Hochschule Fulda
University of Applied Sciences

Global Distributed Software Development

GDSD Team Project WS2020

Global Software Development

Project

FuldaMarkt

Milestone 5

Final Project for GDSD Class Fall 2020

2nd February 2021

Team 2 (local)

Syeda Tasneem Rumy, 1189706 (Team Lead, GitHub Master & Backend Lead),

syeda-tasneem.rumy@informatik.hs-fulda.de

Syed Sumair, 1310771 (Frontend Lead)

Ahmed Abdullah, 1250378 (Backend Developer)

URL of Demo: https://ec2-34-226-123-197.compute-1.amazonaws.com/

CEO & CTO: Prof. Dr. Rainer Todtenhöfer, Fulda University, Germany

Contents

1. Product Summary

- **Name of Product:** “Fulda Markt”
- **List of committed** functions
  - Register User
  - Login User
  - Market Search View
    - based on multiple categories filter
    - based on multiple order of sorting
  - Post a new item for sale
    - Input different fields for an item listing
    - Upload multiple images for an item
  - Item details
    - Item details with multiple images
    - Item details with related items(a/c to selected item category)
  - Messaging
    - Contact Seller via item
    - Contact Buyer via User’s dashboard messages section
  - User Wishlist
    - Add an item to wishlist
    - Remove an item from wishlist
  - Admin rights
    - Approve/decline any new or existing item from items list for market listing
    - Enable/disable new or existing user from users list
    - Change role of any user to make a normal user or give admin rights
- **URL to access product:** https://ec2-34-226-123-197.compute-1.amazonaws.com/

2. Milestone documents: M1, M2, M3, M4

The milestone documents M1, M2, M3 an M4 follows from next page.

Hochschule Fulda
University of Applied Sciences

Global Distributed Software Development

Master AI Team Project WS2020

Global Software Development

Project

FuldaMarkt

Milestone 1

16th November 2020

Team 2

Syeda Tasneem Rumy, 1189706 (Team Lead, GitHub Master & Backend Lead),

syeda-tasneem.rumy@informatik.hs-fulda.de

Syed Sumair, 1310771 (Frontend Lead)

Ahmed Abdullah, 1250378 (Backend Developer)

Chowdhury Amlan Barua, 1310643 (Frontend Developer)

Revision History

Revision	Status	Date
Version 0.1	Submitted	November 16th, 2020
Version 0.2	Revised	November 17th, 2020

CEO & CTO: Prof. Dr. Rainer Todtenhöfer, Fulda University, Germany

Contents

1. Executive Summary

We, the students of Hochschule Fulda are assigned to design a system in Global Distributed Software Development course for students, faculty and other staff of the university. As the life-after-Covid employs, the acceptance/adoption of online solutions are drastically increased all over the world. Especially for the educational institutes and ecommerce applications. This study aims to establish a preliminary assessment, evaluation and understanding of the characteristics of online shopping experience for the students, faculty and university staff.

This system named "FuldaMarkt" is an online web-based application for the above-mentioned target group. Users may offer their passed down goods or services for sale and to search for goods or services available by other users to purchase. It gives user a real-time chat facility for the offered product/service respectively. This makes comfort for both the seller and buyer as the Buyers can get the details on chat, ask for price as well as negotiate over price using this chat facility, similarly sellers wont be wasting time on unexpected deal breakers. FuldaMarkt also allows users to offer online tutoring sessions or extracurricular activities for students and faculty. The credibility of this process is ensured by the administrative approval of every goods or services.

FuldaMarkt web application is available online for all the users having their active Fulda University email address through which they can register in the application. This bail out the integrity by spam users other than Fulda University. Once the registration is successful then the user may avail all the features of the application. All the activities/posts of buyers and sellers can be monitored by the administrators of the system and can be approved or denied depending on their policies.

FuldaMarkt is being developed by Team 2 which consists of 4 members, every team member have their own experience in the field of software development and belongs to different countries, Syeda Tasneem Rumy is the Team Lead, backend lead and Github lead, Ahmed Abdullah is working as backend developer, Syed Muhammad Sumair is working as a front-end lead and Amlan Chowdhury is working as a front-end developer.

2. Personae and main Use Cases

2.1 Personae

In this section, the categories of users who are likely to use the application will be discussed, along with the use cases in which the application will be useful. The application usage will assume that all users accessing it have a stable internet access and a PC/Laptop to allow for adequate viewing of the website.

Category 1: Students:

This application will be a portal for all students of Hochschule Fulda (probable expansion to SFSU in the future) to buy, sell and exchange goods and services, if the items presented are approved by a website administrator. The goal is to provide a trustworthy place for students to exchange common goods and equipment that will likely aid them in their studies or research.

Category 2: University staff and faculty members:

In addition to students, staff and faculty members can use the website to organize extracurricular activity or tutoring sessions through a separate section on the website. This way, students are more aware of how and when these activities are taking place and do not need to scroll through their emails or moodle posts to know what kind of events are taking place at their university.

2.2 Main Use Cases

2.2.1 **Use Case 1:** A student looking to replace their webcam

During his studies, a student of Hochschule Fulda is having difficulties using his laptop webcam to communicate with his tutors in the online class sessions. He is on a tight budget and cannot afford to buy a brand-new webcam from an electronics store. He decides to browse the exchange website to look for good-condition webcams to use in his online classes. After finding a post of another student selling his/her webcam online. They start organizing a meeting via the website chat service to complete the transaction.

2.2.2 **Use Case 2:** A student looking to sell their laptop

After upgrading to a better machine, a student of Hochschule Fulda is looking to sell/giveaway her old laptop to a trustworthy student from the same university. She had previous bad experiences using other e-commerce solutions and wants to make sure the transaction is as safe as possible while benefitting other students in need of a good laptop.

2.2.3 **Use Case 3:** International office trip

The international office of Hochschule Fulda wants to organize a trip for newly enrolled students and other students who are interested to get to know each other. They can post an announcement on the events section of the website to let students know about when the event is taking place and the activities of the event, in addition to fees to be paid for admission. Students can then use the contact details provided in the announcement to register in the event.

2.2.4 **Use Case 4:** Preparation courses

The faculty of applied informatics is looking to host a preparatory/practice session for newly enrolled students or students looking to refresh their skills in subjects such as mathematics or programming. They can post an announcement on the events section of the website. The announcement includes the time and place of the session, fees, contact details and possibly the current number of free seats available for the session.

2.2.5 **Use Case 5:** For students, by students

An outstanding student has noticed that his classmates are struggling in understanding one of the courses they are enrolled in. In order to help his fellow classmates, he has decided to host an online session outside of class times to do more practicing, for a simple fee. He can use the application to post an advert about the time of the session and the fee of registration.

3. List of main data items and entities – data glossary/description

In this section, the types of users who will be implemented and their privileges will be discussed. Additionally, a general description of the data that will be attached to users and their posts will be laid out. This section outlines general information and is highly subject to change as the application development progresses.

Below is a table that describes the website users and their privileges:

User type	Create post in “Event” Section	View post in “Events ” Section	Modify post in “Events” Section	Delete post in “Events” Section	Create post in “Market” Section	View post in “Market” Section	Modify post in “Market” Section	Delete post in “Market” Section
Admin	Y	Y	Y(A)	Y(A)	Y	Y	Y(A)	Y(A)
Student	N	Y	N	N	Y(*)	Y	Y(*)	Y
Faculty	Y	Y	Y	Y	Y	Y	Y	Y

- Administrators have full privileges on the website.
- They can create posts on both sections of the website and modify or delete any posts.
- Students can only view posts on the “Events” section. On the “Market” section they can (*) post and modify their posts but only after administration approval.
- Faculty members can create and view a post on both sections but can only modify and delete their own posts.
- All users can message each other via the website chat service.

Users have the following data attached to them (can be modified in the future):

- Username: The username that is created at first log-in.
- Type: The type of user can be modified by the admin.
- Email: The email has to end with “hs-fulda.de”. No email verification will take place.
- Posts: Posts that the user has posted. Possibly viewable when clicking on the username.

Posts have the following data attached to them (can be modified in the future):

- Post section. Can be either “Events” or “Market”.
- User who created the post.
- Post picture(s).
- Post text.

4. Initial list of functional requirements

1. All users should be able to browse for available items on the website.
2. All users should be able to search for items on the website.
3. All users should be able to filter items on the website by category.
4. All users should be able to sort items by posted date and price in ascending or descending order.
5. All users should be able to view the details of items on the website.
6. Users who provide emails with suffix, “hs-fulda.de” should only be able to register for the website.
7. Registered users should be able to login to the website.
8. Registered users (Sellers) should be able to post items for sale.
9. Registered users (Sellers) should be able to post items that can either be physical products or services.
10. Registered users can be of two types, Student and Faculty of HS Fulda.
11. Registered users should be able to view the contact details of the Seller.
12. Registered users (Buyers) should be able to send message to the seller to buy available items posted by the seller.

13. Registered users (Sellers) should be able to reply to messages sent by the buyers to sell items.
14. Registered users (Buyers and Sellers) should be able to view list of messages sent and received.
15. Registered users (Buyers) should be able to request to buy an item.
16. Registered users should be able to mark an item as ‘favorite.’
17. Registered users should be able to unmark an item from ‘favorite’ list.
18. Registered users (Sellers) should be able to mark an item as sold, unavailable, available etc.
19. Registered users (Buyers and Sellers) should be able to see the list of items they bought or sold.
20. Admin User should be able to approve or disapprove an item posted by a registered user (seller).
21. Admin User should be able to delete inappropriate items posted by a registered user (seller).
22. Admin User should be able to delete/block registered users from the system.

5. List of non-functional requirements

1. Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0.
2. Application shall be optimized for standard desktop/laptop browsers. It will render well on two latest versions of two major browsers: Google Chrome and Firefox.
3. All or selected application functions must render well on mobile devices
4. Data shall be stored in the database on the team’s deployment server.
5. No more than 50 concurrent users shall be accessing the application at any time.
6. Privacy of users shall be protected, and all privacy policies will be appropriately communicated to the users.
7. The language used shall be English.
8. Application shall be very easy to use and intuitive.
9. Application should follow established architecture patterns.
10. Application code and its repository shall be easy to inspect and maintain
11. Google analytics shall be used (optional)
12. No e-mail clients shall be allowed. Interested users can only message to sellers via in-site messaging.
13. Pay functionality, if any shall not be implemented nor simulated in UI.
14. Site security: basic best practices shall be applied for main data items
15. Media formats shall be standard as used in the market today
16. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
17. The application UI (WWW and mobile) shall prominently display the following exact text on all pages "Fulda University Software Engineering Fall 2020. For Demonstration Only” at the top of the WWW page.

6. Competitive analysis

	FuldaMarkt	Amazon	Udemy	eBay Classified
Product Listing	Yes	Yes	No	Yes
Service Listing	Yes	Yes	Yes	No
Target Group	Student/Faculty/Staff	Everyone	Students/Faculty	Everyone
Chat	Yes	No	No	Yes
Delivery/Shipping	No	Yes	No	Yes
Feedback/Rating	Only user feedback	Yes	Yes	Yes
Online Payment	No	Yes	Yes	Yes

There are many well recognized online platforms available in the market which offers ecommerce solutions. FuldaMarkt also offers ecommerce features in this market and have relatively limited features due to the project scope, but it covers almost all core features. FuldaMarkt is like a hybrid system which provide solution for both the ecommerce and online tutoring.

7. High-level system architecture and technologies used

Technology Stack

- Server Host: Amazon AWS EC2 1vCPU 1 GB RAM
- Operating System: Ubuntu Server 20.04 LTS
- Server Database: MySQL v.14.14 Distrib 5.7.27
- Web Server: Apache v.2.4.29 (Ubuntu)
- Server-Side Language: PHP 7.2.24

Additional Technologies

- IDE: Intellij Idea 2020.1
- Web analytics: Google Analytics
- Frontend Library: Bootstrap 4.5

8. Team and roles

Team member	Role
Syeda Tasneem Rumy	Team Lead, GitHub Master, Document Master & Backend Lead
Syed Sumair	Frontend Lead
Ahmed Abdullah	Backend Developer
Chowdhury Amlan Barua	Frontend Developer

9. Checklist

Sl #	Item	Status
1.	So far, all team members are engaged and attending WebEx sessions when required	OK
2.	Team found a time slot to meet outside of the class	OK
3.	Back end, Front end leads, and GitHub master chosen	OK
4.	Team decided and agreed together on using the listed SW tools and deployment server	OK
5.	Team ready and able to use the chosen back and front-end frameworks and those who need to learn are working on learning and practicing	ON TRACK
6.	Team lead ensured that all team members read the final M1 and agree/understand it before submission	ISSUE: Did not get confirmation from **Chowdhury Amlan Barua** if he read the M1 document.
7.	GitHub organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.)	ON TRACK

Hochschule Fulda
University of Applied Sciences

Global Distributed Software Development

GDSD Team Project WS2020

Global Software Development

Project

FuldaMarkt

Milestone 2

1st December 2020

Team 2 (local)

Syeda Tasneem Rumy, 1189706 (Team Lead, GitHub Master & Backend Lead),

syeda-tasneem.rumy@informatik.hs-fulda.de

Syed Sumair, 1310771 (Frontend Lead)

Ahmed Abdullah, 1250378 (Backend Developer)

Chowdhury Amlan Barua, 1310643 (Frontend Developer)

Revision History

Revision	Status	Date
Version 0.1	Draft	November 24th, 2020
Version 0.2	Submitted for review	December 1st, 2020

CEO & CTO: Prof. Dr. Rainer Todtenhöfer, Fulda University, Germany

Contents

1. Functional Requirements - prioritized

PRIORITY 1 - MUST HAVE

Unregistered Users

- Register: Users (*Students, Faculty* and *Staffs* of HS Fulda) should be able to register to the website when providing unique email address with suffix “hs-fulda.de”.

Any User (With/Without Logging in)

- Browse products: User can view the list of latest products posted on the website.
- Search product: User can search products by product title and category.
- Filter product: User can filter published items by category
- Sort product: User can sort published or search result items by published/posted date and price in ascending or descending order.
- View details of product: User can view the details of a published product.

Registered Users (Buyer / Seller /Admin)

- Login: Registered users should be able to login using their correct login credentials unless they are blocked by an Admin user.
- Dashboard: After Login, users should be able to view a Dashboard.
- Post Product: Registered users should be able to post products (physical products/services) for sale
- View Product listing: Registered users should be able to view the list of products they posted.
- Send Message: Registered Users (Buyer/Seller) can send messages to each other
- View Message Details: User (Buyer/Seller) can read the conversation thread with another User (Seller/Buyer)
- View Message listing: Registered User (Buyer/Seller) can see the list of messages sent or received in the Inbox of Dashboard
- View Seller Contact Info: Users should be able to view seller’s contact info
- Closing Deal: Seller can close the deal after selling by marking the published product as “Sold”
- Add product to Favorite List: Users can mark a product to see in his Favorite List
- Remove product from Favorite List: Users can unmark a product from his Favorite List
- View list of products in Favorite list: Users can see their own list of items added to favorite list.

Admin User

- View list of products pending approval: Admin user should be able to view the list of products pending approval.
- Approve Posted Product: Admin user should be able to view details of a product and publish it by marking it as ‘Published’.
- Disapprove Posted Product: Admin user should be able to view details of a product and disapprove it by marking it as ‘Unapproved’.
- Block Posted Product: Admin user can block a posted product after being publishing on the site
- Block User: Admin user can block any user who has posted inappropriate products on the site.

PRIORITY 2 - DESIRED

Registered Users (Buyer)

- Request Purchase: Buyer can request to buy a product to seller by sending automated message.

Registered Users (Seller)

- View list of products by its status: Seller can see the list of Products by its current status (e.g. Created/Approved/Published/Unapproved/Sold/Unavailable).
- Disable unsold Products: Seller can disable a product when not sold.

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 in products details page.

Registered Users (Buyer)

- View list of Products bought: Buyers can see the list of items bought

Registered Users (Seller)

- Disabling of product: Seller can only disable a product when it is available for selling. Sold products cannot be disabled.

Admin User

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

2. List of main data items and entities

In this section, the types of users who will be implemented and their privileges will be discussed. Additionally, a general description of the data that will be attached to users and their posts will be laid out.

The data description is the same from Milestone 1 document but has some expanded definitions and modifications.

Below is a table that describes the website users and their privileges:

User type	Create post in "Events" Section	View post in "Events" Section	Modify post in "Events" Section	Delete post in "Events" Section	Create post in "Market" Section	View post in "Market" Section	Modify post in "Market" Section	Delete post in "Market" Section
Admin.	Y	Y	Y(A)	Y(A)	Y	Y	Y(A)	Y(A)
Student	N	Y	N	N	Y(*)	Y	N	Y
Faculty	Y	Y	Y	Y	Y	Y	Y	Y
Unregistered	N	Y	N	N	N	Y	N	N

- Administrators have full privileges on the website. They can create posts on both sections of the website and modify or delete any posts.
- Students can only view posts on the "Events" section. On the "Market" section they can (*) post their items but only after administration approval. Students cannot modify their posts after administrator approval, but they can delete them freely.
- Faculty members can create (without administrator approval) and view a post on both sections but can only modify and delete their own posts.
- All registered users can message each other via the website chat service.
- Unregistered users can only view posts from both sections.

Registered users have the following data attached to them:

- User ID.
- Username: The username that is created at first log-in.
- Type: The type of user can be modified by the administrator.
- Email: The email has to end with "hs-fulda.de". No email verification will take place.
- Posts: Posts that the user has posted. Possibly viewable when clicking on the username.
- Status: States whether the user is active or suspended by an administrator.
- Items marked as "favorite" by the user.

Posts have the following data attached to them (can be modified in the future):

- Post ID.
- Post title.
- Post section. Can be either "Events" or "Market".
- User who created the post.
- Timestamp. Date only should be sufficient.
- Post picture(s).
- Post text.

