

Project name

Parachain runtime builder (PRB)

1. General Project Information

Short description: MVP Workshop wants to refactor and update an open-sourced project [Substrate Starter Kit](#) which was created to explore and understand the Substrate framework. The new version of the tool, dubbed Parachain Runtime Builder (PRB), further simplifies onboarding with predefined runtimes. Developers can pick runtimes with predefined configurations, customize them, and deploy them to a private GitHub repo. Node operators leverage Parachain Runtime Builder (PRB) via API to deliver 1-click deployment.

Project Category / Type: Infrastructure and Network Operations (improved parachain deployment)

Proponent: Nikola Mandic | MVP Workshop

DOT address: 123jNGxHk9ZV7oVVhFWFtMghNpmnmmTWxSpNxf8TTKzmCSQ2

Requested allocation: 34540 DOT (based on price 21.02.2023, 9 PM), \$255.600

Onchain publish date: 21.02.2023

Governance referenda origin call: Medium spender

Previous treasury proposals: [68●UI & Front-end for Asset Conversion pallet \(referenda68\)](#) - [Project Status Update - Delivery Completed](#)

2. Context

This proposal comes from a recognized need to upgrade the current [Substrate Starter Kit](#) (SSK) used in the Polkadot ecosystem that speeds up the process of creating customized parachains by simply drag-and-dropping pallets to a codebase. This tool has gone through the release back in 2020 by [MVP Workshop](#) as part of the [Substrate Starter Kit GUI grant proposal](#) and has been developed entirely in-house. Since then the Polkadot environment developed and changed, resulting in less usability of the kit as it has not gone through maintenance and so it became outdated, not delivering its potential to the community.

The goal is to update the existing SSK, using RUST, and give the community a fully functioning parachain development platform - Parachain Runtime Builder (PRB). In the new version, developers will easily set up and send a fully configured parachain to a private GitHub repository which they are not able to do in the current constellation. Furthermore, node providers, Blockchain-as-a-Service businesses, and other stakeholders will also benefit from it as they can use our public API to easily integrate the PRB into their system or testing environments, making it

even simpler to launch parachains. As the benefit of the upgrade was recognized by the development community, we are sharing the mutual vision with other teams like [Zeeve](#), whose initial version of the [1-click node deployment, known as the “Sandbox”](#), has already gone through the Web3 Foundation Grants Program. We aim to integrate PRB with Zeeve’s solution, by showcasing early a synergy that will empower both products and offer its full potential to the Polkadot community.

The proponent of this proposal is MVP Workshop, a pioneer in the Web 3 realm with an extensive background in diverse development environments. We are a team of over 50 individuals, with a portfolio of more than 70 blockchain products under our belt. Our reliable collaboration and delivery capabilities are evidenced by our notable partnerships, becoming the official [implementation partner for Polygon](#), [Parity’s Substrate delivery partner](#), and developing a comprehensive tech stack for the Celsius network. Our expertise is further showcased in various competitions, such as this year’s ETH Global Hackathon where we advanced to the top 12 finalists for building dApps atop the FVM (Filecoin Virtual Machine).

In July, MVP Workshop submitted a proposal for the UI & Front-End of the Asset Conversion Pallet on Polkadot, marking the beginning of a significant endeavor. After a tough selection process, our team commenced the project in mid-August and successfully delivered a robust solution in October. Our commitment extends beyond delivery; we have been actively maintaining the project, facilitating seamless integrations for entities such as SubWallet and the Kabocha parachain on Kusama. The overwhelming interest and positive reception of our product have led us to a strategic decision: to expand our presence by [deploying the Asset Conversion Pallet on Kusama](#), coinciding with its launch on this platform. This expansion signifies MVP Workshop's dedication to enhancing interoperability and efficiency within the Polkadot and Kusama ecosystems.

