Applied System Software

Title Slide – Team Info & GitHub Link

•Team X – Section 000

•Members:

Michael Yaghi – 250AEB040

Noah N’Diaye Falcy – 250AEB039

Clovis Le Floc’h – 250AEB050

Maxime Viti – 250AEB014

Axel Pineau – 250AEB022

•GitHub Repository: https://github.com/Noah-ndf/site-web

Project Overview and Motivation

- **Problem**: Psychologists need a modern, user-friendly tool to manage their appointments online.
- **Goal**: Create a professional website that showcases the psychologist's services and enables online booking.
- **Motivation**: Simplify administrative work and enhance accessibility for clients.

Chosen SDLC Methodology – and Why

- **Model Chosen**: Agile
- **Why Agile?**
- Frequent client feedback (mockups, weekly iterations)
- Flexibility to implement evolving requirements (e.g., email notifications, 24h cancellation logic)
- Rapid prototyping and testing

Requirements Gathering & Analysis

•Sources: Direct questions to the psychologist, weekly team journals

•Key Findings:

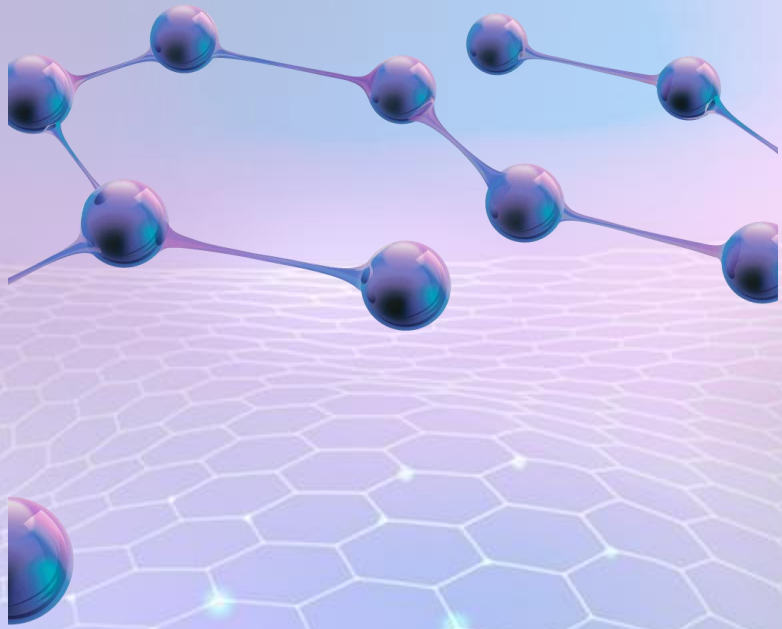
Clear display of services and credentials

Secure client login system

Time-slot-based booking system

Mobile compatibility

Optional features like email alerts and video consultations



Software Requirements Specification

- Functional:

User login/registration

Psychologist sets availability

Clients book/cancel sessions

Dashboard for both roles

Email confirmation

- Non-Functional:

Responsive design

Security (hashed passwords, JWT)

GDPR compliance

Load time < 2s

System and Software Design

- Architecture: Modular RESTful API
- Backend: Node.js, Express, MongoDB
- Frontend: React, Axios, React Router
- Authentication: JWT with role-based access
- Email Module: Asynchronous utility for confirmations

Implementation Phase

- Frontend: Registration, login, booking UI, protected routes
- Backend:
- CRUD API for users & slots
- Authentication middleware
- MongoDB schemas for integrity
- Git & GitHub: Centralized development with pull requests

Testing Strategy

- Manual Testing with Postman for APIs
- Black-box testing for booking flows
- Equivalence Class Testing:
- Valid/invalid credentials
- Available/unavailable slots
- Logged-in vs not logged-in users

Test Results and Validation

- Major Test Cases Passed:
- Booking validation
- JWT route protection
- Form validation & feedback
- Tools Used: Postman, browser DevTools
- Edge Cases:
- Booking expired slots
- Overlapping availabilities
- Cancelling < 24h

Process Challenges and Lessons Learned

- **Challenges**:
  - Syncing frontend and backend logic
  - Managing async email triggers
  - Enforcing booking rules
- **Lessons Learned**:
  - Better planning for roles
  - Importance of consistent testing
  - Modular backend design benefits

Demo Overview – What Features You'll Show

- User registration & login (client and psychologist)
- Psychologist managing availability
- Client booking and cancellation
- Live dashboard updates
- Email notification trigger

Conclusion and Future Work

- **Summary**: Built a working full-stack appointment management app for psychologists
- **Next Steps**:
- Add video consultation feature
- Build admin analytics view
- Deploy to production with CI/CD

Q&A / Feedback

We appreciate your time and attention throughout our presentation.