EduStock Exchange Platform

Team Name: *CodePlus*

Team Members:

Alejandro Diaz Meinecke

Kidus Solomon

Wardan Kazzalbach

Arham Ahmed

Izzie Powell
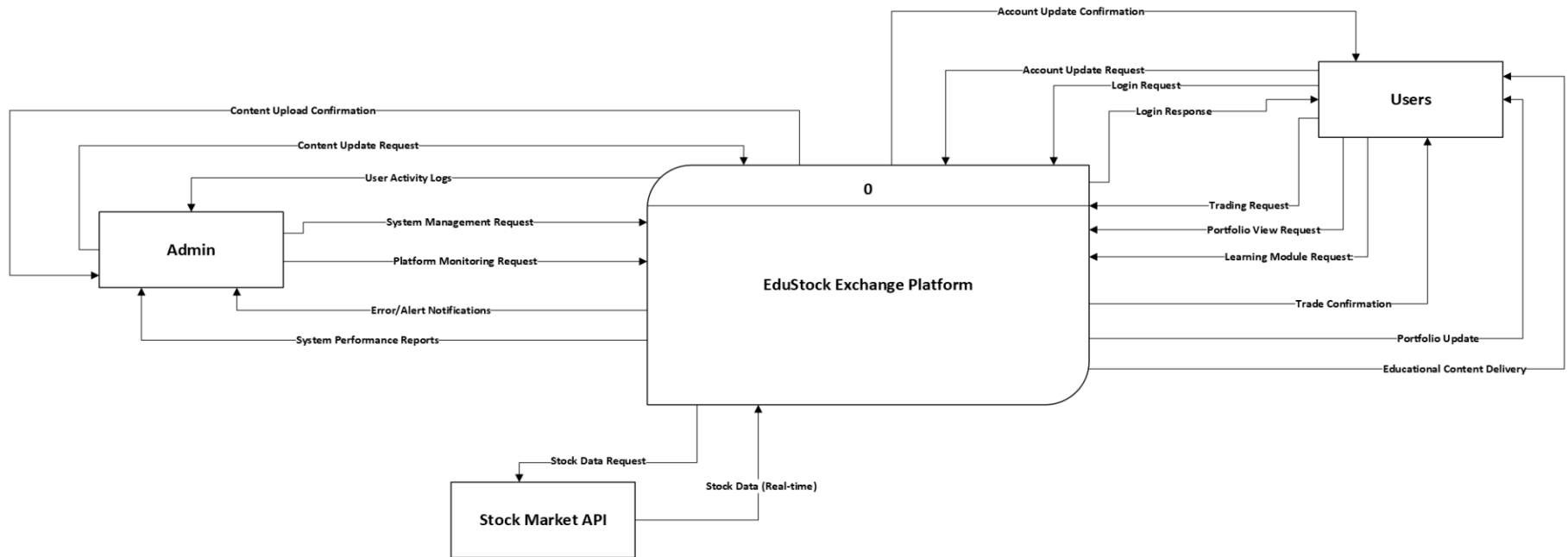
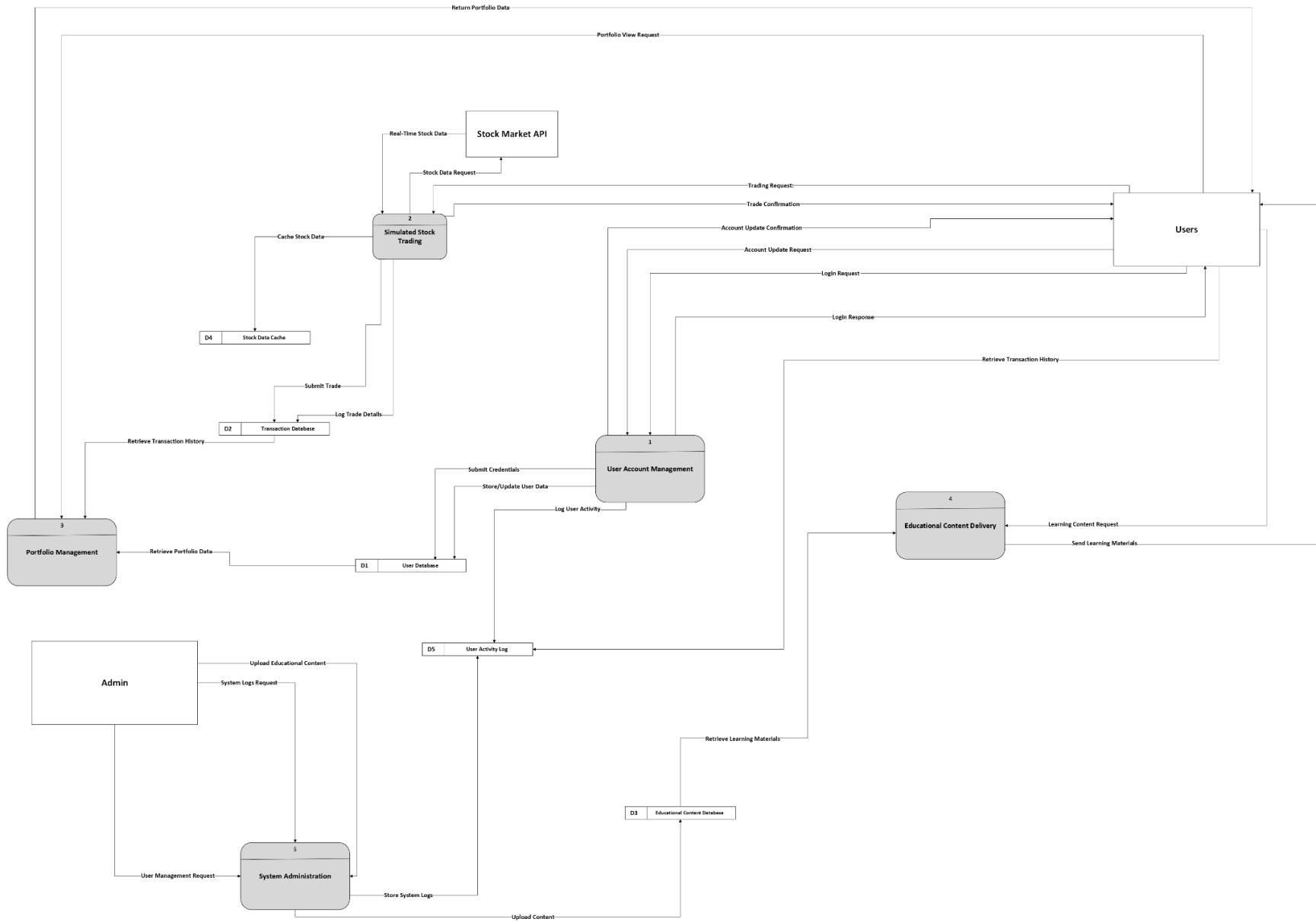**Course:** *IS 436: Structured Systems Analysis and Design*

