

## TRNDii Release Summary

### Team members

Name and Student id	Git Hub id	Number of story points that member was an <b>author</b> on.
<b>Rameen Rastan-Vadiveloo</b> <b>27191863</b>	rameenrastan	18.5
Jacqueline Luo 26938949	LordRoost	8
Michael Mescheder 27202202	michaelgoldfish	16.5
Sam Alexander Moosavi 27185731	sammoosavi	12
Eric Payette 27008058	mosquitodawg	12
Jason Tsalikis 25892120	jason10129	12

### Project summary

TRNDii is a group buying website focused on innovative new products. TRNDii has 'ii tokens' which users can gain and use. When they want to purchase something, unlike regular online shopping, users will pay to commit to buying a product. After paying, users may spend any amount of ii tokens. Once a predetermined number of committed buyers is reached, the product will then be bought from suppliers at a bulk, discounted price. The savings generated from the bulk discount will be redistributed to the buyers based on how many ii tokens they spent and how many ii tokens were spent in total. Anyone who chooses to not spend ii tokens have a chance at winning a free product.

### Velocity

#### Total:

[Iteration 1](#) (1 story, 8 points)

[Iteration 2](#) (6 stories, 16 points)

[Iteration 3](#) (7 stories, 16.5 points)

[Release 1](#) (4 stories, 16 points)

#### Plan for next release:

[Iteration 5](#) (4 story, 10.5 points)

[Iteration 6](#) (3 stories, 19 points)

