**“Arti-GaN”**

FA21 CSCI-P445 Capstone RF 3

Nathanael L. Mann, Mark Bridgewater, Joshua Carrier

Sponsor: Chris Sexton

12 October 2021

**Software Architecture Specification**

**1.1 Overview**

Arti-GaN comprises two primary subsystems, the Front-End Website, and the Back-End GaN generator/trainer. A Brief description of these systems is listed below, and more in-depth Architecture Specification can be found in sections 1.2 onwards.

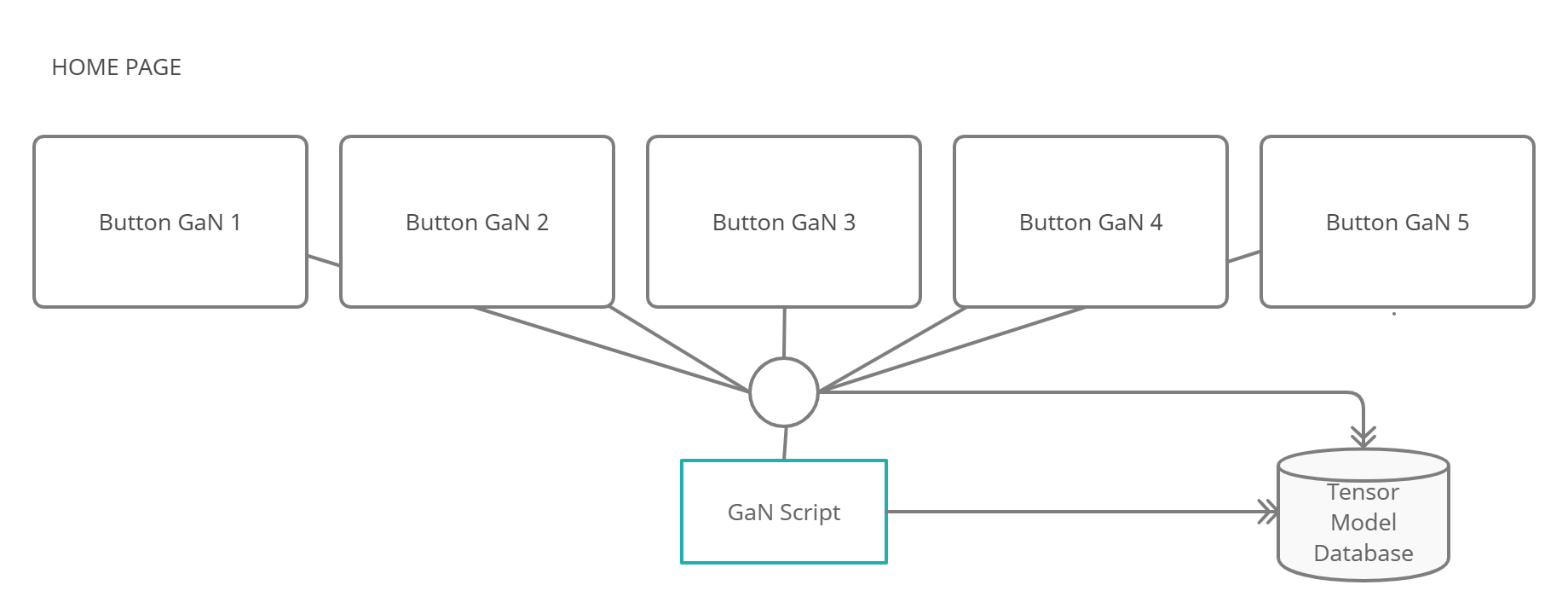
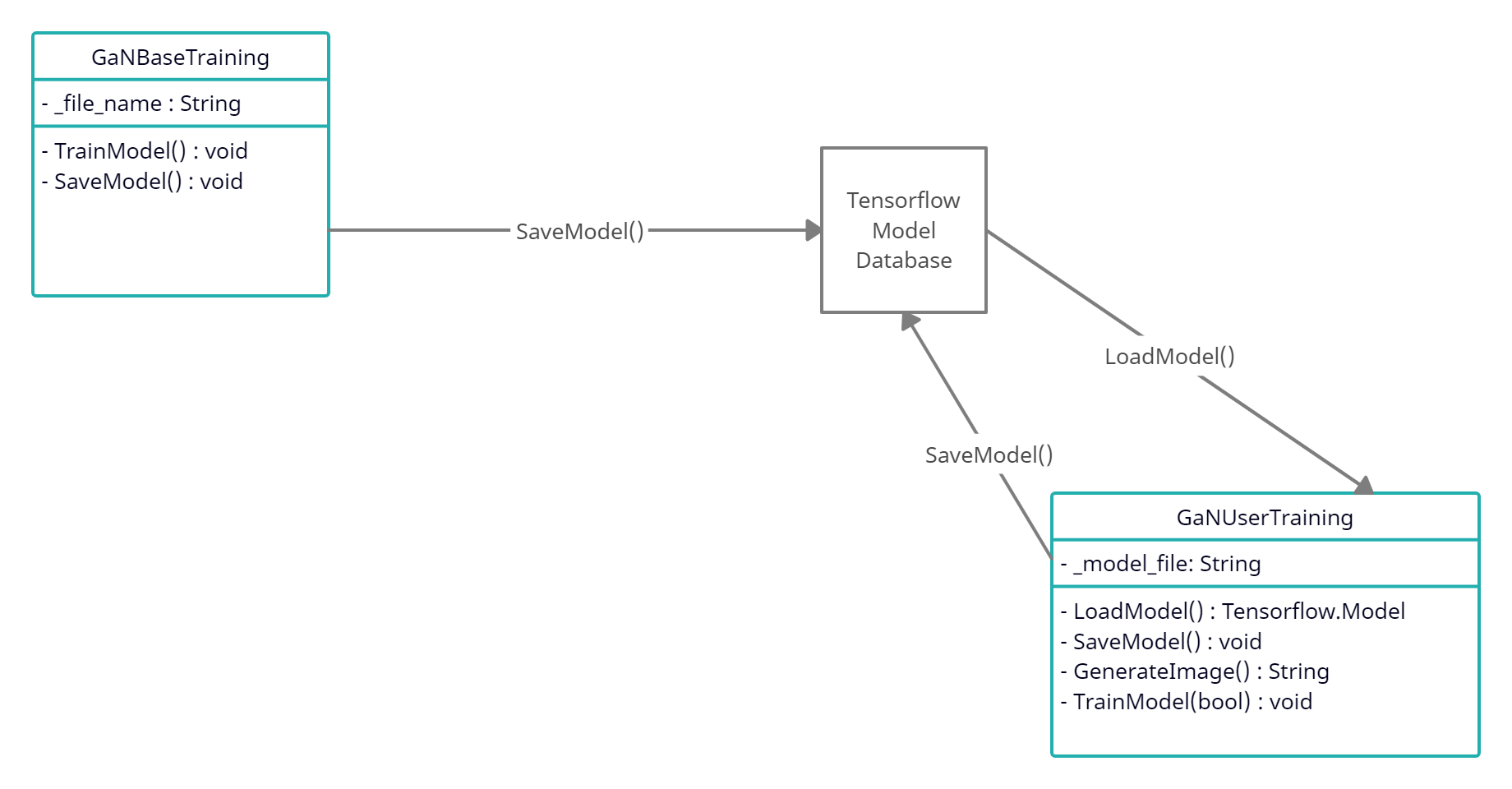
**1.1.1 Front-End Website**

