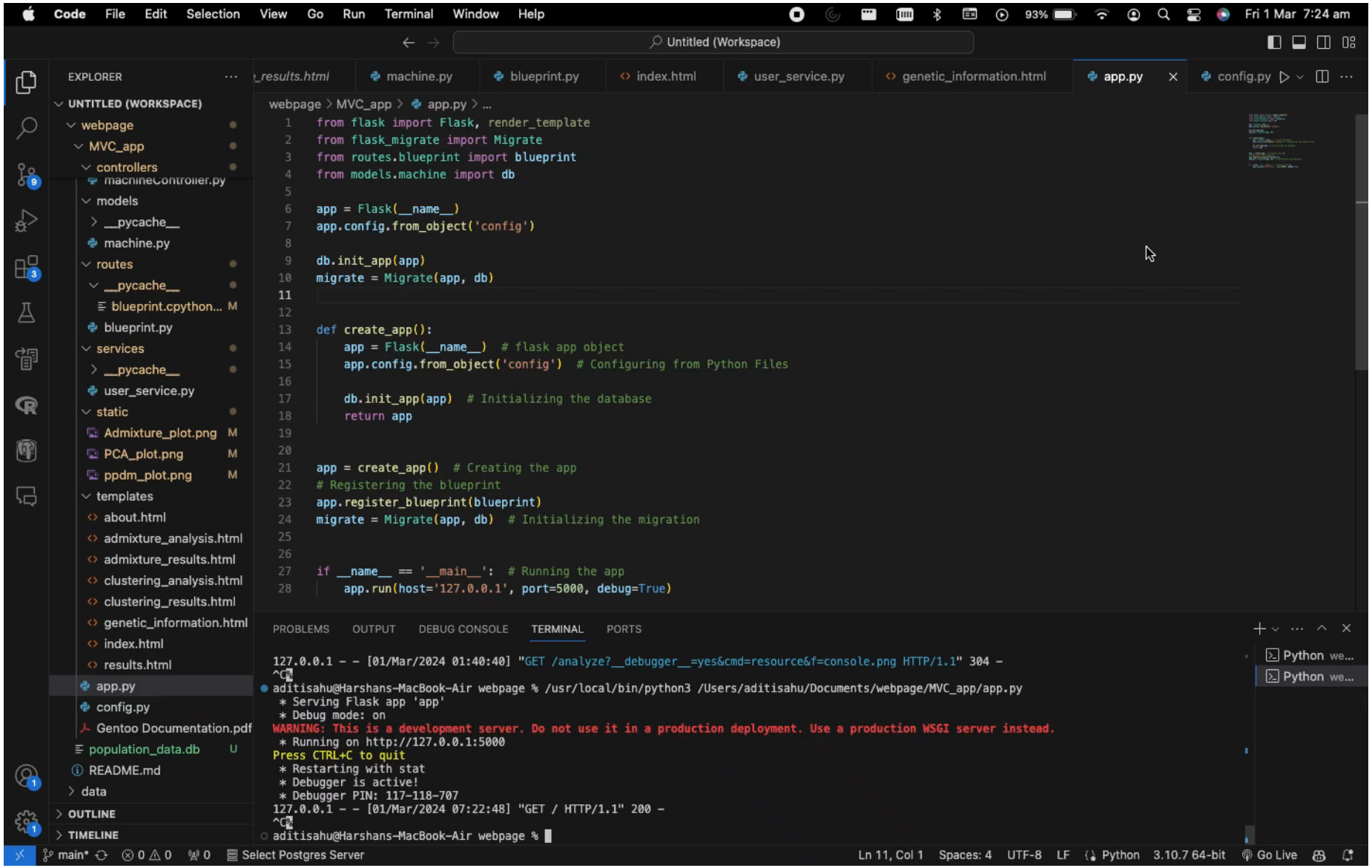
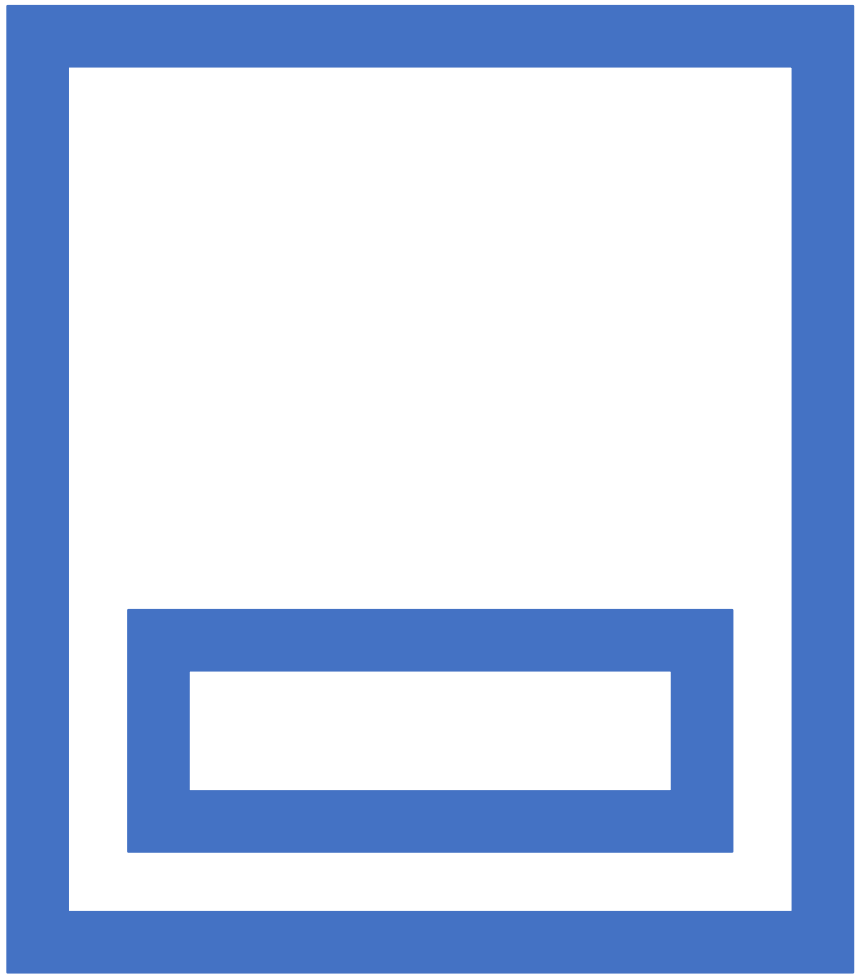
Gentoo App