Our long-term vision is to pioneer a comprehensive Blockchain-as-a-Service (BaaS) platform within the Polkadot ecosystem, internally named SubstrateCloud. This vision aims to bridge a crucial gap in the current ecosystem by providing an array of services that facilitate the creation and management of custom, efficient blockchains. Our envisioned SubstrateCloud will offer features like a user-friendly Blockchain Builder, Managed Infrastructure, Data Tools, and a BlockExplorer. These tools are designed to simplify blockchain creation, enhance interoperability, ensure robust security, and provide comprehensive analytics and user onboarding solutions. Central to this vision is the development of Parachain Runtime Builder (PRB), which serves as the cornerstone for our ambitious project. The SSK will lay the groundwork for the SubstrateCloud by enabling users to easily create and configure custom blockchains using Substrate’s flexible framework. It’s the first critical step towards realizing a fully-fledged AaaS platform, designed to cater to a diverse range of needs - from technical ease for developers to strategic business solutions. By starting with the SSK, we are setting the stage for a broader, more inclusive, and innovative approach to blockchain development in the Polkadot network, ultimately leading to an enriched, more dynamic ecosystem.

3. Problem

This proposal addresses a problem in the Polkadot ecosystem related to deploying blockchains, including parachains and standalone substrate chains. Substrate Starter Kit (SSK) has outdated dependencies, impacting its functionality for configuring and deploying parachain configurations to a private GitHub repository.

This issue slows down deployment speed, particularly for quick Proof of Concept or Minimum Viable Product development, and poses a broader challenge to blockchain deployment ease.

Streamlining this process is crucial for fostering innovation and aiding rapid development within the ecosystem, removing a significant barrier for developers, especially newcomers to the blockchain space.

4. Proposal

Our proposed solution is to revise and update the SSK, transforming it into a robust, Rust-based platform that simplifies the entire deployment process. The revamped platform we call Parachain Runtime Builder (PRB) will provide a 'click and play' feature, eliminating the need for coding during deployment, thus accelerating the initial setup for developers. This is not merely about speeding up the deployment but enhancing the overall experience of blockchain development, whether it's for parachains or standalone substrate chains. Additionally, node providers and other Blockchain-as-a-Service entities will benefit from the enhanced API, which will allow seamless integration of the PRB into their user interfaces or sandbox environments. The overall goals included in this proposal are:

- Rewrite Substrate Starter Kit in Rust to improve performance,
- Refreshing 50+ dependencies in the deployment process,
- Enhance the UI to accommodate in-depth pallet configuration,
- Create an API to enable PRB integration into 3rd party services and 1-click deployment,
- Update the current SSK Website,
- Create educational material and business logic runtime templates to better onboard developers
- Crafting of Runtime templates
- Provide maintenance service to PRB for 1 year.

4.1. Available solutions

Given that there are currently no comparable production-ready solutions within the Polkadot ecosystem, the implementation of a solution like PRB becomes essential.

4.2. Scope of work

The outlined scope of work is segmented into three distinct phases to facilitate a clearer understanding of the deliverables associated with each phase:

- Phase 1: SSK Rust Upgrade and Parachain Deployment Configuration
 - MVP Workshop will convert SSK from a TypeScript-based to a Rust-based project to improve performance, update all the Functional and FRAME pallet configurations so that they would be compatible with the latest pallet versions, and deploy the code into a public GitHub repository. This phase also involves our work on developing the runtime picker, which consists of four parachain templates. These templates will be designed to allow users to easily deploy their preferred parachains. The four parachain templates are:
 - Basic parachain template
 - Smart contract parachain template
 - EVM compatible parachain template
 - Defi parachain template
- Phase 2: Development of integration-ready API and functional user interface
 - MVP Workshop will deploy an API that allows 3rd parties to integrate PRB into their products, and from their own UI configure and build parachains. API documentation will be available as well. We will also introduce a website featuring a standalone UI that utilizes the same API. This UI will allow developers to:
 - Configure custom parachains according to their requirements.
 - Create one-click parachains using pre-existing templates.
 - Modify specific pallet configurations to suit their needs.
 - Deploy the configurations into their private GitHub repositories.
- Phase 3: Maintenance and Integration Assistance
 - MVP Workshop will continually monitor, resolve issues, and review external PRs on the PRB project, as well as update the pallets to maintain compatibility. MVPW will also assist and support 3rd parties seeking to integrate SSK into their products. This service will be provided on an ongoing basis for 1 year.

A detailed breakdown of the phases is further elaborated upon in the Phases section (4.3).

