Università degli Studi di Napoli Federico II Scuola Politecnica e delle Scienze di Base

Dipartimento di Ingegneria Elettrica e Tecnologie dell’Informazione

Corso di Laurea in Informatica - Insegnamento di Ingegneria del Software

Anno Accademico 2023/2024

Progettazione di “DietiDeals24”, una Piattaforma per la Gestione di Aste Online

Antonio Abbatiello - N86003037

Vincenzo Marotta - N86004151

Leonardo Colamarino - N86003586

15**/**04/2024

**Sommario**

[1. Introduzione al Progetto 4](#_Toc163751766)

[1.1 DietiDeals24 4](#_Toc163751767)

[2. Documento dei Requisiti Software 5](#_Toc163751768)

[2.1 Analisi dei Requisiti 5](#_Toc163751769)

[2.1.1 Modellazione dei Casi d’Uso 5](#_Toc163751770)

[2.1.2 Tabelle Cockburn 8](#_Toc163751771)

[2.1.3 Prototipi Mock-Up 13](#_Toc163751772)

[2.1.4 Individuazione del Target degli Utenti 23](#_Toc163751773)

[2.1.5 Valutazione dell’Usabilità 25](#_Toc163751774)

[2.1.6 Glossario 26](#_Toc163751775)

[2.2 Specifica dei Requisiti 27](#_Toc163751776)

[2.2.1 Diagrammi di Classi 27](#_Toc163751777)

[2.2.2 Diagrammi di Sequenza 28](#_Toc163751778)

[2.2.3 Statechart 29](#_Toc163751779)

[3. Documento del Design di Sistema 30](#_Toc163751780)

[3.1 Sviluppo Front-end 30](#_Toc163751781)

[3.2 Sviluppo Back-end 31](#_Toc163751782)

[3.3 Diagrammi di Classi di Design 32](#_Toc163751783)

[3.4 Diagrammi di Sequenza di Design 33](#_Toc163751784)

[4. Testing e Valutazione dell’Usabilità 34](#_Toc163751785)

Capitolo 1

# Introduzione al Progetto

## DietiDeals24

Il progetto Dietideals24 consiste nella realizzazione di una piattaforma completa e intuitiva per la gestione di aste online, che permette agli utenti di partecipare e organizzare aste di diverso tipo, effettuare offerte per beni/servizi e monitorare lo stato delle proprie transazioni, il tutto tramite un'interfaccia user-friendly accessibile tramite dispositivi mobile iOS.

Il sistema è composto da un’applicazione client front-end disponibile come interfaccia utente, e da un’applicazione server back-end per la gestione del database sottostante. La comunicazione tra back-end e front-end sarà resa possibile dalle interfacce di REST API fornite dal sistema server.

Capitolo 2

# Documento dei Requisiti Software

## Analisi dei Requisiti

### Modellazione dei Casi d’Uso

Per ricavare un diagramma di casi d’uso partendo dai requisiti funzionali del sistema, abbiamo modellato due schemi:

* Uno schema per l’accesso/registrazione di un utente al sistema
* Uno schema per l’utilizzo delle funzionalità dell’applicazione di un utente registrato

**Diagramma Use-Case per il sistema di autenticazione di DietiDeals24**

Immagine che contiene diagramma, testo, schizzo, linea

Descrizione generata automaticamente

**Diagramma Use-Case per l’interazione con la piattaforma aste di DietiDeals24**

Immagine che contiene testo, diagramma, documento, ricevuta

Descrizione generata automaticamente

### Tabelle Cockburn

Sono disponibili inoltre descrizioni testuali dettagliate, in forma di tabelle Cockburn, per due casi d’uso dell’applicazione.

I casi d’uso scelti sono:

1. **Use Case #1:** Creazione di un’asta a tempo fisso
2. **Use Case #2:** Presentazione di un’offerta per asta a tempo fisso

