	Default Value
nativeLanguages	List of user's native languages	array	No	No	private	1	None
learningLanguages	List of languages the user is learning	array	No	No	private	1	None
friendsList	List of user's friends	array	No	No	private	1	None
inbox	Inbox for user containing messages	array	No	No	private	1	None
rating	Rating of user given by other users with whom they have practiced with.	double	No	No	private	1	None

Operations:

Name	makePayment
Description	User makes payment for instruction or premium membership
Return Type	bool

Parameters	amount type: double direction: in default: 0.00 (in user's currency)
Visibility	public
Scope	instance
Is Query	No

Processing outline:
User makes payment for service on LingoJive. If payment information on user account valid and sufficient for indicated quantity: process payment. Else: Notify user of invalid payment information or insufficient funds on account.

Name	addFlashCard
Description	User adds flash card to deck for studying later
Return Type	bool
Parameters	flashCard type: FlashCard direction: in default: none
Visibility	public
Scope	instance
Is Query	No

Processing outline:
User adds card to deck of flash cards. If user flash card deck is not full: Add flash card, return true. Else (should be nearly impossible because the maximum flash card amount will be set at a nearly arbitrarily high value): Notify user that flash card cannot be added, return false.

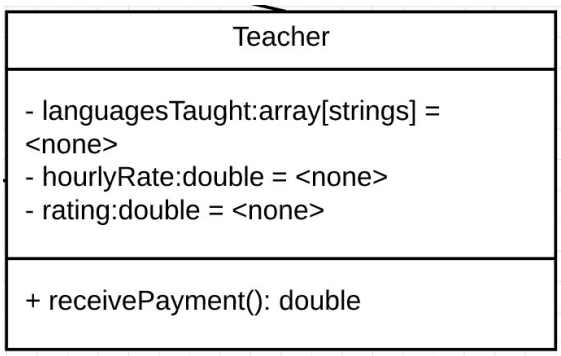
Name	rateUser
Description	User rates another user after language exchange with them
Return Type	void
Parameters	ratedUser type: User direction: in default: none rating type:double direction: in default: none

Visibility	public
Scope	instance
Is Query	No

Processing outline:

User rates another user on a scale of 1-5 stars. Rated user's rating changes based on the new rating being calculated with the mean of all of their other ratings.

Andrew McLain
CSC 3150
System Specification



Attributes:

Name	Description	Data Type	Is Derived	Is Read Only	Visibility	Multiplicity	Default Value
languagesTaught	List of languages taught by teacher	array	No	No	private	1	None
hourlyRate	Hourly rate for instruction by teacher	double	No	No	private	1	None
rating	Average rating of teacher given by their students	double	No	No	private	1	None

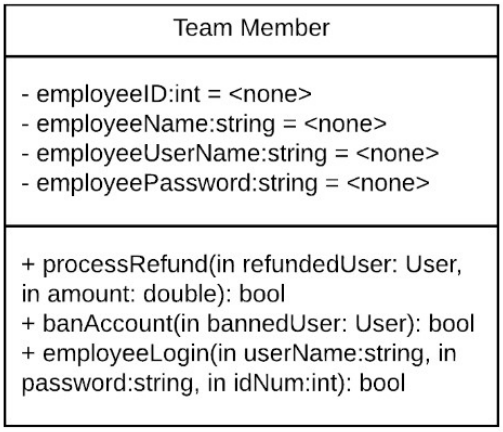
Operations:

Name	receivePayment
Description	Teacher receives money for instruction
Return Type	double
Parameters	none
Visibility	public
Scope	instance
Is Query	No

Processing outline:

Teacher receives payment and amount is paid to their account registered with LingoJive.

Andrew McLain
CSC 3150
System Specification



Attributes:

Name	Description	Data Type	Is Derived	Is Read Only	Visibility	Multiplicity	Default Value
employeeID	ID for employee	int	No	No	private	1	None
employeeName	Name of employee	string	No	No	private	1	None
employeeUserName	Username of employee for login	string	No	No	private	1	None
employeePassword	Password of employee for login	double	No	No	private	1	None

Operations:

Name	processRefund
Description	Team member processes refund
Return Type	bool
Parameters	refundedUser type: User direction: in default: none amount type:double direction: in default: none
Visibility	public

Scope	instance
Is Query	No

Processing outline:
Team member processes refund for user. Returns true if refund successfully processed, false otherwise.