Additional data for market posts:

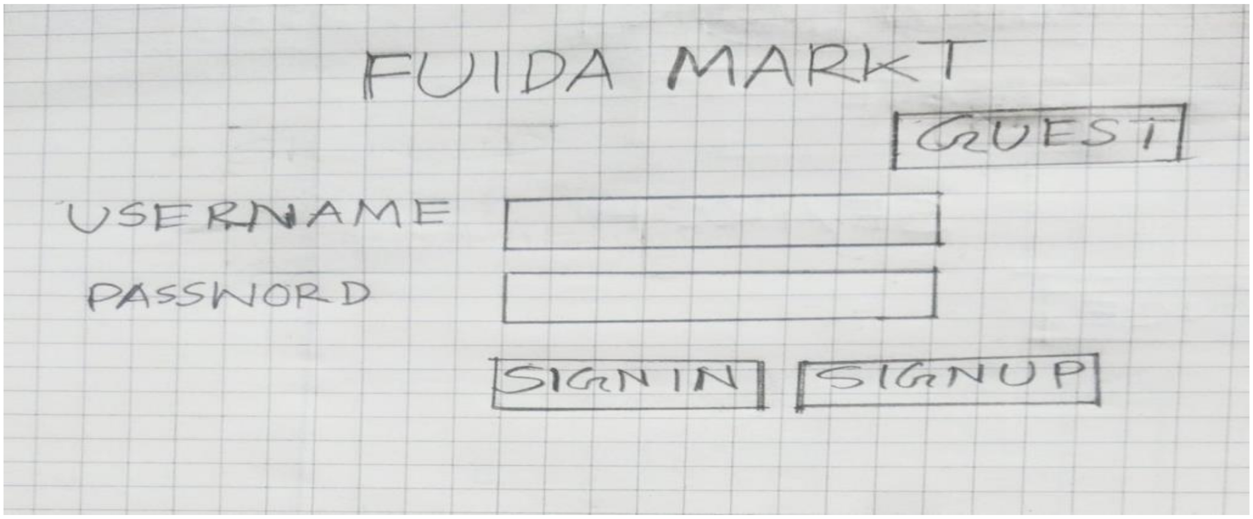
- Post category for “Market” items. For example: “Electronics”, “Service”, “Miscellaneous”.
- Price for the listed item.
- Post status for “Market” items. For example: “Pending approval”, “Available”, “Sold”.
- User which the “Market” item was sold to.
- Users who marked the “Market” post as “favorite”.

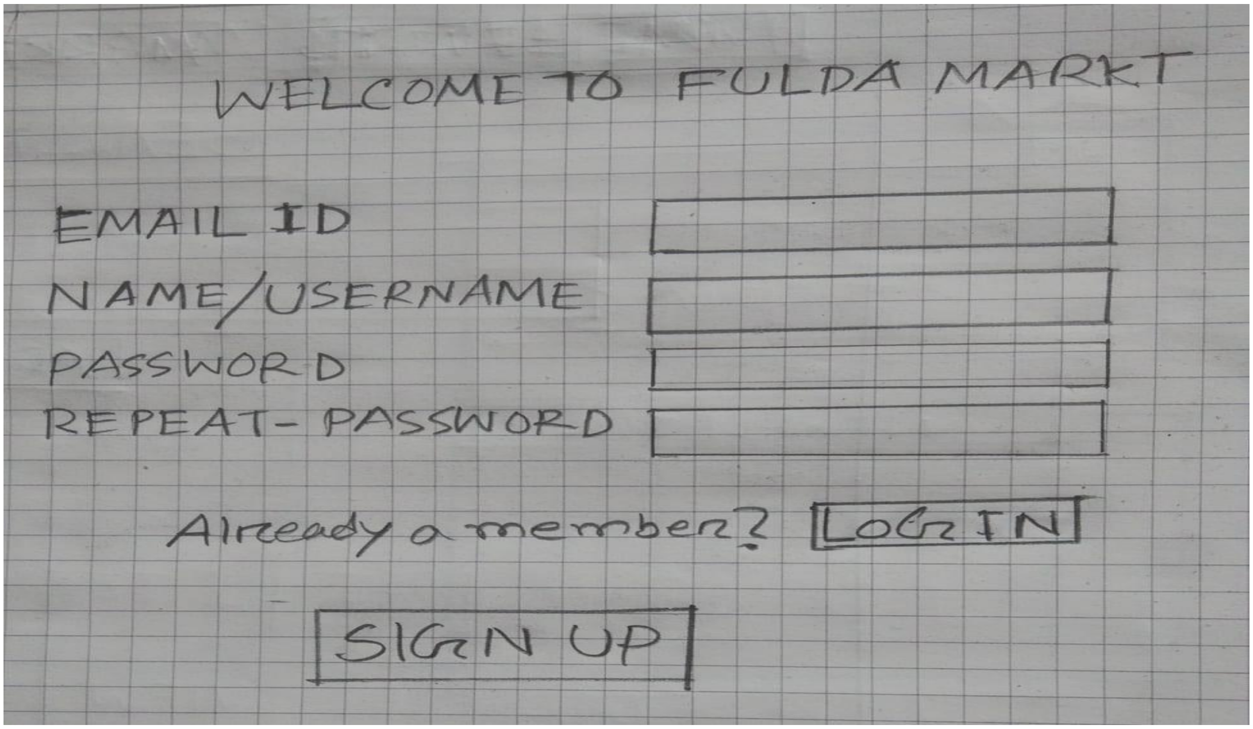
3. UI Mockups and Storyboards

UI Mockup: Below four major use cases have been used for UI mockup design purpose.

Use case 1: Sign up:

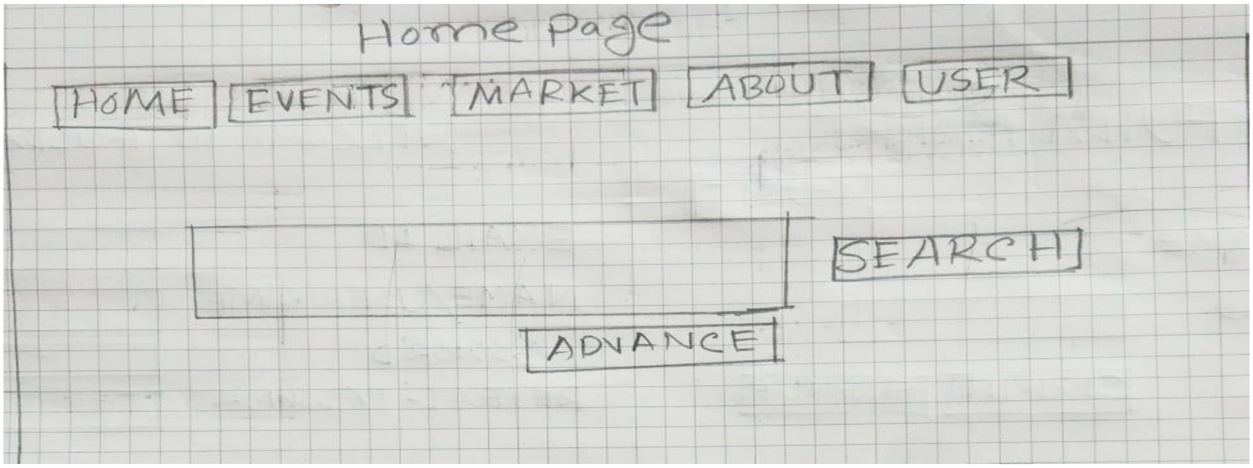
In this storyboard, students or staff wants to sign up for **FuldaMarkt**. He / She goes to login page. He /She notices SIGN UP button in the login page. So, He/ She clicks on **SIGN UP** button. After clicking on **SIGNUP** button, Sign-up page will appear. Then He / She will enter required information. After giving all necessary information, He / She will click SIGNUP button for final registration.

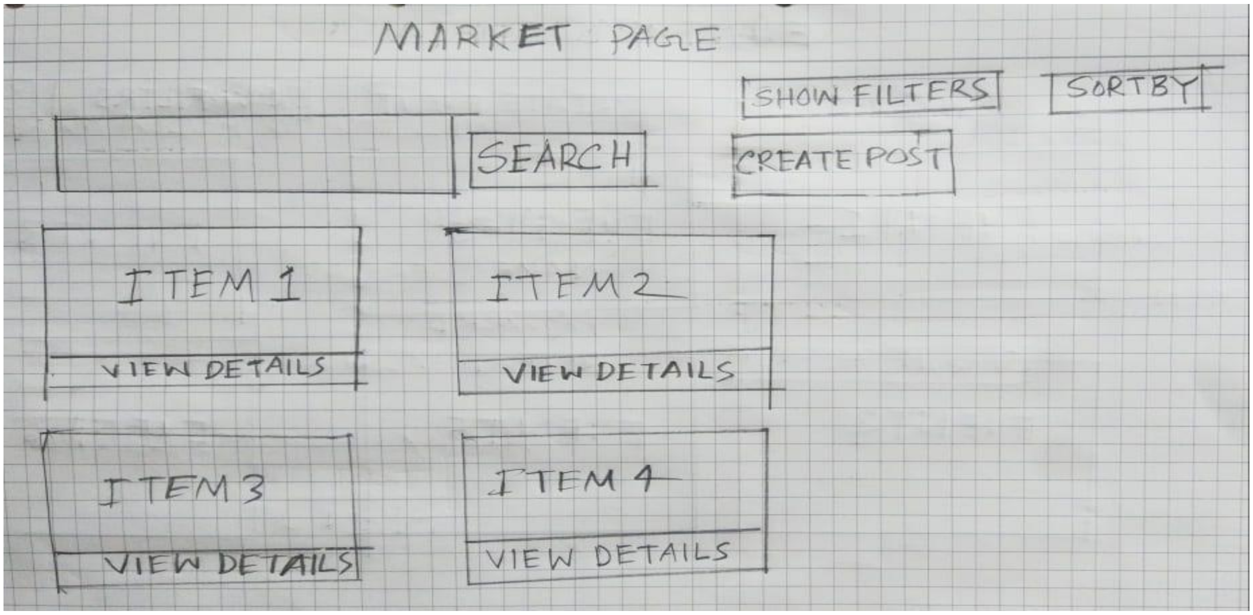


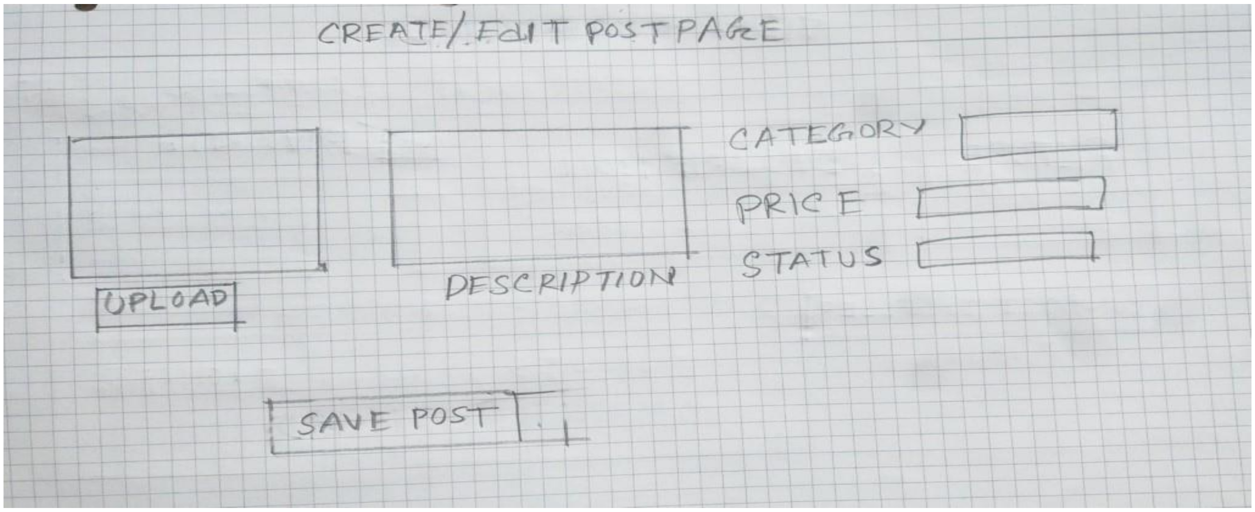


Use Case 2: Sell Item

In this scenario, user wants to sell a laptop which was used by user for a few years. So, user goes to home page. User notice a tab called **Market**. For selling an item, user clicks on tab called Market. A new page appears. Then user clicks on **CREATE POST** Button. A new page appears. Then user gives all required information regarding the item and upload picture of the item. Then user clicks on **SAVE POST** button.

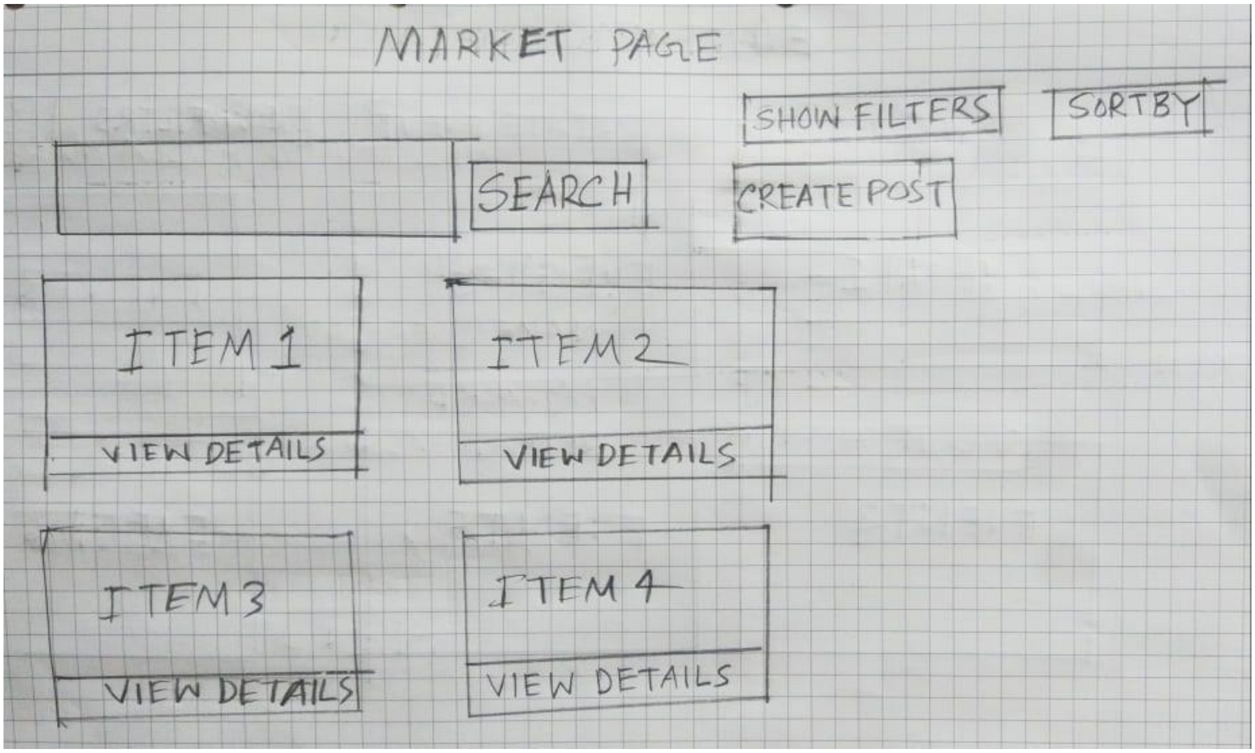


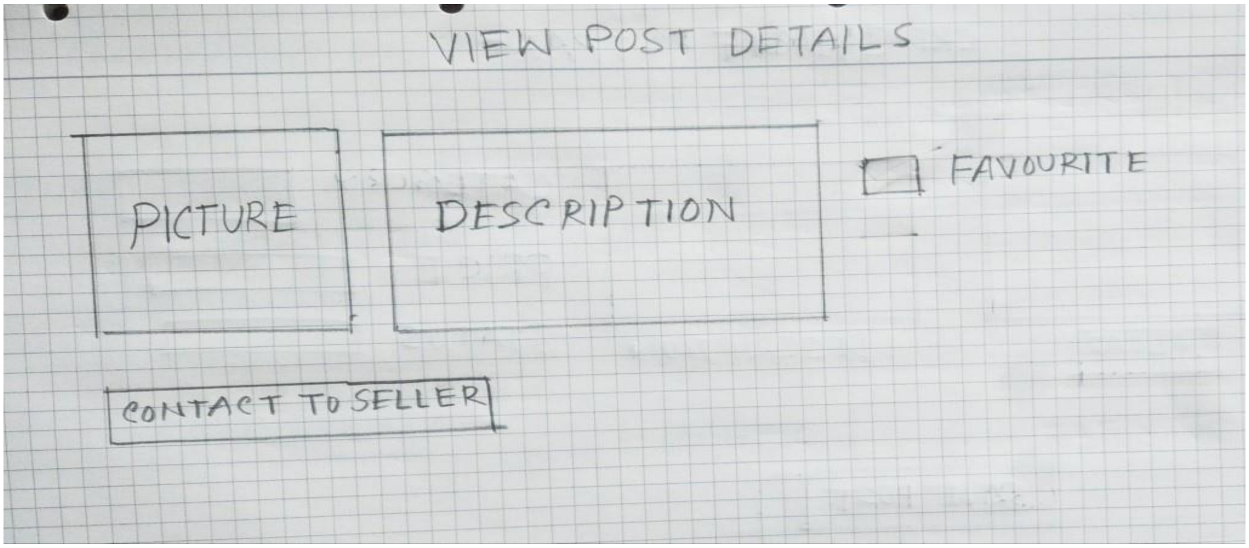


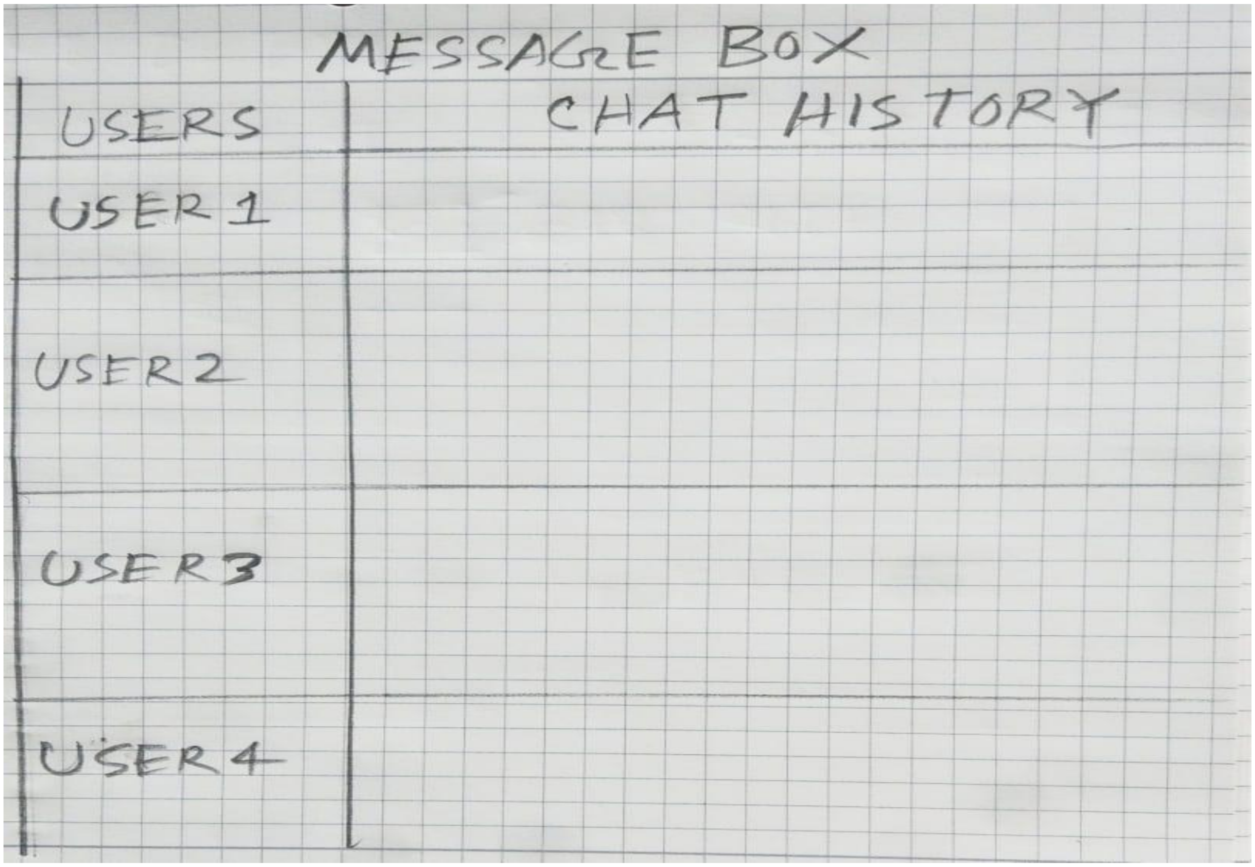


<u>Use Case 3: Buy Item</u>

In this scenario, user wants to buy a laptop for his educational purpose. So, User goes to Home page. User notices a tab called Market. So, for buying an item, user click on tab called **Market**. A new page appears. Then user clicks on **View Details** Button. A new page called 'View Post Details' appears. Then User clicks on **Contact to Seller** button. Then it takes user to **Message** page where user can send message to seller. Now user sends message to the seller about his/ her interest to buy that item.

