CSIT321 Final Year Project
Project requirements
21 Jan 2021

Project Group No: FYP-21-S1-02
Project Topic: Typing Habit Gesture Authentication System
Project Supervisor: Mr Sionggo Japit
Project Members:

Name	UOW ID	Email
Terrence Yap Yong Ting	6573307	yttyap002@mymail.sim.edu.sg
Alicia Tan Chun Ying	6457794	cyatan005@mymail.sim.edu.sg
Tng Min Li	6649737	mltng001@mymail.sim.edu.sg
Chea Cui Hui	6647546	chchea001@mymail.sim.edu.sg
Joel Teo Han Wen	6456467	jhwteo002@mymail.sim.edu.sg

Contents

1 Introduction

Overview

In the digital age, security is of great importance. Authentication has become common activity in daily life. As passwords pose a risk of being compromised due to a variety of reasons such as reusing passwords or using simple passwords, exploring a different approach to authentication could be useful.
The purpose of this project is to develop a typing gesture habit authentication system that can authenticate a person based on his typing habits.

Project Scope

Project Objective

Complete a working typing gesture habit authentication system that can recognize user based on typing habit.

1.2.2 Goals

The website must be able to recognize the user based on their typing habits.

1.2.3 Milestones

- Literature review
- Project specification report
- Progress report
- Coding of system

Project Description

This project is to authenticate users based on their typing habits instead of using passwords. Through analysis of the typing patterns using attributes that are recorded from the user, analysis would be performed on them and then determined if they match the typing habits of the user when he initially registered.

Chosen SDLC Development Methodology

The team has decided on agile Scrum methodology for this project. Scrum allows the developing team to determine the direction of the project as they know the best way to get to the solution. Through collaboration with stakeholders, the development team can better understand the need of the users and adjust the system accordingly.

Scrum consists of repeating periods of 2-3 weeks where a chosen number of user stories are discussed and incorporated into the system. The client would discuss the specifics of the user stories with the development team. After the discussion, the development team would then go on to implement the items discussed through coding and testing. This process is repeated until the system is complete. In this way, the development team can ensure that requirements are consistent with the requirements of the client throughout the process. No changes to the requirement should be made during the sprints as they should be focused on development of the system.

2 Key Stakeholders and List

Key Stakeholders

Internal Stakeholders

Name	Position	Job Description
Tan Chun Ying, Alicia	Project Manager /Developer	*Project* manager This person is key to lead the project to success. The project manager has the overall responsibility for the project's success. A project manager must do: • Planning • Analysis • Research • Design • Execution • Monitoring • Controlling • Closure The project manager is responsible for undertaking the role of decision-making for the team and delivers the expectations to the stakeholders. She has the responsibility, authority and accountability of the overall project management and to coordinate with the rest of the internal (project members) and external stakeholders to obtain required resources. Responsibilities include external and internal activities. Project manager plans and monitors the flow of the project *Developer* Responsible for project design and development activities according to user specifications. He produces clean, efficient code based on specifications. He tests programs to ensure that they meetthe requirements of the specification and that they are error-free. He is responsible for troubleshooting, debugging and upgrading of existing software if necessary. Also, he creates

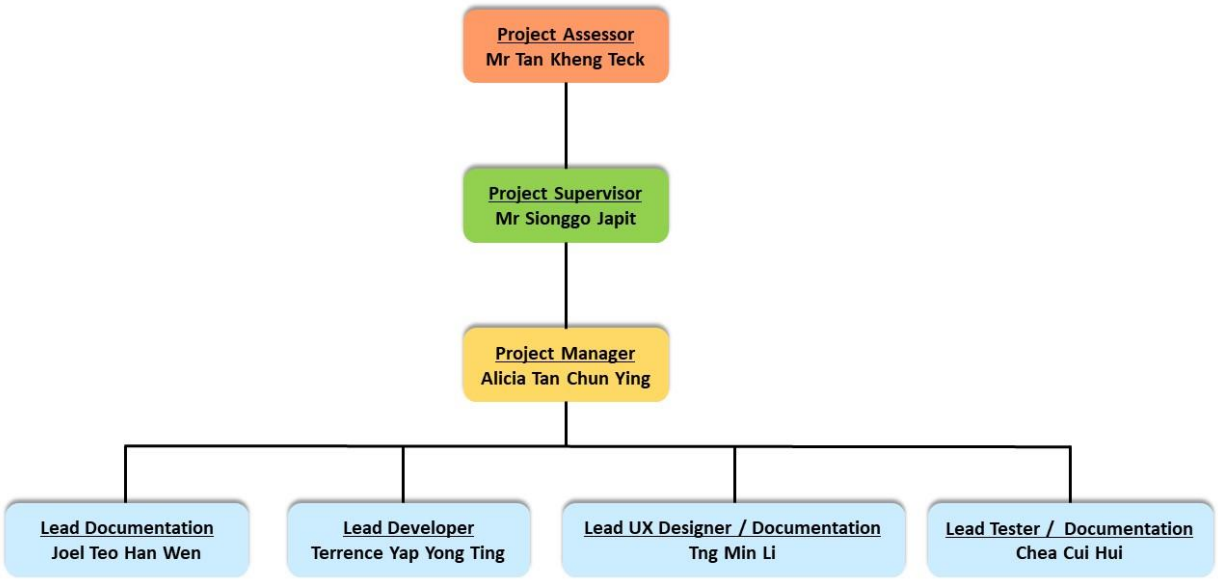
		technical documentation for reference and reporting.
Yap Yong Ting Terrence	Lead Developer	Responsible for project design and development activities according to user specifications. He produces clean, efficient code based on specifications. He tests programs to ensure that they meet the requirements of the specification and that they are error-free. He is responsible for troubleshooting, debugging and upgrading of existing software if necessary. Also, he creates technical documentation for reference and reporting. As lead developer, he is accountable for the project members involved in programming (developers and testers), keeping them on track.
Joel Teo Han Wen	Lead Documentation	Responsible for overseeing the meeting minutes. These minutes are required for a formal account of the members who attended the meeting and the items discussed, as well as the actions agreed upon and to be carried out. He should plan, organize and update the relevant documents and reports As lead documentation, he is accountable for the project members involved in documentation, keeping them on track.
Tng Min Li	Lead UX Designer / Documentation	Responsible for overseeing the meeting minutes. These minutes are required for a formal account of the members who attended the meeting and the items discussed, as well as the actions agreed upon and to be carried out. She should plan, organize and update the relevant documents and reports. As lead UX designer, she is also responsible for the diagrams for this project (eg: use case diagrams).

Chea Cui Hui	Lead tester /Documentation	Lead tester is responsible for reviewing software requirements and preparing test scenarios as well as executing tests on software usability. She prepares reports on all aspects related to the software testing carried out and reports to the design/developer team. She is required to interact with the team to understand product requirements. She participates in design reviews, providing input on requirements, product design and potential problems.

External Stakeholders

Name	Position	Job Description
Japit Sionggo	Project Supervisor	Project supervisor will be playing the role of a facilitator for the project team. He may provide some input to the implementation of the project. At the end of the day, the project supervisor and assessor will be the people assessing the project.
Tan Kheng Teck	Project Assessor	Project assessor will be playing the role of a facilitator for the project team. He may provide some input to the implementation of the project. At the end of the day, the project supervisor and assessor will be the people assessing the project.

Project Team Organizational Chart



3 Feasibility Studies

Literature Review

3.1 What is AS?

Authentication systems (AS) are known as the first and last line of defense for any system. Most of the time, AS can prevent unauthorized users from accessing confidential data (Barkadehi et al., 2018, p. 1502).

3.2 What types of AS are available?

As of now, there are 4 types of Authentication System (AS) models available – Ownership, knowledge-based, inherent-based and mix models (see table 1 below).

Ownership model	Knowledge-based model	Inherent-based model	Mix models
1. Physical keys	1. Passwords	1. Fingerprints	1. Two factor
2. Smart card	2. PIN code	2. Palm	2. Multi-factor
3. NFC	3. Lock pattern	3. Iris	
4. RFID	4. Graphical password	4. Voices	
5. Hardware-token	5. Rhyme based	5. Gestures	
6. Cell-phone	6. Challenge response	6. Face	

Source: Barkadehi, N. (2018)

3.3 What is a Typing Habit Gesture Authentication System?

This is a type of behavioral inherent-based authentication system implemented by recording patterns in a person’s typing habit.

