

Unblinded data report to Data Safety Monitoring Board

params\$sub_title

Dr. Rob van Wijk, Laurynas Mockeliunas, MSc., Prof. Dr. Ulrika Simonsson
Department of Pharmaceutical Biosciences, Uppsala University, Sweden

format(Sys.time(), '%A, %d %B %Y')

Contents

Introduction to trial and reported data	4
Executive summary	5
Glossary	14
Protocol summary	15
Report overview	16
Response to most recent DSMB requests	17
Enrolment status	18
Sample size	20
Participant status	21
Demographics	21
Risk factors	21
Laboratory findings	22
SARS-CoV-2 specific vaccination	25
Primary endpoint	26
Hospitalization due to COVID-19	26
Secondary endpoints	27

COVID-19 cases	28
Total cases of COVID-19	28
Mild (HS = 1)	28
Moderate (HS = 2)	28
Severe (HS = 3)	28
Hospitalized (HS = 4)	29
Hospitalized with oxygen (HS = 5)	29
Hospitalized with ventilation (HS = 6)	29
Death (HS = 7)	29
Repeated infection	29
Post COVID-19 viral syndrome	30
SARS-CoV-2 antibodies	31
Latent tuberculosis infection and COVID-19 or respiratory tract infections	33
Efficacy summary	34
Total hospitalization	34
Respiratory tract infections events	34
Health status	35
Sick leave	36
Adverse events	38
Reported events	38
Health status score	40
Respiratory tract infection events	41
Injection site reaction events	46
Other events	48
Serious adverse events (SAE)	53
Deaths	55
Protocol deviations	57
Adherence	57
Drop-out	57
Quality management data	58
Quality control data analysis	59
Data Definition Table primary endpoint	59
R version:	60

Appendix A: Power calculation	62
Appendix B: Additional summary tables	72
Appendix C: Additional figures	73
Appendix D: Additional data listings	74
Appendix E: Hospitalization by risk factor/classifier	75
Appendix F: KM plots for respiratory tract infections	90
Appendix G: KM plots for RTI per risk factor/classifier	92
Appendix H: Health status score over time	106
Appendix I: MedDRA	113
Appendix J: Risk factors per site	114
Baseline serology for SARS-CoV-2 per site	114
Baseline tuberculosis per site	115
Numerical demographics	117
Binary demographics	118
Categorical demographics	119
Work related demographics, including expected exposure to COVID-19	120
Additional figures per site	123

Introduction to trial and reported data

Report status

```
ifelse(CLOSED == TRUE, "This is the closed report with observations per arm, *please  
close if you are not permitted to read this.*", "This is the open report without observations  
per arm.")
```

Protocol title

Reducing morbidity and mortality in healthcare and other frontline workers at risk of exposure to SARS-CoV-2 by enhancing non-specific immune responses through Bacillus Calmette-Guérin vaccination, a double-blinded, randomized controlled trial.

Protocol number

TASK-008-BCG-CORONA

Protocol version

4.0 (date 20 May 2020) Incorporating amendments 1, 2 and 3

ClinicalTrials.gov Identifier

NCT04379336

Principle investigator

Dr. Caryn Upton

Meeting date

[...]

Data report issued

[...]

Data cut off date

[...]

Date of last data safety monitoring board (DSMB) meeting

[...]

Prepared by

[...]

Executive summary

Report overview	This report is the final trial report with data available in the study database as of <code>format(as.Date(date_cut_off), '%A, %d %B %Y')</code> . Summary tables are provided in the body of the report. Additional tables and figures referenced in the report are provided in the Appendices.
Enrolment status	<code>part_screen</code> participants screened for this study. <code>part_enrol_noQC</code> participants were enrolled. <code>part_enrol</code> participants were enrolled of which data (quality controlled, QC'ed) is in this report. <code>ifelse(CLOSED == T, paste(part_group1, " participants were assigned to the ", Arm1, " arm, ", part_group2, " participants were assigned to the ", Arm2, " arm."), "")</code>
Participant status	<code>ifelse(CLOSED == T, paste(part_followup_10_group1, "participants in the ", Arm1, " arm, and", part_followup_10_group2, "participants in the ", Arm2, " arm have completed week 10 serology follow-up."), paste(part_followup_10, "participants have completed week 10 serology follow-up.))</code> <code>ifelse(CLOSED == T, paste(part_followup_26_group1, "participants in the ", Arm1, " arm, and", part_followup_26_group2, "participants in the ", Arm2, " arm have completed week 26 serology follow-up."), paste(part_followup_26, "participants have completed week 26 serology follow-up.))</code> <code>ifelse(CLOSED == T, paste(part_followup_52_group1, "participants in the ", Arm1, " arm, and", part_followup_52_group2, "participants in the ", Arm2, " arm have completed week 52 serology follow-up."), paste(part_followup_52, "participants have completed week 52 serology follow-up.))</code> <code>ifelse(CLOSED == T, paste(part_discont, " participants have been discontinued (withdrawn), ", part_discont_group1, " in the ", Arm1, " arm and ", part_discont_group2, " in the ", Arm2, " arm.", sep = ""), paste(part_discont, " participants have been discontinued (withdrawn).", sep = ""))</code> <code>ifelse(CLOSED == T, paste("Of those, ", part_discont_losttofu, " participants were lost to follow-up, ", part_discont_losttofu_group1, " in the ", Arm1, " arm and ", part_discont_losttofu_group2, " in the ", Arm2, " arm, and ", part_discont - part_discont_losttofu, " participants lost interest, ", part_discont_group1 - part_discont_losttofu_group1, " in the ", Arm1, " arm and ", part_discont_group2 - part_discont_losttofu_group2, " in the ", Arm2, " arm.", sep = ""), paste("Of those, ", part_discont_losttofu, " participants were lost to follow-up, and ", part_discont - part_discont_losttofu, " participants lost interest.", sep = ""))</code>

**Demographics and
risk factors**

```
ifelse(CLOSED == T, paste("Median age (interquartile range [IQR])  
was ", sum_age_med_group1, " (", sum_age_lwr_group1, "-",  
sum_age_upr_group1, ") years in the ", Arm1, " arm, and ",  
sum_age_med_group2, " (", sum_age_lwr_group2, "-",  
sum_age_upr_group2, ") years in the ", Arm2, " arm.", sep = ''),  
paste("Median age (interquartile range [IQR]) was ", sum_age_med,  
" (", sum_age_lwr, "-", sum_age_upr, ") years.", sep = ''))  
ifelse(CLOSED == T, paste("Median BMI (IQR) was ",  
sum_BMI_med_group1, " (", sum_BMI_lwr_group1, "-",  
sum_BMI_upr_group1, ") kg/m$^2$ in the ", Arm1, " arm, and ",  
sum_BMI_med_group2, " (", sum_BMI_lwr_group2, "-",  
sum_BMI_upr_group2, ") kg/m$^2$ in the ", Arm2, " arm.", sep =  
''), paste("Median BMI (IQR) was ", sum_BMI_med, " (",  
sum_BMI_lwr, "-", sum_BMI_upr, ") kg/m$^2$.", sep = ''))  
ifelse(CLOSED == T, paste("Gender distribution was ",  
nrow(df_risk[df_risk$gender == 'Male' & df_risk$group == Arm1,]),  
" males and ", nrow(df_risk[df_risk$gender == 'Female' &  
df_risk$group == Arm1,]), " females in the ", Arm1, " arm, and ",  
nrow(df_risk[df_risk$gender == 'Male' & df_risk$group == Arm2,]),  
" males and ", nrow(df_risk[df_risk$gender == 'Female' &  
df_risk$group == Arm2,]), " females in the ", Arm2, " arm.", sep  
= ''), paste("Gender distribution was ",  
nrow(df_risk[df_risk$gender == 'Male',]), " males and ",  
nrow(df_risk[df_risk$gender == 'Female',]), " females.", sep =  
''))  
ifelse(CLOSED == T, paste("Ethnicity distribution was ",  
nrow(df_risk[df_risk$ethnicity == 'African' & df_risk$group ==  
Arm1,]), " African, ", nrow(df_risk[df_risk$ethnicity ==  
'Caucasian' & df_risk$group == Arm1,]), " Caucasian, ",  
nrow(df_risk[df_risk$ethnicity == 'Coloured' & df_risk$group ==  
Arm1,]), " Coloured, ", nrow(df_risk[df_risk$ethnicity ==  
'Indian' & df_risk$group == Arm1,]), " Indian, and ",  
nrow(df_risk[df_risk$ethnicity == 'Other' & df_risk$group ==  
Arm1,]), " Other in the ", Arm1, " arm, and ",  
nrow(df_risk[df_risk$ethnicity == 'African' & df_risk$group ==  
Arm2,]), " African, ", nrow(df_risk[df_risk$ethnicity ==  
'Caucasian' & df_risk$group == Arm2,]), " Caucasian, ",  
nrow(df_risk[df_risk$ethnicity == 'Coloured' & df_risk$group ==  
Arm2,]), " Coloured, ", nrow(df_risk[df_risk$ethnicity ==  
'Indian' & df_risk$group == Arm2,]), " Indian, and ",  
nrow(df_risk[df_risk$ethnicity == 'Other' & df_risk$group ==  
Arm2,]), " Other in the ", Arm2, " arm.", sep = ''),  
paste("Ethnicity distribution was ",  
nrow(df_risk[df_risk$ethnicity == 'African',]), " African, ",  
nrow(df_risk[df_risk$ethnicity == 'Caucasian',]), " Caucasian, ",  
nrow(df_risk[df_risk$ethnicity == 'Coloured',]), " Coloured, ",  
nrow(df_risk[df_risk$ethnicity == 'Indian',]), " Indian, and ",  
nrow(df_risk[df_risk$ethnicity == 'Other',]), " Other.", sep =  
''))
```

```

ifelse(CLOSED == T, paste("Job category distribution was ",
nrow(df_risk[df_risk$job_category == 'Doctor' & df_risk$group ==
Arm1,]), " Doctors, ", nrow(df_risk[df_risk$job_category ==
'Nurse' & df_risk$group == Arm1,]), " Nurses, and ",
nrow(df_risk[df_risk$job_category == 'Essential_workers' &
df_risk$group == Arm1,]) , " Essential workers in the ", Arm1, "
arm, and ", nrow(df_risk[df_risk$job_category == 'Doctor' &
df_risk$group == Arm2,]), " Doctors, ",
nrow(df_risk[df_risk$job_category == 'Nurse' & df_risk$group ==
Arm2,]), " Nurses, and ", nrow(df_risk[df_risk$job_category ==
'Essential_workers' & df_risk$group == Arm2,]) , " Essential
workers in the ", Arm2, " arm.", sep = ''), paste("Job category
distribution was ", nrow(df_risk[df_risk$job_category ==
'Doctor',]), " Doctors, ", nrow(df_risk[df_risk$job_category ==
'Nurse',]), " Nurses, and ", nrow(df_risk[df_risk$job_category ==
'Essential_workers',]) , " Essential workers.", sep = ''))
ifelse(CLOSED == T, paste("There were ",
nrow(df_risk[!is.na(df_risk$pack_years) & df_risk$pack_years != 0
& df_risk$group == Arm1,]), " (",
round(100*nrow(df_risk[!is.na(df_risk$pack_years) &
df_risk$pack_years != 0 & df_risk$group ==
Arm1,])/part_group1),"%") current smokers (mean pack years: ",
round(mean(df_risk$pack_years[!is.na(df_risk$pack_years) &
df_risk$pack_years != 0 & df_risk$group == Arm1]), 1),", IQR: ",
sum_smoke_lwr_group1, "-", sum_smoke_upr_group1, ") in the ",
Arm1, " arm, and ", nrow(df_risk[!is.na(df_risk$pack_years) &
df_risk$pack_years != 0 & df_risk$group == Arm2,]), " (",
round(100*nrow(df_risk[!is.na(df_risk$pack_years) &
df_risk$pack_years != 0 & df_risk$group ==
Arm2,])/part_group2),"%") current smokers (mean pack years: ",
round(mean(df_risk$pack_years[!is.na(df_risk$pack_years) &
df_risk$pack_years != 0 & df_risk$group == Arm2]), 1),", IQR: ",
sum_smoke_lwr_group2, "-", sum_smoke_upr_group2, ") in the ",
Arm2, " arm.", sep = ''), paste("There were ",
nrow(df_risk[!is.na(df_risk$pack_years) & df_risk$pack_years !=
0,]), " (", round(100*nrow(df_risk[!is.na(df_risk$pack_years) &
df_risk$pack_years != 0,])/part_enrol),"%") current smokers (mean
pack years: ",
round(mean(df_risk$pack_years[!is.na(df_risk$pack_years) &
df_risk$pack_years != 0]), 1),", IQR: ", sum_smoke_lwr, "-",
sum_smoke_upr, ").", sep = ''))

```

Prevalence of medical conditions

Diabetes mellitus:

```

ifelse(CLOSED == T,
paste(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_dm" & df_med_risk_sum$group == Arm1][[1]], " (",
round(100 * df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_dm" & df_med_risk_sum$group == Arm1][[1]] / part_group1,
1), "%) in the ", Arm1, " arm, ",
df_med_risk_sum$prevalence[df_med_risk_sum$risk == "medhis_dm" &
df_med_risk_sum$group == Arm2][[1]], " (", round(100 *
df_med_risk_sum$prevalence[df_med_risk_sum$risk == "medhis_dm" &
df_med_risk_sum$group == Arm2][[1]] / part_group2, 1), "%) in the
", Arm2, " arm", sep = ''),
paste(sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_dm"]), " (", round(100 *
sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_dm"]) / part_enrol, 1), "%)", sep = ''))
Hypertension:
ifelse(CLOSED == T,
paste(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_hyptens" & df_med_risk_sum$group == Arm1][[1]], " (",
round(100 * df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_hyptens" & df_med_risk_sum$group == Arm1][[1]] /
part_group1, 1), "%) in the ", Arm1, " arm, ",
df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_hyptens" & df_med_risk_sum$group == Arm2][[1]], " (",
round(100 * df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_hyptens" & df_med_risk_sum$group == Arm2][[1]] /
part_group2, 1), "%) in the ", Arm2, " arm", sep = ''),
paste(sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_hyptens"]), " (", round(100 *
sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_hyptens"]) / part_enrol, 1), "%)", sep = ''))
Asthma:
ifelse(CLOSED == T,
paste(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_asthma" & df_med_risk_sum$group == Arm1][[1]], " (",
round(100 * df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_asthma" & df_med_risk_sum$group == Arm1][[1]] /
part_group1, 1), "%) in the ", Arm1, " arm, ",
df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_asthma" & df_med_risk_sum$group == Arm2][[1]], " (",
round(100 * df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_asthma" & df_med_risk_sum$group == Arm2][[1]] /
part_group2, 1), "%) in the ", Arm2, " arm", sep = ''),
paste(sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_asthma"]), " (", round(100 *
sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_asthma"]) / part_enrol, 1), "%)", sep = ''))
Chronic obstructive pulmonary disorder (COPD):

```

```

ifelse(CLOSED == T,
paste(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_copd" & df_med_risk_sum$group == Arm1][[1]], " (",
round(100 * df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_copd" & df_med_risk_sum$group == Arm1][[1]] /
part_group1, 1), "%) in the ", Arm1, " arm, ",
df_med_risk_sum$prevalence[df_med_risk_sum$risk == "medhis_copd"
& df_med_risk_sum$group == Arm2][[1]], " (", round(100 *
df_med_risk_sum$prevalence[df_med_risk_sum$risk == "medhis_copd"
& df_med_risk_sum$group == Arm2][[1]] / part_group2, 1), "%) in
the ", Arm2, " arm", sep = ''),
paste(sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_copd"]), " (", round(100 *
sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_copd"]) / part_enrol, 1), "%)", sep = ''))
Other lung diseases:
ifelse(CLOSED == T,
paste(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_otherlung" & df_med_risk_sum$group == Arm1][[1]], " (",
round(100 * df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_otherlung" & df_med_risk_sum$group == Arm1][[1]] /
part_group1, 1), "%) in the ", Arm1, " arm, ",
df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_otherlung" & df_med_risk_sum$group == Arm2][[1]], " (",
round(100 * df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_otherlung" & df_med_risk_sum$group == Arm2][[1]] /
part_group2, 1), "%) in the ", Arm2, " arm", sep = ''),
paste(sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_otherlung"]), " (", round(100 *
sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_otherlung"]) / part_enrol, 1), "%)", sep = ''))
Cardiovascular disease:
ifelse(CLOSED == T,
paste(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_cvd" & df_med_risk_sum$group == Arm1][[1]], " (",
round(100 * df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_cvd" & df_med_risk_sum$group == Arm1][[1]] / part_group1,
1), "%) in the ", Arm1, " arm, ",
df_med_risk_sum$prevalence[df_med_risk_sum$risk == "medhis_cvd" &
df_med_risk_sum$group == Arm2][[1]], " (", round(100 *
df_med_risk_sum$prevalence[df_med_risk_sum$risk == "medhis_cvd" &
df_med_risk_sum$group == Arm2][[1]] / part_group2, 1), "%) in the
", Arm2, " arm", sep = ''),
paste(sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_cvd"]), " (", round(100 *
sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_cvd"]) / part_enrol, 1), "%)", sep = ''))
Kidney disease:

```

```

ifelse(CLOSED == T,
paste(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_kd" & df_med_risk_sum$group == Arm1][[1]], " (",
round(100 * df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_kd" & df_med_risk_sum$group == Arm1][[1]] / part_group1,
1), "%) in the ", Arm1, " arm, ",
df_med_risk_sum$prevalence[df_med_risk_sum$risk == "medhis_kd" &
df_med_risk_sum$group == Arm2][[1]], " (", round(100 *
df_med_risk_sum$prevalence[df_med_risk_sum$risk == "medhis_kd" &
df_med_risk_sum$group == Arm2][[1]] / part_group2, 1), "%) in the
", Arm2, " arm", sep = ''),
paste(sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_kd"]), " (", round(100 *
sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"medhis_kd"]) / part_enrol, 1), "%)", sep = ''))
Bacillus Calmette-Guérin (BCG) scar:
ifelse(CLOSED == T,
paste(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"bcg_scar" & df_med_risk_sum$group == Arm1][[1]], " (", round(100 *
df_med_risk_sum$prevalence[df_med_risk_sum$risk == "bcg_scar"
& df_med_risk_sum$group == Arm1][[1]] / part_group1, 1), "%) in
the ", Arm1, " arm, ",
df_med_risk_sum$prevalence[df_med_risk_sum$risk == "bcg_scar" &
df_med_risk_sum$group == Arm2][[1]], " (", round(100 *
df_med_risk_sum$prevalence[df_med_risk_sum$risk == "bcg_scar" &
df_med_risk_sum$group == Arm2][[1]] / part_group2, 1), "%) in the
", Arm2, " arm", sep = ''),
paste(sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"bcg_scar"]), " (", round(100 *
sum(df_med_risk_sum$prevalence[df_med_risk_sum$risk ==
"bcg_scar"]) / part_enrol, 1), "%)", sep = ''))

```

Primary endpoint

```

ifelse(CLOSED == T, paste(n_hosp_c19_group1, " (", round(100 *
n_hosp_c19_group1 / part_group1, 1), "%) participants have been
hospitalized with a positive COVID-19 PCR test result in the ",
Arm1, " arm.", sep = ''), paste(n_hosp_c19, " (", round(100 *
n_hosp_c19 / part_enrol, 1), "%) participants have been
hospitalized with a positive COVID-19 PCR test result.", sep =
''))
ifelse(CLOSED == T, paste(n_hosp_c19_group2, " (", round(100 *
n_hosp_c19_group2 / part_group2, 1), "%) participants have been
hospitalized with a positive COVID-19 PCR test result in the ",
Arm2, " arm.", sep = ''), '')