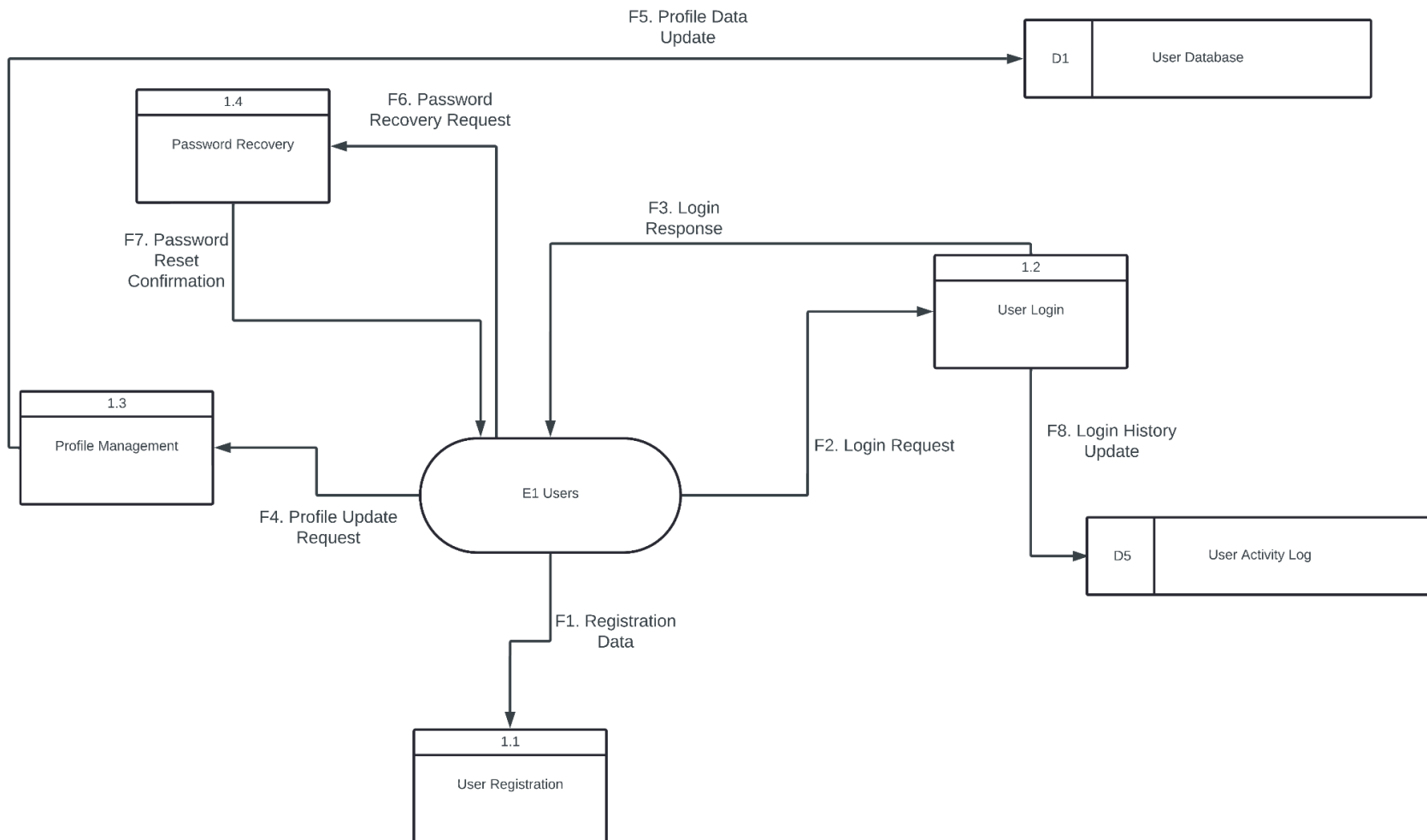**Deliverable 3**

Date: October 27th, 2024

Context Diagram

Users

Account Update Confirmation
Account Update Request
Login Request
Login Response
Trading Request
Portfolio View Request
Learning Module Request
Trade Confirmation
Portfolio Update
Educational Content Delivery

