Ask Your Repository

Bachelorproject Summary

Motivation

- Design Thinking teams record their intermediate results by taking photos of whiteboards and prototypes thereby producing vast amounts of image data
- Images are stored on multiple platforms by different team members and thus not quickly accessible for everyone when needed
- Images rarely organized and if they are, searching big folder structures still takes time
- Design Thinking process is iterative and relies on using previous results

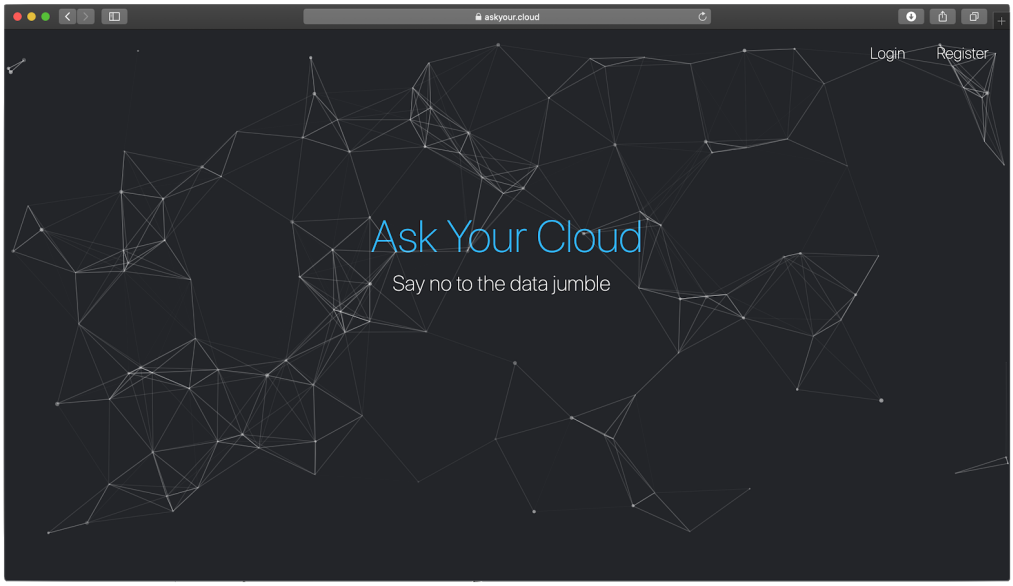
Problem Statement

- If images are not easily accessible the process is slowed down
- Or the knowledge contained in the images is lost because they won't be used again

Solution

We built Ask Your Repository, a web-application that reduces the effort of organizing the images and helps Design Thinkers finding them again quickly later on in the process.

In order to do that our application uses tagging to make images searchable, enhanced by machine learning and an easy to use search interface with voice assistant integration.

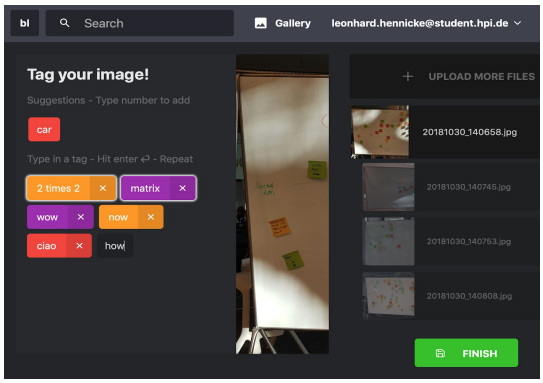


The Project is currently deployed as 'Ask Your Cloud' on askyour.cloud - a domain owned by our team.

Features

Tagging instead of folders

- Fast and intuitive tagging, users can upload and tag several images at the same time
- Implicit grouping to replace folders
- Suggested tags
- Automated tagging using machine learning



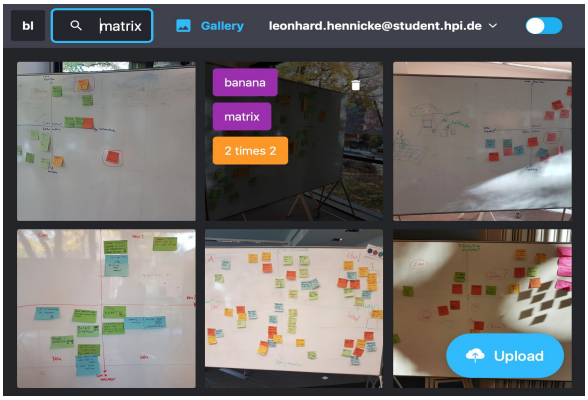
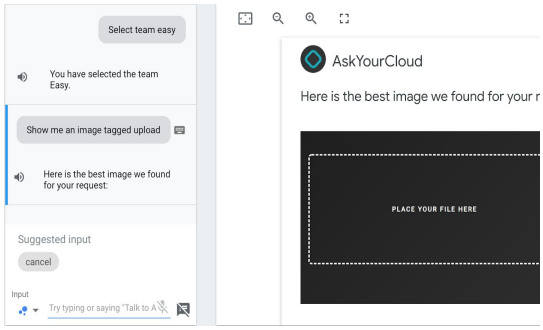


