

# 2024 **FIRST**<sup>®</sup> Robotics Competition

## Guía de Mejora/Iteración del KitBot

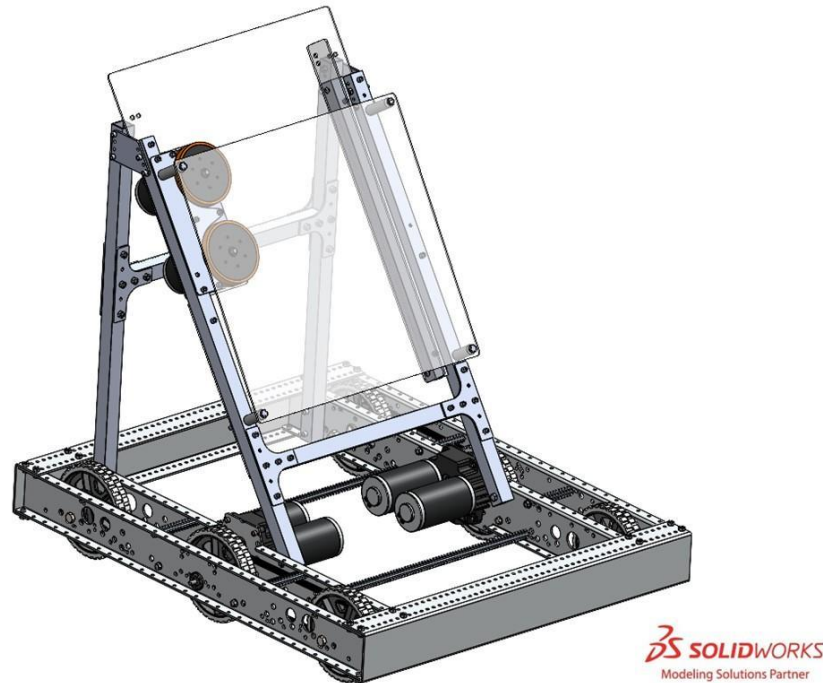
## Tabla de contenido

1	Introducción.....	3
2	Cuando comenzar a pensar sobre cambios .....	4
3	Mejora – Agregando Nuevas Capacidades.....	5
3.1	Análisis del Juego .....	5
3.2	Identificando Mejoras .....	7
3.3	Próximos pasos.....	7
4	Iteración – Mejorando la Capacidad Existente .....	8
4.1	Pruebas con Propósito.....	8
4.2	Idear Mejoras.....	8
4.3	Próximos Pasos.....	14
5	Seleccionando Prioridades - Convirtiendo Ideas en un Plan.....	15
5.1	Priorización .....	15
5.2	Análisis - Estimación de Recursos y Alcance .....	15
5.3	Análisis - Establece tu "presupuesto".....	16
5.4	Elaboración de Planes.....	17
5.5	Próximos Pasos.....	18
6	Apéndice A – Ejemplo de Análisis Básico del Juego, 2023.....	19
6.1	Ejemplo - Lista de Tareas del Robot.....	19
6.2	Ejemplo - Cómo Clasificar Alto.....	20
6.3	Ejemplo - Cómo Ganar Partidos .....	21
6.4	Ejemplo - Mejoras .....	23

# 1 Introduccion

---

Figura 1: 2024 KitBot



El KitBot para CRESCENDO<sup>SM</sup> presentado por Haas es capaz de realizar las siguientes acciones. Algunas acciones requerirán que el equipo agregue código que sean posible (por ejemplo, el código autonomo):

- Manejar por el campo (excepto debajo del Stage) usando una transmisión diferencial (también conocida comúnmente como "tanque") diseñada para una velocidad máxima alcanzable de ~15 pies por segundo (~4,5 m/s).
- Pre-load una Note para el uso autónomo
- Anotar puntos de salir
- Anotar Notes al Speaker
- Conseguir Notes del Source
- Jugar defensa

Este es un conjunto de capacidades bastante básico con respecto a todas las tareas posibles del juego. El KitBot fue diseñado para mantener las cosas muy simples, lo que significa que puede haber oportunidades para iterar y mejorar las capacidades existentes que tiene. Los equipos pueden escoger agarrar piezas del juego del piso, escalar en el stage o más!

## 2 Cuando comenzar a pensar sobre cambios

---

Ya sea antes o después de construir tu KitBot, puede que quieras revisar el juego y determinar si hay capacidades adicionales que te gustaría intentar agregar al robot de tu equipo. El KitBot está diseñado pensando en adiciones futuras, y animamos a la mayoría de los equipos a armarlo y experimentar con él un poco antes de modificarlo o agregarle algo, pero hay beneficios potenciales al considerar adiciones antes de la construcción. Estos son algunos intercambios que pueden ayudarte a decidir si realizar este primer ejercicio antes o después de haber construido el KitBot.

### Antes de Construir el KitBot

- Ventajas:
  - Permite a los miembros del equipo de diseño trabajar de inmediato en lugar de tener que esperar hasta que la construcción esté completa.
  - Es más fácil hacer cambios en las partes del diseño base antes de que estén ensambladas.
- Desventajas:
  - Puede que estés diseñando más robot del que puedes construir con éxito.
  - Puede que estés haciendo cambios en partes del diseño base que no entiendes completamente.

### Después de Construir el KitBot

- Ventajas:
  - Certeza de que tienes un robot que puede jugar algunos aspectos del juego.
  - Mejor capacidad para priorizar la iteración (mejorar capacidades existentes) frente a la mejora (agregar nuevas capacidades).
  - Mejor comprensión de la función del diseño base y qué puede y qué no puede cambiar.
  - Las pruebas y la práctica pueden ocurrir simultáneamente con el trabajo en mejoras.
  - Las mejoras en el desarrollo de software pueden probarse inmediatamente.
- Desventajas:
  - Nada que diseñar hasta que la construcción esté completa.
  - Cambiar las partes base puede requerir desmontar cosas.
  - Puede que prefieras usar tuercas y tornillos en lugar de remaches para desmontar las cosas más fácilmente.