```

	<pre> ifelse(CLOSED == T, paste("A Cox proportional hazard model was utilized to assess the statistical significance of the treatment arm on the primary endpoint. The hazard ratio was ", round(summary(coxph(formula = Surv(time, status) ~ group, data = hosp_c19_km_censor_SAP))\$coefficients[2], 2), " (", round(exp(confint(coxph(formula = Surv(time, status) ~ group, data = hosp_c19_km_censor_SAP))[1]), 2), "-", round(exp(confint(coxph(formula = Surv(time, status) ~ group, data = hosp_c19_km_censor_SAP))[2]), 1) , " 95% confidence interval) for the ", Arm2, " arm relative to the ", Arm1, " arm (p-value = ", round(summary(coxph(formula = Surv(time, status) ~ group, data = hosp_c19_km_censor_SAP))\$coefficients[5], 3), ")", sep = ""), "") </pre> <p>The primary endpoint is here reported based on the intention-to-treat population with right censoring only at withdrawal of consent or lost to follow-up, death, or the end of the trial.</p>
Coronavirus disease 2019 (COVID-19)	<pre> ifelse(CLOSED == T, paste("In total, ", part_c19_group1, " (", round(100 * part_c19_group1 / part_group1, 1), "%) participants had a COVID-19 diagnosis in the ", Arm1, " arm. In total, ", part_c19_group2, " (", round(100 * part_c19_group2 / part_group2, 1), "%) participants had a COVID-19 diagnosis in the ", Arm2, " arm.", sep = ''), paste("In total, ", part_c19, " (", round(100 * part_c19 / part_enrol, 1), "%) participants had a COVID-19 diagnosis.", sep = '')) </pre> <p><i>COVID-19 was diagnosed as respiratory tract infection symptoms with a positive PCR-test.</i></p>
Efficacy summary	<pre> ifelse(CLOSED == T, paste(n_ae_rti_group1, " respiratory tract infection events occurred in ", part_ae_rti_group1, " (", round(100 * part_ae_rti_group1 / part_group1, 1), "%) participants in the ", Arm1, " arm, and ", n_ae_rti_group2, " respiratory tract infection events occurred in ", part_ae_rti_group2, " (", round(100 * part_ae_rti_group2 / part_group2, 1), "%) participants in the ", Arm2, " arm.", sep = ""), paste(n_ae_rti, " respiratory tract infection events occurred in ", part_ae_rti, " (", round(100 * part_ae_rti / part_enrol, 1), "%) participants.", sep = "")) </pre> <pre> ifelse(CLOSED == T, paste(n_hosp_group1, " (", round(100 * n_hosp_group1 / part_group1, 1), "%) participants were hospitalized in the ", Arm1, " arm, and ", n_hosp_group2, " (", round(100 * n_hosp_group2 / part_group2, 1), "%) participants were hospitalized in the ", Arm2, " arm.", sep = ''), paste(n_hosp, " (", round(100 * n_hosp / part_enrol, 1), "%) participants were hospitalized.", sep = '')) </pre> <p>Health status over time per individual participant is shown over time in this data report.</p>
Safety summary	<p>n_ae total adverse events (including respiratory tract infections, see above) occurred in part_ae participants.</p>

```

n_ae_isr injection site reaction (ISR) adverse events occurred in
part_ae_isr (round(100*part_ae_isr/part_enrol, 1)%) participants.
ifelse(CLOSED == T, paste(n_ae_other_group1, "other adverse
events occurred in", part_ae_other_group1, "participants in the",
Arm1, "arm, and", n_ae_other_group2, "other adverse events
occurred in", part_ae_other_group2, "participants in the", Arm2,
"arm."), paste(n_ae_other, "other adverse events occurred in",
part_ae_other, "participants"))

```

```

ifelse(CLOSED == T, paste("Of the ", n_ae_rti, " respiratory
tract infection events, ", n_ae_rti_mild, " (", round(100 *
n_ae_rti_mild / n_ae_rti), "%, ", ae_rti_mild %>% filter(group ==
Arm1) %>% nrow()," in the ", Arm1, " arm and ", ae_rti_mild %>%
filter(group == Arm2) %>% nrow()," in the ", Arm2, " arm) were
considered mild and ", n_ae_rti_moderate, " (", round(100 *
n_ae_rti_moderate / n_ae_rti), "%, ", ae_rti_moderate %>%
filter(group == Arm1) %>% nrow()," in the ", Arm1, " arm and ",
ae_rti_moderate %>% filter(group == Arm2) %>% nrow()," in the ",
Arm2, " arm) were considered moderate.", sep = ''), paste("Of the
", n_ae_rti, " respiratory tract infection events, ",
n_ae_rti_mild, " (", round(100 * n_ae_rti_mild / n_ae_rti), "%)
were considered mild and ", n_ae_rti_moderate, " (", round(100 *
n_ae_rti_moderate / n_ae_rti), "%) were considered moderate.",
sep = ''))

```

Of the n_ae_isr injection site reaction adverse events, n_ae_isr_mild
(round(100 * n_ae_isr_mild / n_ae_isr)%) were considered mild and
n_ae_isr_moderate (round(100 * n_ae_isr_moderate / n_ae_isr)%)
were considered moderate.

```

ifelse(CLOSED == T, paste("Of the ", n_ae_other, " other adverse
events, ", n_ae_other_mild, " (", round(100 * n_ae_other_mild /
n_ae_other), "%, ", ae_other_mild %>% filter(group == Arm1) %>%
nrow()," in the ", Arm1, " arm and ", ae_other_mild %>%
filter(group == Arm2) %>% nrow()," in the ", Arm2, " arm) were
considered mild and ", n_ae_other_moderate, " (", round(100 *
n_ae_other_moderate / n_ae_other), "%, ", ae_other_moderate %>%
filter(group == Arm1) %>% nrow()," in the ", Arm1, " arm and ",
ae_other_moderate %>% filter(group == Arm2) %>% nrow()," in the
", Arm2, " arm) were considered moderate.", sep = ''), paste("Of
the ", n_ae_other, " other adverse events, ", n_ae_other_mild, "
(", round(100 * n_ae_other_mild / n_ae_other), "%) were
considered mild and ", n_ae_other_moderate, " (", round(100 *
n_ae_other_moderate / n_ae_other), "%) were considered
moderate.", sep = ''))

```

n_sae_previous serious adverse events were reported in the previous DSMB
report.

```

ifelse(n_sae - n_sae_previous == 0, "There have been no
additional serious adverse events since the last DSMB meeting.",
ifelse(n_sae - n_sae_previous == 1, paste(n_sae - n_sae_previous,
"additional serious adverse events was reported since the last
DSMB meeting."), paste(n_sae - n_sae_previous, "additional
serious adverse events were reported since the last DSMB
meeting.")))

```

	<pre> ifelse(CLOSED == T, paste("In total,", n_sae_group1, "serious adverse events were reported in", part_sae_group1, "participants in the ", Arm1, " arm, and ", n_sae_group2, "serious adverse events were reported in", part_sae_group2, "participants in the ", Arm2, " arm."), paste("In total,", n_sae, "serious adverse events were reported in", part_sae, "participants.")) SAEs were paste(ifelse(n_sae_unrelated > 0, 'unrelated', ''), ifelse(n_sae_unlikelyrelated > 0, ', or unlikely related', ''), ifelse(n_sae_possiblyrelated > 0, ', or possibly related', ''), ifelse(n_sae_probablyrelated > 0, ', or probably related', ''), ifelse(n_sae_definitely > 0, ', or definitely related', ''), sep = '') to the intervention. </pre>
Other COVID-19 vaccinations	<pre> ifelse(CLOSED == T, paste(part_vaccine_C19_group1, " (", round(100 * part_vaccine_C19_group1 / part_group1, 1), "%) participants recorded a SARS-CoV-2 specific vaccination since the start of the trial in the ", Arm1, " arm.", sep = ''), paste(part_vaccine_C19, " (", round(100 * part_vaccine_C19 / part_enrol, 1), "%) participants recorded a SARS-CoV-2 specific vaccination since the start of the trial.", sep = '')) ifelse(CLOSED == T, paste(part_vaccine_C19_group2, " (", round(100 * part_vaccine_C19_group2 / part_group2, 1), "%) participants recorded a SARS-CoV-2 specific vaccination since the start of the trial in the ", Arm2, " arm.", sep = ''), "") </pre>
Protocol deviations	<pre> n_deviations protocol deviations associated with part_deviations participants have been reported. ifelse(n_deviations_safety == 0, "None of the deviations impacted participant safety.", paste(n_deviations_integrity, "of the deviations impacted participant safety.")) ifelse(n_deviations_integrity == 0, "None of the deviations impacted scientific integrity.", paste(n_deviations_integrity, "of the deviations impacted scientific integrity.")) </pre>
Quality management data	<p>Quality management reviews were performed by TASK and were last completed on <code>format(as.Date(date_last_review), '%A, %d %B %Y')</code>. In <code>n_qc_yes</code> of the reviewed data entries (<code>round(100* n_qc_yes/n_qc)%</code>), a correction was needed.</p>
Quality control data analysis	<p>Quality control review of data analysis scripts were last completed on <code>format(as.Date(date_last_review_data_analysis), '%A, %d %B %Y')</code>, with no major findings.</p>

Glossary

α Significance level for statistical analysis

β Probability of type II error ($1 - \beta = \text{power}$)

AE Adverse event

BCG Bacillus Calmette-Guérin (vaccine)

BMI Body mass index

Censored observations The event of interest is not observed within the study because of for example drop-out of the participant or the study ends before the event of interest occurs

COPD Chronic obstructive pulmonary disorder

COVID-19 Coronavirus disease 2019

Cox proportional hazards model Statistical model for survival analysis (see: survival analysis) of for example hospitalization rate, with the objective to link survival to risk factors (both continuous and categorical covariates)

DSMB Data safety monitoring board

eCRF Electronic case report form

FU Follow-up on for example recorded events

Hazard rate Probability of the event of interest happening

HR Hazard ratio, quantifying the ratio of the hazard of an event in one arm compared to another (for example intervention arm, or for different risk factors). A $HR > 1$ reflects an increasing probability of the event and a decreased survival, and $HR < 1$ reflects a decreasing probability of the event and a increased survival.

HCW Health care worker

HIV Human immunodeficiency virus

HS Health status used to characterize seriousness of events from healthy (0) to fatal (7)

ICF Informed consent form

IgG Immunoglobulin G, the antibody used to test for previous SARS-CoV-2 infections

IGRA Interferon-gamma release assay, used to test for tuberculosis infections

IP Investigational product

IQR Interquartile range (25th-50th percentile) to capture variability in variable

ISR Injection site reaction

Kaplan-Meier plot Figure to visualize survival (proportion) as function of time and variables (for example intervention arm, risk factor)

Markov model Modelling approach of states (for example health status scores) where the future state only depends on the current state

MedDRA Medical dictionary for regulatory activities

NA Not available

NEC Not elsewhere classified

PPE Personal protective equipment

QC Quality control of for example the data analysis script

RTI Respiratory tract infection

SAE Serious adverse event

SAHPRA South African Health Product Regulatory Authority

SARS-CoV-2 Severe acute respiratory syndrome coronavirus 2

Survival analysis Analysis of the duration of time before an event happens

TB Tuberculosis (original indication for BCG vaccine)

Protocol summary

Rationale	<p>A novel betacoronavirus, SARS-CoV-2, is spreading rapidly throughout the world. A large epidemic in South Africa may overwhelm available hospital capacity and healthcare resources which would be worsened by absenteeism of healthcare workers and other frontline staff (HCW). Strategies to prevent morbidity and mortality of HCW are desperately needed to safeguard continuous patient care. Bacillus Calmette-Guérin (BCG) is a vaccine against tuberculosis (TB), with protective non-specific effects against other respiratory tract infections in <i>in vitro</i> and <i>in vivo</i> studies, with reported morbidity and mortality reductions as high as 70%.</p> <p>Due to the novel nature of COVID-19, the impact of genetics on infection, disease severity and outcomes is not well understood. A series of exploratory sub-studies on blood and saliva-samples from a subset of participants will be performed focusing on epigenetic changes and immunogenomics across groups.</p>
Hypothesis	BCG vaccination may reduce the morbidity and mortality of HCW during the SARS-CoV-2 outbreak in South Africa.
Study design	A double-blinded, randomised controlled trial to compare the efficacy of BCG vaccination versus placebo for reducing HCW morbidity and mortality during 52 weeks of follow-up among HCW with risk of exposure to SARS-CoV-2 infected patients.
Duration	52 weeks of follow-up after enrolment.
Intervention	Participants are randomized to administration of BCG vaccine or placebo in a 1:1 ratio.
Study population	A minimum of 500 HCW, defined as nurses, medical doctors and support personnel and other frontline staff (essential workers), working in a facility or at a service where they are likely to have contact with SARS-CoV-2 infected patients. For simplicity “HCW” is used for all these participants.
Risk and benefits	Based on previous experience and randomized controlled trials in adult and elderly individuals, the risks of BCG vaccination are considered low. The objective of this trial is to evaluate the beneficial effects of BCG vaccination through a lower hospital admission of HCW and/or a mitigated clinical course of COVID-19.

Report overview

```
ifelse(CLOSED == TRUE, "**This is the closed report with observations per arm, please  
close if you are not permitted to read this.**", "**This is the open report without  
observations per arm.**")
```

The purpose of this report is to review cumulative enrolment, primary endpoint, efficacy data, and safety data for the “BCG re-vaccination for healthcare workers in SARS-CoV-2 pandemic” trial and reflects data from the database as of `date_last_review`.

```
ifelse(n_dsmb == 1, paste('There has been ', n_dsmb, ' completed DSMB meeting for this  
study and the last meeting was held on ', date_last_dsmb, '.', sep = ''), paste('There  
have been ', n_dsmb, ' completed DSMB meetings for this study and the last meeting was  
held on ', date_last_dsmb, '.', sep = ''))
```

Readers of this report are asked to maintain the confidentiality of the information provided in this report.

Response to most recent DSMB requests

1. List of response to DSMB requests

Enrolment status

part_screen participants screened for this study.

part_enrol_noQC participants were enrolled.

part_enrol participants were enrolled and data underwent quality control and were included in this report.

The trial is at 100% and the database has been locked.

Screening

df_full %>% filter(meet_crit == 'Yes') %>% distinct(PID) %>% nrow() participants met all inclusion criteria and no exclusion criteria.

Inclusion criteria

nrow(df_incl_crit[df_incl_crit\$incl1 == 'Yes',]) participants met inclusion criterion 1 (>18 year).

nrow(df_incl_crit[df_incl_crit\$incl2 == 'Yes',]) participants met inclusion criterion 2 (health care worker).

nrow(df_incl_crit[df_incl_crit\$incl3 == 'Yes',]) participants met inclusion criterion 3 (informed consent).

nrow(df_incl_crit[df_incl_crit\$incl4 == 'Yes',]) participants met inclusion criterion 4 (mobile phone for follow-up).

Exclusion criteria

nrow(df_excl_crit[df_excl_crit\$excl1 == 'Yes',]) participants met exclusion criterion 1 (BCG vaccine allergy).

nrow(df_excl_crit[df_excl_crit\$excl2 == 'Yes',]) participants met exclusion criterion 2 (active TB).

nrow(df_excl_crit[df_excl_crit\$excl3 == 'Yes',]) participants met exclusion criterion 3 (HIV-1 positive).

nrow(df_excl_crit[df_excl_crit\$excl4 == 'Yes',]) participants met exclusion criterion 4 (respiratory tract infection).

nrow(df_excl_crit[df_excl_crit\$excl5 == 'Yes',]) participants met exclusion criterion 5 (immunocompromised).

nrow(df_excl_crit[df_excl_crit\$excl6 == 'Yes',]) participants met exclusion criterion 6 (pregnancy).

nrow(df_excl_crit[df_excl_crit\$excl7 == 'Yes',]) participants met exclusion criterion 7 (excluded medication).

nrow(df_excl_crit[df_excl_crit\$excl8 == 'Yes',]) participants met exclusion criterion 8 (experimental anti-SARS-CoV-2).

Randomization

```
ifelse(CLOSED == T, paste(part_group1, " participants were randomly assigned to the", Arm1, " arm, ", part_group2, " participants were randomly assigned to the ", Arm2, " arm.", sep = ''), "The participants were randomly assigned to arm 1 or 2 in a 1:1 ratio.")
```

Follow-up

The following table reports on the number of participants for which follow-up visit(s) has been recorded.

Follow-up visit	Number of participants
1st	part_followup_visit_1
2nd	part_followup_visit_2
3rd	part_followup_visit_3
4th	part_followup_visit_4
5th	part_followup_visit_5
6th	part_followup_visit_6
7th	part_followup_visit_7
8th	part_followup_visit_8
9th	part_followup_visit_9
10th	part_followup_visit_10
11th	part_followup_visit_11
12th	part_followup_visit_12
13th	part_followup_visit_13
14th	part_followup_visit_14
15th	part_followup_visit_15
16th	part_followup_visit_16
17th	part_followup_visit_17

Sample size

As per the protocol, at least 500 participants were included in the study.

The primary endpoint was proportion of hospitalization per arm. For a two-sample test (1 test, 1 control) for proportions, a power calculation based on the Pearson chi-square test with continuity correction (for smaller sample sizes) was performed. Here, a sample size calculation for two sample populations with $\alpha = 0.05$ and $\beta = 0.2$ (i.e. 80% power) was performed. This calculation did not take into account a drop-out rate.

Power calculation resulted in a sample size of 220 participants per arm assuming an attack rate in health care workers of 30%, hospitalisation rate of 20% and a reduction by vaccination of 75%. These assumptions were based on Liu et al (2020), the case severity reporting from Wuhan, and the reduction in respiratory infections by BCG previously reported by Wardhana et al (2011) and Nemes et al (2018), respectively.

The full detailed power calculations supporting the sample size can be found in Appendix A.

Participant status

Participants in arm 1 received **Arm1**. Participants in arm 2 received **Arm2**.

This report is based on the intention-to-treat population with right censoring only at withdrawal of consent or lost to follow-up, death, or the end of the trial.

Demographics

The following figures display the baseline demographics of the participants `ifelse(CLOSED == T, "stratified per arm", "")`. The solid line represents the smooth density.

Age

Body mass index (BMI)

Dotted lines delimit categories underweight (<18.5), normal (18.5-24.9), overweight (25-29.9), obese (30-39.9), extremely obese (>40).

Currently smoking

If participants self reported as currently smoking, they were asked to report the number of cigarettes packs years they smoke. The distribution is shown below.

Additional tabulated demographic data can be found in Appendix B, additional figures can be found in Appendix C.

Risk factors

The following risk factors/classifiers are identified (dataset variable given in brackets):

- age (see above)
- gender
- BMI (see above)
- ethnicity
- job category
- medical history of diabetes mellitus (`medhis_dm`)
- medical history of hypertension (`medhis_hyptens`)
- medical history of cardiovascular disease (`medhis_cvd`)
- medical history of asthma (`medhis_asthma`)
- medical history of COPD (`medhis_copd`)
- medical history of other lung disease (`medhis_otherlung`)
- medical history of kidney disease (`medhis_kd`)
- presence of a BCG scar (`bcg_scar`)
- currently smoking (`pack_years`, see above)

The full list of potential risk factors/classifiers can be found in Appendix D.

