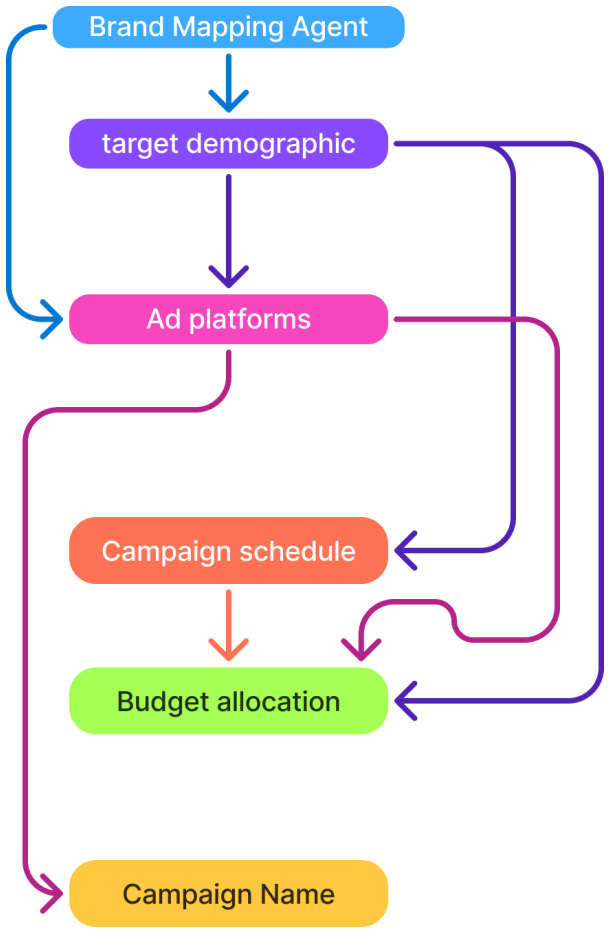
Agentic Technical Flow

- Utilizes CrewAI framework for building collaborative and cohesive Agentic System
- Comprises of the following Agents in the following Crew:
- Campaign Manager Flow
    - Campaign Setup Crew
        - Brand Mapping Agent
        - Audience Identification Agent
        - Channel Recommendation Agent
        - Schedule Date Recommendation Agent
        - Budget Allocation Agent
        - Campaign Name Generator Agent
    - Validator Crew
        - In process.

Utilizes Flow based workflow whereby setting output state for cross crew data management. Inbuilt Chromadb vdb for storing short term and contextual data (Planning to proceed in this fashion). Structured outputs implemented using pydantic model typing.

Agentic Workflow (In terms of input and output flow):



Crew : Campaign Planning

Agent 1: Brand Mapping

Purpose: it will take user inputs regarding a brand to classify the industry type the brand belongs to.

Inputs:

Compulsory inputs (will get every time):

1. Brand name (brand onboarding)
2. Brand description
3. Product Name and Description

Optional inputs

1. Brand Historical data (merged_age,gender_final, Merged_locaton_final ,merged_ads_final)

Tasks:

If Historical Data is present:

1. Use the existing Mapping for the brand
2. function (python code)

-input/output/fallback case/exception handling with error code

If Historical Data is Not present:

1. Predict industry type using a LLM tool intelligently classifying it.

Example Output :

"Food and Beverages", "Beauty Products" ..etc

Outputs To be used in(imp) :

Agent 2: Audience Identification

Agent 3: Channel Recommendation

Agent 2: Audience Identification

Purpose: Analyzes brand information and industry classification to identify key audience segments (e.g., demographics, interests, behaviors)

Inputs:

Compulsory inputs (will get every time):

1. Brand Name
2. Campaign objective
3. Brand Description
4. Brand Industry Type (**From the output of Agent 1: Brand Mapping)**

Optional inputs

1. Brand Historical data (merged_agegender_final, Merged_location_final ,merged_ads_final)

Tasks:

If Historical Data is present: (From testing, seems to be that LLM prediction are accurate)

1. Get audience age, gender, location, and interests from top-performing campaigns and map it. **(**Model not available currently**)**

    1. function (python code calling a model for analysis and identification of best demographic config from past campaign performances)

        – data gather/label/data verification/model/inference /host/API

If Historical Data is not Present:

1. Call an LLM to pass the brand category and description and ask for the best age, gender, and region for the campaign.

    1. The tool (LLM) will take brand details(Name, description, and industry type) and give the best age group, gender(s), location, and interest group. (Tool not available currently)

– Specify prompt template

Example Output :

`
Age Group Range: 25 to 34
Gender: Male, Female and Others
Region: Bangalore, Karnataka, India
Interest group: Fashion Enthusiasts

Outputs To be used in(imp) :

Agent 3: Channel Recommendation
Agent 4: Campaign Schedule Recommendation

Agent 5: Budget Recommendation
Crew : Creative Generation
Crew : Campaign Deployement

Agent 3: Channel Recommendation

Purpose : Analyzes brand information and industry classification to identify key audience segments (e.g., demographics, interests, behaviors)

Inputs:

Compulsory inputs (will get everytime):

1. Campaign objective
2. Brand Name
3. Brand Description
4. Brand Industry Type (**From output of Agent 1: Brand Mapping)**
5. Target Audience (**From output of Agent 2: Audience identification**)
6. Integrated platforms name.

