

---

# Welcome to Project and Portfolio III

---

# Who are we

---

Jason Hinders

Program Director:

[jhinders@fullsail.com](mailto:jhinders@fullsail.com)

Rebecca Leis

Department Chair:

[rleis@fullsail.com](mailto:rleis@fullsail.com)

Steve VanZandt

CD of AHI:

[svanzandt@fullsail.com](mailto:svanzandt@fullsail.com)

Rod Moye

CD of PP2:

[rmoye@fullsail.com](mailto:rmoye@fullsail.com)

John O'Leske

CD of PP3:

[joleske@fullsail.com](mailto:joleske@fullsail.com)

JohnOLeskeFS#4268

# Full Midterm Project Process

---

## Applied Human Computer Interaction I

- Pre Production
  - Design Document
  - Product Backlog
- AHI Topics
  - Nielsen's heuristics
  - Usability
  - UX

## Project and Portfolio II II

- Core Functionality
  - Critical game systems
  - Interface and UI creation
- First Use/Playable
  - Playable complete
  - Experience
  - Fun factor

## Project and Portfolio III III

- Alpha
  - Full Functionality
- Beta
  - Content complete
  - Balancing
- Finalizing
  - QA process
  - Presentation

---

# Midterm Project Grading

---

# Grading Scale: Team Evaluation

## Team Grade

- The total team accomplishments during the project as a whole.
- Sprint Planning: Planning out the milestones and workload amongst the team using the planning poker process

Midterm Project Templates


Sprint Backlog (Alpha)

15 cards

contain a completion screen  
(winning the game, achieving  
achievements, high score)


0/9

(#8) As a user I want, the game  
contain a loss condition and  
to the user when and how

 **As a user I want, the game to contain a loss condition and it is clear to the user when and how they lost** ×

in list [Sprint Backlog \(Alpha\)](#)


☐ Recurring Add #tags ▼ S/E & More ▼


 Description

Edit

Intent:

- Create loss condition and player feedback involved in it

SUGGESTED 

 Join

[Feedback](#)

ADD TO CARD

Butler

# Grading Scale: Team Evaluation

## Team Grade

- The total team accomplishments during the project as a whole.
- Sprint Review: Each milestone is graded on its own based on the sprint goals from that milestone
  - During Alpha are all the features completed?
  - During Beta did we replace all placeholders?

### ALPHA MILESTONE

#### GOALS

CRITERIA	EXCELLENT 100%	SATISFACTORY 80%	NEAR STANDARDS 60%	BELOW STANDARDS 40%	INSUFFICIENT 0%
All Features necessary to complete the game are completed and integrated 40 points	Is fully achieved	Is nearly complete Is partially achieved	Is worked on but not achieved	Little work is done on the task Is not achieved	Is completely missing
The game world is populated with at least 1 example of each game object (all game object types) 30 points	Is fully achieved	Is nearly complete Is partially achieved	Is worked on but not achieved	Little work is done on the task Is not achieved	Is completely missing
Overall game progress can be shown (e.g., multiple levels,	Is fully achieved	Is nearly complete Is partially achieved	Is worked on but not achieved	Little work is done on the task	Is completely missing

# Grading Scale: Personal Evaluation

## Personal Tasks

- Based on completion of the work that you commit to for the duration of the project.
  - Personal Tasks Completed
    - Completed all assigned userstories with a high level of quality and integrated into the build.
  - Team Practices
    - Updated task board tasks and completed peer reviews of completed userstories.

### ALPHA PERSONAL CONTRIBUTION

#### SPRINT PERSONAL EVALUATION

CRITERIA	EXCELLENT 100%	SATISFACTORY 80%	NEAR STANDARDS 60%	BELOW STANDARDS 40%	INSUFFICIENT 0%
<b>Personal Tasks Complete</b> Completes the expected amount of work toward the project while maintaining a high level of quality	Completes all assigned tasks while maintaining a high level of quality of work	Does not complete all tasks assigned or has significant issues in quality	Completes an insufficient amount of tasks on the project	Completes little to no tasks on the project	Does not work on the project
<b>Team Practices</b> Updates task-board tasks on the scrum board showing work progress and completes peer reviews on teammates' tasks	Keeps the scrum board accurate and clean Documents time spent on work Quickly assists in	Needs prompting to update the task board to show work and peer review teammates' work	Inconsistently updates tasks, documents time spent on tasks, and completes peer reviews of tasks	Does not update the task board timely enough for use	Never updates the task board



# Grading Scale: GPS

---

## Professionalism

- The faculty reserves the right to identify what is considered unprofessional and what constitutes a breach of the assignment.
- Common GPS problems include
  - Not complying with imperatives from assigned by CDs
  - Refusal to cooperate with the project methodologies
  - If you choose to come in and work during the other lecture or lab period and are loud or disruptive during the other class' lecture
  - Being off task during free work hours(on campus lectures)



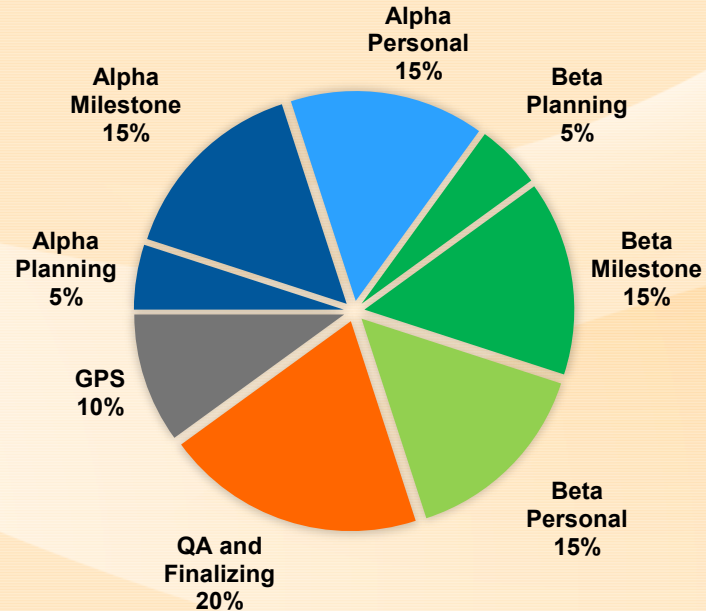


# Grading Scale: PP3 Weighting

---

## Overall Weighting

- Project Milestones (Team): 60%
- Personal tasks: 30%
- GPS: 10%



---

# Schedule

---

# Full Time

---

This is your only class for a reason

- (other than the career module)
- We are now full-time dev on this project
- You need to work on the project every day
  - In person is best
- Stay in contact with us and keep communication open

# Attendance

---

- Lecture and Lab on Monday and Wednesday
- Zoom meetings from 10am– 5pm
  - (With an hour break for lunch)
  - The 304B classroom or Blackmore are available for you on non class days
- Optional on Campus lab support
  - Mondays 3pm– 5pm CR- FS3G-102
  - Additional on campus support possible on request