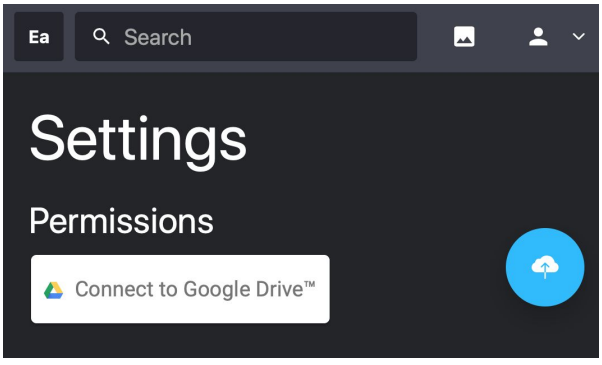
Image search

- Integrated search engine to search images by tags
- Search queries enriched with Natural Language Processing

Google Assistant integration

- Image search even more natural
- Team can ask google assistant
- If Presentation Mode is turned on, images can be searched via phone and displayed on larger devices simultaneously

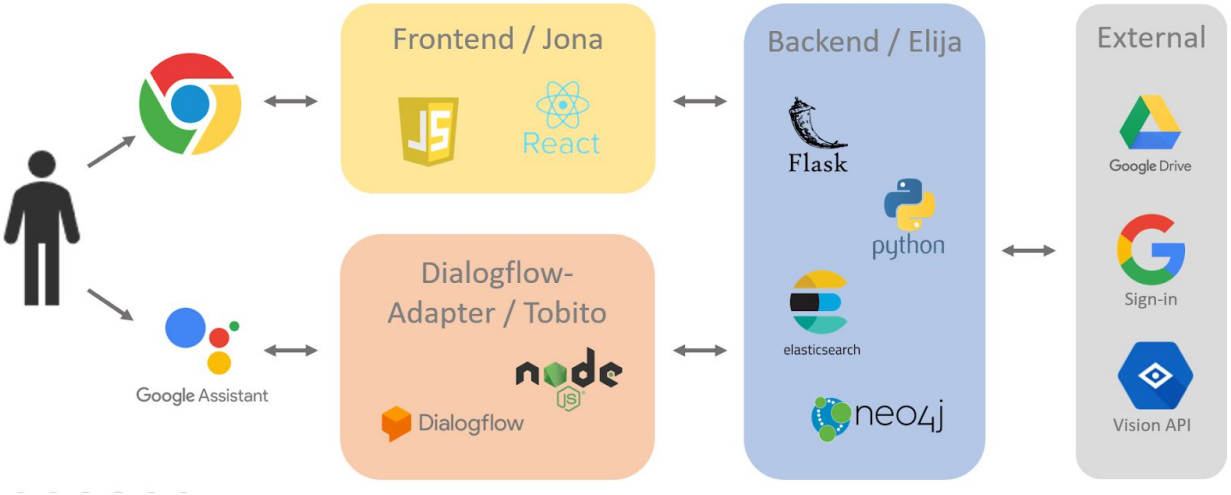




Google Drive Synchronization

- Option to connect a team to a Google Drive folder
- All images from the Google Drive and our Application are kept in sync

Architecture



Frontend

React Web-App for user interaction to visualize data coming from the backend. User-Interface for tagging, searching and displaying images.
https://github.com/hpi-sam/ask-your-repository-web

Dialogflow-Adapter

NodeJS API for handling the voice queries coming in via Google Assistant. This service implements the behavior of the voice assistant.
https://github.com/hpi-sam/ask-your-repository-dialogflow-adapter

Backend

Flask API that manages the data and requests to external services. Handles the automatic enrichment of the image tags, the tag suggestions and manages the Databases.
https://github.com/hpi-sam/ask-your-repository-api

Deployment

- Currently deployed on askyour.cloud, a domain owned by our team
- The deployment is done entirely with docker. There is a docker configuration file (docker-compose.yml) for all services. Tutorial at https://docs.docker.com
- Continuous integration is done with CircleCI. Tutorial at https://circleci.com/docs/
- For more information visit https://github.com/hpi-sam/ask-your-repository-docker
- For the overall project documentation visit https://github.com/hpi-sam/BP2018HG1

License

The whole project is open source under the MIT license, which means that the software can be copied, modified and used privately as well as commercially without restrictions and without warranty.

