STUDENT IT ARCHITECTURE COMPETITION - IV

RECOGNIZING AND PROMOTING THE ART AND SCIENCE OF IT ARCHITECTURE

Architecture Design Document Part II

Contents

OVERVIEW 2

SOFTWARE ARCHITECTURE 3

USER EXPERIENCE / USER INTERFACE ARCHITECTURE 7

SECURITY ARCHITECTURE 12

INFRASTRUCTURE/DEPLOYMENT ARCHITECTURE 14

TECHNOLOGY STACK 16

Glossary 18

1. OVERVIEW

This document outlines various facets of Maitiri's architecture divided into six parts: Software Architecture, UX/UI Architecture, Security Architecture, Infrastructure Architecture, Technology Stack, and Glossary/References. The various architecture sections capture the viewpoints. Each section includes the relevant diagrams supported by description of the components, connectors and relationships.

The software architecture contains the layout of all components and how they interact with each other. It is an eagle-eye view of the complete system layout. The intended audience include: software developers, project managers, and other IT and business stakeholders.

The user experience architecture captures all aspects related to the interactions between the system and the different types of users for web and mobile apps. The intended audience is the end user (personas - typical student, professional aid member, admin), as well as business stakeholders and software engineers (building the interfaces).

The security architecture layout is a unified description of the various components and mechanisms that describe how the potential risks and threats to the system are addressed. The intended audience include developers, architects, and IT stakeholders.

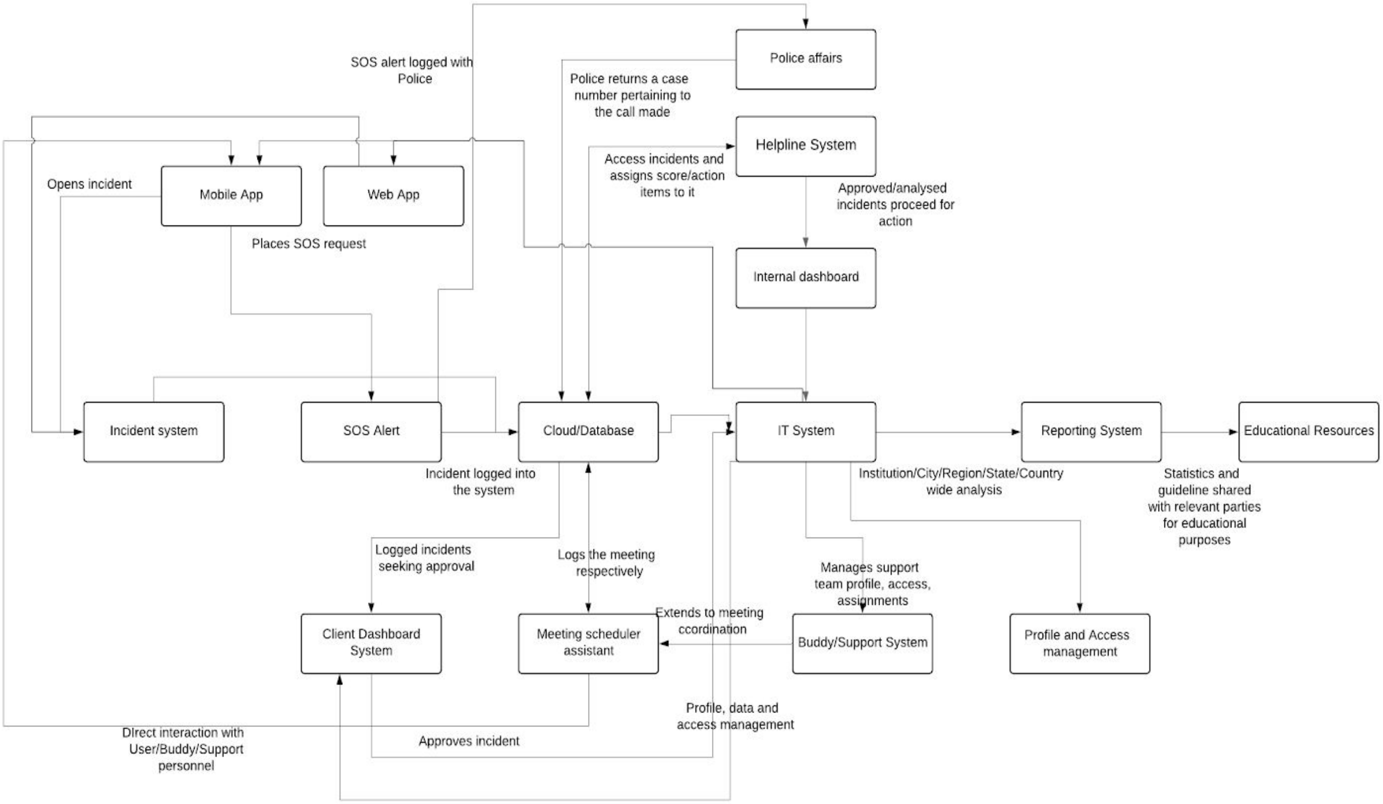
The infrastructure deployment architecture covers information on the hardware that would be used (computing, storage, network). The intended audience is: designers, architectes, project management, and stakeholders.

The technology stack section describes all technology components including OS, DB, Cloud, Mobile used in the architecture. The intended audience is: developers, project managers, and business stakeholders.

Lastly, the Glossary provides a list of terms and abbreviations used in the document that are defined.

2. SOFTWARE ARCHITECTURE

The software architecture described below contains the layout of all components and how they interact with each other. It is an eagle-eye view of the complete system layout.

