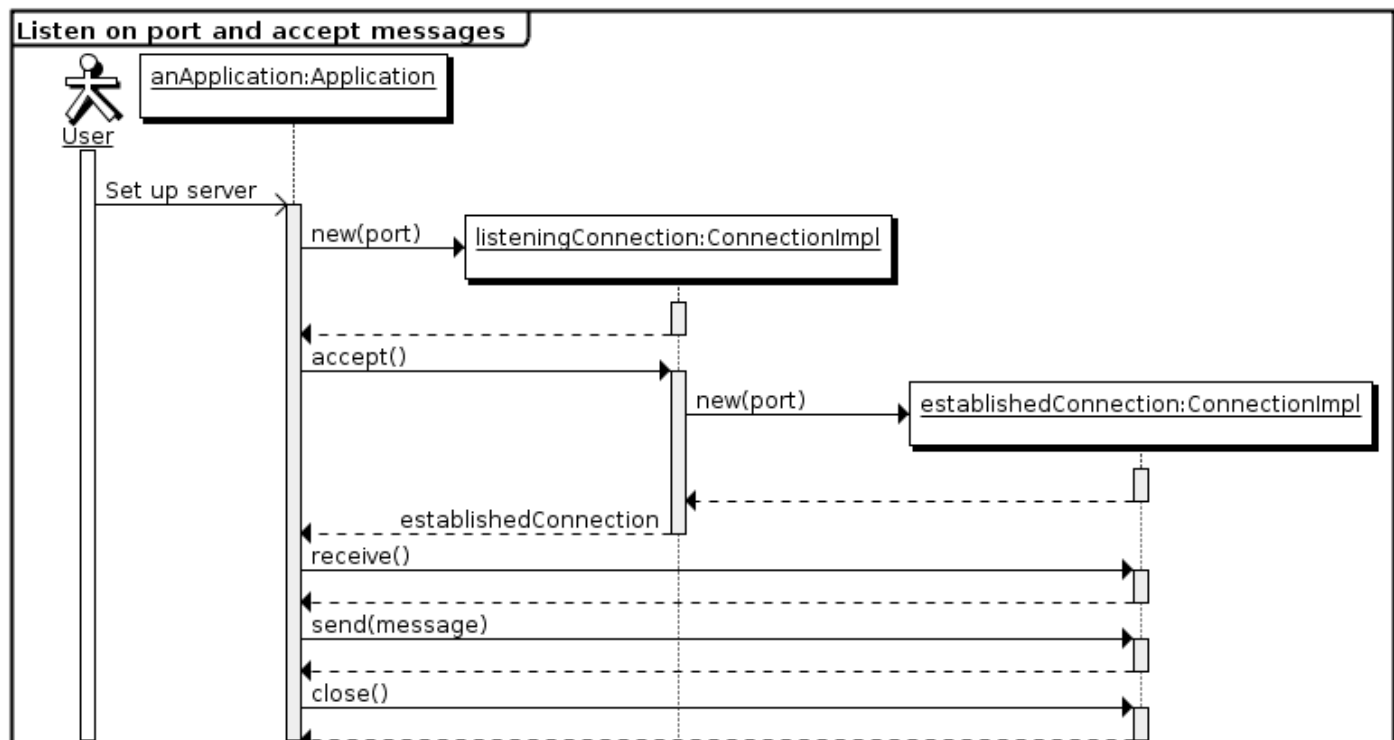
