

# 2025 **FIRST<sup>®</sup>** Robotics Competition

## Guía de mejora e iteración de KitBot

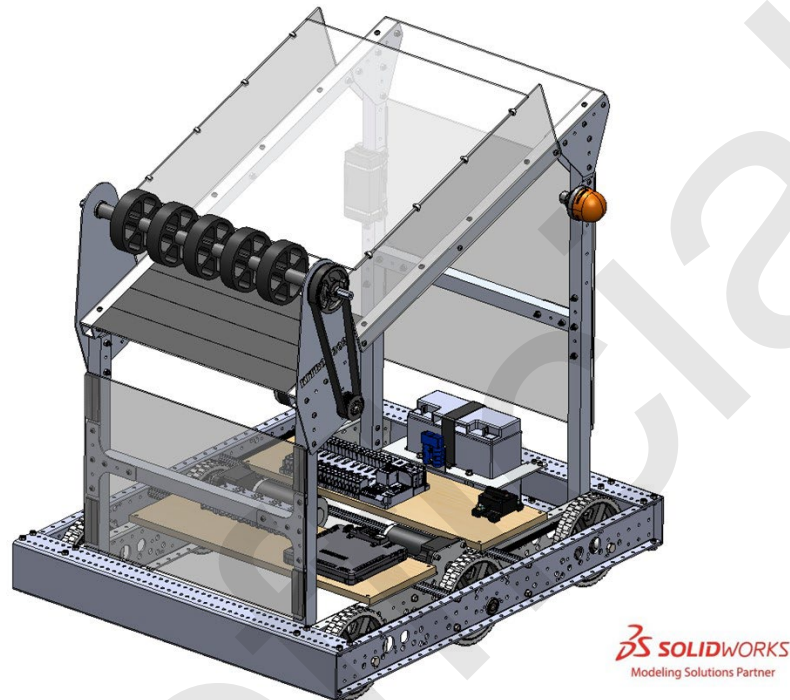
# 1 Índice de contenidos

---

1	Índice de contenidos.....	2
2	Introducción.....	3
3	Cuándo empezar a pensar en los cambios .....	5
4	Mejora - Añadir nuevas funciones.....	6
4.1	Análisis del juego .....	6
4.2	Identificación de mejoras .....	8
4.3	Próximos pasos.....	8
5	Iteración - Mejorar la capacidad existente .....	9
5.1	Pruebas intencionadas.....	9
5.2	Lluvia de ideas para mejorar.....	9
5.3	Próximos pasos.....	15
6	Selección descendente - Convertir las ideas en un plan .....	16
6.1	Priorizar .....	16
6.2	Análisis - Estimación de recursos y alcance .....	16
6.3	Analizar - Fijar su "presupuesto" .....	17
6.4	Hacer planes.....	17
6.5	Próximos pasos.....	19
7	Apéndice A - Ejemplo de análisis del juego básico, 2023 .....	20
7.1	Ejemplo - Lista de tareas del robot .....	20
7.2	Ejemplo - Cómo posicionarse bien.....	21
7.3	Ejemplo - Cómo ganar partidos.....	21
7.4	Ejemplo - Mejoras .....	23

## 2 Introducción

Figura1 : 2025 KitBot



El KitBot para REEFSCAPE<sup>SM</sup> presentado por Haas es capaz de realizar las siguientes acciones. Algunas acciones necesitarán que el equipo añada explícitamente código para que esto sea posible (por ejemplo, el código del Autónomo):

- Conduzca por el campo utilizando un tren de transmisión diferencial (también conocido como "chasis/tanque") diseñado para alcanzar una velocidad máxima alcanzable de aproximadamente 15 pies por segundo (aproximadamente 4,5 m/s). El KitBot no cabe debajo de las jaulas profundas ni de las poco profundas, pero puede maniobrar entre ellas o apartarlas mientras las atraviesa.
- Precargar un Coral para utilizarlo en Autónomo
- Puntuación Dejar puntos
- Puntuación por dejar el Coral en la L1 del arrecife
- Recolectar el Coral de la Estación de Corales
- Jugar a la defensiva

Se trata de un conjunto bastante básico de capacidades con respecto a todas las tareas posibles del juego. Además, el KitBot ha sido diseñado para ser muy sencillo, lo que significa que puede haber oportunidades de iterar y mejorar las capacidades existentes. Teniendo esto en cuenta, los equipos pueden optar por añadir componentes adicionales que permitan al robot recoger piezas de juego del

suelo, subir a la Barcaza, ¡o más! Este documento plantea un posible proceso que podría seguirse para realizar una lluvia de ideas y decidir posibles mejoras.

Unoficial

## 3 Cuándo empezar a pensar en los cambios

---

Antes o después de construir su KitBot, puede que desee revisar el juego y determinar si hay capacidades adicionales que le gustaría tratar de añadir al robot de su equipo. El KitBot está diseñado pensando en futuras ampliaciones, y animamos a la mayoría de los equipos a montarlo y jugar un poco con él antes de modificarlo o añadirle algo, pero hay ventajas potenciales si se consideran las ampliaciones antes de la construcción. Estas son algunas ventajas y desventajas que pueden ayudarte a decidir si hacer este primer ejercicio antes o después de haber construido el KitBot.

### Antes de construir el KitBot

- Pros
  - Permite a los diseñadores trabajar de inmediato en lugar de tener que esperar a que finalice la construcción.
  - Mayor facilidad para realizar cambios en las piezas del diseño base antes de ensamblarlas
- Contras
  - Puede que estés diseñando más elementos del robot del que puede construir con éxito
  - Puedes estar realizando cambios en partes del diseño base que no comprende completamente

### Después de construir el KitBot

- Pros
  - Certeza de tener un robot que puede jugar a algunos aspectos del juego
  - Mejor capacidad para priorizar la iteración (mejorar las capacidades existentes) frente a la mejora (agregar nuevas capacidades)
  - Mejor comprensión de la función del diseño base y de lo que puede y no puede cambiar.
  - Las pruebas y la práctica pueden realizarse simultáneamente con el trabajo en mejoras.
  - Las mejoras en el desarrollo de software pueden probarse inmediatamente
- Contras
  - Nada que diseñar hasta que finalice la construcción
  - Cambiar las piezas de la base puede requerir desmontar cosas
  - Es posible que desee utilizar tuercas y tornillos en lugar de remaches para desmontar las cosas más fácil

## 4 Mejora - Añadir nuevas funciones

Este ejercicio puede realizarse antes o después de construir el KitBot (ver sección 2 de pros/contras de cada opción). Si se realiza este ejercicio después de construir el KitBot, recomendamos completar las partes de lluvia de ideas de todos los ejercicios de Mejora e Iteración (secciones 34 & ) y luego combinarlas con la parte de selección descendente del ejercicio (sección 5) para priorizar los cambios que perseguirá su equipo.