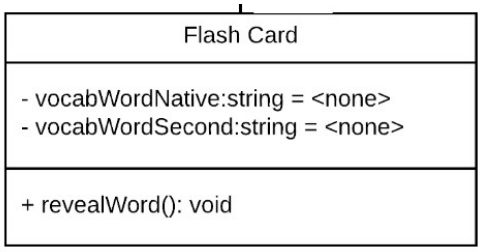
Name	banAccount
Description	Team member bans user
Return Type	bool
Parameters	bannedUser type: User direction: in default: none
Visibility	public
Scope	instance
Is Query	No

Processing outline:
Team member bans user. Returns true if user successfully banned, false otherwise.

Name	employeeLogin
Description	Team member logs in with employee credentials
Return Type	bool
Parameters	userName type: string direction: in default: none password type:string direction: in default: none employeeID type:int direction: in default: none
Visibility	public
Scope	instance
Is Query	No

Processing outline: Team member logs in with username, password, and employee ID number.

Andrew McLain
CSC 3150
System Specification



Attributes:

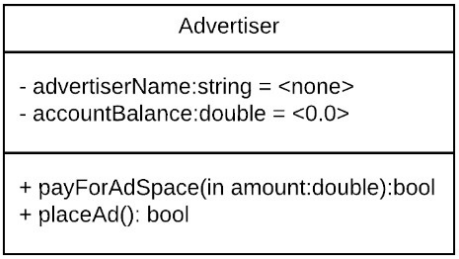
Name	Description	Data Type	Is Derived	Is Read Only	Visibility	Multiplicity	Default Value
vocabWord Native	Vocabulary word in user's native language	String	No	No	private	1	None
vocabWord Second	Vocabulary word in user's second language	String	No	No	private	1	None

Operations:

Name	revealWord
Description	User clicks to reveal translated flash card to check their memorization.
Return Type	void
Parameters	none
Visibility	public
Scope	instance
Is Query	No

Processing outline:
User clicks to reveal the vocabulary word in their native language.

Andrew McLain
CSC 3150
System Specification



Attributes:

Name	Description	Data Type	Is Derived	Is Read Only	Visibility	Multiplicity	Default Value
advertiserName	Name of advertising company	String	No	No	private	1	None
accountBalance	Amount of money in company's advertising fund on LingoJive	Double	No	No	private	1	None

Operations:

Name	payForAdSpace
Description	Advertiser withdraws money from their account balance to pay for more ad space
Return Type	bool
Parameters	amount type:double direction: in default: none
Visibility	public
Scope	instance
Is Query	No

Processing outline:
Advertiser purchases ad space using money from their account balance on LingoJive. Returns false if there are insufficient funds, true otherwise.

Name	placeAd
Description	Advertiser places ad on LingoJive
Return Type	bool
Parameters	none

Visibility	public
Scope	instance
Is Query	No

Processing outline:

Advertiser places an ad on LingoJive. Returns false if ad cannot be placed.

