



# Art Museum App

Focus on User Experience Through Art Discovery and Database Services to the Museum Industry

May Capstone Presentation  
Ryan & Hasha

# Agenda

Introduction -----	4-8
Timeline -----	10
Objectives & Constraints -----	12-14
Hasha - App features and implementation ---	16-25
Ryan- Database features and implementation	26-35
Thoughts & Future -----	36-38
Live DEMO -----	39-40
Appendix -----	42



# Introduction



**Hasha Drabek,  
Nanotechnology**



**Ryan Lockard,  
Bioengineering**



Ryan Lockard, rtlockar@iu.edu  
BS Bioengineering, Minor Math

Languages:  
C, Python, SQL, C++, MATLAB

Research Skills:  
Physiological Signal Analysis,  
Statistical Analysis,  
CAD Design

Significant Courses:  
Modeling & Simulation,  
CAD & ENGR Drawing,  
Intelligent Systems I/II



Hasha Drabek, hddrabek@iu.edu  
BS ISE, BS Physics, BS Maths

Languages:  
Matlab, C++, Mathematica,  
Shell Script, Python

Research Skills:  
CAD Design, Literature Review,  
Statistical Analysis, 3D Printing

Significant Courses:  
Rapid Prototyping,  
Sustainable Product Design,  
Simulating Nanoscale Systems

# An App with Databasing Services for Museums for User Art Discovery



**1** The user opens the app and uses the QR scanner to scan the QR code by the artwork, taking them to a custom website



**2** The QR code directs the user to a custom website that the art museum can easily upload content to and generate more URLs without tech savviness



**3** Users can scroll through art on the front page and discover art to make the excited about the museum, similar to discovery of content through Instagram

An App and Database Service to be used by Museums and Museum Goers where individuals can use the app to discover art and scan QR codes by the art to be taken to a custom website which the museum can create with our program with ease without technical knowledge



We must cater to both museums (B2B) and the public museum goers.



# Timeline (Spring)

Task/Month	Jan	Feb	Mar	Apr	May
Photos for Database					
Database Development					
App Development					
Figma App					
Website Development					
PythonAnywhere					
GUI Program					
Bravatize the App					
Integration					
Fully Impl App					
Final Presentation					
Final Documentation					



# Objectives & Constraints

# Objectives Ranking [priority]

# Constraints

## Phone Application

- Camera integration (for scanning qr code) [3]
- Easy to Use [1]
- Compatible with IOS and android devices [4]
- Extendable for different museums [2]

## Database Server Interface

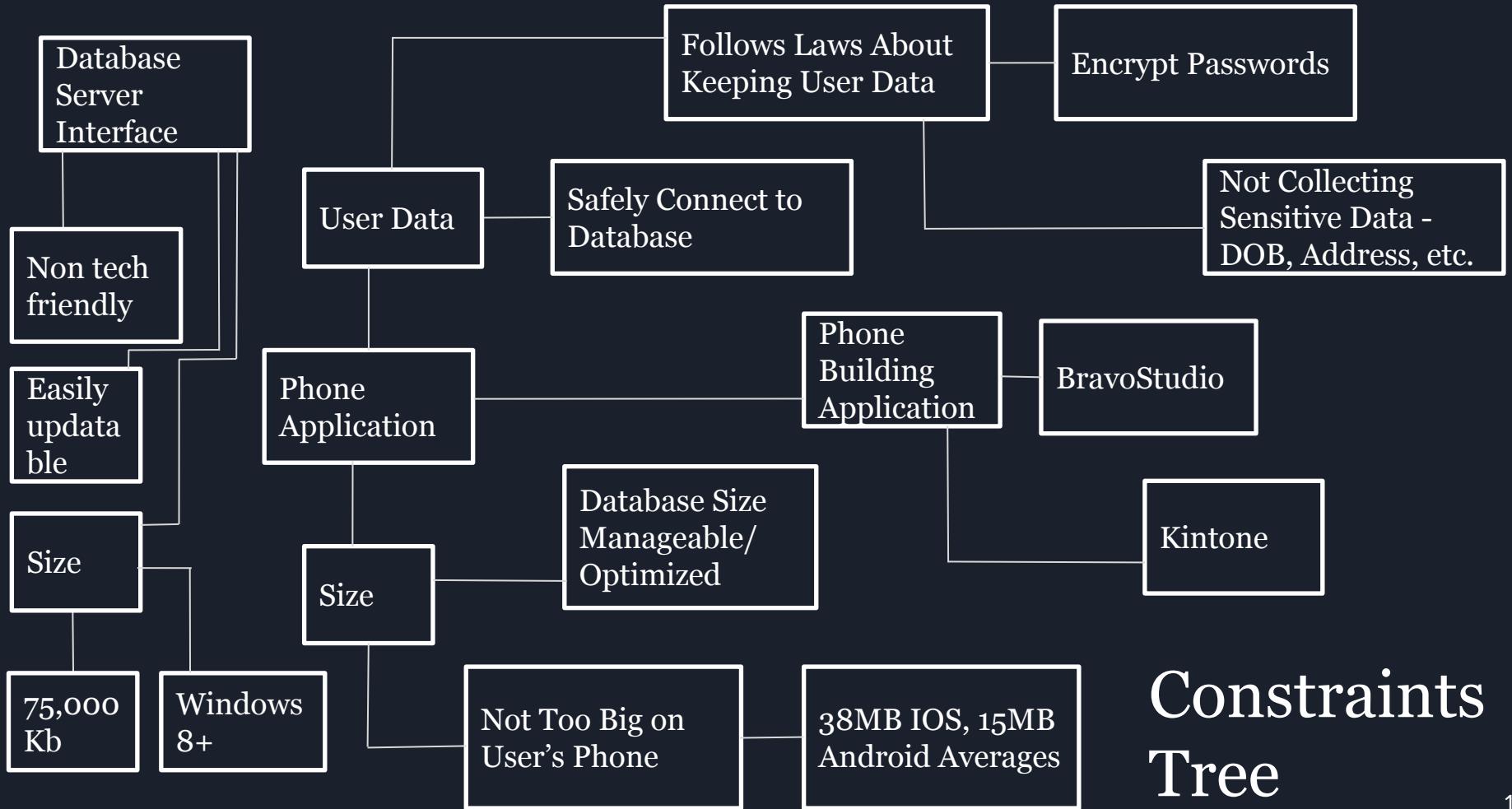
- Easy use for a non-technical person

## Phone Application

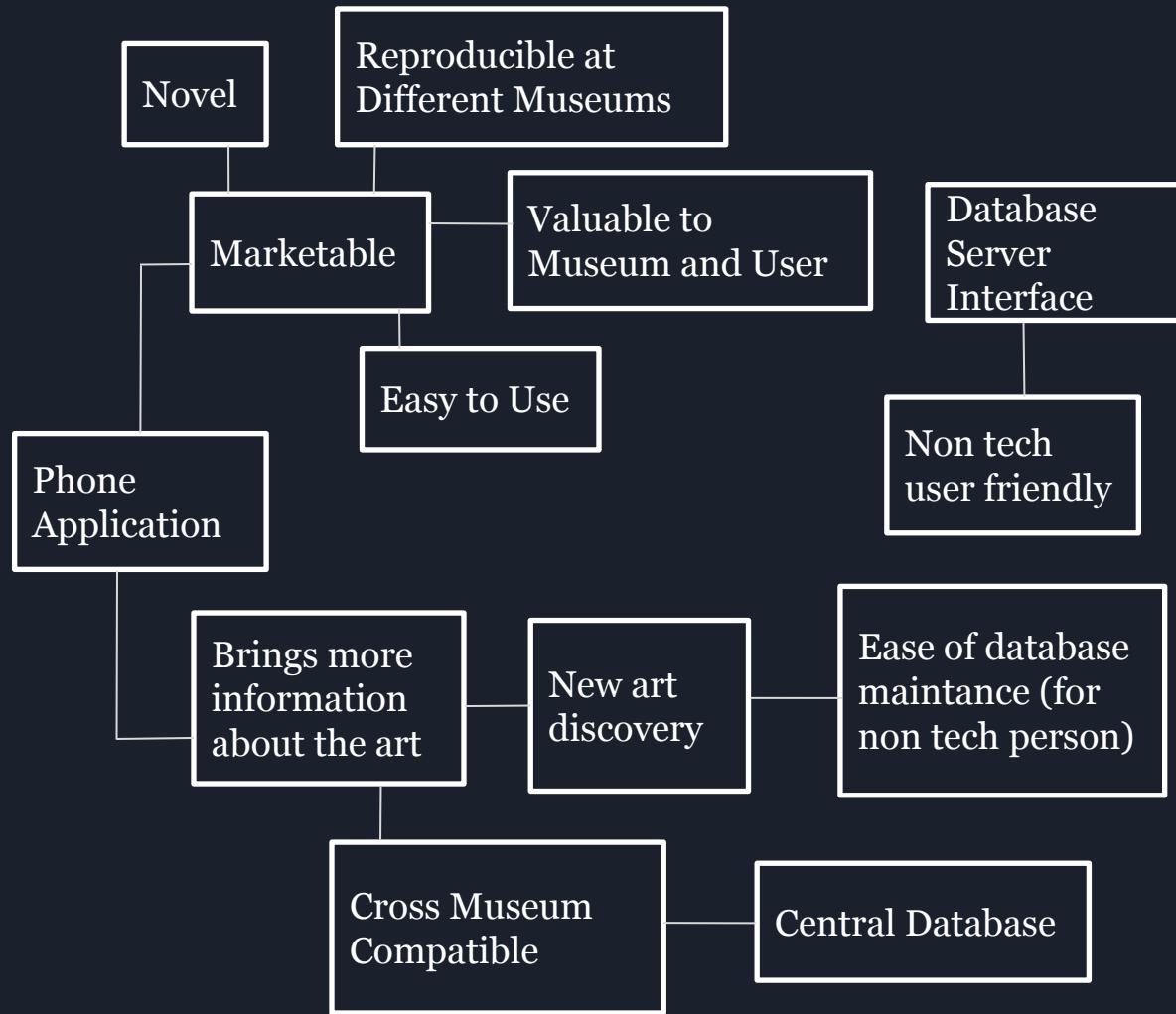
- App size less than 50 MB (averages: 38MB for IOS, 15MB for Android)
- Follow Laws about user data
- Safe and secure download

## Database Server Interface

- 74,000 Kb of disc space
- Windows 8 machine or newer



# Constraints Tree



# Objectives Tree

# Tools We Ended Up Using

## Phone Application

- Figma - containers, tags, design suite
- Bravo Studio - data binding, API connection, implementation, hosting, exporting (iOS & android)
- Airtable - databasing, API key, API URL, JSON
- Firebase - encrypted databasing, API key



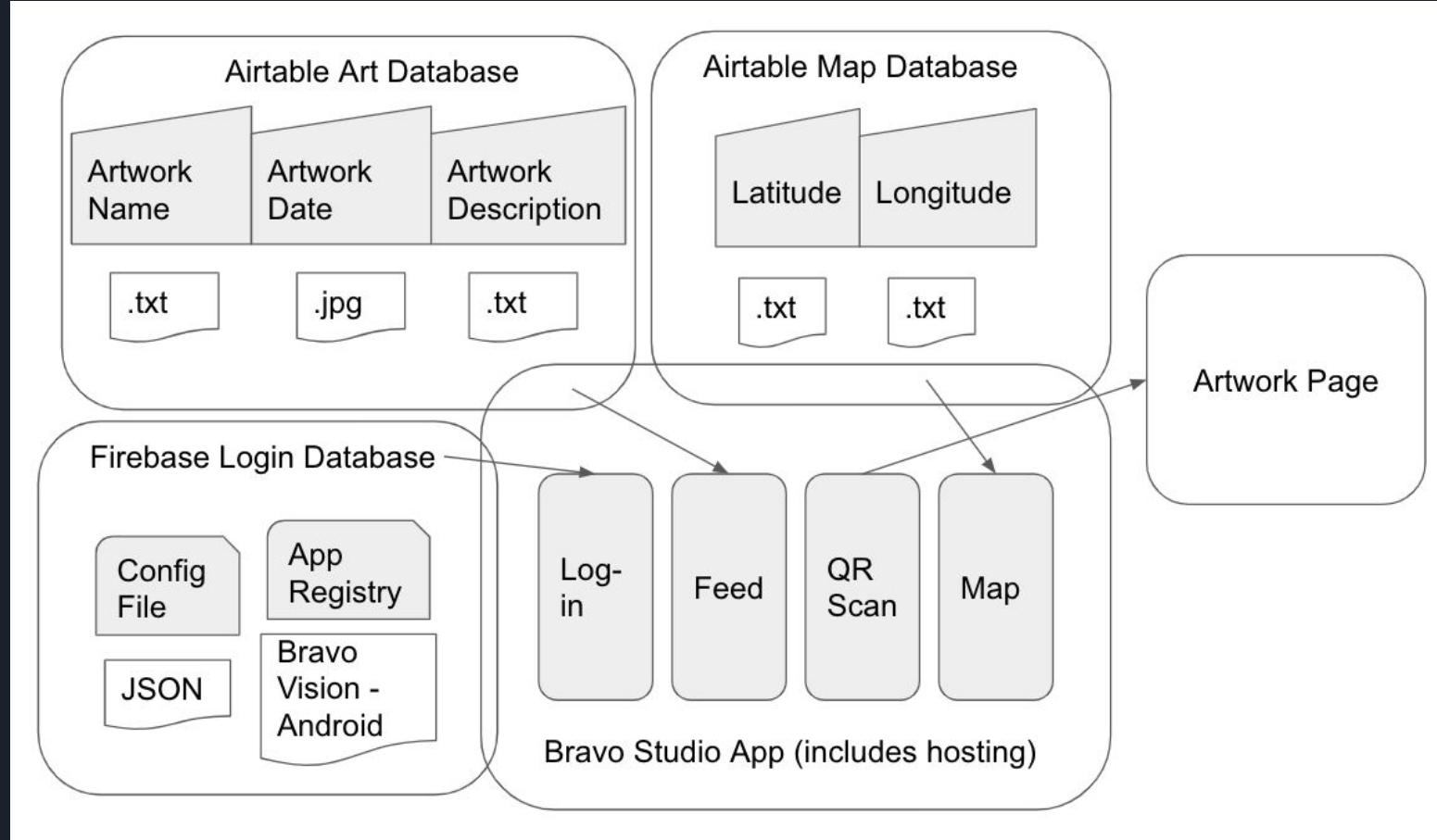
## Database Server Interface

- Webflow - CSS, HTML, JavaScript
- Python - sqlite3 (SQL), flask (web-app), paramiko (ssh), pyinstaller (build exe)
- PythonAnywhere - server
- TextEditor - MS visual Studio Code