**Tabella Cockburn per Use Case #1**





**Tabella Cockburn per Use Case #2**





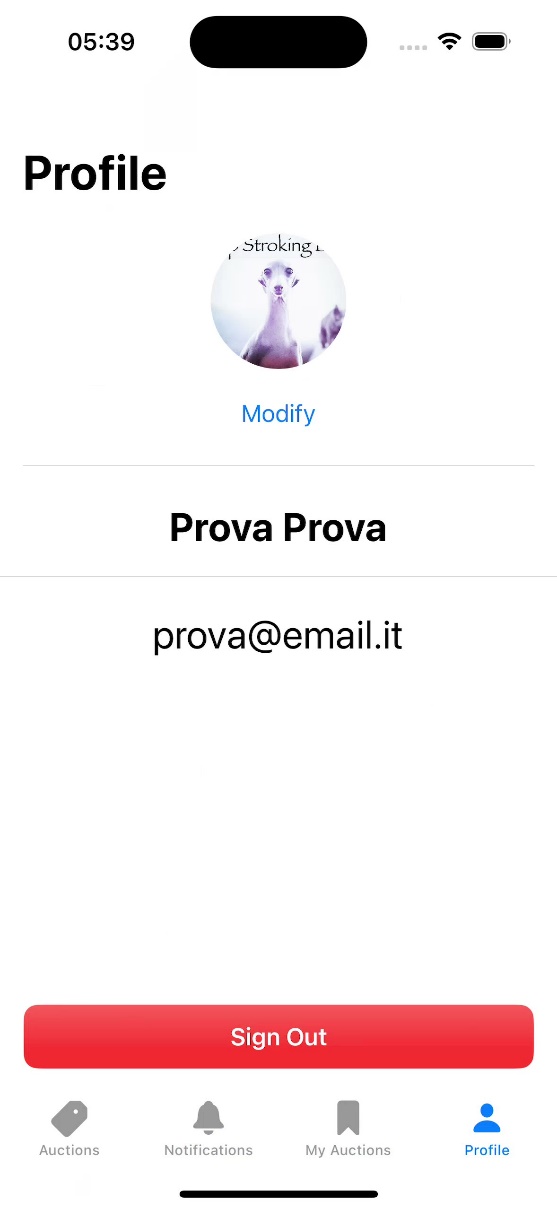
### Prototipi Mock-Up

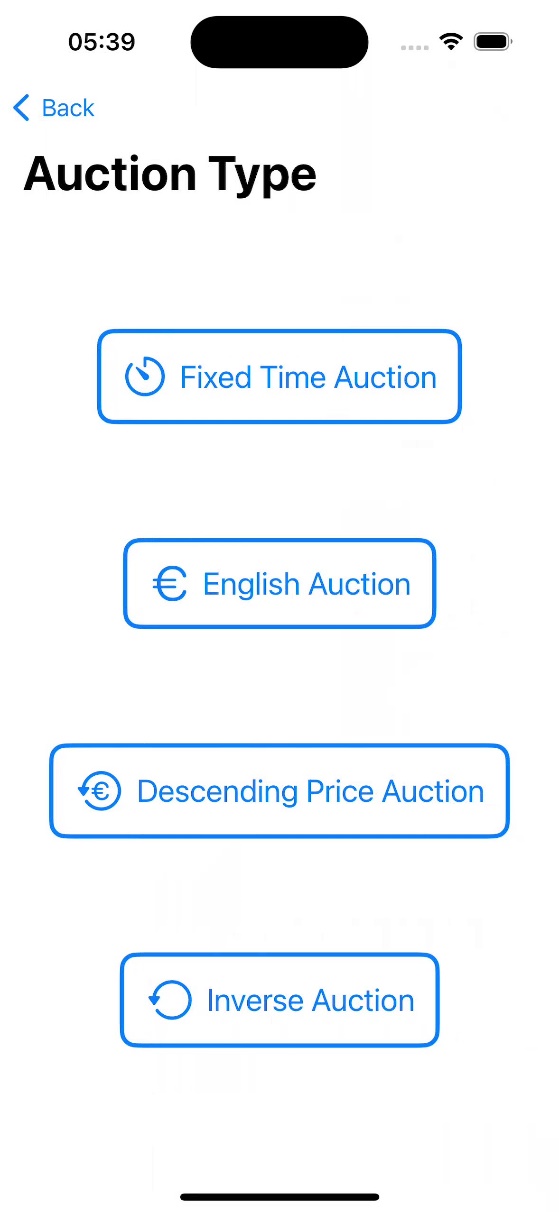
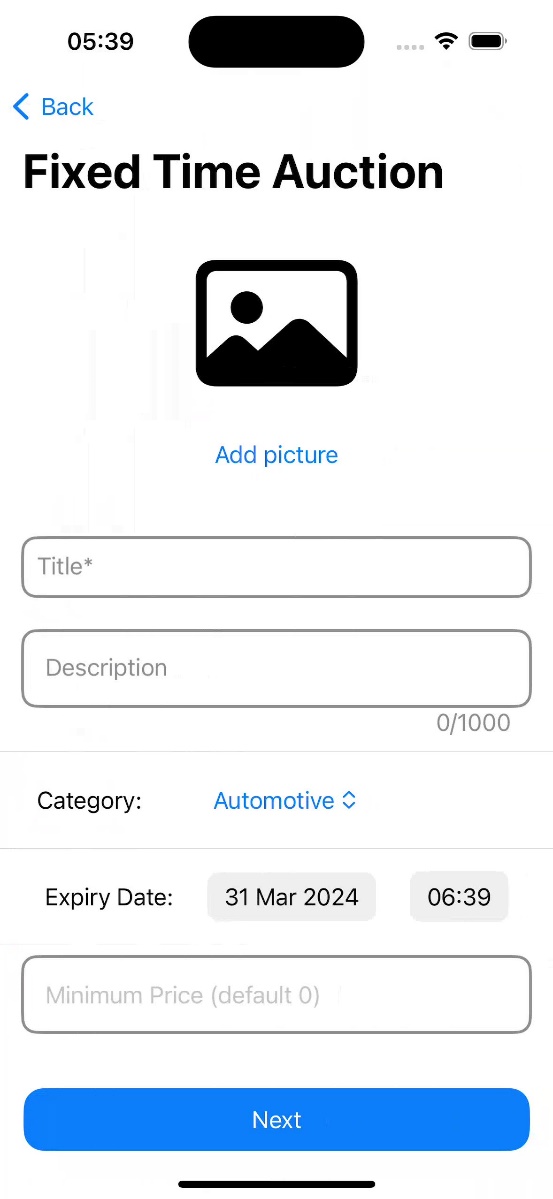
Di seguito sono mostrati dei prototipi mock-up di schermate dell’applicazione, creati usando l’applicazione Figma.

