

PROMPTSAGE

ARCHITECTURE AND API
DOCUMENTATION

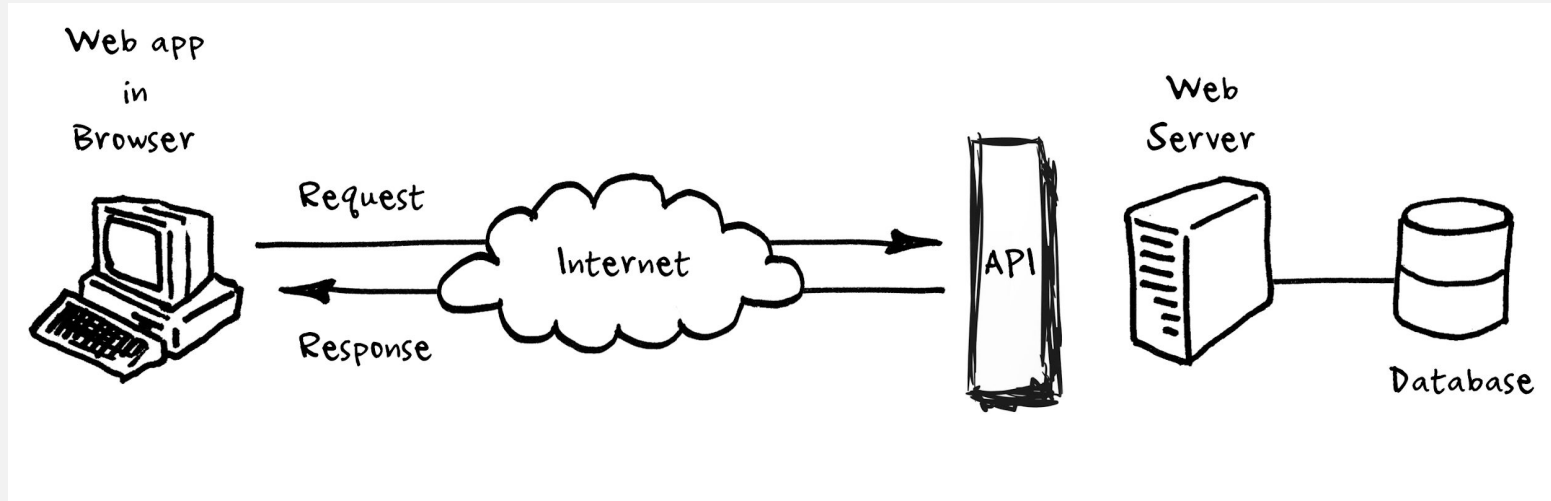
Group-02

Submitted by-

1805095 -Mahbubul Haque

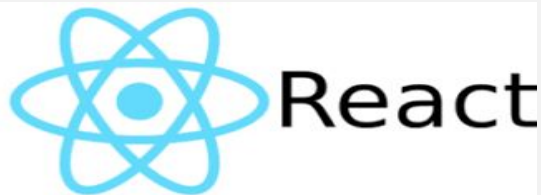
1805118 -Arif Faisal

ARCHITECTURE



TECH STACK

□ Frontend:



□ Backend:



□ Database:



1.1.Creating User Profile

API End-Point	HTTP Method	Request Body	Response Body
api/users	POST	<pre>{ "username": "john_doe", "email": "john@example.com", "password": "securepassword123", "first_name": "John", "last_name": "Doe", "dob": "1990-05-15", // Additional profile information... }</pre>	<pre>HTTP/1.1 201 CREATED { "status": "success", "data": { "user_id": "user123", "username": "john_doe", "email": "john@example.com", "first_name": "John", "last_name": "Doe", "dob": "1990-05-15", // Additional profile information... "created_at": "2023-03-10T08:00:00Z", "message": "User profile created successfully." } }</pre>

