

# THE ULTIMATE GUIDE TO REACH YOUR DELIVERABLES

Stop missing deadlines and taking months on milestones that need to take days.

WWW.ATHENACONSULTING.IO

Presented By:  
Georges Chouchani





# INTRODUCTION

One of the main issues that blockchain startups have been facing is that they often miss deliverable deadlines; for example, product that should have launched in Q3 of 2022 is actually released a later.

Products that may only need a couple of weeks to deliver are taking months to develop, causing an extremely serious issue for owners, managers, developers and leaders as it is costing the startup thousands of dollars that could be spent on much more efficient use.

These slides are a step-by-step guide that will allow Web 3 companies to build and push to production faster, better and cheaper.



Although these slides are intended for Web 3 startups and companies, anyone who is leading a team and thinks that they are moving too slow could study these slides to exponentially increase the team's productivity and set realistic deadlines for the needed outcomes.

# DISCLAIMER



The content provided in this download is for informational purposes only and is not intended to replace professional advice from a qualified consultant. We make no guarantees, expressed or implied, with respect to the accuracy or reliability of the information contained herein. We shall not be liable for any loss or damage resulting from the use of this download as we are not responsible for how the content is applied. Use of this content is at your own risk.

# Step 1: Acknowledging The Issue

Most of you reading this may initially believe that there is no inherent issue in their company; products are being delivered on time and everyone is content.

At first glance, everything may seem normal and nothing needs to be changed, but, after diving deeper, your company may be burning thousands of dollars a month that could be invested in other areas to help the company grow.

The first step is to allow you to understand what are the tasks needed for a certain deliverable and how they are being distributed.

We will go through a checklist that may describe one or more situations currently happening with your team.

# The Ultimate Troubleshooting Checklist

- ▶ We do NOT estimate the time of the tasks of the deliverables needed before starting work.
- ▶ We do NOT divide big tasks into smaller manageable tasks that are easier to estimate and keep track of.
- ▶ We have NO idea how much the project will cost us (even if we are working with our in house team).
- ▶ We do NOT pre-plan reviews and audits of our work.
- ▶ We have NO idea how much the project will cost us (even if we are working with our in house team).
- ▶ The scope of our deliverable may change during the execution phase.
- ▶ We believe that we are saving money by not hiring outside help

**If you believe one or more of these situations represents a situation at your company,**

I can guarantee you that your company can optimize their pipeline and save thousands of dollars in the process.

# The Great Misconception: In-house Means No Extra Cost

## Perception

Many team leaders believe that if they build with in-house development team, then it is impossible to be wasting money since their burn rate is constant.

The team does not assign a certain cost to a project since there is no outside party working on it and there is not an extra cost.

## Reality

In reality, in-house development can cost multiple times more than outsourcing, if the team does not manage the project as they would manage an outsourced project since they do not take into consideration the inherent costs.

Due to having multiple projects in the pipeline, many teams do not treat each one as an independent projects, which causes huge delays in the project which increases the overhead massively.

# Step 2: Dividing The Project Into Manageable Tasks

After acknowledging that there is indeed room for improvement, we can now move to the first step in solving the issue. You can work along side the slides by choosing a project that you have worked with your team on in the past, or an upcoming project.

When you choose a project, we can now move to Step 2, which is dividing the project into manageable small tasks that will make the estimation of time and cost much easier.

As an example, we can use building an NFT marketplace as a project to work on.



# Building an NFT Marketplace: The Breakdown Process

①

The first step in the breakdown process is to list all the areas that the project will focus on.  
In our case these areas are:

Frontend

Design

Backend

Smart Contracts

②

Next, list the major tasks that should be completed under each specific area. (Tasks below are just an example)

Frontend

- ▶ Build UI from Landing Page Design
- ▶ Testing of built pages for glitches etc...

Design

- ▶ Definition of all needed pages
- ▶ Design Scheme for landing page
- ▶ Review of landing page
- ▶ Design Scheme for listing page
- ▶ Review of listing page

Backend

- ▶ Run frontend UI on server
- ▶ Create cache server that queries and stores data from blockchain for faster and seamless UI.

