Project Proposal

Project Name: MySpace

Project Topic: Social Media Platform

Group Members' Names:

• Berşan KORKMAZ – 21SOFT1031

• Volkan DÖLEK – 21SOFT1019

• Özgür YILDIZ – 22COMP1054

Purpose of the Website:

MySpace aims to create an engaging social media platform where users can interact through a unique system of sending stars instead of likes. The platform will feature different user tiers, such as Novice Astronaut, Star Explorer, and Galaxy Pioneer, where each tier offers increased interaction capabilities. This system promotes both social engagement and user progression within the platform. The website allows users to create profiles, share posts, follow others, and interact with content, creating a dynamic and interactive environment.

Users and Their Capabilities:

Users Capabilities:

1. Novice Astronaut:

o Can create and update their profile.

o Can post updates and interact with others' posts by sending stars.

o Can follow other users, view their posts, and leave comments.

o Has a daily limit of 15 stars to send to other posts.

2. Star Explorer:

o Can create and update their profile.

o Can post updates and interact with others' posts by sending stars.

o Can follow other users, view their posts, and leave comments.

o Has a daily limit of 30 stars to send to other posts.

3. Galaxy Pioneer:

o Can create and update their profile.

o Can post updates and interact with others' posts by sending stars.

o Can follow other users, view their posts, and leave comments.

o Has a daily limit of 60 stars to send to other posts.

Admin Capabilities:

o An admin role can be included to manage user accounts, monitor inappropriate

content, and oversee platform operations.

o Admins can moderate posts, suspend or delete user accounts, and access detailed

reports about user activities.

Use Case Scenarios:

Use Case Scenario 1: User creates and updates their profile
Scenario: A user creates and updates their profile on the platform.
Flow of Events:

1. User logs in to the platform.

2. User navigates to the profile creation page.

3. User fills in their personal information (e.g., username, bio, profile picture).

4. User saves the profile details.

5. User goes back to the profile page to view the created profile.

6. User can later update their profile by editing the information and saving changes.

Use Case Scenario 2: User posts content
Scenario: A user posts an update (text, image, or video) on the platform.
Flow of Events:

1. User logs in to the platform.

2. User navigates to the post creation page.

3. User enters the content (text, uploads image/video).

4. User clicks the "Post" button to submit the content.

5. System displays the new post on the user's profile and in the feed of users they follow.

Use Case Scenario 3: User sends stars to interact with a post
Scenario: A user sends stars to a post to show appreciation.
Flow of Events:

1. User logs in to the platform.

2. User browses the feed and finds a post they want to interact with.

3. User clicks on the star icon below the post.

4. System prompts the user to confirm the star action.

5. User confirms and sends stars to the post.

6. System updates the post's star count and shows the user's interaction.

Use Case Scenario 4: User follows another user
Scenario: A user follows another user to view their posts.
Flow of Events:

1. User logs in to the platform.

2. User searches for or finds another user they want to follow.

3. User clicks on the "Follow" button on that user's profile.

4. System confirms the follow action and updates the user's feed to include the followed user's posts.

Use Case Scenario 5: User views popular posts in the Star Waterfall
Scenario: A user views the most popular posts based on stars received.
Flow of Events:

1. User logs in to the platform.

2. User navigates to the "Star Waterfall" section.

3. System displays posts with the most stars in descending order.

4. User browses through the posts and interacts with content by sending stars or leaving comments.

5.

Use Case Scenario 6: Admin manages user accounts
Scenario: Admin manages user accounts, including banning or suspending accounts.
Flow of Events:

1. Admin logs in to the platform using admin credentials.

2. Admin navigates to the user management dashboard.

3. Admin searches for a specific user account or views the list of all users.

4. Admin selects a user account that needs to be managed.

5. Admin reviews the user's activity and determines if action is needed.

6. If the user has violated platform guidelines, the admin chooses to suspend or ban the account.

7. System prompts the admin to confirm the action (suspend/ban).

8. Admin confirms the action, and the system suspends or bans the user's account.

9. System updates the user's account status, and the user is notified of the action.

Use Case Scenario 7: User logs in to the platform
Scenario: A user logs in to access their account and interact with the platform.
Flow of Events:

1. User opens the platform's login page.

2. User enters their registered username and password into the login fields.

3. User clicks the "Login" button.

4. System validates the provided credentials against the database.

   - If the credentials are correct, proceed to step 5.

