Software Requirement Specification

1.Introduction:

1.1 Purpose:

The purpose of a virtual self-driving car is to simulate the behavior and performance of autonomous vehicles in a virtual environment. It can be used for research and development, education and training, testing and validation, and entertainment and gaming purposes. Testing self-driving algorithms and control systems in a virtual environment can help to identify and address issues before they become a problem on the road.

1.2 Scope:

The scope may include developing and testing different algorithms and control systems for self-driving cars, creating virtual environments and scenarios to simulate real-world driving conditions, integrating different sensors and perception systems, ensuring the safety and reliability of the system, providing a platform for education and training, and developing entertaining and engaging experiences for users.

1.3 Definitions, Acronyms, Abbreviations:

A virtual self-driving car refers to the development and testing of autonomous vehicle technology in a simulated environment, without the need for a physical vehicle. The project involves creating a software-based model of a car, which can navigate through a simulated world and interact with various obstacles, traffic rules, and other vehicles. The virtual self-driving car project typically uses advanced machine learning and artificial intelligence techniques to train the software to recognize and respond to different scenarios and challenges that may arise on the road.

Acronyms and abbreviations:

- AV: Autonomous vehicle
- VSDC: Virtual self-driving car

- Lidar: Light Detection and Ranging
- GPS: Global Positioning System
- AI: Artificial Intelligence
- ML: Machine Learning

1.4 References:

- https://carla.org/
- https://pythonprogramming.net/
- https://arxiv.org/abs/1604.07316
- https://www.udacity.com/course/self-driving-car-engineer-nanodegree--nd013
- https://hal.archives-ouvertes.fr/hal-01730991/document
- https://www.mdpi.com/2076-3417/10/19/6962

1.5 Overview:

- **Setting up a virtual environment:** Before beginning any coding, it's a good idea to set up a virtual environment to isolate the project's dependencies. This can be done using tools like virtualenv.

- **Building a simulation environment:** The first step in creating a virtual self-driving car is to build a simulation environment in which the car can navigate. This could involve using a pre-existing simulator like Carla.

- **Collecting sensor data:** Once the simulation environment is set up, the next step is to collect sensor data that the self-driving car will use to navigate. This could include data from lidar, GPS, cameras, and other sensors.

- **Developing machine learning models:** With the preprocessed sensor data in hand, the next step is to develop machine learning models that can use the data to make decisions about how the car should navigate. This could involve using Reinforcement Learning.

- **Implementing control algorithms:** Once the machine learning models are trained, the next step is to implement control algorithms that can translate the model's decisions into actions by the car.

- **Testing and validation:** Finally, the self-driving car system should be tested and validated in the simulation environment.

2. The Overall Description:

2.1 Product Perspective:

2.1.1 System Interfaces:

- Input data interface
- Control interface
- Machine learning interface
- Simulation interface
- Visualization interface
- Logging interface
- Communication interface
- Configuration interface

2.1.2 Hardware Interfaces:

- Lidar
- Cameras
- GPS
- Inertial Measurement Unit (IMU)
- Control actuators

To interface with these hardware components, the virtual self-driving car uses libraries. For example, the lidar sensor interfaced using the ROS (Robot Operating System) library, while the camera may be interfaced using OpenCV (Open Source Computer Vision Library). The GPS and IMU sensors interfaced using the PyNav library, while the control actuators interfaced using the PyGame library.

2.1.3 Software Interfaces:

- Python Libraries
  - OpenCV
  - Tensorflow
  - Pygame
  - Matplotlib
- API Interfaces
  - ROS (Robot Operating System) API
- Simulation Interfaces
  - Carla Simulator

2.1.4 Communication Interfaces:

- Simulation software interface
- Sensor interface
- Control interface
- User interface:
- Network interface

These communication interfaces are essential for a virtual self-driving car to simulate its behavior in a complex environment. By using these interfaces, the virtual car can receive input data, make decisions, and control its movements to achieve its goals in the simulation.

2.1.5 Memory Constraints:

- RAM
- GPU memory
- Storage

To mitigate memory constraints in a virtual self-driving car, developers can optimize the simulation to reduce the amount of data required to run the simulation, simplify the environment, use lower-resolution textures for rendering, and reduce the complexity of the agents and sensors used in the simulation.

