# E-Commerce Platform (KSG Store) Release Summary

## Team members

|  |  |  |
| --- | --- | --- |
| **Spencer Hall (jsh278)**  George Anim (gba37)  Kyana Conway(ktc188) | Spencerhall01  Animkofi  KyanaConway | The number of story points (or ideal hours for tasks) that a member was an **author**. |

Each group member is responsible for counting their own story points. It is the group leader’s duty and responsibility to make sure they are accurate. Please keep in mind that we will check your GitHub stats (go to: “Graphs” on your GitHub project page, for [example](https://github.com/marouen-lamiri/Second-Soul/graphs/contributors)). Note, if your email and GitHub id are not linked properly you will not be counted properly.

You will lose 1 mark if links below are not clickable.

## Project summary (max one paragraph)

This project is an e-commerce platform developed to perform at a high level the actions below based on user type:

1. Buyer – Login, search, compare, buy, return items and logout
2. Seller – Login, add products, sell, receive payments and logout
3. Admin – Login, approve/block new user accounts and products, oversee different user actions and logout

## Velocity and a list of user stories (for [example](https://github.com/solid/user-stories)) and non-story tasks for each iteration

(make sure the iteration is a clickable link to the milestone/sprint on GitHub)

Sprint 1: Create SRS document

Link to github - Click [here](https://github.com/spencerhall01/Group-13/tree/main/srs).

Sprint 2: Create diagrams.

In this sprint the following diagrams were created:

* Use case diagrams
* UML class diagrams
* UML sequence diagrams for each user story
* UML state diagrams and activity diagrams for each user story
* Architectural design diagram and component-level design

Link to github – click [here](https://github.com/spencerhall01/Group-13/tree/main/diagrams)

Sprint 3: Implements features of platform

In this sprint, the group created the e-commerce platform and implemented the following features:

* Created the platform where all users (buyer, seller and admin) can login.
* During account creation, a user can select type of account (buyer or seller)
* Seller account can log in and add products to store only after approval from Admin
* Once approval for added products is obtained, buyers can search and purchase the approved and added items.
* Admin can approve/block accounts as well as products.
* Successfully deployed application on AWS

Link to github: Click [here](https://github.com/spencerhall01/Group-13/tree/main/code).

Sprint 4: Make enhancements based on comments from TA

At the creation of this document, this sprint was still ongoing.

In this sprint, the team made the following enhancements to delivery from sprint 3:

* Name field in login in page was restricted to only strings and does not accept numeric entries
* After buyer searches for item, they can click and it sends them to home where they can make purchase
* Seller can now add how many quantities of items they are listing for sale
* Once items are sold out buyer can’t purchase the sold out item

Link to github:

## Overall Arch and Design

Show us the overall architecture (block diagram) in your system with an architecture diagram.

Show the UML class diagram for your system. If you have multiple packages, show the diagram for at least one package that has more than 10 classes. You can also include these diagrams in your stories on GitHub (by providing URLs).

## Infrastructure

For each library, framework, database, tool, etc

**Name and link**

Max 1 paragraph description of why you are using this framework.

Max 1 paragraph description of other alternatives and why you didn’t choose them.

## Name Conventions

List your naming conventions or just provide a link to the standard ones used online.

For example: [Java naming conventions](http://www.oracle.com/technetwork/java/codeconventions-135099.html)

## Code

Key files: top **5** most important files (full path). We will also be randomly checking the code quality of files. Please let us know if there are parts of the system that are stubs or are a prototype so we grade these accordingly.

|  |  |
| --- | --- |
| File path with a clickable GitHub link | Purpose (1 line description) |
|  |  |

## Testing and Continuous Integration

Each story needs a test before it is complete. If some class/methods are missing unit tests, please describe why and how you are checking their quality. Please describe any unusual/unique aspects of your testing approach.

List the **5** most important unit test with links below.

|  |  |
| --- | --- |
| Test File path with clickable GitHub link | What is it testing (1 line description) |
|  |  |

List the **5** most important acceptance tests with links below.

|  |  |
| --- | --- |
| Test File path (if you automated the test) or as comments in Github issues (if it’s done manually) with clickable GitHub link | Which user story is it testing (1 line description) |
|  |  |

Describe your continuous integration environment. Include a link to your CI.

Describe the choice of the static analysis tool and how do you run it. The static analysis tool should analyze the language that is used in the majority of your source code.

Attach a report as an appendix (not counted for the 6 pages) from static analysis tools by running the static analysis tool on your source code. Randomly select 10 detected problems and discuss what you see.