

Milestone 5

By Team 2

Global Distributed Software Development,

Winter Semester 23/24,

Fulda University of Applied Sciences,

Department of Applied Computer Science.



{ “project_name” : “MediaMajesty” }

Team 2

Team Lead:	Abbas Abbas; Email: abbas.abbas@informatik.hs-fulda.de
GitHub Master:	Shinu Donney
Frontend Lead:	Achraf Boudabous
Backend Lead:	Shifali Kalra
Fronted:	Duru Yilmaz
Backend:	Anwer Al-Dhify

Milestone 5

Submitted on 27.03.2024.

Table of Contents

EXECUTIVE SUMMARY	4
PERSONAS AND MAIN USE CASES:	5
PERSONAS:	5
MAIN USE CASES:	6
LIST OF MAIN DATA ITEMS AND ENTITIES:	7
FUNCTIONAL REQUIREMENTS – PRIORITIZED:	9
LIST OF NON-FUNCTIONAL REQUIREMENTS:.....	12
HIGH LEVEL ARCHITECTURE, DATABASE ORGANIZATION:	13
SOFTWARE STACK:.....	13
ADDITIONAL TECHNOLOGIES:	13
PROPOSED DATABASE ORGANIZATION:.....	13
HIGH LEVEL UML DIAGRAMS:.....	14
DESCRIPTION OF THE CLASSES:	14
USABILITY TEST PLAN	15
QA TEST PLAN	18
CODE REVIEW	19
SELF-CHECK ON BEST PRACTICES FOR SECURITY	24
SELF-CHECK: ADHERENCE TO ORIGINAL NON-FUNCTIONAL SPECS	24
LIST OF NON-FUNCTIONAL REQUIREMENTS:.....	24
COMPETITIVE ANALYSIS:	25
PLANNED ADVANTAGES:	25
PROJECT MANAGEMENT:	26
TASK ASSIGNMENT AND TRACKING:	26
TEAM MEMBERS AND ROLES:	26



Executive Summary

MediaMajesty is an innovative web-based service developed by a dedicated team of 6 students, aimed at enhancing the academic and creative experience of Fulda University students and faculty. Our platform serves as a centralized hub for the Fulda community to seamlessly buy, sell, or share a wide range of digital media, including images, music, videos, and graphics. The primary objective of MediaMajesty is to foster collaboration, knowledge exchange, and creativity among our users.

Why fund MediaMajesty?

In today's fast-paced digital world, sharing and accessing high-quality digital media resources are critical for students and faculty to excel in their academic and creative pursuits. MediaMajesty addresses this need by providing a user-friendly platform tailored to the specific requirements of Fulda University. We envision this platform as a catalyst for collaboration, innovation, and efficient resource utilization.

Key Advantages and Novelty:

MediaMajesty distinguishes itself in several ways:

- **Tailored to Fulda:** Unlike generic digital media platforms, MediaMajesty is exclusively available to Fulda students and faculty. This exclusivity ensures a trusted and closely-knit community for sharing and accessing resources.
- **Safe and Moderated Environment:** To maintain the integrity of the platform, an administrator will oversee every media item, ensuring compliance with community guidelines and eliminating inappropriate content.
- **Free and Secure:** MediaMajesty prioritizes the security of media owners. Users can post items for sale or free use, allowing them to control access and engage with potential buyers securely.
- **User-Friendly Interface:** Our platform is designed for individuals of all technical backgrounds, making it accessible to a broad audience.

About Our Team:

MediaMajesty is the brainchild of a passionate and diverse team of Fulda University students. Our team comprises individuals with varied expertise, from computer science to design, software engineering, and much more. We are committed to building a secure and user-friendly platform that will empower Fulda students and faculty in their academic and creative endeavors. Our dedication and collective vision drive this project forward.

In conclusion, MediaMajesty is more than just a digital media marketplace; it's a community-driven initiative that fosters collaboration, creativity, and resource-sharing within Fulda

University. We seek your support to bring this innovative platform to life, making a positive impact on the Fulda academic community and beyond.

Personas and Main Use Cases:

Personas:

1. Student Sam:

Characteristics: Undergraduate student, tech-savvy, active social media user.

Goals: Sam wants to easily find and share digital media resources for their coursework and creative projects. Sam hopes to connect with other students and faculty for collaboration and resource exchange.

Skills: Proficient in using web applications and social media platforms.

Pain Points: Sam struggles to find high-quality, Fulda-specific digital resources for their projects, and they are concerned about the authenticity and legality of downloaded content.

2. Faculty Member Fiona:

Characteristics: Fulda University professor, experienced in their field, limited tech expertise.

Goals: Fiona seeks a platform that simplifies the process of sharing class materials, research findings, and educational resources with their students. They aim to foster an engaging and collaborative learning environment.

Skills: Limited technical skills, comfortable with basic computer operations.

Pain Points: Fiona finds it challenging to share and manage digital resources within a secure and university-specific environment, often facing copyright and accessibility issues.

3. Media Creator Max:

Characteristics: Aspiring artist, photographer, or musician, passionate about their craft.

Goals: Max wants to showcase and sell their creative digital media to a local and supportive community. They aim to generate income and build a network of customers.

Skills: Proficient in creating digital media but may lack experience in online marketing and sales.

Pain Points: Max finds it difficult to connect with potential buyers and lacks a secure platform to sell their digital media without the risk of copyright infringement.

4. Admin/Administrator Alice:

Characteristics: University staff member responsible for platform moderation.

Goals: Alice is tasked with ensuring the platform's content adheres to the university's guidelines, maintains a safe and secure environment for all users, and resolves disputes or issues that may arise.

Skills: Proficient in platform administration, content moderation, and community management.

Pain Points: Alice's challenges include handling disputes and ensuring that only authorized users gain access to the platform.

Main Use Cases:

Use Case 1: Student Sam Finds and Downloads Digital Resources (Title: "Resource Discovery")

Sam logs into MediaMajesty, enters relevant search criteria, and discovers a selection of digital media resources uploaded by peers and faculty. Sam browses the content, selects a resource, and downloads it in full resolution, facilitating the completion of an assignment. The platform's user-friendly design makes it easy for Sam to find and access resources.

