Project Planning Phase

Project Planning Template (Product Backlog, Sprint Planning, Stories, Story points)

Date	18 October 2023
Team ID	31784ECADC48A50996140F3D510D5CF5
Project Name	Subscribers Galore: Exploring the World's Top YouTube Channels
Maximum Marks	8 Marks

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	Create a Registration Form: Design a user-friendly registration form that includes the following fields: Email Password Confirm Password Front-end Development: Use HTML, CSS, and JavaScript to create the registration form. You can use a framework like React, Angular, or Vue.js for a more dynamic and responsive user interface. Database Storage: Store the user's registration data in a database. You can use SQL databases (e.g., MySQL, PostgreSQL) or NoSQL databases (e.g., MongoDB). Securely hash the password before storing it. Security Considerations: Hash and salt user passwords for security. Use HTTPS for secure data transmission. Implement rate limiting and account lockout	2	High	

			mechanisms to protect against brute force attacks. Apply security best practices to protect against SQL injection and other vulnerabilities. Testing: Thoroughly test the registration process, both for valid and invalid inputs.			
Sprint-1		USN-2	Email Sending Service: Choose and set up an email sending service or SMTP server to send emails. Popular options include SendGrid, Amazon SES, or your own SMTP server. Email Template: Create an email template for the confirmation email. This template should include a link for the user to confirm their email address. Generate a Confirmation Token: When a user registers, generate a unique confirmation token for that user. This token should be associated with the user's account in your database. Include the Token in the Confirmation Link: In the confirmation email template, include the generated token in the link's URL. This token will be used to identify and confirm the user. Sending the Confirmation Email: After a user successfully registers, send the confirmation email to the provided email address using your chosen email sending service. Make sure to include the confirmation link with the token. Email Confirmation Endpoint: Create an endpoint in your application (e.g., /confirm) to handle email confirmation. This endpoint should validate the token and mark the user's account as confirmed in your database.	1	High	

			Token Validation: When a user clicks the confirmation link, your server should validate the token. Ensure that the token is not expired and that it matches a record in your database. Update User Status: If the token is valid, update the user's status in your database to mark their email as confirmed. You can also set other necessary account details (e.g., active status).			
Sprint-2		USN-3	Set Up a Facebook Developer Account: To integrate Facebook registration, you'll need to create a Facebook Developer account and set up a new app. Create a Facebook App: In the Facebook Developer portal, create a new app and configure its basic settings. Make sure to provide a valid privacy policy URL and terms of service URL. Implement the Facebook Login Button: Add a "Log in with Facebook" button on your registration page. When users click this button, they'll be prompted to log in to their Facebook accounts and grant permissions for your app. Handle User Data: Once a user logs in with Facebook, you'll receive user data such as the user's email, name, and Facebook ID. You can use this data to create a new user profile in your application's database. Link Facebook Account to User Account: If the user's Facebook email matches an existing account in your system, you can offer to link the Facebook account with the existing one. Ensure that users can later log in using both methods (email/password and Facebook).	2	Low	

			Store User Data Securely: Handle user data with care and store it securely. Follow best practices for data protection and user privacy. Email Confirmation (Optional): If you require email confirmation for all users, you can still send a confirmation email to the email address provided by Facebook. Error Handling: Handle cases where Facebook registration fails or user data isn't available. Account Security: Implement security measures, such as account recovery options and two-factor authentication, to protect user accounts.		
Sprint-1		USN-4	Set Up a Google Developer Account: To integrate Gmail registration, you'll need to create a Google Developer account and set up a new project. Create a Google Cloud Project: In the Google Developer Console, create a new project and configure its settings. Enable the Gmail API: In your Google Cloud project, enable the Gmail API. This will allow your application to access Gmail for registration purposes. Obtain Google OAuth Client Credentials: After enabling the Gmail API, you'll receive	2	Medium	

| | | | OAuth 2.0 client credentials (Client ID and Client Secret). These credentials will be used to authenticate your app with Google.

Integrate Google Sign-In:
To enable users to register through Gmail, you'll need to integrate Google Sign-In into your application. You can use Google Sign-In for Web for web applications or the Google Sign-In SDK for mobile apps.

Add the "Sign in with Google" Button:
Include a "Sign in with Google" button on your registration page. When users click this button, they'll be prompted to log in to their Gmail accounts and grant permissions for your app.

Access User Data:
After a user signs in with their Gmail account, your app can receive their basic profile information (e.g., email, name) and a unique user ID. You can use this data to create a new user profile in your application's database.

Link Gmail Account to User Account:
If the user's Gmail email matches an existing account in your system, you can offer to link the Gmail account with the existing one. Ensure that users can later log in using both methods (Gmail and email/password).

Securely Store User Data:
Handle user data with care and store it securely. Follow best practices for data protection and user privacy.

Email Confirmation (Optional):
If you require email confirmation for all users, you can still send a confirmation email to the Gmail address provided by Google.

Error Handling:
Handle cases where Gmail registration fails or | | | |
|---|---|---|---|---|---|---|

			user data isn't available. Provide informative error messages to users. Privacy and Permissions: Request only the necessary permissions from Google and respect user privacy. Explain why you need these permissions to users during the registration process. Testing and Debugging: Thoroughly test the Gmail registration feature to ensure it works as expected and handles errors gracefully. User Feedback and UX: Provide feedback to users during the registration process, indicating the progress and any issues that may arise. Compliance with Google Policies: Ensure that your app complies with Google's policies and guidelines for app integration. Account Security: Implement security measures, such as account recovery options and two-factor authentication, to protect user accounts.			
Sprint-1	Login	USN-5	Create a Login Form: Design a user-friendly login form that includes the following fields: Email Password Front-end Development: Use HTML, CSS, and JavaScript to create the login form. You can use a front-end framework like React, Angular, or Vue.js for a more dynamic and responsive user interface. Client-Side Validation: Implement JavaScript code for client-side validation. Verify that the email is in a valid format and check for any other client-side validation needs.	1	High	

			Server-Side Authentication: On the server-side, implement authentication logic. Verify the user's email and password against your database records. If a user with the provided email and password is found, authenticate the user. Password Hashing: For security, hash and salt user passwords before storing them in the database. When authenticating users, hash the entered password and compare it with the stored hash. Session Management: Implement session management to keep the user logged in. You can use JSON Web Tokens (JWT), cookies, or other methods to manage sessions.			
	Dashboard		Front-end Development: Use HTML, CSS, and JavaScript to create the login form. You can use a front-end framework like React, Angular, or Vue.js for a more dynamic and responsive user interface.			

Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2023	29 Oct 2023	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2023	05 Nov 2023		
Sprint-3	20	6 Days	07 Nov 2023	12 Nov 2023		
Sprint-4	20	6 Days	14 Nov 2023	19 Nov 2023		