## 3 Mejora – Agregando Nuevas Capacidades

---

Este ejercicio se puede completar antes o después de construir el KitBot (ver sección 2 para los pros y contras de cada opción). Si completas este ejercicio después de construir el KitBot, recomendamos completar las partes de lluvia de ideas de todos los ejercicios de Mejora e Iteración (Secciones 3 y 4) y luego combinarlas con la parte de selección de la sección 5 para priorizar qué cambios perseguirá tu equipo.

Aunque no es estrictamente necesario, tener un conjunto de objetivos del equipo es muy útil para la parte de estrechar el enfoque de este ejercicio. Para más información sobre la definición de objetivos del equipo, consulta el video de "Establecimiento de Objetivos" enlazado desde la página de Recursos de Gestión del Equipo bajo la sección de Organización del Equipo.

Los ejercicios y técnicas presentados aquí comprenden una versión abreviada de un análisis completo del juego y un proceso de estrategia de robots. Para un recurso más completo sobre este proceso, consulta la sección de Diseño Estratégico de la presentación "Estrategias Efectivas de FIRST® para Diseño y Competición" en la página de Recursos Técnicos bajo la sección Otros Recursos Técnicos->Exploración/Estrategia.

### 3.1 Análisis del Juego

#### 3.1.1 Análisis del juego – ¿Qué más podría hacer nuestro robot?

El primer paso para idear posibles mejoras es analizar el juego y elaborar una lista de posibles cosas que el robot podría hacer. A continuación se proporciona un conjunto abreviado de preguntas para hacer esto. Para un proceso más completo, revisa la "Hoja de Trabajo del Kickoff" en la página de Recursos Técnicos bajo la sección Otros Recursos Técnicos->Kickoff. Esto es diferente a solo una lista de tareas del juego (aunque es un buen punto de partida). No empieces a pensar en cómo el robot realizará estas tareas todavía. A continuación, se presenta una lista de preguntas que pueden guiar a tu equipo en la elaboración de esta lista:

1. ¿Cuáles son las formas de anotar puntos en el juego?
2. ¿Pueden desglosarse alguno de estos elementos (por ejemplo, anotar 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 puedan ayudar a tu alianza (por ejemplo, manipular elementos del campo, formas de ayudar a los compañeros, etc.)?
5. ¿Hay formas de ralentizar el anotar puntos de los oponentes que requieran capacidades únicas del robot (es decir, no solo defendiendo usando el chasis del robot)?

¿Buscas un ejemplo? Consulta el Apéndice A - Ejemplo de Análisis Básico del Juego, 2023 para una posible lista de tareas para el juego de 2023.

### 3.1.2 Análisis del juego - ¿Cómo clasificamos alto?

Si los objetivos de tu equipo involucran intentar clasificar lo más alto posible (por ejemplo, intentar ser capitán de alianza, maximizar Puntos del Distrito para calificar para un Campeonato de Distrito, etc.), es importante revisar los criterios por los cuales se clasifican los equipos. Si los objetivos de tu equipo no involucran clasificar alto (por ejemplo, ser elegido para una alianza, diseñar un mecanismo interesante, ganar un premio, etc.), puedes saltarte esta sección. Los criterios de clasificación se encuentran típicamente en la sección del Torneo del manual y usualmente se basan en Ranking Points (RP), obtenidos por victorias o empates y por completar objetivos específicos del juego. Asumiendo un juego típico de FIRST<sup>®</sup> Robotics Competition con clasificación en este estilo, hazte estas preguntas:

1. Aparte de ganar partidos, ¿cómo pueden las alianzas ganar Puntos de Clasificación?
2. Para cada uno de estos RP, ¿qué capacidades del robot son necesarias para:
  - a. Maximizar nuestra habilidad para que nuestras alianzas ganen el RP?
  - b. Lograr el RP si juegas con otros 2 robots como el nuestro?
  - c. Hacer la contribución mínima hacia el RP para nuestra alianza?
3. ¿Cuáles son los 1.º y 2.º desempates después de los RP (es típicamente improbable tener un empate más allá de esto y muy poco probable que tal empate decida más que un rango o dos como máximo)? Estos no deberían ser una influencia mayor en la selección de capacidades, pero pueden proporcionar un pequeño empujón en caso de una decisión difícil entre dos capacidades al priorizar.

### 3.1.3 Análisis del juego - ¿Ganar partidos?

Si los objetivos de tu equipo involucran ganar partidos, necesitarás dedicar tiempo a mirar el juego y pensar en cómo se ve una alianza ganadora. Dependiendo de tus objetivos exactos, algunas de estas preguntas de muestra pueden ser más o menos relevantes.

1. Enumera todas las tareas de puntuación en el juego con sus valores de puntos.
  - a. ¿Cuántas veces puede una alianza completar cada una de estas tareas? Esto puede no ser una respuesta única y clara, algunos juegos pueden tener tareas que son mutuamente exclusivas.
2. Para cada tarea, estima qué tan rápido crees que se puede lograr.
  - a. Las respuestas a esto variarán altamente basadas en la fortaleza del robot, probablemente querrás elegir una "clase" particular de robot y responder consistentemente con ese nivel en mente. Para puntuación que requiere una pieza de juego, recuerda incluir el tiempo para ir a buscar esa pieza de juego y regresar al lugar de puntuación.
3. Para algunas tareas, también podrías querer considerar una "tasa de éxito". Esto es qué porcentaje del tiempo crees que el robot completará la tarea en el tiempo que has estimado.
  - a. Para una tarea de lanzamiento esto podría ser obvio, qué tan a menudo la pieza de juego entrará en la meta. Para otras tareas puede ser menos obvio, pero intenta considerar con qué frecuencia podrías tener que re-alinearte e intentar de nuevo, por

ejemplo. Expresa la respuesta como un decimal o fracción (por ejemplo, tasa de éxito del 80% = 0.8).

