

Document for the project plan

Core Feature Specs

Background

Feature 1: A system for watching drones

Feature 2: Tactical characteristics system for lane control

Feature 3: Basic Arcade Car System

Feature 4 Levels: Highway, City, and Warehouse Systems

Feature 1: Drone Surveillance System This addition adds a system of AI-powered patrol drones that interact with players in real time to make the game more fun. These drones have sight cones, several warning circumstances, and a configurable search mechanism that lets them respond quickly to what players do. The fundamental purpose of this strategy is to make the game always tense. It does this by giving players rewards for properly planning their covert manoeuvres and enabling them experience exhilarating bursts of fast action. To make sure this feature works, a few metrics have been set up. During each run, skilled players should have detection events that don't go beyond a specified target threshold. Players should also use stealth zones in at least 60% of the times they are able to escape. Also, to make sure the game runs properly, it should keep a frame rate of at least 60 FPS on mid-range devices.

Setting distance and angle parameters for the drone sight cones is one of the functional needs for this capability. The drones will have three different alert states: Passive, Investigating, and Hunting. The speed of the player can also alter how the drone operates. Players will hear and see feedback to let them know when they are being detected and when they are locked on.

Step 1: Ideas and the Basics

1. What does the drone do? For example, does it show problems that can happen in the future, highlight valuable items, and flag safe lanes?

2. Make a rudimentary model of how drones work.

Put a modest sign for a drone over or near the car.

Have it follow the car

Go a little bit ahead in the lane where the player is.

Step 2: Basic Functions of Drones

Looking into basic problems

Before the car comes to the obstacles, have the drone highlight or ping them.

Show warnings or messages that the player can clearly perceive

Stage 3: Interactions with players and tactical benefits

The drone's tactical talents let it use its powers, like zapping everything in its path, picking up things nearby, and putting up temporary shields.

To get people to think ahead

Use energy limits or cooldowns.

Lane suggestion: If one lane is much safer than the others ahead, put up a "Safe Lane" sign. Advanced players should be able to turn hints on and off if they want to.

Stage 4: Bringing the game into the real world

1. You can change the size depending on how hard or fast the game is. For example, at higher speeds or levels, you can make the drone's scan range or hint accuracy better. Sometimes the player can be able to jam or limited of the drone to make things harder or force it to change.

2. Add extra talents, like how to use the drone with different tactical systems. For instance, the drone can ping a dangerous lane, and the player can either use a shield to get through safely or a boost in a lane that the drone says is clear.

Step 5: Mastery and Developed Features

Adaptable AI: The drone would be able to discover on how the player playing and submit on the directions based on by what means carefully they ensure or don't ensure.

The drone could create as a special actions, like extra trails, when it detects the user managing as a smart activities.

3. Various drone scenarios: Add levels or events somewhere the drone can use extra drones for a short time to make combos or get through hard parts from the player.

Needed assets: a drone model, particle effects for search beams, sound effects for scanning, alerting, and locking on, and a UI detection meter.

Feature 2: Lane Control with Tactical Abilities

Interesting ways to utilise on mobile devices: tap on quick-access icons to change lanes, hold to boost, and combine with tactical skills like EMP, Cloak, and Jam. The controls are simple to use simply slide your fingertip about to change lanes and push your finger down to the bottom to speak. EMPs, invisibility, toying with others, and other things are all at your disposal. These goods are readily available for usage at any moment. It is possible to utilise it with just one finger. You could drive right in and play for a little without worrying about pressing any buttons since playing doesn't need any time spent on a blog. The objective is to eliminate all useless items. Everything that is valuable should remain in this system, and there shouldn't be any repercussions that were carefully considered. As a result, it can make compassionate, impulsive judgements. We have discovered that little intervals of time, lasting no more than a minute or two, are what we want. After a few attempts, people may pick up a technique and feel confident executing one or two looks that are close to its preferred ones.

