

Serverless Architecture

Jaime Pallarés

INDEX

- **Introduction**
- **Calendar**
- **Evaluation Model**
- **Initial Tasks**
- **Resources**
- **Installation & Setup**

INTRODUCTION

Content
Design a project
Design Data Model oriented to Documents
User Real Time Database
Protect the Data
Define functions on Database side

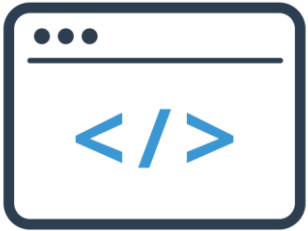
CALENDAR

Date	Description
09/05/2022 13:00	Kick Off. WorkShop
16/05/2022 13:00-15:00	Live Follow Up - Delivery 1
30/05/2022 13:00-15:00	Final App Delivery. Presentation & Demo: 10 minutes/team.

- All Code Deliveries by **GitHub**
- In Each delivery send an **email** with Copy to your team members including the **GitHub URL**



Apps: Mobile / Web



Admin Site

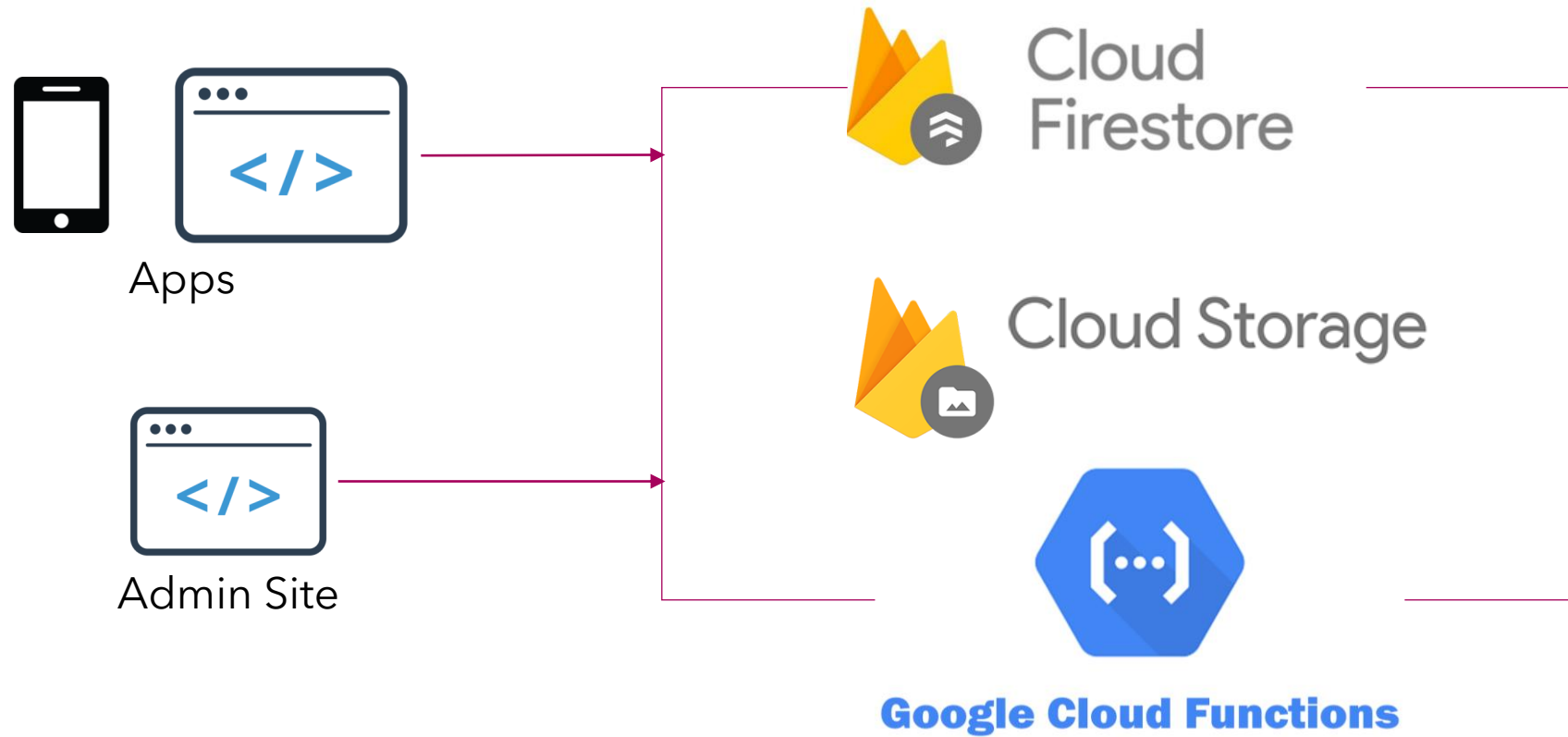
Front End



Back End

TRADITIONAL ARCHITECTURE

- Firebase DB, Cloud Firestore, Cloud Functions



**SERVERLESS
ARCHITECTURE**

APP: WEBAPP

- Firebase DB, Cloud Firestore, Cloud Functions
- CRUD pages for your own >4 Firestore document types with your own Rights
- CRUD pages for base datamodel 4: Users, Groups, Roles, Rights
- In total 8 Type of Documents at the same level
- Users, Groups, Roles, Rights. All Entities are M:N
- Create 3 Groups: Admins, Managers, Users
- Create 3 Roles: Admin, Manager, User
- Integrate with the Firebase Authentication
- Create Rights for each operation and private data: P_USER_x (CRUD), ...
- Protect your data and functions with Effective Rights.
- Initial Data by script to setup the minimum structure: admin user with role, group and rights.
- Host your webapp with on Google Storage

EVALUATION MODEL
THE PROJET TO DELIVER

VERSIONS TO DELIVER

1. Document Store Data model
2. Create Security Rules
3. Create Cloud Functions
4. Store Files
5. Instant Messages between Users
6. Connect to WebApp or Mobile App
- 7. CRUD pages for your own >4 Firestore document types with your own Rights**
8. Integrate with the Firebase Authentication
- 9. CRUD pages for Required Entities (4) : Users, Groups, Roles. All Entities are M:N**
10. Create 3 Groups: Admins, Managers, Users
11. Create 3 Roles: Admin, Manager, User
12. Create Rights for each operation and private data for the correct Roles
13. Protect your data and functions with Rights
14. Validate your Data with Rules
15. Initial Data by script to setup the minimum structure: admin user with role and groups

**8 Type of Documents
at the same level**

**EVALUATION MODEL
THE PROJET TO DELIVER**

Criteria	Max Qualification
Documentation: Data Model	2 Points
Implement the Data Model	2 Points
Security Rules	3 Points
Authentication	2 Points
Store Files	2 Points
Data Validation	2 Points
Cloud Functions: 1. Initial Script for Setup Data, 2. Add Data to Documents for some events. 3 Events for each insert and update	3 Points
Instant Messages	2 Points
Config Files for all Cloud SetUp on your git repo	2 Points

EVALUATION MODEL CRITERIAS

INITIAL TASKS

CREATE A GROUP OF **4 MEMBERS**

CREATE A GITHUB REPOSITORY

SHARE WITH jaime.pallares@epitech.eu