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

Basado en la lista de tiempos y eficiencia que elaboraste, evalúa lo siguiente (salta cualquier cosa que no sea apropiada para los objetivos de tu equipo):

1. ¿Cómo se ve el partido para una alianza ganadora promedio en Clasificaciones?
2. ¿Cómo se ve el partido para una alianza que ganaría aproximadamente el 80% de los partidos de Clasificación?
3. ¿Cómo se ve el partido para una alianza que puede ganar un Partido de Playoff?
4. ¿Cómo se ve el partido para una alianza que gana el evento?

Usando los objetivos de tu equipo y una evaluación de tus capacidades, estima tu rol en cada una de estas alianzas:

1. ¿Cuántos puntos estás tratando de contribuir a la alianza?
2. ¿Hay tareas específicas en estas alianzas en las que crees que necesitas contribuir?
3. ¿Hay tareas específicas que quieras dejar para que los compañeros de alianza contribuyan?

## 3.2 Identificando Mejoras

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

## 3.3 Próximos pasos

Si estás considerando mejoras antes de construir tu KitBot, salta a la Sección 5 para dar algunas ideas sobre cómo convertir estas ideas en un plan. Si ya has construido tu KitBot, procede a la Sección 4 para información sobre cómo probar tu robot con propósito e identificar otras mejoras potenciales para incluir en tu proceso de planificación.

## 4 Iteración – Mejorando la Capacidad Existente

---

La iteración es un componente central del proceso de diseño de ingeniería, y diseñar un robot para FIRST Robotics Competition no es diferente. Una vez que hayas construido tu KitBot, querrás usar cualquier tiempo restante antes o entre tus eventos para hacer dos cosas: practicar e iterar. Para obtener más información sobre la selección de conductores y la práctica, consulta la "Guía para Seleccionar Conductores" y "Mejorando el Rendimiento del Conductor" en la página de Recursos Técnicos bajo la sección Otros Recursos Técnicos.

A continuación se describe un proceso simple para la iteración, que consiste en los siguientes pasos:

- Prueba
- Idear
- Planear
- Implementar

### 4.1 Pruebas con Propósito

El primer paso en un proceso de iteración es identificar áreas para mejorar. Para tu KitBot, recomendamos hacer esto mediante pruebas con propósito. La mayoría de estas pruebas implicarán usar tu robot como esperas hacerlo durante un partido, lo que también servirá como práctica para los conductores. La única excepción es la prueba hasta el fallo. Si tienes suficiente tiempo (recomendamos al menos una semana) antes de tu evento, puedes probar intencionalmente tu robot en situaciones comprometedoras, como chocar agresivamente, adquirir demasiadas piezas de juego, presionar botones en un orden incorrecto, etc., para ver qué se rompe. Mientras pruebas el robot, escribe respuestas a las siguientes preguntas:

1. ¿Qué se rompió? ¿Cómo se rompió?
2. ¿Qué está haciendo el robot de manera inconsistente (por ejemplo, fallar al adquirir piezas de juego, dejar caer piezas de juego, perder oportunidades de puntuación, etc.)?
3. ¿Qué tareas les están tomando mucho tiempo a los conductores para completar (por ejemplo, adquirir piezas de juego, cruzar el campo, alinearse para anotar, liberar la pieza de juego, etc.)?

Si tienes más de una pieza de juego disponible, es posible que quieras hacer una comparación cuidadosa del rendimiento del robot con cada pieza de juego para ver si hay alguna variación. Luego, toma una o dos piezas de juego y cámbialas de maneras que esperas que sucedan durante el juego (sobre infladas o desinfladas, distorsionadas, desgastadas o rotas, etc.). Repite las pruebas del robot y documenta cualquier cambio en el comportamiento.

### 4.2 Idear Mejoras

Ahora que has identificado posibles áreas de mejora, el siguiente paso es generar ideas para mejorar esas debilidades. Este documento divide la lluvia de ideas en dos partes: mecánica y software/eléctrica.



Esto se hace para ayudar a hacer el proceso un poco más manejable y porque estos elementos a menudo requieren diferentes conjuntos de recursos, frecuentemente se pueden perseguir en paralelo.

#### 4.2.1 Lluvia de Ideas para Mejoras - Mecánicas

Para cada una de las áreas de mejora potencial identificadas en tus pruebas, idea algunos cambios mecánicos potenciales que podrían mejorar el rendimiento. Algunas preguntas de estímulo para cada una de las tres categorías se incluyen a continuación:

##### 4.2.1.1 ¿Qué se rompió?

1. ¿Hay formas en que podríamos cambiar la geometría de la parte que se rompió para hacerla más fuerte?
2. ¿Hay materiales alternativos que podríamos usar que fortalezcan la parte con la misma geometría?
  - a. Recuerda que más fuerte no siempre significa más rígido, a veces hacer una parte de un material más flexible (por ejemplo, cambiar una parte de aluminio por policarbonato) puede permitirle absorber energía sin alcanzar deformación permanente.
3. ¿Realmente queremos que la parte sea más fuerte, o queremos hacer repuestos?
  - a. Si la causa de la falla parece anormal y puede suceder con poca frecuencia en un torneo, si es que sucede, es posible que quieras dejar la parte tal como está, o incluso a veces hacerla un poco más débil. Aunque esto pueda parecer contraintuitivo al principio, diseñar un eslabón débil conocido y estar preparado para reemplazarlo si ocurre un daño puede ayudar a prevenir daños inesperados en partes más caras o más difíciles de reemplazar. Los pernos de corte mecánico y los fusibles eléctricos son algunos ejemplos de tales partes sacrificables.