The back end, the three fast lanes, the boost stuff, the tap-for-to-use techniques, and everything else.

Best Steps

Step 1: Basic Lane Control

1. Find out how to utilise the controls in the game

Practise going smoothly from one lane to the next.

Know how quickly and when you can respond.

2.Be careful and look ahead

Always be on the watch for power-ups and other objects that could get in your way. Plan for lane changes ahead of time instead of waiting until the last minute to react.

Step 2: Use your defensive skills wisely

1. Learn how to use each skill. Know what each power-up or tactical move does, such as a shield or a speed boost. Know when to use a cooldown and what the rules are.

2. When you need them, use your skills. Don't use boosters or shields at random moments; use them right before the hardest parts of the course.

Step 3: Advanced Lane Strategy.

1. Become the leader of the safe, calm lane. Whenever you can, choose lanes with fewer obstacles and more prizes. Put the lanes for future strategic power-ups at the top.

2. Plan your weaving a few seconds ahead of time.

Use shields or leaps to switch lanes in a more hazardous yet lucrative way. Setting up lane control and tactical movements

1. Make combinations in a smart way. Use a shield to stay safe from a lot of threats, and then accelerate up in an open lane. Magnets can pick up anything in a lane, but you can go around them.

2. Use your talents to get out of tough groupings or dead ends. Only use your skills when you need to get away or score.

Step five: Learning and growing

1. See levels and patterns. Look for patterns of barriers and incentives to find the best lanes.

2. The game becomes faster as you go. To go through harder, faster stages, work on your tactical talents.

3. Change your plans even if they don't help, and when an ability cools down, choose the safest way. If you have power, choose the riskier pathways for the highest rewards.

Need buttons for the UI, icons for abilities, boost VFX, haptic patterns, and overlays with instructions.

Part 3

Setting Up a Basic Arcade Car

You need simple controls like buttons, taps, and swipes to manoeuvre the car left or right in this kind of game. To dodge drones and other dangers, players alternate between two and four lanes. It also doesn't have a clear finish line in the area; it just keeps flowing.

The goal is to stay alive and do well. Players must stay away from walls, other cars, and other random or game-controlled dangers. You may get shields, point multipliers, equipment, and lane speed improvements.

These factors make players want to take chances and show off their abilities. The action becomes faster and harsher, which makes it difficult for the player to stay alive.

Steps That Are Perfect

Step 1: Basic Movement and Function.

1. Get the car ready. You can design or import a box vehicle model and then add Unity's rigid body and collider elements to turn it into a game object. 2. Do something easy. When going forward and backward, create code that runs quickly and slowly. For basic steering, turn left and right.

3. Controls for linking Map touch inputs to speed and turn.

Step 2: Touch and Physics

4. To make physics more resemble an arcade game or more responsive, change the mass, drag, and angular drag. To manage the automobile, set its top speed.

Step 3: Feedback with sound and video

1. 4. Change the weight, vertical drag, and lateral drag to make it more fun or responsive.

Pick the fastest speed that lets you keep the car under control.

Step 3: Give feedback with sounds and pictures

1. Sound Feedback: Add engine sound, a file that loops and speeds up and slows down with input. Optional: Sounds of tires squealing while going around sharp turns.

Step 4: User Experience and Camera

8. Get the next camera ready to track the car without any problems. Add bounce or lag to make it seem more professional. 9. HUD Elements: If your designer asks for it, include a speedometer or a simple speed, lap, or health UI.

Stage 5: Issues, crashes, and completing 10. Hit. 9. HUD Elements: a timer or a simple UI for speed or life.

Stage 5: Mistakes and finishing touches

10. Crash. When colliders strike the track or obstacles, they should bounce, slow down, and so on. You may add knockbacks or visual effects to hits.

11. Experience creating changes and testing them. Playtest it to find out how great and entertaining it is. Make changes to the feedback, speed, and physics as needed.