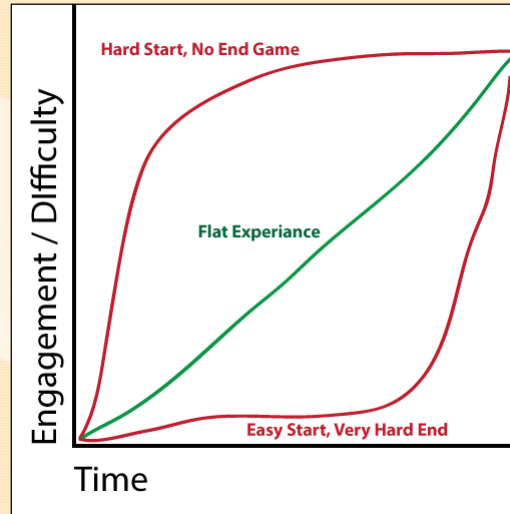
---

# IPM Documents and Difficulty Curves

---

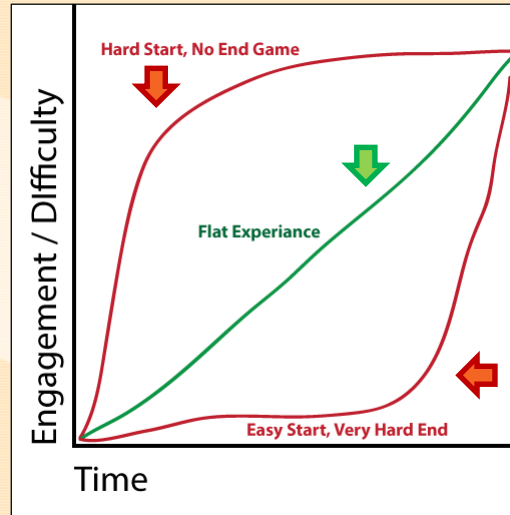
# Difficulty Curve

A plotting of the challenges the user faces while playing a game, against the time that that user has used the product



# Difficulty Curve: Issues

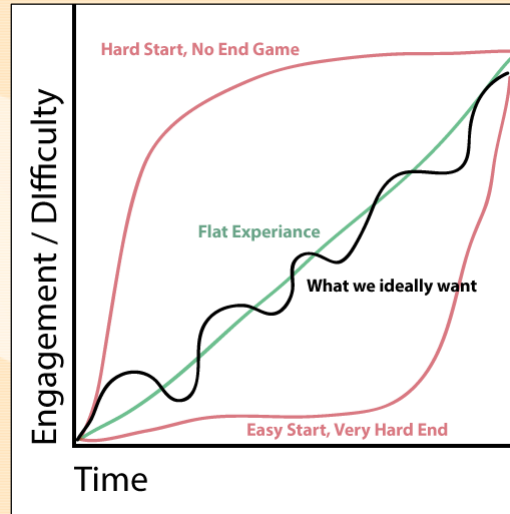
- Dropping everything on the user at once
- Little challenge with a big spike at end game
- Flat experience





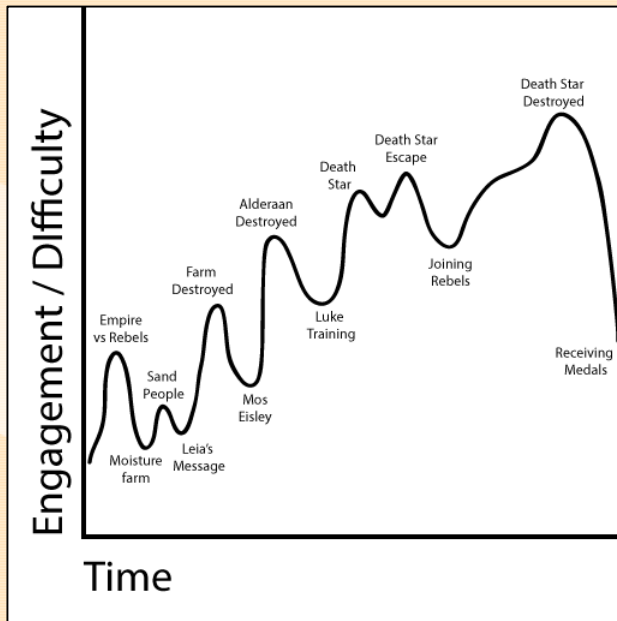
# Difficulty Curve: Ideal

- Feeling of power and control over the game's challenges
- Feeling of loss of power and control over the game's challenges
- Challenges spread throughout the product



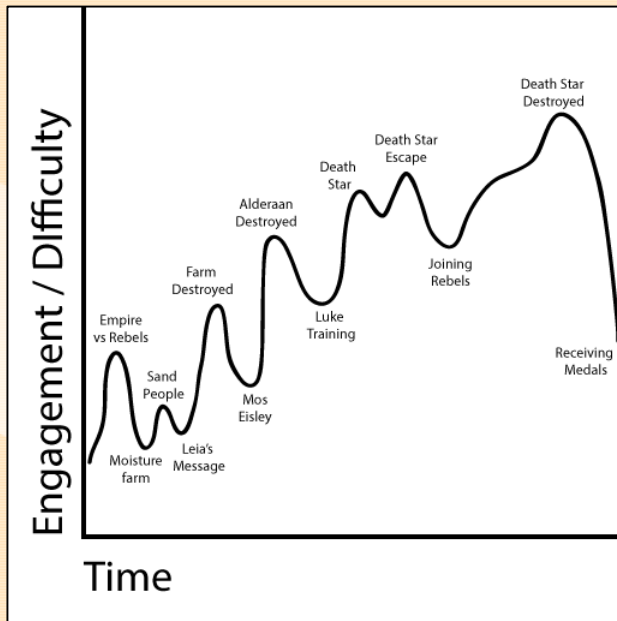
# Dramatic Arc

There is a similar concept to this in general entertainment based on viewer engagement called the dramatic arc



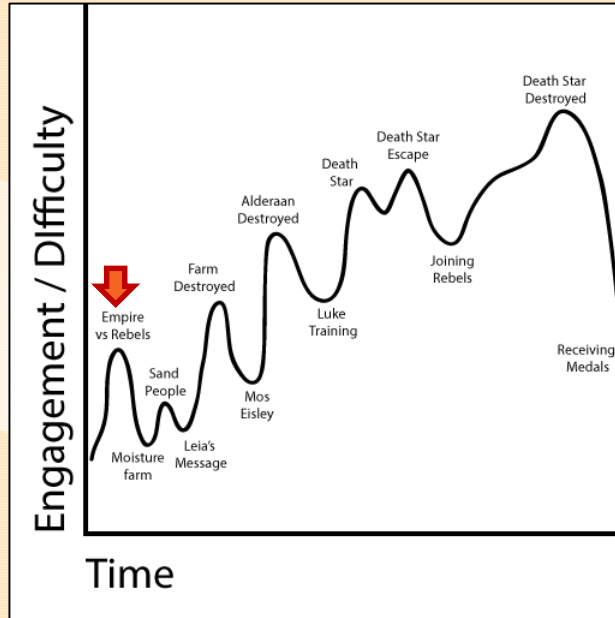
# Dramatic Arc

We don't want a flat line of experiences we want a series of peaks and valleys with few plateaus



