Global Distributed Software Development
Fulda Fall 2019

Project

LetStuffGo

Milestone 2

21st November 2019

Team G6

Syeda Tasneem Rumy, syedatasneemrumy@gmail.com, (Team Lead & Backend Lead)
Emin Gasimov, 1155853 (GitHub Master)
Md Firoj Kabir, 1300934 (Frontend Lead)
Muhammad Naeem Afzal, 1248326 (Frontend)

Revision History

Revision	Status	Date
Version 0.1	Created	20-11-2019
Version 0.2	Submitted for review	21-11-2019

CEO & CTO: **Prof. Rainer Todtenhoefer,** Fulda University, Germany

Table of Contents

1. Functional Requirements (Prioritized)

Priority 1 - Must have

Any User

Search product: User can search products by category.

Sort product: User can sort the search results by Price and Latest Post

View details of product: User can view the details of a product.

Anonymous Users

Register: Users should be able to **register** to the website

Login: Users should be able to **login**

Registered Users (Buyer / Seller)

Dashboard: After Login users will have a Dashboard

Post Product: Registered users should be able to post product for sale

Send Message: Registered Users (Buyer/Seller) can send messages to each other

View Message listing: Registered User (Buyer/Seller) can see the list of messages sent or received in their Inbox of Dashboard

View Message Details: Buyer/Seller can read the conversation thread with another Seller/Buyer

View List of Products bought: Buyer can see the list of items bought

Mark Product as Sold: Seller can mark a product as sold and remove item from search result filtering

Cancel Purchase: Seller can cancel the purchase and make it available for sale again.

Admin User

Approve Posted Product: Moderator/Admin can approve a posted product for publishing on the site

Disapprove Posted Product: Moderator/Admin can disapprove a posted product for publishing on the site

Block Posted Product: Moderator/Admin can block a posted product after being publishing on the site

Block User: Moderator/Admin can block user who has posted inappropriate products on the site

Priority 2 - Desired

Registered Users (Buyer / Seller)

Request Purchase: Buyer can request to buy a product to seller

Approve Purchase: Seller can approve the purchase

Closing Deal: Buyer and Seller can close the deal after purchase.

View List of Items created / approved and on sale/rejected/sold: Seller can see the list of products created / approved and on sale/rejected/sold

Delete / Disable unsold Products: Seller can delete/disable a product when not sold

Blocking of / Deletion / Disabling of sold product: Seller can not delete/disable a product when it has already been sold

☐**Add item to Favourite List:** Buyer can mark a product to see in his **Favourite List**

☐**Remove item from Favourite List:** Buyer can unmark a product from his **Favourite List**

Priority 3 - Opportunistic

Any User

☐**View Products on sale from the same seller:** Users can view a list of all products on sale by the seller with his contact information.

Registered User

☐**View Seller Contact Info:** All registered users can view seller's contact info

Admin User

Delete/Block bulk products: Moderator/Admin can delete / block a group of posted product from one or more sellers.