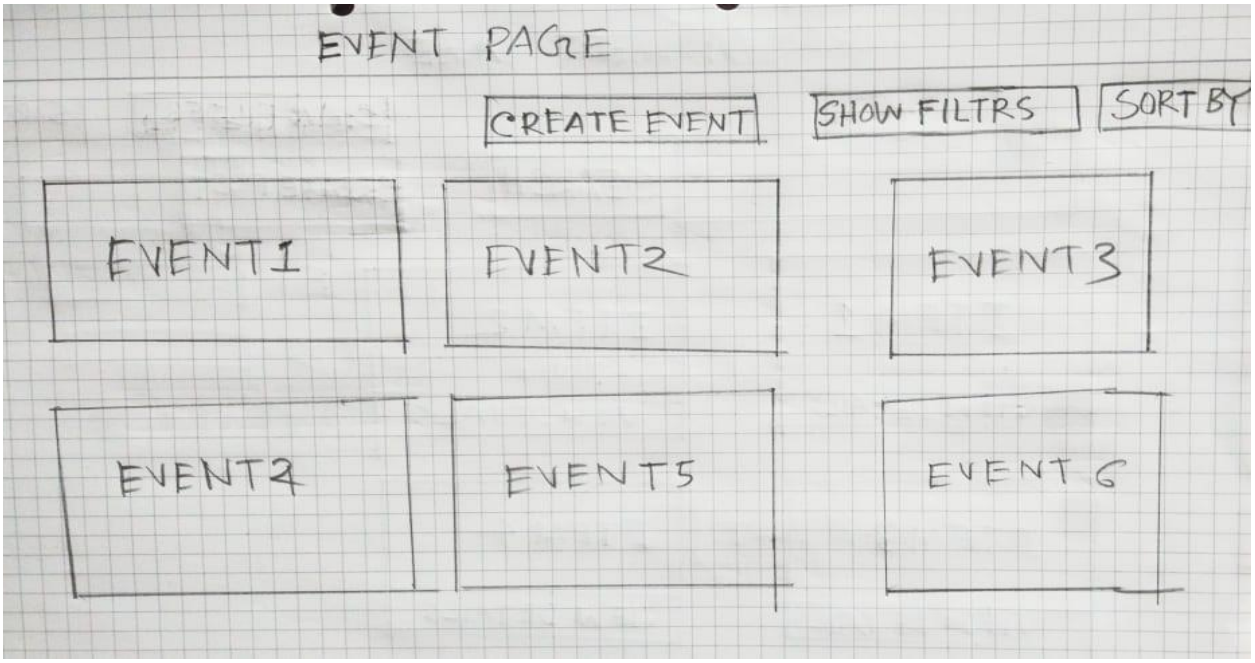



VIEW POST DETAILS
PICTURE
DESCRIPTION
FAVOURITE
CONTACT TO SELLER


MESSAGE BOX
USERS
CHAT HISTORY
USER 1
USER 2
USER 3
USER 4

User Case 4: Create/Update Event

In this scenario, user wants to create a online seminar in order to raise awareness regarding Covid-19 pandemic. So, user wants to create an event in FuldaMarkt website. User goes to Home Page and there user notices a tab called Event. In order to create or update an event, user clicks on tab called **Event**. A new page appears then. Then user clicks on **CREATE Event** Button. A new page regarding creating event appears. Then user gives all required information regarding the item and click on **SAVE EVENT** button.



4. High level Architecture, Database Organization

In this section, the database schema and its tables will be discussed. Additionally, initial details regarding media storage and the implementation of the search/filter for database items will be discussed as well.

Database Schema:

This is the initial, high-level description of the database schema and its tables. Additional tables might be created, and current tables may change based on future decisions and feature implementations.

The schema will include four tables detailed as follows:

- USER_TABLE: This is the table that contains all relevant user information:

Column	Description
User_id	Unique identifier. Will be used as primary key
Username	Username input by the user at first log-in
Type	Default is “student”. Can be changed by admin
Email	Email input by the user at first log-in
Status	Shows status of the user as either “active” or “suspended”

- EVENTS_TABLE: This is the table that holds data of the events section, where events and announcements by the faculty are posted.

Column	Description
Post_id	Unique identifier. Will be used as primary key
Title	Title of the post
Author_id	ID of the user who created the post
Timestamp	Date when the post was created
Picture	Picture object of the post
Text_Body	Body of text detailing the post

- MARKET_TABLE: This is the table that holds data of the market section, where items are posted for sale.

Column	Description
Post_id	Unique identifier. Will be used as primary key
Title	Title of the post
Author_id	ID of the user who created the post
Timestamp	Date when the post was created
Picture	Picture object of the post
Text_Body	Body of text detailing the post
Market_category	Category of the item posted
Price	Price of the item listed
Status	Status of item (ex. pending approval, available, sold)
Sold_to	ID of the user to which the item was sold

- USER_FAVORITES: This table will list which user marked which post from the market table as “favorite”. The primary key here is a combination of two foreign keys: user_id from the user_table and post_id from the market_table.

Column	Description
User_id	User who marked the item as “favorite”
Post_id	Post which was marked as “favorite”

Media Storage:

For media storage, database BLOBs will be used as it makes organization of data easier. The expected size of one picture is between 1MB to 3MBs. The expected size of the text body is 1000 characters.

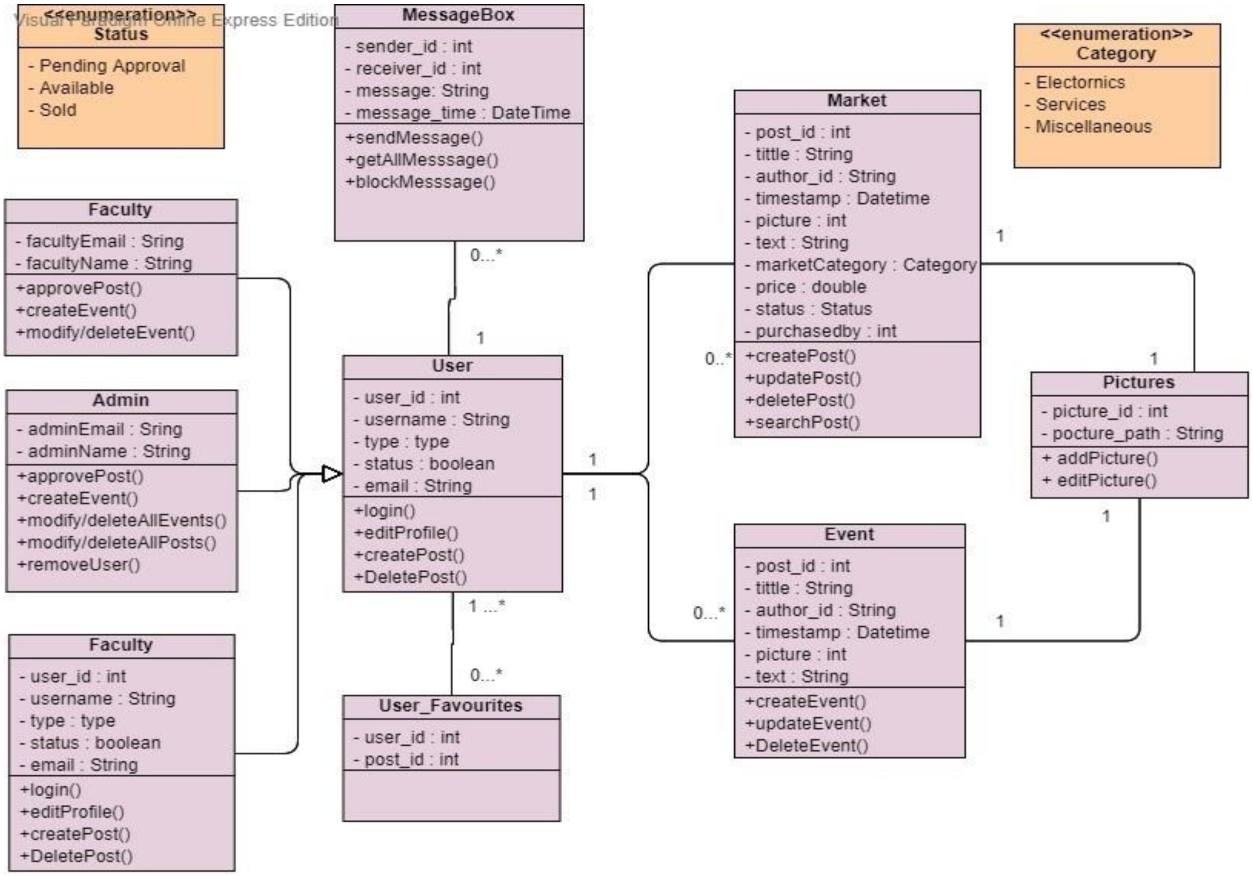
Search/Filter Implementation:

For the implementation of the search/filter: The Market categories will be available as a drop-down for the user to select one to be applied to the query. The terms that will be searched for are the “username” from the user table and the “title” from either the market table or the events table. In the case of searching for similar words, the LIKE operator will be used, which will look for similarities between user input and the columns “title” and “username”.

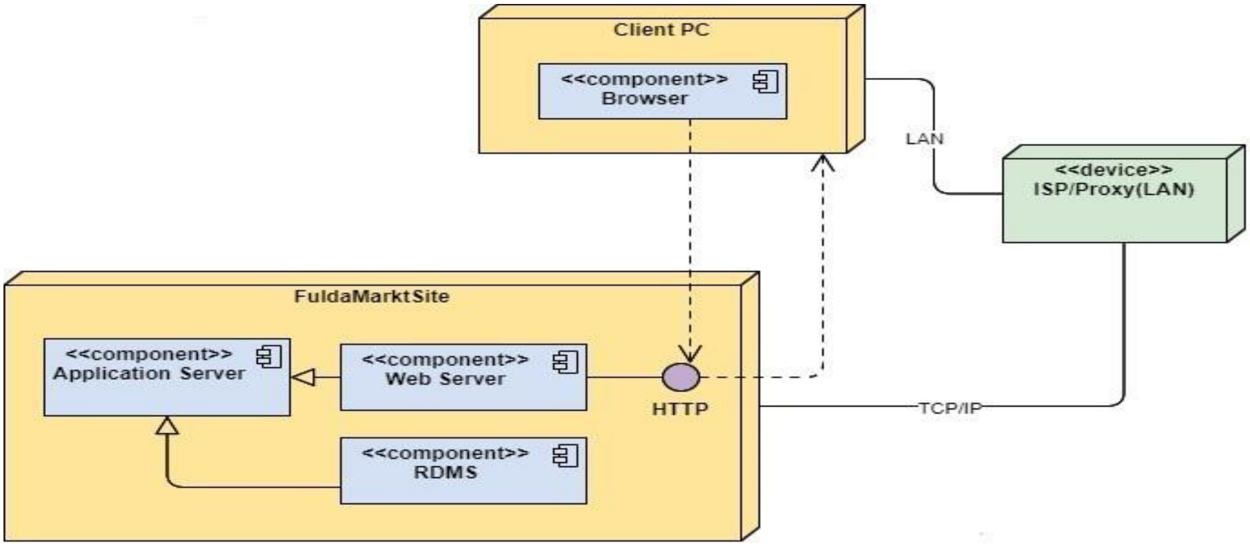
For sorting by date or price, the terms “timestamp” or “price” will be sorted either in ascending or descending order using the “ORDER BY” and “ASC” or “DESC” operations. For example, if the user selected to sort by pricing in ascending order, the query will use “ORDER BY price ASC”.

5. High-level UML class diagrams

UML Diagram:



Component and Deployment Diagram:



6. Key Risks

1. Lack of skills:

We as the students have software development experience on different software platform in our bachelor's programs and/or internships. While committing to a specific solution for this course project, we as a team may lack skills for individuals to cover all the aspects of the project.

Eventually team members will learn the skills, this learning may include programming languages, frameworks and other skills related to software development practices. We may have online sessions to cover up with these issues and team members who are having experience in the skills which we are going to use in our project can be taught by them.

2. Inaccurate estimations:

Project estimations can be foul-up when project tasks and scheduled release are not analyzed properly. Schedule risks mainly affect a project and may lead to project failure. The reasons behind these slips can be wrong deadline estimations, inappropriate tracking of resources like staff, systems and skills of individuals and failure to identify and develop complex functionalities of the project.

To mitigate these risks, we will use planning documents, such as specifications and project plans and perform a detailed task analysis of the work to be performed so that we can reduce the critical paths and dependencies available.

3. Technical risks:

Technical risks may lead to failure of functionality or performance. This may occur if we use deprecated frameworks/plugins or any dependencies which needs to be updated all the time to maintain the consistency of the project.

To mitigate this risk, we will be using the up-to-date frameworks, tools and plugins or any other dependencies with the official documentations available.

4. Teamwork risk:

Since team members are sharing most of the responsibilities to deliver outcomes, some individuals may need to do additional work to make up for those not fully contributing their share of efforts. This may lead some negative perception that can make the team less effective.

We should be clearly mentioning the task responsibilities and accountability for individual contributions to the group effort.

5. Legal/content risks

It is important in this world of advanced technologies to credit the author or the developer of certain APIs, plugin or software snippet and avoiding patent misuse. Failing to do so may lead to copyright allegations and claims.

We will be working with the services of opensource communities wherever possible and give proper credits and acknowledgements wherever needed.

7. Project Management

To develop a high-quality project and improve project consistency, we as a team need to coordinate through some project management tool to carry out the tasks with assigned persons and deadlines.

There are a lot of project management tools available in the market for project management which offers extensive features and customized functionalities. Initially we started with the **Microsoft Teams** as our centralized channel for every communication, files sharing, task monitoring, task assigning, and any other activities carried out within the team. For now it’s working smoothly for the whole team. As an alternate if we fall into some problems or have any limitations in this tool, We will be using **Trello** for our project as it provides a straightforward user experience for organizing and monitoring project tasks at a glance.

Hochschule Fulda
University of Applied Sciences

Global Distributed Software Development

GDSD Team Project WS2020

Global Software Development

Project

FuldaMarkt

Milestone 3 Part 1 review summary GDSD Team project 2020 team 2

19th January 2021

Team 2 (local)

Syeda Tasneem Rumy, 1189706 (Team Lead, GitHub Master & Backend Lead),

syeda-tasneem.rumy@informatik.hs-fulda.de

Syed Sumair, 1310771 (Frontend Lead)

Ahmed Abdullah, 1250378 (Backend Developer)

Revision History

Revision	Status	Date
Version 0.1	Submitted	19th January 2021

CEO & CTO: Prof. Dr. Rainer Todtenhöfer, Fulda University, Germany

Contents

Summary of feedback and tasks to do

Feedback from meeting (12.01.2021) regarding the project so far:

1. HTTPS Security should be implemented on the production server.
2. For final release admin approval of posts should be possible from the website.
3. So far, all requirements have been met and the customer is satisfied with the current progress.
4. Next focus will be on developing test cases for the software (Milestone 4).

List of tasks the team chose to focus on and implement for final delivery.

1. **View list of products pending approval:** Admin user should be able to view the list of products pending approval.
2. **Approve Posted Product:** Admin user should be able to view details of a product and publish it.
3. **Disapprove Posted Product:** Admin user should be able to view details of a product and disapprove it.
4. **Block Posted Product:** Admin user can block a posted product after being publishing on the site.
5. **Add product to Wishlist:** Users can mark a product to see in his Wishlist.
6. **Remove product from Wishlist:** Users can unmark a product from his Wishlist.
7. **View list of products in Wishlist:** Users can see their own list of items added to Wishlist.
8. **View Product listing:** Registered users should be able to view the list of products they posted.

List of final products P1 functions agreed at the meeting.

1. **Register**: Users (*Students, Faculty* and *Staffs* of HS Fulda) should be able to register to the website when providing unique email address with suffix “hs-fulda.de”.
2. **Login**: Registered users should be able to login using their correct login credentials unless they are blocked by an Admin user.
3. **Dashboard**: After Login, users should be able to view a Dashboard.
4. **Search product:** User can search products by product title and category.
5. **Filter product:** User can filter published items by category.
6. **Sort product:** User can sort published or search result items by published/posted date and price in ascending or descending order.
7. **View details of product:** User can view the details of a published product.
8. **View Related Products:** Users can view a list of all related published products in products details page from the same category.
9. **Post Product:** Registered users should be able to post products (physical products/services) for sale.
10. **View Product listing:** Registered users should be able to view the list of products they posted.
11. **Send Message:** Registered Users (Buyer/Seller) can send messages to each other.
12. **View Message listing:** Registered User (Buyer/Seller) can see the list of messages sent or received in the Inbox of Dashboard.
13. **Add product to Wishlist:** Users can mark a product to see in his Wishlist.
14. **Remove product from Wishlist:** Users can unmark a product from his Wishlist.
15. **View list of products in Wishlist:** Users can see their own list of items added to Wishlist.
16. **View list of products pending approval:** Admin user should be able to view the list of products pending approval.
17. **Approve Posted Product:** Admin user should be able to view details of a product and publish it.
18. **Disapprove Posted Product:** Admin user should be able to view details of a product and disapprove it.
19. **Block Posted Product:** Admin user can block a posted product after being publishing on the site.
20. **Block User:** Admin user can block any user who has posted inappropriate products on the site.