Aunque no es estrictamente necesario, contar con una serie de objetivos de equipo es muy útil para la parte de este ejercicio dedicada a delimitarlos. Para obtener más información sobre el establecimiento de objetivos de equipo, consulte el vídeo "Establecimiento de objetivos" enlazado desde la página [Recursos para la gestión de equipos](#), en la sección Organización del equipo.

Los ejercicios y técnicas que aquí se presentan constituyen una versión abreviada de un proceso completo de análisis del juego y de la estrategia del robot. Para un recurso más completo sobre este proceso, vea la sección de Diseño Estratégico de la presentación "Estrategias Efectivas de FIRST® para Diseño y Competencia" de la página de [Recursos Técnicos](#) bajo la sección Otros Recursos Técnicos->Scouting/Estrategia.

### 4.1 Análisis del juego

#### 4.1.1 Analizar el juego - ¿Qué más podría hacer nuestro robot?

El primer paso para pensar en posibles mejoras es analizar el juego y elaborar una lista de cosas que podría hacer el robot. A continuación se ofrece un conjunto abreviado de preguntas para hacerlo. Para un proceso más completo, consulte la "Hoja de trabajo para la puesta en marcha" en la página de [Recursos técnicos](#), en la sección Otros recursos técnicos->Puesta en marcha. No se trata sólo de una lista de tareas del juego (aunque es un buen punto de partida). No empieces a pensar todavía en cómo realizará estas tareas el robot. A continuación se incluye una lista de preguntas que pueden ayudar a orientar a su equipo en la elaboración de esta lista:

1. ¿Cuáles son las formas de conseguir puntos en el juego?
2. ¿Puede desglosarse más alguno de estos elementos (por ejemplo, puntuar desde diferentes distancias puede ser una capacidad única)?
3. ¿Cómo adquiere el robot las piezas del juego? A menudo hay múltiples formas de adquirir que requieren capacidades únicas.
4. ¿Hay otras tareas que podrían ayudar a su alianza (por ejemplo, elementos de campo que manipular, formas de ayudar a los integrantes de la alianza, etc.)?
5. ¿Existen formas de ralentizar la puntuación de los oponentes que requieran capacidades únicas del robot (es decir, que no se limiten a defender utilizando el tren motriz del robot)?

¿Busca un ejemplo? Echa un vistazo [Apéndice A - Ejemplo de análisis del juego](#) básico, 2023 a la posible lista de tareas para el partido de 2023.

#### 4.1.2 Analizando el juego - ¿Cómo nos clasificamos?

Si los objetivos de tu equipo implican intentar clasificarse lo más alto posible (por ejemplo, intentar ser capitán de una alianza, intentar maximizar los Puntos de Distrito para clasificarse para un Campeonato de Distrito, etc.), es importante revisar los criterios por los que se clasifican los equipos. Si los objetivos de tu equipo no implican una clasificación alta (por ejemplo, ser elegido para una alianza, diseñar un mecanismo genial, ganar un premio, etc.) puedes saltarte esta sección. Los criterios de clasificación se encuentran normalmente en la sección de torneos del [manual](#) y suelen basarse en los puntos de clasificación RP (Ranking Point), que se obtienen por victorias o empates y por completar objetivos de juego específicos. Suponiendo un juego típico de FIRST<sup>®</sup> Robotics Competition con clasificación en este estilo, hágase estas preguntas:

1. Aparte de ganar partidos, ¿cómo pueden las alianzas ganar puntos de clasificación?
2. Para cada uno de estos RP(Ranking Point) qué capacidades robóticas se necesitan para:
  - a. maximizar la capacidad de nuestras alianzas para ganar el RP(Ranking Point) ?
  - b. ¿Conseguir el RP(Ranking Point) si se juega con otros 2 robots como el nuestro?
  - c. ¿Hacer la contribución mínima al RP(Ranking Point) para nuestra alianza?
3. ¿Cuáles son el <sup>primer</sup> y el <sup>segundo criterio</sup> de desempate después de RP(Ranking Point) (normalmente es poco probable que haya un empate más allá de esto y muy poco probable que tal empate decida más de un rango o dos como máximo)? No deberían ser una influencia importante a la hora de seleccionar capacidades, pero pueden suponer un pequeño empujón en caso de decisión difícil entre dos capacidades a la hora de establecer prioridades.

#### 4.1.3 Analizar el juego - ¿Ganar partidos?

Si los objetivos de tu equipo pasan por ganar partidos, tendrás que dedicar algo de tiempo a analizar el juego y pensar en cómo es una alianza ganadora. Dependiendo de sus objetivos exactos, algunas de estas preguntas de ejemplo pueden ser más o menos pertinentes.

1. Enumera todas las tareas puntuables del juego con sus valores en puntos.
  - a. ¿Cuántas veces puede completar una alianza cada una de estas tareas? Puede que no sea una respuesta única y limpia, algunos juegos pueden tener tareas que se excluyen mutuamente.
2. Para cada tarea, calcula la rapidez con la que crees que puede realizarse.
  - a. Las respuestas a esta pregunta variarán mucho en función de la fuerza del robot, por lo que es probable que desee elegir una "clase" particular de robot y responder de forma coherente con ese nivel en mente. Para la puntuación que requiere una pieza de juego, recuerde incluir el tiempo para ir a buscar esa pieza de juego y volver al lugar de puntuación.
3. Para algunas tareas, también puede considerar un "índice de éxito". Es el porcentaje de veces que crees que el robot completará la tarea en el tiempo que has estimado.
  - a. Para una tarea de lanzamiento esto podría ser obvio, con qué frecuencia entrará la pieza de juego en la portería. Para otras tareas puede ser menos obvio, pero trate de considerar la frecuencia con la que puede tener que volver a alinear e intentarlo de

nuevo, por ejemplo. Expresa la respuesta en decimales o fracciones (por ejemplo, 80% de éxito = 0,8).