Distribution of the identified risk factors/classifiers (absolute and relative) are shown below `ifelse(CLOSED == T, "stratified per arm", "")`.

Laboratory findings

Serology for SARS-CoV-2

At week 0, 10, 26, and 52, participants were tested for SARS-CoV-2 infection based on immunoglobulin G (IgG) serology.

```
ifelse(CLOSED == T, paste(n_c19_serology_group1, " (", round(100 * n_c19_serology_group1
/ part_c19_serology_group1, 1), "%) participants had at least one positive serology
result for SARS-CoV-2 based on IgG in the ", Arm1, " arm. \n", n_c19_serology_group2,
" (", round(100 * n_c19_serology_group2 / part_c19_serology_group2, 1), "%) participants
had at least one positive serology result for SARS-CoV-2 based on IgG in the ", Arm2, "
arm.", sep = ''), paste(n_c19_serology, " (", round(100 * n_c19_serology / part_c19_serology,
1), "%) participants had at least one positive serology result for SARS-CoV-2 based on
IgG.", sep = ''))
```

Full details on the serology findings can be found in SARS-CoV-2 antibodies.

Tuberculosis

At week 0 and 52, participants were tested for TB infection based on an interferon-gamma release assay (IGRA).

```
Week 0 ifelse(CLOSED == T, paste(n_TB_baseline_pos_group1, " (", round(100 * n_TB_baseline_pos_group1
/ part_TB_week0_group1, 1), "%) participants had a positive IGRA result for TB at baseline
in the ", Arm1, " arm. \n", n_TB_baseline_pos_group2, " (", round(100 * n_TB_baseline_pos_group2
/ part_TB_week0_group2, 1), "%) participants had a positive IGRA result for TB at baseline
in the ", Arm2, " arm.", sep = ''), paste(n_TB_baseline_pos, " (", round(100 * n_TB_baseline_pos
/ part_TB_week0, 1), "%) participants had a positive IGRA result for TB at baseline.",
sep = ''))
```

```
ifelse(CLOSED == T, paste(n_TB_baseline_int_group1, " (", round(100 * n_TB_baseline_int_group1
/ part_TB_week0_group1, 1), "%) participants had a intermediate IGRA result for TB at
baseline in the ", Arm1, " arm. \n", n_TB_baseline_int_group2, " (", round(100 * n_TB_baseline_int_group2
/ part_TB_week0_group2, 1), "%) participants had a intermediate IGRA result for TB at
baseline in the ", Arm2, " arm.", sep = ''), paste(n_TB_baseline_int, " (", round(100 *
n_TB_baseline_int / part_TB_week0, 1), "%) participants had a intermediate IGRA result
for TB at baseline.", sep = ''))
```

```
ifelse(CLOSED == T, paste(n_TB_baseline_neg_group1, " (", round(100 * n_TB_baseline_neg_group1
/ part_TB_week0_group1, 1), "%) participants had a negative IGRA result for TB at baseline
in the ", Arm1, " arm. \n", n_TB_baseline_neg_group2, " (", round(100 * n_TB_baseline_neg_group2
/ part_TB_week0_group2, 1), "%) participants had a negative IGRA result for TB at baseline
in the ", Arm2, " arm.", sep = ''), paste(n_TB_baseline_neg, " (", round(100 * n_TB_baseline_neg
/ part_TB_week0, 1), "%) participants had a negative IGRA result for TB at baseline.",
sep = ''))
```

```
ifelse(CLOSED == T, paste(n_TB_baseline_nr_group1, " (", round(100 * n_TB_baseline_nr_group1
/ part_TB_week0_group1, 1), "%) participants had a 'no result' IGRA result for TB at
baseline in the ", Arm1, " arm. \n", n_TB_baseline_nr_group2, " (", round(100 * n_TB_baseline_nr_group2
/ part_TB_week0_group2, 1), "%) participants had a 'no result' IGRA result for TB at
baseline in the ", Arm2, " arm.", sep = ''), paste(n_TB_baseline_nr, " (", round(100 *
n_TB_baseline_nr / part_TB_week0, 1), "%) participants had a 'no result' IGRA result for
TB at baseline.", sep = ''))
```

```

Week 52 ifelse(CLOSED == T, paste(n_TB_week52_pos_group1, " (", round(100 * n_TB_week52_pos_group1
/ part_TB_week52_group1, 1), "%) participants had a positive IGRA result for TB at week
52 in the ", Arm1, " arm. \n", n_TB_week52_pos_group2, " (", round(100 * n_TB_week52_pos_group2
/ part_TB_week52_group2, 1), "%) participants had a positive IGRA result for TB at week
52 in the ", Arm2, " arm.", sep = ''), paste(n_TB_week52_pos, " (", round(100 * n_TB_week52_pos
/ part_TB_week52, 1), "%) participants had a positive IGRA result for TB at week 52.",
sep = ''))

ifelse(CLOSED == T, paste(n_TB_week52_int_group1, " (", round(100 * n_TB_week52_int_group1
/ part_TB_week52_group1, 1), "%) participants had a intermediate IGRA result for TB at
week 52 in the ", Arm1, " arm. \n", n_TB_week52_int_group2, " (", round(100 * n_TB_week52_int_group2
/ part_TB_week52_group2, 1), "%) participants had a intermediate IGRA result for TB at
week 52 in the ", Arm2, " arm.", sep = ''), paste(n_TB_week52_int, " (", round(100 *
n_TB_week52_int / part_TB_week52, 1), "%) participants had a intermediate IGRA result for
TB at week 52.", sep = ''))

ifelse(CLOSED == T, paste(n_TB_week52_neg_group1, " (", round(100 * n_TB_week52_neg_group1
/ part_TB_week52_group1, 1), "%) participants had a negative IGRA result for TB at week
52 in the ", Arm1, " arm. \n", n_TB_week52_neg_group2, " (", round(100 * n_TB_week52_neg_group2
/ part_TB_week52_group2, 1), "%) participants had a negative IGRA result for TB at week
52 in the ", Arm2, " arm.", sep = ''), paste(n_TB_week52_neg, " (", round(100 * n_TB_week52_neg
/ part_TB_week52, 1), "%) participants had a negative IGRA result for TB at week 52.",
sep = ''))

ifelse(CLOSED == T, paste(n_TB_week52_nr_group1, " (", round(100 * n_TB_week52_nr_group1
/ part_TB_week0_group1, 1), "%) participants had a 'no result' IGRA result for TB at week
52 in the ", Arm1, " arm. \n", n_TB_week52_nr_group2, " (", round(100 * n_TB_week52_nr_group2
/ part_TB_week0_group2, 1), "%) participants had a 'no result' IGRA result for TB at
week 52 in the ", Arm2, " arm.", sep = ''), paste(n_TB_week52_nr, " (", round(100 *
n_TB_week52_nr / part_TB_week0, 1), "%) participants had a 'no result' IGRA result for
TB at week 52.", sep = ''))

Week unknown ifelse(CLOSED == T, paste(n_TB_unknown_pos_group1, " (", round(100 *
n_TB_unknown_pos_group1 / part_TB_unknown_group1, 1), "%) participants had a positive
IGRA result for TB at an unknown week in the ", Arm1, " arm. \n", n_TB_unknown_pos_group2,
" (", round(100 * n_TB_unknown_pos_group2 / part_TB_unknown_group2, 1), "%) participants
had a positive IGRA result for TB at an unknown week in the ", Arm2, " arm.", sep = ''),
paste(n_TB_unknown_pos, " (", round(100 * n_TB_unknown_pos / part_TB_unknown, 1), "%)
participants had a positive IGRA result for TB at an unknown week.", sep = ''))

ifelse(CLOSED == T, paste(n_TB_unknown_int_group1, " (", round(100 * n_TB_unknown_int_group1
/ part_TB_unknown_group1, 1), "%) participants had a intermediate IGRA result for TB at
an unknown week in the ", Arm1, " arm. \n", n_TB_unknown_int_group2, " (", round(100 *
n_TB_unknown_int_group2 / part_TB_unknown_group2, 1), "%) participants had a intermediate
IGRA result for TB at an unknown week in the ", Arm2, " arm.", sep = ''), paste(n_TB_unknown_int,
" (", round(100 * n_TB_unknown_int / part_TB_unknown, 1), "%) participants had a intermediate
IGRA result for TB at an unknown week.", sep = ''))

ifelse(CLOSED == T, paste(n_TB_unknown_neg_group1, " (", round(100 * n_TB_unknown_neg_group1
/ part_TB_unknown_group1, 1), "%) participants had a negative IGRA result for TB at an
unknown week in the ", Arm1, " arm. \n", n_TB_unknown_neg_group2, " (", round(100 *
n_TB_unknown_neg_group2 / part_TB_unknown_group2, 1), "%) participants had a negative
IGRA result for TB at an unknown week in the ", Arm2, " arm.", sep = ''), paste(n_TB_unknown_neg,
" (", round(100 * n_TB_unknown_neg / part_TB_unknown, 1), "%) participants had a negative
IGRA result for TB at an unknown week.", sep = ''))

```

```

ifelse(CLOSED == T, paste(n_TB_unknown_nr_group1, " (", round(100 * n_TB_unknown_nr_group1
/ part_TB_unknown_group1, 1), "%) participants had a 'no result' IGRA result for TB at
an unknown week in the ", Arm1, " arm. \n", n_TB_unknown_nr_group2, " (", round(100 *
n_TB_unknown_nr_group2 / part_TB_unknown_group2, 1), "%) participants had a 'no result'
IGRA result for TB at an unknown week in the ", Arm2, " arm.", sep = ''), paste(n_TB_unknown_nr,
" (", round(100 * n_TB_unknown_nr / part_TB_unknown, 1), "%) participants had a 'no
result' IGRA result for TB at an unknown week.", sep = ''))

```

```

Active      tuberculosis ifelse(CLOSED == T, paste(n_active_TB_group1, " (", round(100 *
n_active_TB_group1 / part_group1, 1), "%) participants had a case of active TB during
the trial in the ", Arm1, " arm. \n", n_active_TB_group2, " (", round(100 * n_active_TB_group2
/ part_group2, 1), "%) participants had a case of active TB during the trial in the ",
Arm2, " arm.", sep = ''), paste(n_active_TB, " (", round(100 * n_active_TB / part_enrol,
1), "%) participants had a case of active TB during the trial.", sep = ''))

```


SARS-CoV-2 specific vaccination

Vaccination of trial participants with SARS-CoV-2 specific vaccines is reported below.

```
ifelse(CLOSED == T, paste(part_vaccine_C19_group1, " (", round(100 * part_vaccine_C19_group1 / part_group1, 1), "%) participants recorded a SARS-CoV-2 specific vaccination since the start of the trial in the ", Arm1, " arm. \n", part_vaccine_C19_group2, " (", round(100 * part_vaccine_C19_group2 / part_group2, 1), "%) participants recorded a SARS-CoV-2 specific vaccination since the start of the trial in the ", Arm2, " arm.", sep = ''), paste(part_vaccine_C19, " (", round(100 * part_vaccine_C19 / part_enrol, 1), "%) participants recorded a SARS-CoV-2 specific vaccination since the start of the trial.", sep = ''))
```

Kaplan-Meier estimates of the proportion of subject which were not vaccinated with a SARS-CoV-2 specific vaccine versus time. `ifelse(CLOSED == T, "The figure is stratified by study arm. ", "")`The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. Vertical lines show censoring events (withdrawal or lost to follow-up, or death, whichever comes first). The table below the figure shows the number of subjects still elligble for SARS-CoV-2 specific vaccination`ifelse(CLOSED == T, ", stratified by study arm", "")`, versus time.

Primary endpoint

The primary endpoint was proportion of hospitalization due to COVID-19 per arm.

Hospitalization due to COVID-19

```
ifelse(CLOSED == T, paste(n_hosp_c19_group1, " (", round(100 * n_hosp_c19_group1 / part_group1, 1), "%) participants were hospitalized in the ", Arm1, " arm with a positive COVID-19 PCR test result. \n", n_hosp_c19_group2, " (", round(100 * n_hosp_c19_group2 / part_group2, 1), "%) participants were hospitalized in the ", Arm2, " arm with a positive COVID-19 PCR test result.", sep = ''), paste(n_hosp_c19, " (", round(100 * n_hosp_c19 / part_enrol, 1), "%) participants were hospitalized with a positive COVID-19 PCR test result.", sep = ''))
```

The endpoint is here reported based on the intention-to-treat population with right censoring only at withdrawal of consent or lost to follow-up, death, or the end of the trial.

Kaplan-Meier estimates of the proportion of subject which were not hospitalized with a positive COVID-19 PCR test result versus time. `ifelse(CLOSED == T, "The figure is stratified by study arm. ", "")` The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized `ifelse(CLOSED == T, " , stratified by study arm", "")`, versus time. A figure with a enhanced y-axis is shown as well for clearer visualization.

Primary endpoint per site

The primary endpoint of hospitalization due to COVID-19 is reported below per site.

Kaplan-Meier estimates of the proportion of subject which were not hospitalized with a positive COVID-19 PCR test result versus time. `ifelse(CLOSED == T, "The figure is stratified by study arm. ", "")` The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized `ifelse(CLOSED == T, " , stratified by study arm", "")`, versus time. A figure with a enhanced y-axis is shown as well for clearer visualization.

Kaplan-Meier plots stratified per risk factor can be found in Appendix E.

Primary endpoint analysis

A Cox proportional hazard model was utilized to assess the statistical significance of the treatment arm on the primary endpoint. The dataset contains the intention-to-treat population and included censoring because of withdrawal from the trial or lost to follow-up, or death, as shown above in the Kaplan-Meier curves.

The hazard ratio was `round(summary(m0_group_itt)$coefficients[2], 2) (round(exp(confint(m0_group_itt))[1, 2]) - #round(exp(confint(m0_group_itt))[2, 1] 95% confidence interval) for the Arm2 arm relative to the Arm1 arm (p-value = round(summary(m0_group_itt)$coefficients[5], 3))`.

The results were not sensitive to different initial estimates corresponding to a hazard ratio range from 0.1 to 10 (100-fold). The forest plot below visualized the hazard ratio including its 95% confidence interval.

AIC = Akaike information criterion

Secondary endpoints

The table below shows the secondary endpoints and their corresponding section of the report.

Secondary objective	Description	Report section
a	To determine the incidence of SARS-CoV-2 infection in HCW by molecular or serological testing (as available) at entry, 10, 26 and/or 52 weeks. The timing of serological testing will be adjusted to the availability of validated tests and the course the epidemic of COVID-19 will take.	SARS-CoV-2 antibodies (page 31)
b	To compare the incidence of symptoms of respiratory tract infection per arm.	Respiratory tract infections events (page 34)
c	To compare the number of days of (unplanned) absenteeism because of documented SARS-CoV-2 infection or COVID-19 per arm.	Recorded sick leave during the trial (page 36)
d	To compare the number of days of (unplanned) absenteeism for any reason per arm.	Recorded sick leave during the trial (page 36)
e	To compare the incidence of hospitalization of HCW for any reason per arm.	Total hospitalization (page 34)
f	To compare the incidence of intensive care admission of HCW due to COVID-19 per arm.	Hospitalized with ventilation (HS = 6) (page 29)
g	To compare the incidence of intensive care admission of HCW for any reason per arm.	Intensive care admissions (page 39)
h	To compare the incidence of death of HCW due to COVID-19 per arm.	Death (HS = 7) (page 29)
i	To compare the incidence of death of HCW for any reason per arm.	Deaths (page 55)
j	To describe the prevalence of latent TB infection as determined by interferon gamma release assay (IGRA) at enrolment and at week 52.	Tuberculosis (page 22)
k	To compare the incidence of active TB of HCW per arm.	Active tuberculosis (page 24)
l	To compare the effect of latent TB infection on morbidity and mortality of HCW due to COVID-19 per arm.	Latent tuberculosis infection and COVID-19 or respiratory tract infections (page 33)
m	To compare the incidence of grade 2 or higher adverse events and vaccination site reactions per arm.	Grade 2 or higher (page 39)

COVID-19 cases

Please be aware not all participants are tested as per guidelines. Participants can come in with proof of a positive test from a third party (PCR or other). Serology testing at week 0, 10, 26, and 52 showed IgG-based test results of SARS-CoV-2. Health status (HS) score reflects the highest HS recorded per event.

Total cases of COVID-19

```
ifelse(CLOSED == T, paste("In total, ", n_c19_event_group1, " COVID-19 cases diagnosed  
with a positive PCR test have been reported in ", part_c19_group1, " (", round(100 *  
part_c19_group1 / part_group1, 1), "%) participants in the ", Arm1, " arm. \nIn total,  
", n_c19_event_group2, " COVID-19 cases diagnosed with a positive PCR test have been  
reported in ", part_c19_group2, " (", round(100 * part_c19_group2 / part_group2, 1),  
"%) participants in the ", Arm2, " arm.", sep = ''), paste("In total, ", n_c19_event, "  
COVID-19 cases diagnosed with a positive PCR test have been reported in ", part_c19, "  
(", round(100 * part_c19 / part_enrol, 1), "%) participants.", sep = ''))
```

The endpoint is here reported based on the intention-to-treat population with right censoring only at withdrawal of consent or lost to follow-up, death, or the end of the trial.

Kaplan-Meier estimates of the proportion of subject without a positive COVID-19 PCR test result versus time. `ifelse(CLOSED == T, "The figure is stratified by study arm. ", "")`The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized.`ifelse(CLOSED == T, ", stratified by study arm", "")`, versus time. A figure with a enhanced y-axis is shown as well for clearer visualization.

