TRNDii Iteration 1 Summary

Team members

Jacqueline Luo 26938949
Michael Mescheder 27202202
Sam Alexander Moosavi 27185731
Eric Payette 27008058
Rameen Rastan-Vadiveloo 27191863
Jason Tsalikis 25892120

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

As this is the first iteration of the project, it was mostly based around getting a project approved, meeting with the stakeholders and setting up our project. In this initial 2 week period, we worked on 1 story, login, worth 8 points.

Overall Arch and Class diagram

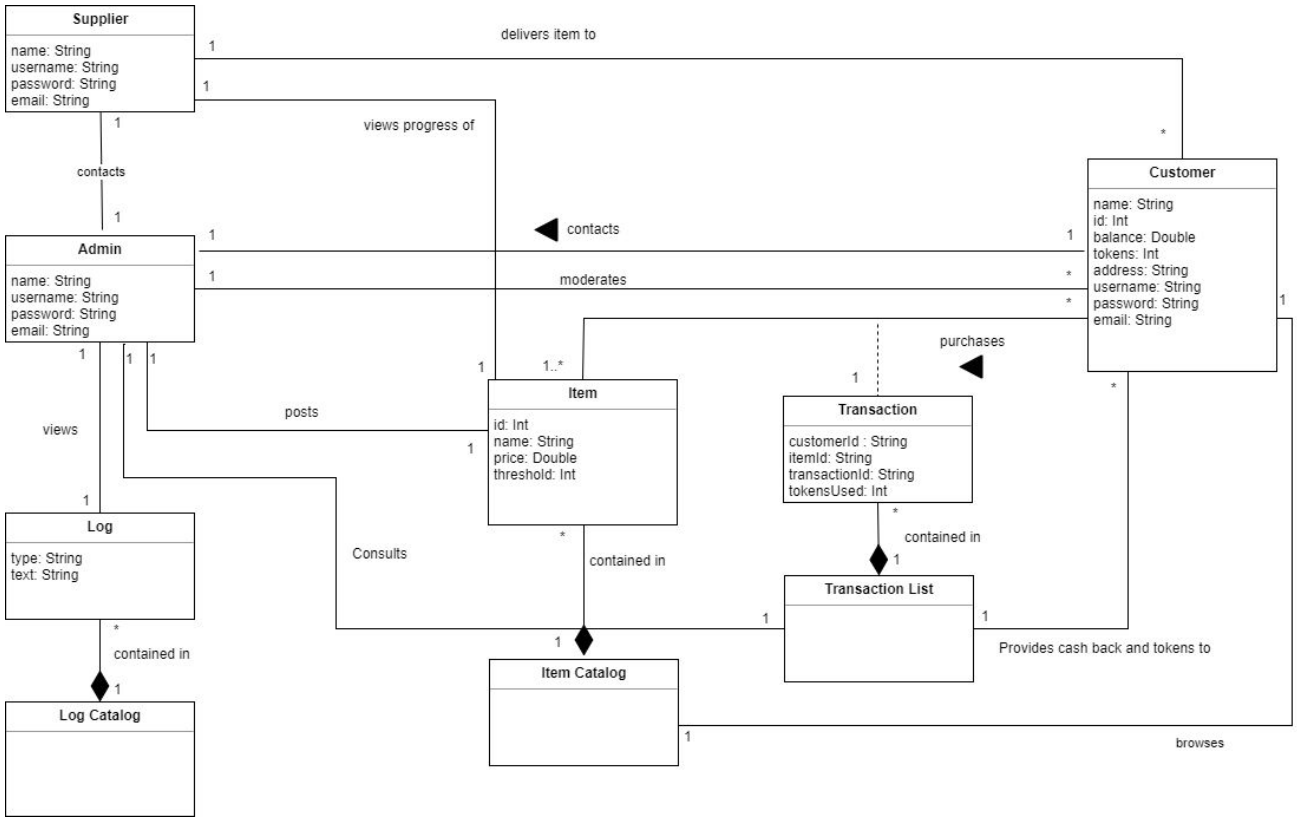

Diagram 1: Conceptual Class Diagram

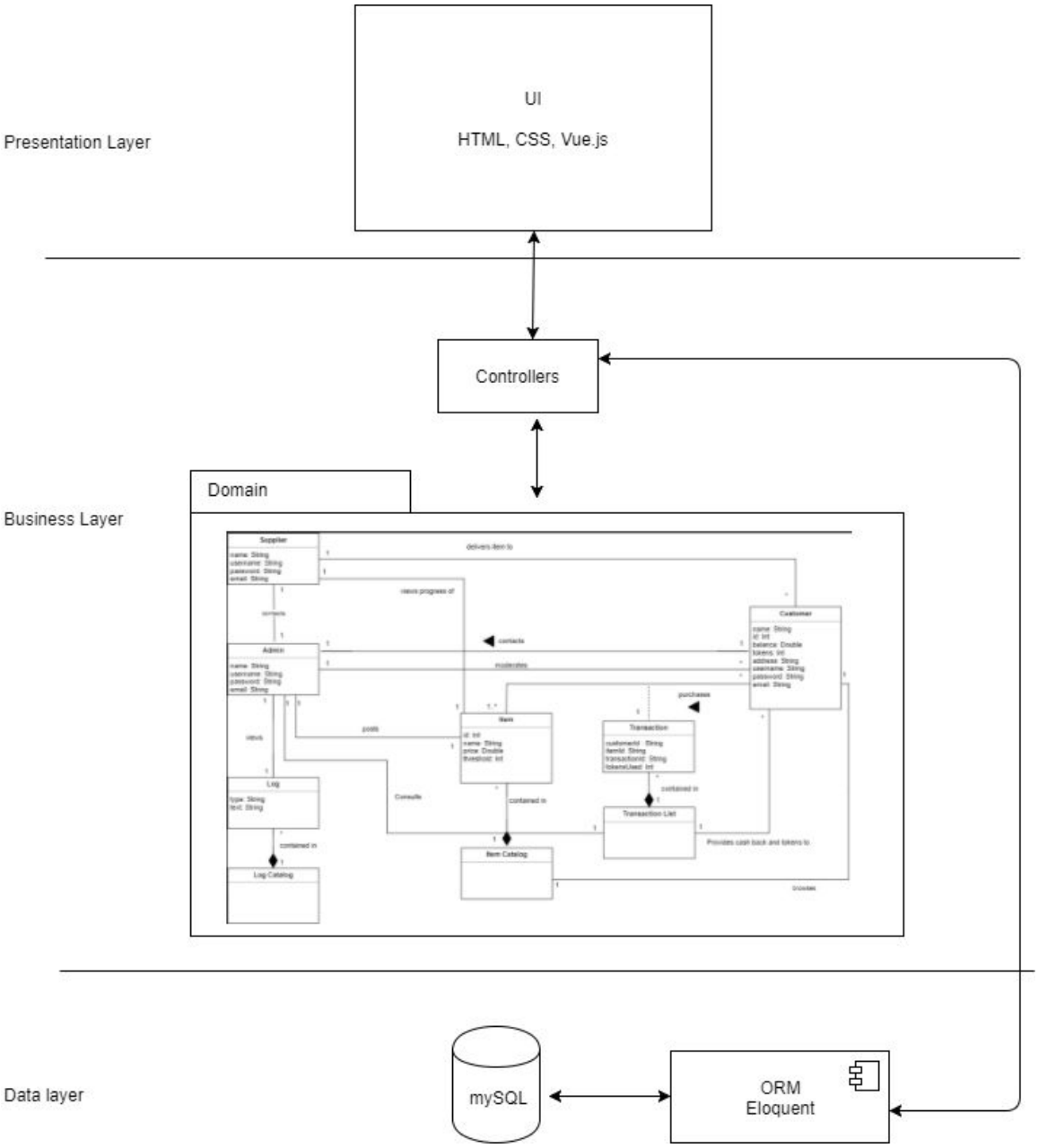

Diagram 2: Architectural Diagram

Plan up to next release

Iteration 2 (16 points)
Iteration 3, Release 1 (11.5 points, 35.5 total points)

Infrastructure

Bootstrap

Bootstrap is one of our front-end design framework of choice because it comes in a package with Laravel. Additionally, it is the most popular CSS front-end framework, meaning there are lots of resources and tutorials about it. It is responsive which is excellent as our stakeholder wants the site to work well on desktop and 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. Laravel has extensive documentation and makes it simple to implement many important things 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.

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. Also, it is easy to learn and its speed is very fast. It is also what most of our team have 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 light, 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.

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