



**Progetto MYTHOS**

## **System Design Document**

Partecipanti

Egidio Mario 0512105122

Pagliari Vincenzo 0512105546

Project Manager

De Lucia Andrea

Pecorelli Fabiano

Sommario

- 1 Introduzione ..... 3**
  - 1.1 Scopo del sistema..... 3
  - 1.2 Obiettivi di design ..... 3
- 2 Architettura software proposta..... 5**
  - 2.1 Panoramica ..... 5
  - 2.2 Decomposizione del sistema ..... 6
  - 2.3 Mapping hardware/software..... 6
  - 2.4 Gestione dati persistenti..... 8
  - 2.5 Controllo degli accessi e della sicurezza ..... 8
  - 2.6 Gestione del controllo globale ..... 9
  - 2.7 Condizioni boundary ..... 9
- 3 Servizi dei sottosistemi ..... 10**

# 1. Introduzione

## 1.1. Scopo del sistema

Nei locali italiani la maggior parte dei problemi è dovuto alla difficoltà di 'gestione' degli ordini, del magazzino e principalmente del controllo del rendiconto finale.

Infatti, dopo aver dialogato con un gestore di una discoteca (Mythos) è sorta la sua difficoltà nell'organizzazione interna del locale. Esso, infatti, trova difficoltà nel mantenere le informazioni relative ad ogni tavolo del privé (es. budget rimanente, #persone ecc.) e nell'inoltare i relativi ordini in modo efficiente alla cambusa la quale si occupa di preparare l'ordine, proprio perché essendo una discoteca il locale risulta alquanto confusionale.

Un'altra difficoltà riscontrata è quella di mantenere correttamente sotto controllo la quantità di ogni prodotto in magazzino, un rendiconto dei guadagni di ogni serata e, soprattutto, uno storico di tutti gli ordini effettuati.

Un problema secondario è dovuto dal fatto che la discoteca ha a disposizione alcune persone (dette p.r., addette alle relazioni pubbliche) che si occupano di formare delle liste (composte da nome e cognome) che consentono l'ingresso al locale delle persone presenti nella lista.

Siccome queste liste possono essere modificate anche durante la serata stessa, risulta complicato mantenerle sempre aggiornate in formato cartaceo, pertanto si vuole automatizzare anche questo aspetto.

## 1.2. Obiettivi di design

Il sistema Mythos deve essere, quanto più possibile, efficiente ed intuitivo in maniera tale da non far perdere molto tempo nella fase di ordinazione e prenotazione. Tale efficienza sarà raggiunta grazie a tempi di risposta del sistema molto rapidi. Sarà realizzata un'interfaccia molto intuitiva e l'efficienza sarà soddisfatta grazie all'uso di un DBMS il quale consente operazioni di lettura e salvataggio di dati persistenti in maniera efficiente ed ottimizzata. Inoltre, si punterà ad avere una buona manutenibilità attraverso il facile inserimento di nuove funzionalità permesso grazie all'utilizzo di un linguaggio object-oriented come Java.

Per ottenere tutti gli obiettivi finali, verranno rispettati i seguenti criteri di progettazione.

### 1.2.1. Criteri di performance

<b>Tempo di risposta</b>	Il sistema deve essere molto reattivo soprattutto nella fase di ordinazione e relativa stampa dell'ordine (la quale deve avvenire entro 5 secondi dalla conferma dell'ordine). Tuttavia, essendo un sistema web che funziona in locale la reattività dell'applicativo dovrebbe essere garantita in ogni condizione a patto che non ci siano problemi di connessione di rete.
<b>Throughput</b>	Il sistema deve garantire picchi di carico, fino a 10 utenti collegati simultaneamente, senza rallentamenti e soprattutto garantire consistenza dei dati in tutte le operazioni che richiedono un accesso al database.
<b>Memoria</b>	Il sistema utilizza un DBMS per la memorizzazione dei dati pertanto, sicuramente, non ci sarà grande mole di dati da gestire. Inoltre, alcuni dati saranno gestiti attraverso un sistema che consente di mantenerli in memoria volatile; questo fa in modo che siano subito a disposizione senza che venga effettuato un accesso al DBMS; questo per garantire maggiore efficienza in alcune operazioni la cui persistenza non è importante averla subito.