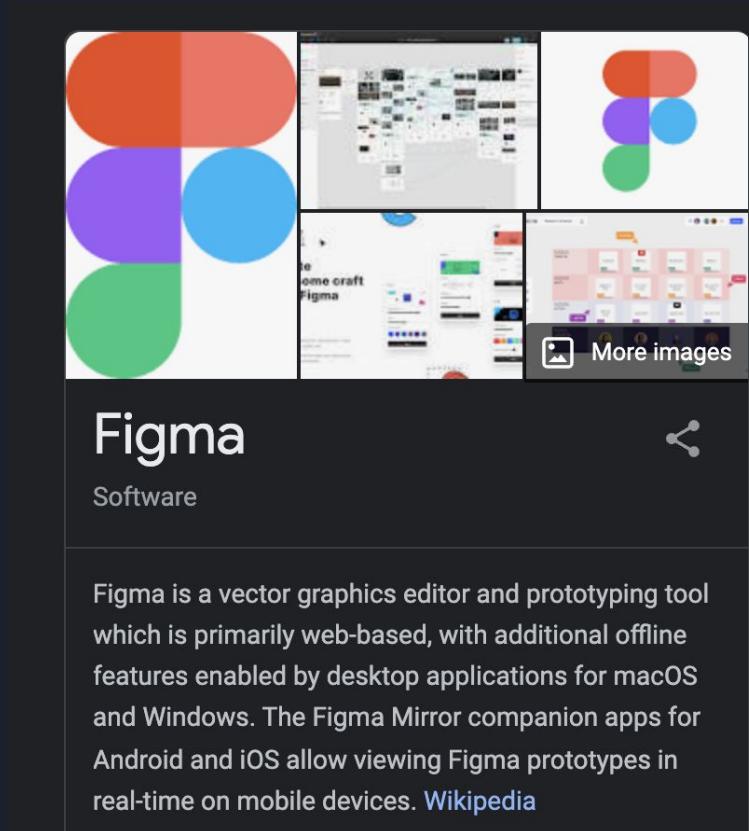


# Hasha

# Glass Box - App



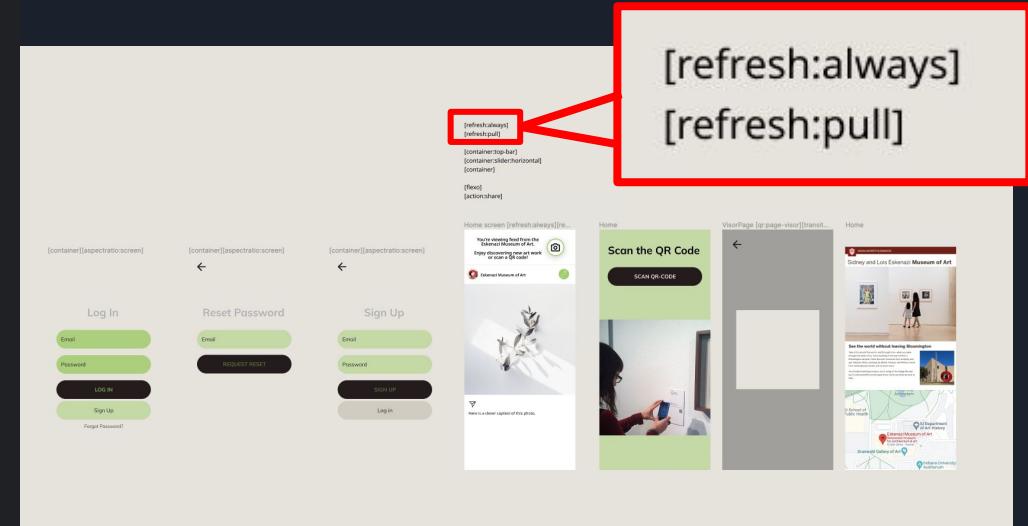
# Mobile Application Design - Hasha



The image shows the Figma software interface. At the top left is the Figma logo, which consists of four overlapping colored circles (orange, red, purple, blue) on a white background. To the right of the logo is a screenshot of the Figma workspace showing various wireframes and components. Below the logo is the word "Figma" in large white letters, followed by "Software" in smaller white letters. A share icon is located to the right of the title. At the bottom of the slide is a detailed description of Figma:

Figma is a vector graphics editor and prototyping tool which is primarily web-based, with additional offline features enabled by desktop applications for macOS and Windows. The Figma Mirror companion apps for Android and iOS allow viewing Figma prototypes in real-time on mobile devices. [Wikipedia](#)

- **Container** - abstraction to the app layer that packages code and dependencies together
- **Tag**- a text string placed in the layer name of the Figma file. After import into Bravo Studio, the layer with the Tag will turn into a mobile component or action in Bravo Vision



The image shows the Bravo Vision interface, which displays a mobile application prototype. On the right side of the screen, there is a callout box with a red border containing two text labels: "[refresh(always)]" and "[refresh(pull)]". A red arrow points from this callout box to a specific layer in the Figma prototype. The Figma prototype includes several screens: Log In, Reset Password, Sign Up, Home screen, Scan the QR Code, and VisorPage. The Home screen layer has the "[refresh(always)]" and "[refresh(pull)]" tags applied to it.

# Mobile Application Execution - Hasha

**Build  
mobile apps  
smarter.**

Your no-code mobile app builder for iOS and Android. Create MVPs, validate ideas and publish on App Store and Google Play Store.

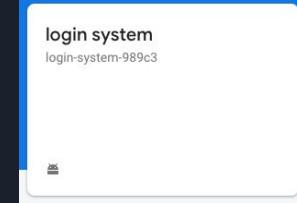
- Cheap
- Easy (kinda)
- API integration (lots)
- iOS and Android publishable
- Huge Figma support



- Resources
- Tutorials
- If there's not a plug-in/tag for it, it basically can't be done yet (eg. no google maps API support yet)
- Learning curve (similar to learning an Adobe suite)

# Login - Firebase - Hasha

## Google Developed Login Service for Apps



x Add Firebase to your Android app

**BravoVision - Intermediate Hosting**

1 Register app  
Android package name: com.dolelesslabs.app

2 Download config file  
[Download google-services.json](#)

Switch to the Project view in Android Studio to see your project root directory.

Move the google-services.json file you just downloaded into your Android app module root directory.

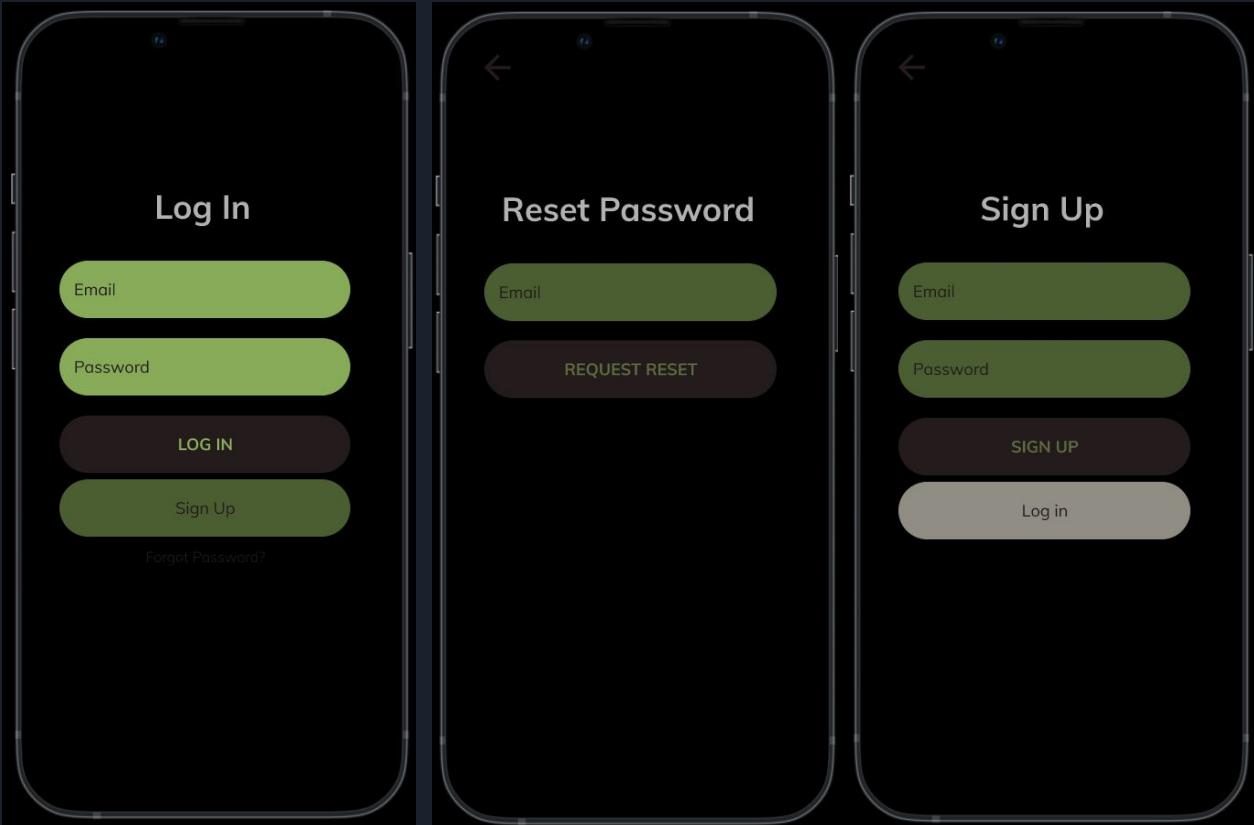
Different for Android or iOS, but can do both

 google-services.json

Previous [Next](#)

