**Blue and red letters with white text

Description automatically generated**

**Assignment Of**

**Human-Computer Interaction (2218)**

**Submitted By:**

**PAVAN MAHESH PATEL**

**(MAH23208175)**

**Submitted To:**

**Lecturer Mr Mossab Salih**

**MSc Computer Science**

**2024/025**

**UNIVERSITY COLLEGE BIRMINGHAM**

**Table of Contents**

[**Table of Contents 2**](#_b6gjxfmhl06w)

[**Introduction 2**](#_k7e6qw6ckeom)

[**Task 1 - System Design 4**](#_60vi8ee6plg9)

[Mood Boards for Layout Ideas 5](#_1i8qm59e7fp1)

[Client Requirements 7](#_na2oirlmr1ye)

[Hardware and Software Requirements 7](#_7k18tsy7jlns)

[Detailed Flowcharts 8](#_xsoffb9qjqd7)

[Annotated Storyboards 9](#_cfmugn4ajdkb)

[**Justification of HCI Principles Use 10**](#_l8w0v5mnujxc)

[**Task 2 - Implementation 11**](#_xzg7cgork0ff)

[Certification Tab Implementation: 13](#_g05rgv3n1n2y)

[Enrol Course Form 16](#_4i25ouemtrrf)

[Add Tutor Form: 17](#_thjccsyt6tt5)

[Main Application Window 18](#_rlula93gbdcd)

[**Task 3 - Evaluation 19**](#_jjs9f2y0a9pt)

[Evaluation of the HCI Solution for PassITApp - Suitability for the Target Audience 19](#_at43f1vgo4s4)

[Usability 19](#_ka1wkxfgkgv7)

[Reliability 20](#_xsug3jtt0up)

[Maintainability 20](#_qmbc39nlduhb)

[Portability 20](#_3sb0klegfr32)

[Robustness 21](#_f1ddlixsxslg)

[Legal and Ethical Considerations 21](#_qhrzda5c0q0d)

[Strengths of the Solution 21](#_1j9jheuoktaz)

[Weakness 22](#_kmb6v09ds96l)

[Recommendations 23](#_vh3gtqoanz8j)

[**Conclusion 23**](#_fm57ir21gt0k)

[**References 25**](#_li04zbp8dn1y)

#### Introduction

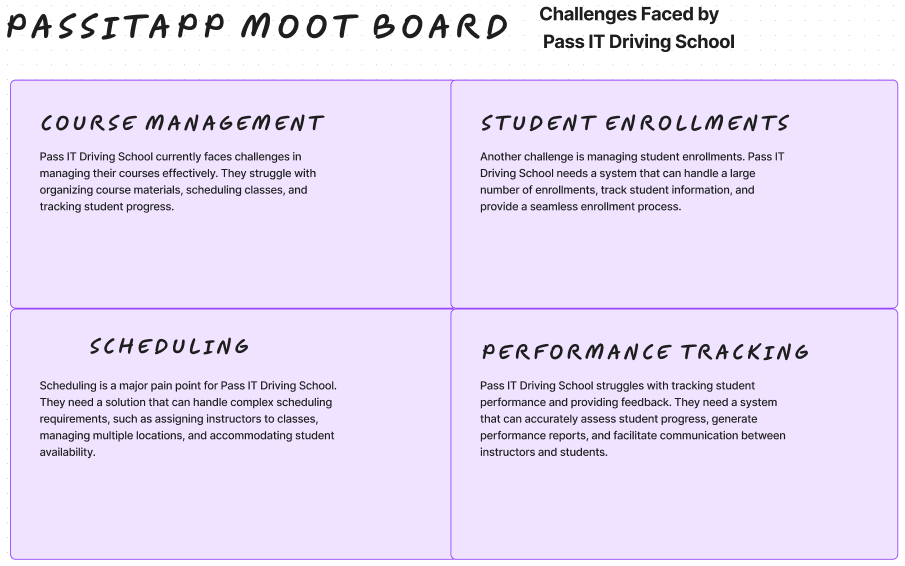
PassITApp is a full-blown Human-Computer Interaction (HCI) product built to automate Pass IT Driving School's administrative and educational procedures. It has been designed with one primary goal: to permit the interaction between the school staff, the students, and all the processes with the best possible provision of course management, enrolments, scheduling, and performance tracking effectively and efficiently. The application utilises the principles of HCI to build a user-friendly application with interfaces that are intuitive and responsive to the diversified needs of its users. The main requirements of PassITApp are robust functionalities for different users, such as administrators, tutors, and students. Admins should supervise the whole system, courses, and tutor assignments. It is supposed to provide tutors with information concerning their schedules, the learning progress of students, and the resources required for the lesson at a particular moment. The student should be able to register for the course, view their schedules, access certificates, and feedback.