**Prototipo Mock-Up per Use Case #1**

Sono riportate le seguenti schermate:

* Schermata “Profilo”
* Schermata “Le Mie Aste”
* Selezione di asta da creare
* Creazione di un’asta a tempo fisso
* Inserimento di dati per la creazione dell’asta
* Visualizzazione di un sommario dell’asta da pubblicare
* Popup di pubblicazione con successo dell’asta
* Schermata “Le Mie Aste” (ora contenente l’asta appena pubblicata)
* Visualizzazione asta a tempo fisso

****

****

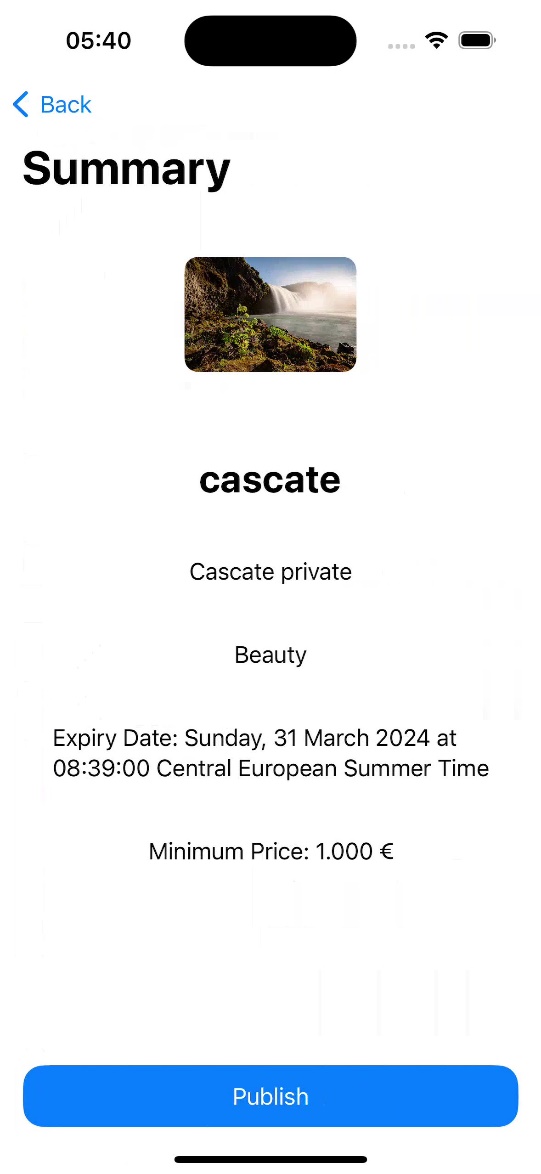
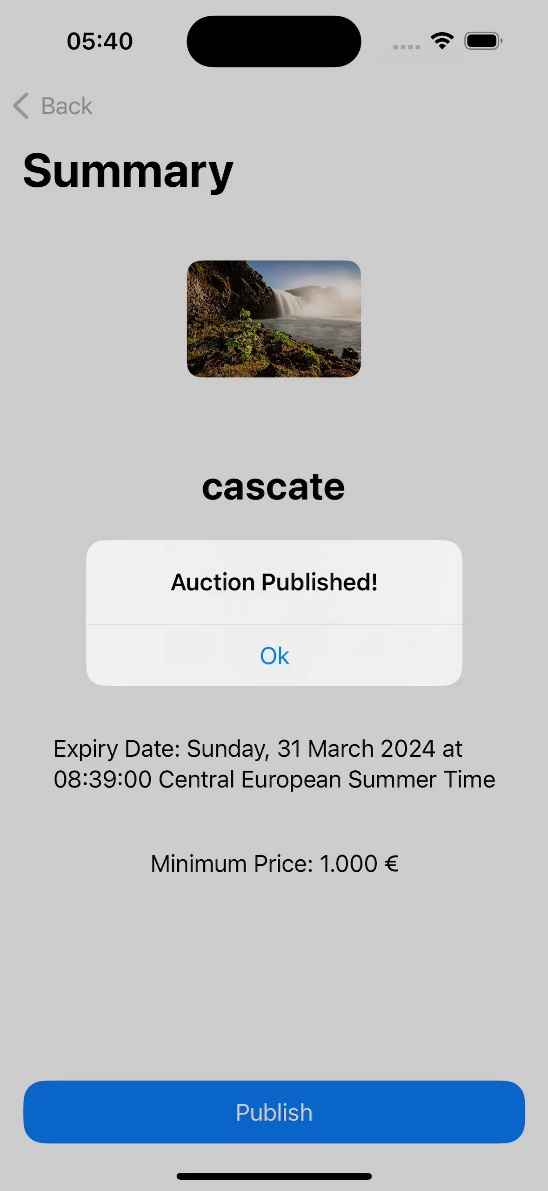


