**Tutor Match CRM Project — Final Documentation & Presentation**

**Introduction**

**Tutor Match CRM** is a Salesforce-based project designed to streamline the process of connecting students with tutors. It enables students to easily book sessions, tutors to manage schedules, and administrators to gain insights through dashboards and reports — all while ensuring security and scalability.

**Phase 1: Requirement Gathering & Planning**

* Identified challenges: tutor discovery, scheduling conflicts, limited visibility for administrators.
* Defined objectives: simplify booking, improve communication, automate processes, and ensure secure access.

**Phase 2: Data Model Design**

* Created **Custom Objects:**
  + **Student\_\_c** – captures student details.
  + **Tutor\_\_c** – stores tutor information and expertise.
  + **Course\_\_c** – lists courses offered.
  + **Booking\_\_c** – records student session requests.
  + **Session\_\_c** – manages confirmed sessions.
* Established relationships: Lookups and Master-Detail to link Students, Tutors, and Courses.

**Phase 3: User Interface Setup**

* Customized **Page Layouts** to show relevant fields per object.
* Used **Lightning App Builder** to design record pages with Highlights Panel, Related Lists, and Tabs.
* Implemented **Dynamic Forms** to show fields conditionally based on record values.

**Phase 4: Data Import & Management**

* Used **Data Import Wizard** for small volumes (Students, Tutors, Courses).
* Used **Data Loader** for large datasets.
* Created CSV templates for structured data import.
* Ensured field mapping to Salesforce standard and custom fields.

**Phase 5: Automation with Flows & Rules**

* **Record-Triggered Flows** – auto-create sessions when a booking is confirmed.
* **Scheduled Flows** – reminder notifications before sessions.
* **Validation Rules** – prevent invalid data (e.g., booking date in the past).
* **Approval Processes** – manage booking discounts or tutor changes.

**Phase 6: Integration**

* **Email & SMS Notifications:** Queueable Apex callouts to notify tutors/students.
* **Payment Gateway Stub:** Apex class simulating payment process with token, amount, currency.
* Demonstrated external API handling through callouts.

**Phase 7: Apex Programming**

* **Batch Apex:** Designed for large data processing (e.g., reassigning counselors).
* **Queueable Apex:** For asynchronous operations like background enrollment updates.
* **Scheduled Apex:** For periodic tasks such as monthly reports.

**Phase 8: Reporting & Dashboards**

* **Reports:** Built Tabular, Summary, Matrix, and Joined reports for different perspectives.
* **Report Types:** Standard and Custom Report Types created for specific reporting needs.
* **Dashboards:** Real-time charts for bookings, tutor performance, and student engagement.
* **Dynamic Dashboards:** Personalized views depending on the logged-in user.

**Phase 9: Security Review**

* **Sharing Settings:** Configured org-wide defaults, role hierarchy, and sharing rules.
* **Field-Level Security:** Restricted sensitive information like student contact and tutor payment details.
* **Session Settings:** Configured login timeout and forced re-authentication.
* **Login IP Ranges:** Restricted access to trusted networks.
* **Audit Trail:** Enabled to monitor configuration changes.

**Phase 10: Final Presentation & Demo Day**

* **Pitch Deck:** Highlighted problem, solution, features, implementation, and impact.
* **Demo Walkthrough:**
  1. Student books a course session.
  2. Tutor receives a real-time notification.
  3. Admin monitors activities through dashboards.
* **Benefits:** Time savings, improved matching, data-driven insights.
* **Future Enhancements:** AI-driven recommendations, video conferencing integration, mobile app.

**Conclusion**

The **Tutor Match CRM project** demonstrates how Salesforce can be leveraged to solve real-world education challenges. With its custom objects, automation, integrations, and reporting capabilities, it provides a complete end-to-end solution for tutor-student management. The project ensures scalability, security, and adaptability for future needs.