Smart Contracts

- ▶ Build Marketplace Smart Contract
- ▶ Write unit tests for smart contract
- ▶ Deploy and test smart contracts on mainnet

③ After listing the major tasks, it is time to break down the tasks to small manageable tasks. The rule of thumb is to breakdown a task into tasks that do not take up more than 5 hours each (estimated).



The reason is that, in the next step, you will estimate with your team the time it takes to complete each task. When tasks are small and manageable, its easier to estimate the time to complete and the error margin is usually smaller.



For example, building the UI on frontend might be estimated from 25 to 35 hours, while if the tasks are broken down into: 5 tasks which take up 5 hours each, it is easier to estimate the time of smaller tasks and therefor estimated time becomes more accurate. Tasks should also be done by one person in the team. If the task should be done by multiple people, then its not broken down enough.



Another reason is that breaking down to smaller tasks makes it easier to distribute tasks to the workforce and understand what tasks can be done in parallel or if there is any dependency.

# Breaking Down The Task: Build Marketplace Smart Contract



Build List function for  
Marketplace Contract



Build Purchase  
function for  
Marketplace Contract



Build List Query for  
Marketplace Contract



Build Single Listing  
Query



Build List function unit  
test



Build Purchase  
function unit test



Build List Query unit  
test



Build Single Listing  
Query Unit Tests

# Step 3: Estimate The Time For Each Individual Task (Three Point Shelling Method)



One of the most important tasks is to be able to as accurately as possible estimate the time for each individual task.



The Three Point Shelling Method is, in our experience, one of the best methods that allow to accurately estimate the time to estimate each task.



It consists of gathering the project team that will work on the project and allow them to estimate individually (by researching), the time (in hours) that each task may take. They should provide 3 estimations: Worst Time, Best Time, Average Time



The estimate of the expert in a certain task would be used as the benchmark for estimation, for example the smart contract developer's estimates would be used as benchmarks. If there is multiple experts, choose the main by seniority.

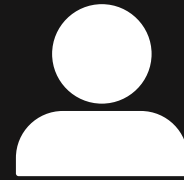


After setting the benchmark, take each estimation provided by the project team and check that the range exists inside the benchmark's range.



If the range given is in the benchmark's range, keep them and discard the rest. We then find the average of the provided estimate times in the range to provide the most accurate estimated time possible.

# Example on Estimating the Time for the Task: Build Purchase function for Marketplace Contract



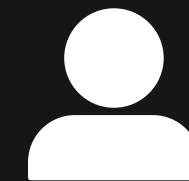
Smart Contract Developer:  
Best: 2 hours, Average: 3 hours, Worst: 4 hours



Project Manager:  
Best: 5 hours, Average: 6 hours, Worst: 7 hours



Frontend Developer:  
Best: 3 hours, Average: 4.5 hours, Worst: 5.5 hours



Backend Developer:  
Best: 3 hours, Average: 4.5 hours, Worst: 5.5 hours

 Out Of Range

 In Range

 In Range

Then, the estimated time of the project is the average of the prediction of the benchmark, and the predictions in range:

$$\text{Average Expected Time of Task} = 3 + 4.5 + 4.5 / 3 = 4 \text{ hours}$$

# Step 4: Identify Parallel Tasks and Workforce Needed

- ➔ After breaking down main tasks into smaller manageable tasks and after estimating the time of each task, it is time to identify tasks that can run in parallel and what is our workforce for the project.
- ➔ It is first important to set the constraints for the project team. What is the maximum number of backend developers you can have work on the project? What about the maximum number of designers?
- ➔ Then, from all the tasks you have divided for the project, you set all projects that can work in parallel. For example the designs for the landing page and for the listing page can be worked on in parallel.
- ➔ The number of tasks in parallel for each area decides what is the optimal number for workers for each area.
- ➔ The lower number between the constraint and the optimal number is the number that we will use for the next step.

