

## 4 Design

### 4.1 Design Content

Our design attempts to address the needs of the client by creating two fundamental systems; a frontend that provides the UI/UX to the user as well as display the rendering of the convoy displayed to the user, and a backend that handles data access, computation, and visualization rendering.

### 4.2 Design Complexity

Provide evidence that your project is of sufficient technical complexity. Use the following metric or argue for one of your own. Justify your statements (e.g., list the components/subsystems and describe the applicable scientific, mathematical, or engineering principles)

1. The design consists of multiple components/subsystems that each utilize distinct scientific, mathematical, or engineering principles
  - a. Frontend
    - i. Authentication/Authorization
      1. Users need to view only data assigned to them.
    - ii. UI/UX
      1. Database, algorithm, parameter selection
      2. View Job Status
      3. View Results
    - iii. Visualization of convoys
      1. Interactive
      2. 3D
  - b. Backend
    - i. Data Access/Storage
      1. Upload Datasets
      2. Load Datasets for computation
    - ii. Convoy Computation
      1. Multiple Algorithms
      2. Job Queueing
    - iii. Visualization Rendering
      1. Reduce Size for Output
      2. Reduce Resolution
2. The problem scope contains multiple challenging requirements that match or exceed current solutions or industry standards.
  - a. As we know of, there are no other comprehensive tools for selection, computation, and visualization of molecular convoys.
  - b. We are utilizing newly developed/published convoy detection algorithms.
  - c. Project requirements demand the creation of multiple systems/subsystems with constant interactivity between them

### 4.3 Modern Engineering Tools

What modern engineering tools were used for this design? Their roles.

- React.js <https://react.dev/> [1]
  - o An industry standard frontend JavaScript framework that is widely utilized.
- Flask <https://flask.palletsprojects.com/en/3.0.x/> [2]
  - o A Python web framework that we are going to utilize
- Plotly <https://plotly.com/python/> [3]
  - o A Python graphing framework that we are utilizing to create animated 3D graphs

### 4.4 Design Context

Describe the broader context in which your design problem is situated. What communities are you designing for? What communities are affected by your design? What societal needs does your project address?

Our project is designed to be used as an aid for drug manufacturers to detect how certain substrates and drug molecules may interact over time as well as detect events of interest such as convoys. Because of this, both chemists and non-technical industry professionals need to be able to use and understand our software.

List relevant considerations related to your project in each of the following areas:

Area	Description	Examples
Public health, safety, and welfare	How does your project affect the general well-being of various stakeholder groups? These groups may be direct users or may be indirectly affected (e.g., solution is implemented in their communities)	Aid in providing a comprehensive tool to automate convoy detection and allow chemists to develop potentially lifesaving drugs faster.
Global, cultural, and social	How well does your project reflect the values, practices, and aims of the cultural groups it affects? Groups may include but are not limited to specific communities, nations, professions, workplaces, and ethnic cultures.	Chemists' workload will be lightened by removing the tedious repetitive task of detecting convoys by hand enabling a more effective exchange of findings and processes.
Environmental	What environmental impact might your project have? This can include indirect effects, such as deforestation or unsustainable practices related to materials manufacture or procurement.	The servers that run the convoy detection algorithms will use energy, however, our system will enable more efficient separation of potential reactions among atoms without sacrificing the effectiveness.
Economic	What economic impact might your project have? This can include the financial viability of your product within your team or company, cost to consumers, or broader	Drug research can be extremely costly and have risks involved. Our system will mitigate some of these costs by using simulated data and

	economic effects on communities, markets, nations, and other groups.	detecting convoys with efficient algorithms. This can make for faster drug research as well as quicker time to market.
--	--	--

#### 4.5 Prior Work/Solutions

There are comprehensive systems that provide convoy detection and visualization using a cohesive user interface. There has been prior work done with studying molecular [flocks](#) [4] and [convoys](#) [5]. The downside of these two studies is that they neglect to consider tolerance gaps in convoy detection which is still interesting to drug manufactures. There also exists software that can visualize molecular trajectories but none that work in conjunction with a convoy detection system.

