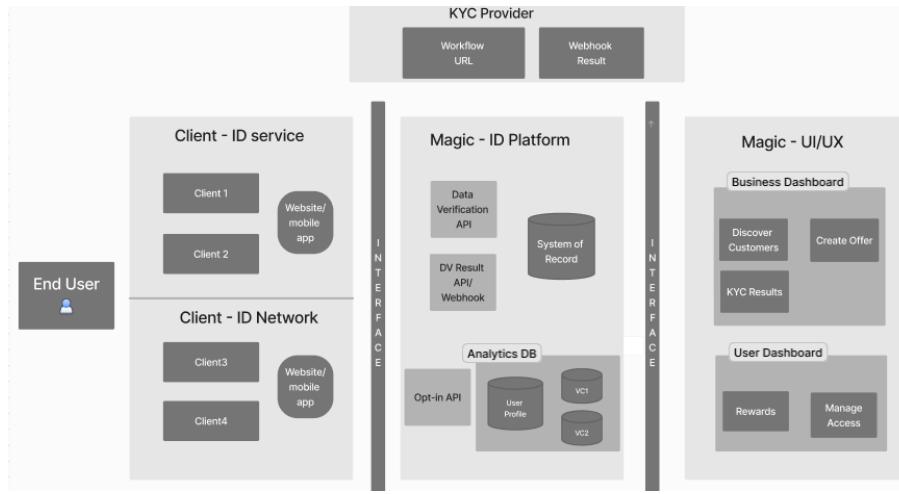
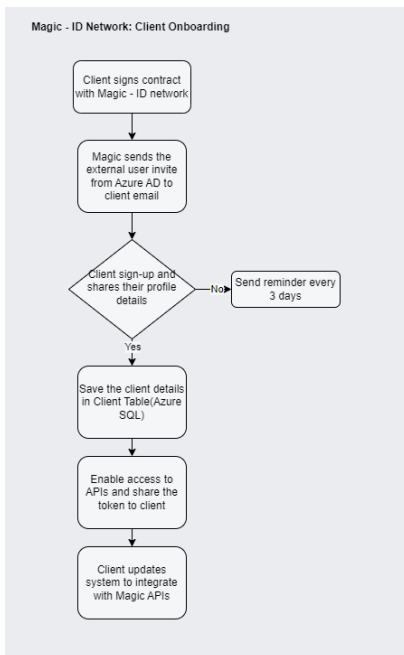


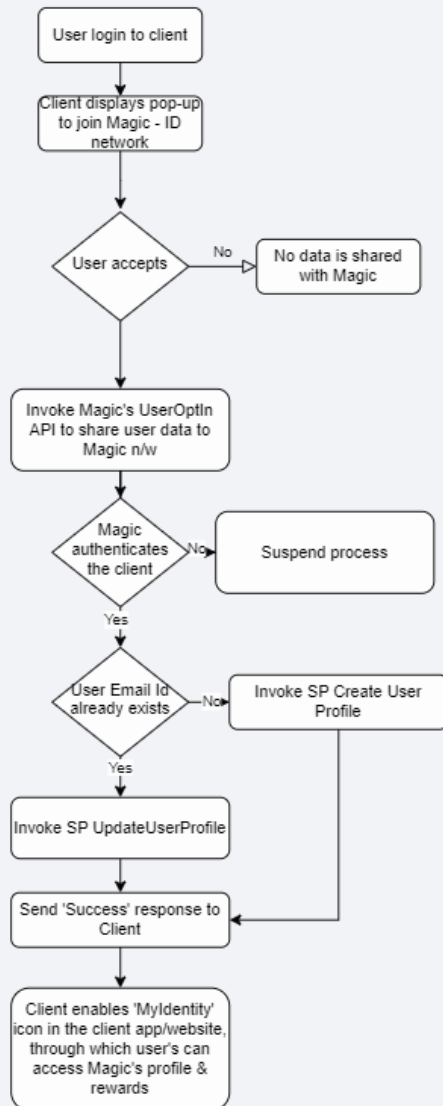
Draft Architecture



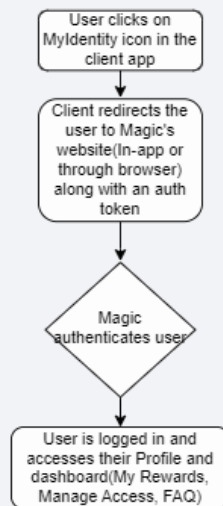
Use-cases:



Magic - ID Network: User Onboarding



User accessing their profile in Magic



Technology Decisions & Assumptions:

Component	Beta	Comments
Cloud Infrastructure	Azure	
Business Dashboard	Power pages, Javascript, embedded Power BI	Already implemented
Client Authentication	Azure AD - external user	
System of Record	Azure SQL	Already implemented basic structure (Users, Client, Rewards tables)
Analytics DB	Azure SQL	
User Dashboard	?	Please provide your suggestion Option 1: Azure-hosted custom web app Option 2: Power page which uses a single premium license on the backend through the Power app to execute transactions. Not sure if it provides RBAC Option3: Any other low code front end
User Authentication	Tokens?	Please provide your suggestion. Prefer if the client can share the token when they redirect?
Device Authentication	?	Please provide your suggestion , if we need any additional authentication for the client or user(certificates, keys, etc.,)
UserOptIn API	?	Please provide your suggestion Azure function or Azure logic app.
Secret store - Password, certificates	Azure key vault	Please provide your suggestion
Build and deploy	Azure DevOps	AzureDevOps or GitHub We have free GitHub Enterprise License for 6 months and Azure free credits(not sure if that covers Azure DevOps). Please provide your suggestion
Monitoring		Please provide your suggestion , if we need anything at this stage?
API Config	Json	Please provide your suggestion. Prefere to keep it simple
RBAC for users, clients, developers and admin		To be implemented
Data verification API	Azure logic app	
Data verification Result API/webhook	Azure logic app	