### 1.2.2. Criteri di Affidabilità

<b>Disponibilità</b>	Il sistema deve essere sempre disponibile alle eventuali richieste degli utenti.
<b>Robustezza</b>	Il sistema deve poter mantenere i dati anche in caso di problemi elettrici a patto che non ci siano guasti alla memoria persistente della macchina principale. Il sistema deve sempre informare l'utente con messaggi di errore l'inserimento di dati errati.
<b>Sicurezza</b>	Le tecniche utilizzate, per garantire la sicurezza, sono basate prevalentemente su un 'login', la quale permette il riconoscimento dell'utente. Le funzionalità messe a disposizione dell'utente vengono settate nella fase di autenticazione in base al tipo di utente che ha effettuato l'accesso pertanto solo un determinato tipo di utente può accedere a determinate funzioni.

### 1.2.3. Criteri di Costo

<b>Costi di sviluppo e installazione</b>	I costi di sviluppo prevedono la retribuzione di 350 euro per ogni componente del team per un totale di 700 euro. Il server finalizzato per ospitare "la parte online" del sistema è stato offerto da un conoscente pertanto non sono previsti costi. Software come Visual Paradigm, Balsamic e Microsoft Office sono stati utilizzati con la licenza studente, quindi gratuita.
<b>Costi di manutenzione</b>	Per i costi di manutenzione, l'utente finale ha a disposizione 3 mesi per individuare e comunicare tutti i bug del sistema i quali saranno risolti gratuitamente; dopo questo periodo di tempo per ogni bug sarà applicata una specifica tariffa. Per quanto riguarda i miglioramenti del sistema questi sono esclusi dai costi di sviluppo pertanto per ogni richiesta sarà applicata una specifica tariffa valutata a seconda del miglioramento.
<b>Costi amministrativi</b>	Poiché non sono stati presi accordi riguardo questo aspetto, per il momento non sono previsti costi amministrativi annui.

### 1.2.4. Criteri di manutenzione

<b>Estensibilità e Modificabilità</b>	Il sistema sarà implementato utilizzando linguaggi come: HTML5, CSS3, JavaScript, Java e PHP. Questo consente la facile estensibilità del sistema poiché viene utilizzato un paradigma object-oriented il quale consente di aggiungere nuove funzionalità o di modificare quelle già esistenti in maniera del tutto efficiente.
<b>Portabilità</b>	La portabilità sarà garantita da un server Tomcat il quale consente all'applicativo di essere eseguito su tutte le piattaforme.
<b>Leggibilità</b>	Il codice sarà ben strutturato, documentato e commentato pertanto si capirà con rapidità la funzione di ogni singola operazione del sistema.
<b>Tracciabilità dei requisiti</b>	Il codice verrà strutturato quanto più possibile secondo le categorie dei requisiti specificati, questo per consentire l'efficiente tracciabilità di essi e la possibilità di effettuare modifiche necessarie al corretto funzionamento di ogni singolo requisito.

### 1.2.5. Criteri per l'utente finale

<b>Utilità</b>	Il sistema deve essere utile all'utente finale soprattutto da un punto di vista economico e gestionale. Si richiede, infatti, che il sistema aiuti il moderatore nel mantenimento consistente dei dati del magazzino e che calcoli in maniera precisa il rendi conto delle prenotazioni.
<b>Usabilità</b>	Il sistema deve essere di facile apprendimento. Maggiore attenzione è richiesta nella fase di ordinazione in quanto gli operatori che effettuano gli ordini possono variare nel tempo; si ha bisogno, pertanto, di un'interfaccia molto semplice e intuitiva la quale consente di padroneggiare le funzionalità in rapido.

## 2. Architettura software proposta

Il software che si vuole sviluppare consiste in un'automazione di un sistema attualmente cartaceo.

### 2.1. Panoramica

Il sistema proposto è un software web-based che automatizza la gestione di una discoteca facilitando la gestione degli ingressi in lista dei clienti, le prenotazioni e le ordinazioni ai tavoli e la gestione di tutti i prodotti in magazzino.

Le tipologie di utenti si dividono in:

- Moderatore, colui che gestisce la discoteca.
- Cameriere, colui che effettua le ordinazioni ai tavoli.
- Cassiere, colui che gestisce le prenotazioni e gli ingressi in lista.
- P.R., colui che gestisce gli inserimenti in lista.
- Utente Registrato, colui che non ha ancora effettuato l'accesso alla piattaforma.

