

Marapp, Flutter App

25.06.2025

Enrico Pezzano

Product Manager

4825087

enriicola@proton.me

Table of Contents

[Overview](#)

[Goals](#)

[Specifications \(app features\)](#)

[The client](#)

[Expected goals](#)

[I. Market Alignment and Brand Launch](#)

[II. System Stability and Integration Completion](#)

[III. Break-Even and ROI Verification](#)

[Background](#)

[Reference Contract Documents](#)

[Site dependencies](#)

[Constraints](#)

[Project Roadmap](#)

[Project Structure](#)

[OBS – Organizational Breakdown Structure](#)

[Product Breakdown Structure \(PBS\)](#)

[Activity Breakdown Structure \(ABS\)](#)

[Work Breakdown Structure \(WBS\)](#)

[Supporting Management Activities](#)

[Risk Management](#)

[Quality Management](#)

[Development](#)

[Configuration Management](#)

[Acceptance Criteria & Testing](#)

[Reporting and Phase Reviews](#)

[Master Plan](#)

[Milestones & Gantt Chart](#)

[Technical Quotation](#)

[Costs](#)

[Purchasing Plan](#)

[Price & Payment](#)

[Billing and Collections](#)

[Economic Value Added \(EVA\)](#)

Overview

Marapp is a Flutter app for a concept e-commerce for my mom (from this moment onward defined as “the client”). It is designed to enable on-the-go ordering of homemade sweets produced by the client’s artisanal bakery. It will serve as a mobile storefront, integrating secure authentication, product browsing, shopping cart functionality, and payment processing. Inspired by standard PMP structures used in manufacturing projects like the ones we have seen during lectures , this plan adapts those frameworks to a software delivery context.

Goals

1. Deliver a stable, intuitive mobile app (iOS & Android) for customers to browse and purchase homemade sweets.
2. Integrate backend APIs for product catalog, user management, and order processing.
3. Achieve an initial launch (MVP) within 6 months, with a scalable architecture for future features.

Specifications (app features)

- User registration, login, and password recovery.
- Product catalog with images, descriptions, and pricing.
- Shopping cart and checkout flows.
- Payment gateway integration (Stripe or similar).
- Order confirmation, status tracking, and notifications.

The client

The client is a stay-at-home mom artisanal bakery specializing in homemade sweets. The app will support her expanding direct-to-consumer strategy.

Expected goals

I. Market Alignment and Brand Launch

By this milestone, the Marapp Flutter E-Commerce App will be fully aligned with the bakery's strategic goal of expanding direct-to-consumer sales. The app will be live in both iOS and Android stores, carrying all approved branding and messaging. This milestone ensures stakeholder buy-in, secures early adopter feedback, and positions the client to reach a 10% increase in online orders within the first quarter post-launch.

II. System Stability and Integration Completion

At this point, all core technical components will be delivered and validated: backend APIs fully integrated with the Flutter frontend, payment gateway configured and tested, and push-notification services operational. All automated tests (unit, integration, and performance) will pass with $\geq 90\%$ coverage, and the app will exhibit end-to-end functionality without critical defects on supported OS versions (Android 8+ and iOS 13+).

III. Break-Even and ROI Verification

This milestone sets the benchmark for economic viability: total cumulative revenue generated through in-app sales must cover the combined development and operating costs up to this date. Based on the initial price and cost structure, Marapp must achieve at least €28,400 in sales (development fee plus 12-month hosting) to break even. Additionally, the first formal ROI report will be delivered, demonstrating positive cash flow and outlining next-year profitability targets.

Background

Reference Contract Documents

The reference contract documents include:

- Project proposal
- Requests for Proposals or Requests for Quotations
- Contract Agreement
- Statement of Work
- Warranty Agreement
- Technical specifications
- Acceptance Testing Criteria

Site dependencies

- Cloud hosting (e.g., AWS Amplify or Firebase) for backend services.
- Third-party services: Payment gateway, image CDN, push notification service.
- Developer tools: CI/CD pipeline in GitHub Actions, code repository on GitHub.