Use Case 2: Faculty Member Fiona Shares Class Materials (Title: "Classroom Collaboration")

Fiona, a professor at Fulda University, logs into MediaMajesty. She uploads lecture slides, study materials, and additional resources for her students to access. Fiona uses the platform to facilitate seamless communication with her students, ensuring all necessary materials are readily available, improving the learning experience.

Use Case 3: Media Creator Max Sells Digital Artwork (Title: "Creative Marketplace")

Max, an aspiring artist, wants to sell their digital artwork. Max registers on MediaMajesty, uploads their artwork, sets a selling price, and provides descriptions. A fellow student or faculty member discovers Max's artwork, agrees to the purchase terms, and buys the digital media, empowering Max to monetize their talent.

Use Case 4: Admin Alice Ensures Platform Compliance (Title: "Moderation and Content Oversight")

Alice, the platform administrator, logs into MediaMajesty to oversee and approve new media uploads. She ensures that all uploaded content complies with university guidelines and resolves any disputes or inappropriate items. Alice's active involvement in the moderation process ensures a safe and secure environment for all users, maintaining the platform's integrity.

Use Case 5: Guest Explorer Grace Explores MediaMajesty (Title: "Guest Exploration")

Grace, an external visitor without a MediaMajesty account, visits the platform out of curiosity. As a guest user, she can browse through a limited selection of public resources and get a glimpse of the diverse content available. While unable to download or contribute without an

account, this use case allows potential users to preview the platform's offerings and functionality, potentially encouraging them to create an account for full access in the future.

List of main data items and entities:

1. User Profiles: User profiles represent the information about registered individuals using MediaMajesty, including students, faculty, and administrators.

- Sub-items:
 - ✓ Username: The chosen name that identifies a user on MediaMajesty.
 - ✓ Email Address: The unique email associated with a user's account.
 - ✓ Password: Encrypted information allowing user access.
 - ✓ Profile Picture: Image chosen by the user to represent their identity.
 - ✓ Bio: A brief description provided by the user about themselves.
 - ✓ Role: Indicates whether the user is a student, faculty, or administrator.

2. Media Items: Media items represent digital content uploaded by users for sharing, selling, or viewing.

- Sub-items:
 - ✓ File: The actual digital content (image, video, audio, etc.).
 - ✓ Title: The name given to the media item.
 - ✓ Description: A summary or details about the media item.
 - ✓ Tags: Keywords associated with the media item for searchability.
 - ✓ Upload Date: The date and time when the media item were uploaded.
 - ✓ Views: The number of times the media item has been accessed.

3. Messaging System: The messaging system is a feature that enables users to communicate with each other.

- Sub-items:
 - ✓ Conversations: Threads of messages between users.
 - ✓ Participants: Users involved in a particular conversation.
 - ✓ Timestamps: Indicates when each message was sent.
 - ✓ Attachments: Option to share media items within messages.

4. Admin Controls: Admin controls refer to the functionality used by the platform administrator to manage user accounts, uploaded media, and resolve disputes.

- Sub-items:
 - ✓ User Management: Tools for creating, modifying, or deleting user accounts.
 - ✓ Media Moderation: Features to monitor and manage uploaded media content.
 - ✓ Dispute Resolution: System for resolving conflicts reported by users.
 - ✓ Access Logs: Records of administrative activities within the platform.

5. Terms and Conditions: Terms and Conditions outline the rules and guidelines for using MediaMajesty.

- Sub-items:
 - ✓ Acceptance: User agreement to adhere to platform rules.
 - ✓ Privacy Policy: Guidelines on how user data is collected and used.
 - ✓ Code of Conduct: Defines acceptable behavior on the platform.
 - ✓ Dispute Resolution: Procedures for handling conflicts between users.

6. User Privileges: User privileges define the actions and access rights available to different user types (Student, Faculty, Admin).

- Sub-items:
 - ✓ Upload Permissions: Determines who can upload media items.
 - ✓ Access Levels: Defines the scope of features accessible to different user roles.
 - ✓ Account Management: Specifies actions users can take regarding their accounts.
 - ✓ Reporting: Ability to report inappropriate content or behavior.

7. Search Queries: Search queries are user-generated requests for specific media items.

- Sub-items:
 - ✓ Keywords: Terms entered by users to find specific content.
 - ✓ Filters: Options to refine search results (e.g., by date, type, popularity).
 - ✓ Search History: Records of past user searches.

8. Accessibility Settings: Accessibility settings allow users to customize their experience, such as text size and screen reader compatibility.

- Sub-items:
 - ✓ Text Size: Options for adjusting the size of text on the platform.
 - ✓ Color Contrast: Settings to enhance visibility for users with visual impairments.
 - ✓ Screen Reader Compatibility: Features to improve accessibility for screen reader users.
 - ✓ Keyboard Shortcuts: Quick commands for users with mobility challenges.

These key data items and entities form the foundation of MediaMajesty, shaping user interactions and defining the structure of the platform's database and user interface. Consistent use of these terms will guide the development and design process.

Functional requirements – prioritized:

Requirement ID	FR-01
Title	User Registration
Priority	1
Description	Users can create accounts with their name, email, and password. User type selection is mandatory during registration.
Requirement ID	FR-02
Title	User Login
Priority	1
Description	Registered users can log in with their credentials to access the platform
Requirement ID	FR-03
Title	Media Upload
Priority	1
Description	Users can upload digital media items, providing details like title, description, type, and resolution.
Requirement ID	FR-04
Title	Media Search and Discovery
Priority	1
Description	Users can search for media items based on keywords, type, and other filters.
Requirement ID	FR-05
Title	Messaging System
Priority	1
Description	Users can communicate with each other through a messaging system to discuss media items and transactions.
Requirement ID	FR-06
Title	User Profile Management
Priority	1
Description	Users can edit their profiles, update personal information, and change settings.
Requirement ID	FR-07
Title	Content Moderation



Priority	1
Description	The administrator can review and approve media items, as well as remove inappropriate content.
Requirement ID	FR-08
Title	Version Control
Priority	1
Description	Users can upload updated versions of their media items, with the platform maintaining a version history for each item.
Requirement ID	FR-09
Title	User Privilege Assignment
Priority	1
Description	Admins have the ability to assign and adjust users roles based on their responsibilities on the platform.
Requirement ID	FR-10
Title	User Authentication
Priority	1
Description	Users prompted to reset their passwords if forgotten
Requirement ID	FR-11
Title	Resolution Control
Priority	2
Description	Access to full-resolution media items is only granted after users agree on terms (purchase or free use).
Requirement ID	FR-12
Title	User Privilege Management
Priority	2
Description	The administrator can assign roles and privileges to users based on their type and responsibilities.
Requirement ID	FR-13
Title	User Feedback and Rating
Priority	2
Description	Users can provide feedback and ratings for media items and other users to build trust.



