Use Case Diagram for JMessenger App

JMessenger is an app that:
- Provides a repository for software development projects
- Authorized users can connect to the secured access database.
- Authorized users can query the database.
- Authorized users can save files to the database.
- Authorized users can receive files from the database.
- Authorized users can save files to local storage.
- The database will provide version control.
- Authorized users can access the database remotely via vpn to send and receive files from outside the intranet.

Use Case ID / Link:	USE CASE	**Use Case Name:**	Send Code
Description	Send code in JMessenger		
Primary Actor or Persona	JMessenger USER		
Pre-Conditions	*None*		
Post-Conditions	*None*		
Triggers	*User clicks the send*		
Main/Happy Path	*User opens program* *User pastes code into message window* *User selects send* *Message / file sent to recipient* *User selects receive* *Waiting message / file are retrieved*		
Alternative / Exception Flows	*User opens program* *User selects remote send* *Message / file sent to recipient*		
Use Case Frequency	*2 Times per user per day*		
Non-functional Requirements	*Not Applicable*		

Use Case ID / Link:	USE CASE	Use Case Name:	Receive
Description		Receive code in JMessenger	
Primary Actor or Persona		JMessenger USER	
Pre-Conditions		*Code exists to receive*	
Post-Conditions		*None*	
Triggers		*User clicks the receive*	
Main/Happy Path		*User opens program* *User selects receive* *Message / file received to recipient*	
Alternative / Exception Flows		*User opens program* *User selects remote receive* *Message / file received*	
Use Case Frequency		*2 Times per user per day*	
Non-functional Requirements		*Not Applicable*	

Use Case ID / Link:	USE CASE	Use Case Name:	Save
Description		Save and Open in the JMessenger	
Primary Actor or Persona		JMessenger USER	
Pre-Conditions		*none*	
Post-Conditions		*none*	
Triggers		*User clicks the save button* *User clicks the open button*	
Main/Happy Path		● *User opens program* ● *User sends and receives messages / files* ● *User selects save as* ● *Pop-up window with save destination location* ● *User inputs name for file* ● *User selects save* ● *File saved with name in destination location*	
Alternative / Exception Flows		● *User opens program* ● *User selects open* ● *Pop-up window with files to open* ● *User selects what file to open* ● *File is opened and displayed to user*	
Use Case Frequency		*20 per day*	
Non-functional Requirements		*n/a*	