Andrew McLain
CSC 3150
System Specification

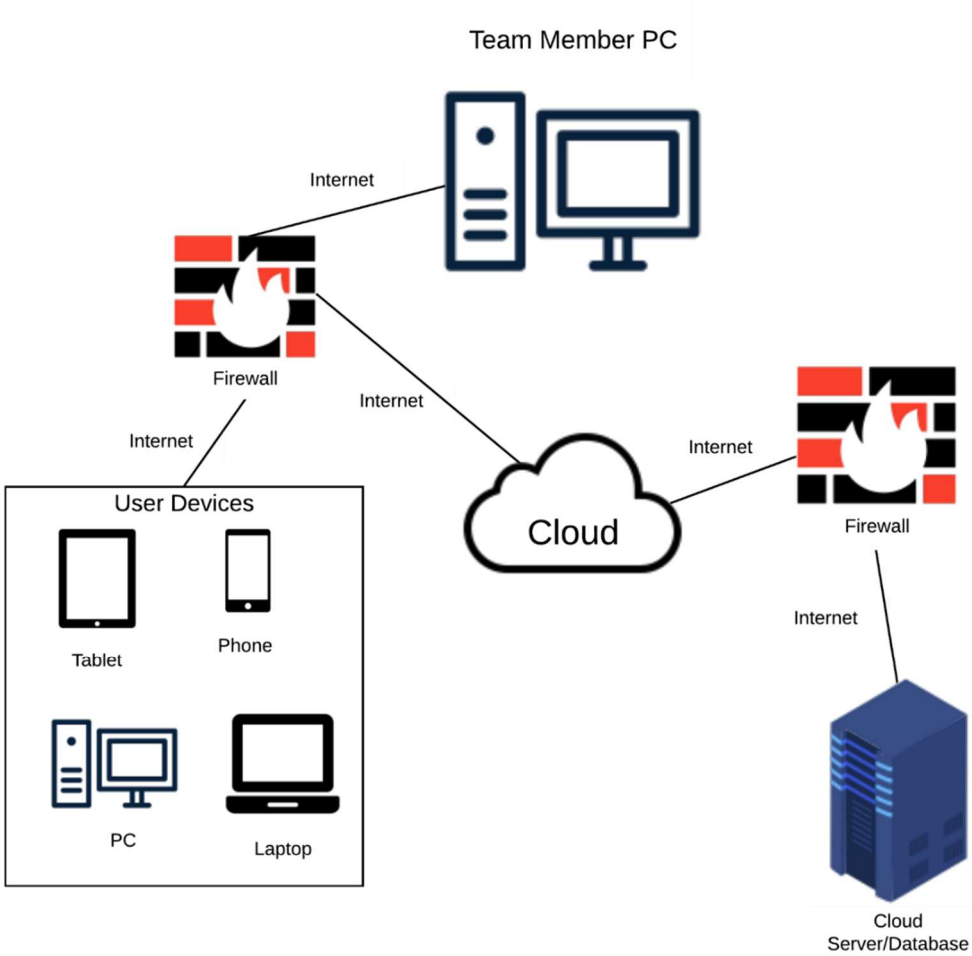
III. Architecture Design

A. Introduction

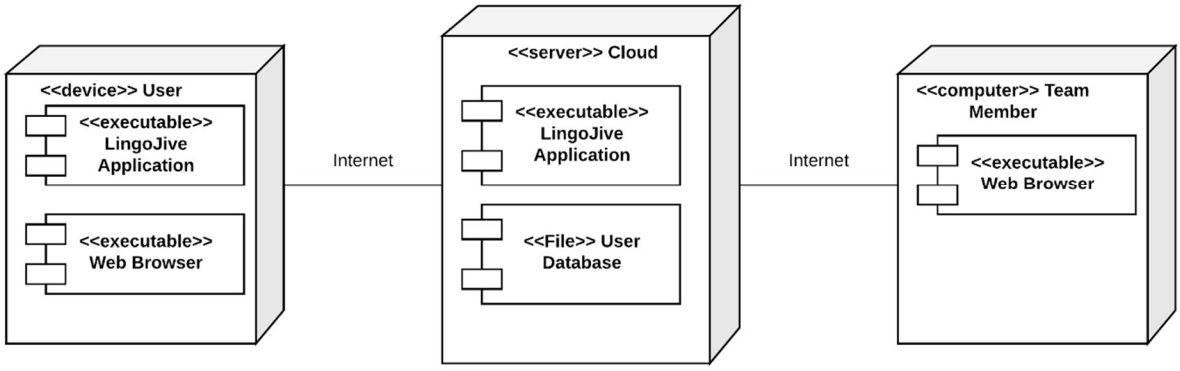
Section B includes the architecture overview of LingoJive. The architecture overview is a macroscopic view of LingoJive's cloud-based implementation, depicting the relationships between the different pieces of hardware in the system. Section C contains nodes and artifacts which depict how some of the software components relate to the hardware components in the system. LingoJive will be implemented using a thin-client architecture because most of the users will access LingoJive using their mobile devices, which have limited processing power. The processing for LingoJive will be outsourced to a third-party provider such as AWS (Amazon Web Services) for convenience, flexibility, and reliability.

