Title of Document

Subtitle of document



Technical support team – Feb 2023

Context

[**Introduction** 3](#_Toc138254419)

[**Technical info** 4](#_Toc138254420)

[**System Parts** 5](#_Toc138254421)

[**API List** 6](#_Toc138254422)

[**API 1**  6](#_Toc138254423)

# **Introduction**

The Introduction section of a System Document provides an overview of what the system is, how it works, and what it does. This section aims to give the reader a general understanding of the system's purpose and function.

In this document, we will introduce our new inventory management system. The system is designed to help businesses keep track of their inventory levels and streamline the ordering process. With our system, users can easily monitor stock levels, reorder items when necessary, and receive notifications when stock levels fall below certain thresholds.

The system works by utilizing a centralized database that stores all information related to inventory levels, orders, and shipments. Users can access this database through a web-based interface, which allows them to view real-time inventory data and generate reports on demand. The system also includes automated order processing and shipment tracking features, which help to reduce errors and improve efficiency.

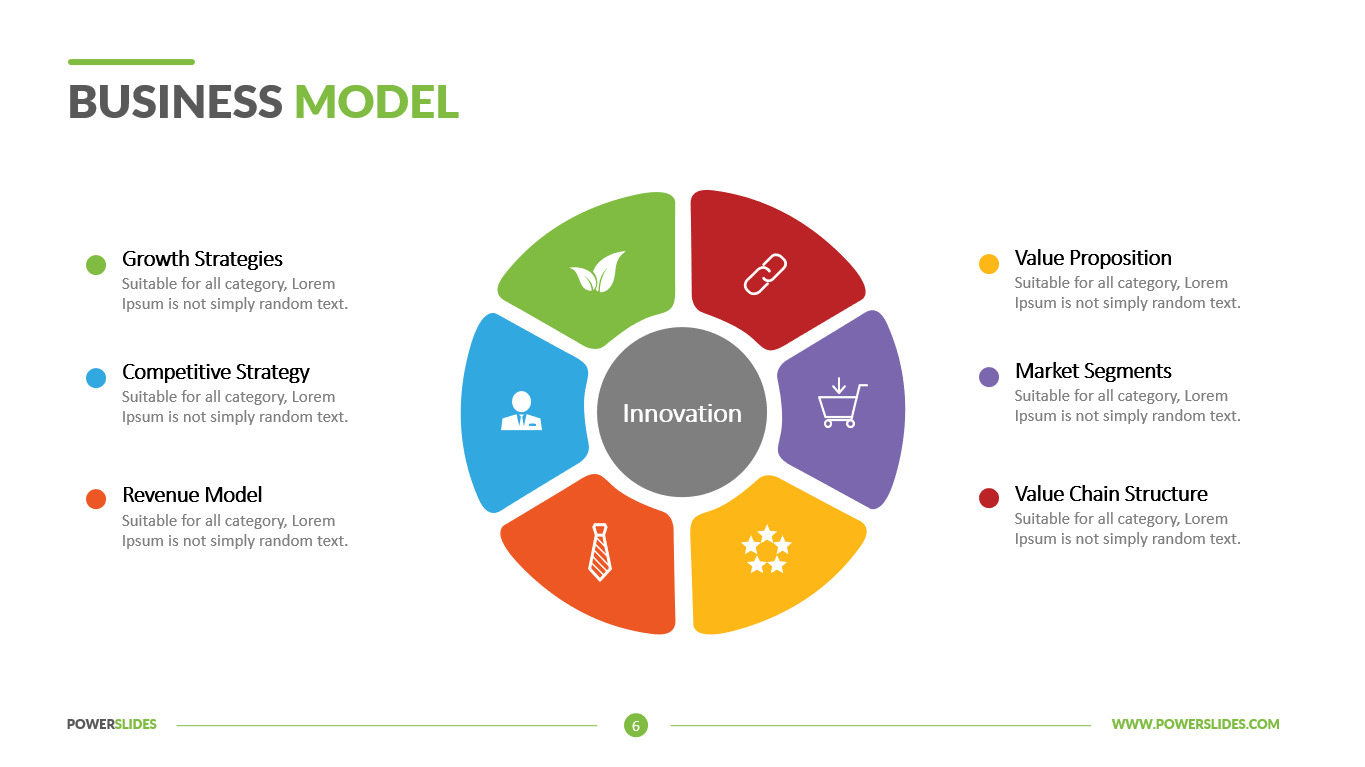
Overall, our inventory management system is a powerful tool for any business looking to improve their inventory control processes. With its user-friendly interface, comprehensive reporting capabilities, and advanced automation features, the system provides users with a seamless experience that enables them to focus on growing their business.

Figure 1 - System info diagram or picture

**Technical info**

Any technical information of service could be here.

The system is built using Python programming language and leverages the Flask framework for web development. It also utilizes various tools such as Git for version control, Docker for containerization, and Amazon Web Services (AWS) for hosting and infrastructure management.

The application architecture follows a Model-View-Controller (MVC) pattern to ensure separation of concerns and maintainability. Additionally, various libraries such as NumPy, Pandas, and Scikit-Learn are used for data processing and machine learning tasks.

The UI is designed using HTML, CSS, and JavaScript with the help of Bootstrap framework for responsive web design. Overall, the system is designed to be scalable, robust, and maintainable while utilizing cutting-edge technologies and tools.

|  |  |  |
| --- | --- | --- |
|  | Title | Explanation |
| 1 | System capacity | 1000 users per second |
| 2 | Logging | Yes |
| 3 | Monitoring | Yes |
| 4 | API Lists | Yes, Postman collection |

Table 1- System technical info

# **System Parts**

The system consists of several interconnected parts that work together to provide the users with a seamless experience.