   - If the credentials are incorrect, the system displays an error message and prompts the user to try again.

5. System redirects the user to their homepage or dashboard.

6. User can now interact with the platform (e.g., post content, follow other users, send stars).

7. If the user chooses to log out, they can click the "Logout" button, and the system will log them out and return to the login page.

Use Case Scenario 7: User logs out of the platform

Scenario: A user logs out of their account to end their session.
Flow of Events:

1. User logs in to the platform.

2. User navigates to the logout option in the menu.

3. User clicks the "Logout" button.

4. System confirms the action and terminates the session.

5. System redirects the user to the login page.

Use Case Scenario 8: User unfollows another user

Scenario: A user unfollows another user to stop viewing their posts.
Flow of Events:

1.   User logs in to the platform.

2.   User navigates to the profile of a user they currently follow.

3.   User clicks the "Unfollow" button on that user's profile.

4.   System confirms the action and updates the user's feed to remove posts from the unfollowed user.

Use Case Scenario 9: User comments on a post

Scenario: A user leaves a comment on another user's post.
Flow of Events:

1.   User logs in to the platform.

2.   User browses the feed and selects a post to comment on.

3.   User types a comment in the comment box below the post.

4.   User clicks the "Submit" button.

5.   System validates the comment and adds it to the post's comment section.

6.   System notifies the post owner of the new comment.

Use Case Scenario 10: User deletes a comment

Scenario: A user deletes one of their comments from a post.
Flow of Events:

1.   User logs in to the platform.

2.   User navigates to the post containing their comment.

3.   User selects their comment and clicks the "Delete" button next to it.

4.   System prompts the user to confirm the deletion.

5.   User confirms, and the system removes the comment from the post's comment section.

Use Case Scenario 11: User sends a Super Star

Scenario: A user sends a "Super Star" to a post as a special interaction (e.g., premium users only or limited per day).
Flow of Events:

1. User logs in to the platform.

2. User browses the feed and selects a post they want to send a "Super Star" to.

3. User clicks the "Super Star" icon below the post.

4. System checks if the user has "Super Stars" available (based on daily or tier-based limits).

    o   If available, proceed to step 5.

    o   If not, system notifies the user of insufficient "Super Stars."

5. User confirms the action, and the system sends the "Super Star" to the post.

6. System updates the post's Super Star count and highlights the user's interaction.

Use Case Scenario 12: Admin deletes a post

Scenario: Admin removes a post that violates platform guidelines.
Flow of Events:

1. Admin logs in to the platform using admin credentials.

2. Admin navigates to the content moderation dashboard.

3. Admin searches for the post in question or reviews flagged posts.

4. Admin selects the post and clicks the "Delete Post" button.

5. System prompts the admin to confirm the deletion.

6. Admin confirms, and the system removes the post from the platform.

7. System notifies the post owner about the deletion and the reason for removal.

Use Case Scenario 13: User reports a post

Scenario: A user reports a post for violating platform guidelines.
Flow of Events:

1. User logs in to the platform.

2. User browses the feed and selects a post they find inappropriate.

3. User clicks the "Report" button on the post.

4. System prompts the user to provide a reason for reporting.

5. User submits the report, and the system forwards it to the admin team for review.

6. Admin reviews the report and takes necessary action (e.g., warning, post deletion, account suspension).

Use Case Scenario 15: Admin bans a user

Scenario: Admin bans a user for violating platform guidelines.
Flow of Events:

1. Admin logs in to the platform using admin credentials.

2. Admin navigates to the user management dashboard.

3. Admin searches for or selects a user account to review.

4. Admin reviews the user's activity and determines that a ban is necessary.

5. Admin clicks the "Ban User" button.

6. System prompts the admin to confirm the action.

7. Admin confirms, and the system updates the user's account status to "Banned."

8. System sends a notification to the banned user explaining the reason for the ban.

9. The user can no longer access the platform and is shown a "Banned" message if they attempt to log in.

Use Case Scenario 16: User checks notifications

Scenario: A user views notifications related to their account activity.
Flow of Events:

1. User logs in to the platform.

2. User navigates to the notifications section.

3. System displays a list of recent notifications (e.g., new followers, post interactions, comments, admin warnings).

4. User clicks on a specific notification to view more details.

5. System redirects the user to the relevant section (e.g., post, profile, or warning page).

Use Case Scenario 17: User sets their profile

Scenario: A user customizes their profile settings.
Flow of Events:

1. User logs in to the platform.

2. User navigates to the profile settings page.

3. User uploads a profile picture, enters or updates their bio, and adjusts visibility settings (e.g., public or private profile).

