



Project handbook

Smart-Wand

1

Version 4
Project manager: Thomas Toferer

Date: 16/12/25

Content

1 Project plans	5
1.1 Project Assignment.....	6
1.2 Project Objectives (objectives, non-objectives)	6
1.3 Description of Pre- and Post Project Phase	7
1.4 Project Environment Analysis.....	8
1.5 Relationship to Other Projects andthe Organisations's Strategy.....	10
1.6 Project Organisation Chart	11
1.7 Plan of Objects of Consideration of the Project.....	13
1.8 Work Breakdown Structure (WBS).....	14
1.9 Project Work-Package Specification	15
1.1 Spell Detection	16
1.10 1.2 Communication.....	18
1.11 1.3 Setting up Hardware.....	19
1.12 1.4 Documentation.....	20
1.13 1.5 Web-Dashboard.....	21
1.14 1.6 Design of the Wand (3D Model).....	22
1.15 Project Responsibility Matrix	23
1.16 Milestoneplan.....	25
1.17 Project Bar Chart	26
1.18 Resource Plan	27
1.19 Project Cost Plan	28
1.20 Project Communication	29
1.21 Project „Rules“	30
1.22 Project Risk Analysis	31
1.23 Project Documentation	32
2 Project Start	33
2.1 Minutes– Project Start	34
2.1.1 Project Start Workshop	34
2.1.2 Follow-up Workshop.....	34
2.1.3 Project Owner Meeting	34
3 Project Co-ordination.....	34
3.1 Approval of Work-packages	35
3.2 Minutes – Project Co-ordination.....	35
4 Project Controlling	35
4.1 Project Status Report.....	36
4.2 Additional Project Status Reports	37
4.3 Minutes – Project controlling	37
4.3.1 Project Controlling Meetings	37

4.3.2	Project Owner Meetings.....	37
5	Project Close Down.....	37
5.1	Project Close Down Report.....	38
5.2	Minutes – Project Close Down	38
5.2.1	Project Close Down Workshop	39

Document versions

Distribution list

1 Project plans

1.1 Project Assignment

Smart-Wand 1		PROJECT-ASSIGNMENT																	
Project start event: <ul style="list-style-type: none"> • 2.12.2025 		Project start date: <ul style="list-style-type: none"> • 3.12.2025 																	
project close down event in terms of content: <ul style="list-style-type: none"> • End of Mai Formal project close down event: <ul style="list-style-type: none"> • End of Mai 		Project close down dates: <ul style="list-style-type: none"> • End of Mai 																	
Project objectives: <ul style="list-style-type: none"> • Creating useable Wand for Smart Devices • send commands to Home Assistant • Distinguish different Spells by Movement 		Non-objectives: <ul style="list-style-type: none"> • Expanding support for other Smart-Home management Systems (Google Home, Loxone etc) • Lifetime Support • Further Development 																	
Main tasks (Project phases): <ul style="list-style-type: none"> • Detecting Spells (Gyro, Acceleration) • Send commands to Home-Assist • Refine spell detection 		Project resources and costs*: <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">resource/type of cost</th> <th style="text-align: left;">unit</th> <th style="text-align: left;">Costs (€)</th> </tr> </thead> <tbody> <tr> <td>ESP 32</td> <td>1</td> <td>7€</td> </tr> <tr> <td>RaspberryPi 5</td> <td>1</td> <td>75€</td> </tr> <tr> <td>MPU6050</td> <td>1</td> <td>7,55€</td> </tr> <tr> <td>Shelly Plug</td> <td>1</td> <td>23,8€</td> </tr> </tbody> </table>			resource/type of cost	unit	Costs (€)	ESP 32	1	7€	RaspberryPi 5	1	75€	MPU6050	1	7,55€	Shelly Plug	1	23,8€
resource/type of cost	unit	Costs (€)																	
ESP 32	1	7€																	
RaspberryPi 5	1	75€																	
MPU6050	1	7,55€																	
Shelly Plug	1	23,8€																	
Project owner: <ul style="list-style-type: none"> • HTL-Wels 		Project manager: <ul style="list-style-type: none"> • Toferer Thomas 																	
Project team members: <ul style="list-style-type: none"> • Codoban Simon • Durkowitsch Laurenz 																			
HTL Wels (Project owner)		Toferer Thomas (Project manager)																	