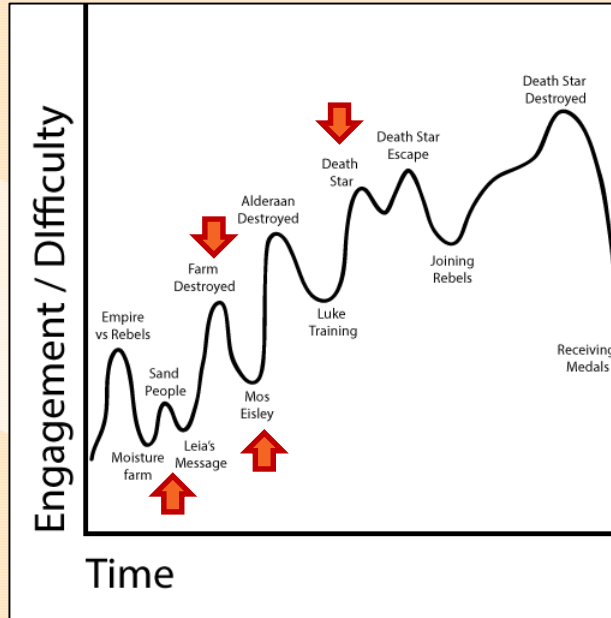
# Dramatic Arc

Start with something to hook the audience



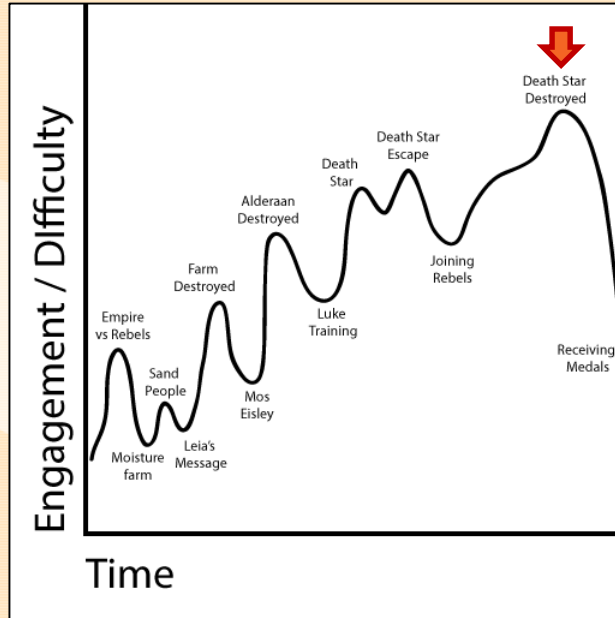
# Dramatic Arc

Continuous dipping and building back up to keep the audience engaged



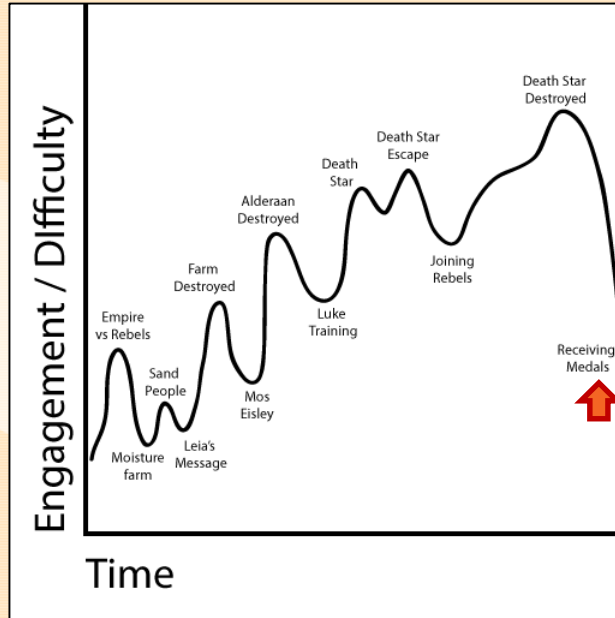
# Dramatic Arc

Reach the peak at the most opportune moment



# Dramatic Arc

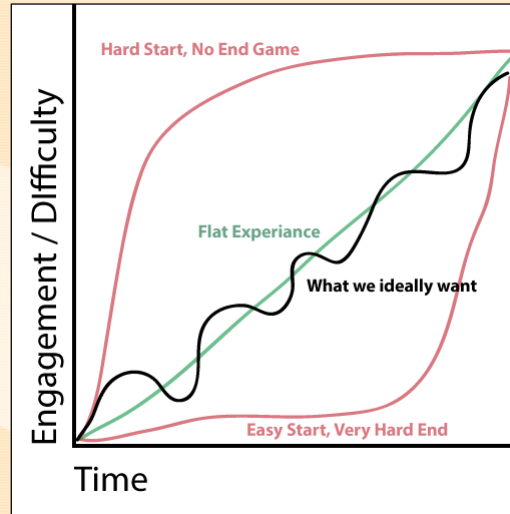
Allow the audience to come back down to wrap up the experience





# Difficulty Curve

This same ebb and flow is what we want with a difficulty curve as well.



# Difficulty Curve

---

You can't expect this curve to just appear and feel natural in the game automatically

It takes conscious thought and focus to make it happen

- Document it
- Find faults
- Fix them before wasting time in production

# Difficulty Curve

---

Three phases to be aware of

- Phases of increased difficulty (Introduction)
- Periods of normalcy (Practice)
- Dips in difficulty due to the player becoming acclimated (Mastery)

# Difficulty Curve: Example Items

---

Control mechanics or player abilities

- Where is it explained to the user?
- Where can they fumble and make mistakes without dire consequences?
- When do we expect the player to experiment with subtleties in the controls?
- When does that control become second nature to them?

# Difficulty Curve: Example Items

---

Types of enemy

- When does the player first encounter it?
- Where can the player acclimate themselves to the scenario's the enemies create?
- When does the player feel like they are in total control again?
- When does the player feel uninterested with an enemy?

# Difficulty Curve: Example Items

---


## Types of obstacles

- When does the player first encounter it?
- Where can the player acclimate themselves to the different ways the obstacle can function
- When does the player feel confident in traversing the obstacles?

# IPM Document

List out all points of interest and make that one axis of our chart.

- Mechanics
- Challenges
- Control schemes
- Enemy types
- Weapon types
- Item types
- ...
- Any potential stumbling point for the player




Point of Interest	Complexity (1-10)	Stage									
		1	2	3	4	5	6	7	8	9	10
Player Movement	1	I	P	M							
Precision Jumping	4		I			M					
Double Jump	2			I		M					
Fireball	2		I	M							
Dash	3				I	P	P	M			
Dash Attack	5						I	P	P	P	M
Enemy Dude	2	I	P	P	M						
Enemy Ninja	6		I	P	P	M					
Total		3	9	13	11	15	14	8	5	5	5



# IPM Document

The other axis lists out the time table of our users' experiences in whatever best fits our product

- Playable levels/stages
- Character level
- Time spent in game
- (Not all at once in a tutorial level)



Point of Interest	Complexity (1-10)	Stage									
		1	2	3	4	5	6	7	8	9	10
Player Movement	1	I	P	M							
Precision Jumping	4		I			M					
Double Jump	2			I		M					
Fireball	2		I	M							
Dash	3				I	P	P	M			
Dash Attack	5						I	P	P	P	M
Enemy Dude	2	I	P	P	M						
Enemy Ninja	6			I	P	P	M				
	Total	3	9	13	11	15	14	8	5	5	5

# IPM Document

Define, for each of points of interest,

- When they are introduced to the user
- When does the user put it in practice
- When is the user expected to have mastered it