B. Infrastructure Model

1) Deployment Diagram 1 – Architecture Overview



2) Deployment Diagram 2 – Nodes and Artifacts



C. Hardware and Software Requirements

Required Hardware Components

- PCs for team members to work from.
- Cloud server (subscription-based service such as AWS)
- User devices such as smartphones, tablets, and PCs. These are purchased by the users and are not factored into expenses for LingoJive.

Required Software Components

- Must support widely used operating systems such as Microsoft Windows (10.0 or later), macOS (10.0 or later), Android (4.4 or later), and Linux (5.0 or later).
- Must be supported on all widely used browsers such as Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge.

D. Security Plan

Security Table:

Components\Threats	Power Loss	Physical Theft/Loss	Natural Disaster	Unauthorized Access	Virus
User Devices	X	X	X	4, 6, 7, 8	3
Team Member PC	2	5, 9	1, 2, 5	4, 6, 7, 8	3
Cloud Server	2	5, 9	1, 2, 5	4, 6, 7, 8	3

1. Natural disaster contingency plans.
2. Back-up power supply.
3. Antivirus software.
4. Firewall between device and cloud server.
5. Insurance for hardware purchased by LingoJive.

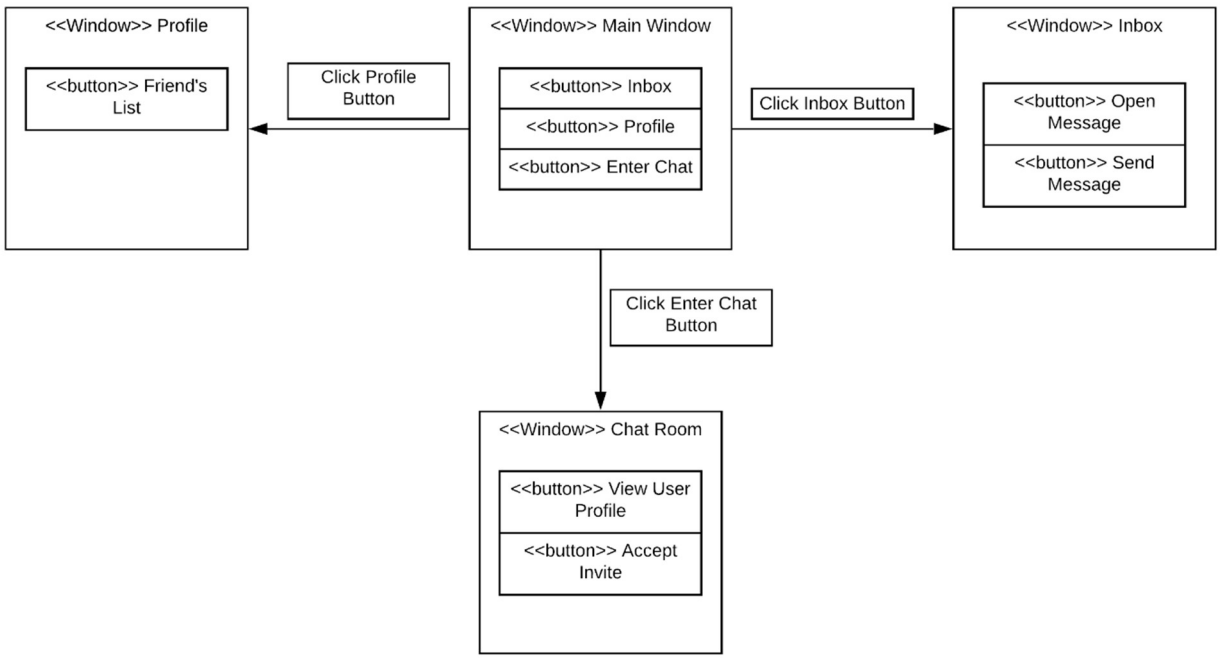
6.. High quality encryption software from a reliable source.
7. Support only reputable and secure payment methods.
8. Periodically scheduled third-party security audits.
9. Secure working facility with locks and professional security system.

IV. User-Interface

A. User-Interface Requirements and Constraints

This section helps visualize the UI for LingoJive. The highest priority for LingoJive's UI is simplicity and ease of use. Navigating the interface should be self-explanatory, and all features should be highly visible and easily accessible. We also want to minimize the amount of text on each screen so that the aesthetic of LingoJive's UI does not vary significantly in different languages. This section will roughly wireframe the three main screens for a typical user: Inbox, profile, and chat room. When users log in, the first screen they will see is their own profile. From there, they can quickly navigate to the message and chat room screens.