* Possible categories of total Project budget:

Category A: cover of the shelly module

Category B: full cover of equipment

1.2 Project Objectives (objectives, non-objectives)

Smart Wand		
1	PROJECT OBJECTIVES	
Type of objective	Project objectives	Adjusted project objectives as of...
objectives: <ul style="list-style-type: none"> • Main objectives 	<ul style="list-style-type: none"> • Detect multiple Spells • Communicate with Home-Assist • Create a reliable and nice experience using the device 	
<ul style="list-style-type: none"> • Additional objectives 	<ul style="list-style-type: none"> • Add Support for another microcontroller • Web interface to manage Spells 	
<ul style="list-style-type: none"> • Non-objectives 	<ul style="list-style-type: none"> • Add Support for other IOT Management Solutions • Offer Lifetime Support • Further Developement 	<ul style="list-style-type: none"> • If there is enough time left before end of mai this objective could be transformed into an additional object.

1.3 Description of Pre- and Post Project Phase

Smart Wand

1

DESCRIPTION OF PRE- AND POST- PROJECT PHASE

1) Pre-project phase

What triggered the project?

- The project was proposed by a school assignment in ITP.
- The idea of the project was given to us by a former student of HTL Wels.
- The project idea was exactly what we wanted for an educational project. To develop this accessory, we need to build skills in various fields of computer science, offering a good overview of choosing the right career path in the future.

Relevant documents for the project („Minutes“, ... ONLY documents and no content necessary)

- SmartWand_PSP
- SmartWand_Goals
- Requirement Spezification(Pflichtenheft)

Experience from similar projects

- Xdj100s build (Toferer)
- FPV Drones (Durkowitsch)

2) Post-project phase

What will happen after the project has ended?

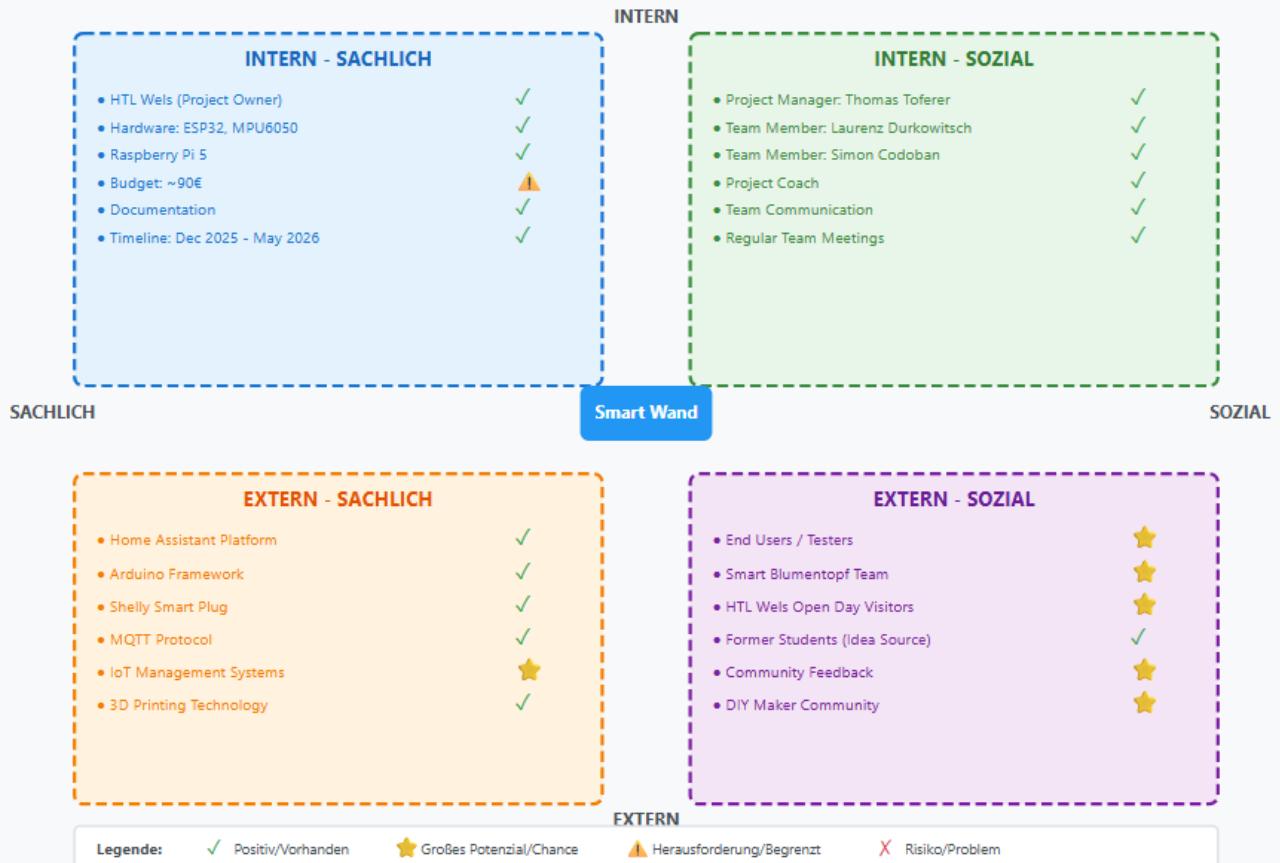
- Demonstration of the project on "open day" of HTL Wels 2026

