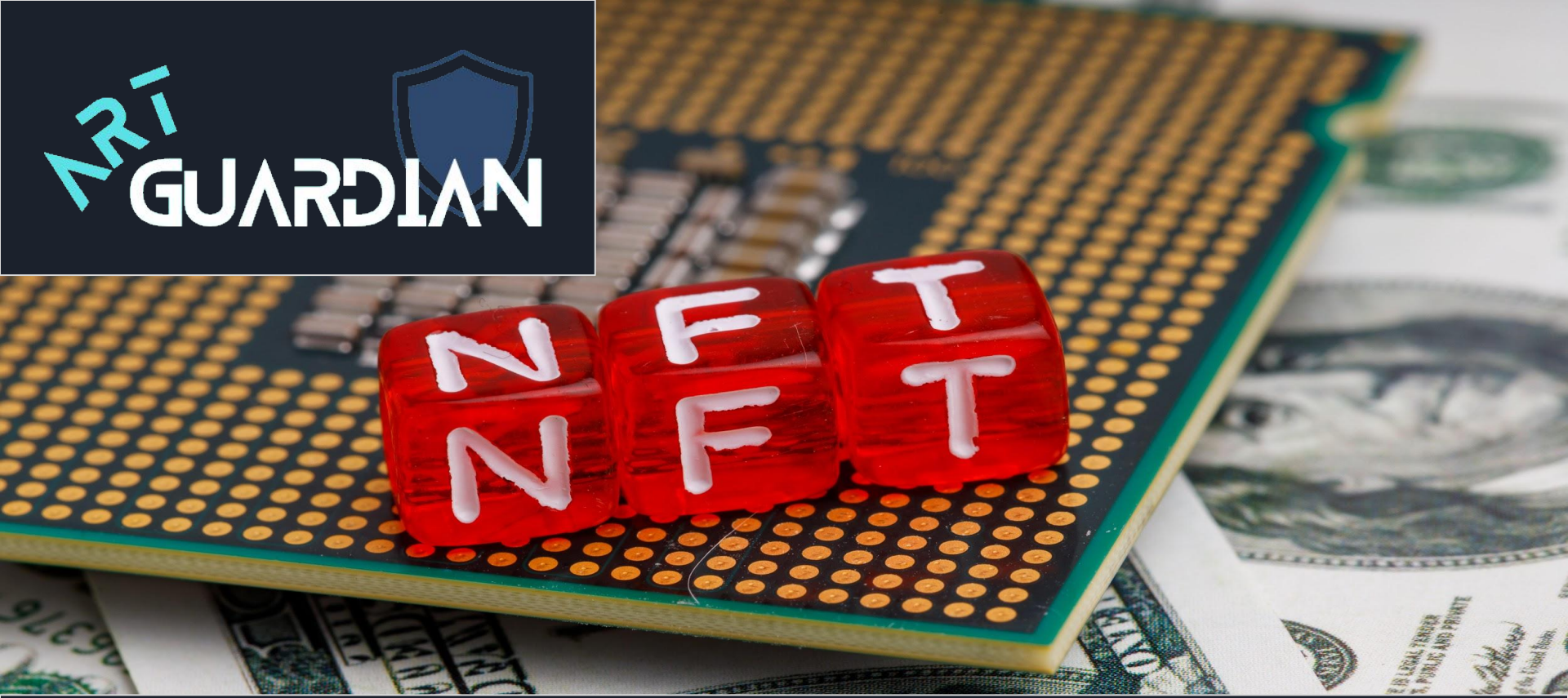
ART GUARDIAN



Team Blue

CS411 Fall 2022 Prototype Presentation #2

Table of Contents

Team Blue

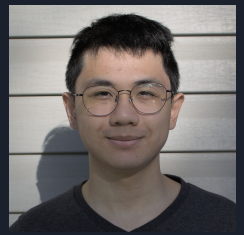
Carlo Diaz
Frontend/Webmaster

Paolo Ihde
Frontend/Backend

Brandon Kidd
Database Specialist/Backend

Michael Thompson
Backend/Frontend

Autumn Roberts
Team Leader/Frontend



Tobin Zheng
Backend / Web Master

Spencer Hite
Database Specialist/Backend

Sprint 0 & 1 Task Breakdown

Team:

- Setup Dev Environments
- Setup Cloud Services and Database

Sprint 2 Task Breakdown

Front-End:

- Create Landing Pages
- Create Sign in and login pages

Back-End:

- Tested an Image Matching algorithm

Database:

- Finish GraphQL data table for Settings.
- Start GraphQL data table to hold a images' metadata.
- Generate common GraphQL mutations and queries to read/write to DynamoDB database.
- Launch AWS S3 bucket to hold User Artwork images.