Immagine che contiene testo, schermata

Descrizione generata automaticamente

****

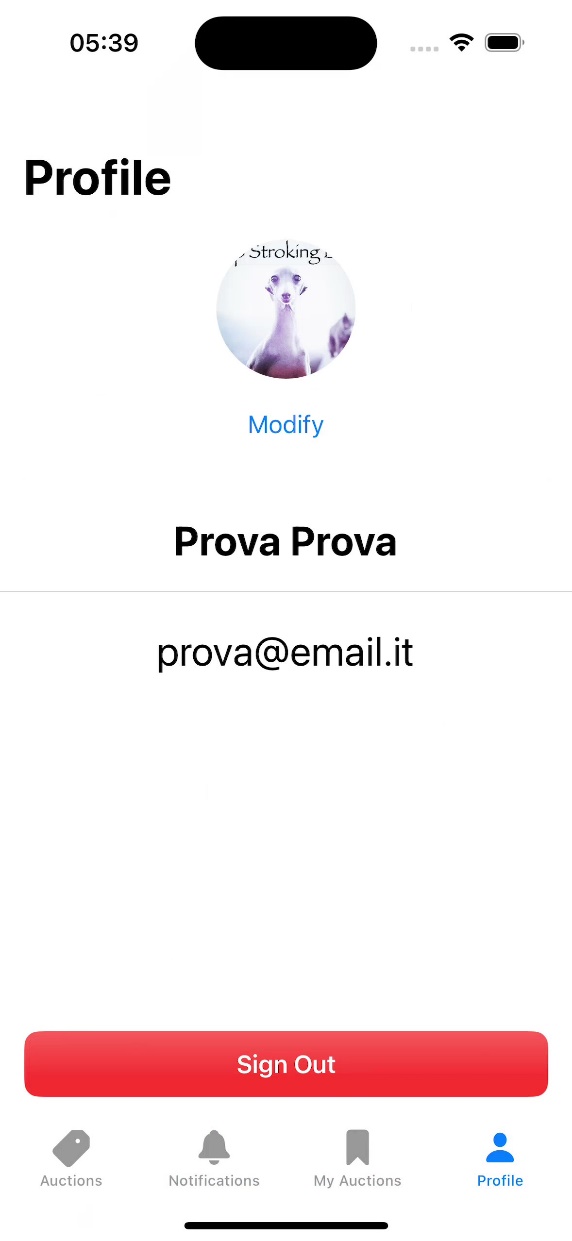
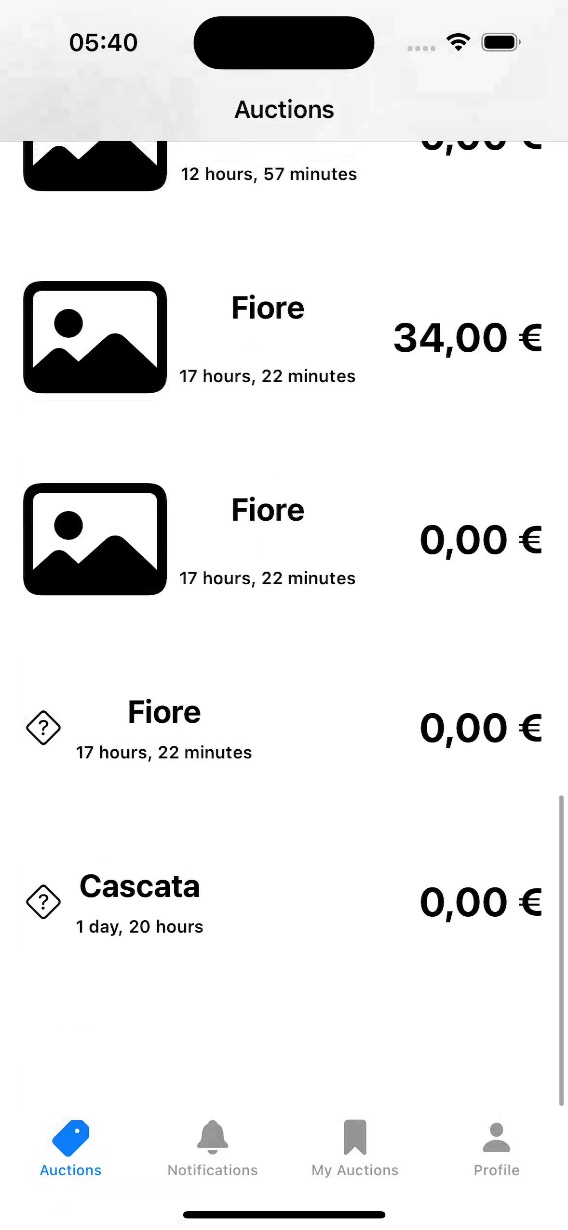
**Immagine che contiene testo, schermata