#### 4.6 Design Decisions

##### 1. Backend Framework

- a. Needs to be able to work with the data and algorithms provided by the client.
- b. Needs to be able to scale with multiple users.
- c. Needs authentication support
- d. Needs to be able to communicate with frontend using HTTP
- e. Support for rendering visualizations

Given the criteria we narrowed down our options to two different frameworks:

- Java Spring
  - o Group Familiarity/Experience
  - o Fast
  - o Authentication Support
  - o Testing Support / Dependency Injection
  - o Good Documentation
  - o Bloated Codebase
  - o Not compatible with given datasets
- Python Flask
  - o Older framework
  - o Little group experience
  - o Compatible with given datasets and algorithms
  - o Performant
  - o Lightweight
  - o Can work with many visualization frameworks

After analysis of our two options, we have decided to go with Flask to create our backend. Its compatibility with the given datasets being its biggest draw. It will also give the group an opportunity to learn a new framework and for some of us a new language. Though this will come at the cost of added time as members may have to do more initial research before completing their tasks. Python also has a larger number of graphing/visualization framework available which also lead us to choose Flask.

##### 2. Frontend Framework

- a. Needs to be able to give a good user experience
- b. Interface with our visualization framework to display information

Given the criteria we narrowed down our options to two different frameworks:

- React
  - o Most popular frontend framework
  - o Group Familiarity
  - o Flexible
  - o Larger community
  - o Larger file size
- Vue
  - o Second most popular frontend framework
  - o Smaller file sizes
  - o Better for simplicity

After analysis of our two options, we have decided to go with React to create our frontend. Of all our members, we have the most experience with React. Additionally, it has great community support as it is the most popular frontend framework that developers use.

3. Visualization Framework
  - a. Display convoys through various time periods
  - b. Output multiple convoys
  - c. Highlight hydrogen bonds within convoys
  - d. Output results to our frontend
  - e. Fast and efficient

Given the criteria we narrowed down our options to two different frameworks:

- VMD
  - o External application
  - o Unknown API
  - o Suggested by client
  - o Built for modeling molecules
  - o Built-in scripting
  - o Lacks interactivity
- Plotly
  - o Python library
  - o Interactive
  - o Flexible
  - o May lack complexity and tooling for molecule modeling

After analysis of our two options, we have decided to go with Plotly to create our visualization framework. We concluded that VMD would require too much work to interact with our frontend and it would be better to use a more flexible tool like Plotly over one tool designed for molecule modeling. Plotly being a Python library also allows for simplicity without backend framework being developed in Python as well.

## 4.7 Proposed Design

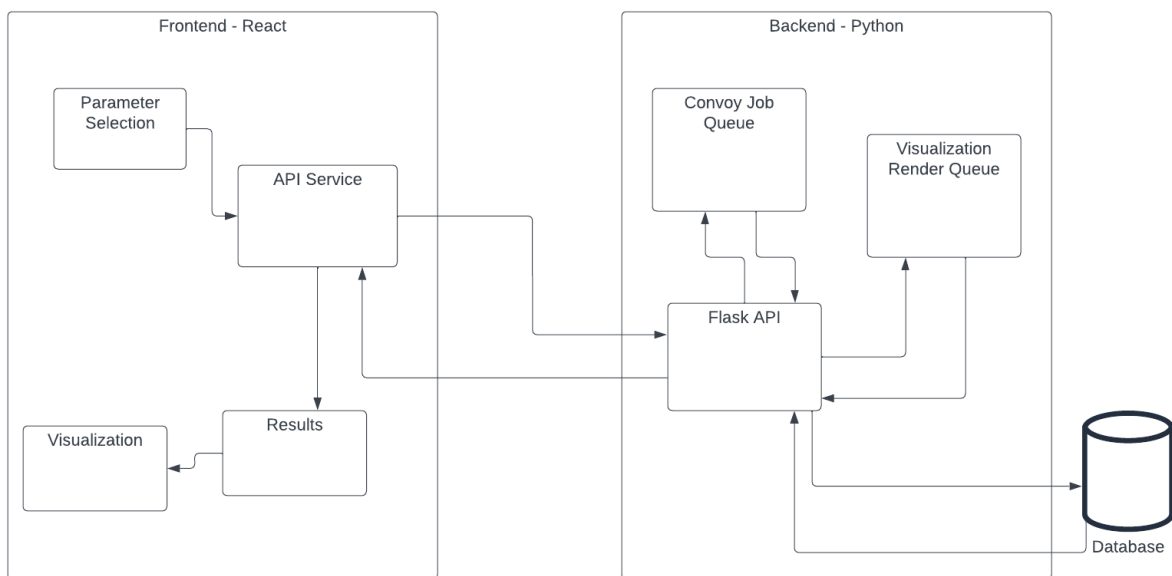
### 4.7.1 Design 0 (Initial Design)