Closing a Deal: Moderator/Admin can close a deal on behalf of a seller and buyer if required.
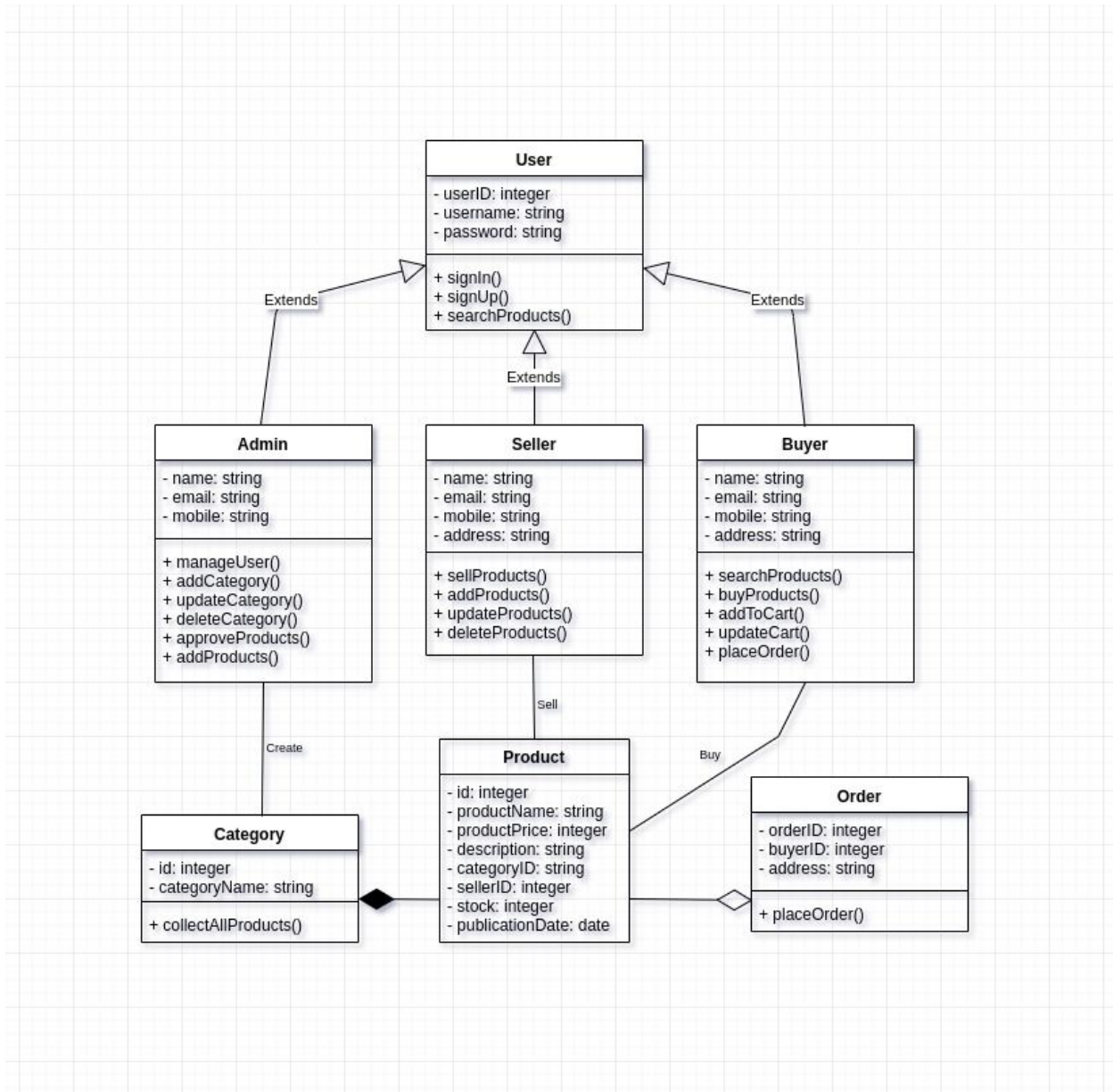

☐ **= Unique features of our product**
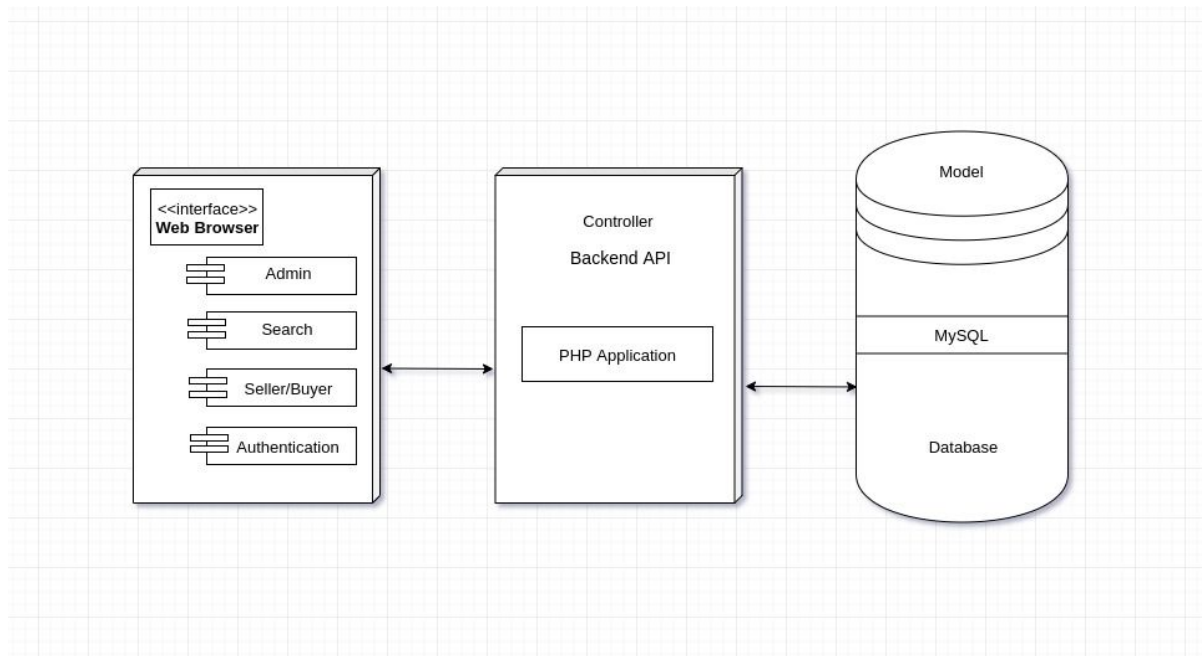
2. UI Mockups and Storyboards (High-Level)

