|  |  |
| --- | --- |
| **Project Manager’s** | |
| Name and Surname | Safi CENGİZ |
| Student Number | 2030150004 |

|  |  |
| --- | --- |
| **Project’s** | |
| Title | Shortest path Dijkstra algorithm |
| Description  *(definition of problem, aim, algorithm,etc.)* | Made small graph for Shortest path Dijkstra algorithm and gave solution for Shortest path as possible. Mark all nodes unvisited. Create a set of all the unvisited nodes called the unvisited set.  Assign to every node a tentative distance value: set it to zero for our initial node and to infinity for all other nodes. Since initially no path is known to any other vertex than the source itself (which is a path of length zero), all other tentative distances are initially set to infinity. Compare the newly calculated tentative distance to the current assigned value and assign the smaller one.Then show all edges and vertexes as dictionary and shortest distance between them. |
| Main tool for development\* | Python  Libraries: matplot,networkx,sys,csv,image PIL |
| Additional tools to be used  *(if any)* | None |
| Milestones  *(main steps)* | For first review; I created manual list of nodes and vertices and showed shortest path used drawned picture(for explain better) for it no graph at all it was small graph.  For 2nd review; I changed manual list to matrix which you showed us in lectures make sketch for more nodes.  For final review; I made graph with newtorkx lib and added more nodes. |
| Methodology | Dijkstra algorithm assign to every node a tentative distance value then visit all possible ways for destination node. Then gives us shortest way possible. If the edges not shortest it just pass it. |
| \*Attach the code of project https://colab.research.google.com/drive/1KSX1-TlCPRHB9waSHHfU7W4bGMUahkt6#scrollTo=1RFA-AiN4\_kC | |