Admin

Content Upload Confirmation
Content Update Request
User Activity Logs
System Management Request
Platform Monitoring Request
Error/Alert Notifications
System Performance Reports

0

EduStock Exchange Platform

Stock Data Request
Stock Data (Real-time)

Stock Market API

Level 0

Stock Market API

Users

2 Simulated Stock Trading

1 User Account Management

3 Portfolio Management

4 Educational Content Delivery

5 System Administration

Admin

D4	Stock Data Cache
D2	Transaction Database
D1	User Database
D5	User Activity Log
D3	Educational Content Database

Return Portfolio Data

Portfolio View Request

Real-Time Stock Data

Stock Data Request

Trading Request

Trade Confirmation

Account Update Confirmation

Account Update Request

Login Request

Login Response

Cache Stock Data

Retrieve Transaction History

Submit Trade

Log Trade Details

Retrieve Transaction History

Submit Credentials

Store/Update User Data

Log User Activity

Learning Content Request

Send Learning Materials

Retrieve Portfolio Data

Upload Educational Content

System Logs Request

Retrieve Learning Materials

User Management Request

Store System Logs

Upload Content
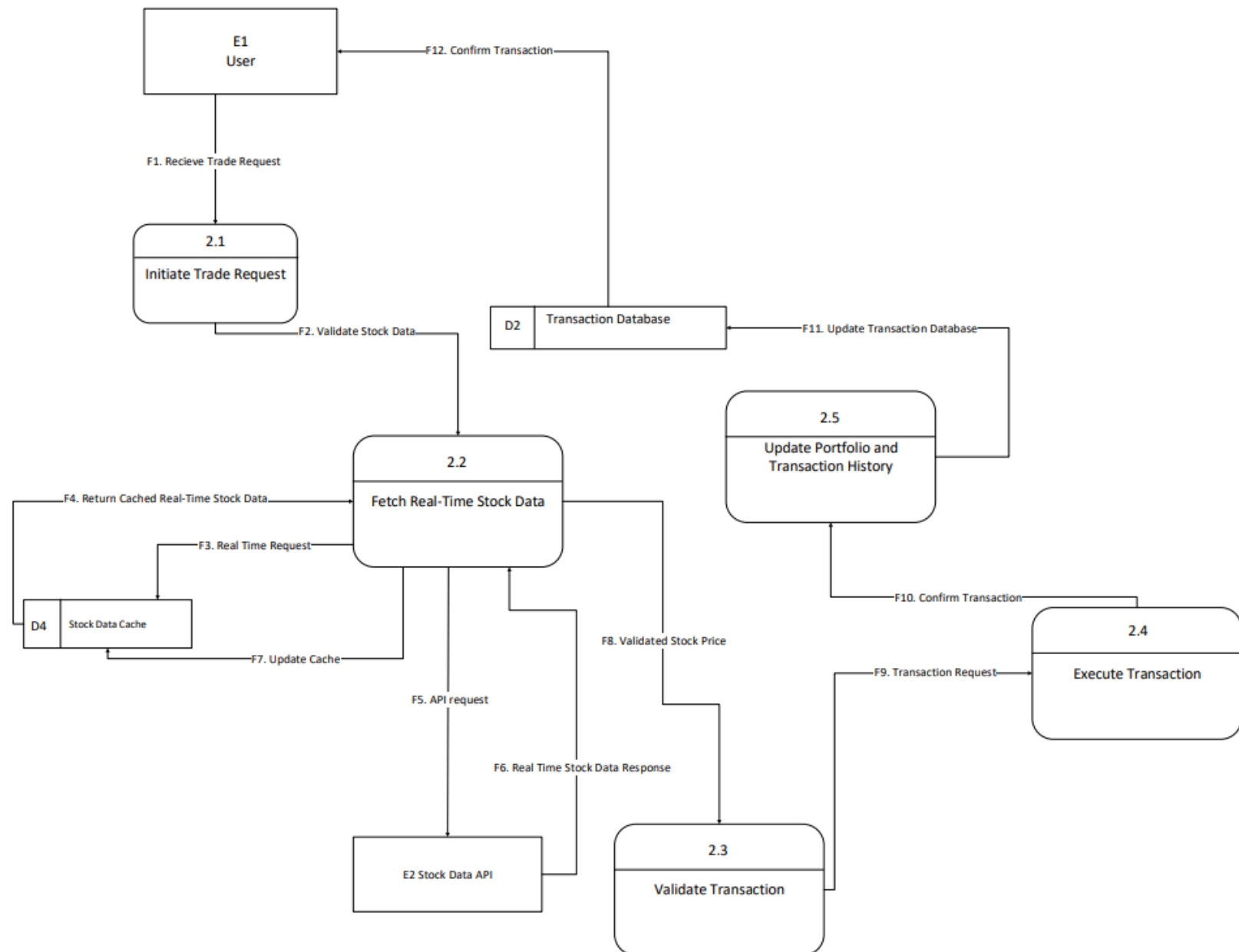
Process 1: Level 1

User Account Management

Text Definitions:
- Processes:
  - Process 1: User Account Management
    - This process takes inputs from the User entity (E1) and interacts with the User Database (D1) and User Activity Log (D5) data stores to allow users to register, log in, update their profiles, and recover passwords securely. The outputs from this process include login responses and password reset confirmations, which are sent back to the User entity (E1).
  - Process 1.1: User Registration
    - This process captures new user information; email, password, and profile information, and sends a confirmation email to verify user identity.
  - Process 1.2: User Login
    - This process verifies the users email and password with the system, and retrieves the user roles to personalize the user experience.
  - Process 1.3: Profile Management
    - This process allows users to view and update their personal information, like passwords and preferences, and provides access to previous activities like trades and learning content.
  - Process 1.4: Password Recovery
    - This process generatea a secure password recovery link and send it by email, and allows users to reset their passwords securely.
- Entities:
  - E1. User:
    - This entity is the user making requests like registration, login, and profile updates.
- Datastores:
  - D1: User Database
    - This datastore stores user credentials, profile details, and roles, and maintains password recovery data.
  - D5: User Activity Log
    - This datastore maintains login history.
- Data Flows:
  - F1: Registration Data