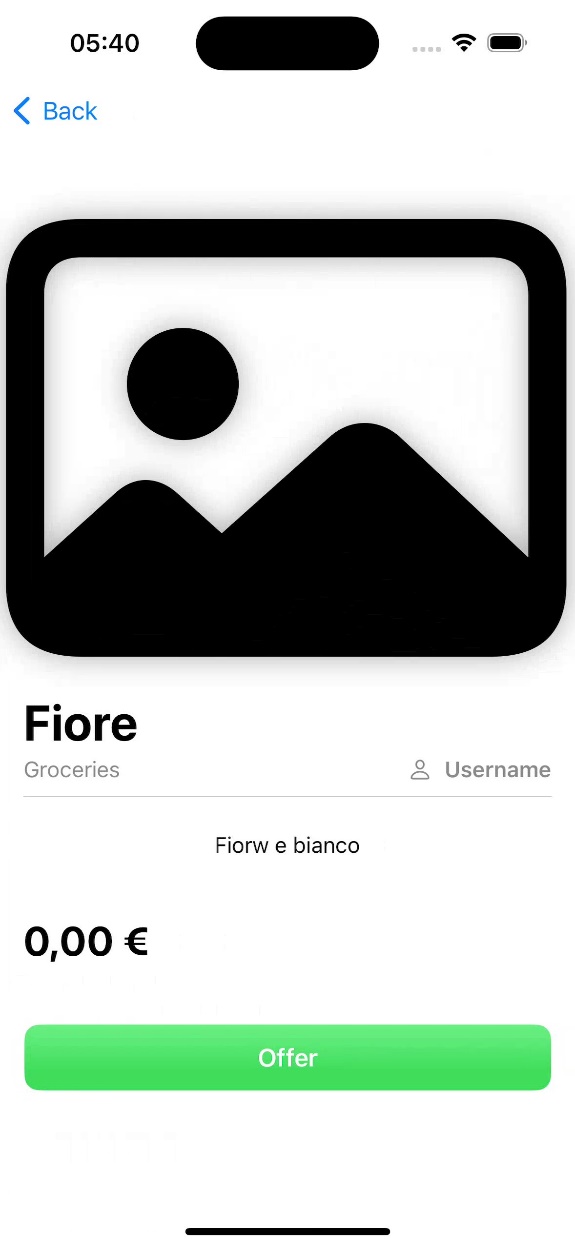
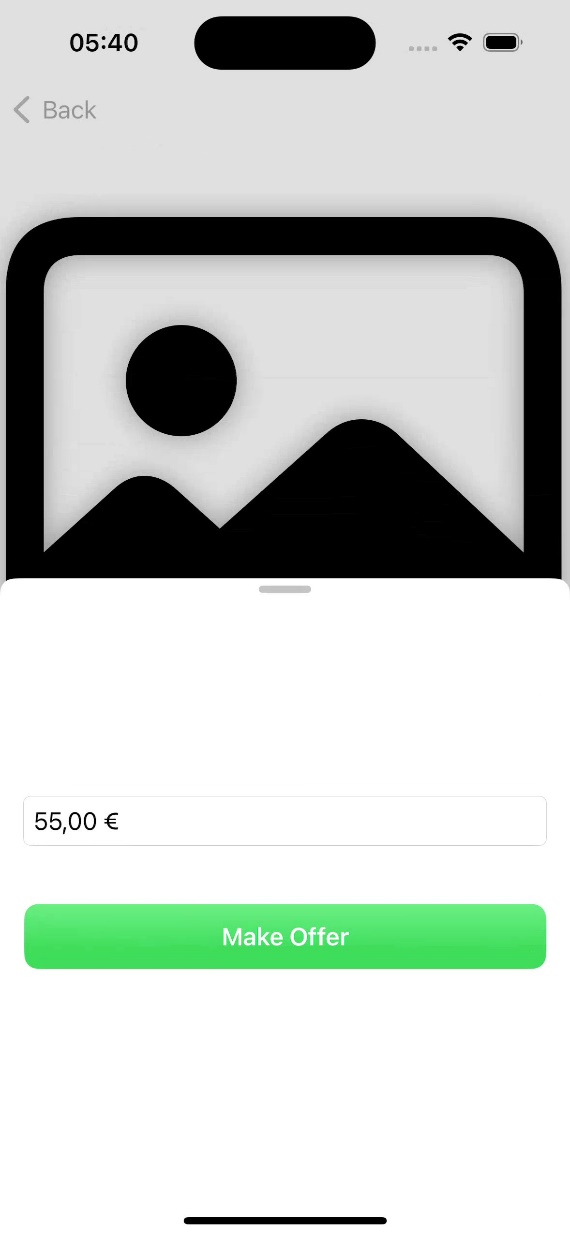
Descrizione generata automaticamente**

