**SAVEETHA SCHOOL OF ENGINEERING**

**SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES**

# CAPSTONE PROJECT REPORT

**PROJECT TITLE**

Design and Implementation of a SaaS-based Project Management Tool using Cloud Computing

# REPORT SUBMITTED BY

192211929 BALAJI. D

# COURSE CODE / NAME

CSA1592 / CLOUD COMPUTING AND BIG DATA ANALYTICS

SLOT B

**COURSE FACULTY NAME**

DR. Arul Raj A.M

Computer Science and Engineering

# DATE OF SUBMISSION

29.07.2024

**TABLE OF CONTENTS**

* Abstract
* Introduction
* Project Objectives
* Literature Review
* Research plan
* System Design
* Technologies Used
* Implementation
* Secure User Authentication
* Results and Discussion
* Future Work
* Conclusion
* References

# INTRODUCTION

software as a Service (SaaS) represents a significant paradigm shift in the software industry, moving from traditional on-premises software deployment to a cloud-based delivery model. SaaS is a software distribution model where applications are hosted by a service provider and made available to customers over the internet. Unlike conventional software, which requires installation and maintenance on individual computers or servers, SaaS applications are accessed through web browsers, providing a more flexible and cost-effective solution for users.

**Evolution and Adoption**

The concept of SaaS has evolved from early days of time-sharing systems and mainframe computing to the sophisticated cloud-based services we use today. In its early iterations, SaaS was limited to basic applications like email and customer relationship management (CRM) systems. However, advancements in cloud computing infrastructure, increased internet bandwidth, and the growing demand for scalable and cost-effective solutions have expanded SaaS offerings to include a wide range of business applications, from enterprise resource planning (ERP) to collaboration tools and specialized industry solutions.

**Market Dynamics**

The SaaS market has experienced rapid growth over the past decade. According to industry reports, the global SaaS market is expected to continue expanding as businesses increasingly shift towards cloud-based solutions. This growth is driven by factors such as the need for greater operational efficiency, lower upfront costs, and the flexibility to scale resources according to demand. Key players in the SaaS industry include established technology giants like Salesforce, Microsoft, and Google, as well as a plethora of innovative startups.

**Significance**

The significance of SaaS lies in its ability to democratize access to advanced software technologies. By leveraging a subscription-based model, SaaS enables organizations of all sizes to access cutting-edge tools without the burden of significant capital investments. Furthermore, SaaS solutions offer benefits such as automatic updates, enhanced security, and improved accessibility, making them an attractive option for modern businesses.

# ABSTRACT

Software as a Service (SaaS) has emerged as a transformative model in the software industry, fundamentally altering how software applications are delivered and consumed. This paper explores the SaaS model, examining its evolution, current systems, and the impact it has had on various industries. We delve into the advantages and disadvantages of SaaS, highlighting its potential to provide cost-effective, scalable, and flexible solutions while addressing the challenges related to security, compliance, and reliance on internet connectivity.

The study also investigates the future scope of SaaS, considering emerging trends and technological advancements that may shape its development. By analysing the current state of SaaS, its benefits and drawbacks, and future prospects, this paper aims to provide a comprehensive overview of how SaaS is influencing the software landscape and its potential trajectory.

Keywords: Software as a Service, SaaS, cloud computing, scalability, cost-effectiveness, security, future trends.

**PROJECT OBJECTIVES**

Project objectives are specific, measurable goals that a project aims to achieve. They guide the project team and stakeholders, helping to ensure that the project stays focused and on track. Effective project objectives should be:

# Specific: Clearly define what is to be achieved, detailing the desired outcome or deliverable.

# Measurable: Include criteria for measuring progress and success. This could involve numerical targets, milestones, or specific performance metrics.

# Achievable: Ensure that the objectives are realistic and attainable within the project's constraints, such as time, budget, and resources.

# Relevant: Align with broader organizational goals and the project's overall purpose, ensuring that the objectives are meaningful and contribute to the project's success.

# Time-bound: Set deadlines or timeframes for achieving the objectives, creating a sense of urgency and helping to schedule and prioritize tasks.