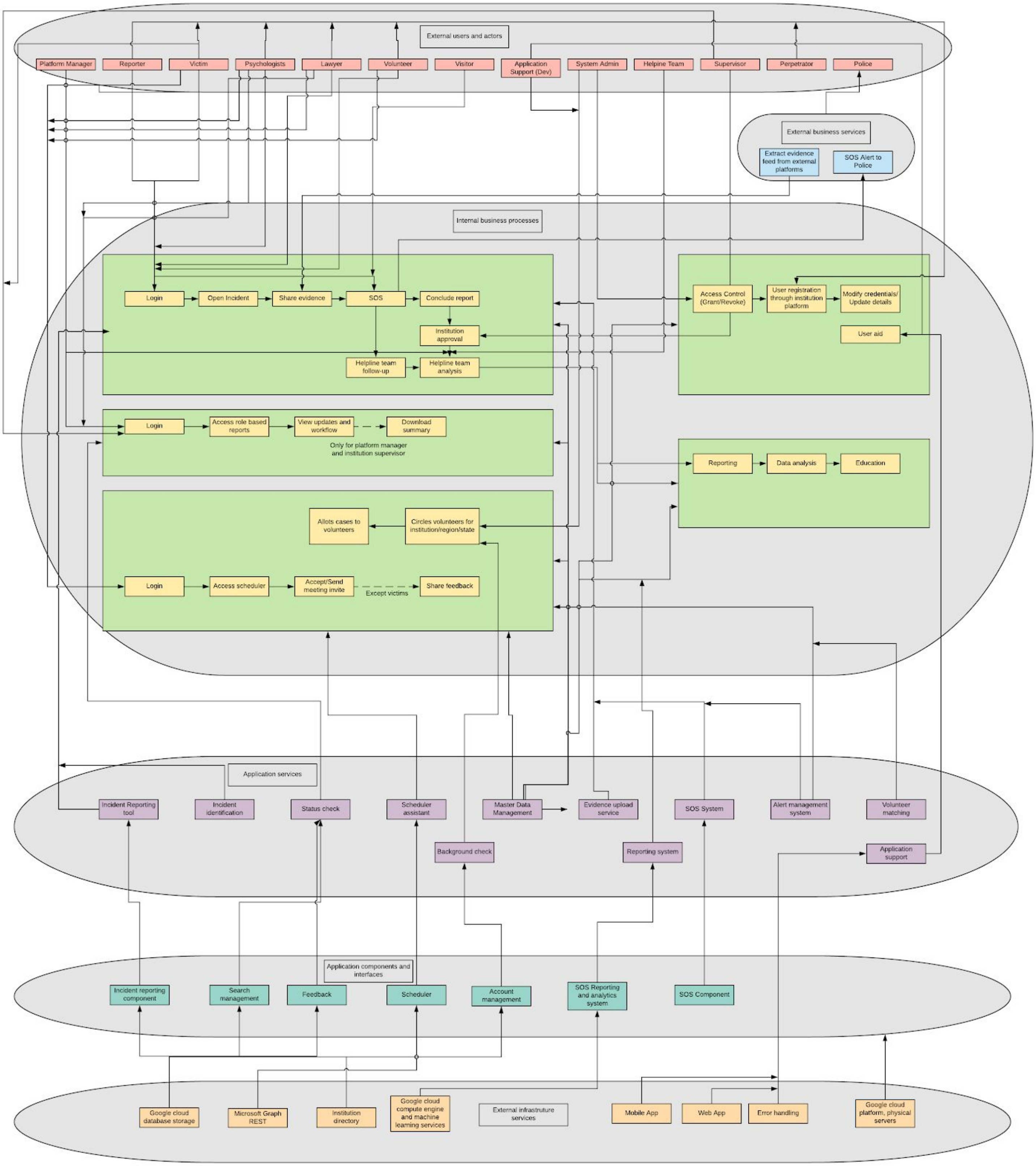

Police affairs
SOS alert logged with Police
Police returns a case number pertaining to the call made
Helpline System
Access incidents and assigns score/action items to it
Approved/analysed incidents proceed for action
Opens incident
Mobile App
Web App
Places SOS request
Internal dashboard
Incident system
SOS Alert
Cloud/Database
IT System
Reporting System
Educational Resources
Incident logged into the system
Institution/City/Region/State/Country wide analysis
Statistics and guideline shared with relevant parties for educational purposes
Logged incidents seeking approval
Logs the meeting respectively
Manages support team profile, access, assignments
Extends to meeting coordination
Client Dashboard System
Meeting scheduler assistant
Buddy/Support System
Profile and Access management
Profile, data and access management
DIrect interaction with User/Buddy/Support personnel
Approves incident

Description of the components

Component	Description	Key Non-Functional Qualities that are relevant
Web App	This is a web accessible UI for internal and external users to raise incidents, check status, accept/initiate meetings. All data processing information is streamed to the portal near real time. All the user inputs are adopted on real time basis through.	Security – authentication and access through institution based login platform. Response – Page load, Loading of images, Loading of options to either raise an incident or check status or update profile information. Availability - as per SLA
Mobile Application	This is a web accessible UI for internal and external users to raise incidents, check status, accept/initiate meetings and initiate SOS calls. All data processing information is streamed to the portal near real time through. All the user inputs are adopted on real time basis through.	Security – authentication and access through institution based login platform. Response – Page load, Loading of images, Loading of options to either raise an incident or check status or update profile information. Availability - as per SLA
Incident System	This component is a collaboration of incident registration interfaces. User can open an incident in real time by uploading evidence and sharing details of incident. User is allowed to place calls in case of emergency (only through mobile app). The users can also search the workflow of past incidents as they are updated by going to the platform and searching for them.	Security - the details are captured in real time and transferred by masking via an encrypted medium to the server. Reliability - the system will store the details as being inputted by the user and would not get cleaned in case the save function was unsuccessful. Availability- The system will be accessible 24*7. SOS functionality will be available 24*7, but the processing of incidents would be over the working days.
SOS System	This component is used to raise emergency call which are diverted to nearest police facility. The calls are made in real-time and patched to the closest police station	SOS functionality will be available 24*7 Usability, Performance
Cloud Database System	All the data entries made in the system, irrespective of the user or role are saved in the system in real-time and would be accessible to users based on the permissions.	Modifiability -- the data should be modifiable
IT System	This system is a common point for interaction between users, institution heads, helpline team and business team. It would contain all business rules and logic that allow for decision making, processing and data handling.	Reliability - the system should be reliable to host the fundamental operations at all times, while following business rules.

Client dashboard system	All the incidents opened for a particular institution are forwarded first to the institution body (in a batch) for approval and then can be cascaded to helpline team for processing (through a batch process). The client institution heads can also extract reports of incidents in their campus for internal purposes.	Availability - the data on dashboards for institutional heads should be updated periodically
Helpline System	The helpine body moderates the incidents received to assign a severity score and intervene (on a case by case basis, later forwarded to IT system in a batch), escalate in case of dire situations. This phase would be automated in advanced stages of the platform growth.	Availability - this system should be available at all time
Internal Dashboard	Primarily for business heads of Maitri, platform manager and system admins to access curated dashboards for reports,processing access/granting of new users / bullies.	Availability - as per SLA Security - the data should be secure
Meeting Scheduler	The victims, lawyers, buddies and psychologists can extend and accept meeting invite by accessing the scheduler assistant on the app. The invite is sent through in real time.	Availability - as per SLA Modifiability - the meetings should be modifiable
Buddy/Support System	This system maps and maintains volunteers to be allotted to different cases. They are able to see cases assigned to them , arrange meetings and provide feedback.	Usability - the system is only usable if the mapping is systematically for feasible options.
Profile/Access management	The internal team will manage and control access at the time when a institution partners with Maitiri and when an incident occurs that requires access control intervention. They also help out users manage their accounts.	Security - the data should be secure Modifiability - the access controls should be modifiable
Reporting System	The system will help business groups and institution heads analyse the problem in their area and craft anti-bullying solutions and curb the problem foundationally.	Availability - as per SLA Security - the report will be shared keeping privacy check points and compliances in system rules.
Educational Resources	The information acquired from anti bullying case report will be used to spread awareness about bullying incidents in an area and develop consciousness amongst people.	
Police affairs	The police will be alerted about an incident through SOS call. Later they can provide Maitri the official case number, if any, to be updated in database.	Availability - 24*7