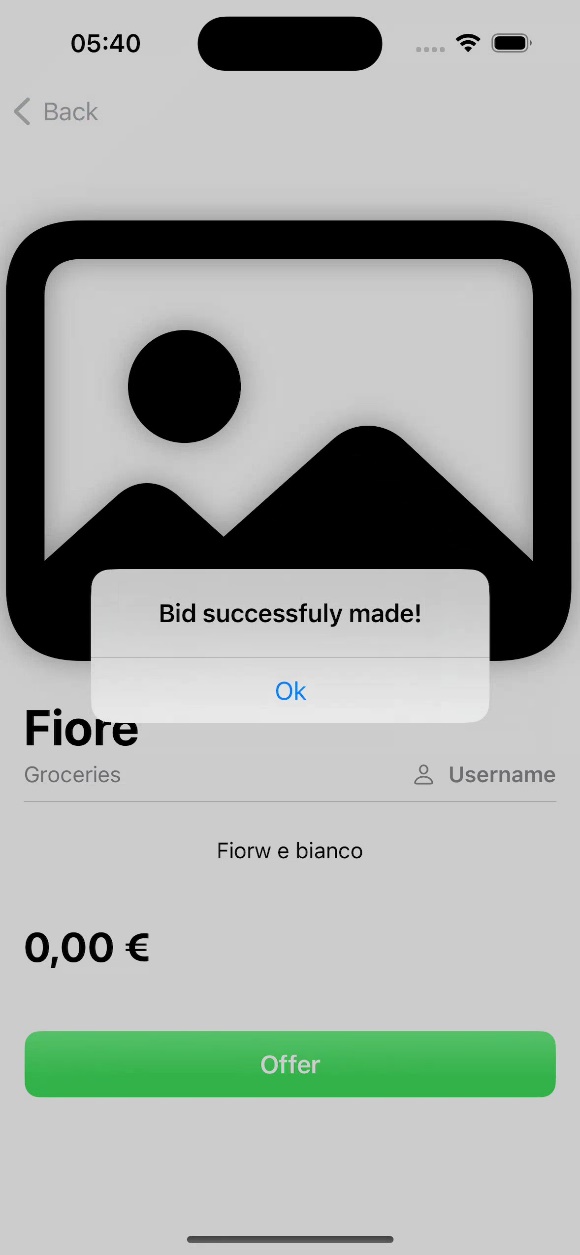
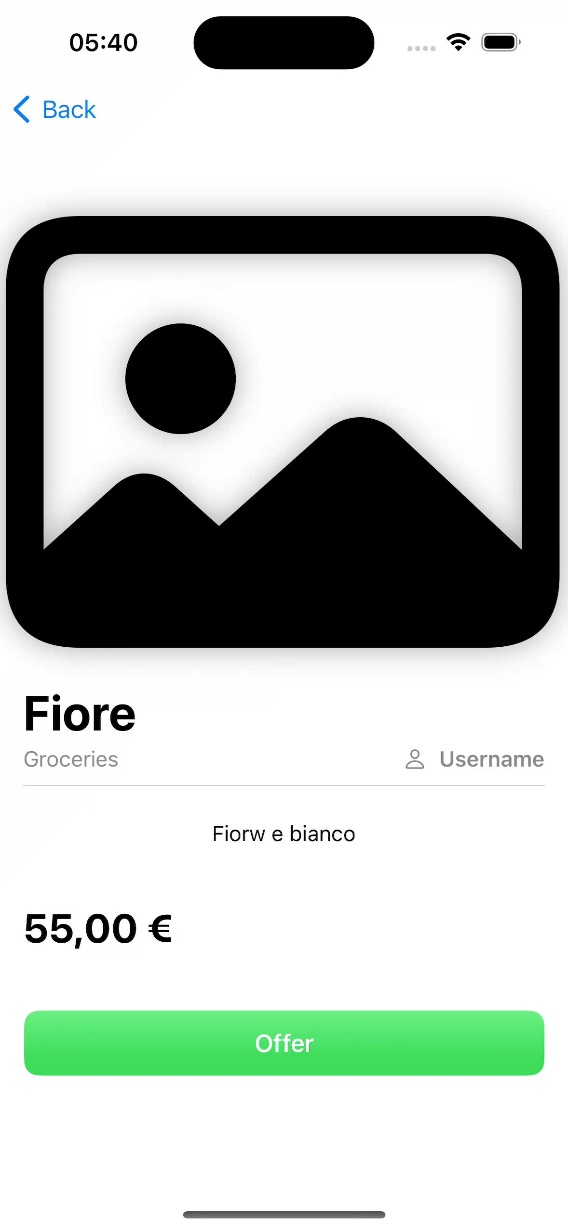
**Prototipo Mock-Up per Use Case #2**

Sono riportate le seguenti schermate:

* Schermata “Profilo”
* Schermata “Aste”
* Schermata di visualizzazione asta e di presentazione offerta
* Popup di presentazione offerta
* Popup di offerta avvenuta con successo
* Schermata di visualizzazione asta (ora con ultima offerta aggiornata)