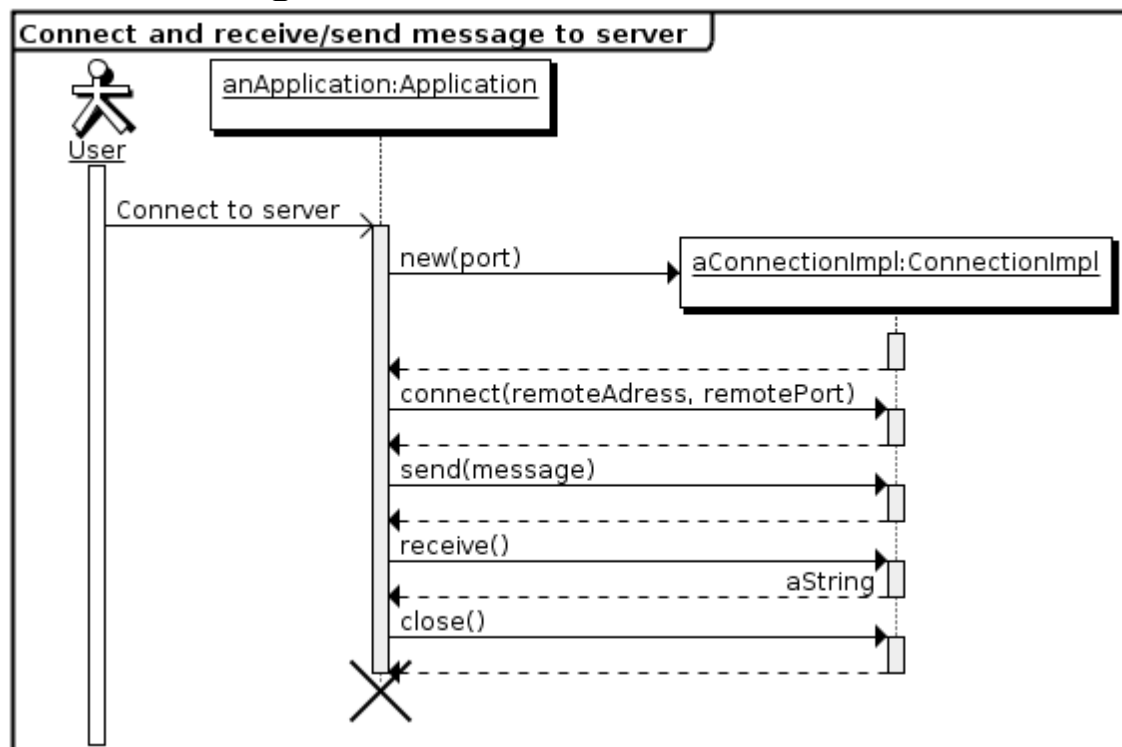
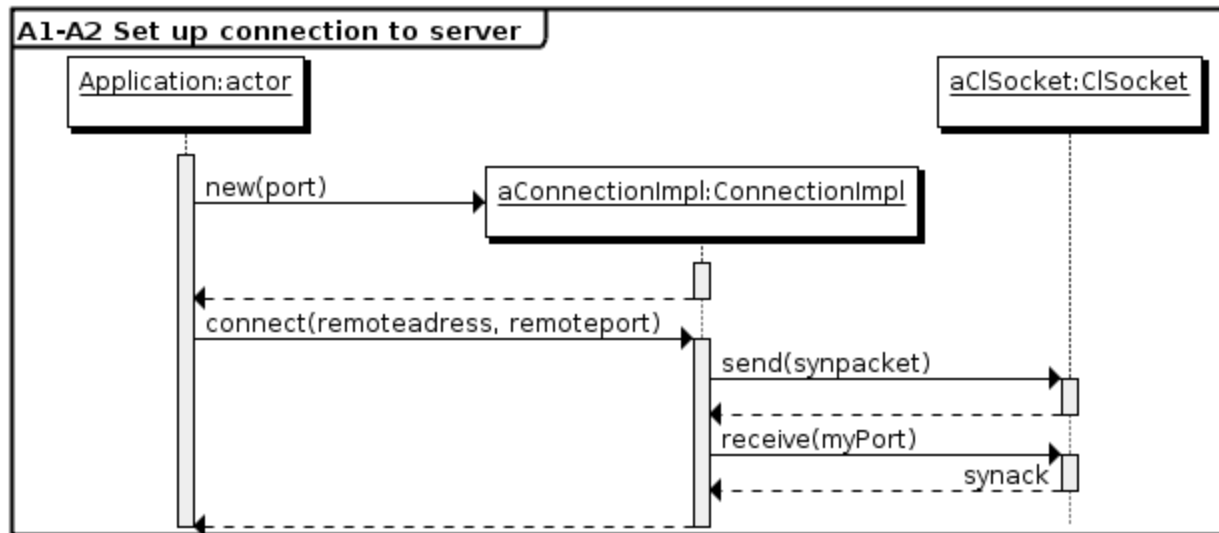