What you need

Make a car from a drawing or get a free Unity asset.

Levels: Highway, City, and Warehouse Systems

Steps

To provide a cyber atmosphere, think of using bright colours, neon lights, holograms, futuristic ads, and digital signs.

Think about what makes each road, city, or warehouse "cyber" friendly. Blade Runner, Ghost in the Shell, and Akira are all amazing cyberpunk movies to draw and make

concept art from.

Find the most important parts:

streets,

settlements

keeping things in the digital age.

Robot for free vehicles

billboard with electricity

Roads that are slippery

Hacks on vehicles cause traffic jams.

Artificial intelligences that don't follow the rules.

Grey boxing blackout time Highway: big roads with plenty of lanes, tunnels, and other things in the way. There are big neon signs and car drones flying above the city. There are also skybridges, little roads with neon undergrowth, night markets, and business buildings. Put in conveyor belts, lit boxes, self-driving forklifts, and drones to make the warehouse seem like a tunnel.

Step 1: Add spawning, faster speeds, and tiny boundaries to the prototype Cyber Highway so that people may travel forever.

Use neon barriers, self-driving automobiles, and shattered holograms to create themed traffic and impediments.

When you first try to draw, use neon signs, glowing guardrails, and enormous skyscrapers in the distance instead of barriers.

First tests: Check the speed and utilisation of the internet.

4. Change the roads and go to Cyber City:

Add ramps, crossroads, and countless "city blocks" that make things happen.

More advanced AI includes self-driving banner boards and groups of drones that don't always perform what you say.

It tastes like seeing neon lights in the rain, big holographic projections, flashing crosswalks, and electric cables everywhere.

The first part of the city-specific SFX includes light rain, alarms that may be heard from far away, and club sounds that are hard to hear.

Step 4: Layer for a polished look Repeat images:

Make neon seem nicer,

every surface shows

more holograms.

Add moving parts like damaged glass and signs that change as you pass by.

A cyberpunk synth-wave soundtrack with sound effects that show how to perform combo techniques and dodge obstacles.

Making things hard

Change how often obstacles appear

prizes for playing without limits

7. Test each part of the game independently and provide feedback.

Stay away from bright lighting and effects that make you perform worse.

Because cyberpunk may seem visually busy, use player feedback and indications to make it easier to go about.

Check that the stages are fair for both games and presentations.

8. Delivery and Rollout on Stage

First, play the demo of the cyber highway. Get some ideas. Cybercity with more risks and parts that seem like highways.

The final level of the game is the warehouse, which makes it seem fresh.

Step 5: Skills at a higher level

Reactive AI: A drone has to learn how players respond and suggest paths depending on how well they do.

If a drone keeps seeing a smart person, it may create more material trails.

You may make fake cyber period roads in the Endless Car Game. The roads are fast and wide, but they don't go through or between cities very often. Long pathways and lanes.

When the player changes paths, they have to move rapidly and avoid cars, trucks, barricades, and other dangers. There might be traffic, sudden falls, construction sites, and wrecked automobiles in the path.

Neon-lit roads, gigantic holographic ads, and city skylines make for a lively atmosphere, much like in the digital era.

Purpose Features: It begins with smooth, clear, "flowy" action to assist players get into driving and speed up combo chains. City (The Never-Ending Car Game)

Setting: cities that are busy with plenty of people, buildings, streets, and intersections.

How to Play

Pay attention to sharper turns, quick lane changes, and staying away from buses. Crosswalks, buses pulling out, construction delays, and delivery drones flying low are all things that get in the way.

To make a cyberpunk setting, there should be a lot of people, bright lights, digital advertisements, traffic lights, and maybe even rain. The feature makes the game tougher by adding obstacles every second, making the environment harder to see, and modifying how enemies and AI respond so you have to be quicker and sharper.

Factory surroundings, such big warehouses with shelves, containers, and machines.