1.4 Project Environment Analysis

Smart Wand
1

PROJECT ENVIRONMENT GRAPHIC

Project Environment Analysis - Smart Wand



Smart Wand
1

PROJECT ENVIRONMENT TABLE

Environment	Relationship (potential/conflict)	Measures	Who / when PSP Code
Internal	Investor	Regular updates on project progress	HTL Wels alltime
Tester/User	Feedback, Finding Bugs	Requirements gathering, early feedback	Everybody, Demo 1.1.1.4

1.5 Relationship to Other Projects and the Organisations's Strategy

Smart Wand

1

RELATIONSHIP TO OTHER PROJECTS

Programs/ Projects/	Relationship (potential/conflict)	Sanctions	Who / when WBS Code
Arduino Framework	We need those Open-Source Libraries to get sensor data. (get prohibited)	Development will be more difficult	1.1
Home Assistant	Managing IOT Devices and REST API calls. (get prohibited)	The project will have to use another IOT Management-Software	
Smart Blumentopf	Management Cooperation. Argument between PMs	Similar problems both projects could have, can't be solved with their knowledge.	

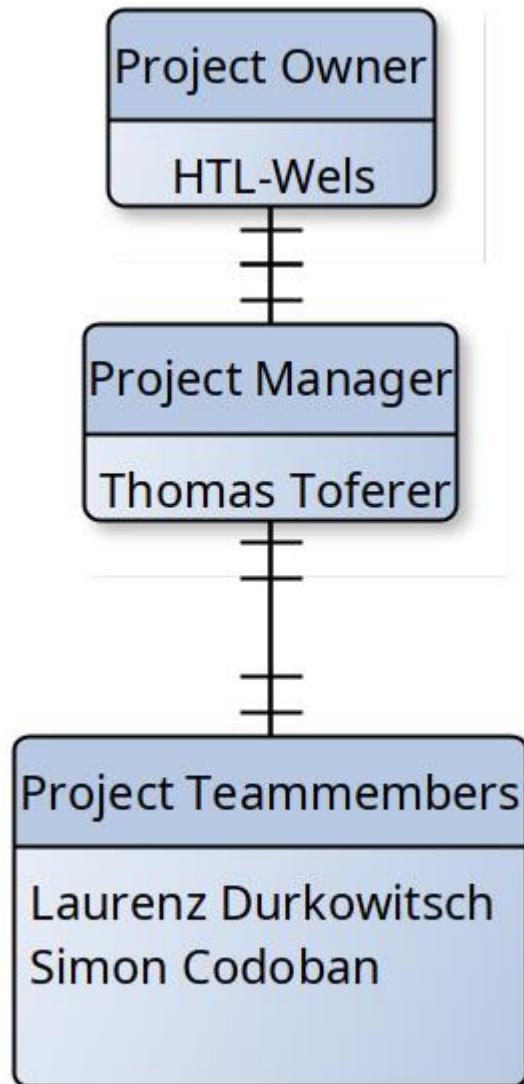
Smart Wand

1

CONNECTION TO THE ORGANISATION'S STRATEGY

Strategy	Description of connection/relationship
-	-

1.6 Project Organisation Chart



Smart Wand

1

PROJECT- ORGANISATION

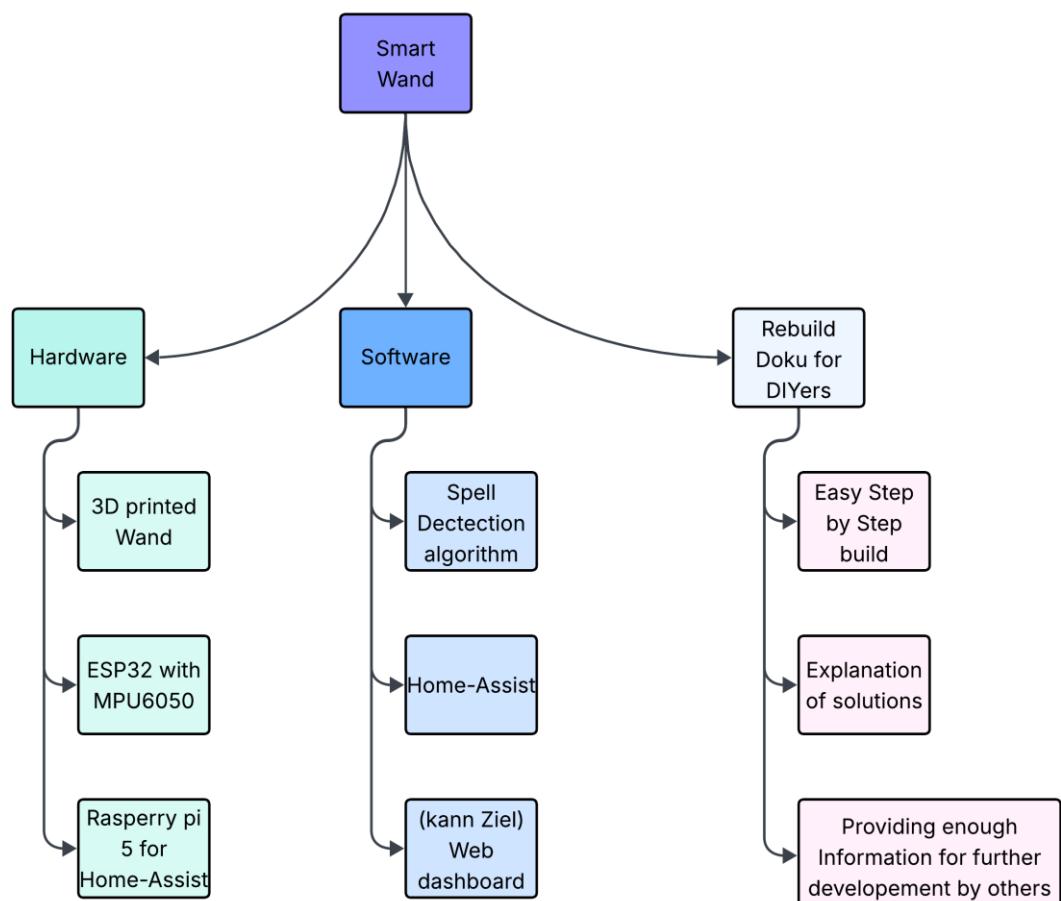
Role in Project	Field of duties/Skills	Name
Project owner	Advisor for technical problems. Investor	HTL Wels
Project manager	Coordination, Development of Spell Detection	Toferer Thomas
Project team members	1. Development of Rest API 2. Helping with smaller tasks	1. Laurenz Durkowitsch 2. Simon Codoban
Project members		