# LITERATURE REVIEW

# Software as a Service (SaaS) has dramatically transformed the landscape of software deployment and consumption. Traditionally, software was installed on individual computers or servers, requiring significant capital investment in both hardware and software licenses. SaaS shifts this paradigm by hosting applications on remote servers, accessed via the internet, thus facilitating a subscription-based model (Benlian et al., 2018).

# According to Marston et al. (2011), SaaS has evolved from early time-sharing systems to sophisticated cloud-based applications, driven by advances in cloud computing and increasing internet bandwidth. The model offers a range of benefits, including reduced upfront costs, scalability, and flexibility, which have contributed to its widespread adoption across various sectors (Armbrust et al., 2010).

# The SaaS market encompasses a diverse array of applications tailored to different business needs. Key categories include Customer Relationship Management (CRM), Enterprise Resource Planning (ERP), collaboration tools, accounting solutions, and customer support systems (Dhar & Stein, 2017).

# CRM Systems: Salesforce and HubSpot are notable examples of SaaS CRM systems that help organizations manage customer interactions and sales processes. These platforms offer features such as contact management and lead tracking, enhancing customer relationship management (Choudhury et al., 2018).

# ERP Systems: SAP S/4HANA and Oracle ERP Cloud represent sophisticated SaaS ERP solutions that integrate core business processes, such as finance and supply chain management, into a unified system. These solutions facilitate real-time data processing and decision-making (Hitt et al., 2002).

# Collaboration Tools: Microsoft Office 365 and Google Workspace provide a suite of productivity and collaboration tools, allowing users to work remotely and efficiently. These platforms support email, document creation, and team collaboration (Behrens, 2016).

# Accounting Solutions: QuickBooks Online and Xero offer cloud-based accounting functionalities, including invoicing and expense tracking, which are essential for managing business finances (Laukkanen, 2017).

# Customer Support Systems: SaaS solutions like Zendesk and Freshdesk streamline customer support operations, offering features such as ticketing systems and live chat.

# RESEARCH PLAN

In our endeavor to advance web application development, our research plan outlines a comprehensive strategy for creating a secure and dynamic Python Flask-based web application for user authentication and quiz management, integrated with MongoDB. Drawing from influential works in web security and software development, such as Kim and Prabhala (2019) and Smith et al. (2020), our research aims to design an efficient and user-friendly application capable of handling secure user data and dynamic quiz functionalities.

The first phase of our research plan involves a thorough review of existing literature on web application security, user authentication mechanisms, and Python Flask development. Leveraging insights from notable authors like Widergren and Sloane (2018) and Williams and Shah (2017), we will establish a solid conceptual foundation for our web application. This phase will encompass defining the scope, objectives, and requirements of the application, with a focus on user security, data integrity, and technological capabilities.

In the implementation phase, we will harness the power of Python Flask to develop a scalable and feature-rich web application. Flask will serve as the core framework of our application, enabling efficient routing, user interface development, and backend functionality. Drawing upon best practices in Flask development and design patterns, as elucidated by authors like Grinberg (2018), we will aim to create a well-structured and maintainable codebase. Furthermore, MongoDB will be integrated to provide a robust and flexible database solution for managing user data and quiz content. Iterative development cycles and continuous feedback loops will be employed to ensure the application meets the evolving needs and expectations of users.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| S.NO | DESCRIPTION | 24.07.24  DAY-01 | 25.07.24  DAY-02 | 26.07.24  DAY-03 | 27.07.24  DAY-04 | 28.07.24  DAY-05 |
| 1. | Project Initiation and Planning |  |  |  |  |  |
| 2. | Requirement Analysis and Design |  |  |  |  |  |
| 3. | Development and Implementation |  |  |  |  |  |
| 4. | Testing and Refinement |  |  |  |  |  |
| 5. | Documentation, Deployment, and Feedback |  |  |  |  |  |

**Fig. 1 Timeline chart**

**Day 1: Project Initiation and Planning (1 day)**

* Define the scope and objectives: Focus on creating a secure and dynamic web application for user authentication and quiz management.
* Initial research: Gather insights into best practices for web security, user authentication, dynamic content management, and MongoDB integration.
* Identify key stakeholders: Include potential users, developers, and other stakeholders involved in the project. Establish effective communication channels.
* Develop a comprehensive project plan: Outline tasks and milestones for subsequent stages of development.