- This data flow is input from the User entity (E1) to User Registration (1.1), which occurs when a user submits their registration information. The data is stored in the User Database (D1).
  - F2. Login Request:
    - This data flow is input from the User entity (E1) to User Login (1.2), which occurs when a user inputs an email and password for authentication.
  - F3. Login Response:
    - This data flow is output from User Login (1.2) to the User entity (E1), which occurs after the system processes the login request and responds with a successful or error message.
  - F4. Profile Update Request:
    - This data flow is input from the User entity (E1) to Profile Management (1.3), which occurs when a user submits new profile details like an updated password or personal information.
  - F5. Profile Data Update:
    - This data flow is output from Profile Management (1.3) to the User Database (D1), which occurs when updated profile information is saved in the user database.
  - F6. Password Recovery Request:
    - This data flow is input from the User entity (E1) to Password Recovery (1.4), which occurs when a user requests to reset their password by entering the registered email.
  - F7. Password Reset Confirmation:
    - This data flow is output from Password Recovery (1.4) to the User entity (E1), which occurs after the user successfully resets their password using the reset link.
  - F8. Login History Update:
    - This data flow is output from User Login (1.2) to the User Activity Log (D5), which occurs after each successful login to maintain a login history in the database.

Process 2: Level 1

Simulated Stock Trading

Text Definitions:

- Processes:
  - Process 2: Simulated Stock Trading
    - This process takes inputs from the User entity (E1) and interacts with the Stock Data API (E2), Stock Data Cache (D4), and Transaction Database (D2) to allow users to initiate and execute trades with real-time stock data. The process validates transactions, updates user portfolios, and confirms completed trades. The outputs from this process include real-time stock data updates and transaction confirmations, which are sent back to the User entity (E1).
  - Process 2.1: Initiate Trade Request:
    - This process receives a trade request from the user and validates the basic structure of the request, including the stock ID and trade type (buy or sell).
  - Process 2.2: Fetch real-Time Stock Data
    - This process obtains the most relevant current stock data by checking the stock data cache for any existing real-time data. If no data is available, the cache requests new data from the API.
  - Process 2.3: Validate Transaction
    - This process validates the trade by synthesizing relevant data and checking if the user has available funds for buying or sufficient holdings for selling. If the funds or holdings are validated, the trade request is forwarded to execution.
  - Process 2.4: Execute Transaction
    - This process executes the validated trade, adds or subtracts funds from the user's account, and holds the transaction details to be saved in the system.
  - Process 2.5: Update Portfolio and Transaction History
    - This process updates the user portfolio to reflect the completed transaction and stores the details in the transaction database for record-keeping purposes.
- Entities:
  - E1 User:
    - This entity represents the user who initiates the trade request that is pushed to the transaction system.
  - E2 Stock Data API:
    - This entity provides real-time market data to the EduStock Platform, including information such as prices, trading volumes, and portfolio updates.
- Datastores:
  - D2 Transaction Database:

- ■ This datastore stores transaction and trade details for each user, maintaining a running history of all trades.
    - ○ D4 Stock Data Cache:
        - ■ This datastore temporarily stores real-time stock data retrieved from the API, reducing the time needed to fetch repetitive data and making the platform more efficient.
- Data Flow:
    - ○ F1: Receive trade request
        - ■ This data flow is input from the User entity (E1) to the Initiate Trade Request (2.1) process, which occurs when the user sends a request to either buy or sell. The request includes a stock ID, trade type, and the number of shares.
    - ○ F2: Validate stock data
        - ■ This data flow is output from the Initiate Trade Request (2.1) process to the Validate Transaction (2.3) process, which occurs when the stock ID, trade type, and number of shares are passed for validation using the currently available stock data.
    - ○ F3: Real-time request
        - ■ This data flow is input from the Validate Transaction (2.3) process to the Stock Data Cache (D4), which occurs when the system checks the cache using the most recent stock data based on the stock ID.
    - ○ F4: Return real-time stock data
        - ■ This data flow is output from the Stock Data Cache (D4) to the User entity (E1), which occurs when real-time stock data is provided to the user.
    - ○ F5: Real-time API request
        - ■ This data flow is output from the Stock Data Cache (D4) to the Stock Data API (E2), which occurs when a request for live stock data is made if the cache does not have the most recent stock data.
    - ○ F6: Real-time stock data request
        - ■ This data flow is output from the Stock Data API (E2) to the Stock Data Cache (D4), which occurs when the API provides the most relevant stock data in response to the request.
    - ○ F7: Update cache
        - ■ This data flow is input from the Stock Data API (E2) to the Stock Data Cache (D4), which occurs when the cache is updated with the newest and most relevant stock data.
    - ○ F8: Validated stock price
        - ■ This data flow is output from the Stock Data Cache (D4) to the Validate Transaction (2.3) process, which occurs when the stock price is carried for validation.