##### 4.2.1.2 ¿Qué es inconsistente?

1. ¿Hay formas en que podríamos ajustar la geometría de la parte para tratar de mejorar la consistencia?
  - a. Las ideas para esto a menudo incluyen agregar o quitar restricciones o guías, aumentar o disminuir la compresión en una pieza de juego, eliminar huecos o grietas con las que una pieza de juego pueda estar interactuando inadvertidamente, o agregar ruedas o rodillos adicionales conectados a un sistema existente.
2. ¿Hay formas en que podríamos cambiar los materiales para tratar de mejorar la consistencia?
  - a. Las ideas para esto suelen buscar aumentar o reducir la flexibilidad o fricción.
3. ¿Hay formas en que agregar actuación adicional podría aumentar la consistencia?
  - a. Ejemplos de esto incluyen expandir activamente un mecanismo para adquirir piezas de juego, agregar pestillos o compuertas adicionales en una trayectoria de movimiento de la pieza de juego, agregar actuación adicional para ayudar a asegurar la orientación de la pieza de juego en un sistema, etc.

#### 4.2.1.3 ¿Qué lleva mucho tiempo?

1. ¿Hay guías mecánicas que se puedan agregar para acelerar la tarea del conductor?
  - a. Ejemplos incluyen cuñas o embudos que interactúan con elementos del campo, o un marcador físico en un robot que ayude al conductor a alinear mejor con un elemento del campo.
2. ¿Hay formas de hacer al robot más robusto ante desalineaciones?
  - a. Ejemplos incluyen aumentar el tamaño de apertura para la adquisición de piezas de juego o cambiar la geometría de anotación para funcionar desde diferentes alineaciones.
3. ¿El problema es cuánto tiempo tarda el mecanismo en completar la acción? ¿Se puede acelerar cambiando el engranaje?
  - a. Si planeas acelerar un mecanismo, asegúrate de verificar cuánta corriente está consumiendo en su configuración actual. La corriente escalará aproximadamente linealmente con el engranaje, por lo que si acelerar el mecanismo empujará la corriente a niveles preocupantes, es posible que necesites agregar otro motor también.

#### 4.2.1.4 Ideas Específicas para el KitBot 2024

A continuación se presentan algunas sugerencias específicas para posibles mejoras mecánicas para el KitBot 2024. Omite esta sección si prefieres apegarte a las ideas que generes tú mismo.

1. **Agregar blindaje para prevenir posesión inadvertida de Notas** - El KitBot tiene lugares donde una Nota podría quedar atrapada inadvertidamente (debido a un alineamiento incorrecto al adquirir, rebote en el Altavoz, etc.) lo que contaría como tu única Nota permitida para controlar. Se podría agregar blindaje en estos lugares para reducir esta probabilidad.
2. **Cerrar la brecha de adquisición de anillo** - Para mantener las cosas simples, el KitBot adquiere Notas 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. Un mecanismo actuado podría usarse para llenar este hueco después del inicio del partido, extendiendo la rampa sobre los bumpers. Probablemente quieras detenerte en o ligeramente antes del borde del bumper para permitir que el bumper continúe absorbiendo cualquier impacto con la pared.
3. **Reducir el tiempo de aceleración del lanzador**. -El lanzador en el KitBot necesita un segundo para acelerarse antes de estar listo para lanzar. Podrías reducir este tiempo aumentando la potencia del motor aplicada a la rueda lanzadora cambiando a un motor más potente o agregando un segundo motor para alimentar la rueda. También podrías mitigar esto con un cambio de software cambiando los controles para permitir que tu conductor acelere mientras el robot todavía se está posicionando y luego lance solo cuando esté completamente listo.

## 4.2.2 Lluvia de Ideas para Mejoras – Eléctricas/Software

Las mejoras eléctricas y de software se agrupan porque a menudo van de la mano. Hacer mejoras de software en un robot a menudo requiere la entrada de sensores para permitir que el software reaccione a lo que está sucediendo. Al igual que en la lluvia de ideas mecánicas, revisa cada una de tus áreas de mejora potencial y piensa en cómo la retroalimentación de software o la automatización podrían proporcionar mejoras.

### 4.2.2.1 ¿Qué se rompió?

1. ¿Podría haberse evitado la falla con límites de viaje para un mecanismo?
  - a. Estos límites suelen implementarse utilizando interruptores de límite para denotar los bordes del recorrido permitido. Una alternativa es usar "límites suaves" para mecanismos que ya tienen retroalimentación de posición absoluta (es decir, sabes que tu mecanismo está a 30 grados, por lo que puedes limitarlo con software para que no viaje más en esa dirección).
2. ¿Podría haberse evitado la falla con límites de corriente?
  - a. Los límites de corriente pueden ayudar a reducir la fuerza máxima que un mecanismo puede emitir y pueden limitar el sobrecalentamiento del motor si un motor se estanca debido a un atasco inesperado del mecanismo.
3. ¿Podría haberse evitado la falla con un bloqueo de software?
  - a. A veces, los robots tienen mecanismos que solo deben activarse cuando el robot está en un cierto estado; de lo contrario, algo podría romperse. Estas exclusiones se pueden hacer cumplir en el software para ayudar a prevenir que los conductores cometan un error y dañen el robot.