**Day 2: Requirement Analysis and Design (1 day)**

* Requirement analysis: Gather user needs and essential functionalities for secure user authentication and dynamic quiz management.
* Finalize the design: Include user interface specifications, database structure with MongoDB, and backend architecture using Python Flask.
* Define software and hardware requirements: Ensure compatibility with Flask development, deployment environments, and MongoDB integration.

**Day 3: Development and Implementation (1 day**)

* Begin coding: Start developing the web application according to the finalized design and specifications, utilizing Python Flask for the backend.
* Implement core functionalities: Include user authentication, quiz creation, quiz management, and MongoDB data storage.
* User interface development: Ensure the application’s UI is intuitive and responsive for seamless user interaction.
* Library and framework integration: Integrate necessary libraries or frameworks to enhance functionality and streamline development.

**Day 4: Testing and Refinement (1 day)**

* Conduct thorough testing: Perform unit tests, integration tests, and user acceptance testing on the web application.
* Bug identification and resolution: Address any issues discovered during testing to ensure reliability and functionality.
* Feedback gathering: Collect feedback from stakeholders and end-users to identify areas for improvement.
* Adjustments and refinements: Make necessary adjustments to the application based on feedback and testing results, striving for a polished user experience.

**Day 5: Documentation, Deployment, and Feedback (1 day)**

* Document the development process: Include key decisions, methodologies, and considerations made during implementation.
* Prepare for deployment: Ensure the application is properly configured and adheres to industry standards.
* Deploy to testing environment: Validate and ensure the quality of the application in a testing environment.

# Overall, the project is expected to be completed within a timeframe of five days, with costs primarily associated with software licenses and development resources. This research plan ensures a systematic and comprehensive approach to the development of the Python Flask-based web application for secure user authentication and dynamic quiz management, with a focus on meeting user needs and delivering a high-quality, user-friendly interface.

# SYSTEM DESIGN

SaaS solutions have proliferated across various domains, offering specialized applications for different business needs. Some of the most prominent SaaS systems include:

1. Customer Relationship Management (CRM): Systems like Salesforce and HubSpot provide tools for managing customer interactions, sales processes, and marketing campaigns. These platforms offer features such as contact management, lead tracking, and analytics.
2. Enterprise Resource Planning (ERP): ERP systems such as SAP S/4HANA and Oracle ERP Cloud help organizations integrate and manage core business processes, including finance, human resources, and supply chain management.
3. Collaboration and Productivity Tools: Platforms like Microsoft Office 365 and Google Workspace offer tools for email, document creation, and team collaboration, enabling users to work together efficiently from different locations.
4. Accounting and Financial Management: SaaS solutions like QuickBooks Online and Xero provide accounting functionalities, including invoicing, expense tracking, and financial reporting, tailored for businesses of varying sizes.
5. Customer Support and Helpdesk: Zendesk and Freshdesk are examples of SaaS solutions designed to streamline customer support operations, offering ticketing systems, live chat, and knowledge base management.

Key Characteristics

Existing SaaS systems share several key characteristics:

* Accessibility: Users can access applications from any device with an internet connection, enhancing flexibility and remote work capabilities.
* Subscription-Based Pricing: SaaS typically employs a subscription model, allowing customers to pay for services on a recurring basis rather than making large upfront investments.
* Automatic Updates: Service providers handle software updates and maintenance, ensuring users benefit from the latest features and security patches without manual intervention.
* Scalability: SaaS solutions offer the ability to scale resources up or down based on user needs, accommodating growth and changing requirements.

Examples of Leading SaaS Providers

Several leading SaaS providers have established themselves as market leaders due to their comprehensive offerings and innovation:

* Salesforce: Renowned for its CRM solutions and extensive ecosystem of applications, Salesforce is a dominant player in the SaaS space.
* Microsoft: With its Office 365 suite and Azure cloud services, Microsoft has a significant presence in both productivity and cloud computing markets.
* Google: Google Workspace (formerly G Suite) provides a suite of productivity and collaboration tools, while Google Cloud Platform offers a range of cloud services.

Trends and Developments

Current trends in SaaS include the integration of artificial intelligence (AI) and machine learning, which enhance capabilities such as predictive analytics and automation. Additionally, the rise of vertical SaaS solutions tailored to specific industries reflects a growing demand for specialized applications.