4.3. Phases

Phase 1. – SSK Rust Upgrade and Parachain Deployment Configuration				
Task	Description	Resource	Deliverable	Time
SSK Update to RUST	Rewrite the SSK backend for modifying parachain template files and publishing generated code to a GitHub repo from TypeScript to Rust	2 FTE Backend Engineers 1 FTE Product Manager	Github repo	5 weeks
Reconfiguration base template for parachain deployment	Upgrade the currently used substrate node template to a parachain-based template .	2 FTE Backend Engineers 1 FTE Product Manager	Github repo	1 week
Integration of all Functional & Parachain FRAME pallets	Update configurations for runtime and genesis config for FRAME pallets	2 FTE Backend Engineers 1 FTE Product Manager	Github repo	4 weeks
Runtime picker: parachain templates	Developing four parachain templates involves analyzing the best industry examples and integrating appropriate pallets into the parachain configuration. After detailed testing of each parachain configuration, these templates will be ready to use via PRB UI. Templates are: Basic parachain template, Smart contract parachain template, EVM compatible parachain template, DeFi parachain template.	2 FTE Backend Engineers 1 FTE Product Manager	Github repo	3 weeks

Create an automatic pallet upgrade mechanism	Develop a checker that can detect whenever new versions of pallets become available. Implement a systematic process for automatically upgrading the pallets when updates are detected.	2 FTE Backend Engineer 1 FTE Product Manager	Github repo	1 week
Total time				14 weeks

Phase 2. – Development of integration-ready API and Interface Enhancements				
Task	Description	Resource	Deliverable	Time
Development of integration-ready API	MVP Workshop will implement an API enabling third-party integration of PRB into their products. This API will empower users to configure and construct parachains directly from their user interfaces. Comprehensive documentation for this API will also be provided.	2 FTE Backend Engineers 1 FTE Product Manager	Github repo API doc	4 weeks
A fully functional UI with the ability to configure individual pallets	Redesign of the current product to enable users to adjust specific configuration parameters for each pallet they are utilizing via a user-friendly interface (UI).	1 FTE Frontend Engineer 1 FTE Product Designer 1 FTE Product Manager	Github repo	6 weeks
Upgrading SSK website	Develop the project's homepage and incorporate additional pages containing important information such as Terms of Service, GitHub repository, API documentation, and other relevant project data.	1 FTE Frontend Engineer 1 FTE Product Manager	Website	2 weeks
Total time				12 weeks

Phase 3. – Maintenance and Integration Assistance			
Task	Description	Resource	Deliverable
Ongoing maintenance of the project and its infrastructure	This will be our continuous effort to maintain the project's stability. It includes monitoring system performance and ensuring continuous operation.	1 FTE Backend	99.9% uptime
Parachain templates and pallet updates	Add or upgrade new pallets as new versions are available. Configure and test parachain templates after each update and integrate with the project's repo.	1 FTE Backend	GitHub Pull Requests
Bug fixes	Addressing identified issues promptly	1 FTE Backend	GitHub Pull Requests
Reviews of 3rd party contributors (feedback, PRs)	Review of contributions made by the community including examining submitted code, assessing quality, and providing feedback for improvement.	1 FTE Backend	GitHub Pull Requests
Assistance to 3rd parties integrating SSK into their products	Providing technical support, addressing tickets, and offering guidelines on integration.	1 FTE Backend	GitHub Issues and Pull Requests