Il moderatore sarà l'amministratore della piattaforma e potrà accedere alle funzionalità che si occupano di gestire i prodotti, i p.r., le liste di clienti, le ordinazioni e le prenotazioni;

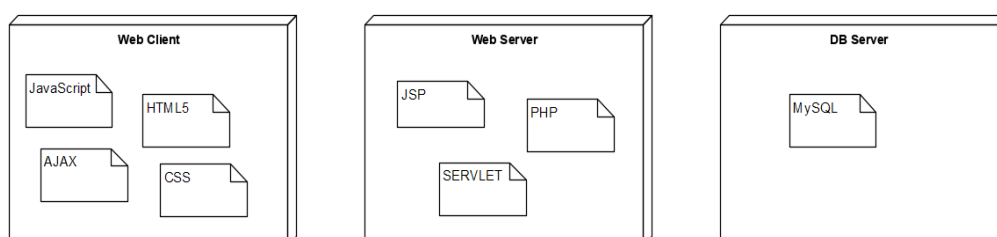
Il cameriere prenderà le ordinazioni ai tavoli;

Il cassiere potrà gestire l'ingresso dei clienti che sono già in lista e potrà, inoltre, gestire le prenotazioni ai tavoli, visualizzandole, inserendone di nuove, eliminandole e così via;

Il p.r. gestirà l'inserimento in lista dei clienti.

L'architettura del sistema si presenta come una architettura Client/Server a tre livelli per i seguenti motivi:

- Il sistema (essendo web-based) potrà essere utilizzato su una varietà macchine e sistemi operativi.
- Il sistema sarà in grado di fornire i propri servizi al singolo utente senza interferire con gli altri utenti del sistema.
- Il sistema garantirà reattività per i task degli utenti collegati; questo è possibile poiché i dispositivi sono collegati sotto la stessa rete (locale), pertanto la latenza sarà molto bassa.
  - Eccezione da segnalare, però, riguarda i P.R., essi, infatti, accederanno al sistema situato su un server online pertanto la reattività dipenderà dalla loro connessione attuale.
- Il sistema sarà in grado di gestire gli utenti che effettuano disparate operazioni in contemporanea.
- Il sistema fornirà un'interfaccia grafica intuitiva e con funzionalità specifiche per il ruolo dell'utente che effettuerà l'accesso.