#### Design Visual and Description

Include a visual depiction of your current design. Different visual types may be relevant to several types of projects. You may include: a block diagram of individual components or subsystems and their interconnections, a circuit diagram, a sketch of physical components and their operation, etc.

Describe your current design, referencing the visual. This design description should be in sufficient detail that another team of engineers can look through it and implement it.

Justify each component in the design with respect to requirements.



This project can be broken down into two main parts, the frontend and the backend. The frontend will encompass parameter selection, the API service, the Results, and the Visualization. The backend will contain the Flask API, the Convoy Job Queue, the Visualization Render Queue, and communicate with the database. The process will start on the frontend, where a user is able to upload a dataset for the convoy detection algorithms to run on. This will go through the API Service to communicate with the Flasks API, which will then upload that data to the database. After this is done, the user can use the Parameter Selection component to determine the specifics of how the convoy detection algorithm will work. It will pass that information to the API service, which will communicate with the Flask API. The Flask API will send this information to the Convoy Job Queue. This will execute the algorithm needed to detect convoys and return the results to the API. Next, it will send these results to the Visualization Render Queue. This will create a visualization using plotly to display information about the convoys. Finally, the Flask API will send back this visualization to the API Service, which will send this information to the frontend for the user to view.

Components in relation to requirements:

- Frontend
  - Parameter Selection

- This helps to satisfy the requirement of taking input from the user including the dataset, parameters, and algorithm. It also helps to ensure the parameters are valid to prevent crashing.
  - API Service
    - This is a necessary component because it handles the communication between the data/parameters and the execution of the algorithms.
  - Results
    - This component is needed to process the results being returned from the backend, necessary for generating the visualization on the frontend.
  - Visualization
    - This helps to meet the requirement of displaying a visualization for the user to view on the frontend.
- Backend
  - Flask API
    - This is a necessary part of the communication between the frontend and backend, allowing for the parameters and dataset to be properly used.
  - Convoy Job Queue
    - This component satisfies the requirement of executing the algorithms on a remote server.
  - Visualization Render Queue
    - This component meets the requirement of creating the visualization so it can eventually be outputted on the frontend for the users to view.
  - Database
    - This enables the program to access the data both quickly and reliably.

## Functionality

Describe how your design is intended to operate in its user and/or real-world context. This description can be supplemented by a visual, such as a timeline, storyboard, or sketch.

How well does the current design satisfy functional and non-functional requirements?

This design is intended to operate in the real world by first allowing chemists who need to view the formation of convoys and hydrogen bonds between molecules upload their data to the database. Next, they can select the necessary parameters and algorithm to execute. Both this and the visualization of the results will be done remotely and send back to the frontend for the users to view. This process will meet the requirements as specified above.

### 4.7.2 Design 1 (Design Iteration)

Include another most matured design iteration details. Describe what led to this iteration and what are the major changes that were needed in Design o.

#### Design Visual and Description

Include a visual depiction of this design as well highlighting changes from Design o. Describe these changes in detail. Justify them with respect to requirements.

NOTE: The following sections will be included in your final design document but do not need to be completed for the current assignment. They are included for your reference. If you have ideas for these sections, they can also be discussed with your TA and/or faculty adviser.

#### 4.8 Technology Considerations

Highlight the strengths, weakness, and trade-offs made in technology available.

Discuss possible solutions and design alternatives

#### 4.9 Design Analysis

- Did your proposed design from 4.7 work? Why or why not?
- What are your observations, thoughts, and ideas to modify or iterate further over the design?

#### References

1. <https://react.dev/>
2. <https://flask.palletsprojects.com/en/3.0.x/>
3. <https://plotly.com/python/>
4. <https://dl.acm.org/doi/10.1145/1183471.1183479>
5. <https://dl.acm.org/doi/10.14778/1453856.1453971>