4. Utiliza los números de los tres pasos anteriores para calcular los puntos/segundo de cada tarea puntuable (éxito\*puntos/segundos) y ordénala de más a menos eficiente.

Basándose en la lista de tiempos y eficacia que ha elaborado, evalúe lo siguiente (omite lo que no sea apropiado para los objetivos de su equipo):

1. ¿Cómo es el partido para una alianza ganadora media en Calificaciones?
2. ¿Qué aspecto tiene el partido para una alianza que ganaría alrededor del 80% de los partidos de clasificación?
3. ¿Cómo es el partido para una alianza que puede ganar un partido de desempate?
4. ¿Cómo es el partido para una alianza que gane el evento?

Utilizando los objetivos de sus equipos y una evaluación de sus capacidades, calcule su papel en cada una de estas alianzas:

1. ¿Cuántos puntos pretende aportar a la alianza?
2. ¿Hay tareas específicas en estas alianzas a las que crea que debe contribuir?
3. ¿Hay tareas específicas que quiera dejar para que contribuyan los socios de la alianza?

## 4.2 Identificación de mejoras

Ahora compara las capacidades del KitBot con la lista de tareas y totales de puntos que acabas de generar en función de los objetivos de tu equipo. ¿Qué capacidades necesita añadir para cumplir los puntos y capacidades identificados? ¿Cuál de ellas es la más importante?

## 4.3 Próximos pasos

Si estás pensando en realizar mejoras antes de construir tu KitBot, pasa a la sección [6](#) de ideas sobre cómo convertir estas ideas en un plan. Si ya has construido tu KitBot, pasa a la sección [5](#) para obtener información sobre cómo probar tu robot e identificar otras posibles mejoras para incluirlas en tu proceso de planificación.



## 5 Iteración - Mejorar la capacidad existente

La iteración es un componente esencial del proceso de diseño de ingeniería, y el diseño de un robot para la FIRST Robotics Competition no es diferente. Una vez que hayas construido tu KitBot, querrás utilizar el tiempo que quede antes o entre tus eventos para hacer dos cosas: practicar e iterar. Para más información sobre la selección y la práctica de los controladores, consulte la "Guía para la selección de controladores" y la "Mejora del rendimiento de los controladores" en la página de [Recursos](#) Técnicos de la sección Otros Recursos Técnicos.

A continuación se describe un sencillo proceso de iteración que consta de los siguientes pasos:

- Prueba
- Idear
- Planificar
- Implementar

### 5.1 Pruebas intencionadas

El primer paso en un proceso de iteración consiste en identificar las áreas susceptibles de mejora. Para tu KitBot recomendamos hacerlo mediante pruebas intencionadas. La mayor parte de estas pruebas consistirán en utilizar el robot como se espera que lo haga durante un partido, lo que también servirá como práctica de conducción. La única excepción son las pruebas hasta el fallo. Siempre que disponga de tiempo suficiente (recomendamos al menos una semana) antes de su evento, puede probar intencionadamente su robot en situaciones comprometidas, como chocar con cosas de forma agresiva, adquirir demasiadas piezas de juego, pulsar botones en un orden incorrecto, etc. para ver qué se rompe. Mientras pruebas el robot, anota las respuestas a las siguientes preguntas:

1. ¿Qué se ha roto? ¿Cómo se rompió?
2. ¿Qué está haciendo el robot de forma incoherente (por ejemplo, no consigue adquirir las piezas del juego, se le caen las piezas del juego, pierde oportunidades de puntuar, etc.)?
3. ¿Qué tareas llevan mucho tiempo a los conductores (por ejemplo, adquirir las piezas del juego, atravesar el campo, alinearse para marcar, soltar la pieza del juego, etc.)?

Si dispone de más de una pieza de juego, puede comparar cuidadosamente el rendimiento del robot con cada pieza de juego para ver si hay alguna variación. A continuación, coge una o dos piezas del juego y cámbialas de la forma que esperas que ocurra durante el juego (infladas en exceso o en defecto, deformadas, arañadas o rotas, etc.). Repita las pruebas del robot y documente cualquier cambio en el comportamiento.

### 5.2 Lluvia de ideas para mejorar

Ahora que ya ha identificado las posibles áreas de mejora, el siguiente paso es generar ideas para mejorar esos puntos débiles. Este documento divide la lluvia de ideas en dos partes: mecánica y software/eléctrica. Esto se hace para ayudar a que el proceso sea un poco más manejable, y dado que

estos temas a menudo requieren diferentes conjuntos de recursos, con frecuencia pueden perseguirse en paralelo.

### 5.2.1 Lluvia de ideas para mejorar - Mecánica

Para cada una de las áreas de mejora potencial identificadas en sus pruebas, haga una lluvia de ideas sobre algunos posibles cambios mecánicos que podrían mejorar el rendimiento. A continuación se incluyen algunas preguntas para cada una de las tres categorías:

#### 5.2.1.1 ¿Qué se ha roto?

1. ¿Hay alguna forma de cambiar la geometría de la pieza que se ha roto para hacerla más resistente?
2. ¿Existen materiales alternativos que podamos utilizar para reforzar la pieza con la misma geometría?
  - a. Recuerde que más fuerte no siempre significa más rígido, a veces hacer una pieza de un material más flexible (por ejemplo, cambiar una pieza de aluminio por policarbonato) puede permitirle absorber energía sin llegar a la deformación permanente.
3. ¿Realmente queremos que la pieza sea más resistente o queremos fabricar repuestos?
  - a. Si la causa del fallo parece anormal y puede ocurrir con poca frecuencia en un torneo, si es que ocurre, puede que quiera dejar la pieza como está, o a veces incluso hacerla ligeramente más débil. Aunque esto pueda parecer contraintuitivo al principio, diseñar un eslabón débil conocido y estar preparado para sustituirlo si se produce un daño puede ayudar a evitar daños inesperados en piezas más caras o más difíciles de sustituir. Los pasadores de seguridad mecánicos y los fusibles eléctricos son algunos ejemplos de este tipo de piezas\_sacrificables.