## Overall Arch and Class diagram

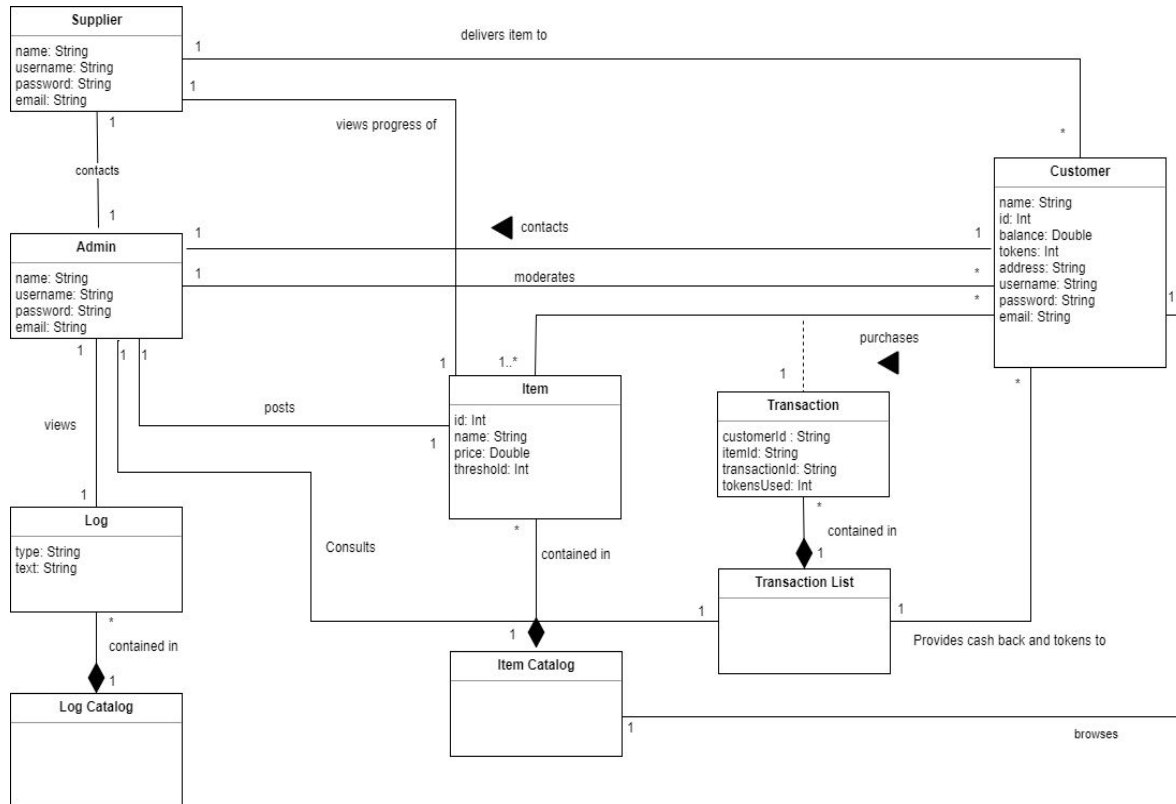


Diagram 1: Domain Model

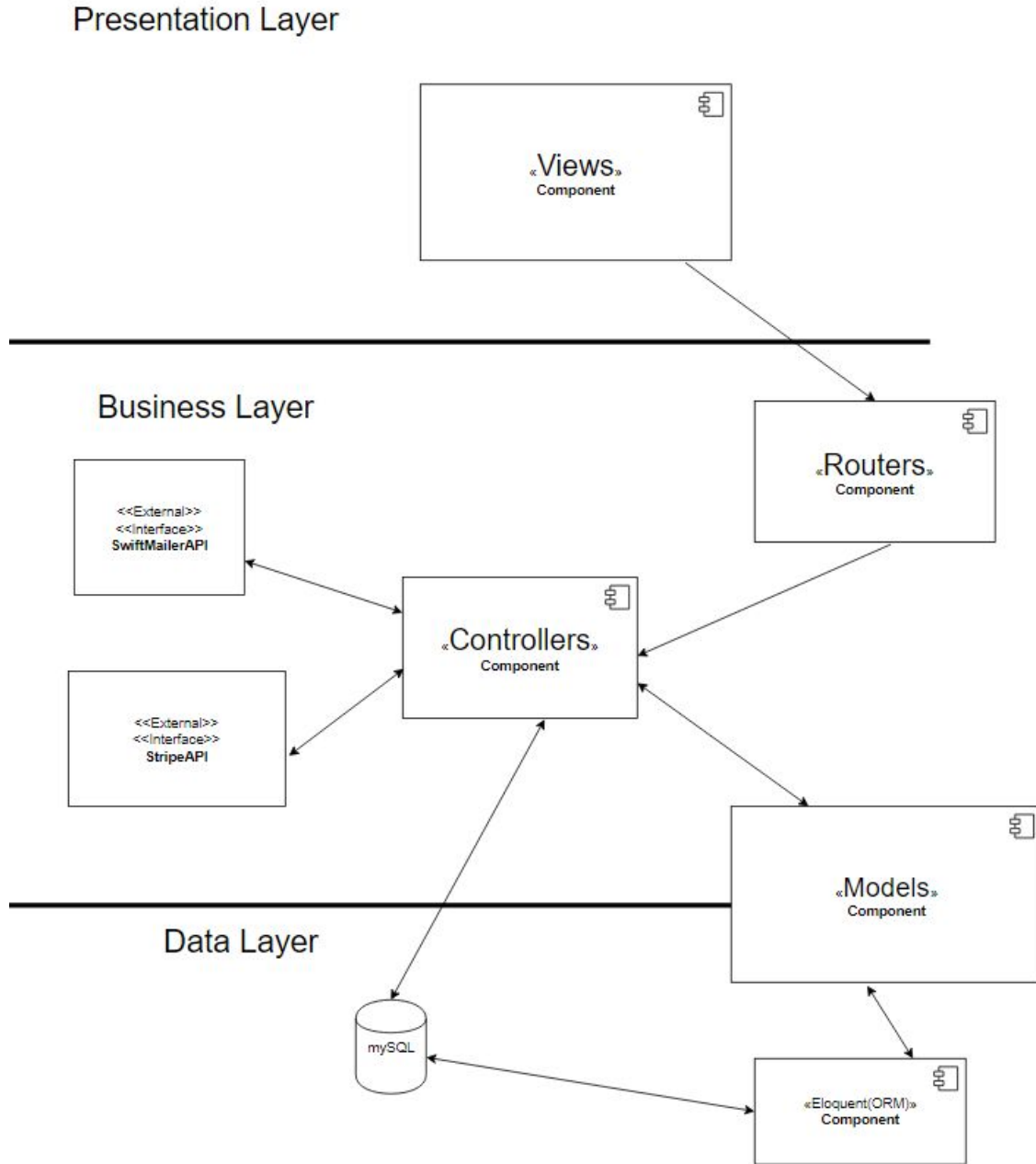


Diagram 2: Layers of the System

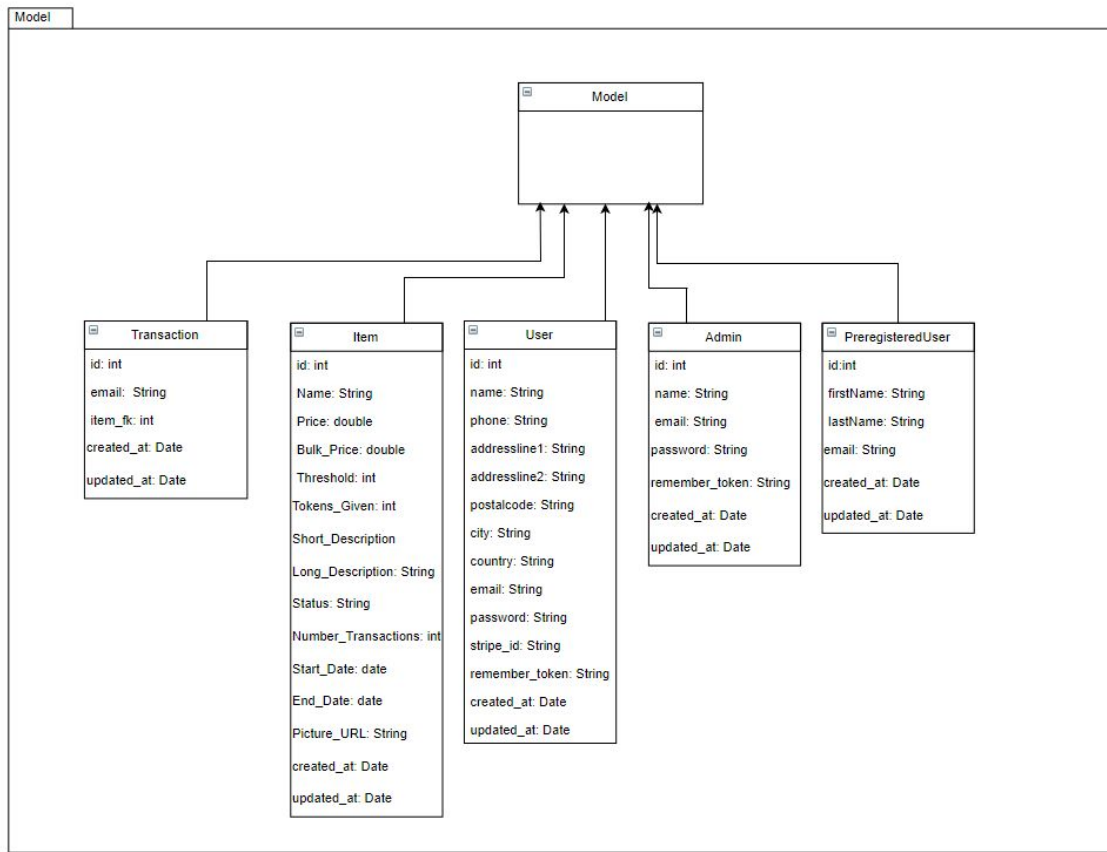


Diagram 3: Model Package

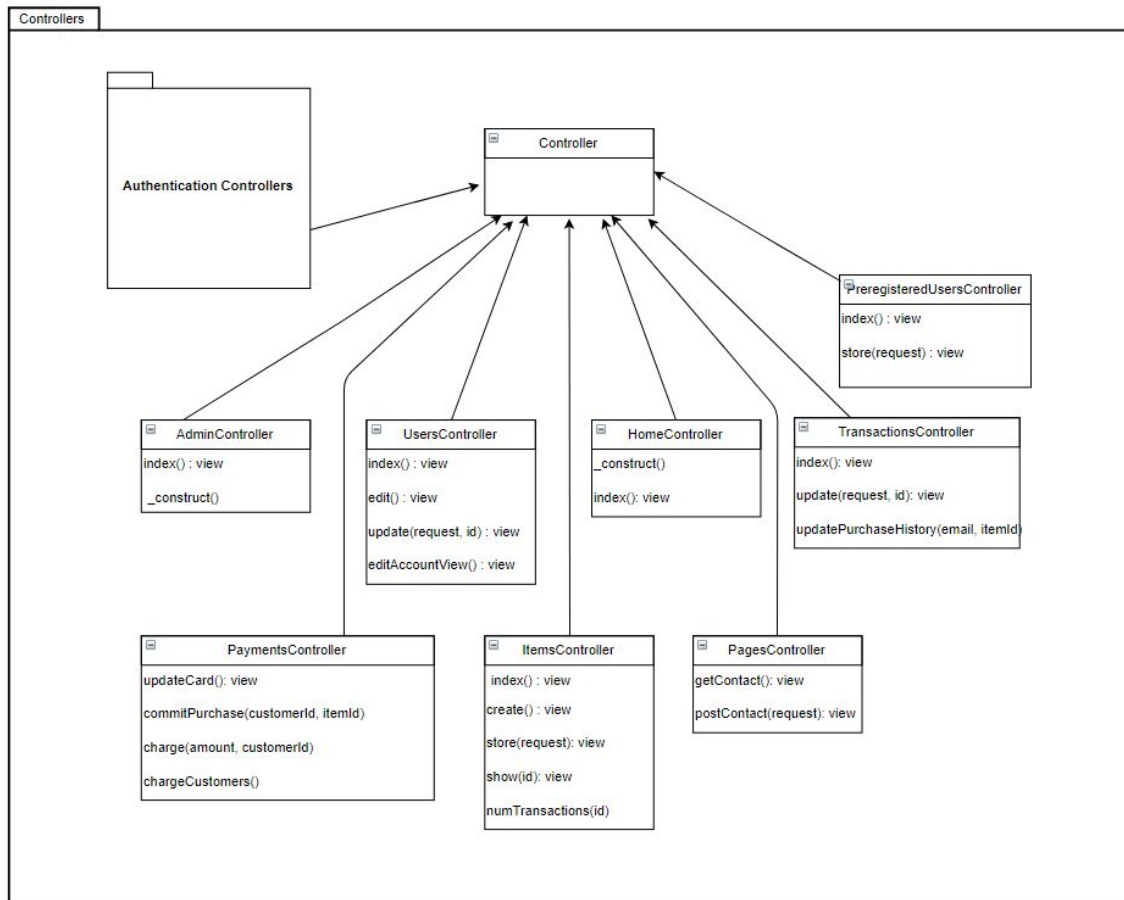