Hochschule Fulda
University of Applied Sciences

Global Distributed Software Development

GDSD Team Project WS2020

Global Software Development

Project

FuldaMarkt

Milestone 4

Beta Launch, QA and Usability Testing and Final Commitment for Product Features (P1 list)

2nd February 2021

Team 2 (local)

Syeda Tasneem Rumy, 1189706 (Team Lead, GitHub Master & Backend Lead),

syeda-tasneem.rumy@informatik.hs-fulda.de

Syed Sumair, 1310771 (Frontend Lead)

Ahmed Abdullah, 1250378 (Backend Developer)

Revision History

Revision	Status	Date
Version 0.1	Submitted	2nd February 2021

CEO & CTO: Prof. Dr. Rainer Todtenhöfer, Fulda University, Germany

Contents

1. Product Summary

- **Name of Product:** “Fulda Markt”
- **List of committed** functions
    - Register User
    - Login User
    - Market Search View
        - based on multiple categories filter
        - based on multiple order of sorting
    - Post a new item for sale
        - Input different fields for an item listing
        - Upload multiple images for an item
    - Item details
        - Item details with multiple images
        - Item details with related items(a/c to selected item category)
    - Messaging
        - Contact Seller via item
        - Contact Buyer via User’s dashboard messages section
    - User Wishlist
        - Add an item to wishlist
        - Remove an item from wishlist
    - Admin rights
        - Approve/decline any new or existing item from items list for market listing
        - Enable/disable new or existing user from users list
        - Change role of any user to make a normal user or give admin rights
    -
- **URL to access product:** https://ec2-34-226-123-197.compute-1.amazonaws.com/

2. Usability Test Plan

- **Test Objectives**:

The goal of this test is to measure effectiveness and efficiency of the search function, which will be the function used most frequently on the website. Additionally, in order to gauge user satisfaction and ensure that the search function meets the user's expectation of the product presented, a simple Lickert scale questionnaire will be provided and filled out by the tester.

- **Test Background and Setup:**

The test assumes that the server instance is running, and user has access to a web browser with internet access. The server uses the following technology stack:

- Server Host: Amazon AWS EC2 1vCPU 1 GB RAM

- Operating System: Ubuntu Server 20.04 LTS

- Server Database: MySQL v.14.14 Distribution 5.7.27

- Web Server: Apache v.2.4.29 (Ubuntu)

- Server-Side Language: PHP 7.2.24

The starting point for the task to be carried out by the tester is to open the website.

The URL for the website is : https://ec2-34-226-123-197.compute-1.amazonaws.com/

The intended users behind the test are users looking for a specific product (e.g. looking to buy new headphones). The intention behind the test is to measure user satisfaction after executing the task of searching for a desired item on the website.

- **Usability Task Description:**

The tester should follow the instructions below, then provide feedback to assess efficiency, effectiveness, and user satisfaction.

Instructions:

1. Open the website by copying the URL in the section "Test Background and Setup" into a browser.
2. Click on the "Market" item found in the user dashboard in the upper side of the website.
3. Type into the search bar what item to look for (e.g. iphone) and click on the magnifier icon.
4. Click on a desired item to view its details.

Feedback form to be filled out by the tester:

- Describe how would you measure the effectiveness of the search function in three lines or less:

The function was working as expected and should satisfy the expectations of our customer base.

- Describe how would you measure the efficiency of the search function in three lines or less:

The function presented was sufficiently efficient in providing fast results to the user's search criteria in a responsive way.

- Express your level of satisfaction with the presented user interface:

☐Highly Satisfied

☒Satisfied

☐Neutral

☐Dissatisfied

☐Highly dissatisfied

- Express your level of satisfaction with the responsiveness (i.e. load times, rendering):

☒Highly Satisfied

☐Satisfied

☐Neutral

☐Dissatisfied

☐Highly dissatisfied

- How likely are you to use the website again in the future?

☐Highly likely

☒Likely

☐Somewhat likely

☐Unlikely

☐Highly unlikely

3. QA Test Plan

Test Objectives

We'll run through the main functionality "Search" of the application "Fulda Markt", that needs to be tested as this function is used almost every time when the user uses this application. This test plan includes the trivial use cases performed by the users by setting different search parameters. This results the run rate and the pass rate of the search function.

HW and SW setup

A laptop and mobile with a stable internet connection and an internet browser are required to perform this test plan.

The starting point for the task to be carried out by the tester is to open the website.

The URL for the website is : https://ec2-34-226-123-197.compute-1.amazonaws.com/

Feature to be tested

Search functionality with different parameters

QA Test Plan

Test #	Title	Environment/Device	Description	Input	Expected/Correct Output	Test Result
1	Search by Category	Laptop/PC browser	Select specific category to see all listing of only respective category	Select "Electronics" by category dropdown	Page should show all the listing of Electronics category type	Pass
2	Search by Keyword	Laptop/PC browser	Enter an item keyword to search if it shows the list with relevant keyword	Enter "Chair" in the search text area	Page should show all the items which contains "Chair" in their title	Pass
3	Search by Sorting	Laptop/PC browser	Select any sort function to show listing in specific order	Select "Price Ascending" by sort dropdown	Page should show all the listing in order of price low to high	Pass
4	Search by Category	Mobile	Select specific category to see all listing of only respective category	Select "Services" by category dropdown	Page should show all the services	Pass
5	Search by Keyword	Mobile	Enter an item keyword to search if it shows the list with relevant keyword	Enter "iphone" in the search text area	Page should show all the items which contains "iphone" in their title	Pass
6	Search by Sorting	Mobile	Select any sort function to show listing in specific order	Select "Date Descending" by sort dropdown	Page should show all the listing in order of date new to old	Pass

Run rate is ratio between number test cases executed/total test cases of test specification. In our case, the test specification has total 6 test cases, and the tester has executed all 6 test cases, So the run rate is 6/6= 1.0(100%)

Pass rate is ratio between numbers test cases passed / test cases executed. In our case, there are 6 test cases executed, and the number of test cases that are passed are also 6, so the pass rate is 6/6= 1(100%)

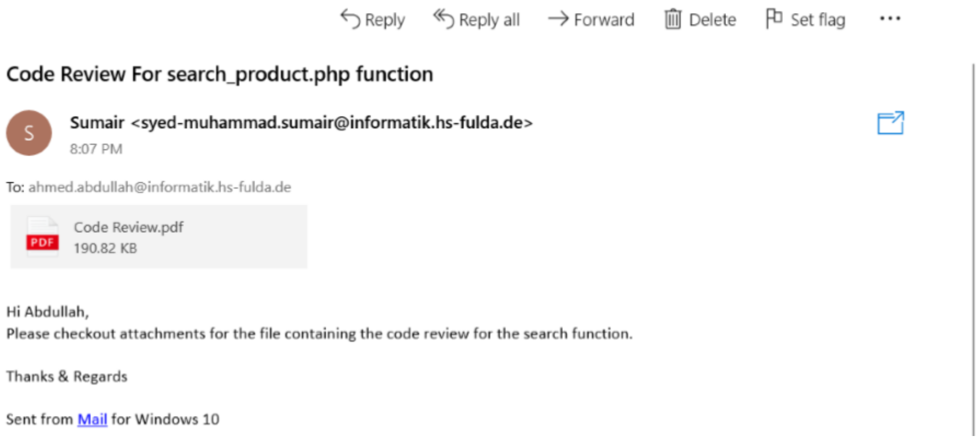
4. Code Review

Name of File reviewed: search_product.php

Author of File: Ahmed Abdullah

Function of File: Search functionality, Retrieve data from database based on query provided by frontend.

Email Screenshot:



Code Review For search_product.php function

S Sumair <syed-muhammad.sumair@informatik.hs-fulda.de>
8:07 PM

To: ahmed.abdullah@informatik.hs-fulda.de

PDF Code Review.pdf
190.82 KB

Hi Abdullah,
Please checkout attachments for the file containing the code review for the search function.

Thanks & Regards

Sent from Mail for Windows 10

Code Review:

```
<?php

        require 'project.php';

//This function should return an associative array of results.
        function get_product($dbi, $errors, $search_by_data)
        {
        $executed = false;//[Code Review by Syed Sumair] This variable should be named as
is_executed to indicate its usage properly
        $text_input = $search_by_data['text_input'];
        $category = $search_by_data['category'];
        $sort_by = $search_by_data['sort_by'];

        //Input validation to check if input is alphanumeric and less than 40 characters
        if(!empty($text_input)){
        if(!ctype_alnum(str_replace(' ','',$text_input)) || (strlen($text_input)) >= 40){
////[Code Review by Syed Sumair] such static values should be defined in a separate
constants file.
        return 1;
        }
        }

        //check database connection
        if(!$dbi ){
        die('Could not connect: ' . mysqli_error());
        $errors[] = "Search query failed: " . $ex->getMessage();
        return $executed;
        }

        mysqli_select_db($dbi, 'fuldamarkt_proddb');


        // check sorting to see whether to sort by ascending or descending order for
price and date sorting
        if(strpos($sort_by,"ascending") !== false){
        if(strpos($sort_by,"Price") !== false){
        $sort_by = "Price"; //formating
        }
        if(strpos($sort_by,"date") !== false){
        $sort_by = "date_inserted";
        }
        $sql_query = "SELECT * FROM MARKET_TABLE WHERE STATUS='available' AND TITLE LIKE
'%$text_input%' AND market_category LIKE '%$category%' ORDER BY $sort_by ASC;";
        }
        else if(strpos($sort_by,"descending") !== false){
        if(strpos($sort_by,"Price") !== false){
        $sort_by = "Price";
        }
        if(strpos($sort_by,"date") !== false){
        $sort_by = "date_inserted";
        }
        $sql_query = "SELECT * FROM MARKET_TABLE WHERE STATUS='available' AND TITLE LIKE
'%$text_input%' AND market_category LIKE '%$category%' ORDER BY $sort_by DESC;";
        }
        else{ //[Code Review by Syed Sumair] This statement could have been handled by
another [else if] condition so it doesn't execute all the time whenever above conditions
fails.
        $sql_query = "SELECT * FROM MARKET_TABLE WHERE STATUS='available' AND TITLE LIKE
'%$text_input%' AND market_category LIKE '%$category%' ORDER BY $sort_by ASC;";
        }



        try {
        $query_result = mysqli_query($dbi, $sql_query);
        // $executed = $query->execute(true); //[Code Review by Syed Sumair] There should
be no commented out code left. if necessary then mention its purpose or in which case
this needs to be uncommented
        } catch (\Exception $ex) {
        $errors[] = "Search query failed: " . $ex->getMessage();

        return $executed;
        }

        $query_result2 = $query_result->fetch_all(MYSQLI_ASSOC);
        return $query_result2;
        }
        $temp_name = 'market.twig';
        $title = 'Market Page';
        $body = 'Search for items';
        $errors = '';
        $message = '';
        $search_data = array();
        $search_data_pictures = array(); /*array containing multiple entries of
(directory:images), the directory is a string containg the location of a posts' images.
the ['images'] is a subarray containing names of pictures inside the directory */
//[Code Review by Syed Sumair] variable names should be simple and possibly short


        if ($request->getMethod() == "POST") {
        $search_by_data = array(
        'text_input' => trim($request->get('text_input')),
        'category' => trim($request->get('categories')),
        'sort_by' => trim($request->get('sort_by'))
        );



        $search_data = get_product($dbi, $errors, $search_by_data); //this is a 2d array
with index starting at 0 for first row


        if($search_data != 1){
        if(!empty($search_data)){ //check if there are results to the search query
        foreach($search_data as $val){//[Code Review by Syed Sumair] $val variable name
should be named by its type of object so it can be distinguished
        if(isset($val['picture'])){ //check if a post has a picture directory
        if(file_exists($_SERVER['DOCUMENT_ROOT'] . $val['picture'])){
        $files = array_diff( scandir( $_SERVER['DOCUMENT_ROOT'] . $val['picture']),
array(".", "..") ); //get list of image files found in the post's directory
        $temp_array = array('directory' => $val['picture'],'images' => $files);
        array_push($search_data_pictures, $temp_array);
        //[Code Review by Syed Sumair] for $files & $temp_array No object should exist
longer than necessary
        }
        }
        }
        unset($val);
        }
        else{
        $message = "No Data Found";
        }
        }
        else{
        $message = "Search input needs to be alphanumeric and less than 40 characters";
        }
        }

// necessary to forward to the twig template so that the "create post" button only
appears when a user is logged in
        $check_auth = $session->get('is_authenticated');


        /* Setting Template Variable, $page_data */
        if(!empty($search_data)){
        $page_data = ['title' => $title, 'body' => $body, 'errors'=> $errors, 'message'
=> $message, 'result' => $search_data,
        'search_data_pictures' => $search_data_pictures, 'Doc_root' =>
$_SERVER['DOCUMENT_ROOT'], 'check_auth' => $check_auth, 'session' => $session];
        }
        else{
        $page_data = ['title' => $title, 'body' => $body, 'errors'=> $errors, 'message'
=> $message, 'check_auth' => $check_auth, 'session' => $session];
        }
        try {
        echo $template->render($temp_name, $page_data);
        } catch (\Exception $ex) {
        $errors[] = "Template render error: " . $ex->getMessage();
        var_dump($errors);
        }
```

5. Self-check on Security Best Practices

Major Assets being protected

1. **Server Login credentials**
2. **Database**
3. **Deployed Application Code**

Major threats for each Asset above and how they are protected

1. **Server Login credentials**
    a. Private Key Compromise - Private key is only accessible to one administrator who manages the server.
    b. Unauthorized SSH Access - SSH credentials to server is only known to one administrator.
2. **Database**
    a. Privilege abuse – Database has only one user who administers the database deployed in server.
    b. SQL Injection – SQL injection has been mitigated by using data sanitizing filtering for all search, signup, login, product and message posting forms inputs.
    c. Exploitation of Vulnerable, Misconfigured Databases – Database can not be accessed from the web or without first logging in using the SSH credentials of the server.
    d. Denial of Service – Server is deployed using Amazon web services which prevents the Denial-of-Service vulnerability.
    e. Storage media exposure – No media stored in the server can be accessed from outside the application interface which is ensured by the Apache 2.0 configuration installed in the server.
3. **Deployed Application Code**
    a. Security Issues of Programming Language used – Usage of PHP 7 in server ensures better security enhancements, e.g. a filtered unserialize function or a set of functions cryptographically secure random numbers (random_bytes() and random_init()). Testing applications and finding hidden bugs becomes simpler due to scalar type and return type declarations.
    b. Directory traversal attack – It is prevented by avoiding passing of user-supplied input to filesystem APIs altogether.

Confirmation of User Information Security: All user supplied passwords are saved as encrypted values in the database.

Confirmation of Input data validation: FuldaMarkt codes validates,

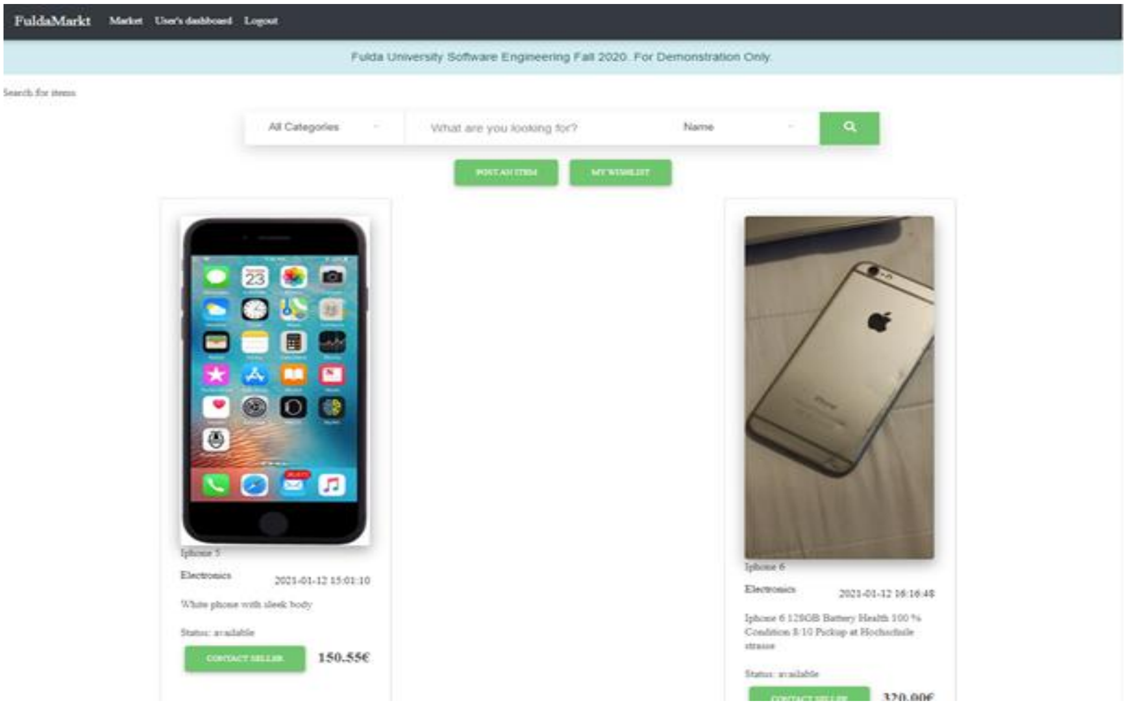
1. Signup form – Emails for suffix, character limits and data type.
2. Login form – Validity of User Email and Password.
3. Product posting – Character limits for each form text inputs.
4. Search bar – Search inputs up to 40 alphanumeric characters.

6. Adherence to original Non-functional specs

No.	Non-Functional Requirement	Status	Notes (If any)
1	Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0 (some may be provided in the class, some may be chosen by the student team but all tools and servers have to be approved by class CTO).	DONE	
2	Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers	DONE	
3	All or selected application functions must render well on mobile devices	ON TRACK	
4	Data shall be stored in the database on the team's deployment server.	DONE	
5	No more than 50 concurrent users shall be accessing the application at any time	DONE	
6	Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.	DONE	
7	The language used shall be English (no localization needed)	DONE	
8	Application shall be very easy to use and intuitive	DONE	
9	Application should follow established architecture patterns	DONE	
10	Application code and its repository shall be easy to inspect and maintain	DONE	
11	Google analytics shall be used (optional)	ISSUE	No, it has not been implemented. It was optional.
12	No e-mail clients shall be allowed. Interested users can only message to sellers via in-site messaging.	DONE	
13	Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.	DONE	
14	Site security: basic best practices shall be applied (as covered in the class) for main data items	DONE	
15	Media formats shall be standard as used in the market today	DONE	
16	Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development	DONE	
17	The application UI (WWW and mobile) shall prominently display the following exact text on all pages "Fulda University Software Engineering Fall 2020. For Demonstration Only" at the top of the WWW page. (Important so as to not confuse this with a real application).	ON TRACK	

3. Product Screen Shots

3.1 Home Page / Search Result Page



Fulda University Software Engineering Fall 2020. For Demonstration Only.