#### 5.2.1.2 ¿Qué es incoherente?

1. ¿Hay alguna forma de modificar la geometría de las piezas para mejorar la coherencia?
  - a. Las ideas para ello suelen incluir la adición o eliminación de restricciones o guías, el aumento o la disminución de la compresión sobre una pieza de juego, la eliminación de huecos o hendiduras con los que una pieza de juego puede estar interactuando inadvertidamente, o la adición de ruedas o rodillos adicionales conectados a un sistema existente.
2. ¿Hay alguna forma de cambiar los materiales para intentar mejorar la coherencia?
  - a. Por lo general, las ideas en este sentido buscan aumentar o reducir la flexibilidad o la fricción.
3. ¿Existen formas de aumentar la coherencia añadiendo un accionamiento adicional?
  - a. Ejemplos de esto incluyen la expansión activa de un mecanismo para la adquisición de piezas de juego, la adición de pestillos o puertas adicionales en una ruta de movimiento de piezas de juego, la adición de actuación adicional para ayudar a asegurar la orientación de piezas de juego en un sistema, etc.

### 5.2.1.3 ¿Qué lleva mucho tiempo?

1. ¿Existen guías mecánicas que puedan añadirse para agilizar al conductor la realización de esta tarea?
  - a. Algunos ejemplos son las cuñas o embudos que interactúan con los elementos del campo, o un marcador físico en un robot que ayuda al conductor a ver mejor la alineación con un elemento del campo.
2. ¿Hay alguna forma de hacer que el robot sea más resistente a la desalineación?
  - a. Algunos ejemplos son el aumento del tamaño de la abertura para la adquisición de piezas de juego o el cambio de la geometría de la puntuación para trabajar desde diferentes alineaciones.
3. ¿La cuestión es cuánto tarda el mecanismo en completar la acción? ¿Se puede acelerar cambiando la marcha?
  - a. Si piensas acelerar un mecanismo, asegúrate de comprobar cuánta corriente consume en su configuración actual. La corriente aumentará aproximadamente de forma lineal con el engranaje, por lo que si al acelerar el mecanismo la corriente alcanza niveles preocupantes, puede que también tengas que añadir otro motor.

### 5.2.1.4 Ideas concretas para el KitBot de 2025

A continuación se presentan algunas sugerencias específicas de posibles mejoras mecánicas para el KitBot 2025. Sáltate esta sección si prefieres ceñirte a las ideas que tú mismo generas.

1. **Añadir blindaje para evitar la posesión** inadvertida de Coral - Mientras que el KitBot tiene blindaje para evitar las formas más probables de que el Coral se atasque (al recoger o entregar) todavía tiene lugares donde el Coral podría atascarse inadvertidamente (cae accidentalmente de un compañero u oponente, etc.) que contaría como su único Coral permitido para controlar. Podría añadirse blindaje a estos lugares para reducir esta probabilidad.
2. **Cerrar las brechas** de adquisición/entrega de Coral - Para simplificar las cosas, el KitBot adquiere Coral con una rampa completamente pasiva. Esto deja un hueco entre la rampa y la pared causado por la necesidad de que la rampa se detenga en el borde del perímetro del marco. El KitBot también suministra Coral desde completamente dentro del perímetro del marco, dejando un hueco entre el final de la rampa y el Arrecife. Se podría utilizar un mecanismo accionado para rellenar este hueco tras el inicio del partido, extendiendo la rampa por encima de los parachoques. Es probable que desee detenerse en el borde del parachoques o ligeramente antes para permitir que éste continúe absorbiendo cualquier impacto con la pared o el arrecife.
3. Alineación del coral - Con el diseño del KitBot es posible que el coral no caiga en el robot y descansa contra el rodillo completamente recto, lo que a veces hace que se expulse de forma inconsistente. ¿Hay alguna forma de añadir elementos mecánicos adicionales (motorizados o no) que puedan mejorar esto?

## 5.2.2 Lluvia de ideas sobre mejoras - Eléctricas/Software

Las mejoras eléctricas y de software se agrupan porque suelen ir de la mano. La introducción de mejoras en el software de un robot suele requerir la intervención de sensores para que el software reaccione a lo que está sucediendo. Al igual que en el brainstorming mecánico, repase cada una de sus áreas de mejora potencial y haga una lluvia de ideas sobre cómo la retroalimentación del software o la automatización podrían proporcionar mejoras.

### 5.2.2.1 ¿Qué se ha roto?

1. ¿Podría haberse evitado el fallo limitando el recorrido de un mecanismo?
  - a. En la mayoría de los casos, se utilizan finales de carrera para indicar los límites del recorrido admisible. Una alternativa es el uso de "límites suaves" para los mecanismos que ya tienen retroalimentación de posición absoluta (es decir, usted sabe que su mecanismo está a 30 grados por lo que puede limitar con el software de viajar más lejos en esa dirección).
2. ¿Podría haberse evitado el fallo con límites de corriente?
  - a. Los límites de corriente pueden ayudar a reducir la fuerza máxima que puede emitir un mecanismo y pueden limitar el sobrecalentamiento del motor si éste se cala debido a un atasco inesperado del mecanismo.
3. ¿Podría haberse evitado el fallo mediante un bloqueo de software?
  - a. A veces los robots tienen mecanismos que sólo deben accionarse cuando el robot se encuentra en un estado determinado, de lo contrario algo podría romperse. Estas exclusiones pueden aplicarse en el software para evitar que los conductores cometan un error y dañen el robot.

