Discovery Tool

An idea management system

Team 3

Project overview
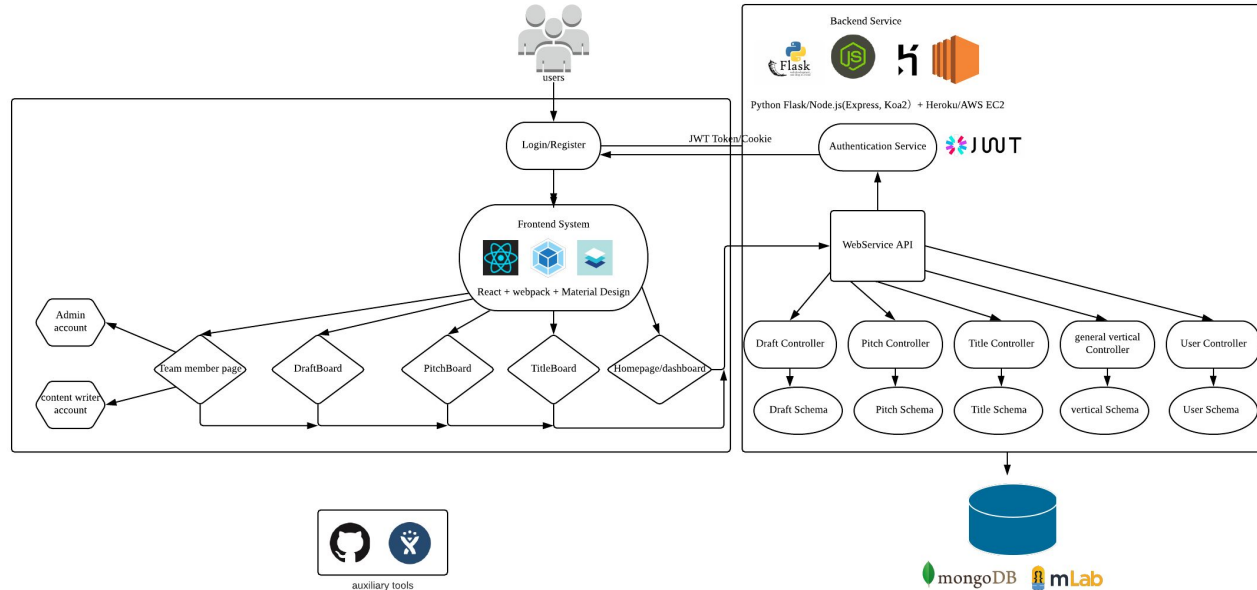
Discovery Tool is a web app based **idea management system**. It serves as an **internal tool** to System1. It provides a solution to **help System1 admins and content contributors to streamline, manage, accelerate their workflow**. It gives the admins necessary tooling to quickly **scale the team as well as content production**. Overall, the system supports the team to work cross-functionally, leveraging resources to meet needs as efficiently as possible.

Stakeholders	Initiatives	Value Propositions	Beneficiaries
-System1 Super Admin -System1 Admins -System1 Contributors -System1 Maintainer	-Admins and contributors correctly use the app -Maintainers maintain the whole system after handoff occured -Admins and contributors are responsive to the different stages of idea formulation -Developers implement system to mimic and streamline existing process on idea formulation	-Modernized frontend and backend technology -Streamlined process with real-time feedback -Ease of scaling the project and team size -Monitored and transparent team performance, progress, and goals -Improved organization, tracking, and resolution of content disputes	-Super Admin -Admins -Contributors -Maintainer

Technologies

- Frontend: React/Redux + webpack + Material Design
- Backend: Python Flask/Node.js(Express/Koa2)
- Database: mLab(mongoDB)
- The whole architecture is based on RESTFul API, MVC pattern and microservices, realize frontend and backend separation
- High performance on scalability, availability and usability based on mature cloud services(AWS EC2 Auto Scaling, Heroku distributed dynos, mLab DAAS features, etc)

NDI/NCS Products	Purposes	Comments
React	Front-end Framework	Positive Points: ● Better performance ● Better compatibility among web browsers ● Better ecosystem Negative Points: ● Learning curve
Flask	Back-end Framework	Positive Points: ● Open source ● Light Weight ● Familiar by the client Negative Points: ● Have to use third party extensions

NDI/NCS Products	Purposes	Comments
MongoDB	Database	Positive Points: ● Flexible and schemaless ● Support large scale of data Negative Points: ● Lack of community support with respect to Flask integration
AWS	Cloud Server	Positive Points: ● Easier than building local server ● Cheaper when having small data scale ● Safe Negative Points: ● More complex setup compared to alternatives (Heroku)