Diagram 4: Controller Package

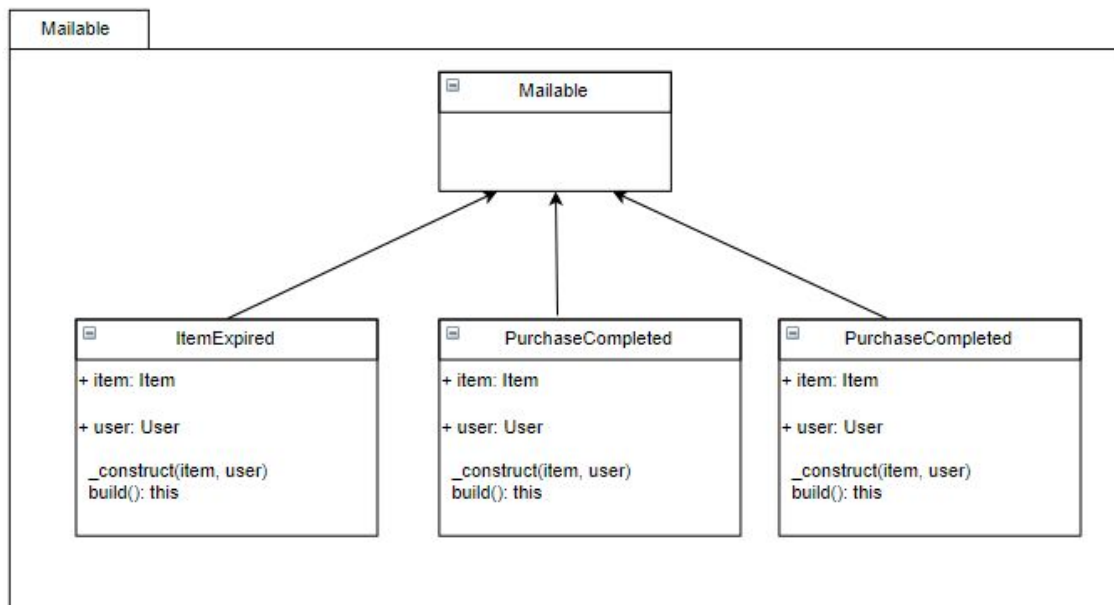


Diagram 5: Mailable Package

## Infrastructure

### Bootstrap

Bootstrap is one of our front-end design frameworks of choice as it comes in a package with Laravel. Additionally, it is the most popular CSS front-end framework, meaning there are lots of resources and documentation on it. It is responsive which is excellent as our stakeholder emphasized that the website should work well on desktop as well as on phones. Bootstrap also supports all major browsers.

