



## CST-323 Design Report Template

<b>Topic:</b>	Milestone 2																																											
<b>Date:</b>	11/26/2024																																											
<b>Revision:</b>	2.0																																											
<b>Team:</b>	1. Kaya Nelson																																											
	2. Tyson martin																																											
	3.																																											
	4.																																											
	5.																																											
	6.																																											
<b>Weekly Team Status Summary:</b>	<table border="1"> <thead> <tr> <th>User Story</th><th>Team Member</th><th>Hours Worked</th><th>Hours Remaining</th></tr> </thead> <tbody> <tr> <td><i>As a developer, I find it beneficial to incorporate sprints, use a burndown chart, and establish a repository setup for managing projects effectively.</i></td><td>Tyson</td><td>1</td><td>0</td></tr> <tr> <td><i>As a developer, I prefer to have foundational tools that assist in building server and web applications.</i></td><td>Kaya</td><td>3</td><td>0</td></tr> <tr> <td><i>As a developer, I aim to implement a database along with server routes that efficiently handle data operations.</i></td><td>Kaya</td><td>5</td><td>0</td></tr> <tr> <td><i>As a developer, I would value having comprehensive and up-to-date documentation for our service API and client application.</i></td><td>Kaya/Tyson</td><td>1</td><td>0</td></tr> <tr> <td><i>As a developer, I want to design a landing page that encourages users to log in or register.</i></td><td>Tyson</td><td>3</td><td>0</td></tr> <tr> <td><i>As a developer, I need to create a page that enables users to log into their accounts.</i></td><td>Kaya</td><td>1</td><td>0</td></tr> <tr> <td><i>As a developer, I would like a high-level video presentation that highlights the code, web design, and features to illustrate our current progress.</i></td><td>Tyson</td><td>1</td><td>0</td></tr> <tr> <td></td><td></td><td></td><td></td></tr> <tr> <td></td><td></td><td></td><td></td></tr> </tbody> </table>				User Story	Team Member	Hours Worked	Hours Remaining	<i>As a developer, I find it beneficial to incorporate sprints, use a burndown chart, and establish a repository setup for managing projects effectively.</i>	Tyson	1	0	<i>As a developer, I prefer to have foundational tools that assist in building server and web applications.</i>	Kaya	3	0	<i>As a developer, I aim to implement a database along with server routes that efficiently handle data operations.</i>	Kaya	5	0	<i>As a developer, I would value having comprehensive and up-to-date documentation for our service API and client application.</i>	Kaya/Tyson	1	0	<i>As a developer, I want to design a landing page that encourages users to log in or register.</i>	Tyson	3	0	<i>As a developer, I need to create a page that enables users to log into their accounts.</i>	Kaya	1	0	<i>As a developer, I would like a high-level video presentation that highlights the code, web design, and features to illustrate our current progress.</i>	Tyson	1	0								
User Story	Team Member	Hours Worked	Hours Remaining																																									
<i>As a developer, I find it beneficial to incorporate sprints, use a burndown chart, and establish a repository setup for managing projects effectively.</i>	Tyson	1	0																																									
<i>As a developer, I prefer to have foundational tools that assist in building server and web applications.</i>	Kaya	3	0																																									
<i>As a developer, I aim to implement a database along with server routes that efficiently handle data operations.</i>	Kaya	5	0																																									
<i>As a developer, I would value having comprehensive and up-to-date documentation for our service API and client application.</i>	Kaya/Tyson	1	0																																									
<i>As a developer, I want to design a landing page that encourages users to log in or register.</i>	Tyson	3	0																																									
<i>As a developer, I need to create a page that enables users to log into their accounts.</i>	Kaya	1	0																																									
<i>As a developer, I would like a high-level video presentation that highlights the code, web design, and features to illustrate our current progress.</i>	Tyson	1	0																																									