Bachelor Theses

The goal of our theses was to explore other, more innovative approaches for the improvement of our tool. The explorative work on our theses allowed for more complex and creative solutions than our previous process. However, this means that not all thesis results can be integrated. The relevant code bases for each thesis will be linked in the documentation.

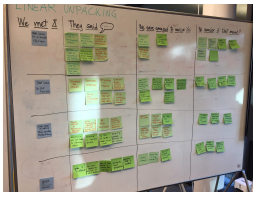
Detecting Design Thinking Methods [...]
(Leonhard Hennicke)

Motivation

- Images have to be tagged in order to be searchable
- Tags have to match what users remember about the image

Problem Statement

- Tagging by Design thinking method would be valuable, as users might remember the method better than that content
- Automatically generated tags cannot capture this as of now
- Would be very repetitive manual work



Solution

- Implement automatic tagging for Design Thinking methods (A and B)
- Explore two approaches: machine learning and crowdsourcing

Evaluation

- Crowdsourcing can be accurate (>90%), but only with high pricing, which too costly in the long run
- Too little training data available for machine learning to be accurate enough (reliably about 75%)

Integration

- Both approaches implemented, but not yet integrated together in Ask Your Repository
- With extra funding, crowdsourcing could be used at scale to generate training data for machine learning

Sketch based Image retrieval
(Jascha Beste)

Motivation:

- Keyword search relies on a well tagged image database
- Images are often not sufficiently tagged by users
- Automatic tagging is often not of high enough quality to allow accurate search by keyword
- Users may remember what an image looked like but not how it was tagged anyways

Problem statement:

- Manually tagging all the images with descriptions would be tedious work

Solution

- Use sketches by users as query instead of text
- Find images that are visually similar to the sketches of users
- Find a descriptor, meaning a method to effectively calculate the similarity of images and sketches
- In addition to a standard image descriptor (histogram of oriented gradients), two different improvements were tried: Sliding window and Image Pyramid

Evaluation

- Overall retrieval rate was “okay” (~50%-70% hit rate in searches by sketch).
- Both algorithm improvements yielded no significant improvement in retrieval quality
-

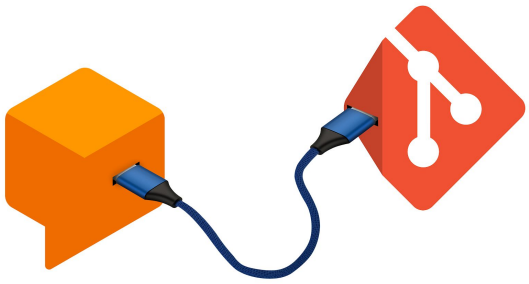
Integration

- Research codebase exists
- Not ready for usage in Ask Your Repository production version, would require further performance optimization

Supporting Iterative Development of Voice Interfaces using a DSL
(Arne Zerndt)

Motivation
- Version Control is needed for Iterative development of voice assistants

Problem statement:
- Source control for voice assistants is currently insufficient
- JSON representation of voice assistant configuration can be saved in version control of Dialogflow-Adapter source code but is close to unreadable



Solution
- Design a Domain Specific Language (DSL) for configuring voice assistants, which allows to more efficiently describe voice assistants
- DSL has much better readability than JSON version

Evaluation
- The DSL produces configuration files that are (on average) more than 85% shorter, compared to the JSON representation
- The DSL is - by all applicable standards - more readable than the JSON version

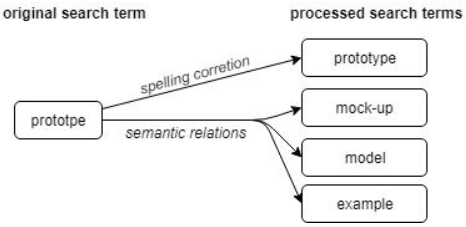
Integration
- DSL is integrated into deployment and the voice assistant is working
- Usage of the DSL is recommended only to development teams willing to familiarise themselves with its inner workings, otherwise the standard Dialogflow UI by Google (while tedious to use) might be easier to start with

Optimizing search queries for the retrieval of tagged images
(Luise Benkert)

Motivation:
- The only way for users to retrieve images is through text search
- Even slight deviations can hinder the user's search terms from being mapped to a related image tag

Problem statement:
- Users may search for an image with wrong, misspelled, too vague or generally unsuited search terms
  → the image results will not be ideal
  Search tools should avoid this.

