

Project Name: **MediaPlanner**

Project approach:

1. Layered Architecture (web API, Asp.Net MVC, Database)
2. Microservices (for data export, integration with Google, Roles & authentication) and can be called internally using gRPC
3. Dependency injection, Service Locator Pattern
4. Entity Framework (core), database first approach

Some **Key Features** of current implementation:

1. Created DataStore to perform CRUD operations for current task.
2. Master data is cached in memory, first time it will be loaded from DB and then from cache memory.
3. Used Microsoft's in-built Service locator pattern for DataStoreContext and Configuration class.
4. Used Javascript template to append dynamic data.
5. Responsive & interactive UI (bootstrap-4). Please refer to screenshots for more details.
6. Suggested approach is layered architecture and Microservice architecture (Microservice is for RoleBasedAuthorization and Integrations with FB, Google, Or any third party, finance application)
7. Common access to web API, can be consumed across any channel (MVC, Integrations, RoleBasedAuthentication)
8. Validation in UI, to sanitize input from end-user
9. Have used latest .Net Core technology and EF Core (database first approach)

Suggested Team Size: 5

1 Business Analyst/Product Owner – To validate the use cases and code review

2 Developers – For the development from ground

1 Tester – Having tester is luxury

1 DevOps – For deployment in staging, production

Time Estimations: 55 working days

Technologies Used:

Front-end: Asp.Net MVC (Razor View), .Net Core 2.0, Razor syntax, Javascript jQuery template

Back-end: Web API (.Net core 2.0), Class Libraries (.Net Core), MS SQL Server 2017

Database: Azure Database, Entity Framework (Core 3.1.0)

External References: Microsoft.EntityFrameworkCore.SqlServer (3.1.0)

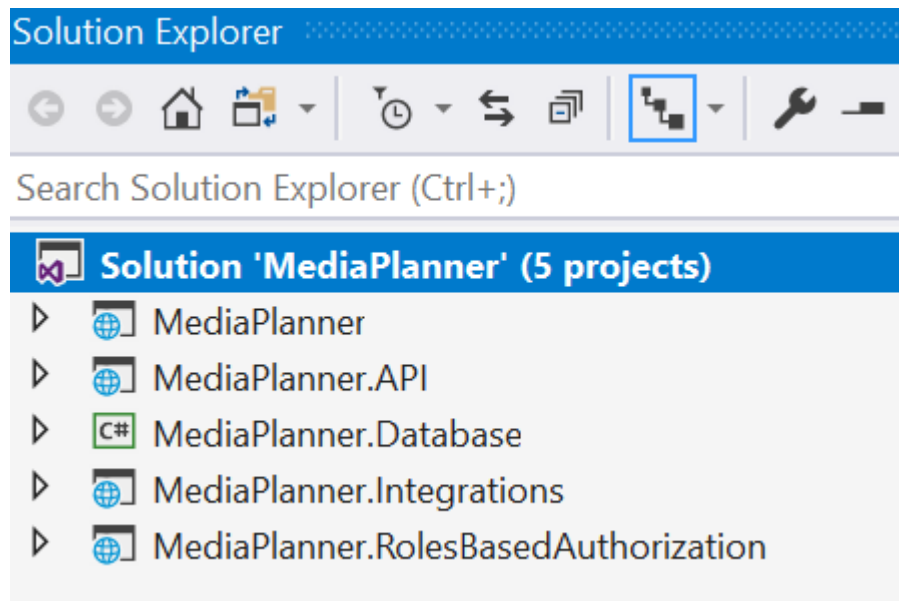
Development Tools: VS 2017, MS SQL Server 17

Suggested Methodology:

Agile, for better quality, fast development, early failure detection

Project tracking tools: JIRA (for creating user stories, BDD, tasks) & InvisionApp (for wireframes)

Project Architecture:



MediaPlanner: This is the main UI, built using Asp.Net MVC, Razor syntax

MediaPlanner.API: This is for CRUD operations for UI

MediaPlayer.Database: Used Entity Framework (Database first approach)

MediaPlanner.Integrations: For external integrations, like integrations with FB/Google media manager, integrations with Finance applications

MediaPlanner.RolesBasedAuthorization: This is for custom roles based authentication and authorization (Microsoft in-built Roles based authentication)