1.7 Plan of Objects of Consideration of the Project

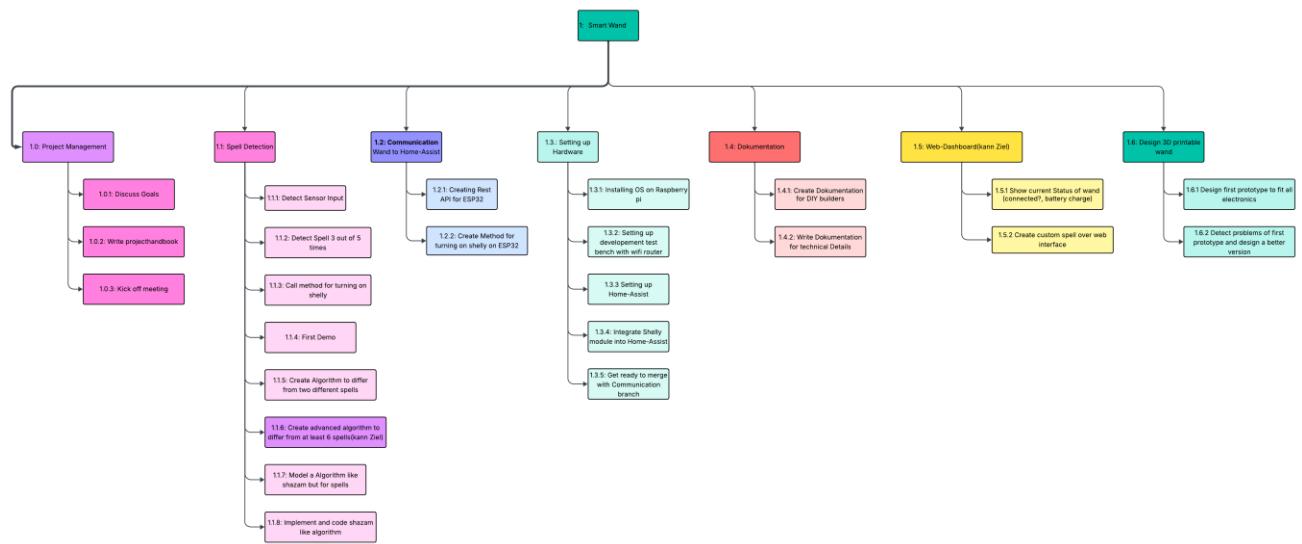
Smart Wand

1

PLAN OF OBJECTS OF CONSIDERATION



1.8 Work Breakdown Structure (WBS)



PSP_Smart-Wand_V2.
pdf

1.9 Project Work-Package Specification

Smart Wand

1

PROJECT WORK-PACKAGE SPECIFICATION

1.1 Spell Detection

Work Package	WP Content (What shall be done?)	Non-WP Content (What shall not be done?)	WP Result (What is achieved after WP was finished?)	Progress Measurement (How is progress measured?)
1.1.1 Detect Sensor Input	Implement sensor data collection from wand components Establish baseline for movement detection	Process sensor data for spell recognition Implement data filtering algorithms	Reliable sensor input detection system	Sensor data visible in development environment Consistent data stream verified for 5 minutes
1.1.2 Detect Single Spell 4 out of 5 times	Develop algorithm to recognize one specific spell movement Test and refine recognition accuracy	Implement multiple spell detection Add voice recognition capabilities	Reliable single spell detection system	Detection of spell 4 out of 5 times performing the movement
1.1.3 Call method for turning on Shelly	Implement API call to trigger Shelly device Integrate spell detection with device control	Create new Shelly devices Modify Shelly firmware	Successful triggering of Shelly device via spell	Shelly device responds to spell detection 4 out of 5 times
1.1.4 First Demo	Prepare demonstration of basic spell detection Showcase wand movement to device control functionality	Include multiple spells in the demo Demonstrate advanced features not yet implemented	Working demo showing basic functionality	Successful demo presented to project stakeholders Basic spell detection and device control demonstrated
1.1.5 Create Algorithm to	Develop algorithm to distinguish between two	Implement more than two spell	System that can reliably differentiate	Correct identification of