Solution:
- Using Natural Language Processing to transform and enrich the user's search terms
- Methods used: Lemmatization, Word Stemming, Spelling Correction, Semantic Relations, Filtering
- Example:



Evaluation:
- Search quality was evaluated by using well-accepted measurements on two image sets (30 images each)
- Processed search queries perform on average 10% better than non-processed ones
- Processed queries perform so well that there is no need manual tagging (automatically generated tags were sufficient for all test sets)

Integration:
- Version with aforementioned improvements is fully implemented
- Due to its scientific purpose, code may not be fully standard-compliant
- Feature can easily be integrated, but should be refactored and optimized

Implementing a recommendation system for single-user image collections
(Erik Ziegler)

Motivation
- Users want to explore their image collection
- Providing a way to navigate through vast amounts of images needed

Problem Statement
- Recommending images to the user that they would likely want to see next or that are relevant to her
- Recommendation systems usually rely on large amounts of users and items: similarly between items is computed by checking if like-minded users liked/purchased the item
- In this case, there is only one user and only implicit feedback

Solution 1: Tag-Based Approach
- Similarity between items (images) is computed by using the image tags, meaning images that share more tags are more similar to each other
- By taking the nature of the images the user already looked at into account, recommendations are computed

Solution 2: Property-Based Approach
- Images have properties like date, locations, image content etc.
- For each property, similarity between images is calculated according to that property
- Recommendations are computed based on the overall similarity

Evaluation
- The algorithms were compared in an offline experiment regarding their prediction accuracy
- Both recommenders had sufficiently high precision for most test cases with the property-based approach performing slightly better

Integration
- Working research code base
- Ready for integration in Ask Your Repository

Handwritten Text Detection on Whiteboard Images
(Adrian Steppat)

Motivation:
- To search for whiteboard images it would be useful to extract text

Problem statement:
- Currently available solutions to extract text from images are either not good enough or costly

Solution:
- A segmentation pipeline that can segment a whiteboard image and detect Post-its, originally developed as a preprocessing step
- A pipeline to detect and recognize text which currently does not use the segmentation pipeline because it can already detect small text on Post-its

Evaluation:
- Post it segmentation was evaluated on a dataset of 47 images
- Text detection and recognition results are good enough to extract some keywords, but the Google Vision API achieves better results

Integration:
- Current version is not fast enough to be production ready (>20 sec)
- Solution will be developed further
- Could be integrated as an additional service, I will keep in touch regarding integration

User Testing

Results of our two test runs with D-School participants in the First Track and Advanced Track:

- Many minor bugs have been reported and fixed
- Most feature requests were for usability features
- A lot of participants wanted the ability to interact with images and edit them in our tool

➔ Users were initially excited about the Design Thinking specific features and the advantages they provide.
➔ Our tool (being a work in progress) lacked in usability compared to professional platforms, so that users ended up switching back to those for convenience.

Conclusion

Ask Your Repository provides numerous benefits to the users that are specific to the Design Thinking process. Used in that context, it can facilitate the process significantly. However, as with every new tool, using it comes with the extra effort of learning how to use it. This is made even more complicated by the fact that Ask Your Repository is work in progress and thus does not have the refined usability features of other professional platforms created by big companies with lots of funding. In our test runs we found that the best way for us to counter this was to work closely together with a few voluntary users, so their issues can be handled personally and as quick as possible in order to minimize any frustration that they might otherwise cause. In the long term tough, improving the integration with Google Drive and other professional platforms further could minimize the switching costs, thus making Ask Your Repository the default tool for future D-School teams.

Future Work and Ideas

- Decide which of the thesis resuts are relevant, improve the respective code and integrate them
- Implement an admin panel
- Expand on the Design Thinking specific image recognition (for recognizing more Design Thinking methods, maybe also phases etc.)
- Track the evolution of ideas through different images and thus throughout the process in the right order e.g as a timeline
- Digital whiteboard as a way to edit images, put them together and add comments
- Include other data formats (PDFs, videos, audio data etc.) and make them searchable
- Implement graph-like gallery visualization that shows connections between images
- Visualize design cycles and make images searchable by their respective cycles