Search for items

Iphone 5

Electronics 2021-01-12 15:01:10

White phone with sleek body

Status: available

150.55€

Iphone 6

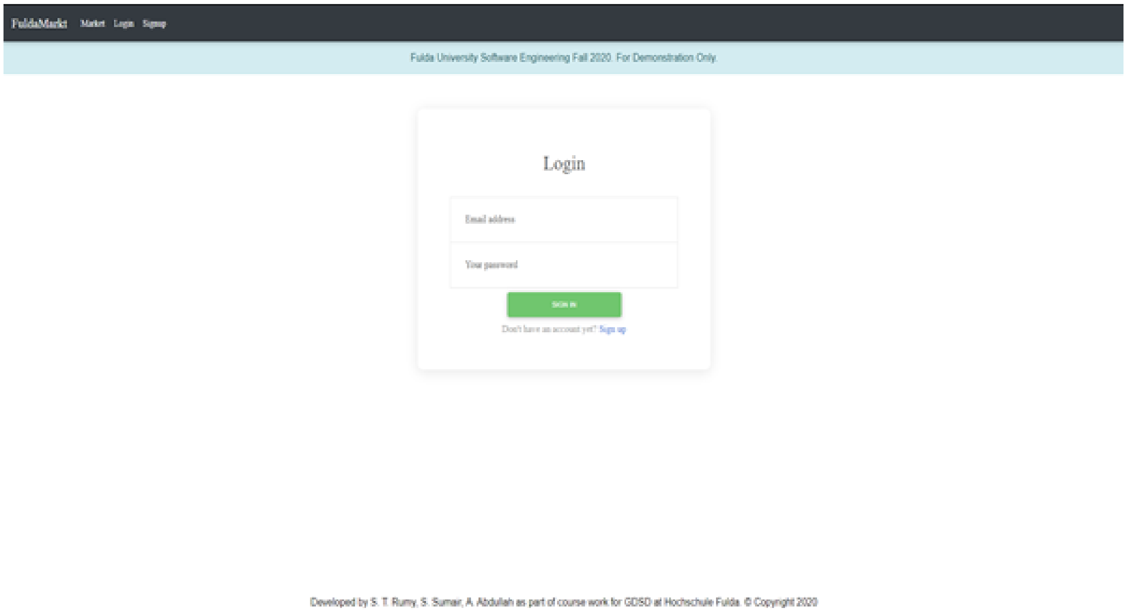
Electronics 2021-01-12 16:16:48

Iphone 6 128GB Battery Health 100 % Condition 8/10 Pickup at Hochschule strasse

Status: available

320.00€

3.2 Login Page

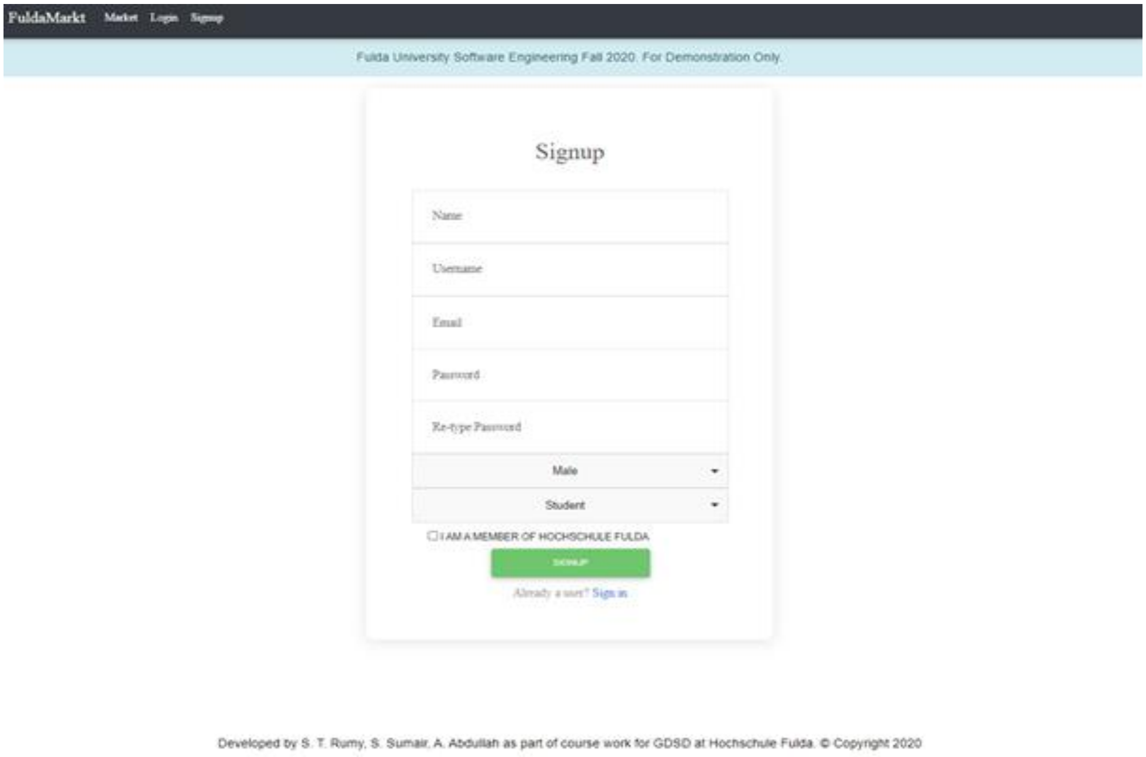


Fulda University Software Engineering Fall 2020. For Demonstration Only.

Don't have an account yet? Sign up

Developed by S. T. Rumy, S. Sumair, A. Abdullah as part of course work for GDSD at Hochschule Fulda. © Copyright 2020

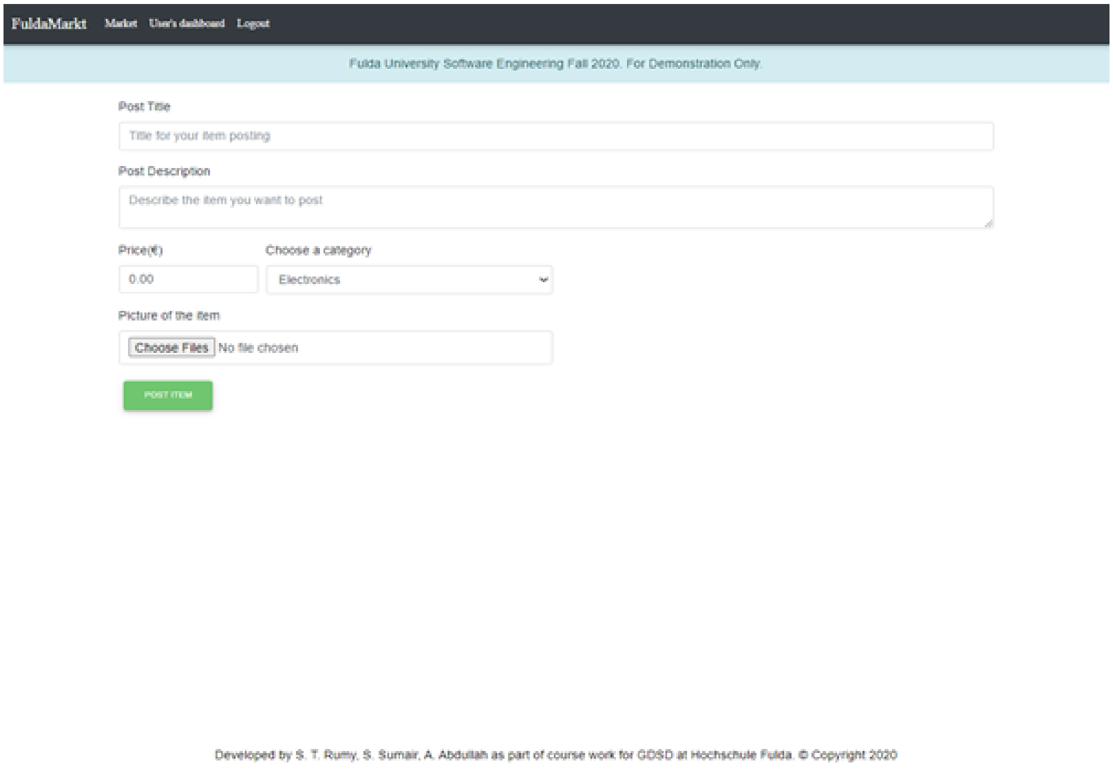
3.3 Signup Page



Fulda University Software Engineering Fall 2020. For Demonstration Only.

Developed by S. T. Rumy, S. Sumair, A. Abdullah as part of course work for GDSD at Hochschule Fulda. © Copyright 2020

3.4 Post Product Page



Fulda University Software Engineering Fall 2020. For Demonstration Only.

Post Title

Title for your item posting

Post Description

Describe the item you want to post

Price(€)

0.00

Choose a category

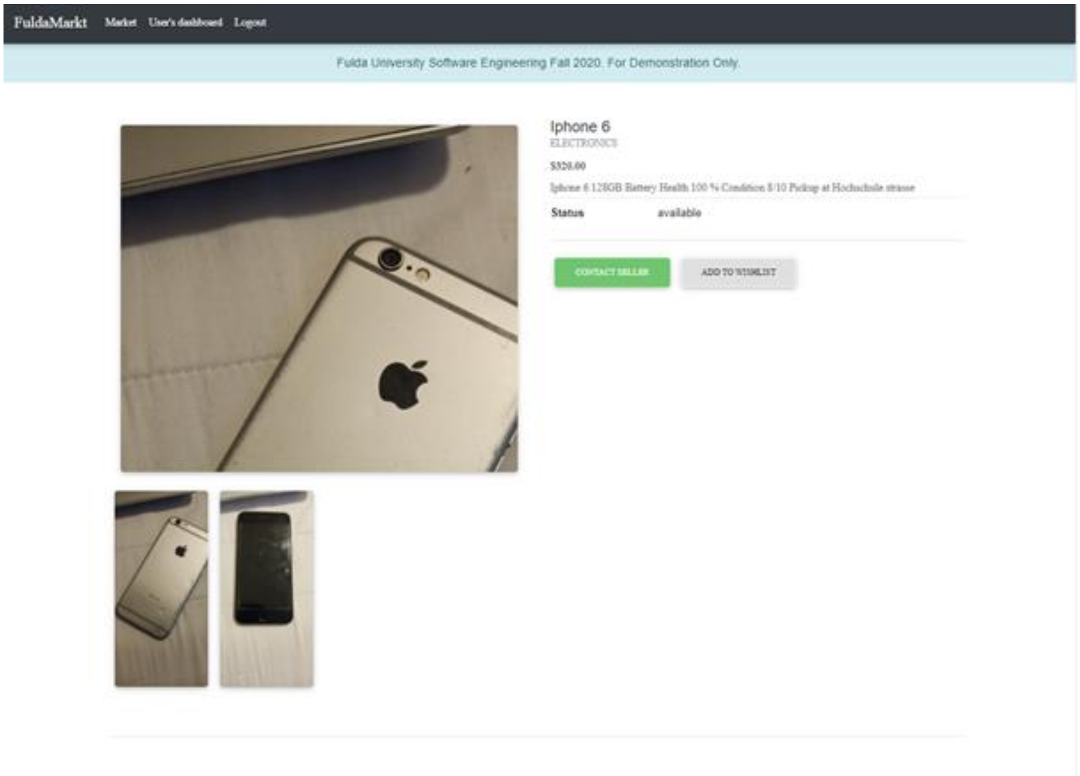
Electronics

Picture of the item

Choose Files No file chosen

Developed by S. T. Rumy, S. Sumair, A. Abdullah as part of course work for GDSD at Hochschule Fulda. © Copyright 2020

3.5 Product Details Page



Fulda University Software Engineering Fall 2020. For Demonstration Only.

Iphone 6

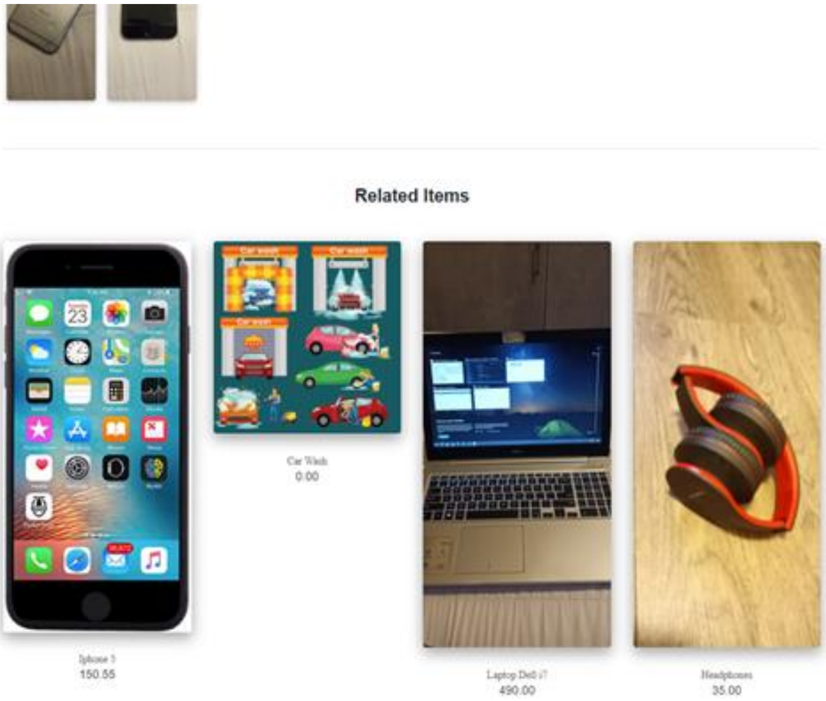
ELECTRONICS

$320.00

Iphone 6 128GB Battery Health 100 % Condition 8/10 Pickup at Hochschule strasse

Status available

3.6 Related Products View in Product Details Page



Related Items

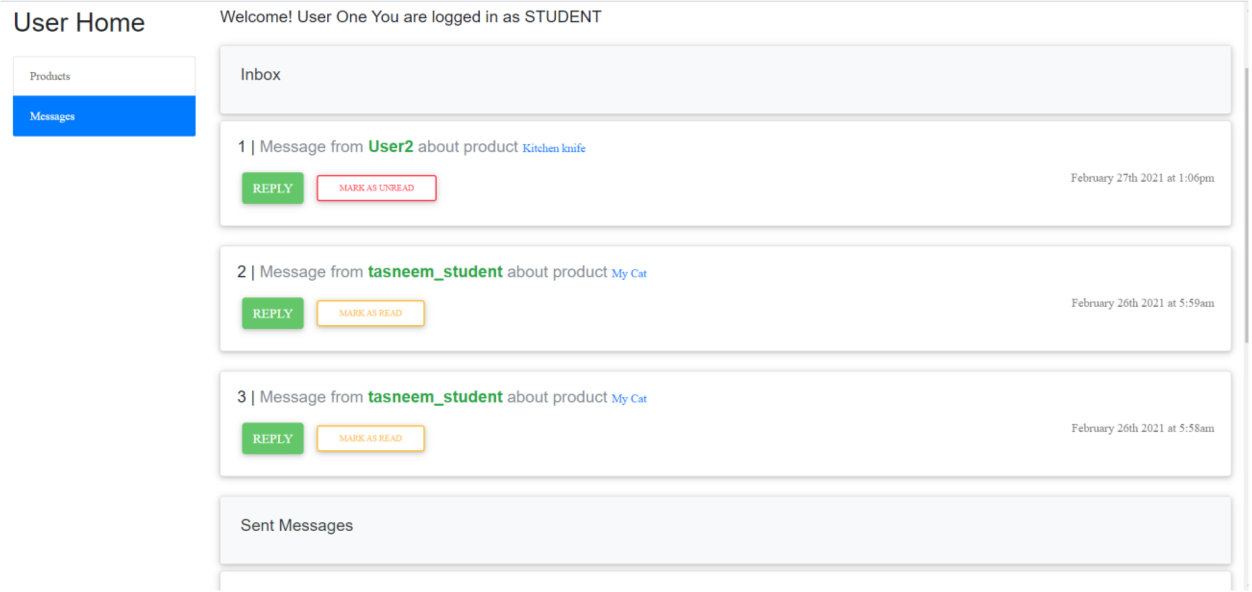
Iphone 5
150.55

Car Wash
0.00

Laptop Dell i7
490.00

Headphones
35.00

3.7 User Dashboard – Messages Inbox and Sent Messages List



Welcome! User One You are logged in as STUDENT

Inbox

1 | Message from User2 about product Kitchen knife

February 27th 2021 at 1:06pm

2 | Message from tasneem_student about product My Cat

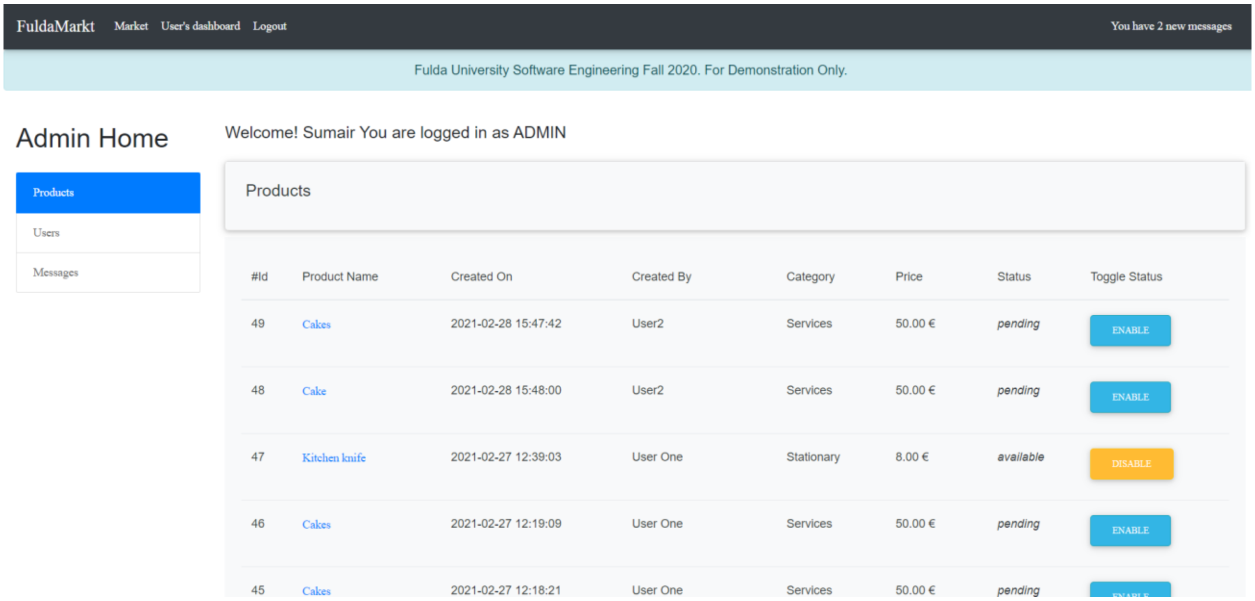
February 26th 2021 at 5:59am

3 | Message from tasneem_student about product My Cat

February 26th 2021 at 5:58am

Sent Messages

3.8 Admin Dashboard – Product List View



Fulda University Software Engineering Fall 2020. For Demonstration Only.