Collaborators: Harshan, Jeremiah, Osman


Code File Edit Selection View Go Run Terminal Window Help
Fri 1 Mar 7:24 am
Untitled (Workspace)
EXPLORER
UNTITLED (WORKSPACE)
webpage
MVC_app
controllers
models
routes
services
static
templates
app.py
config.py
Gentoo Documentation.pdf
population_data.db
README.md
data
OUTLINE
TIMELINE
machine.py
blueprint.py
index.html
user_service.py
genetic_information.html
app.py
config.py
webpage > MVC_app > app.py > ...
from flask import Flask, render_template
from flask_migrate import Migrate
from routes.blueprint import blueprint
from models.machine import db
app = Flask(__name__)
app.config.from_object('config')
db.init_app(app)
migrate = Migrate(app, db)
def create_app():
app = Flask(__name__) # flask app object
app.config.from_object('config') # Configuring from Python Files
db.init_app(app) # Initializing the database
return app
app = create_app() # Creating the app
# Registering the blueprint
app.register_blueprint(blueprint)
migrate = Migrate(app, db) # Initializing the migration
if __name__ == '__main__': # Running the app
app.run(host='127.0.0.1', port=5000, debug=True)
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
127.0.0.1 - - [01/Mar/2024 01:40:40] "GET /analyze?__debugger__=yes&cmd=resource&f=console.png HTTP/1.1" 304 -
aditisahu@Harshans-MacBook-Air webpage % /usr/local/bin/python3 /Users/aditisahu/Documents/webpage/MVC_app/app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 117-118-707
127.0.0.1 - - [01/Mar/2024 07:22:48] "GET / HTTP/1.1" 200 -
aditisahu@Harshans-MacBook-Air webpage %
Python we...
Python we...
main*
Select Postgres Server
Ln 11, Col 1
Spaces: 4
UTF-8
LF
Python
3.10.7 64-bit
Go Live

