

Documentazione Ingegneria del Software

Silviu Filote, Simone Ronzoni, Jonathan Bommarito, Nicolò Carissimi

January 25, 2022

Contents

1	Software Life Cycle	2
2	Configuration Management	2
3	People Management and Team Organization	3
4	Requirement Engineering	4
5	Software Architecture	8
6	Software Design	10
7	Software Testing	10
8	Software Maintenance	11

1 Software Life Cycle

Per il progetto si è deciso di utilizzare il modello Agile Scrum. Siccome non vi è la necessità di comunicare volta per volta con il cliente, ad ogni sprint verrà eletto un product owner tra i diversi componenti del gruppo che avrà il compito di:

- pianificare il lavoro per lo sprint corrente;
- assicurarsi che le task completate siano state chiuse;

2 Configuration Management

Il sistema di configuration management utilizzato è GitHub, sia per la creazione dei task che per il versioning del codice.

Per quanto riguarda la creazione dei task, partendo dalle note presenti nel Backlog, si creano gli Issue che si decide di implementare per lo sprint corrente e vengono assegnati ad un membro del team.

Per la gestione e il monitoraggio dell'avanzamento degli issue viene usato il project board presente su GitHub con il template "Basic kanban".

Per il completamento di ogni task si è utilizzata la seguente modalità:

- Creazione del branch utilizzando come naming convention il numero progressivo del task
- Implementazione del codice necessario per completare il task
- Aggiunta e commit delle modifiche sul proprio branch locale
- Push branch da origin a remote
- Push modifiche sul branch remoto
- Apertura merge request verso main

- Assegnazione review e compilazione merge request
- Approvazione dei reviewers
- Merge
- Cancellazione branch remoto e locale