differ from two different spells	different spell movements Implement classification logic for multiple gestures	recognition Add machine learning components	between two spells	both spells 4 out of 5 times
1.1.6 Create advanced algorithm to differ from at least 6 spells	Expand algorithm to recognize at least 6 different spell movements Optimize performance for multiple spell recognition	Implement real-time feedback system Add voice command integration	Reliable multi-spell detection system	All 6 spells detected correctly 4 out of 5 times
1.1.7 Model a Algorithm like shazam but for spells	Research and model audio fingerprinting concepts for movement patterns Create conceptual framework for spell recognition	Implement full shazam-like system Deploy to production environment	Theoretical model for advanced spell recognition	Model documented and validated with sample data Concept approved by technical lead
1.1.8 Implement and code shazam like algorithm	Code the advanced spell recognition algorithm Integrate with existing detection system	Optimize for production deployment Add user interface components	Functional advanced spell recognition system	Algorithm successfully distinguishing spells with 90% accuracy Code integrated with main detection system

1.2 Communication

Work Package	WP Content (What shall be done?)	Non-WP Content (What shall not be done?)	WP Result (What is achieved after WP was finished?)	Progress Measurement (How is progress measured?)
1.2.1 Creating Rest API for ESP32	Develop REST API interface for ESP32 microcontroller Enable communication between wand and central system	Implement MQTT protocol in this phase Create web interface for API testing	Functional REST API on ESP32	API responds to basic HTTP requests Successful data transmission verified

1.2.2 Create Method for turning on shelly on ESP32	Implement specific API endpoint for Shelly device control Create command structure for device activation	Modify Shelly device firmware Implement multiple device control	Reliable method to control Shelly devices via API	Shelly device responds to API commands 4 out of 5 times Command structure documented
--	---	--	---	---

1.3 Setting up Hardware

Work Package	WP Content (What shall be done?)	Non-WP Content (What shall not be done?)	WP Result (What is achieved after WP was finished?)	Progress Measurement (How is progress measured?)
1.3.1 Installing OS on Raspberry pi	Install operating system on Raspberry Pi Configure basic system settings	Install application-specific software Configure network services	Functional Raspberry Pi with operational OS	OS successfully boots Basic system commands work
1.3.2 Setting up development test bench with wifi router	Configure isolated network environment for testing Set up necessary development tools	Configure production network settings Install final application software	Ready-to-use development environment	Test bench operational with network connectivity All team members can access the environment
1.3.3 Setting up Home-Assist	Install and configure Home Assistant platform Set up basic system structure	Integrate with Shelly devices yet Configure advanced automation	Operational Home Assistant instance	Home Assistant accessible via web interface Basic configuration completed
1.3.4 Integrate Shelly module into Home-Assist	Add Shelly device integration to Home Assistant Configure basic device control	Create complex automations Modify Shelly device settings directly	Shelly devices controllable through Home Assistant	Shelly devices appear in Home Assistant interface Basic on/off control working

1.3.5 Get ready to merge with Communication branch	Prepare hardware configuration for integration Document current setup for team reference	Implement communication features Modify existing communication code	Hardware setup ready for integration with communication system	Hardware configuration documented Readiness confirmed by both hardware and communication teams
--	--	---	--	--

1.4 Documentation

Work Package	WP Content (What shall be done?)	Non-WP Content (What shall not be done?)	WP Result (What is achieved after WP was finished?)	Progress Measurement (How is progress measured?)
1.4.1 Create Documentation for DIY builders	Document all steps required for DIY builders to replicate the project Create clear, step-by-step instructions	Include advanced troubleshooting beyond basic setup Document internal development processes	Complete DIY builder guide	Guide covers all necessary steps for project replication Guide validated by external tester
1.4.2 Write Documentation for technical Details	Document technical specifications and implementation details Create reference material for developers	Include user-facing instructions Document features not yet implemented	Comprehensive technical documentation	All technical components documented Documentation reviewed and approved by technical lead