There are 2 types of typing habits: fixed and free text (Identifying Users Using Keystroke Dynamics and Contextual Information, 2018)

Fixed

The system prompts the user to repeatedly type the same predefined text several times. The predefined text is always the same. For example, the text could be a password, or a string consisting of your information such as: name, surname, login or password.

Free

The system will prompt the user to type long portions of text that mimics the concept of typing whatever they want without limitations. The system’s algorithm makes use of this input to filter the necessary features and build a unique model for the user. The algorithm should determine the validity of the samples.

In this project, the team be focusing more on typing habits based on fixed texts on a regular computer (QWERTY) keyboard.

3.4 What do you understand about typing gesture habits?

Typing gesture habits have an alternative name which is also called as keystroke dynamics. Unlike other forms of biometric such as fingerprint scanning, iris scanning, and facial recognition, keystroke dynamics don't require active input. Instead, keystroke dynamics analyzes user typing patterns, which may include typing rhythms, frequent mistakes, which shift keys used for capitalization and pace.

It creates a baseline for user typing and then uses the baseline to check for abnormalities. If an abnormality is detected, a different authentication factor may be requested to verify the user or to terminate the session immediately, depending on the security policy.

Keystroke dynamics fall under the category of “behavioral biometrics;” these use the behaviors of the users as an authentication factor. As such, hackers cannot “steal them” because they are integral to the personality of the users; they also cannot replicate them for the same reasons. So, instead of considering what words users' type, how they type becomes of special interest (Canner, 2020).

3.5 How are typing gesture habits be used for authentication?

According to Fabian Monroe and Aviel Rubin, proceedings of the 4th ACM conference on Computer and communications security:

When a person types, the latency between successive keystrokes, keystroke durations, finger placement and applied pressure on the keys can be used to construct a unique signature for that individual. For well known, regularly typed strings, such signatures can be quite consistent. Furthermore, keystroke dynamics are not intrusive, making it very applicable to computer access security as the users will be typing on the keyboard anyway.

Unlike conventional passwords and PINs, behavioral typing gestures cannot be stolen or lost, and implementing it can serve as a safe and cheap way as it only requires a keyboard to achieve authentication.

3.6 Parameters of gesture typing

3.6.1 Text entry speed: Words per minute

Things to consider:

- When did the timing for the phrase begin?
- When did it end?

- Did timing begin with the first character or with a START button or some other signal before the first character?
- Did timing end with the last character or by pressing the ENTER after the last character?
- If timing begins with the first character and ends with the last character, then, arguably, the first character should not count, since timing excludes the time leading up to the entry of the first character.

3.6.2 Accuracy / Error rate: ration of incorrect characters to total characters

Insertion errors & deletion errors may lead to accuracy problems and hence, might affect the recognition of the user.

3.6.3 Dwelling time: text entry by eye typing

Things to consider:

1. A study carried out in 1996 reported that “for people with severe disabilities it can take anywhere from 15 minutes to many months to acquire eye control skill to run the system.” (Tecce, 1998, p. 320). The speed of real experts has not been systematically measured for any of the eye-controlled text entry systems.

2. As another example, consider (Wigdor & Balakrishnan, 2005, p. 212) TiltText, a technique for mobile phone text entry that uses the orientation of the device to resolve the ambiguity of letters on keys on the mobile phone keypad. In addition to conventional error rate analyses, they defined and used “button error” and “tilt error” as dependent variables. Button errors were the ratio of errors due to pressing the wrong button, and tilt errors were the ratio of errors due to tilting the device in the wrong direction.

PiePad is a gesture-based entry method for numeric entry (Quinn & Zhai, 2016, p. 235; (MacKenzie & Tanaka-Ishii, 2007, pp. 1–3). Gestures were stylus strokes conforming to a clock metaphor: right for 3, down for 6, left for 9, and so on. In addition to analyzing the time to make gestures, they defined and used "preparation time" and "scripting time" as dependent variables. Preparation time was the time between gestures, from stylus up after the previous character to stylus down for the current character. Scripting time was the gesturing time, from stylus down to stylus up.

3. Experiments have been performed with text entry by gaze using a common, "off-the-shelf" video camera with the GazeTalk system (Itoh et al., 2006, p. 65). They observed text entry speeds of 3–5 WPM by untrained users using large (3 × 4) on-screen buttons. It is not surprising that systems requiring hierarchical navigation or multiple gestures are slower than on-screen Qwerty keyboards with accurate eye trackers, although the prediction and completion features can improve the text entry speed. Their use depends heavily on both individual styles and extended use, so they are difficult to evaluate without longitudinal studies.

4. The language of the text also influences text entry speed. All the figures given so far are for English. GazeTalk and Dasher also support entering text in Japanese. A study with the results on the typing speed of 22–24 Kanji characters per minute, with performance improving from 19 to 23–25 characters per minute over seven short trials over 3 days was also tested (Itoh et al., 2006, p. 65). Both systems reached these text entry rates.

3.7 What is AI?

Artificial Intelligence (AI) refers to the simulation of human intelligence in machines programmed to think like humans and to imitate their actions. The term can also be applied to any machine that exhibits features associated with the human mind, such as learning and problem-solving.

The ideal characteristic of artificial intelligence is its ability to rationalize and take action, that has the best chance of achieving a specific goal. A subset of artificial intelligence is machine learning, which refers to the concept that computer programs can automatically learn from and adapt to new data without the assistance of humans. Deep learning techniques enable automatic learning through the absorption of vast amounts of unstructured data, such as text, images or video (Frankenfield, 2021).

An example of this is that Keystrokes generated maliciously that do not normally match human typing and can be easily detected. Using artificial intelligence, however, the Malboard attack independently generates user-style commands, injects keystrokes as malicious software into the keyboard, and avoids detection. The keyboards used in the research were the products of Microsoft, Lenovo and Dell.

"Our proposed detection modules are reliable and secure, based on information that can be measured from side-channel resources, in addition to data transmission," says BGU student Nitzan Farhi.

“These include (1) the keyboard’s power consumption; (2) the keystrokes’ sound; and (3) the user’s behavior associated with his or her ability to respond to typographical errors.”

Dr. Nissim adds, “Each of the proposed detection modules is capable of detecting the Malboard attack in 100% of the cases, with no misses and no false positives. Using them together as an ensemble detection framework will assure that an organization is immune to the Malboard attack as well as other keystroke attacks.” (Security, 2020).

3.8 What is machine learning?

Machine learning involves using algorithms to obtain statistical information on massive amounts of data. Data can come in many different forms such as numbers, words or

even pictures. The uses of machine learning could be used to gain an advantage in marketing or show recommendations on sites such as Google.

2 common methods of typing gesture authentication are fixed text or free text. Fixed text being a set paragraph that the user must type and is predetermined by the system, or free text that is randomly generated by the system so each time would give different words. A machine learning algorithm requires training, by using supervised or unsupervised learning using new samples, before being used (Azevedo et al., 2007b, p. 69).

Supervised learning:

Datasets with a known result are used to train a model. The algorithm learns and adapts from the data provided to give results consistent with the results provided by the inputs. When an unknown sample is required to be evaluated, the algorithm can draw on what it has previously learned to provide and answer. The greater the number of samples used to train the model, the better it performs. Common methods of supervised learning algorithms include Support Vector Machine, Neural Networks and k-nearest neighbors.
Unsupervised learning:

This method does not know which samples to learn from as all data is unlabeled. The algorithm is learned as samples are fed into it. A common method of unsupervised learning algorithms are Bayesian classifiers and clustering algorithms. The result can be ambiguous if the initial samples are misleading.