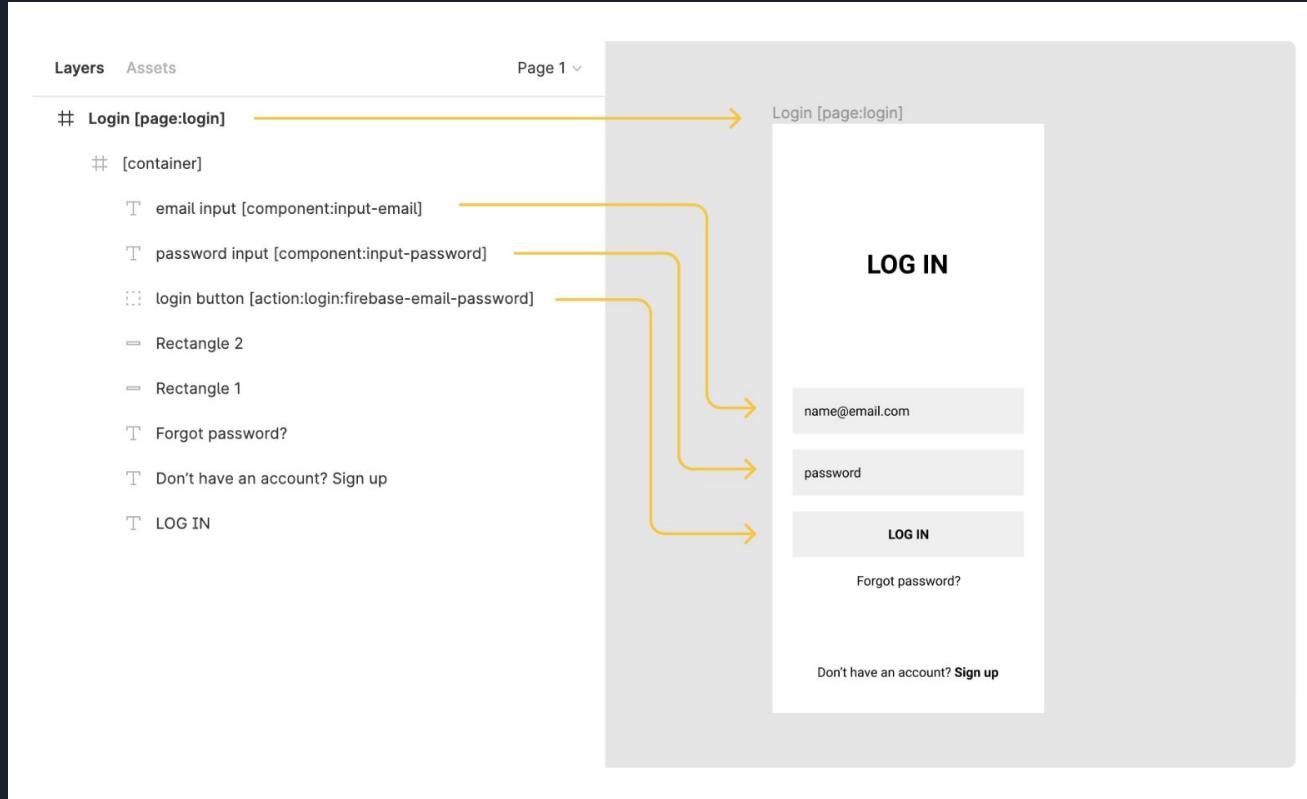
3 Add Firebase SDK

4 Read the Get Started Guide for Android

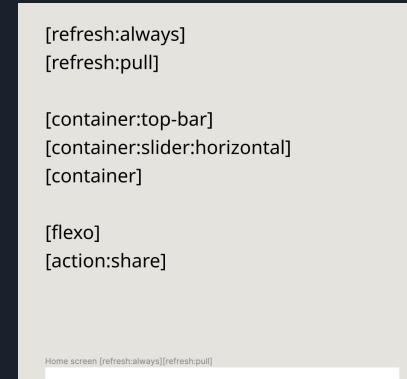
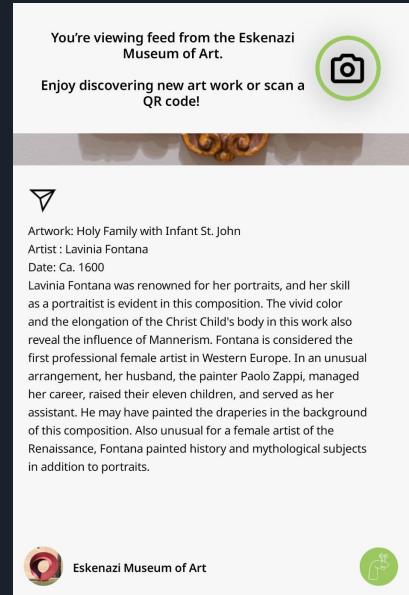
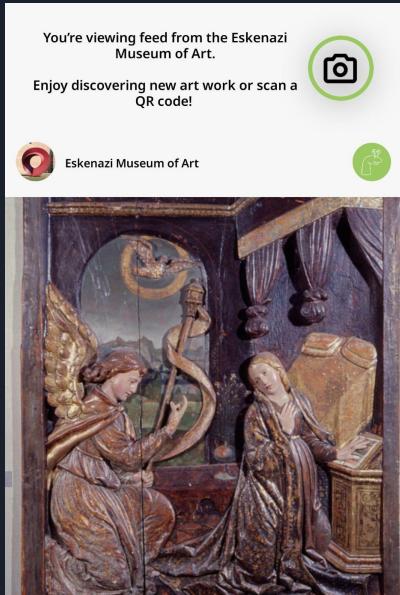
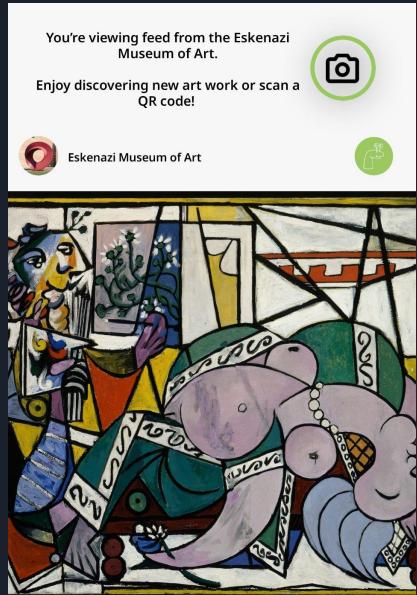


# Login - Firebase - Hasha

## Google Developed Login Service for Apps



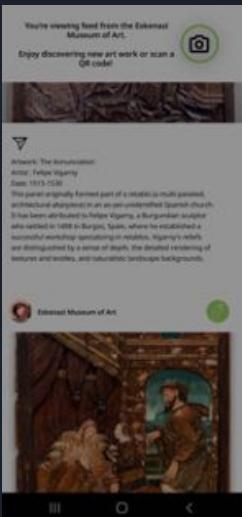
# Mobile Application Execution - Hasha



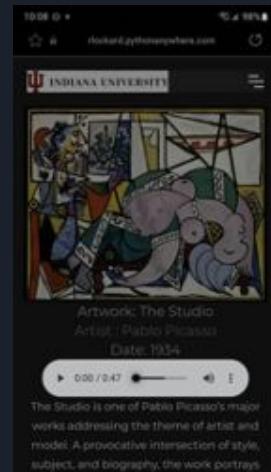
After TAGS are in place, you need to connect the API url from Airtable where the Museum Content resides to Bravostudio, where the app is hosted. An extra step is needed when fully exporting the app.

# Comments and Discovery Feed - Hasha

## Discovery Feed for User



## Museum Content



## Database

	Caption	Image	Posted
1	Artwork: Saint Eustace Artist : Albrecht Durer...		4/25/2022 14:00
2	Artwork: The Studio Artist : Pablo Picasso...		4/25/2022 13:02
3	Artwork: Holy Family with Infant St. John...		4/25/2022 13:00
4	Artwork: Judith with the Head of Holofernes...		4/21/2022 10:09
5	Artwork: Laborers Harvesting Bath Stone...		12/17/2020 07:57
6	Artwork: Madonna and Child with Apple and Pears...		12/17/2020 07:19
7	Artwork: The Annunciation Artist : Felipe Vigarny...		12/11/2020 09:34
8	Artwork: The Birth of the Virgin Artist : Felipe Vigarny...		12/11/2020 07:24
9	Artwork: The Nativity Artist : Felipe Vigarny...		12/9/2020 11:03
+			

# Airtable API for Discovery Feed - Hasha

Saved

## Requests

Deals from product

+ Deals from product

Lead - Delete

Note - Delete

Edit product

Notes from deal

Add note to lead

Insights - Today summary

Notes from lead

Add lead

Deals - Today

Insights - This month summary

Insights - Sales by agent

Deals from product

Request URL

GET https://api.airtable.com/v0/appWADqQ7bynJnV/Deals?filterByFormula=%7BProductId%7D%3D\${productId}&sort%5B0%5D%5Bfield

Send

Headers Test Values Body Pagination Query Parameters

Headers

HTTP headers to send with your request

Authorization

.....