This system design employs the basic rules of HCI design: simplicity, consistency, feedback, and error prevention, which results in an environment where minimal training and confusion allow a user to execute the task. Some significant elements utilised in designing the system are mood boards to reflect the visual design, detailed flow charts of the processes, annotated storyboards of how users interact, and technical specifications that clearly define the software and hardware requirements to make the system scalable and maintainable. These components are collectively implemented to ensure that function and user experience fit into practical and aesthetic needs in the driving school's digital landscape.

#### Task 1 - System Design

The system architecture is meticulously designed with provisions for administrators and instructors for features to perform the most varied operations like handling the registrations, scheduling courses, certifying the students, and generating real-time reports and information about all the educational activities. The user interface adopts tabbed navigation, which, according to Oulasvirta (2020), breaks the application into different "zones" of functionality, thus reducing the cognitive burden and helping in better navigation (Oulasvirta, 2020). The design process starts with creating mood boards, which would help conceive the aesthetic and layout of the application. The colour palette is represented by some soft greens and greys, which set the general mood for calmness and concentration. The selection of icons and typography considered their readability and universal recognition to make the interface understandable for everyone, regardless of their experience with digital applications.

##### Mood Boards for Layout Ideas

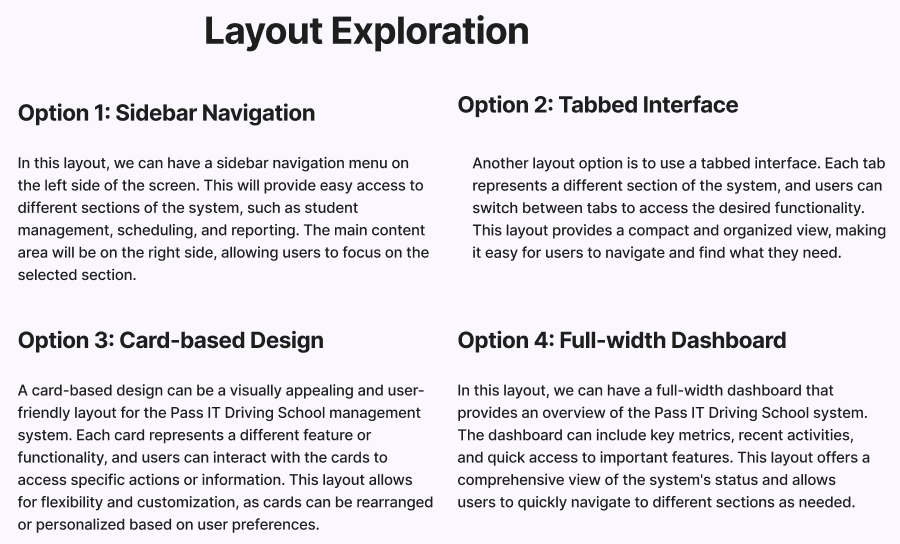


**Figure1: Challenges Mootboard**

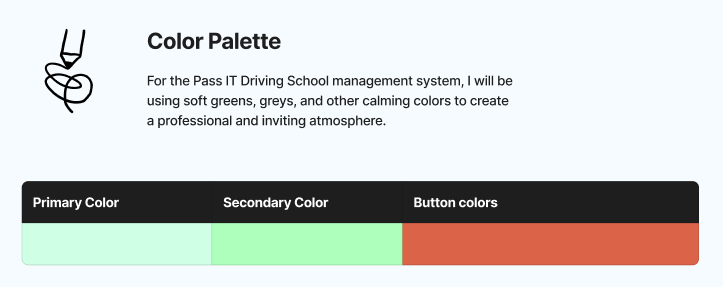
A page of a storyboard

Description automatically generated

**Figure 2: Mood board of expected core features**

****

**Figure 3: Mood board of layout options**

****

**Figure 4: Mood board colour selection**

##### Client Requirements

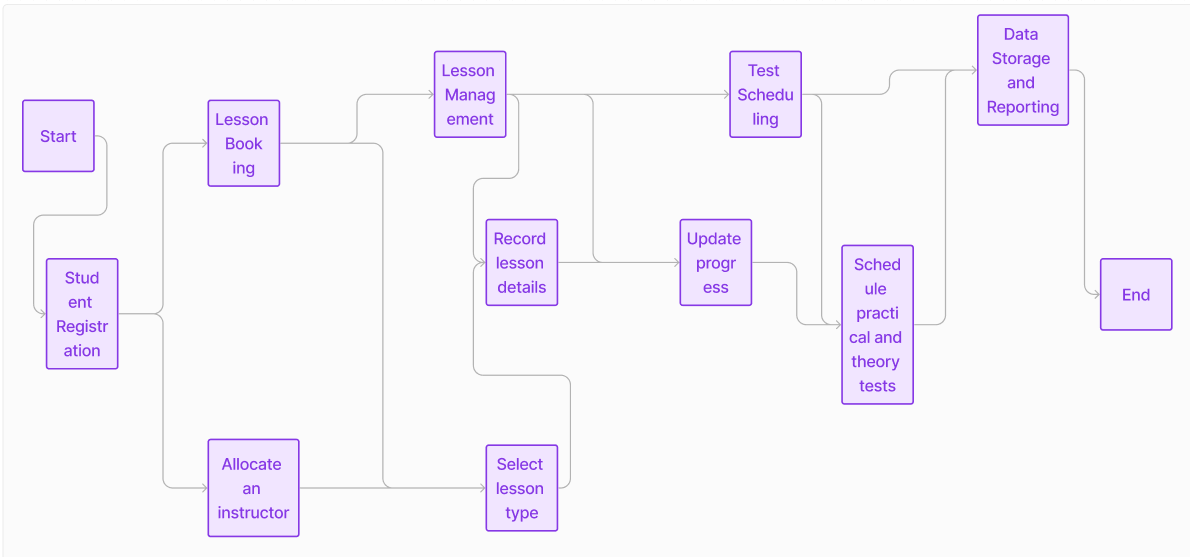
The app satisfies the client's requirement for an educational management system that is robust enough to closely monitor the student's progress, tutor effectiveness, and course scheduling. The app facilitates interactions, starting with student registration and continuing with the certificate generation to complete the course. The application of HCI design principles, apparent in the design and building of the management system for Pass IT Driving School, ensured an interface that is user-friendly and appropriate for the everyday life usage of road warriors. Since this system is based on a user-centred design, the efficiency and effectiveness of the educational management processes are enhanced, providing a powerful tool that supports the administrative and educational activities of the school.