Other frameworks include Foundation and Semantic UI. Foundation is not too different from Bootstrap. It also works on a grid system and is very responsive, but the community around it is smaller and we do not really need to use its extra functionalities. Semantic UI is also similar to the previous two frameworks, but its tags are supposed to be more semantic, making it easier for programmers to code. However, it is large and apparently buggy.

### Laravel

We have decided to use Laravel because it is the most popular framework for PHP, our back-end language of choice, and it strongly enforces the MVC pattern. Laravel has extensive documentation and makes it simple to implement many important features such as authentication or logging. It is also easy to setup and start coding with. Additionally, Laravel is well suited for testing as there is built-in support for testing with PHPUnit.

Other php back-end libraries are Symfony and CodeIgniter. Symfony is known to be a mature, stable framework that can make the development of web apps very secure and maintainable. However, it is known for having a steep learning curve and we do not want to be spending most of our time figuring out a complex framework. CodeIgniter is a lightweight and easy to learn framework. However, it is now slightly outdated and is no longer officially supported.

### PHPUnit

PHPUnit is our unit testing framework of choice. It is already packaged and integrated with our framework (Laravel), and is the most popular unit testing framework for PHP, which means it has a lot of information and resources that can be consulted.

An alternative to PHPUnit is atoum; it essentially serves the same functionality as phpunit however given it is not the defacto choice of the community, there are less references and documentation online for it, hence a major factor in going with phpunit.

### Dusk

We are using Laravel Dusk to perform browser tests. This testing framework uses ChromeDriver in order to perform tests therefore we don't need to install additional plugins or softwares. A large quantity of documentation is available online to help us build our browser tests. The learning curve for Dusk is very low due to the simplicity in the algorithms of the tests.

Another popular automated web testing tool is Selenium. While both of these tools utilize ChromeDriver, the fact that Dusk is integrated in Laravel and runs through phpunit makes it much more simple to setup and quick to use, and reduces the need for other installations.

### MySQL

We are using MySQL as our database management system. This is because it is very well supported and it has a lot of GUI managing tools. It is also easy to learn and is very fast. It is also what most of our team has experience with, meaning we do not have to learn a new tool.

Other database management systems are Oracle Database, SQLite and PostgreSQL. Oracle Database needs to be paid for, which is unsuitable for our team. SQLite is an embedded database which has no networking capabilities. It also has issues with concurrency, making it inappropriate for our project. PostgreSQL is a completely open source alternative to MySQL, but it is less popular and thus difficult to get support or to google as many questions for it. PostgreSQL can also run slower for read-heavy operations.

### Vue.js

Vue.js is our front-end library of choice. It integrates well with Laravel; it is provided in a package while downloading Laravel. Vue can also be easily integrated with other front-end libraries, making it very versatile. Other benefits of Vue are the facts that it is quick and lightweight, ideal for our website as we do not want users to wait for pages to load. Vue is also simple to learn, which is ideal.

Other popular frameworks for the front-end are Angular, React and Ember. These all have a steep or steeper learning curve than Vue. We would have to learn JSX for React and Typescript for Angular. Ember has issues processing quick changes, which can be for the user experience. Additionally, it is big and heavy.

### Stripe

Stripe is our payment service of choice. It allows our website to accept payments without credit card information reaching our server, which alleviates many security concerns associated with a website that accepts payments. It provides a very simple dashboard and API to handle transactions. Stripe also provides excellent documentation which allows the learning curve to be much less of a hassle.

Alternatives are Braintree and Paypal. The main reason we decided not to go with Paypal is that it limits the form of payment, as users will not be able to use credit cards. However, we may implement Paypal later on as an alternative form of payment. Our research indicated that Braintree is a bit more of a hassle to implement as the documentation and API is not as clear, and also does not provide features such as discounts.