3. High Level Architecture, Database Organization
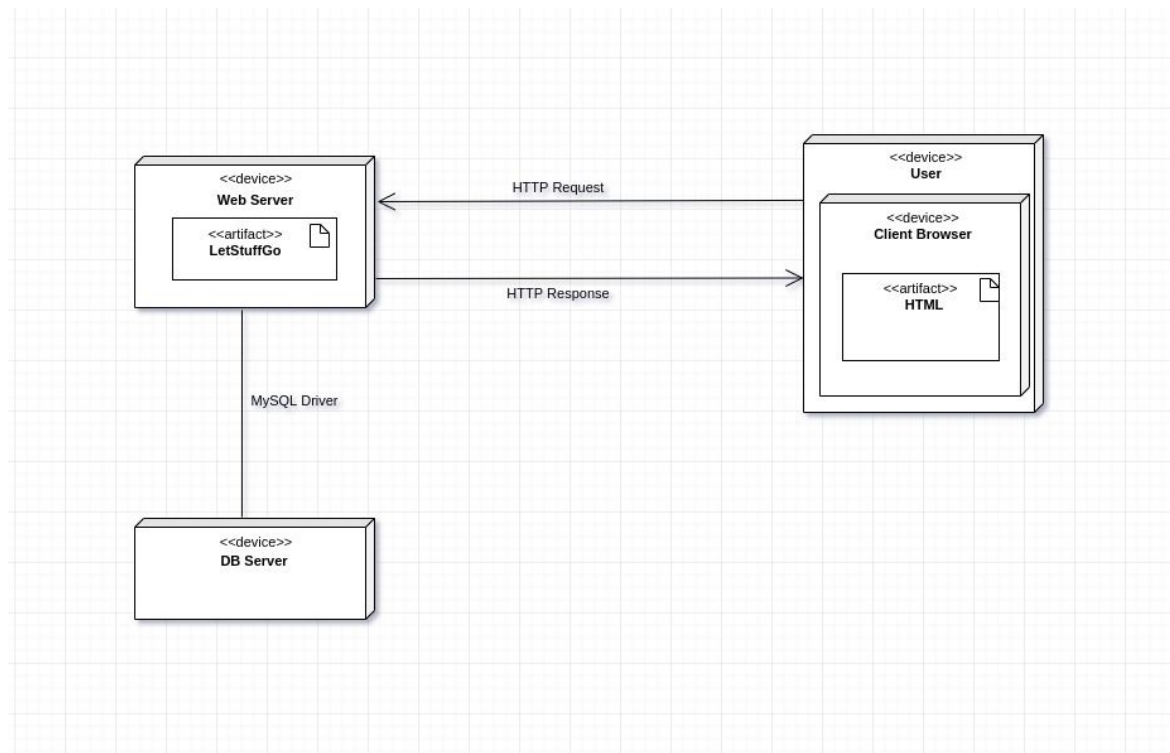
4. High-Level UML Diagrams

4.1 Class Diagram

4.2 Component Diagram

4.3 Deployment Diagram

5. Identify Actual Key Risks For Your Project At This Time

Skill Risks

❏ **Lack of experience in back-end technology:** We are 4 members in the group and 2 are front end and 2 are working on the back-end. Lack of experience in back-end at 2 members and getting experience from other group members.

❏ **Working with console:** Every team member does not experience working on the console. As a team, we are sharing our experiences with each other and try to get used to it.

Schedule Risks

❏ **Vertical SW Prototype:** The vertical SW Prototype needs a lot of time because our group is very small consisting of only 4 members.

❏ **Nonuniform knowledge about tools:** Not everyone has the similar knowledge on every tool. Before using any new tool some of us are learning it then discussed it in the team. This can take more time to teach each other.

Team Risks

❏ Different teammates have different schedules and fixing time for meeting can take time.

Content/Legal Risk

❏ **Selling illegal items:** The admin will observe all of the products uploaded by any seller. If there is any illegal item, the admin will ask for the illegal part before publishing it. After fixing it from the seller side, it can be published by Admin.

6. Project Management

From the beginning of our project starting from M0 milestone, we as a team have been using a task management tool, Asana, All group members have been invited to use Asana's online platform which has convenient task creation, assignment and viewing capabilities. It has a unified dashboard, status and task update notification, calendar view and many more productive features. It is nowadays very popular among most software companies in the market.

In Asana, the team lead created a project for each on-going milestone and created milestone specific tasks under that project. Then as per discussion in the team meetings, tasks were assigned to interested members by team lead. Team members can also choose any task from the list and assign it to themselves. Tasks that were related to user interface or front-end were assigned to those who are interested and have skills in front-end development, Tasks related to backend are assigned to those interested and have experience related to backend technology. Besides assigning tasks we also set a deadline for each task by estimating possible time requirement based on member feedback, milestone deadline, team capability and task load. Whenever a task is completed by a member, they login to their online Asana account and update their task status from 'Open' to 'Completed'. Team lead manages the tasks to keep it up-to-date.