##### Hardware and Software Requirements

The system is optimised for low- to moderate-specification hardware, which is common in educational environments. It requires a lightweight Python runtime and SQLite and can run on various operating systems such as Windows, macOS, and Linux.

##### Detailed Flowcharts

Every application module, from user login to data management, was mapped with detailed flowcharts that outline the logic and sequence of operations, ensuring all user actions lead to a predictable and intended outcome. Storyboards for key user interactions were annotated to detail the step-by-step process, providing a visual script for the application’s flow and enhancing the understanding of user and system interactions.

****

**Figure 5: Flow chart expected solution**

##### Annotated Storyboards

****

**Figure 6: Annotated storyboard of the expected solution**

***Technical Specifications:*** Technical specifications include using SQLite for database management and ensuring data integrity and speed. The application’s responsiveness is designed to operate within the bandwidth limitations of institutional networks.

***Technical Designs-Code:*** The codebase is modular, enhancing maintainability and scalability. Functions are clearly defined for user authentication, database operations, and UI updates, facilitating more accessible updates and debugging.

***Consideration of Design Rules:*** The application adheres to recognised design rules such as the Eight Golden Rules of Interface Design. This calls for a consistent and intuitive user experience by uniformly employing familiar interface elements across screens (Mazumder & Das, 2014).

##### Justification of HCI Principles Use

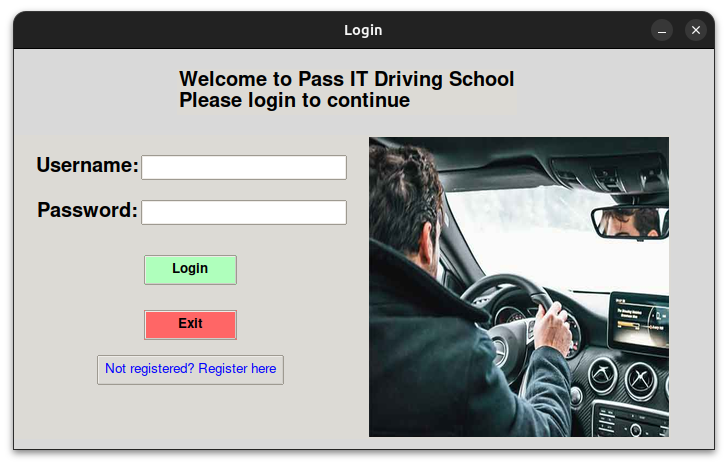
At the core of the system's design is a commitment to user control and autonomy, which is crucial in a learning environment where different users have varying needs and technical skills. The tabbed interface, a central feature of the design, allows users to navigate the system effortlessly. A good design choice significantly reduces the cognitive burden on users by organising distinct functionalities like course management, student and tutor records, and scheduling into dedicated tabs (Hollender et al., 2010). This separation helps compartmentalise tasks and the time spent looking for functions, enhancing the user experience in the long run.