### 5.2.2.2 ¿Qué es incoherente?

1. ¿Son los controles intuitivos y obvios?
  - a. Si los conductores cometen errores, podría significar que necesitan más práctica, pero también que los controles no son intuitivos. Una vez seleccionados los controladores, asegúrate de trabajar con ellos para que los controles tengan sentido para ellos. Lo que puede ser obvio para una persona puede ser extraño para otra. Un mal diseño de los controles también puede incluir el uso de un único botón para alternar entre un estado y otro que los conductores no pueden ver. Por ejemplo, si al pulsar un solo botón se enciende la rueda del lanzador y al pulsarlo de nuevo se apaga, ¿cómo sabes si lo has pulsado y te has acordado de apagarlo?
2. ¿Existen problemas de sincronización que la automatización del software podría ayudar a evitar?
  - a. A veces la incoherencia se debe a que un conductor hace algo en el momento equivocado. La automatización de estos tiempos mediante programas informáticos puede ayudar a eliminar las incoherencias humanas del proceso. Un ejemplo de ello es la expulsión de Coral. Si el botón se mantiene pulsado brevemente (de modo que el

- rodillo deja de recibir energía antes de que la Coral abandone por completo el robot), ¿sucede algo diferente que si el botón se mantiene pulsado durante más tiempo?
3. ¿Existen secuencias que un programa informático pueda gestionar automáticamente?
    - a. A veces, para realizar una acción, un robot tiene que accionar varios mecanismos en un orden concreto, con o sin una temporización específica. ¿Puede utilizarse la automatización de programas informáticos para imponer este orden o automatizar toda la secuencia para evitar errores?
  4. ¿Existen bloqueos de software que puedan ayudar a evitar problemas?
    - a. A veces, la incoherencia puede deberse a que se realiza una acción cuando el robot no se encuentra en el estado adecuado (no está alineado con el campo, el mecanismo de interacción no está bien preparado, etc.). ¿Puede utilizarse software para detectar alguno de estos estados incorrectos e impedir que el conductor realice la acción?
  5. ¿El comportamiento del robot parece incoherente incluso con lo que parecen acciones correctas del conductor?
    - a. A veces, esta incoherencia se debe a cambios en el robot o en el entorno, como un voltaje diferente de la batería o un poco más de fricción a medida que se desgasta una pieza. Añadir sensores y controles de software puede ayudar a garantizar que el robot haga siempre lo mismo.

#### 5.2.2.3 ¿Qué lleva mucho tiempo?

1. ¿Puede automatizarse la alineación?
  - a. La forma más habitual es utilizar sensores de visión para detectar la ubicación del robot en el campo y utilizarlos para navegar hasta la ubicación deseada, pero dependiendo de la tarea también pueden ser útiles otros sensores como telémetros o interruptores de haz.
2. ¿Puede dar más información a los conductores?
  - a. Los conductores pueden hacer todo lo posible con los ojos para ver lo que ocurre, pero el robot también puede ayudar enviando señales adicionales. La transmisión de una cámara al puesto del conductor, la visualización de los estados del robot en el salpicadero, el uso de luces en el robot para señalar el estado y la adición de retroalimentación del controlador (si procede) son ejemplos de formas de proporcionar información sobre el estado del robot a los conductores.

#### 5.2.2.4 Ideas concretas para el KitBot de 2025

A continuación se ofrecen algunas sugerencias específicas para la posible automatización del software del KitBot 2025. Sáltate esta sección si prefieres ceñirte a las ideas que generas tú mismo.

1. Control **del rodillo en bucle** cerrado: cambiando el motor que acciona la rueda delantera del lanzador a un motor con codificador incorporado (Venom, NEO/Vortex, Falcon, Kraken) o añadiendo un codificador directamente al motor existente o engranado en él, se puede pasar del actual control en bucle abierto (estimación de la velocidad de la rueda basada en la tensión suministrada) a un control en bucle cerrado (control preciso de la velocidad de la rueda basado

en la realimentación). Para obtener información sobre la velocidad del sensor elegido, deberá consultar la documentación del motor o sensor correspondiente. Para utilizar esta información para controlar la velocidad de la rueda, consulte la sección Introducción a los controles [avanzados de la documentación](#) WPILib, especialmente el artículo sobre "Ajuste de un controlador de velocidad del volante."

2. Detección de corales - Se podría añadir al robot un interruptor, un sensor de proximidad o un sensor de haz de rotura para detectar cuándo hay corales en el robot. Podrías utilizar esta información para proporcionar información al conductor a través de las luces, el salpicadero o la vibración del mando (si utilizas un mando con vibración, como un mando de Xbox).
3. Alineación de la visión - El KitBot puede alinearse fácilmente con el Reef debido a la gran apertura, la proximidad a sus controladores, y la capacidad de ejecutar los Bumpers contra el Reef para alinear la profundidad. Alinearse con la Estación Coral puede ser un poco más difícil debido al ángulo desde sus conductores, pero el Jugador Humano probablemente tenga mucho espacio para moverse y ajustarse. La adición de una cámara para procesar [AprilTags](#) puede utilizarse para alinearse con cualquiera de las ubicaciones de forma más rápida y precisa. Puedes hacerlo añadiendo una webcam conectada al roboRIO o añadiendo un sistema de visión externo como un [Limelight](#), [RaspberryPi con WPILibPi](#) o placa [coprocesadora con PhotonVision](#). Una vez que disponga de los datos de AprilTag, podrá utilizarlos para alinear el robot:
  - a. Hacer que el conductor coloque el robot cerca del objetivo y apuntando aproximadamente en la dirección correcta, de forma que esté seguro de que la etiqueta está a la vista.
  - b. Hacer que un conductor pulse y mantenga pulsado un botón para iniciar la orientación visual.
  - c. Mientras se mantiene pulsado el botón, controla la rotación de la transmisión utilizando la ubicación de la etiqueta. Probablemente puedas utilizar un control proporcional simple para acercarte lo suficiente. Compara la ubicación de la etiqueta con el centro de la imagen, multiplica el resultado por alguna constante que afines y aplica el resultado a la rotación de la transmisión (aplica el resultado como el valor de giro de la función de transmisión arcade o la mitad del valor en signos opuestos a cada lado de una función de transmisión tanque).
  - d. Permite al conductor mantener el control del componente de propulsión del tren motriz, ya sea continuando aplicando el valor del joystick al movimiento adelante/atrás de una función de conducción arcade, o añadiendo el valor del joystick al valor AprilTag para una conducción de tanque (para la conducción de tanque también tendrás que detectar si cualquiera de los dos valores excede 1 y escalar ambos lados hacia abajo si lo hace).
  - e. El conductor puede entonces conducir el robot hacia el objetivo mientras éste se mantiene apuntando a la etiqueta. Cuando los Bumpers alcanzan el objetivo, el centro del robot debe estar aproximadamente alineado con el centro del objetivo. Es posible que el robot aún no esté alineado a ras con la pared. Continúe empujando el robot hacia el objetivo y los Bumpers deberán cuadrarse contra la pared. También puede

soltar el botón de alineación y aplicar un pequeño giro en la dirección adecuada para facilitar esta parte del proceso.