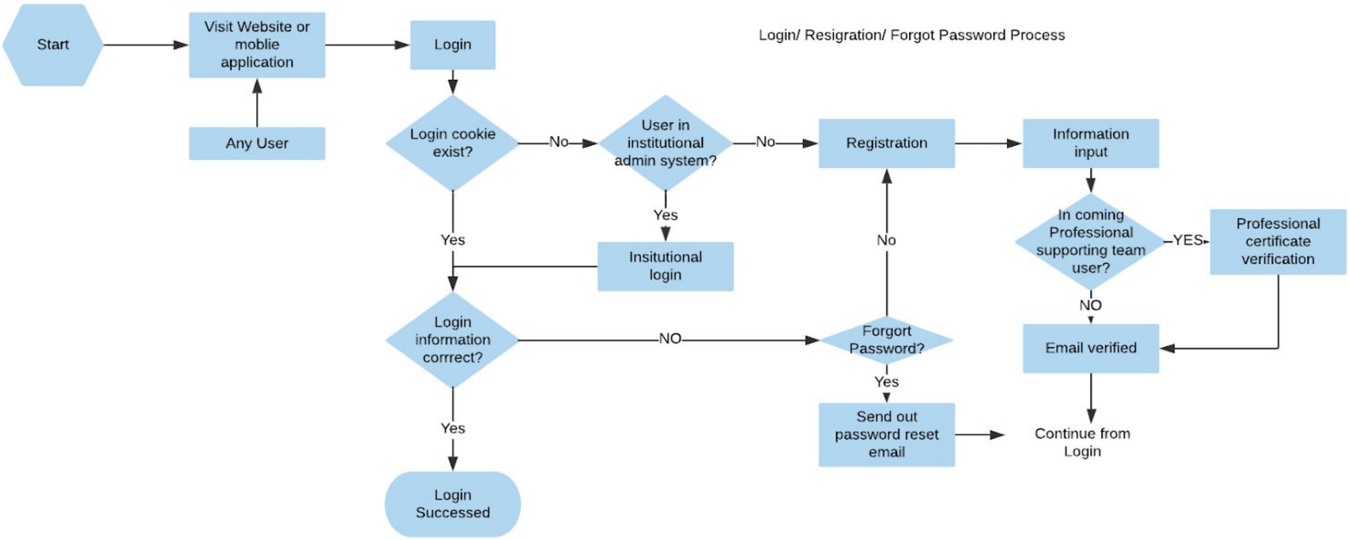
Layered architecture:



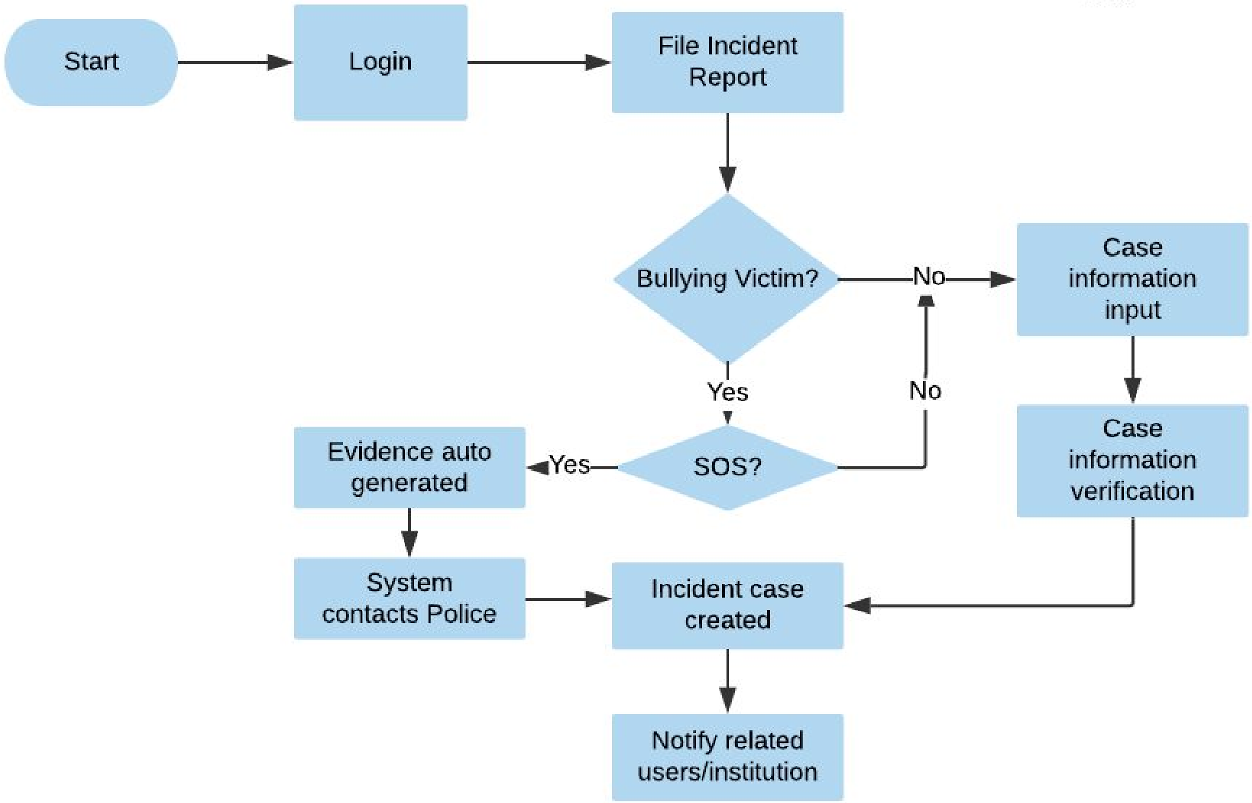
3. USER EXPERIENCE / USER INTERFACE ARCHITECTURE

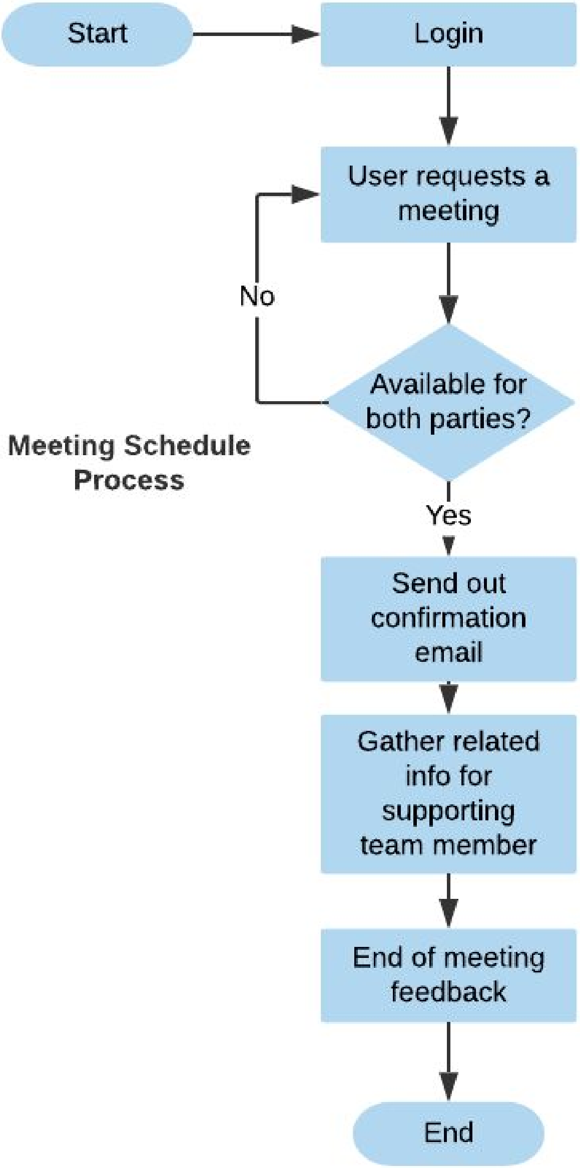
User Experience architecture captures all aspects related to the interactions between the system and the different types of users for web and mobile apps. It covers aesthetic appearance, application screens, navigation, and the content presented to the user.

Navigation flows:


Login/ Resigration/ Forgot Password Process
Start
Visit Website or moblie application
Any User
Login
Login cookie exist?
No
User in institutional admin system?
No
Registration
Information input
Yes
Insitutional login
Yes
No
In coming Professional supporting team user?
YES
Professional certificate verification
NO
Email verified
Login information corrrect?
NO
Forgort Password?
Yes
Send out password reset email
Continue from Login
Yes
Login Successed