Gameplay Centre: Short passages, moving machinery, and locations for those with limited eyesight all include surprise turns.

There are problems like automatic forklifts, conveyor belts, sliding doors, laser-grid security, and stacks of containers.

The space is gloomy and hard to see because of the bright lights and flashing neon. The lights that flicker enhance tension and don't seem "loose" or random.

Highlight Function: Like a "turn" or rapid, unexpected drift shift that never stops. It works that you run differently, are pushed to an injunction for drama, and alter at the difficulty curve.

1. Plan and goal.

The right environment for the internet age: Think about brilliant colours, neon lights, holograms, commercials that seem like they came from the future, and digital signs. Think about what makes each road, city, or warehouse "cyber" friendly.

Make drawings of your ideas. You may get ideas from cyberpunk movies like Blade Runner, Ghost in the Shell, and Akira.

Figure out what's most important: What do highways, towns, and storage facilities in cyberspace all have in common? Self-driving vehicles, electronic billboards, smooth highways, and maybe even hacked traffic bottlenecks for autos or AI.

The grey boxing blackout phase Make roads with more than one lane, dividers, and tunnels.

There are big neon signs and automobile drones flying about the city. There are also skybridges, walkways that go up, small lanes with neon lights, night markets, and business offices. Add conveyor belts, illuminated containers, forklifts, and self-driving drones to the warehouse to make it appear like a tunnel.

First, create CyberHighway

Basic mechanics of a prototype: Add spawning, speed enhancements, and minor obstacles to make driving last forever.

Set up themed traffic and things that get in the way. Neon barriers, self-driving automobiles, and shattered holograms are all perilous.

First try at art: Instead of barriers, use bright neon signs, luminous guardrails, and buildings that are far away.

First tests: Look at the speed and traffic on the internet.

2. Go to Cyber City Change roads by adding ramps, junctions, and an endless "city block" spawn.

More complicated AI: swarms of self-driving drones that don't always follow the rules and self-driving banner boards.

Taste with your eyes: neon lights falling from the sky, huge holographic projections, crosswalks flashing, and electric cables everywhere.

Rain, distant sirens, and club beats that are hard to hear are all examples of city-specific SFX.

5. Cyberpunk Stage 3: Change in the warehouse's mechanical parts: Laser grids, tight turns, and moving machines may create new problems.

There are random warehouse hallways with robot arms for procedural lanes. Flickering lights, moving parts that are lit up by fluid, and containers with changing displays are all examples of images.

Mechanical clinks, distant footfall, and a drone. 6. Stage 4: Add a layer to improve and polish.

Repeat visuals: Make neon better by adding rain, fog, reflections on all surfaces, and holograms.

Add things that change as you drive by, including signs that change, broken glass, and people in the city that react.

A cyberpunk synth-wave soundtrack with sounds that tell you when to use combinations and when to avoid obstacles.

Getting harder: Change the number of rewards and obstacles to play forever.

7. Tests and comments

Playtest each part on its own.

Watch out for bright lights and effects that make performance worse.

Cyberpunk may become visually busy, so listen to what players say to make it easier to go about and provide tips.

Make sure the levels are balanced for both presentation and play.

8. Delivery and Rollout on Stage

The first playable stage or demo should be the cyber highway.

Get some ideas.

Add a cyber city based on a roadway, with more hazards and features.

The final level of the game is called Warehouse, and it's a twist on the cycle.
Staged delivery makes each scenario more authentic and illustrates how people act.

In general: Structures for game range

Stage Explanation Goal Possible Game Least likely to make that shows the basic
gameplay loop: driving, changing lanes, avoiding hazards, and getting basic feedback.
Prototype/demo that can be played

Full Game / Done Game Completely mastered skill with control, adaptable from the AI,
in different areas (Highway → City → Warehouse), and covered approaches. The game
itself has been announced.