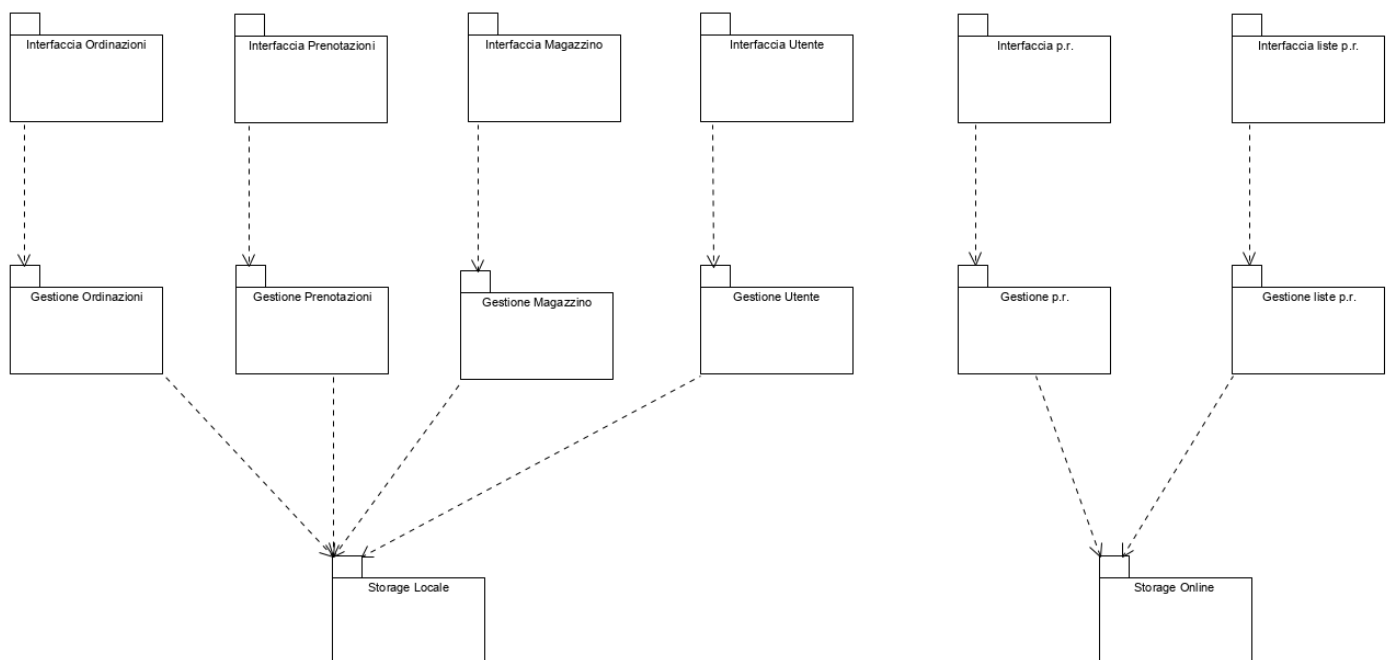
## 2.2. Decomposizione del sistema

Per la realizzazione del software Mythos, si è scelto un misto tra un'architettura three-layer in versione Client/Server e un'architettura basata sul pattern MVC. Questa particolare architettura software prevede la divisione dell'applicazione in 3 strati, dedicati rispettivamente all'interfaccia utente, alla logica di business e alla gestione dei dati persistenti.

- **Il presentation Layer:** include tutte le interfacce grafiche e in generale i boundary objects, come i form con cui l'utente interagisce e la logica di controllo. L'interfaccia verso l'utente è rappresentata maggiormente da contenuti statici (es. pagine HTML).
- **L'application Layer:** include tutti gli oggetti relativi all'elaborazione dei dati. Questo livello interagisce con lo storage layer affinché possa generare contenuti dinamici e accedere ai dati persistenti.
- **Lo storage Layer:** effettua la memorizzazione, il recupero e l'interrogazione degli oggetti persistenti. I dati, i quali possono essere acceduti dall'application layer, sono depositati in maniera persistente su un database tramite DBMS.

Il sistema comprende 6 sottosistemi

- **Ordinazioni**, composto da Interfaccia Ordinazioni, Gestione Ordinazioni e Storage Locale.
- **Prenotazioni**, composto da Interfaccia Prenotazioni, Gestione Prenotazioni e Storage Locale.
- **Magazzino**, composto da Interfaccia Magazzino, Gestione Magazzino e Storage Locale.
- **Utenti**, composto da Interfaccia Utente, Gestione Utente e Storage Locale.
- **P.R.**, composto da Interfaccia p.r., Gestione p.r. e Storage Online.
- **Liste p.r.**, composto da Interfaccia Liste p.r., Gestione Liste p.r. e Storage Online.



## 2.3. Mapping hardware/software

Dalle esigenze espresse del committente è necessario scomporre il sistema in due strutture hardware separate, infatti:

La prima struttura hardware (che definiremo come "struttura locale") si può definire come la struttura principale dell'intera piattaforma poiché tutte le principali operazioni vengono eseguite su questa configurazione.

Questa, infatti, è composta da una macchina principale chiamata "Server" all'interno della quale sono incapsulati il Web Server Tomcat che si occupa di produrre l'output in base alle varie richieste dei client e il DBMS (Database

Management System) il quale si occupa della persistenza dei dati e il relativo prelevamento in maniera del tutto efficiente; la struttura è, inoltre composta dai client che non sono altro che devices come PC, Tablet e Smartphone (Mobile) che interagiscono esclusivamente con il Server.

Tutti questi componenti saranno connessi fra di loro attraverso un'unica rete, quella locale.

I client che desiderano accedere alla piattaforma, infatti, dovranno, attraverso l'utilizzo di un browser, collegarsi all'indirizzo IP locale (assegnato staticamente) del Server specificando la porta sulla quale il Web Server è in ascolto e fare delle richieste.

L'altra struttura hardware (che definiremo come "struttura online") riguarda, invece, una parte più piccola del sistema, infatti qui vengono coinvolti solo la maggior parte dei servizi offerti dai sottosistemi: "Gestione liste p.r." e "Gestione p.r.".