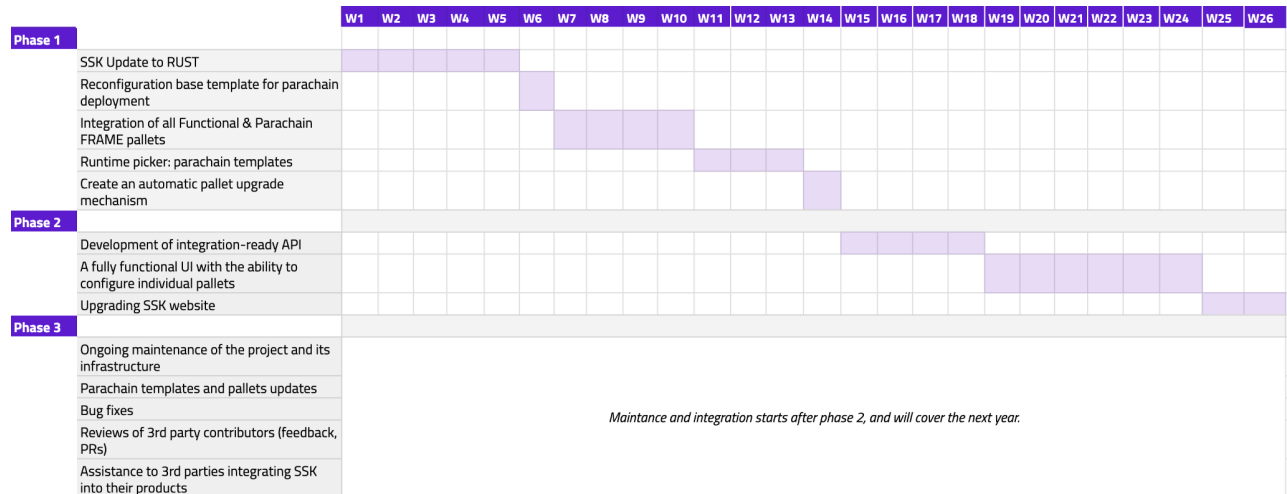
**Given that phase 3 primarily involves maintenance and integrations, no time estimations are presented here.*

4.4. Timeline

PRB Project Phases		
Phase	Description	Time
1	SSK Rust Upgrade & Parachain Deployment Configuration	14 weeks
2	Development of integration-ready API and Interface Enhancements	12 weeks
3	Maintenance and Integration Assistance	1 year

Total time	26 weeks (development) 1 year (maintenance and support)
-------------------	--

PRB's Development Roadmap



4.5. Budget

BACKEND	36 WEEKS OF EFFORT	3.230€	116.280€
WEB APP	8 WEEKS OF EFFORT	2.770€	22.160€
DESIGN	6 WEEKS OF EFFORT	2.310€	13.860€

PRODUCT	26 WEEKS OF EFFORT	2.770€	72.020€
MAINTENANCE	12 MONTHS OF EFFORT	1.000€	12.000€
Total			236.320€ (255.600\$)

4.6. Payment condition

Proponent: Nikola Mandic, Business Developer @ MVP Workshop

DOT address: 123jNGxHk9ZV7oVVhFWFtMghNpmnmmTWxSpNxf8TTKzmCSQ2

Requested allocation: 56.595 DOT (based on price 27.10.2023, 12 PM), \$253.430

Governance referenda origin call: Medium spender

Final budget	
Phase 1	139.750 \$
Phase 2	102.850 \$
Phase 3	Free of charge
Infrastructure Maintenance (full year)	13,000 \$
Budget	255.600 \$

Treasury request	
DOT/USD EMA30 on 27/10/2023 ¹	7.37\$
USD to DOT	255.600 \$ / 7.37\$ = 34.540 DOT
Total amount requested	34540 DOT

Sources:

- ¹ DOT/USD EMA30 source: <https://polkadot.subscan.io/tools/charts?type=price>

4.7. Key deliverables

Phase 1

- Public, open-source GitHub repository with up-to-date configurations of all Functional & FRAME pallets, parachain templates, and easily customizable pallet configurations via great product UI.

Phase 2

- Integration-ready API, enabling 3rd parties to integrate PRB into their products
- API Documentation
- Functional user interface on a public website that communicates with PRB via API

Phase 3

- Ongoing project support, maintenance and updates for 1 year
- Assistance to 3rd parties seeking to integrate PRB into their products

4.8. Objectives/Success criteria

The primary goal of the Parachain Runtime Builder (PRB) project is to significantly enhance and revolutionize blockchain development within the Polkadot ecosystem. By upgrading and refining the PRB, we aim to create a tool that is not only robust and efficient but also intuitive and user-friendly for developers at all levels.

Key Objectives and Strategies:

1. Refactoring and Modernization:

- Rewrite the existing SSK using the Rust programming language to optimize performance.
- Refresh all dependencies to ensure the platform remains contemporary and efficient, addressing any existing issues and preparing it for future scalability.

2. User Interface Enhancement:

- Develop a sophisticated yet accessible user interface for the PRB.
- Enable developers to effortlessly configure and manage pallets, thus simplifying the process of creating customized blockchains.
- Focus on usability to cater to both seasoned developers and newcomers in the ecosystem.