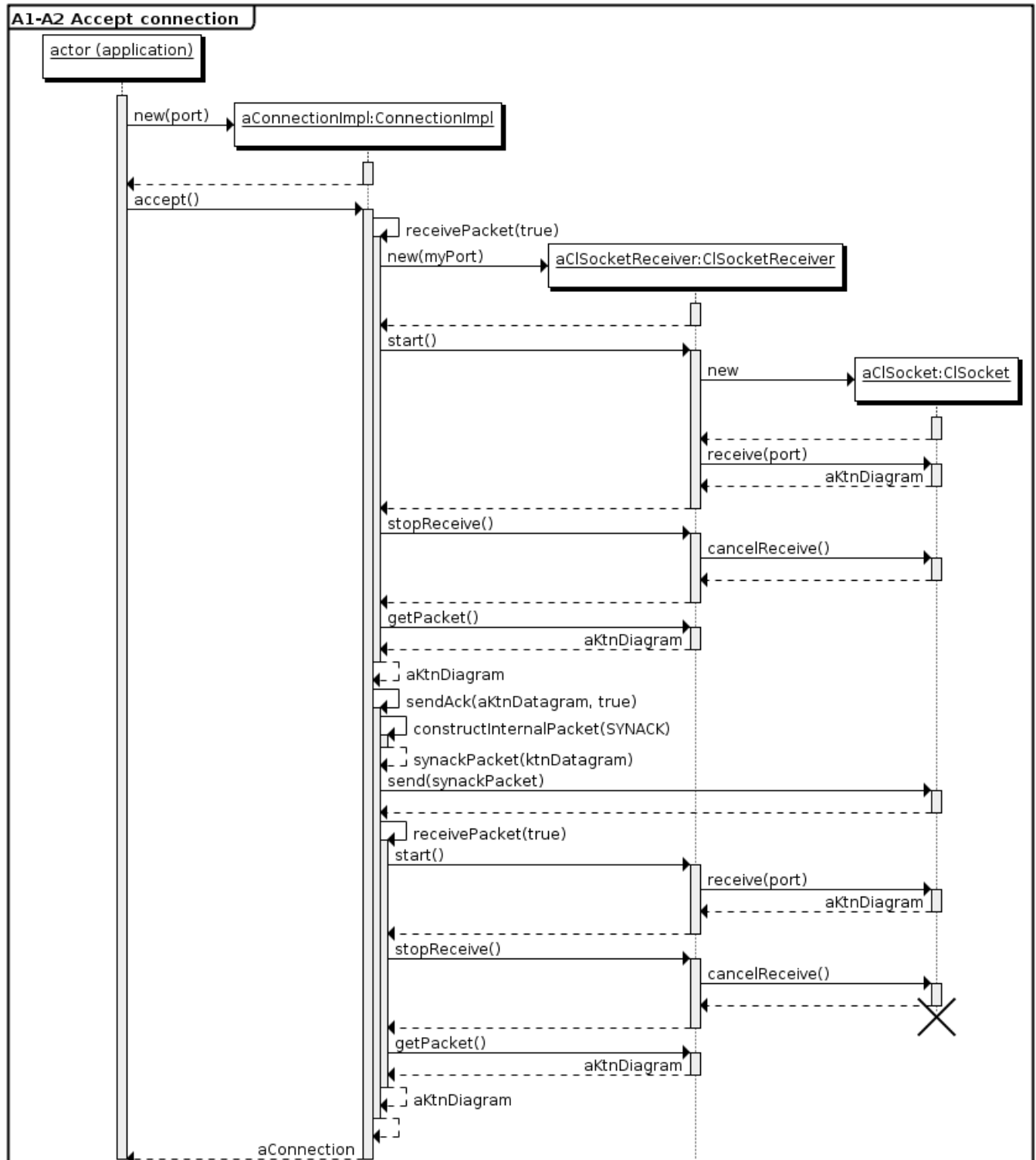