Il mapping hardware è pressoché identico a quello descritto in precedenza ma presenta comunque delle piccole differenziazioni, infatti:

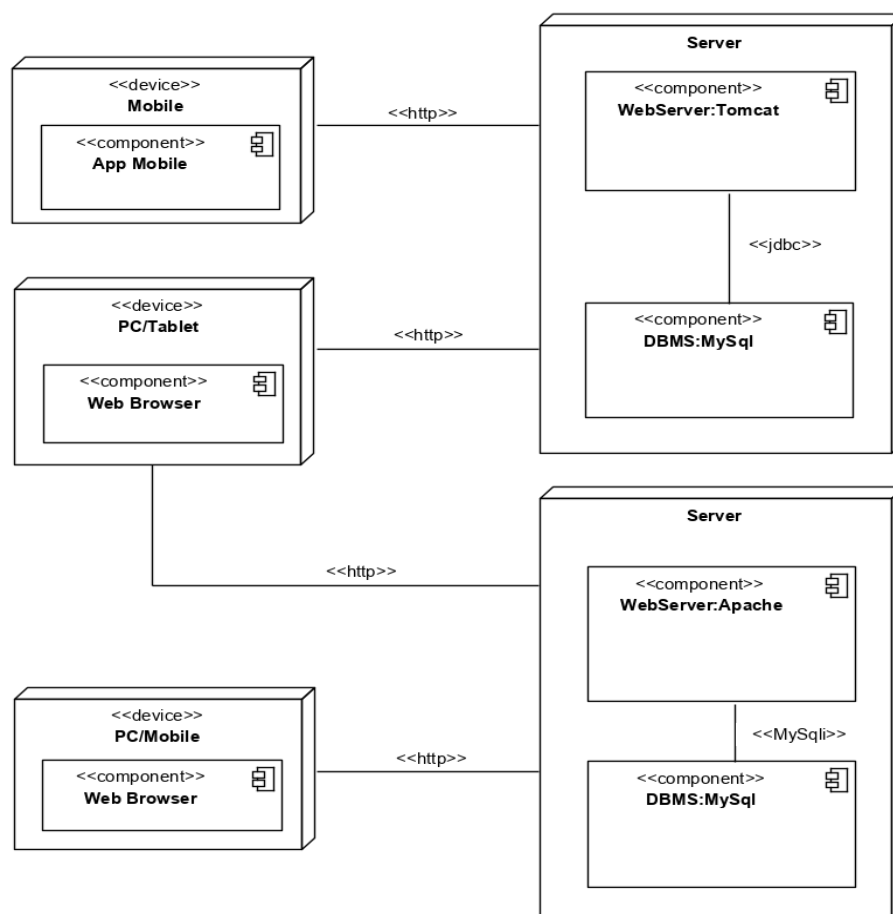
I client (PC o smartphone) in questo caso però, si collegheranno alla piattaforma (con l'aiuto di un browser) attraverso un dominio registrato ad un host online (Server) il quale risponderà alle loro richieste; ovviamente questo è responsabile soltanto dei servizi dei sottosistemi citati in precedenza.

Il server online presenta comunque la stessa struttura interna però al suo interno troviamo un Web Server Apache il quale interagirà con il DBMS attraverso l'estensione "MySQLi".

Una forte precisazione riguarda il fatto che i client appartenenti alla struttura locale ed in particolare i PC e i Tablet possono interagire con la struttura online richiedendo dei particolari servizi che sono offerti solo dall'application server online.

I client e i server, in entrambe le strutture descritte, comunicheranno attraverso il protocollo http e saranno connesse ad una rete che utilizza il protocollo TCP/IP.

Segue lo schema delle strutture sopra descritte:



## 2.4. Gestione dati persistenti

Si rimanda al file "DBD.pdf"

## 2.5. Controllo degli accessi e della sicurezza

Mythos è un sistema multiutente la cui autenticazione alla piattaforma è indispensabile per usufruire dei servizi che la piattaforma offre. Ogni singolo utente, dopo essersi loggato alla piattaforma attraverso username e password, avrà accesso ad un determinato insieme di funzionalità assegnate dal sistema nella fase di autenticazione e scelte in base al ruolo che quell'utente assume nel sistema.

Per comprendere meglio le operazioni che ogni tipo di utente può svolgere, di seguito viene rappresentata una matrice degli accessi nella quale sarà possibile consultare, per ogni attore, quali servizi sono ad esso riservati. Ma non verrà implementata al primo rilascio del sistema.

