

## 5COSC021W Coursework 2 - GROUP template 2021\_22

- Use this template to structure the GROUP element of coursework 2. Ensure that the correct information is in each white box. The advice for each box is basic guidance to help you focus your answer.
- YOU MUST USE THIS TEMPLATE FOR THE GROUP WORK OF COURSEWORK 2.
- The current size of the boxes is not indicating how much you should write; change their size as you need.
- When you save the file, put your name and registration number in the file name, eg **'5COSC003W\_cwk2\_group\_Kelly\_Garret\_12345678.doc'**.
- A reminder of plagiarism: If you use bits of another's group report in yours or if you give your report to another group to use this will be an academic offence called 'collusion'.
- In order for the tutors to be able to assess your work you must ensure the following for your software submission:
  - Submit a zipped NetBeans project folder of the **COMPLETE** working project (i.e. the parts of each group member incorporated in one program, not just your part). If you have not been able to incorporate your part with that of the group, then submit only your part – it should be able to run though by itself.
  - Make sure that the submitted project will run using the software provided by the University. Contact your tutor if you have any problems with this.
  - Make sure that the project folder should contain all files necessary to run the program e.g. excel files etc.
  - Make sure that file I/O code does not use absolute file paths.
  - Make sure that the submission contains all usernames and passwords necessary to test the program.
  - Each group must upload a video describing the work in a google drive

Surname	Dhali
Forename	Monesha
Registration No:	w1719240

### By submitting this coursework you agree to the following:

I confirm that I understand what plagiarism is and have read and understood the section on Assessment Offences in the Essential Information for Students. The work that I have submitted is entirely my own. Any work from other authors is duly referenced and acknowledged.

I confirm

List here the team name and the other members of your group	Team name: PolyGlot Aisana Zharmagambetova Monesha Dhali Rahat Amit Haider Rafael Sally Zain Cheema Joshua Nwachukwu
---	--

## 1. Application Front End (HCI)– group part (10 marks)

(if you have not been able to incorporate your work in the group project do not fill in this section, instead fill in section 1a below)

**Guidance:** Attach here a screenshot of the front end of your application, incorporating the elements from each group member.

- Sum up all the main HCI issues of the application you have addressed and how they were addressed within the group, in particular as the individual elements came together. Include any HCI issues still remaining.
- Comment on how well all parts of the application are linked
- Marking of this section will also include the defence of your work during the demonstration

<b>Members of the group that worked in this:</b>	<b>Aisana - prepared images and set up icons, colour and font for menubar view and other pages</b> <b>Everyone else, used menubar view as example and fixed their pages accordingly (colour, font)</b> <b>Monesha - adjusted design by Aisana so it looks fine on other devices, and made coloured buttons design</b> <b>Aisana and Monesha collaborated to polish final design of the pages</b> <b>See Appendix 1 below for the Application pages screenshots</b>
--	--

**Issue:** Maintaining same font and components location with Frame size(with different device models):

different device resolutions would shift and reposition components of a frame and make it look slightly off. Even though it doesn't prevent users from using the application it still dissatisfying in the look

**Resolution:** We managed to resolve the font issue by choosing a font that existed on all devices: "Dialogue". For future improvement, it would be useful to have a method that would set up font and colours for everything when the program starts running, thus we would need to change font or colour just one time to change it everywhere.

**Further work:** The issue with various device screens we decided to leave as an opportunity for improvement in the future. Suggestion: make the application get the screen resolution of the currently used device and calculate the fitting size based on it.

**Issue:** Icons for menubar buttons design:

The menubar buttons design required buttons to have icons and change their size look when the button is clicked and disabled. The issue was that switching the size 60x60 image to 90x90 would require enlarging the button and adding a new image, which would make the button reposition its location, which would appear either on the screen or half-cut, especially with different screens resolutions.

**Resolution:** It was adjusted by Aisana so that Icon can enlarge on click and be disabled. The issue was that there was a need to see it works on other devices, from here Aisana passed work to Monesha to make sure the look of button icons is consistently fine among all devices. As a safety measure, Aisana set button size to largest 100x100 and thus switched icons from small to large wouldn't shift icons out of their locations. However, there still remained an issue with the proper Frame size that would fit everything on any device.

**Issue:** Locating the images for icons:

There was an issue that files I imported and tried to create images for couldn't be found with the class.getResource method and the file name I set.

**Resolution:** Resolved the issue by creating a resource file in the Source Package, which generated the other Sources package and allowed us to locate images and set up icons to buttons and labels (Rahat Amit helped Aisana to sort folder issues)

**Issue:** the design of displaying error messages

**Resolution:** Because we left design for the last part, the programming of the main code required to display error some way, thus team members used JOptionPane to display error. Due to time limits, we decided to leave this as an improvement to be done in the future

**Further Work:** Next improvement to the design would be to make error messages occur at the place suggested in the final interface design

**Issue:** colours choice:

It was needed to select an appropriate colour scheme that would go well without being a pain to watch. Plus the colour contrast needed to be evident for colourblind people to distinguish between buttons, and text.