GRAND CANYON  
UNIVERSITY™

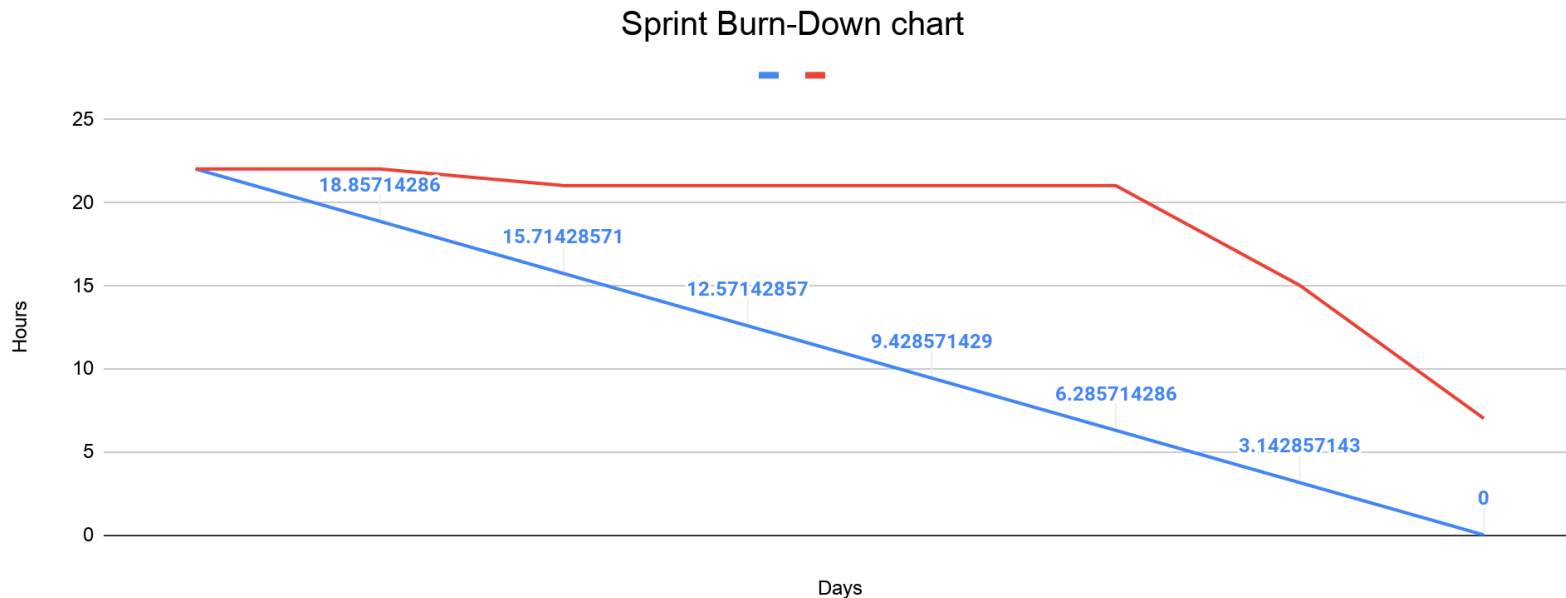
<b>GIT URL:</b>	<a href="https://github.com/Kdeshun/CST323-Milestone">https://github.com/Kdeshun/CST323-Milestone</a>			
<b>Peer Review:</b>	<a href="#">Y/N</a>	We acknowledge that our team has reviewed this report and we agree to the approach we are all taking.		



## Planning Documentation

### Agile Scrum Burn Down Chart:

*This needs to contain a URL to Bitbucket Scrum Burn Down Chart Artifact.*



### Agile Retrospective Results:

*The following table should be completed after each Retrospective on Things That Went Well (Keep Doing). An alternative to the following table is to use a Mind Mapping tool such as Coggle. If you use a Mind Mapping tool, you must include a URL or Image File.*

What Went Well
Having a centralized method of communication helped communicate team standards to the group on how to complete and submit work.
The initial setup with react/ node/ npm/ and base project file worked flawlessly



*The following table should be completed after each Retrospective on Things That Didn't Go Well (Stop Doing) and What Would Be Done Differently Next Time with an Action Plan to Improve (Try Doing and Continuous Improvement). An alternative to the following table is to use a Mind Mapping tool such as Coggle. If you use a Mind Mapping tool, you must include a URL or Image File.*

What Did Not Go Well	Action Plan	Due Date
<b>Integration Challenges:</b> There were issues integrating with third-party APIs, leading to delays in functionality.	Set up a testing environment to test integrations before deployment.	12/1
<b>Documentation Shortcomings:</b> The documentation was not comprehensive, making it hard for new developers to onboard.	Create onboarding guides for new developers to facilitate easier integration into the team.	12/1
<b>Performance Bottlenecks:</b> The application experienced slow response times, particularly during peak usage.	Conduct a performance review to identify bottlenecks.	12/1

## Design Documentation

### Install Instructions:

*Include step-by-step instructions for setting up your database, configuring, and deploying/installing your application. This section should also include detailed instructions for what configuration files are required by your application, what configuration settings need to be adjusted for various runtime (development or production) environments, and where the files need to be deployed to. This section should also contain detailed instructions for how to clone your application source code from BitBucket and deploy the application to an externally hosted site.*

To get started, the first step is to clone the repository onto your local machine. Once that's done, navigate to the `/app` folder. Open your terminal and type `npm install` to set up the necessary packages. Next, switch to the `/client` directory and run `npm install` (or simply `npm i`) once more.

