

<b>Obelix (C5)</b> <b>Presentazione: 24</b>	<b>Giudizio complessivo sui documenti: 19</b>
Consegna e considerazioni generali	<p><u>Consegna</u>: insufficiente attenzione nella verifica. <u>Lettera di Presentazione</u>: l'indicazione della data prevista di consegna non è sufficientemente precisa. La denominazione dell'affiliazione del committente è errata. <u>Organizzazione dei documenti</u>: Il nome dei documenti non ne riporta il numero di versione. Inoltre, essi non includono la lista di distribuzione. <u>Registro delle modifiche</u>: per facilità di consultazione, il registro delle modifiche va ordinato per versione (quindi chiave di prima colonna), dalla più recente alla più lontana. Le modifiche registrate dovrebbero essere più precisamente localizzabili. Riferimenti: dei documenti soggetti a versionamento (p.es., Glossario) serve specificare la versione riferita. <u>Verbali</u>: i verbali, distinti tra interni ed esterni, devono essere raccolti in apposite cartelle dedicate. Essi dovranno contenere un resoconto tracciabile delle decisioni prese. <u>Versionamento e approvazione</u>: non è buona prassi che un documento venga rilasciato all'esterno prima dell'approvazione da parte del responsabile. Questo accade, per esempio, per le Norme di Progetto.</p>
Presentazione	Impianto grafico da migliorare. Qualità di esposizione discreta. Ottima introduzione alla visione generale del prodotto.
Studio di Fattibilità	Bene.
Norme di Progetto	<p>Struttura discreta, che si sforza di aderire alla <i>best practice</i> suggerita, ma le interpreta in modo ingenuo. La struttura gerarchica adottata (e conforme) pone al primo livello le categorie di processi e al secondo gli specifici processi di interesse; per questo motivo, l'espressione "processo di" nei titoli di secondo livello è ridondante. Alcuni processi di supporto, potenzialmente di interesse strategico del fornitore (versionamento, configurazione?), sono stati omessi. Quelli organizzativi paiono poco conosciuti. Contenuti insufficienti per ampiezza e per profondità, soprattutto per accompagnare le attività tecniche (sviluppo e verifica), che sono destinate a diventare preponderanti con il superamento della RR. Nel complesso, da <b>rivedere</b> secondo le indicazioni.</p>
Analisi dei Requisiti	<p>§1.4.2: rivedere la formattazione. §2.1 deve scendere in maggior dettaglio. Inoltre, deve includere un diagramma che riporti gli attori del sistema e le eventuali relazioni tra essi. UC0: perché UC1 non è considerato una "funzionalità" alla stregua di tutti i sotto-casi di UC2? Lo scenario principale degli eventi deve essere <b>sempre</b> presente nella descrizione dei casi d'uso, anche se ovvio. UC1: quali informazioni devono essere fornite per creare una bolla? Vi sono più tipi di bolle? Le funzionalità offerte dall'API devono essere descritte in maggior dettaglio. Quanto avete per ora fornito può rappresentare al più un punto di partenza. UC3-cv: chi è l'attore principale del caso d'uso? La "Fonte Esterna" è un attore secondario; in realtà, è una operazione eseguita dal sistema e pertanto non è tra quelle da descrivere in un documento di AR. Devono invece esserne descritti gli effetti. Analogamente per UC2-mt, UC4-tr. UC1-ls: sono presenti due sotto-casi con medesimo codice identificativo. RObFu011.2: requisito troppo generico. Il rapporto tra requisiti e casi d'uso è 1:1, il che segnala una analisi molto superficiale. Prima di poter iniziare la stesura dell'architettura, l'analisi deve essere molto più approfondita, così da comprendere bene quali funzionalità il sistema dovrà fornire. RObQu010: non è misurabile. Non sono presenti requisiti di qualità, che devono porre vincoli al processo di sviluppo. RObDi008 è di qualità. RObDi001-cv è funzionale. RObDi007 è opportuno specificare quale <i>framework</i> viene utilizzato. Quali piattaforme saranno supportate? Quali <i>browser</i>? Che versioni? Manca il tracciamento casi d'uso – requisiti.</p> <p>Nella versione attuale, il documento è solo un inizio di AR. L'analisi deve essere approfondita e migliorata il processo con il quale la svolgete. Correggere gli errori UML, approfondire i requisiti e rivedere la loro classificazione. Documento da <b>rivedere</b>, inserendo il tracciamento mancante.</p>

Piano di Progetto	<p>§1.5: il progetto didattico non attraversa l'intero di ciclo di vita del <i>software</i>. Quello che vi interessa è limitato al modello di sviluppo. (Attenzione all'uso, intermittente, del termine "fase".)</p> <p>§2: buona l'analisi dei rischi, ma la presentazione narrativa e a lista ne diminuisce l'efficacia. Meglio invece una struttura tabellare, che è di più immediata consultazione e anche induce alla sintesi. Buona l'attualizzazione (e non "attuazione": evitate il copia-incolla acritico, che porta a importare errori inconsapevolmente) dell'analisi al periodo di rendicontazione.</p> <p>§3: la pianificazione che descrivete pare guidata solo dalla produzione dei documenti richiesti in ingresso alle revisioni di progresso, invece che dalle esigenze di sviluppo vere e proprie (come dovrebbe invece derivare dalla scelta di un modello incrementale). §3.3: la quantità di impegno riservata per il consolidamento dei requisiti appare esigua e quindi ottimistica.</p> <p>§6.7.2: i costi, computati secondo le regole del capitolato, non sono arrotondabili, né per difetto, perché applicare uno "sconto" al committente vi mette a rischio di sforamenti, né per eccesso, perché il preventivo riporta (e deve riportare) il costo reale.</p> <p>§7: includere il consuntivo di periodo nel PdP serve primariamente a trarne indicazioni su come raffinare e correggere il preventivo del periodo rimanente. Non basta dunque riportare gli scostamenti, ma serve invece ragionare sulla loro origine e su come evitarli in futuro.</p> <p>Documento buono per organizzazione e discreto per contenuti, ma da <b>rivedere</b> secondo le indicazioni.</p>
Piano di Qualifica	Documento insoddisfacente per interpretazione di struttura (ereditata acriticamente) e profondità di contenuti. Da <b>rivedere</b> in profondità.
Glossario	Bene.