**Resolution:** In the end, we decided to use blue and white as the main colour of the app. The red colour was used for the buttons to alert and bring the attention to the user on the action they are about to do. The colour scheme is made so that everyone can access including colourblind people, especially we added exclamation mark icons to point our error inputs as well as messages.

**Further work:** it may be good to add a solid visible border to the buttons we made red(significant) in order to make it visible as important to pay attention to.

## 2. Application Front End (SECURITY)– group part (10 marks)

### Guidance:

- Sum up all the main **security** issues of the application and how they were addressed and any security risks still remaining
- Marking of this section will also include the defence of your work during the demonstration

<b>Members of the group that worked in this:</b>	<b>Rahat Amit, Zain</b>
--	-------------------------

### By Rahat Amit

#### PASSWORD

- **Issue:** When a new or old user uses their credentials to log in into the app, this information gets stored into a database. However, information such as passwords can not be stored as plain text inside the database, as it will become vulnerable.
- **Resolution:**
  - To sort out this issue, an encryption method has been implemented into the application, the hash method. This technique consists of creating a salt value that will be added to the password, with an encryption key to make the hash generated hard to be decrypted. (**Mayoral F., 2013**)
  - On top of that, whenever a new user wants to register, they need to enter a strong password, and to tackle this, we implemented a password validation, where the password needs to have a certain length and some special characters in order to be accepted.

#### SQL INJECTION

- **Issue:**
  - The application doesn't have any sort of protection against these attacks, as it is still a prototype, even though there should be at least one method implemented into the application.
  - There are several types of SQLi attacks such as in-band SQLi, blind SQLi, and out-of-band SQLi (**Acunetix, 2022**).
  - In-band SQLi is one of the most common SQL Injection attacks, and this occurs when the hacker uses the same communication channel to attack and gather results. Meanwhile, with the blind SQLi, the attacker doesn't pass any values, but they try to reconstruct the database by looking at their behaviours.
  - Compared to the other methods, the out-of-band SQL Injection attack is not very common, because this occurs when the attacker isn't able to use the same channel to attack and gather results. This type of attack relies on the database server's ability to make a DNS or HTTP requests to deliver an attack. (**Acunetix, 2022**)
- **Resolution:**
  - As the development of the application goes on, more safeguards will be added against these attacks.
  - One way to prevent SQL Injection, is to use parameterized queries, through PreparedStatement, where the user inputted data will no longer be concatenated with the SQL query, but instead it will be substituted with a placeholder in the form of ?.
  - The first step is to create a PreparedStatement and then bind it with the values entered by the user, making it a safe way to run queries with the values passed by the user. (**Schults, 2022**)

**References:**

Mayoral, F 2013, Instant Java Password and Authentication Security, Packt Publishing, Limited, Olton. Available from: ProQuest Ebook Central. [4 May 2022].

Acunetix. 2022. *What is SQL Injection (SQLi) and How to Prevent Attacks*. [online] Available at: <<https://www.acunetix.com/websitesecurity/sql-injection/>> [Accessed 4 May 2022].

Acunetix. 2022. *Types of SQL Injection?*. [online] Available at: <<https://www.acunetix.com/websitesecurity/sql-injection2/>> [Accessed 4 May 2022].

Schults, C., 2022. *Preventing SQL injections in Java (and other vulnerabilities)* - Sqreen Blog. [online] Sqreen Blog. Available at: <<https://blog.sqreen.com/preventing-sql-injections-in-java-and-other-vulnerabilities/>> [Accessed 4 May 2022].

**By Zain**

### **Weak Backend**

**Issue:** Weak backend (Weak backend access controls) result from incorrect restrictions applied to an authenticated user. Threat actors can exploit these flaws to gain unauthorised access, which includes. User accounts, looking into sensitive files, manipulating other user data and accessing rights.

These types of issues can cause major security implications that can tarnish the reputation of the company and only hurt users of the application.

**Resolution:** To resolve such issues we would need to impose the correct restrictions on users. Such as, correct rulesets on what the user can and cannot do. Hide any backdoor access that are visible to the user (to not allow them to have access to any upper-level control). Differentiate what access a user should have on a page, and what access an admin level user should have.

Imposing these valid restrictions would prevent any of these security issues. Strengthening the backend only leads to a safer and more secure environment for the users.

If the strengthened restrictions were to be exploited. Then we would have to include these issues in an update.

#### **Examples of Weak backend systems:**

- *Injection, Broken Access Control and Insecure deserialization (Geekflare, 2022)*
- *Detailed look into Sql Injection (Medium, 2022)*

### **Logging**

Insufficient logging, monitoring and the lack of detecting the threats could allow malicious actors to tamper, extract, or destroy data, as well as further attack systems and pivot to more systems.

If actors gained access to the data by exploiting restrictions. If not logged correctly could lead to huge loss in data. A lack of logging and monitoring can cause any threats by actors to go unnoticed.

