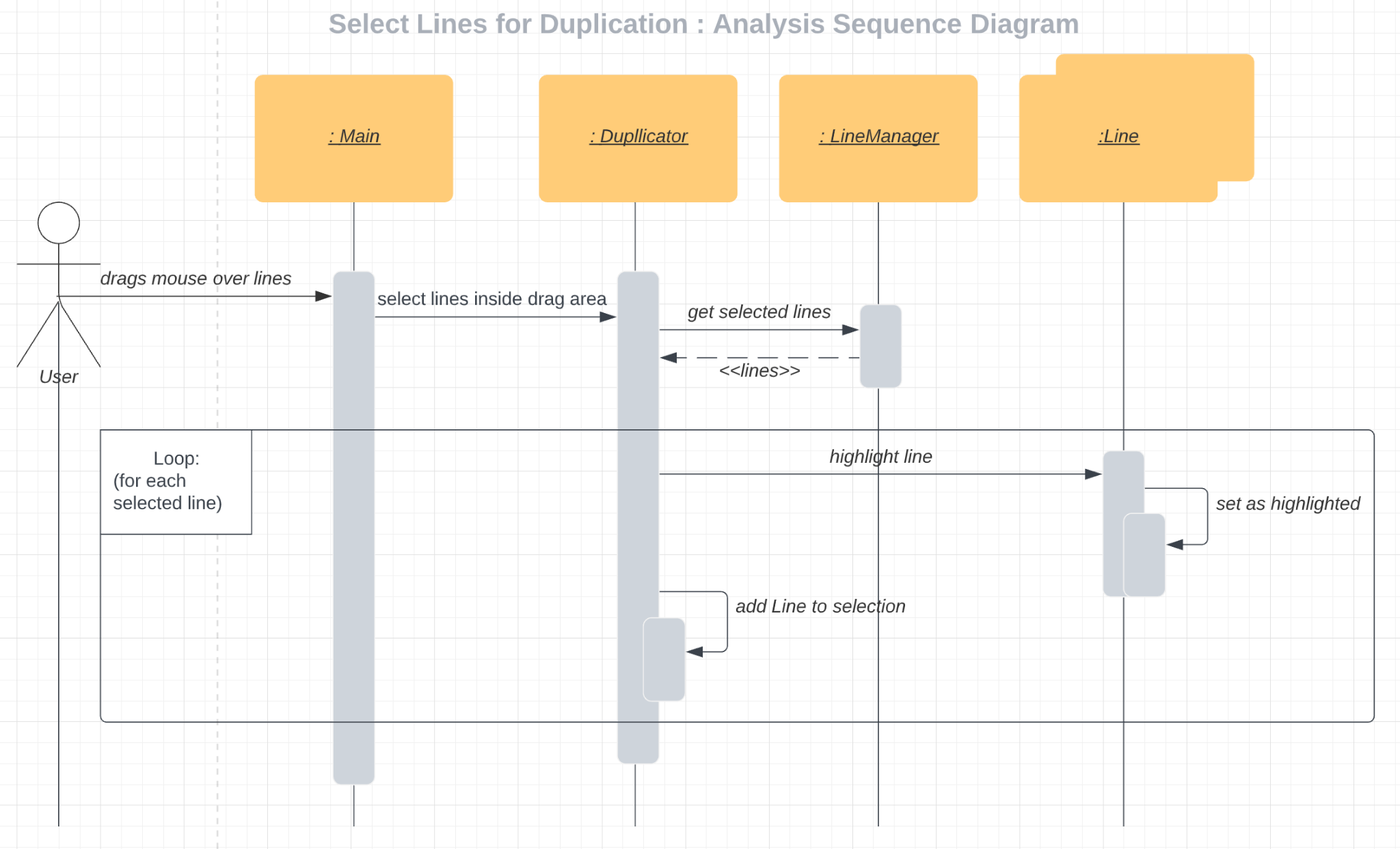
Analysis Sequence Diagrams

Group 5

Deliverable 4

**1) DUPLICATE LINES -> Select Lines for Duplication**

Analysis sequence diagram:



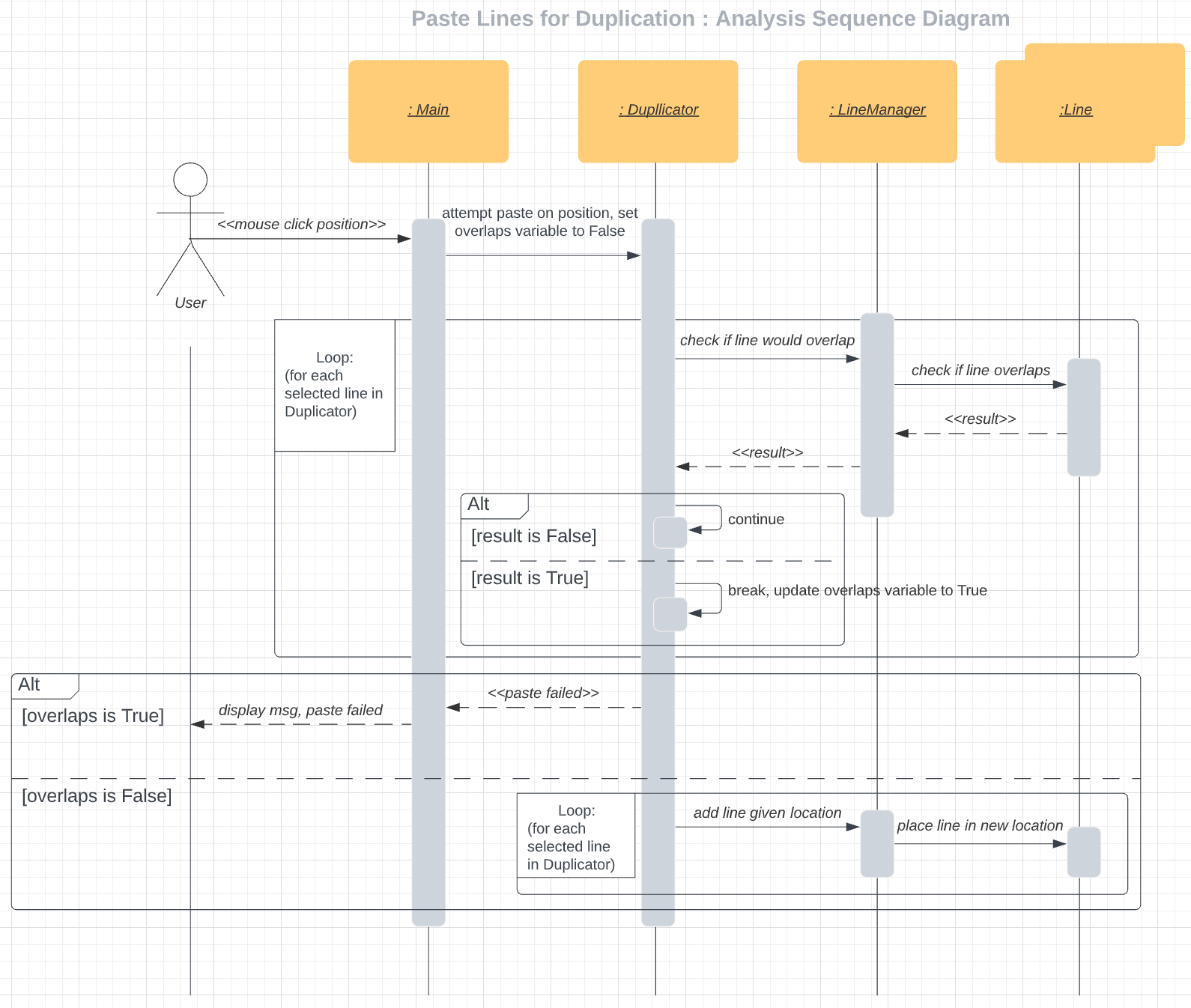
Note: To address feedback, Duplicator is acting as a controller for the duplication function.

Scenario description:

| # | Subject | Subject Action | Parameters | Object Acted Upon |
| --- | --- | --- | --- | --- |
| 1 | User | drags | mouse | Main |
| 2 | Main | selects lines | drag area | Duplicator |
| 3 | Duplicator | gets selected lines | drag area | LineManager |
| 4 | LineManager | returns | selected lines | Duplicator |
| 5 | Loop, for each selected line: | | | |
| 5.1 | Duplicator | sends | highlight | Line |
| 5.2 | Line | sets | highlighted | Line |
| 5.3 | Duplicator | adds to selection | Line | Duplicator |

**2) DUPLICATE LINES -> Place duplicated selection**

Analysis sequence diagram:

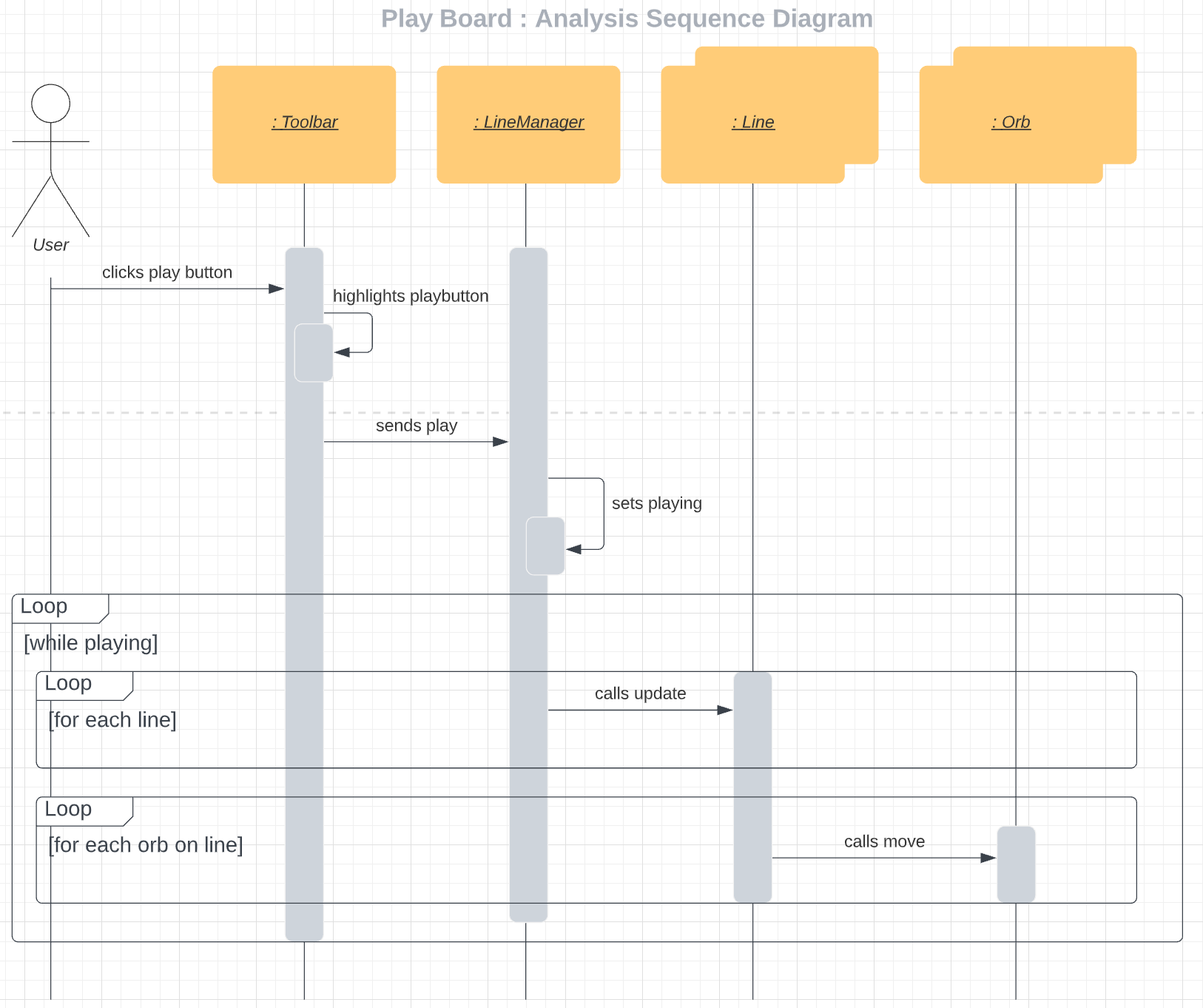


Scenario description:

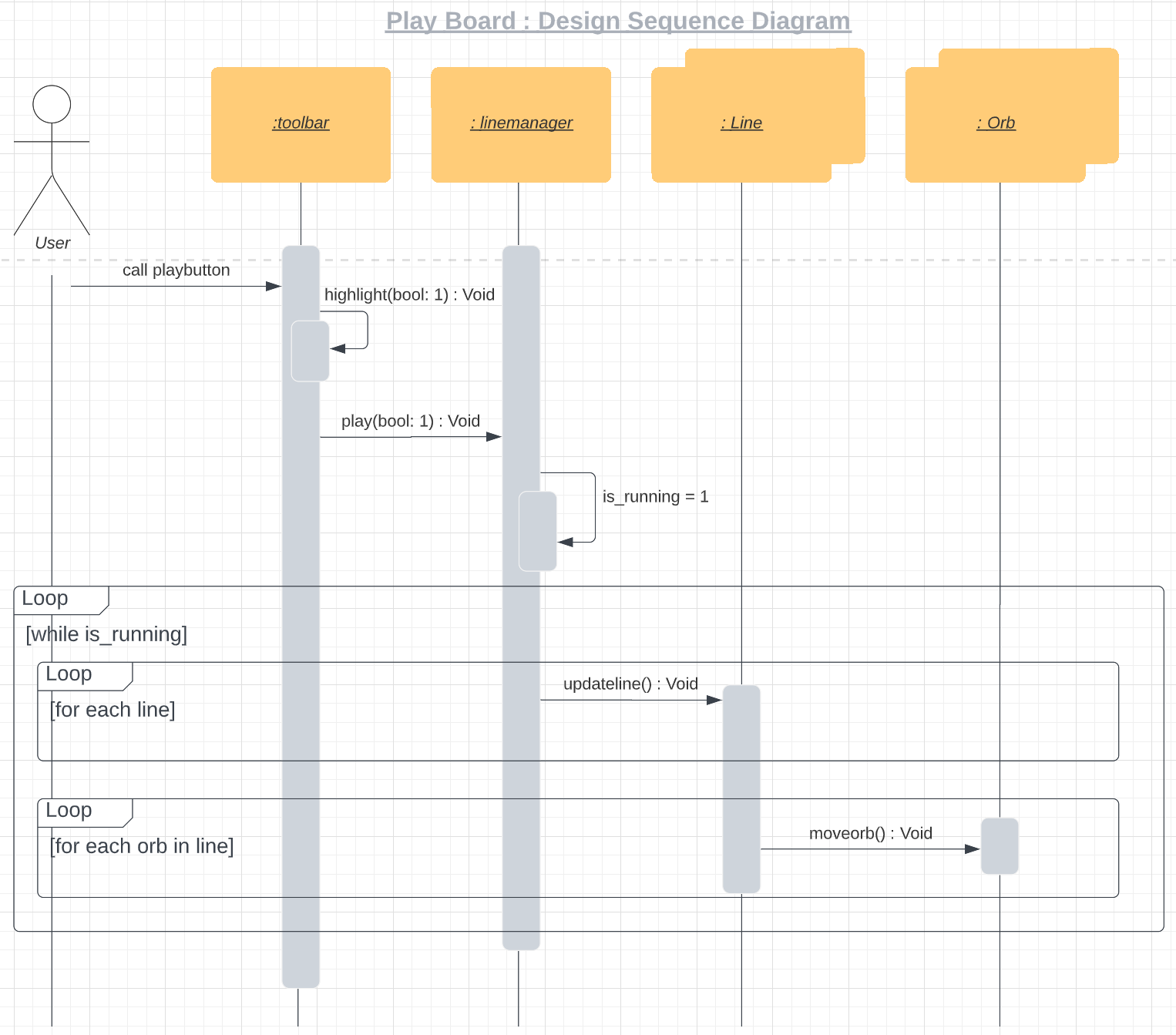
| # | Subject | Subject Action | Parameters | Object Acted Upon |
| --- | --- | --- | --- | --- |
| 1 | User | clicks mouse | position | Main |
| 2 | Main | sends | Paste, overlaps = False | Duplicator |
| 3 | Loop, for each selected Line: | | | |
| 3.1 | Duplicator | requests | overlaps | LineManager |
| 3.2 | LineManager | requests | overlaps | Line |
| 3.3 | Line | returns | overlaps | LineManager |
| 3.4 | LineManager | returns | overlaps | Duplicator |
| 3.3 | If line doesn’t overlap: | | | |
| 3.3.1 | Duplicator | continues |  | Duplicator |
| 3.4 | Else if line overlaps: | | | |
| 3.4.1 | Duplicator | breaks, updates overlaps to True |  | Duplicator |
| 4 | If overlaps is True: | | | |
| 4.1 | Duplicator | sends | "paste failed" | Main |
| 4.2 | Main | displays msg | "paste failed" | User |
| 5 | If overlaps is False: | | | |
| 5.1 | Loop, for each selected Line: | | | |
| 5.2 | Duplicator | sends | line location | LineManager |
| 5.3 | LineManager | places on location | new line | LineManager |