Backend Demo

UI Live Demo

Login Page

Auto redirect to admin dashboard or content writer homepage

Content Contributor Homepage

Work progress indicator
indicate assigned vertical, and pitch

Sidebar
nav to other board

Pitch table
submitted pitches

Content Contributor Draft Board

Draft Table
display all pitch draft's detail
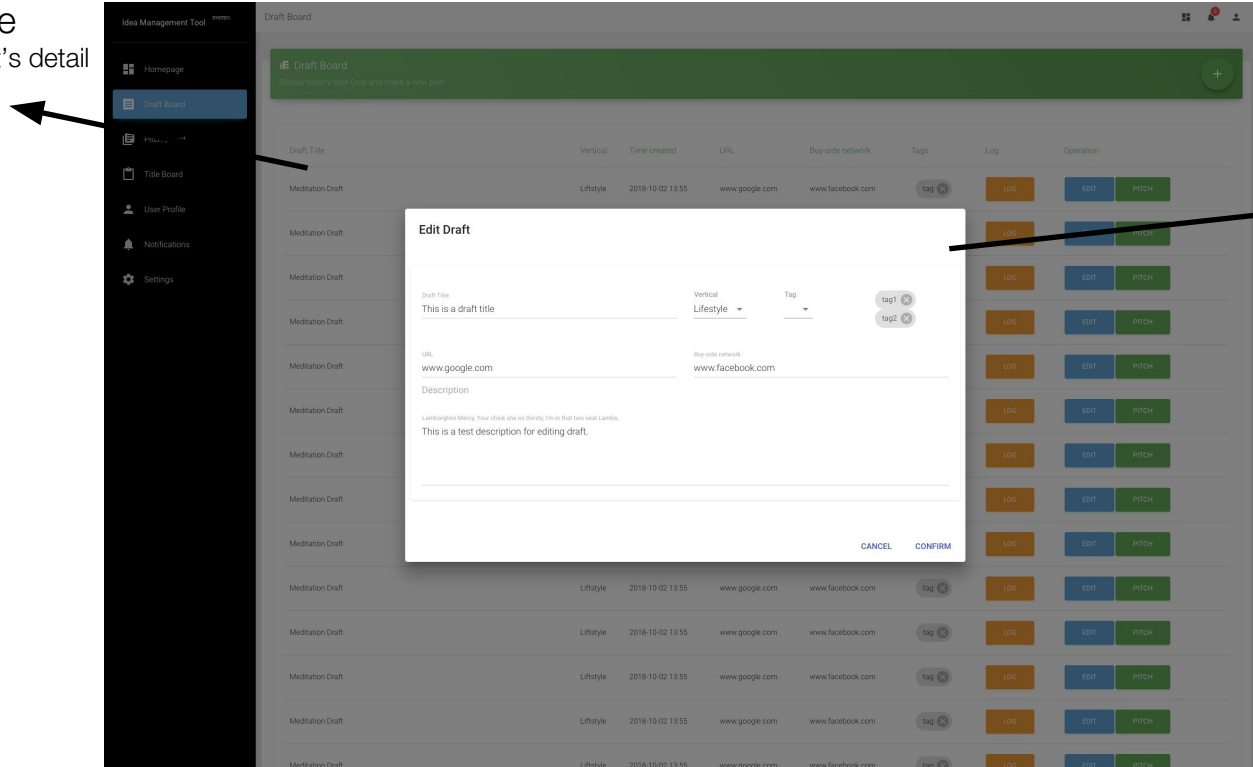
Edit Draft
edit a draft's detail
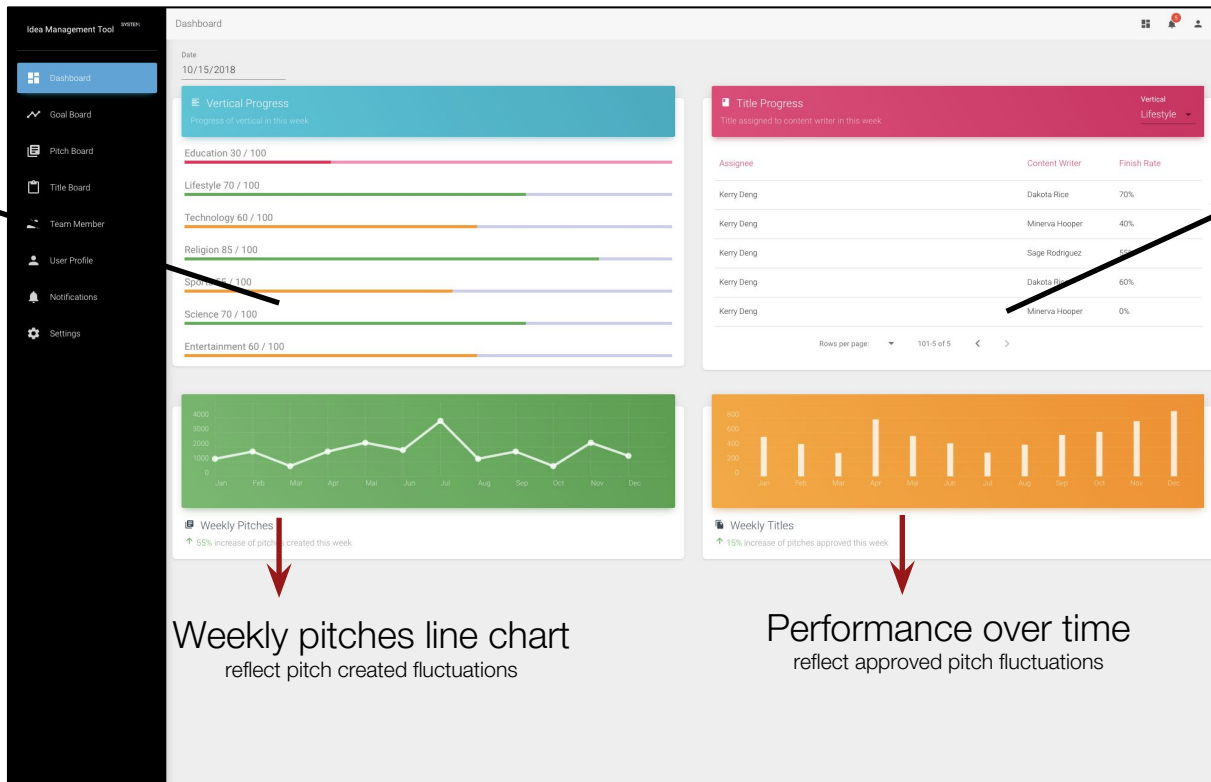
Content Contributor Global Pitch Board