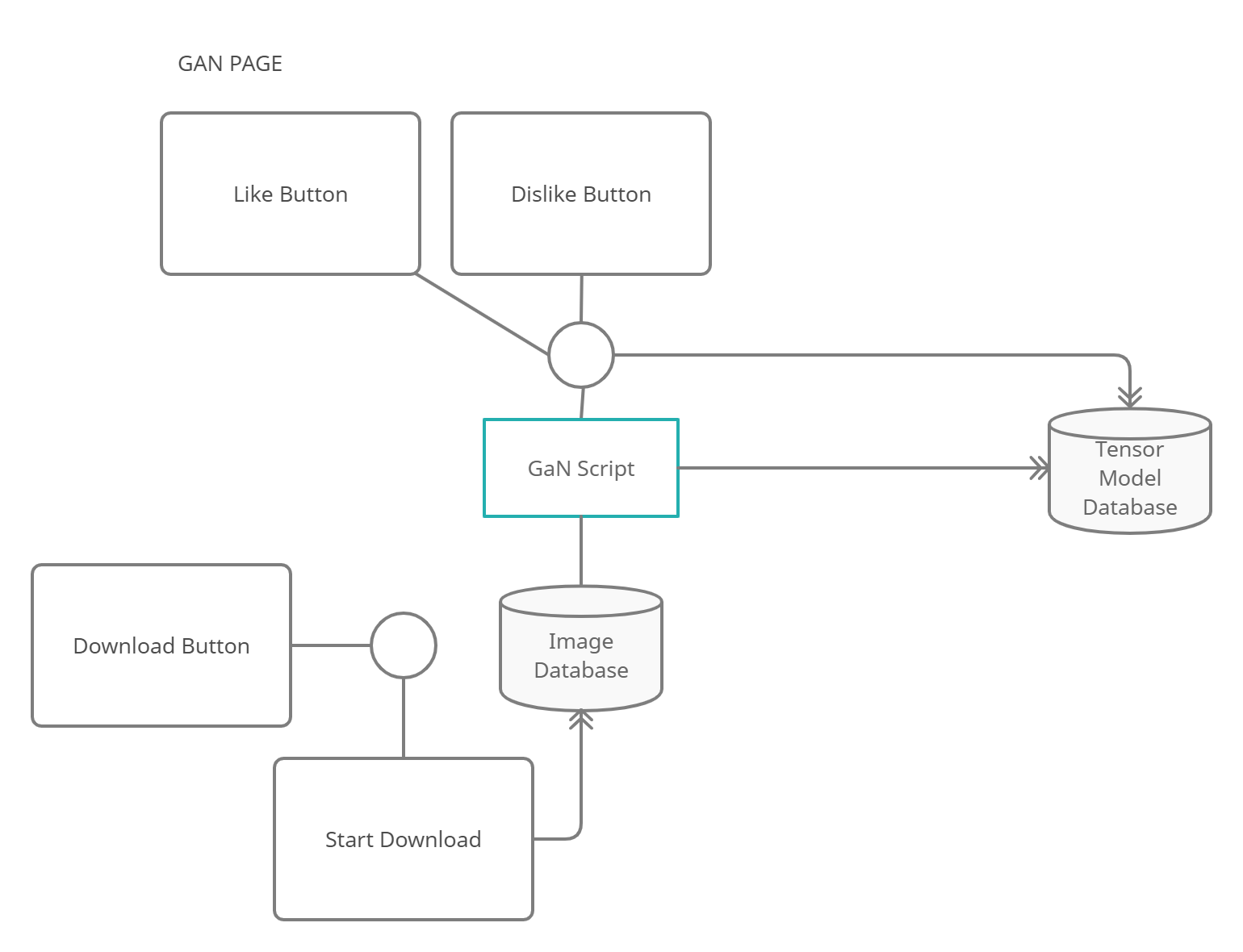
The Front-End Website consists of 5 GaN’s that visitors can interact with. These GaN’s will generate images derived from its training set that visitors can rate as either ‘Good’ or ‘Bad.’