3) **PLAY BOARD -> Play Board (move orbs)**

Analysis sequence diagram:



Design Sequence Diagram:

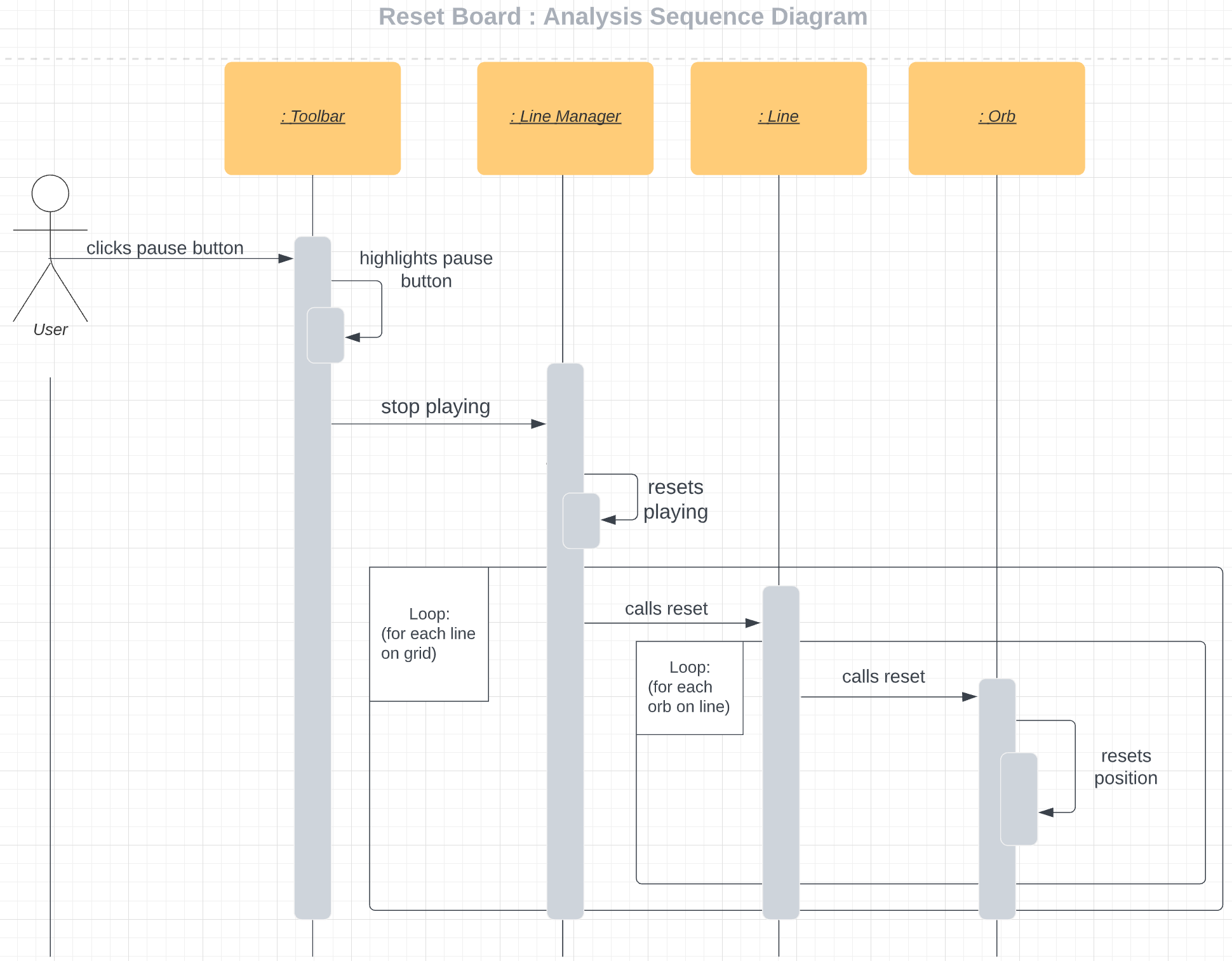


Scenario description:

| # | Subject | Subject Action | Parameters | Object Acted Upon |
| --- | --- | --- | --- | --- |
| 1 | User | clicks | play button | Toolbar |
| 2 | Toolbar | highlights | play button | Toolbar |
| 3 | Toolbar | sends | play | LineManager |
| 4 | LineManager | sets | playing | LineManager |
| 5 | While playing: |  |  |  |
| 5.1 | For each line: | | | |
| 5.1.1 | LineManager | calls | update | Line |
| 5.2 | For each orb on line: | | | |
| 5.2.1 | Line | calls | move | Orb |