NFR - Beta

- Secure communication channel
- Basic GDPR compliance - PII user data (text only)
- Zero trust security architecture
- RBAC for client and user data segregation
- As it is a Beta version, the solution should be simple to manage/change without relying on developers unless for major modifications.
- 2 environments - Sandbox & Live
- SQL - RBAC for the internal team (Super Admin, Support)
- Contributor role and permission for external developers (Azure & SQL)
- Up to 5 clients with up to 500 users - roughly 10,000 transactions/month/azure API
- UK-based clients and users. (Future expansion to EU and global)
- Minimum cost and use free credits/licenses wherever possible.
- Microsoft for Start-up credits with the below benefits:
 - Azure credits - \$150,000
 - M365 Business edition - 10 seats
 - Power apps premium license - 10 users for a year
 - 6 months GitHub Enterprise License

Out of Scope

- Business Dashboard - Power page development is already complete.
- Azure AD external user setup for client to access Power page dashboard is already complete
- SQL server and the basic structure of tables(User,Client,Rewards) with sample data are already set-up.
- Ignore the Data verification API and webhook, for later
- Ignore the analytical DB. Basic SQL DB that has been setup is good enough for beta

Ask from 55(Immediate)

- Review architecture, tech stack table, and use cases. Share any Immediate Queries & Feedback, especially to start the UserOptIn API implementation.
- Provide suggestions as requested in the comments section of the tech stack to help finalize pending tech choices
- Provide estimated cost and timelines for the implementation of UserOptIn API along with the required cloud security configuration apart from what's already complete(Out of scope)
- UserOptIn API - develop the API according to specification(refer to use case diagram Magic ID Network – User Onboarding)
-
- RBAC (Azure components & SQL) as explained in NFR

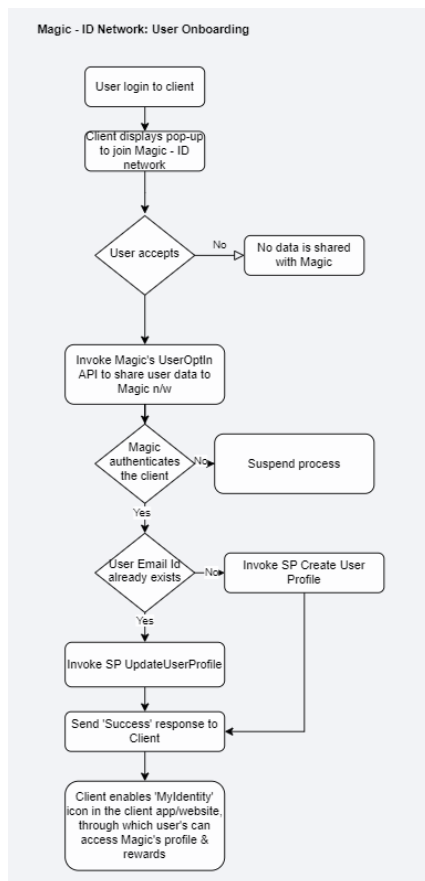
- Any other suggestion in architecture & tech choice based on NFR and our verbal discussion which I have overlooked?

Ask from 55(Future, subject to agreement)

- Cloud running cost and optimization suggestions
- DevOps, monitoring solution and environment setup - Sandbox and Prod
- Development of other APIs - Rewards API, Data verification API, Data verification Result API/webhook
- Development of Front end - User dashboard
- M365 - User Admin and device policies

UserOptIn API specification:

Commented [A1]: lakshmi.nagaraja@fiftyfivetech.io FYR, added the spec and mapping here



- SQL Table Users already exist within MagicIDNetwork database.

- Below is the mapping from API to SQL table – Users

Target	Input	
	API - Request Object	Conditional map
Users\UserName	VerifiedLegalName	
Users\DateOfBirth	VerifiedDateOfBirth	
Users\Address	VerifiedAddress	
Users\Email	Email	
Users\Phone	Phone	
Users\ClientId		Retrieve ClientId based on authentication (based on how authentication happens)
Users\ClientUserURN	ClientUserURN	

- Authenticate the request originating from the Client. Clients are already configured in Azure AD for their web app(Power pages) access. But not yet set up for API integration and authentication and authorization.

Definition of Done:

- Documentation and walkthrough of DevOps process, how to modify/deploy the api.
- Application documentation with the list of components created(Azure function, SP and anything else)
- Appropriate authentication, authorization and RBAC setup, relevant to UserOptIn API
- Code reviewed for best practices and Secrets stored in Key vault, check in to Azure DevOps.
- Azure DevOps setup.
- 2 separate Config files and pipelines to be created. For Dev and Prod.
- Unit Tested.

Acceptance criteria:

Postman call to UserOptIn API with Sample user details and valid client credentials should be traced in SQL table – User. ClientId should reflect the authentication.

Postman call to UserOptIn API with invalid credentials to be rejected.