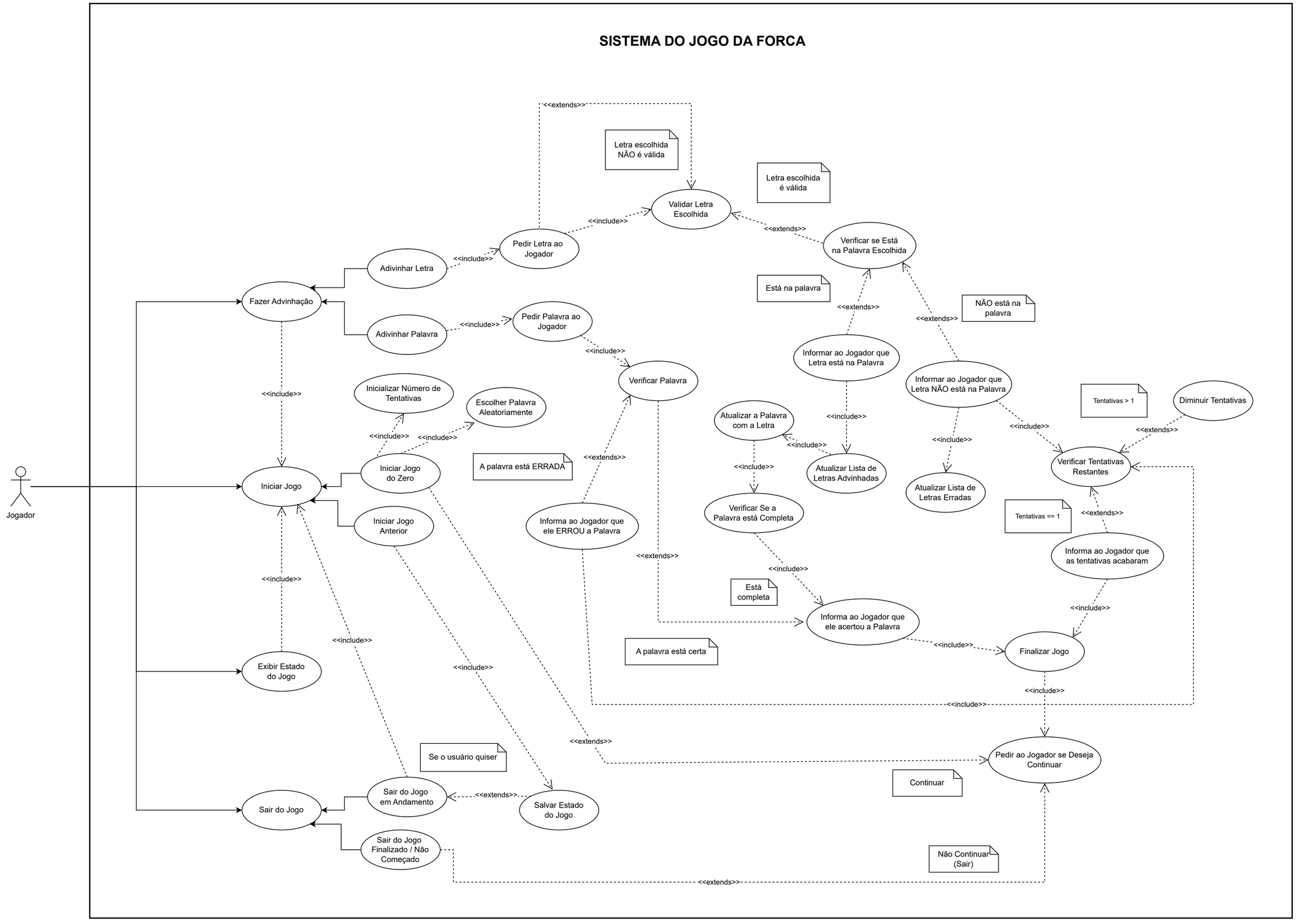
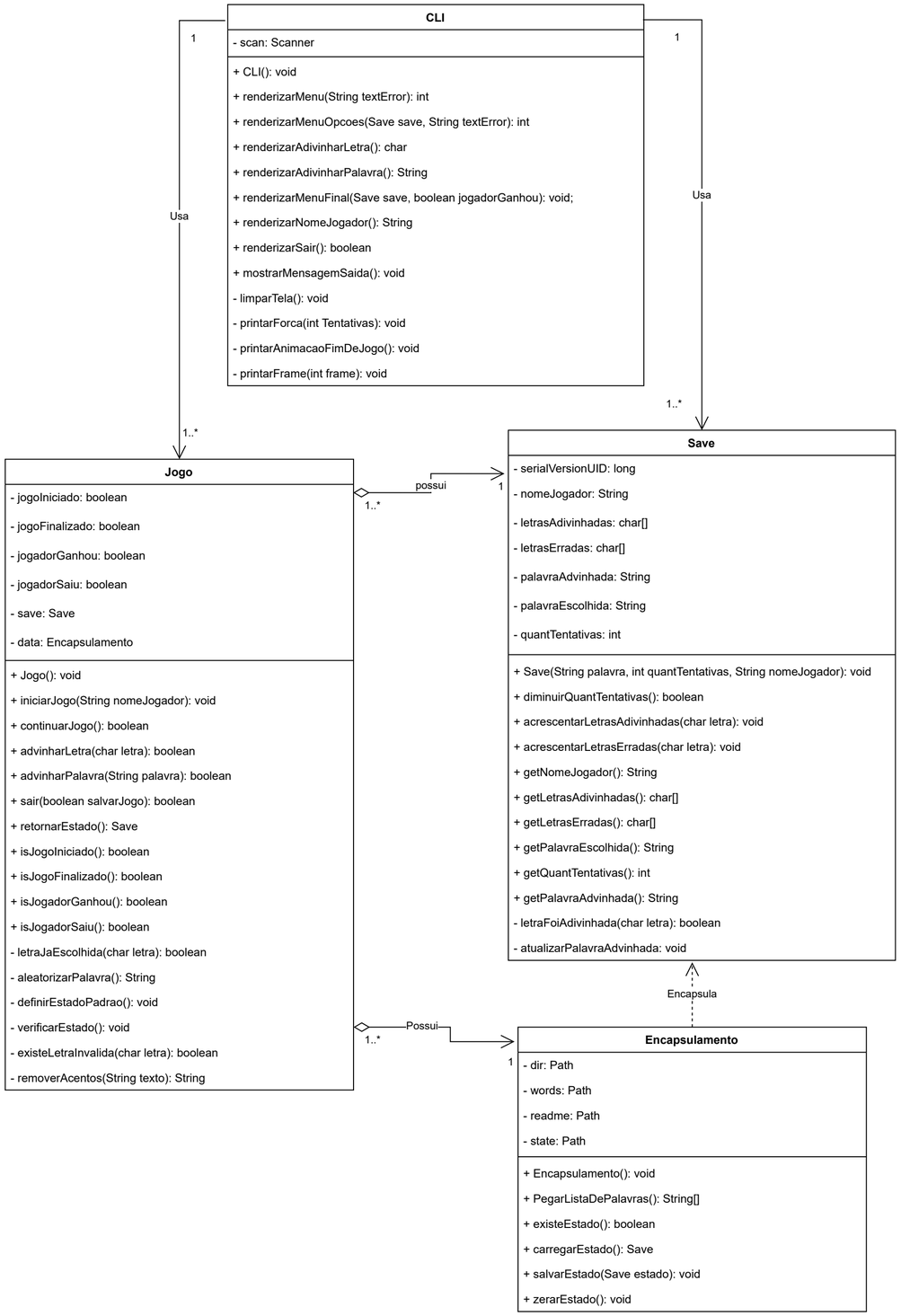
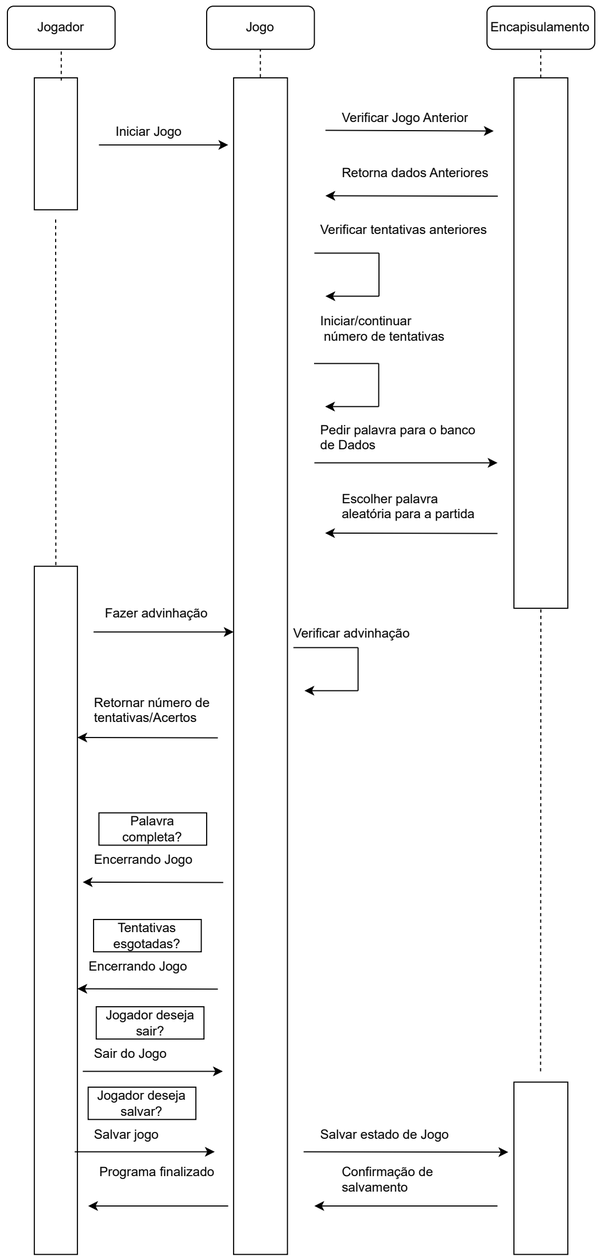
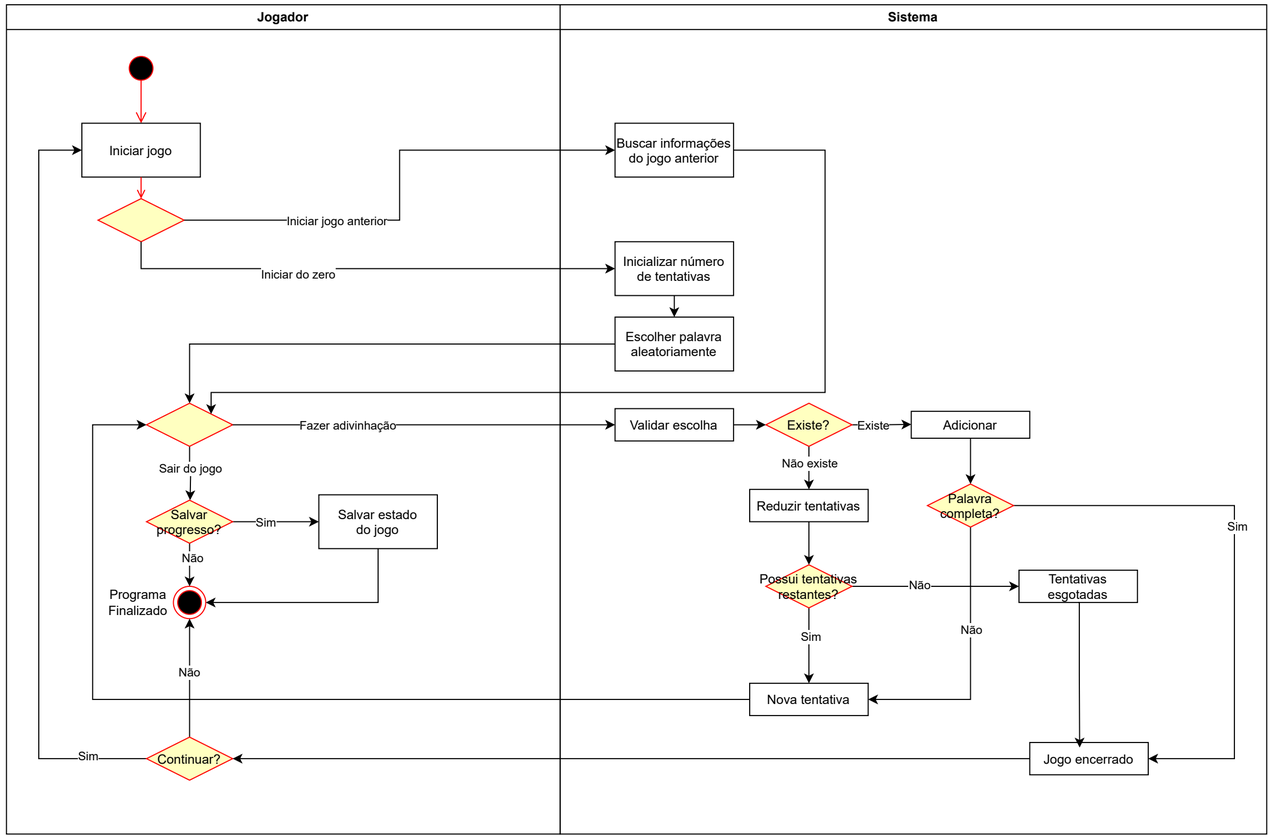
SISTEMA DO JOGO DA FORCA




CLI
- scan: Scanner
+ CLI(): void
+ renderizarMenu(String textError): int
+ renderizarMenuOpcoes(Save save, String textError): int
+ renderizarAdivinharLetra(): char
+ renderizarAdivinharPalavra(): String
+ renderizarMenuFinal(Save save, boolean jogadorGanhou): void;
+ renderizarNomeJogador(): String
+ renderizarSair(): boolean
+ mostrarMensagemSaida(): void
- limparTela(): void
- printarForca(int Tentativas): void
- printarAnimacaoFimDeJogo(): void
- printarFrame(int frame): void
1
Usa
1..*
1
Usa
1..*
Jogo
- jogoIniciado: boolean
- jogoFinalizado: boolean
- jogadorGanhou: boolean
- jogadorSaiu: boolean
- save: Save