### 4.2.2.2 ¿Qué es inconsistente?

1. ¿Son los controles intuitivos y obvios?
  - a. Si los conductores cometen errores, podría significar que necesitan más práctica, pero también podría significar que los controles no son intuitivos. Una vez que hayas seleccionado a tus conductores, asegúrate de trabajar con ellos para asegurar que los controles tengan sentido para ellos. Lo que puede ser obvio para una persona puede ser extraño para otra. Un mal diseño de control también puede incluir usar un solo botón para alternar un estado de ida y vuelta que los conductores no pueden ver.
2. ¿Hay problemas de sincronización que la automatización de software podría ayudar a prevenir?
  - a. A veces la inconsistencia proviene de que un conductor haga algo en el momento equivocado. Automatizar esta sincronización con software puede ayudar eliminando la inconsistencia humana del proceso. Un ejemplo de esto es el lanzador de Notas. Para que funcione eficazmente, la rueda delantera tiene que girar un poco antes de que la rueda trasera empuje la Nota hacia adelante. En lugar de que los conductores intenten cronometrar esto, el software proporcionado maneja esta sincronización

automáticamente controlando toda la secuencia con una sola pulsación y mantenimiento de un botón.

3. ¿Hay secuencias que podrían ser manejadas automáticamente por el software?
  - a. A veces, para realizar una acción, un robot tiene que activar múltiples mecanismos en un orden específico, con o sin un tiempo específico. ¿Se puede utilizar la automatización de software para hacer cumplir este orden o automatizar toda la secuencia para prevenir errores?
4. ¿Hay bloqueos de software que podrían ayudar a prevenir problemas?
  - a. La inconsistencia a veces proviene de realizar una acción cuando el robot no estaba en el estado correcto (no alineado con el campo, mecanismo de interacción no preparado adecuadamente, etc.). ¿Se puede usar el software para detectar cualquiera de estos estados incorrectos y evitar que el conductor realice la acción?
5. ¿El comportamiento del robot parece inconsistente incluso con lo que parece ser acciones correctas del conductor?
  - a. A veces esta inconsistencia proviene de cambios en el robot o el entorno, como un voltaje de batería diferente o un poco más de fricción a medida que una parte se desgasta. Agregar sensores y controles de software puede ayudar a asegurar que el robot haga lo mismo cada vez.

#### 4.2.2.3 *¿Qué lleva mucho tiempo?*

1. ¿Se puede automatizar la alineación?
  - a. La forma más común es usar sensores de visión para detectar la ubicación del robot en el campo y usar eso para navegar un robot a una ubicación deseada, pero dependiendo de la tarea, otros sensores como telémetros o sensores de haz de ruptura también pueden ser útiles.
2. ¿Puedes dar más información a los conductores?
  - a. Los conductores pueden hacer lo mejor que pueden usando sus ojos para ver lo que está sucediendo, pero el robot también puede ayudar enviando señales adicionales. Transmitir una cámara a la estación del conductor, mostrar estados del robot en el tablero, usar luces en el robot para señalar el estado y agregar retroalimentación del controlador (si corresponde) son ejemplos de formas de proporcionar información sobre el estado del robot a los conductores.

#### 4.2.2.4 *Ideas Específicas para el KitBot 2024*

A continuación se presentan algunas sugerencias específicas para la automatización de software potencial para el KitBot 2024. Omita esta sección si prefiere apegarse a las ideas que generó usted mismo.

1. **Control de lanzador de bucle cerrado:** –Cambiando el motor que alimenta la rueda lanzadora frontal por un motor con un codificador incorporado (Venom, NEO/Vortex, Falcon, Kraken) o agregando un codificador directamente o engranado del motor existente, puedes cambiar del

control de bucle abierto actual (estimando la velocidad de la rueda basada en el voltaje y tiempo proporcionados) a control de bucle cerrado (controlando precisamente la velocidad de la rueda basada en retroalimentación). Para obtener información de velocidad de tu sensor elegido, necesitarás consultar la documentación del motor o sensor apropiado. Para usar esta información para controlar la velocidad de la rueda, consulta la sección de Introducción a Controles Avanzados de la documentación de WPILib, especialmente el artículo sobre "Ajustar un Controlador de Velocidad de Volante".

2. **Detección de Notas:** –Se podría agregar un interruptor, sensor de proximidad o sensor de haz de ruptura al robot para detectar cuándo una Nota está en el robot. Podrías usar esta información para detener automáticamente el funcionamiento de las ruedas hacia adentro para reducir el calentamiento del motor y podrías proporcionar retroalimentación al conductor a través de luces, tablero o vibración del controlador (si usas un controlador con vibración como un controlador de Xbox). Esto también puede reducir la probabilidad de adquirir múltiples Notas al mismo tiempo.
3. **Alineación visual:** – El KitBot se puede alinear fácilmente con el Altavoz debido a la gran apertura, la proximidad a tus conductores y la capacidad de correr los Parachoques contra el Subwoofer para alinear la profundidad. Alinear con la Fuente puede ser más difícil debido a la distancia de tus conductores y el ancho más estrecho de las ranuras en la Fuente en comparación con el Altavoz. Agregar una cámara para procesar AprilTags se puede usar para alinear con cualquiera de las dos ranuras exteriores en la Fuente. Puedes hacer esto agregando una cámara web conectada al roboRIO o agregando un sistema de visión externo como un Limelight, RaspberryPi con WPILibPi o una placa de coprocesador con PhotonVision. Una vez que tengas datos del AprilTag, puedes usarlos para alinear el robot con la Fuente siguiendo estos pasos:
  - a. Posicionar el robot cerca de la Fuente y apuntar aproximadamente en la dirección correcta para asegurarte de que la etiqueta esté a la vista.
  - b. Mantener presionado un botón para iniciar el objetivo de visión.
  - c. Mientras se mantiene presionado el botón, controlar la rotación del tren de conducción usando la ubicación de la etiqueta. Puedes usar un control proporcional simple para acercarte lo suficiente. Compara la ubicación de la etiqueta con el centro de la imagen, multiplica el resultado por una constante que ajustes y aplica el resultado a la rotación del tren de conducción (aplica el resultado como el valor de giro de la función de conducción de arcade o la mitad del valor en signos opuestos a cada lado de una función de conducción de tanque).
  - d. Permitir que el conductor mantenga el control de la componente de propulsión del tren de conducción aplicando el valor del joystick al movimiento hacia adelante/atrás de una función de conducción de arcade, o sumando el valor del joystick al valor de AprilTag para una conducción de tanque (para la conducción de tanque también tendrás que detectar si alguno de los valores supera 1 y escalar ambos lados hacia abajo si es necesario).
  - e. El conductor puede entonces conducir el robot hacia la Fuente mientras el robot se mantiene apuntando a la etiqueta. Cuando los Parachoques lleguen a la pared de la