4. User selects preferences such as notification settings and display themes.

5. User clicks the "Save Changes" button.

6. System validates the inputs and updates the user's profile.

7. System redirects the user to their updated profile page.

Use Case Scenario 18: User views another user's profile

Scenario: A user checks the profile of another user on the platform.
Flow of Events:

1. User logs in to the platform.

2. User navigates to the search bar and searches for another user by name or username.

3. System displays search results.

4. User selects a profile from the search results.

5. System loads the selected user's profile, showing their bio, posts, followers, and other public information.

6. If the profile is private, the system limits the visible information and displays a "Follow to view more" prompt.

Use Case Scenario 19: User sends a direct message to another user

Scenario: A user initiates a private conversation with another user through direct messaging.
Flow of Events:

1. User logs in to the platform.

2. User navigates to the messages section or selects a user profile to start a conversation.

3. User clicks the "Message" button.

4. System opens a chat window.

5. User types a message and clicks "Send."

6. System delivers the message to the recipient and updates the chat history.

7. Recipient receives a notification about the new message.

Use Case Scenario 20: User receives an admin warning

Scenario: A user views a warning issued by the admin.
Flow of Events:

1. Admin logs in and identifies a user who has violated platform rules.

2. Admin issues a warning to the user.

3. System sends a notification to the user, including details of the warning and actions required.

4. User logs in to the platform and navigates to the notifications section.

5. User views the warning message and any guidelines provided.

Use Case Scenario 21: User edits or deletes their post

Scenario: A user modifies or removes one of their posts.
Flow of Events:

1. User logs in to the platform.

2. User navigates to their profile and selects a post.

3. User clicks the "Edit" or "Delete" button.

4. If editing, the system opens an editor for the user to modify the post content.

    o   User makes changes and clicks "Save," and the system updates the post.

5. If deleting, the system prompts the user to confirm the deletion.

6. User confirms, and the system removes the post from the platform.

Use Case Scenario 22: User likes a comment

Scenario: A user interacts with a comment by liking it.
Flow of Events:

1. User logs in to the platform.

2. User navigates to a post and views the comments.

3. User clicks the "Like" icon under a comment.

4. System updates the like count of the comment.

5. System notifies the comment author of the interaction.

Use Case Scenario 23: User blocks another user

Scenario: A user blocks another user to avoid interactions.
Flow of Events:

1. User logs in to the platform.

2. User navigates to another user's profile or a post made by that user.

3. User clicks the "Block" button.

4. System prompts the user to confirm the block action.

5. User confirms, and the system:

    o   Prevents the blocked user from viewing the blocker's profile or posts.

o   Removes the blocked user from the blocker's follower list (if applicable).

6. System notifies the user that the block was successful.

Use Case Scenario 24: User reports a post

Scenario: A user reports a post for inappropriate content.
Flow of Events:

1. User logs in to the platform.

2. User navigates to a post they find inappropriate.

3. User clicks the "Report" button.

4. System prompts the user to select a reason for reporting (e.g., spam, offensive content, copyright violation).

5. User selects a reason and clicks "Submit."

6. System logs the report and notifies the admin for further review.

Use Case Scenario 25: Admin reviews reported posts

Scenario: Admin reviews posts flagged by users.
Flow of Events:

1. Admin logs in to the platform using admin credentials.

2. Admin navigates to the "Reported Posts" section.

3. Admin views a list of reported posts along with the reasons and user details.

4. Admin reviews the content of the reported posts.

5. Admin decides whether to take action (e.g., remove the post, issue a warning, ban the user).

6. System updates the status of the report and notifies the involved users of the decision.