Mild (HS = 1)

```
ifelse(CLOSED == T, paste("In total, ", n_c19_HS1_group1, " (", round(100 * n_c19_HS1_group1  
/ part_group1, 1), "%) participants had a positive PCR test result for COVID-19 with  
mild symptoms in the ", Arm1, " arm. \nIn total, ", n_c19_HS1_group2, " (", round(100  
* n_c19_HS1_group2 / part_group2, 1), "%) participants had a positive PCR test result  
for COVID-19 with mild symptoms in the ", Arm2, " arm.", sep = ''), paste("In total, ",  
n_c19_HS1, " (", round(100 * n_c19_HS1 / part_enrol, 1), "%) participants had a positive  
PCR test result for COVID-19 with mild symptoms.", sep = ''))
```

Moderate (HS = 2)

```
ifelse(CLOSED == T, paste("In total, ", n_c19_HS2_group1, " (", round(100 * n_c19_HS2_group1  
/ part_group1, 1), "%) participants had a positive PCR test result for COVID-19 with  
moderate symptoms in the ", Arm1, " arm. \nIn total, ", n_c19_HS2_group2, " (", round(100  
* n_c19_HS2_group2 / part_group2, 1), "%) participants had a positive PCR test result for  
COVID-19 with moderate symptoms in the ", Arm2, " arm.", sep = ''), paste("In total, ",  
n_c19_HS2, " (", round(100 * n_c19_HS2 / part_enrol, 1), "%) participants had a positive  
PCR test result for COVID-19 with moderate symptoms.", sep = ''))
```

Severe (HS = 3)

```
ifelse(CLOSED == T, paste("In total, ", n_c19_HS3_group1, " (", round(100 * n_c19_HS3_group1  
/ part_group1, 1), "%) participants had a positive PCR test result for COVID-19 with
```

```
severe symptoms in the ", Arm1, " arm. \nIn total, ", n_c19_HS3_group2, " (", round(100 * n_c19_HS3_group2 / part_group2, 1), "%) participants had a positive PCR test result for COVID-19 with severe symptoms in the ", Arm2, " arm.", sep = ''), paste("In total, ", n_c19_HS3, " (", round(100 * n_c19_HS3 / part_enrol, 1), "%) participants had a positive PCR test result for COVID-19 with severe symptoms.", sep = ''))
```

Hospitalized (HS = 4)

```
ifelse(CLOSED == T, paste("In total, ", n_c19_HS4_group1, " (", round(100 * n_c19_HS4_group1 / part_group1, 1), "%) hospitalized participants had a positive PCR test result for COVID-19 in the ", Arm1, " arm. \nIn total, ", n_c19_HS4_group2, " (", round(100 * n_c19_HS4_group2 / part_group2, 1), "%) hospitalized participants had a positive PCR test result for COVID-19 in the ", Arm2, " arm.", sep = ''), paste("In total, ", n_c19_HS4, " (", round(100 * n_c19_HS4 / part_enrol, 1), "%) hospitalized participants had a positive PCR test result for COVID-19.", sep = ''))
```

Hospitalized with oxygen (HS = 5)

```
ifelse(CLOSED == T, paste("In total, ", n_c19_HS5_group1, " (", round(100 * n_c19_HS5_group1 / part_group1, 1), "%) hospitalized participants with oxygen had a positive PCR test result for COVID-19 in the ", Arm1, " arm. \nIn total, ", n_c19_HS5_group2, " (", round(100 * n_c19_HS5_group2 / part_group2, 1), "%) hospitalized participants with oxygen had a positive PCR test result for COVID-19 in the ", Arm2, " arm.", sep = ''), paste("In total, ", n_c19_HS5, " (", round(100 * n_c19_HS5 / part_enrol, 1), "%) hospitalized participants with oxygen had a positive PCR test result for COVID-19.", sep = ''))
```

Hospitalized with ventilation (HS = 6)

```
ifelse(CLOSED == T, paste("In total, ", n_c19_HS6_group1, " (", round(100 * n_c19_HS6_group1 / part_group1, 1), "%) hospitalized participants with ventilation had a positive PCR test result for COVID-19 in the ", Arm1, " arm. \nIn total, ", n_c19_HS6_group2, " (", round(100 * n_c19_HS6_group2 / part_group2, 1), "%) hospitalized participants with oxygen had a positive PCR test result for COVID-19 in the ", Arm2, " arm.", sep = ''), paste("In total, ", n_c19_HS6, " (", round(100 * n_c19_HS6 / part_enrol, 1), "%) hospitalized participants with ventilation had a positive PCR test result for COVID-19.", sep = ''))
```

Death (HS = 7)

```
ifelse(CLOSED == T, paste("In total, ", n_c19_HS7_group1, " (", round(100 * n_c19_HS7_group1 / part_group1, 1), "%) participant who died had a positive PCR test result for COVID-19 in the ", Arm1, " arm. \nIn total, ", n_c19_HS7_group2, " (", round(100 * n_c19_HS7_group2 / part_group2, 1), "%) participant who died had a positive PCR test result for COVID-19 in the ", Arm2, " arm.", sep = ''), paste("In total, ", n_c19_HS7, " (", round(100 * n_c19_HS7 / part_enrol, 1), "%) participant who died had a positive PCR test result for COVID-19.", sep = ''))
```

Repeated infection

```
ifelse(CLOSED == T, paste(part_repeated_infect_group1, " (", round(100 * part_repeated_infect_group1 / part_group1, 1), "%) participant had a repeated COVID-19 infection in the ", Arm1, "
```

```
arm. \n", part_repeated_infect_group2, " (", round(100 * part_repeated_infect_group2
/ part_group2, 1), "%) participant had a repeated COVID-19 infection in the ", Arm2, "
arm.", sep = ''), paste(part_repeated_infect, " (", round(100 * part_repeated_infect /
part_enrol, 1), "%) participant had a repeated COVID-19 infection.", sep = ''))
```

Post COVID-19 viral syndrome

```
ifelse(CLOSED == T, paste(nrow(part_post_group1), " (", round(100 * nrow(part_post_group1)
/ part_c19_group1, 1), "%) participants experienced post COVID-19 viral syndrome in the
", Arm1, " arm. \n", nrow(part_post_group2), " (", round(100 * nrow(part_post_group2)
/ part_c19_group2, 1), "%) participants experienced post COVID-19 viral syndrome in the
", Arm2, " arm.", sep = ''), paste(nrow(part_post), " (", round(100 * nrow(part_post) /
part_c19, 1), "%) participants experienced post COVID-19 viral syndrome.", sep = ''))
```

SARS-CoV-2 antibodies

Overall ifelse(CLOSED == T, paste(n_c19_serology_group1, " (", round(100 * n_c19_serology_group1 / part_c19_serology_group1, 1), "%, n=", part_c19_serology_group1, ") participants had at least one positive serology result for SARS-CoV-2 based on IgG in the ", Arm1, " arm.\n", n_c19_serology_group2, " (", round(100 * n_c19_serology_group2 / part_c19_serology_group2, 1), "%, n=", part_c19_serology_group2, ") participants had at least one positive serology result for SARS-CoV-2 based on IgG in the ", Arm2, " arm.", sep = ''), paste(n_c19_serology, " (", round(100 * n_c19_serology / part_c19_serology, 1), "%, n=", part_c19_serology, ") participants had at least one positive serology result for SARS-CoV-2 based on IgG.", sep = ''))

Without COVID-19 symptoms ifelse(CLOSED == T, paste(n_c19_serology_nosym_group1, " (", round(100 * n_c19_serology_nosym_group1 / part_c19_serology_group1, 1), "%, n=", part_c19_serology_group1, ") participants had a positive serology result (at baseline or week 10, 26, or 52, or at an unknown week) for SARS-CoV-2 based on IgG but showed no symptoms (no COVID-19 events reported) in the ", Arm1, " arm, assuming that those positive at baseline did not report any symptoms before they were included. \n", n_c19_serology_nosym_group2, " (", round(100 * n_c19_serology_nosym_group2 / part_c19_serology_group2, 1), "%, n=", part_c19_serology_group2, ") participants had a positive serology result (at baseline or week 10, 26, or 52, or at an unknown week) for SARS-CoV-2 based on IgG but showed no symptoms (no COVID-19 events reported) in the ", Arm2, " arm, assuming that those positive at baseline did not report any symptoms before they were included.", sep = ''), paste(n_c19_serology_nosym, " (", round(100 * n_c19_serology_nosym / part_c19_serology, 1), "%, n=", part_c19_serology, ") participants had a positive serology result (at baseline or week 10, 26, or 52, or at an unknown week) for SARS-CoV-2 based on IgG but showed no symptoms (no COVID-19 events reported), assuming that those positive at baseline did not report any symptoms before they were included.", sep = ''))

Baseline ifelse(CLOSED == T, paste(n_c19_serology_baseline_group1, " (", round(100 * n_c19_serology_baseline_group1 / part_c19_serology_week0_group1, 1), "%, n=", part_c19_serology_week0_group1, ") participants had a positive serology result for SARS-CoV-2 based on IgG at baseline in the ", Arm1, " arm. \n", n_c19_serology_baseline_group2, " (", round(100 * n_c19_serology_baseline_group2 / part_c19_serology_week0_group2, 1), "%, n=", part_c19_serology_week0_group2, ") participants had a positive serology result for SARS-CoV-2 based on IgG at baseline in the ", Arm2, " arm.", sep = ''), paste(n_c19_serology_baseline, " (", round(100 * n_c19_serology_baseline / part_c19_serology_week0, 1), "%, n=", part_c19_serology_week0, ") participants had a positive serology result for SARS-CoV-2 based on IgG at baseline.", sep = ''))
ifelse(CLOSED == T, paste("A Pearson's chi-squared test for the baseline serology between the two groups showed a p-value of ", round(prop.test(x = c(n_c19_serology_baseline_group1, n_c19_serology_baseline_group2), n = c(part_c19_serology_week0_group1, part_c19_serology_week0_group2))), 3), '.', sep = ''), '')
ifelse(CLOSED == T, paste('Of these baseline seropositive participants, ', n_rti_serobaseline_event_group1, " (", round(100 * n_rti_serobaseline_event_group1 / n_c19_serology_baseline_group1, 1), "%) participants had an RTI event after enrolment in the ", Arm1, " arm, and ", n_rti_serobaseline_event_group2, " (", round(100 * n_rti_serobaseline_event_group2 / n_c19_serology_baseline_group2, 1), "%) participants had an RTI event after enrolment in the ", Arm2, " arm.", sep = ''), paste('Of these baseline seropositive participants, ', n_rti_serobaseline_event, " (", round(100 * n_rti_serobaseline_event / n_c19_serology_baseline, 1), "%) participants had an RTI event after enrolment.", sep = ''))
ifelse(CLOSED == T, paste('Of these baseline seropositive participants, ', n_c19_serobaseline_event_group1, " (", round(100 * n_c19_serobaseline_event_group1 / n_c19_serology_baseline_group1, 1), "%) participant had a COVID-19 event after enrolment in the ", Arm1, " arm, and ",

```
n_c19_serobaseline_event_group2, " (", round(100 * n_c19_serobaseline_event_group2 /
n_c19_serology_baseline_group2, 1), "%) participant had a COVID-19 event after enrolment
in the ", Arm2, " arm.", sep = ' '), paste('Of these baseline seropositive participants,
', n_c19_serobaseline_event, " (", round(100 * n_c19_serobaseline_event / n_c19_serology_baseline,
1), "%) participant had a COVID-19 event after enrolment.", sep = ' '))
```

```
Week 10 ifelse(CLOSED == T, paste(n_c19_serology_week10_group1, " (", round(100 *
n_c19_serology_week10_group1 / part_c19_serology_week10_group1, 1), "%, n=", part_c19_serology_week10_g
") participants had a positive serology result for SARS-CoV-2 based on IgG at week 10 in
the ", Arm1, " arm. \n", n_c19_serology_week10_group2, " (", round(100 * n_c19_serology_week10_group2
/ part_c19_serology_week10_group2, 1), "%, n=", part_c19_serology_week10_group2, ")
participants had a positive serology result for SARS-CoV-2 based on IgG at week 10 in the
", Arm2, " arm.", sep = ' '), paste(n_c19_serology_week10, " (", round(100 * n_c19_serology_week10
/ part_c19_serology_week10, 1), "%, n=", part_c19_serology_week10, ") participants had a
positive serology result for SARS-CoV-2 based on IgG at week 10.", sep = ' '))
```

```
Week 26 ifelse(CLOSED == T, paste(n_c19_serology_week26_group1, " (", round(100 *
n_c19_serology_week26_group1 / part_c19_serology_week26_group1, 1), "%, n=", part_c19_serology_week26_g
") participants had a positive serology result for SARS-CoV-2 based on IgG at week 26 in
the ", Arm1, " arm. \n", n_c19_serology_week26_group2, " (", round(100 * n_c19_serology_week26_group2
/ part_c19_serology_week26_group2, 1), "%, n=", part_c19_serology_week26_group2, ")
participants had a positive serology result for SARS-CoV-2 based on IgG at week 26 in the
", Arm2, " arm.", sep = ' '), paste(n_c19_serology_week26, " (", round(100 * n_c19_serology_week26
/ part_c19_serology_week26, 1), "%, n=", part_c19_serology_week26, ") participants had a
positive serology result for SARS-CoV-2 based on IgG at week 26.", sep = ' '))
```

```
Week 52 ifelse(CLOSED == T, paste(n_c19_serology_week52_group1, " (", ifelse(part_c19_serology_week52
== 0, 0, round(100 * n_c19_serology_week52_group1 / part_c19_serology_week52_group1, 1)),
"%, n=", part_c19_serology_week52_group1, ") participants had a positive serology result
for SARS-CoV-2 based on IgG at week 52 in the ", Arm1, " arm. \n", n_c19_serology_week52_group2,
" (", ifelse(part_c19_serology_week52_group2 == 0, 0, round(100 * n_c19_serology_week52_group2
/ part_c19_serology_week52_group2, 1)), "%, n=", part_c19_serology_week52_group2, ")
participants had a positive serology result for SARS-CoV-2 based on IgG at week 52 in the
", Arm2, " arm.", sep = ' '), paste(n_c19_serology_week52, " (", ifelse(part_c19_serology_week52
== 0, 0, round(100 * n_c19_serology_week52 / part_c19_serology_week52, 1)), "%, n=",
part_c19_serology_week52, ") participants had a positive serology result for SARS-CoV-2
based on IgG at week 52.", sep = ' '))
```

```
Seroconversion ifelse(CLOSED == T, paste(n_c19_seroconversion_group1, " (", round(100 *
n_c19_seroconversion_group1 / part_c19_secondssample_group1, 1), "%, n=", part_c19_secondssample_group1,
") participants had seroconversion for SARS-CoV-2 in the ", Arm1, " arm. \n", n_c19_seroconversion_gro
" (", round(100 * n_c19_seroconversion_group2 / part_c19_secondssample_group2, 1), "%,
n=", part_c19_secondssample_group2, ") participants had seroconversion for SARS-CoV-2 in
the ", Arm2, " arm.", sep = ' '), paste(n_c19_seroconversion, " (", round(100 * n_c19_seroconversion
/ part_c19_secondssample, 1), "%, n=", part_c19_secondssample, ") participants had seroconversion
for SARS-CoV-2.", sep = ' '))
ifelse(CLOSED == T, paste('Of these seroconverted participants, ', n_rti_seroconversion_event_group1,
" (", round(100 * n_rti_seroconversion_event_group1 / n_c19_seroconversion_group1, 1),
"%) participants had an RTI event in the seroconversion period (period between last
negative and first positive serology result) in the ", Arm1, " arm, and ", n_rti_seroconversion_event_g
" (", round(100 * n_rti_seroconversion_event_group2 / n_c19_seroconversion_group2, 1),
```



```
"%) participants had an RTI event in the seroconversion period in the ", Arm2, " arm.",
sep = ''), paste('Of these seroconverted participants, ', n_rti_seroconversion_event, "
(", round(100 * n_rti_seroconversion_event / n_c19_seroconversion, 1), "%) participants
had an RTI event in the seroconversion period (period between last negative and first
positive serology result).", sep = '')) ifelse(CLOSED == T, paste('Of these seroconverted
participants, ', n_c19_seroconversion_event_group1, " (", round(100 * n_c19_seroconversion_event_group1
/ n_c19_seroconversion_group1, 1), "%) participants had a COVID-19 event in the seroconversion
period in the ", Arm1, " arm, and ", n_c19_seroconversion_event_group2, " (", round(100
* n_c19_seroconversion_event_group2 / n_c19_seroconversion_group2, 1), "%) participants
had a COVID-19 event in the seroconversion period in the ", Arm2, " arm.", sep = ''),
paste('Of these seroconverted participants, ', n_c19_seroconversion_event, " (", round(100
* n_c19_seroconversion_event / n_c19_seroconversion, 1), "%) participants had a COVID-19
event in the seroconversion period.", sep = ''))
```

Serology sampling takes place at baseline and at week 10, 26, and 52. Percentages reported are relative to the total number of samples with results reported at that timepoint, or to the number of participants with at least two serology results for the seroconversion (reported as n).

Latent tuberculosis infection and COVID-19 or respiratory tract infections

```
ifelse(CLOSED == T, paste("Of the reported COVID-19 cases, ", n_c19_TB_group1, " were
preceded by a positive IGRA test for TB in ", part_c19_TB_group1, " (", round(100 *
part_c19_TB_group1 / part_c19_group1, 1), "%) participants in the ", Arm1, " arm, and
", n_c19_TB_group2, " in ", part_c19_TB_group2, " (", round(100 * part_c19_TB_group2 /
part_c19_group2, 1), "%) participants in the ", Arm2, " arm.", sep = ''), paste("Of the
reported COVID-19 cases, ", n_c19_TB, " were preceded by a positive IGRA test for TB in
", part_c19_TB, " (", round(100 * part_c19_TB / part_c19, 1), "%) participants.", sep =
''))
```

```
ifelse(CLOSED == T, paste("Of the reported RTI cases, ", n_rti_TB_group1, " were preceded
by a positive IGRA test for TB in ", part_rti_TB_group1, " (", round(100 * part_rti_TB_group1
/ part_ae_rti_group1, 1), "%) participants in the ", Arm1, " arm, and ", n_rti_TB_group2,
" in ", part_rti_TB_group2, " (", round(100 * part_rti_TB_group2 / part_ae_rti_group2,
1), "%) participants in the ", Arm2, " arm.", sep = ''), paste("Of the reported RTI
cases, ", n_rti_TB, " were preceded by a positive IGRA test for TB in ", part_rti_TB,
" (", round(100 * part_rti_TB / part_ae_rti, 1), "%) participants.", sep = ''))
```

Efficacy summary

Total hospitalization

Total hospitalization included participants hospitalized (i.e. $HS \geq 4$) due to COVID-19 (see primary endpoint) as well as due to other causes.

```
ifelse(CLOSED == T, paste(n_hosp_group1, " (", round(100 * n_hosp_group1 / part_group1, 1), "%) participants were hospitalized in the ", Arm1, " arm. \n", n_hosp_group2, " (", round(100 * n_hosp_group2 / part_group2, 1), "%) participants were hospitalized in the ", Arm2, " arm.", sep = ''), paste(n_hosp, " (", round(100 * n_hosp / part_enrol, 1), "%) participants were hospitalized.", sep = ''))
```

The endpoint is here reported based on the intention-to-treat population with right censoring only at withdrawal of consent or lost to follow-up, death, or the end of the trial.

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. `ifelse(CLOSED == T, "The figure is stratified by study arm. ", "")` The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized `ifelse(CLOSED == T, "The figure is stratified by study arm. ", "")`, versus time. A figure with a enhanced y-axis is shown as well for clearer visualization.

Kaplan-Meier plots stratified per risk factor can be found in Appendix E.

Respiratory tract infections events

```
ifelse(CLOSED == T, paste(part_ae_rti_group1, " (", round(100 * part_ae_rti_group1 / part_group1, 1), "%) participants had an RTI event in the ", Arm1, " arm. \n", part_ae_rti_group2, " (", round(100 * part_ae_rti_group2 / part_group2, 1), "%) participants had an RTI event in the ", Arm2, " arm.", sep = ''), paste(part_ae_rti, " (", round(100 * part_ae_rti / part_enrol, 1), "%) participants had an RTI event.", sep = ''))
```

The endpoint is here reported based on the intention-to-treat population with right censoring only at withdrawal of consent or lost to follow-up, death, or the end of the trial.

The numbers in the risk table are based on the number of participants that have reached that timepoint.

Kaplan-Meier estimates of the proportion of subject which did not have an RTI event ($HS > 0$) versus time. Only first event is shown. `ifelse(CLOSED == T, "The figure is stratified by study arm. ", "")` The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of an RTI event `ifelse(CLOSED == T, "The figure is stratified by study arm. ", "")`, versus time. A figure with a enhanced y-axis is shown as well for clearer visualization.

Kaplan-Meier plots for all health status scores can be found in Appendix F.

Health status

Health status	Description
0	Healthy
1	Mild symptoms
2	Moderate symptoms
3	Severe symptoms
4	Hospitalized
5	Hospitalized, oxygen
6	Hospitalized, ventilated
7	Death

Health status over time for respiratory tract infections

Health status over time for RTIs for each subject in the study (dashed lines).`ifelse(CLOSED == T, " The figure is stratified and coloured by the study arm.", "")`

A health status score of 4 or higher means hospitalization of the participant (horizontal dotted line).