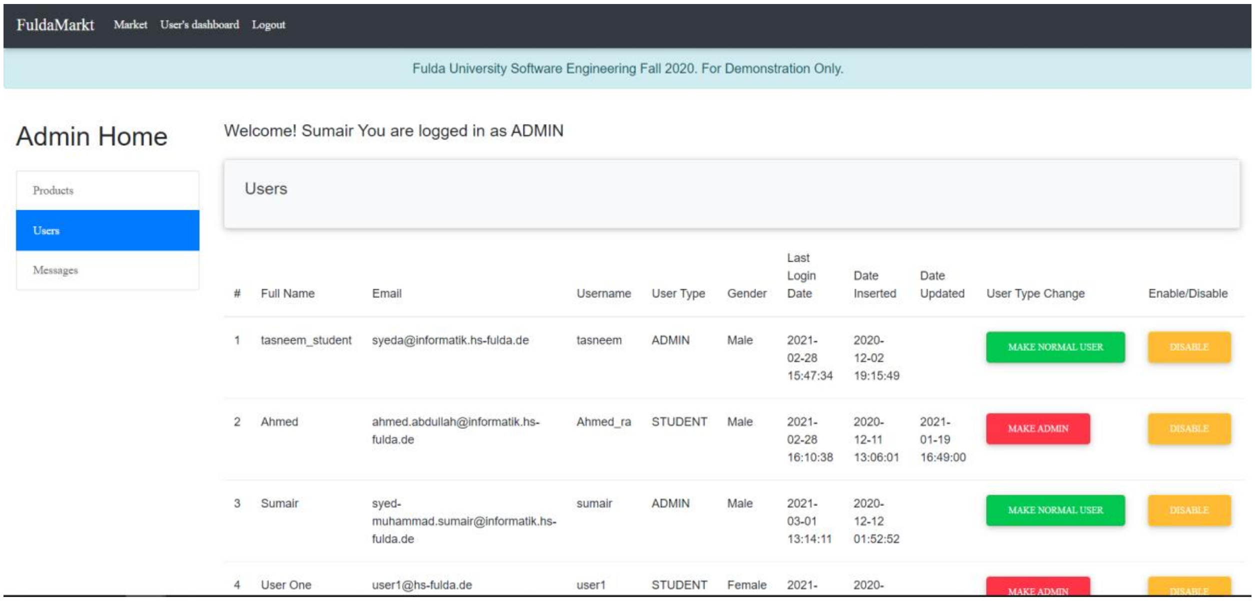
Admin Home

Welcome! Sumair You are logged in as ADMIN

Products

#Id	Product Name	Created On	Created By	Category	Price	Status	Toggle Status
49	Cakes	2021-02-28 15:47:42	User2	Services	50.00 €	*pending*	ENABLE
48	Cake	2021-02-28 15:48:00	User2	Services	50.00 €	*pending*	ENABLE
47	Kitchen knife	2021-02-27 12:39:03	User One	Stationary	8.00 €	*available*	DISABLE
46	Cakes	2021-02-27 12:19:09	User One	Services	50.00 €	*pending*	ENABLE
45	Cakes	2021-02-27 12:18:21	User One	Services	50.00 €	*pending*	ENABLE

3.9 Admin Dashboard – Users List View



Fulda University Software Engineering Fall 2020. For Demonstration Only.

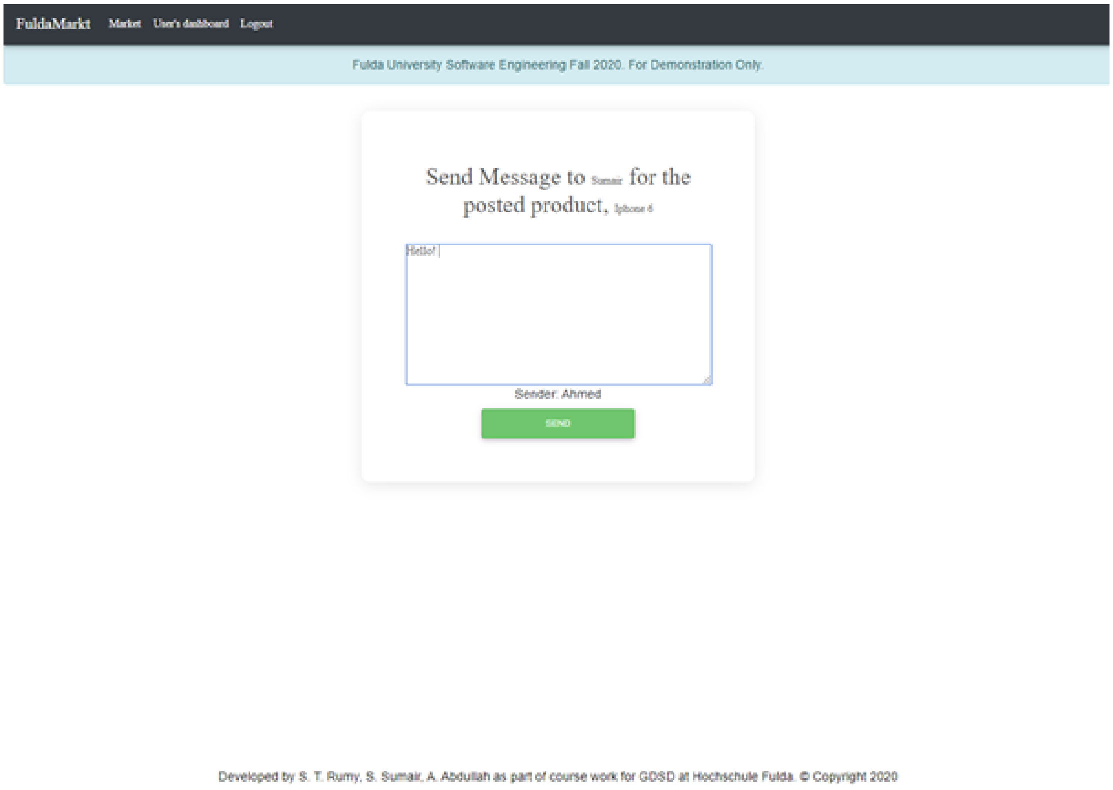
Admin Home

Welcome! Sumair You are logged in as ADMIN

Users

#	Full Name	Email	Username	User Type	Gender	Last Login Date	Date Inserted	Date Updated	User Type Change	Enable/Disable
1	tasneem_student	syeda@informatik.hs-fulda.de	tasneem	ADMIN	Male	2021-02-28 15:47:34	2020-12-02 19:15:49		MAKE NORMAL USER	DISABLE
2	Ahmed	ahmed.abdullah@informatik.hs-fulda.de	Ahmed_ra	STUDENT	Male	2021-02-28 16:10:38	2020-12-11 13:06:01	2021-01-19 16:49:00	MAKE ADMIN	DISABLE
3	Sumair	syed-muhammad.sumair@informatik.hs-fulda.de	sumair	ADMIN	Male	2021-03-01 13:14:11	2020-12-12 01:52:52		MAKE NORMAL USER	DISABLE
4	User One	user1@hs-fulda.de	user1	STUDENT	Female	2021-	2020-		MAKE ADMIN	DISABLE

3.10 Contact Seller Form page



Fulda University Software Engineering Fall 2020. For Demonstration Only.

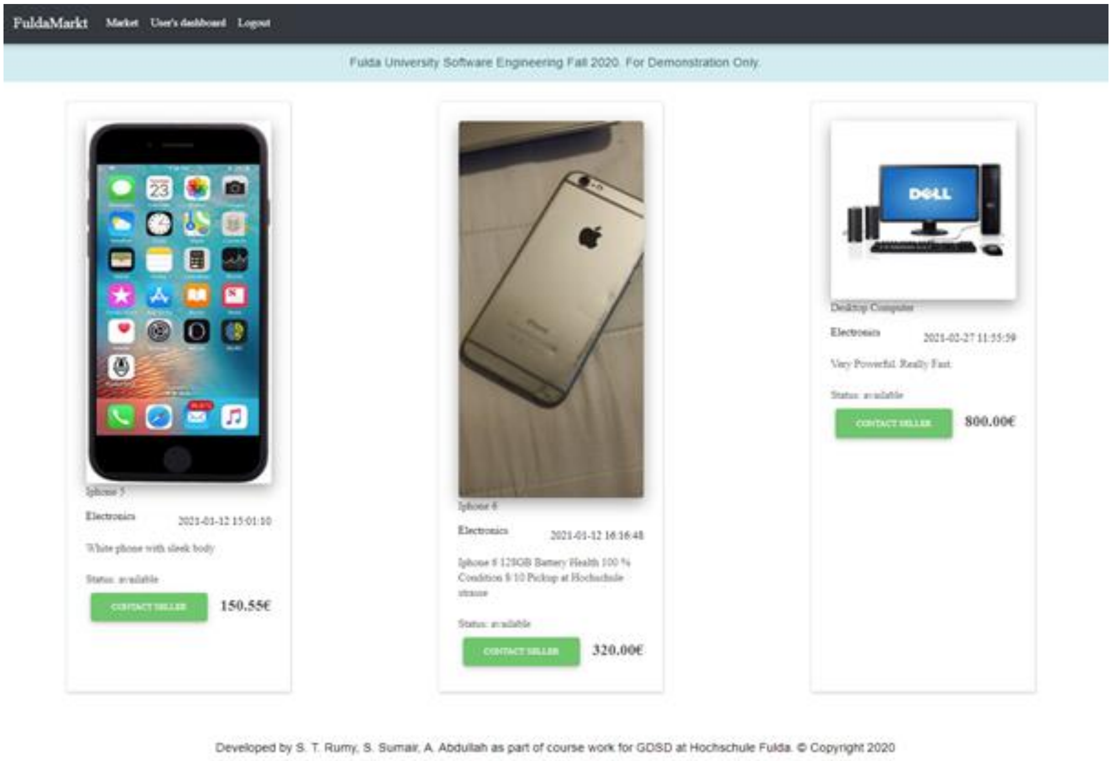
Send Message to Sumair for the posted product, Iphone 6

Hello!

Sender: Ahmed

Developed by S. T. Rumy, S. Sumair, A. Abdullah as part of course work for GDSD at Hochschule Fulda. © Copyright 2020

3.11 Wishlist View



Fulda University Software Engineering Fall 2020. For Demonstration Only.

Iphone 5

Electronics 2021-01-12 15:01:10

White phone with sleek body

Status: available

150.55€

Iphone 6

Electronics 2021-01-12 16:16:48

Iphone 6 128GB Battery Health 100 % Condition 8/10 Pickup at Hochschule strasse

Status: available

320.00€

Desktop Computer

Electronics 2021-02-27 11:55:59

Very Powerful. Really Fast.

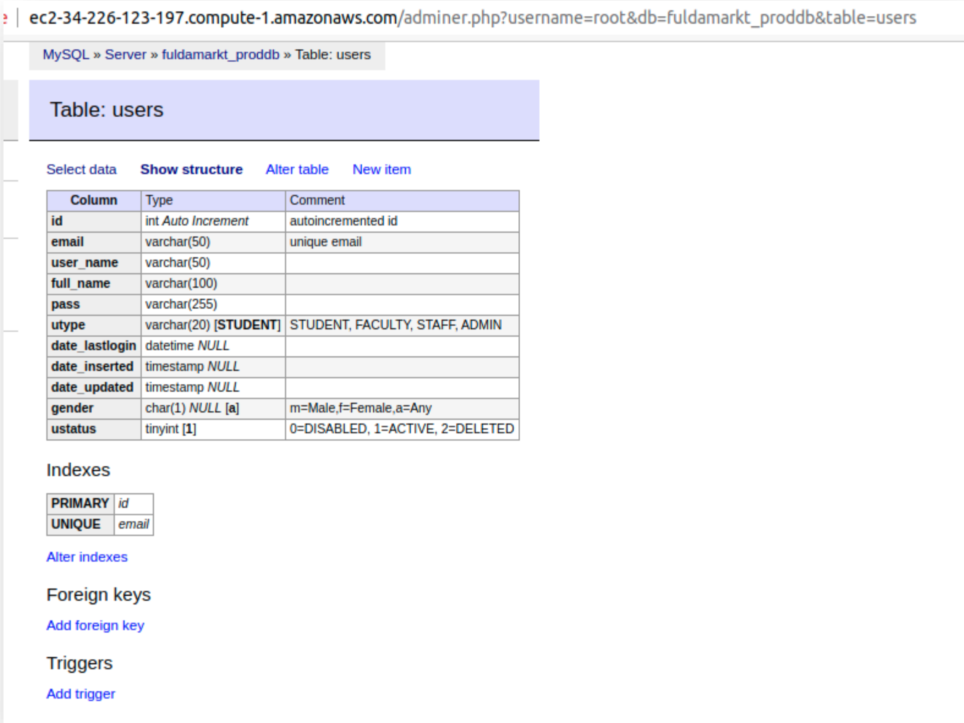
Status: available

800.00€

Developed by S. T. Rumy, S. Sumair, A. Abdullah as part of course work for GDSD at Hochschule Fulda. © Copyright 2020

4. Database Organization

4.1 ‘users’ table schema



ec2-34-226-123-197.compute-1.amazonaws.com/adminer.php?username=root&db=fuldamarkt_proddb&table=users

MySQL » Server » fuldamarkt_proddb » Table: users

Table: users

Select data | **Show structure** | Alter table | New item

Column	Type	Comment
id	int *Auto Increment*	autoincremented id
email	varchar(50)	unique email
user_name	varchar(50)	
full_name	varchar(100)	
pass	varchar(255)	
utype	varchar(20) **[STUDENT]**	STUDENT, FACULTY, STAFF, ADMIN
date_lastlogin	datetime *NULL*	
date_inserted	timestamp *NULL*	
date_updated	timestamp *NULL*	
gender	char(1) *NULL* **[a]**	m=Male,f=Female,a=Any
ustatus	tinyint **[1]**	0=DISABLED, 1=ACTIVE, 2=DELETED

Indexes

PRIMARY	*id*
UNIQUE	*email*

Alter indexes

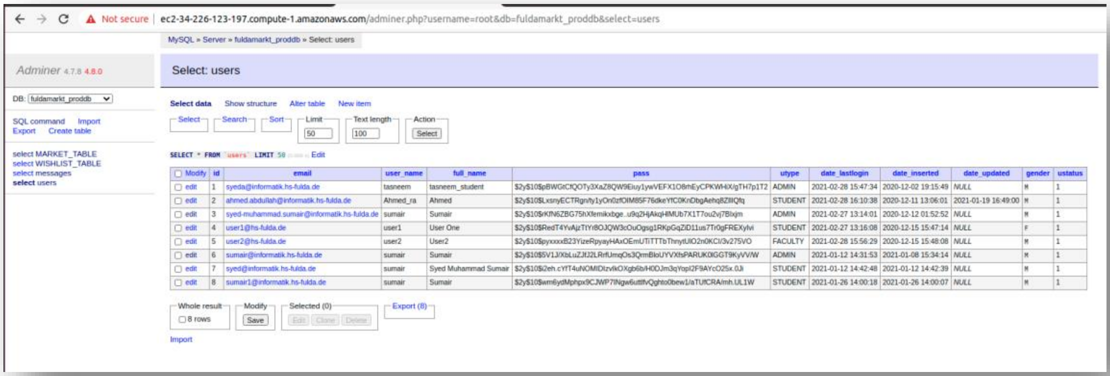
Foreign keys

Add foreign key

Triggers

Add trigger

4.2 ‘users’ table data



Not secure | ec2-34-226-123-197.compute-1.amazonaws.com/adminer.php?username=root&db=fuldamarkt_proddb&select=users

MySQL » Server » fuldamarkt_proddb » Select: users

Select: users

Select data | Show structure | Alter table | New item

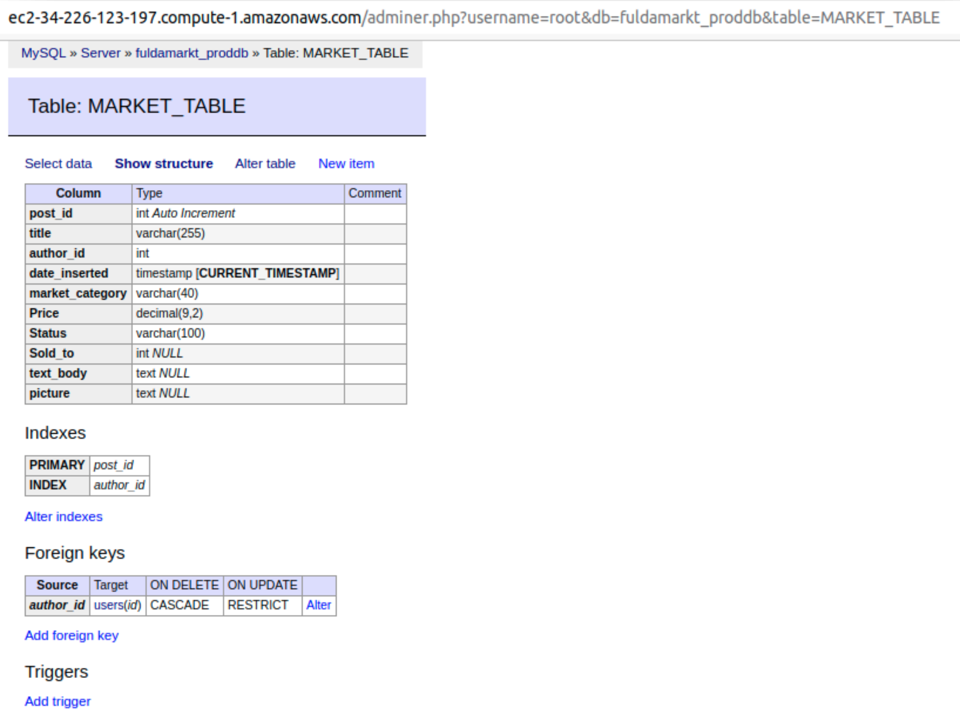
Select | Search | Sort | Limit: 50 | Text length: 100 | Action: Select

```
SELECT * FROM `users` LIMIT 50
```