Additionally, using cloud-based solutions and distributed computing can help distribute the computational load and reduce the memory requirements on any single machine.

2.1.6 Operations:

The operations of a self-driving car involve a complex and highly sophisticated system of technology, algorithms, and controls that work together to ensure safe and efficient transportation.

- Sensing and Perception
- Route Planning
- Control and Execution
- Human-Machine Interface
- Maintenance and Monitoring
- Safety Features

2.1.7 Site Adaptation Requirements:

Adapting a site for a virtual self-driving car project involves creating an environment that is safe and conducive to testing and development of the technology.

- Infrastructure
- Controlled Environment
- Realistic Environment
- Safety Features
- Data Collection
- Security
- Accessibility

2.2 Product Functions:

- **Accurate perception:** The virtual self-driving car must be able to accurately perceive the environment it is operating in, including other vehicles, pedestrians, and obstacles.

- **Safe driving**: The virtual self-driving car must be programmed to drive safely and obey traffic laws, including speed limits, traffic signals, and other regulations.

- **Navigation:** The virtual self-driving car must be able to navigate to a destination using a variety of input sources, including GPS, maps, and other sensors.

- **Communication:** The virtual self-driving car must be able to communicate with other vehicles and infrastructure, including traffic lights, to ensure safe and efficient driving.

- **Human interaction:** The virtual self-driving car must be able to interact with human passengers in a variety of situations, including emergencies and unexpected events.

- **Security:** The virtual self-driving car must be secure from hackers and other malicious actors who may attempt to take control of the vehicle or steal data.

- **Maintenance:** The virtual self-driving car must be easy to maintain and update, with regular software updates and patches to ensure optimal performance.

- **Energy efficiency:** The virtual self-driving car should be energy-efficient, with a low carbon footprint and the ability to operate on renewable energy sources.

- **Cost-effective:** The virtual self-driving car should be cost-effective, with a reasonable price point that makes it accessible to a wide range of users.

- **Accessibility:** The virtual self-driving car should be designed to accommodate users of all abilities, including those with disabilities or mobility challenges.

2.3 User Characteristics:

- **Familiarity with technology:** Users of a virtual self-driving car may need to have a certain level of comfort and familiarity with technology, as they will likely need to interact with the car's virtual interface and controls.
- **Comfort with autonomous systems:** Users of a virtual self-driving car may need to be comfortable with autonomous systems and be willing to trust the car's decision-making processes.

- **Accessibility requirements:** Users with accessibility requirements, such as those who are visually impaired, may require special accommodations in order to interact with the virtual self-driving car's interface.
- **Language requirements:** Depending on the market and location of the virtual self-driving car, users may require language support in order to interact with the car's interface and controls.
- **User preferences:** Users may have specific preferences for the car's features and functionality, such as the ability to customize the car's route, speed, or other driving characteristics.

2.4 Constraints:

Virtual self-driving cars are computer simulations and, as such, they have certain constraints that may impact their functionality and capabilities.

- **Limited data and accuracy:** Virtual self-driving cars rely on data and algorithms to make decisions and navigate, which may be limited or less accurate than real-world data. This can affect the car's ability to make safe and effective driving decisions.
- **Technical limitations:** The technology and computing power required to simulate a self-driving car can be complex and expensive. This may lead to technical limitations, such as processing speed, memory constraints, or system crashes, which can affect the car's performance.
- **Lack of physical interaction:** Virtual self-driving cars do not physically interact with the environment, which means they cannot detect or respond to physical obstacles or unexpected events that occur outside of their virtual environment.
- **Limited ability to adapt:** Virtual self-driving cars may have limited ability to adapt to changing road conditions or unexpected events. They rely on pre-programmed algorithms to respond to various scenarios, which may not account for all possible situations.
- **Legal and regulatory constraints:** The development and deployment of virtual self-driving cars may be subject to legal and regulatory constraints, which can

impact their functionality and availability. This may include issues related to data privacy, liability, and compliance with existing road safety regulations.