1.5 Web-Dashboard

Work Package	WP Content (What shall be done?)	Non-WP Content (What shall not be done?)	WP Result (What is achieved after WP was finished?)	Progress Measurement (How is progress measured?)
1.5.1 Show current Status of wand	Display wand connection status Show battery percentage information Present connection status information	Allow spell triggering via dashboard Include advanced configuration options	Dashboard showing all key wand status information	All status parameters displayed correctly Information updates in real-time

1.5.2 Create custom spell over web interface	Implement interface for creating new spell patterns Allow users to define and save custom spells	Allow real-time spell execution from dashboard Implement advanced spell editing features	Functional interface for custom spell creation	Users can create and save at least one custom spell Spell data correctly transmitted to wand system
--	---	---	--	--

1.6 Design of the Wand (3D Model)

Work Package	WP Content (What shall be done?)	Non-WP Content (What shall not be done?)	WP Result (What is achieved after WP was finished?)	Progress Measurement (How is progress measured?)
1.6.1 Design first prototype to fit all electronics	Create 3D model of wand with space for all components Ensure ergonomic design while accommodating electronics	Create final production-ready design Optimize for mass manufacturing	First functional prototype design	3D model completed with all components fitting Design reviewed and approved by team
1.6.2 Detect problems of first prototype and design a better version	Test first prototype and identify issues Create improved design based on testing feedback	Create multiple design variations Finalize for production without further testing	Improved wand design addressing prototype issues	Issues from first prototype documented Improved design created and validated with component fit test

1.10 Project Responsibility Matrix

SmartWand
1

**PROJECT-
RESPONSIBILITY-
MATRIX**

WBS-Code	WP-Title	Roles & Environment		External project owner	Project owner	Project manager	Project assistant	Project team member 1	Project team member 2	Project team member 3	Project team member 4	Project team member 5	Project member 1	Project member 2	Project member 3	Project coach
		I	R													
1.0	Project Management	I	R				C	C								
1.0.1	Set goals for project	I	R				C	C								
1.0.2	Write Project-Handbook	I	R				C	C								
1.0.3	Kick-off meeting	I	R				C	C								
1.1	Spell Detection	I	R				C	I								
1.1.1	First spell detection	I	R				C	I								
1.1.1.1	Detect Sensor Input	I	R				I	I								
1.1.1.2	Detect spell 4 out of 5 time	I	R				I	I								
1.1.1.3	Call method for turning on shelly	I	R				C	I								
1.1.1.4	First Demo	I	R				C	C								
1.1.1.5	Create Algorithm for 2 spells	I	R				I	I								
1.1.2	Advanced Algorithm	I	R				I	C								
1.1.2.1	Model a Algorithm like shazam but for spells	I	R				I	I								
1.1.2.2	Implement and code shazam like Algorithm	I	R				I	I								
1.2	Communication	I	I				R	I								
1.2.1	Creating Rest API for ESP32	I	I				R	I								
1.2.2	Create Method for turning shelly on ESP32	I	I				R	I								
1.3	Setting up Hardware	I	I				R	C								
1.3.1	Installing OS on Raspberry pi	I	I				R	C								
1.3.2	Setting up development-bench	I	I				I	R								
1.3.3	Setting up Home-Assist	I	I				R	C								
1.3.4	Integrate Shelly module into Home-Assist	I	R				C	R								
1.3.5	Get ready to accept calls from ESP	I	I				R	C								
1.4	Dokumentation	I	C				I	R								
1.4.1	Create Docu for rebuilders	I	I				C	R								
1.4.2	Write technical documentation	I	C				C	R								
1.5	Webdashboard	I	I				R	C								
1.5.1	Show current status of wand	I	I				R	C								

1.5.2	Create custom spells over webdashboard		I	C		R	C						
1.6	Design wand		I	C		R	I						
1.6.1	Design first prototype		I	C		R	I						
1.6.2	Redesign first design		I	C		R	I						

Functions

RResponsible

CContribution

Ihas to be informed

1.11 Milestoneplan

<Project name>
 <Project no.>

MILESTONEPLAN

WBS-Code	Milestone	Plan date	Revised date	Actual date
	Setup of HomeAssist and addinional Hardeware	15 January		
	Spell Detection	1 Feburary		
	Communication from wand to HomeAssist	20 Feburary		
	Preperation for first Demo	1 March		

*In order of plan dates.