Edit

□ Modify	id	email	user_name	full_name	pass	utype	date_lastlogin	date_inserted	date_updated	gender	ustatus
□ edit	1	syeda@informatik.hs-fulda.de	tasneem	tasneem_student	$2y$10$pBWGtCfQOTy3XaZ8QW9Eiuy1ywVEFX1O8rhEyCPKWHiX/gTH7p1T2	ADMIN	2021-02-28 15:47:34	2020-12-02 19:15:49	*NULL*	M	1
□ edit	2	ahmed.abdullah@informatik.hs-fulda.de	Ahmed_ra	Ahmed	$2y$10$LxsnyECTRgn/ty1yOn0zfOIM85F76dkeYfC0KnDbgAehq8ZlIQfq	STUDENT	2021-02-28 16:10:38	2020-12-11 13:06:01	2021-01-19 16:49:00	M	1
□ edit	3	syed-muhammad.sumair@informatik.hs-fulda.de	sumair	Sumair	$2y$10$rKfN6ZBG75hXfemikxbge..u9q2HjAkqHIMUb7X1T7ou2vj7Bixjm	ADMIN	2021-02-27 13:14:01	2020-12-12 01:52:52	*NULL*	M	1
□ edit	4	user1@hs-fulda.de	user1	User One	$2y$10$RedT4YvAjzTiYr8OJQW3cOuOgsg1RKpGqZiD11us7Tr0gFREXylvi	STUDENT	2021-02-27 13:16:08	2020-12-15 15:47:14	*NULL*	F	1
□ edit	5	user2@hs-fulda.de	user2	User2	$2y$10$pyxxxxB23YizeRpyayHAxOEmUTiTTTbThnytUlO2n0KCl/3v275VO	FACULTY	2021-02-28 15:56:29	2020-12-15 15:48:08	*NULL*	M	1
□ edit	6	sumair@informatik.hs-fulda.de	sumair	Sumair	$2y$10$5V1J/XbLuZJfJ2LRrfUmqOs3QrmBloUYVXfsPARUK0IGGT9KyVV/W	ADMIN	2021-01-12 14:31:53	2021-01-08 15:34:14	*NULL*	M	1
□ edit	7	syed@informatik.hs-fulda.de	sumair	Syed Muhammad Sumair	$2y$10$i2eh.cYfT4uNOMIDlzvlkOXgb6b/H0DJm3qYopI2F9AYcO25x.0Ji	STUDENT	2021-01-12 14:42:48	2021-01-12 14:42:39	*NULL*	M	1
□ edit	8	sumair1@informatik.hs-fulda.de	sumair	Sumair	$2y$10$wm6ydMphpx9CJWP7INgw6uttlfvQghto0bew1/aTUfCRA/mh.UL1W	STUDENT	2021-01-26 14:00:18	2021-01-26 14:00:07	*NULL*	M	1

4.3 ‘MARKET_TABLE’ table schema



ec2-34-226-123-197.compute-1.amazonaws.com/adminer.php?username=root&db=fuldamarkt_proddb&table=MARKET_TABLE

MySQL » Server » fuldamarkt_proddb » Table: MARKET_TABLE

Table: MARKET_TABLE

Column	Type	Comment
post_id	int *Auto Increment*	
title	varchar(255)	
author_id	int	
date_inserted	timestamp [**CURRENT_TIMESTAMP**]	
market_category	varchar(40)	
Price	decimal(9,2)	
Status	varchar(100)	
Sold_to	int *NULL*	
text_body	text *NULL*	
picture	text *NULL*	

Indexes

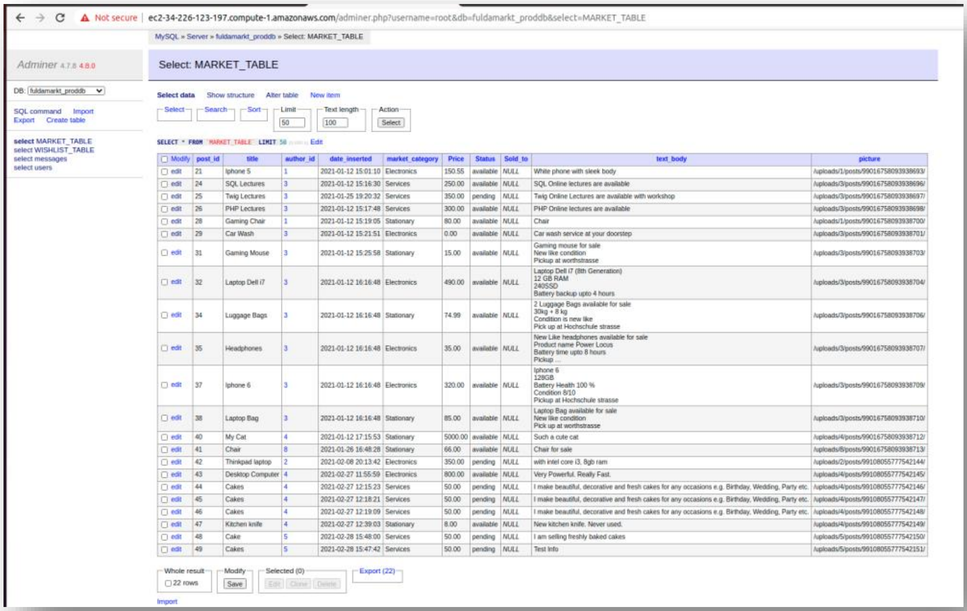
PRIMARY	*post_id*
INDEX	*author_id*

Foreign keys

Source	Target	ON DELETE	ON UPDATE	
author_id	users(*id*)	CASCADE	RESTRICT	Alter

Triggers

4.4 ‘MARKET_TABLE’ table data



Not secure | ec2-34-226-123-197.compute-1.amazonaws.com/adminer.php?username=root&db=fuldamarkt_proddb&select=MARKET_TABLE

MySQL » Server » fuldamarkt_proddb » Select: MARKET_TABLE

Select: MARKET_TABLE

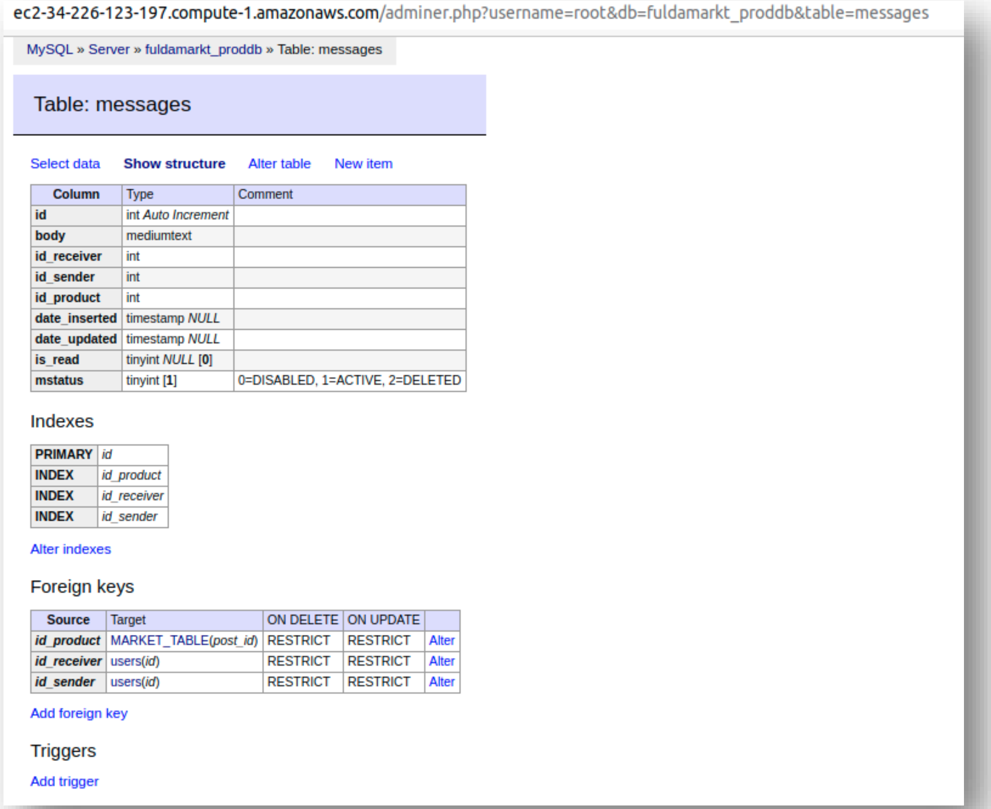
Limit: 50 Text length: 100 Action: Select

```
SELECT * FROM `MARKET_TABLE` LIMIT 50
```

□ Modify	post_id	title	author_id	date_inserted	market_category	Price	Status	Sold_to	text_body	picture
□ edit	21	Iphone 5	1	2021-01-12 15:01:10	Electronics	150.55	available	*NULL*	White phone with sleek body	/uploads/1/posts/99016758093938693/
□ edit	24	SQL Lectures	3	2021-01-12 15:16:30	Services	250.00	available	*NULL*	SQL Online lectures are available	/uploads/3/posts/99016758093938696/
□ edit	25	Twig Lectures	3	2021-01-25 19:20:32	Services	350.00	pending	*NULL*	Twig Online Lectures are available with workshop	/uploads/3/posts/99016758093938697/
□ edit	26	PHP Lectures	3	2021-01-12 15:17:48	Services	300.00	available	*NULL*	PHP Online lectures are available	/uploads/3/posts/99016758093938698/
□ edit	28	Gaming Chair	1	2021-01-12 15:19:05	Stationary	80.00	available	*NULL*	Chair	/uploads/1/posts/99016758093938700/
□ edit	29	Car Wash	3	2021-01-12 15:21:51	Electronics	0.00	available	*NULL*	Car wash service at your doorstep	/uploads/3/posts/99016758093938701/
□ edit	31	Gaming Mouse	3	2021-01-12 15:25:58	Stationary	15.00	available	*NULL*	Gaming mouse for sale New like condition Pickup at worthstrasse	/uploads/3/posts/99016758093938703/
□ edit	32	Laptop Dell i7	3	2021-01-12 16:16:48	Electronics	490.00	available	*NULL*	Laptop Dell i7 (8th Generation) 12 GB RAM 240SSD Battery backup upto 4 hours	/uploads/3/posts/99016758093938704/
□ edit	34	Luggage Bags	3	2021-01-12 16:16:48	Stationary	74.99	available	*NULL*	2 Luggage Bags available for sale 30kg + 8 kg Condition is new like Pick up at Hochschule strasse	/uploads/3/posts/99016758093938706/
□ edit	35	Headphones	3	2021-01-12 16:16:48	Electronics	35.00	available	*NULL*	New Like headphones available for sale Product name Power Locus Battery time upto 8 hours Pickup ...	/uploads/3/posts/99016758093938707/
□ edit	37	Iphone 6	3	2021-01-12 16:16:48	Electronics	320.00	available	*NULL*	Iphone 6 128GB Battery Health 100 % Condition 8/10 Pickup at Hochschule strasse	/uploads/3/posts/99016758093938709/
□ edit	38	Laptop Bag	3	2021-01-12 16:16:48	Stationary	85.00	available	*NULL*	Laptop Bag available for sale New like condition Pick up at worthstrasse	/uploads/3/posts/99016758093938710/
□ edit	40	My Cat	4	2021-01-12 17:15:53	Stationary	5000.00	available	*NULL*	Such a cute cat	/uploads/4/posts/99016758093938712/
□ edit	41	Chair	8	2021-01-26 16:48:28	Stationary	66.00	available	*NULL*	Chair for sale	/uploads/8/posts/99016758093938713/
□ edit	42	Thinkpad laptop	2	2021-02-08 20:13:42	Electronics	350.00	pending	*NULL*	with intel core i3, 8gb ram	/uploads/2/posts/99108055777542144/
□ edit	43	Desktop Computer	4	2021-02-27 11:55:59	Electronics	800.00	available	*NULL*	Very Powerful. Really Fast.	/uploads/4/posts/99108055777542145/
□ edit	44	Cakes	4	2021-02-27 12:15:23	Services	50.00	pending	*NULL*	I make beautiful, decorative and fresh cakes for any occasions e.g. Birthday, Wedding, Party etc.	/uploads/4/posts/99108055777542146/
□ edit	45	Cakes	4	2021-02-27 12:18:21	Services	50.00	pending	*NULL*	I make beautiful, decorative and fresh cakes for any occasions e.g. Birthday, Wedding, Party etc.	/uploads/4/posts/99108055777542147/
□ edit	46	Cakes	4	2021-02-27 12:19:09	Services	50.00	pending	*NULL*	I make beautiful, decorative and fresh cakes for any occasions e.g. Birthday, Wedding, Party etc.	/uploads/4/posts/99108055777542148/
□ edit	47	Kitchen knife	4	2021-02-27 12:39:03	Stationary	8.00	available	*NULL*	New kitchen knife. Never used.	/uploads/4/posts/99108055777542149/
□ edit	48	Cake	5	2021-02-28 15:48:00	Services	50.00	pending	*NULL*	I am selling freshly baked cakes	/uploads/5/posts/99108055777542150/
□ edit	49	Cakes	5	2021-02-28 15:47:42	Services	50.00	pending	*NULL*	Test Info	/uploads/5/posts/99108055777542151/

Whole result: □ 22 rows

4.5 ‘messages’ table schema



ec2-34-226-123-197.compute-1.amazonaws.com/adminer.php?username=root&db=fuldamarkt_proddb&table=messages

MySQL » Server » fuldamarkt_proddb » Table: messages

Table: messages

Column	Type	Comment
id	int *Auto Increment*	
body	mediumtext	
id_receiver	int	
id_sender	int	
id_product	int	
date_inserted	timestamp *NULL*	
date_updated	timestamp *NULL*	
is_read	tinyint *NULL* **[0]**	
mstatus	tinyint **[1]**	0=DISABLED, 1=ACTIVE, 2=DELETED

Indexes

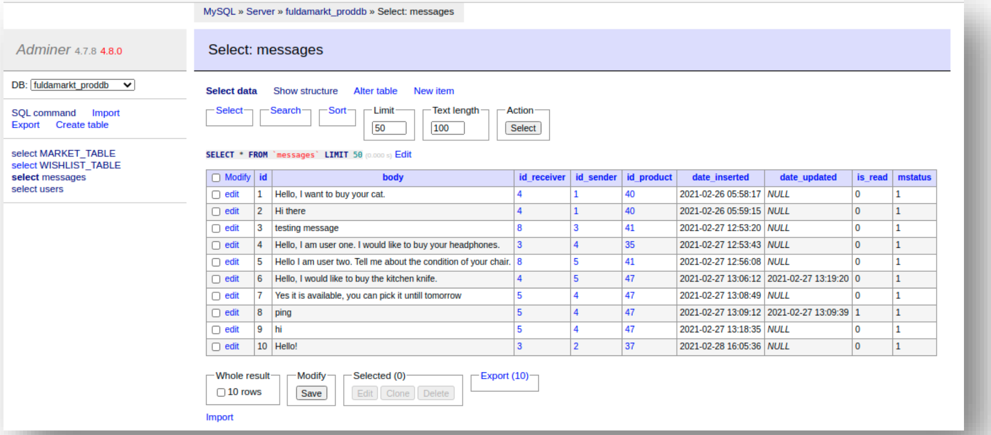
PRIMARY	*id*
INDEX	*id_product*
INDEX	*id_receiver*
INDEX	*id_sender*

Foreign keys

Source	Target	ON DELETE	ON UPDATE	
id_product	MARKET_TABLE(*post_id*)	RESTRICT	RESTRICT	Alter
id_receiver	users(*id*)	RESTRICT	RESTRICT	Alter
id_sender	users(*id*)	RESTRICT	RESTRICT	Alter

Triggers

4.6 ‘messages’ table data



MySQL » Server » fuldamarkt_proddb » Select: messages

Select: messages

Limit: 50

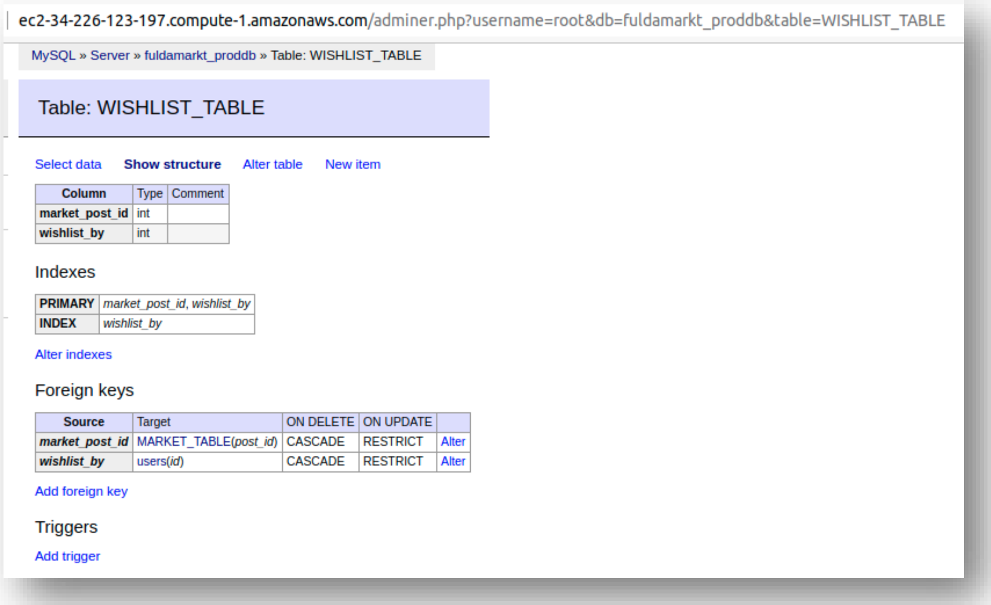
Text length: 100

```
SELECT * FROM `messages` LIMIT 50 (0.000 s) Edit
```