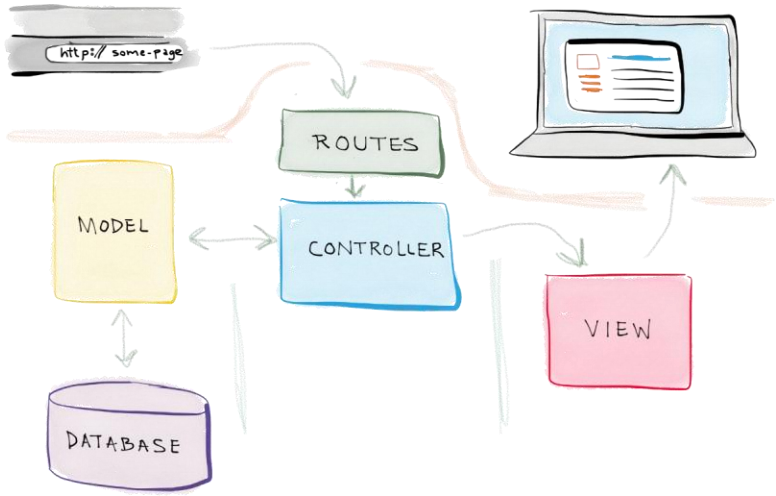
What are the functions of this app?

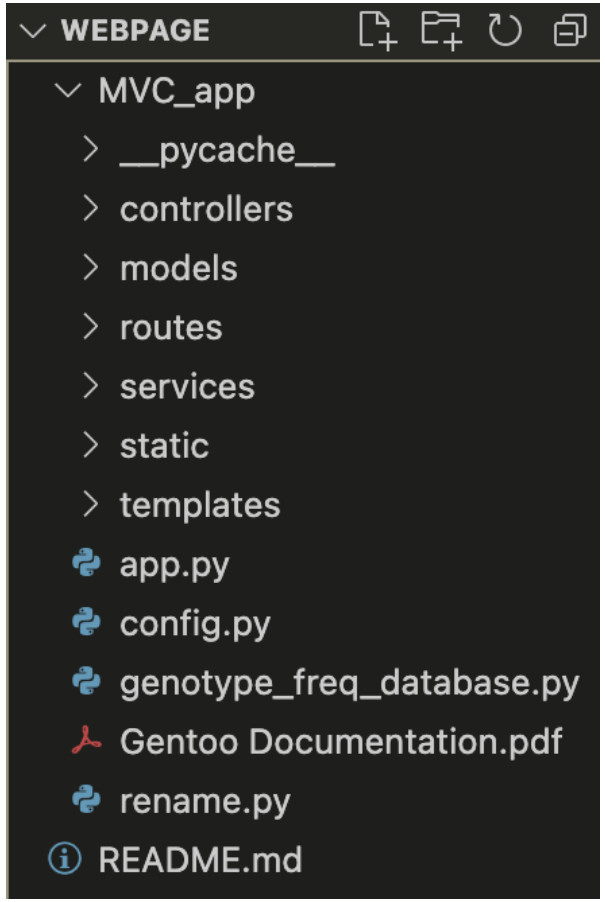
- **Clustering and Admixture Analyses**
- **Allele & Genotype Frequency**
- **Pairwise Population Genetic Differentiation**