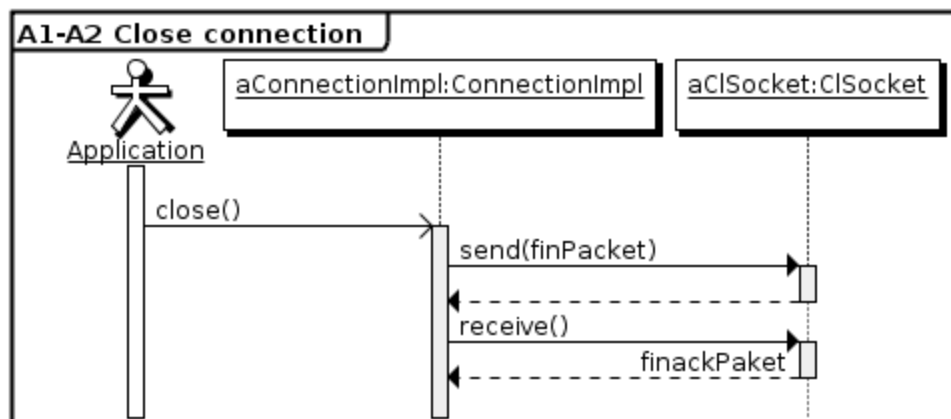
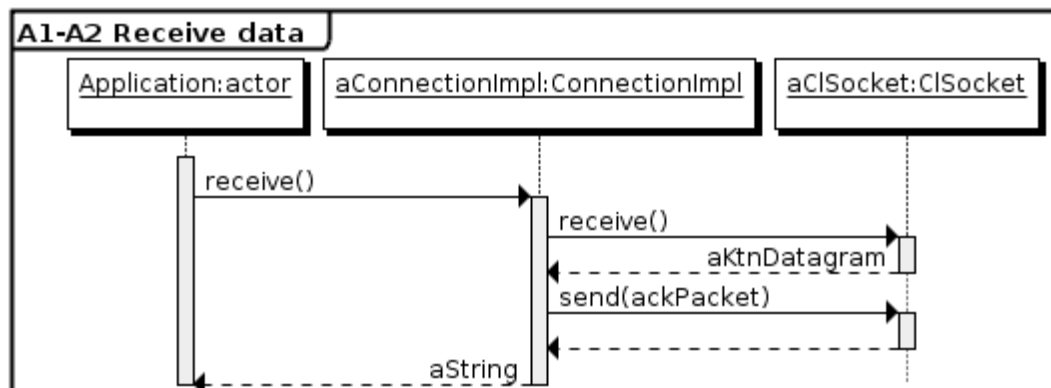
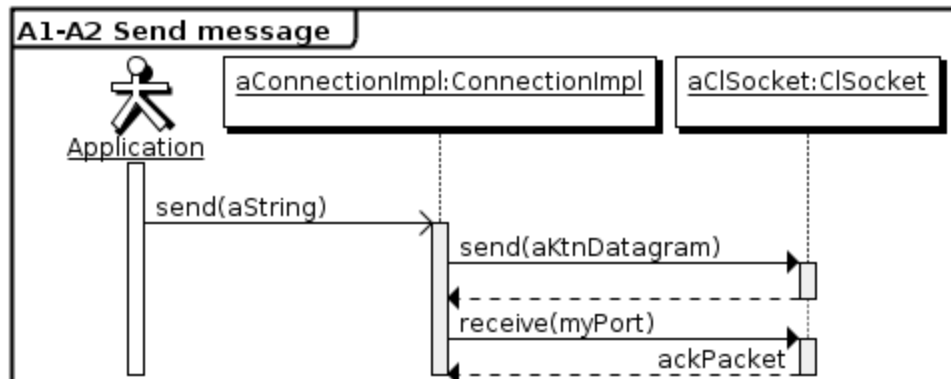
KTN1 phase I