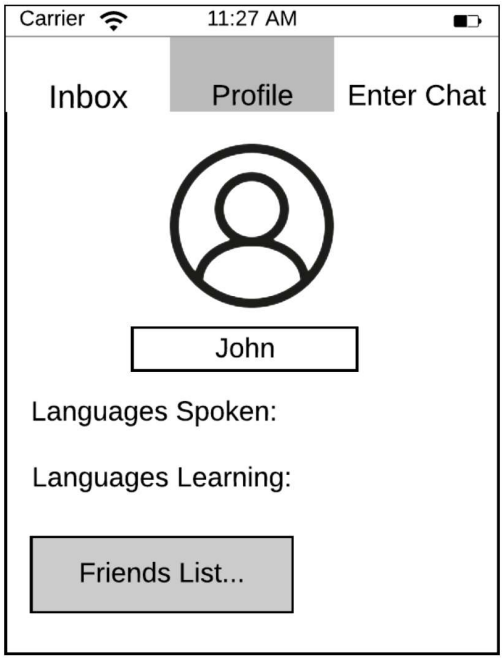
B. Window Navigation Diagram



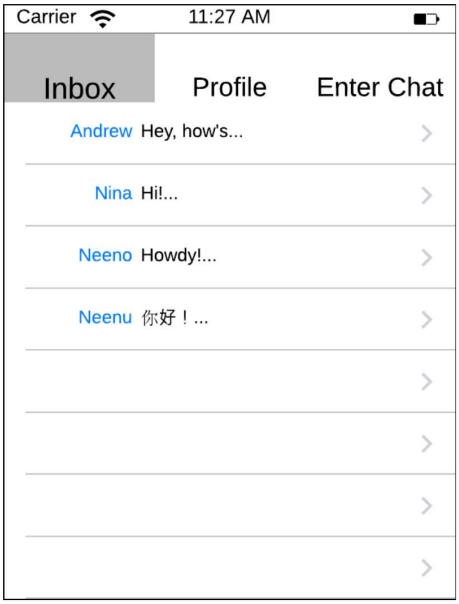
Andrew McLain
CSC 3150
System Specification

C. Forms: Screen / User-Interaction Design

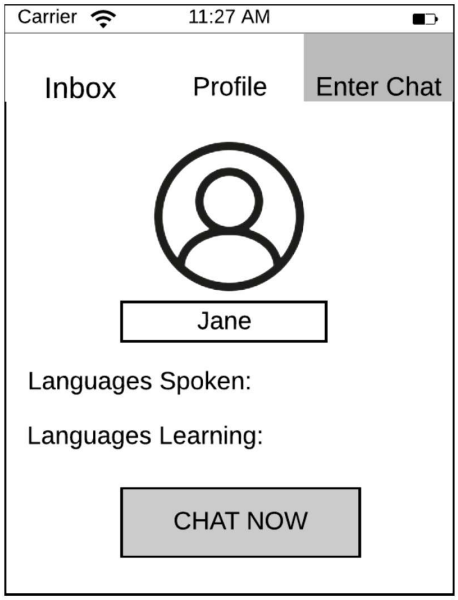
Window of user viewing their own profile.



Window of user viewing their inbox messages.



Window of user viewing chat invitation and profile of language partner in chat room.



V. Appendices

A. Bibliography / References

1. HelloTalk – Free Language Exchange: Language Partners.
   URL: https://www.hellotalk.com
   Accessed May 26, 2020.
2. Tandem Language Exchange
   URL: https://www.tandem.net
   Accessed May 26, 2020
3. Bilingua
   URL: https://bilingua.io
   Accessed May 26, 2020
4. italki: Learn a language online
   URL: https://www.italki.com
   Accessed May 26, 2020
5. Larman, Craig (2004) Applying UML and Patterns-Prentice Hall
6. Seattle Pacific University CSC 3150 Class Slides
7. Lucidchart: Online Diagram Software & Visual Solution
   URL: https://www.lucidchart.com