Sprint 3 Task Breakdown

Front-End:

- Create user art upload and gallery
- Create skeleton code for DMCA Generation
- User Account Security and Encryption
- Login Cookies

Back-End:

- Create whitelisting functionality
- Implement Image Matching as part of back-end
- Implement detection of Art on NFT Marketplaces
- Create fake NFTs

Database:

- Finish GraphQL data table to hold a images' metadata.
- Start GraphQL data table to hold DMCA copyright information.
- Update GraphQL queries and mutations to include any new or modified data tables.

Sprint 4 Task Breakdown

Front-End:

- Execute DMCA Takedown Requests
- Account Settings
- Account Profile

Back-End:

- DMCA takedown generation
- Integrate and test Gmail API

Database:

- Finish GraphQL data table to hold DMCA copyright information
- Start GraphQL data table to hold notification metadata
- Update GraphQL queries and mutations to include any new or modified data tables.

Sprint 5 Task Breakdown

Front-End:

- Art Protection Guidelines
- App Tutorial
- Generate Push Notification

Back-End:

- DMCA Cataloging
- DMCA Tracking

Database:

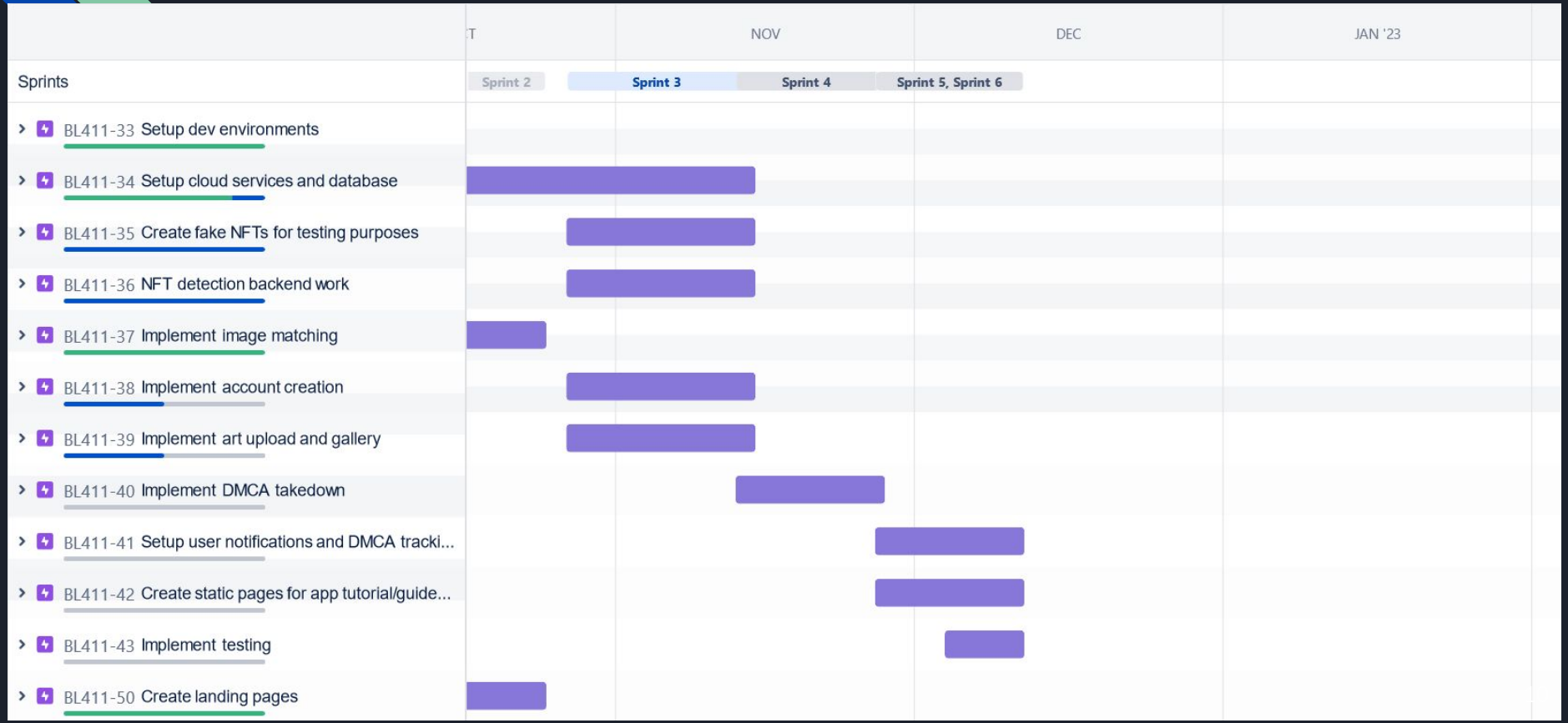
- Finish GraphQL data table to hold Notification metadata
- Database maintenance