At this stage, all dependencies should be properly installed and linked. Now, to fire up the development server, head back to the `/app` directory and execute `npm run start`. This command will launch the development server via `server.js` and kick off the React script in the `/client` folder.

When you're ready to prepare the application for deployment, switch over to the `/client` directory again. Run `npm run build` and sit back while the build process completes. If you're deploying the app as a Node.js application on Heroku or another platform, that service will handle the installation of dependencies at the `/app` root level and can start the application server with `npm run start`.

### General Technical Approach:

*In your own words describe your approach and design here. You should also summarize any meeting notes, brain storming sessions, and so forth that you want to retain thru the design of your project.*

Our team's vision is to craft a blogging platform that thrives on the open web. To foster collaboration, we'll harness a diverse set of tools tailored to our workflow:

- **Discord** will be our go-to hub for communication, keeping us connected and engaged as we tackle projects throughout the week.
- **GitHub** will act as our code repository, enabling efficient source code management and version control to streamline our development process.
- For tracking progress, we'll utilize **Google Docs** to create and update our sprint and burndown charts, along with design reports.
- To bring our ideas to life visually, we'll turn to **Draw.io** for wireframing and UML diagrams.
- Finally, our ER diagrams will be crafted with **MySQL Workbench**, ensuring our database design is robust and well-structured.

## Key Technical Design Decisions:

*Any final technical design decisions, (e.g., framework decisions) should be documented here. List the technology/framework, its purpose in the design, and why it was chosen.*

Our team is set to create an intuitive blogging platform. For seamless communication, we'll harness the power of Discord. Documentation will be organized and accessible through Google Docs, while GitHub will serve as our version control hub. We'll primarily use Visual Studio Code as our code editor of choice, though any alternative editor can be easily swapped in. For video presentations, Loom will be our tool of preference.

## Technology Stack:

Our application will be built using a pure JavaScript ecosystem, featuring:

- **React:** A dynamic framework for frontend development.
- **NodeJS:** A robust server-side framework for backend development.
- **MySQL:** A reliable database engine favored by major players like Uber.

## Libraries and Packages:

To maximize the potential of our React/NodeJS setup, we'll integrate a variety of libraries and packages to enhance our application's functionality. All of these resources can be easily accessed via CDN or NPM:

- [MySQL](#)
- [EditorJS](#)
- [Axios](#)
- [UUID](#)
- [React Bootstrap](#)

## Known Issues:

*Any anomalies or known issues in the code or functionality should be documented here.*



**GRAND CANYON**  
UNIVERSITY™

One of the most pressing challenges we face is the underwhelming frontend user interface. Currently, the frontend pages are quite sparse and haven't integrated with the backend service-level API. This limitation stems from restricted team engagement and collaboration leading up to this point, as reflected in the burn-down chart and the breakdown of user stories.

In our upcoming sprint, our top priority will be to enhance the UI, ensuring it is complete and fully operational with the database before we tackle the next set of milestone requirements.