Point of Interest	Complexity (1-10)	Stage									
		1	2	3	4	5	6	7	8	9	10
Player Movement	1	I	P	M							
Precision Jumping	4					M					
Double Jump	2					M					
Fireball	2		I	M							
Dash	3				I	P	P	M			
Dash Attack	5						I	P	P	P	M
Enemy Dude	2	I	P	P	M						
Enemy Ninja	6				I	P	M				
	Total			13	1	15	14	8	5	5	5

# How an IPM is used

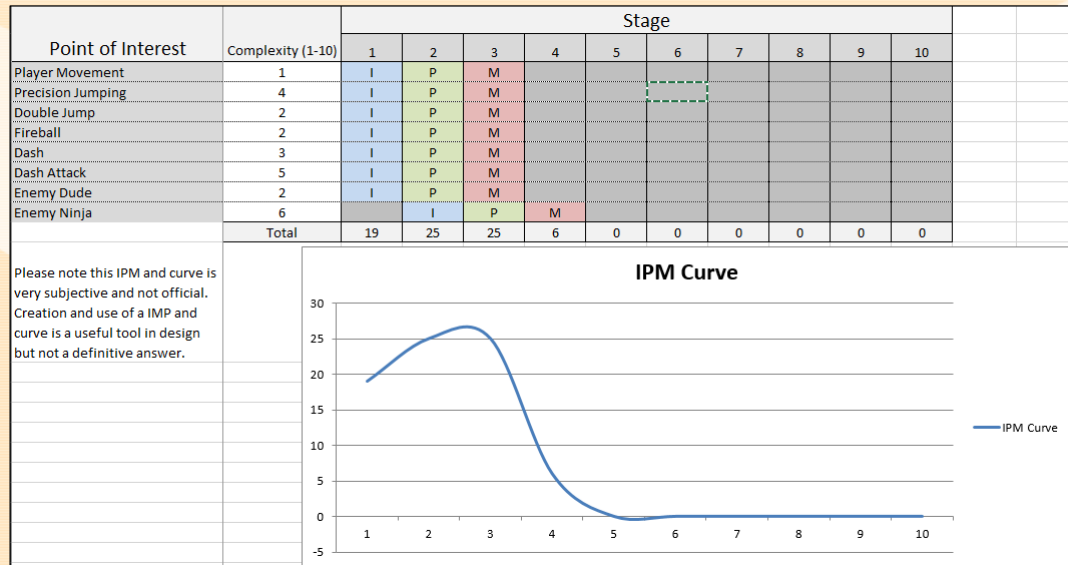
---

## Understanding the game

- The act of creating this documents helps teams understand the depth and breadth of the game's challenges

# How an IPM is used

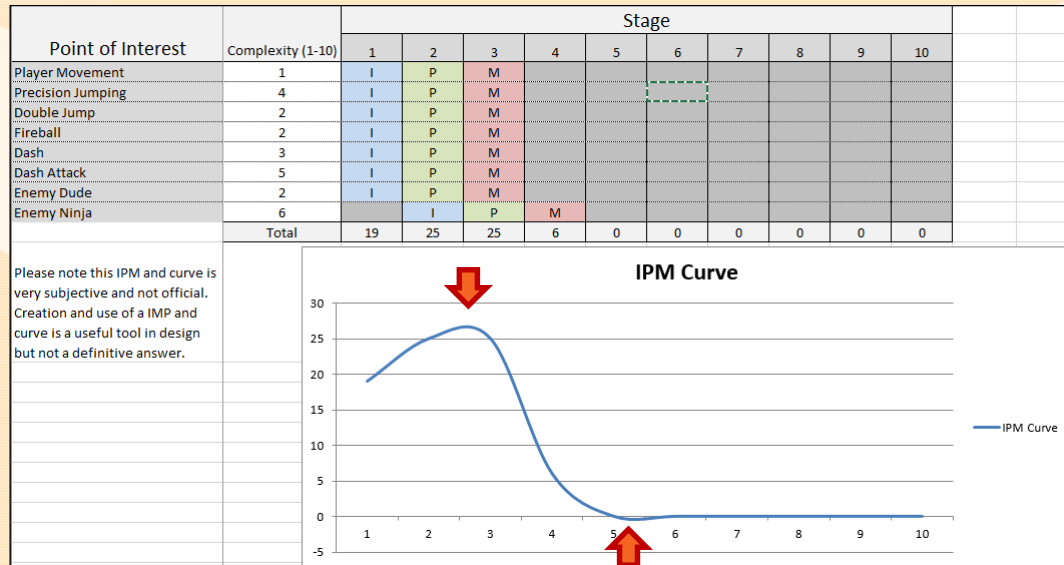
From the IPM matrix a graphical representation of our overall learning/engagement curve can be generated.



# How an IPM is used

## Discover hot spots

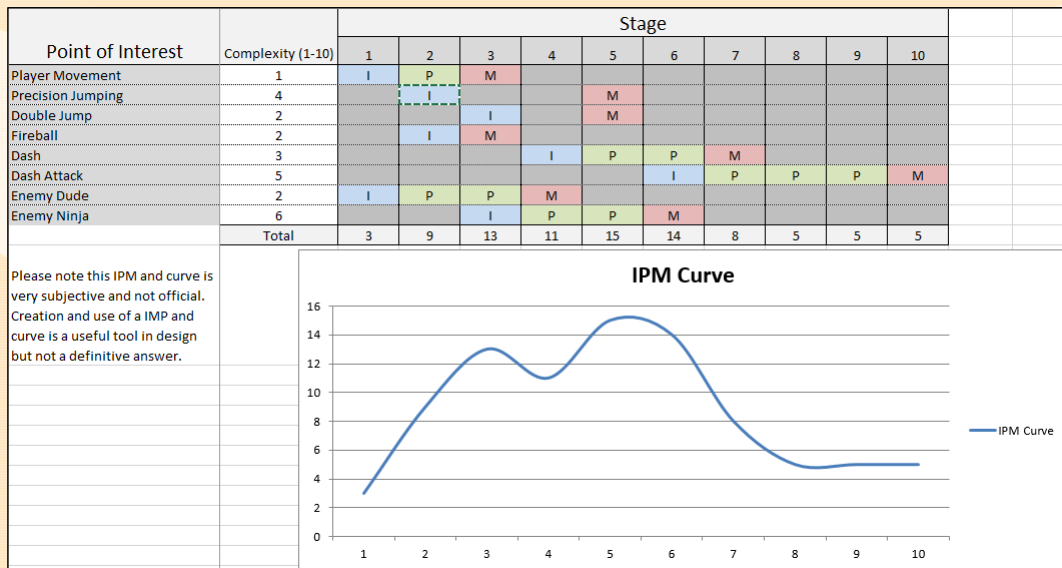
- With that curve we can see if there are any points of the curve that the slope is too aggressive or if at any point the game becomes stagnant.