- data: Encapsulamento
+ Jogo(): void
+ iniciarJogo(String nomeJogador): void
+ continuarJogo(): boolean
+ advinharLetra(char letra): boolean
+ advinharPalavra(String palavra): boolean
+ sair(boolean salvarJogo): boolean
+ retornarEstado(): Save
+ isJogoIniciado(): boolean
+ isJogoFinalizado(): boolean
+ isJogadorGanhou(): boolean
+ isJogadorSaiu(): boolean
- letraJaEscolhida(char letra): boolean
- aleatorizarPalavra(): String
- definirEstadoPadrao(): void
- verificarEstado(): void
- existeLetraInvalida(char letra): boolean
- removerAcentos(String texto): String
possui
1..*
1
Save
- serialVersionUID: long
- nomeJogador: String
- letrasAdivinhadas: char[]
- letrasErradas: char[]
- palavraAdvinhada: String
- palavraEscolhida: String
- quantTentativas: int
+ Save(String palavra, int quantTentativas, String nomeJogador): void
+ diminuirQuantTentativas(): boolean
+ acrescentarLetrasAdivinhadas(char letra): void
+ acrescentarLetrasErradas(char letra): void
+ getNomeJogador(): String
+ getLetrasAdivinhadas(): char[]
+ getLetrasErradas(): char[]
+ getPalavraEscolhida(): String
+ getQuantTentativas(): int
+ getPalavraAdvinhada(): String
- letraFoiAdivinhada(char letra): boolean
- atualizarPalavraAdvinhada: void
Encapsula
Possui
1..*
1
Encapsulamento
- dir: Path
- words: Path
- readme: Path
- state: Path
+ Encapsulamento(): void
+ PegarListaDePalavras(): String[]
+ existeEstado(): boolean
+ carregarEstado(): Save
+ salvarEstado(Save estado): void
+ zerarEstado(): void


Jogador
Jogo
Encapisulamento
Iniciar Jogo
Verificar Jogo Anterior
Retorna dados Anteriores
Verificar tentativas anteriores
Iniciar/continuar número de tentativas
Pedir palavra para o banco de Dados
Escolher palavra aleatória para a partida
Fazer advinhação
Verificar advinhação
Retornar número de tentativas/Acertos
Palavra completa?
Encerrando Jogo
Tentativas esgotadas?
Encerrando Jogo
Jogador deseja sair?
Sair do Jogo
Jogador deseja salvar?
Salvar jogo
Salvar estado de Jogo
Programa finalizado
Confirmação de salvamento


Jogador
Sistema
Iniciar jogo
Iniciar jogo anterior
Iniciar do zero
Buscar informações do jogo anterior
Inicializar número de tentativas
Escolher palavra aleatoriamente
Fazer adivinhação
Sair do jogo
Salvar progresso?
Sim
Salvar estado do jogo
Não
Programa Finalizado
Não
Continuar?
Sim
Validar escolha
Existe?
Existe
Adicionar
Não existe
Reduzir tentativas
Palavra completa?
Sim
Possui tentativas restantes?
Não
Tentativas esgotadas
Sim
Não
Nova tentativa
Jogo encerrado