System Requirements Specification

1. The system should be running and continuously taking pictures via a qualified camera during the lecture time and keep adding the attendance of the people it identifies until the end of the lecture. This gathered information is then stored and structured in a database. Student's data will be added to the attendance table with the time of attending and the lecture number.
2. A database must be made to hold all the important information about and required by the system. This database should be a relational database that holds tables about different entities of the system with all the attributes or fields necessary about that specific entity. One of these entities is the student; the system should have the full name, student id, email, university, faculty, and the different courses that the student is registered in. Also, the face encodings of that specific student so that the system knows his face. Another entity is the Course and each of these must have an id and the course code and name. Also, the Faculty is an important entity and it has a many-to-many relationship with the University entity.
3. The database should follow the ACID (atomicity, consistency, isolation, durability) properties which is a set of properties of database transactions intended to guarantee data validity despite errors, power failures, and other mishaps. In the context of databases, a sequence of database operations that satisfies the ACID properties (which can be perceived as a single logical operation on the data) is called a transaction. For example, a transfer of funds from one bank account to another, even involving multiple changes such as debiting one account and crediting another, is a single transaction. Therefore, the database must have a backup and replications. The Quality of the database is maintained in such a way so that it can be very user friendly to all the users of the database. The database will be SQLite.
4. Different universities can be added to the system and work as a distributed system with maintaining the integrity and security of each one of them. The admins or university staff at each university can only work with the data for this specific university and not be able to comprise or manipulate any other data. Also, each doctor or faculty staff is authorized to access the least amount of data possible. A report is generated with the requested data for the admin so that they can seek information anytime they need. Also, the attendance for each course must be restricted to only the doctors responsible for this course and can have live updates in real-time of their attendance. Also, the system should be running and available all the time of the day and every day.
5. The university staff must provide the necessary information about the different faculties, doctors, and courses for the university before the system production. This must be achieved through adding the required data directly to the database on the server through some python functions. These data must be accurate and complete so that when going to production, the system is ready to handle user requests. These data should include all faculties in the university, all doctors applied in each of the faculties, and the courses required in each of the faculties with which doctors are working on them.
6. The student must register on the system to be able to identify him before he can take attendance and that happens via a web form the student must fill and provide his picture to upload it on the server. Each student will be able to choose his faculty while registering. This web form contains all columns of the student table besides the courses he/she applied to.
7. After student registration, the system should analyze the picture he provided and detect his face and reject the picture if the picture turns out to have multiple faces or if doesn’t contain a face at all. And if the picture passes these conditions, the system should get the encodings of that face and store them in the database. Another important edge case is if this specific encoding is already in the database, in this case the picture is also rejected.
8. For testing and usability measurements, the student can verify attendance by either uploading a picture or live video feeding to the system throughout his\her webcam.
9. To log in as a doctor or admin the user must provide his full name or username as well as his password. Then the system checks if the user data is registered in the database. If the data is valid then redirect the user to the admin panel otherwise show an error message, and if the user tries to access the admin panel without login, he will be redirected to the login page.
10. Any passwords or sensitive information cannot be stored in plain text but rather must be hashed to secure it. Using argon2 hash generator algorithm to generate a hash of each password in the database to protect the user's password in case of a data leak. the hash value will be unique for each password and argon2 is not fast to crack while md5 or sh1 are easy to brute-force thousands of keys in on each second.
11. When a user submits the registration form all the user data must be carefully checked and sanitized to prevent SQL injection, stored cross-site scripting, and object deserialization. which can escalate further to remote code execution or database leak.
12. Doctors can change their passwords while keeping security constraints like password length and including lower and upper letters and special characters. the new password is sent to the server and on the server-side constraints are checked again to make sure the new pass is valid then it generates the argon2 hash for the pass and update the database. all this is done only if is not log in this he has no access to this page.
13. Each Doctor can add his courses to the database. Each course has a unique code and table of students. Students will be able to register for these courses and then attend it. If the student provided a course that does not exist in the database, that course should be simply ignored.
14. Doctors will send warnings to students if they are near to the allowed limit of absence in any course. It will make it easier for doctors to communicate with students and easier for students to keep track of their absence. The allowed limit of absence will be set by the faculty staff. Also, the doctor will receive remainders or warnings if there is a deadline or some of the students are about to violate one of the university attendances rules (e.g., absence more than three lectures in a row). Therefore, the students should be informed and send the reasons for their absence.
15. The system should which course to take attendance into and which lecture of that course via knowing the faculty schedule before hand or the doctor will configure these data before the lecture starts. But for testing sake, we will let the student provide the name of the course and the lecture number before submitting his\her attendance.
16. Allow the user to view their input of the password field in plain text to be able to notice any misspelled word in their input. And that can be achieved by a “show password” checkbox that the user can enable or disable.
17. Prevent unauthorized URL traversal like access admin page without login this done by check for user credential in user session on every page if the user was not unauthorized to view that page, he will be redirected to the login page. The session id is stored in a cookie on the user browser this allows the server to distinguish each user.
18. In uploading pictures, either in registration or attendance, the system should check the image type and the extension in the filename. And if the type is in the allowed types (e.g., jpg, png, jpeg) then try to resize the picture and if successful then proceed to the next steps. This must be done in both the client side and the server side to avoid malicious files to be uploaded.
19. The system will send an error message to the student if the uploaded image wasn't clear enough to identify him/her. This will increase the system's accuracy. Each student will know the problem with his/her registration and this will make a better user experience.
20. Any student is able to stream a live video to the dedicated window on the administration website in order to submit the attendance in order that the uploaded image is not sufficient to recognize the standing person.
21. There is a green led attached to to protect the led from over-current operation. This governed by ohm law which states that the current through a conductor between two points is directly proportional to the voltage across the two points. Introducing the constant of proportionality, the resistance, one arrives at the usual mathematical equation that describes this relationship: where I is the current through the conductor in units of amperes, V is the voltage measured across the conductor in units of volts, and R is the resistance of the conductor in units of ohms. More specifically, Ohm's law states that the R in this relation is constant, independent of the current. There are two led in our microprocessor-based module depending on the validation of the image matching operation which indicates whether the student is recognized or not which led to update the database field or not as well.
22. The power supply requirements differ by Raspberry Pi model. All models require a 5.1V supply, but the current supplied generally increases according to model. All models up to the Raspberry Pi 3 require a *microUSB* power connector, whilst the Raspberry Pi 4 uses a USB-C connector. Exactly how much current (mA) the Raspberry Pi requires is dependent on what you connect to it. The current requirements for Raspberry Pi 4 Model B are specified as the following:

* Recommended PSU current capacity: *3.0 A*
* Maximum total USB peripheral current draw: *1.2 A*
* Typical bare-board active current consumption: *600 mA*

The power requirements of the Raspberry Pi increase as you make use of the various interfaces on the Raspberry Pi. The GPIO pins can draw 50mA safely, distributed across all the pins; an individual GPIO pin can only safely draw 16mA. The HDMI port uses 50mA, the camera module requires 250mA, and keyboards and mice can take as little as 100mA or over 1000mA! We have to provide stable electrical power source with the previous requirements and specification from two separated devices to ensure power availability and system efficient performance.

1. Construct a system with enough computing capabilities in order to do image enhancement locally on the microprocessor chip so we must provide a microprocessor with enough processing capabilities such as sufficient RAM size to store the image in an appropriate manner as well as the raspberry pi 3 board come with ARM processor to provide processing at an efficient time and many other resources with appropriate specifications as following:

* Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
* 2GB, 4GB or 8GB LPDDR4-3200 SDRAM (depending on model)
* 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE
* Gigabit Ethernet
* 2 USB 3.0 ports; 2 USB 2.0 ports.
* Raspberry Pi standard 40 pin GPIO header (fully backwards compatible with previous boards)
* 2 × micro-HDMI ports (up to 4kp60 supported)
* 2-lane MIPI DSI display port
* 2-lane MIPI CSI camera port
* 4-pole stereo audio and composite video port
* H.265 (4kp60 decode), H264 (1080p60 decode, 1080p30 encode)
* OpenGL ES 3.0 graphics
* Micro-SD card slot for loading operating system and data storage
* 5V DC via USB-C connector (minimum 3A\*)
* 5V DC via GPIO header (minimum 3A\*)
* Power over Ethernet (PoE) enabled (requires separate PoE HAT)
* Operating temperature: 0 – 50 degrees C ambient