**4) PLAY BOARD -> Reset Board and Stop Animation**

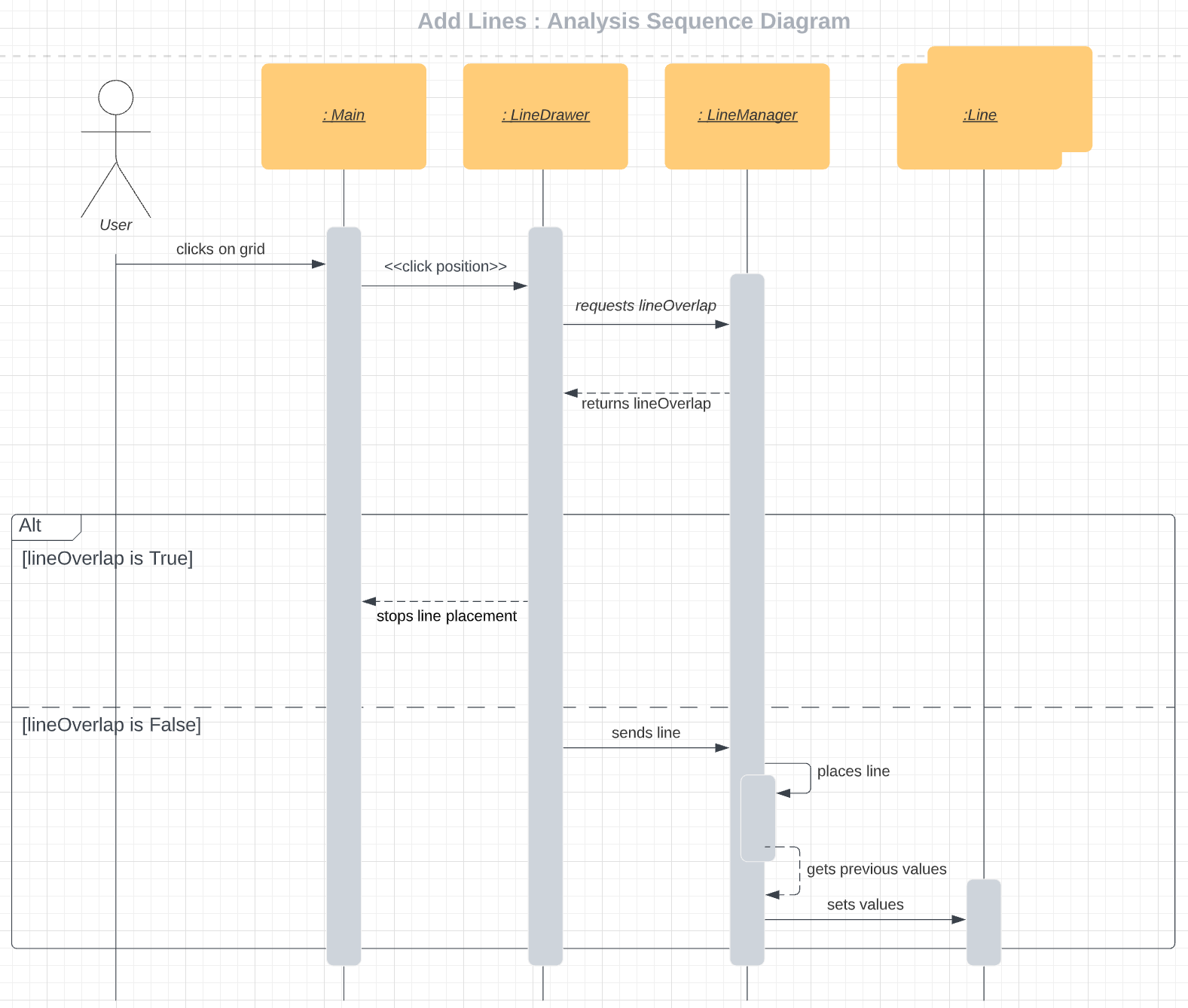
Analysis sequence diagram:



Scenario description:

| # | Subject | Subject Action | Parameters | Object Acted Upon |
| --- | --- | --- | --- | --- |
| 1 | User | clicks | pause button | Toolbar |
| 2 | Toolbar | highlights | pause button | Toolbar |
| 3 | Toolbar | sends | stop playing | LineManager |
| 4 | LineManager | resets | playing | LineManager |
| 5 | For each line on grid: | | | |
| 5.1 | LineManager | calls | reset | Line |
| 5.2 | For each orb on line: |  |  |  |
| 5.2.1 | Line | calls | reset | Orb |
| 5.2.2 | Orb | resets | position | Orb |