The figures below shows the proportion of participant-events for each HS over time. Participant-events was used as participants could have multiple events with separate HS at the same time.

Transitions

The table below shows the number of transitions between the HS states (from the HS identified in the row to the HS identified in the column).

From ↓ to →	0	1	2	3	4	5	6	7
0	hs_time_transition_00 %>% nrow()	hs_time_transition_01 %>% nrow()	hs_time_transition_02 %>% nrow()	hs_time_transition_03 %>% nrow()	hs_time_transition_04 %>% nrow()	hs_time_transition_05 %>% nrow()	hs_time_transition_06 %>% nrow()	hs_time_transition_07 %>% nrow()
1	hs_time_transition_10 %>% nrow()	hs_time_transition_11 %>% nrow()	hs_time_transition_12 %>% nrow()	hs_time_transition_13 %>% nrow()	hs_time_transition_14 %>% nrow()	hs_time_transition_15 %>% nrow()	hs_time_transition_16 %>% nrow()	hs_time_transition_17 %>% nrow()
2	hs_time_transition_20 %>% nrow()	hs_time_transition_21 %>% nrow()	hs_time_transition_22 %>% nrow()	hs_time_transition_23 %>% nrow()	hs_time_transition_24 %>% nrow()	hs_time_transition_25 %>% nrow()	hs_time_transition_26 %>% nrow()	hs_time_transition_27 %>% nrow()
3	hs_time_transition_30 %>% nrow()	hs_time_transition_31 %>% nrow()	hs_time_transition_32 %>% nrow()	hs_time_transition_33 %>% nrow()	hs_time_transition_34 %>% nrow()	hs_time_transition_35 %>% nrow()	hs_time_transition_36 %>% nrow()	hs_time_transition_37 %>% nrow()
4	hs_time_transition_40 %>% nrow()	hs_time_transition_41 %>% nrow()	hs_time_transition_42 %>% nrow()	hs_time_transition_43 %>% nrow()	hs_time_transition_44 %>% nrow()	hs_time_transition_45 %>% nrow()	hs_time_transition_46 %>% nrow()	hs_time_transition_47 %>% nrow()
5	hs_time_transition_50 %>% nrow()	hs_time_transition_51 %>% nrow()	hs_time_transition_52 %>% nrow()	hs_time_transition_53 %>% nrow()	hs_time_transition_54 %>% nrow()	hs_time_transition_55 %>% nrow()	hs_time_transition_56 %>% nrow()	hs_time_transition_57 %>% nrow()
6	hs_time_transition_60 %>% nrow()	hs_time_transition_61 %>% nrow()	hs_time_transition_62 %>% nrow()	hs_time_transition_63 %>% nrow()	hs_time_transition_64 %>% nrow()	hs_time_transition_65 %>% nrow()	hs_time_transition_66 %>% nrow()	hs_time_transition_67 %>% nrow()

Health status scores over time stratified per risk factors can be found in Appendix H.

Sick leave

Baseline sick leave

Baseline sick leave recorded at enrolment is plotted below `ifelse(CLOSED == T, " per arm", "")`. This was the number of days sick leave taken in the four weeks prior to enrolment.

Recorded sick leave during the trial

During the monthly participant follow-up, the questionnaire asked how many days of sick leave were taken since the last contact. These days were summed per participant and shown below `ifelse(CLOSED == T, " per arm", "")`.

```
ifelse(CLOSED == T, paste(df_full %>% filter(group == Arm1) %>% filter(sick_leave_any == 'Yes') %>% distinct(PID) %>% nrow(), " participants in the ", Arm1, " arm took sick leave, ", df_full %>% filter(sick_leave_any == 'Yes', c19_positive == 'Yes', group == Arm1) %>% distinct(PID) %>% nrow(), " of which had a preceding positive COVID-19 test. ",
```

```

df_full %>% filter(group == Arm2) %>% filter(sick_leave_any == 'Yes') %>% distinct(PID)
%>% nrow(), " participants in the ", Arm2, " arm took sick leave, ", df_full %>% filter(sick_leave_any
== 'Yes', c19_positive == 'Yes', group == Arm2) %>% distinct(PID) %>% nrow()," of which
had a preceding positive COVID-19 test.", sep = ""), paste(df_full %>% filter(sick_leave_any
== 'Yes') %>% distinct(PID) %>% nrow(), " participants took sick leave, ", df_full %>%
filter(sick_leave_any == 'Yes', c19_positive == 'Yes') %>% distinct(PID) %>% nrow()," of
which had a preceding positive COVID-19 test.", sep = ""))

ifelse(CLOSED == T, paste('Mean number of sick days after a positive COVID-19 test was
', round(df_full %>% filter(sick_leave_any == 'Yes', c19_positive == 'Yes', group ==
Arm1) %>% summarize(mean = mean(sick_leave_days))), ' days in the ', Arm1, ' arm, and ',
round(df_full %>% filter(sick_leave_any == 'Yes', c19_positive == 'Yes', group == Arm2)
%>% summarize(mean = mean(sick_leave_days))), ' days in the ', Arm2, ' arm.', sep = ''),
paste('Mean number of sick days after a positive COVID-19 test was ', round(df_full %>%
filter(sick_leave_any == 'Yes', c19_positive == 'Yes') %>% summarize(mean = mean(sick_leave_days))),
' days.', sep = '')) ifelse(CLOSED == T, paste('Mean number of sick days without a positive
COVID-19 test was ', round(df_full %>% filter(sick_leave_any == 'Yes', c19_positive !=
'Yes', group == Arm1) %>% summarize(mean = mean(sick_leave_days))), ' days in the ',
Arm1, ' arm, and ', round(df_full %>% filter(sick_leave_any == 'Yes', c19_positive !=
'Yes', group == Arm2) %>% summarize(mean = mean(sick_leave_days))), ' days in the ',
Arm2, ' arm.', sep = ''), paste('Mean number of sick days without a positive COVID-19
test was ', round(df_full %>% filter(sick_leave_any == 'Yes', c19_positive != 'Yes') %>%
summarize(mean = mean(sick_leave_days))), ' days.', sep = ''))

```

Adverse events

Reported events

Events are recorded in three categories: respiratory tract infection, injection site reaction, and other. For each event, a health status score was recorded. The highest health status score per event, defined by unique event number per participant, is reported in this report.

n_ae total adverse events occurred in part_ae participants.

```
ifelse(CLOSED == T, paste(n_ae_rti_group1, "respiratory tract infection events occurred in", part_ae_rti_group1, "participants in the ", Arm1, " arm, and", n_ae_rti_group2, "respiratory tract infection events occurred in", part_ae_rti_group2, "participants in the ", Arm2, " arm."), paste(n_ae_rti, "respiratory tract infection events occurred in", part_ae_rti, "participants."))
```

n_ae_isr injection site reaction adverse events occurred in part_ae_isr participants.

```
ifelse(CLOSED == T, paste(n_ae_other_group1, "other adverse events occurred in", part_ae_other_group1, "participants in the ", Arm1, " arm, and", n_ae_other_group2, "other adverse events occurred in", part_ae_other_group2, "participants in the ", Arm2, " arm."), paste(n_ae_other, "other adverse events occurred in", part_ae_other, "participants"))
```

```
ifelse(CLOSED == T, paste("Of the ", n_ae_rti, " respiratory tract infection events, ", n_ae_rti_mild, " (", round(100 * n_ae_rti_mild / n_ae_rti), "%, ", ae_rti_mild %>% filter(group == Arm1) %>% nrow(), " in the ", Arm1, " arm and ", ae_rti_mild %>% filter(group == Arm2) %>% nrow(), " in the ", Arm2, " arm) were considered mild and ", n_ae_rti_moderate, " (", round(100 * n_ae_rti_moderate / n_ae_rti), "%, ", ae_rti_moderate %>% filter(group == Arm1) %>% nrow(), " in the ", Arm1, " arm and ", ae_rti_moderate %>% filter(group == Arm2) %>% nrow(), " in the ", Arm2, " arm) were considered moderate.", sep = ''), paste("Of the ", n_ae_rti, " respiratory tract infection events, ", n_ae_rti_mild, " (", round(100 * n_ae_rti_mild / n_ae_rti), "%) were considered mild and ", n_ae_rti_moderate, " (", round(100 * n_ae_rti_moderate / n_ae_rti), "%) were considered moderate.", sep = ''))
```

Of the n_ae_isr injection site reaction adverse events, n_ae_isr_mild (round(100 * n_ae_isr_mild / n_ae_isr)%) were considered mild and n_ae_isr_moderate (round(100 * n_ae_isr_moderate / n_ae_isr)%) were considered moderate.

```
ifelse(CLOSED == T, paste("Of the ", n_ae_other, " other adverse events, ", n_ae_other_mild, " (", round(100 * n_ae_other_mild / n_ae_other), "%, ", ae_other_mild %>% filter(group == Arm1) %>% nrow(), " in the ", Arm1, " arm and ", ae_other_mild %>% filter(group == Arm2) %>% nrow(), " in the ", Arm2, " arm) were considered mild and ", n_ae_other_moderate, " (", round(100 * n_ae_other_moderate / n_ae_other), "%, ", ae_other_moderate %>% filter(group == Arm1) %>% nrow(), " in the ", Arm1, " arm and ", ae_other_moderate %>% filter(group == Arm2) %>% nrow(), " in the ", Arm2, " arm) were considered moderate.", sep = ''), paste("Of the ", n_ae_other, " other adverse events, ", n_ae_other_mild, " (", round(100 * n_ae_other_mild / n_ae_other), "%) were considered mild and ", n_ae_other_moderate, " (", round(100 * n_ae_other_moderate / n_ae_other), "%) were considered moderate.", sep = ''))
```

n_sae_previous serious adverse events were reported in the previous DSMB report.

```
ifelse(n_sae - n_sae_previous == 0, "There have been no additional serious adverse events since the last DSMB meeting.", ifelse(n_sae - n_sae_previous == 1, paste(n_sae - n_sae_previous, "additional serious adverse events was reported since the last DSMB meeting."), paste(n_sae - n_sae_previous, "additional serious adverse events were reported since the last DSMB meeting.")))
```

```
ifelse(CLOSED == T, paste("In total,", n_sae_group1, "serious adverse events were reported in", part_sae_group1, "participants in the ", Arm1, " arm, and ", n_sae_group2, "serious
```

```
adverse events were reported in", part_sae_group2, "participants in the ", Arm2, " arm."),
paste("In total,", n_sae, "serious adverse events were reported in", part_sae, "participants.))
paste("SAEs were ", ifelse(n_sae_unrelated > 0, 'unrelated', ''), ifelse(n_sae_unlikelyrelated
> 0, ', or unlikely related', ''), ifelse(n_sae_possiblyrelated > 0, ', or possibly
related', ''), ifelse(n_sae_probablyrelated > 0, ', or probably related', ''), ifelse(n_sae_definitely
> 0, ', or definitely related', ''), " to the intervention.", sep = '')
```

Grade 2 or higher

```
ifelse(CLOSED == T, paste(n_ae_grade_234_group1, " adverse events of grade 2 or higher
occured in ", part_ae_grade_234_group1, " (", round(100 * part_ae_grade_234_group1 /
part_group1, 1), "%) participants in the ", Arm1, " arm, and ", n_ae_grade_234_group2,
" adverse events of grade 2 or higher occured in ", part_ae_grade_234_group2, " (",
round(100 * part_ae_grade_234_group2 / part_group1, 1), "%) participants in the ", Arm2,
" arm.", sep = ''), paste(n_ae_grade_234, " adverse events of grade 2 or higher occured
in ", part_ae_grade_234, " (", round(100 * part_ae_grade_234 / part_group1, 1), "%)
participants", sep = '')) ifelse(CLOSED == T, paste('Of these, ', n_ae_grade_3_group1,
" were grade 3 (in ", part_ae_grade_3_group1, " (", round(100 * part_ae_grade_3_group1
/ part_group1, 1), "%) participants) and ", n_ae_grade_4_group1, " were grade 4 (in ",
part_ae_grade_4_group1, " (", round(100 * part_ae_grade_4_group1 / part_group1, 1), "%)
participants) in the ", Arm1, " arm, and ", n_ae_grade_3_group2, " were grade 3 (in ",
part_ae_grade_3_group2, " (", round(100 * part_ae_grade_3_group2 / part_group2, 1), "%)
participants) and ", n_ae_grade_4_group2, " were grade 4 (in ", part_ae_grade_4_group2,
" (", round(100 * part_ae_grade_4_group2 / part_group2, 1), "%) participants) in the
", Arm2, " arm.", sep = ''), paste('Of these, ', n_ae_grade_3, " were grade 3 (in ",
part_ae_grade_3, " (", round(100 * part_ae_grade_3 / part_enrol, 1), "%) participants)
and ", n_ae_grade_4, " were grade 4 (in ", part_ae_grade_4, " (", round(100 * part_ae_grade_4
/ part_enrol, 1), "%) participants).", sep = ''))

ifelse(CLOSED == T, paste(n_ae_grade_234_ISR_group1, " injection site reaction of grade 2
or higher occured in ", part_ae_grade_234_ISR_group1, " (", round(100 * part_ae_grade_234_ISR_group1
/ part_group1, 1), "%) participant in the ", Arm1, " arm, and ", n_ae_grade_234_ISR_group2,
" injection site reactions of grade 2 or higher occured in ", part_ae_grade_234_ISR_group2,
" (", round(100 * part_ae_grade_234_ISR_group2 / part_group1, 1), "%) participants in
the ", Arm2, " arm.", sep = ''), paste(n_ae_grade_234_ISR, " injection site reactions of
grade 2 or higher occured in ", part_ae_grade_234_ISR, " (", round(100 * part_ae_grade_234_ISR
/ part_group1, 1), "%) participants", sep = ''))
```

Intensive care admissions

```
ifelse(CLOSED == T, paste(n_ae_icu_group1, " total intensive care admissions occurred in
", part_ae_icu_group1, " participants in the ", Arm1, " arm, and ", n_ae_icu_group2,
" total intensive care admissions occurred in ", part_ae_icu_group2, " participants
in the ", Arm2, " arm.", sep = ''), paste(n_ae_icu, " total intensive care admissions
occurred in ", part_ae_icu, "participants", sep = '')) ifelse(CLOSED == T, paste('Of
these, ', nrow(c19_HS6_group1), " were due to COVID-19 in the ", Arm1, " arm, and ",
nrow(c19_HS6_group2), " in the ", Arm2, " arm.", sep = ''), paste('Of these, ', nrow(c19_HS6),
' were due to COVID-19', sep = ''))
```

More details are described per event type below. Full details of the adverse events can be found in Appendix B.

Health status score

Every adverse event received a health status score (see Health status definitions for their definitions). Health status scores had a weekly time resolution, within which interval the highest experienced health status was recorded.

Respiratory tract infection events

MedDRA lowest level terms were used to describe events, see for the full definitions Appendix I. Definition of event seriousness followed recorded corresponding health status (see Health status definitions).

Classification	Recorded events	Event details
Total events	n_ae_rti	RTI events in part_ae_rti participants.
Healthy events	n_ae_rti_healthy	if(CLOSED == F){unique(ae_rti_healthy\$LLT)} healthy RTI events in part_ae_rti_healthy participants. ifelse(CLOSED == T, if(CLOSED == T){unique(ae_rti_healthy\$LLT[ae_rti_healthy\$group == paste(nrow(ae_rti_healthy[ae_rti_healthy\$group == Arm1,])), " healthy RTI events in the ", Arm1, " arm", sep = ""), ""}) ifelse(CLOSED == T, if(CLOSED == T){unique(ae_rti_healthy\$LLT[ae_rti_healthy\$group == paste(nrow(ae_rti_healthy[ae_rti_healthy\$group == Arm2,])), " healthy RTI events in the ", Arm2, " arm", sep = ""), ""})
Mild events	n_ae_rti_mild	if(CLOSED == F){unique(ae_rti_mild\$LLT)} mild RTI events in part_ae_rti_mild participants. ifelse(CLOSED == T, if(CLOSED == T){unique(ae_rti_mild\$LLT[ae_rti_mild\$group == Arm1])}) paste(nrow(ae_rti_mild[ae_rti_mild\$group == Arm1,])), " mild RTI events in the ", Arm1, " arm", sep = ""), ""})

Classification	Recorded events	Event details
	<pre> ifelse(CLOSED == T, paste(nrow(ae_rti_mild[ae_rti_mild\$group == Arm2,])), " mild RTI events in the ", Arm2, " arm", sep = ""), "") </pre>	<pre> if(CLOSED == T){unique(ae_rti_mild\$LLT[ae_rti_mild\$group == Arm2])} </pre>
Moderate events	<pre> n_ae_rti_moderate moderate RTI events in part_ae_rti_moderate participants. ifelse(CLOSED == T, paste(nrow(ae_rti_moderate[ae_rti_moderate\$group == Arm1,])), " moderate RTI events in the ", Arm1, " arm", sep = ""), "") ifelse(CLOSED == T, paste(nrow(ae_rti_moderate[ae_rti_moderate\$group == Arm2,])), " moderate RTI events in the ", Arm2, " arm", sep = ""), "") </pre>	<pre> if(CLOSED == F){unique(ae_rti_moderate\$LLT)} if(CLOSED == T){unique(ae_rti_moderate\$LLT[ae_rti_moderate\$group == Arm1])} if(CLOSED == T){unique(ae_rti_moderate\$LLT[ae_rti_moderate\$group == Arm2])} </pre>
Severe events	<pre> n_ae_rti_severe severe RTI events in part_ae_rti_severe participants. ifelse(CLOSED == T, paste(nrow(ae_rti_severe[ae_rti_severe\$group == Arm1,])), " severe RTI events in the ", Arm1, " arm", sep = ""), "") </pre>	<pre> if(CLOSED == F){unique(ae_rti_severe\$LLT)} if(CLOSED == T){unique(ae_rti_severe\$LLT[ae_rti_severe\$group == Arm1])} if(CLOSED == T){unique(ae_rti_severe\$LLT[ae_rti_severe\$group == Arm2])} </pre>

Classification	Recorded events	Event details
		<pre> ifelse(CLOSED == T, paste(nrow(ae_rti_severe[ae_rti_severe\$group == Arm2,]), " severe RTI events in the ", Arm2, " arm", sep = ""), "") </pre>
Hospitalization	n_ae_rti_hosp RTI events leading to hospitalization in part_ae_rti_hosp participants.	<pre> if(CLOSED == F){unique(ae_rti_hosp\$LLT)} ifelse(CLOSED == T, paste(nrow(ae_rti_hosp[ae_rti_hosp\$group == Arm1,]), " RTI events leading to hospitalization in the ", Arm1, " arm", sep = ""), "") ifelse(CLOSED == T, paste(nrow(ae_rti_hosp[ae_rti_hosp\$group == Arm2,]), " RTI events leading to hospitalization in the ", Arm2, " arm", sep = ""), "") </pre>
Hospitalization with oxygen	n_ae_rti_hosp_oxy RTI events leading to hospitalization with oxygen in part_ae_rti_hosp_oxy participants.	<pre> if(CLOSED == F){unique(ae_rti_hosp_oxy\$LLT)} </pre>

Classification	Recorded events	Event details
	<pre> ifelse(CLOSED == T, paste(nrow(ae_rti_hosp_oxy[ae_rti_hosp_oxy\$group == Arm1,])), " RTI events leading to hospitalization with oxygen in the ", Arm1, " arm", sep = "", "") ifelse(CLOSED == T, paste(nrow(ae_rti_hosp_oxy[ae_rti_hosp_oxy\$group == Arm2,])), " RTI events leading to hospitalization with oxygen in the ", Arm2, " arm", sep = "", "") </pre>	<pre> if(CLOSED == T){unique(ae_rti_hosp_oxy\$LLT[ae_rti_hosp_oxy\$group == Arm1,])} if(CLOSED == T){unique(ae_rti_hosp_oxy\$LLT[ae_rti_hosp_oxy\$group == Arm2,])} </pre>
Hospitalization with ventilation	<pre> n_ae_rti_hosp_vent RTI event leading to hospitalization with ventilation in part_ae_rti_hosp_vent participant. </pre>	<pre> if(CLOSED == F){unique(ae_rti_hosp_vent\$LLT)} if(CLOSED == T){unique(ae_rti_hosp_vent\$LLT[ae_rti_hosp_vent\$group == Arm1,])} if(CLOSED == T){unique(ae_rti_hosp_vent\$LLT[ae_rti_hosp_vent\$group == Arm2,])} </pre>
	<pre> ifelse(CLOSED == T, paste(nrow(ae_rti_hosp_vent[ae_rti_hosp_vent\$group == Arm1,])), " RTI event leading to hospitalization with ventilation in the ", Arm1, " arm", sep = "", "") </pre>	

Classification	Recorded events	Event details
	<pre> ifelse(CLOSED == T, paste(nrow(ae_rti_hosp_vent[ae_rti_hosp_vent\$group == Arm2,]), " RTI event leading to hospitalization with ventilation in the ", Arm2, " arm", sep = ""), "") </pre>	<pre> if(CLOSED == T){unique(ae_rti_hosp_vent\$LLT[ae_rti_hosp_vent\$group == Arm2])} </pre>
Death	<pre> n_ae_rti_dead fatal RTI event in part_ae_rti_dead participant. </pre> <pre> ifelse(CLOSED == T, paste(nrow(ae_rti_dead[ae_rti_dead\$group == Arm1,]), " fatal RTI events in the ", Arm1, " arm", sep = ""), "") ifelse(CLOSED == T, paste(nrow(ae_rti_dead[ae_rti_dead\$group == Arm2,]), " fatal RTI events in the ", Arm2, " arm", sep = ""), "") </pre>	<pre> if(CLOSED == F){unique(ae_rti_dead\$LLT)} if(CLOSED == T){unique(ae_rti_dead\$LLT[ae_rti_dead\$group == Arm1])} if(CLOSED == T){unique(ae_rti_dead\$LLT[ae_rti_dead\$group == Arm2])} </pre>

Injection site reaction events

MedDRA lowest level terms were used to describe events, see for the full definitions Appendix I. Definition of event seriousness followed recorded corresponding health status (see Health status definitions).