Feature 1: Drone Surveillance System. Purpose: AI drones that are always on and can
see and react to what players do to provide danger, stealth, and duties that change
based on what players do.

Factor Est. Effort Priority

fundamental drone model and journey A straightforward floating entity that views or
hovers above the player's automobile 1 week Very high

Zone of little detection A simple mesh (like a Unity spotlight) makes a visible cone that
sets off an alarm. High

Three Alert states Passive, Investigating, and Hunting, each with sound and colour
feedback One week High

Simple audio and visual feedback Hoots, a flashing UI, or little indicators to help you
find things Three days Medium

Regular goal Improving the drone scripts and limiting the number of active drones Still
going High

Aftermath: patrols of rigid drones that react to the player, as well as adjustments to the
stealth gameplay.

As Add-ons for Finished Games

Factor Est. Effort Priority

Adaptive AI (knowledge behaviour) Drone changes the scan radius dependent on the
circles of the players. Two to three weeks High

Strategic drone abilities: EMP zap, shield help, and picking up items 2 weeks Medium

Multiple drone situations Chases that are planned or coordinated tactics 1–2 weeks
Medium

Advanced SFX and VFX Scan beams, noises that notify you, and animation 1 week High

System for measuring stealth control and issues Detect-event ratio, planning in secret
1 week Medium

Result: Reflex, the growth of the drone ecosystem that guides the curve's pressure with enthusiasm.

Feature 2: Control of lanes and tactical skills

Goal

Basic input loop: lane switching and tactical abilities that make short play sessions enjoyable and easy to get into.

Factor Explanation Estimated Effort System for moving in priority lanes Input for mobile devices that uses swipes or taps 1-week High Boost mechanic Hold to speed up; speed up for 2–3 days High

One protective talent, like a shield Ability to survive with a cooldown 3-day Medium

Basic HUD Lane indication, ability button, and cooldown UI Three days Average

Simple instructional with instructions for tapping and swiping 2 days Low

Aftermath: Simple, intuitive lane-switching navigation that thinks it can work on mobile.

Add-ons for Finished Games Description, Estimated Effort, and Priority

3–4 tactical skills (EMP, Cloak, boost) Using quick-entry UI icons 2 weeks High

System for haptic feedback Vibration signals for impacts and enhancements Three days Medium

Ability timing score multiplier for the combo system. 1 week Medium

Adaptive instruction: gives players the right recommendations depending on their mistakes 1 week Medium

Managing energy and cooldown Push the limitations to get people to plan. 1 week High

The outcome: A mobile control system that is rich yet simple, with significant player decision loops.

Feature 3: Basic Arcade Car Setup Purpose

The most important parts of gameplay mechanics are the unending drive, avoiding obstacles, and how the car feels.

Part	Description	Estimated Effort	Priority
------	-------------	------------------	----------

Car "prefab"	as a box collider and a stiff body	Easy automobile system for players	1 to 4 days Very high
--------------	------------------------------------	------------------------------------	--------------------------

Moving on and finding your way	Written forward force and side-by-side movement (left or right system)	1 to 4 days	High
--------------------------------	--	-------------	------

Accident decision	Hit an obstacle. Reset or slow down.	2–3 days	High
-------------------	--------------------------------------	----------	------

A system that follows a camera that follows smoothly without any little but no lagging		Five days	High
--	--	-----------	------

Simple power-ups and obstacles	Making a basic with a lot of obstacles and a few pickups	1.5 weeks	High
Aftermath:	An everlasting driving encompass should be working from start to finish.		

Add-ons for Finished Games	Factor	Description	Est. Effort	Priority
----------------------------	--------	-------------	-------------	----------

Tuning physics	More precise handling and a responsive feel	1-week	High-Quality
----------------	---	--------	--------------

VFX/SFX	Sound of the engine, screeching tires, and neon trails	1 week	Medium
---------	--	--------	--------

Damage to vehicles and feedback	Cracks that you can see, sparks, and the phone shaking	On-going	Medium
---------------------------------	--	----------	--------

scoring system:	As a local scoring system inside	On-going	High
-----------------	----------------------------------	----------	------

Result: an arcade control that is fluid and easy to use, with all the additional characteristics it needs.

