

Project Name: European Vacation

Project Members: Theta Gang (Blake, Muzzy, Jared, Felix)

Test ID: 1

Purpose of Plan: Make sure all code runs effectively and smoothly with no errors/problems

Scope of the Test Plan: All code will be tested to get desired outputs

Overall Test Strategy: Regression Testing

- Break up code into smaller parts each time we run our regression test
- Have test code to check every possible path that could break the program until the desired result is found

Feature's tested from Traveler's Perspective:

- Login functionality for traveler/user
- List of Cities is properly displayed
- List of foods is properly displayed
- plan a trip starting from Berlin to the initial 11 European cities
 - Will plan and display the trip in the most efficient order
- plan the shortest trip starting from Paris
 - Number of cities wanting to visit is specified by traveler
 - Trip is planned starting in Paris and taking the shortest route between each city
- Plan a custom trip
 - Specify starting European city
 - Traveler will pick all other cities they wish to visit
- When planning any trip the order of cities will be in the most efficient order
 - Least distance between each city
- Traveler can purchase multiple food items when visiting each city
- With each travel plan total distance should be displayed

Features Tested from Admin's Perspective:

- Admin login functionality

- Add new cities
 - Distances to other cities
- Add new food items
 - Create price for food item
- Change prices for food items
- Delete food items

Entry Criteria:

- Database Complete
- Login page complete
- Displays for cities and food complete
- Algorithm for determining shortest distances complete

Exit Criteria:

- When everything works properly

Suspension Criteria:

- When a new feature is being added or an undesired result is found

Roles and Responsibilities:

- Product owner:
 - Talk to customer about what results need to achieve
 - Make sure team members are following what the customer desires
 - Make sure stories are being completed
 - Oversees product testing

- Update backlog everytime a test is run with desired result/outcome of the test
 - If test produces an undesirable result, log the result and instruct dev team/scrum master to fix
- Scrum Master:
 - Ensure smooth interactions between team members
 - Ensure goals and of the product owner are understood from team members
 - Help with/check to see if stories are complete
 - Report to product owner
- Team Member
 - Implement testing criteria to what the customer wants to see
 - Help with stories
 - Check in regularly with scrum master

Test Team:

- Product Owner: oversees tests and provides insight to what should be done
- Scrum Master: Helps implement tests, mainly gives guidance on the more important things being tested first
- Team Members: implement a majority of the testing based off details from scrum master/product owner

Approval Process:

- When the product owner is satisfied with the output and when they believe it fits within the customer's wants/needs

Test Strategy:

- Black Box Testing

Schedule:

- Every night, but if nothing is added during the day no need to run at night
- Make sure test when smoothly the day before, if not fix problems and rerun test
- Sprint 1 should test functionality of login page and admin utilities
 - Ensure both an admin and traveler can login
 - Admin should be able to edit the database
- Sprint 2 should test functionality of menus for displaying travel plans to traveler
 - List of Cities should be able to be displayed
 - List of food should be displayed
 - Traveler should be able to plan a trip(order does not matter at this point)
- Sprint 3 implement algorithm to determine the most efficient travel path
 - Traveler can start from Berlin, Paris, or a custom trip
 - After specifying all locations for the trip, it will be displayed in the most efficient order(least distance in between each city)

Training:

- Go over regression testing and black box testing. Make sure everyone is familiar and knows what each component does

Environment Description:

- QT, SQLITE, MACOS, Windows
- Software will be running off a windows platform for the final
- Team is split between macOS and windows so precautions must be taken into consideration when merging work from the two platforms

Configuration Management Plan:

- Each member has their own branch on top of a master branch. Each member will upload their required files from each sprint to their own branch and things needed by everyone such as database and homepage will be in the master branch. If a test fails we must fix the problem by debugging and making our code more sucure and efficient.

Test Deliverables:

- Black box testing
- Regression testing
- The execution report(pass/fail) and the errors needed to fix
- UML Diagrams
- Agile stories
- Coding standards
- Team rules
- Doxygen
- Test Plan
- Scrum log

Documents:

- <https://www.guru99.com/black-box-testing.html>
- <https://www.guru99.com/regression-testing.html>
- <https://ivc.instructure.com/courses/41588/assignments/791592>

Glossary of Terms:

- GIT: a distributed version-control system for tracking changes in source code during software development.
- QT: a free and open-source widget toolkit for creating graphical user interfaces as well as cross-platform applications that run on various software and hardware platforms such as Linux, Windows, macOS, Android or embedded systems with little or no change in the underlying codebase while still being a native application with native capabilities and speed.
- SQLITE: a relational database management system contained in a C library.
- Black box testings: a method of software testing that examines the functionality of an application without peering into its internal structures or workings.
- Regression Testing: re-running functional and non-functional tests to ensure that previously developed and tested software still performs after a change.