1.2. Getting user information for Profile

API End-Point	HTTP Method	Request Body	Response Body
api/users/{userid}	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "id":1 "rank":2, "followers": 256, "joined_date": "2023-01-01", "username": "Travis", "email": "travis@gmail.com", "bio": "Hi! I'm a prompt engineer in GPT.", } }

1.3. Getting user monthly-stats of sales and earning for Profile

API End-Point	HTTP Method	Request Body	Response Body
api/users/{userid}/monthly-stats	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "monthly_stats": [{ "month": "January", "year": 2023, "sales": 25, "earnings": 5000.00 }, { "month": "February", "year": 2023, "sales": 30, "earnings": 6000.00 },] } } }

1.4. Getting user created prompts for Profile

API End-Point	HTTP Method	Request Body	Response Body
api/users/{userid}/created-prompts	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "prompts": [{ "id": 1, "title": "Merry Christmases From Your Pets", "picture": https://a.com/images/1.jpg "platform": "DALL-E" }, { "id": 2, "title": "Vintage Christmas Card Illustrations", "picture": https://a.com/images/2.jpg "platform": "DALL-E" }] } }

1.5. Getting user bought prompts for Profile

API End-Point	HTTP Method	Request Body	Response Body
api/users/{userid}/bought-prompts	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "prompts": [{ "id": 1, "title": "Merry Christmases From Your Pets", "description": "Enjoy festive joy with 'Merry Christmas from Your Pets' prompts. " "category": "Photography", "platform": "DALL-E" "created_at": "2023-02-15T12:30:45Z", "updated_at": "2023-02-16T09:45:00Z" }, { //Other prompts }] } } }

1.6. Edit user Profile

API End-Point	HTTP Method	Request Body	Response Body
api/users/{userid}/	PUT	{ "email": "new_email@example.com", "bio": "Updated bio info" }	HTTP/1.1 201 CREATED { "status": "success" }
			HTTP/1.1 204 NO CONTENT { "status": "error" }

1.7. Delete a prompt by user

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/users/{user_id}/prompts/{prompt_id}</code>	DELETE		HTTP/1.1 201 CREATED { "status": "success", "data": { "message": "Prompt deleted successfully." } }

2.1. Getting Popular prompts for Home Page

API End-Point	HTTP Method	Request Body	Response Body
api/homepage/popular-prompts	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "prompts": [{ "id": 1, "title": "Merry Christmases From Your Pets", "description": "Enjoy festive joy with 'Merry Christmas from Your Pets' prompts. " "category": "Photography", "platform": "DALL-E", "likes": 100, }, { //Other prompts }] } }

2.2. Getting Featured prompts for Home Page

API End-Point	HTTP Method	Request Body	Response Body
api/homepage/featured-prompts	GET	<pre>{ }</pre>	<pre>HTTP/1.1 200 OK { "status": "success", "data": { "prompts": [{ "id": 1, "title": "Merry Christmases From Your Pets", "description": "Enjoy festive joy with 'Merry Christmas from Your Pets' prompts. " "category": "Photography", "platform": "DALL-E" }, { //Other prompts }] } }</pre>

3.1. Getting prompts for Market-Place Page

API End-Point	HTTP Method	Request Body	Response Body
api/marketplace/prompts	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "prompts": [{ "id": 1, "title": "Merry Christmases From Your Pets", "description": "Enjoy festive joy with 'Merry Christmas from Your Pets' prompts. " "category": "Photography", "platform": "DALL-E" }, { //Other prompts }] } }

4.1. Input from user to generate prompts for Generate Page

API End-Point	HTTP Method	Request Body	Response Body
api/generate-prompt	POST	<pre>{ "input_text": "User's input text for generating a prompt." }</pre>	<pre>HTTP/1.1 201 CREATED { "status": "success", "data": { "generated_prompt": "Generated prompt text based on user input." } }</pre>

4.2. Generate Image from prompt for Generate Page

API End-Point	HTTP Method	Request Body	Response Body
api/generate-image	POST	<pre>{ "prompt_text": "Generated prompt text based on user input." }</pre>	<pre>HTTP/1.1 201 CREATED { "status": "success", "data": { "generated_image_url": "https://example.com/images/generated_image.jpg" } }</pre>