# For Example:

The maximum number of each area I can get to work on my project are:



- 1 Designer
- 2 Frontend Developers
- 2 Backend Developers
- 3 Smart Contract Developers

In Smart Contract  
Development:

Build List function for  
Marketplace Contract

Build Purchase  
function for  
Marketplace Contract

Build List Query for  
Marketplace Contract

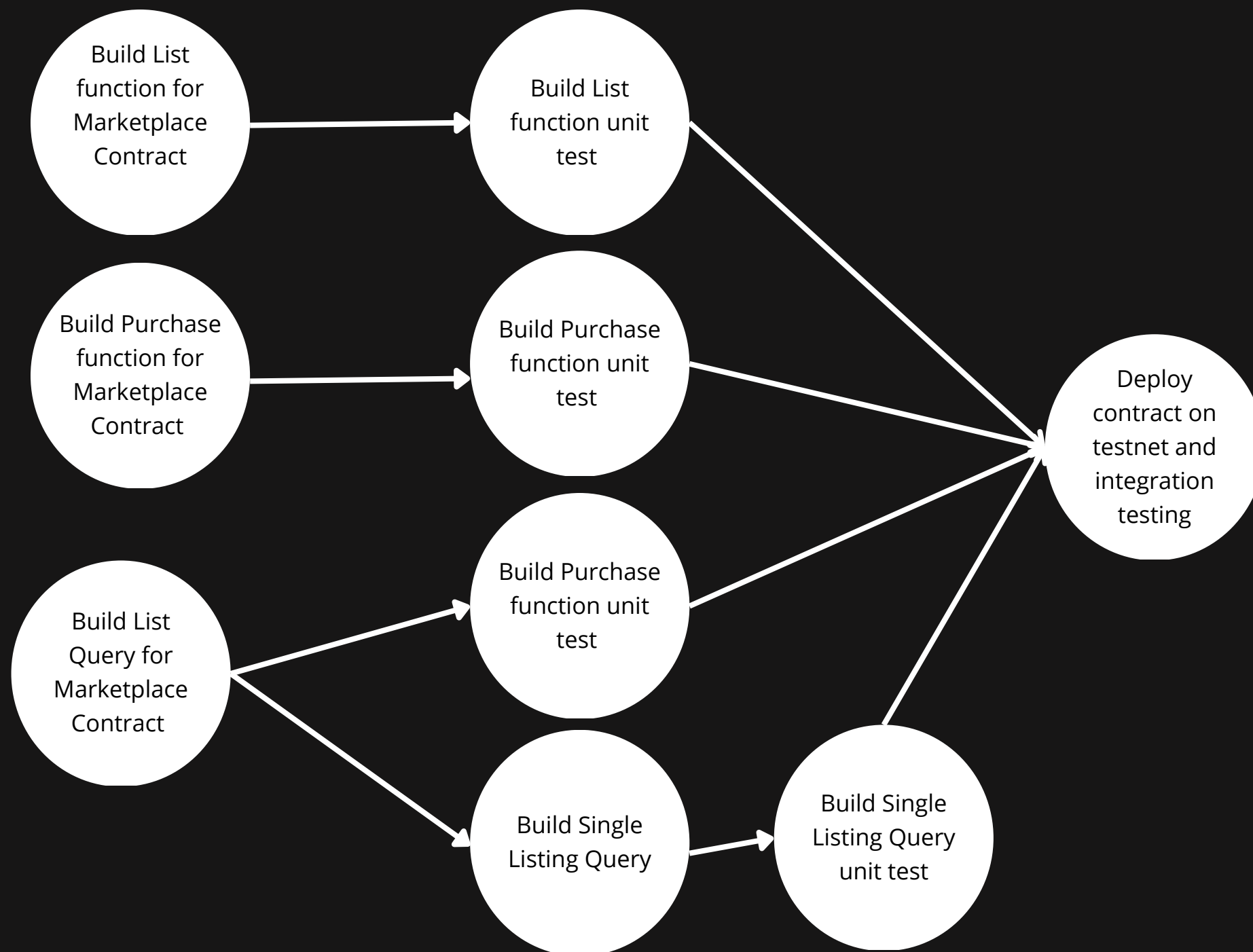
Build Single Listing  
Query

The above 4 tasks can be done in parallel, but since we have 3 smart contract developers, then we can only make 3 smart contract tasks work in parallel.

# Step 5: Drawing the Project Graph



After estimating the workforce, time of each task and tasks that can work in parallel, you can now draw the project graph that consists of the tasks in order depending on their dependency. For example, the smart contract development graph would look like this.



Note that although building the list query, and listing query could be done in parallel, but since we have only 3 backend developers that can work on the project, we must set the fourth parallel task after one of the first 3 tasks are completed.