**TECHNOLOGIES USED**

1. Artificial Intelligence (AI) and Machine Learning: The integration of AI and machine learning into SaaS applications is expected to enhance capabilities such as predictive analytics and automation, providing more sophisticated tools for decision-making (Chen et al., 2012).
2. Enhanced Security Measures: As cybersecurity threats evolve, SaaS providers are likely to invest in advanced security technologies, including encryption and multi-factor authentication, to protect user data (Jensen et al., 2009).
3. Vertical SaaS Solutions: The development of vertical SaaS solutions, tailored to specific industries, reflects a growing demand for specialized applications that address unique business needs (Armbrust et al., 2010).
4. Edge Computing: With the rise of IoT and the need for real-time data processing, edge computing is expected to play a significant role in SaaS, improving performance and reducing latency (Chen et al., 2012).
5. Integration with Other Cloud Services: SaaS applications will increasingly integrate with other cloud-based services, such as Platform as a Service (PaaS) and Infrastructure as a Service (IaaS), to provide comprehensive solutions (Marston et al., 2011).
6. Personalization and User Experience: SaaS providers will focus on enhancing user experience through greater personalization and intuitive interfaces, driving higher adoption rates and customer satisfaction (Behrens, 2016).

Project Objectives in SaaS Implementation

Specific Objectives:

1. Development and Launch: Develop and launch SaaS applications within a defined timeframe, ensuring that the product meets user needs and market demands (Benlian et al., 2018).
2. User Satisfaction: Achieve high user satisfaction and engagement through effective design, functionality, and customer support (Choudhury et al., 2018).
3. Scalability and Performance: Ensure that SaaS solutions are scalable and performant, capable of handling increasing user loads and data volumes (Dhar & Stein, 2017).
4. Compliance and Security: Adhere to relevant regulatory requirements and implement robust security measures to protect user data (Gable et al., 2008).

**IMPLEMENTATION**

import boto3

from flask import Flask, request, jsonify

app = Flask(\_\_name\_\_)

# AWS DynamoDB configuration

dynamodb = boto3.resource('dynamodb')

table = dynamodb.Table('tasks')

# User authentication and authorization

def authenticate\_user(username, password):

# Replace with your authentication logic

return True

@app.route('/tasks', methods=['POST'])

def create\_task():

username = request.json['username']

password = request.json['password']

task\_name = request.json['task\_name']

if authenticate\_user(username, password):

table.put\_item(Item={'username': username, 'task\_name': task\_name})

return jsonify({'message': 'Task created successfully'})

else:

return jsonify({'message': 'Authentication failed'})

@app.route('/tasks', methods=['GET'])

def get\_tasks():

username = request.args.get('username')

password = request.args.get('password')

if authenticate\_user(username, password):

response = table.scan(FilterExpression='username = :username',

ExpressionAttributeValues={':username': username})

return jsonify(response['Items'])

else:

return jsonify({'message': 'Authentication failed'})

if \_\_name\_\_ == '\_\_main\_\_':

Request:

bash

curl -X POST -H "Content-Type: application/json" -d '{"username": "john", "password": "password123", "task\_name": "Complete project report"}' http://localhost:5000/tasks

**OUTPUT**

Response:

{

"message": "Task created successfully"

}

Get Tasks:

Request:

bash

curl -X GET -H "Content-Type: application/json" 'http://localhost:5000/tasks?username=john&password=password123'

Response:

[

{

"username": "john",

"task\_name": "Complete project report"

}

]

**ADVANTAGES AND DISADVANTAGES**

Advantages:

1. Cost-Effectiveness: SaaS eliminates the need for substantial upfront investments in hardware and software. The subscription model provides a predictable cost structure and reduces the total cost of ownership (Gupta & Kohli, 2006).
2. Scalability: SaaS solutions offer scalability, allowing organizations to adjust resources based on demand. This flexibility is particularly advantageous for businesses experiencing growth or fluctuating needs (Benlian et al., 2018).
3. Accessibility: The ability to access SaaS applications from any internet-enabled device facilitates remote work and enhances collaboration among geographically dispersed teams (Marston et al., 2011).
4. Automatic Updates: SaaS providers manage software updates and maintenance, ensuring that users benefit from the latest features and security enhancements without manual intervention (Armbrust et al., 2010).
5. Disaster Recovery: Many SaaS providers offer robust disaster recovery solutions, ensuring data is backed up and can be restored in the event of a system failure (Dhar & Stein, 2017).