5.1. Getting Top Mid-Journey prompt Engineers for Hire page

API End-Point	HTTP Method	Request Body	Response Body
api/hire/mid-journey	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "top_engineers": [{ "id":1, "name":"John Doe", "pro_pic": " https://a.com/images/1.jpg " }, { "id":2, "name":"travis", "pro_pic": " https://a.com/images/2.jpg " },] } }

5.2. Getting Top DALL-E prompt Engineers for Hire page

API End-Point	HTTP Method	Request Body	Response Body
api/hire/dall-e	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "top_engineers": [{ "id": 1, "name": "John Doe", "pro_pic": " https://a.com/images/1.jpg " }, { "id": 2, "name": "travis", "pro_pic": " https://a.com/images/2.jpg " },] } }

5.3. Getting Top Stable Diffusion prompt Engineers for Hire page

API End-Point	HTTP Method	Request Body	Response Body
api/hire/stable-diffusion	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "top_engineers": [{ "id": 1, "name": "John Doe", "pro_pic": " https://a.com/images/1.jpg " }, { "id": 2, "name": "travis", "pro_pic": " https://a.com/images/2.jpg " },] } }

6.1. Getting info of travis for Hire-travis page

API End-Point	HTTP Method	Request Body	Response Body
api/user/{user id}/hire/{user id}	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "id":2 "rank":2, "followers": 256, "joined_date": "2023-01-01", "username": "Travis", "email": "travis@gmail.com", "bio": "Hi! I'm a prompt engineer in GPT.", } }

6.2. Getting prompts of travis for Hire-travis page

API End-Point	HTTP Method	Request Body	Response Body
api/user/{user id}/hire/{user id}/prompts	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "prompts": [{ "id": 1, "title": "Merry Christmases From Your Pets", "picture": https://a.com/images/1.jpg "platform": "DALL-E" }, { "id": 2, "title": "Vintage Christmas Card Illustrations", "picture": https://a.com/images/2.jpg "platform": "DALL-E" }] } }

6.3. Getting conversation with travis for Hire-travis-chat page

API End-Point	HTTP Method	Request Body	Response Body
api/user/{user id}/hire/{user id}/chat	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "chat_history": [{ "sender_id": 101, "receiver_id": 102, "message": "Hi there!", "timestamp": "2023-05-01T09:30:00Z" }, { "sender_id": 102, "receiver_id": 101, "message": "Hello! How can I help you today?", "timestamp": "2023-05-01T09:35:00Z" }] } }

6.4. Sending text to travis for Hire-travis-chat page

API End-Point	HTTP Method	Request Body	Response Body
api/user/{user id}/hire/{user id}/chat	POST	<pre>{ "receiver_id": 102, "message": "This is a test message." }</pre>	<pre>HTTP/1.1 201 Created { "status": "success" }</pre>

6.5. Getting Info about Amount for Hire-travis-payment page

API End-Point	HTTP Method	Request Body	Response Body
api/user/{user id}/hire/{user id}/payment-info	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "id":2 "username": "Travis", "amount_per_prompt": 120 } }

7.1. Getting ChatList with users for Chat page

API End-Point	HTTP Method	Request Body	Response Body
api/user/{user id}/chat	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "chat_list": [{ "user_id": 102, "username": "JaneDoe", "last_message": "Hello!", "last_message_timestamp": "2023-05-01T09:35:00Z" }, { "user_id": 103, "username": "JohnSmith", "last_message": "Sure, I can help with that.", "last_message_timestamp": "2023-05-02T14:20:00Z" }] } }

7.2. Getting Notifications for Notification page

API End-Point	HTTP Method	Request Body	Response Body
api/user/{user id}/notifications	GET	{ }	HTTP/1.1 200 OK { "status": "success", "data": { "notifications": [{ "id": 601, "message": "Your bought prompt 'abc' has added in your profile.", "timestamp": "2023-05-10T14:00:00Z" }, { "id": 602, "message": "Your prompt 'Graphic Design Project' has been featured.", "timestamp": "2023-05-11T09:30:00Z" }] } } }

