Monday, 19 August 2024 8:21 PM

Japanese Reading old text handwriting

→ Web application (Open on browser)
→ Desktop application

Stakeholder :- Occupation
↳ Client ↳ linguist
↳ researcher

Researcher → Secure
↳ hard to hack/confidential

Software

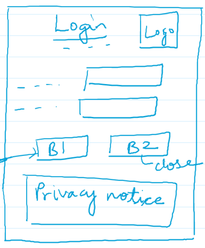
Desktop	Web
① Not that easily accessible	① Easily accessible anywhere
② Highly secure as it works only on a particular system	② Highly breachable despite security measures
③ Hardly any addit° cost. Use Visual studio 2022. It has inbuilt security check.	③ Additional cost many ↳ Domain purchase ↳ SONAR purchase ↳ IP SECURE purchase

→ Based on the client's occupat°, desktop appl° is the most feasible option.

Software design → Desktop application

→ Main form
↳ log → specific folder
↳ redirecting to next page

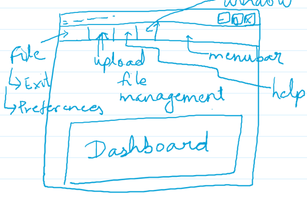
→ Login page (GUI)



→ Only connect via anu id or ANU Global connect.

→ If they are outside and not connected to ANU network, they should get a message "kindly connect to the ANU n/w or ANU Global connect."

→ Main Page (GUI)



Dashboard
↳ Graphs
↳ charts for statistics
↓ Example
No of files uploaded : 10
No of files having old Japanese Text : 05
No of Japanese Text decrepted 100
Success rate : 25%

Software creation

→ Data-access layer (DAL)
→ Research layer (RL)
→ UI layer (GUI)

DAL → Connect to database/backend
↳ Decide which DB to use
db.sqlite3, Postgres SQL, My SQL, Django, etc.
↳ RDBMS concepts → apply all
↳ EER, HLD, LLD, Tables, Keys, Indexes, views, security.

RL → Intermediary to DAL & GUI
↳ Common function
↳ Global variable
↳ packet descriptions (TCP/UDP)
↳ CV, Text reading, Neural N/W, etc

GUI → Front-end