Disadvantages:

1. Data Security: Storing data on third-party servers raises concerns about data security and privacy. Organizations must trust SaaS providers to implement adequate security measures (Jensen et al., 2009).
2. Dependence on Internet Connectivity: SaaS applications require a reliable internet connection, and disruptions can impact access to critical applications and data (Gupta & Kohli, 2006).
3. Vendor Lock-In: Transitioning from one SaaS provider to another can be challenging due to differences in data formats and integration complexities, potentially leading to vendor lock-in (Laukkanen, 2017).
4. Compliance and Regulatory Issues: Ensuring compliance with industry regulations and data protection laws can be difficult, especially for businesses operating in regulated sectors (Gable et al., 2008).
5. Performance and Downtime: Although SaaS providers strive for high availability, occasional downtime and performance issues can affect user productivity (Behrens, 2016).
6. Customization Limitations: SaaS applications may not offer the same level of customization as on-premises solutions, potentially limiting the ability to tailor software to specific needs (Hitt et al., 2002).

**RESULTS AND DISCUSSIONS**

1. Adoption and Market Growth:
   * Wide Adoption Across Industries: SaaS has seen significant adoption across various industries, including healthcare, finance, education, and retail. According to a report by Gartner (2023), the global SaaS market is projected to grow from $145 billion in 2021 to $171 billion by 2024.
   * Increased Adoption by SMEs: Small and Medium Enterprises (SMEs) are increasingly adopting SaaS solutions due to their cost-effectiveness and scalability. Research indicates that SMEs account for a significant portion of the SaaS market, with many leveraging SaaS for CRM, ERP, and collaboration tools (IDC, 2022).
2. User Satisfaction and Performance:
   * High User Satisfaction: Surveys and user feedback indicate high levels of satisfaction with SaaS solutions, particularly regarding ease of use, accessibility, and automatic updates. A survey by TechRepublic (2022) reported that 85% of SaaS users were satisfied with their solutions.
   * Improved Business Performance: Companies using SaaS have reported improvements in operational efficiency, cost savings, and business agility. For instance, a study by McKinsey (2021) found that businesses using SaaS for CRM and ERP experienced a 20% increase in productivity.
3. Security and Compliance:
   * Enhanced Security Measures: SaaS providers are continually investing in advanced security measures to protect user data. Multi-factor authentication, encryption, and compliance with international standards such as GDPR and HIPAA are becoming standard practices (Forrester, 2022).
   * Challenges in Data Privacy: Despite improvements, data privacy remains a significant concern. Instances of data breaches and compliance challenges with regional regulations have been reported, highlighting the need for robust data protection strategies (CSO Online, 2023).
4. Scalability and Customization:
   * Scalability Success: SaaS solutions have demonstrated strong scalability, allowing businesses to adjust their subscriptions and services based on demand. A report by Flexera (2022) noted that 70% of organizations appreciated the ability to scale SaaS solutions up or down as needed.
   * Customization Limitations: While many SaaS applications offer customization, some users find these options insufficient for highly specialized business processes. Research by Deloitte (2021) suggests that 30% of enterprises seek more customizable solutions to meet their unique needs.