- ○ F9: Transaction request
    - ■ This data flow is output from the Validate Transaction (2.3) process to the Execute Transaction (2.4) process, which contains all the validated trade details and any other relevant transaction information.
- ○ F10: Transaction confirmation
    - ■ This data flow is output from the Execute Transaction (2.4) process to the Update Portfolio and Transaction History (2.5) process, which confirms the execution of the trade and provides all details for updating the transaction and portfolio database.
- ○ F11: Update database
    - ■ This data flow is input from the Update Portfolio and Transaction History (2.5) process to the Transaction Database (D2), which occurs when confirmed transaction details are stored.
- ○ F12: Confirm transaction
    - ■ This data flow is output from the Update Portfolio and Transaction History (2.5) process to the User entity (E1), which occurs when a transaction confirmation, along with updated portfolio details and a trade summary, is sent to the user.

Process 3: Level 1

Portfolio Management
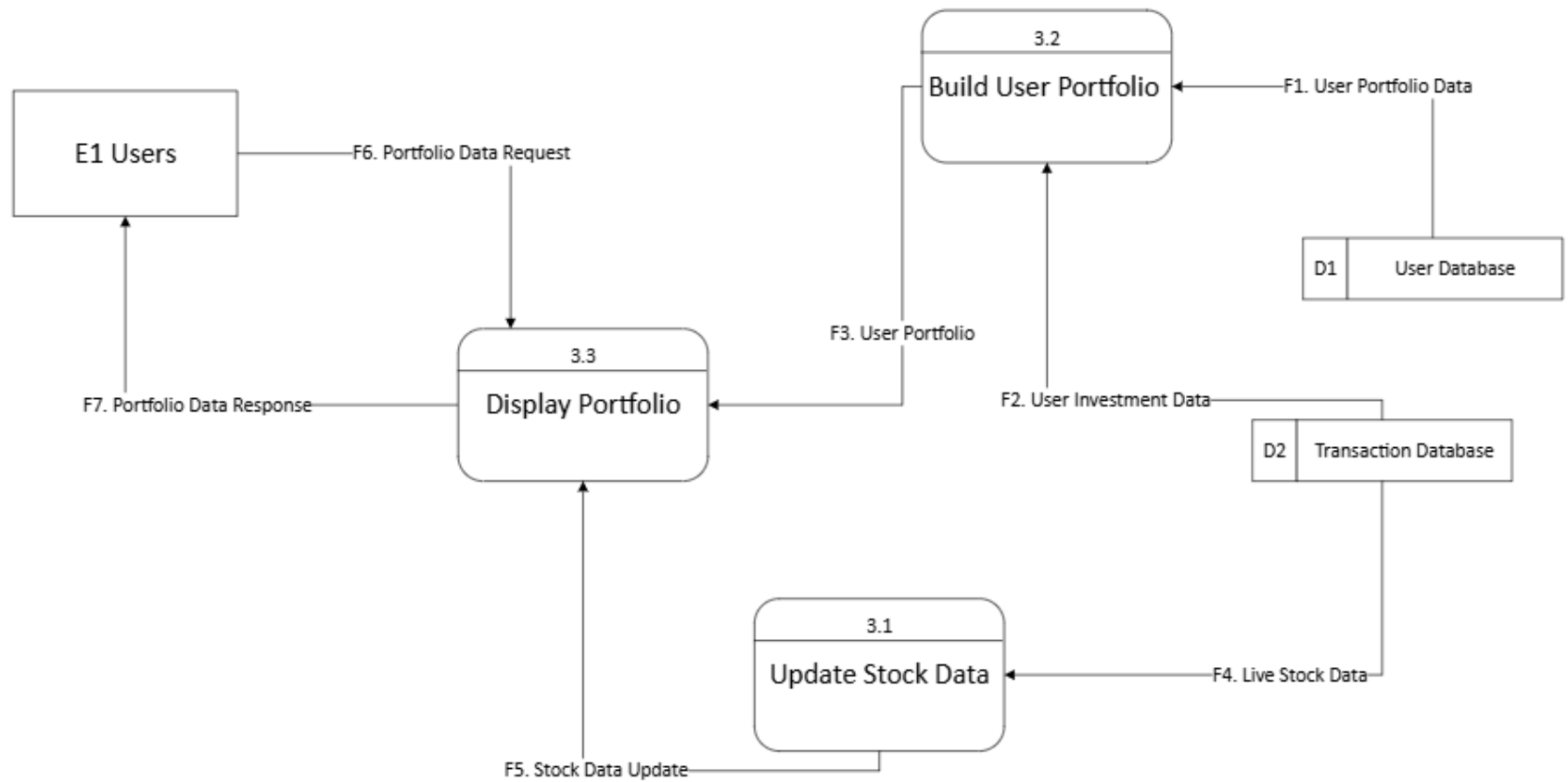
Text Definitions:

- Processes:
  - Process 3: Portfolio Management
    - This process takes the input Portfolio Data Request (F6) from the Users entity (E1). It interacts with the User Database (D1) and Transaction Database (D2) to provide the necessary information to build user portfolios. The portfolio data is then returned to the Users entity (E1).
  - Process 3.1: Update Stock Data
    - This process allows real-time stock data to be delivered to the portfolio for users to view stock data trends.
  - Process 3.2: Build User Portfolio
    - This process allows administrators to assemble the user's stock portfolio with the User Database (D1) and Transaction Database (D2) data stores.
  - Process 3.3: Display Portfolio
    - This process allows the user to view their stock portfolio including real-time stock data.
- Entities:
  - E1: Users
    - This entity represents users who can access their stock portfolio, which displays their stocks and current stock trends.
- Data Stores:
  - D1: User Database
    - This datastore contains information about the user that is used to build their portfolio.
  - D2: Transaction Database
    - This datastore contains information about the user's transaction history, which is used to build their portfolio.
- Data Flows:
  - F1: User Portfolio Data
    - This data flow is input from the User Database (D1) to the Build User Portfolio (3.2) process, which occurs when the database shares user information to assist in building their portfolio.
  - F2: User Investment Data
    - This data flow is input from the Transaction Database (D2) to the Build User Portfolio (3.2) process, which occurs when the database shares the user's investment information to assist in building their portfolio.
  - F3: User Portfolio