Sekvensdiagrammer









Overordnet beskrivelse:

Vi bruker Admin til logging og for å sette sannsynligheter på hvor stor/mye feil det skal være i A2.

Three-way handshake:

1. Klienten er først i en CLOSED state.
2. klienten sender et spesielt TCP-segment til serveren (via A2), som ikke inneholder applikasjonslag-data, men kun SYN flagget = 1. Dette gjøres ved å bruke metoden `constructInternalPacket(Flag flag)` i `AbstractConnection` i `connect()`-metoden i `ConnectionImpl`. Klienten havner så en i SYN_SENT state.
3. Serveren mottar SYN fra klienten. Når serveren mottar SYN, sender `accept()` en SYNACK tilbake til klienten.

4. Klienten venter på svar fra serveren. Vi har tre forskjellige tilfeller
 - 4.1 ingen svar eller korrupt svar: send nytt segment (steg 1)
 - 4.2 godkjent svar (connection granted). Vi sender deretter et segment tilbake som annerkjenner den mottatte SYNACK-en fra serveren. Klienten er nå i ESTABLISHED state.
- (5. Klienten kan sende og motta segmenter med pauload.)

Closing a connection:

1. Når klienten bestemmer seg for å avslutte forbindelsen til serveren sender klienten en FIN = 1 til serveren og går inn i en FIN_WAIT_1 state.
2. Mens klienten er i FIN_WAIT_1 state, venter den på annerkjennelse fra serveren. Serveren sender en FINACK tilbake.
3. Når klienten mottar FINACK, settes tilstanden til klienten seg i FIN_WAIT_2 state. Mens klienten er i denne tilstanden venter den på en FIN = 1 fra serveren.
4. Når klienten mottar FIN = 1 fra serveren går klienten inn i TIME_WAIT state og sender en ACK tilbake til serveren.
5. Mens klienten er i TIME_WAIT state, sender klienten en ny ACK i tilfelle den blir borte.
6. Når TIME_WAIT state er over (typisk 30 sec, 1 min eller 2 min) blir forbindelsen brutt.

Receive:

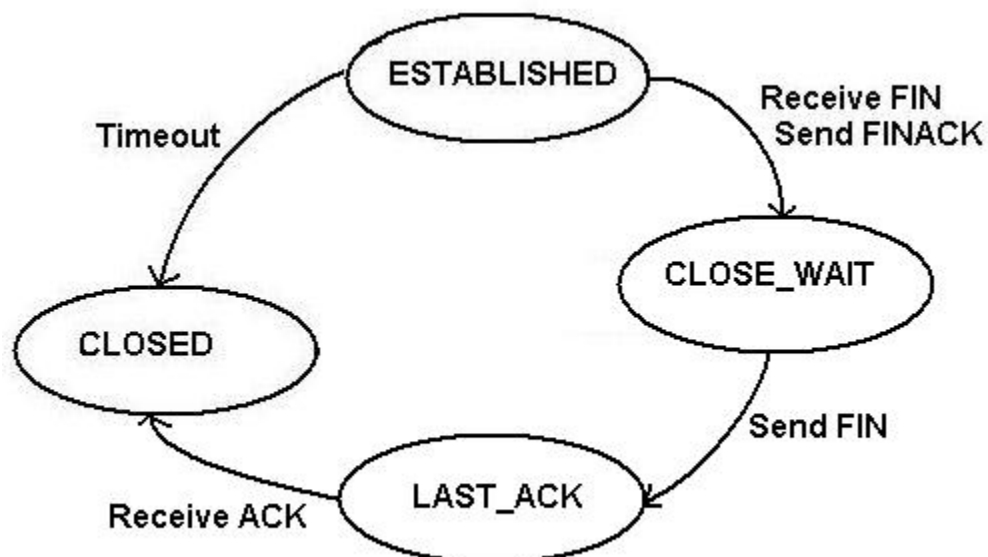
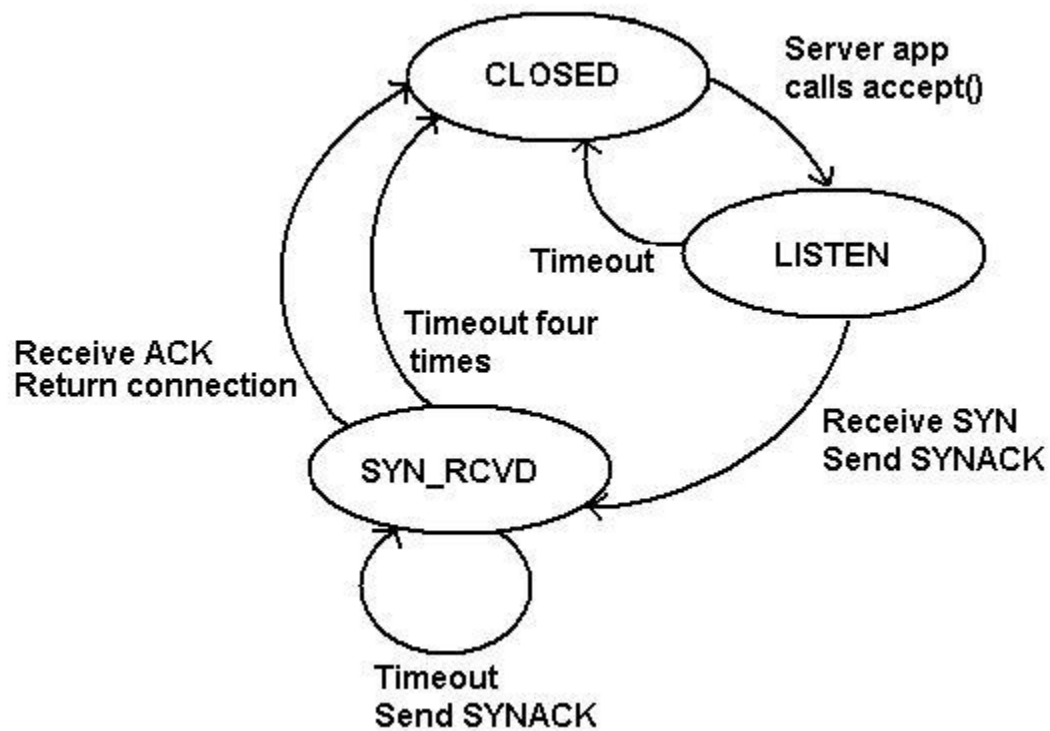
1. En pakke hentes fra A2 ved hjelp av `AbstractConnection.receivePacket(false)`
2. Hvis pakken har riktig sekvensnummer sendes det en ack med `AbstractConnection.sendAck()`, hvis ikke, gå til 1.
3. Pakken sjekkes for feil med `isValid()`.
4. Strengen i pakken returneres til applikasjonen.