Requirement ID	FR-14
Title	Reporting System
Priority	2
Description	Users can report inappropriate content or issues, triggering action by the platform administrator
Requirement ID	FR-15
Title	Categorization
Priority	2
Description	Users can organize, and filter content based on predefined categories.
Requirement ID	FR-16
Title	Privacy Control
Priority	2
Description	Users can set privacy settings for each media Item, choosing between public, private and shared with specific users.
Requirement ID	FR-17
Title	Users Onboarding Tutorials
Priority	2
Description	New users receive steps by steps tutorials highlighting the key features.
Requirement ID	FR-18
Title	Creative Marketplace Features
Priority	2
Description	Media creators can set prices for their digital media items. And users can view a list of media items available for sale in a dedicated "Marketplace" section
Requirement ID	FR-19
Title	Terms and Conditions Acceptance
Priority	3
Description	Users must agree to the platform's terms and conditions during registration.
Requirement ID	FR-20
Title	Emergency response
Priority	3

Description

Admins have a feature to respond quickly to emergency situations, such as handling harmful or inappropriate content.

List of non-Functional requirements:

- Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in Milestone 0. Application delivery shall be from chosen cloud server.
- Application shall be optimized for standard desktop/laptop browsers e.g., must render correctly on the two latest versions of two major browsers.
- All our selected application functions must render well on mobile devices.
- Data shall be stored in the database on the team's deployment cloud server.
- Full resolution free media shall be downloadable directly, and full resolution media for selling shall be obtained after contacting the seller/owner.
- No more than 50 concurrent users shall be accessing the application at any time.
- Privacy of users shall be protected, and all privacy policies will be appropriately communicated to the users.
- The language used shall be English (no localization needed)
- Application shall be very easy to use and intuitive.
- Application should follow established architecture patterns.
- Application code and its repository shall be easy to inspect and maintain.
- Google analytics shall be used (optional for Fulda teams)
- No email clients shall be allowed.
- Pay functionality, if any (e.g., paying for goods and services) shall not be implemented nor simulated in UI.
- Site security: basic best practices shall be applied (as covered in the class) for main data items.
- Application shall be media rich (images, video etc.). Media formats shall be standard as used in the market today.
- Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development.

- For code development and management, as well as documentation like formal milestones required in the class, each team shall use their own GitHub to be set-up by class instructors and started by each team during Milestone 0
- The application UI (WWW and mobile) shall prominently display the following exact text on all pages "Fulda University of Applied Sciences Software Engineering Project, Fall 2023 For Demonstration Only" at the top of the WWW page. (Important to not confuse this with a real application).

High level Architecture, Database Organization:

Software stack:

Sever Host: Azure virtual machine Standard B1ms

Operating System: Ubuntu 22.04

Database: MySQL version 8.2.0

Web Server: NGINX 1.24.0

Server-Side Language: Python

Client-Side Language: JavaScript

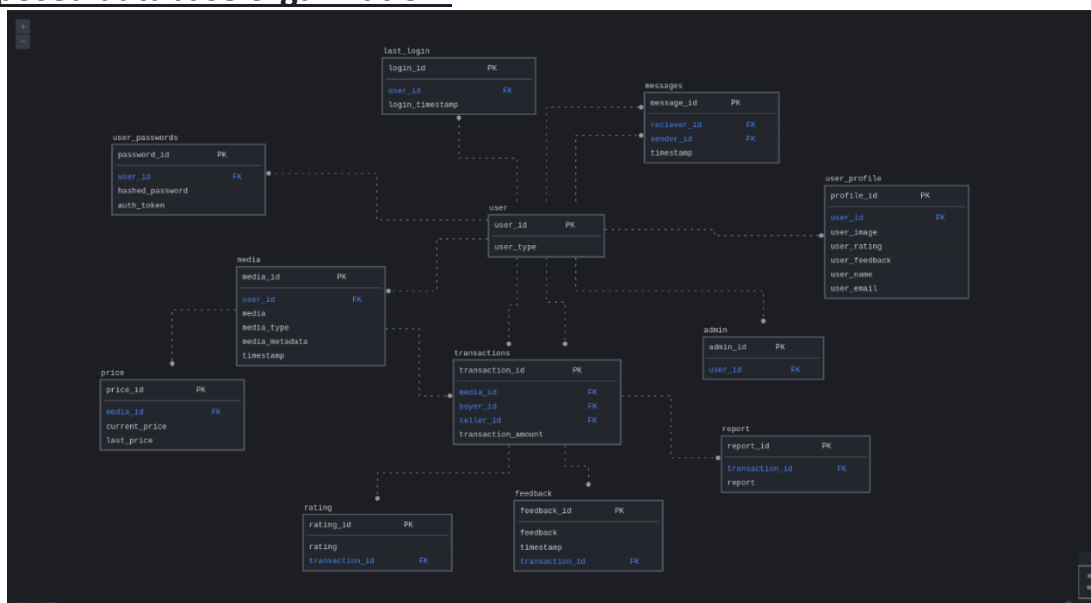
Additional Technologies:

Web Frameworks: Django,

IDE: VSCode, NeoVim, PyCharm

SSL Cert: Lets Encrypt (Cert Bot)