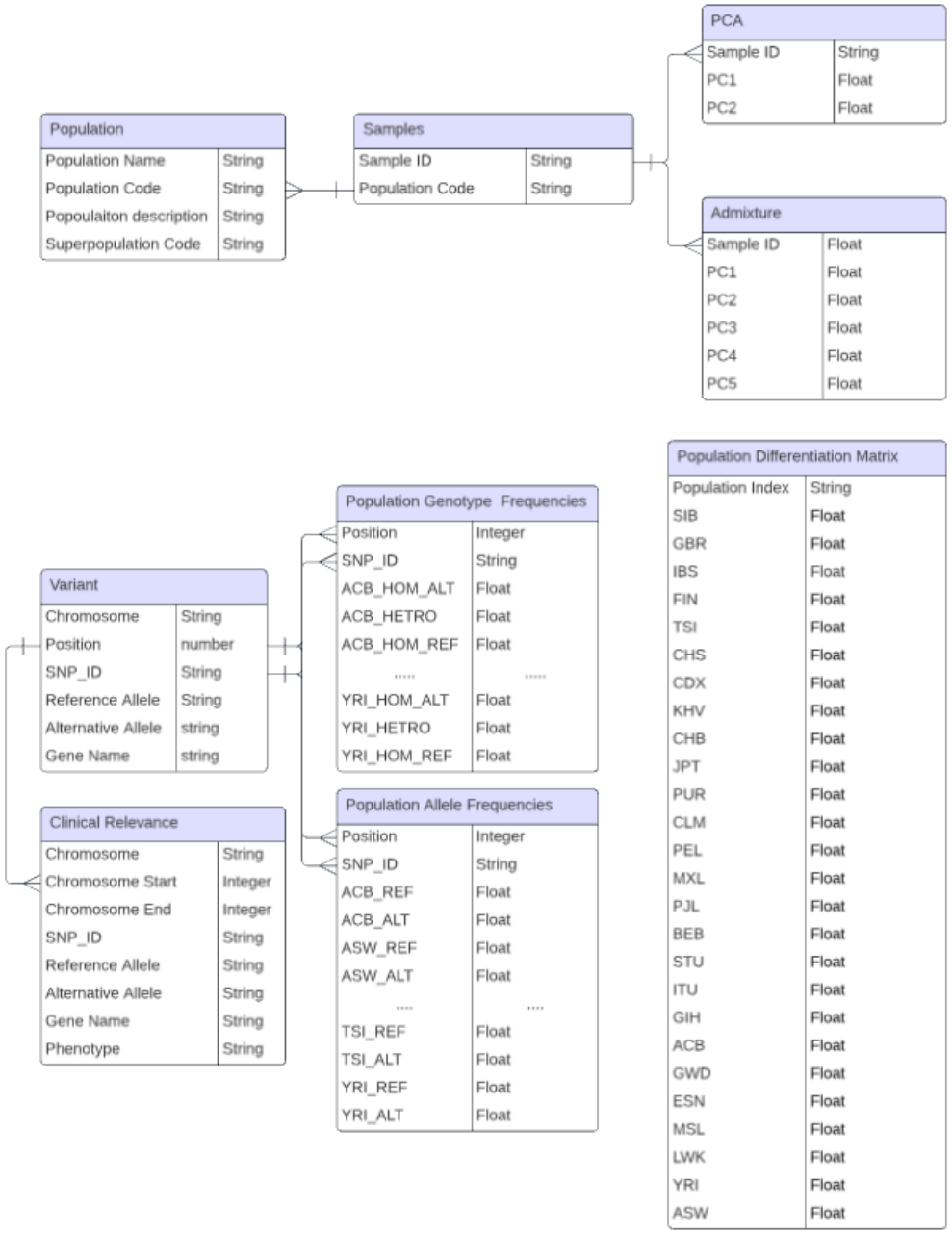
Software Design Pattern

- **Models:**
  - Represent data and its associated logic, serving as the core of our application.
- **Views:**
  - Responsible for presenting data to users in a clear and accessible manner.
- **Controllers:**
  - Act as intermediaries between views and models, orchestrating the flow of data and business logic.
- **Routes:**
  - Define URL endpoints and handle incoming HTTP requests, directing them to corresponding controller actions for processing.
- **Services:**
  - Encapsulate reusable business logic and operations, enhancing code reusability and maintainability.
- **Flask Blueprints:**
  - Utilized for modularizing our application, dividing it into smaller, manageable pieces.
  - Allow registration of multiple components at distinct URL prefixes for efficient routing and resource management.