**1.1.2 Back-End GaN**

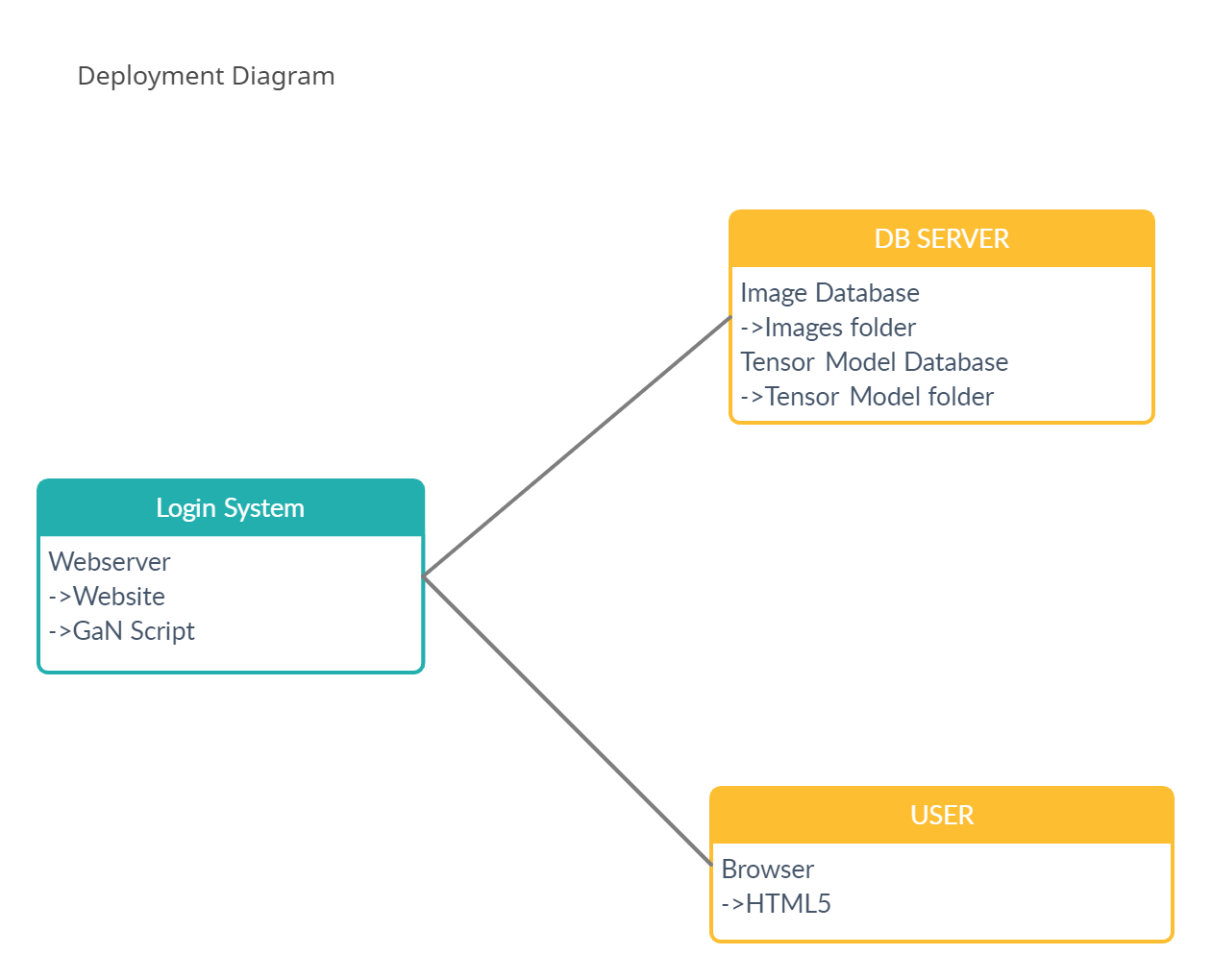
The Back-End GaN generator is the software program that generates and trains GaN’s based on a set of input images. The output of this software is then moved to the Front-End Site where it can be interacted with.