3.9 Software development methodologies, programming languages and implementation

We plan to implement our project with the SCRUM methodology. With regular discussions with stakeholders and using user stories, use case descriptions, class diagram, BCE diagrams, sequence diagrams to assist us in creating the framework of the whole program. With the use of Java programming language, we will be using GUI

to create an interface for the user to input their data and using the concept of Object Orientated Programming to build our program.

3.10 Algorithm and implementation

“Keystroke dynamics is the detailed timing information that describes exactly when each key was pressed and when it was released when concrete person is typing at a keyboard of a computer, gadget etc.” (Kochegurova et al., 2017, p. 12073). In keystroke dynamics, features such as dwell time, intervals between key presses and overlapping of key presses are factors used to capture data. Dwell time is the time when a key is in a pressed state. Intervals between key presses is the period between two keys being pressed. Overlapping of keypresses is when 1 key is not released before another key is pressed, the increase of speed will cause a greater number of keys being overlapped.

How we apply it to our program is by having the user store their keystroke dynamics characteristics into a profile first. For the algorithm, we will be using Euclidean distance to use as comparison between the profile, stored with the user’s keystroke dynamics characteristics and the current user’s keystroke characteristics for authentication.

Euclidean distance can be calculated by:

$$P = \sqrt{\sum_{i=1}^{N} (t_{et} - t_{cur})^2}$$

where N – the amount of different characters,

t_{et} – the standard dwell time for a key;

t_{cur} – the current dwell time for this key.

Source: (Kochegurova et al., 2017, p. 12073)

3.11 Recent developments in the area and case studies

3.11.1 Bayesian regularized neural network

The authentication system was trained using fixed text to determine the user.

Method:

Their algorithm makes use of Bayesian regularized neural network trained using users' typing habits like *dwell time*, *flight time*, as well as total amount of time used to type password to record continuous keystroke dynamics data in different sessions on separate days. Dwell time refers to the time the user spent pressing the key, while flight time refers to the interval or pause when the previous key is up, and the next key is pressed.

Operating environment:

This is a Java API-based application to be run on the computer on a basic keyboard. Upon startup, a screen is displayed where the user is asked to enter the username, session ID and their password. When users submit, the keystrokes are saved in the database. User is authenticated by their unique username and keystroke dynamics logged from sample text entered by that user (Zareen et al., 2018, p. 65).

3.11.2 Typing habit authentication system on android phone

Method:

The user is prompted to set their password and repeat it thirty times for the system to learn their typing pattern. The data provided by the user is fed into the system's artificial neural network. Then, a weight function is applied on the network. Support Vector Machine (SVM) and K-Mean strategy is adopted to let the system learn to recognize users better. This authentication system makes use of three separate SQLite databases for these purposes: training values, saving password and storing trained weights from the artificial neural network (Mishra et al., 2018).

Operating environment:

This authentication system runs on android 5.0 (or later) consisting of components such as artificial neural network to keystroke values and authenticate user. The keystroke dynamics recorded are based on PIN and patterned passwords set by the user. The front-end user interface was written in JAVA and XML, while the database connected to the program was SQLite

3.11.3 Typing habit Authentication for banking transaction system

Method:

Registration

Users are prompted to register their actions (performing a banking transaction). During registration, user is prompted to input their password 10 times so that the keystroke dynamics are analyzed, and a threshold of the user's typing pattern is determined.

Login

Users is prompted to complete their login process. Three tries will be given should they fail to log in. After account verification, the system authenticates the user by their keystroke habits. If the variance from the recorded typing pattern is little to none, full access will be given. Additionally, if variance is more than that but still within the threshold, partial access will be given to the user. Lastly, is the variance recorded is past the threshold, no access will be granted.

Operating Environment:

The authentication system is a JAVA application is run on windows XP (or higher) with an SQL database (Chourasia, 2014).

Preliminary Project Risk Analysis & Assessment

Risk Register 1

15 Jan 2021

Project Name: Behavioural Password Protection

ID No.	Rank	Risk	Category	Potential Responses	Risk Owner	Probability	Impact	Residual Risk
1	1	Unable to get the message across. Risk of miscommunication.	People	project manager to intervene and understand source of miscommunication	Alicia	Low	High - Dispute results in zero progress	Low
2	2	Inability to achieve the programming requirement.	Technical	do proper research and seek advice	Terrence	Low	High - Unable to deliver project specifications	Low
3	5	Unable to maximize individual's roles for the project. Unable to delegate roles to specific individual.	People	project manager to step in and know what specialties each individual has	Alicia	Low	Medium - Requires more communication and impede progress	Low
4	4	Inability to achieve users' expectation	People	Gathering of users' feedback and suggestions to make corrections and improve our product.	Joel	Low	Medium - Readdressing the requirements will result in time taken.	Low
5	3	Change of project scope.	People	Team members to discuss on the changes required for the prototype programme.	Joel	Medium	Medium - Need to add in and change some of the information and the programme.	Low
6	6	Programmer's computer break down during development	Technical	There should also be an online copy of the program on the internet / Github where the developers can retrieve from.	Terrence /Alicia/Cui hui	Low	High - Unable to deliver project specifications	Low
7	7	Programmer's computer break down during presentation	Technical	Make sure that there are at least 2 computers that has the software pre-loaded into their computers. There should also be an online copy on the internet / Github and a hard copy in a external hard disk.	Terrence /Alicia/Cui hui	Low	High - Unable to present our code, hence signication deduction of marks	Low

Figure xx: A screenshot of the project risk register

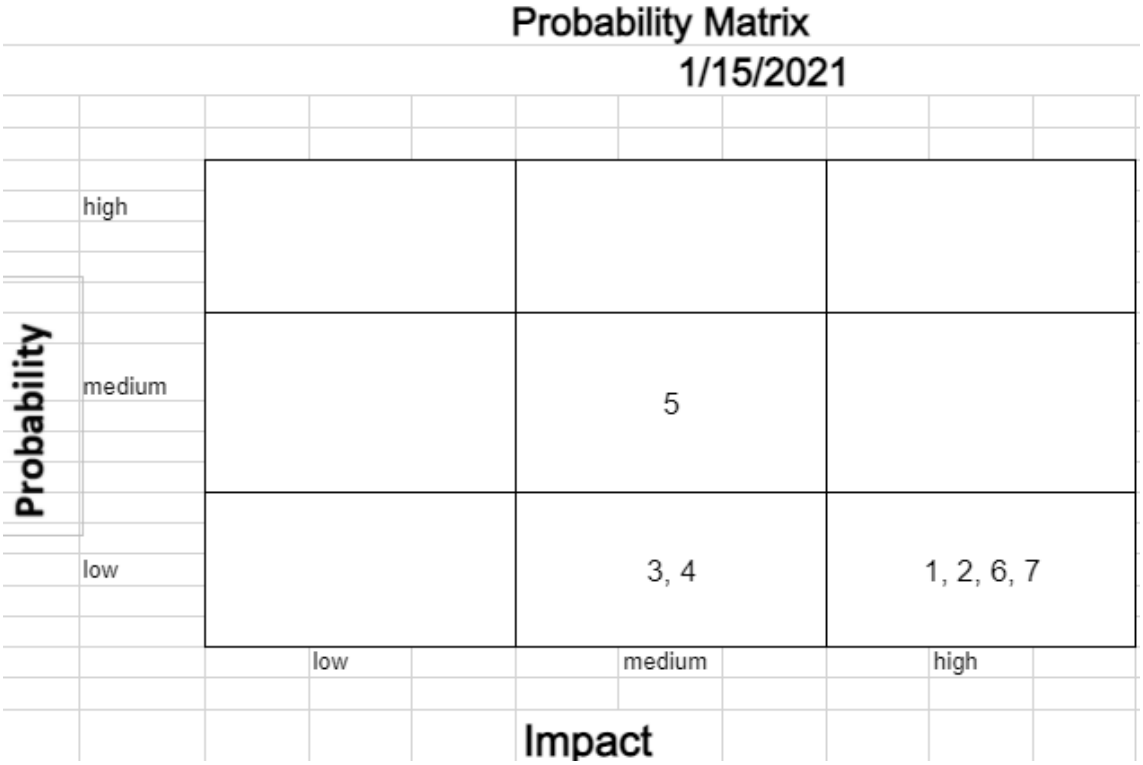


Figure xx: A screenshot of the project probability matrix

Proposed duration and schedule

Table xx: Brief overview of project schedule

Project Item	Dates
Project Website	Start: 15 Jan End: 21 Jan, 8pm
Literature Review	Start: 15 Jan End: 25 Jan, 10pm
Project specifications report	Start: 1 Feb End: 5 Feb, 8pm
Progress report	Start: 15 Feb End: 19 March, 8pm
Prototype demonstration	27 March
Coding	Start: 15 March Mid (ready to start user testing phase): 15 May End: 22 May, 10pm
Final Presentation	29 May