2.5 Assumptions and Dependencies:

Assumptions:

- The technology will continue to improve at a fast pace, allowing for more reliable and efficient self-driving cars.
- Laws and regulations will be adapted to allow for the safe deployment of self-driving cars on public roads.
- There will be sufficient demand for self-driving cars to justify the investment in the technology.
- There will be sufficient funding and resources available to support the development and testing of self-driving cars.
- The general public will eventually accept self-driving cars as a safe and reliable mode of transportation.

Dependencies:

- Availability and reliability of sensors, algorithms, and control systems needed to create a functional self-driving car.
- The availability of suitable testing sites that can simulate a variety of real-world driving scenarios.
- The development of reliable and secure communication systems that allow for self-driving cars to communicate with other vehicles, pedestrians, and infrastructure.
- The availability of skilled personnel to design, develop, and maintain the self-driving car technology.
- The availability of necessary infrastructure to support self-driving cars, such as charging stations and data centers.

2.6 Apportioning of Requirements:

- **User requirements:** Requirements related to user needs and preferences might include factors such as ease of use, accessibility features, customization options, and language support. These requirements would need to be designed and developed with input from potential users, including those with special needs.
- **Technical requirements:** Technical requirements might include factors such as system performance, reliability, and scalability. These requirements would need to be designed and developed by technical experts, such as software developers, engineers, and data scientists.
- **Safety requirements:** Safety requirements would be critical for a self-driving car, whether virtual or physical. These requirements would need to address issues such as collision detection and avoidance, emergency response protocols, and cybersecurity measures.
- **Regulatory requirements:** Depending on the market and location of the virtual self-driving car, there may be regulatory requirements that need to be met in order to ensure compliance with local laws and regulations. These requirements would need to be identified and addressed in the development and deployment of the virtual car.
- **Business requirements:** Business requirements might include factors such as cost, revenue potential, market demand, and competition. These requirements would need to be considered in order to ensure the virtual self-driving car is a viable and sustainable product.

3. Specification Requirements:

3.1 External Interfaces:

- **Hardware Interfaces:** The self-driving car will require various hardware components to operate, including sensors, cameras, GPS devices, and other electronic systems. These components may need to interface with each other and with the central computer that controls the car's operations.

- **Software Interfaces:** The self-driving car will require various software components to operate, including operating systems, algorithms, and applications that enable the car to perform its functions.

- **Human Interfaces:** The self-driving car may require various human interfaces to enable passengers and operators to interact with the car. These interfaces may include displays, buttons, touchscreens, voice recognition systems, and other user input/output devices.

- **Infrastructure Interfaces:** The self-driving car will need to interface with various infrastructure systems, including traffic signals, road signs, and other transportation systems. The car may use dedicated short-range communications (DSRC), cellular networks, or other communication technologies to interface with these systems.

3.2 Functions:

A virtual self-driving car is a computer simulation of a self-driving car, and as such, it can perform a range of functions that are similar to those of a physical self-driving car. Here are some of the key functions that a virtual self-driving car might perform:

- **Navigation:** A virtual self-driving car can be programmed to navigate a virtual environment, using predefined maps and routes, as well as sensors and algorithms to detect and avoid obstacles and hazards.
- **Control:** A virtual self-driving car can be controlled using a virtual interface, such as a computer keyboard or a mobile app, allowing users to adjust the car's speed, direction, and other driving parameters.
- **Simulation:** A virtual self-driving car can simulate a range of driving scenarios, allowing developers to test the car's performance under different conditions, such as varying weather conditions, traffic patterns, and road types.

- **Learning:** A virtual self-driving car can be programmed to learn from its experiences and improve its performance over time. This might include machine learning algorithms that analyze data from the car's sensors and adjust its driving behavior accordingly.
- **Interaction:** A virtual self-driving car can interact with other virtual objects and entities within its simulated environment, such as other cars, pedestrians, and traffic signals.
- **Communication:** A virtual self-driving car can communicate with other virtual entities, as well as with human users, using virtual interfaces and natural language processing technologies.

Overall, the functions of a virtual self-driving car are designed to simulate the behavior and capabilities of a physical self-driving car, while also providing a safe and controlled environment for testing and development.