Database Schema

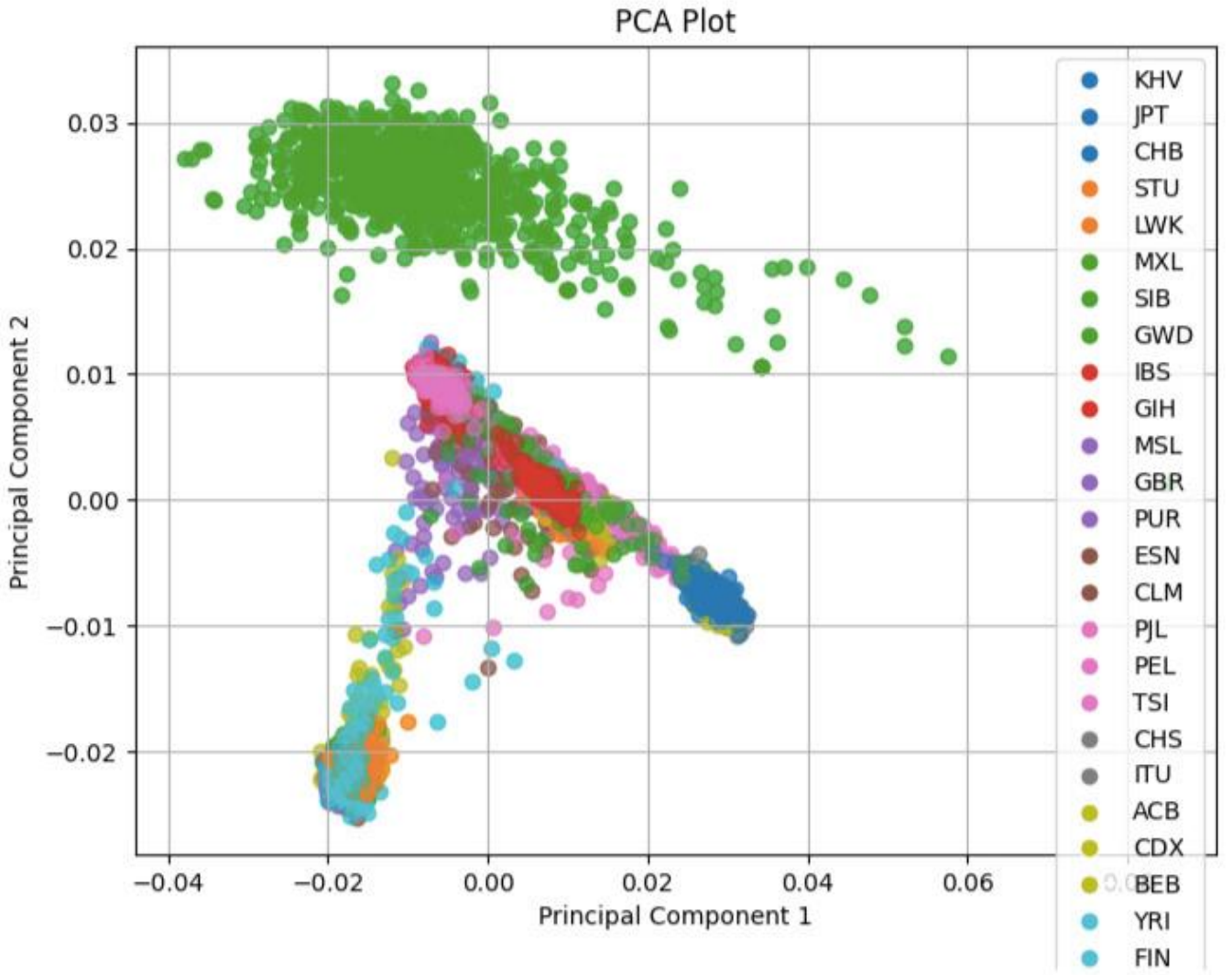


Plink

- Implemented PLINK for data preprocessing and analysis within our application.
- Offers comprehensive functionalities including data quality control, association testing, and PCA.
- Required data format for PCA analysis in PLINK: BED/BIM/FAM.
- Converted VCF file into BED, BIM, and FAM formats using PLINK.
- PLINK's scalability and performance handle large-scale genetic datasets efficiently.