Classification	Recorded events	Event details
Total events	<code>n_ae_isr</code> ISR events in <code>part_ae_isr</code> participants.	
Healthy events	<code>n_ae_isr_healthy</code> healthy ISR events in <code>part_ae_isr_healthy</code> participants.	<code>unique(ae_isr_healthy\$LLT)</code>
Mild events	<code>n_ae_isr_mild</code> mild ISR events in <code>part_ae_isr_mild</code> participants.	<code>unique(ae_isr_mild\$LLT)</code>
Moderate events	<code>n_ae_isr_moderate</code> moderate ISR events in <code>part_ae_isr_moderate</code> participants.	<code>unique(ae_isr_moderate\$LLT)</code>
Severe events	<code>n_ae_isr_severe</code> severe ISR events in <code>part_ae_isr_severe</code> participants.	<code>unique(ae_isr_severe\$LLT)</code>
Hospitalization	<code>n_ae_isr_hosp</code> ISR events leading to hospitalization in <code>part_ae_isr_hosp</code> participants.	<code>unique(ae_isr_hosp\$LLT)</code>
Hospitalization with oxygen	<code>n_ae_isr_hosp_oxy</code> ISR events leading to hospitalization with oxygen in <code>part_ae_isr_hosp_oxy</code> participants.	<code>unique(ae_isr_hosp_oxy\$LLT)</code>
Hospitalization with ventilation	<code>n_ae_isr_hosp_vent</code> ISR events leading to hospitalization with ventilation in <code>part_ae_isr_hosp_vent</code> participants.	<code>unique(ae_isr_hosp_vent\$LLT)</code>

Classification	Recorded events	Event details
Death	n_ae_isr_dead fatal ISR events in part_ae_isr_dead participants.	unique(ae_isr_dead\$LLT)

Other events

MedDRA lowest level terms were used to describe events, see for the full definitions Appendix I. Definition of event seriousness followed recorded corresponding health status (see Health status definitions).

Classification	Recorded events	Event details
Total events	n_ae_other	
	other events in	
	part_ae_other	
	participants.	
Healthy events	n_ae_other_healthy	if(CLOSED == F){unique(ae_other_healthy\$LLT)}
	healthy other	
	events in	
	part_ae_other_healthy	
	participants.	
		<pre> ifelse(CLOSED == T, if(CLOSED == T){unique(ae_other_mild\$LLT[ae_other_healthy\$group == paste(nrow(ae_other_healthy[ae_other_healthy\$group == Arm1,])), " healthy other events in the ", Arm1, " arm", sep = ""), ""} ifelse(CLOSED == T, if(CLOSED == T){unique(ae_other_mild\$LLT[ae_other_healthy\$group == paste(nrow(ae_other_healthy[ae_other_healthy\$group == Arm2,])), " healthy other events in the ", Arm2, " arm", sep = ""), ""} </pre>
Mild events	n_ae_other_mild	if(CLOSED == F){unique(ae_other_mild\$LLT)}
	mild other	
	events in	
	part_ae_other_mild	
	participants.	
		<pre> ifelse(CLOSED == T, if(CLOSED == T){unique(ae_other_mild\$LLT[ae_other_mild\$group == paste(nrow(ae_other_mild[ae_other_mild\$group == Arm1,])), " mild other events in the ", Arm1, " arm", sep = ""), ""} </pre>

Classification	Recorded events	Event details
	<pre> ifelse(CLOSED == T, paste(nrow(ae_other_mild[Arm2,])d[ae_other_mild\$group == Arm2,]), " mild other events in the ", Arm2, " arm", sep = ""), "") </pre>	<pre> if(CLOSED == T){unique(ae_other_mild\$LLT[ae_other_mild\$group == Arm2,])d[ae_other_mild\$group == Arm2,]) </pre>
Moderate events	<pre> n_ae_other_moderate moderate other events in part_ae_other_moderate participants. ifelse(CLOSED == T, paste(nrow(ae_other_moderate[Arm1,])d[ae_other_moderate\$group == Arm1,]), " moderate other events in the ", Arm1, " arm", sep = ""), "") ifelse(CLOSED == T, paste(nrow(ae_other_moderate[Arm2,])d[ae_other_moderate\$group == Arm2,]), " moderate other events in the ", Arm2, " arm", sep = ""), "") </pre>	<pre> if(CLOSED == F){unique(ae_other_moderate\$LLT)} if(CLOSED == T){unique(ae_other_moderate\$LLT[ae_other_moderate\$group == Arm1,])d[ae_other_moderate\$group == Arm1,]) if(CLOSED == T){unique(ae_other_moderate\$LLT[ae_other_moderate\$group == Arm2,])d[ae_other_moderate\$group == Arm2,]) </pre>
Severe events	<pre> n_ae_other_severe severe other events in part_ae_other_severe participants. ifelse(CLOSED == T, paste(nrow(ae_other_severe[Arm1,])d[ae_other_severe\$group == Arm1,]), " severe other events in the ", Arm1, " arm", sep = ""), "") </pre>	<pre> if(CLOSED == F){unique(ae_other_severe\$LLT)} if(CLOSED == T){unique(ae_other_severe\$LLT[ae_other_severe\$group == Arm1,])d[ae_other_severe\$group == Arm1,]) </pre>

Classification	Recorded events	Event details
		<pre> ifelse(CLOSED == T, paste(nrow(ae_other_severe[ae_other_severe\$group == Arm2,]), " severe other events in the ", Arm2, " arm", sep = ""), "") </pre>
Hospitalization	<pre> n_ae_other_hosp other events leading to hospitalization in part_ae_other_hosp participants. </pre>	<pre> if(CLOSED == F){unique(ae_other_hosp\$LLT)} ifelse(CLOSED == T, T){unique(ae_other_hosp\$LLT[ae_other_hosp\$group == Arm1,]), " other events leading to hospitalization in the ", Arm1, " arm", sep = ""}, "") ifelse(CLOSED == T, T){unique(ae_other_hosp\$LLT[ae_other_hosp\$group == Arm2,]), " other events leading to hospitalization in the ", Arm2, " arm", sep = ""}, "") </pre>
Hospitalization with oxygen	<pre> n_ae_other_hosp_oxy other event leading to hospitalization with oxygen in part_ae_other_hosp_oxy participant. </pre>	<pre> if(CLOSED == F){unique(ae_other_hosp_oxy\$LLT)} </pre>

Classification	Recorded events	Event details
	<pre> ifelse(CLOSED == T, paste(nrow(ae_other_hosp_oxy[ae_other_hosp_oxy\$group == Arm1,]), " other event leading to hospitalization with oxygen in the ", Arm1, " arm", sep = ""), "") ifelse(CLOSED == T, paste(nrow(ae_other_hosp_oxy[ae_other_hosp_oxy\$group == Arm2,]), " other event leading to hospitalization with oxygen in the ", Arm2, " arm", sep = ""), "") </pre>	<pre> if(CLOSED == T){unique(ae_other_hosp_oxy\$LLT[ae_other_hosp_oxy\$group == Arm1,])} if(CLOSED == T){unique(ae_other_hosp_oxy\$LLT[ae_other_hosp_oxy\$group == Arm2,])} </pre>
Hospitalization with ventilation	<pre> n_ae_other_hosp_vent other events leading to hospitalization with ventilation in part_ae_other_hosp_vent participants. </pre>	<pre> if(CLOSED == F){unique(ae_other_hosp_vent\$LLT)} if(CLOSED == T){unique(ae_other_hosp_vent\$LLT[ae_other_hosp_vent\$group == Arm1,])} other events leading to hospitalization with ventilation in the ", Arm1, " arm", sep = ""), "") </pre>

Classification	Recorded events	Event details
	<pre> ifelse(CLOSED == T, paste(nrow(ae_other_hosp_vent[ae_other_hosp_vent\$group == Arm2,])), " other events leading to hospitalization with ventilation in the ", Arm2, " arm", sep = ""), "") </pre>	<pre> if(CLOSED == T){unique(ae_other_hosp_vent\$LLT[ae_other_hosp_vent\$group == Arm2,])} </pre>
Death	<pre> n_ae_other_dead fatal other events in part_ae_other_dead participants. </pre> <pre> ifelse(CLOSED == T, paste(nrow(ae_other_dead[ae_other_dead\$group == Arm1,])), " fatal other events in the ", Arm1, " arm", sep = ""), "") ifelse(CLOSED == T, paste(nrow(ae_other_dead[ae_other_dead\$group == Arm2,])), " fatal other events in the ", Arm2, " arm", sep = ""), "") </pre>	<pre> if(CLOSED == F){unique(ae_other_dead\$LLT)} if(CLOSED == T){unique(ae_other_dead\$LLT[ae_other_dead\$group == Arm1,])} if(CLOSED == T){unique(ae_other_dead\$LLT[ae_other_dead\$group == Arm2,])} </pre>

Serious adverse events (SAE)

Classification	Recorded events	Causality	Event details
Total SAEs	n_sae SAEs in part_sae participants.	<pre> if(CLOSED == F){ paste(ifelse(n_sae_unrelated > 0, paste(n_sae_unrelated, 'events unrelated to IP. '), ''), ifelse(n_sae_unlikelyrelated > 0, paste(n_sae_unlikelyrelated, 'event unlikely related to IP. '), ''), ifelse(n_sae_possiblyrelated > 0, paste(n_sae_possiblyrelated, 'events possibly related to IP. '), ''), ifelse(n_sae_probablyrelated > 0, paste(n_sae_probablyrelated, 'events probably related to IP. '), ''), ifelse(n_sae_definitely > 0, paste(n_sae_definitely, 'events definitely related to IP. '), ''), sep = '')} </pre>	<pre> if(CLOSED == F){sae\$LLT[order(sae\$LLT)]} </pre>

Classification	Recorded events	Causality	Event details
<pre> ifelse(CLOSED == T, "**SAEs per arm**", "") </pre>	<pre> ifelse(CLOSED == T, paste(nrow(sae[sae\$group == Arm1,]), " SAEs in the ", Arm1, " arm.", sep = ''), "") </pre>	<pre> if(CLOSED == T){ paste(ifelse(n_sae_unrelated_group1 == Arm1))} paste(n_sae_unrelated_group1, 'events unrelated to IP. '), ''), ifelse(n_sae_unlikelyrelated_group1 > 0, paste(n_sae_unlikelyrelated_group1, 'events unlikely related to IP. '), ''), ifelse(n_sae_possiblyrelated_group1 > 0, paste(n_sae_possiblyrelated_group1, 'events possibly related to IP. '), ''), ifelse(n_sae_probablyrelated_group1 > 0, paste(n_sae_probablyrelated_group1, 'events probably related to IP. '), ''), ifelse(n_sae_definitely_group1 > 0, paste(n_sae_definitely_group1, 'events definitely related to IP. '), ''), sep = '')} </pre>	<pre> if(CLOSED == T){sae\$LLT[sae\$group == Arm1]} paste(sae\$LLT[sae\$group == Arm1]) </pre>

Classification	Recorded events	Causality	Event details
	<pre> ifelse(CLOSED == T, paste(nrow(sae[sae\$group == Arm2,]), " SAEs in the ", Arm2, " arm.", sep = ''), "") </pre>	<pre> if(CLOSED == T){ paste(ifelse(n_sae_unrelated_group2 == Arm2,))} paste(n_sae_unrelated_group2, 'events unrelated to IP. '), ''), ifelse(n_sae_unlikelyrelated_group2 > 0, paste(n_sae_unlikelyrelated_group2, 'events unlikely related to IP. '), ''), ifelse(n_sae_possiblyrelated_group2 > 0, paste(n_sae_possiblyrelated_group2, 'events possibly related to IP. '), ''), ifelse(n_sae_probablyrelated_group2 > 0, paste(n_sae_probablyrelated_group2, 'events probably related to IP. '), ''), ifelse(n_sae_definitely_group2 > 0, paste(n_sae_definitely_group2, 'events definitely related to IP. '), ''), sep = '')} </pre>	<pre> if(CLOSED == T){sae\$LLT[sae\$group == Arm2,])} if(CLOSED == T){sae\$LLT[sae\$group == Arm2,])} </pre>

Full details of the adverse events can be found in Appendix B.

Serious adverse events per site

In the modelling approach, all relevant risk factors will be tested. Afterwards, site can be tested as well, but the signal will probably be accounted for by the risk factors already.

Full risk factors per site can be found in Appendix J.

Deaths

```

n_deaths total fatal eventsifelse(CLOSED == F, paste(", with causes of deaths:", sep = ""),
"")`` if(CLOSED == F){unique(ae_dead$LLT)}.

```

```

ifelse(CLOSED == T, paste(nrow(ae_dead[ae_dead$group == Arm1,]), " fatal events in the ",
Arm1, " arm, with cause of deaths: ", sep = ""), "")`` if(CLOSED == T){unique(ae_dead$LLT[ae_dead$group == Arm1])}`` if(CLOSED == T){'.'}

```

```

ifelse(CLOSED == T, ifelse(nrow(ae_dead[ae_dead$group == Arm2,]) <1, paste("No fatal
events in the ", Arm2, " arm", sep = ''), paste(nrow(ae_dead[ae_dead$group == Arm2,]), "
fatal events in the ", Arm2, " arm, with cause of deaths: ", sep = "")), "")`` if(CLOSED
== T){unique(ae_dead$LLT[ae_dead$group == Arm2])}`` if(CLOSED == T){'. '}}

```

Risk factors

In the table below, the demographics and risk factors of the participants that suffered a fatal event (as identified by LLT) are shown. Cells are empty in case of no data available.

```

if(CLOSED == T) {details_deaths %>% select(-PID) %>% t() %>% kable()} else {details_deaths
%>% select(-PID, -group) %>% t() %>% kable()}

```


Protocol deviations

n_deviations protocol deviations associated with part_deviations participants were reported.

```
ifelse(n_deviations_safety == 0, "None of the deviations impacted participant safety.",
paste(n_deviations_integrity, "of the deviations impacted participant safety."))
```

```
ifelse(n_deviations_integrity == 0, "None of the deviations impacted scientific integrity.",
paste(n_deviations_integrity, "of the deviations impacted scientific integrity."))
```

Adherence

A risk in the trial was the lack of adherence, when health care workers might get their own BCG vaccination once it would become clear it was efficacious.

```
ifelse(CLOSED == T, paste(part_vaccine_group1, " (", round(100 * part_vaccine_group1 /
part_group1, 1), "%) participants recorded a flu, BCG, or other (non-COVID-19) vaccination
since the start of the trial in the ", Arm1, " arm. \n", part_vaccine_group2, " (",
round(100 * part_vaccine_group2 / part_group2, 1), "%) participants recorded a flu,
BCG, or other (non-COVID-19) vaccination since the start of the trial in the ", Arm2,
" arm.", sep = ''), paste(part_vaccine, " (", round(100 * part_vaccine / part_enrol, 1),
"%) participants recorded a flu, BCG, or other (non-COVID-19) vaccination since the start
of the trial.", sep = ''))
```

Drop-out

```
ifelse(CLOSED == T, paste(part_discont, " participants were discontinued (withdrawn),
", part_discont_group1, " in the ", Arm1, " arm and ", part_discont_group2, " in the ",
Arm2, " arm.", sep = ""), paste(part_discont, " participants were discontinued (withdrawn).",
sep = "")) ifelse(CLOSED == T, paste("Of those, ", part_discont_losttofu, " participants
were lost to follow-up, ", part_discont_losttofu_group1, " in the ", Arm1, " arm and ",
part_discont_losttofu_group2, " in the ", Arm2, " arm, and ", part_discont - part_discont_losttofu,
" participants lost interest, ", part_discont_group1 - part_discont_losttofu_group1, " in
the ", Arm1, " arm and ", part_discont_group2 - part_discont_losttofu_group2, " in the
", Arm2, " arm.", sep = ""), paste("Of those, ", part_discont_losttofu, " participants
were lost to follow-up, and ", part_discont - part_discont_losttofu, " participants lost
interest.", sep = "")) A drop-out model could be considered when drop-out was substantial and
informative.
```

Quality management data

`n_qc` records were subject of quality review, which took place on `qc_date`.

In `n_qc_yes` of the reviewed data entries ($\text{round}(100 * n_qc_yes / n_qc) \%$), correction was needed.

Quality control data analysis

This script were QC'ed by `reviewer_data_analysis` on `date_last_review_data_analysis`:

The QC was documented in the QC report, no major findings were reported.