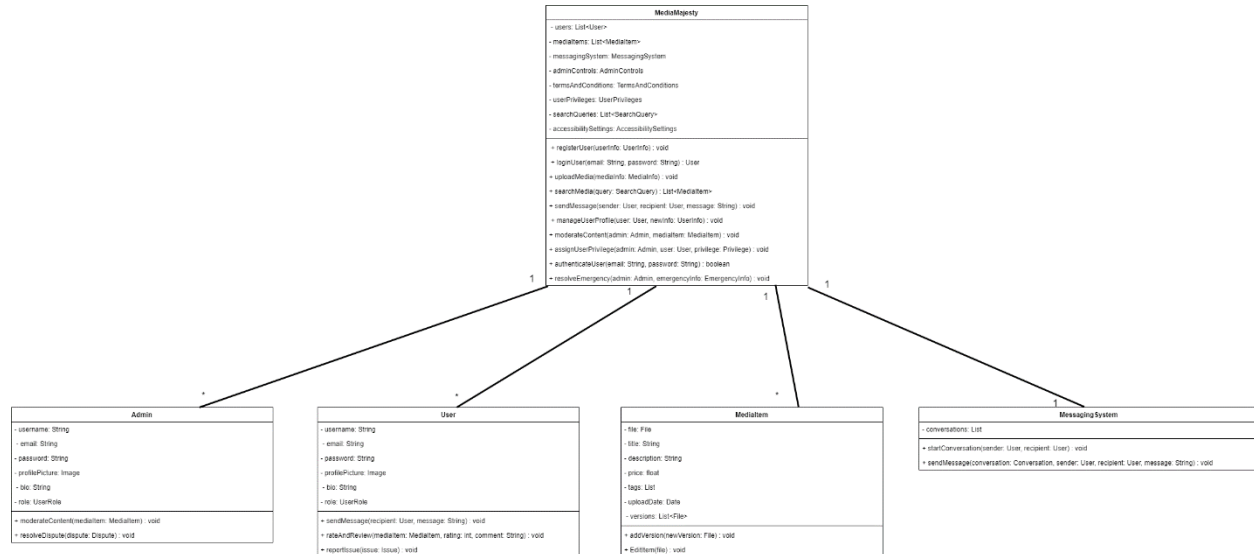
Proposed database organization:





High Level UML Diagrams:

Class diagram is one of the static structure diagrams that shows the main entities of the projects and how they communicate with each other. It is designed for the whole system to give an overview of the attributes of the main entities and the relationships between them.



Description of the classes:

The **MediaMajesty** class represents the main platform for the MediaMajesty system. It serves as a centralized hub for Fulda University students and faculty to buy, sell, or share digital media. This class manages users, digital media items, messaging systems, administrative controls, terms and conditions, user privileges, search queries, and accessibility settings.

The **User** class represents individuals registered on the MediaMajesty platform. Users, including students and faculty, have profiles with attributes such as username, email, password, profile picture, bio, and role. Users can interact with the platform by sending messages, providing ratings and reviews, and reporting issues.

The **Admin** class represents administrators responsible for overseeing the platform. Admins have profiles like users but possess additional responsibilities, including moderating content, resolving disputes, and managing the platform's integrity.

The **Mediatem** class represents digital media items uploaded by users on the platform. It includes attributes such as the media file, title, description, tags, upload date, and a version history. Users can upload, update, and manage their digital media items using this class.

The **MessagingSystem** class represents the messaging system within MediaMajesty. It manages user conversations and facilitates communication between users. Users can start conversations, send messages, and share media items through this system.

Usability Test Plan

Functions to be tested: Media Upload and Download Functionality

Test Objectives:

The primary objective of this usability test is to evaluate the user experience of the media upload and download functionality on the website. Specifically, we aim to assess the effectiveness, efficiency, and user satisfaction related to the process of uploading media (videos, images, audios, and PDFs) to Azure Blob Storage and subsequently viewing and downloading them on the website.

Test Background and Setup:

System Setup: The test will be conducted on the staging environment that replicates the production setup. The Azure Blob Storage integration and website functionalities will be identical to those in the live system.

Starting Point: Testers will start at the homepage of the website with a pre-logged-in user account.

Intended Users: The intended users for this test are individuals with basic to intermediate technical proficiency, representing the target audience for the website.

Measurement Focus (M4): The primary focus of measurement will be on user satisfaction, assessed through Likert scale evaluations. Other aspects, such as task completion time and error rates, will also be considered to gauge the overall usability.

Usability Task Description:

Task: Upload a video file to Azure Blob Storage using the provided functionality on the website.

Instructions:

- a. Locate the "Upload Media" section on the website.
- b. Select a video file from your local device.
- c. Complete any required metadata or description fields.
- d. Initiate the upload process.
- e. Navigate to the "Media page" and locate the uploaded video.
- f. Download the video file to your local device.

Measurement:

Effectiveness: Measure the percentage of successfully uploaded and downloaded media files.

Efficiency: Record the time taken by the user to complete the entire task.

Likert Scale Questions:

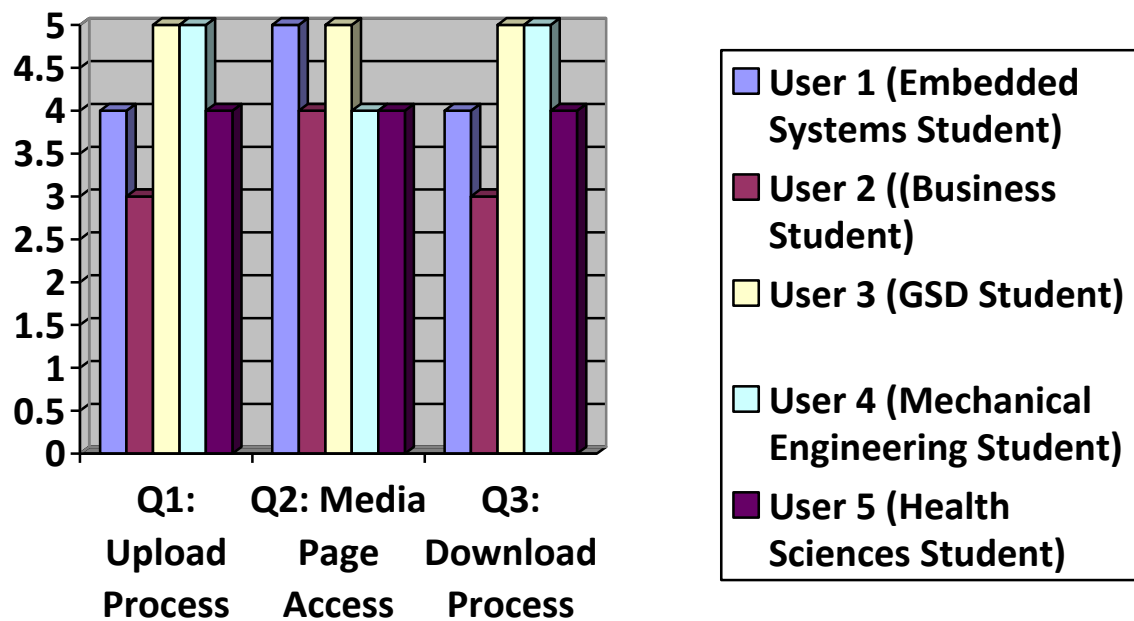
After completing the task, users were asked to rate their satisfaction on a Likert scale (1-5), where 1 indicates "Strongly Disagree" and 5 indicates "Strongly Agree."