Feature 4: Levels: Highway, City, and Warehouse

Goal

Separate parts of an environment that are easy to see and becoming harder for the challenging areas

Factor Est. Effort Priority Description

Highway sector (looped) Never-ending track of procedures 1–2 weeks High Procedural obstacle spawning Random lanes, with a week between them High

Basic skybox and neon lights Cyberpunk environment prototype: 3 days; Medium Performance check (lighting) Optimising a cellphone for 60 FPS is still going on. High Aftermath: a playable vertical slice that includes a highway unending run with fundamental components.

Add-ons for Finished Game

Factor Est. Effort Priority

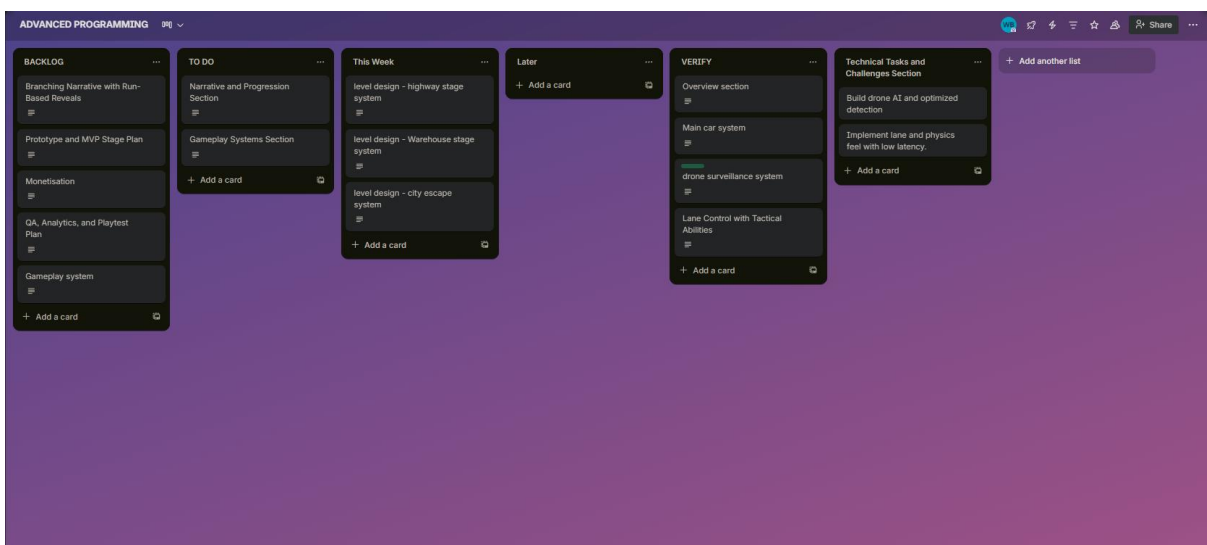
Cyber City sector: complicated intersections, traffic, and drones 2 weeks High Warehouse sector Laser grids, forklifts, and pictures inside Two weeks High Environmental Interfaces Signs, glass, fog, and reflections 1–2 weeks Medium

Curves of adaptive difficulty Adjust how often monsters spawn with passion for 1-week High-quality Cyberpunk music and sound effects Synth-wave and rain sounds if you want them 1 week Medium

Result: The multi-sector endless runner was fully completed in a cyberpunk setting.

Planner for Project Management

I am preparing these things on Trello, which is an advanced programming tool. It helps me plan what to accomplish this week, what to do later, what to verify, and any technical **chores and issues I could have.**



(Screenshot 02/11/2025)

Link: <https://trello.com/b/ey8TUfXP/advanced-programming>