- f. Si desea apuntar a algo que no sea el centro de la AprilTag (por ejemplo, para alinear de tal manera que 2 corales encajen uno al lado del otro en el Arrecife), se necesitará un poco más de complejidad.

O, para un uso más avanzado, puedes utilizar las AprilTags para obtener información completa sobre la posición de tu robot y alinearlos o planificar una trayectoria en cualquier parte del campo.

### 5.3 Próximos pasos

Si ha pensado en mejoras antes de construir su robot, decida si quiere volver a realizar una lluvia de ideas sobre ese tema para añadirlo a su grupo de iteraciones, o si tiene una lista de ideas sobrantes que reconsiderar. Si aún no ha considerado las mejoras, recopile toda la lista de ideas de ambos ejercicios y pase a la sección [6](#) de consejos sobre cómo priorizarlas.

## 6 Selección descendente - Convertir las ideas en un plan

Si estás pensando en realizar mejoras antes de construir tu KitBot, es posible que aún no tengas ninguna idea para la iteración. Realice este ejercicio sólo con las mejoras. Siempre puedes volver y hacerlo de nuevo con las ideas de mejora restantes más nuevas ideas de iteración después de haber construido y probado tu robot. Si ya ha construido su robot, le recomendamos que combine las ideas para nuevas capacidades con las ideas para la iteración en un solo ejercicio de selección descendente.

Convertir su lista de posibles mejoras en una lista priorizada sobre la que actuar es, sin duda, un ejercicio difícil que implica sopesar muchos factores. Hay muchas formas de sopesar los beneficios percibidos de cada cambio frente a la dificultad de aplicarlo y los recursos de equipo necesarios para llevarlo a cabo. A continuación se detalla un posible método.

### 6.1 Priorizar

Basándose en su análisis del juego y en su visión del papel de sus robots en las alianzas, priorice sus posibles mejoras de mayor a menor impacto. No te preocupes todavía por lo que es fácil y lo que es difícil, o qué utiliza qué recursos, simplemente crea una "lista de sueños" ordenada de mejoras del robot. A continuación se ofrece un ejemplo:

1. Nueva capacidad robótica
2. Auto cosa que ayuda con RP(Ranking Point)
3. Final del juego que ayuda con RP(Ranking Point)
4. Ajustes para mejorar la precisión de la puntuación
5. Hacer repuestos del componente X

### 6.2 Análisis - Estimación de recursos y alcance

En esta sección, va a examinar su lista y evaluar los recursos necesarios para ejecutar cada mejora. En lugar de asignar un solo concepto o número a la dificultad de una idea, prueba a crear un par de categorías diferentes que representen los distintos recursos que pueden ser necesarios. Algunas de estas categorías pueden representar limitaciones de recursos físicos, como el material disponible, la disponibilidad de máquinas o el presupuesto para adquirir nuevos componentes/material, y otras pueden representar recursos más abstractos, como el tiempo total/capacidad de los estudiantes y mentores en un aspecto específico del equipo.

Un posible conjunto de categorías es: Diseño, Fabricación, Software, Materiales/Coste. Para cada idea, evalúe el coste de los recursos para cada una de estas categorías y asigne un valor de 0 a 5 (siéntase libre de sustituirlo por la escala que desee), siendo 0 que no se requieren recursos y 5 que se requieren recursos muy importantes. Para la prioridad, utilice el rango que creó en [el Paso 6.1](#), del 1 al N (no pasa nada si están en una escala diferente a los valores de la categoría).

En el caso de un mecanismo nuevo, lo ideal sería poder hacer números aproximados sin necesidad de hacer una lluvia de ideas y seleccionar qué mecanismo específico se utilizaría para completar la tarea, pero para algunos equipos esto puede resultar difícil. Es posible que en este punto tenga que



desviarse para realizar una lluvia de ideas o prototipos de ideas de mecanismos específicos con el fin de evaluar sus necesidades de recursos; consulte la sección [6.5](#) de recursos adicionales.

En Cuadro 1, se ofrece un ejemplo agnóstico de juego y robot.

*Cuadro 1 : Ejemplo de recursos de mejora*

Mejora	Prioridad	Diseño	Fabricación	Software	Materiales/Costos
Nueva capacidad robótica	1	5	3	3	3
Auto cosa que ayuda con RP(Ranking Point)	2	0	1	3	0
Final del juego que ayuda con RP(Ranking Point)	3	3	2	1	1
Ajustes para mejorar la precisión de la puntuación	4	1	3	0	1
Hacer repuestos del componente X	5	0	2	0	2

### 6.3 Analizar - Fijar su "presupuesto"

Ahora que ha fijado los costos relativos de los recursos de sus ideas, necesita establecer un "presupuesto" que describa las capacidades de su equipo en cada categoría. Para cada una de las categorías, utilizando los costes asignados como referencia para la escala, considere los recursos de su equipo en el tiempo que queda hasta el evento. Asegúrese de tener en cuenta el tiempo que necesitará justo antes del acontecimiento para probar y practicar con cualquier mejora y empaquetar su equipo para el evento regional. Asigne a cada categoría un presupuesto global que represente los recursos de que dispone en esa categoría de aquí al próximo evento.

Como ejemplo, consideremos un equipo con 2 estudiantes de CAD y un mentor de CAD, sólo 1-2 estudiantes de software sin mentor de software, una empresa que ayuda con la fabricación de piezas además del equipo en el taller del equipo, y un importante stock de materias primas y piezas COTS. Este equipo podría describir su presupuesto para estas categorías como:

- Diseño - 5
- Programas informáticos - 3
- Fabricación - 10
- Materiales - 10

### 6.4 Hacer planes

Ahora que ya tienes un presupuesto y algunos costes, es hora de reunirlos en un plan sobre qué ideas llevar a cabo. Un enfoque consiste simplemente en ir bajando por la lista de prioridades hasta agotar

el presupuesto de cualquier categoría, y luego seguir bajando añadiendo los elementos que sólo afecten a las categorías con presupuesto restante.

Otro planteamiento más complejo consiste en pedir a individuos o a pequeños grupos que elaboren planes de muestra y luego los evalúen en equipo y seleccionen el que el grupo considere que dará como resultado el mejor robot. A veces, esto puede dar lugar a un uso más óptimo de los recursos.