Data Definition Table primary endpoint

For the Cox proportional hazard model, the following data structure was used:

Data item	Definition	Possible values	Unit	Data source
PID	Personal identifier	BCGxxxx (x = numerical)	-	enrolment_randomisation.dat; participant_ID.dat
group	Study arm	1 or 2	-	group.dat
time	Time of recorded event	0 (onset) - 52 (end of trial)	Week	event_isr.dat; event_isr_other_fu.dat; event_other.dat; event_rti.dat; event_rti_cont.dat; event_rti_fu.dat
status	Hospitalization of participant ($HS \geq 4$)	0 or 1	-	event_isr.dat; event_isr_other_fu.dat; event_other.dat; event_rti.dat; event_rti_cont.dat; event_rti_fu.dat
age	Age of participant as calculated from date of birth	≥ 18	Year	demographics_contact.dat
gender	Gender of participant	Male, Female	-	demographics_contact.dat
BMI	Body mass index of participant	≥ 15	kg/m ²	vital_signs.dat
ethnicity	Ethnicity of participant	<code>unique(df_full\$ethnicity)</code>	-	demographics_contact.dat
job_category	Job category of participant	<code>unique(df_full\$job_category)</code>	-	pre_assessment.dat
medhis_dm	Medical history of diabetes mellitus	Yes, No	-	medical_history.dat
medhis_hyptens	Medical history of hypertension	Yes, No	-	medical_history.dat
medhis_cvd	Medical history of cardiovascular diseases	Yes, No	-	medical_history.dat
medhis_kd	Medical history of kidney disease	Yes, No	-	medical_history.dat
medhis_asthma	Medical history of asthma	Yes, No	-	medical_history.dat
medhis_copd	Medical history of chronic obstructive pulmonary disease	Yes, No	-	medical_history.dat
medhis_otherlung	Medical history of other lung conditions	Yes, No	-	medical_history.dat
bcg_scar	Scar of previous BCG vaccination	Yes, No	-	meds.dat

Data item	Definition	Possible values	Unit	Data source
pack_years	Smoking habit reported by participant	≥ 0	packs of cigarettes/year	social_history.dat
site	Clinical site	Central, Eden, UCT	-	enrolment_randomisation.dat; participant_ID.dat (based on PID, 1-450 Central, 451-500 Eden, 5000-5500 UCT)
serobase	SARS-CoV-2 serology status at baseline	Positive, negative, equivocal, no result	-	serology.dat
censor	Censoring due to loss of follow-up or withdrawal of consent	0 or 1	-	serology.dat
expect_interact	Expectation of the participant of interaction with COVID-19 positive patients	Yes, No	-	exposure_assessment.dat

R version:

```
version$version.string
```

Packages:

KableExtra Version `packageVersion('kableExtra')`

To cite package 'kableExtra' in publications use:

Zhu H (2021). `_kableExtra: Construct Complex Table with 'kable' and Pipe Syntax_`. R package version 1.3.4,
<<https://CRAN.R-project.org/package=kableExtra>>.

Knitr Version `packageVersion('knitr')`

To cite the 'knitr' package in publications use:

Yihui Xie (2022). `knitr: A General-Purpose Package for Dynamic Report Generation in R`. R package version 1.39.

Yihui Xie (2015) `Dynamic Documents with R and knitr`. 2nd edition. Chapman and Hall/CRC. ISBN 978-1498716963

Yihui Xie (2014) `knitr: A Comprehensive Tool for Reproducible Research in R`. In Victoria Stodden, Friedrich Leisch and Roger D. Peng, editors, `Implementing Reproducible Computational Research`. Chapman and Hall/CRC. ISBN 978-1466561595

Survival Version `packageVersion('survival')`

To cite package 'survival' in publications use:

Therneau T (2022). *_A Package for Survival Analysis in R_*. R package version 3.3-1, <<https://CRAN.R-project.org/package=survival>>.

Terry M. Therneau, Patricia M. Grambsch (2000). *_Modeling Survival Data: Extending the Cox Model_*. Springer, New York. ISBN 0-387-98784-3.

Survminer Version `packageVersion('survminer')`

To cite package 'survminer' in publications use:

Kassambara A, Kosinski M, Biecek P (2021). *_survminer: Drawing Survival Curves using 'ggplot2'_*. R package version 0.4.9, <<https://CRAN.R-project.org/package=survminer>>.

Tidyverse Version `packageVersion('tidyverse')`

To cite package 'tidyverse' in publications use:

Wickham et al., (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43), 1686, <https://doi.org/10.21105/joss.01686>

Viridis Version `packageVersion('viridis')`

To cite `viridis/viridisLite` in publications use:

Simon Garnier, Noam Ross, Robert Rudis, Antônio P. Camargo, Marco Sciaini, and Cédric Scherer (2021). *Rvision - Colorblind-Friendly Color Maps for R*. R package version 0.6.2.

Appendix A: Power calculation

This document provides preliminary sample size calculations for a clinical trial into vaccination against SARS-CoV-2 to prevent COVID-19 in health care workers.

Assumptions

- Attack rate (probability of getting infected) was assumed to be 30-80% which was highly variable
 - 9-12% was reported by the international council of nurses
 - 30% was reported for close contacts
 - 80% was reported for unmitigated epidemics
 - These reports are based on tests on symptomatic patients and the likely number might be 5-10 fold higher
- Healthcare workers use personal protective equipment (PPE) and would be on the lower end of this rate
- Of those infected, 20% would likely be hospitalised (15% severe, 5% critical following reporting in Wuhan)
- A reduction of 30% in these numbers would be considered significant
- Primary endpoint was the proportion of participants hospitalised

Estimate informed by literature references

The power calculation resulted in a sample size of 220 participants per arm assuming an attack rate in health care workers of 30%, hospitalisation rate of 20% and a reduction by vaccination of 75%. These assumptions were based on Liu et al (2020), the case severity reporting from Wuhan, and the reduction in respiratory infections by BCG previously reported by Wardhana et al (2011) and Nemes et al (2018), respectively.

Sample size

Using the assumptions stated above, the **sample size per study arm** ranged between `ceiling(power.prop.test(p1 = 0.8*0.2, p2 = 0.8*0.2*0.70, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')$n)` for 80% attack or infection rate to `ceiling(power.prop.test(p1 = 0.3*0.2, p2 = 0.3*0.2*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')$n)` for 30% attack or infection rate to even `ceiling(power.prop.test(p1 = 0.1*0.2, p2 = 0.1*0.2*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')$n)` for 10% attack or infection rate. All scenarios are shown in the tables below, with varying attack or infection rate, hospitalisation rate, and reduction by the vaccine.

Power calculations

This clinical trial had a proportional endpoint and power calculations to determine sample size were performed for different scenarios based on the stated assumptions. The group size increased with a smaller vaccine effect, which would result in a smaller difference between the test and control group. It was therefore important to have a realistic estimate of the difference. The group size also increased with a decreasing attack or infection rate of health care professionals (the more rare an event, the larger the group to observe it).

For a two-sample test (1 test, 1 control) for proportions, a power calculation based on the Pearson chi-square test with continuity correction (for smaller sample sizes) was performed. Here, a sample size calculation for two sample populations with a proportion of 80% positive (control) and 50% reduction (test) with $\alpha = 0.05$ and $\beta = 0.2$ (i.e. 80% power) and a one-sided test was performed.

These scenarios did not take into account a drop-out rate, which should be considered for the final trial design.

Table of sample size for 10% hospitalisation

Attack rate	Hospitalisation rate	Reduction by vaccination	Sample size per arm
80%	10%	75%	<code>ceiling(power.prop.test(p1 = 0.8*0.1, p2 = 0.8*0.1*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
80%	10%	50%	<code>ceiling(power.prop.test(p1 = 0.8*0.1, p2 = 0.8*0.1*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
80%	10%	30%	<code>ceiling(power.prop.test(p1 = 0.8*0.1, p2 = 0.8*0.1*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
80%	10%	10%	<code>ceiling(power.prop.test(p1 = 0.8*0.1, p2 = 0.8*0.1*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
50%	10%	75%	<code>ceiling(power.prop.test(p1 = 0.5*0.1, p2 = 0.5*0.1*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
50%	10%	50%	<code>ceiling(power.prop.test(p1 = 0.5*0.1, p2 = 0.5*0.1*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
50%	10%	30%	<code>ceiling(power.prop.test(p1 = 0.5*0.1, p2 = 0.5*0.1*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>

Attack rate	Hospitalisation rate	Reduction by vaccination	Sample size per arm
50%	10%	10%	<code>ceiling(power.prop.test(p1 = 0.5*0.1, p2 = 0.5*0.1*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	10%	75%	<code>ceiling(power.prop.test(p1 = 0.3*0.1, p2 = 0.3*0.1*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	10%	55%	<code>ceiling(power.prop.test(p1 = 0.3*0.1, p2 = 0.3*0.1*0.45, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	10%	50%	<code>ceiling(power.prop.test(p1 = 0.3*0.1, p2 = 0.3*0.1*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	10%	45%	<code>ceiling(power.prop.test(p1 = 0.3*0.1, p2 = 0.3*0.1*0.55, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	10%	33%	<code>ceiling(power.prop.test(p1 = 0.3*0.1, p2 = 0.3*0.1*0.67, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	10%	30%	<code>ceiling(power.prop.test(p1 = 0.3*0.1, p2 = 0.3*0.1*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	10%	10%	<code>ceiling(power.prop.test(p1 = 0.3*0.1, p2 = 0.3*0.1*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>

Attack rate	Hospitalisation rate	Reduction by vaccination	Sample size per arm
20%	10%	75%	<code>ceiling(power.prop.test(p1 = 0.2*0.1, p2 = 0.2*0.1*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
20%	10%	50%	<code>ceiling(power.prop.test(p1 = 0.2*0.1, p2 = 0.2*0.1*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
20%	10%	30%	<code>ceiling(power.prop.test(p1 = 0.2*0.1, p2 = 0.2*0.1*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
20%	10%	10%	<code>ceiling(power.prop.test(p1 = 0.2*0.1, p2 = 0.2*0.1*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
10%	10%	75%	<code>ceiling(power.prop.test(p1 = 0.1*0.1, p2 = 0.1*0.1*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
10%	10%	50%	<code>ceiling(power.prop.test(p1 = 0.1*0.1, p2 = 0.1*0.1*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
10%	10%	30%	<code>ceiling(power.prop.test(p1 = 0.1*0.1, p2 = 0.1*0.1*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
10%	10%	10%	<code>ceiling(power.prop.test(p1 = 0.1*0.1, p2 = 0.1*0.1*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>

Table of sample size for 20% hospitalisation

Attack rate	Hospitalisation rate	Reduction by vaccination	Sample size per arm
80%	20%	75%	<code>ceiling(power.prop.test(p1 = 0.8*0.2, p2 = 0.8*0.2*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
80%	20%	50%	<code>ceiling(power.prop.test(p1 = 0.8*0.2, p2 = 0.8*0.2*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
80%	20%	30%	<code>ceiling(power.prop.test(p1 = 0.8*0.2, p2 = 0.8*0.2*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
80%	20%	10%	<code>ceiling(power.prop.test(p1 = 0.8*0.2, p2 = 0.8*0.2*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
50%	20%	75%	<code>ceiling(power.prop.test(p1 = 0.5*0.2, p2 = 0.5*0.2*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
50%	20%	50%	<code>ceiling(power.prop.test(p1 = 0.5*0.2, p2 = 0.5*0.2*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
50%	20%	30%	<code>ceiling(power.prop.test(p1 = 0.5*0.2, p2 = 0.5*0.2*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
50%	20%	10%	<code>ceiling(power.prop.test(p1 = 0.5*0.2, p2 = 0.5*0.2*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>

Attack rate	Hospitalisation rate	Reduction by vaccination	Sample size per arm
30%	20%	75%	<code>ceiling(power.prop.test(p1 = 0.3*0.2, p2 = 0.3*0.2*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	20%	55%	<code>ceiling(power.prop.test(p1 = 0.3*0.2, p2 = 0.3*0.2*0.45, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	20%	50%	<code>ceiling(power.prop.test(p1 = 0.3*0.2, p2 = 0.3*0.2*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	20%	40%	<code>ceiling(power.prop.test(p1 = 0.3*0.2, p2 = 0.3*0.2*0.6, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	20%	30%	<code>ceiling(power.prop.test(p1 = 0.3*0.2, p2 = 0.3*0.2*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	20%	10%	<code>ceiling(power.prop.test(p1 = 0.3*0.2, p2 = 0.3*0.2*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
20%	20%	75%	<code>ceiling(power.prop.test(p1 = 0.2*0.2, p2 = 0.2*0.2*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
20%	20%	50%	<code>ceiling(power.prop.test(p1 = 0.2*0.2, p2 = 0.2*0.2*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>

Attack rate	Hospitalisation rate	Reduction by vaccination	Sample size per arm
20%	20%	30%	<code>ceiling(power.prop.test(p1 = 0.2*0.2, p2 = 0.2*0.2*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
20%	20%	10%	<code>ceiling(power.prop.test(p1 = 0.2*0.2, p2 = 0.2*0.2*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
10%	20%	75%	<code>ceiling(power.prop.test(p1 = 0.1*0.2, p2 = 0.1*0.2*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
10%	20%	50%	<code>ceiling(power.prop.test(p1 = 0.1*0.2, p2 = 0.1*0.2*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
10%	20%	30%	<code>ceiling(power.prop.test(p1 = 0.1*0.2, p2 = 0.1*0.2*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
10%	20%	10%	<code>ceiling(power.prop.test(p1 = 0.1*0.2, p2 = 0.1*0.2*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>

Table of sample size for 30% hospitalisation

Attack rate	Hospitalisation rate	Reduction by vaccination	Sample size per arm
80%	30%	75%	<code>ceiling(power.prop.test(p1 = 0.8*0.3, p2 = 0.8*0.3*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>

Attack rate	Hospitalisation rate	Reduction by vaccination	Sample size per arm
80%	30%	50%	<code>ceiling(power.prop.test(p1 = 0.8*0.3, p2 = 0.8*0.3*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
80%	30%	30%	<code>ceiling(power.prop.test(p1 = 0.8*0.3, p2 = 0.8*0.3*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
80%	30%	10%	<code>ceiling(power.prop.test(p1 = 0.8*0.3, p2 = 0.8*0.3*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
50%	30%	75%	<code>ceiling(power.prop.test(p1 = 0.5*0.3, p2 = 0.5*0.3*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
50%	30%	50%	<code>ceiling(power.prop.test(p1 = 0.5*0.3, p2 = 0.5*0.3*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
50%	30%	30%	<code>ceiling(power.prop.test(p1 = 0.5*0.3, p2 = 0.5*0.3*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
50%	30%	10%	<code>ceiling(power.prop.test(p1 = 0.5*0.3, p2 = 0.5*0.3*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	30%	75%	<code>ceiling(power.prop.test(p1 = 0.3*0.3, p2 = 0.3*0.3*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>

Attack rate	Hospitalisation rate	Reduction by vaccination	Sample size per arm
30%	30%	50%	<code>ceiling(power.prop.test(p1 = 0.3*0.3, p2 = 0.3*0.3*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	30%	45%	<code>ceiling(power.prop.test(p1 = 0.3*0.3, p2 = 0.3*0.3*0.55, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	30%	33%	<code>ceiling(power.prop.test(p1 = 0.3*0.3, p2 = 0.3*0.3*0.67, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	30%	30%	<code>ceiling(power.prop.test(p1 = 0.3*0.3, p2 = 0.3*0.3*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
30%	30%	10%	<code>ceiling(power.prop.test(p1 = 0.3*0.3, p2 = 0.3*0.3*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
20%	30%	75%	<code>ceiling(power.prop.test(p1 = 0.2*0.3, p2 = 0.2*0.3*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
20%	30%	50%	<code>ceiling(power.prop.test(p1 = 0.2*0.3, p2 = 0.2*0.3*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
20%	30%	30%	<code>ceiling(power.prop.test(p1 = 0.2*0.3, p2 = 0.2*0.3*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>

Attack rate	Hospitalisation rate	Reduction by vaccination	Sample size per arm
20%	30%	10%	<code>ceiling(power.prop.test(p1 = 0.2*0.3, p2 = 0.2*0.3*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
10%	30%	75%	<code>ceiling(power.prop.test(p1 = 0.1*0.3, p2 = 0.1*0.3*0.25, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
10%	30%	50%	<code>ceiling(power.prop.test(p1 = 0.1*0.3, p2 = 0.1*0.3*0.5, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
10%	30%	30%	<code>ceiling(power.prop.test(p1 = 0.1*0.3, p2 = 0.1*0.3*0.7, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>
10%	30%	10%	<code>ceiling(power.prop.test(p1 = 0.1*0.3, p2 = 0.1*0.3*0.9, sig.level = 0.05, power = 0.8, n = NULL, alternative = 'one.sided')\$n)</code>

Appendix B: Additional summary tables

Missing variables

Numerical demographics

Binary demographics

Categorical demographics

Work related demographics, including expected exposure to COVID-19

Respiratory tract infection events (full details)

Injection site reaction events (full details)

Other events (full details)

Serious adverse events (full details)

Appendix C: Additional figures

Demographics as bar graphs (number of participants)

The following figures display the baseline demographics of the participants `ifelse(CLOSED == T, "stratified by study arm", "")`.

Age (bar graph)

Baseline body weight (bar graph)

Height (bar graph)

BMI (bar graph)

Dotted lines delimit categories underweight (<18.5), normal (18.5-24.9), overweight (25-29.9), obese (30-39.9), extremely obese (>40).

Current smoking (bar graph)

Appendix D: Additional data listings

The following variables are recorded in the different databases for each participant.

Screening and enrolment

`names(df1_full)`

Events

`names(df2_full)`

Laboratory results

`names(df3_full)`

Follow-up

`names(df4_full)`

Appendix E: Hospitalization by risk factor/classifier

The following Kaplan-Meier figures show Kaplan-Meier results for first hospitalization event stratified by the identified risk factors/classifiers.

Risk factor: age

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor age. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor age, versus time. The y-axis is enhanced for clearer visualization.

Risk factor: gender

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor gender. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor gender, versus time. The y-axis is enhanced for clearer visualization.

Risk factor: Body mass index (BMI)

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor BMI. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor BMI, versus time. The y-axis is enhanced for clearer visualization.

Classifier: ethnicity

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` classifier ethnicity. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` classifier ethnicity, versus time. The y-axis is enhanced for clearer visualization.

Risk factor: job category

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor job category. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor job category, versus time. The y-axis is enhanced for clearer visualization.

Risk factor: medical history of diabetes mellitus

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor diabetes mellitus. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor diabetes mellitus, versus time. The y-axis is enhanced for clearer visualization.

Risk factor: medical history of hypertension

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor hypertension. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor hypertension, versus time. The y-axis is enhanced for clearer visualization.

Risk factor: medical history of cardiovascular disease

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor cardiovascular disease. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor cardiovascular disease, versus time. The y-axis is enhanced for clearer visualization.

Risk factor: medical history of asthma

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor asthma. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor asthma, versus time. The y-axis is enhanced for clearer visualization.

Risk factor: medical history of COPD

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor COPD. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor COPD, versus time. The y-axis is enhanced for clearer visualization.

Risk factor: medical history of other lung disease

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor other lung disease. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor other lung disease, versus time. The y-axis is enhanced for clearer visualization.

Risk factor: medical history of kidney disease

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor kidney disease. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor kidney disease, versus time. The y-axis is enhanced for clearer visualization.

Risk factor: presence of BCG scar

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor BCG scar. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor BCG scar, versus time. The y-axis is enhanced for clearer visualization.

Risk factor: currently smoking

Kaplan-Meier estimates of the proportion of subject which were not hospitalized versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor currently smoking. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of being hospitalized, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor currently smoking, versus time. The y-axis is enhanced for clearer visualization.

Currently smoking is grouped in 0, 1-20, 21-40, and >40 pack years.