Attori	Moderatore	Cassiere	Cameriere	P.R.
Gestore				
Utente	<ul style="list-style-type: none"><li>VerificaUtente</li></ul>	<ul style="list-style-type: none"><li>VerificaUtente</li></ul>	<ul style="list-style-type: none"><li>VerificaUtente</li></ul>	<ul style="list-style-type: none"><li>VerificaPr</li></ul>
Ordine	<ul style="list-style-type: none"><li>VisualizzazioneOrdine</li></ul>		<ul style="list-style-type: none"><li>Nuovo Ordine</li><li>Conferma Ordine</li><li>Conferma Dettagli</li><li>AggiungiProdottoAlCarrello</li><li>EliminaProdottoDalCarrello</li><li>EliminaDettaglioDalCarrello</li><li>AggiungiDettaglioAlCarrello</li></ul>	
Prenotazione	<ul style="list-style-type: none"><li>AzzeraPrenotazioni</li><li>Rendiconto Tavoli</li><li>ModificaCostoPrenotazioni</li><li>ModificaTipoPrenotazioneTavolo</li><li>ListaTavoli</li><li>RicavaCostoPrenotazioni</li></ul>	<ul style="list-style-type: none"><li>ElencoPrenotazioni</li><li>ModificaPrenotazione</li><li>NuovaPrenotazione</li><li>CancellazionePrenotazione</li><li>TavoliDisponibili</li><li>ListaTavoli</li><li>RicavaCostoPrenotazioni</li></ul>	<ul style="list-style-type: none"><li>InfoPrenotazione</li><li>ListaTavoli</li></ul>	
Magazzino	<ul style="list-style-type: none"><li>InserimentoNuovoProdotto</li><li>VisualizzaTuttiProdotti</li><li>ModificaProdotto</li><li>CancellazioneProdotto</li><li>RipristinoProdottoCancellato</li><li>VisualizzazioneSoloProdottiBar</li></ul>		<ul style="list-style-type: none"><li>VisualizzazioneSoloProdottiCassa</li><li>VisualizzaDettagliOrdine</li></ul>	
Liste P.R.	<ul style="list-style-type: none"><li>Abilita/Disabilita inserimento in lista</li><li>AzzeramentoListaClienti</li><li>VisualizzazioneListaClientiPr</li><li>visualizzaStatoListe</li></ul>	<ul style="list-style-type: none"><li>ModificaStatoClientiInLista</li><li>RicercaClientiInLista</li></ul>		<ul style="list-style-type: none"><li>VisualizzazioneListaClientiSingoloPr</li><li>InserimentoClientiInLista</li><li>CancellazioneClientiDaLista</li></ul>
P.R.	<ul style="list-style-type: none"><li>InserimentoNuovoPr</li><li>CancellazionePr</li><li>VisualizzazionePr</li></ul>			



## **2.6. Gestione del controllo globale**

Il controllo del flusso software viene gestito dai controller che interagendo con il client, il quale si interfaccia tramite un web browser, svolgono le varie operazioni. Il server smista ogni nuova richiesta ai controller adeguati, inoltrando poi la risposta al client. Il sistema Mythos utilizzerà un controllo del flusso event-driven, in quanto prevede che il flusso del programma è largamente determinato dal verificarsi di eventi esterni. Gli eventi esterni saranno innescati (triggered) dagli utenti collegati alla piattaforma tramite i link ed i pulsanti presenti sulle pagine. Un event “handler” provvederà poi a prendere in gestione la richiesta ed inoltrarla al dispatcher. Quest’ultimo effettuerà la chiamata verso il servizio che può prenderla in carico e gestirla.

Inoltre, per permettere l’invio degli ordini in contemporanea da parte di più camerieri, il sistema Mythos dovrà fornire dei meccanismi di sincronizzazione quando si conferma l’ordine. Nel dettaglio quando si deve confermare un ordine va resa come operazione atomica la sequenza di operazioni “Verifica disponibilità prodotto” e “Scala quantità prodotto

## **2.7. Condizioni Boundary**

Le condizioni limite riguardano, dal lato Server, l’accensione e lo spegnimento del sistema. Riguardano invece, dal lato Client, gli errori di connessione al server.

