Project Title: Shahid

Supervisor TA: Dalia Elalfy

Team ID: 48

Team Members:

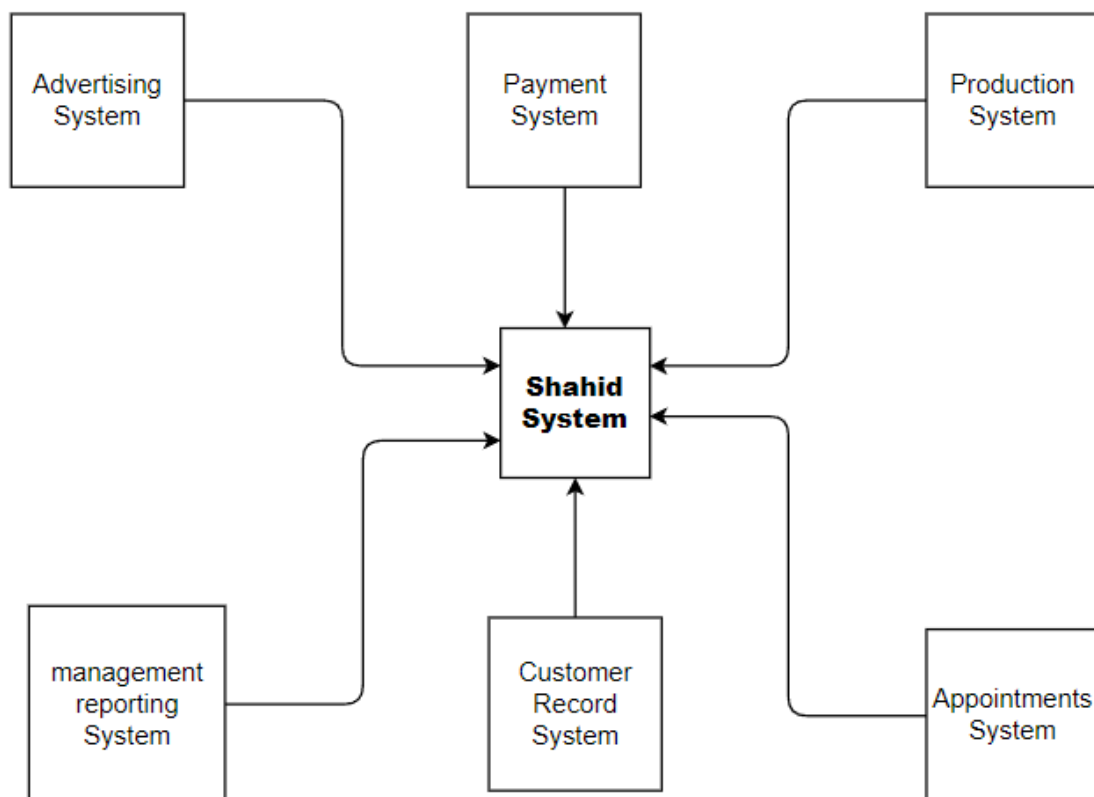| Name	ID	Department	Section
عصام الدين شريف جاب الله حسن	20191700387	CS	3
عبد الله رمضان السيد ابراهيم	20191700353	CS	2
حسن وليد حسن	20191700215	CS	2
عبد الرحمن سعيد ذكي	20191700363	general	17
هناء أشرف رزق محمد	20191700739	IS	6
جنه محمد احمد عبده	20191700198	SC	2

➢ software requirements specification (SRS)

Introduction:

A platform for showing movies, series and TV shows, and a section dedicated to children, with a monthly and annual subscription.

The system includes many functions, the most important of which are the video streaming service, the easy payment process, the protected authentication system, and the search process.

Increasing the organization's profit and facilitating the user's access to television content are among the strategic goals of the platform and Shahid VIP has been entering into strategic partnerships with relevant companies to increase accessibility.