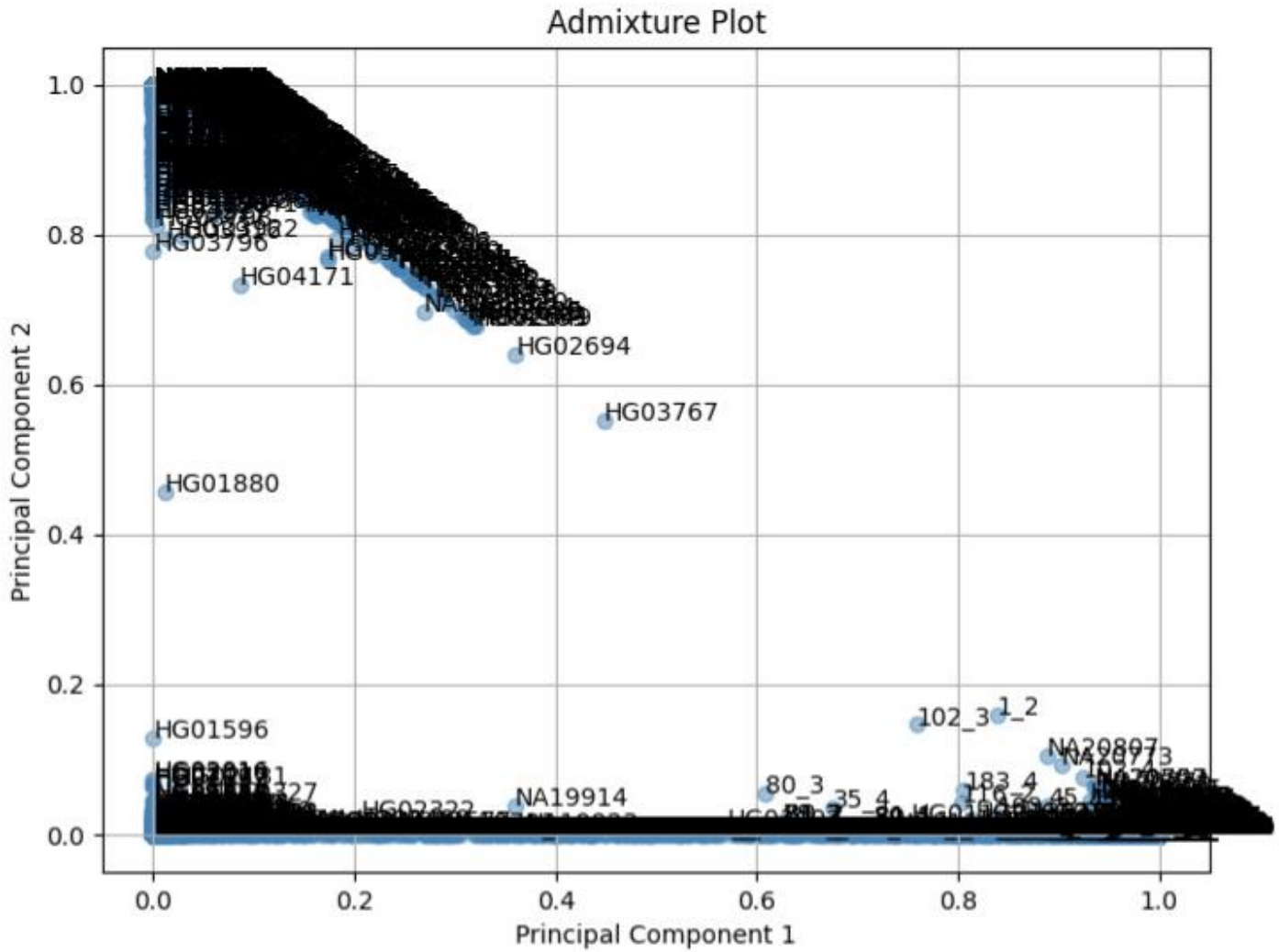
PCA

- PCA is a pivotal technique for dimensionality reduction within our web application's clustering analysis framework.
- PCA transforms original variables into a new set of orthogonal variables called principal components.
- This transformation preserves data variance while simplifying it, aiding in visualization.
- PCA helps unravel underlying patterns and structures within complex datasets.
- Widely used in genetics for understanding relationships and identifying clusters.
- Enhances data visualization and comprehension of complex genetic data.



Admixture

- Admixture analysis in the app: Allows users to analyze population genetic data.
- Key role: Understanding genetic ancestry, population migrations, and diversity.
- Statistical model: Based on maximum likelihood estimation.
- Incorporating cross-validation: Evaluates model performance.
- Optimal K value determination: Utilises cross-validation to minimize error (industry standard).
- Result: Optimal K value determined as 5.
- Admixture analysis outputs: Files with individual ancestry proportions.
- Insights provided: Genetic makeup, historical events, migrations, diversity.
- Integration with PLINK: Streamlines analysis process.



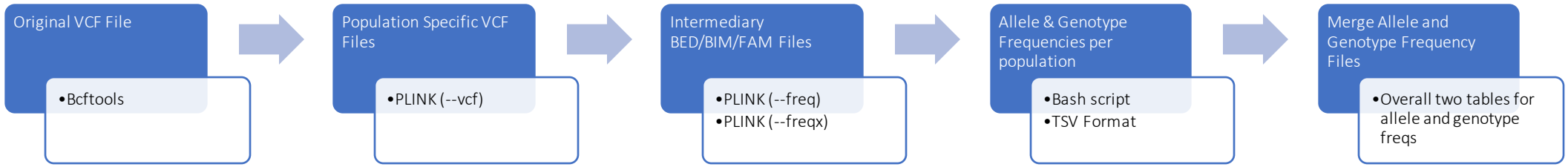
Allele & Genotype Frequencies

Allele Frequency Table

POS	ID	ALT_ACB	REF_ACB	ALT_ASW	REF_ASW
10397	1:10399:C:A	0	1	0	1
10420	1:10420:A:C	0	1	0	1
10437	1:10437:T:C	0	1	0	1
10438	1:10438:A:T	0	1	0	1