- This data flow is input from the Build User Portfolio (3.2) process to the Display Portfolio (3.3) process, which occurs when the user's portfolio is uploaded for the user to view.
  - F4: Live Stock Data
    - This data flow is input from the Transaction Database (D2) to the Update Stock Data (3.1) process, which occurs when real-time stock data is shared with the system.
  - F5: Stock Data Update
    - This data flow is output from the Update Stock Data (3.1) process to the Display Portfolio (3.3) process, which occurs when real-time stock data is updated into the portfolio for users to see live stock trends.
  - F6: Portfolio Data Request
    - This data flow is input from the Users entity (E1) to the Display Portfolio (3.3) process, which occurs when users request to view their stock portfolio.
  - F7: Portfolio Data Response
    - This data flow is output from the Display Portfolio (3.3) process to the Users entity (E1), which occurs when the user's stock portfolio is delivered to them, allowing them to view real-time stock data along with their investment information.

Process 4: Level 1

Educational Content Delivery
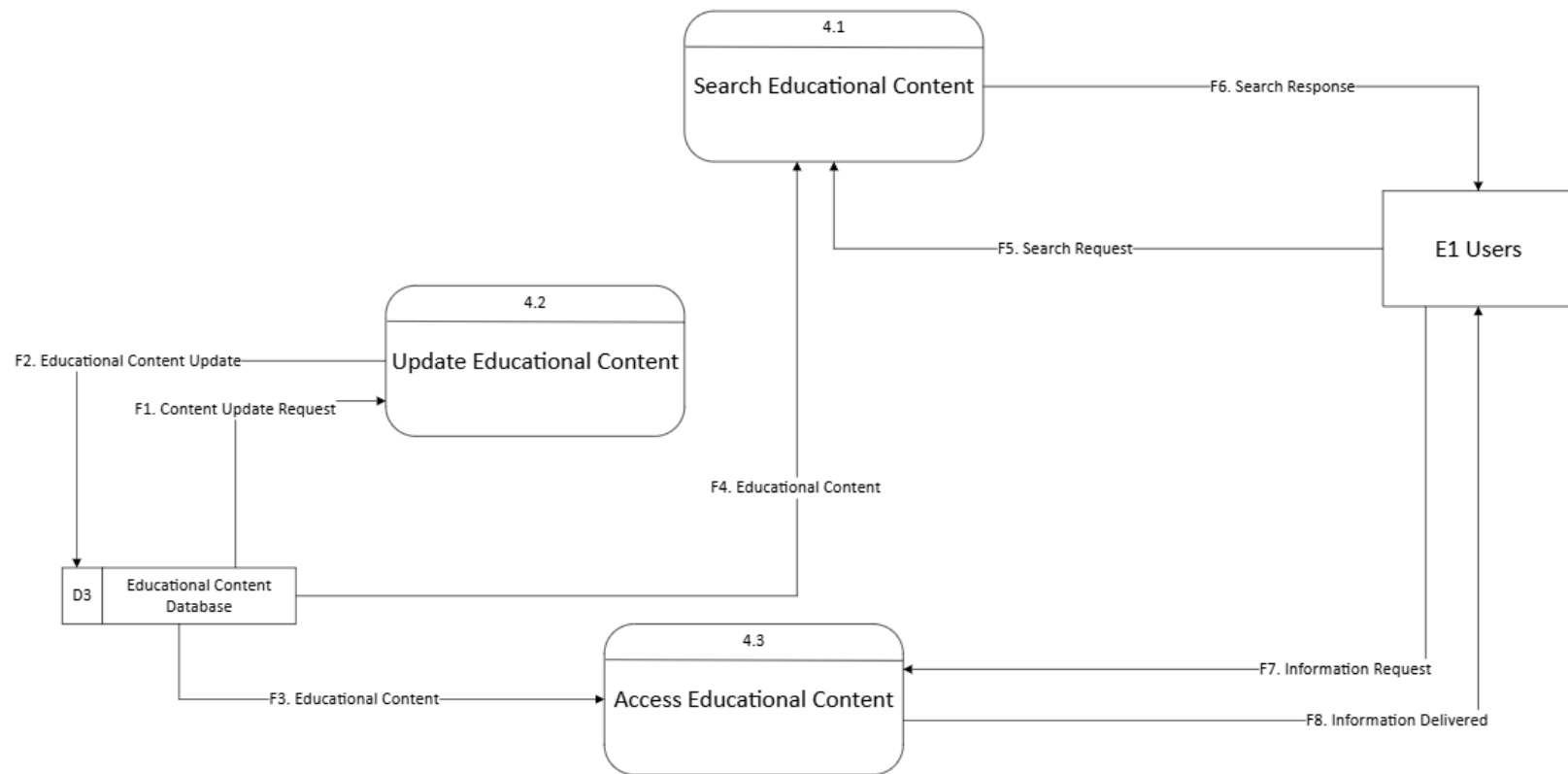
Text Definitions:

- Processes:
  - Process 4: Educational Content Delivery
    - This process takes inputs from the Users entity (E1) as Search Request (F4) and Information Request (F7). It interacts with the Educational Content Database (D3) to upload and provide users with access to educational content. The outputs from this process include educational content and search results, which are delivered back to the Users entity (E1).
  - Process 4.1: Search Educational Content
    - This process allows users to search for educational content related to stock trading. The search results are returned to users.
  - Process 4.2: Upload Educational Content
    - This process allows educational content to be uploaded for users to access. The educational content database (D3) is updated to include the new educational resources for users.
  - Process 4.3: Access Educational Content
    - This process allows users to access educational content from the educational content database (D3).
- Entities:
  - E1: Users
    - This entity represents users who request to search and access educational content.
- Data Stores:
  - D3: Educational Content Database
    - This data store maintains and updates educational content for users.
- Data Flows:
  - F1: Content Update Request
    - This data flow is input from the Educational Content Database to the Upload Educational Content (4.2) process, which occurs when new educational content is requested to be included in the database for users to access.
  - F2: Educational Content Update
    - This data flow is output from the Upload Educational Content (4.2) process to the Educational Content Database (D3), which occurs when the database is updated with new educational content.
  - F3: Educational Content

- ■ This data flow is input from the Educational Content Database (D3) to the Access Educational Content (4.3) process, which occurs when users request to access educational content from the database.
  - ○ F4: Educational Content
    - ■ This data flow is input from the Educational Content Database (D3) to the Search Educational Content (4.1) process, which occurs when the database provides new educational content for users to search.
  - ○ F5: Search Request
    - ■ This data flow is input from the Users entity (E1) to the Search Educational Content (4.1) process, which occurs when users search the database to find learning resources.
  - ○ F6: Search Response
    - ■ This data flow is output from the Search Educational Content (4.1) process to the Users entity (E1), which occurs when learning resources are provided to users after searching the database.
  - ○ F7: Information Request
    - ■ This data flow is input from the Users entity (E1) to the Access Educational Content (4.3) process, which occurs when users request to view the learning resources provided by the database.
  - ○ F8: Information Delivered
    - ■ This data flow is output from the Access Educational Content (4.3) process to the Users entity (E1), which occurs when learning resources are delivered to users from the database.

Process 5: Level 1

System Administration
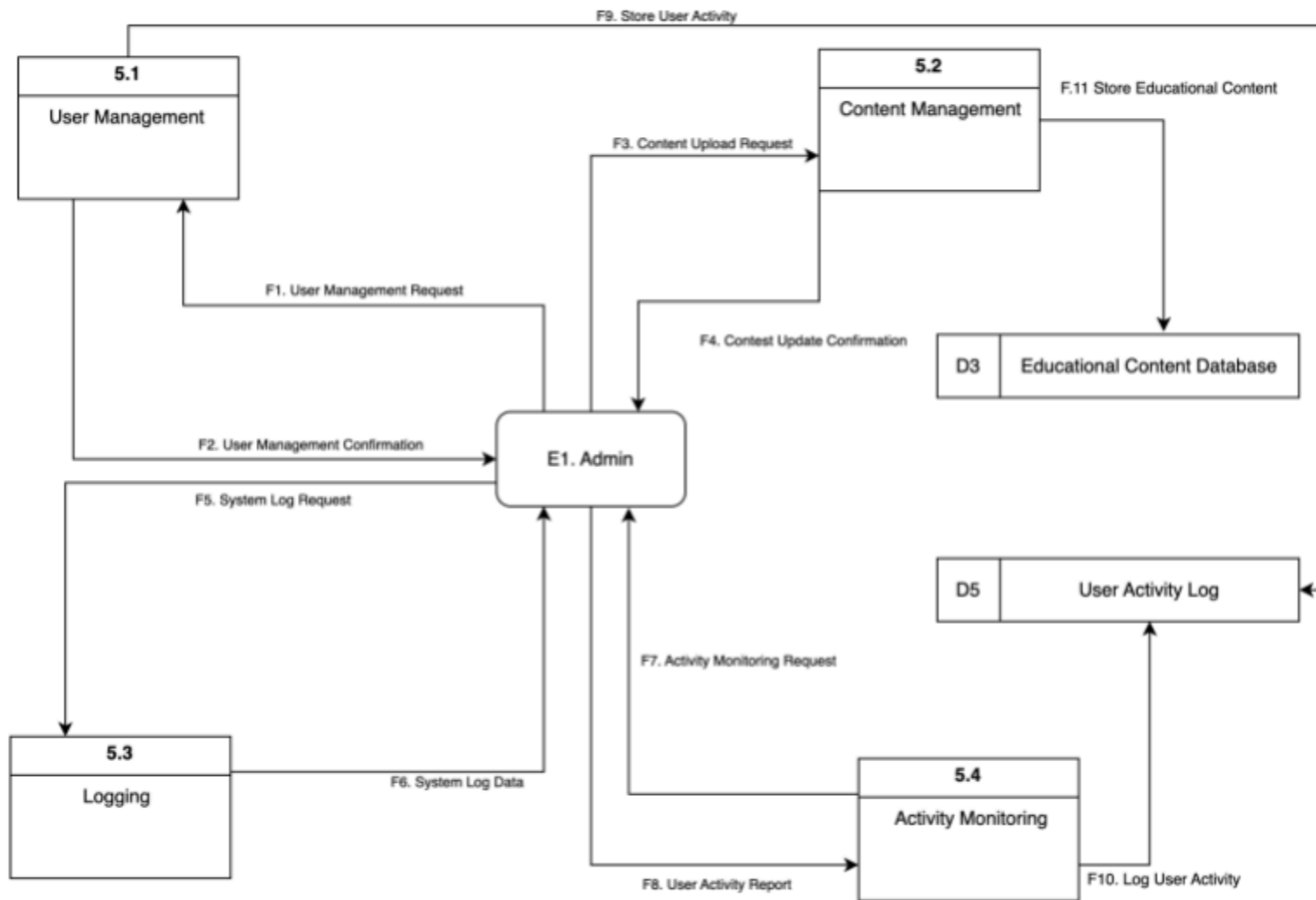
Text Definitions:

- Processes:
  - Process 5: System Administration
    - The System Administration process takes inputs exclusively from the Admin entity and manages essential system functions. It interacts with the User Activity Log (D5) and Educational Content Database (D3) data stores to handle user account actions, manage educational content, retrieve system logs, and monitor user activities. Outputs from this process include confirmations and activity reports, sent back to Admin.
  - Process 5.1: User Management
    - This process handles user-related requests from Admin, such as adding, updating, or deleting user profiles, and interacts with the User Activity Log (D5) to store user activities. Outputs include confirmation messages to Admin.
  - Process 5.2: Content Management
    - This process manages uploading and updating educational content, interacting with the Educational Content Database (D3) to store new and revised resources. Outputs include confirmation of content uploads or updates to Admin.
  - Process 5.3: Logging
    - This process retrieves system logs upon Admin's request, providing activity reports for monitoring purposes.
  - Process 5.4: Activity Monitoring
    - This process tracks and logs user actions in the system for accountability, storing data in the User Activity Log (D5). Outputs include activity reports sent to Admin.
- Entities:
  - E1: Admin
    - This entity represents administrators who are responsible for all management actions, including user administration, content management, logging, and activity monitoring.
- Datastores:
  - D3: Educational Content Database
    - This datastore stores all educational content for system use, supporting Content Management processes.
  - D5: User Activity Log
    - This datastore records user actions like logins, profile updates, and other activities, supporting User Management and Activity Monitoring processes.
- Data Flows:
  - F1: User Management Request
    - This data flow is input from Admin (E1) to User Management (Process 5.1), initiating actions like adding, updating, or deleting a user.
  - F2: User Management Confirmation

- ■ This data flow is output from User Management (Process 5.1) to Admin (E1), providing confirmation of the requested action (e.g., user added, updated, or deleted).
- ○ F3: Content Upload Request
  - ■ This data flow is input from Admin (E1) to Content Management (Process 5.2), indicating a request to upload or update educational content.
- ○ F4: Content Update Confirmation
  - ■ This data flow is output from Content Management (Process 5.2) to Admin (E1), confirming that the content was successfully uploaded or updated.
- ○ F5: System Log Request
  - ■ This data flow is input from Admin (E1) to Logging (Process 5.3), requesting system logs for review.
- ○ F6: System Log Data
  - ■ This data flow is output from Logging (Process 5.3) to Admin (E1), containing the requested system logs.
- ○ F7: Activity Monitoring Request
  - ■ This data flow is input from Admin (E1) to Activity Monitoring (Process 5.4), requesting an activity report or monitoring of user activities.
- ○ F8: User Activity Report
  - ■ This data flow is output from Activity Monitoring (Process 5.4) to Admin (E1), providing details on recent user activities for review.
- ○ F9: Store User Activity
  - ■ This data flow is output from User Management (Process 5.1) to the User Activity Log (D5), storing records of user actions, such as logins and profile changes.
- ○ F10: Log User Activity
  - ■ This data flow is output from Activity Monitoring (Process 5.4) to the User Activity Log (D5), maintaining a log of ongoing user activities in the system.
- ○ F11: Store Educational Content
  - ■ This data flow is output from Content Management (Process 5.2) to the Educational Content Database (D3), storing new or updated educational content.

Student roles and contributions

Project Manager: Wardan Kazzalbach (kj31410@umbc.edu)
- Wardan worked on the context diagram and the level 1 diagram. He also mitigated the workload throughout the team, posted tasks on our GitHub project board, and led each meeting during this deliverable.

Frontend Developer/Lead Designer: Izzie Powell (ipowell2@umbc.edu)
- Izzie worked on the Educational Content Delivery and Portfolio Management level 1 diagrams and definitions for these diagrams.

Backend Developer: Alejandro Diaz Meinecke (alejand4@umbc.edu)
- Alejandro worked on the User Management level 1 process diagram and its definitions of them.

Database/Network Developer: Kidus Solomon (kiduss1@umbc.edu)
- Kidus worked on the Simulated Stock Trading level 1 process diagram and its definitions of them.

Systems Analyst: Arham Ahmed (kahmed4@umbc.edu)
- Arham worked on the level 1 System Administration process diagram and its definitions of them.

Scheduled Meeting Time Outside of Class:
a. Bi-weekly Meetings: Our team has been meeting every other Friday at noon to discuss project progress, address any challenges, and plan upcoming tasks for any deliverables or requirements we have.
b. Communication Platform: We have been using Discord as our primary communication channel to share updates, coordinate class deliverables, and ensure everyone understands what needs to be done.
c. Task Allocation: Responsibilities have been assigned based on each member's roles, and we have been collaborating to complete tasks efficiently and within the set timelines. We also have been helping each other out when needed.

Project Contact Person: Wardan Kazzalbach
UMBC Gmail: KJ31410@umbc.edu
GitHub Link: https://github.com/users/wkazz246/projects/4/views/1