# How an IPM is used

## Spread out challenge

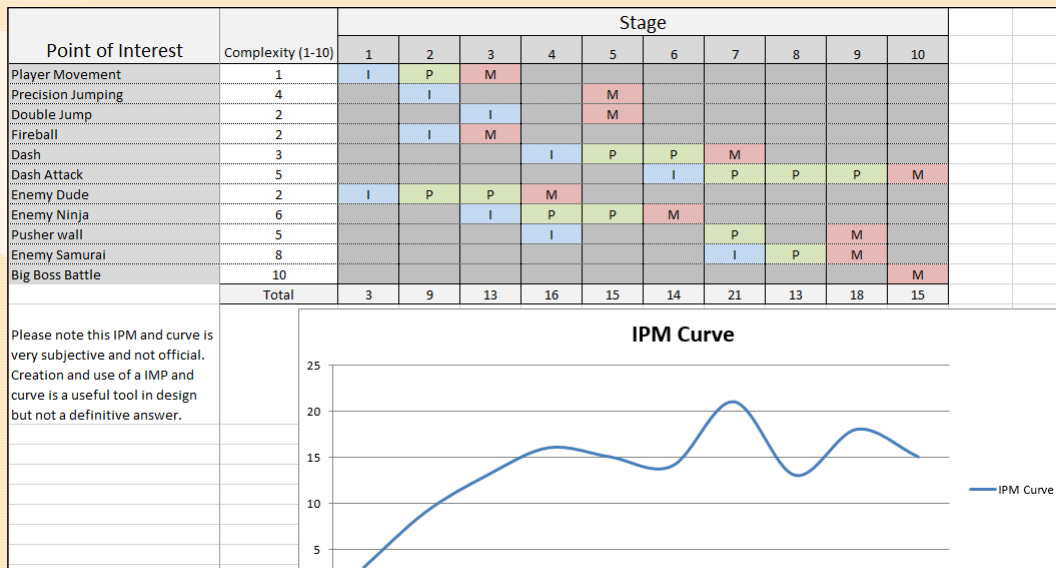
- The challenges can then be redistributed to make for a more even and engaging user experience



# How an IPM is used

Discover gaps in the overall game engagement

- Helps the team discover when there is a lack of content variety and push for more advanced gameplay





# How an IPM is used

## Use in level design

- A level designer uses the chart by looking down the column of the level they are going to create and know what features they are responsible for
- Breaks “blank canvas paralysis” as it pertains to level design

Point of Interest	Complexity (1-10)	Stages											
		1	2	3	4	5	6	7	8	9	10		
Player Movement	1	I	P	M									
Precision Jumping	4		I			M							
Double Jump	2			I		M							
Fireball	2		I	M									
Dash	3				I	P		M					
Dash Attack	5						I	P	P	P	M		
Enemy Dude	2	I	P	P	M								
Enemy Ninja	6			I	P	P	M						
Pusher wall	5				I			P		M			
Enemy Samurai	8							I	P	M			
Big Boss Battle	10										M		
	Total	3	9	13	16	15	14	21	13	18	15		


Please note this IPM and curve is very subjective and not official. Creation and use of a IMP and curve is a useful tool in design but not a definitive answer.

**IPM Curve**


# How an IPM is used

## Use in level design

- A level designer uses the chart by looking down at the level they are going to create and knows what features they are responsible for.
- Stage 6
  - Mastery of the enemy ninja
  - Practicing dash movement
  - Introduction to the dash attack
  - Other features available but not planned for use

Point of Interest	Complexity (1-10)	Stage 											
		1	2	3	4	5	6	7	8	9	10		
Player Movement	1	I	P	M									
Precision Jumping	4		I			M							
Double Jump	2			I		M							
Fireball	2		I	M									
Dash	3				I	P	P	M					
Dash Attack	5						I	P	P	P	M		
Enemy Dude	2	I	P	P	M								
Enemy Ninja	6			I	P	P	M						
Pusher wall	5				I			P		M			
Enemy Samurai	8							I	P	M			
Big Boss Battle	10										M		
Total		3	9	13	16	15	14	21	13	18	15		

Please note this IPM and curve is very subjective and not official.

 IPM Curve

# Part of Alpha

---

Create this document during alpha

- During alpha we will be finalizing our game's features
- We will create this document while those features are being finalized
- This document will help us pace it out where they will be used
- The plan being formulated wont be fully executed until the end of the beta milestone

---

Where are we now?

---

# <Activity> Playtesting

---

30 minutes to play each others games

- Watch for what people find fun
- Watch for what people find frustrating
- Is the experience what you wanted it to be?

---

# Sprint Process

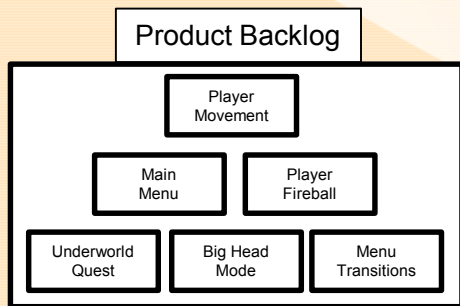
**REVIEW**

# Product backlog

---

Everything that could be in the product is collected into a list called the product back log

- Things can get added to the product back log as needed
- Only a wish list for now, Not promises that need to be fulfilled
- In userstory format

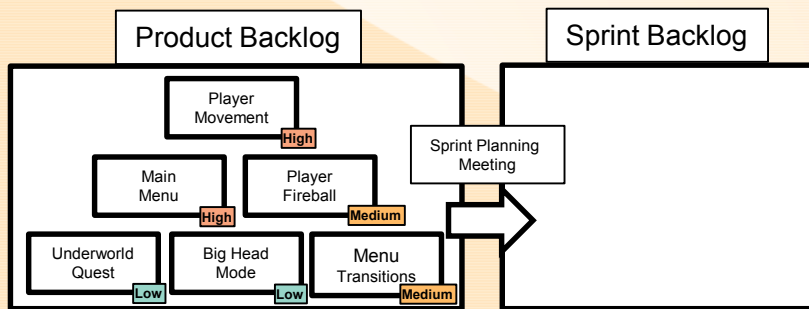


**REVIEW**

# Sprint Backlog

## Sprint Planning

- The entire team will meet to:
- First
  - Determine an overall sprint goal
  - Select stories from the product back log to achieve that goal



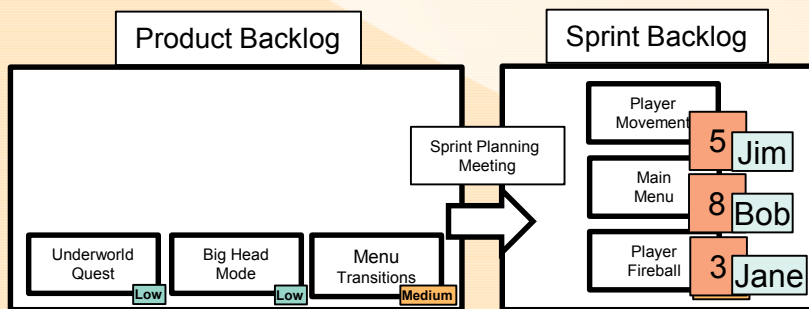
**REVIEW**



# Sprint Backlog

## Sprint Planning

- The entire team will meet to:
- Second
  - Evaluate the difficulty/hours/complexity of the stories selected
  - Distributing the workload among the team



**REVIEW**

# Planning poker

---

After the userstories have been selected each userstory is evaluated individually by the group.