1.12 Project Bar Chart

1.13 Resource Plan

<Project name>
 <Project no.>

Resource Plan

WBS-Code	Phase/Work-package	Type of resource	Planned quantity (days)	Revised quantity (days)	Actual quantity (days)	Deviation (days)
	Detecting spells	Time, ESP32, MPU6050	30			
	Developing Rest API	Time	30			
	Configure Hardware with Home-Assist	Time, Raspberry pi,	15			
	Writing Dokumentation	Time	30			

1.14 Project Cost Plan

Smart Wand 1		PROJECT COST PLAN			
WBS-Code, WP-Title	Type of Cost	Planned cost	Revised cost	Actual cost	Deviation
	• Personnel	0	0	0	
	• Material	< 90€			
	• External services	0	0	0	
	• Other	0	0	0	
	Total	< 90€			
	• Personnel				
	• Material				
	• External services				
	• Other				
	Total				
	• Personnel				
	• Material				
	• External services				
	• Other				
	Total				
Project cost					

1.15 Project Communication

<Project name>
 <Project no.>

PROJECT- COMMUNICATION

Title	Objectives, Content	Participants	Schedule	Location
Project owner meeting	<ul style="list-style-type: none"> • project status • decisions • acceptance of progress report 	Project owner, Project manager		
Project controlling meeting	<ul style="list-style-type: none"> • project status • controlling of tasks, schedule, resources, costs • controlling of project environments • social controlling • prepare proposal for decision 	Project manager, Project team, Project coach		
Subteam meeting	<ul style="list-style-type: none"> • Coordination of subteams • Discussion of problems 	Subteam		
	•			

1.16 Project „Rules“

1.17 Project Risk Analysis

1.18 Project Documentation

Area	Description
File	
Access Authorisation	
Naming convention	
Rules	

2 Project Start

2.1 Minutes– Project Start

2.1.1 Project Start Workshop

2.1.2 Follow-up Workshop

2.1.3 Project Owner Meeting

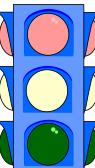
3 Project Co-ordination

3.1 Approval of Work-packages

3.2 Minutes – Project Co-ordination

4 Project Controlling

4.1 Project Status Report

<Project name> <Project no.>	PROJECT STATUS REPORT as of
 Project crisis Project in difficulties Project according to plan ✓	Overall Status: •
2) Status Project objectives • •	Activities: • •
3) Status Project progress • •	Activities: • •
4) Status Schedule • •	Activities: • •
5) Status Resources/costs • •	Activities: • •
6) Status Context • •	Activities: • •
7) Status Organisation/culture • •	Activities: • •

4.2 Additional Project Status Reports

4.3 Minutes – Project controlling

4.3.1 Project Controlling Meetings

4.3.2 Project Owner Meetings

5 Project Close Down

5.1 Project Close Down Report

<Project name>
 <Project no.>

PROJECT CLOSE DOWN REPORT

1) Overall impression

2) Reflection: Fulfilment of objectives

3) Reflection: Deliverables / Schedule

4) Reflection: Resources / Costs

5) Reflection: Internal Organisation / Environmental Relationships

6) Performance appraisal (Project owner, Project manager, Project member)

7) Lessons learned (Summary of Experiences and suggestions for improvement)

8) Post-Project Phase Planning, Additional Tasks

To-Do	Owner	Schedule

9) Project Close Down

 <Name> (Project owner)

 <Name> (Project manager)

5.2 Minutes – Project Close Down

5.2.1 Project Close Down Workshop