Optional inputs

1. Brand Historical data (merged_agegender_final, Merged_locaton_final ,merged_ads_final)

Tasks:

If Historical Data is present:

1. Analyze Historical Camapign performance for top campaigns and their channels for the specific age, gender, region for that brand and return best performing ad channels in those campaigns.**(**Model not available currently**)**

    1. Function(python code calling channel recommendation model that will analyze past campaign performances for a brand and demographics and suggest channels)

        – – data gather/label/data verification/model/inference /host/api

If Historical Data is Not present:

1. Call an LLM to pass brand category, description and target audience to determine best ad channel(s) combination out of meta,

google and linkedin to run the campaign.(Tool not available currently)

1. Tool (LLM) that will take brand details and target audience details and give best Channel(s) among meta, google and linkedin for that ad campaign .

   – Specify prompt template, select LLM, prevent hallucinations.

   “

If any of the recommended Ad account is not integrated:

1. Remove that platform name from suggestion output so that it is not used to run campaign as the account is not integrated currently

1. Function (Python code that calls an api that checks weather the channel suggested by previous funcitons or tools is integrated or not with the workspace and then removes that channel from suggestion output)

Example Output :

Suggested Channels: Meta & Google
Final Suggestion: Meta Ads //Since google ads account is not integrated

Outputs To be used in(imp) :

Agent 5: Budget Recommendation
Crew: Creative Generation
Crew: Campaign deployment

Agent 4: Schedule Recommendation

Purpose : An agent that will suggest most optimal start date and end date for a campaign objective, target demographic and

Inputs:

Compulsory inputs (will get everytime):

1. Campaign Objective
2. Target Audience (**From output of Agent 2: Audience identification**)

Optional inputs

1. Brand Historical data (merged_agegender_final, Merged_locaton_final ,merged_ads_final)

Tasks:

If Historical Data is present:

1. Analyze brand's past campaign performance across the specific ad channels , campaign objective and target audiences to determine best start and end date for the campaign.**(**Model not available currently**)**

    1. A function (Python code calling the campaign schedule suggestion model)

        .– – data gather/label/data verification/model/inference /host/api

If Historical Data is Not present:

1. Pass the inputs (campaign objective, target audience and Ad channels) into an LLM and ask it to suggest best start date and end date (end date is optional for a campaign) . (Tool not available currently)

    1. A Tool (LLM) that will take the inputs and return start date and end date of a campaign (end date optional)

        – Specify prompt , select & test LLM, hallucination prevention

Example Output :

"Food and Beverages", "Beauty Products" ..etc

Outputs To be used in(imp) :

Agent 5: Budget Allocation
Agent 6: Campaign Name Recommendation
Crew : Campaign Deployement

Agent 5: Budget Allocation

Purpose : An agent that will suggest most optimal total daily budget and will also determine how that amount should be distributed between selected ad channels (meta, google and linkedin)

Inputs:

Compulsory inputs (will get everytime):

1. Campaign objective (eg: “Brand Awareness”)
2. Target Audience (**From output of Agent 2: Audience identification**)
3. Ad channel(s) (**From output of Agent 3: Channel Recommendation**)
4. Campaign Start date (**From output of Agent 4: Schedule Recommendation)**

Optional inputs

1. Campaign end date
2. Brand Historical data (merged_agegender_final, Merged_locaton_final ,merged_ads_final)

Tasks:

If Historical Data is present:

1. A The refined version of current budget allocation model will be called to analyze the past campaign performance data for the specific brand and suggest total budget and its distribution across channesl).

(We need to refine and utilize already existing general budget allocation model trained on synthetic data by instead we will be applying that model on the real historical data of the brand to suggest budget**)**

1. A function (Python code calling the Enhanced Budget Allocation model).

   – – data gather/label/data verification/model/inference /host/api

If Historical Data is Not present:

1. Use General Budget allocation model that uses your target demographic,ad channels and campaign schedule to determine start and end date **(General Budget Allocation Model already present)**

1. A function (Python code calling the General Budget Allocation model).

   – – data gather/label/data verification/model/inference /host/api

Example Output :

Total Budget: Rs. 5000
Distribution on Meta : Rs. 2000 (40%)
Distribution on Google: Rs 3000 (60%)

Outputs To be used in following(imp) :

Crew: Campaign Deployement

Agent 6: Campaign Name Recommendation

Purpose : An agent that will suggest a proper campaign name that is easily identifiable by user and is relevant to the campaign configuration.

Inputs:

Compulsory inputs (will get everytime):

1. Campaign objective (eg: “Brand Awareness”)
2. Target Audience (**From output of Agent 2: Audience identification**)
3. Campaign Start date (**From output of Agent 4: Schedule Recommendation)**

Optional inputs

N.A

Tasks:

1. Call the LLM that will determine a suitable campaign name for this ad campaign configuration for the user.
    2. Tool(LLM) that will generate a campaign name using the inputs

– specify prompt template, select LLM, prevent hallucinations

Outputs To be used in following(imp) :

Crew: Campaign Deployment

Current fallbacks:

Insufficient and unsuitable datasets for historical data models.