When eventually noticed, it would be too late to impose any type of restriction in stopping the loss of data from the threat actors.

To resolve such an issue is to make logging more sufficient. As well as have backups in case of these types of security threats. Having a proper set of logging and monitoring can prevent this issue much ahead of time. Or even allow another instance of the application to run over the currently threatened version. That then allows users to have access to a safe backup while the current is being squashed out.

#### **Examples of Logging infrastructure:**

- *Logging Framework (SentinelOne, 2022)*
- *Logging System ([eso.org](#), 2022)*
- *Safe Logging Practice ([paladion.net](#), 2022)*
- *Logging for Security Purpose (2022)*

**References:**

1. Geekflare. 2022. 6 Web Backend Security Risks to Consider in Development. [online] Available at: <https://geekflare.com/web-backend-security-risk>
2. Medium. 2022. SQL Injections In Ethical Hacking & Cybersecurity Part II. [online] Available at: <https://systemweakness.com/sql-injections-in-ethical-hacking-cybersecurity-part-ii-a9a572180ec0>
3. SentinelOne. 2022. What To Look for in a Logging Framework | Scalyr. [online] Available at: <https://www.sentinelone.com/blog/logging-framework/#:~:text=A%20logging%20framework%20is%20a,organizations%20also%20roll%20their%20own>.
4. Eso.org. 2022. 2.11 LOGGING SYSTEM. [online] Available at: <https://www.eso.org/projects/vlt/sw-dev/wwwdoc/JAN2006/VLT-MAN-ESO-17210-0619/Output/log.html#:~:text=The%20purpose%20of%20the%20logging,information%20related%20to%20operative%20conditions>.
5. Paladin.net. 2022. Application Logs: Security Best Practices. [online] Available at: <https://www.paladion.net/blogs/application-logs-security-best-practices#:~:text=Safe%20Practices%20in%20Logging&text=Secure%20the%20system%20on%20which,malfunctions%20or%20is%20shut%20down>.
6. 2022. [online] Available at: <https://www.ncsc.gov.uk/guidance/introduction-logging-security-purposes>

### 3. Professional conduct: Legal & Ethical (10 marks)

#### 3.1 Legal

Which group members worked on this: Monesha

**Guidance:** List here the legal issues that would affect both the development and the use of your application. You need to support this work with research. The marks in this section also include marks for references (see end of document).

##### Legal Issues:

There are many legal issues to consider which could affect the development and the use of the app. A list of those legal issues is included and explained below.

**Legal issue:** absence of a confidential agreement would result in not being able to protect ideas originated by a person. This would affect the development of the application as other people would be able to justifiably steal ideas from the application without getting fined. To reduce such effects, it is recommended to sign a non-disclosure agreement. A non-disclosure agreement is intellectual property rights that prevent business ideas/products from being copyrighted (Droids On Roids, 2021). This legal agreement would ensure that the business idea/application will be secure. This also helps the owner of the idea to take legal action against anyone leaking the idea or violating the terms of a non-disclosure agreement (Anon, 2018). Due to our application being a student project less rigorous means have been adopted as the project is not confidential. Students took precautions using their university id to take ownership of their creation.

Another way to protect **intellectual property from copycat is to get patent protection**. Patent protection means that the business idea cannot be commercially made, used, distributed, imported or sold by anyone without the consent of the owner of the patent ([www.wipo.int](http://www.wipo.int), n.d.). Such means should be taken whilst developing an application so that the owner can take any legal action if some circumstances do get arisen. The team did not take any action to get patent protection as the project is not confidential thereby does not need such a contract at this stage. However, measures have been taken to show ownership over the intellectual property by adding the member's id over any work they have contributed.

**Legal Issues: Independent Contractor Agreement** should be signed at the start of the development of the application to reduce related legal issues. This is an agreement that defines the responsibilities of all the parties involved whilst the creation of the application (Droids On Roids, 2021). This leads to a decrease in misunderstandings occurring throughout the whole development of the application. This is due to defining the responsibilities, rights and scope of work for all the people involved. This agreement is beneficial especially when an exit plan is drawn up. This is because throughout the development of the application the cooperation might cease before the planned duration due to unexpected variables (Droids On Roids, 2021). This agreement then would be beneficial as all the parties involved will be able to take effect the necessary steps without resorting to any conflicts related to funds or property rights. To resolve any such legal issues, a simple contract has been created and implemented at the start of the project. This contract stated all the responsibilities each member of the team has. Additional terms and conditions have also been added to tackle problems such as consequences to be followed if a member of the team breaks any terms of the contract.

There are **privacy issues** involved when an application collects the data of any children below the age of 13 (under American law). It is the Children Online Privacy Protection Rule which imposes certain requirements on an application that uses/collects personal info of children under the age of 13 (Federal Trade Commission, 2013). Our application audience is students above the age of 13 thus, this law does not apply to LangWise (our application). Therefore, we do not knowingly collect any data on children below the age of 13.