**1.2 Subsystem decomposition**



****

**1.3 Hardware/software mapping**



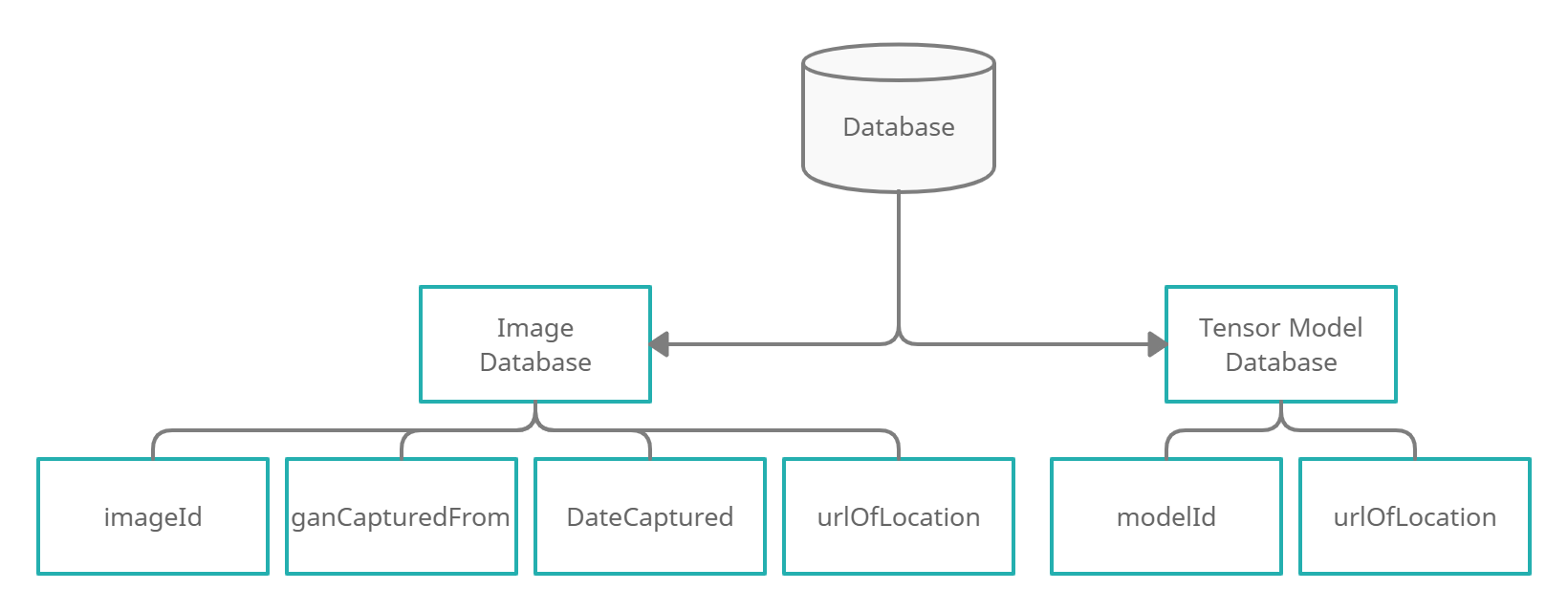
**1.4 Persistent data management**

Persistent Elements

* GaN Structure → Backend, but will need to be stored on the frontend. Technically only half the GaN will need to be stored on-site (the generator)
* Daily “Final” Image → 5 Images per day (one per GaN)

XFS file systems for holding the images, tensor models, and the HTML/CSS/ts files.