Sprint 6 Task Breakdown

Team:

- Create Testing Environment
- Prototype Test
- Project Cleanup

Features Roadmap



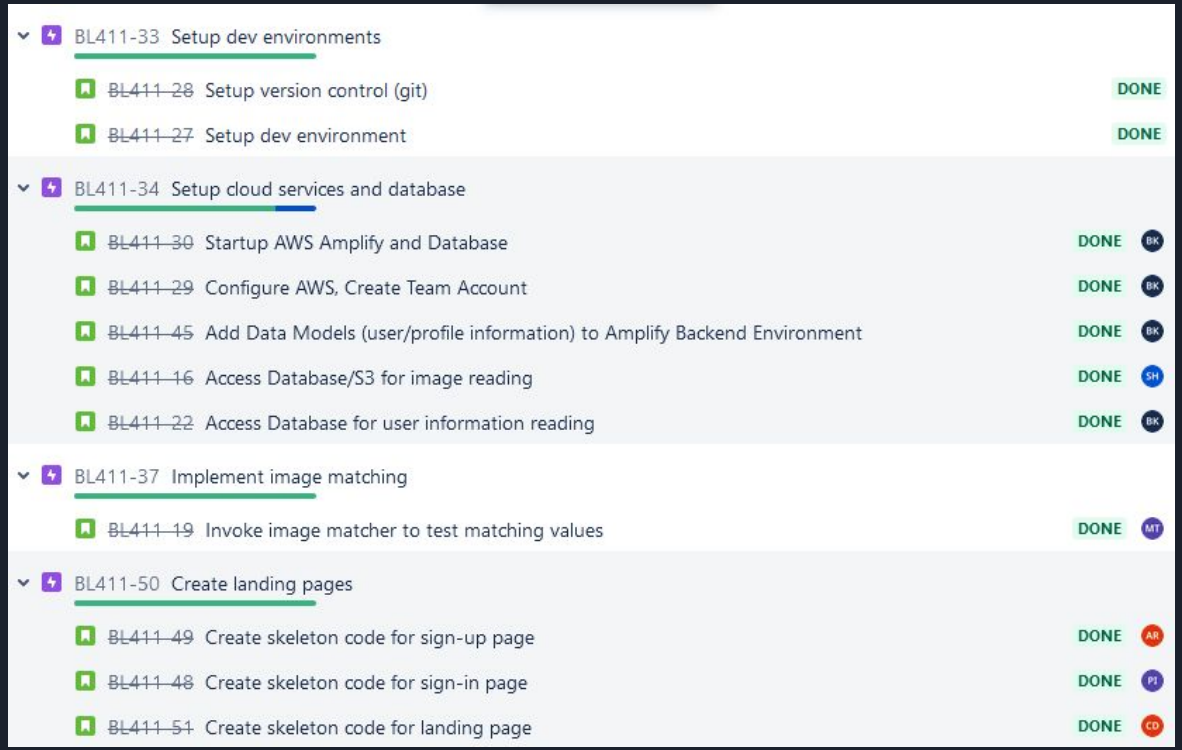
Team Roles and Responsibilities

- Scrum Masters - Autumn, Tobin
  - Sprint planning
  - Alternating leaders for team meetings (Scrum, Stand-ups, etc.)
- Frontend Developer - Carlo, Autumn, Paolo
  - AWS Amplify, ReactJS, CSS development
  - App Feel and overarching themes
  - User Accessibility
- Backend Developer - Michael, Tobin
  - Lambda, Step Functions, Python, OpenCV
  - Implementation of algorithms:
    - Whitelisting
    - Counterfeit art detection
    - Image matching
    - DMCA
- Database Developer - Brandon Kidd, Spencer Hite
  - AWS Amplify Database implementation and maintenance.
  - GraphQL mutations/queries implementation and maintenance.