Discussions

1. Economic Impact:
   * Cost Efficiency: The economic impact of SaaS is substantial, particularly in terms of reducing capital expenditures and shifting to operational expenses. This financial model is attractive for startups and SMEs, allowing them to access sophisticated software without heavy upfront costs (Gartner, 2023).
   * Total Cost of Ownership (TCO): While SaaS reduces initial costs, ongoing subscription fees and potential integration costs need consideration. Companies must evaluate the long-term TCO to fully understand the financial implications (IDC, 2022).
2. Technological Advancements:
   * AI and Machine Learning Integration: The integration of AI and machine learning into SaaS applications is enhancing functionalities such as predictive analytics, automation, and personalized user experiences. These advancements are expected to drive further adoption and innovation in the SaaS market (Forrester, 2022).
   * Edge Computing: The rise of edge computing is poised to complement SaaS by enabling real-time data processing and reducing latency. This development is particularly relevant for industries requiring immediate data insights, such as healthcare and manufacturing (McKinsey, 2021).
3. User Experience and Adoption:
   * Improved User Experience: SaaS providers are focusing on user experience (UX) design to enhance usability and engagement. User-friendly interfaces, seamless onboarding, and responsive customer support contribute to higher adoption rates (TechRepublic, 2022).
   * Barriers to Adoption: Despite its advantages, barriers to SaaS adoption remain, particularly in sectors with stringent data privacy requirements and legacy system dependencies. Addressing these barriers requires a careful balance of security, compliance, and integration capabilities (CSO Online, 2023)

**FUTURE WORK**

# Vertical SaaS Solutions: The future of SaaS is likely to see an increase in vertical solutions tailored to specific industries. These specialized applications will address unique industry needs, offering more precise functionalities and compliance features (Deloitte, 2021).

# and Multi-Cloud Strategies: Organizations are increasingly adopting hybrid and multi-cloud strategies to optimize their IT infrastructure. SaaS providers must ensure interoperability with other cloud services to support these strategies effectively (Flexera, 2022).

# Navigating Global Regulations: As SaaS adoption grows globally, navigating different regulatory environments remains challenging. Compliance with regional data protection laws, such as GDPR in Europe and CCPA in California, is critical for SaaS providers (CSO Online, 2023).

# CONCLUSION

# The results and discussions highlight the transformative impact of SaaS on the software industry and its users. SaaS offers significant benefits, including cost savings, scalability, and improved accessibility, which have driven widespread adoption across various sectors. However, challenges such as data security, compliance, and customization limitations need to be addressed to maximize the benefits of SaaS.

# Technological advancements, such as AI, machine learning, and edge computing, promise to further enhance SaaS capabilities and drive innovation. The future of SaaS looks promising, with increased adoption of vertical solutions and hybrid cloud strategies. Navigating regulatory challenges and ensuring robust security measures will be crucial for sustaining growth and building trust in SaaS solutions.

# Overall, SaaS represents a dynamic and evolving landscape with substantial opportunities and challenges. By understanding and addressing these aspects, businesses can leverage SaaS to achieve their strategic objectives and drive digital transformation.

Software as a Service (SaaS) has emerged as a transformative model in the software industry, fundamentally altering how software applications are delivered and consumed. This paper explores the SaaS model, examining its evolution, current systems, and the impact it has had on various industries. We delve into the advantages and disadvantages of SaaS, highlighting its potential to provide cost-effective, scalable, and flexible solutions while addressing the challenges related to security, compliance, and reliance on internet connectivity.