Q1: The process of uploading media files was straightforward.

Q2: I could easily find and access the uploaded media in the Media Page.

Q3: The download process for media files was intuitive.

User	Q1: Upload Process	Q2: Media Page Access	Q3: Download Process
User 1 (Embedded Systems Student)	4	5	4
User 2 ((Business Student)	3	4	3
User 3 (GSD Student)	5	5	5
User 4 (Mechanical Engineering Student)	5	4	5
User 5 (Health Sciences Student)	4	4	4



Likert Scale:

1 (Strongly Disagree) 2 (Disagree) 3 (Neutral) 4 (Agree) 5 (Strongly Agree)

Interpretation:

User 1 (Embedded Systems Student):

- Background: Currently studying embedded systems, User 1 is likely tech-savvy, detail-oriented, and has a strong interest in the functionality of systems.
- Objectives: User 1 might be using the media functionality for collaborative projects involving software development, requiring efficient file management, and sharing.
- Interpretation: User 1 found the upload process and media page access satisfactory, indicating alignment with their technical background. However, they were slightly less satisfied with the download process, suggesting potential areas for improvement in the context of software development collaboration.

User 2 (Business Student):

- Background: Studying business, User 2 likely prioritizes efficiency, user-friendliness, and practicality in software tools.
- Objectives: User 2 might use the media functionality for business-related projects involving international teams or clients, emphasizing effective communication and collaboration.
- Interpretation: User 2 had a moderately positive experience, finding the tasks somewhat agreeable. Their responses could suggest that the system meets basic usability requirements, but there may be opportunities for enhancements to better support global collaboration within a business context.

User 3 (GSD Student - Global Software Development):

- Background: Studying Global Software Development, User 3 likely has a comprehensive understanding of software development processes.
- Objectives: User 3 might be using the media functionality for collaborative coding, documentation, and project management in a distributed software development environment.
- Interpretation: User 3 expressed high satisfaction with all aspects of the functionality. This could indicate that the system is well-suited for users in the field of Global Software Development, meeting their specific needs for efficient collaboration, file sharing, and media management in a global context.

User 4 (Mechanical Engineering Student):

- Background: Currently studying mechanical engineering, User 4 is likely to have a strong technical background with a focus on mechanical systems, design, and manufacturing processes.
- Objectives: User 4 might use the media functionality for managing CAD (Computer-Aided Design) files, project documentation, and multimedia elements related to mechanical design projects.
- Interpretation: User 4's feedback could prioritize the seamless integration of technical drawings, 3D models, and other media assets commonly used in mechanical engineering

projects. Their Likert scale responses may reflect the importance of an organized and accessible platform for managing diverse types of engineering files.

User 5 (Health Sciences Student):

- Background: Engaged in health sciences studies, User 5 is likely to be involved in medical studies, data analysis, and healthcare-related projects.
- Objectives: User 5 might use the media functionality for managing medical images, research data, and multimedia content related to health sciences research.
- Interpretation: User 5's Likert scale responses could emphasize the importance of a secure and efficient system for handling sensitive medical data and multimedia content. Their feedback may reflect the need for features that facilitate collaboration among healthcare professionals in a research setting.

QA Test Plan

Test Objectives:

The primary objective of the QA test is to ensure the reliability and functionality of the media upload and download feature on the website. This includes verifying the successful upload and download of various media types to and from Azure Blob Storage.

HW and SW Setup:

Environment: Staging environment replicating the production setup.

URL: <http://20.51.194.250:8000/>

Feature to be Tested: Media Upload and Download Functionality

QA Test Plan Table:

Test #	Test Title	Test Description	Test Input	Expected Correct Output	Test Results (Browser 1 / Browser 2)
1	Upload Video File	Verify the successful upload of a video file.	Select a video file named "TestFile1" from the local device.	Confirm "TestFile1" is uploaded successfully.	PASS / PASS
2	Download Uploaded Video	Verify the successful download of an uploaded video.	Purchase a video named "PurchasedMedia1," and navigate to the download section.	Confirm "PurchasedMedia1" is downloaded successfully.	PASS / PASS

3	Metadata Completeness	Ensure metadata completeness during media upload.	Upload media with incomplete metadata (e.g., missing title, description) and name it "IncompleteMedia."	Verify that the system prompts for missing data.	PASS / PASS
---	-----------------------	---	---	--	-------------

Results of Testing:

Tested on Browser 1 (e.g., Chrome) and Browser 2 (e.g., Firefox).

User / Test #	1 (Upload)	2 (Download)	3 (Metadata)
User 1 (Embedded Systems Student)	PASS	PASS	PASS
User 2 ((Business Student)	PASS	PASS	PASS
User 3 (GSD Student)	PASS	PASS	PASS

Interpretation:

All test cases were successfully passed on both browsers, indicating the reliability and functionality of the media upload and download feature.

This structured QA test plan ensures comprehensive validation of the media functionality while maintaining clarity and professionalism.