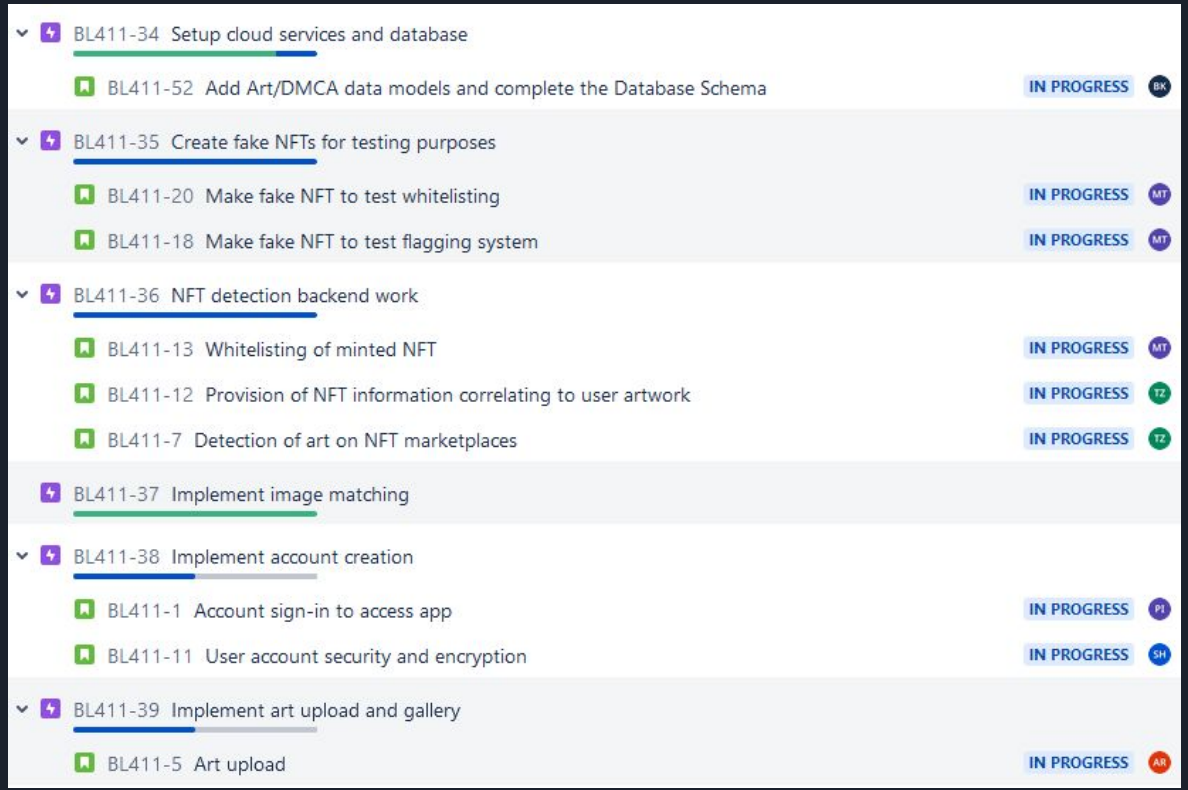
Accountability Synopsis

- Keep track of commits and pushes
- Assigning members to tasks on Jira
- Honor System:
  - I pledge to support the Honor System of Old Dominion University. I will refrain from any form of academic dishonesty or deception, such as cheating or plagiarism. I am aware that as a member of the academic community it is my responsibility to turn in all suspected violations of the Honor Code. I will report to a hearing if summoned.
- Weekly meetings to discuss progress

Features Completed



Features in Progress



Prototype Changes

- Upload and Gallery now one page
- Created Logo
- Changed gallery database from Dynamo to S3
- Utilized amplify features
- Major changes to Art Detection feature:
  - Images will not be batched
  - Future of periodic searches unclear
- Adjusted weights for Image Matching Algorithm

Near-term Priorities

- Login Cookies
- Integrate algorithms into workflow / step functions
- Finish database implementation

Concerns

- Art Gallery separating and showing art based on user
- Time constraints
- Team Availability
- Difficulty curve
    - Step Functions / AWS Lambda