Fuente, el centro del robot debería estar aproximadamente alineado con el centro de la ranura. El robot puede no estar alineado completamente contra la pared. Continuar empujando el robot hacia la pared de la Fuente y los Parachoques deberían cuadrarse contra la pared. También puedes potencialmente soltar el botón de alineación y aplicar un poco de giro en la dirección adecuada para facilitar esta parte del proceso.

Un proceso similar podría usarse para alinear con el Altavoz. O, para un uso más avanzado, podrías usar los AprilTags para obtener información completa de la posición de tu robot y alinear o planificar una ruta en cualquier lugar del campo.

### 4.3 Próximos Pasos

Si consideraste mejoras antes de construir tu robot, decide si deseas visitar la lluvia de ideas sobre ese tema para agregar a tu conjunto de iteraciones, o si tienes una lista de ideas sobrantes para reconsiderar. Si aún no has considerado mejoras, recopila tu lista completa de ideas de ambos ejercicios y procede a la Sección 5 para obtener consejos sobre cómo priorizarlas.

## 5 Seleccionando Prioridades - Convirtiendo Ideas en un Plan

Si estás considerando mejoras antes de construir tu KitBot, es posible que aún no tengas ideas para la iteración. Procede con este ejercicio solo con las mejoras. Siempre puedes volver y hacerlo de nuevo con las ideas de mejora restantes más las nuevas ideas de iteración después de haber construido y probado tu robot. Si ya has construido tu robot, recomendamos combinar las ideas para nuevas capacidades con las ideas de iteración en un único ejercicio de selección de prioridades.

Convertir tu lista de posibles mejoras en una lista priorizada para actuar es, sin duda, un ejercicio difícil que implica sopesar muchos factores. Hay muchas formas de abordar el equilibrio entre el beneficio percibido de cada cambio frente a la dificultad de implementarlo y qué recursos del equipo pueden ser necesarios para implementarlo. A continuación se detalla un método posible.

### 5.1 Priorización

Apoyándote en tu análisis del juego y en tu visión del papel de tus robots en las alianzas, prioriza tus posibles mejoras y mejoras desde las más impactantes hasta las menos impactantes. No te preocupes aún por lo que es fácil y lo que es difícil, o qué usa qué recursos, solo crea una lista ordenada "de sueños" de mejoras en el robot. A continuación se proporciona un ejemplo:

1. Nueva capacidad fantástica del robot
2. Cosa autónoma que ayuda con los Puntos de Clasificación (RP)
3. Cosa de juego final que ayuda con los RP
4. Ajuste para mejorar la precisión de puntuación
5. Hacer repuestos del componente X

### 5.2 Análisis - Estimación de Recursos y Alcance

En esta sección, vas a mirar tu lista y evaluar los recursos necesarios para ejecutar cada mejora. En lugar de asignar solo un concepto o número a la dificultad de una idea, intenta crear un par de categorías diferentes que representen los diversos recursos que pueden ser necesarios. Algunas de estas categorías pueden representar limitaciones de recursos físicos como material disponible, disponibilidad de máquinas o presupuesto para comprar nuevos componentes/materiales y algunas pueden representar recursos más abstractos como el tiempo/habilidad total de estudiantes y mentores en un aspecto específico del equipo.

Un posible conjunto de categorías es: Diseño, Fabricación, Software, Materiales/Costo. Para cada idea, evalúa el costo de recursos para cada una de estas categorías y asigna un valor de 0 a 5 (siéntete libre de sustituir la escala que desees) siendo 0 sin recursos necesarios y 5 un requerimiento de recursos muy significativo. Para la prioridad, usa el rango que creaste en el Paso 5.1, del 1 al N (está bien si estos están en una escala diferente a los valores de categoría).

Para un nuevo mecanismo, idealmente podrías idear números aproximados sin necesidad de idear y seleccionar qué mecanismo específico usarías para completar la tarea, pero para algunos equipos esto puede ser difícil. Es posible que necesites hacer un desvío en este punto para idear o prototipar ideas de mecanismos específicos para evaluar sus requisitos de recursos, consulta la Sección 5.5 para recursos adicionales.

Se proporciona un ejemplo agnóstico del juego y del robot en la Tabla 1.

*Tabla 1: Ejemplo de mejoras de recursos*

Mejora	Prioridad	Diseño	Manufactura	Software	Costo Material
Capacidad nueva del robot	1	5	3	3	3
Autónomo para RP	2	0	1	3	0
Fin de juego para RP	3	3	2	1	1
Mejorar precisa la puntuación	4	1	3	0	1
Hacer repuestos	5	0	2	0	2

### 5.3 Análisis - Establece tu "presupuesto"

Ahora que has establecido los costos relativos de recursos de tus ideas, necesitas establecer un "presupuesto" que describa las capacidades de tu equipo en cada categoría. Para cada una de tus categorías, utilizando tus costos asignados como referencia de escala, considera los recursos de tu equipo en el tiempo entre ahora y tu evento. Asegúrate de considerar el tiempo que necesitarás justo antes del evento para probar y practicar con cualquier mejora y empacar tu equipo para el evento. Asigna un presupuesto general para cada categoría que represente tus recursos disponibles en esa categoría entre ahora y tu próximo evento.