- Estimating workload
- Understanding dependencies
- Assigning tasks

**REVIEW**

# Planning poker: Step 1: Bidding

---

## Step 1: Bidding

- Userstory and test cases is read out to the team
  - Answers questions if there are any
  - Modify test cases where needed
    - (Client is involved in this for externally produced projects)
- Each team member
  - Evaluates how difficult they believe the story is to completing, without bias from other members
  - Pick which of the possible bids best represents how difficult they evaluate the task to be
  - Done with cards placed face down in person
  - Done with web tools when remote ([planitpoker.com](https://planitpoker.com))

**REVIEW**

# Planning poker: Step 1: Bidding

Bid Value : Estimated Work

- 0 hr
- ½ hr
- 1 hr
- 2 hrs
- 3 hrs
- 5 hrs
- 8 hrs
- 13 hrs (1 day and a half)
- 20 hrs (half a week)
- 40 hrs (1 week)
- 100 hrs (2 weeks)
- Unknown
- Infinite

- The number pattern reflects one of the faults in making estimates
- The larger the estimate the more room for error



**REVIEW**

# Planning poker: Step 1: Bidding

Bid Value : Estimated Work

- 0 hr
- ½ hr
- 1 hr
- 2 hrs
- 3 hrs
- 5 hrs
- 8 hrs
- 13 hrs (1 day and a half)
- 20 hrs (half a week)
- 40 hrs (1 week)
- 100 hrs (2 weeks)
- Unknown
- Infinite

- Each value should be through as a range from the bid below it up
  - Can I get this done in 3? No. Can I get this done in 5?...

1	2	3	4	5	6	7	8	9	10	11	12	13
1	2	3	5		8			13				

**REVIEW**

# Planning poker: Step 1: Bidding

---

## Avoid Bias

- This first step (bidding) must be done in a vacuum
  - Allows everyone to think about the story
  - Gives people a place to defend and forces them to make their estimate for a reason
- Helps avoid “group think”
  - Bob thinks it is X so I guess it must be X

**REVIEW**

# Planning poker: Step 2: Negotiation

---

## Step 2: Negotiation

- Each team member reveals what bid they decided upon on the previous step at the same time
- If bids differ the team must discuss why and come to an agreement on the task's value

**REVIEW**

# Planning poker: Step 3: Allocation

---

## Step 3: Allocation

- After every user story has agreed upon values, user stories must have owners committed to them.
- The story's owner will be the person
  - Best equipped to tackle the story
  - Responsible for completing all task related to the story before the end of the sprint

**REVIEW**



# Planning poker: Step 3: Allocation

---

## Balance the workload

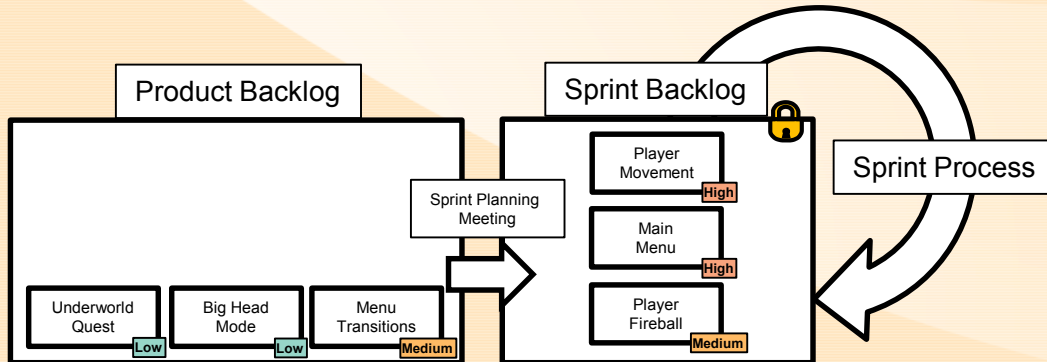
- Make sure each team member is contributing equally
  - Redistribute stories if they are not
- Make sure the workload matches up with the sprint length
  - Not enough hours to fill the schedule = take more stories from the product backlog
  - Over hours = Discuss with the product owner to return things to pull back on the sprint goal

**REVIEW**

# Sprint Backlog

## Sprint Planning

- Once the sprint planning is completed and the sprint has started a commitment has been made for those tasks
- Neither the product owner nor the developers should change a sprint plan once in motion

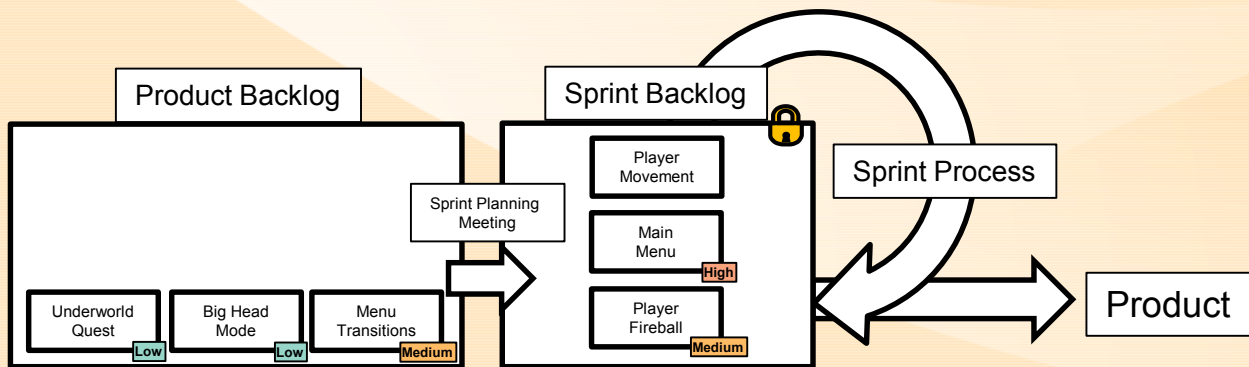


**REVIEW**

# Sprint Process

Teams then work through the sprint to complete the agreed upon tasks

- Completing the tasks
- Integrating into the master build



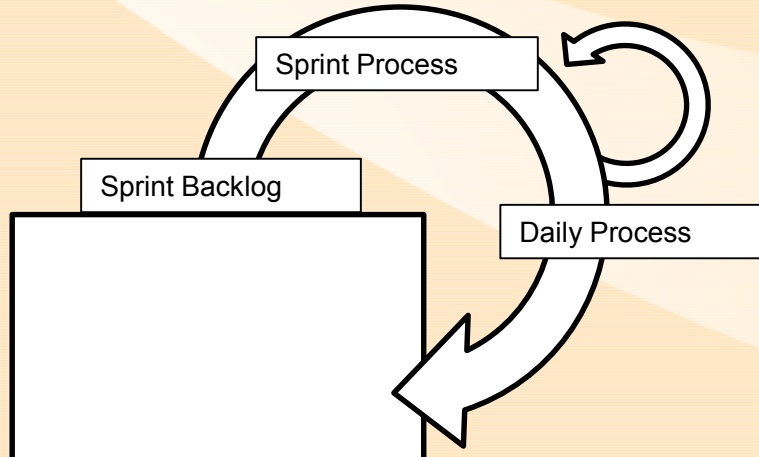
**REVIEW**

# Workday in scrum

---

Teams meet every day for a scrum “stand up” meeting

- Maintain transparency
- Hold each other accountable
- Set up help when needed



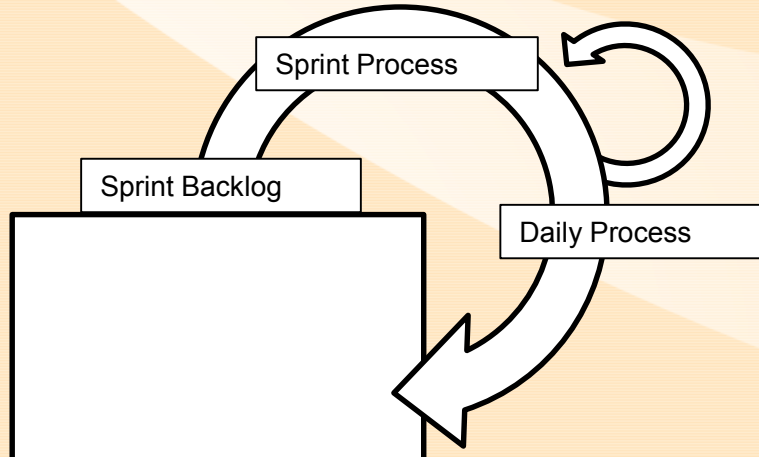
**REVIEW**

# Workday in scrum

---

Key points of scrum “stand up” meetings

- The meeting should be the start of our working day
- Maximum of 15 minutes.