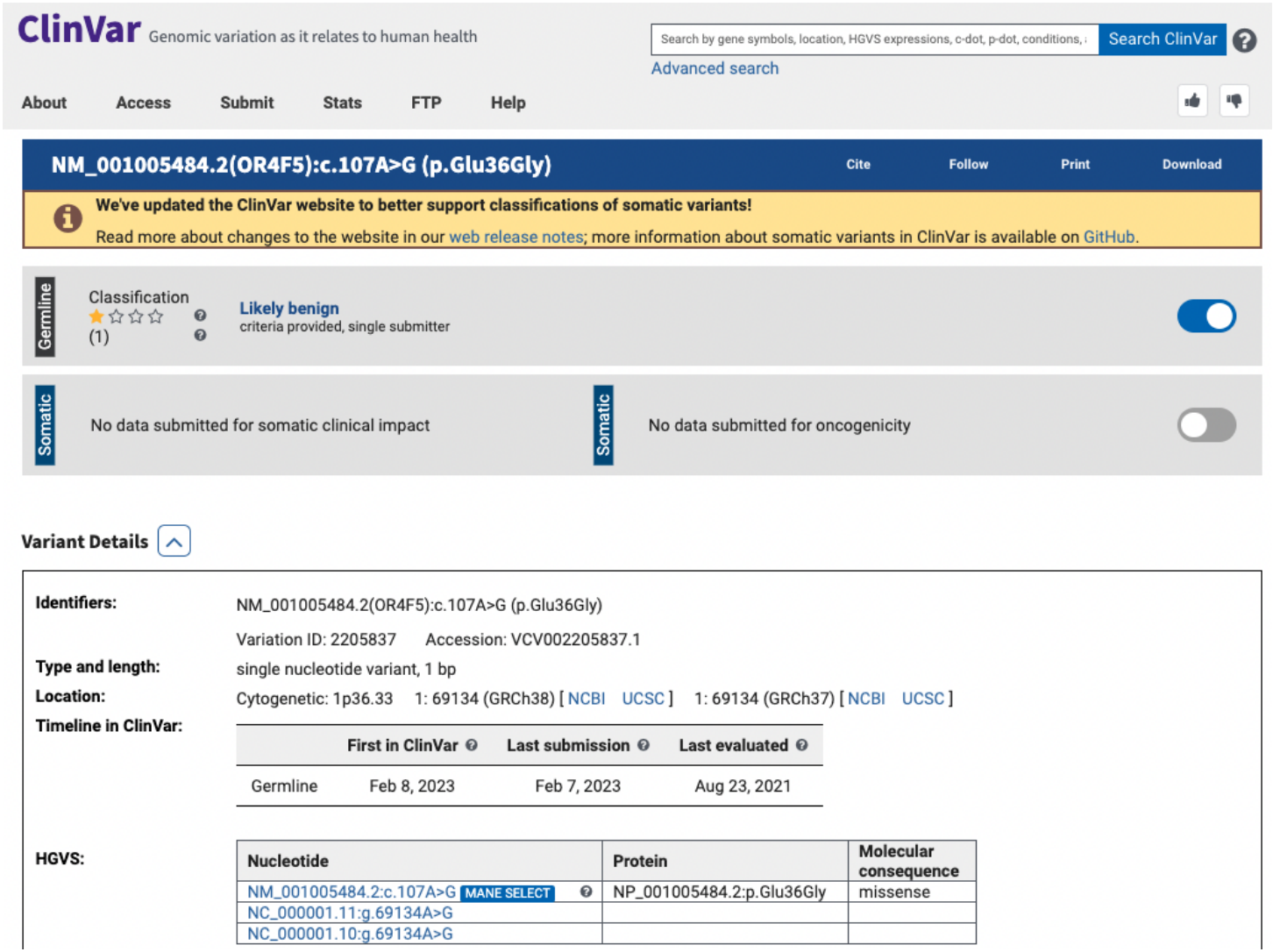
Genotype Frequency Table

POS	ID	HOM_ALT_ACB	HETRO_ACB	HOM_REF_ACB
10397	1:10399:C:A	0	0	116
10420	1:10420:A:C	0	0	116
10437	1:10437:T:C	0	0	116
10438	10438:A:T	0	0	116