Constraints

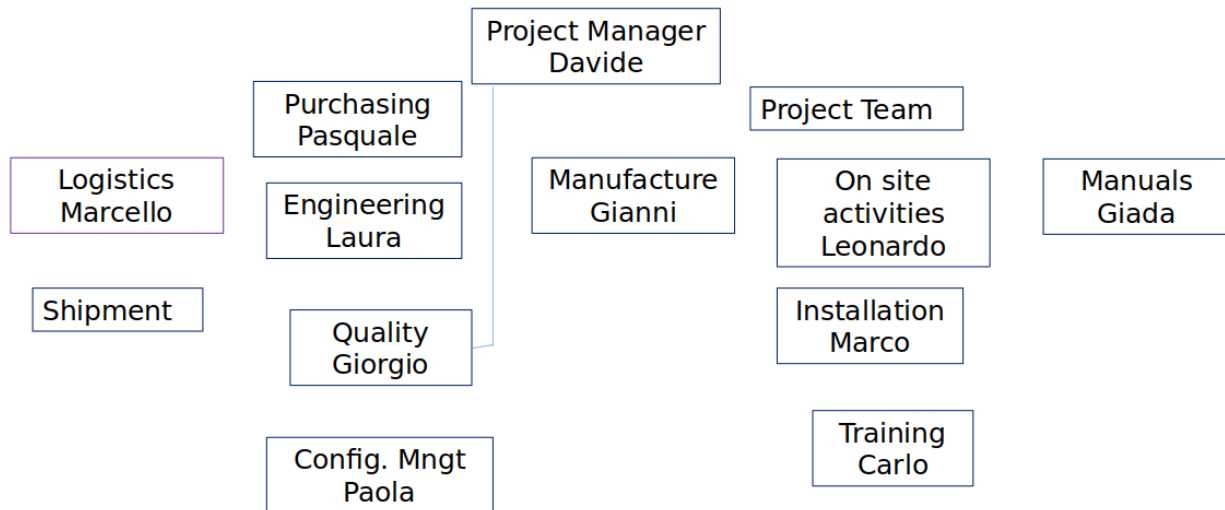
- **Time:** MVP launch in 6 months; full feature set in 9 months.
- **Budget:** €20,000 maximum development budget (internal costs).
- **Platform:** Flutter cross-platform; must support Android 8+ and iOS 13+.

Project Roadmap

- **T0 (Project Kick-off):** 2025-07-01
- **Design Completion:** T0 + 1.5 months
- **Core Development End (MVP):** T0 + 6 months
- **Beta Release & Testing:** T0 + 7 months
- **Production Launch:** T0 + 9 months

Project Structure

OBS – Organizational Breakdown Structure



- **Project Manager:** Davide – overall project accountability, stakeholder communication, and steering.
- **Purchasing:** Pasquale – procurement of third-party services (hosting, CDN, payment gateway).
- **Engineering Lead:** Laura – technical oversight of backend and frontend development.
- **Quality Assurance:** Giorgio – test planning, execution, and quality controls.
- **Configuration Management:** Paola – version control strategy, release branching, and deployment governance.
- **Logistics & Deployment Support:** Marcello – app distribution approvals, store submissions, and release packaging.
- **Deployment Coordination:** (Shipment) – coordination of release artifacts delivery to production environments.
- **Manufacturing (Build & CI/CD):** Gianni – CI/CD pipeline configuration, build automation, and production deployments.

Product Breakdown Structure (PBS)

The major deliverables and activities that will compose the final product are:

1. Mobile App (Flutter)
2. Backend Services (APIs)

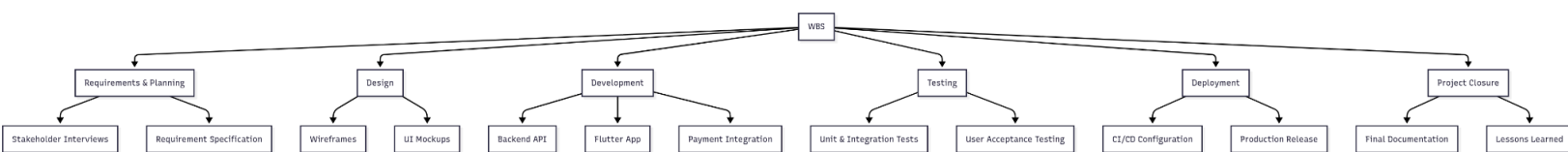
3. CI/CD Pipeline
4. Documentation
5. User Training Materials

Activity Breakdown Structure (ABS)

The project will consist of the following activities:


- Requirements Analysis
- UI/UX Design
- API Design & Development
- Flutter Frontend Development
- Integration & Testing
- Deployment & Hosting Setup
- Release Management
- Documentation
- Training & Handover

Work Breakdown Structure (WBS)



graph TD

A[WBS] --> A1[Requirements & Planning]



A1 --> A11[Stakeholder Interviews]
A1 --> A12[Requirement Specification]
A --> A2[Design]
A2 --> A21[Wireframes]
A2 --> A22[UI Mockups]
A --> A3[Development]
A3 --> A31[Backend API]
A3 --> A32[Flutter App]
A3 --> A33[Payment Integration]
A --> A4[Testing]
A4 --> A41[Unit & Integration Tests]
A4 --> A42[User Acceptance Testing]
A --> A5[Deployment]
A5 --> A51[CI/CD Configuration]
A5 --> A52[Production Release]
A --> A6[Project Closure]
A6 --> A61[Final Documentation]
A6 --> A62[Lessons Learned]

Supporting Management Activities

Risk Management

ID	Risk Description	Mitigation	Owner	Status	Risk
----	------------------	------------	-------	--------	------

R1	Delays in API delivery	Parallel development of stubs & mocks	PM & Dev	Open	2
R2	Third-party service outages	Select redundant services; caching strategies	DevOps	Open	1
R3	Flutter platform quirks on older OS versions	Early compatibility testing	QA & Dev	Open	3
TOTAL					6

Key risks that have been identified in this project include:


1. **Schedule delays:** These delays may arise due to dependencies on external factors, delays in the delivery of equipment or services, or unexpected obstacles encountered during the project execution.
2. **Technical challenges:** The project may face difficulties resulting from limitations in technology or compatibility issues between different components or systems involved.

The project team will prioritize managing these risks to ensure the successful completion of the project within the defined timeframe and without compromising the desired outcomes.

Quality Management

- Code reviews via pull requests on GitHub.
- Automated test coverage (target $\geq 80\%$).
- Weekly regression test cycles.

Implementing effective quality management strategies is essential to ensure that the project meets the desired standards and delivers high-quality results. Within the team,



there is a specific role for project quality control and management: the person who fulfills this role will be responsible for developing and implementing a Quality Management Plan.

Periodic inspections and tests will be conducted to verify that the project follows best practices, industry standards, and specifications. This will allow for the early identification of errors, non-conformities, or deficiencies, providing opportunities for improvement.

Proper documentation of this process (e.g., code reviews, automated test coverage, weekly regression test cycles, and non-conformity reports) facilitates traceability and provides evidence of compliance with quality standards.

Development

The project will be developed using the agile methodology, structured into two-week sprints. A prioritized product backlog, managed by the Product Owner, will drive sprint planning. Daily stand-up meetings will ensure team alignment, impediment removal, and transparent progress tracking. Each sprint will conclude with a sprint review to demonstrate completed features to stakeholders and a retrospective to capture lessons learned and continuously improve team processes. This iterative approach enables early delivery of a minimum viable product (MVP), supports continuous integration and automated testing, and allows for rapid adaptation to evolving requirements, ensuring both high quality and stakeholder satisfaction

Configuration Management

Thanks to Paola, Marapp employs a thorough Git workflow of feature, develop, release, and main branches. There are also pull-request reviews and Semantic Versioning. Any approved change automatically initiates the CI/CD pipeline to build, test, and package the application. Environment variables and secrets are kept secure so that all releases can be replicated and traced.

Acceptance Criteria & Testing

Acceptance will be based on explicit, simple-to-understand rules for each story: proper setup of accounts that work, easy product navigation, robust order entry and payment processing, and a simple user interface on all devices. Unit and integration testing are included in our quality process as part of the continuous integration and delivery pipeline, and we also have exploratory and user acceptance testing sessions done manually. This way, each feature fulfills functional requirements and also satisfies end users. A release will be permitted into production only after all acceptance tests pass—no significant defects, high performance metrics, and stakeholder panel approval.