Consistency across these tabs and standardised use of colours, typography, and button styles further reinforce the system's usability by maintaining a uniform interface; users can quickly grasp the system's operation. This will minimise errors and reduce the system's learning curve. Consistency also enables the patterns of interaction across the system to be predictable and familiar to the user, giving a sense of knowing one's way and being in control. Another fundamental concept of the HCI approach adopted by this system is the emphasis on error avoidance and the need for effective feedback. Every user action receives an immediate system reaction: feedback on finalising a booking, a warning about a scheduling conflict, or a notification about possible data input errors. Such interactions are very important, as they maintain a clear line of communication between the system and the user, keeping the user in the know at all times about the status of their actions and, therefore, building their trust in the system to lead them in the right direction. For example, when booking lessons, the system cross-checks the existing entries to avoid double booking, and in case of a hitch, it instantly informs the user. Therefore, it saves operational interruptions and preserves the integrity of the school schedule.

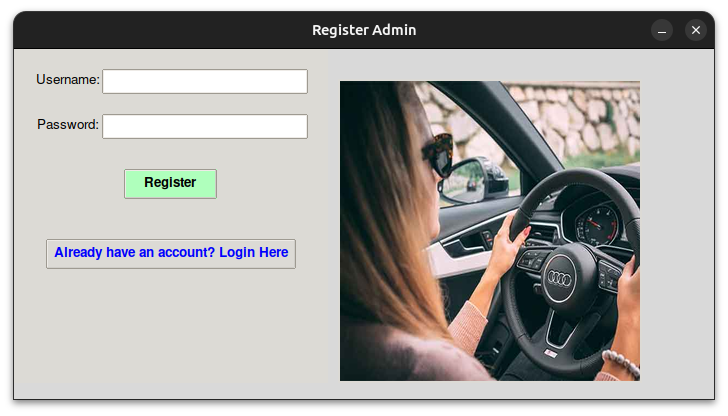
By acknowledging that human error mostly cracks systems, this system was designed with the automation of repetitive tasks and processes in mind to a great extent. For instance, the system automatically updates the records of those students who finish their courses and stores their lesson histories. Automation reduces the administrative burden on staff, allowing them to focus more on direct educational activities than routine data management tasks. Such thoughtful integration of automated processes underlines the system’s role not just as a tool for administrative management but as an active facilitator of the educational goals of the driving school. The system uses conventions familiar to the target user group, such as icons for editing or deleting entries and clear and jargon-free language. This mirrors the real-world educational administration processes, reducing the learning curve. The interface minimises the user’s memory load by making information visible or easily retrievable. According to Proctor and Vu, 2007, user actions are often guided by visible cues rather than requiring the recall of instructions, thereby simplifying the interaction (Proctor & Vu, 2007).

#### Task 2 - Implementation

Regarding interaction design, the application includes features like image carousels in the RegisterDialog and LoginDialog, which make the interface lively and provide visual cues about the organisational context. To ensure seamless interactivity, using modal dialogues for tasks such as registering or logging in ensures that the users complete necessary actions without distraction, a practice that aligns with maintaining user focus (Bailey & Konstan, 2006). The application includes dialogues for login and registration, featuring clear differentiation between fields and actions. Error prevention is facilitated through immediate feedback on incorrect entries, and visual cues are provided to successfully guide the user through the login process.