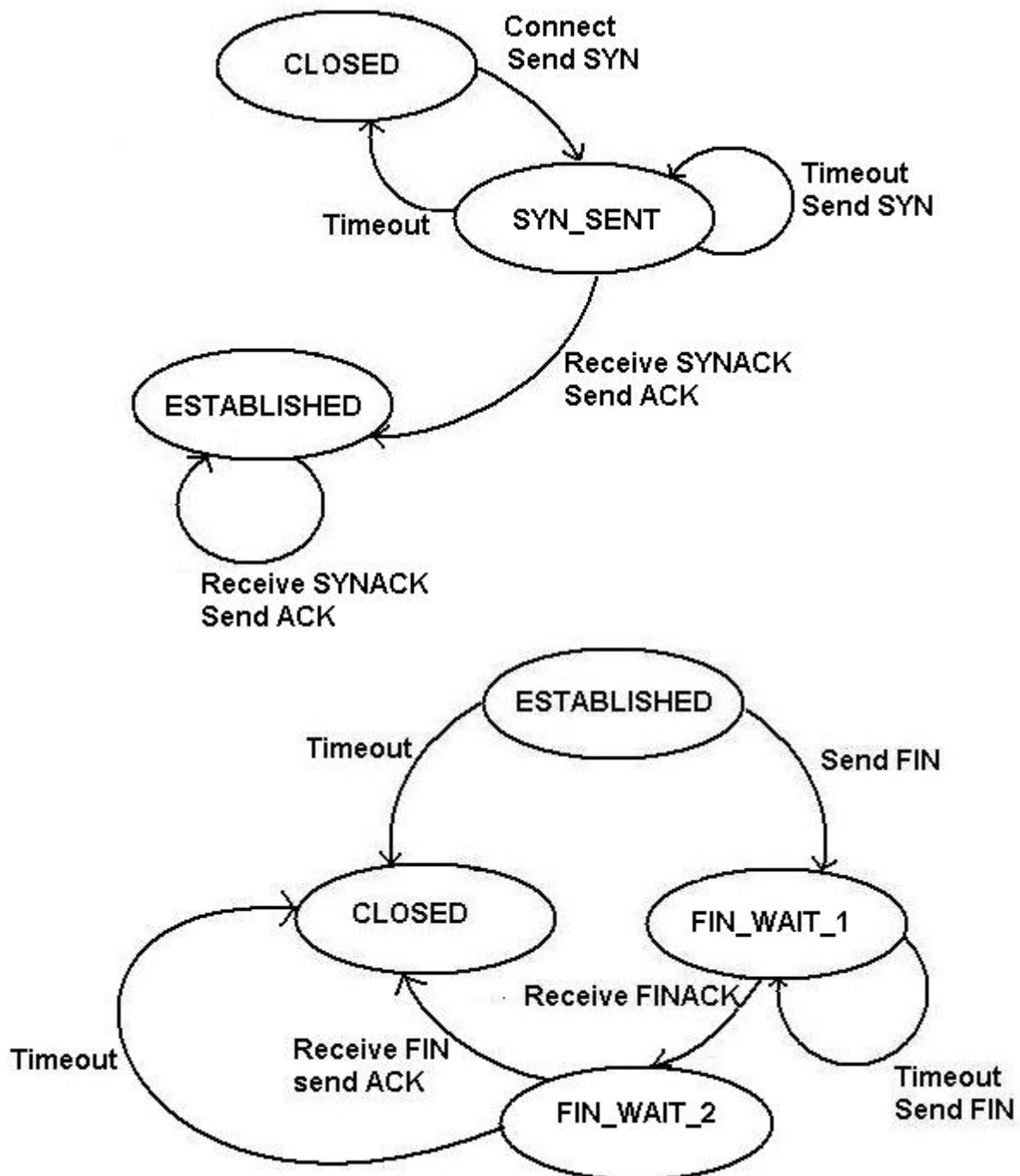
Send:

1. Datastrengen pakkes inn i et `ktnDatagram` ved hjelp av `AbstractConnection.constructDataPacket()`
2. Pakken sendes ved hjelp av `AbstractConnection.sendDataPacketWithRetransmit()`
3. Acken sjekkes for sekvensnummer, hvis det er feil, gå til 2

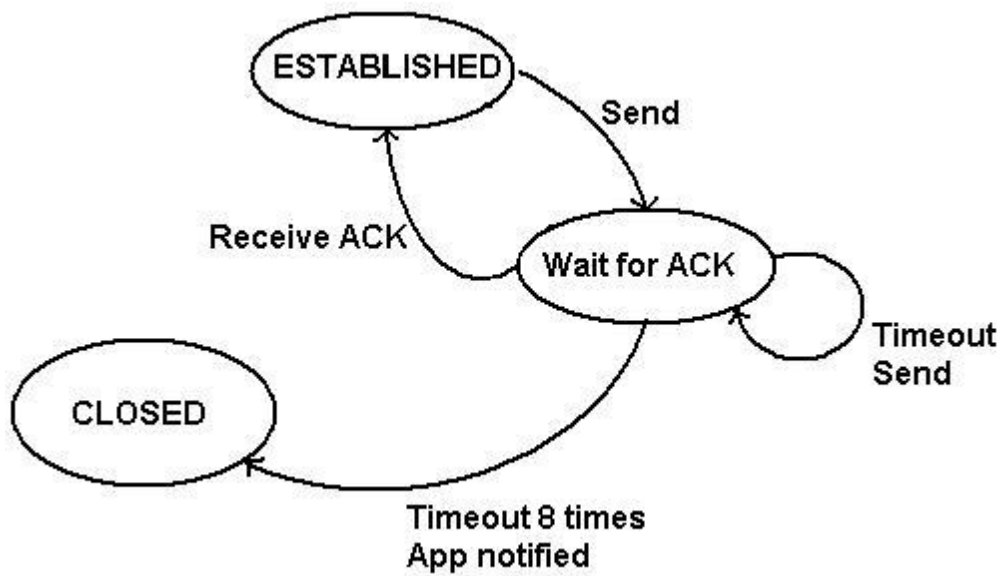
Tilstandsdiagrammer for A1

Tjener:

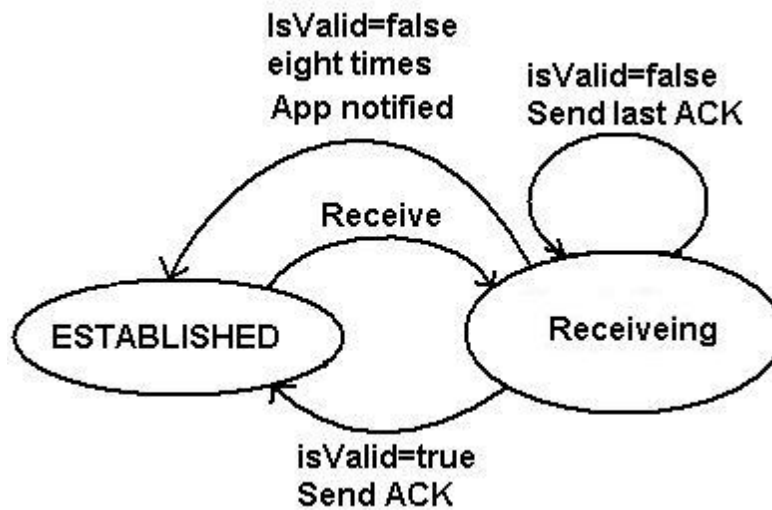


Klient:

Send:



Motta:



Feilhåndtering

Feil	Håndtering
Tapt pakke	Vi bruker positive/negative acknowledgments og en timer slik at sender sender pakken til mottaker en gang til. Etter x antall forsøk regnes forbindelsen som brutt.
Forsinket pakke	Vi bruker sekvensnummer slik mottaker vet det er et duplikat. Mottaker avslår dermed pakken, men sender ACK til sender.
Pakken har feil	isValid() sjekker alle pakker ved hjelp av checksum og mottaker sender dermed en NACK til sender.
Spøkelsepakke	isValid() sjekker alle pakker ved hjelp av checksum og eliminerer dermed spøkelsepakker.

Testplan

Id	Formål	Tilstand	Input	Respons
001	Klient ber om en tilkobling	CLOSED	Sender SYN	Mottar SYNACK, setter tilstand til SYN_SENT
002	Respons på SYNACK	SYN_SENT	Mottar SYNACK	Sender ACK, setter tilstand til ESTABLISHED
003	Tjener oppretter en tilkobling	LISTEN	Mottar SYN	Sender SYNACK, setter tilstand til SYN_RCVD
004	Tjener vidrekobler	SYN_RCVD	Mottar ACK	Ny tilkobling opprettet, setter tilstand til CLOSED
005	Klient ber om avslutning av tilkobling	ESTABLISHED	Sender FIN	Setter tilstand til FIN_WAIT
006	Klient avslutter tilkobling	FIN_WAIT	Mottar FINACK	Setter tilstand til CLOSED
007	Tjener avslutter tilkobling	ESTABLISHED	Mottar FIN	Sender FINACK, setter tilstand til CLOSED
008	Sending av pakke	ESTABLISHED	Pakke	Mottar ACK, eller forrige ACK hvis pakken er korrupt
009	Mottak av pakke	ESTABLISHED	Mottar pakke	1. Hvis isValid er false er pakken korrupt. Sender dermed forrige ACK. 2. Hvis isValid er true sender ACK

Sannsynlighet

TestID	Tapt pakke	Forsinket pakke	Feil i pakke	Spøkelsespakke
	Sannsynlighet	Sannsynlighet	Sannsynlighet	Sannsynlighet
1	0	0	0	0
2	0.1	0	0	0
3	0.5	0	0	0
4	0	0.1	0	0
5	0	0.5	0	0
6	0	0	0.1	0
7	0	0	0.5	0
8	0	0	0	0.1
9	0	0	0	0.5

Kombinerte feil

Om nettverket er tregt vil en få flere tapte og forsinkede pakker, vi tester dette scenarioet i test 1.

Vi ønsker å teste isValid()-funksjonen grundig, dette tester vi i test 2 og 3.

Test 4, 5 og 6 er stresstester. I test 6 skal applikasjonen gi feilmelding om at nettverkstilkoblingen er for ustabil.

TestID	Tapt pakke	Forsinket pakke	Feil i pakke	Spøkelsepakke
	Sannsynlighet	Sannsynlighet	Sannsynlighet	Sannsynlighet
1	0.5	0.5	0	0
2	0	0	0.5	0.5
3	0	0	0.9	0.9
4	0.4	0.3	0.3	0.3
5	0.4	0.6	0.6	0.6
6	0.1	0.9	0.9	0.45

Gruppe 45

Kim, John, Johan, Stian, Andreas