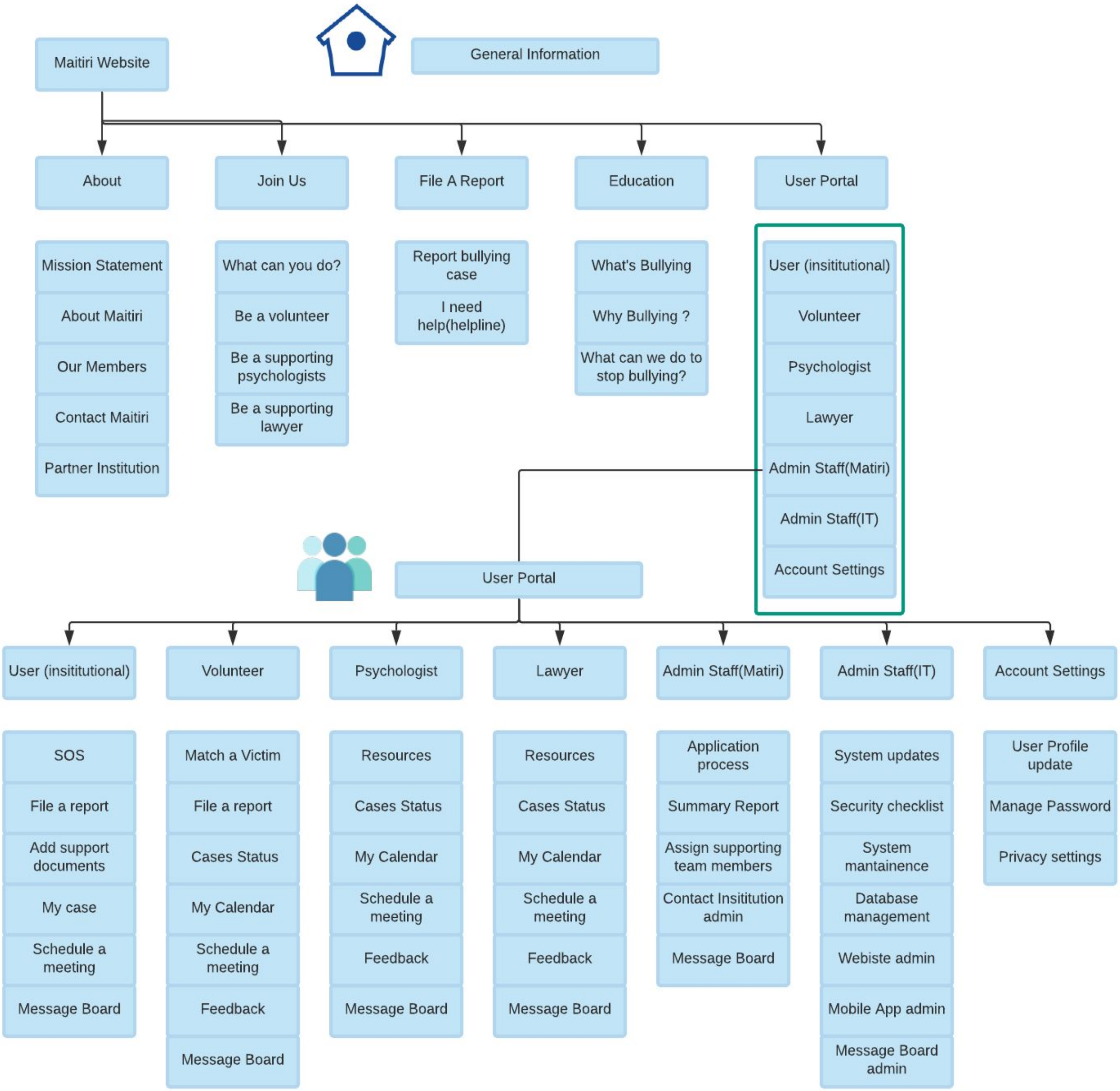
Reporting / SOS process flow



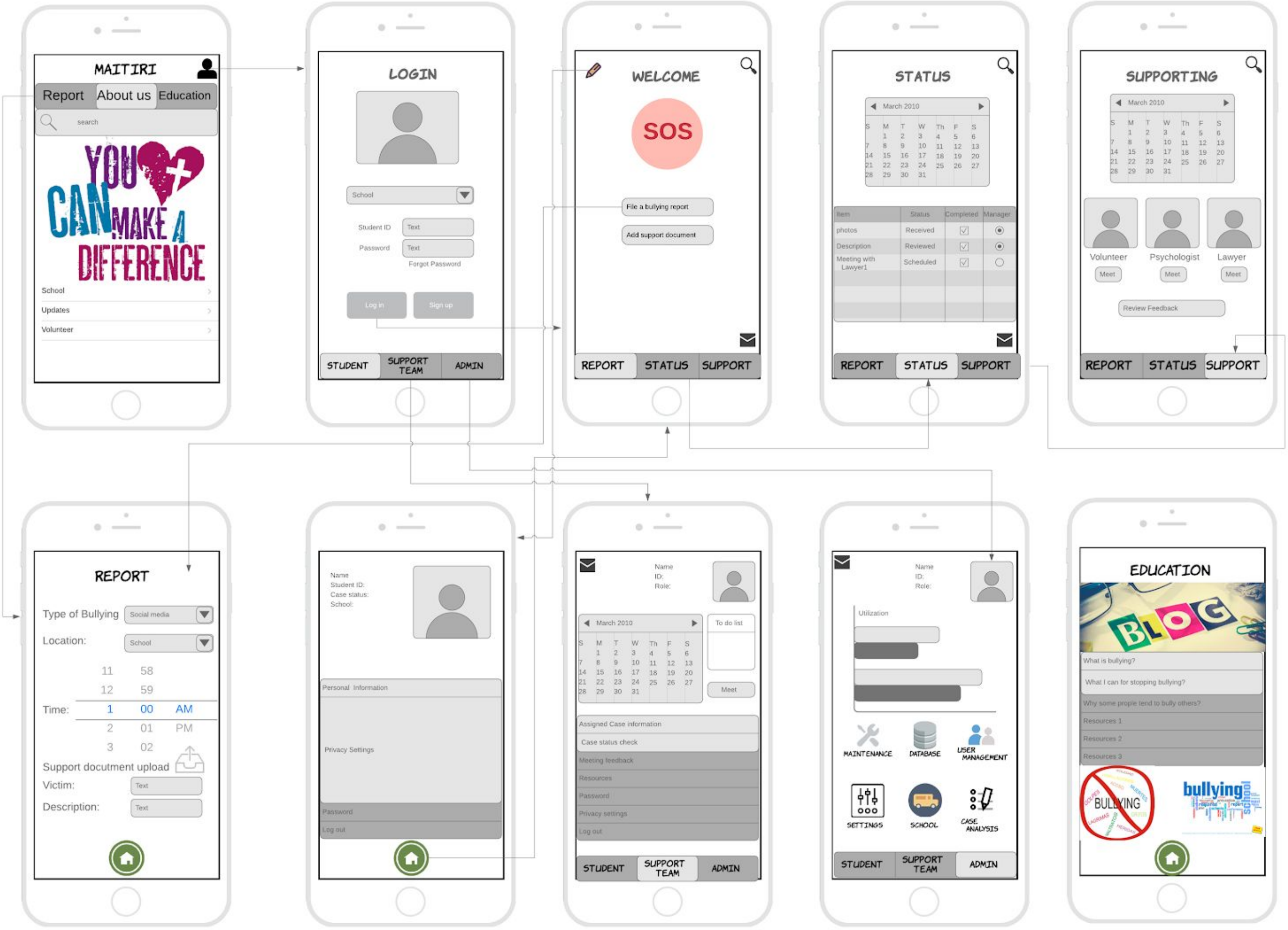

Start
Login
User requests a meeting
No
Available for both parties?
Yes
Send out confirmation email
Gather related info for supporting team member
End of meeting feedback
End

Meeting Schedule Process

Sitemap:


Maitiri Website
General Information
About
Join Us
File A Report
Education
User Portal
Mission Statement
About Maitiri
Our Members
Contact Maitiri
Partner Institution
What can you do?
Be a volunteer
Be a supporting psychologists
Be a supporting lawyer
Report bullying case
I need help(helpline)
What's Bullying
Why Bullying ?
What can we do to stop bullying?
User (insititutional)
Volunteer
Psychologist
Lawyer
Admin Staff(Matiri)
Admin Staff(IT)
Account Settings
User Portal
User (insititutional)
Volunteer
Psychologist
Lawyer
Admin Staff(Matiri)
Admin Staff(IT)
Account Settings
SOS
File a report
Add support documents
My case
Schedule a meeting
Message Board
Match a Victim
File a report
Cases Status
My Calendar
Schedule a meeting
Feedback
Message Board
Resources
Cases Status
My Calendar
Schedule a meeting
Feedback
Message Board
Resources
Cases Status
My Calendar
Schedule a meeting
Feedback
Message Board
Application process
Summary Report
Assign supporting team members
Contact Insititution admin
Message Board
System updates
Security checklist
System mantainence
Database management
Webiste admin
Mobile App admin
Message Board admin
User Profile update
Manage Password
Privacy settings

Wireframe:



UI Screen description: App preview for the users

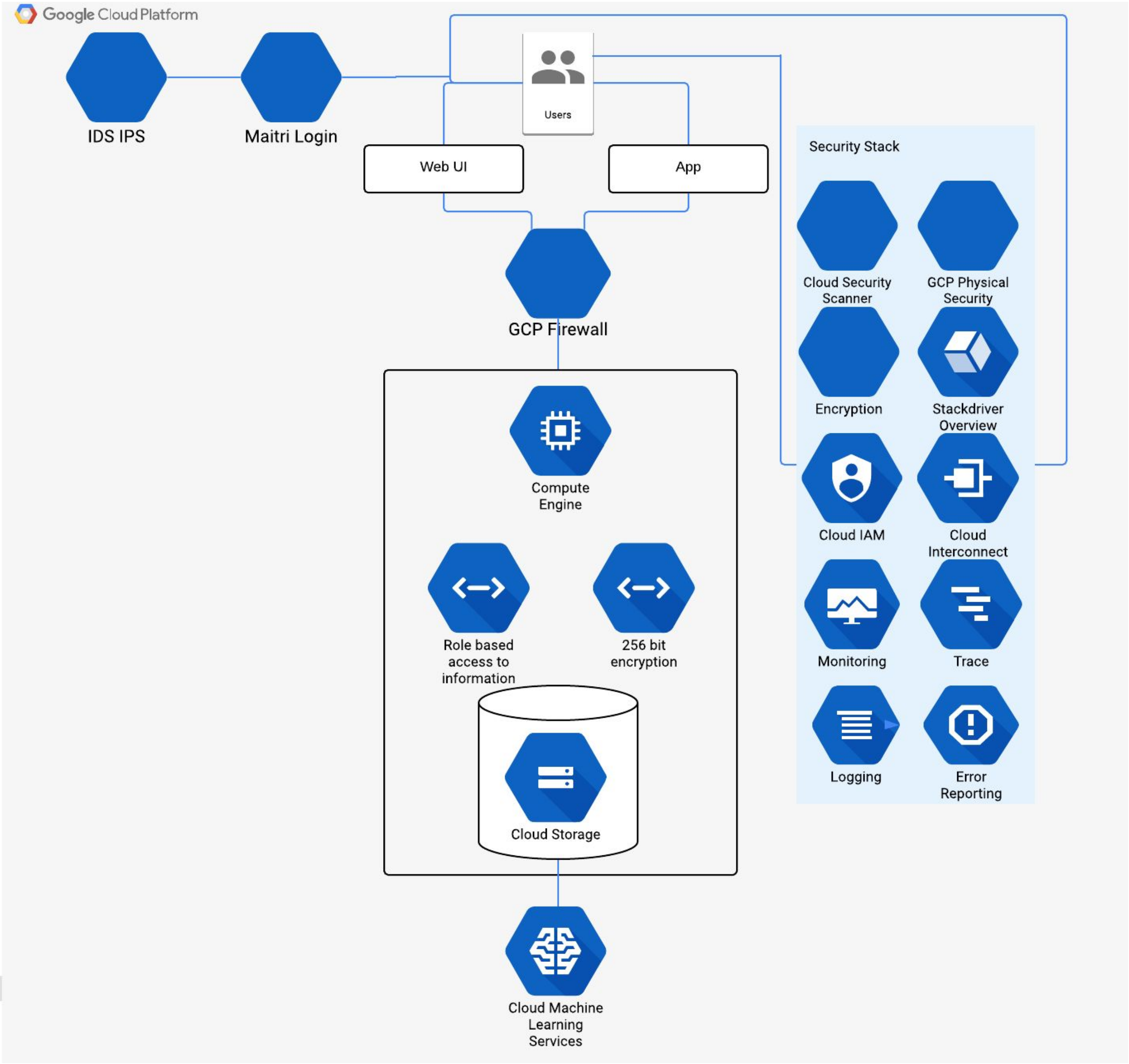
1. Home page – shows the main features (reporting, about us, education, and personal login) for users to navigate their needs
2. Login pages – This screen will be displayed when the user try to login to their own account, and users can choose their roles to enter different login credentials (which will be verified upon login)
3. Student user personal home page – the student user can tap on the SOS for the emergency aid, file a bullying report, check their case status, and schedule the meeting with Maitiri's special supporting team
4. Status check for students – Student user can view their case's process status and mark dates on their calendar.
5. Meeting schedule screen for students -- for support features, the student can schedule the meeting with his assigned volunteer, psychologists and lawyer (and vice-versa). The support team members can add feedback for the meeting.

6. Report filing pages - the user can enter the type of bullying he wants to report from the drop down menu, location, time, related file, victim’s name if possible and description for the bullying incidents
7. Student user’s personal settings screen -- in this page the student can manage their personal file, password and privacy settings
8. Supporting team’s personal dashboard -- the supporting team member can track their cases’ status, meeting schedules, provide feedbacks, related professional resources, password management and privacy settings.
9. Admin team’s personal dashboard -- the Maitiri’s platform admin will have the access to the summary report of the utilization of all the resource available on the platform, system maintenance, database management, user management, settings, insititual partnership and case analysis.
10. Education section -- the page can be entered through the homepage of the mobile app and has a list of related resource for the bullying
11. Mailbox icon will direct user to the messages or notification they received
12. Pencil icon will direct user to edit their profile

4. SECURITY ARCHITECTURE

Security architecture layout is a unified description of the various components and mechanisms that describe how the potential risks and threats to the system are addressed. It also specifies when and where would Maitri apply security controls.

Google Cloud Platform-Based, Custom Security Solution:



The key attributes of the description of security architecture are as follows:

- Components –

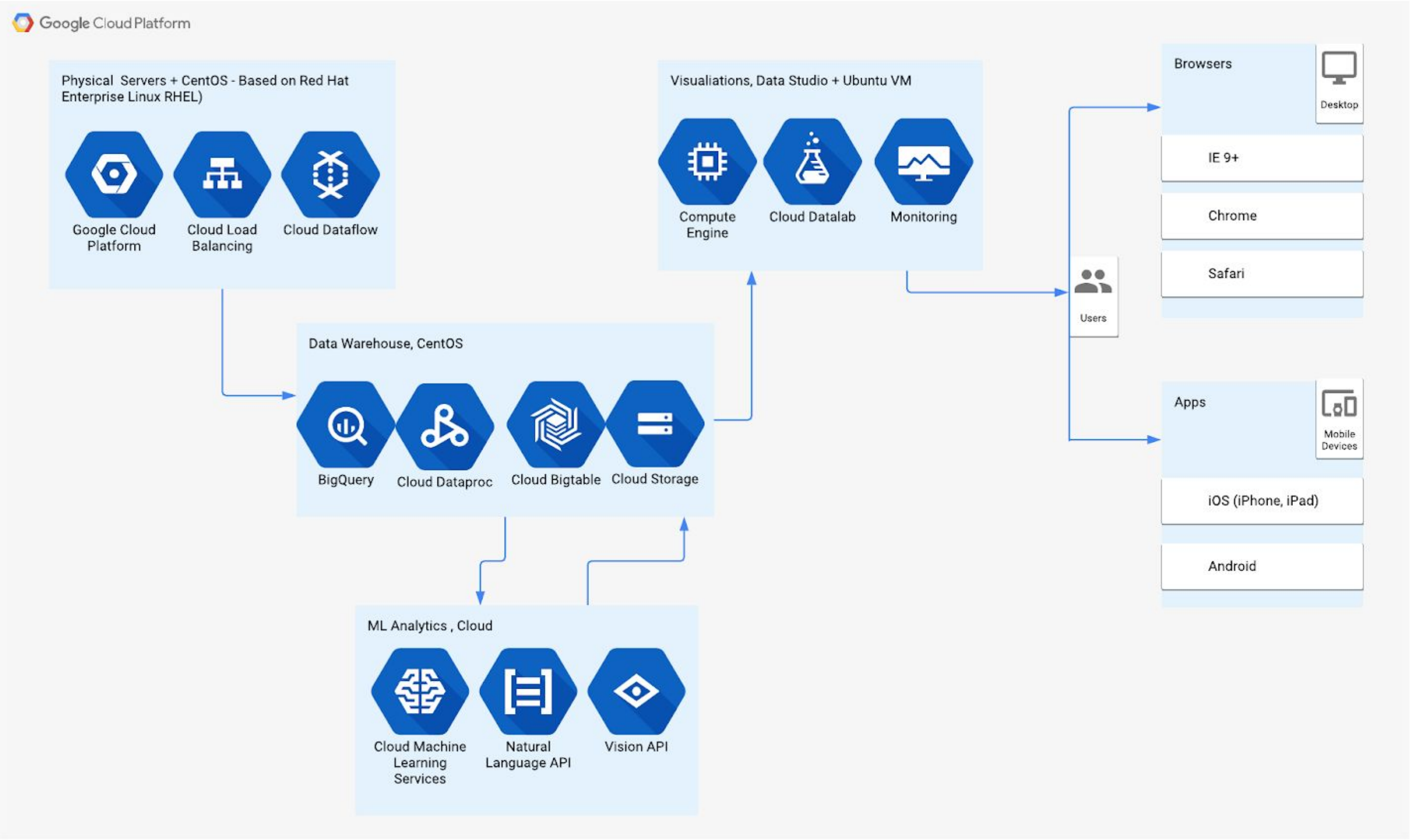
The system employs Google Cloud Platform's provided security services and components. A built-in firewall adds a layer of protection between the users and the servers/compute engines. Broadly speaking, they allow for the following functionalities:

- Identify/Access Manager (Cloud IAM): gives admins fine-grained access control (authentication, authorization settings) and visibility for managing resources.
- Encryption: hashing data prior to being written to disk, and that data at rest is also protected at the level of AES256
- Denial of Service: though scanning + stackdriver, enables an "armor" which provides defense at scale against infrastructure and application Distributed Denial of Service (DDoS) attacks
- Data Loss Prevention: through monitoring + logging, as well as redundancy in backups, tracks the flow of data coming in/out and safeguards against isolated incidents. Also allows for easy auditing and reporting.
- Mechanisms – protocols, monitoring techniques
  - Additional intrusion detection systems (IDS) and intrusion prevention systems (IPS) supported
  - Key management solution: Cloud KMS is a cloud-hosted key management service that manages and saves cryptographic keys with AES256. Cloud KMS is integrated with Cloud IAM
  - Monitoring. Error Reporting, Tracing protocols placed between the user and the app/cloud
    - Monitoring through Stackdriver - provides visibility into the performance, uptime, and overall health of cloud-powered applications on Google Cloud Platform
    - Error reporting through Stackdriver - counts, analyzes and aggregates crashes in running cloud services
    - Tracing - collects latency data from applications and displays it in the console dashboard
- Standards followed – Encryption standards, Legal standards
  - Encryption at rest: by default with Google Cloud Platform, which means that Data is automatically encrypted prior to being written to disk, each encryption key is itself encrypted with a set of master keys, and encryption policies are managed the same way, in the same keystore
  - Encryption in transit: by default with Google Cloud Platform, systems use the HTTPS protocol to communicate over the Internet (TLS connection, BoringSSL, Google Certificate Authority).
  - Secure Emails: for employees of Maitri, disabling download of Word/PDF documents from unknown senders to prevent phishing and malware infection of computers (depending on the role of the employee); turning on automatic identification of SPAM emails and providing regular training opportunities on cybersecurity for staff
  - Regular OS updates and patching on employees' computers as a practice, requiring user passwords to have higher entropy
  - Coppa - The Children's Online Privacy Protection Act (COPPA) is a law created to protect the privacy of children under 13. The Act took effect in April 2000. COPPA is managed by the Federal Trade Commission (FTC). Because we are dealing with information about students, this applies.
  - EU GDPR - The General Data Protection Regulation on data protection and privacy for all individuals within the European Union and the European Economic Area. Although we are not operating in the EU at the moment, we could expand within the next 5 years (bringing the app to the global app stores on iOS/Android), hence, this regulation should be considered in advance.
  - CCPA - The California Consumer Privacy Act of 2018, is a bill that enhanced privacy rights and consumer protections for residents of the US state of California. Applies to us, as we operate within CA. CCPA standards are comparable to GDPR.
- Controls – Access control, User groups, audit logs
  - Handled primarily through the aforementioned Cloud IAM dashboard, allowing granular access controls and defining of specific user groups (i.e. students, school admins, professional aid, etc.)

5. INFRASTRUCTURE/DEPLOYMENT ARCHITECTURE

Deployment architecture covers information on the hardware that would be used (computing, storage, network). It also describes a deployment diagram and shares what form of cables and network selection would be taken up Maitri’s ecosystem to facilitate communication.

Deployment Architecture Diagram: Data Warehouse & Reporting

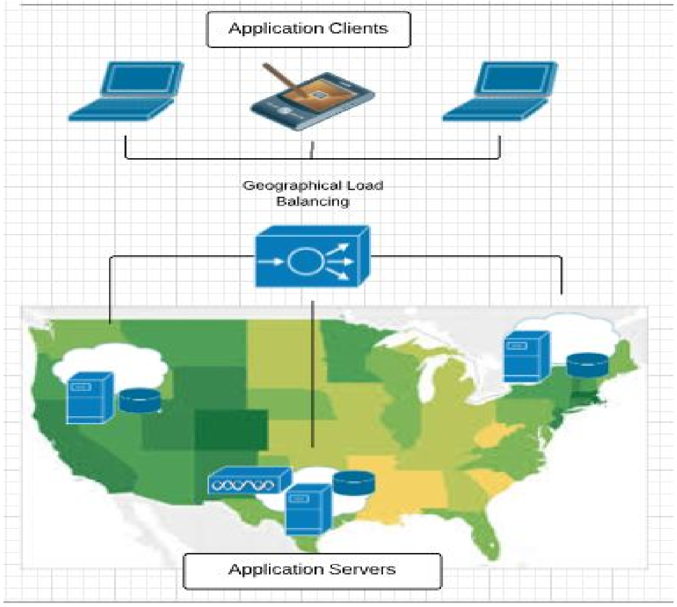


The data warehouse is on a Cloud Dataproc (Spark + Hadoop) System, where BigQuery (RESTful service) is used for Analytics. Further ML processing is handled by the Cloud Machine Learning Services, which saves data into a provisioned datamart (Cloud Storage for static content, Cloud Bigtable for dynamic). The data transformation is done using a Google product, Cloud Dataflow/Dataprep. The transformed data is ultimately consumed via Cloud Datalab for visualizations fed into the apps. The general security layer is provided by the Google Cloud Platform.

Component	Description
Cloud Dataflow/ Dataprep	This is the data transformation stage software running on physical machines in the Google Data Center on Linux (CentOS 7.0 based on Red Hat Enterprise RHEL). This is implemented on Google’s Cloud Platform compute engine (i.e. VM instance with n1-standard-1, n1-highcpu-2, with load balancing), on a virtual private cloud with firewall.