Work Breakdown Schedule

Table xx: Table of work breakdown schedule

Project Title: Typing Habit Gesture Authentication System
Date: 15 January 2021 **Prepared by:** Alicia Tan

WBS Categories

1.Initiating

1.1 Hold project kick off meeting
1.2 Create project charter
1.3 Project scope statement
1.4 Project Description
1.5 Team charter
1.6 Team member profiles

2. Planning

2.1a Work breakdown Structure
2.1b Project Schedule
2.1c Roles & responsibilities
2.2 Research/Literature review
2.3 FURP functional and non-functional requirements
2.4 User stories
2.5 Hold project meeting minutes (1)
2.6 Use case
2.7 Use Case Description
2.8 Milestone meeting
2.9 Project methodology
2.10 Hold project meeting minutes (2)
2.11 Prepare stakeholder register
2.12 Create project communications management
2.13 Create project risk register
2.14 Probability Impact Matrix
2.15 System Architecture
2.16 Operating Environment
2.17 Hold project meeting minutes (3)
2.18 Project Requirements Document Submissions
2.19 Sequence diagram
2.20 State diagram
2.21 Class diagram
2.22 Activity diagram
2.23 Hold project meeting minutes (4)

3. Executing

3.1 Sprint 1
3.2 Hold project meeting minutes (5)
3.3 Sprint 2
3.4 Project Progress Documentation
3.5 Hold project meeting minutes (6)
3.6 Sprint 3

3.7 Prototype System design and development 3.8 Document application development 3.9 Hold project meeting minutes (7) 3.10 Sprint 4 3.11 Test and integrate system components 3.12 Review and evaluate the system 3.13 Hold project meeting minutes (8) 3.14a Prototype presentation 3.14b Submit Project Progress Documentation 3.15 Hold project meeting minutes (9) 3.16 Final System design and development 3.17 Hold project meeting (10) 3.18 Make any final tweaks to system 3.19 Test and integrate system components 3.20 Hold project meeting (11) 3.21 Review and evaluate the system 3.21 Finalize testing and deployment
4. Closing
4.1 Review inputs, outputs and procedures 4.2 Hold project meeting (12) 4.3 Final Project Presentation 4.4 Submission of final product documentation and weekly diaries 4.5 Project closing and completed by date

Detailed Overview of the Project Schedule

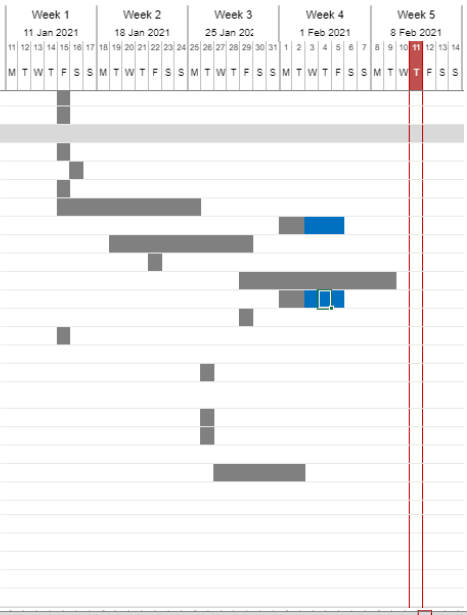
FYP Typing Habits Gesture Authentication System

FYP-21-S1-02

Project Start Date: 1/15/2021 (Friday) | Display Week: 1

Project Lead: Alicia Tan

WBS	TASK	LEAD	PREDECESSOR	PLANNED START	PLANNED END	ACTUAL START	ACTUAL END	DAYS	% DONE
1.5	Team charter	Alicia	-	Fri 1/15/21	Fri 1/15/21	Fri 1/15/21	Fri 1/15/21	1	100%
1.6	Team member profiles	Alicia	-	Fri 1/15/21	Fri 1/15/21	Fri 1/15/21	Fri 1/15/21	1	100%
2	**Planning**								
2.1a	Work breakdown Structure	Alicia	-	Fri 1/15/21	Fri 1/15/21	Fri 1/15/21	Fri 1/15/21	1	100%
2.1b	Project Schedule	Alicia	2.1a	Sat 1/16/21	Sat 1/16/21	Sat 1/16/21	Sat 1/16/21	1	100%
2.1c	Roles & responsibilities	Alicia	-	Fri 1/15/21	Fri 1/15/21	Fri 1/15/21	Fri 1/15/21	1	100%
2.2	Research/Literature review	Alicia	1.1	Fri 1/15/21	Mon 1/25/21	Fri 1/15/21	Mon 1/25/21	7	100%
2.3	FURP functional and non-functional requirements	Alicia	2.2	Mon 2/01/21	Fri 2/05/21	Mon 2/01/21	Fri 2/05/21	4	50%
2.4	User stories	Joel	-	Tue 1/19/21	Fri 1/29/21	Tue 1/19/21	Fri 1/29/21	11	100%
2.5	Hold project meeting minutes (1)	Alicia	1.1	Fri 1/22/21	Fri 1/22/21	Fri 1/22/21	Fri 1/22/21	1	100%
2.6	Use case diagram	Cui Hui	2.4	Fri 1/29/21	Fri 2/05/21	Fri 1/29/21	Tue 2/09/21	6	100%
2.7	Use Case description	Min Li	2.4	Mon 2/01/21	Fri 2/05/21	Mon 2/01/21	Fri 2/05/21	5	50%
2.8	Milestone meeting	Alicia	2.7	Fri 1/29/21	Fri 1/29/21	Fri 1/29/21	Fri 1/29/21	1	100%
2.9	Project methodology	Alicia	-	Fri 1/15/21	Fri 1/15/21	Fri 1/15/21	Fri 1/15/21	1	100%
2.10	Hold project meeting minutes (2)	Alicia	2.5	Fri 2/05/21	Fri 2/05/21		-	1	0%
2.11	Prepare stakeholder register	Alicia	-	Tue 1/26/21	Thu 2/22/42	Tue 1/26/21	Tue 1/26/21	1	100%
2.12	Create project communications management	Terrence	-	Wed 1/20/21	Wed 1/25/21	Wed 1/20/21	Wed 1/25/21	5	100%
2.13	Create project risk register	Alicia	1	Tue 1/26/21	Tue 1/26/21	Tue 1/26/21	Tue 1/26/21	1	100%
2.14	Probability Impact Matrix	Alicia	2.13	Tue 1/26/21	Tue 1/26/21	Tue 1/26/21	Tue 1/26/21	1	100%
2.15	System Architecture	Terrence	2.2	Mon 2/01/21	Fri 2/05/21	Wed 1/27/21		5	0%
2.16	Operating Environment	Terrence	2.2	Mon 2/01/21	Fri 2/05/21	Wed 1/27/21	Tue 2/02/21	5	100%
2.17	Hold project meeting minutes (3)	Alicia	2.10	Fri 2/12/21	Fri 2/12/21		-	1	0%
2.18	Project Requirement Specifications Document Submissions	Alicia	-	Sun 2/14/21	Sun 2/14/21			1	0%
2.19	Sequence diagram	Terrence	2.18	Mon 2/15/21	Fri 3/19/21		-	3	0%
2.20	State diagram	Min Li	2.18	Mon 2/15/21	Fri 3/19/21		-	4	0%
2.21	Class diagram	Min Li	2.18	Mon 2/15/21	Fri 3/19/21		-	5	0%
2.22	Activity diagram	Min Li	2.18	Mon 2/15/21	Fri 3/19/21		-	5	0%
2.23	Hold project meeting minutes (4)	Alicia	2.17	Fri 2/19/21	Fri 2/19/21		-	1	0%
3	**Executing**								
3.1	Sprint 1	Alicia	2	Mon 2/15/21	Sun 2/21/21		-	7	0%
3.2	Hold project meeting minutes (5)	Alicia	2.23	Fri 2/26/21	Fri 2/26/21		-	1	0%
3.3	Sprint 2	Alicia	3.1	Mon 2/22/21	Sun 2/28/21		-	7	0%
3.4	Project Progress Documentation	Joel	2.18	Mon 2/15/21	Fri 3/19/21		-	6	0%
3.5	Hold project meeting minutes (6)	Alicia	3.2	Fri 3/05/21	Fri 3/05/21		-	1	0%
3.6	Sprint 3	Alicia	3.3	Mon 3/01/21	Sun 3/07/21		-	7	0%
3.7	Prototype System design and development	Terrence	3.1	Mon 2/15/21	Fri 3/19/21		-	33	0%
3.8	Document application development	Joel	3.1	Mon 2/15/21	Fri 3/19/21		-	33	0%
3.9	Hold project meeting minutes (7)	Alicia	3.5	Fri 3/12/21	Fri 3/12/21		-	1	0%
3.10	Sprint 4	Alicia	3.6	Mon 3/08/21	Sun 3/14/21		-	7	0%
3.11	Test and integrate system components	Cui Hui	3.10	Mon 2/15/21	Sun 3/14/21		-	4	0%
3.12	Review and evaluate the system	Cui Hui	3.11	Mon 2/15/21	Sun 3/14/21		-	3	0%
3.13	Hold project meeting minutes (8)	Alicia	3.9	Fri 3/19/21	Fri 3/19/21		-	1	0%
3.14a	Prototype presentation	Alicia	3.12	Sat 3/27/21	Sat 3/27/21		-	1	0%
3.14b	Submit Project Progress Documentation	Alicia	3.13	Sat 3/27/21	Sat 3/27/21		-	1	0%
3.15	Hold project meeting minutes (9)	Alicia	3.13	Fri 3/26/21	Fri 3/26/21		-	1	0%
3.16	Final System design and development	Terrence	3.14	Sat 3/27/21	Thu 4/01/21		-	6	0%
3.17	Hold project meeting (10)	Alicia	3.13	Fri 4/02/21	Fri 4/02/21		-	1	0%
3.18	Make any final tweaks to system	Terrence	3.16	Sat 3/27/21	Thu 4/01/21		-	6	0%
3.19	Test and integrate system components	Cui Hui	3.18	Mon 1/04/21	Sat 4/03/21		-	3	0%
3.20	Hold project meeting (11)	Alicia	3.17	Fri 4/09/21	Fri 4/09/21		-	6	0%
3.21	Review and evaluate the system	Terrence	3.19	Sat 4/03/21	Sat 4/03/21		-	1	0%
3.22	Finalize testing and deployment	Cui Hui	3.21	Thu 5/20/21	Sat 5/22/21		-	3	0%
4	**Closing**				-		-		
4.1	Review inputs, outputs and procedures	Alicia	3.21		-		-	1	0%
4.2	Hold project meeting (12)	Alicia	3.20	Sat 5/29/21	Sat 5/29/21			1	0%
4.3	Final Project Presentation	Alicia	4.1	Sat 5/29/21	Sat 5/29/21			1	0%
4.4	Submission of final product documentation and weekly diaries	Alicia	4.2	Sat 5/22/21	Sat 5/22/21			1	0%
4.5	Project closing and completed by date	Alicia	4.3	Sat 5/29/21	Sat 5/29/21			1	0%