8.1. Processing Payment for Payment page

API End-Point	HTTP Method	Request Body	Response Body
/api/payment/process	POST	<pre>{ "order_id": "abc123", "amount": 50.00, "payment_method": "credit_card", "card_details": { "card_number": "***** **** * 1234", "expiration_date": "MM/YYYY", "cvv": "123" } }</pre>	<pre>HTTP/1.1 200 OK { "status": "success", "transaction_id": "xyz789", "message": "Payment processed successfully." }</pre>

8.2. Get Transaction Details for Payment page

API End-Point	HTTP Method	Request Body	Response Body
/api/payment/transactions/{transaction_id}	GET	{ }	HTTP/1.1 200 OK { "transaction_id": "xyz789", "order_id": "abc123", "amount": 50.00, "timestamp": "2023-01-15T14:30:00Z", "status": "completed" }

8.3. Verify Payment Status for Payment page

API End-Point	HTTP Method	Request Body	Response Body
/api/payment/verify/{order_id}	GET	{ }	{ "order_id": "abc123", "status": "completed", "amount_paid": 50.00 }

8.4. User Wallet Integration for Payment page

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/payment/wallet/topup</code>	POST	<pre>{ "user_id": "user123", "amount": 100.00}</pre>	<pre>{ "wallet_balance": 150.00, "transaction_id": "topup123", "message": "Wallet successfully topped up."}</pre>

8.5. List Transactions for Payment page(User Side)

API End-Point	HTTP Method	Request Body	Response Body
/api/payment/user/{user_id}/transactions	GET	{ }	[{ "transaction_id": "xyz789", "order_id": "abc123", "amount": 50.00, "timestamp": "2023-01-15T14:30:00Z", "status": "completed" }, // Additional transactions...]

9.1. Create a post for Discussion Page

API End-Point	HTTP Method	Request Body	Response Body
/api/forum/posts	POST	<pre>{ "user_id": "user123", "title": "How to generate creative prompts?", "content": "I'm looking for tips on generating unique and creative prompts. Any suggestions?"}</pre>	<pre>{ "post_id": "post123", "timestamp": "2023-02-10T12:30:00Z", "message": "Post created successfully."}</pre>

9.2. Getting a post details for Discussion Page

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/forum/posts/{post_id}</code>	GET	<pre>{</pre>	<pre>{ "post_id": "post123", "user_id": "user123", "title": "How to generate creative prompts?", "content": "I'm looking for tips on generating unique and creative prompts. Any suggestions?", "timestamp": "2023-02-10T12:30:00Z", "replies_count": 3 }</pre>

9.3. Getting Lists of the posts for Discussion Page

API End-Point	HTTP Method	Request Body	Response Body
/api/forum/posts/	GET	{ }	[{ "post_id": "post123", "user_id": "user123", "title": "How to generate creative prompts?", "timestamp": "2023-02-10T12:30:00Z", "replies_count": 3 }, // Additional posts...]

9.4. Reply to a post for Discussion Page

API End-Point	HTTP Method	Request Body	Response Body
/api/forum/posts/{post_id}/replies	POST	<pre>{ "user_id": "user456", "content": "One approach is to..." }</pre>	<pre>{ "reply_id": "reply789", "timestamp": "2023-02-11T09:45:00Z", "message": "Reply posted successfully." }</pre>

9.5. Get Reply to a post for Discussion Page

API End-Point	HTTP Method	Request Body	Response Body
/api/forum/posts/{post_id}/replies	GET	{ }	[{ "reply_id": "reply789", "user_id": "user456", "content": "One approach is to...", "timestamp": "2023-02-11T09:45:00Z" }, // Additional replies...]