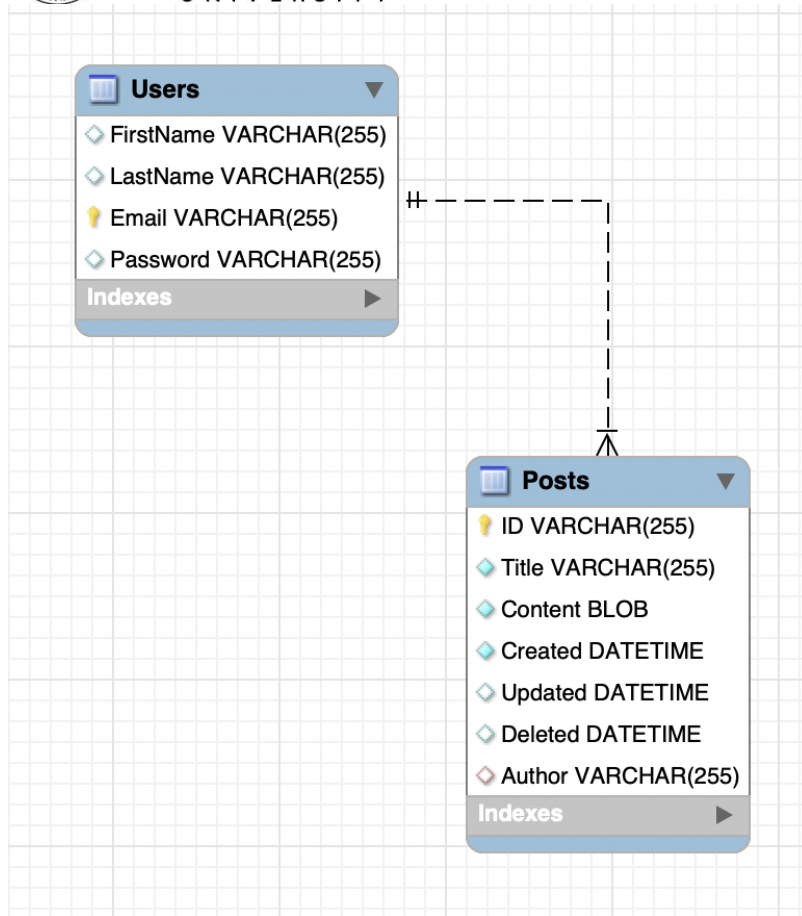
### **Risks:**

*Any risks, unknowns, or general project elements that should be tracked for risk management should be documented here.*

No known Risks at this time.

### **ER Diagram:**

Below is an image of the current database ER diagram.



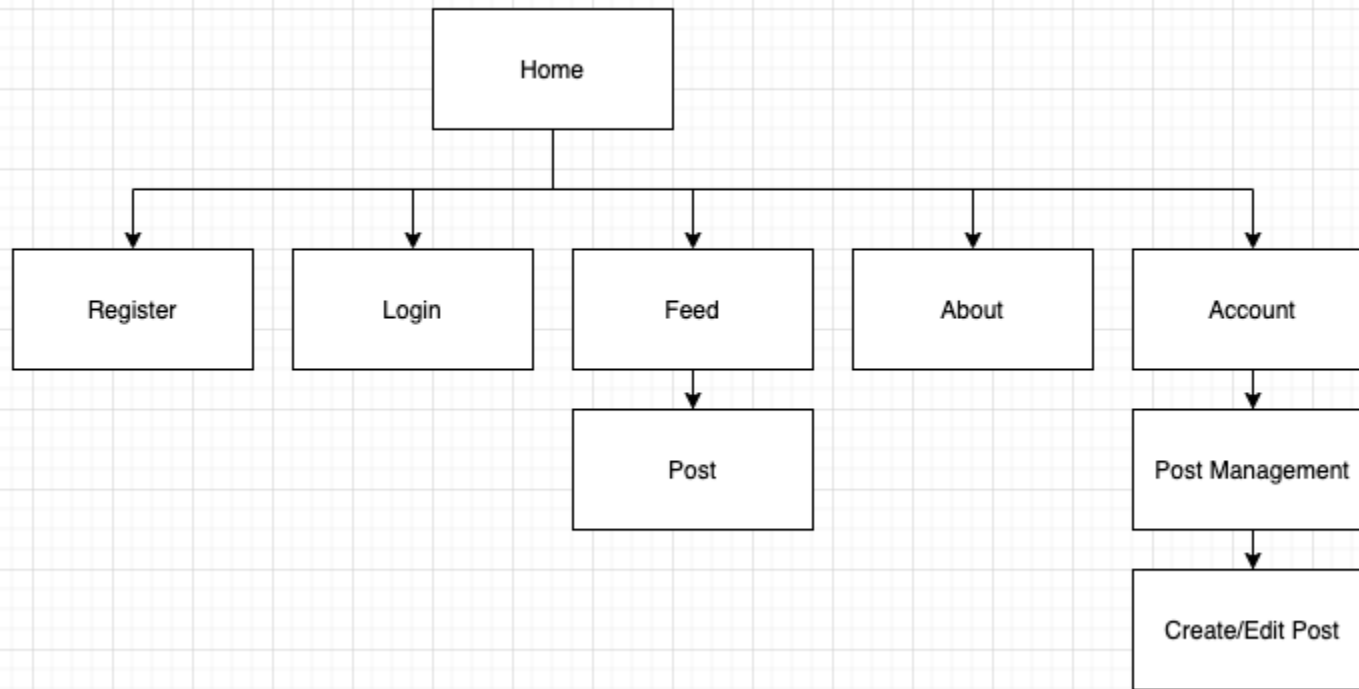
### DDL Scripts:

The current DDL Create script can be found at the following link <https://github.com/Kdeshun/CST323-Milestone>

### Sitemap Diagram:

*Include an image file of your Sitemap diagram.*







GRAND CANYON  
UNIVERSITY™

**User Interface Diagrams:**

*You should insert any wireframe drawings or white board concepts that were developed to support your application. If you have no supporting documentation, please explain the rationale for labeling this section N/A.*



BACK

## EDIT POST

Title

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

DELETE POST

SAVE CHANGES



BACK

## CREATE ACCOUNT

FIRST NAME

John

EMAIL

Smith

EMAIL

eg. name@gmail.com

PASSWORD

\*\*\*\*\*

REGISTER



BACK

## TITLE OF POST

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

ALL POSTS



BACK

## POST MANAGEMENT

WRITE NEW POST

EDIT A POST

POST TITLE

POST TITLE

POST TITLE

POST TITLE



BACK

## MY ACCOUNT

WRITE OR EDIT POSTS

FIRST NAME

John

LAST NAME

Smith

EMAIL

eg. name@gmail.com

PASSWORD

\*\*\*\*\*

SAVE CHANGES



BACK

## LOGIN

EMAIL

eg. name@gmail.com

PASSWORD

\*\*\*\*\*

LOG IN

Don't have an account?

REGISTER





GRAND CANYON  
UNIVERSITY™

THE "BLOGS ARE COOL" BLOG

[HOME](#)

[ABOUT](#)

[FEED](#)

[ACCOUNT](#)

[LOGIN](#)

# Read interesting things, even if you don't want to.

[CREATE ACCOUNT](#)

[LOGIN](#)



BACK

## ALL ARTICLES

TITLE OF ARTICLE

TITLE OF ARTICLE

TITLE OF ARTICLE

TITLE OF ARTICLE

### Class Diagrams:

*You should insert any class diagrams here. Your class diagrams should be drawn correctly with the three appropriate class compartments, + and – minus to indicate accessibility, and the data types for the state/properties as well as method arguments and return types. If you have no supporting documentation, please explain the rationale for labeling this section N/A.*

## UML Diagram:

No UML diagram at this time, while we sort out how we will organize and structure our code layers between the front/back ends.

## Service API Design:

*This section should fully document any Third Party Service Interface API's being consumed or application specific Service API's being published, how to access the service, what parameters are required by the API, and the detailed JSON data format specification that could be used by a third party developer to integrate with the service and API.*

At this stage, we don't anticipate the need to integrate any third-party APIs or external services.

## Service API Overview

We will implement an API layer that bridges the web client and our database, accessible at `/api` on port 3001. This layer will facilitate RESTful interactions with both Users and Posts records.

### User Endpoints:

- **GET /user/:email** - Retrieve user details by email.
- **GET /users** - Fetch a list of all users.
- **POST /user/create** - Create a new user.
- **POST /user/update** - Update existing user information.

### Post Endpoints:

- **GET /post/:id** - Get post details by ID.
- **GET /posts** - List all posts.
- **POST /post/create** - Add a new post.
- **POST /post/update** - Modify an existing post.



**GRAND CANYON**  
UNIVERSITY™

Further documentation detailing the expected parameters for POST requests and the structure of return objects for each endpoint will be added to this documentation shortly.

### **Security Design:**

*This section should outline the design for how authentication and authorization was supported. This section should also contain all of the roles and privileges that are supported by the design.*

Users will access their accounts using their email addresses along with a distinctive password. To ensure security, we'll apply salting and hashing techniques to the passwords before saving them. When a user attempts to log in, we'll salt and hash the password they provide, then compare this processed string to the stored hash. This method guarantees that their original password is never stored in plain text within our database.

### **Other Documentation:**

*You should insert any additional drawings, storyboards, white board pictures, project schedules, tasks lists, and so forth that support your approach, design, and project. If you have no supporting documentation, please explain the rationale for labeling this section N/A.*