3. API Development and Integration:

- Design and implement a comprehensive public API to facilitate the integration of PRB into third-party services and development environments.
- Ensure that the API is robust, well-documented, and easy to use, thereby broadening the accessibility and utility of the PRB.

4. Educational and Community Engagement:

- Establish a PRB Playground as a platform for educational resources, practical examples, and real-world use cases.
- Focus on providing clear, concise, and practical learning materials to help developers understand the usage and configuration of pallets for various business applications.
- Engage with the community through forums, workshops, and interactive sessions to gather feedback and foster a collaborative development environment.

To measure the success and completion of the Parachain Runtime Builder (PRB) project within the Polkadot ecosystem, considering the information from the provided documents, the following metrics can be employed:

1. PRB Playground Metrics:

- **Number of GitHub Repositories Deployed:** Track the count of GitHub repositories utilizing the PRB. A higher number indicates widespread adoption and utility of the kit among developers.
- **Time Spent on Educational Materials:** Monitor the duration developers spend on learning materials related to pallet usage and configurations. This metric will help gauge the effectiveness and engagement of the educational content.

2. Production Metrics:

- **Number of Staging Environments:** Measure the number of staging environments created using PRB. This reflects how frequently developers are testing and preparing their projects for production, indicating the practical utility of the PRB in real-world applications.

3. Community Adoption Metrics:

- **Developer Feedback and Engagement:** Collect feedback from the developer community to assess the kit's usability, performance, and relevance. Engaging with the community through forums, surveys, and discussion boards can provide qualitative insights.
- **Collaborations and Partnerships:** Track the number of collaborations and partnerships formed as a result of the PRB. Partnerships with other teams and projects within the Polkadot ecosystem can be a strong indicator of the kit's value and adoption.
- **Integration in New Projects:** Monitor the number of new projects in the Polkadot ecosystem that integrate the PRB. This shows the kit's contribution to ecosystem growth and innovation.

4. Long-term Vision Alignment - Substrate Cloud:

- **Community Interest and Feedback:** Assess the community's interest in the long-term vision of the Substrate Cloud. This can be done through surveys, community discussions, and feedback sessions.
- **Progress Towards Phases:** Evaluate the progress made towards achieving the Phases set for the Substrate Cloud. This includes the development of the Blockchain Builder and other features outlined in the proposal.

4.9. Known constraints

The Parachain Runtime Builder (PRB) project may face schedule and budget impacts due to changes in the Polkadot ecosystem, technical challenges, and market conditions. Updates in Polkadot's protocols could require us to adjust our development strategy. Financially, market fluctuations could affect resource costs and funding. We have ensured no conflict of interest with other Polkadot proposals, and any potential conflicts or significant project changes will be promptly disclosed.

4.10. Reporting

To uphold transparency and accountability in the Parachain Runtime Builder (PRB) project, we have established a comprehensive reporting process, ensuring the Polkadot community stays informed about our progress, achievements, and challenges. Monthly reports will be our standard communication routine, providing consistent updates on our ongoing activities, developments, and any immediate issues or successes. These reports will be primarily shared on Polkasassembly, ensuring accessibility and visibility within the community. For significant Phases, we will prepare detailed updates, highlighting key achievements and insights gained upon the completion of each major project phase. These Phase reports will not only be shared on Polkasassembly but also disseminated through Discord and the Element Polkadot Direction channel, fostering a broad-based engagement and dialogue with the community. This approach balances regular updates with in-depth reporting at critical junctures, reflecting our commitment to transparency and community engagement throughout the lifecycle of the PRB project.

4.11. Communication strategy

In our commitment to transparent and responsive communication for the Parachain Runtime Builder (PRB) project, we will employ a cohesive strategy, utilizing Polkasassembly and Element as our primary channels for community engagement and updates. Polkasassembly will serve as the forum for detailed project discussions, proposal updates, and community feedback, while Element will facilitate real-time communication and direct interactions with our team. We have established a policy to respond to all inquiries and feedback within a 72-hour window, ensuring that our community and stakeholders are heard, acknowledged, and informed throughout the project's lifecycle. This integrated approach will guarantee consistent, clear, and timely communication, reinforcing our dedication to the Polkadot ecosystem and its community.