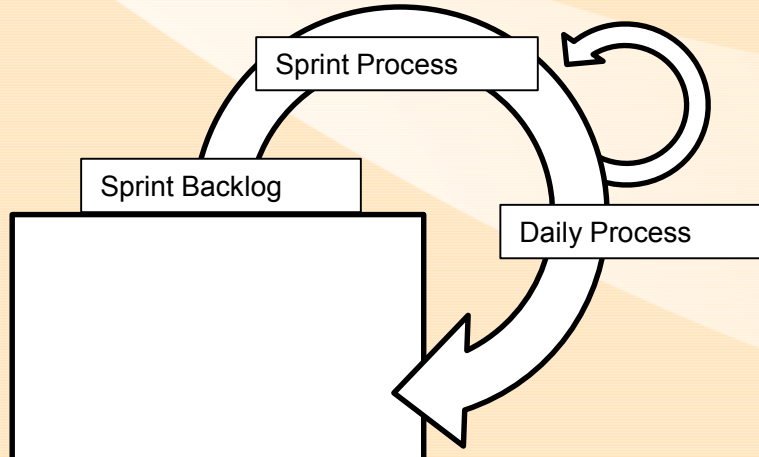
**REVIEW**

# Workday in scrum

---

The daily meeting needs to answer the following for each team member

- What did you do?
- What are you about to do?
- What currently stands in your way?



**REVIEW**

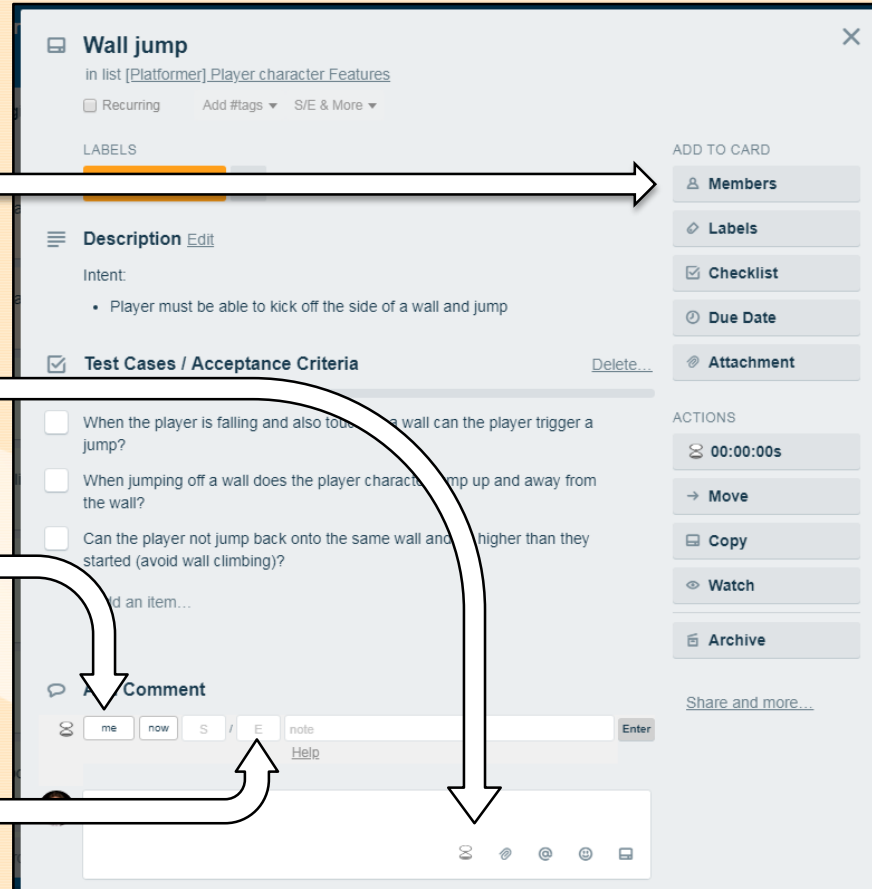
# Logging Hours with Trello

Add yourself as a member of any card you are responsible for

Click the hourglass to start logging hours if interface isn't already visible

The person who is taking ownership of the task and hours.  
Defaults to "me"; the person entering the hours on the card

Log the hours here  
E for estimate (sprint planning)  
S for time spent (tracking your progress)

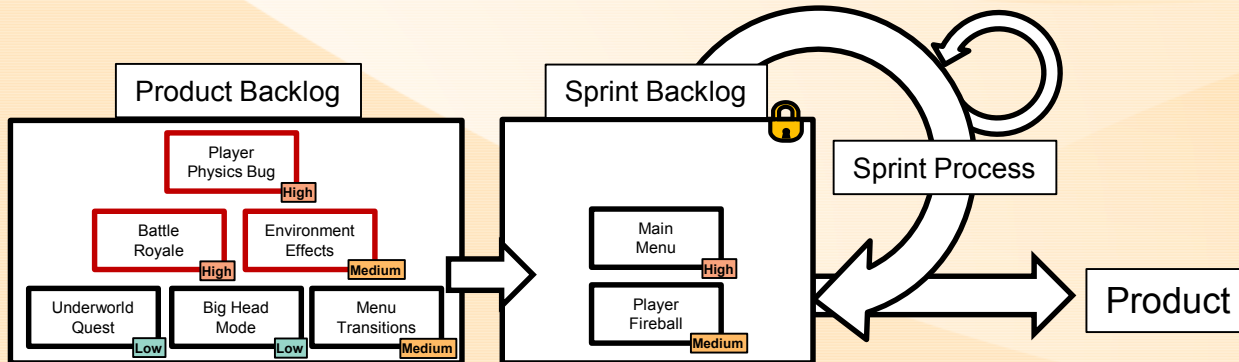


**REVIEW**

# Sprint Process: Change Requests

During the sprint, things are added to the product backlog if

- Discovered to make the product better
- Added from outside influences
- Changes in product expectation from client





---

# Todos for today

---

# Sprint Planning

---

- **Understand the overall sprint goals**
- **Select Userstories**
  - Using the product backlog select userstories that achieve the goals and move those userstories to this sprint board
  - If necessary create userstories for tasks that were previously unknown/overlooked
  - Review and edit test cases where needed
- **Evaluate Estimates**
  - Through planning poker the difficulty/hours/complexity is agreed on by all team members
- **Dividing work:**
  - Distributing the workload among the team by assigning owners for all of the stories
  - Each team member with roughly the same amount of work
  - Each team member with roughly the amount of work expected for the duration of the sprint
- **Submit sprint plan for approval**
  - Send a message to CD that the plan is ready for review
  - Sprint plan is not accepted until it has been reviewed by the CD and confirmed as approved directly.
  - It may take several passes to clear up all aspects of the sprint plan.