Volvamos a nuestra tabla de ejemplo anterior con nuestro equipo teórico que ha establecido su presupuesto de recursos de diseño en 5 puntos (vamos a ignorar su presupuesto en otras categorías para simplificar). Con el primer enfoque, este equipo optaría por añadir una "nueva capacidad de lujo", agotando todo su presupuesto de diseño, y luego "hacer algunos repuestos del componente X".

*Cuadro2 : Ejemplo de plan de mejora - Opción 1*

Mejora	Prioridad	Diseño	Fabricación	Software	Materiales/Costos
Nueva capacidad robótica	1	5	3	3	3
Hacer repuestos del componente X	5	0	2	0	2

Este plan realiza dos mejoras y utiliza 5 puntos de recursos de fabricación, 3 de software y 5 de materiales.

Con el segundo enfoque, alguien puede sugerir que el equipo se salte esa primera mejora y reparta los puntos de su presupuesto de diseño entre las tres mejoras siguientes.

*Cuadro3 : Ejemplo de plan de mejora - Opción 2*

Mejora	Prioridad	Diseño	Fabricación	Software	Materiales/Costos
Auto cosa que ayuda con RP(Ranking Point)	2	1	1	3	0
Final del juego que ayuda con RP(Ranking Point)	3	3	2	1	1
Ajustes para mejorar la precisión de la puntuación	4	1	3	0	1
Hacer repuestos del componente X	5	0	2	0	2

Aunque este plan no incluye el elemento de mayor prioridad individual, introduce 4 mejoras en lugar de dos y puede aprovechar mejor los recursos de fabricación (8 frente a 5) y software (4 frente a 1) disponibles. El equipo tendría entonces que sopesar el valor de estas 4 mejoras frente a las 2 del otro plan y decidir qué enfoque producirá el mejor robot para cumplir los objetivos del equipo.

## 6.5 Próximos pasos

### 6.5.1 Mejoras - Nuevos mecanismos

Para añadir nuevos mecanismos, el siguiente paso es empezar a hacer una lluvia de ideas sobre posibles implementaciones y luego crear prototipos para determinar una dirección. Los siguientes recursos de la página [Recursos técnicos](#), en Recursos mecánicos->Generales, pueden ser un punto de partida útil:

- "Prototipos 101"
- "Hoja de trabajo de creación de prototipos"
- "Hoja de trabajo sobre mecanismos"
- "Guía de diseño robótico RAP de la NASA"

### 6.5.2 Mejoras - Capacidades de software

Para añadir nuevas funciones de software, consulte la "Documentación del sistema de control y programación", así como la "Programación 101" de la sección Software/Recursos eléctricos->Software de la página [Recursos técnicos](#).

### 6.5.3 Plan del proyecto

Lo último que probablemente quiera hacer antes de ponerse manos a la obra es esbozar un plan de proyecto aproximado. Puede ser tan sencillo o tan detallado como quieras, pero es probable que quieras al menos unos cuantos hitos entre ahora y tu próximo evento. Esto le ayudará a comprobar si va por buen camino y le permitirá añadir recursos (como tiempo de reunión) o reducir funciones antes de llegar a un punto en el que se presente en el evento con un robot inacabado o sin probar. Recortar una o dos funciones para asegurarte de que tienes tiempo para probarlas, y la práctica probablemente dará sus frutos al final.

## 7 Apéndice A - Ejemplo de análisis del juego básico, 2023

### 7.1 Ejemplo - Lista de tareas del robot

A continuación se muestra un ejemplo de lista de tareas del robot (ver sección [4.1.1](#)) para el juego 2023 *FIRST* Robotics Competition CHARGED<sup>UPSM</sup> presentado por Haas. Esta lista se ha confeccionado a posteriori; es posible que cada equipo haya incluido más o menos tareas en su propia lista.

Formas de marcar para un robot:

- Cono alto
- Cono medio
- Cono bajo
- Cubo alto
- Cubo medio
- Cubo bajo
- Puente de equilibrio

Mayor desglose de la puntuación:

- Puente de equilibrio
  - Equilibrio conduciendo sobre el puente desde el borde ancho
  - Equilibrio desde el lado estrecho de un puente ya equilibrado

Formas de adquirir piezas de juego:

- Subestación doble
  - Cubos
  - Conos con punta
  - Conos verticales
- Subestación única directa
  - Cubos
  - Conos con punta
  - Conos verticales
- Suelo
  - Cubos
  - Conos con punta
    - Punto primero
    - Brida primero
  - Conos verticales

Adquirir las piezas del juego desde el lado opuesto del robot desde donde se marcan puede acelerar los recorridos, especialmente importante en Auto (puede importar menos para los trenes motrices que pueden girar mientras se mueven, como mecanum o swerve).

Otras tareas:

Rev. 4 de Enero de 2025

Página 20

- ¿Dar fichas de juego a los compañeros?

Tareas defensivas:

- ¿Adquisición de piezas de juego del oponente? (Capacidad de adquisición sin extensión para recoger legalmente piezas de la ZONA DE CARGA del adversario)

## 7.2 Ejemplo - Cómo posicionarse bien

A continuación se muestra un ejemplo de cómo clasificar alto en el 2023 *FIRST* Robotics Competition juego CHARGED UP presentado por Haas.