4 System Description

Operating Environment

The operating environment for the authentication system is as follows:

- Operating system: Windows
- Platform: Java
- Database: MySQL

Preliminary User Stories

User

Essential

1) As a user, they should be able to get authenticated so that they can log in to the application.
2) As a user, they should be able to log out so that they can exit the application safely.
3) As a user, they should be able to view a sentence upon login so that they can be able to type and authenticate themselves.
4) As a user, they should be able to see the generated sentence so that they can type it for authentication.
5) As a user, they should be able to re-calibrate their typing habits so that they are able to pass the authentication system even if their typing habits change.
6) As a user, they should be able to register account so that they can log in to the system.

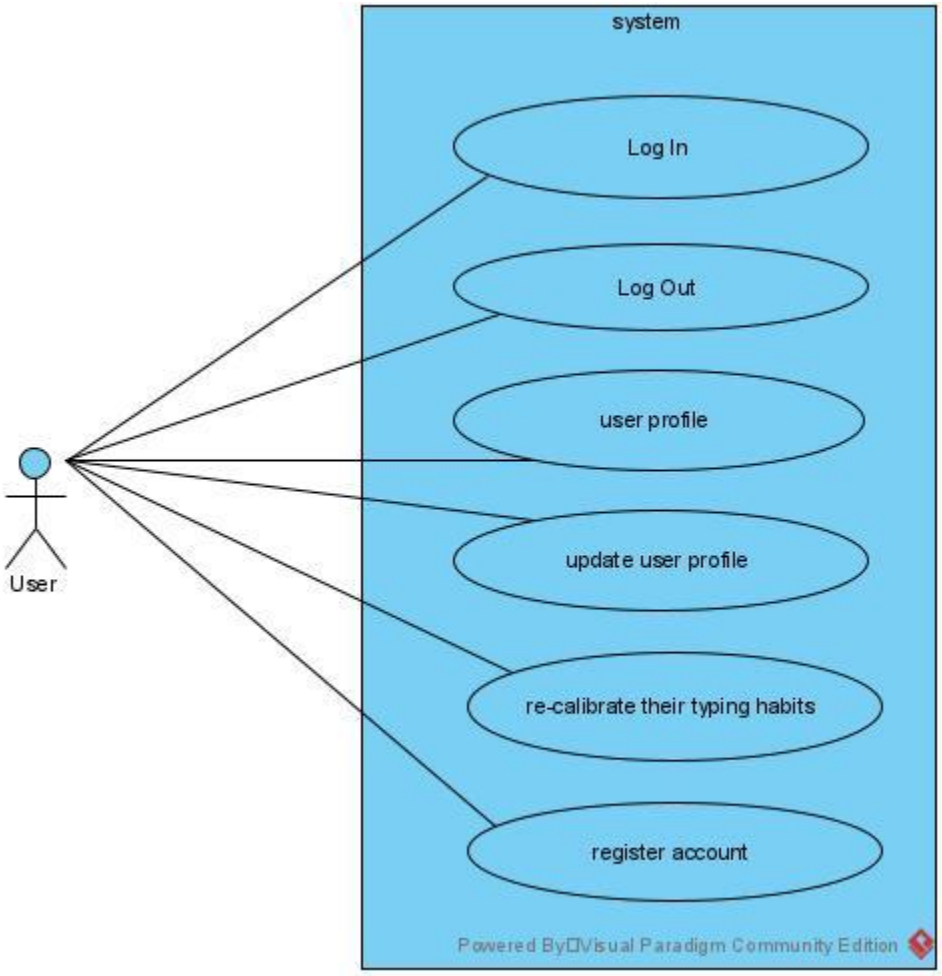
Additional

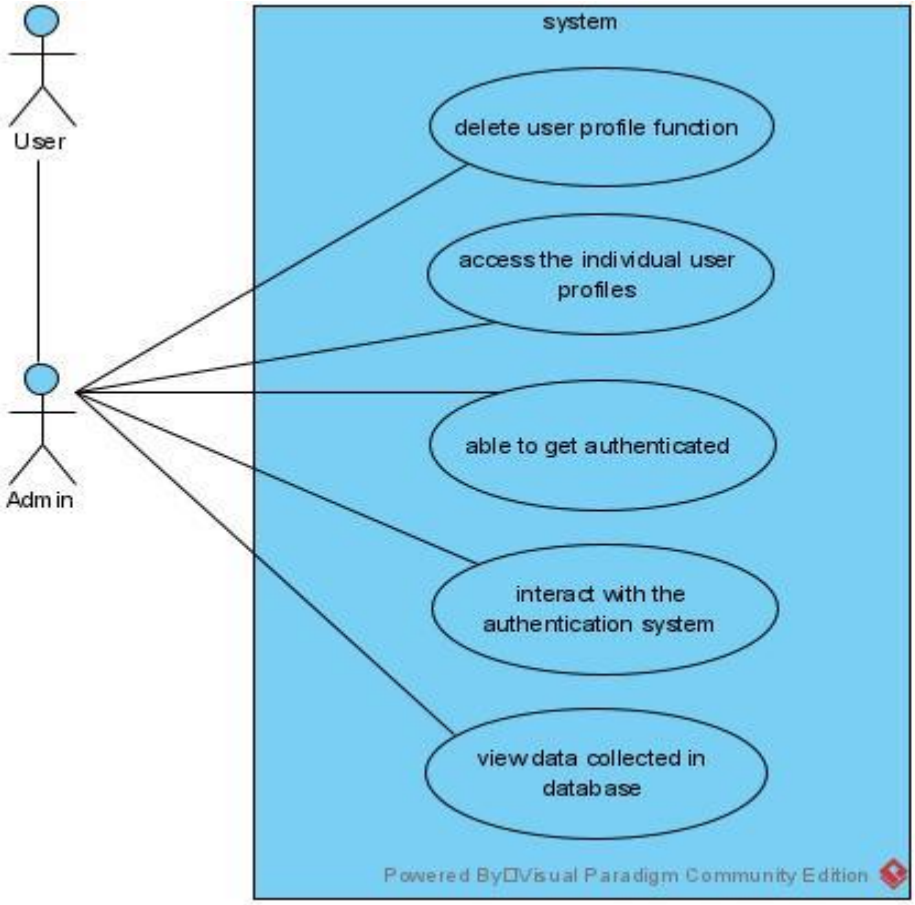
7) As a user, they should be able to authenticate themselves using a security question in case their typing habits change.

Admin

8) As an admin, they should be able to see all the user profiles so that they can manage the user profiles.
9) As an admin, they should be able to delete user profiles so that they can remove profiles of inactive users.
10) As an admin, they should be able to access the individual user profiles so that they can see user profile details.
11) As an admin, they should be able to log in as admin so that they can be distinguished as a different role.
12) As an admin, they should be able to log out so that they can exit the system.
13) As an admin, they should be able to get authenticated so that they can log in to the application.
14) As an admin, they should be able to interact with the authentication system so that they can perform maintenance on it.
15) As an admin they should be able to view data collected in database by authentication system so that they can analyze any anomalies.

Preliminary Use Case Diagram




system
User
Admin
delete user profile function
access the individual user profiles
able to get authenticated
interact with the authentication system
view data collected in database
Powered By Visual Paradigm Community Edition

Preliminary Use Case Description

5 User: User Story #1 (Inclusive of 3, 4, 11 & 13)

User Story #1: As a user, they should be able to get authenticated so that they can log in to the application.	
Name: Login	**Taiga ID:** 1
Stakeholders and Goals: User wants to login into the Typing Habit Gesture Authentication System.	
Descriptions: The user is able to login into the system with the credentials that they have registered with the System.	
Actors: User	
Trigger: 1. User clicks on the Login button on the Welcome page.	
Normal Flow: User clicks on the Login button on the Welcome page. The system retrieves and displays the Login page with a self-generated sentence. User enters their username and sentence generated by the System and clicks on the Login button to submit. The system verifies and validates login credentials with the database and redirects User to their respective home dashboard. End.	
Sub flows: 3a User clicks on the refresh icon beside the generated sentences. The system will re-generate a sentence and display it within the sentence text field. 3b User selects Password Login option. The system will retrieve and display the Login Password page. 3c User selects Back option. The system will display an error message and redirect the user back to the Welcome page.	
Alternate/Exception flows:	

2a The system is unable to retrieve the Login page.
The system will display an error message. The use case terminates.

2b The system is unable to display the Login page.
The system will display an error message. The use case terminates.

3a No username and/or sentence entered in the username and/or sentence input text field.

The system will prompt the User to re-input the missing text field at Step 3.

3b Invalid value in the username text field.

The system will display an error message and prompt User to re-input the username at Step 3.

4a The system is unable to find a matching username in the database with the username that User has entered.

The system will display a message indicating the user may not have registered before and prompt the User to register as a new user.