Como ejemplo, consideremos un equipo con 2 estudiantes de CAD y un mentor de CAD, solo 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 stock sustancial de materiales en bruto y partes COTS. Este equipo podría describir su presupuesto para estas categorías como:

- Diseño – 5
- Software – 3
- Fabricación – 10
- Materiales – 10



## 5.4 Elaboración de Planes

Ahora que tienes un presupuesto y algunos costos, es hora de juntarlos en un plan de qué ideas perseguir. Un enfoque es simplemente seguir la lista de prioridades hasta que el presupuesto en cualquier categoría se agote, y luego continuar añadiendo cualquier ítem que solo afecte a categorías con presupuesto restante.

Otro enfoque más complejo es que individuos o pequeños grupos elaboren planes de muestra y luego evaluar estos planes como equipo y seleccionar lo que el grupo cree que resultará en el mejor robot. Esto a veces puede resultar en un uso más óptimo de los recursos.

Revisemos nuestro ejemplo de la tabla 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 por simplicidad). Usando el primer enfoque, este equipo elegiría agregar una "Nueva capacidad fantástica", agotando todo su presupuesto de diseño, y luego "Hacer algunos repuestos del componente X".

*Tabla 2: Opción 1 de plan de mejora*

Mejora	Prioridad	Diseño	Manufactura	Software	Costo Material
Capacidad nueva del robot	1	5	3	3	3
Hacer repuestos	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.

Usando el segundo enfoque, alguien podría sugerir en cambio que el equipo se salte esa primera mejora y distribuya sus puntos de presupuesto de diseño entre las siguientes tres mejoras.

*Tabla 2: Opción 2 de plan de mejora*

Mejora	Prioridad	Diseño	Manufactura	Software	Costo Material
Autónomo para RP	2	1	1	3	0
Fin de juego para RP	3	3	2	1	1
Mejorar precisa la puntuación	4	1	3	0	1
Hacer repuestos	5	0	2	0	2

Aunque este plan no incluye el punto con la prioridad individual más alta, realiza 4 mejoras en lugar de dos y puede utilizar mejor los recursos de fabricación (8 vs 5) y software (4 vs 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 con los objetivos del equipo.

## 5.5 Próximos Pasos

### 5.5.1 Mejoras – Nuevos Mecanismos

Para agregar nuevos mecanismos, el siguiente paso es comenzar a idear posibles implementaciones y luego prototipar para determinar una dirección. Los siguientes recursos de la página de Recursos Técnicos, bajo Recursos Mecánicos->General, pueden ser un buen punto de partida:

- “Prototipado 101”
- “Hoja de Trabajo de Prototipado”
- “Hoja de Trabajo de Mecanismos”
- “Guía de Diseño de Robótica de NASA RAP”

### 5.5.2 Mejoras – Capacidades de Software

Para agregar nuevas capacidades de software, consulta la “Documentación del Sistema de Control y Programación” así como la “Programación 101” de la sección Recursos de Software/Electricidad->Software de la página de Recursos Técnicos.

### 5.5.3 Plan de Proyecto

Lo último que probablemente quieras hacer antes de comenzar con el trabajo es esbozar un plan de proyecto aproximado. Esto puede ser tan simple o detallado como desees, pero probablemente quieras al menos algunos hitos entre ahora y tu próximo evento. Estos te ayudarán a verificar si estás en el camino correcto y te permitirán agregar recursos (como tiempo de reunión) o reducir características antes de llegar a un punto en el que te presentes en el evento con un robot sin terminar o sin probar. Recortar una o dos características para asegurarte de tener tiempo para probar y practicar probablemente dará sus frutos al final.

## 6 Apéndice A – Ejemplo de Análisis Básico del Juego, 2023

---

### 6.1 Ejemplo - Lista de Tareas del Robot

A continuación se muestra un ejemplo de lista de tareas del robot (ver Sección 3.1.1) para el juego de la Competición de Robótica FIRST 2023 CHARGED UPSM presentado por Haas. Esta lista se hizo en retrospectiva, los equipos individuales pueden haber tenido más o menos tareas en sus propias listas.

Formas en que un Robot puede anotar:

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

Desglose adicional de la puntuación:

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

Formas de adquirir piezas de juego:

- Doble subestación
  - Cubos
  - Conos inclinados
  - Conos verticales
- Subestación simple directa
  - Cubos
  - Conos inclinados
  - Conos verticales
- Suelo
  - Cubos
  - Cubos inclinados
    - Punto primero
    - Borde primero
    - Conos verticales

Adquirir piezas de juego desde el lado opuesto del robot desde donde se anotan puede acelerar los caminos, especialmente importante en Auto (puede importar menos para trenes de conducción que pueden rotar mientras se mueven, como mecanum o swerve).

Otras tareas:

- ¿Alimentar piezas de juego a tu alianza?

Tareas defensivas:

- ¿Adquirir piezas de juego del oponente? (Capacidad para adquirir sin extensión para agarrar legalmente piezas de las ZONAS DE CARGA del oponente)