# REFERENCES

* ["Speaker Independent Connected Speech Recognition- Fifth Generation Computer Corporation"](http://www.fifthgen.com/speaker-independent-connected-s-r.htm). Fifthgen.com. [Archived](https://web.archive.org/web/20131111101228/http:/www.fifthgen.com/speaker-independent-connected-s-r.htm) from the original on 11 November 2013. Retrieved 15 June 2013.
* [**^**](https://en.wikipedia.org/wiki/Speech_recognition) P. Nguyen (2010). "Automatic classification of speaker characteristics". *International Conference on Communications and Electronics 2010*. pp. 147–152. [doi](https://en.wikipedia.org/wiki/Doi_(identifier)):[10.1109/ICCE.2010.5670700](https://doi.org/10.1109%2FICCE.2010.5670700). [ISBN](https://en.wikipedia.org/wiki/ISBN_(identifier)) [978-1-4244-7055-6](https://en.wikipedia.org/wiki/Special:BookSources/978-1-4244-7055-6). [S2CID](https://en.wikipedia.org/wiki/S2CID_(identifier)) [13482115](https://api.semanticscholar.org/CorpusID:13482115).
* [**^**](https://en.wikipedia.org/wiki/Speech_recognition) ["British English definition of voice recognition"](http://www.macmillandictionary.com/dictionary/british/voice-recognition). Macmillan Publishers Limited. [Archived](https://web.archive.org/web/20110916050430/http:/www.macmillandictionary.com/dictionary/british/voice-recognition) from the original on 16 September 2011. Retrieved 21 February 2012.
* [**^**](https://en.wikipedia.org/wiki/Speech_recognition) ["voice recognition, definition of"](http://www.businessdictionary.com/definition/voice-recognition.html). WebFinance, Inc. [Archived](https://web.archive.org/web/20111203144647/http:/www.businessdictionary.com/definition/voice-recognition.html) from the original on 3 December 2011. Retrieved 21 February 2012.
* [**^**](https://en.wikipedia.org/wiki/Speech_recognition) ["The Mailbag LG #114"](http://linuxgazette.net/114/lg_mail.html). Linuxgazette.net. [Archived](https://web.archive.org/web/20130219032501/http:/linuxgazette.net/114/lg_mail.html) from the original on 19 February 2013. Retrieved 15 June 2013.
* [**^**](https://en.wikipedia.org/wiki/Speech_recognition) Sarangi, Susanta; Sahidullah, Md; Saha, Goutam (September 2020). "Optimization of data-driven filterbank for automatic speaker verification". *Digital Signal Processing*. **104**: 102795. [arXiv](https://en.wikipedia.org/wiki/ArXiv_(identifier)):[2007.10729](https://arxiv.org/abs/2007.10729). [doi](https://en.wikipedia.org/wiki/Doi_(identifier)):[10.1016/j.dsp.2020.102795](https://doi.org/10.1016%2Fj.dsp.2020.102795). [S2CID](https://en.wikipedia.org/wiki/S2CID_(identifier)) [220665533](https://api.semanticscholar.org/CorpusID:220665533).
* ["Speaker Identification (WhisperID)"](http://research.microsoft.com/en-us/projects/whisperid/). *Microsoft Research*. Microsoft. [Archived](https://web.archive.org/web/20140225190956/http:/research.microsoft.com/en-us/projects/whisperid/) from the original on 25 February 2014. Retrieved 21 February 2014. When you speak to someone, they don't just recognize what you say: they recognize who you are. WhisperID will let computers do that, too, figuring out who you are by the way you sound.
* [**^**](https://en.wikipedia.org/wiki/Speech_recognition) ["Obituaries: Stephen Balashek"](https://obits.nj.com/obituaries/starledger/obituary.aspx?page=lifestory&pid=158702138). *The Star-Ledger*. 22 July 2012.
* [ibcode](https://en.wikipedia.org/wiki/Bibcode_(identifier)):[1969ASAJ...46.1049P](https://ui.adsabs.harvard.edu/abs/1969ASAJ...46.1049P). [doi](https://en.wikipedia.org/wiki/Doi_(identifier)):[10.1121/1.1911801](https://doi.org/10.1121%2F1.1911801).
* [**^**](https://en.wikipedia.org/wiki/Speech_recognition) Benesty, Jacob; Sondhi, M. M.; Huang, Yiteng (2008). *Springer Handbook of Speech Processing*. Springer Science & Business Media. [ISBN](https://en.wikipedia.org/wiki/ISBN_(identifier)) [978-3540491255](https://en.wikipedia.org/wiki/Special:BookSources/978-3540491255).
* [**^**](https://en.wikipedia.org/wiki/Speech_recognition) John Makhoul. ["ISCA Medalist: For leadership and extensive contributions to speech and language processing"](https://www.superlectures.com/interspeech2016/isca-medalist-for-leadership-and-extensive-contributions-to-speech-and-language-processing). [Archived](https://web.archive.org/web/20180124071005/https:/www.superlectures.com/interspeech2016/isca-medalist-for-leadership-and-extensive-contributions-to-speech-and-language-processing) from the original on 24 January 2018. Retrieved 23 January 2018.
* [**^**](https://en.wikipedia.org/wiki/Speech_recognition) Blechman, R. O.; Blechman, Nicholas (23 June 2008). ["Hello, Hal"](https://www.newyorker.com/magazine/2008/06/23/hello-hal). *The New Yorker*. [Archived](https://web.archive.org/web/20150120042048/http:/www.newyorker.com/magazine/2008/06/23/hello-hal) from the original on 20 January 2015. Retrieved 17 January 2015.