4b The system is able to find a matching username however the System is unable to match the sentence's typing biometrics against User's typing biometrics in the database.

The system will prompt the User to re-input the sentence at Step 3.

4bi The system is unable to match the sentence's typing biometrics against User's typing biometrics in the database more than 3 times to the same username.

The system will display an error message and prompt User to login using Password.

4c The system is unable to retrieve the respective home dashboard according to User's role.

The system will display an error message. The use case terminates.

4d The system is unable to display the respective home dashboard according to User's role. The system will display an error message. The use case terminates.

6 User: User Story #2 (Inclusive of User story #12)

User Story #2: As a user, they should be able to log out so that they can exit the application safely.	
Name: Logout	**Taiga ID:** 2
Stakeholders and Goals: User wants to logout from the Typing Habit Gesture Authentication System.	
Descriptions: The user is able to logout from the system.	
Actors: User	
Trigger: 1. User clicks on the Login button on their HomeDashboard page.	
Normal Flow: User clicks on the Logout button on their HomeDashboard page. The system retrieves and displays the Welcome page. End.	
Sub flows: None	
Alternate/Exception flows: 2a The system is unable to retrieve the welcome page. The system will display an error message. The use case terminates. 2b The system is unable to display the Welcome page. The system will display an error message. The use case terminates.	

7 User: User Story #5

User Story #5: As a user, they should be able to re-calibrate their typing habits so that they are able to pass the authentication system even if their typing habits change.	
Name: Calibration	**Taiga ID:** 5

Stakeholders and Goals: User wants to re-calibrate their typing biometrics.

Descriptions: The user is able to re-calibrate their typing biometrics with the System.

Actors: User

Trigger:

1. User clicks on the Calibration option on their HomeDashboard page.

Normal Flow:
User clicks on the Calibrate option on the HomeDashboard page.
The system retrieves and displays the Calibration page with a self-generated sentence.
User enters the sentence generated into the Input text field.
User clicks on the Submit button.
The system will verify and register the user's typing biometrics into the system.
The system and User will repeat Steps 2 through 5 for 10 times.
After the 10th time the System will re-calibrate using User's typing biometrics, update the database and display a success message.
User clicks Okay button.
The system redirects User to their HomeDashBoard page.
End.

Sub flows:

3a User selects Back option.

The system will display an error message and redirect user back to their HomeDashboards.

Alternate/Exception flows:

2a The system is unable to retrieve the Calibration page.
The system will display an error message. The use case terminates.

2b The system is unable to display the Calibration page.
The system will display an error message. The use case terminates.

3a No input of the sentence is entered in the sentence input text field.

The system will prompt the user to re-input the missing text field at step 3.

3b Invalid value in the sentence text field.

The system will display an error message and prompt user to re-input the username at step 3. 5a The system is unable to verify and register the user's typing biometrics in the database. The system will display an error message. User clicks 'Okay' button. The system goes back to step 2. 7a The system is unable to re-calibrate the User's typing biometrics and update it in the database. They System will display an error message. User clicks 'Okay' button. The system goes back to step 2. 9a The system is unable to retrieve the respective home dashboard according to user's role. The system will display an error message. The use case terminates. 4d The system is unable to display the respective home dashboard according to user's role. The system will display an error message. The use case terminates.

8 User: User Story #6

User Story #6: As a user, they should be able to register account so that they can log in to the system	
Name: Register	**Taiga ID:** 6
Stakeholders and Goals: User wants to register an account.	
Descriptions: The user is able to create an account.	
Actors: User	
Trigger: 1. User clicks on the Register option on the Welcome page.	