## **Other Libraries**

The main reason we've selected the following libraries to handle things such as task scheduling, data persistence, dependency injections and logging is due to the fact that they are integrated with Laravel, and Laravel has considerable documentation on how to utilize the following libraries. This will make implementation much more simple, and will avoid significant overhead due to potential compatibility issues:

### **Laravel Command Scheduler**

Laravel's integrated command scheduler will be used to schedule tasks that need to be scheduled on the server.

### **HTTP Session**

Laravel's HTTP Session will be used for data persistence.

### **Service Container**

Laravel's service container will handle class dependency management and dependency injections.

### **Cache**

The cache.php file provided by Laravel will be used as our caching mechanism in order to cache our backend as needed.

### **Monolog**

The Monolog library will be used for logging. We've decided to use this library as it is integrated with Laravel.

### **SwiftMailer**

The SwiftMailer library, integrated with Laravel's mailable classes, will be used to develop and send email messages through our website. Our email testing environment will be through mailtrap.io, which allows us to send fake emails and validate them through a development inbox.



## Name Conventions

For the backend:

[PSR-1 - PHP basic coding standards](#)

[PSR-2](#)

[PSR-4](#)

For the frontend:

[Javascript conventions](#)

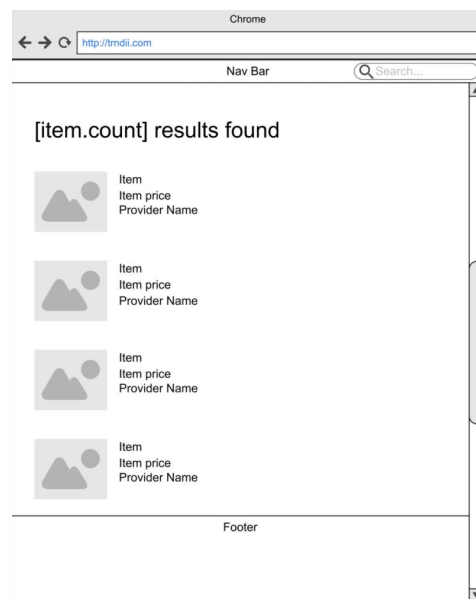
[HTML conventions](#)

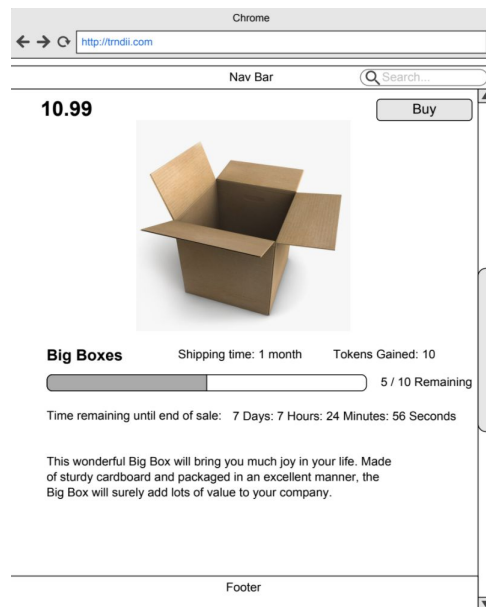
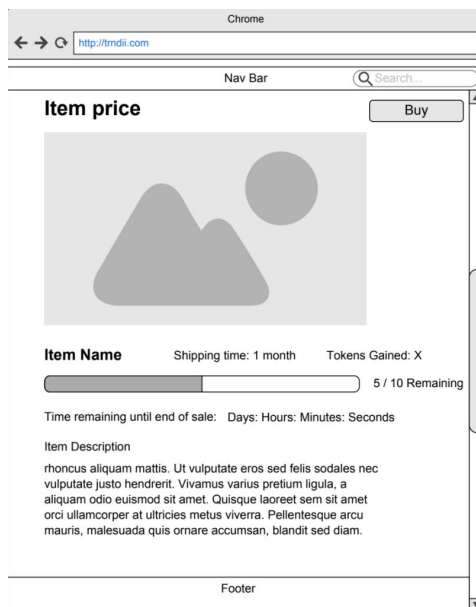
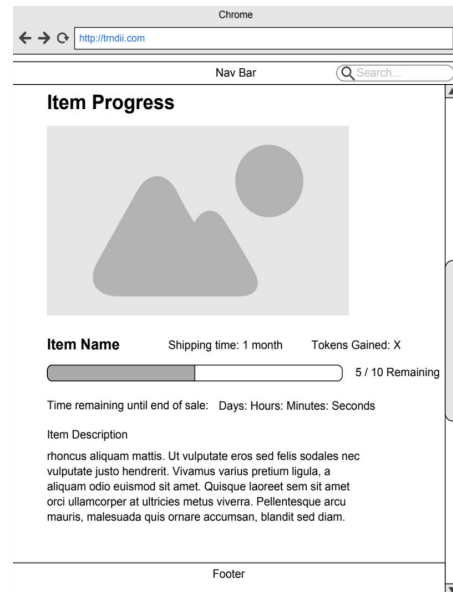
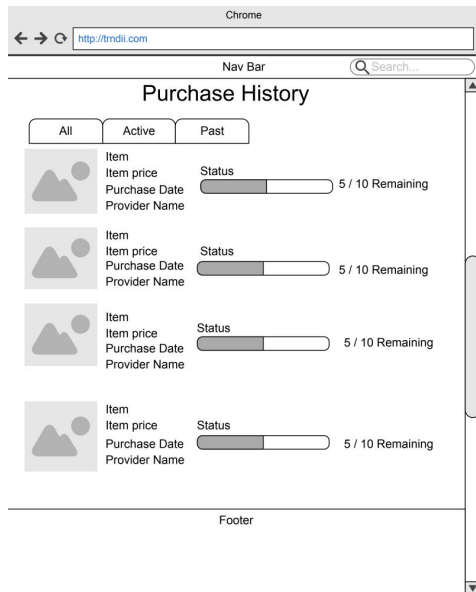
[CSS conventions](#)