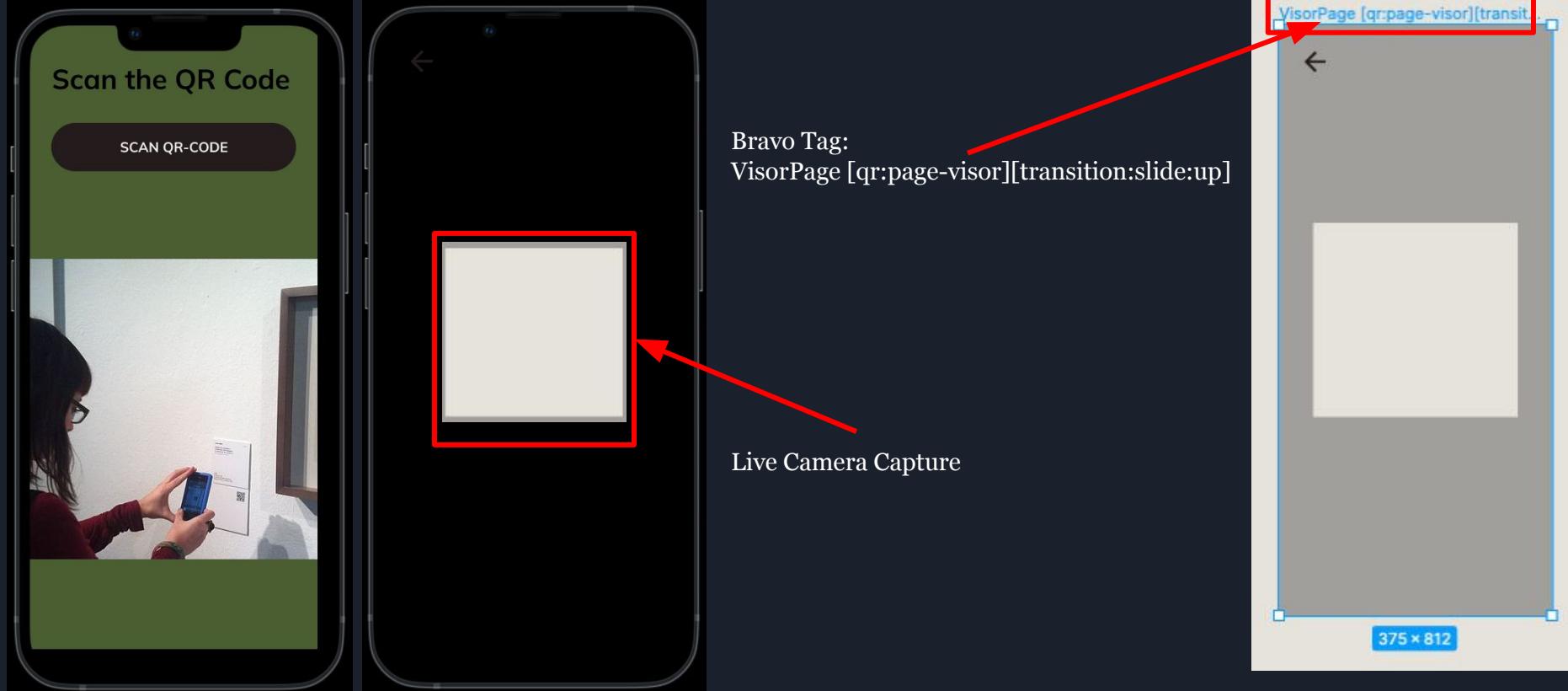
Key

Please enter the key first

Received Data Selected Data Debug

Hit send to get a response

# QR Code Scanner Implementation - Hasha

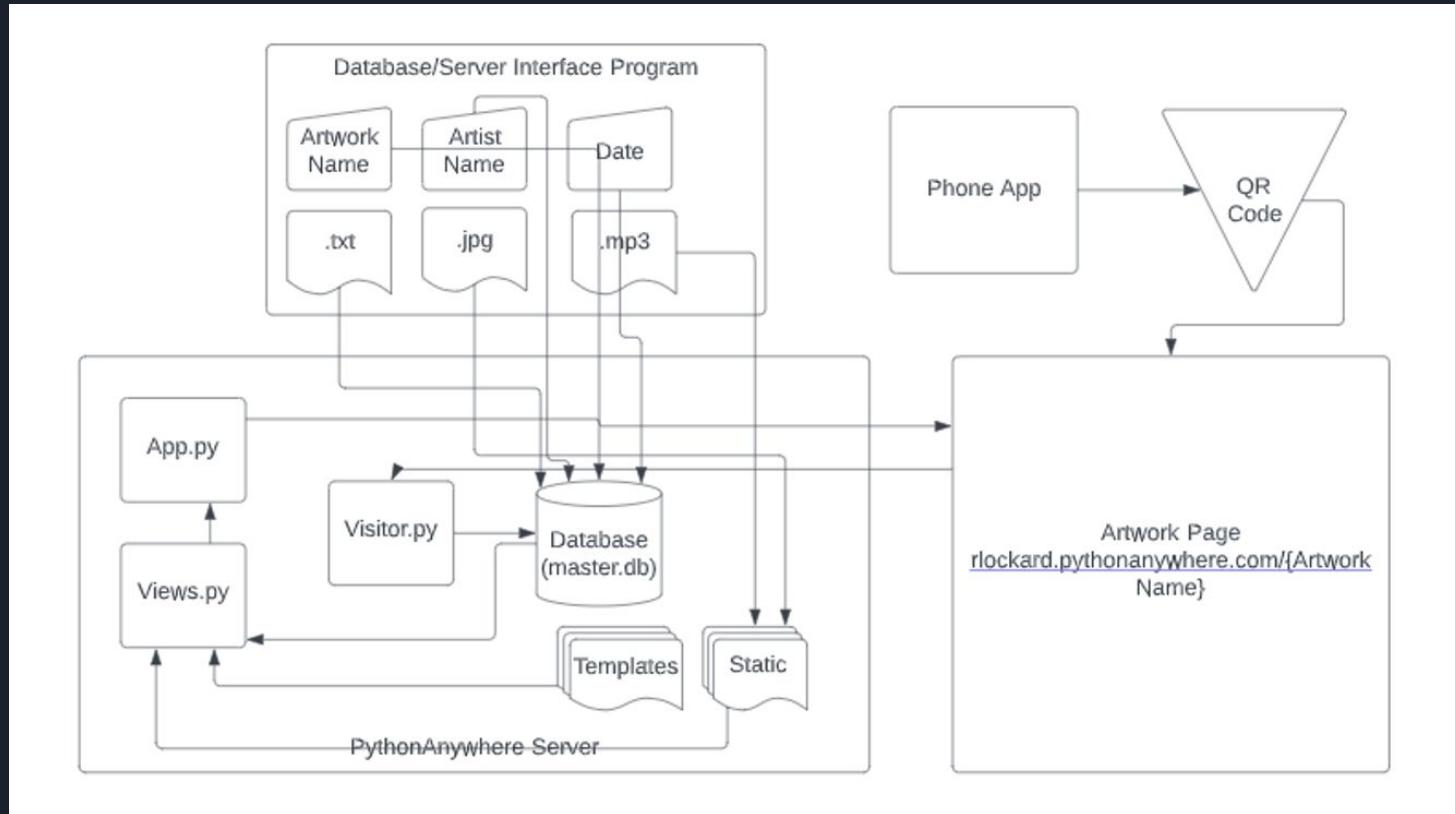


(note - will only work with Bravo for implementation/extraction (but this does not affect iOS/Android use - it will still work))

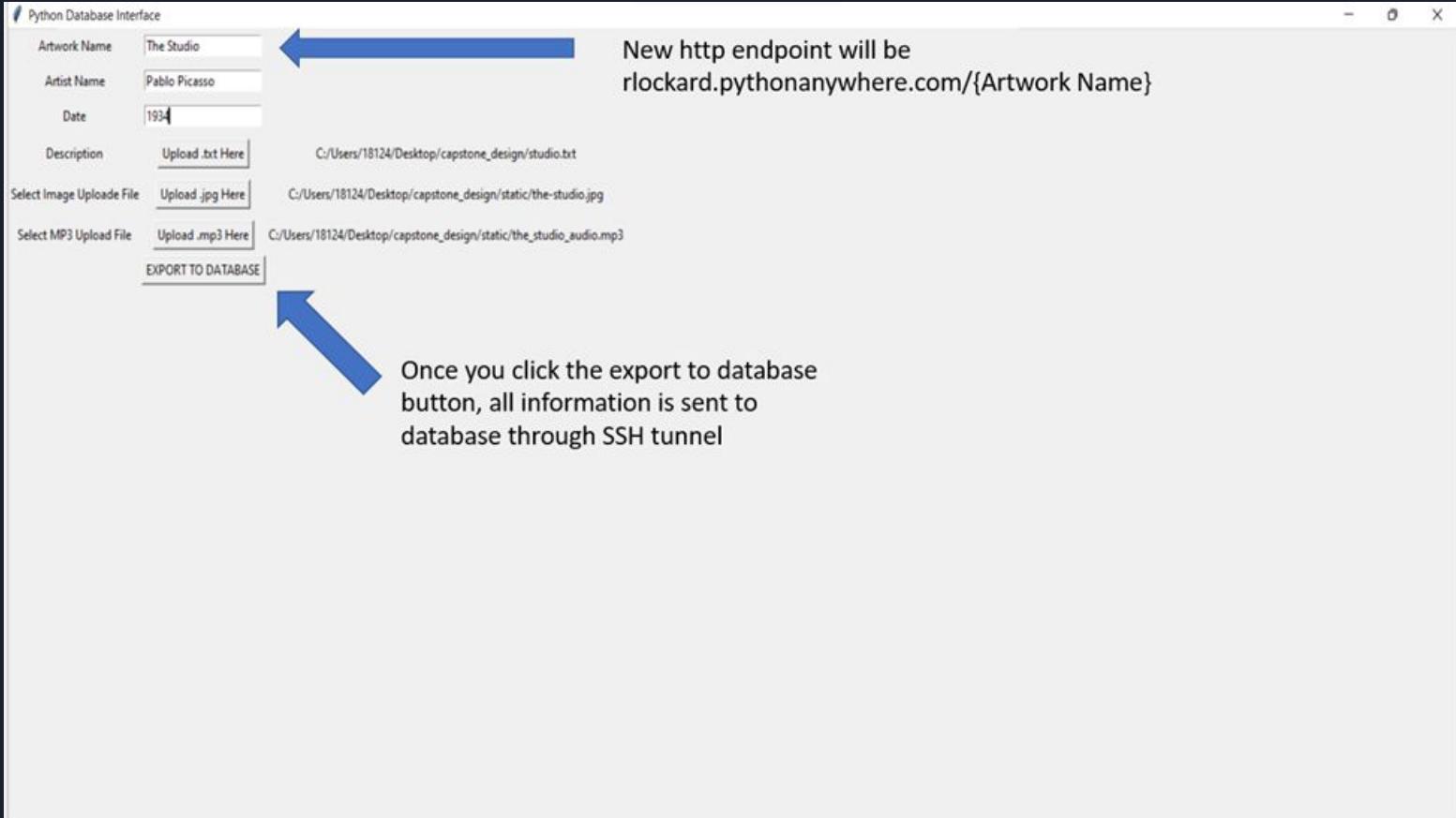


# Ryan

# Glass Box - Database



# Database Server Interface - Ryan



# Database with SQL - Ryan

location	name	artist	date	description	image_file_name	audio
Filter	Filter	Filter	Filter	Filter	Filter	Filter
Modern Art	The Studio	Pablo Picasso	1934	The Studio is one of Pablo Picasso's major works...	data:image/jpeg;base64,/9j/...	string
Modern Art	Beth Aleph	Morris Louis	1960	Morris Louis was a member of the Washington ...	NULL	string
European and American Art, Medieval to 1900	Saint Eustace	Albrecht Durer	Ca. 1501	Since prints are made as multiples, museums ...	NULL	string
European and American Art, Medieval to 1900	Holy Family with Infant St. John	Lavinia Fontana	Ca. 1600	Lavinia Fontana was renowned for her portraits, ...	NULL	string
European and American Art, Medieval to 1900	Judith with the Head of Holofernes	Matteo di Giovanni	Ca. 1490	This sword-wielding woman has traditionally bee...	NULL	string
European and American Art, Medieval to 1900	The Annunciation	Felipe Vigormy	1515-1530	This panel originally formed part of a retablo (a ...	NULL	string
European and American Art, Medieval to 1900	The Nativity	Felipe Vigormy	1515-1530	This panel originally formed part of a retablo (a ...	NULL	string
European and American Art, Medieval to 1900	The Birth of the Virgin	Felipe Vigormy	1515-1530	This panel originally formed part of a retablo (a ...	NULL	string
Modern Art	Homage to the Square: Bright Reminder	José Albers	1956	In 1920, José Albers began a course of study at ...	NULL	string
Modern Art	Portrait of Lisa Bigelow	Alfred Leslie	1964	Alfred Leslie, who pioneered the renewal of ...	NULL	string
European and American Art, Medieval to 1900	Madonna and Child with Apple and Pears	Studio of Bernard van Orley	1530	Flemish artists were renowned for vivid and ...	NULL	string
European and American Art, Medieval to 1900	Laborers Harvesting Bath Stone	Benjamin Barker, the Younger	Ca. 1800	This painting focuses on a picturesque subject, ...	NULL	string
European and American Art, Medieval to 1900	Laborers Harvesting Bath Stone	Benjamin Barker, the Younger	Ca. 1800	This painting focuses on a picturesque subject, ...	NULL	string
European and American Art, Medieval to 1900	Laborers Harvesting Bath Stone	Benjamin Barker, the Younger	Ca. 1800	This painting focuses on a picturesque subject, ...	NULL	string
Modern Art	The Studio	Pablo Picasso	NULL	hello	NULL	string

Tracking Statistics -  
Number of views of the  
HTML page

page_name	number_of_views
Filter	Filter
The Birth of the Virgin	0
Saint Eustace	0
The Studio	10
Beth Aleph	0
Judith with the Head of Holofernes	0
Holy Family with Infant St. John	1
The Annunciation	0
The Nativity	0
Portrait of Lisa Bigelow	0
Madonna and Child with Apple and Pears	0
Laborers Harvesting Bath Stone	0
The Birth of the Virgin	0
Saint Eustace	0
The Studio	10
Beth Aleph	0
Judith with the Head of Holofernes	0
Holy Family with Infant St. John	0
The Annunciation	0
The Nativity	0
Portrait of Lisa Bigelow	0
Madonna and Child with Apple and Pears	0
Laborers Harvesting Bath Stone	0

# Dynamic HTML generation - Ryan

```
views.py > @the_studio
1  from flask import Blueprint, render_template
2  import sqlite3
3  from sqlite3 import Error
4  import visitor
5
6  from matplotlib import image
7
8  db_connection = sqlite3.connect('master.db', check_same_thread=False)
9  db_cursor = db_connection.cursor()
10
11 views = Blueprint(__name__, "views")
12 @views.before_request
13 def track_visitor():
14     visitor.track_visitor()
15
16 @views.route("/")
17 def home():
18     return render_template("eskenazi.html")
19
20 @views.route("/<page_name>")
21 def pages(page_name):
22     artwork_name = page_name
23     sql = "SELECT * FROM eskenazi WHERE name = ?"
24     db_cursor.execute(sql, (artwork_name,))
25     df = db_cursor.fetchone()
26     artwork_name = df[2]
27     artist_name = df[3]
28     artwork_date = df[4]
29     im = df[6]
30     print(im[:1000])
31     artwork_description = df[5]
32     audio = df[7]
33     artwork_description = artwork_description.replace("\\", "")
```

```
34     </pre>
35
36     <head>
37         <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
38         <link href="{{ url_for('static', filename='webflow.css') }}" rel="stylesheet" type="text/css">
39         <link href="{{ url_for('static', filename='style.css') }}" rel="stylesheet" type="text/css">
40         <script src="https://ajax.googleapis.com/ajax/libs/webfont/1.6.26/webfont.js" type="text/javascript"></script>
41         <script type="text/javascript">WebFont.load({ google: { families: ['Montserrat:100,100italic,200,200italic,300,300italic,400,400italic,500,500italic,600,600italic,700,700italic'] } })</script>
42         <!-- [if lt IE 9]><script src="https://cdnjs.cloudflare.com/ajax/libs/html5shiv/3.7.3/html5shiv.min.js" type="text/javascript"></script><![endif]-->
43         <script type="text/javascript">!function(o,r){var n=c.documentElement,t="w-mod";n.className+=t+"js",("ontouchstart"in o||o.DocumentTouch&c.documentElement.touches)in o||c.documentElement.classList.add("touch")}</script>
44         <link href="{{ url_for('static', filename='favicon.ico') }}" rel="shortcut icon" type="image/x-icon">
45         <link href="{{ url_for('static', filename='webclip.png') }}" rel="apple touch icon">
46     </head>
47     <body>
48         <div data-collapse="medium" data-animation="default" data-duration="400" data-easing="ease" data-easing2="ease" role="banner" class="navigation w-nav">
49             <div class="navigation-wrap">
50                 <a href="index.html" class="logo-link w-nav-brand"></a>
51                 <div class="menu">
52                     <nav role="navigation" class="navigation-items w-nav-menu">
53                         <a href="{{url_for('views.home')}}" class="navigation-item w-nav-link">Home</a>
54                         <a href="{{url_for('views.team')}}" class="navigation-item w-nav-link">Team</a>
55                     </nav>
56                     <div class="menu-button w-nav-button"></div>
57                 </div>
58             </div>
59         </div>
60         <div class="section">
61             <div class="container">
62                 <div class="w-layout-grid projects-grid">
63
64                     <div id="w-node_2870ff7d-26fe-9471-3c4e-7762e2030cdc-c4f3b6e5" data-w-id="1102" data-easing="ease" data-easing2="ease" data-duration="400" data-animation="default" data-collapse="medium" data-style="normal" data-type="grid-item" data-x="0" data-y="0" data-width="100%" data-height="100%" data-align="center" data-aligny="middle" data-justify="center" data-justifyy="middle" data-orientation="vertical" data-spacing="0" data-spacingx="0" data-spacingy="0" data-style="normal" data-type="grid-item">
65                         <div class="project-name-wrap">
66                             <div class="paragraph-light">Artwork: {{ artwork_name }}</div>
67                             <div class="paragraph-light">Artist : {{ artist_name }}</div>
68                             <div data-w-id="1102" data-easing="ease" data-easing2="ease" data-duration="400" data-animation="default" data-collapse="medium" data-style="normal" data-type="grid-item" data-x="0" data-y="0" data-width="100%" data-height="100%" data-align="center" data-aligny="middle" data-justify="center" data-justifyy="middle" data-orientation="vertical" data-spacing="0" data-spacingx="0" data-spacingy="0" data-style="normal" data-type="grid-item">
69                                 
70                         <div class="w-richtext">
71                             <div>
72                                 <audio src="{{ url_for('static', filename='audio_filename) }}" controls>
73                                     </audio>
74                                     <div class="w-f-section">
75                                         <div>{{ description }}</div>
76                                     </div>
77                                     </div>
78                                     <div>
79                                         <div>
80                                             <script src="https://d3e54v1033q8gb.cloudfront.net/js/jquery-3.5.1.min.js?site-622/fb17/d7d7005/f3b6e2" type="text/javascript" integrity="sha256-9/aliU8dd2tLb05s" data-w-id="1102" data-easing="ease" data-easing2="ease" data-duration="400" data-animation="default" data-collapse="medium" data-style="normal" data-type="grid-item" data-x="0" data-y="0" data-width="100%" data-height="100%" data-align="center" data-aligny="middle" data-justify="center" data-justifyy="middle" data-orientation="vertical" data-spacing="0" data-spacingx="0" data-spacingy="0" data-style="normal" data-type="grid-item">
81                                     <script src="{{ url_for('Static', filename='Webflow.js') }}" type="text/javascript"></script>
82                                     <!-- [if lt IE 9]><script src="https://cdnjs.cloudflare.com/ajax/libs/placeholders/3.0.2/placeholders.min.js"></script><![endif]-->
83                                 </div>
84                             </div>
85                         </div>
86                     </div>
87                 </div>
88             </div>
89         </div>
90     </body>
91 
```