Le condizioni limite vanno definite tenendo conto che una parte dell’applicativo verrà pubblicato su un host online e l’altra parte sarà gestita in una rete locale.

Avendo due server diversi l’avvio, la terminazione e il fallimento del sistema saranno gestiti in due modi diversi.

### **Sistema Locale**

#### **2.7.1. Configurazione del sistema**

La configurazione della piattaforma Mythos consisterà nell’inserimento, da parte dell’amministratore, dei vari utenti che potranno autenticarsi alla piattaforma così come concordato con il cliente.

#### **2.7.2. Avvio del sistema**

Sul server saranno registrati come servizi il WebServer e il DBMS, di conseguenza per avviare il sistema Mythos al moderatore basterà avviare il PC che fa da server.

Dopo l’avvio del sistema, esso presenterà per ogni client un’interfaccia di login.

Ogni utente, dopo aver effettuato l’autenticazione, potrà accedere alle proprie funzionalità.

#### **2.7.3. Terminazione del sistema**

Prima della completa terminazione e disattivazione di database-server e di application-server, tutti i client connessi al sistema dovranno essere disconnessi dal sistema. Prima della disattivazione di application-server e database-server è consigliabile la disattivazione di tutti i client attraverso la funzione di logout.

#### **2.7.4. Fallimento del sistema**

Nel caso si verifichi un errore riguardante il software o l’hardware, si tenterà il ripristino del sistema alla configurazione iniziale. Essendo i dati gestiti dai DBMS, il rischio di perdita è quasi nullo. Tuttavia, è da tenere in conto un eventuale guasto al supporto di memorizzazione dei dati nel database-server che potrebbe portare alla perdita di tali dati. Questo rischio verrà minimizzato eseguendo manualmente dei backup del database.

### **Sistema Online**

Il sistema online non è sotto il controllo degli amministratori Mythos pertanto, l’avvio e la terminazione non sono gestibili. Il sistema, essendo pubblicato su un host online, sarà sempre attivo e quindi sempre raggiungibile, tranne nel caso in cui si presentano eventuali fallimenti. Per poter gestire un fallimento del sistema si dovrà contattare un amministratore dell’host tramite ticket (segnalazione) che si occuperà di risolvere le problematiche.

### 3. Servizi dei sottosistemi

#### 3.1. Gestione Ordinanze

<b>Sottosistema</b>	Gestione Ordinazione
<b>Descrizione</b>	Sottosistema che gestisce tutte le operazioni che riguardano le ordinazioni e che consentono di gestire il ciclo di vita di un ordine.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
VisualizzazioneOrdine	Questa funzionalità permette di visualizzare tutti gli ordini effettuati ad uno specifico tavolo
NuovoOrdine	Questa funzionalità permette di selezionare un tavolo per cui effettuare un'ordinazione e di selezionare un metodo di pagamento tra quelli disponibili.
ConfermaOrdine	Questa funzionalità permette di confermare l'ordine con i prodotti/dettagli presenti nel carrello
ConfermaDettagli	Questa funzionalità permette di confermare l'ordine con i soli dettagli presenti nel carrello
AggiungiProdottoAlCarrello	Questa funzionalità permette di inserire un nuovo prodotto nel carrello
EliminaProdottoDalCarrello	Questa funzionalità permette di eliminare un prodotto dal carrello
EliminaDettaglioDalCarrello	Questa funzionalità permette di inserire un nuovo dettaglio nel carrello
AggiungiDettaglioAlCarrello	Questa funzionalità permette di eliminare un dettaglio dal carrello
EliminaOrdine	Questa funzionalità permette di eliminare un ordine memorizzato sullo storage

#### 3.2. Gestione Prenotazioni

<b>Sottosistema</b>	Gestione Prenotazione
<b>Descrizione</b>	Sottosistema che gestisce tutte le operazioni che riguardano le prenotazioni dei tavoli.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
ElencoPrenotazioni	Questa funzionalità consente di visualizzare tutti i tavoli prenotati
ModificaPrenotazione	Questa funzionalità consente di modificare i campi di una prenotazione
NuovaPrenotazione	Questa funzionalità consente di registrare una nuova prenotazione per un tavolo
CancellazionePrenotazione	Questa funzionalità consente di eliminare una prenotazione
TavoliDisponibili	Questa funzionalità consente di visualizzare tutti i tavoli non ancora prenotati
RendicontoTavoli	Questa funzionalità consente di visualizzare, per ogni tavolo, quale è stato l'incasso
ModificaTipoPrenotazioneTavolo	Questa funzionalità consente di stabilire se un tavolo è adibito ad una prenotazione "luxus" o "normale"