To consider the **legal aspects** of the use of the application it is recommended to use terms and conditions. Terms and Condition is used to protect the company and the application rights that were developed (Droids On Roids, 2021). It provides clarity over what actions and results will occur in any given situation. This is beneficial to the company as it offers clients and commercial partners an understanding of their rights, roles, boundaries and responsibilities (Droids On Roids, 2021). It clearly states the standing of all the parties, and clients involved when using or being related to the product in question. The use of such conditions allows the customer or any parties to abide by rules imposed by the company/ business owner. Thus, resulting in less conflict and legal issues the business idea encounters. To consider such a legal issue, we have decided to implement the terms and conditions on the application. The terms and conditions were generated from an online terms and conditions generator (Terms And Conditions Generator, n.d.).

**Legal issues such as user privacy** would affect both the development and the use of the application if it is not handled properly. Therefore, our application follows the Data Protection Act 2018 (which strongly relates to GDPR) legislation. It imposes requirements on those who collect any personal data (Gov.Uk, 2018). To process the personal data it is recommended to collect a limited amount of user data (only the necessary data) and to use the user data only for the required purpose which the user gave consent for (Droids On Roids, 2021).

In response to such an issue, we have designed to collect a decreased amount of information about the user. The data collected would not be used without the knowledge of the user. The data collected are mainly to serve the user and they are protected through technological means at our best effort.

## Professional conduct – Legal & Ethical

### 3.2 Ethical

Which group members worked on this:

Rafael  
Joshua

**Guidance:** With the aid of a table list here the ethical issues that would affect both the development and the use of your application. You need to support this work with research. The marks in this section also include marks for references (see end of document).

#### By Rafael

**Professional standards :** Working in the field of computer science and information technology frequently requires that people have industry skills and experience, most likely with official qualifications, and will continue to improve their abilities and knowledge, having a diverse set of industry abilities etc(BBC). While working on the assignment, it became clear that not all of us had the same set of skills in all areas, so it was decided that splitting the work according to each individual's strength and then putting everything together would be the best option to have a well-done project.

**Data protection legislation :** Personal information is confidential and should only be accessed by those with the appropriate permissions. : While developing the app an issue with allowing certain information to only be accessible by certain users for privacy purposes was encountered, and it was solved by giving the users (teacher and student) access to different information in the app.

As mentioned in the BBC, computer-stored digital files are simple to access, copy, and share. Personal data held by others must be kept private and not altered or deleted, which necessitates protection. As a result, the Data Protection Act of 1998/2018 was enacted to safeguard personal information(BBC).

**The Copyright :** When a person develops anything, they are the exclusive owner of it. Copyright gives the owner of the copyright the sole right to publish, copy, distribute, and sell their work. Without authorization, no one else can use the work(Peter Wayner, 2014). When you execute a programme on a computer, it's nearly difficult to avoid copying some of the code because the programme is usually copied automatically within the computer's memory in order for the software to work. Copyright is infringed not only by obtaining a direct duplicate of the original work, but also by altering versions of the original, which is unique to software. To avoid copyright infringement the project and most of the code in the project is our own not a duplicate of other work.

**By Joshua**

**Ethical Constraint: Bug fixes**

- It's unethical for one member to do their own thing completely ignoring the specifications other members have given and proceed to push their own changes through and call it a day. It's important that we follow what the specification says we do rather than what a member thinks the client would prefer.
- An ethical irresponsibility like a developer figuring out a software breaking bug but the key members responsible for the code decides it's not important now even though it will greatly affect user experience will be avoided if possible as bugs that are left alone could cause a serious problem to the entire program, fixing bugs should not only be important when the value of having them fixed exceeds the cost of fixing it (Spolsky, 2001).

**How it was solved/will be solved**

- In order to avoid encountering too many bugs and having to fix them all, we made sure to work in our intended environment, a mobile application size and width while also making sure there were no major issues that would affect user experience like text being too small or buttons not being easy to click.
- After our program has been submitted, having one or more members keep an eye out for bug reports would be ideal as there is no point in just ignoring a software we spent months building just for marks, especially when it's created for a real client.

**Ethical Constraint: Communication**

- There are a variety of communication services that allow us to communicate and share information quickly so there was no reason for one of us to publicly reveal information to the users that a bug exists which would be unethical and unprofessional.

**How it was solved/will be solved**

- The way the group was able to avoid this issue was to communicate via helpful applications, Discord and Github. The former allowed us to communicate and show our work as if we were physically present while the latter allowed us to share our code and work on the same program efficiently.

**Ethical Constraints: Misleading Users**

- In the program we created, we made sure that our program is easy to read for our users and ensured no part of our design tricks our users (Scacca, 2022).
- Each part, design and text in the program was simple, clear and concise for our user. We also made sure that when the user chooses a language they want to study, it is correct in both their language and the language they want to study.

