User Story

Not Done Yet Almost Done Finished

CS1D – Project 2

Team: Austin Merando, John Zavala, Nhan Phan, Sina Heydarimolaei, Kayvon Haghighi

1. As a football fan, I want to have access to all the stadiums and their corresponding information: the teams, stadiums’ seating capacities and their surfaces
   1. Description:

* Create a basic GUI in order to display all the stadiums and the information associated with the stadium.
* Football fan will also be able to view the additional list in the GUI:
  + A lists of NFL teams sorted by team name
  + A list of NFL stadiums and their corresponding team name sorted by team name.
  + A list of American Football Conference teams sorted by team name.
  + Create a list of National Football Conference teams sorted by team name.
  + A list of stadiums that have an “open” stadium roof type and their corresponding team name sorted by stadium name.
  + A list of NFL star players and their corresponding team name sorted by team name.
  + A list of NFL teams, their stadium names, their seating capacity, their corresponding location sorted by seating capacity.
  + A list of NFL teams, their stadium names, their surface type, and their corresponding location sorted by surface type
* Determine what underlying data structure for a map (except STL or Qt Container) will be used to store those information: Linked list
  1. Assumption:
     + All of the teams’ information has been stored in the program
  2. Tasks:
     + Create the following:
       - A Stadium class which hold the all information about the stadium
       - A linked list with map ADT (begin, end, size, empty, put, remove)
  3. Test:
     + Black box texting for this story
       - Checking if each list is correctly sorted.
       - Visual check on the GUI: Check if things are output in GUI
  4. Priority:8
  5. Story point: 6
  6. Definition of Done:
     + A map
     + The lists created from the description has to be correctly sort according to the criteria from each list
     + Those lists can be seen on GUI
  7. Assignee(s): Austin Merando, John Zavala, Nhan Phan, Sina Heydarimolaei, Kayvon Haghighi

1. As a football fan, I want to choose a starting stadium, in which I clearly know the number of stadiums closest to Los Angeles Memorial Coliseum, so that later on I can travel on a trip. This trip uses a shortest path between each stadium and it starts from Los Angeles Memorial Coliseum. That way, I can take the most efficient route.
   1. Description:
      * Football fan will start the trip from Los Angeles Memorial Coliseum
      * An option to travel to the number of stadiums will be shown to the fan and he/she must be able to decide the number to travel. This also means that fan can visit all of the stadiums.
      * Dijkstra’s algorithm will be using to find the shortest distance.
      * With the selected stadiums, the fan will see the overall shortest route which starts from Los Angeles Memorial Coliseum
   2. Assumption:
      * User Story 1 has been finished by the assignee(s)
      * The assignee(s) for this Story already know Dijkstra’s algorithm
      * GUI are made with the lists of stadiums
   3. Tasks:
      * Implement Dijkstra’s algorithm to find the shortest distance
   4. Test: Black box testing will be used to see if the shortest path is correctly made
   5. Priority: 2
   6. Story point: 4
   7. Definition of Done: During the trip starting at Los Angeles Memorial Coliseum, football fan will be able to check the distance between ever stadium visited and the total distance
   8. Assignee(s): Nhan Phan, Sina Heydarimolaei, John Zavala
2. As a football fan, I would like to choose any restaurant that I want and make a trip with that stadium as a starting location. This trip will include the shortest paths between one stadium and another (also between one stadium and Los Angeles Memorial Coliseum). That is because I want to take the most efficient route.
   1. Description:
      * Football fan will start the trip from a stadium selected by that fan
      * With the list of stadium, football fan will be able to choose the stadiums that he/she wants to travel to.
      * Dijkstra’s algorithm will be using to find the shortest distance.
      * With the selected stadiums, the fan see the overall shortest route which starts from Los Angeles Memorial Coliseum
   2. Assumption:
      * User Story 1 has been finished by the assignee(s)
      * The assignee(s) for this Story already know Dijkstra’s algorithm
      * GUI are made with the lists of stadiums
   3. Tasks:
      * Implement Dijkstra’s algorithm to find the shortest distance
   4. Test: Black box testing will be used to see if the shortest path is correctly made
   5. Priority:2
   6. Story point: 4
   7. Definition of Done: During the trip starting at any stadium, football fan will be able to check the distance between ever stadium visited and the total distance
      * Trip starts any restaurant that is based on the fan’s choice
   8. Assignee(s): Nhan Phan, Sina Heydarimolaei, John Zavala
3. As a football fan, I want to know the mileage, which associated to the minimum spanning tree connecting all of the NFL stadiums.
   1. Description:
      * The path between one stadium and one of its adjacent stadiums must be the shortest distance among all path from that stadium.
      * The mileage consists of those paths
   2. Assumption:
      * Each stadium already has the distance from that stadium to the adjacent stadiums
      * The assignee(s) for this Story already know Prim’s or Kruskal’s algorithm
   3. Tasks: TBA
   4. Test: Using black box testing to see if the mileage is calculated correctly
   5. Priority:2
   6. Story point: 4
   7. Definition of Done: The mileage obtained from the minimum spanning tree connected all stadiums must be correctly.
   8. Assignee(s): Nhan Phan, Sina Heydarimolaei, John Zavala
4. As a football fan, I want to track the souvenirs that I purchased on the trip along with the corresponding costs, and also a grand total for all the stadium.
   1. Description:
      * Football fan will be able to see for each trip ONLY
      * Football fan will be able to see the cost of the souvenirs that he/she purchase
      * Football fan will be able to know the total spent after the trip is finish, which include all of the souvenirs bought at the last stadium
   2. Assumption:
      * User story 1 is finished by the assignee(s)
      * Beside the complete information, each team also need its own list of souvenirs to football fan to buy
   3. Tasks: TBA
   4. Test: Black box testing checking the price from purchasing souvenirs in each stadium during the trip and also the total price
   5. Priority:3
   6. Definition of Done: The price when football fan purchases the souvenirs is shown in UI, including the total
      * Must be correctly calculated for the total price
   7. Assignee(s): John Zavala
5. As an administrator, I want to modify a stadium information of the team in which that team moves into that stadium.
   1. Description:
      * Administrator will be about to change the name of the team’s stadium
      * Change the stadium’s name must not adversely affect other information of the same team
   2. Assumption:
      * Administrator’s page is made
      * The information of each team is already given and available for administrator to modify
   3. Tasks:
      * Create class(es) to modify to access the stadium information and make modification.
      * Make the modification works in UI
   4. Test:
   5. Priority:
   6. Story point: 2
   7. Definition of Done: The information related to the stadium must be able to be changed upon football fan’s choice
6. Assignee(s): Nhan Phan, John Zavala, Kayvon Haghighi
7. As a football fan, I want to know the information related to my selected football team (team name, stadium name, seating capacity, location, conference, surface type, stadium roof type, star player).
   1. Description:
      * The system will output the team’s information: the name, its stadium’s name, seating capacity of its stadium, stadium location, conference, surface type, stadium roof type and its start player
      * The list will be shown in the GUI
   2. Assumption:
      * Each team’s information is given and ready to use
   3. Tasks:
      * Create a method in the stadium list class to access one of the stadium
   4. Test:
   5. Priority:2
   6. Story point: 2
   7. Definition of Done: Football fan can see the information in GUI
      * About the team name and it corresponding information
   8. Assignee(s): Kayvon Haghighi, Sina Heydarimolaei, Nhan Phan