2. User has entered an invalid username on the login page and the system prompts user to register.

Normal Flow:

The system retrieves and displays the register page.
User enters first name, last name, username, email, password, confirm password, typing habit.
User clicks on the 'Register' button.
The system creates a new row in the database and displays a success message.
User clicks 'Okay' button.
The system retrieves the user Login page.
End.

Sub flows:

2a User selects Back option.

The system will display an error message and redirect the user back to the welcome page.

Alternate/Exception flows:

1a The system is unable to retrieve the Register page.
The system will display an error message. The use case terminates.

1b The system is unable to display the Register page.
The system will display an error message. The use case terminates.

2a No first name entered in the First Name text field.

The system will prompt User to re-input first name at Step 2.

2b Invalid value in the First Name text field.

The system will display an error message and prompt User to re-input first name at Step 3.

2c No last name entered in the Last Name text field.

The system will prompt User to re-input last name at Step 2.

2d Invalid value in the Last Name text field.

The system will display an error message and prompt User to re-input last name at Step 3.

2e No username entered in the Username text field.

The system will prompt User to re-input username at Step 2.

2f Invalid value in the Username text field.

The system will display an error message and prompt User to re-input username at Step 3.

2g No email entered in the Email text field.

The system will prompt User to re-input email at Step 2.

2h Invalid value in the Email text field.

The system will display an error message and prompt User to re-input email at Step 3.

2i No password entered in the Password text field.

The system will prompt User to re-input password at Step 2.

2j Invalid value in the Password text field.

The system will display an error message and prompt User to re-input password at Step 3.

2k No confirm password entered in the Confirm Password text field.

The system will prompt User to re-input confirm password at Step 2.

2l Invalid value in the Confirm Password text field.

The system will display an error message and prompt User to re-input confirm password at Step 3.

4a The system is unable to create a new profile in the database as there is an existing account. The system will display a message indicating that the user may have registered before and prompt User to Login.   6a The system is unable to retrieve the login page. The system will display an error message. The use case terminates. 6b The system is unable to display the login page. The system will display an error message. The use case terminates.

9 User – Additional: User Story # 7 (Changes)

User Story #7: As a user, they should be able to authenticate themselves using a password in case their typing habits change.	
Name: LoginPassword	**Taiga ID:** 7
Stakeholders and Goals: User wants to login into the Typing Habit Gesture Authentication System using password.	
Descriptions: The user is able to login into the system with the password credentials that they have registered with the System.	
Actors: User	
Trigger: 1. User clicks on the Login button on the Welcome page. 2. The system has successfully updated User's new password entered in ForgetPassword page.	
Normal Flow: User clicks on the Password Login button on the Login page. The system retrieves and displays the LoginPassword. User enters their username and password and clicks on the Login button to submit. The system verifies and validates login credentials with the database and redirects User to their respective home dashboard. End.	
Sub flows: 3a User selects Gesture Login option.	

The system will retrieve and display the login page.

3b User selects Back option.

The system will display an error message and redirects the user back to the welcome page.

Alternate/Exception flows:

2a The system is unable to retrieve the LoginPassword page.
The system will display an error message. The use case terminates.

2b The system is unable to display the LoginPassword page.
The system will display an error message. The use case terminates.

3a No username and/or password entered in the username and/or password text field.

The system will prompt the User to re-input the missing text field at Step 3.

3b Invalid value in the username and/or password text field.

The system will display an error message and prompt User to re-input the username and/or password at Step 3.

4a The system is unable to find a matching username in the database with the username that User has entered.

The system will display a message indicating the user may not have registered before and prompt the User to register as a new user.

4b The system is able to find a matching username however the System is unable to match the password enter by User against User's password in the database.

The system will prompt the User to re-input the sentence at Step 3.

4bi The system is unable to match the password against User's password in the database more than 3 times to the same username.

The system will display an error message and prompt User to reset password.

4c The system is unable to retrieve the respective home dashboard according to User's role. The system will display an error message. The use case terminates. 4d The system is unable to display the respective home dashboard according to User's role. The system will display an error message. The use case terminates.

10 Admin: User Story #8

User Story #8: As an admin, they should be able to see all the user profiles so that they can manage the user profiles.	
Name: ManageUser	**Taiga ID:** 8
Stakeholders and Goals: Admin wants to view and manage all user profiles within the system.	
Descriptions: The user is able to access all user profiles within the system.	
Actors: Admin	
Trigger: 1. Admin clicks on the Manage User button or icon in AdminHomeDashboard page.	
Normal Flow: Admin clicks on the Manage User button or icon in AdminHomeDashboard page. The system retrieves and displays the ManageUser page. End.	
Sub flows: None	
Alternate/Exception flows: 2a The system is unable to retrieve the ManageUser page. The system will display an error message. The use case terminates. 2b The system is unable to display the ManageUser page. The system will display an error message. The use case terminates.	

11 Admin: User Story #9

User Story #9: As an admin, they should be able to delete user profiles so that they can remove profiles of inactive users.	
Name: DeleteUser	**Taiga ID:** 9
Stakeholders and Goals: Admin wants to delete and remove user profiles of inactive users within the system.	
Descriptions: The admin is able to delete user profile within the system.	
Actors: Admin	
Trigger: 1. Admin clicks on the 'Remove User' button on ManageUserDetails page.	
Normal Flow: Admin clicks on the 'Remove User' button on ManageUserDetails page. The system displays a confirmation message. Admin clicks on the confirm option. The system retrieves and deletes the row containing the inactive user from the database and displays a success message. Admin clicks 'Okay' button. The system retrieves and displays the ManageUser page. End	
Sub flows: 3a Admin selects cancel option. The system redirects Admins back to the ManageUserDetails page.	
Alternate/Exception flows: 2a The system is unable to display the confirmation message. The system will display an error message. The use case terminates. 4a The system is unable to retrieve and delete the inactive user from the database. The system will display an error message. The use case terminates. 6a The system is unable to retrieve the ManageUser page. The system will display an error message. The use case terminates.	

6b The system is unable to display the ManageUser page.

The system will display an error message. The use case terminates.

12 Admin: User Story #10

User Story #10: As an admin, they should be able to access the individual user profiles so that they can see user profile details.	
Name: ManageUserDetails	**Taiga ID:** 10
Stakeholders and Goals: Admin wants to view individual user profile details within the system.	
Descriptions: The user is able to access individual user profile details within the system.	
Actors: Admin	
Trigger: 1. Admin selects individual users on the ManageUser page. 2. Admin selects Cancel option on confirmation message for Remove User.	
Normal Flow: Admin selects individual users on the ManageUser page. The system retrieves and displays the selected individual user details on the ManageUserDetails page. End.	
Sub flows: None	
Alternate/Exception flows: 2a The system is unable to retrieve the ManageUserDetails page. The system will display an error message. The use case terminates. 2b The system is unable to display the ManageUserDetails page. The system will display an error message. The use case terminates.	

13 Admin: User Story #14

User Story #14: As an admin, they should be able to interact with the authentication system so that they can perform maintenance on it.	
Name: AccessSystem	**Taiga ID:** 14
Stakeholders and Goals: Admin wants access to the Typing Habit Gesture Authentication System for maintenance.	
Descriptions: The admin is able to access the system for maintenence.	
Actors: Admin	
Trigger: 1. Admin clicks on the Access System button/icon in AdminHomeDashboard page.	
Normal Flow: Admin clicks on the Access System button/icon in AdminHomeDashboard page. The system retrieves and displays an AccessSystem page. End	
Sub flows: None	
Alternate/Exception flows: 2a The system is unable to retrieve the AccessSystem page. The system will display an error message. The use case terminates. 2b The system is unable to display the AccessSystem page. The system will display an error message. The use case terminates.	

14 Admin: User Story #15

User Story #15: As an admin, they should be able to view data collected in database by authentication system so that they can analyze any anomalies.	
Name: AccessDatabase	**Taiga ID:** 15
Stakeholders and Goals: Admin wants access to database used by the Typing Habit Gesture Authentication System to analyze for anomalies.	
Descriptions: The admin is able to access database for analyzing anomalies.	