**How it was solved/will be solved**

- We made sure we did not intentionally add different meanings to phrases in dialogue that would trick users into learning incorrect dialogue in the language they want to study.
- Our users must have control over the dialogue they want to learn and it is our job to make sure the application we are providing works correctly.

### **Ethical Constraint: Responsibility**

- Just like it's our job to make sure we create as few bugs as possible, it is also the user's responsibility to report the bugs they encounter to us. This would cause the bugs to be fixed as soon as possible which in turn makes the user experience much better rather than ignoring the bugs.
- However, ethical responsibility should extend to all areas of software development (Bittner and Hornecker, 2005) It is important that we follow regular business ethics and specific ethical standards for the software we are creating.
- Though some of us would have to make easy or sometimes hard decisions when it comes to bugs, for example, should someone work overnight trying to fix as much as they can, or should they choose which bug can they ignore, and which one is more damaging? (Wayner, 2014).
- No matter how difficult the decisions mentioned may be, each of us has a responsibility to make decisions as a team, not alone as each one of us have specialised knowledge that we might have, allowing us to run like a well-oiled machine to get our software up and running perfectly.

### **How it was solved/will be solved**

- We always made sure to follow professional ethics by being fair, patient and supportive to one another and always working in the best interest of our client by making sure our work is to their specifications and we plan to respond to any bugs should they arise.
- Every major decision was made via scheduled meetings on Discord and there are more experienced developers in our team that we were able to count on their help and a developer is able to help them out when they are struggling with their part of the code.
- The less experienced developers were able to count on the more experienced developers when they were struggling with their code. It was much easier to share knowledge than ignore the members in need.

## References (marks included in each of the main sections)

Sections 3 and 4 must be supported by research.

List below your sources, using Harvard referencing. Make sure that your references are referred to correctly from the relevant text of your work.

If you are not clear how to reference read:

<https://www.westminster.ac.uk/library-and-it/support-and-study-skills/guides-and-tutorials/referencing-your-work>

Here's how we'll assess it:

- No research sources: that's very bad for level 5 work
- There is one source with all information, copied directly as if it's your own text: that is plagiarism
- There is one source with all information, referenced and discussed: that is bad research
- There are a few different sources, referenced and discussed in the text: this is getting better
- There are quite a few good sources from many different places, referenced and discussed in the text: this gets good marks.

### Section 3.1 references

Anon, (2018). *Top 7 Legal Issues To Consider In Mobile App Development*. [online] Available at: [https://opengeekslab.com/blog/top-legal-issues-in-mobile-app-development/#4\\_Intellectual\\_Property\\_Rights\\_to\\_Prevent\\_Your\\_Product\\_from\\_Copycats](https://opengeekslab.com/blog/top-legal-issues-in-mobile-app-development/#4_Intellectual_Property_Rights_to_Prevent_Your_Product_from_Copycats) [Accessed 25 April 2022]

Droids On Roids. (2021). *5 Key Legal Issues to Consider in your Mobile App Development in 2021*. [online] Available at: <https://www.thedroidsonroids.com/blog/legal-issues-in-mobile-app-development>. [Accessed 25 April 2022]

Federal Trade Commission. (2013). *Children's Online Privacy Protection Rule ('COPPA')*. [online] Available at: <https://www.ftc.gov/legal-library/browse/rules/childrens-online-privacy-protection-rule-coppa>. [Accessed 25 April 2022]

Gov.Uk (2018). *Data Protection Act*. gov.uk. [online] Available at: <https://www.gov.uk/data-protection> [Accessed 25 April 2022]

www.wipo.int. (n.d.). *Patents*. [online] Available at: <https://www.wipo.int/patents/en/#:~:text=In%20principle%2C%20the%20patent%20owner> [Accessed 25 April 2022]

### Section 3.2 references

(Rafael's):

BBC : Ethical, legal and environmental impacts of technology. Available at : <https://www.bbc.co.uk/bitesize/guides/zd726yc/revision/1> (Accessed: 20/04/2022).

Peter Wayner. (2014) 12 ethical dilemmas gnawing at developers today. Available at : <https://www.infoworld.com/article/2607452/12-ethical-dilemmas-gnawing-at-developers-today.html> (Accessed: 20/04/2022).

(Joshua's):