## 6.2 Ejemplo - Cómo Clasificar Alto

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

1. Aparte de Ganar Partidos, ¿cómo pueden las Alianzas ganar Puntos de Clasificación?
  - a. RP de Enlace: Anotar al menos 5 Enlaces o al menos 4 Enlaces con Cooperación (esto se refiere al valor original de Kickoff, no al valor modificado del Campeonato).
  - b. RP de Estación de Carga: Anotar al menos 26 puntos de estación de carga.
2. Para cada uno de estos RP, ¿qué capacidades de robot son necesarias para:
  - a. Maximizar nuestra capacidad para que nuestras alianzas ganen el RP?
    - i. Anotar 15 piezas de juego en total mezcladas entre ambos tipos en al menos la fila inferior y media para crear 5 Enlaces.
    - ii. Equilibrar la Estación de Carga en Auto para que solo sea necesario un equilibrio de 2 Robots en el juego final para asegurar el RP.
  - b. ¿Lograr el RP si juegas con otros 2 robots como el nuestro?
    - i. Anotar aproximadamente 5 piezas de juego en total mezcladas entre ambos tipos en al menos la fila inferior y media para ayudar a crear 5 Enlaces.
    - ii. Equilibrar la Estación de Carga en Auto para que solo sea necesario un equilibrio de 2 Robots en el juego final para asegurar el RP.
  - c. ¿Hacer la contribución mínima hacia el RP para nuestra alianza?
    - i. Anotar aproximadamente 3-5 piezas de juego en total en al menos la fila inferior para contribuir hacia los Enlaces.
    - ii. Ayudar a equilibrar la Estación de Carga en el Juego Final para contribuir hacia los puntos de Estación de Carga.
3. ¿Cuáles son los 1.º y 2.º desempates después de los RP (es típicamente improbable tener un empate más allá de esto y muy poco probable que tal empate decida más que un rango o dos como máximo)? Estos no deberían ser una influencia mayor en la selección de capacidades, pero pueden proporcionar un pequeño empuje en caso de una decisión difícil entre dos capacidades al priorizar.

- a. Puntos de Estación de Carga
- b. Puntos de Autonomo

### 6.3 Ejemplo - Cómo Ganar Partidos

A continuación se muestra un ejemplo de análisis de puntos/tiempo para el juego de 2023. Para simplificar ligeramente, este análisis observa solo el período de Teleop e ignora el Autónomo. También asume que anotar piezas de juego contribuye a los Enlaces, lo cual puede no ser una suposición válida si se elige una estrategia solo de Cubos. Las estimaciones de tiempo y la tasa de éxito utilizadas son aproximaciones y representan a un robot tratando de llegar a los playoffs (~percentil 50) en un evento regional o distrital promedio.

*Tabla 4: Ejemplo de puntos con tiempos*

Action de puntos	Puntos	Tiempo	Consistencia	Pts/Segundos
Cubo Bajo	2 + 5/3 (Link)	22	95	.158
Cono Bajo	2 + 5/3 (Link)	30	95	.116
Medio Cubo	3 + 5/3 (Link)	27	90	.122
Medio Cono	3 + 5/3 (Link)	35	85	.088
Alto Cubo	5 + 5/3 (Link)	30	85	.160
Alto Cono	5 + 5/3 (Link)	38	80	.119
Balance 1 Robot	10	15	90	.6
Balance 2 Robots	20	20	85	.85
Balance 3 Robots	30	25	80	.96

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

1. ¿Cómo se ve el partido para una alianza ganadora promedio en las Clasificaciones?
  - 3x Movilidad = 9 puntos
  - Auto Acoplado y Enganchado = 10 puntos
  - 3x piezas de juego en Auto = +3 puntos
  - 2x Enlaces Bajos = 25 + 62 = 22 puntos
  - 2x Enlaces Altos = 25 + 65 = 40 puntos
  - Acoplado y Enganchado x2 Robots = 20 puntos
  - Estacionamiento = 2 puntos
  - Total = 106 puntos

2. ¿Cómo se ve el partido para una alianza que ganaría aproximadamente el 80% de los partidos de Clasificación?

3x Movilidad = 9 puntos  
Auto Acoplado y Enganchado = 10 puntos  
4x piezas de juego en Auto = +4 puntos  
3x Enlaces Bajos = 35 + 92 = 33 puntos  
2x Enlaces Altos = 25 + 65 = 40 puntos  
Acoplado y Enganchado x2 Robots = 20 puntos  
Estacionamiento = 2 puntos  
Total = 118 puntos

3. ¿Cómo se ve el partido para una alianza que puede ganar un partido de Playoff?

3x Movilidad = 9 puntos  
Auto Acoplado y Enganchado = 10 puntos  
4x piezas de juego en Auto = +4 puntos  
3x Enlaces Bajos = 35 + 92 = 33 puntos  
2x Enlaces Altos = 25 + 65 = 40 puntos  
Acoplado y Enganchado x2 Robots = 20 puntos  
Estacionamiento = 2 puntos  
Total = 118 puntos

4. ¿Cómo se ve el partido para una alianza que gana el evento?

3x Movilidad = 9 puntos  
Auto Acoplado y Enganchado = 10 puntos  
5x piezas de juego en Auto = +5 puntos  
3x Enlaces Bajos = 35 + 92 = 33 puntos  
1x Enlace Medio = 5 + 3\*3 = 14 puntos  
3x Enlaces Altos = 35 + 95 = 60 puntos  
Acoplado y Enganchado x3 Robots = 30 puntos  
Estacionamiento = 2 puntos  
Total = 162 puntos

Finalmente, usando un equipo que intenta llegar a los playoffs como objetivo, algunas respuestas aproximadas a lo que el equipo debería intentar contribuir:

1. ¿Cuántos puntos estás tratando de contribuir a la alianza?  
~30-40 puntos
2. ¿Hay tareas específicas en estas alianzas en las que crees que necesitas contribuir?  
Acoplarse y engancharse en Auto
3. ¿Hay tareas específicas que quieras dejar para que los compañeros de alianza contribuyan?

No

## 6.4 Ejemplo - Mejoras

Supongamos que el KitBot 2023 fue diseñado solo para entregar Cubos Bajos. Mirando el total de puntos y las tareas identificadas anteriormente, agregar una rutina Autónoma de Acoplamiento y Enganche permitiría al robot cumplir con las tareas y totales de puntos identificados. Mirando la tabla de puntos/tiempo, la siguiente capacidad a considerar agregar probablemente serían los Cubos Medios. Esto es casi tan eficiente como los cubos bajos (y podría ser mejor si se puede hacer tan rápido como los Cubos Bajos) y brinda al equipo 3 oportunidades más de puntuación.