9.6. Upvote to a post for Discussion Page

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/forum/posts/{post_id}/upvote</code>	POST	<pre>{ }</pre>	<pre>{ "upvotes": 15, "message": "Post upvoted successfully." }</pre>

9.7. Downvote to a post for Discussion Page

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/forum/posts/{post_id}/downvote</code>	POST	<pre>{ }</pre>	<pre>{ "downvotes": 5, "message": "Post downvoted successfully." }</pre>

9.8. Search a post using keyword for Discussion Page

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/forum/posts/search?q=keyword</code>	GET	<pre>{ }</pre>	<pre>[{ "post_id": "post123", "user_id": "user123", "title": "How to generate creative prompts?", "timestamp": "2023-02-10T12:30:00Z", "replies_count": 3 }, // Additional posts matching the search...]</pre>

10.1. Admin (View pending prompts)

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/admin/prompts/pending</code>	GET	<pre>{ }</pre>	<pre>HTTP/1.1 200 OK { "status": "success", "data": [{ "prompt_id": "prompt123", "user_id": "user456", "content": "A unique prompt submission.", "timestamp": "2023-02-15T09:30:00Z" }, // Additional pending prompts...] }</pre>

10.2. Admin (Approve or reject a prompt submitted by a user)

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/admin/prompts/{prompt_id}/approval</code>	PUT	<pre>{ "approval_status": "approved", "admin_comment": "Prompt meets quality standards." }</pre>	<pre>HTTP/1.1 200 OK { "status": "success", "message": "Prompt approved successfully." }</pre>

10.3. Admin (View reports on user activity and behaviour)

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/admin/reports/users</code>	GET	<pre>{ }</pre>	<pre>HTTP/1.1 200 OK { "status": "success", "data": [{ "user_id": "user789", "reports": [{ "report_type": "inappropriate_content", "description": "User posted inappropriate content.", "timestamp": "2023-02-20T15:45:00Z" }, // Additional user reports...] }, // Additional user reports...] }</pre>

10.4. Admin (Suspend user)

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/admin/users/{user_id}/suspend</code>	PUT	<pre>{ "reason": "Multiple violations of content guidelines." }</pre>	<pre>HTTP/1.1 200 OK { "status": "success", "message": "User suspended successfully." }</pre>

10.5. Admin (View transaction history)

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/admin/transactions</code>	GET	<pre>{ }</pre>	<pre>HTTP/1.1 200 OK { "status": "success", "data": [{ "transaction_id": "txn789", "user_id": "user123", "amount": 50.00, "timestamp": "2023-02-25T18:20:00Z" }, // Additional transactions...] }</pre>

10.6. Admin (get system stats for monitoring)

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/admin/stats</code>	GET	<pre>{</pre>	<pre>HTTP/1.1 200 OK { "status": "success", "data": { "total_users": 1000, "active_prompts": 500, "total_transactions": 2000, // Additional system statistics... } }</pre>

10.7. Admin (manage/modify VIP memberships details)

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/admin/memberships/{membership_id}</code>	PUT	<pre>{ "benefits": ["priority_support", "exclusive_prompts"], "duration_months": 12 }</pre>	<pre>HTTP/1.1 200 OK { "status": "success", "message": "VIP membership updated successfully." }</pre>

10.8. Admin (create system announcement)

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/admin/announcements</code>	POST	<pre>{ "title": "Scheduled Maintenance", "content": "Our platform will undergo maintenance on...", "start_time": "2023-03-01T03:00:00Z", "end_time": "2023-03-01T06:00:00Z" }</pre>	<pre>HTTP/1.1 201 CREATED { "status": "success", "data": { "announcement_id": "announcement456", "timestamp": "2023-02-28T14:00:00Z", "message": "Announcement created successfully." } }</pre>

10.9. Admin (Manage user roles)

API End-Point	HTTP Method	Request Body	Response Body
<code>/api/admin/users/{user_id}/roles</code>	PUT	<pre>{ "roles": ["admin", "moderator"] }</pre>	<pre>HTTP/1.1 200 OK { "status": "success", "message": "User roles updated successfully." }</pre>

THANK YOU