Base user story details:

1. As a football fan, I want to know the total seating capacity of the NFL teams.
   1. Description:
      * Sum up the total seating capacity of all of the stadiums from every NFL team
      * Football fan will be able to see that total value
   2. Assumption:
      * Each team already has the its own seating capacity at the stadium
   3. Tasks: TBA
   4. Test: TBA
   5. Priority:2
   6. Story point: 1
   7. Definition of Done: The total seating capacity is calculated from all NFL teams
   8. Assignee(s): Nhan Phan
2. As an administrator, I want add a new team and the associated information (team name, stadium name, seating capacity, location, conference, surface type, stadium roof type, star player)
   1. Description:
      * The information of new team should be stored in the database with all of the inputs from the administrator
   2. Assumption:
      * Administrator is successfully logged on
      * Administrator’s page is made
      * The information of each team is available
   3. Tasks:
   4. Test:
   5. Priority:3
   6. Story point: 9
   7. Definition of Done: New team and the corresponding information is successfully added by the administrator.
   8. Assignee(s): John Zavala, Nhan Phan, Kayvon Haghighi
3. As an administrator, I want to be able to make some modifications on the souvenirs’ list such as adding, deleting or changing the item along with its price per NFL team (?)
   1. Description:
      * Administrator will be able to add, delete the souvenirs from the list of souvenirs from one team.
      * A souvenir consists a name and a price
      * Administrator will be able to change the price of any exist souvenir from the list, BUT not the name
   2. Assumption:
      * Administrator is successfully logged on
      * Administrator’s page is made
      * The information of each team is available
   3. Tasks:
   4. Test: Black box testing on checking if the item is modified successfully.
      * Check if the item is added, changed the way the administrator wants
      * Check if the deleted item is completely removed in the system
   5. Priority: 3
   6. Story point: 14
   7. Definition of Done: Administrator can add, delete, change souvenirs without any problem
   8. Assignee(s): John Zavala
4. As a team member, I want to have a more understand about this project by visualizing its models.
   1. Description: Create a UML
   2. Assumption: The assignee already has the knowledge about UML
   3. Task: Create the following diagrams
      * UML State Diagram
      * UML Class Diagram
      * UML Use Case Program
   4. Test: Visual check for the UML(s) meet the requirements
   5. Priority:1
   6. Story Point: 2
   7. Definition of Done: A complete UML of the project including all kinds of UML diagrams
      * If UML is made during the software development, UML must be always up-to-date to the development process
   8. Assignee(s): Sina Heydarimolaei, Nhan Phan
5. As team members, we want to have a documents for testing during the software development of our group.
   1. Description: Create a test plan
   2. Assumption: The assignee already has the knowledge about creating a test plan
   3. Task:
      * Decide the testing technique
      * Implement continuous integration testing
      * Create the following information for the test plan:
        + Description
        + What to be tested
        + Test technique
        + Test cases
        + Overall strategy
   4. Priority:1
   5. Story Point: 2
   6. Definition of Done: A fully documented Test Plan that covers as much test cases as possible
      * Most features of the software must be tests not just one single or a few
      * Test plan must include all of the test cases that were done by the members.
   7. Assignee(s): Nhan Phan

Sprint 1:

* Story #1, #8, #11, #12

Sprint 2:

* Story #6, #7, #9, #10

Sprint 3 (last):

* Story #2, #3. #4, #5

Team Rule:

* Language using: C++ and framework Qt
* Online meeting:
* Using Slack
* Every Thursday at 9:00pm to 9:15pm