**5) ADD LINES**

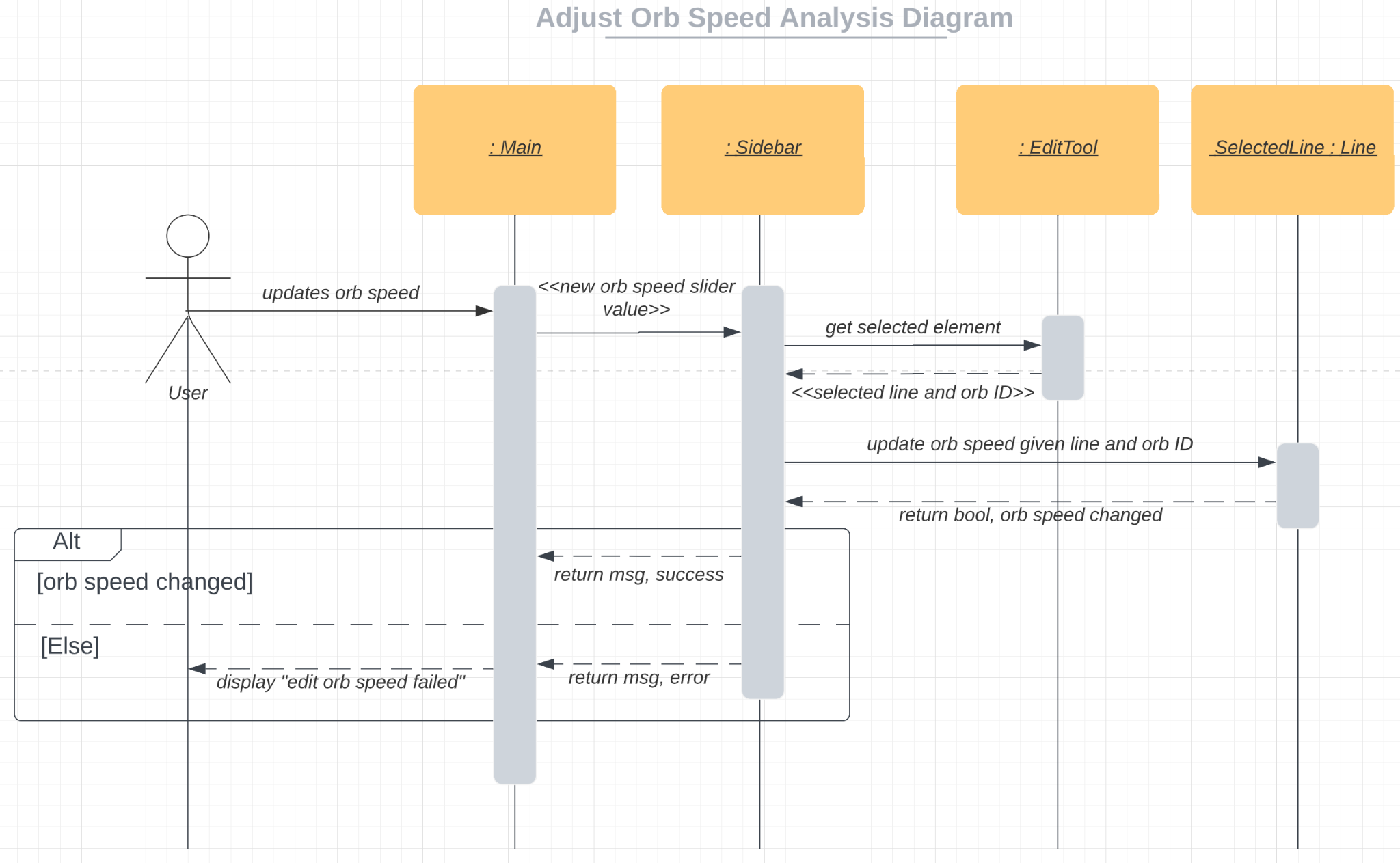


Scenario description:

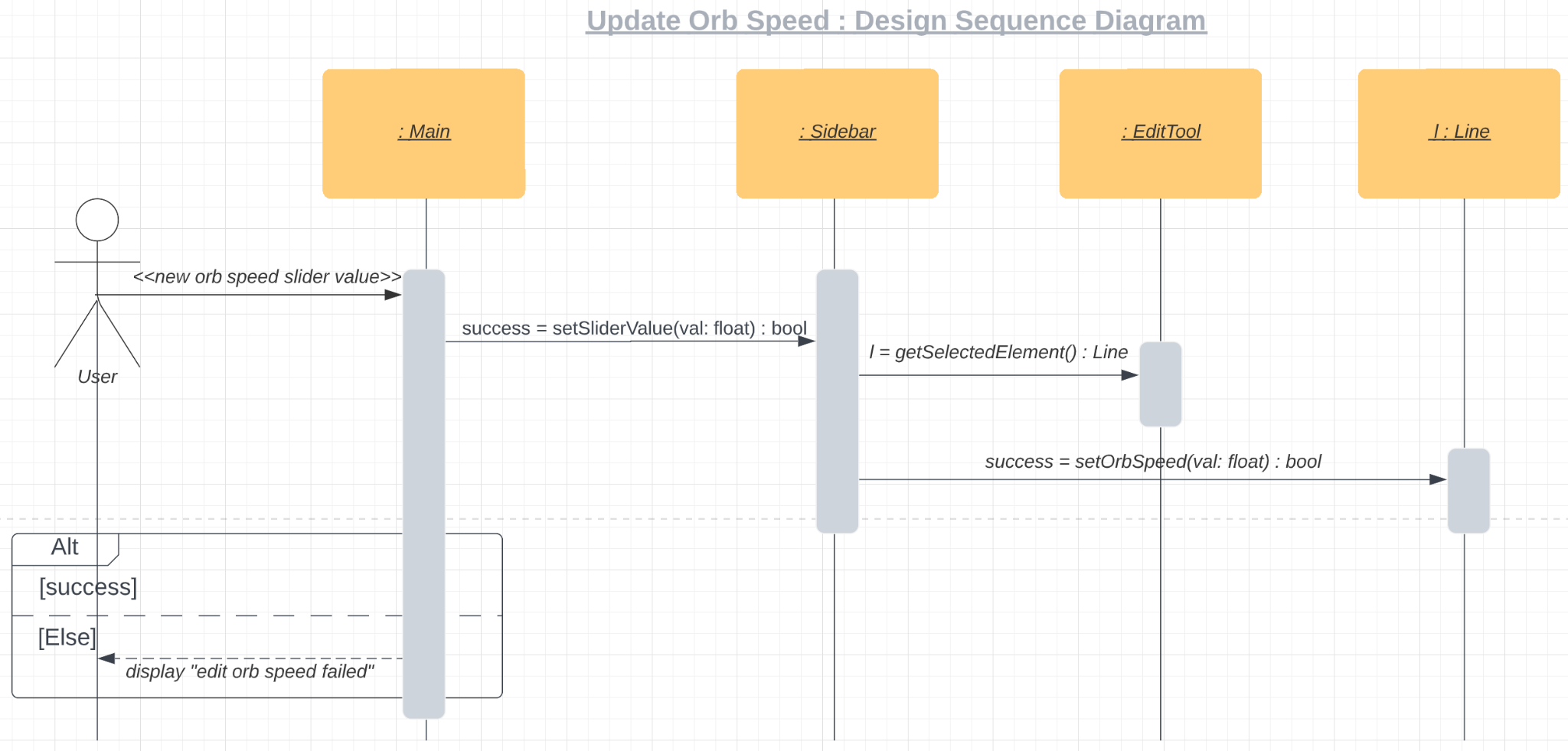
| # | Subject | Subject Action | Parameters | Object Acted Upon |
| --- | --- | --- | --- | --- |
| 1 | User | clicks grid | position | Main |
| 2 | Main | sends | click position | LineDrawer |
| 3 | LineDrawer | requests | lineOverlaps | LineManager |
| 4 | LineManager | returns | lineOverlaps | LineDrawer |
| 5 | If lineOverlaps: | | | |
| 4.1 | LineDrawer | stops | placement | Main |
| 5 | Else if lineOverlap is False: | | | |
| 5.1 | LineDrawer | sends | Line | LineManager |
| 5.2 | LineManager | places | Line | LineManager |
| 5.3 | LineManager | gets | previous values | LineManager |
| 5.4 | LineManager | sets | values | Line |