Data Warehouse	It is Google BigQuery for analytics, with Cloud Dataproc (Spark + Hadoop) system for distributed computing. Management machine running on CentOS 7. The data warehouse will have redundancy in the form of backups (in case of single server failure).
ML Analytics	It is using Cloud Machine Learning Services (tensorflow with APIs for inputting NLP - text mining/sentiment analysis - and vision - convolutional neural networks scanning photos on social media) for processing users' data inputs. Results post-processing are fed into a provisioned datamart (Cloud Storage for static content, Cloud Bigtable dynamic content)
Visualization Platform	This is Google Datalab, running on clustered Virtual Ubuntu machines (compute engine instances). The Compute Engine will also be handling business logic components, such as domain services and data access management.
Browsers	These are the major versions of browsers (Web UI) that connect to the visualizations built with Google Datalab.
Apps	These are the mobile apps that connect to the Google Datalab information server to pull in the data and visualizations.

- Geographical distribution diagram
  The aforementioned cloud load balancing may include an added feature of geographical load balancing. The following architecture would be observed:



Whereby different application servers can be adjusted depending on inconsistent traffic from sub-georegions. However, because Maitri plans to only operate within the United States (at least, initially) this feature is not as urgent as if it were to be performed on a global scale (i.e. load balancing for different countries). We anticipate potential user base to be fairly well-distributed across the U.S.

Specifically, we will select Google Compute data centers physically located in Los Angeles, Iowa, and North Virginia to ensure fair coverage of east/mid/west states with minimal latency as possible.

- Trend/SLA/Growth

For Maitri, during the Term of the Google Compute Engine License Agreement, the Covered Service will provide a Monthly Uptime Percentage to Customer of at least 99.99% (the “Service Level Objective ” or “SLO”). If Google does not meet the SLO, and if Customer meets its obligations under this SLA, Customer will be eligible to receive Financial Credits as compensation.

For our customers: the SLA for percentage uptime of app/service will follow industry practices (that is, not as rigorously enforced as the service provider - especially if Maitri will be offered as a free tool for institutions). Unusual spikes in activity are expected to be handled by the load balancer. Expected growth for customer base will result in additional compute engine/server resources and storage capacities being expanded on (yearly).

- Communication details
    - ❑ Type of cable: Ethernet, RJ-45 connections (fiber optic not needed) communicating via TCP/IP protocols.
    - ❑ Network: Standard Tier network, with average performance but improved ingress costs.
    - ❑ Speeds: Per core on each VM, subject to a 2 Gbps cap for performance (Google Cloud).

6. TECHNOLOGY STACK

The technology stack comprises of technology components that allows Maitri to function. This includes description of the programming language, UI Frameworks, Operating system, system software, web components, database, and hardware components.

List of Criteria:

- ○ Cost effectiveness, development speed
- ○ Engineering talent required to develop and support this technology
- ○ Maintainability of the technology and support from vendor
- ○ Scalability required in the future
- ○ Legal/Licensing complexities

Technology stack table:

Architecture component	Technology Choice	Justification
Programming Language	Python	User friendly, convenient for machine learning implementation, light in weight
Database Type	Google Cloud Storage, BigTable	Cost benefit
Operating system (development)	Windows / Ubuntu	Easier to find backend engineers familiar with the platform
API frameworks	Flask, Microsoft Graph REST	Compatibility with Python
Web UI	AngularJS	Realtime data binding and easy UI UX for users to comprehend
Server OS	CentOS 7.0 (based on RHEL)	Free for Google Compute Servers, based on RHEL industry standards
Mobile App UI	PyMob	Allows Python to be used for development of iOS, Android apps

		without requiring programmers to learn other languages (objective-C, Java)
Google Compute Engine Config. Specs	n1-standard-1, n1-highcpu-2	n1-standard is the basic/cheapest tier, ni-highcpu may be needed when called for processing ML under heavy loads
Queries, Real-Time Views	Google BigQuery for analytics, with Cloud Dataproc	Simple compatibility and integration with existing Google services selected, built-in
ML Processing	Google Cloud Machine Learning Services	Tensorflow with APIs for inputting NLP - text mining/sentiment analysis - and vision - convolutional neural networks
IDE	Sublime Text Editor 3	Supports Python, free to install
Browsers supported (user side)	Chrome, IE 9+, Safari	In an MVP Stage, we are focussing on generic and popular platforms
App OS supported (user side)	iOS, Android OS	In an MVP Stage, we are focussing on generic and popular platforms
Persistence	Redundancy Backup	Protects against isolated server crashes/loss of data
Security	Google Compute Platform Services, Encryption	Simple compatibility and integration with existing Google services selected, built-in
Standards	COPPA, EU GDPR, CCPA(Jan1,2020)	Working with students/minors, app will be available in CA and EU in future

Cost Table (Agile framework)

Work Item	Number of resources	Time	Cost
Development cost	6	5 months	$300000 (@$10000 per person per month)
Testing cost	2	2 months	$25000 (@$6250 per person per month)
Maintenance cost	3~2	24 months (tentatively)	$25000 (@$6250 per person per month)
Helpline personnel	4	12 months	$96000 (@$2000 per person)
Google Compute Engine	3 standard, 1 high CPU	24hr/7days wk, 10hrs/5days	$ 87.73 per 1 month
Google Cloud Storage	3TB (1TB per U.S. region)	1 month	$ 76.92 per 1 month
Google Cloud ML Services	Basic training, online predict	3 hours/day for 1 month	$34.30 per 1 month
Google Cloud NL API	Automatic:Sentiment, Syntax,Content classification	90,000 instances per type each month	$117.50 per 1 month
Google Cloud Vision API	Automatic: Label, Explicit Content, Facial, Web classifications	40,000 instances per type each month	$192.00 per 1 month
Google Cloud Dataflow	Streaming 150 hours (batch)	1 Month	$51 per 1 month
Google Cloud Dataproc	4 vCPUs total, 300 hours	1 Month	$5 per month

Google Cloud IAM	-	1 Month	Free (included)
Google Cloud Stackdriver	1GB Monitor of data/profiles	1 Month	$72 per 1 month
Google Cloud BigTable Data	1TB	1 Month	$45 per 1 month
Google Business Email Suite	Unlimited storage, archiving	1 Month	$5 per month
Google Cloud Pub/Sub for simple notifications	First 10GB	1 Month	Free (included)
Certification cost (SSL)	-	1 year	~$1000

7. Glossary

Term	Description
NFR	Non-Functional Requirement(s)
ML	Machine Learning
ERD	Entity relationship diagram
IAM	Identity and Access Management
API	Application programming interface
IDE	Integrated development environment
GCP	Google Cloud Platform
IDS/IPS	Intrusion Detection and Prevention Systems
IaaS	Infrastructure as a Service