Appendix F: KM plots for respiratory tract infections

Kaplan-Meier plots for all health status scores for the first RTI event.

Health status 1: mild symptoms (RTI events)

Kaplan-Meier estimates of the proportion of subject which did not have an RTI event (HS=1) versus time. Only first event is shown. `ifelse(CLOSED == T, " The figure is stratified by study arm.", "")` The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of an RTI event `ifelse(CLOSED == T, ", stratified by study arm", "")`, versus time. The y-axis is enhanced for clearer visualization.

Health status 2: moderate symptoms (RTI events)

Kaplan-Meier estimates of the proportion of subject which did not have an RTI event (HS=2) versus time. Only first event is shown. `ifelse(CLOSED == T, " The figure is stratified by study arm.", "")` The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of an RTI event `ifelse(CLOSED == T, ", stratified by study arm", "")`, versus time. The y-axis is enhanced for clearer visualization.

Health status 3: severe symptoms (RTI events)

Kaplan-Meier estimates of the proportion of subject which did not have an RTI event (HS=3) versus time. Only first event is shown. `ifelse(CLOSED == T, " The figure is stratified by study arm.", "")` The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of an RTI event `ifelse(CLOSED == T, ", stratified by study arm", "")`, versus time. The y-axis is enhanced for clearer visualization.

Health status 4: hospitalized (RTI events)

Kaplan-Meier estimates of the proportion of subject which did not have an RTI event (HS=4) versus time. Only first event is shown. `ifelse(CLOSED == T, " The figure is stratified by study arm.", "")` The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of an RTI event `ifelse(CLOSED == T, ", stratified by study arm", "")`, versus time. The y-axis is enhanced for clearer visualization.

Health status 5: hospitalized, oxygen (RTI events)

Kaplan-Meier estimates of the proportion of subject which did not have an RTI event (HS=5) versus time. Only first event is shown. `ifelse(CLOSED == T, " The figure is stratified by study arm.", "")` The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up,

death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of an RTI event `ifelse(CLOSED == T, ", stratified by study arm", "")`, versus time. The y-axis is enhanced for clearer visualization.

Health status 6: hospitalized, ventilated (RTI events)

Kaplan-Meier estimates of the proportion of subject which did not have an RTI event (HS=6) versus time. Only first event is shown. `ifelse(CLOSED == T, " The figure is stratified by study arm.", "")` The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of an RTI event `ifelse(CLOSED == T, ", stratified by study arm", "")`, versus time. The y-axis is enhanced for clearer visualization.

Health status 7: death (RTI events)

Kaplan-Meier estimates of the proportion of subject which did not have an RTI event (HS=7) versus time. `ifelse(CLOSED == T, " The figure is stratified by study arm.", "")` The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk of an RTI event `ifelse(CLOSED == T, ", stratified by study arm", "")`, versus time. The y-axis is enhanced for clearer visualization.

Appendix G: KM plots for RTI per risk factor/classifier

The following Kaplan-Meier figures show Kaplan-Meier results for RTI events stratified by the identified risk factors/classifiers.

Risk factor: age

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor age. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor age, versus time.

Risk factor: gender

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor gender. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor gender, versus time.

Risk factor: Body mass index (BMI)

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor BMI. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor BMI, versus time.

Classifier: ethnicity

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` classifier ethnicity. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` classifier ethnicity, versus time.

Risk factor: job category

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor job category. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor job category, versus time.

Risk factor: medical history of diabetes mellitus

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor diabetes mellitus. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor diabetes mellitus, versus time.

Risk factor: medical history of hypertension

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor hypertension. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor hypertension, versus time.

Risk factor: medical history of cardiovascular disease

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor cardiovascular disease. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor cardiovascular disease, versus time.

Risk factor: medical history of asthma

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor asthma. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor asthma, versus time.

Risk factor: medical history of COPD

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor COPD. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor COPD, versus time.

Risk factor: medical history of other lung disease

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor other lung disease. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor other lung disease, versus time.

Risk factor: medical history of kidney disease

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor kidney disease. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor kidney disease, versus time.

Risk factor: presence of BCG scar

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor BCG scar. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor BCG scar, versus time.

Risk factor: currently smoking

Kaplan-Meier estimates of the proportion of subject without RTI events versus time. The figure is stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor currently smoking. The shaded area represent the 95% confidence interval of the Kaplan-Meier estimates. The figure is based on the intention-to-treat population, vertical lines show censoring events (withdrawal or lost to follow-up, death, or end of trial, whichever comes first). The table below the figure shows the number of subjects at risk, stratified by `ifelse(CLOSED == T, " study arm and", "")` risk factor currently smoking, versus time.

Currently smoking is grouped in 0, 1-20, 21-40, and >40 pack years.

Appendix H: Health status score over time

Health status score over time plots stratified by the identified risk factors/classifiers.

Risk factor: age

Health status over time for each subject in the study (dashed lines). The figure is stratified by the risk factor `age` if `else(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each risk factor strata if `else(CLOSED == T, " and study arm", "")` over time.

Risk factor: gender

Health status over time for each subject in the study (dashed lines). The figure is stratified by the risk factor `gender` if `else(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each risk factor strata if `else(CLOSED == T, " and study arm", "")` over time.

Risk factor: body mass index (BMI)

Health status over time for each subject in the study (dashed lines). The figure is stratified by the risk factor BMI if `else(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each risk factor strata if `else(CLOSED == T, " and study arm", "")` over time.

Classifier: ethnicity

Health status over time for each subject in the study (dashed lines). The figure is stratified by the classifier ethnicity if `else(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each classifier strata if `else(CLOSED == T, " and study arm", "")` over time.

Risk factor: job category

Health status over time for each subject in the study (dashed lines). The figure is stratified by the risk factor job category `ifelse(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each risk factor strata `ifelse(CLOSED == T, " and study arm", "")` over time.

Risk factor: medical history of diabetes mellitus

Health status over time for each subject in the study (dashed lines). The figure is stratified by the risk factor diabetes mellitus `ifelse(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each risk factor strata `ifelse(CLOSED == T, " and study arm", "")` over time.

Risk factor: medical history of hypertension

Health status over time for each subject in the study (dashed lines). The figure is stratified by the risk factor hypertension if `else(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each risk factor strata if `else(CLOSED == T, " and study arm", "")` over time.

Risk factor: medical history of cardiovascular disease

Health status over time for each subject in the study (dashed lines). The figure is stratified by the risk factor cardiovascular disease if `else(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each risk factor strata if `else(CLOSED == T, " and study arm", "")` over time.

Risk factor: medical history of asthma

Health status over time for each subject in the study (dashed lines). The figure is stratified by the risk factor `asthmaifelse(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each risk factor `strataifelse(CLOSED == T, " and study arm", "")` over time.

Risk factor: medical history of COPD

Health status over time for each subject in the study (dashed lines). The figure is stratified by the risk factor `COPDifelse(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each risk factor `strataifelse(CLOSED == T, " and study arm", "")` over time.

Risk factor: medical history of other lung disease

Health status over time for each subject in the study (dashed lines). The figure is stratified by the risk factor other lung disease `ifelse(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each risk factor strata `ifelse(CLOSED == T, " and study arm", "")` over time.

Risk factor: medical history of kidney disease

Health status over time for each subject in the study (dashed lines). The figure is stratified by the risk factor kidney disease `ifelse(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each risk factor strata `ifelse(CLOSED == T, " and study arm", "")` over time.

Risk factor: presence of BCG scar

Health status over time for each subject in the study (dashed lines). The figure is stratified by the risk factor BCG scar if `else(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each risk factor strata if `else(CLOSED == T, " and study arm", "")` over time.

Risk factor: currently smoking

Health status over time for each subject in the study (dashed lines). The figure is stratified by the risk factor currently smoking if `else(CLOSED == T, " and coloured by the study arm", "")`. The solid lines represent a loess smooth of the score in each risk factor strata if `else(CLOSED == T, " and study arm", "")` over time.

Individual plots of health status scores over time

The figures below show the health status score over time per individual subject for RTI events, of which the event number is shown in bar above the plot.

Appendix I: MedDRA

Medical Dictionary for Regulatory Activities (MedDRA) terms are used in this report.

LLT = Lowest level terms, PT = Preferred terms, HLT = High-level terms, HLGT = High-level group terms, SOC = System organ class

Appendix J: Risk factors per site

Baseline serology for SARS-CoV-2 per site

At baseline participants were tested for SARS-CoV-2 infection based on immunoglobulin G (IgG) serology.

Central

```
ifelse(CLOSED == T, paste(c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'Central', group == Arm1) %>% distinct(PID) %>% nrow(), " (", round(100 * (c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'Central', group == Arm1) %>% distinct(PID) %>% nrow()) / part_enrol_central_group1, 1), "%) participants had a baseline positive serology result for SARS-CoV-2 based on IgG in the ", Arm1, " arm. \n", c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'Central', group == Arm2) %>% distinct(PID) %>% nrow(), " (", round(100 * (c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'Central', group == Arm2) %>% distinct(PID) %>% nrow()) / part_enrol_central_group2, 1), "%) participants had a baseline positive serology result for SARS-CoV-2 based on IgG in the ", Arm2, " arm.", sep = ''), paste(c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'Central') %>% distinct(PID) %>% nrow(), " (", round(100 * (c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'Central') %>% distinct(PID) %>% nrow()) / part_enrol_central, 1), "%) participants had a baseline positive serology result for SARS-CoV-2 based on IgG.", sep = ''))
```

Eden

```
ifelse(CLOSED == T, paste(c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'Eden', group == Arm1) %>% distinct(PID) %>% nrow(), " (", round(100 * (c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'Eden', group == Arm1) %>% distinct(PID) %>% nrow()) / part_enrol_edden_group1, 1), "%) participants had a baseline positive serology result for SARS-CoV-2 based on IgG in the ", Arm1, " arm. \n", c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'Eden', group == Arm2) %>% distinct(PID) %>% nrow(), " (", round(100 * (c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'Eden', group == Arm2) %>% distinct(PID) %>% nrow()) / part_enrol_edden_group2, 1), "%) participants had a baseline positive serology result for SARS-CoV-2 based on IgG in the ", Arm2, " arm.", sep = ''), paste(c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'Eden') %>% distinct(PID) %>% nrow(), " (", round(100 * (c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'Eden') %>% distinct(PID) %>% nrow()) / part_enrol_edden, 1), "%) participants had a baseline positive serology result for SARS-CoV-2 based on IgG.", sep = ''))
```

UCT

```
ifelse(CLOSED == T, paste(c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'UCT', group == Arm1) %>% distinct(PID) %>% nrow(), " (", round(100 * (c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'UCT', group == Arm1) %>% distinct(PID) %>% nrow()) / part_enrol_uct_group1, 1), "%) participants had a baseline positive serology result for SARS-CoV-2 based on IgG in the ", Arm1, " arm. \n", c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'UCT', group == Arm2) %>% distinct(PID) %>% nrow(), " (", round(100 * (c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'UCT', group == Arm2) %>% distinct(PID) %>% nrow()) / part_enrol_uct_group2, 1), "%) participants had a baseline positive serology result for SARS-CoV-2 based on IgG in the ", Arm2, " arm.",
```

```
sep = ''), paste(c19_serology %>% filter(visit_week_id_3 == 0) %>% filter(site == 'UCT')
%>% distinct(PID) %>% nrow(), " (", round(100 * (c19_serology %>% filter(visit_week_id_3
== 0) %>% filter(site == 'UCT') %>% distinct(PID) %>% nrow()) / part_enrol_uct, 1), "%)
participants had a baseline positive serology result for SARS-CoV-2 based on IgG.", sep =
''))
```

Baseline tuberculosis per site

At baseline, participants are tested for TB infection based on an interferon-gamma release assay (IGRA).

Central

```
ifelse(CLOSED == T, paste(df_full %>% filter(igra == 'POS') %>% filter(site == 'Central',
group == Arm1) %>% distinct(PID) %>% nrow(), " (", round(100 * (df_full %>% filter(igra
== 'POS') %>% filter(site == 'Central', group == Arm1) %>% distinct(PID) %>% nrow())
/ part_enrol_central_group1, 1), "%) participants had a baseline positive result for
IGRA result for TB in the ", Arm1, " arm. \n", df_full %>% filter(igra == 'POS') %>%
filter(site == 'Central', group == Arm2) %>% distinct(PID) %>% nrow(), " (", round(100
* (df_full %>% filter(igra == 'POS') %>% filter(site == 'Central', group == Arm2) %>%
distinct(PID) %>% nrow()) / part_enrol_central_group2, 1), "%) participants had a baseline
positive result for IGRA result for TB in the ", Arm2, " arm.", sep = ''), paste(df_full
%>% filter(igra == 'POS') %>% filter(site == 'Central') %>% distinct(PID) %>% nrow(),
" (", round(100 * (df_full %>% filter(igra == 'POS') %>% filter(site == 'Central') %>%
distinct(PID) %>% nrow()) / part_enrol_central, 1), "%) participants had a baseline
positive result for IGRA result for TB.", sep = ''))
```

Eden

```
ifelse(CLOSED == T, paste(df_full %>% filter(igra == 'POS') %>% filter(site == 'Eden',
group == Arm1) %>% distinct(PID) %>% nrow(), " (", round(100 * (df_full %>% filter(igra
== 'POS') %>% filter(site == 'Eden', group == Arm1) %>% distinct(PID) %>% nrow()) /
part_enrol_edan_group1, 1), "%) participants had a baseline positive result for IGRA
result for TB in the ", Arm1, " arm. \n", df_full %>% filter(igra == 'POS') %>% filter(site
== 'Eden', group == Arm2) %>% distinct(PID) %>% nrow(), " (", round(100 * (df_full %>%
filter(igra == 'POS') %>% filter(site == 'Eden', group == Arm2) %>% distinct(PID) %>%
nrow()) / part_enrol_edan_group2, 1), "%) participants had a baseline positive result
for IGRA result for TB in the ", Arm2, " arm.", sep = ''), paste(df_full %>% filter(igra
== 'POS') %>% filter(site == 'Eden') %>% distinct(PID) %>% nrow(), " (", round(100 *
(df_full %>% filter(igra == 'POS') %>% filter(site == 'Eden') %>% distinct(PID) %>%
nrow()) / part_enrol_edan, 1), "%) participants had a baseline positive result for IGRA
result for TB.", sep = ''))
```

UCT

```
ifelse(CLOSED == T, paste(df_full %>% filter(igra == 'POS') %>% filter(site == 'UCT',
group == Arm1) %>% distinct(PID) %>% nrow(), " (", round(100 * (df_full %>% filter(igra
== 'POS') %>% filter(site == 'UCT', group == Arm1) %>% distinct(PID) %>% nrow()) / part_enrol_uct_group
1), "%) participants had a baseline positive result for IGRA result for TB in the ",
Arm1, " arm. \n", df_full %>% filter(igra == 'POS') %>% filter(site == 'UCT', group ==
Arm2) %>% distinct(PID) %>% nrow(), " (", round(100 * (df_full %>% filter(igra == 'POS')
%>% filter(site == 'UCT', group == Arm2) %>% distinct(PID) %>% nrow()) / part_enrol_uct_group2,
```

```

1), "%) participants had a baseline positive result for IGRA result for TB in the ",
Arm2, " arm.", sep = ''), paste(df_full %>% filter(igra == 'POS') %>% filter(site ==
'UCT') %>% distinct(PID) %>% nrow(), " (", round(100 * (df_full %>% filter(igra == 'POS')
%>% filter(site == 'UCT') %>% distinct(PID) %>% nrow()) / part_enrol_uct, 1), "%) participants
had a baseline positive result for IGRA result for TB.", sep = ''))

```

Numerical demographics

Central

Missing values are reported as NA in the `ifelse(CLOSED == T, 'tables', 'table')` below.

```
ifelse(CLOSED == T, paste("***the ", Arm1, " arm**", sep = ''), "")
ifelse(CLOSED == T, paste("***the ", Arm2, " arm**", sep = ''), "")
```

Eden

Missing values are reported as NA in the `ifelse(CLOSED == T, 'tables', 'table')` below.

```
ifelse(CLOSED == T, paste("***the ", Arm1, " arm**", sep = ''), "")
ifelse(CLOSED == T, paste("***the ", Arm2, " arm**", sep = ''), "")
```

UCT

Missing values are reported as NA in the `ifelse(CLOSED == T, 'tables', 'table')` below.

```
ifelse(CLOSED == T, paste("***the ", Arm1, " arm**", sep = ''), "")
ifelse(CLOSED == T, paste("***the ", Arm2, " arm**", sep = ''), "")
```

Binary demographics

Empty records are assigned either '' (blank) or NA.

Central

```
ifelse(CLOSED == T, paste("***the ", Arm1, " arm**", sep = ''), "")  
ifelse(CLOSED == T, paste("***the ", Arm2, " arm**", sep = ''), "")
```

Eden

```
ifelse(CLOSED == T, paste("***the ", Arm1, " arm**", sep = ''), "")  
ifelse(CLOSED == T, paste("***the ", Arm2, " arm**", sep = ''), "")
```

UCT

```
ifelse(CLOSED == T, paste("***the ", Arm1, " arm**", sep = ''), "")  
ifelse(CLOSED == T, paste("***the ", Arm2, " arm**", sep = ''), "")
```

Categorical demographics

Central

```
ifelse(CLOSED == T, paste("***the ", Arm1, " arm**", sep = ''), "")  
ifelse(CLOSED == T, paste("***the ", Arm2, " arm**", sep = ''), "")
```

Eden

```
ifelse(CLOSED == T, paste("***the ", Arm1, " arm**", sep = ''), "")  
ifelse(CLOSED == T, paste("***the ", Arm2, " arm**", sep = ''), "")
```

UCT

```
ifelse(CLOSED == T, paste("***the ", Arm1, " arm**", sep = ''), "")  
ifelse(CLOSED == T, paste("***the ", Arm2, " arm**", sep = ''), "")
```

Work related demographics, including expected exposure to COVID-19

Central

```
ifelse(CLOSED == T, paste("***the ", Arm1, " arm**", sep = ''), "")
```

```
ifelse(CLOSED == T, paste("***the ", Arm2, " arm**", sep = ''), "")
```

Legend work_hours: 1_40 means 1-40 hour/week, 41_80 means 41-80 hour/week, more_than_80 means more than 80 hour/week.

Eden

```
ifelse(CLOSED == T, paste("***the ", Arm1, " arm**", sep = ''), "")
```

```
ifelse(CLOSED == T, paste("***the ", Arm2, " arm**", sep = ''), "")
```

Legend work_hours: 1_40 means 1-40 hour/week, 41_80 means 41-80 hour/week, more_than_80 means more than 80 hour/week.

UCT

```
ifelse(CLOSED == T, paste("***the ", Arm1, " arm**", sep = ''), "")
```

```
ifelse(CLOSED == T, paste("***the ", Arm2, " arm**", sep = ''), "")
```

Legend work_hours: 1_40 means 1-40 hour/week, 41_80 means 41-80 hour/week, more_than_80 means more than 80 hour/week.

Additional figures per site

Demographics as bar graphs (number of participants)

The following figures display the baseline demographics of the participants `ifelse(CLOSED == T, "stratified by study arm", "")`.

Age (bar graph)

Baseline body weight (bar graph)

Height (bar graph)

BMI (bar graph)

Dotted lines delimit categories underweight (<18.5), normal (18.5-24.9), overweight (25-29.9), obese (30-39.9), extremely obese (>40).

Current smoking (bar graph)

This report took `(proc.time() - ptm) [3]` seconds to compile.