Pitch table
display all pitches submitted by current user

Admin Dashboard

Vertical(pitch) overview
display all verticals work progress

Team Member Performance

Weekly pitches line chart
reflect pitch created fluctuations

Performance over time
reflect approved pitch fluctuations

Super Admin Goal board

Goal board
display weekly targets

New goal creation
create a new goal for a specified week

Admin Global Pitch board

Pitch table
admin can view, evaluate global pitches

Admin Team member List

Team member table
display all users profile
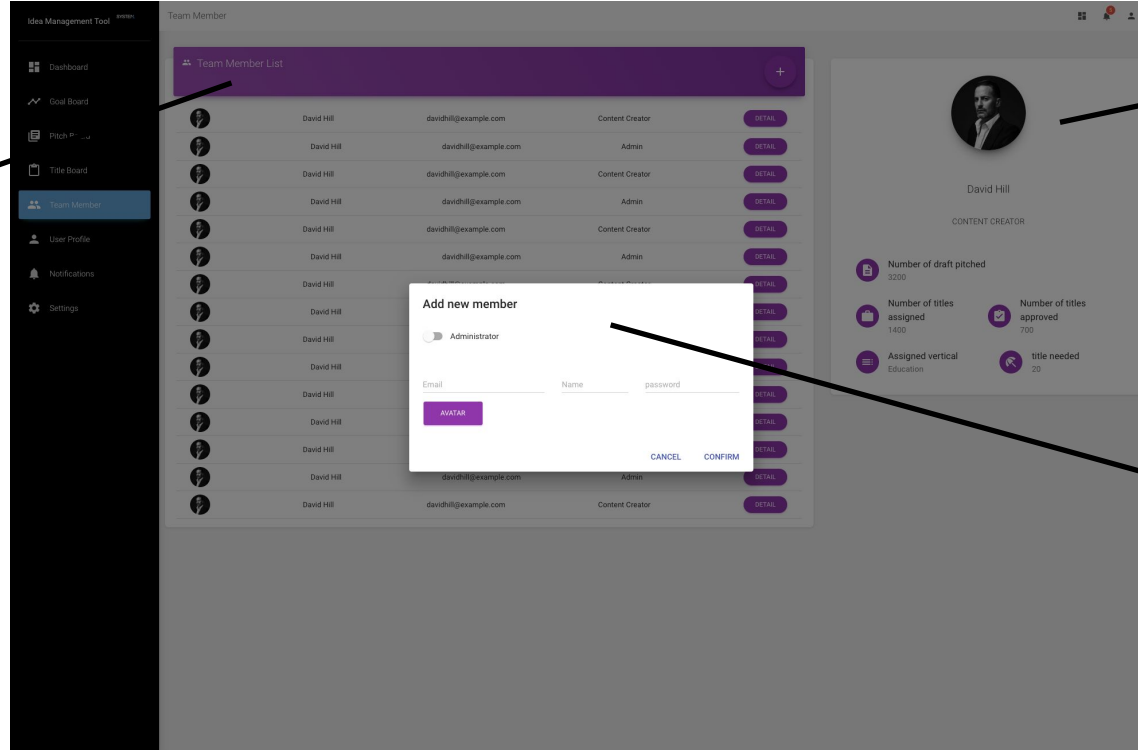
User profile detail
show a specific user's work progress

New member
the only way for admin to add new users

Lesson Learned

Product Development
- The client does not always know what they want (ikiwisi)
- All success-critical stakeholders should be involved as early as possible
- Feedback loops should be shortened where possible

Process
- Willing commitments yield better work
- Openly stated and publicly hosted commitment tracking can be used as an effective accountability and motivation mechanism

Moving Forward…

- Implement changes based on feedback from most recent client meeting
- Finalize UI and backend API
- Start discussions with the maintainer