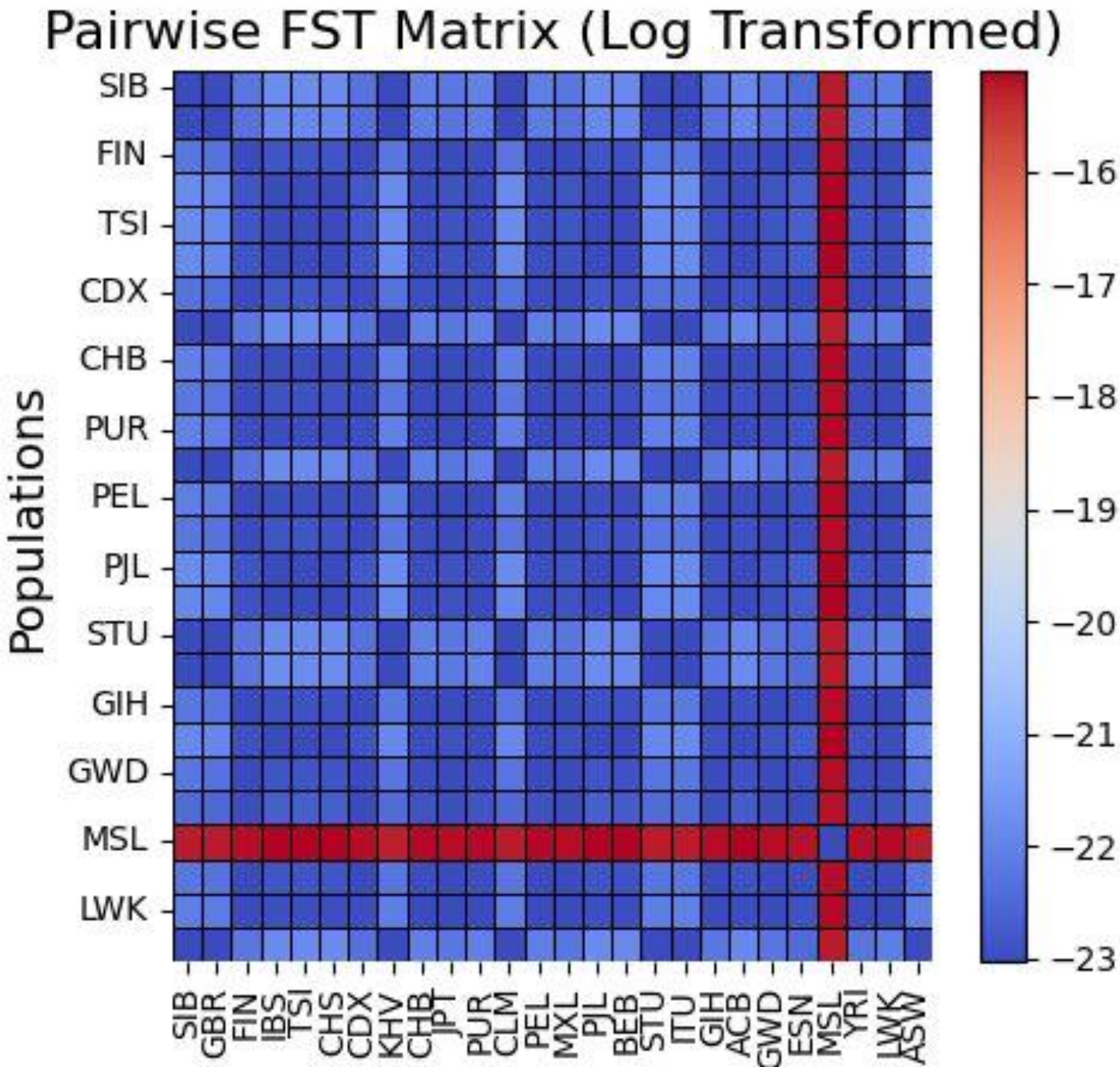
Clinical Relevance

- ClinVar was used to obtain information regarding clinical relevance of SNPs.
- Data was gathered for all genes located on chromosome 1.
- We downloaded the chr1 information from the ClinVar website and integrated it into our database.
- ClinVar was used because chromosome positions can be used to index the database.



Population Pairwise Differentiation Matrix

- Genetic differentiation was determined via fixation indexing (FST).
- The approach used to calculate FST values was Wright's F-statistics.
- FST was calculated for alternative allele frequency for each population.
- The FST values were log-transformed for better visualisation of the data.



Limitations & Future Developments

- FST calculations are not SNP specific hence genetic differentiation data could be lost when comparing populations. Improvements in the app should focus on building in this functionality.
- The user is not able to download a text file of the pairwise population differentiation analysis. Future developments should focus on having a download button to export the results onto the user device.
- In terms of clinical relevance, the user does not receive much information hence there could be a link that redirects the user to the relevant SNP in the ClinVar database.
- We could implement the ClinVar API into our application so that in future whenever any new data is added, it will automatically