Code Review

Our team has adopted GitHub as the primary platform for our code review process. Every team member has submitted a substantial portion of the code for peer review. The review has been conducted entirely within the GitHub environment, utilizing the platform's built-in features such as pull requests and inline comments. The peer review includes constructive feedback, suggestions, and any necessary comments directly within the codebase on GitHub. Screenshots of the review process, capturing the inline comments and overall feedback, will be submitted to document and ensure a transparent and collaborative code review experience for the team. This approach leverages the version control capabilities of GitHub to enhance code quality and collaboration within our development workflow.



mediamajesty/mediamajesty/settings.py Outdated

```

156 + AZURE_ACCOUNT_KEY = 'sEQH8IaY8dUxmFoKu8LMtIdiJVgSWFJcQuN/8A51iobkUxK1IVJpX+QppnhrUoXB58EtodqeZ700+AStMCyxfg=='
157 +
158 + # Azure Storage Container settings
159 + AZURE_CONTAINER = 'media-majesty-container'

```

7ze 2 weeks ago

Don't add your credentials to Github!

👍 1

```

149 +
150 +
151 + # Use AzureBlobStorage for storing static files.
152 + DEFAULT_FILE_STORAGE = str(os.getenv("DEFAULT_FILE_STORAGE"))
153 + AZURE_ACCOUNT_NAME = str(os.getenv("AZURE_ACCOUNT_NAME"))
154 + AZURE_ACCOUNT_KEY = str(os.getenv("AZURE_ACCOUNT_KEY"))
155 + AZURE_CONTAINER = str(os.getenv("AZURE_CONTAINER"))

```

Figure 1 Anwar code review

mediamajesty/items/approval_view.py Outdated

```

17 +
18 +
19 + @user_passes_test(lambda u: u.is_staff) # type: ignore
20 + def approve_item(_, id):

```

AnwerHSFulda 2 weeks ago

In Django views, the first parameter is typically the request object, so it might be better to change it.

👍

```

17
18
19 @user_passes_test(lambda u: u.is_staff) # type: ignore
20 + def approve_item(request, id):
21     item = get_object_or_404(Item, id=id)
22     item.is_approved = True
23     item.save()

```

Figure 2 Tom code review



```

10     blob_client = blob_service_client.get_blob_client(container=container_name, blob=blob_name)
11
12     with open(file=local_path, mode="wb") as local_blob:
13         download_stream = blob_client.download_blob()
14         local_blob.write(download_stream.readall())
15 -
16     @login_required
17     def download(request, id):
18         item = get_object_or_404(Item, id=id)
19 @@ -26,11 +25,9 @@ def download(request, id):
20
21
22
23
24
25
26     container_name = AZURE_CONTAINER
27
28
29 -     local_path = os.path.join('Downloads', blob_name.split("/")[-1])
30 -     if not os.path.exists('Downloads'):
31 -         current_directory = os.getcwd()
32 -         print("Current Working Directory:", current_directory)
33 -
34
35     download_blob_to_file(blob_service_client, container_name, blob_name, local_path)
36

```

7ze marked this conversation as resolved. Hide resolved

7ze 2 weeks ago ...

this code writes the blob to the server

```

9     blob_client = blob_service_client.get_blob_client(container=container_name, blob=blob_name)
10
11     with open(file=local_path, mode="wb") as local_blob:
12         download_stream = blob_client.download_blob()
13         local_blob.write(download_stream.readall())
14 +
15     @login_required
16     def download(request, id):
17         item = get_object_or_404(Item, id=id)
18
19
20
21
22
23
24
25     container_name = AZURE_CONTAINER
26
27
28 +     downloads_folder = os.path.join(os.path.expanduser("~"), 'Downloads')
29 +
30 +     local_path = os.path.join(downloads_folder, blob_name.split("/")[-1])
31
32
33     download_blob_to_file(blob_service_client, container_name, blob_name, local_path)
34

```

Figure 3 Achraf code review



mediamajesty/core/templates/core/about.html
Outdated

...
...
@@ -3,5 +3,25 @@
3
3
{% block title %}About{% endblock %}
4
4
5
5
{% block content %}
6
- <h1>About. This is the about page!</h1>
6
+

7ze 4 days ago
...

when adding images, put them in a `static` folder, that can be referenced by django. Absolute paths would break in production.

5
{% block content %}
6
<div class="bg-white p-6 rounded-lg shadow-lg">
7
<div class="w-20 h-20 mx-auto bg-gray-300 rounded-full overflow-hidden">
8

9
</div>
10
<h3 class="text-xl font-semibold mt-4">Abbas Abbas</h3>
11
<p class="text-gray-600">Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
12
</div>
13
14
<div class="bg-white p-6 rounded-lg shadow-lg">
15
<div class="w-20 h-20 mx-auto bg-gray-300 rounded-full overflow-hidden">
16

17
</div>

Figure 2 Duru code review

mediamajesty/chats/templates/chats/inbox.html

17
-
</p>
18
-
</div>
19
-

11
+

7ze 35 minutes ago
Author
...

we need to rework this in the future, ie make both inbox and chats into one page / view

Figure 3 Abbas code review

```

34 mediamaJesty/items/templates/items/item.html
42 - class="inline-block bg-yellow-500 hover:bg-yellow-600 text-white font-bold mt-6 py-2 px-4 rounded-sm">
43 - Download </a>
44 - {% else %}
45 - <button class="inline-block bg-yellow-500 hover:bg-yellow-600 text-white font-bold mt-6 py-2 px-4 rounded-sm"
46 - id="purchaseBtn">Purchase</button>
47 - {% endif %}
39 + {% if item.price == 0.0 %}
40 + <a href="{% url 'items:download' item.id %}" class="inline-block bg-yellow-500 hover:bg-yellow-600 text-white font-bold mt-6 py-2 px-4 rounded-sm">
41 + Purchase
42 + </a>
43 + {% else %}
44 + <button class="inline-block bg-yellow-500 hover:bg-yellow-600 text-white font-bold mt-6 py-2 px-4 rounded-sm" id="purchaseBtn">
45 + Purchase
46 + </button>
47 + {% endif %}
48 + {% endif %}
49 </div>
50 </div>
51
52 <div class="mt-6 px-12 py-12 bg-gray-100 rounded-sm">
53 <h2 class="mb-6 text-2xl text-center">Related items</h2>
54 <div class="grid grid-cols-4 gap-10">
55 - {% for item in related_items %}
55 + {% for related_item in related_items %}
56 <div>
57 - <a href="{% url 'items:item' item.id %}">
57 + <a href="{% url 'items:item' related_item.id %}">
58 <div>
59 - 
59 + 
60 </div>
61 <div class="p-6 bg-white rounded-b-xl">
62 - <h2 class="text-2xl">{{ item.name }}</h2>
62 + <h2 class="text-2xl">{{ related_item.name }}</h2>
63 - <p class="text-gray-500">Price: {{ item.price }}€</p>
63 + <p class="text-gray-500">Price: {{ related_item.price }}€</p>
64 </div>
65 </a>
66 </div>
67 {% endfor %}
68 </div>
69 </div>
70 +
71 + <div class="mt-6 px-12 py-12 bg-gray-100 rounded-sm">
72 + <h2 class="mb-6 text-2xl text-center">Product Reviews</h2>
73 + {% if feedbacks %}
74 + <div class="grid grid-cols-3 gap-6">
75 + {% for feedback in feedbacks %}
76 + <div class="bg-white p-4 rounded-md shadow-md">
77 + <p class="text-lg font-semibold mb-2"><strong>{{ feedback.user.username }}</strong></p>
78 + <p class="text-md mb-2">Rating: {{ feedback.rating }}</p>
79 + <p class="text-gray-700">{{ feedback.feedback }}</p>
80 + </div>
81 + {% endfor %}
82 + </div>
83 + <p class="text-center mt-4 text-xl font-semibold">Average Rating: {{ average_rating }}</p>
84 + {% else %}
85 + <p class="text-center text-gray-700">No feedback available for this item.</p>
86 + {% endif %}
87 + </div>
88 +
89 +
71 90 {% endblock %}
72 91
73 92 {% block scripts %}