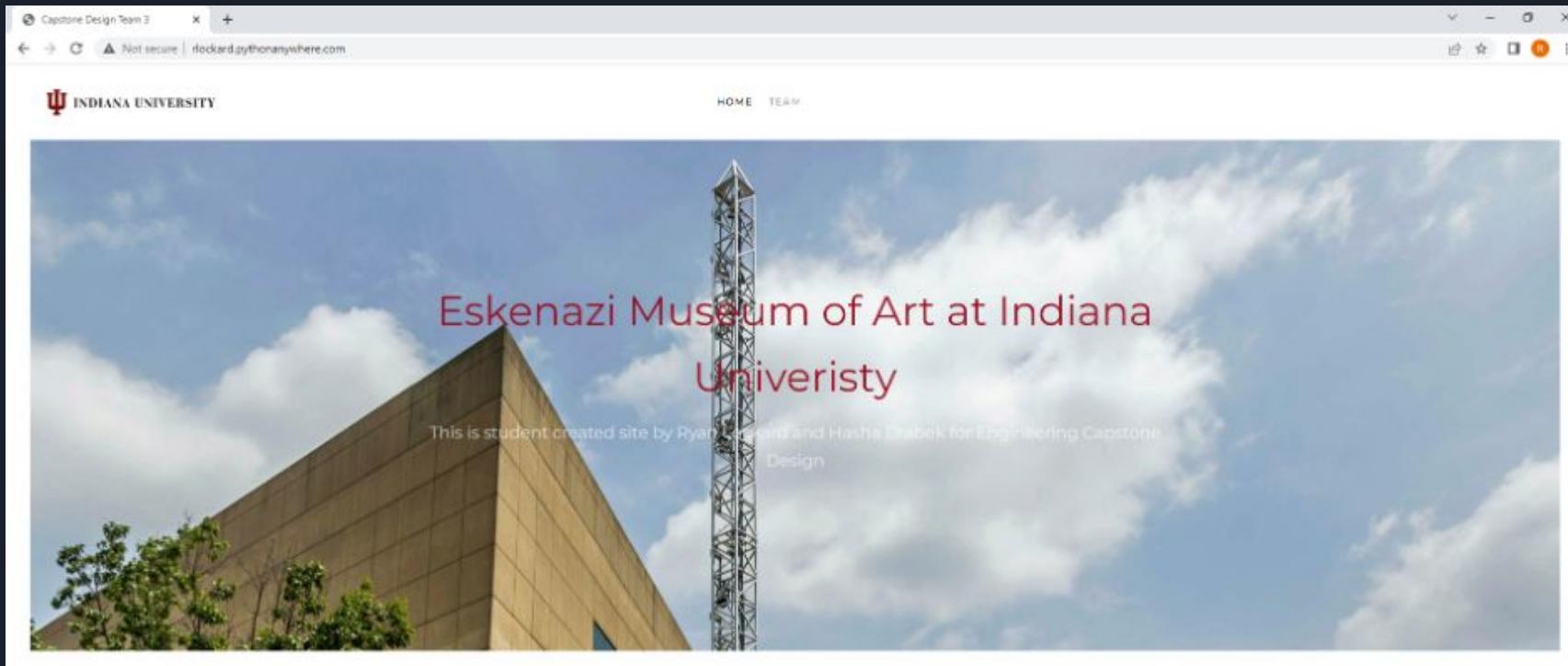
# Running Code - Ryan

```
views.py > cd the_studio
1  from flask import Blueprint, render_template
2  import sqlite3
3  from sqlite3 import Error
4  import visitor
5
6  from matplotlib import image
7
8  db_connection = sqlite3.connect('master.db', check_same_thread=False)
9  db_cursor = db_connection.cursor()
10
11 views = Blueprint(__name__, "views")
12 @views.before_request
13 def track_visitor():
14     visitor.track_visitor()
15
16 @views.route("/")
17 def home():
18     return render_template("eskenazi.html")
19
20 @views.route("/<page_name>")
21 def pages(page_name):
22     artwork_name = page_name
23     sql = "SELECT * FROM eskenazi WHERE name = ?"
24     db_cursor.execute(sql, (artwork_name,))
25     df = db_cursor.fetchmany()
26     artwork_name = df[2]
27     artist_name = df[3]
28     artwork_date = df[4]
29     im = df[6]
30     print(im[:1000])
31     artwork_description = df[5]
32     audio = df[7]
33     artwork_description = artwork_description.replace("\\\\\'\\\'")
34
35     return render_template("the_studio.html", audio_filename="the_studio_audio.mp3", image_filename="the-studio.jpg", artwork_name=artwork_name, artist_name=artist_name, date=artwork_date, descri
```

```
1  from asyncio import threads
2  from distutils.log import debug
3  from pickle import TRUE
4  from flask import Flask
5  from views import views
6  from waitress import serve
7
8
9
10 app = Flask(__name__, static_folder="static")
11 app.register_blueprint(views, url_prefix="/")
12
13
14 if __name__ == '__main__':
15     app.run(debug=True, port=8080, host="0.0.0.0")
16
17
18
```

Code to Display Webpage

# Website - Ryan



# Website - Ryan

The screenshot shows a web browser window displaying the 'Capstone Design Team 3' website. The page features a header with the Indiana University logo and navigation links for 'HOME' and 'TEAM'. The main content area is titled 'Capstone Design Team 3'. It includes two profile sections: one for Hasha Drabek, a woman with long brown hair, and one for Ryan Lockard, a young man with short brown hair. Each profile includes a photo, the team member's name, and a brief description of their role and responsibilities.

Capstone Design Team 3

Indiana UNIVERSITY

HOME TEAM

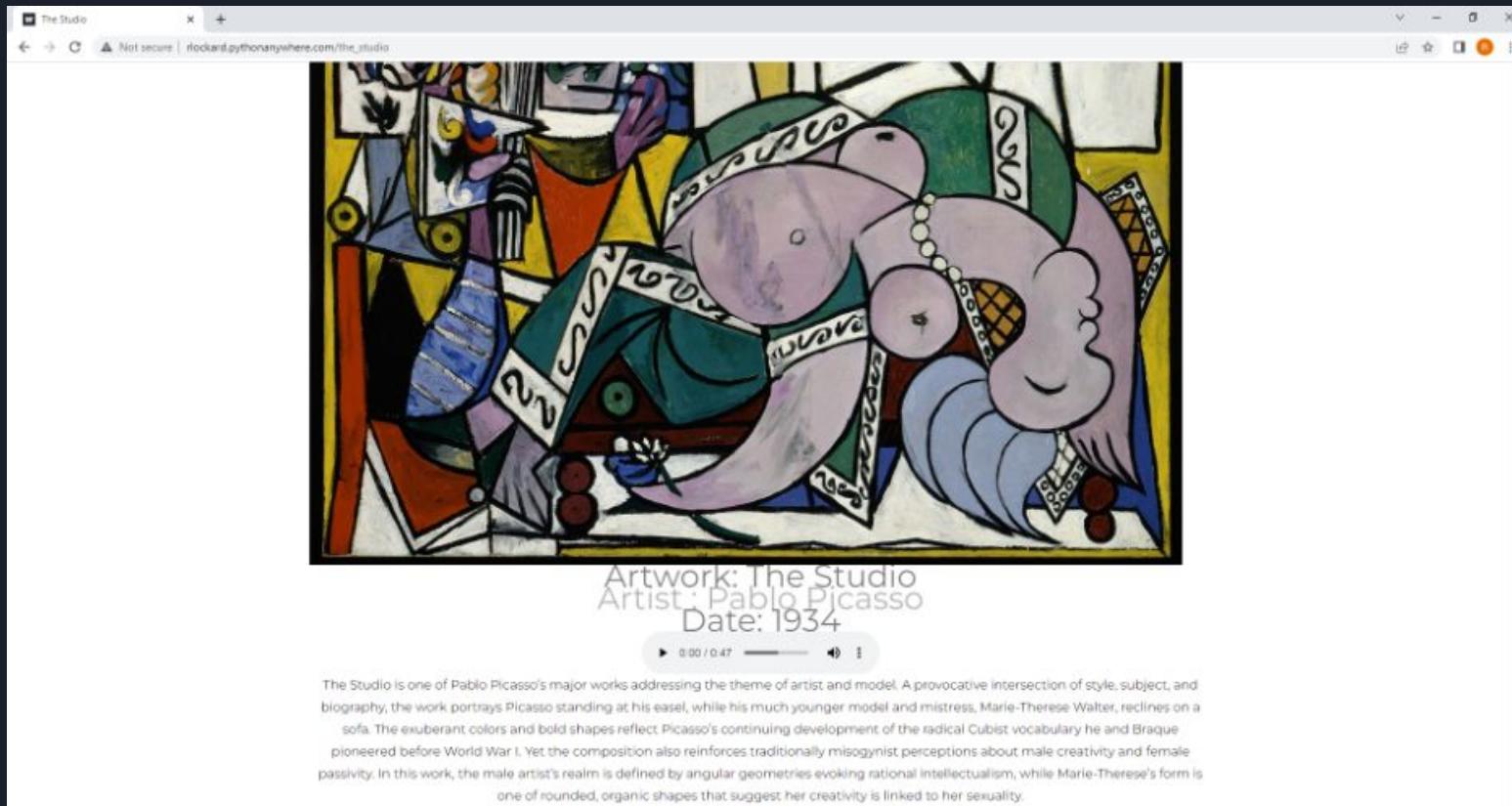
Hasha Drabek

Hasha is the project coordinator of our team. Hasha was responsible for designing and deploying the app portion of our project using a platform called Figma.

Ryan Lockard

Ryan is the system integrator of our team. Ryan was responsible for building and deploying this website for the team. Ryan was also responsible for designing and maintaining the database that powers this site and logs user information back to the museum.

# Website - Ryan



The Studio

Not secure | rockard.pythonanywhere.com/the\_studio

Artwork: The Studio  
Artist: Pablo Picasso  
Date: 1934

0:00 / 0:47

The Studio is one of Pablo Picasso's major works addressing the theme of artist and model. A provocative intersection of style, subject, and biography, the work portrays Picasso standing at his easel, while his much younger model and mistress, Marie-Therese Walter, reclines on a sofa. The exuberant colors and bold shapes reflect Picasso's continuing development of the radical Cubist vocabulary he and Braque pioneered before World War I. Yet the composition also reinforces traditionally misogynist perceptions about male creativity and female passivity. In this work, the male artist's realm is defined by angular geometries evoking rational intellectualism, while Marie-Therese's form is one of rounded, organic shapes that suggest her creativity is linked to her sexuality.