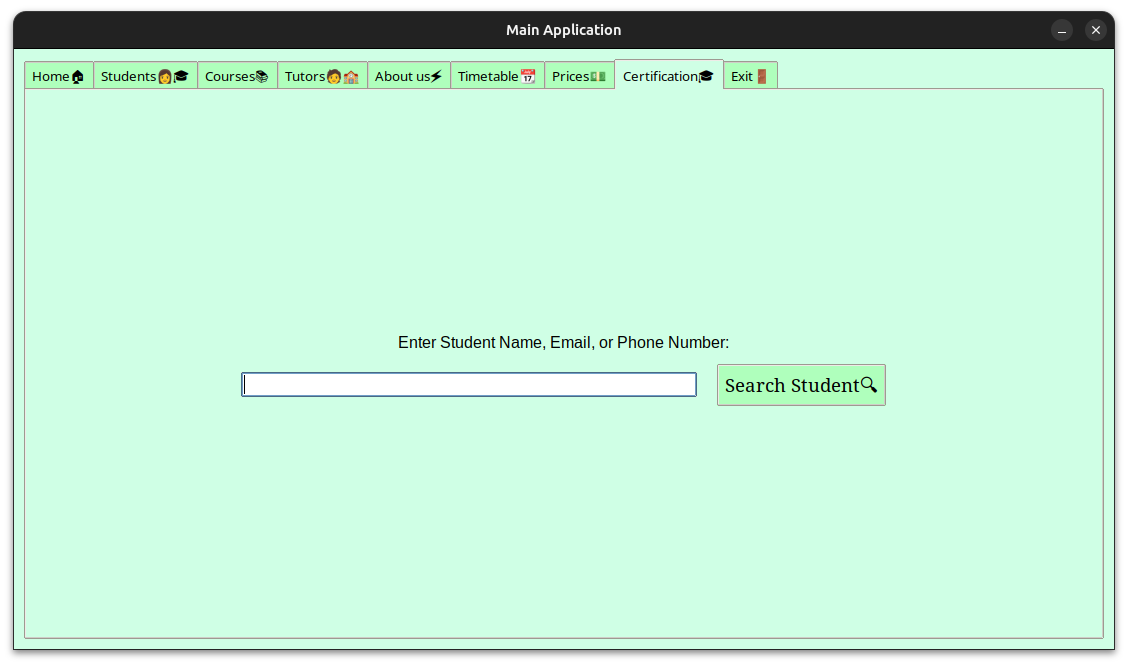
**Figure 7: Login Page**

****

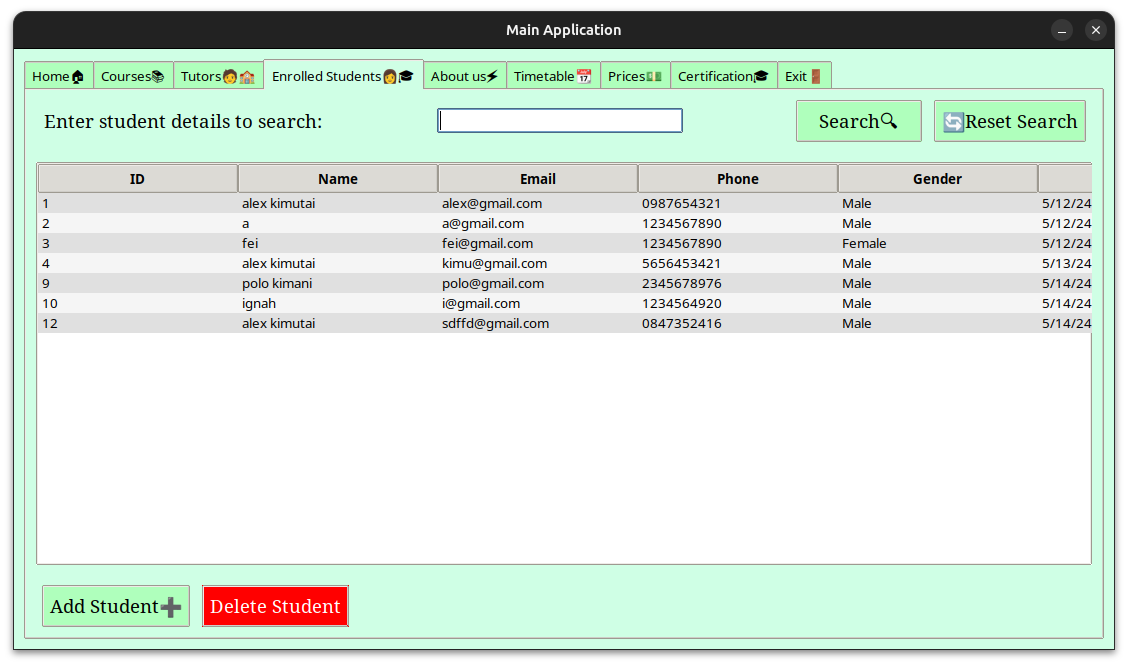
**Figure 8: Registration Dialog**

###### Certification Tab Implémentation:

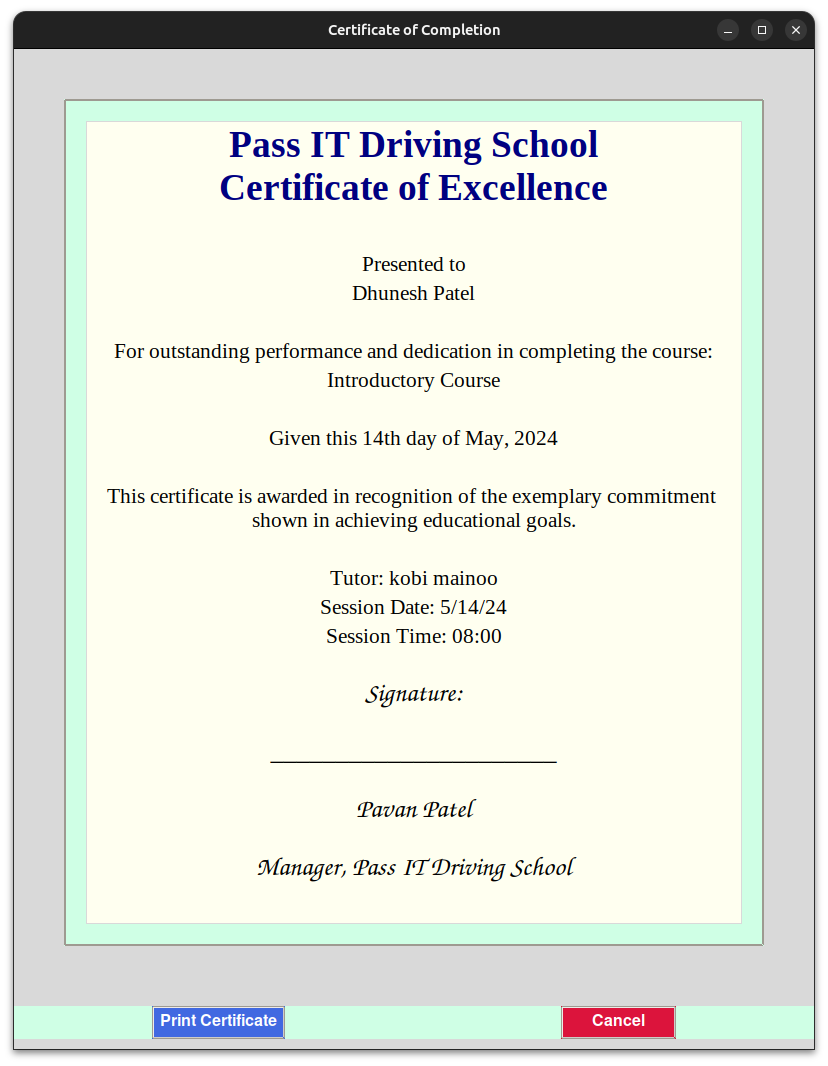
The Certification Tab allows users to search for students and generate certificates of completion. This tab was designed with a clear and straightforward layout to minimise user error and ensure ease of use. A search bar allows a quick lookup of student records by name, email, or phone number, followed by a real-time display of the corresponding certificate details in a new window. According to Fairclough, 2009, this real-time feedback is crucial for maintaining user engagement and ensuring accuracy (Fairclough, 2009).