1. Aparte de ganar partidos, ¿cómo pueden las alianzas ganar puntos de clasificación?
  - a. RP(Ranking Point) de enlace: consigue al menos 5 enlaces o al menos 4 enlaces con Co-Op (esto se refiere al valor original de inicio, no al valor modificado del campeonato).
  - b. Estación de carga RP(Ranking Point) - Consigue al menos 26 puntos de estación de carga
2. Para cada uno de estos RP, ¿qué capacidades robóticas se necesitan para
  - a. maximizar la capacidad de nuestras alianzas para ganar el RP(Ranking Point) ?
    - i. Anota 15 piezas de juego en total mezcladas entre ambos tipos en al menos la fila inferior y media para crear 5 Enlaces
    - ii. Equilibra la Estación de Carga en Auto para que sólo se necesite un balance de 2 Robots en el final del juego para asegurar el RP(Ranking Point).
  - b. ¿Conseguir el RP(Ranking Point) si se juega con otros 2 robots como el nuestro?
    - i. Anota ~5 piezas de juego totales mezcladas entre ambos tipos en al menos la fila inferior y media para ayudar a crear 5 Enlaces
    - ii. Equilibra la Estación de Carga en Auto para que sólo se necesite un balance de 2 Robots en el final del juego para asegurar el RP(Ranking Point).
  - c. ¿Hacer la contribución mínima al RP(Ranking Point) para nuestra alianza?
    - i. Anota ~3-5 piezas de juego en total en al menos la fila inferior para contribuir a los Enlaces.
    - ii. Ayuda a equilibrar la estación de carga en la partida final para contribuir a los puntos de la estación de carga.
3. ¿Cuáles son el <sup>primer</sup> y el <sup>segundo</sup> criterio de desempate después de RP(Ranking Point) (normalmente es poco probable que haya un empate más allá de esto y muy poco probable que tal empate decida más de un rango o dos como máximo)? No deberían ser una influencia importante en la selección de capacidades, pero pueden suponer un pequeño empujón en caso de decisión difícil entre dos capacidades a la hora de priorizar.
  - a. Puntos de recarga
  - b. Puntos automáticos

## 7.3 Ejemplo - Cómo ganar partidos

A continuación se muestra un ejemplo de análisis de puntos/tiempo para el partido de 2023. Para simplificar un poco, este análisis sólo tiene en cuenta el periodo Teleop e ignora el Autónomo. También asume que las piezas de juego que puntúan están contribuyendo a los Enlaces, lo que puede

no ser una suposición válida si se elige una estrategia de sólo Cubo. Las estimaciones de tiempo y la tasa de éxito utilizadas son aproximaciones y representan a un robot que intenta llegar a las eliminatorias (~50° percentil) en un evento regional o de distrito medio.

*Cuadro4 : Ejemplo de análisis de puntos a lo largo del tiempo*

<b>Acción puntuable</b>	<b>Puntos</b>	<b>Tiempo</b>	<b>Tasa de éxito</b>	<b>Pts/Segundo</b>
Cubo bajo	2 + 5/3 (Enlace)	22	95	.158
Cono bajo	2 + 5/3 (Enlace)	30	95	.116
Cubo central	3 + 5/3 (Enlace)	27	90	.122
Cono central	3 + 5/3 (Enlace)	35	85	.088
Cubo alto	5 + 5/3 (Link)	30	85	.160
Cono alto	5 + 5/3 (Link)	38	80	.119
Equilibrio 1 Robot	10	15	90	.6
Equilibrio 2 Robots	20	20	85	.85
Equilibrio 3 Robots	30	25	80	.96

A continuación, algunas respuestas aproximadas a las preguntas sobre las alianzas ganadoras.

1. ¿Cómo es el partido para una alianza ganadora media en Calificaciones?

3x Movilidad = 9 puntos  
 Auto acoplado y activado = 10 puntos  
 3x Auto piezas de juego = +3 Puntos  
 2x Enlaces Bajos =  $2*5 + 6*2 = 22$  Puntos  
 2x Enlaces Altos =  $2*5 + 6*5 = 40$  Puntos  
 Robots acoplados y enganchados x2 = 20 Puntos  
 Parque = 2 Puntos  
 Total = 106 puntos

2. ¿Qué aspecto tiene el partido para una alianza que ganaría alrededor del 80% de los partidos de clasificación?

3x Movilidad = 9 puntosAuto  
 Acoplado y Comprometido = 10 puntos4x  
 Auto piezas de juego = +4 Puntos3x

Enlaces Bajos =  $3 \cdot 5 + 9 \cdot 2 = 33$  Puntos $2x$

Enlaces Altos =  $2 \cdot 5 + 6 \cdot 5 = 40$

PuntosAcoplado y Comprometido x2 Robots = 20 PuntosAparcamiento

= 2 PuntosTotal

= 118 Puntos

3. ¿Cómo es el partido para una alianza que puede ganar un partido de desempate?

3x Movilidad = 9 puntosAuto

Acoplado y Comprometido = 10 puntos $4x$

Auto piezas de juego = +4 Puntos $3x$

Enlaces Bajos =  $3 \cdot 5 + 9 \cdot 2 = 33$  Puntos $2x$

Enlaces Altos =  $2 \cdot 5 + 6 \cdot 5 = 40$

PuntosAcoplado y Comprometido x2 Robots = 20 PuntosAparcamiento

= 2 PuntosTotal

= 118 Puntos

4. ¿Cómo es el partido para una alianza que gane el evento?

3x Movilidad = 9 puntos

Auto acoplado y activado = 10 puntos

5x Auto piezas de juego = +5 Puntos

3x Enlaces Bajos =  $3 \cdot 5 + 9 \cdot 2 = 33$  Puntos

1x Enlace Medio =  $5 + 3 \cdot 3 = 14$  Puntos

3x Enlaces altos =  $3 \cdot 5 + 9 \cdot 5 = 60$  Puntos

Robots Atracados y Comprometidos x3 = 30 Puntos

Parque = 2 Puntos

Total = 162 puntos

Por último, utilizando como objetivo un equipo que intenta llegar a los playoffs, algunas respuestas aproximadas a lo que el equipo debería intentar aportar:

1. ¿Cuántos puntos pretende aportar a la alianza?  
~30-40 puntos
2. ¿Hay tareas específicas en estas alianzas a las que crea que debe contribuir?  
Acoplar y acoplar en Auto
3. ¿Hay tareas específicas que quiera dejar para que contribuyan los socios de la alianza?  
No

## 7.4 Ejemplo - Mejoras

Supongamos que el KitBot 2023 se diseñó para suministrar sólo Low Cubes. Teniendo en cuenta el total de puntos y las tareas identificadas anteriormente, la adición de una rutina de acoplamiento y conexión autónoma permitiría al robot cumplir con las tareas y los totales de puntos identificados. Si nos fijamos en la tabla de puntos/tiempo, la siguiente capacidad a considerar sería probablemente la de los cubos intermedios. Esto es casi tan eficiente como los cubos bajos (y podría ser mejor si se puede hacer tan rápido como Cubos Bajos) y le da al equipo 3 oportunidades más de anotar.

Unoficial