For the server:

[SQL naming conventions](#)

## Mockups





## Code

File path with clickable GitHub link	Purpose (1 line description)
<a href="#">/trndiiapp/app/Http/Controllers/PaymentsController.php</a>	Handles inputs and business logic relating to payments.
<a href="#">/trndiiapp/app/Http/Controllers/TransactionsController.php</a>	Handles inputs and business logic relating to transactions.
<a href="#">/trndiiapp/app/Http/Controllers/ItemsController.php</a>	Handles inputs and business logic relating to items.
<a href="#">/trndiiapp/app/Http/Controllers/UsersController.php</a>	Handles inputs and business logic relating to user information.
<a href="#">/trndiiapp/resources/views/layouts/purchasehistory.blade.php</a>	View displaying users orders (active and past transactions)

## Testing and Continuous Integration

Test File path with clickable GitHub link	What is it testing (1 line description)
<a href="#">/trndiiapp/tests/Unit/PurchaseTest.php</a>	Tests purchasing an item.
<a href="#">/trndiiapp/tests/Unit/PurchaseHistoryTest.php</a>	Tests viewing purchase history.
<a href="#">/trndiiapp/tests/Browser/ViewItemTest.php</a>	Tests viewing an item.
<a href="#">/trndiiapp/tests/Browser/EditAccountTest.php</a>	Tests editing account information.
<a href="#">/trndiiapp/tests/Browser/AdminAddItemCompleteTest.php</a>	Tests an admin creating a new item.

## Appendix

### Analogical Case Study

Our project TRNDii is an e-commerce site that operates in a group buying manner with a focus on offering innovative new products to its users. The one competitor that is most similar to our project is Massdrop, also a group buying website. The way it functions, Massdrop is website that has polls for what users want to buy. The highest voted item will be put into a “drop”; a sale for a limited time. Users can join a drop; the more people join, the cheaper the price will be, up to a limit. When the deadline arrives, if the minimum number of people joined, Massdrop buys the items at a bulk rate and then ship them to the users. Massdrop is probably the most well known of the group buying sites that deals with products; compared to TouchofModern or even sites like Groupon or SocialLiving, it gets more web traffic and visits, according to the web traffic data analytics website Alexa. Massdrop is also fairly popular among the mechanical keyboard community, since the site has a whole category dedicated to offering these.

Massdrop does have some issues. Due to the nature of the site, the time it takes from when one commits to a drop and when they receive the product can be quite long, spanning months and even years in rare cases. Another issue of the site is that their customer service is apparently quite poor, with slow or no responses and a confusing or tedious return system for items that arrived defective. Some users get no help at all with orders that arrived different than what was listed or missing parts. Furthermore, some people may not realize that the website is not the average online shopping site, but is instead a group buying site. They become shocked at the long shipping times, leading them to write negative reviews of the company on consumer protection and review sites.