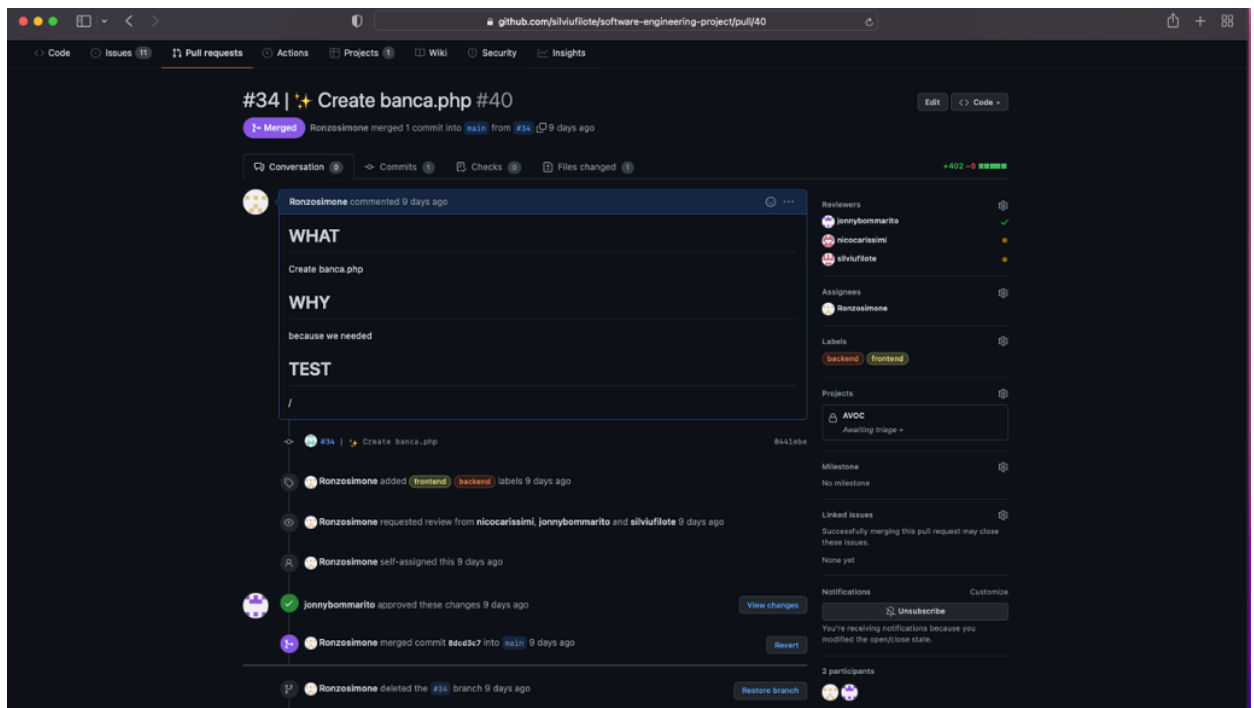


Figure 1: Merge Request tipica su GitHub

3 People Management and Team Organization

Siccome nel progetto si è utilizzato un approccio agile, l'organizzazione all'interno del team non ha una rappresentazione rigida. Nel caso in cui si palesino difficoltà, queste vengono esposte nello stand up meeting, chiarite e, se necessario, vengono schedate delle sessioni di Pair programming.

4 Requirement Engineering

1. Introduction

(a) **Purpose**

Avoc è un web app realizzata per il concessionario Boninsegna s.r.l. che consenta ai clienti di richiedere un preventivo per comprendere la convenienza tra un noleggio e un leasing, consentendo un'eventuale sottoscrizione.

(b) **Scope**

Il prodotto che si vuole realizzare automatizza le funzionalità descritte nel punto precedente. Lo scopo del servizio è di agevolare i clienti del concessionario attraverso le funzionalità offerte online. Questo permetterà di estendere la clientela passando dai soli utenti locali, ad un bacino più ampio.

(c) **Definitions, acronyms and abbreviations**

Admin: per admin ci si riferisce ad un dipendente del concessionario che possiede il privilegio di aggiungere, rimuovere o modificare macchine.

Utente: per utente si intende un qualsiasi utilizzatore del sito che non sia admin.

(d) **Overview**

Nella sezione 2 si trova un overview generale del sistema, nella sezione 3 vengono invece descritti i requisiti funzionali e non funzionali.

2. Overall description

(a) **Product perspective**

Si utilizza il database già presente all'interno dell'organizzazione, contenente tutte le informazioni relative alle vetture, e si aggiunge la tabella relativa agli utenti e quella relativa finanziamenti che essi stipulano.

(b) **Product functions**

- Funzionalità per visualizzare macchine e sottoscrivere contratti per l'utente;
- Funzionalità di gestione del parco macchine per gli admin;

Le funzionalità vengono esposte in automatico, in base al tipo di utenza inserita in fase di login.

(c) **Constraints**

Gli utenti possono solo richiedere un preventivo e visualizzare i finanziamenti sottoscritti, non sono abilitati ad eseguire modifiche relative alle auto.

3. Specific requirements

(a) **Hardware interfaces**

L'interfaccia utente è responsive, così che possa adattarsi alla dimensione dello schermo di qualsiasi dispositivo.

(b) **Software interfaces**

La connessione al database è stata possibile tramite l'estensione mysqli.

(c) **Functional requirements**

i. USER

- SignIn

L'utente ha a disposizione una finestra di dialogo dove inserire i parametri per effettuare il login. i suddetti parametri sono il codice utente e la propria password. In caso quest'ultimo sia andato a buon fine, l'utente sarà stato correttamente loggato

- SignUp

L'utente ha a disposizione una finestra di dialogo dove inserire i parametri per effettuare la registrazione tra cui:

- Nome
- Cognome
- CodiceFiscale
- DataDinascita
- E-mail
- Password

- CalcolaFinanziamento

L'utente può scegliere dal menù a tendina il brand, il modello e la versione di una macchina presente nel concessionario. Successivamente viene calcolato il prezzo e viene evidenziato il button per la configurazione del finanziamento. Attraverso la modale sono richiesti i seguenti parametri:

- Tipologia
- RataMensile
- Anticipo
- Chilometreggio

ii. ADMIN

- SignIn

L'admin ha a disposizione una finestra di dialogo dove inserire i parametri per effettuare il login tra cui il codice utente e la propria password. In caso quest'ultimo sia andato a buon fine, l'utente sarà stato correttamente loggato con privilegi amministratore

- gestisci Macchine

All'interno di questa view è possibile eseguire le seguenti operazioni:

- inserimento Macchina

Per questa funzione viene visualizzata una modale che permette l'inserimento delle nuove auto. i parametri richiesti sono:

- * marca
- * modello
- * versione
- * AnnoImmatricolazione
- * prezzo
- * peso
- * lunghezza
- * larghezza
- * posti