**Figure 9: Search Bar in the Certifications Tab**

****

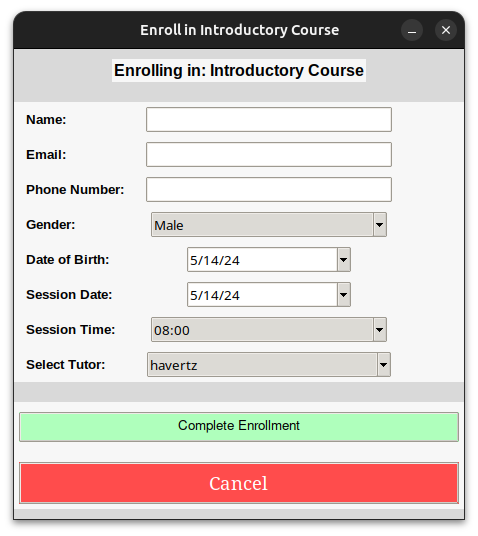
**Figure 10: Student UI**



**Figure 11: Certification UI**

###### Enrol Course Form

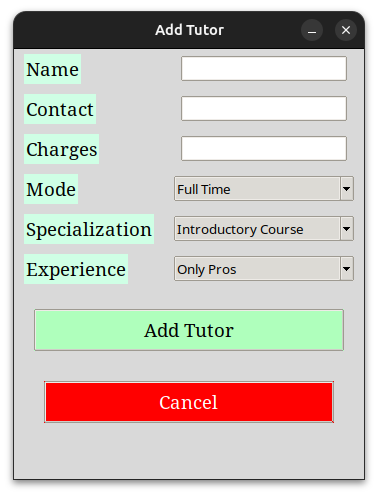
The Enrol Course Form is a modal window that captures student information for course enrolment. As advocated by Hollender et al. in “Computers in Human Behaviour*”,* 2010, this form is designed to reduce cognitive load by grouping related information, such as personal details and course-specific details, which adheres to the HCI principle of organised interfaces (Hollender et al., 2010). The form also uses combo boxes for fields like gender and session time, constraining input options to valid choices, thereby preventing user error.