Lead Communicator:

Team member: Nikola Mandić | Business development lead

Email: nikola.mandic@mvpworkshop.co

Polkasassembly post:

Element: @mandaweb3:matrix.org

Discord: maandic

Technical Communicator:

Team member: Dragana Koprivica | Product Manager

Email: dragana.koprivica@mvpworkshop.co

Polkasassembly post:

Element: @gasha994:matrix.org

Discord: gasha1513

5. Team

(List the team members that will actively work on the project. If possible, provide team credentials and background.)

- Nemanja Aleksić
- Filip Jovakarić
- Marko Jovanović
- Žarko Jevtić
- Filip Moldvai
- Milica Spasojević

5.1. Kusama/Polkadot reputation

- [1st Belgrade Polkadot Day](#) - Organized this June as part of [Belgrade Blockchain Week](#), Polkadot Day 2023 featured panel discussions, workshops, and networking opportunities. Attendees had the chance to hear from some of the most prominent figures in the Polkadot ecosystem, including Parity, Astar, Moonbeam, and other popular & growing parachain leaders.
- [MACI Pallet](#) - Minimal Anti Collusion Pallet. The grant was initially approved, but after discovering complex technical issues preventing the grant completion, it was terminated.
- [Substrate Starter Kit GUI](#) - a GUI where users can quickly select the runtime pallets, enter their genesis configurations, and satisfy their dependencies without forking the template repository and writing any RUST code. The application would add/remove the necessary files and set configurations and dependencies in the codebase behind the scene. In the end, the user will generate a code base with all the components. The grant was successfully completed in 2020.
- Astar Tokenomics - MVP Workshop developed a dynamical-systems model using the cadCAD Python library to test Astar Network's new token economy. This model effectively calculates stakers' APR relative to TVL fluctuations and adapts the number of slots per tier based on bear and bull market scenarios. It also determines dApp builder rewards and token burn rates under various market conditions and tier fullness, while comparing transaction fees for EVM and WASM. The audit and subsequent changes to Astar Network are set to enhance sustainability and equity, featuring aligned fees, reduced inflation, fairer rewards, and an advanced token burn mechanism, ultimately optimizing token utilization within the system.
- UI & Front-End for Asset Conversion Pallet - We successfully completed our UI and Front-end KIT for Asset Conversion Pallet, which has been deployed and tested on Westend and Rococo networks. The [UI Kit repository](#) is now fully operational and open for implementation and customization. Additionally, you can experience the UI Kit firsthand on our live staging environment on both [Rococo](#) and [Westend](#). For any assistance, our [testing instructions](#) are readily available. Also, in response to the growing interest from various projects, we are excited to inform you that we are planning to expand our deployment to the Kusama network, broadening our reach and utility in the ecosystem.

6. Why Polkadot?

Leveraging our deep expertise in infrastructure and enterprise solutions, we recognize the transformative potential of Polkadot in driving technological adoption across industries. Our focus is on harnessing Polkadot's scalability and robust security to deliver customized, enterprise-grade solutions. Polkadot's architecture, which facilitates efficient cross-chain data transfer and asset scalability, aligns perfectly with the needs of businesses seeking agile and scalable systems. This

capability enables us to tailor solutions precisely to the varied and specific requirements of different enterprises.

However, we are also mindful of the challenges associated with Polkadot's relative novelty, including its complexity and the potential for bugs. The cost implications, particularly regarding parachain auctions and on-chain bonds, are significant considerations for businesses. To address these, our approach includes utilizing Substrate for deploying chains that can integrate into the Polkadot ecosystem as parachains. This strategy not only mitigates go-to-market (GTM) impacts but also efficiently allocates resources for building and deploying customized or templated runtimes.

Crucially, we understand the difference between Substrate-based chains and Polkadot parachains which helped us to understand potential that Polkadot brings to this solution regarding its shared security and consensus mechanism. Our primary focus will be helping the parachains. This involves configuring and building pallets, deploying runtimes, benchmarking low-level infrastructure, and deploying parachains. Our approach considers the specifics of current parachain deployment, including auctions, on-demand parachains, and agile coretime, ensuring that our infrastructure-oriented project remains adaptable, efficient, and aligned with the evolving needs of the industry.