The main components of the system include the user interface, the application server, the database server, and the network infrastructure.

The user interface is responsible for presenting information to the users in a clear and concise manner. It allows them to interact with the system and perform various tasks such as creating, editing, and deleting data.

The application server serves as the backbone of the system. It processes requests from the user interface and communicates with the database server to retrieve or store data.

The database server stores all the data related to the system, including user information, transaction records, and other relevant details.

Finally, the network infrastructure connects all the components of the system and enables communication between them. Together, these parts ensure that the system operates efficiently and effectively, providing users with a reliable and intuitive platform for their needs

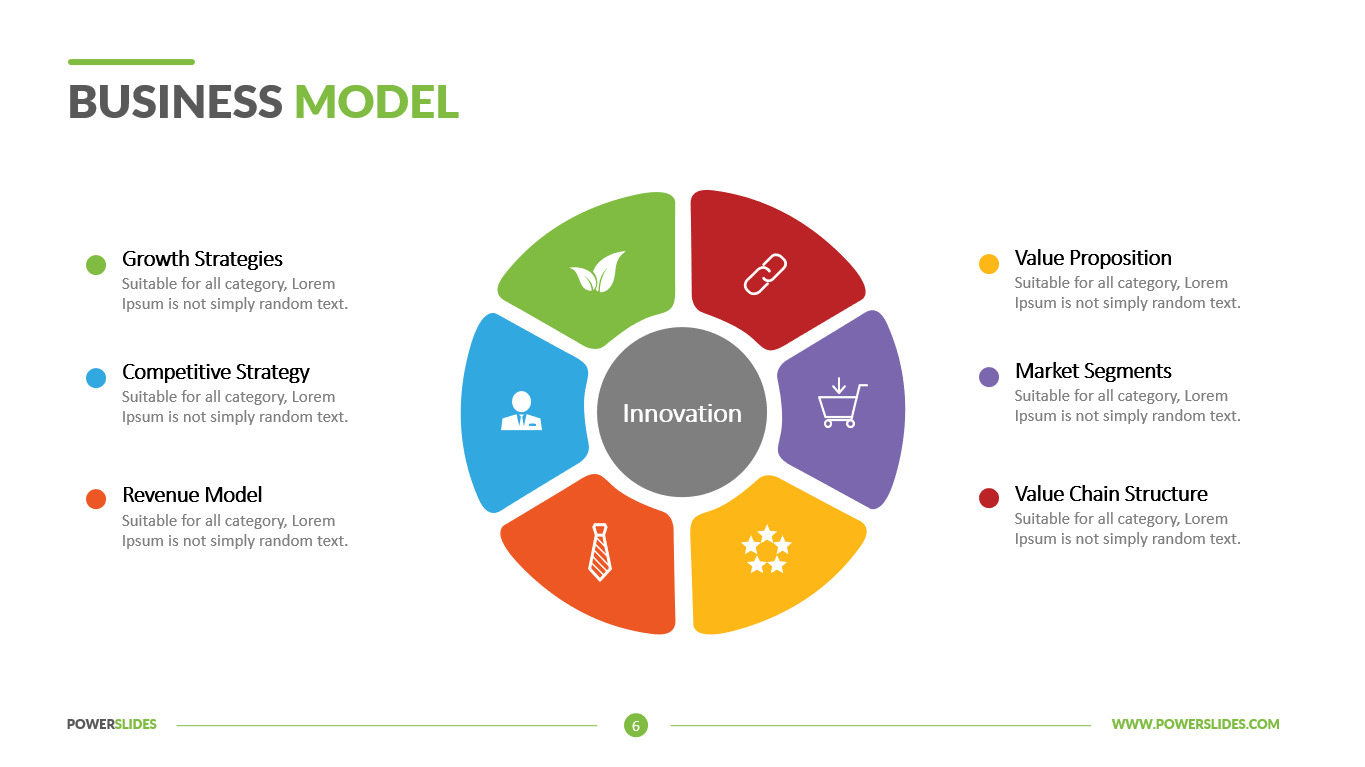


Figure 2- System parts image

# **API List**

List of APIs that this system provided:

1. [API](file:///C:\Users\moslem.babaee\Downloads\مستند_SV_-_1.1.docx#_اعتبارسنجی_گواهی) 1
2. [API 2](file:///C:\Users\moslem.babaee\Downloads\مستند_SV_-_1.1.docx#_اعتبارسنجی_فایل_pdf)

## **API 1**

What does it do?

|  |  |
| --- | --- |
| Title | Title of API |
| URL | /api/Controller/Action |
| HTTP Method | HTTP Post |

* 1. **Input and output parameters**

**Input parameters:**

|  |  |  |
| --- | --- | --- |
| Parameter name | type | Explanation |
| BirthDate | nullable | yyyy-MM-ddThh:mm:ssZ |
| Stage | \* | This data is from list below: ListItem1, ListItem2, ListItem3 |

Table 2- Input parameters of API1

**Output parameters:**

|  |  |  |
| --- | --- | --- |
| Parameter name | Explanation | Value |
| BirthDate | Birthdate of user | DateTime |
| Stage | Stage of user | Int : 1-2-3 |

Table 3-Output parameters of API1

**Explanation of output parameters:**

* **BirthDate**: this is when user born.
* **Stage:** This field explain the stage of user in system
  1. **Input and output samples**

**Input Sample:**

{

  " BirthDate ": "2022-08-13T07:09:34.435Z",

  " Stage ": "ListItem1 ",

}

**Output sample:**

{

  " BirthDate ": "2022-08-13T07:09:34.435Z",

  " Stage ": "1 ",

}