# Making an Endpoint - Ryan



# Future Additions to the Project

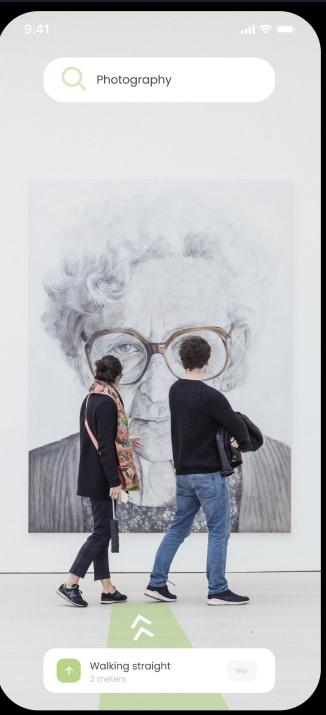
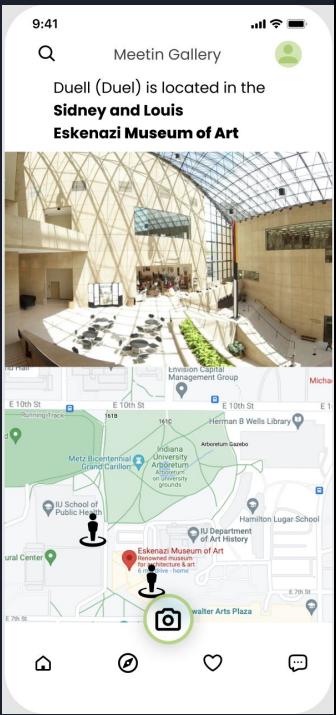
Add App Features:  
Better Location

Comments

VR?

Automate User  
Tracking

Export to  
TestFlight



page_name	number_of_views
Filter	Filter
The Birth of the Virgin	0
Saint Eustace	0
The Studio	10
Beth Aleph	0
Judith with the Head of Holofernes	0
Holy Family with Infant St. John	1
The Annunciation	0
The Nativity	0
Portrait of Lisa Bigelow	0
Madonna and Child with Apple and Pears	0
Laborers Harvesting Bath Stone	0
The Birth of the Virgin	0
Saint Eustace	0
The Studio	10
Beth Aleph	0
Judith with the Head of Holofernes	0
Holy Family with Infant St. John	0
The Annunciation	0
The Nativity	0
Portrait of Lisa Bigelow	0
Madonna and Child with Apple and Pears	0
Laborers Harvesting Bath Stone	0



Revisit AI?





Ryan Lockard, rtlockar@iu.edu  
BS Bioengineering, Minor Math

**Lifelong Learning from Project:**  
Networking (Technical), CSS, HTML,  
JavaScript, Improved Python Knowledge

**Immediate Future Plans:**  
Electrical Controls Engineer  
(Meyer Tools, Cincinnati OH)

**Longterm Future Plans:**  
Work on Da Vinci Robot  
(Intuitive, Remote), MSEE, MBA



Hasha Drabek, hddrabek@iu.edu  
BS ISE, BS Physics, BS Maths

**Lifelong Learning from Project:**  
Figma Software, BravoStudio, Database  
connecting, UI Design

**Immediate Future Plans:**  
Risk Analyst (Grant Thornton, Remote),  
MS ISE, Grow Current Businesses (product  
sourcing industry)

**Longterm Future Plans:**  
Big 4 Finance, Automate Current  
Businesses, Move to Alaska (work remote)

# QR Codes - Try One!



# Live DEMO!



Thank you.  
Questions?

# Appendix

Mentor -----	43
Role Assignments -----	44
More App Pages -----	45-47
Finances -----	48
Functions & Means -----	49-50
WorkFlow -----	51
OG Block Diagram -----	53
Market Study/Industry Interviews	55-59
Safety & Ethics -----	61-63
Fall Gannt Chart -----	65
Fall AI results -----	67-71
Sustainability -----	72
Takeaways -----	73
Software Package -----	74-76
Regulations -----	77





Hannah Jiao, Graduate Student  
Human Computer Interaction Design

Technical Skills:  
Figma, Adobe Suite, Final Cut Pro

Research Skills:  
Storyboard, Usability Testing, Sitemap,  
Personas, SWOT Analysis

Significant Courses:  
Experience Design, Foundation of HCI

Please email for portfolio,  
[yongjiao@iu.edu](mailto:yongjiao@iu.edu)

# Team Roles

**Hasha**

**Project Coordinator  
Presentation Manager**

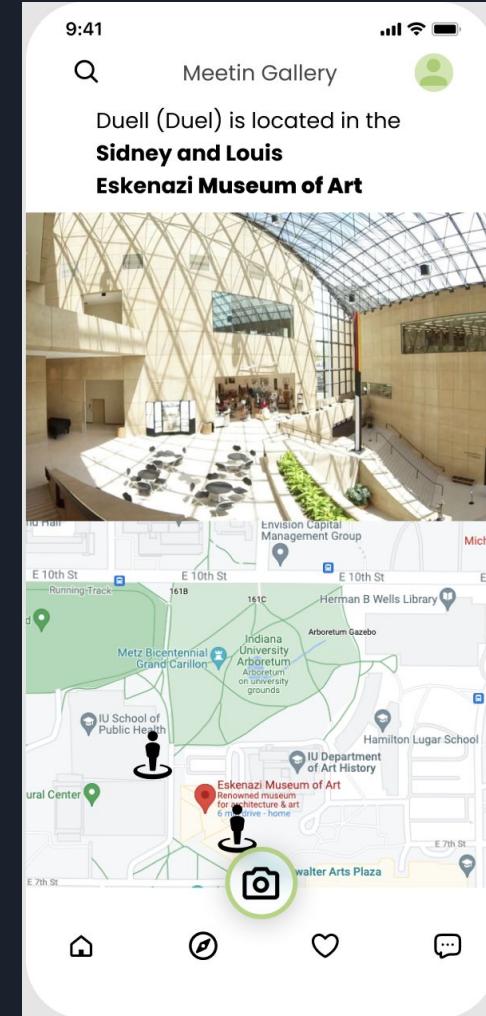
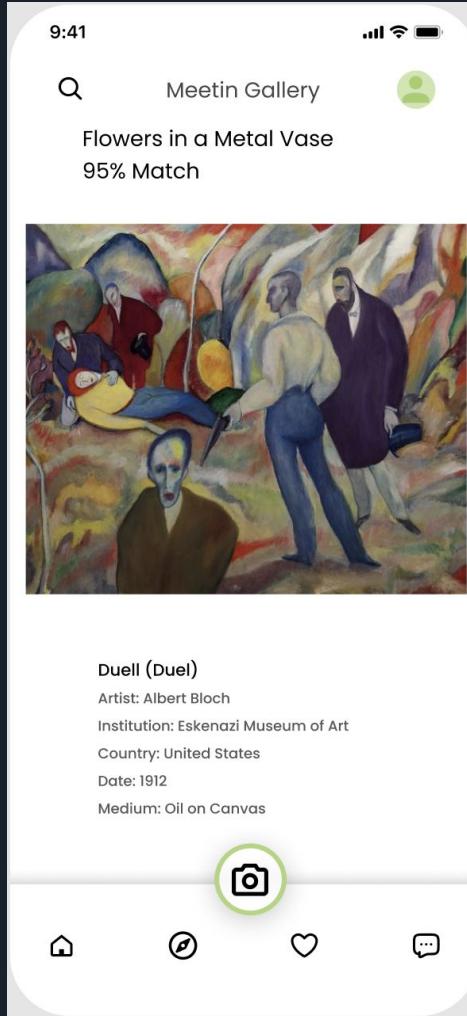
**App Design  
App Implementation  
Market Study**