****

**Figure 12: Enrol Course Form**

###### Add Tutor Form:

Adding a tutor involves filling out a detailed form with fields for name, contact, charges, and specialisation. This form validates input to ensure data integrity; for example, it uses a validation function for charge entries to accept only numeric input. According to Guide, 2012, consistency in label placements and button styles across forms aligns with HCI guidelines for familiar user experiences, promoting recognition rather than recalling (Guide, 2012).

****

**Figure 13: Add Tutor Form**

###### Main Application Window

Upon successful login, the main application window presents a tabbed interface allowing easy navigation between different functionalities like Home, Students, Courses, Tutors, and more. Using tabs reduces the clutter and organises the interface into manageable sections, improving the overall user experience. Each tab is designed with a consistent style and layout, maintaining visual and operational consistency throughout the application. Also, within the application, different tabs are used to separate functionalities, making the interface intuitive and easy to navigate—which, as per Marra (2012), is a fundamental HCI principle (Marra, 2012). For instance, the 'Students' tab allows managing student information, searching, adding, or deleting entries. Each operation within this tab is designed to be straightforward, with clear labels and buttons that are easy to locate, enhancing the overall usability and accessibility of the system.



**Figure 14: Main Application Window showing tabbed navigation**

#### Task 3 - Evaluation

##### Evaluation of the HCI Solution for PassITApp - Suitability for the Target Audience

The PassITApp is a desktop-based application developed using Python and the Tkinter library, designed to manage various educational and administrative functions within a driving school. This includes handling registrations, scheduling, and certifications for students and instructors. With educational institutions rapidly moving towards digital implementation to automate their overall working, an appropriate design for human-computer interaction becomes essential. Against this backdrop, this paper critically reviews the HCI features and functionality of the PassITApp for its effectiveness in being practical and functional for its users, mainly educational administrators and staff. The PassITApp has been developed to focus on the user base—administrative staff and instructors of the driving school—and, therefore, it offers the management of courses, students, and instructors for day-to-day operations. It is suitable for non-technical people who require an effective way to handle educational and administrative tasks. It has a simple interface with user-friendly tabs. Each tab has a function: student registration, course management, or tutor assignment. It allows the user to move through the system very fast without hitting the user with too much information.

##### Usability

The use of familiar stylings in the GUI of the application enhances the usability of the PassITApp since it uses buttons, entry fields, and clear labels that are consistently used throughout the application, simplifying for users to navigate through the system. According to Hsieh et al., 1994, in "Journal of Organisational and End User Computing (JOEUC)", there are elements of the system used consistently that the user has known about. For example, the comboboxes facilitate data entry and prevent errors and waste of time. More so, the mouse-over tooltips and confirmation messages help the user, thus reducing the learning curve significantly; it is quite a delightful user experience.

##### Reliability

By ensuring good data management, PassITApp is very reliable since the most straightforward functions are performed with uniform results. In its structure, the application appropriately implements data management tasks through SQLite, a lightweight, fast, and reliable database for small-to-medium applications. This decision guarantees average operation performance typical for driving school administrative work.

##### Maintainability

Code application is well-classified within classes and methods corresponding to different school administration process functionalities, such as handling student enrolment or tutor scheduling. The modular design applied to the system makes it understandable, maintainable, and easily updatable and alterable. This means, for instance, if new functionality needs to be added or existing features must be modified. Developers could track down and update the code sections without affecting irrelevant modules.

##### Portability

Built using Python and Tkinter, PassITApp capitalises on the cross-platform features that the previously mentioned technologies possess. Minimal changes make the application portable across most operating systems, including Windows, macOS, and Linux. Therefore, according to Gonzalez-Martinez et al., this kind of portability is a significant advantage in strategic terms for deploying cross-platform software since it can run in other computing environments, hence enabling ease for users who switch between multiple operating systems (Gonzalez-Martinez et al., 2015).

##### Robustness

The strength of the PassITApp is evident in its ability to manage varied inputs and data user interaction without a single point of failure. The application uses fundamental error-handling mechanisms that can help avoid crashes and, in turn, ensure data corruption, hence maintaining the system in a stable state and functional under conditions of unexpected user input and operation.

##### Legal and Ethical Considerations

PassITApp is designed to raise awareness of the legal and ethical responsibilities and concerns of managing educational data. It has thus been designed to address any such concerns and through the safe handling of personal information, mainly about student and instructor information—generally critical under privacy laws such as the EU General Data Protection Regulation or equivalent enactments in other jurisdictions. The application ensures that personal data is stored securely and that access is restricted to authorised users, aligning with best data protection practices.

##### Strengths of the Solution

The significant strengths of PassITApp are related to ease of use and efficiency in performing tasks emanating from a user-centred design. The system is designed with a clear and well-organised interface so users can complete their tasks without confusion and error. The other strength is database integration in the practical storage and retrieval of data to maintain accurate records in an educational environment. The system is designed with some scalability in the sense that it allows for the addition of new tabs or better reporting features, and so on, without having to do much redesigning.