TRNDii differs from Massdrop because the items on sale are not voted by the users. Instead they will be curated by the owners of TRNDii to be new and innovative products. This gives TRNDii a more focused direction, targeting not the average online shopper, but instead people who like to buy unique and cool products, who may trend younger and more technologically inclined. Additionally, instead of focusing on discounts, TRNDii is more concentrated on having the user engage in its token system. The token system allows users to spend tokens when they buy a product in order to get money back. The amount of money they get back depends on the total amount of tokens that were spent by users who want to buy a specific item. To ensure fairness to the users, the amount of tokens that users get is equivalent to if they bought an item at its discounted price. Additionally, each users start with tokens so that they may participate in the system without first having to spend money. To give a clarifying example, if an item costs 100\$ at regular price but 75\$ at bulk price, users would get 25 tokens from purchasing the item. If every user spent 25 tokens while buying this item, everyone would get 25\$ back, so it would be as if they bought the product at bulk price. In practice, the owners of TRNDii will take a

small cut of the money in order to keep operating, but the majority of the money will be redistributed to the users. We cannot really affect the shipping times of the suppliers of TRNDii, but the site will be built in order to make it easy for item suppliers to get the shipping information of buyers, in order to minimize the time taken from between product completion and actual shipping of products. TRNDii is transparent, the fact that it is a group buying website, the way to obtain tokens and the way people get money back will not be secret; everyone will know how the system functions and we aim to have everything be obvious and clear. The gamification of buying can be morally ambiguous, but we strive to make it fair for all users.

### **Contemporary Practice**

As technology and the internet get more and more entrenched into our daily lives, online shopping has become more and more popular as the years go by. According to a report<sup>1</sup> that was published in June 2016 by United Parcel Service (UPS) and comScore, shoppers now do 51% of their shopping online, with millennials being the demographic that is most likely to purchase things online. Additionally, mobile phones are becoming more and more popular for online shopper, although the majority still do their shopping on their computers (desktop or laptop).

Although there are a myriad of online shopping sites, such as Amazon or Wish, group buying sites are not as numerous, although there still exists a few of them. One such example is the previously mentioned Massdrop. Another website similar to ours is Touch of Modern; an e-commerce website that focuses on selling modern and trendy things to men. The site is members only and features things such as knives, ties, watches, etc. Items are listed on the site at a sale price until a specific time. The items only ship once the sale ends, the site operates as a group buy behind the scenes and thus the owners buy the items in bulk and then ship them out to whoever bought a specific item. TRNDii differs from Touch of Modern because the items sold are going to be more innovative and cutting edge type products. Another difference is that Touch of Modern obscures the fact that it is a group buying website while TRNDii is made with group buying at the forefront. Additionally, instead of offering discounts on items, TRNDii functions on users getting money back.

Another competitor of TRNDii is Groupon. Groupon is an e-commerce marketplace mainly focused on offering local attractions and restaurants at a cheaper price. Although they mostly offer things like Escape Room games, horseback riding or culinary experiences, they do also have physical items on sale. Groupon also allows businesses to work with it, by featuring their service or product as a sale on their site. TRNDii differs from Groupon because it is focused on products and not experiences. Also, Groupon no longer functions as a group buying site while TRNDii

---

1

[https://pressroom.ups.com/mobile0c9a66/assets/pdf/pressroom/white%20paper/2016\\_UPS\\_Pulse%20of%20the%20Online%20Shopper\\_white%20paper%20final.pdf](https://pressroom.ups.com/mobile0c9a66/assets/pdf/pressroom/white%20paper/2016_UPS_Pulse%20of%20the%20Online%20Shopper_white%20paper%20final.pdf)

does. The items on sale at Groupon are mostly more mundane while TRNDii's products are more cutting edge and exciting.

The problem of some of these sites is that there is not much user engagement. Massdrop does have its users vote which items goes on sale, but Groupon is currently aiming at getting new users in order to make money. TRNDii's token system and gamification aims to make users engaged with the site and thus stay with the site. No other group buying site currently has a system like TRNDii's. Additionally, with the items being things like innovative technologies or gadgets, millennials should be interested in these things and since they are most like to shop online and use smart phones, the website is designed to be responsive for desktop and mobile.