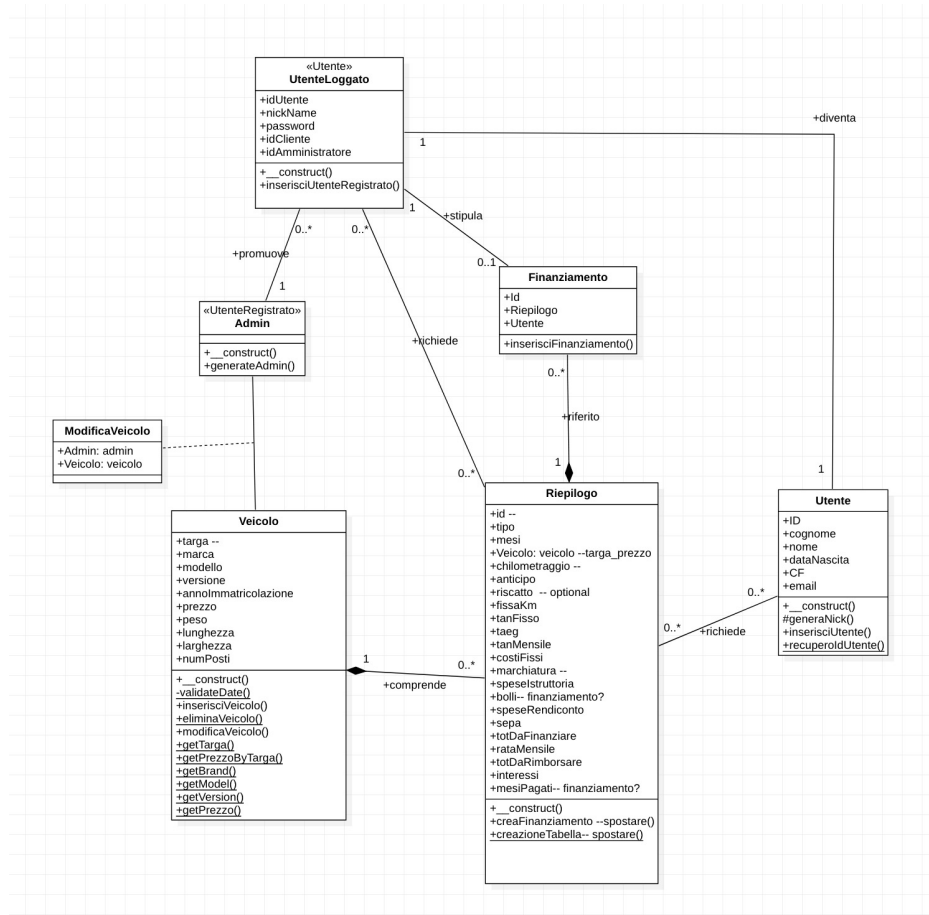
- Modifica Macchina

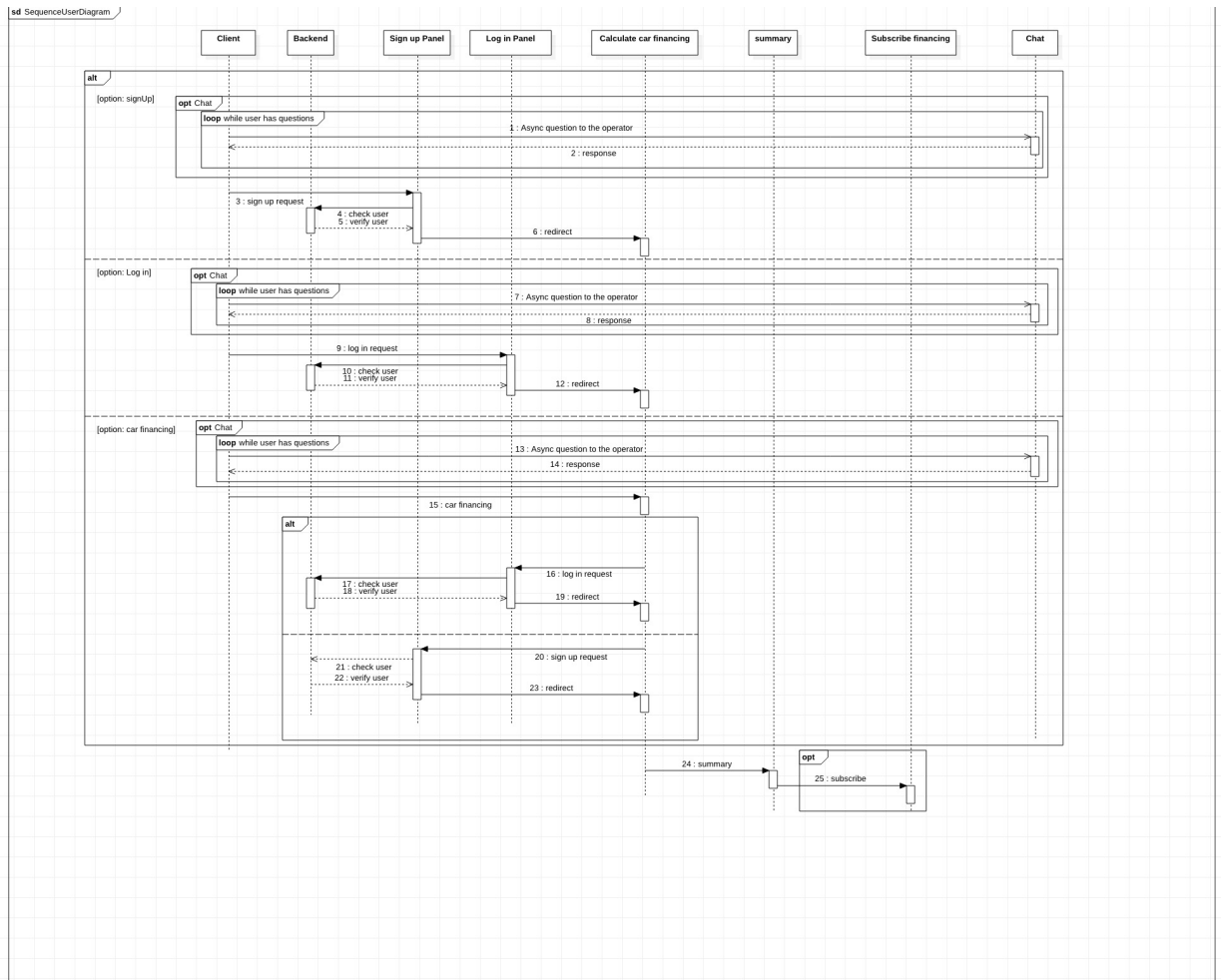
Per questa funzione è necessario cliccare sul campo che si vuole modificare e confermare attraverso il pulsante 'Modifica'

– Elimina Macchina

Per questa funzione è necessario cliccare sull'apposito pulsante dedicato nella colonna 'Azioni' della tabella dei record delle macchine.

5 Software Architecture





6 Software Design

DESIGN PATTERN

Delegation:

La classe admin è in grado di utilizzare tutte le funzioni della classe veicolo, tra cui:

- aggiungi veicolo;
- elimina veicolo;
- modifica veicolo;

7 Software Testing

Test documentation based on IEEE 928

- **Test plan:**
il piano di test consiste nell'importare tutte le classi con le relative dipendenze e testare tutti i metodi cercando di ottenere la massima copertura possibile.
- **Test design specification:**
i nomi dei componenti richiamano le funzionalità che svolgono
 - Admin.php
 - Riepilogo.php
 - Utente.php
 - UtenteRegistrato.php
 - Veicolo.php

- **Test procedure specification:**
Se i metodi sono statici vengono richiamati passando gli opportuni parametri, per i metodi non statici vengono creati degli oggetti di supporto.
- **Test item transmittal report:**
Gli elementi testati si trovano nella cartella ./classi e verranno testati tutti i metodi all'interno degli stessi.
- **Test log:**
Testing a partire dalle classi meno popolate fino ad arrivare a quelle più complesse.
- **Test incident report:**
Nulla da trasmettere.
- **Test summary report:**
I test sono stati esaustivi e hanno evidenziato alcuni importanti bug.

8 Software Maintenance

Le attività di Refactor sono state le seguenti:

- Refactor della homepage suddividendola in più moduli per favorire la granularità;
- Introduzione del PDO nelle query al fine di migliorare la sicurezza;
- Gestito il redirect tramite .htaccess;