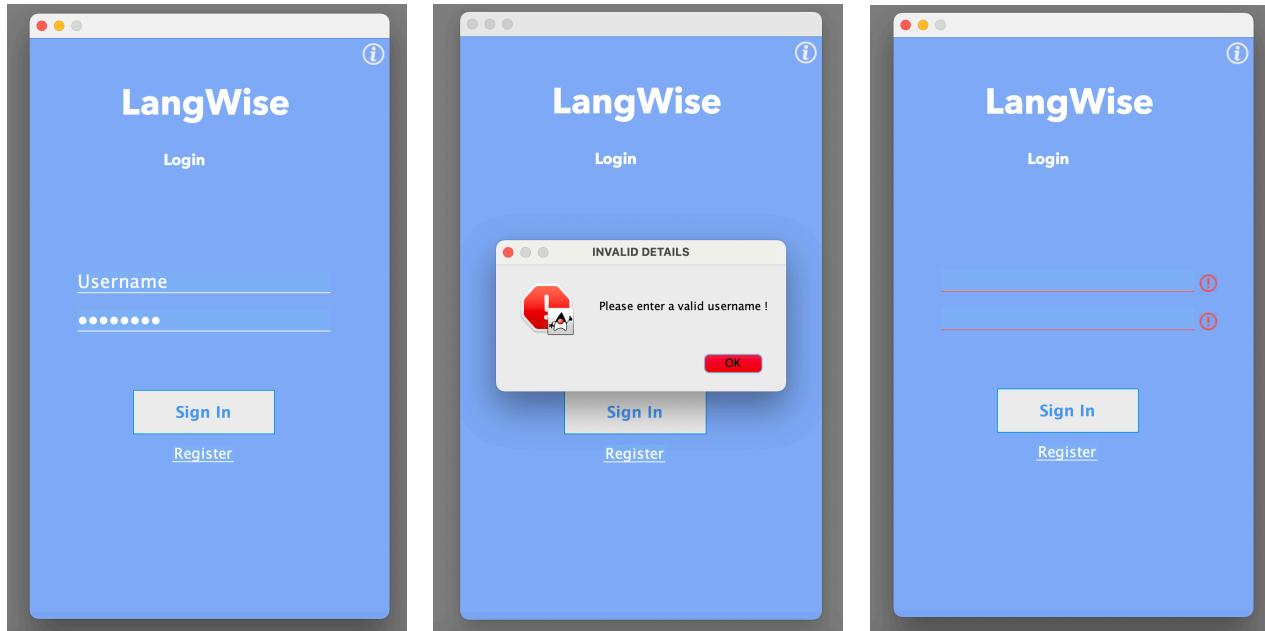
Spolsky, J., 2001. *Hard-assed Bug Fixin'*. [online] Joel on Software. Available at: <<https://www.joelonsoftware.com/2001/07/31/hard-assed-bug-fixin/>> [Accessed 14 April 2022].

Bittner, P. and Hornecker, E., 2005. *Proceedings of the 4th Decennial Conference on Critical Computing*. New York, NY: Association for Computing Machinery, pp.69-78.

Wayner, P., 2014. *12 ethical dilemmas gnawing at developers today*. [online] Info World. Available at: <<https://www.infoworld.com/article/2607452/12-ethical-dilemmas-gnawing-at-developers-today.html>> [Accessed 14 April 2022].

Scacca, S., 2022. *Why Ethical Design Is Critical for Mobile App Designers*. [online] Telerik Blogs. Available at: <<https://www.telerik.com/blogs/why-ethical-design-is-critical-for-mobile-app-designers>> [Accessed 4 May 2022].

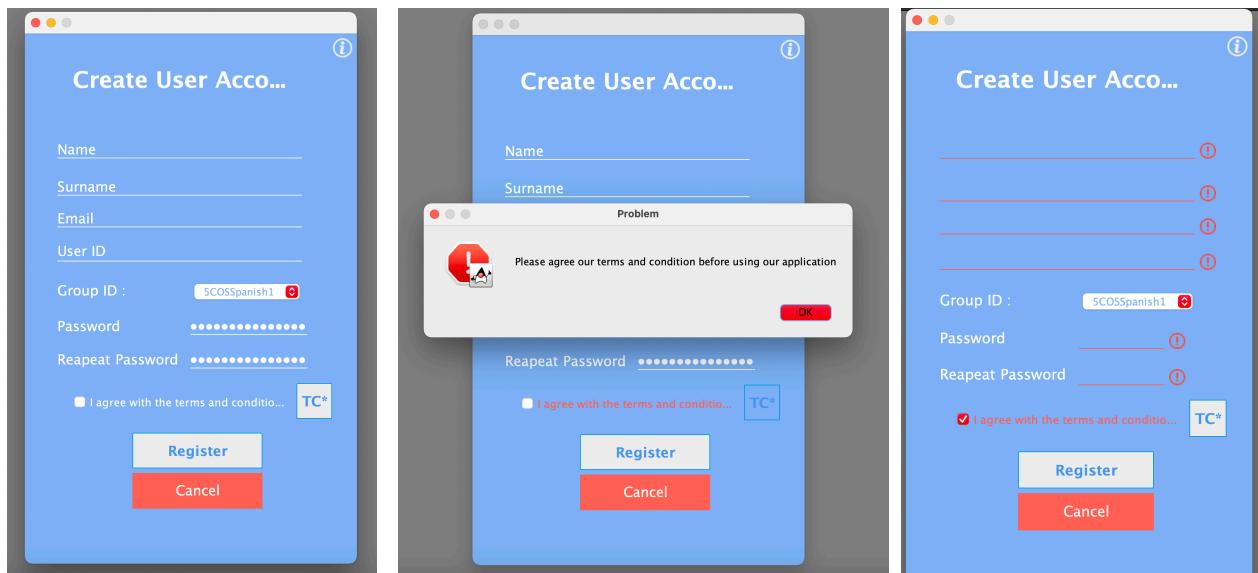
## Appendix 1: Screenshot of Application pages and who did them.