ModificaCostoPrenotazione	Questa funzionalità consente di modificare il prezzo della prenotazione “luxus” e “normale”
InfoPrenotazione	Questa funzionalità consente di visualizzare le informazioni di una specifica prenotazione
AzzeraPrenotazioni	Questa funzionalità consente di azzerare tutte le prenotazioni con i relativi ordini effettuati
ListaTavoli	Questa funzionalità consente di ottenere l’elenco di tutti i tavoli.
RicavaCostoPrenotazioni	Questa funzionalità consente di ottenere il costo delle prenotazioni.

### 3.3. Gestione Magazzino

<b>Sottosistema</b>	Gestione Magazzino
<b>Descrizione</b>	Sottosistema che permette di gestire i prodotti presenti in magazzino.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
InserimentoNuovoProdotto	Questa funzionalità consente di aggiungere un nuovo prodotto nel catalogo
VisualizzaTuttiProdotti	Questa funzionalità consente di visualizzare tutti i prodotti presenti nel catalogo
ModificaProdotto	Questa funzionalità consente di modificare i campi di un prodotto
CancellazioneProdotto	Questa funzionalità consente di eliminare un prodotto dal catalogo
RipristinoProdottoCancellato	Questa funzionalità consente di ripristinare un prodotto precedentemente cancellato
VisualizzazioneSoloProdottiCambusa	Questa funzionalità consente di visualizzare tutti i prodotti presenti in cambusa nel gestionale (locazione “cambusa”)
VisualizzazioneSoloProdottiBar	Questa funzionalità consente di visualizzare tutti i prodotti presenti in bar nel gestionale (locazione “bar”)
VisualizzaDettagliOrdine	Questa funzionalità consente di visualizzare tutti i dettagli ordine che il cameriere può aggiungere durante un ordine.

### 3.4. Gestione Liste P.R.

<b>Sottosistema</b>	Gestione Liste P.R.
<b>Descrizione</b>	Sottosistema che riguarda tutte le operazioni che si possono effettuare sulle liste dei p.r.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
Abilita/Disabilita inserimento in lista	Questa funzionalità consente di abilitare o disabilitare l’inserimento dei clienti in lista da parte dei p.r.
AzzeraListaClienti	Questa funzionalità consente di azzerare tutte le liste di tutti i p.r.
ModificaStatoClienteInLista	Questa funzionalità consente di contrassegnare l’ingresso nel locale di un cliente in lista

VisualizzazioneListaClientiSingoloPr	Questa funzionalità consente di visualizzare tutti i clienti aggiunti in lista da uno specifico p.r.
InserimentoClienteInLista	Questa funzionalità consente al p.r. di aggiungere un cliente nella sua lista
CancellazioneClienteDaLista	Questa funzionalità consente al p.r. di eliminare un cliente della sua lista che ha aggiunto precedentemente
VisualizzazioneListaClientiPr	Questa funzionalità consente di visualizzare tutti i clienti aggiunti in lista da tutti i p.r.
RicercaClienteInLista	Questa funzionalità consente di cercare un cliente in lista
VisualizzaStatoListe	Questa funzionalità consente di visualizzare lo stato attuale delle liste (abilitate o disabilitate)

### 3.5. Gestione P.R.

<b>Sottosistema</b>	Gestione P.R.
<b>Descrizione</b>	Sottosistema che permette le principali operazioni di memorizzazione e recupero dati dei p.r.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
InseirmentoNuovoPr	Questa funzionalità consente di aggiungere un nuovo p.r.
CancellazionePr	Questa funzionalità consente di cancellare un p.r. esistente
VisualizzazionePr	Questa funzionalità consente di visualizzare tutti i p.r.
verificaPr	Questa funzionalità consente di verificare le credenziali dei Pr (verificaUtente di gestione utente).

### 3.6. Gestione Utenti

<b>Sottosistema</b>	Gestione Autenticazione
<b>Descrizione</b>	Sottosistema che permette le principali operazioni di recupero dati degli utenti registrati.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
VerificaUtente	Questa funzionalità consente di verificare le credenziali degli utenti.