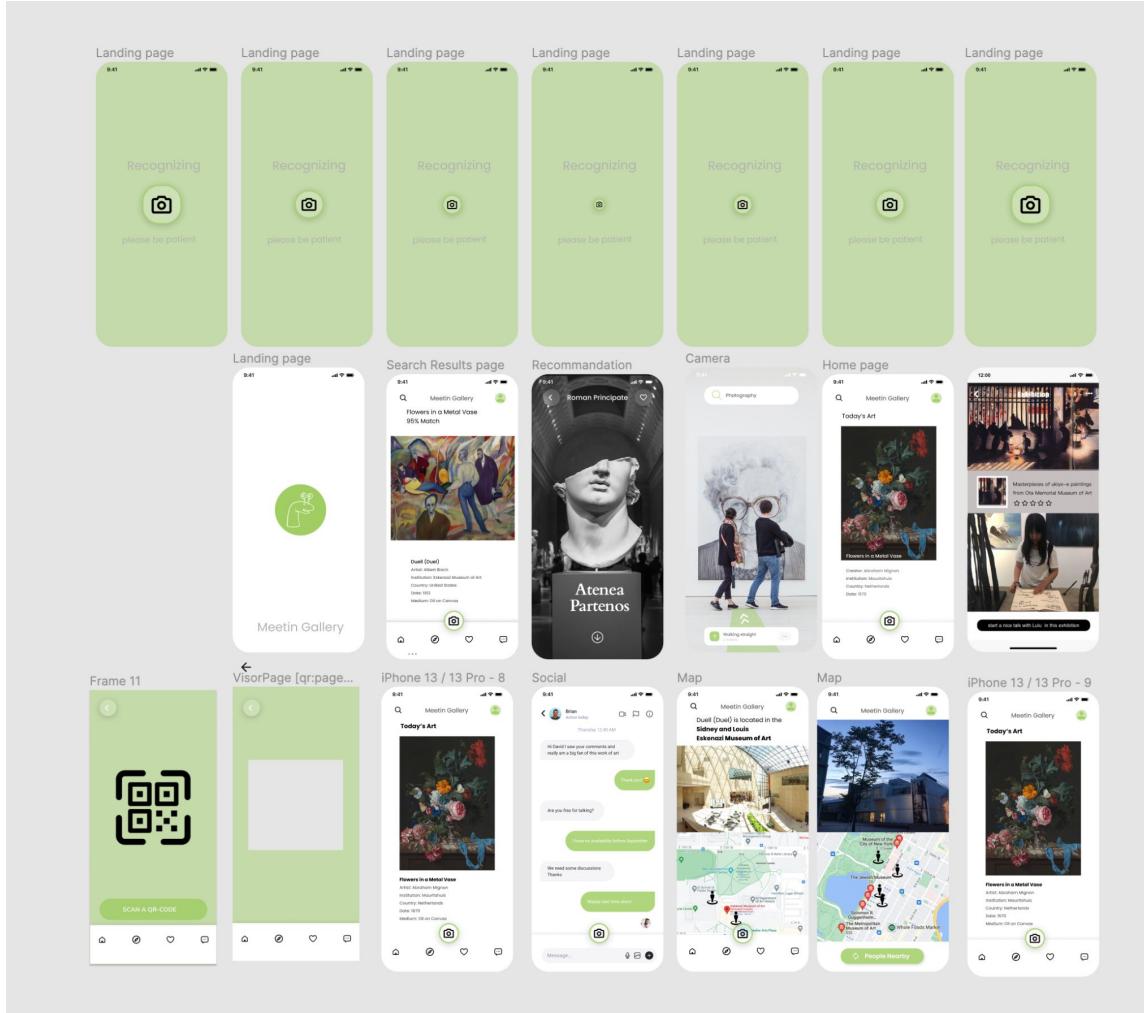
**Ryan**

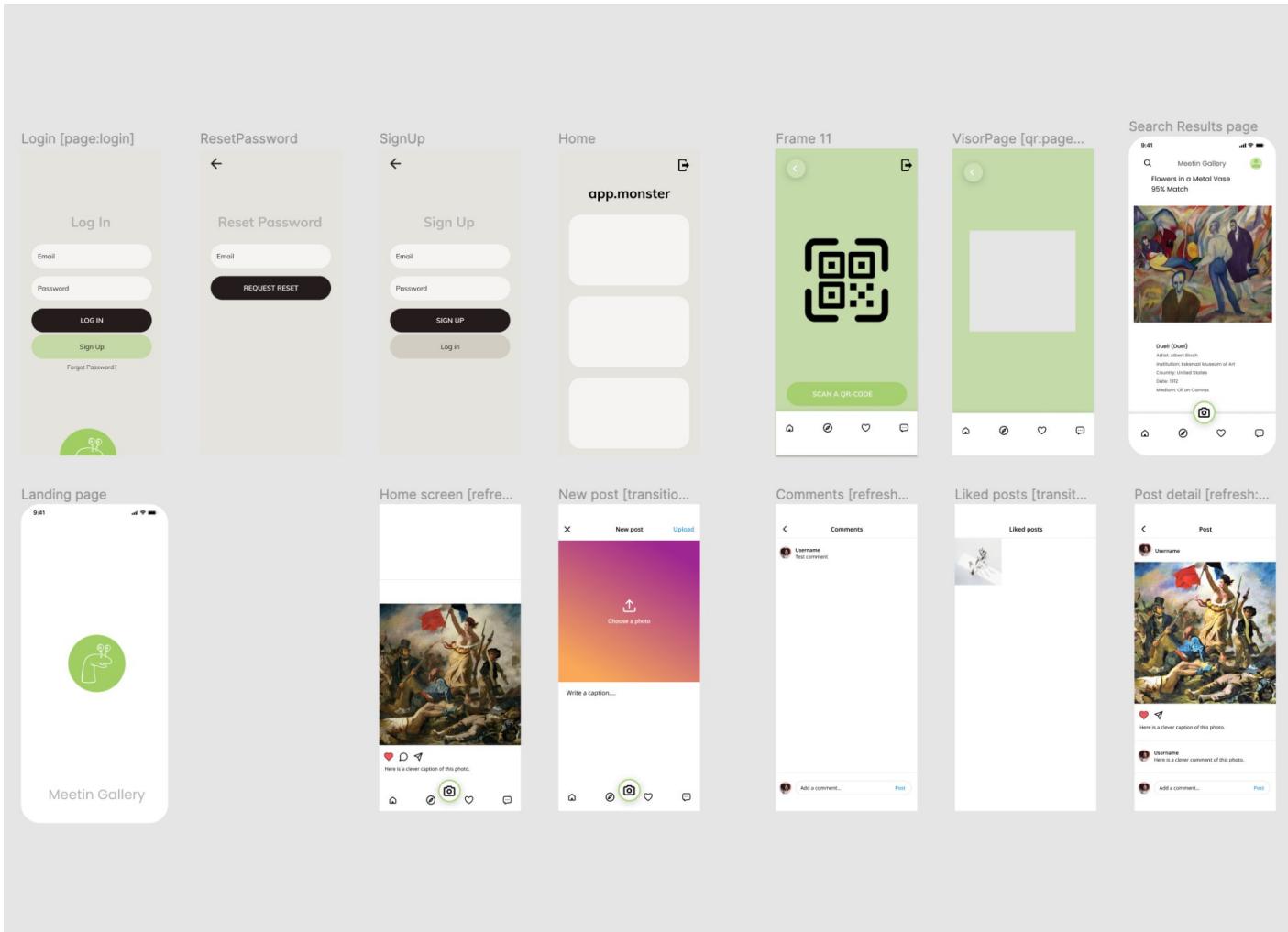
**System Manager  
Prototype Integrator**

**Data Base  
Automation  
Website  
Server Setup**

**App**







Standard/Law/Regulation	How it Applies to Us
California Consumer Privacy Act of 2018- United States	All private data must be encrypted and at rest in encryption. Security must be ‘reasonable’ to protect the data.
Federal Information Processing Standards, mandated by NIST- United States	We use Firebase an encryption based security system to house our user’s login data. If we are able to certify that our data is protected, we can become FIPS certified, and any encryption system must follow FIPS 140-3, to mitigate this risk, we are using google’s Firebase, which is FIPS 140-3 certified.
GDPR, General Data Protection Regulation - European Union	This is the major governing regulation about user data in the EU. Specifically, Chapter 4 article 32 says “...shall implement appropriate technical and organisational measures to ensure a level of security appropriate to the risk. We are using Firebase by Google, which we trust is compliant with this regulation.

Functions	Means			
Interacting With User	Java -Android App	Swift -Apple App	Python-IDE Tkinter	
Identifying Art	Search Bar	QR Code	<b>Machine Learning</b>	Art Historian
Storing Data	<b>Database-(SQL)</b>	Excel File	CSV	
App Hosting	IU Servers	<b>Amazon Servers</b>	Personal Server	IBM
Connecting Between App and Server	Flask	AJAX-Client	JSON-Server	Django



## Function

## Means

Art Identifier -----> ML App / QR Code

Art Info -----> SQL database

User Info -----> SQL database

Mobile Application -----> Figma

App Hosting -----> AWS

Database Hosting -----> AWS



# Flow of Execution

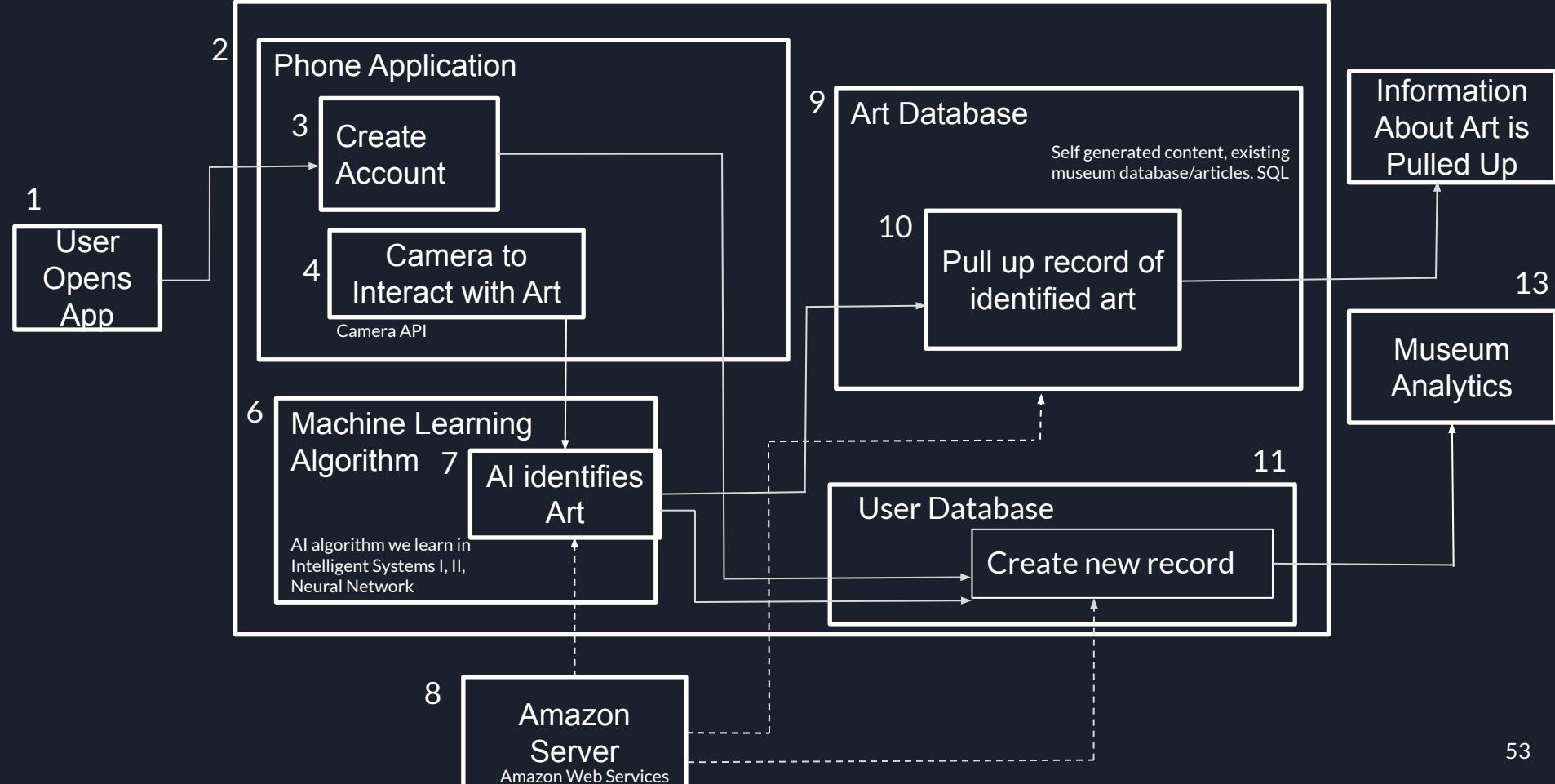
- [1] User Creates an account (Mobile Application)
- [2] Creates record of user (User Database)
- [3] User interacts with Art (Mobile Application)
- [4] Identifier identifies the piece (Art Identifier)
- [5] Information about art is pulled from art database and displayed to user (Mobile Application)
- [6] Creates record of art interaction (User Database)



# Box Diagrams

# Glass Box

12



53



Competition,  
Market Study,  
Intended Market

# Closest Mobile Competitor- DailyArt

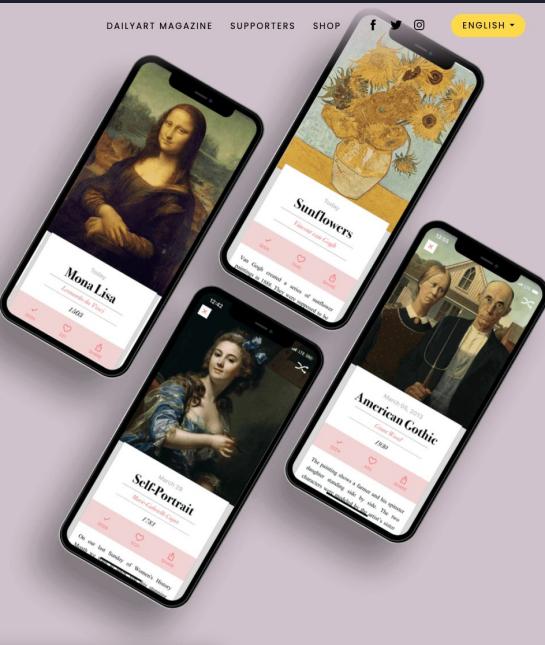
**DailyArt**  
Your daily dose of art

**Every day get one piece of fine art with a short story about it**

Straight to your phone or tablet, for free

[Download on the App Store](#) [GET IT ON Google Play](#)

App Store featured      Google Play featured



DailyArt is a mobile application recommending a new 2D art piece to its users everyday.

DailyArt does not provide a customized user experience, or the opportunity for community commentary.

The art is curated and commented on by an in house curruator.

# Cleveland Museum of Art - Gallery One



Gallery One consists of CMA's permanent collection, and allows for a technology driven user selected interaction with each work.



Gallery One removes the curation aspect, it also does not allow for individual commentary.

# Interviews with Industry Experts



Melody Barnett  
Deusner, PhD  
Expert in Museum  
Industry



Maria  
Domene-Danés, PhD  
Art Interpretation  
Expert



Margaret Graves, PhD  
Museum Industry and  
Interpretation Expert



Faye R. Gleisser, PhD  
Museum Curation  
Expert

# Interviews with Industry Experts- Key Takeaways

- Data analytics for museums
  - Our app can provide engagement analytics museums can use in their own grant applications.
- Database services would update small museums
  - A centralized database would allow one museum to ‘advertise’ themselves to a user.
  - A centralized database would show relationships between art at different museums.
- Gives a voice to controversial objects
  - Civil war monuments that cannot be easily moved or commented on
  - Provides a voice to things like sculptures outside of a museum where no little paragraph is present

# Interviews with Industry Experts- Key Takeaways

- Art historians are probably less likely to use the app
  - This app provides a safe place for the novice art enthusiast to engage and participate in art commentary
  - People behave strangely in museums, and are hesitant to engage with curators or their resources
- Many voices can be heard in one place
  - Traditional curation is limited to 200 words curated by a limited number of individuals, or even one individual
  - Users can select pieces and contribute to a growing dialogue about the pieces
- Update on the isolating earbud audio experiences many museums implemented in the mid 2000's



# Safety & Ethics

# Safety

## Maintaining and Protecting User Data

- Affected Blocks
  - 2 Phone Application Hasha
- Mitigation
  - Make sure data is stored on secured servers (Amazon Web Services)
  - Store data with encryption
- Applicable Standards
  - 15 U.S. Code 41 et seq. Law that protects user data, privacy laws about user data
  - 18 U.S. Code 2721 et seq. Law that governs privacy and disclosure of gathered personal information

# Safety

## Overheated Battery

- Affected Blocks
  - 2 Phone Application Hasha
  - 6 Machine Learning Algorithm Ryan
- Mitigation
  - Test code thoroughly especially for things like infinite bugs
  - Test length usage, leave open a long time
- Applicable Standards
  - UL 1020 Thermal cutoffs for use in electrical appliances and components

# Ethics

## Proper Citation of Art and Descriptors

- Affected Blocks
  - 2 Phone Application Hasha
  - 9 Database Ryan
- Mitigation
  - Specifically cite each work and its associated institution properly
  - Make sure the art we include is able to even be reproduced on our application (not copyrighted or permission explicitly denied)
  - Get written permission to use database description of the art
- Applicable Standards
  - 17 U.S. Code 106A. Law that protects claim to work



# Gantt Chart

Task/Month	Oct	Nov	Dec
Research			
December Presentation			
Photos for Database			
Database Design/Development			
App Development			
December Proposal			