Installation Guide:

1. Database script is available with the code. Script contains Schema and Master Data.

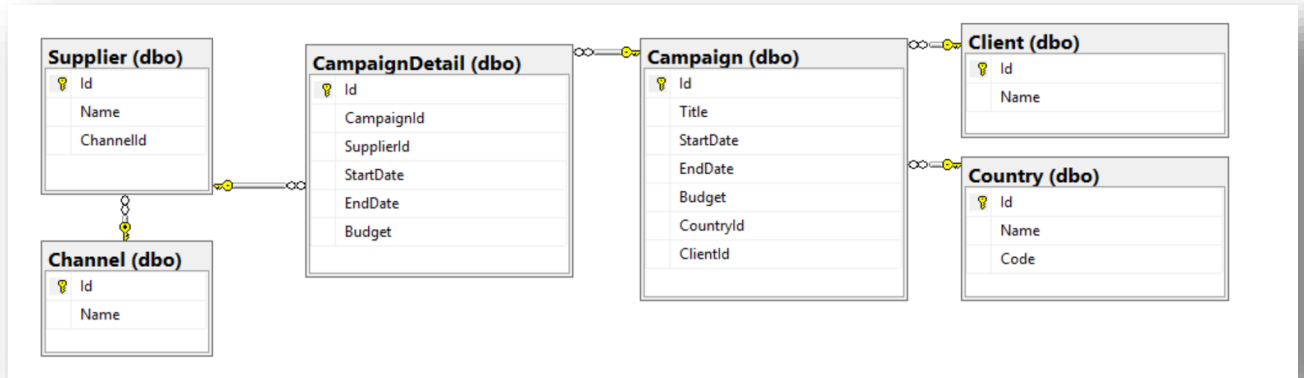


MediaPlannerDB.Script.sql

2. Configure the connection string in MediaPlanner.Database > MediaPlannerDbContext.cs class
3. Go to solution and set MediaPlanner and MediaPlanner.API as startup projects.

4. Make sure that API endpoints configured in application matched with your runtime environment.
5. For more information, you may contact Dheeraj.Bansal14@gmail.com

Screenshots from Database:



dbmedia.MediaPlannerDB - Diagram_0* SQLQuery1.sql - db...B (db_admin (118))*

```

1 Select * from Client
2 Select * from Country
3 Select * from Channel
4 Select * from Supplier
  
```

100 %

Results Messages

	Id	Name	Code
1	1	United Arab Emirates	UAE
2	2	United States	US
3	3	Canada	CAD
4	4	India	IN
5	5	Singapore	SG

	Id	Name
1	1	Out-Of-Home
2	2	Digital
3	3	TV

	Id	Name	ChannelId
1	1	Facebook	2
2	2	Instagram	2
3	3	Twitter	2
4	4	Gulf News	2
5	5	MCB 1	3
6	6	MCB 2	3
7	7	MCB 3	3
8	8	Billboard	1

Screenshots from running application (MediaPlanner)

The screenshot shows a web browser window with the address bar displaying 'localhost:63141/#advertisement'. The application has a dark header with 'MediaPlanner' and navigation links 'Home', 'Report', and 'Contact'. Below the header is a tabbed interface with 'Campaign', 'Media Channel', 'Ad Management', and 'Preview'. The 'Campaign' tab is active, showing a form with the following fields:

- Title:** Campaign_Dubai_DutyFree
- Client:** Emirates (dropdown menu)
- Market:** United Arab Emirates (dropdown menu)
- Start Date:** 12/19/2019 (calendar icon)
- End Date:** 12/21/2019 (calendar icon)
- Budget:** \$ 52000

Below the form is a 'Next' button and a footer with '© 2019 - MediaPlanner'.

The screenshot shows a web browser window with the address bar displaying 'localhost:63141/#mediaChannel'. The application has a dark header with 'MediaPlanner' and navigation links 'Home', 'Report', and 'Contact'. Below the header is a tabbed interface with 'Campaign', 'Media Channel', 'Ad Management', and 'Preview'. The 'Media Channel' tab is active, showing a form with the following fields:

- Media Channel:** Out-Of-Home, Digital, TV (dropdown menu)

Below the form is a 'Back Next' button and a footer with '© 2019 - MediaPlanner'.