**6) Adjust Orb Speed**

Analysis sequence diagram:



Design Sequence Diagram:

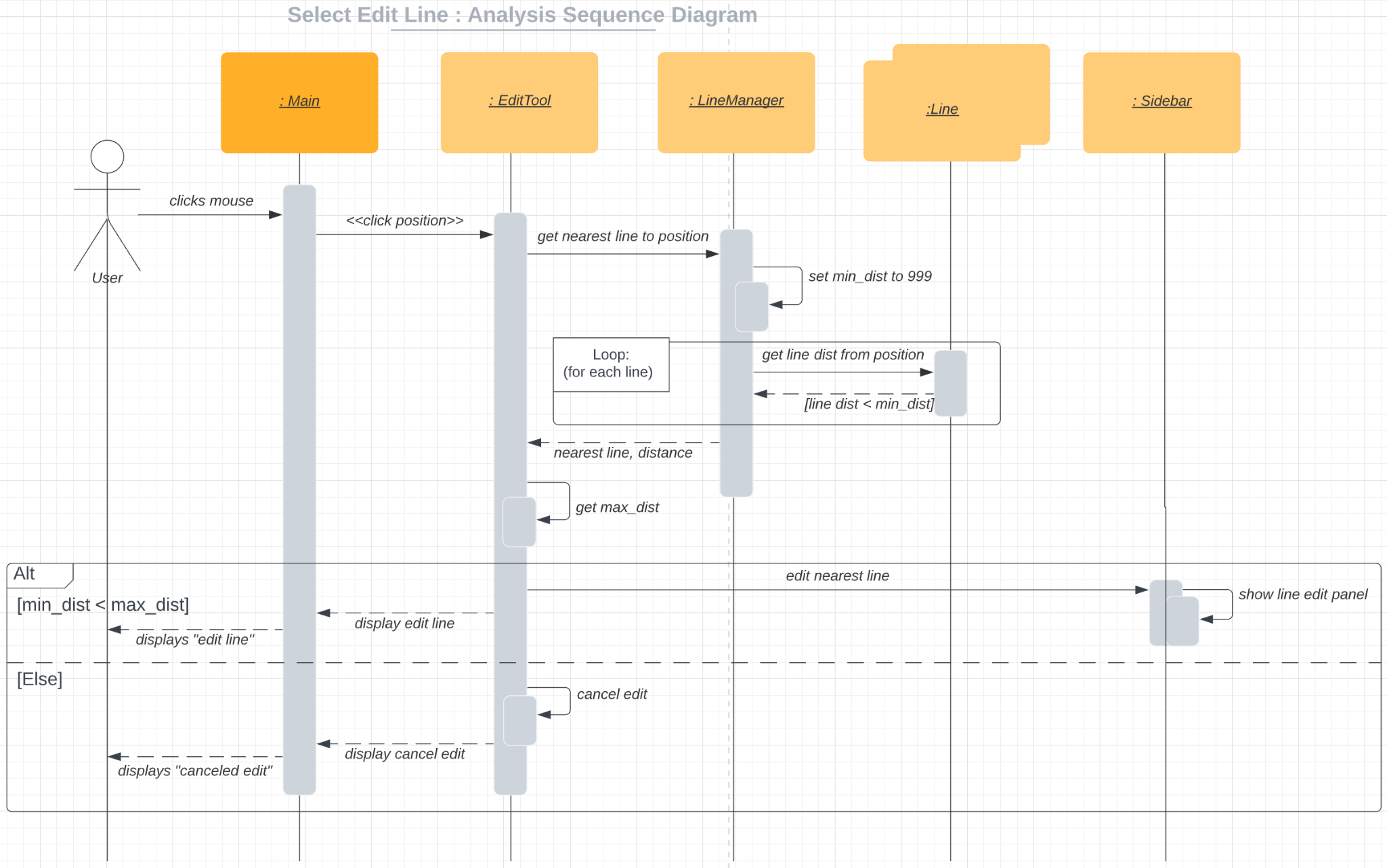


Scenario description:

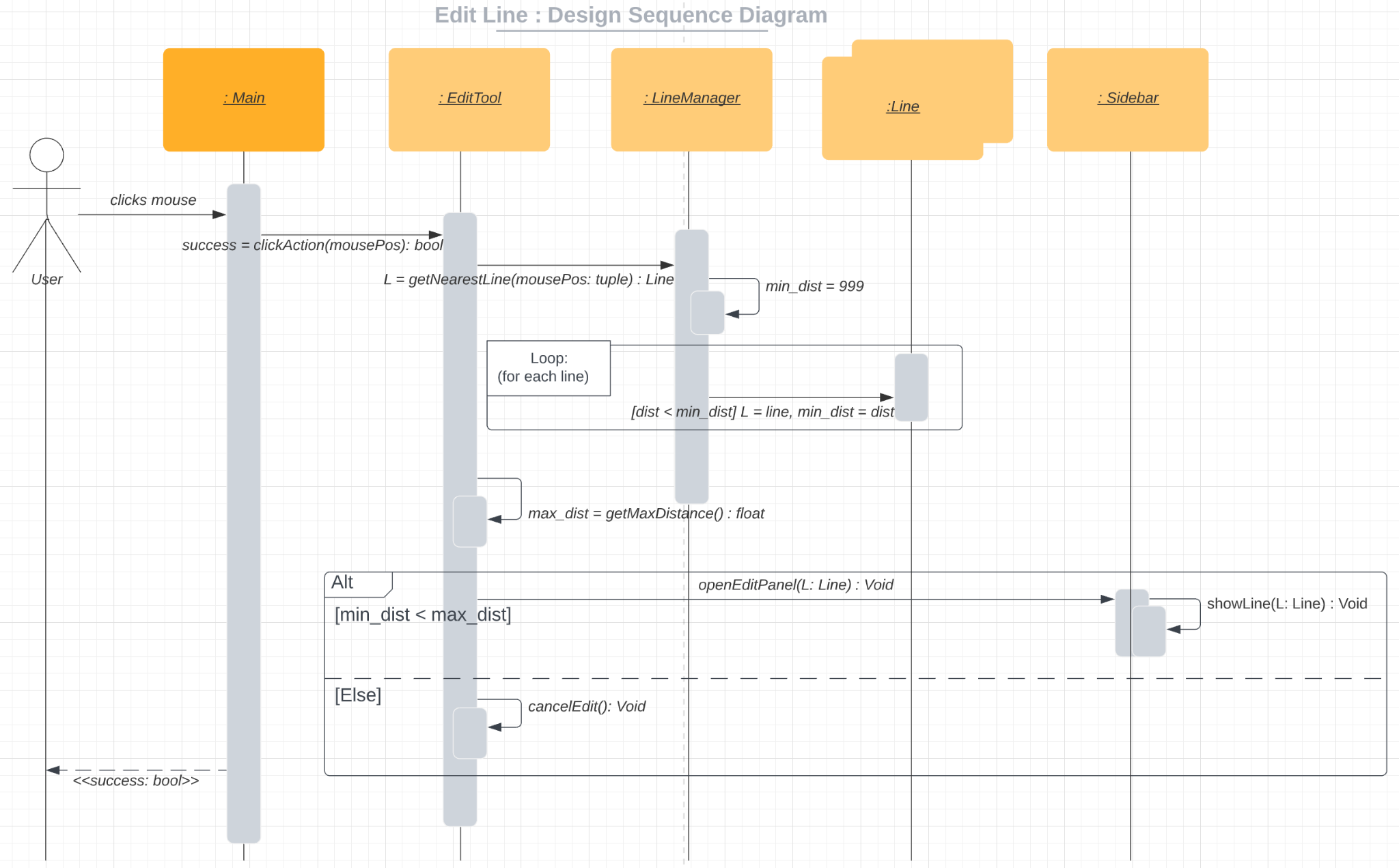
| # | Subject | Subject Action | Parameters | Object Acted Upon |
| --- | --- | --- | --- | --- |
| 1 | User | updates | orb speed | Main |
| 2 | Main | sends | orb speed slider value | Sidebar |
| 3 | Sidebar | gets | selected element | EditTool |
| 4 | EditTool | returns | selected element (Line and orb ID) | Sidebar |
| 5 | Sidebar | updates orb speed | Line, orb ID | Line |
| 6 | Line | returns bool | orb speed changed | Sidebar |
| 7 | If orb speed changed: | | | |
| 7.1 | Sidebar | returns msg | success | Main |
| 8 | Else: | | | |
| 8.1 | Sidebar | returns msg | error | Main |
| 8.2 | Main | displays msg | “edit orb speed failed” | User |

**7) SELECT EDIT LINE**

Analysis sequence diagram:

****

Design Sequence Diagram:



Scenario description:

| # | Subject | Subject Action | Parameters | Object Acted Upon |
| --- | --- | --- | --- | --- |
| 1 | User | clicks mouse | position | Main |
| 2 | Main | sends click | position | EditTool |
| 3 | EditTool | gets nearest Line | position | LineManager |
| 4 | LineManager | sets | min\_distance to 999 | LineManager |
| 5 | For each line in LineManager: | | | |
| 5.1 | LineManager | gets line position |  | Line |
| 5.1.1 | If line distance < min\_distance: | | | |
| 5.1.2 | Line | returns | min\_dist, line | LineManager |
| 5.2.2 | Else: (nothing happens, not included in diagram) | | | |
| 6 | LineManager | returns | nearest line, distance | EditTool |
| 7 | EditTool | gets | max\_distance | EditTool |
| 8 | If min\_dist < max\_dist: | | | |
| 8.1 | EditTool | edits | nearest line | Sidebar |
| 8.2 | Sidebar | shows | Line edit | Sidebar |
| 8.3 | EditTool | sends | edit line msg | Sidebar |
| 8.4 | EditTool | display | edit line | Main |
| 8.5 | Main | displays msg | “edit line” | User |
| 9 | Else: | | | |
| 9.1 | EditTool | cancels edit |  | EditTool |
| 9.2 | EditTool | display | cancel edit | Main |
| 9.3 | Main | displays msg | “canceled edit” | User |