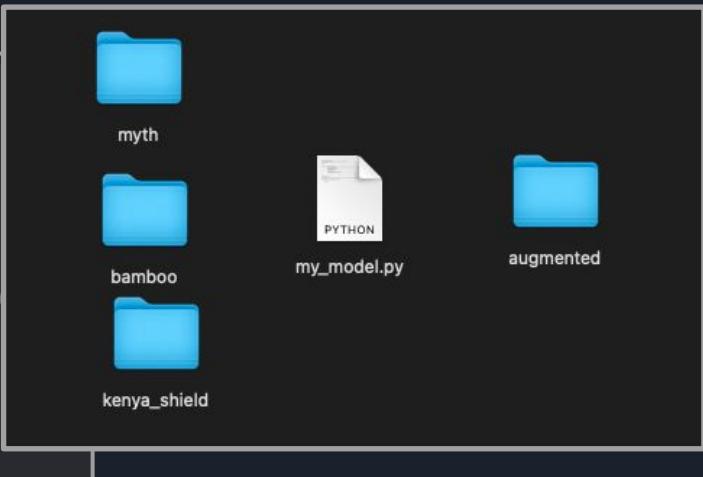


# Progress & First Results

```

1 import tensorflow as tf
2 from keras.preprocessing.image import ImageDataGenerator
3 from skimage import io
4 import os
5
6
7 datagen = ImageDataGenerator(
8     rotation_range=10, # 0 to 360 degrees the duplicates will be removed
9     width_shift_range=0.1,
10    height_shift_range=0.1,
11    shear_range=.05,
12    zoom_range=.25,
13    brightness_range=(1.6,1.6),
14    fill_mode='constant', cval=128)
15
16
17 i = 0
18 hello = os.getcwd()
19 for batch in
20     datagen
21         .flow_from_directory(directory=hello,batch_size=64,target_size=
22             (128,128),color_mode='rgb',save_to_dir='augmented',save_prefix='aug'
23             ,save_format='png'):
24     i += 1
25     if i > 1:
26         break

```



```

Last login: Sat Dec  4 15:16:26 on console
[(base) hasha@MacBook-Pro ~ % cd Desktop
[(base) hasha@MacBook-Pro Desktop % cd images
[(base) hasha@MacBook-Pro images % python3 my_model.py
Found 12 images belonging to 5 classes.

```

Process Name	% CPU
python3.8	99.9

```
1 import tensorflow as tf
2 from keras.preprocessing.image import ImageDataGenerator
3 from skimage import io
4 import os
5
6
7 datagen = ImageDataGenerator(
8     rotation_range=10, # 0 to 360 degrees the duplicates will be rotated
9     width_shift_range=0.1,
10    height_shift_range=0.1,
11    shear_range=.05,
12    zoom_range=.25,
13    brightness_range=(1.6,1.6),
14    fill_mode='constant', cval=128)
15
16
17 i = 0
18 hello = os.getcwd()
19 for batch in
20     datagen
21         .flow_from_directory(directory=hello,batch_size=64,target_size=
22             (128,128),color_mode='rgb',save_to_dir='augmented',save_prefix='aug'
23         ,save_format='png'):
24             i += 1
25             if i > 1:
26                 break
27
28 |
```







Best: 94.5%

```
> vjpyg
Epoch 2/30
2/20 [=====] - 61s 3s/step - loss: 1.8186 - accuracy: 0.3088 - val_loss: 1.1165 - val_accuracy: 0.5031
Epoch 3/30
2/20 [=====] - 63s 3s/step - loss: 2.0081 - accuracy: 0.2250 - val_loss: 2.0286 - val_accuracy: 0.1656
Epoch 4/30
2/20 [=====] - 63s 3s/step - loss: 2.0054 - accuracy: 0.2236 - val_loss: 2.0569 - val_accuracy: 0.1875
Epoch 5/30
2/20 [=====] - 62s 3s/step - loss: 2.0254 - accuracy: 0.2688 - val_loss: 1.8735 - val_accuracy: 0.2656
Epoch 6/30
2/20 [=====] - 62s 3s/step - loss: 1.9074 - accuracy: 0.3109 - val_loss: 1.9358 - val_accuracy: 0.3469
Epoch 7/30
2/20 [=====] - 61s 3s/step - loss: 1.3815 - accuracy: 0.5172 - val_loss: 0.3185 - val_accuracy: 1.0000
Epoch 8/30
2/20 [=====] - 62s 3s/step - loss: 0.7405 - accuracy: 0.7875 - val_loss: 0.0138 - val_accuracy: 1.0000
Epoch 9/30
2/20 [=====] - 62s 3s/step - loss: 0.1883 - accuracy: 0.9465 - val_loss: 0.0167 - val_accuracy: 1.0000
Epoch 10/30
2/20 [==>.....] - ETA: 41s - loss: 0.1449 - accuracy: 0.9375
```

Worst:  
About Equal  
Distribution  
of Percentages

```
vjpyg
pg > vjpyg
> vjpyg
DEBUG CONSOLE
10/10 [=====] - 17s 2s/step - loss: 1.0715 - accuracy: 0.4305 - val_loss: 1.0732 - val_accuracy: 0.4000
Epoch 11/20
10/10 [=====] - 17s 2s/step - loss: 1.0908 - accuracy: 0.3940 - val_loss: 1.0719 - val_accuracy: 0.4000
Epoch 12/20
10/10 [=====] - 17s 2s/step - loss: 1.1034 - accuracy: 0.3510 - val_loss: 0.9470 - val_accuracy: 0.8000
10/10 [=====] - 18s 2s/step - loss: 1.0839 - accuracy: 0.4031 - val_loss: 1.0716 - val_accuracy: 0.4000
Epoch 14/20
10/10 [=====] - 18s 2s/step - loss: 1.0817 - accuracy: 0.4125 - val_loss: 1.2267 - val_accuracy: 0.0000e+00
10/10 [=====] - 17s 2s/step - loss: 1.0853 - accuracy: 0.4040 - val_loss: 0.9953 - val_accuracy: 0.6000
Epoch 16/20
10/10 [=====] - 17s 2s/step - loss: 1.0691 - accuracy: 0.4437 - val_loss: 1.0697 - val_accuracy: 0.4000
Epoch 17/20
10/10 [=====] - 17s 2s/step - loss: 1.0896 - accuracy: 0.3841 - val_loss: 1.0326 - val_accuracy: 0.6000
Epoch 18/20
10/10 [=====] - 18s 2s/step - loss: 1.0851 - accuracy: 0.4094 - val_loss: 1.0693 - val_accuracy: 0.4000
Epoch 19/20
10/10 [=====] - 16s 2s/step - loss: 1.0853 - accuracy: 0.4205 - val_loss: 1.2458 - val_accuracy: 0.0000e+00
Epoch 20/20
10/10 [=====] - 19s 2s/step - loss: 1.0968 - accuracy: 0.3688 - val_loss: 1.0311 - val_accuracy: 0.6000
[[0.27148393 0.326136 0.40238002]]
PS C:\Users\18124\Desktop\images>
```

# Sustainability



**Physical:** QR code

**Mitigation:** 100% recyclable (paper or plastic), and reusable for many years as long as website is maintained

**Software:** Applications, databases, website

**Mitigation:** Monitoring size of applications and power consumption of applications as we build, joining cluster of computers for website hosting, optimizing data stored in database (removing unnecessary data)



# Key Takeaways

- The business value is in:
  - a centralized database
  - a user generated comment section
  - unique analytics
- The value the AI brings is:
  - for marketing
  - a streamlined user experience

# Server/Database Upload

- Completely packaged tkinter python application
- User interface gives client the ability to upload images, audio, and txt files to the database
- Server code automatically generates a new http endpoint when new database entry is committed.

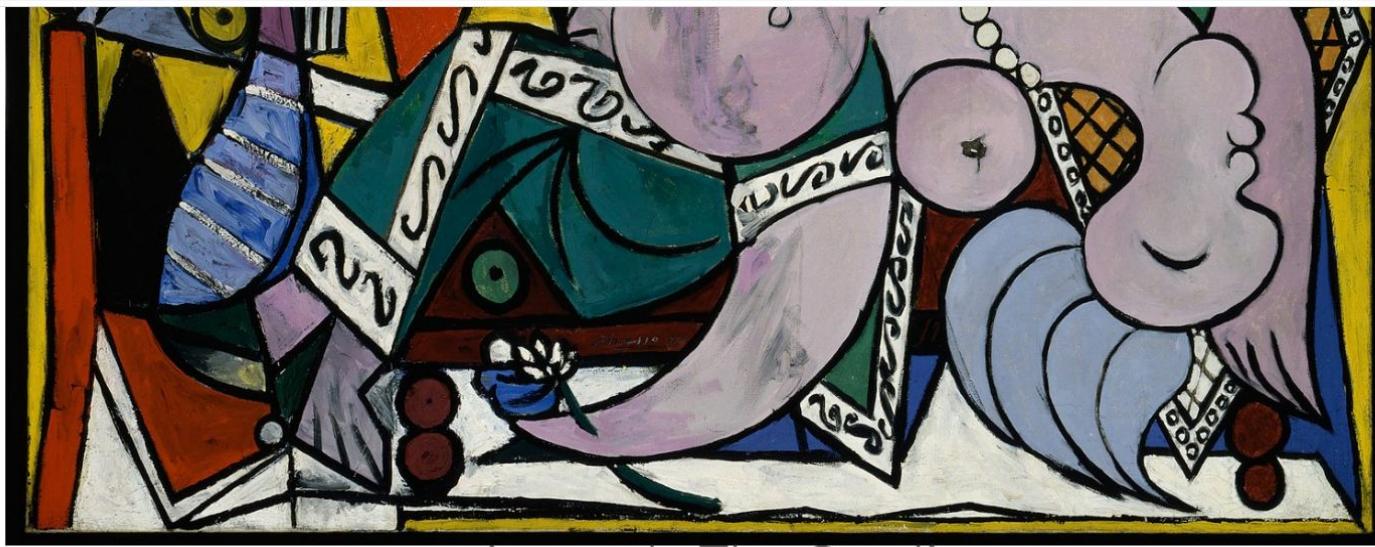
Artwork Name	The Studio
Artist Name	Pablo Picasso
Date	1934
Description	<input type="button" value="Upload .txt Here"/> C:/Users/18124/Desktop/capstone_design/studio.txt
Select Image Upload File	<input type="button" value="Upload .jpg Here"/> C:/Users/18124/Desktop/capstone_design/static/the-studio.jpg
Select MP3 Upload File	<input type="button" value="Upload .mp3 Here"/> C:/Users/18124/Desktop/capstone_design/static/the_studio_audio.mp3

New http endpoint will be  
rlockard.pythonanywhere.com/{Artwork Name}



Once you click the export to database button, all information is sent to database through SSH tunnel

New http endpoint created



Artwork: The Studio  
Artist : Pablo Picasso  
Date: 1934

▶ 0:00 / 0:47

The Studio is one of Pablo Picasso's major works addressing the theme of artist and model. A provocative intersection of style, subject, and biography, the work portrays Picasso standing at his easel, while his much younger model and mistress, Marie-Therese Walter, reclines on a sofa. The exuberant colors and bold shapes reflect Picasso's continuing development of the radical Cubist vocabulary he and Braque pioneered before World War I. Yet the composition also reinforces traditionally misogynist perceptions about male creativity and female passivity. In this work, the male artist's realm is defined by angular geometries evoking rational intellectualism, while Marie-Therese's form is one of rounded, organic shapes that suggest her creativity is linked to her sexuality.

L—	Description	Requirement	Notes
B1	Intended Market Geography	North America, EU. Or other English Speaking Museum Entities, pre-approve	
B2	Intended Market Demography	B2B aspect: Small to medium sized museums in need of database services and/or increasing public interaction  Public aspect: Any museum goer with a smart phone (apple or android)	
B3	Ave List Sales Price - ASP	B2B: \$15K per 1,000 pieces (?), ongoing data hosting fees  Public aspect: free download with no premium features	
B4	Estimated Annual Volume	# 1 one new contract per month, 1 month turn around	# of units/year
B5	Max Material Cost (Parts)	\$ 5000 one time camera costs  \$ X server costs (?)  \$ ongoing labor costs, per diem when traveling to museums  \$ ongoing copywriting costs (might be free if user generated)	\$20/hr, it takes about 1 hour per 100 pieces, about \$2000 labor costs for salary and per diem for 1 week for 15,000 pieces  AI computing services will be much more
B6	Max Assembly & Test Cost	\$ N/A	Must be less than ASP
B7	Product Life Target	Evaluate need for updates for app every 6 months or sooner if apparent	Will be adjusted after reliability evaluation
B8	Target Warranty Length	Charge museums for updated database services they request, open warranty for 3 months post contract	Will be adjusted after reliability evaluation
B9	Min Life Cycle Period	Support as long as business is profitable and museum is paying data hosting fees	Will be adjusted after reliability evaluation