Note that the project graph should include all the tasks that are needed to complete the project and areas usually mesh together. For example, frontend tasks cannot be started until some design tasks are completed.



# Step 6: Finding The Critical Path



After drawing the project graph, you can now input your findings on any critical path online solver (we recommend <https://sanjiverat.com/cpm/>)



You will need to input the task estimated duration and if it depends on any previous task. There is then two outputs expected:

- The project duration, which is found by adding the duration of tasks on the critical path.
- The activities on the critical path.



How do these help?

- The critical path is the longest path of the project. It gives the minimum time needed in order to complete the project.
- The project duration is the time you expect the project to last.



The tasks that are in the critical path are the most important tasks to not delay the project. A delay in any of these tasks leads to a delay in the whole project.



You would be surprised as how much the project duration is shorter than your expectations, which shows that good management and organization of the project lead to a shorter and cheaper project.

# Current Outcomes

- The time the project should take.
- The human resources needed for the project.
- Small manageable tasks with their estimated duration.
- The tasks that lie on the critical path.
- The total hours of work needed from all resources.

# Step 7: Resource and Reserve Analysis

- It is important to note that not everything will go according to plan. A developer may become sick for a week, a designer might suddenly leave the team or other unforeseen situations may arise during the course of the project.
- Although it may seem that unforeseen issues could jeopardize the work done in the planning and preceding steps, effective preparation by the team leader can mitigate potential issues and ensure that deliverables and deadlines are not impacted.
- The best way to manage unknown variables during the planning phase is to allocate a time buffer, or time reserve, for the critical path and its associated tasks.
- To determine an appropriate buffer, it's often best practice to plan for a worst-case scenario in terms of project delays, and then adjust the contingency linearly as more information becomes available during the execution phase.
- Although a full quantitative analysis can be used to determine the appropriate buffer, we recommend that team leaders start with a buffer of 20% of the estimated time and adjust it as their confidence increases that issues will not arise.

# Step 8: Handling Scope Change

- Scope changes that occur in the middle or end of the execution phase can be one of the most frustrating situations for team leaders.
- The first rule to understand is that everything you've built and planned thus far is not set in stone. Project graphs, critical paths, cost estimations, and other factors can change.
- While it can be frustrating, it's important to remember that changes are a common occurrence in most projects. Be sure to keep clients, users, and stakeholders informed of any changes as they happen.
- Go back through the steps and ensure that the project graph, critical path, and tasks accurately reflect the latest tasks that need to be completed.
- The mark of a great team and leader is how they handle change and succeed in difficult situations.



Note that not all scope changes are negative and sometimes can turn the project cheaper and easier to complete.

# Step 9: Cost Estimation



Estimating the total cost of the project is one of the final and most important steps in the planning process. Fortunately, if all the preceding steps have been completed successfully, this step will be relatively easy.

①

Revisit the breakdown of tasks into the three different areas, and assign them to resources (employees) on your team based on their areas of expertise.

②

To calculate the estimated cost of each task, multiply each team member's hourly rate by the estimated time for the task. To get the hourly rate, divide the team member's annual salary by the number of working hours in a year.

③

Finally add up all the estimated costs of each task to get the total cost of the project.



Do not forget to go back to step 7 and take into consideration the time buffer for the project that adds a cost contingency that would be smart to keep in place for any situation that arises.

# Step 10: BUILD, BUILD, BUILD

If you completed the 10-step guide then your team is ready to start building.

You are now armed with:

- An accurate estimated cost.
- A shorter and more concise deadline.
- Critical Path tasks that should NOT be delayed.
- All the tasks needed with the assigned team member to complete the deliverable.

# Did You Like The Guide?

Athena Consulting strives to provide top-notch blockchain implementations and solutions to startups, and companies looking to leverage the power of blockchain.

We offer a full refund on our services anytime during the process if the client is not satisfied (although we highly doubt that will be necessary).

If you are interested to learn more about our services and how we can help you reach your goals, please feel free to contact us from our website linked below.

Thank you for your time in reading and going through our guide. Stay tuned for updates coming soon.

Georges Chouchani  
Founder of Athena Consulting



<https://www.athenaconsulting.io>