Context Diagram:

User Requirements:

1) Easily search for movies and TV shows
2) The user chooses the quality of viewing he wants
3) user friendly interface (simple ui/ux)
4) The recommendation system is based on user choices
5) video content management
6) videos available on offline mode
7) security and privacy

Functional Requirements:

Register Function:

- Description: Allow visitors to Register to the community platform, Registration should request visitors to Enter their information.
- Inputs: Email, Password, age, Phone number
- Source: User
- Pre-condition: Open Register Form
- Post-condition: Send Confirmation message to the Email or Phone number, Store account in DB
- Output: Go to home page

Log in Function:

- Description: Entry page to a website that requires user identification and authentication
- Inputs: Email, Password
- Source: Database
- Pre-condition: Open Log in Form
- Post-condition: Check Validation of inputs
- Output: Go to home page

Search Function:

- Description: Providing a search function that searches your Web pages is a design strategy that offers users a way to find content.
- Inputs: movies name or series name or TV shows name
- Source: User
- Pre-condition: Open Search text box
- Post-condition: if request is valid go to details page with selected item, else print not found message
- Output: Matched results or not found message

Purchase Package Function:

- Description: Payment processing is how businesses complete credit card and debit card transactions, services expedite card transactions, and payment gateways securely transmit data so money from a customer's issuing bank can be transferred to a merchant's account.

- Inputs: select package then select payment method, user information
- Source: User, Database
- Pre-condition: User must register to system
- Post-condition: if request is valid go to video Player page with selected item and send confirmation message to user and save process in database, else print Error message for user
- Output: go to video Player page or print Error message

Display Video Function:

- Description: control all the video features such as Playback speed, Subtitles, Quality and Screen-size
- Inputs: Click on Subcategory and then on video
- Source: User
- Pre-condition: System will check in database for Video related to that category and user have account and available
- Post-condition: add video to his Recommend list for Recommendation system.
- Output: Video related to that category will be displayed to user

Add to Mylist Function:

- Description: help user to easily retrieve the best videos that he loves and helps the recommendation system to display similar topics
- Inputs: Click on add to Mylist
- Source: User
- Pre-condition: user make login in system
- Post-condition: System will add the video to your Mylist.
- Output: Confirmation message to user

Feedback Function:

- Description: The organization helps determine the quality of the product and the extent of customer satisfaction
- Inputs: Name, Address, Contact number and feedback
- Source: User
- Pre-condition: User must register to system
- Post-condition: Save feedback into database
- Output: Display Feedback successfully submitted message

Cancel the subscription Function:

- Description: In the event of any reason, the user will terminate the service
- Inputs: Cancel request
- Source: User
- Pre-condition: User must register to system and have package
- Post-condition: Canceling the package and deleting the package
- Output: Package cancelled successfully

Non-Functional Requirements:

Performance Requirement:

- If video is not there, then system should display related videos.
- The new videos will be uploaded each day.
- Each page must load within 2 seconds.

Maintainability Requirement:

- Only maintainers are allowed to add new videos to system
- installation of a new version shall leave all database contents and all personal settings unchanged

Security Requirements:

- Users are advised to change their Password regular basis.
- The maximum number of attempts to enter a Password will be three.

Usability Requirements:

- **users should be able to complete the main actions once they see the interface**
- users can reach their goals quickly
- the design is pleasant to use

Reliability Requirements:

- The system defect rate shall be less than 1 failure per 1000 hours of operation.
- No more than 1 per 1000000 transactions shall result in a failure requiring a system restart.

Response time Requirements:

- The System must be able to search fast as possible in the database
- login process and registration process should take at max 5 second

Testability Requirements:

- The system shall include unit tests that ensure 100% coverage (requirements coverage, code coverage, payment process coverage).
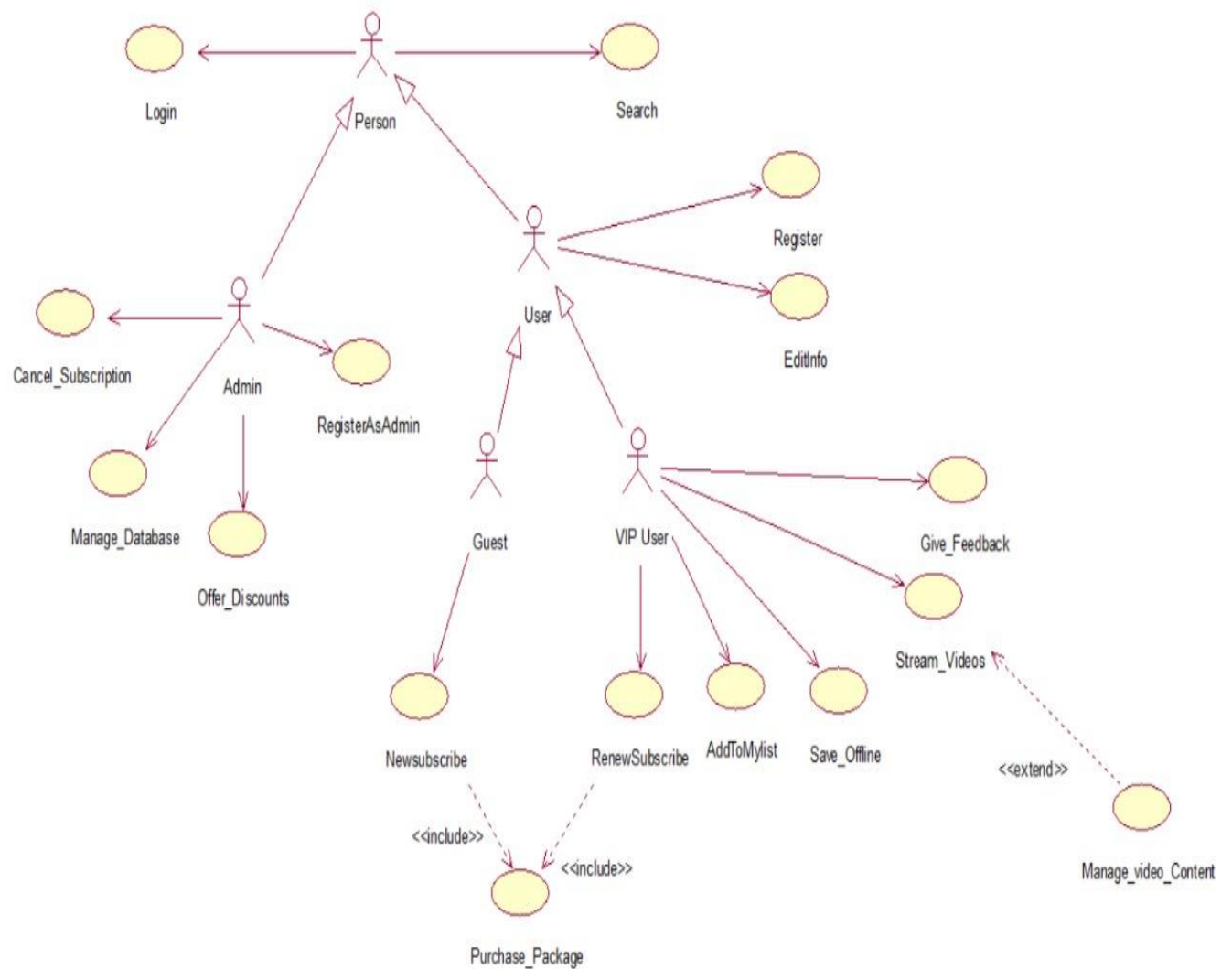
Availability Requirements:

- shahid is available 24 hours a day with all features.

Database Requirement:

- The System must be able to store all kinds of data. The System should be frequently connectable to a database

➢ **Use case diagram**

➢ **Sequence diagram**

Sequence diagram with lifelines: Customer, Payment Site, Processes, Bank Account, Database.

- Customer → Payment Site: click on buy the package button
- Payment Site → Processes: registrate order()
- Processes → Database: save order()
- Database --> Processes: return state()
- Processes --> Payment Site: return state()
- Customer → Payment Site: Make payment()
- Payment Site → Processes: Make payment()
- Processes → Bank Account: Make payment()
- Bank Account --> Processes: Enter card number()
- Processes --> Payment Site: Enter card number()
- Customer → Payment Site: Enter card number()
- Payment Site → Processes: set card number()
- Processes → Bank Account: set card number()
- Bank Account --> Processes: Enter password()
- Processes --> Payment Site: Enter password()
- Customer → Payment Site: Enter password()
- Payment Site → Processes: set password()
- Processes → Bank Account: set password()
- Bank Account: check card details
- Bank Account --> Processes: return state()

alt
[state == true & balance >= amounts]
- Processes → Bank Account: Make payment()
- Bank Account: update_credit_card()
- Bank Account --> Processes: done()
- Processes → Database: set order and payment status()
- Processes --> Payment Site: Confirmation process()

[state == true & balance < amounts]
- Bank Account --> Processes: balance_is_not_enough()
- Processes --> Payment Site: Failed process()

[state == false]
- Bank Account --> Processes: Wrong_card_info()
- Processes --> Payment Site: check_card_data()
- Processes --> Payment Site: Failed process()