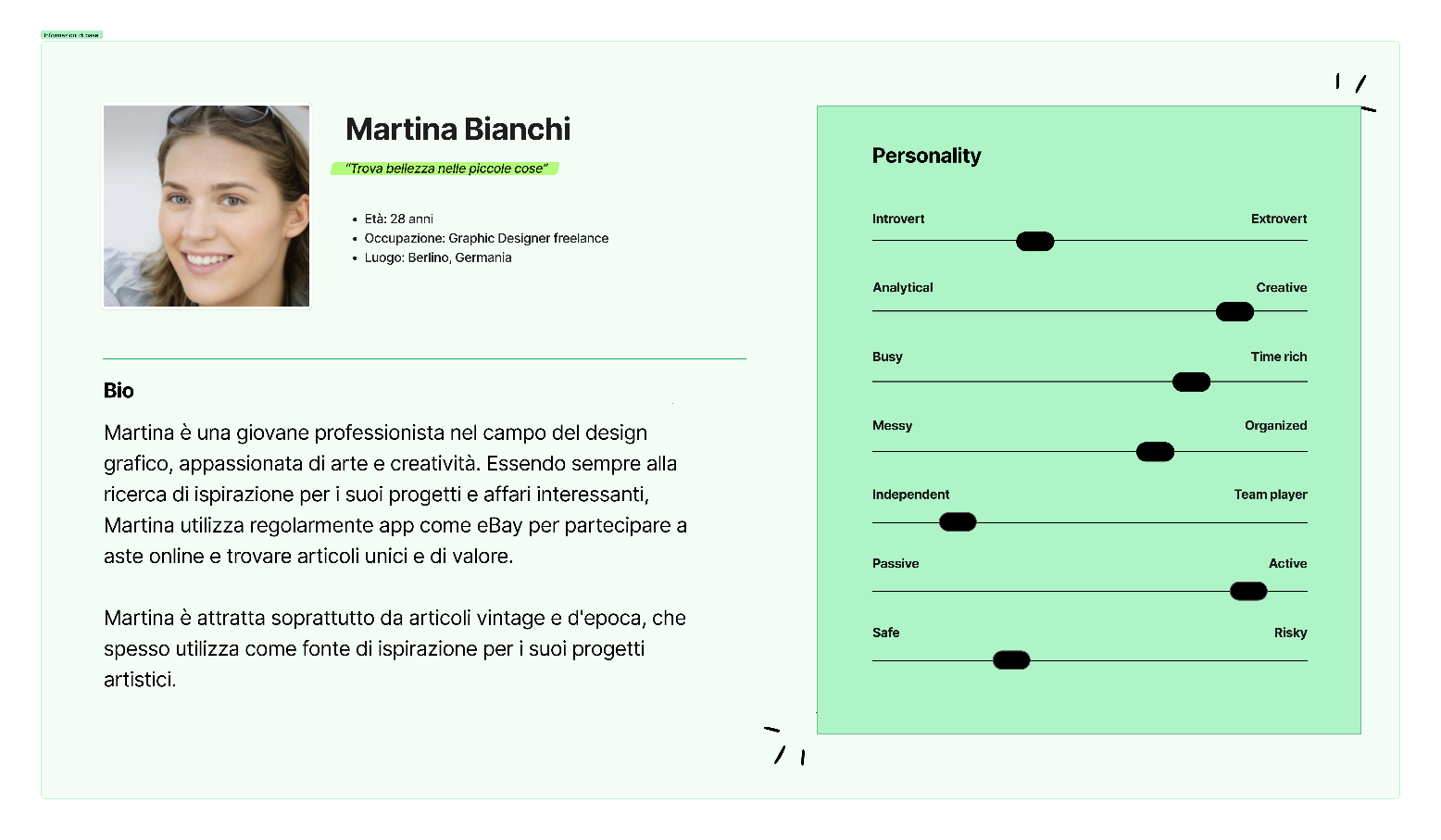


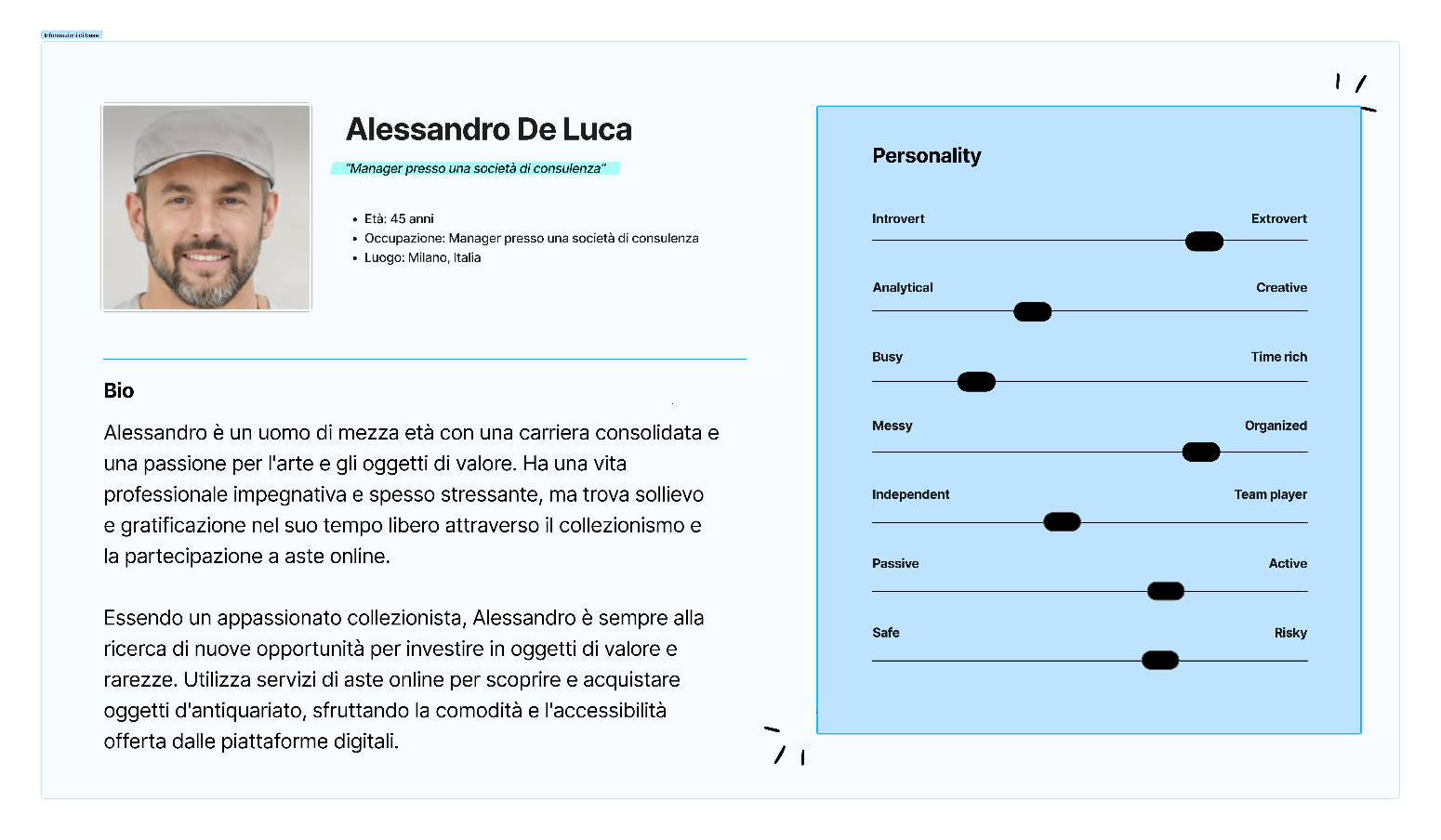


### Individuazione del Target degli Utenti

Immagine che contiene testo, Viso umano, persona, schermata

Descrizione generata automaticamente





### Valutazione dell’Usabilità

### Glossario

## Specifica dei Requisiti

### Diagrammi di Classi

### Diagrammi di Sequenza

### Statechart

Capitolo 3

# Documento del Design di Sistema

## Sviluppo Front-end

L'applicazione front-end è stata sviluppata utilizzando il linguaggio di programmazione **Swift**, garantendo una fluida esperienza utente su dispositivi Apple iOS. L’interfaccia intuitiva permette agli utenti di navigare facilmente tra le aste in corso, visualizzare dettagli su prodotti in vendita e inoltre di effettuare offerte in tempo reale.

Swift è un linguaggio di programmazione sviluppato da Apple ed è ideato per la creazione di applicativi destinati a dispositivi mobili iOS. L'adozione di Swift come linguaggio di programmazione per lo sviluppo dell'interfaccia utente offre una serie di vantaggi significativi, sia per lo sviluppo dell’applicazione che per l’usabilità del prodotto finale.

L’utilizzo di Swift come linguaggio di programmazione per lo sviluppo del front-end offre una serie di vantaggi, tra i quali:

* Offre un'eccellente varietà di servizi ed un **alto livello di mantenimento e portabilità**. Esso offre una zona di lavoro sicura per gli sviluppatori, ed a sua volta garantisce un'esperienza ottimale per gli utenti.
* Swift ha inoltre una **sintassi chiara e intuitiva**, che lo rende più accessibile anche per sviluppatori principianti e alle prime armi nel campo dello sviluppo di applicazioni front-end. Ciò ha reso possibile lo sviluppo dell’applicazione in tempi relativamente ristretti.