```

Figure 6 Shifali code review

Self-Check on Best Practices for Security

Various protective measures are employed across key assets to fortify the platform against potential threats. In terms of user data, the focus is on averting unauthorized access and data breaches by regularly updating security protocols. Media content is shielded from unauthorized copying or distribution, as well as offensive material, through the implementation of content moderation and manual review processes. The messaging system is safeguarded against unauthorized access and phishing attacks with a comprehensive set of protections, including Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), SQL injection, Clickjacking, and SSL/HTTPS. User authentication credentials are protected against password cracking and unauthorized access through stolen credentials by storing hashed and salted passwords. Overall security measures include encryption of user passwords, data validation to prevent injection attacks, regular security audits, user role management, password reset mechanisms, and strict access controls. Notably, the encryption of passwords in the database and input data validation for the search bar are highlighted as crucial components of the security architecture. These collective measures ensure the confidentiality, integrity, and resilience of the system across various assets.

Self-Check: Adherence To Original Non-Functional Specs

List of non-Functional requirements:

- Application shall be developed, tested, and deployed using tools and servers approved by Class CTO and as agreed in Milestone 0. Application delivery shall be from chosen cloud server. ✓
- Application shall be optimized for standard desktop/laptop browsers e.g., must render correctly on the two latest versions of two major browsers. ✓
- All our selected application functions must render well on mobile devices. ✓
- Data shall be stored in the database on the team's deployment cloud server. ✓
- Full resolution free media shall be downloadable directly, and full resolution media for selling shall be obtained after contacting the seller/owner. ✓
- No more than 50 concurrent users shall be accessing the application at any time. ✓
- Privacy of users shall be protected, and all privacy policies will be appropriately communicated to the users. ✓
- The language used shall be English (no localization needed) ✓
- Application shall be very easy to use and intuitive. ✓
- Application should follow established architecture patterns. ✓
- Application code and its repository shall be easy to inspect and maintain. ✓

- Google analytics shall be used (optional for Fulda teams)
- No email clients shall be allowed. ✓
- Pay functionality, if any (e.g., paying for goods and services) shall not be implemented nor simulated in UI. ✓
- Site security: basic best practices shall be applied (as covered in the class) for main data items. ✓
- Application shall be media rich (images, video etc.). Media formats shall be standard as used in the market today. ✓
- Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development. ✓
- For code development and management, as well as documentation like formal milestones required in the class, each team shall use their own GitHub to be set-up by class instructors and started by each team during Milestone 0. ✓
- The application UI (WWW and mobile) shall prominently display the following exact text on all pages "Fulda University of Applied Sciences Software Engineering Project, Fall 2023 For Demonstration Only" at the top of the WWW page. (Important to not confuse this with a real application). ✓

Competitive analysis:

Feature	Behance	Dribbble	Etsy	MediaMajesty
Media Upload	Yes	Yes	Yes (digital)	Yes
Messaging System	Yes	Yes	Yes	Yes
Resolution Control	No	No	No	Yes
Content Moderation	Limited	Yes	Limited	Yes
User Privilege Management	No	No	No	Yes

Planned Advantages:

MediaMajesty distinguishes itself by catering specifically to the Fulda University community, offering a platform where students and faculty can share and access digital media. It provides unique features like resolution control, enabling users to regulate access to high-resolution

media. It also includes robust content moderation to maintain a secure environment, which some competitors lack. Furthermore, MediaMajesty's user privilege management system empowers administrators to assign roles and privileges, enhancing the platforms.

Project management:

To efficiently manage and plan M2 and future tasks, we have adopted Trello as our primary task management tool. Trello provides a user-friendly interface and a unified dashboard view, allowing us to streamline collaboration, track progress, and ensure transparency across all assigned tasks.

Task Assignment and Tracking:

Assignment of Tasks:

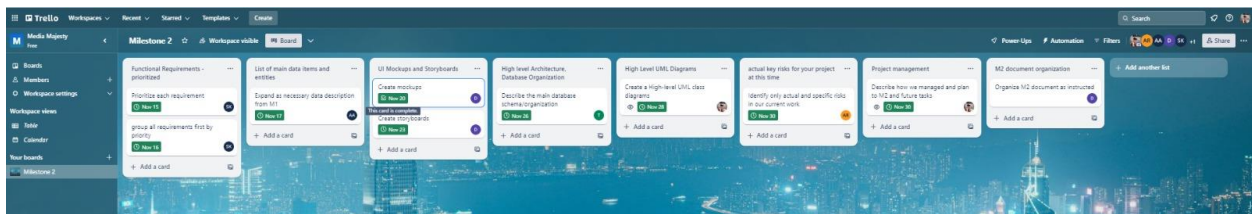
- We began by assigning specific tasks for M2 to team members based on their expertise and project requirements.
- Tasks were categorized into manageable units to enhance clarity and focus.