Actors: Admin
Trigger: 1. Admin clicks on the Access Database button/icon in AdminHomeDashboard page.
Normal Flow: Admin clicks on the Access Database button/icon in AdminHomeDashboard page. The system retrieves and displays an AccessDatabase page. End
Sub flows: None
Alternate/Exception flows: 2a The system is unable to retrieve the AccessDatabase page. The system will display an error message. The use case terminates. 2b The system is unable to display the AccessDatabase page. The system will display an error message. The use case terminates.

15 System Requirements

Functional requirements

GUI Application

- Allows user to sign up for an account and register their typing habits
- Allows user to log in and authenticate themselves (at log in)
- Register user account based on 4 steps:
    - 1st: Enter username, name and email.
    - 2nd: Type sentence according to generated sentence.
    - 3rd: Typing habit is recorded.
    - 4th: Repeat 2nd and 3rd steps at least 5 – 10 times
- Authenticate user logins based on 3 steps:
    - 1st: Correct username
    - 2nd: Correct Input according to generated sentence
    - 3rd: User is authenticated if typing habit recorded is within the user's threshold.
- Based on roles:
    - Users:
        - Able to recalibrate their typing habits
        - Able to modify their email/personal details.
    - Admins:
        - Able to view all user profiles
        - Able to delete user profile
        - Able to view and edit data in database

Account creation and Management

- Application has admin users to manage the regular users.

Non-Functional Requirements

Usability

- GUI interface will be simple and straightforward with a few buttons.

Reliability

- The application should be always available whenever authentication is required.

Performance

- The authentication process should be as fast as possible while maintaining high accuracy in identifying anomalies.

Supportability

- All users should be able to run the application on their computer.
- The application is easy to test with various test cases.
- Application is only supported on computer.

Security

- Users should not be able to see each other's typing profile.
- Admin privileges should only be given to the user with the admin role.
- Users should not be able to log into another user's account.
- Users should only be able to log into their own account based on their typing habit.

Scalability

- Code for machine learning should be simple for future implementations of this authentication system.
- Machine should learn users' habits even during usage of program (eg: during login).
- Updates on the machine learning system (to improve the system / remove bugs from the system) should only be accessible by admin/programmers with the given permission.

External Interface Requirements

Software requirements

Our team have decided to implement the authentication system using the following software:

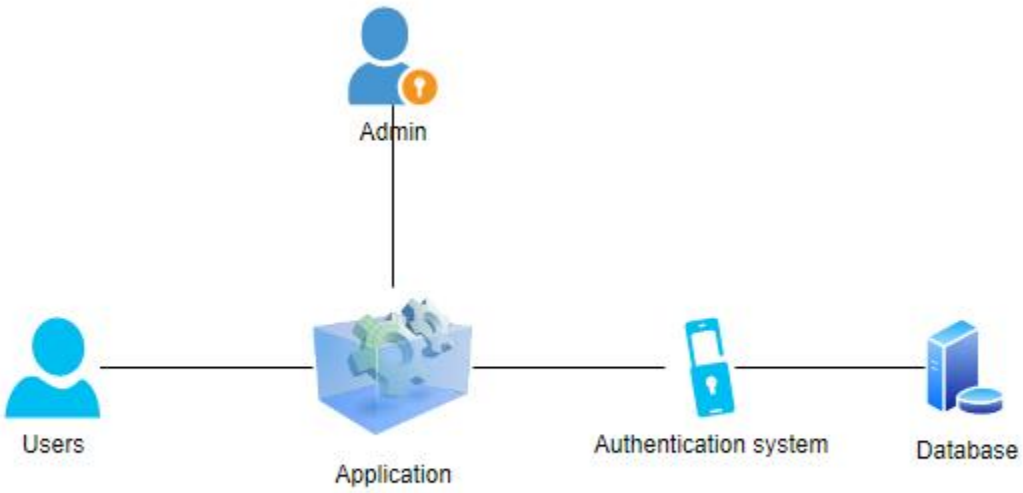
Table xx: Table depicting our software requirements

Software used	**Description**
Operating System	We have chosen **Windows** operating system for several reasons: most popular operating system, having the best support and user-friendliness. Windows version required: Windows 10 (.NET Framework 3.5) or later
Database	To save the user records, password records we have chosen an **SQL** database.
Java integrated development environments (IDEs)	We have chosen to implement the authentication system as a **JAVA-based GUI** using IDEs such as Eclipse and Netbeans. JAVA version requirement: 1.8.0_191 or later

Hardware requirements

Computer should have windows operating system. Internet connection is not necessary to show implementation of the authentication system.

System Architecture



16 References

1. Azevedo, G. L. F. B. G., Cavalcanti, G. D. C., & Filho, E. C. B. C. (2007b). Hybrid Solution for the Feature Selection in Personal Identification Problems through Keystroke Dynamics. 2007 International Joint Conference on Neural Networks, 67–72. https://doi.org/10.1109/ijcnn.2007.4371256.

2. Barkadehi, M. H., Nilashi, M., Ibrahim, O., Zakeri Fardi, A., & Samad, S. (2018). Authentication systems: A literature review and classification. Telematics and Informatics, 35(5), 1491–1511. https://doi.org/10.1016/j.tele.2018.03.018.

3. Canner, B. (2020, June 25). What are Keystroke Dynamics? How Can It Improve Your... Top Identity & Access Management Software, Vendors, Products, Solutions, & Services. https://solutionsreview.com/identity-management/what-are-keystroke-dynamics-how-can-it-improve-your-authentication/.

4. Chourasia, N. (2014, April 20). Authentication of the user by keystroke dynamics for banking transaction system. Https://Www.Digitalxplore.Org/. https://www.digitalxplore.org/up_proc/pdf/64-139807550041-45.pdf.

5. Frankenfield, J. (2021, January 6). How Artificial Intelligence Works. Investopedia. https://www.investopedia.com/terms/a/artificial-intelligence-ai.asp.

6. Identifying users using Keystroke Dynamics and contextual information (No. 3). (2018). https://www.researchgate.net/publication/323028812_Identifying_users_using_Keystroke_Dynamics_and_contextual_information.

7. Itoh, K., Aoki, H., & Hansen, J. P. (2006). A comparative usability study of two Japanese gaze typing systems. Proceedings of the 2006 Symposium on Eye Tracking Research & Applications - ETRA '06, 59–66. https://doi.org/10.1145/1117309.1117344.

8. Kochegurova, E. A., Gorokhova, E. S., & Mozgaleva, A. I. (2017). Development of the Keystroke Dynamics Recognition System. Journal of Physics: Conference Series, 803, 012073. https://doi.org/10.1088/1742-6596/803/1/012073.

9. MacKenzie, S. I., & Tanaka-Ishii, K. (2007). Text Entry Systems: Mobility, Accessibility, Universality (Morgan Kaufmann Series in Interactive Technologies) (1st ed.). Morgan Kaufmann.

10. Mishra, V., Gupta, R., Sood, G., & Patni, J. C. (2018). User Authentication using Keystroke Dynamics. Https://Research.Ijcaonline.Org/. https://research.ijcaonline.org/icrtstmsd2018/number1/icrtstmsd201806.pdf.

11. Quinn, P., & Zhai, S. (2016). Modeling Gesture-Typing Movements. Human–Computer Interaction, 33(3), 234–280. https://doi.org/10.1080/07370024.2016.1215922

12. Security, H. N. (2020, January 31). New user keystroke impersonation attack uses AI to evade detection. Help Net Security. https://www.helpnetsecurity.com/2019/06/10/user-keystroke-impersonation-attack/.

13. Tecce, J. (1998). Eye movement control of computer functions. International Journal of Psychophysiology, 29(3), 319–325. https://doi.org/10.1016/s0167-8760(98)00020-8.

14. Wigdor, D., & Balakrishnan, R. (2005). Empirical Investigation into the Effect of Orientation on Text Readability in Tabletop Displays. ECSCW 2005, 205–224. https://doi.org/10.1007/1-4020-4023-7_11.

15. Zareen, F. J., Matta, C., Arora, A., Singh, S., & Jabin, S. (2018). An authentication system using keystroke dynamics. International Journal of Biometrics, 10(1), 65. https://doi.org/10.1504/ijbm.2018.090129.