Login page [by Amit]

Login page error(1): input warning pop out

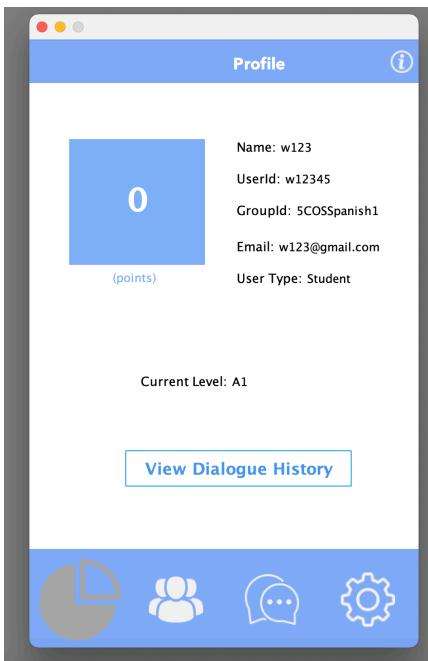
Login Page error (2): design error



Register page [by Monesha]

Register page error(1): input warning pop out

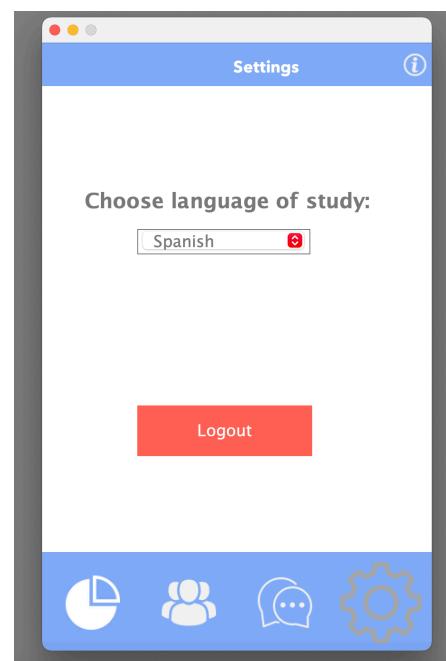
Register Page error (2): design error



Profile Page [by Rafael]



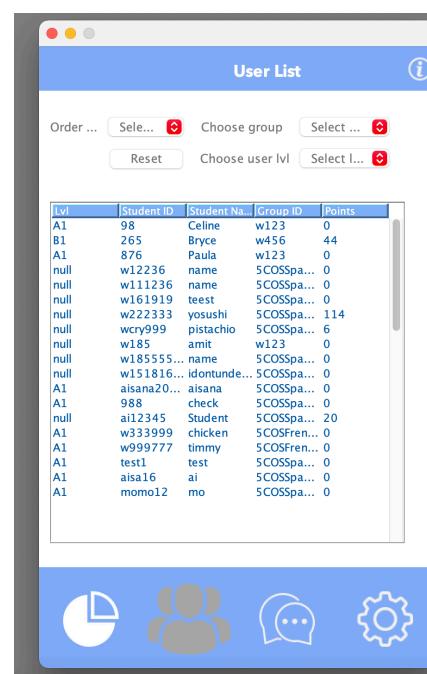
Dialogue history page [by Rafael]



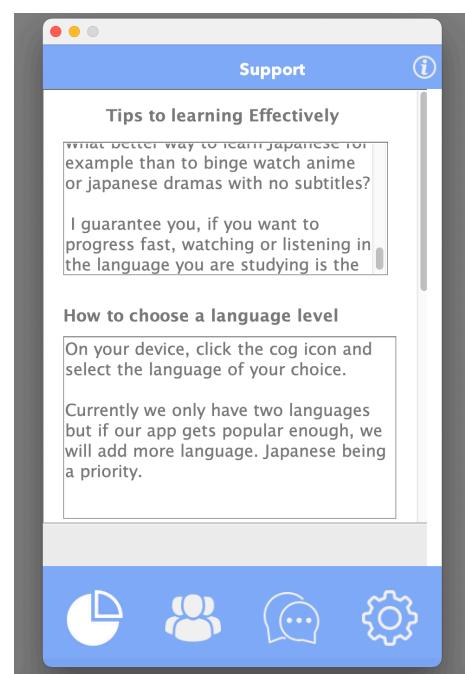
Settings page [by Monesha]