Trello Board Setup:

- We created a dedicated Trello board for M2 and subsequent tasks, with lists representing different stages of the workflow.
- Each task was represented as a Trello card within the appropriate list.

Member Responsibilities:

- Team members were assigned responsibility for specific cards, outlining the scope, deadlines, and any dependencies.



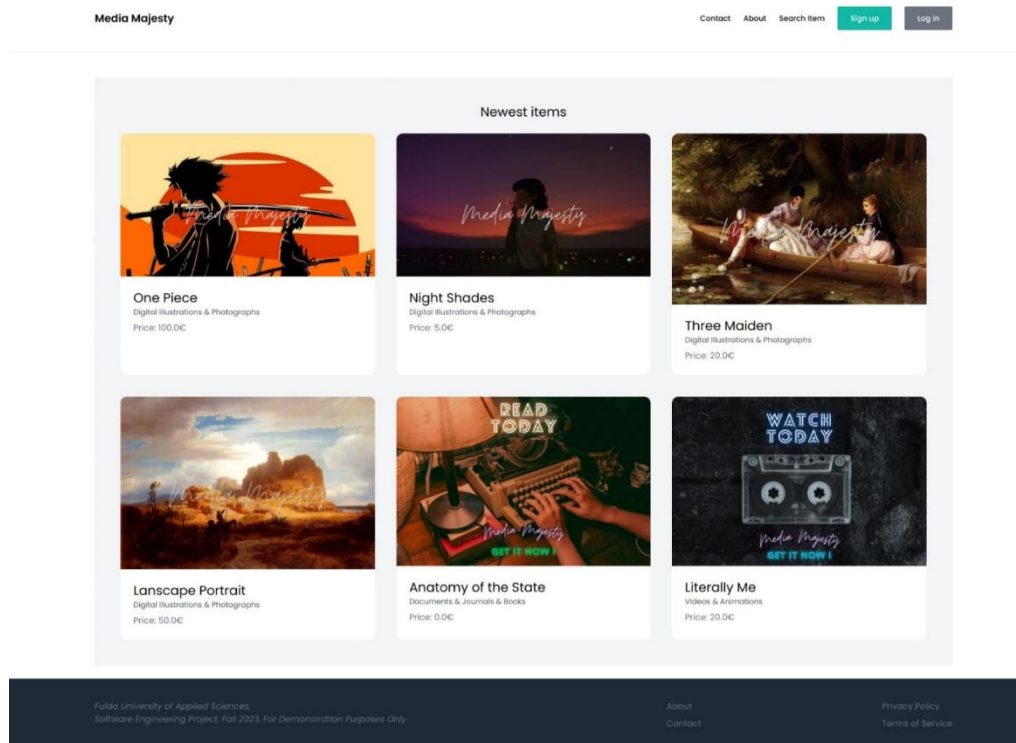
Team members and roles:

Name	Email	Nickname	Role	Can meet these times outside the class
Abbas	abbas.abbas@informatik.hs-fulda.de	Abbas	Team Lead	Yes
Achraf	achraf.boudabous@informatik.hs-fulda.de	Achraf	Frontend lead	Yes

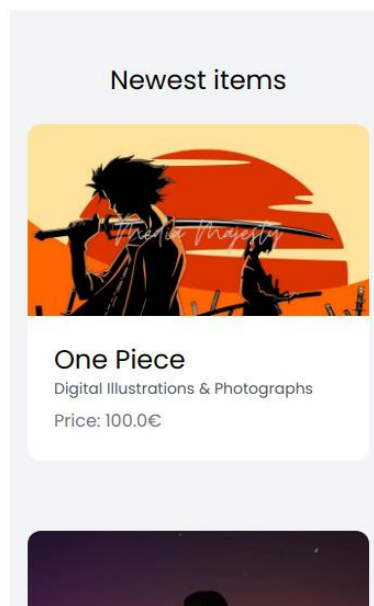
Shinu	shinu.donney@informatik.hs-fulda.de	Tom	GitHub master	Yes
Shifali	Shifali.shifali@informatik.hs-fulda.de	Shifali	Backend lead	Yes
Duru	duru.yilmaz@informatik.hs-fulda.de	Duru	Frontend	Yes
Anwer	anwer-ahmed.al-dhify@informatik.hs-fulda.de	Anwer	Backend	Yes

Screenshots from the website:

Landing page:



Media Majesty



Search results page:

Media Majesty

[Contact](#)
[About](#)
[Search Item](#)
[Sign up](#)
[Log in](#)

Categories


Digital Illustrations & Photographs
Documents & Journals & Books
Music & Audio
Videos & Animations

Filters


Minimum Price:

Maximum Price:


Apply Price
Clear Filters




After Dark
Music & Audio
Price: 5.0€




American upon city
Documents & Journals & Books
Price: 15.0€




Anatomy of the State
Documents & Journals & Books
Price: 0.0€




But benefit
Videos & Animations
Price: 5.0€




Can
Digital Illustrations & Photographs
Price: 40.0€



Eye hair important
Documents & Journals & Books
Price: 7.0€



Landscape Portrait
Digital Illustrations & Photographs
Price: 50.0€



Literally Me
Videos & Animations
Price: 20.0€

Media Majesty

Categories

Digital Illustrations & Photographs
Documents & Journals & Books
Music & Audio
Videos & Animations

Filters


Minimum Price:

Maximum Price:

Apply Price
Clear Filters

Product details page:

Media Majesty
Contact About Search Item Sign Up Log In




Landscape Portrait
50.00€
Sold by tom
Digital Illustrations & Photographs
Landscape


Contact Seller
Purchase
Report

Product Reviews
No feedback available for this item.


Related Items




Cam
Price: 45.00€




Night Shades
Price: 9.00€



One Piece
Price: 100.00€



Samurai Horse
Price: 9.00€



Three Maiden
Price: 35.00€

Fulda University of Applied Sciences
Software Engineering Project Fall 2020 For Demonstration Purposes Only

About
Contact

Privacy Policy
Terms of Service

Media Majesty



One Piece

100.00€

Sold by tom

Digital Illustrations &
Photographs

Contact Seller

Purchase

Report