##### Weakness

While PassITApp is quite functional and designed with a user-friendly interface, several respects can be considered impractical and helpful from the users' perspective. The first big weakness of the application is the interface, which, though very intuitive, is still much dependent on traditional desktop paradigms that most modern web users may not find effective. This may make the software feel outdated and possibly deter engagement, especially from younger staff and those more accustomed to modern software solutions. Second, and an even more significant limitation, the software is incompatible with mobile devices. According to Burford and Park, 2014, in today’s environment, where mobile device usage is prevalent, the inability to access the system on smartphones or tablets is a significant drawback (Burford & Park, 2014). This restricts users from interacting with the system on the go, potentially decreasing productivity and responsiveness. Additionally, the system's performance scalability has not been thoroughly addressed. There may be issues with database performance, problems in handling user loads, and data processing in real-time as the volume of the users and data keeps escalating. Such problems make a system respond slowly, making users less satisfied, especially in peak administrative periods. Also, it does not have multi-language capability at present, which is necessary in educational environments that are diversified worldwide, where most users would like to interact with the system in their local languages. This could limit its use and accessibility for non-English-speaking users and, in turn, its adaptability in international markets if Pass IT chooses to migrate its services or lease its software.

##### Recommendations

While PassITApp does serve its functional requirements of addressing basic needs in managing educational and administrative tasks within a driving school, there are a few areas where improvements need to be made to complete the user's satisfaction and meet modern software standards. To further improve it, it should introduce cloud-based database solutions that would enable its remote accessibility, make its user interface responsive, and implement tight security features to meet the prescribed legal standards for handling data. Moreover, increased feedback mechanisms and improved error handling will significantly enhance user experience and system reliability. By looking into these areas, PassITApp can better serve the stakeholders it serves and evolve with new educational administration trends.

#### Conclusion

The PassITApp has been designed for Pass IT driving schools, with all-around views of this powerful tool to improve driving school activity management and an outstandingly designed user interface. This application uses principles of human-computer interaction to make it an integrated environment in which staff and instructors can work on booking and managing driving lessons while maintaining students' progress. On its base basis, PassITApp is engineered for optimum operations to make education management a natural and efficient process. The system was architected to handle the school's needs: more than one type of driving lesson, dynamic rescheduling of instructors, and record maintenance of student progress and test results. The design of such an approach will meet not only the requirements of the driving school in practice but also ensure that users can cope with the system and navigate it with minimal preparation. An example of such a well-designed, thoughtful interface based on HCI principles is the PassITApp, which will revolutionise the administrative and educational processes of the driving school. It will enhance the operational support of the school but, at the same time, make the learning process more certain for students, who will be assured of their well-managed and documented educational experience. This project is not only for the current needs of the Pass IT driving school but also scalable and flexible for future enhancement. Thus, it is a long-term viable project as an educational management tool.

#### References

Bailey, B. P., & Konstan, J. A. (2006). On the need for attention-aware systems: Measuring effects of interruption on task performance, error rate, and affective state. Computers in human behaviour, 22(4), 685-708.

Burford, S., & Park, S. (2014). The impact of mobile tablet devices on human information behaviour. Journal of documentation, 70(4), 622-639.

Fairclough, S. H. (2009). Fundamentals of physiological computing. Interacting with computers, 21(1-2), 133-145.

González-Martínez, J. A., Bote-Lorenzo, M. L., Gómez-Sánchez, E., & Cano-Parra, R. (2015). Cloud computing and education: A state-of-the-art survey. Computers & Education, 80, 132-151.

Guide, D. (2012). User Interface.

Hollender, N., Hofmann, C., Deneke, M., & Schmitz, B. (2010). Integrating cognitive load theory and concepts of human–computer interaction. Computers in human behaviour, 26(6), 1278-1288.

Hsieh, C. T., Lu, M. T., & Lin, E. (1994). Consistency in human-computer interfaces for end-users. Journal of Organisational and End User Computing (JOEUC), 6(2), 3-10.

Marra, R. (2012). Human-computer interface design. In Hypermedia learning environments (pp. 115-135). Routledge.

Mazumder, F. K., & Das, U. K. (2014). Usability guidelines for usable user interface. International Journal of Research in Engineering and Technology, 3(9), 79-82.

Oulasvirta, A., Dayama, N. R., Shiripour, M., John, M., & Karrenbauer, A. (2020). Combinatorial optimization of graphical user interface designs. Proceedings of the IEEE, 108(3), 434-464.

Proctor, R. W., & Vu, K. P. L. (2007). Human information processing: an overview for human-computer interaction. The human-computer interaction handbook, 69-88.