In sintesi, l'uso di Swift come linguaggio per lo sviluppo dell'interfaccia utente offre facilità di utilizzo e integrazione con l'ecosistema iOS, che lo rendono una scelta ideale per la creazione di app iOS moderne.

## Sviluppo Back-end

Parallelamente, il sistema back-end è stato implementato utilizzando Java e il framework Spring Boot, fornendo una base per la gestione delle logiche di business e la comunicazione con il database. Questo sistema è distribuito come container Docker, garantendo scalabilità, flessibilità e facilità di gestione.

L’utilizzo di Spring Boot come framework per lo sviluppo del back-end offre una serie di vantaggi, tra i quali:

* I meccanismi di configurazione sono per la maggior parte automatizzati, e ciò permette agli sviluppatori di concentrarsi sulla scrittura del codice, dei test e sulla gestione degli errori, senza doversi preoccupare di configurazioni complesse.
* Spring Boot fa inoltre parte dell'ecosistema Spring, che offre un'ampia gamma di librerie e moduli per affrontare una varietà di requisiti nel mondo dello sviluppo di applicazioni enterprise.
* È offerto un supporto completo per la creazione di API RESTful, che sono essenziali per la comunicazione tra il front-end e il back-end. Tramite Spring Boot è relativamente semplice definire i controller REST per gestire le richieste HTTP in ingresso, implementando operazioni CRUD (Create, Read, Update, Delete) su risorse come aste, utenti e offerte.
* La gestione delle dipendenze Maven all'interno di un'applicazione è semplificata dalo strumento Spring Initializr, che permette agli sviluppatori di avviare nuovi progetti con una configurazione predefinita e selezionare le dipendenze desiderate con pochi clic, semplificando così il processo di configurazione del progetto.

## Diagrammi di Classi di Design

## Diagrammi di Sequenza di Design

Capitolo 4

# Testing e Valutazione dell’Usabilità

## Unit Testing

## Valutazione dell’Usabilità