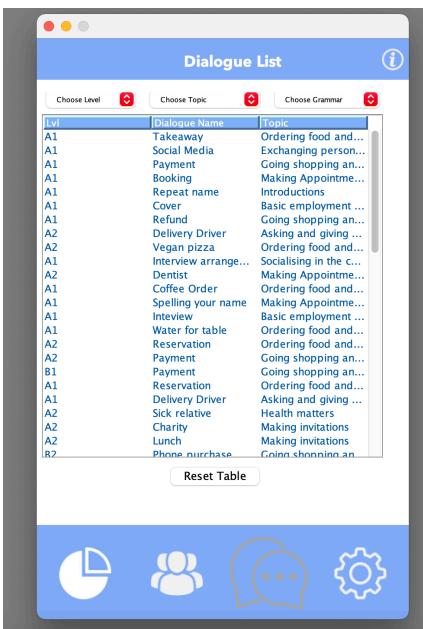
User List Page (Student View) [by Joshua]



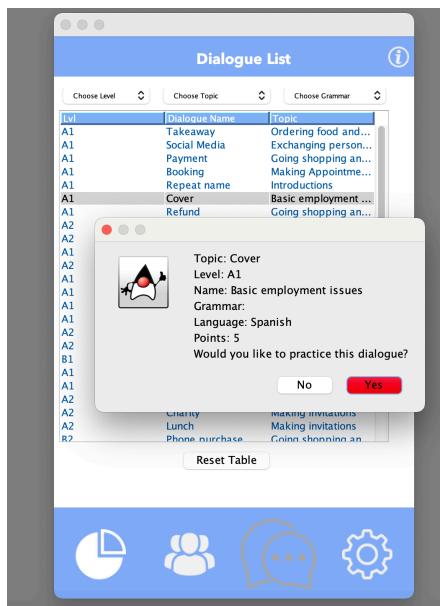
User List Page (Teacher View) [by Amit]



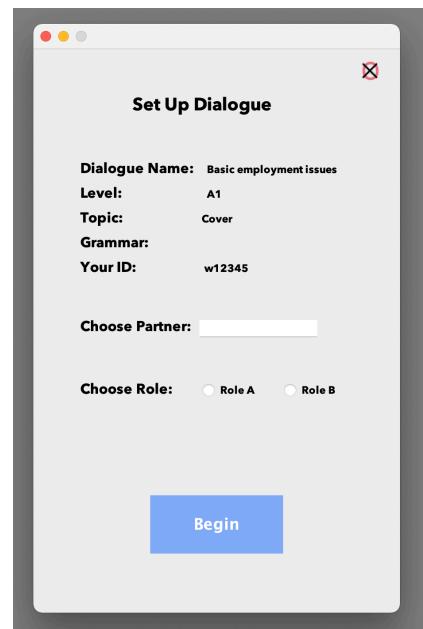
Support Page (for signed in user) [by Joshua]



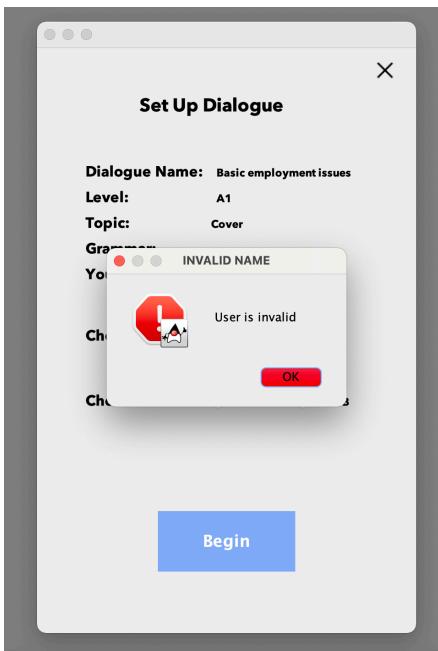
Dialogue List page [by Aisana]



Dialogue List Page: selecting Dialogue



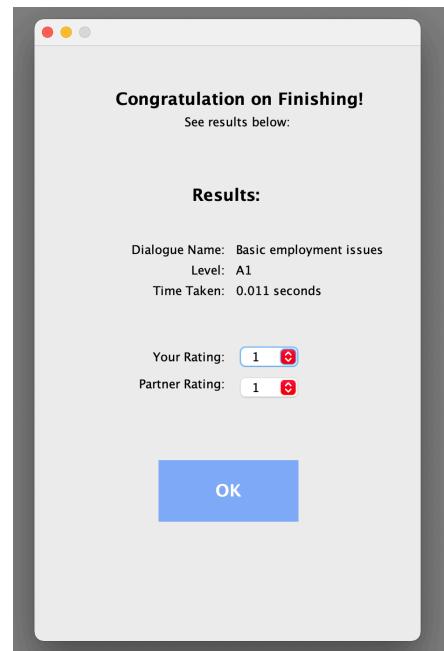
Setup Dialogue page [by Zain & Aisana]



Setup Dialogue page error input warning



Card Scroll page [by Zain]



Completing dialogue confirmation window [by Zain]

Design of the pages done primarily by:

- Monesha, Amit, Aisana
- Aisana managed all the button icons