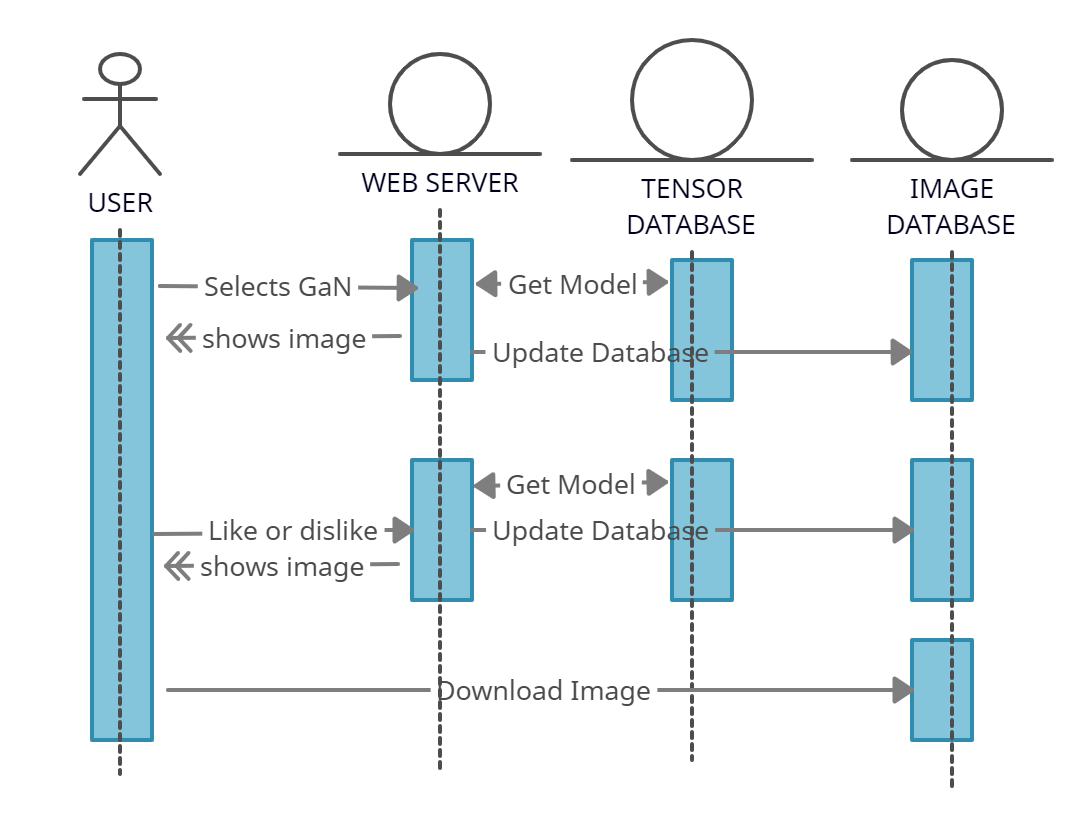
The XFS will also be used to hold scripts for running the GaN.



**1.5 Access control and security**

Anyone with the website URL will be able to log on and interact with the GaN’s. To prevent rapid input, a small delay will be issued between reviews of a generated image both to prevent spam and ensure that the user considers the quality of the artwork before submitting it. Since the reviews are subjective, accidentally, or intentionally “incorrect” responses are not a problem. No data will be collected from the end-users (aside from their input to the AI) so there is no need for complex security features on the front end.

**1.6 Global software control**



**1.7 Boundary conditions**

Once the GaN Generation software is developed, the website will be launched, hosting 5 of the GaN’s for users to interact with. The initial plan is for startup to happen by late January 2022, and the website will be live for the duration of the Spring Semester, at which point the site will be terminated, but the source code will remain on GitHub where it could be hypothetically reinstated at a later date with little effort. Long-term data storage is not a concern within the scope of this research project, since the site only needs to be active long enough to gather sufficient data.

**Breakdown of individual contributions**

* Nathanael - Product Design & Documentation
* Mark - Website and Repository Management
* Joshua - Technical Research & Development

**Key Personnel information**

Nathanael L. Mann (Project Manager)

Email: mannnl@iu.edu

Git: IAMNameless9t9

Mark Bridgewater (Primary Front End Developer)

Email: mobridge@iu.edu

Git: MarkoBridgewater

Joshua Carrier (Primary Back End Developer)

Email: carriejr@iu.edu

Git: jcarrier0026