□ Modify	id	body	id_receiver	id_sender	id_product	date_inserted	date_updated	is_read	mstatus
□ edit	1	Hello, I want to buy your cat.	4	1	40	2021-02-26 05:58:17	*NULL*	0	1
□ edit	2	Hi there	4	1	40	2021-02-26 05:59:15	*NULL*	0	1
□ edit	3	testing message	8	3	41	2021-02-27 12:53:20	*NULL*	0	1
□ edit	4	Hello, I am user one. I would like to buy your headphones.	3	4	35	2021-02-27 12:53:43	*NULL*	0	1
□ edit	5	Hello I am user two. Tell me about the condition of your chair.	8	5	41	2021-02-27 12:56:08	*NULL*	0	1
□ edit	6	Hello, I would like to buy the kitchen knife.	4	5	47	2021-02-27 13:06:12	2021-02-27 13:19:20	0	1
□ edit	7	Yes it is available, you can pick it untill tomorrow	5	4	47	2021-02-27 13:08:49	*NULL*	0	1
□ edit	8	ping	5	4	47	2021-02-27 13:09:12	2021-02-27 13:09:39	1	1
□ edit	9	hi	5	4	47	2021-02-27 13:18:35	*NULL*	0	1
□ edit	10	Hello!	3	2	37	2021-02-28 16:05:36	*NULL*	0	1

Whole result: □ 10 rows

4.7 ‘WISHLIST_TABLE’ table schema



ec2-34-226-123-197.compute-1.amazonaws.com/adminer.php?username=root&db=fuldamarkt_proddb&table=WISHLIST_TABLE

MySQL » Server » fuldamarkt_proddb » Table: WISHLIST_TABLE

Table: WISHLIST_TABLE

Column	Type	Comment
market_post_id	int	
wishlist_by	int	

Indexes

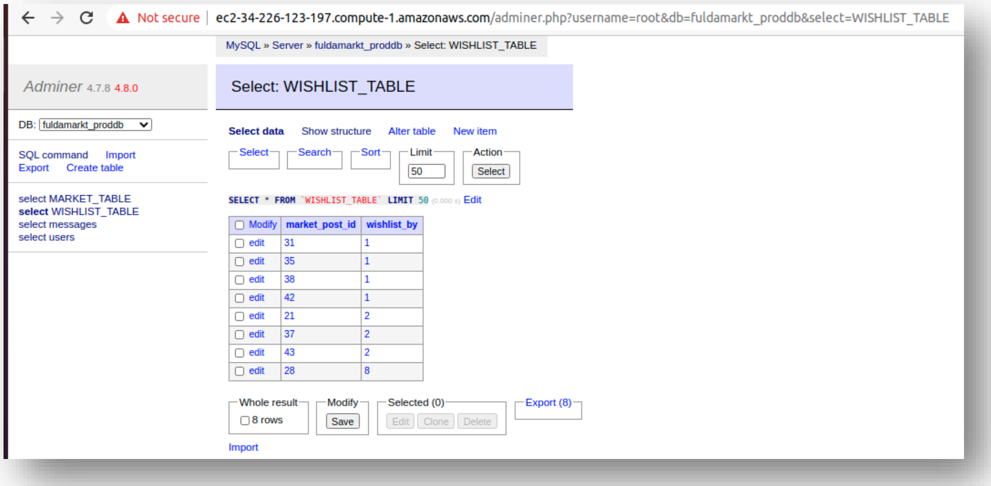
PRIMARY	*market_post_id, wishlist_by*
INDEX	*wishlist_by*

Foreign keys

Source	Target	ON DELETE	ON UPDATE	
market_post_id	MARKET_TABLE(*post_id*)	CASCADE	RESTRICT	Alter
wishlist_by	users(*id*)	CASCADE	RESTRICT	Alter

Triggers

4.8 ‘WISHLIST_TABLE’ table data



ec2-34-226-123-197.compute-1.amazonaws.com/adminer.php?username=root&db=fuldamarkt_proddb&select=WISHLIST_TABLE

MySQL » Server » fuldamarkt_proddb » Select: WISHLIST_TABLE

Select: WISHLIST_TABLE

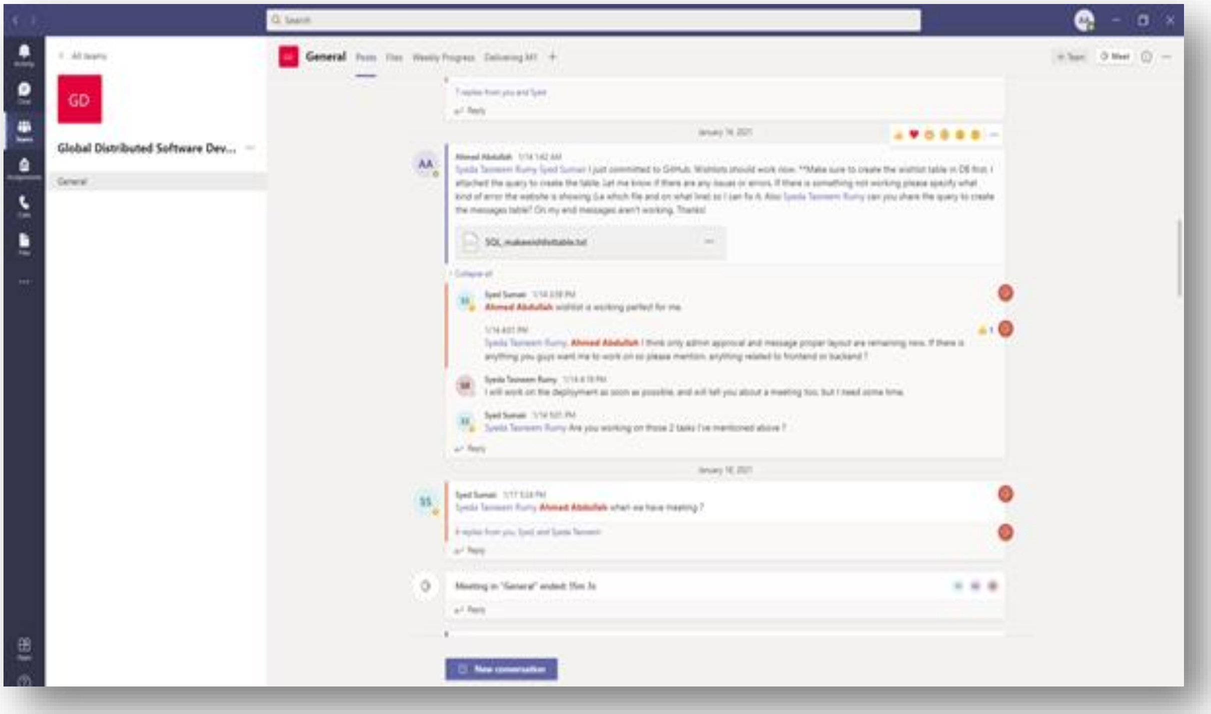
```
SELECT * FROM `WISHLIST_TABLE` LIMIT 50 (0.000 s) Edit
```

Modify	market_post_id	wishlist_by
edit	31	1
edit	35	1
edit	38	1
edit	42	1
edit	21	2
edit	37	2
edit	43	2
edit	28	8

5. Project management

Platform used: **Microsoft Teams.**

5.1 Discussions:



7 replies from you and Syed

Reply

January 14, 2021

Ahmed Abdullah 1/14 1:42 AM
Syeda Tasneem Rumy Syed Sumair I just committed to GitHub. Wishlists should work now. **Make sure to create the wishlist table in DB first. I attached the query to create the table. Let me know if there are any issues or errors. If there is something not working please specify what kind of error the website is showing (i.e which file and on what line) so I can fix it. Also Syeda Tasneem Rumy can you share the query to create the messages table? On my end messages aren't working. Thanks!

SQL_makewishlisttable.txt

Syed Sumair 1/14 3:59 PM
Ahmed Abdullah wishlist is working perfect for me.

1/14 4:01 PM
Syeda Tasneem Rumy, Ahmed Abdullah I think only admin approval and message proper layout are remaining now. If there is anything you guys want me to work on so please mention. anything related to frontend or backend ?

Syeda Tasneem Rumy 1/14 4:19 PM
I will work on the deployment as soon as possible, and will tell you about a meeting too. but I need some time.

Syed Sumair 1/14 5:01 PM
Syeda Tasneem Rumy Are you working on those 2 tasks I've mentioned above ?

Reply

January 17, 2021

Syed Sumair 1/17 5:24 PM
Syeda Tasneem Rumy Ahmed Abdullah when we have meeting ?

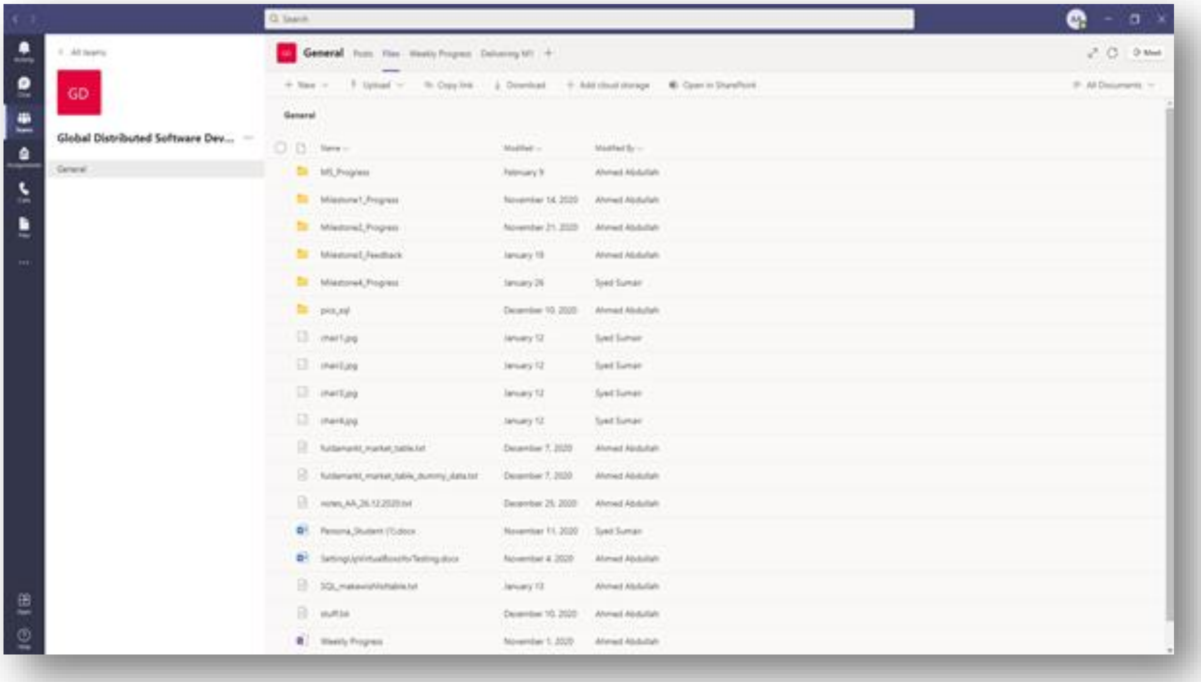
4 replies from you, Syed and Syeda Tasneem

Reply

Meeting in "General" ended: 15m 3s

Reply

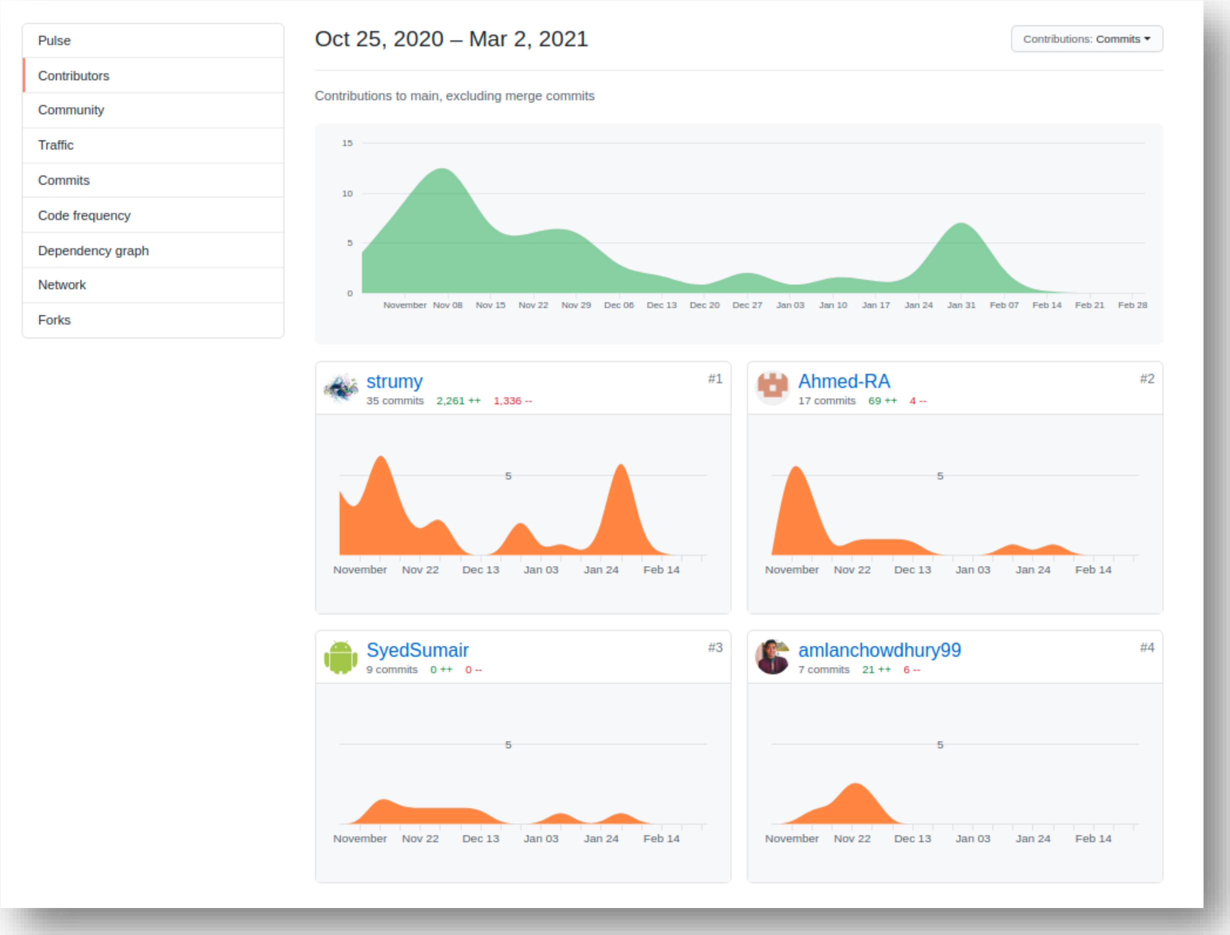
5.2 Document contribution and work:



General

Name	Modified	Modified By
M5_Progress	February 9	Ahmed Abdullah
Milestone1_Progress	November 14, 2020	Ahmed Abdullah
Milestone2_Progress	November 21, 2020	Ahmed Abdullah
Milestone3_Feedback	January 19	Ahmed Abdullah
Milestone4_Progress	January 26	Syed Sumair
pics_sql	December 10, 2020	Ahmed Abdullah
chart1.jpg	January 12	Syed Sumair
chart2.jpg	January 12	Syed Sumair
chart3.jpg	January 12	Syed Sumair
chart4.jpg	January 12	Syed Sumair
fuldamarkt_market_table.txt	December 7, 2020	Ahmed Abdullah
fuldamarkt_market_table_dummy_data.txt	December 7, 2020	Ahmed Abdullah
notes_AA_26.12.2020.txt	December 25, 2020	Ahmed Abdullah
Persona_Student (1).docx	November 11, 2020	Syed Sumair
SettingUpVirtualBoxforTesting.docx	November 4, 2020	Ahmed Abdullah
SQL_makewishlisttable.txt	January 13	Ahmed Abdullah
stuff.txt	December 10, 2020	Ahmed Abdullah
Weekly Progress	November 1, 2020	Ahmed Abdullah

5.3 GitHub Statistics:



Oct 25, 2020 – Mar 2, 2021

Contributions to main, excluding merge commits

strumy #1
35 commits 2,261 ++ 1,336 --

Ahmed-RA #2
17 commits 69 ++ 4 --

SyedSumair #3
9 commits 0 ++ 0 --

amlanchowdhury99 #4
7 commits 21 ++ 6 --

6. Team member self-assessment and contributions

6.1 Syeda Tasneem Rumy (Team Lead, Github Master, Lead Developer)

a) Contributions to team project and teamwork (technical and any other)

1. Collaborating with team and class CTO as team lead of group 2,
2. Managed git repository as GitHub master
3. Played role of lead developer by setting up and maintaining server in Amazon web services and created project setup of the FuldaMarkt
4. Developed the Login, Registration, User Dashboard, Admin Dashboard, Admin control over Product and User enable/disable, and Messaging feature.

b) Number of submissions to GitHub team Dev. branch: 40

c) Main challenges encountered in team project:

I was a member of team 2 in our course GDSD. Within our 3-member team we had different or inadequate skills in the programming language HTML/CSS as well as PHP which was the main tool of our application development. Although I had experience developing web applications using HTML/CSS and JavaScript in the frontend and PHP/MySQL in the backend it was still challenging for me to develop the software using raw PHP without any framework. Moreover, it was challenging within the team members when work ethics conflicted due to someone modifying someone else’s code. The inconsistent understanding of product quality or features requirement or project framework design as a team was also a challenge that we had to cope with.

d) What could be done better next time based on what was learned in the class about SE management and processes.

Requirements of the projects could be discussed more thoroughly within team to have a clear picture of what was expected from the software or the final product so that the task planning could be done more effectively.

6.2 Ahmed Abdullah (Backend Developer)

a) Contributions to team project and teamwork

1. Taken the role of backend development.
2. Developed features: item search (including ability to filter by category and sort by date, name, or price), item post, add or remove item from user wishlist.

b) Number of commits to GitHub team Dev. branch: 39

c) Main challenges:

Adapting to a team development environment, on some occasions did not adequately communicate intentions beforehand which led to work delays. Lack of knowledge about how to implement some assigned features (e.g., implementation of image uploads and displays).

d) Improvements going forward:

Better communication with team members to ensure no misunderstandings occur that would possibly cause work delays. Since familiarity with the used technologies has been established, it is now easier to estimate time needed to finish tasks.

6.3 Syed Muhammad Sumair (Frontend Lead)

a) Contributions to team project and teamwork (technical and any other)

1. Frontend Lead
2. Proposed initial layout with material controls and layouts for base design which is followed all over the application.
3. Handling responsiveness in overall layouts.
4. Worked on Item details and similar items module

b) Number of submissions to GitHub team Dev. branch: 23

c) Main challenges encountered in team project:

After a lot of struggle with a very new language and environment to me, things were very complicated and messed up in the start due to the lack of knowledge in web applications, but this project marks my initial steps towards the web application development. It was also a challenging task to complete the project on time with consecutive milestones and weekly tasks which led us to gain time management as well as team work skills

d) What could be done better next time based on what was learned in the class about SE management and processes.

It should be identified by whole team in the beginning of the project about the business needs and requirements for the project and everyone should be on same page regarding project knowledge as it becomes more complex if there is misunderstanding in the initial phase.