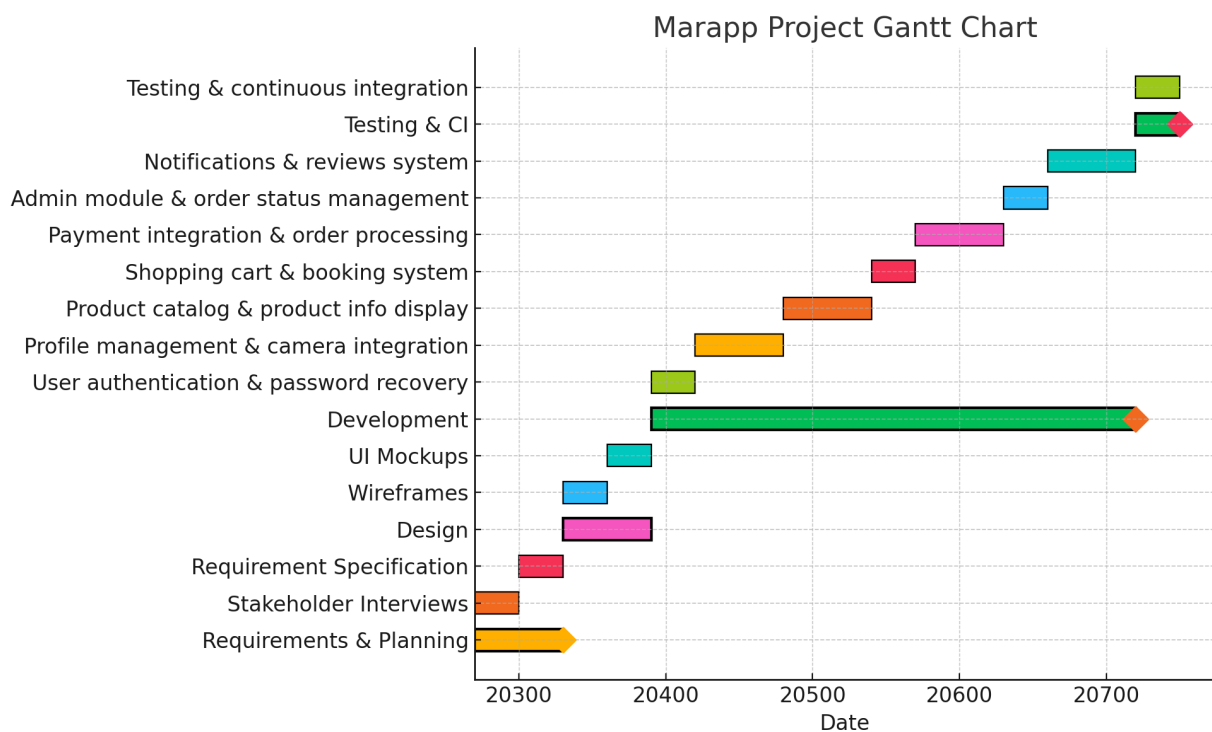
Reporting and Phase Reviews

The team will conduct a review at the end of each phase of the project, to ensure that a stage is completed before moving on to the next one.

Master Plan

Milestones & Gantt Chart

- **M1 – Project Kick-off:** 2025-07-01
- **M2 – Design Complete:** 2025-08-15
- **M3 – MVP Delivery:** 2025-12-01
- **M4 – Beta Release:** 2026-01-01
- **M5 – Production Launch:** 2026-04-01



Technical Quotation

Costs

Based on the IT Management table: total BCWS = €8,300.

This amount covers various costs, such as the production costs of Marapp, app design, development software purchase, production of manuals, user training, and project management activities.

Purchasing Plan

With Pasquale at the steering wheel, we will request and compare prices from various vendors of Flutter plugins, hosting, and payment APIs. Then we will bundle our buys to get bulk discounts. All our purchase orders and contracts will be kept in our version-controlled repository for easy records and renewals.

Price & Payment

We'll bill €10 750—covering €8 600 of costs plus 25 % margin—in three installments: 30 % on signing, 40 % on beta delivery and 30 % on final acceptance. Electronic invoices carry net-30 terms with a 2 % early-payment discount to accelerate cash flow.

Billing and Collections

As stated in the contract, the payments are planned as follows:

1. 20% at the order
2. 20% at the delivery on site
3. 20% at the end of the installation
4. 40% at the final acceptance

Economic Value Added (EVA)

Economic Value Added (EVA) With an estimated NOPAT of €2,150 and a 10% cost of capital of €8,600 (€860 annual charge), Marapp's EVA is €1,290. The positive EVA indicates that our project returns exceed the cost of financing, which aids in making future investment decisions.