# Alpha Sprint Goals

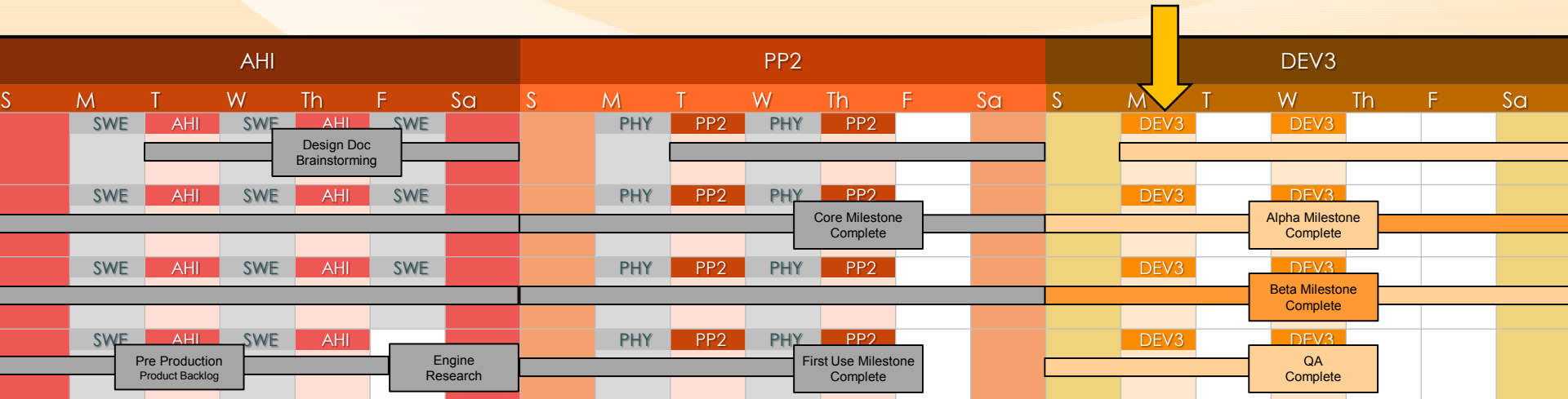
---

- The game world is populated with at least one example of each game object
  - All game object types
- Game progress can be shown
  - Multiple levels, multiple objectives, persistent data saved between levels...
  - Ability to progress. Not final level designs
- The game contains its win/loss conditions
- Cleared Technical debt
- Feature complete
  - Enough asset creation complete to prove the use of features
- Any feature not completed by the end of this sprint must be cut from the product

# Schedule

Hourly commitment

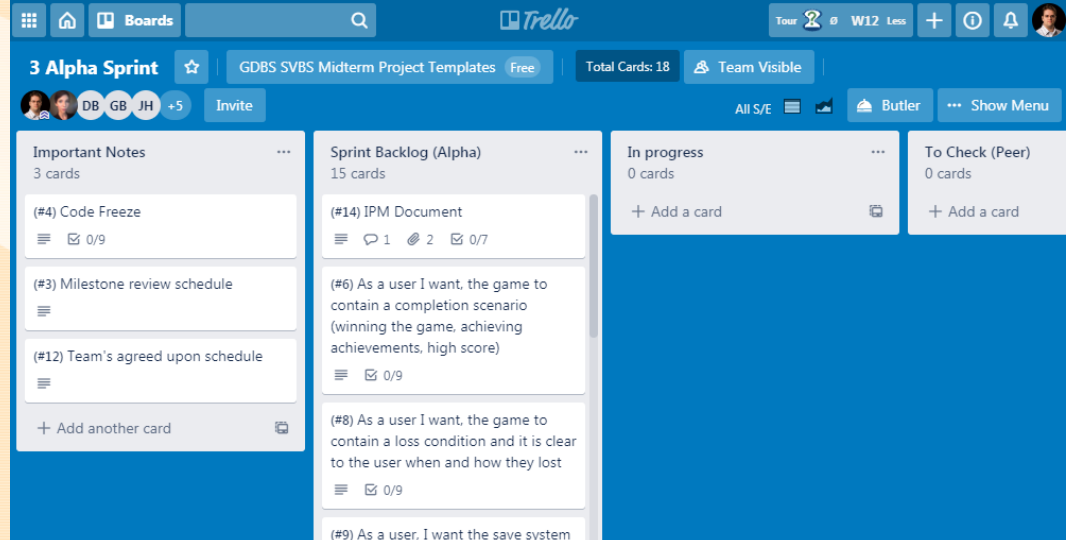
- 6 dedicated workdays
- ~42 hours a person
- Sprint review day 4 of PP3



# Alpha board is created

Move cards from product backlog here that are needed to achieve sprint goals

- Review and edit test cases
- Write new cards if needed



# Logging Hours with Trello

Add yourself as a member of any card you are responsible for

Click the hourglass to start logging hours if interface isn't already visible

The person who is taking ownership of the task and hours.  
Defaults to "me"; the person entering the hours on the card

Log the hours here  
E for estimate (sprint planning)  
S for time spent (tracking your progress)

The screenshot shows a Trello card titled "Wall jump" with the following details:

- in list [Platformer] Player character Features
- ☐ Recurring Add #tags S/E & More
- LABELS
- Description [Edit](#)
  - Intent:
    - Player must be able to kick off the side of a wall and jump
- ☒ Test Cases / Acceptance Criteria [Delete...](#)
  - ☐ When the player is falling and also touching a wall can the player trigger a jump?
  - ☐ When jumping off a wall does the player character jump up and away from the wall?
  - ☐ Can the player not jump back onto the same wall and end up higher than they started (avoid wall climbing)?
  - ☐ Add an item...
- [Comment](#)
- Input fields:  me  now  S  /  E  note  [Help](#)

Annotations on the right side of the card:

- An arrow points from the "Add yourself as a member of any card you are responsible for" text to the "Members" button in the "ADD TO CARD" section.
- An arrow points from the "Click the hourglass to start logging hours if interface isn't already visible" text to the hourglass icon in the "ACTIONS" section.
- An arrow points from the "The person who is taking ownership of the task and hours. Defaults to 'me'; the person entering the hours on the card" text to the "me" input field in the comment section.
- An arrow points from the "Log the hours here E for estimate (sprint planning) S for time spent (tracking your progress)" text to the "E" and "S" input fields in the comment section.

A red "REVIEW" stamp is visible in the bottom right corner of the screenshot.

# Alpha Sprint

---

## Sprint Goals

- The game world is populated with at least one example of each game object
  - All game object types
- Overall game progress can be shown
  - (multiple levels, multiple objectives...)
- The game contains its win/loss conditions
- Clear Technical debt
- Feature complete
  - Enough asset creation complete to prove the use of features
- Any feature not completed by the end of this sprint must be cut from the product

## Hourly commitment

- 6 dedicated work days
- ~42 hours a person